Administering Entrust/PKI 6.0 on Windows

Entrust® Technologies

We Bring Trust to e-Business™

© 2001 Entrust Technologies. All rights reserved.

Entrust is a registered trademark of Entrust Technologies Limited. All Entrust product names are trademarks of Entrust Technologies Limited. All other company and product names may be trademarks or registered trademarks of their respective owners.

This information is subject to change as Entrust Technologies Limited reserves the right to, without notice, make changes to its products as progress in engineering or manufacturing methods or circumstances may warrant.

Export and/or import of cryptographic products may be restricted by various regulations in various countries. Licenses may be required.

Printed in Canada.

Release 6.0.

Contents

About this book **15**
Typographic conventions 16

CHAPTER 1
Working with audit logs and creating reports **17**
About audit logs 18
Secured audit logs 18
Error messages in manager.log and mgraudit.log 19
Modifying the default behavior of log files 19
Viewing audit logs in Entrust/RA 19
Viewing an audit log in Entrust/RA 22
Description of audit log fields in Entrust/RA 23
Sorting audit logs in Entrust/RA 24
Clearing audit logs in Entrust/RA 25
Saving audit logs in Entrust/RA 25
Viewing audit logs in the Windows Event Viewer 27
Description of audit log fields in the Event Viewer 29
Producing reports in Entrust/RA 30
Explanation of report fields 30

CHAPTER 2

Other administrative tasks 33

Using the Directory Browser 34
Logging in to the Directory Browser 34
Adding a Directory entry 35
Finding a Directory entry 37
Deleting a Directory entry 38
About Directory attributes 39
Adding an attribute to a Directory entry 40
Adding a Directory attribute value 41
Replacing a Directory attribute value 41
Deleting a Directory attribute or attribute value 42
Showing and hiding the Entrust/RA toolbar 44
Setting up Security Officers for key recovery 45
Changing passwords and password rules 48
Changing a Master User password 48
Changing Master User password rules 50
Changing an administrative user profile password 52
Changing the CA Directory Access password 53
Changing the Entrust Directory Administrator password 54
Changing password length 55
Changing the number of authorizations for sensitive operations 57
Changing the number of Master User authorizations 57
Resetting Security Officer authorizations to one 59
Setting a new user limit 60
Setting Entrust/RA preferences 62
Modifying Entrust/RA to improve database searches 71
Updating the time zone for database reporting 72
Changing algorithms 73
Changing the certificate hashing algorithm 73
Changing the protocol encryption algorithm 73
Changing the length of the user encryption key 74
Assigning multiple processes to a subsystem 75
Guidelines for assigning processes 75

Using a Combined CRL 77
Turning the Combined CRL on and off 78
Changing the Combined CRL lifetime 79
Publishing the Combined CRL to a file 80
Limiting the size of the published Combined CRL 81
Specifying additional CRL distribution points 83
Specifying the CDP in URL form 83
Recovering the CA profile 86
Exporting the CA certificate 88
Viewing the contents of a CA certificate 90

CHAPTER 3
Using CA hardware 93

Updating the CA hardware password 94
Updating the CA key pair on hardware 96
Moving the CA key pair to a PKCS #11 v1 hardware device 97
Updating the CA key pair to a second hardware device 98
Hardware-based database protection 100
CA hardware devices and redundancy 100
When to enable hardware-based database protection 100
Identifying CA hardware libraries when using separate devices . . . 101
Enabling and disabling hardware-based database protection 102

CHAPTER 4

Managing Entrust/Authority data for Informix 105

Backing up Entrust/Authority data 106
What is a backup? 107
What types of backups does Entrust/Authority perform? 107
Where is Entrust/Authority data backed up? 110
Scheduling automatic backups with the graphical interface 113
Changing the number of backups with the graphical interface 117
Changing the backup directory with the graphical interface 118
Configuring automatic backup settings from the command line . . . 120
Initiating manual backups 122
Verifying full backups 124
Archiving backups 124
Validating the database 126
Restoring data from backups 127
Deciding which backup to use when you restore 127
Restoring the Entrust/Authority data files 128
Restoring the file authauto.ini 129
Restoring the file onconfig 130
Restoring the Entrust/Authority database 130
What's next? 132
Restoring Entrust/Authority data to the Directory 133
Re-encrypting the database 136
Changing the Informix database password 139

CHAPTER 5

Customizing Entrust/PKI 141

Creating roles 142
Overview of roles 142
When to create a role 148
Examples of roles you can create 149
Viewing roles 151
Viewing and saving a record of a role's settings in a file 156
Descriptions of permission categories 157
Creating a role 161
Checking permission dependencies 165
Setting Entrust/PKI user policies 167
Overview of policy settings 168
Changing a user policy 184
Creating a new user policy 185
Assigning default certificate settings to a role 187
Enforcing the use of custom policy certificates 187
Modifying the New User dialog box 189
Exporting the template definition file 190
Customizing the New User dialog box 191
Importing the template definition file 198
Testing the template definition file 199

CHAPTER 6

Cross-certifying with other CAs 203

About cross-certification 204
Establishing trust using cross-certification 205
Cross-certifying online 208
Online cross-certification requirements 208
Performing online cross-certification 208
Offline cross-certification 218
Offline cross-certification requirements 218
Performing offline cross-certification 218
Importing the cross-certificate you requested 226
Displaying information about cross-certified CAs 233

Removing the cross-certificate you requested from the Directory 236
Revoking a cross-certificate 238
Updating a cross-certificate 241

CHAPTER 7
Creating a hierarchy of CAs 243

About hierarchy of CAs 244
Implications of a hierarchy 245
When to create a hierarchy of CAs 246
Creating a hierarchy of CAs 248
Adding a subordinate 248
Becoming subordinate to an Entrust or third-party superior CA . . . 252
Removing a subordinate CA from a hierarchy 257
When to revoke a subordinate CA 257
Reinstalling a subordinate CA 259
Updating a third-party subordinate CA's certificate 265
Saving a CA certificate to file 266
Preparing the Directory 268
Chaining and referring Directories 268

CHAPTER 8
Customizing certificates 269

About certificates 271
Entrust/Authority certificate categories 271
Dividing certificate categories into certificate types 272
Why customize certificates? 273
Working with Entrust/Authority certificate definitions 274
Working with the master.certspec file 277
Creating the master.certspec file 278
Opening the master.certspec file 279
Processing changes to the master.certspec file 280

Customizing Enterprise and Web certificates 282
Creating and modifying a user certificate type 283
Editing a certificate type definition 285
Editing certificate extensions 287
Editing a variable 297
Setting up default values 300
Example of user certificate types 302
Entrust/Timestamp and Entrust/Profile Server types 303
Sample certificate types 303
Customizing policy certificates 306
Creating and modifying a certificate type 307
Editing a certificate type definition 308
Editing certificate attributes 310
Editing a variable for a certificate attribute 313
Customizing cross-certificates 314
Creating and modifying a cross-certificate type 314
Editing a certificate type definition 315
Editing certificate extensions 318
Examples of cross-certificate types 323
Customizing CA certificates 325
Excluding certificate extensions 328
Making a certificate type obsolete 330
Defining a replacement certificate type 330
Customizing database fields 332
Defining a database field 333
Example of database fields 334
Removing X.509 certificate extensions 336
X.509 certificates 336
X.509 CA certificates 337
Removing default extensions from an X.509 certificate 337
Turning off revocation checking for foreign certificates 339

CHAPTER 9
Command-line interface for Entrust/Authority Master Control 341

Before you begin 343

Initializing Entrust/PKI 345
Starting and stopping Entrust Master Control Command Shell 347
Starting Entrust Master Control Command Shell 347
Stopping Entrust Master Control Command Shell 347
Logging in and out 349
Logging in to Entrust/Authority Master Control 349
Logging out of Entrust/Authority Master Control 349
Changing Master User passwords 351
Starting and stopping the Entrust/Authority service 352
Displaying the status of the Entrust/Authority service 352
Starting the Entrust/Authority service 353
Stopping the Entrust/Authority service 354
Starting commands automatically 354
Enabling and disabling Entrust/Authority subsystems 357
Administering the database 360
Re-encrypting the database 360
Backing up the Entrust/Authority database 361
Updating the time-zone information for reports 363
Listing advanced configuration variables 364
Validating the integrity of the database 365
Enabling and disabling hardware-based database protection 367
Administering the Directory 368
Restoring CA data or all certificates to the Directory 368
Writing CRLs and ARLs to the Directory 369
Setting the CA or Directory Administrator password 370
Changing the Directory Administrator DN 371
Setting the security policy 373
Changing the certificate hashing algorithm 373
Changing the protocol encryption algorithm 374
Changing the length of the user encryption key 375
Changing the number of Master User authorizations 377
Distributing setup information 378

Maintaining the CA 381
Recovering the CA profile 381
Resetting the certificate serial number counter 382
Displaying a list of CA certificates 382
Exporting a CA certificate to a file 383
Changing the lifetime of the next CA certificate 385
Changing the CA key pair algorithm 386
Updating the CA key pair 389
Revoking the CA certificate 392
Recovering all users 394
About revocation lists 395
Lifetimes and update periods 395
Exporting revocation lists 396
Changing the ARL lifetime 398
Examples 400
Modifying behavior for certificate and revocation list extensions 402
Excluding the privateKeyUsage extension from user certificates . . . 402
Excluding the privateKeyUsage extension from CA certificates . . . 404
Excluding the basicConstraints extension from user certificates . . . 405
Setting the format of key identifier extensions 406
Excluding the reasonCode extension from revocation lists 408
Using a Combined CRL 410
Turning the use of Combined CRL on and off 410
Changing the Combined CRL lifetime 411
Establishing compatibility with Entrust-Ready applications 413
Adding information to user entries 415
Encoding DNs 417
Offline initialization of a subordinate CA 419
Creating and recovering Security Officers 423
Creating a Security Officer 423
Recovering a Security Officer 424
Changing the number of Security Officer authorizations 425
Reissuing a Security Officer's policy certificate 427
Listing Security Officers 428
Accessing certificate definition files 430

Monitoring Entrust/PKI 432
Checking audit logs 432
Using performance counters 434
Getting help on Entrust Master Control Command Shell 438
Getting help on command names and syntax 438
Obtaining a description of an error code 440
Displaying the command shell version 440
Entrust Master Control Command Shell security classes 442
Command summary 443

APPENDIX A
Audit logs 445

Audits logged for revocation lists 482

APPENDIX B
Customizing the initialization files 483

Note on default values 484
Customizing the entrust.ini file 485
Customizing the entrustra.ini file 520
Customizing the entmgr.ini file 527

APPENDIX C
Navigating the log files 551

APPENDIX D
Disaster recovery and high availability 553

Overview 554
What is disaster recovery planning? 554
What is high availability? 555
Resources 555
Identifying and assessing risks 557
Resources 560

Disaster recovery strategies 561
Creating a backup and restore strategy 561
Using remote replication 561
Resources 562
Disaster-proofing your Entrust/PKI 564
Backup and restore for Entrust/PKI 564
Enhanced database protection and remote replication 564
Example architecture using a disaster recovery site 564
Ensuring a successful transfer to the disaster recovery site 565
Ensuring service availability 566
Clustering 566
Load balancing 566
Hardware redundancy 566
Power source protection 567
Monitoring a highly available system 567
Resources 568
Ensuring data availability 569
Disk mirroring 569
RAID 569
Intelligent disk arrays 570
Resources 570
Determining Entrust/PKI availability requirements 572
Determining availability requirements for client applications 572
Understanding available operations in various scenarios 573
Determining availability requirements for components 574
Using shadow Directories with Entrust/PKI 574
Configuring Entrust/Authority in a high availability system 576
Installing on a shared drive versus local drives 576
Using multiple shadow Directories 576
Configuring high availability software for use with Entrust/PKI . . . 577
Monitoring a highly available Entrust/PKI 577
Using multiple network interface cards and virtual interfaces 578
Sample Entrust/PKI architecture incorporating high availability . . . 580
Ensuring a successful Entrust/PKI failover 582
Why a failover may be necessary 582
What happens during failover? 582

Case studies of highly available Entrust/PKI systems 583
Case Study 1 583
Case Study 2 585

APPENDIX E
Running Entrust/PKI in an Identrus environment 589

Entrust/PKI in the Identrus environment 590
Interoperability with the root CA 590
Interoperability with OCSP responders 590
Interoperability with smartcards 591
Installing Entrust/PKI in the Identrus environment 592
Providing Identrus-specific input for installation prompts 592
Configuring the entmgr.ini file 592
Specifying the value that represents the OCSP responder URL 594
Providing timely revocation information to an OCSP responder . . . 595
Providing information to your smartcard vendor 595
Changes to the certificate definition file 597
Disabling revocation list checking 600
Changing the URL of an OCSP responder 602

APPENDIX F
PKIX and FPKI compliance 603

PKIX and FPKI compliance checks 604
Strict PKIX compliance checking 604
Strict FPKI compliance checking 604
References 605

APPENDIX G
Named elliptic curves. 607

APPENDIX H
Server security. 611

Security checklist 612

Index. 615

About this book

This book is one of two user guides included with Entrust/PKI 6.0. It includes advanced-level information and customization procedures that are not part of the ordinary day-to-day operation of Entrust/PKI 6.0. For the day-to-day operations performed using Entrust/RA 6.0, see *Using Entrust/PKI 6.0 on Windows*. For installation instructions, read *Installing Entrust/PKI 6.0 on Windows.*

This guide does not contain information for users of Entrust-Ready applications. Consult the documentation that accompanies each application.

You don't have to read the whole guide to make the most of Entrust/PKI. You may want to read the "About Entrust/PKI" chapter in *Using Entrust/PKI 6.0 on Windows* to familiarize yourself with concepts about public-key infrastructures and the Entrust cryptographic model.

Typographic conventions

Following are typographic conventions which appear in this user guide:

- Commands and text that you must enter appear in bold Courier type and look like this:

  `user_setattribute`

- Variables appear in italic Courier type and look like this:

  `name_of_person`

- Options (which by definition are things you may or may not choose to specify) appear in Courier type in angle brackets. For example:

  `<option>`

Chapter 1

Working with audit logs and creating reports

Using Entrust/RA, administrative users can monitor Entrust/PKI activity and create general or highly detailed reports about this activity.

This chapter includes the following sections:

- "About audit logs" on page 18
- "Viewing audit logs in the Windows Event Viewer" on page 27
- "Producing reports in Entrust/RA" on page 30

About audit logs

Audit logs record all transactions that occur when you run Entrust/PKI, such as administrative user logins and logouts, user activation, key updates, and any system failures. Use audit logs to troubleshoot the system and generally monitor user activity.

Only certain default administrative roles can view audit logs in Entrust/RA. See Table 1 and "Viewing roles" on page 151.

Table 1: Default administrative user roles that can view audit logs

Administrative role	Authorization
Security Officer	Can view all logs
Administrator	Can view all logs
Directory Administrator	Cannot view logs
Auditor	Can view all logs

Audit logs are stored in the \authdata\log\ folder on the server that hosts Entrust/Authority. Audit logs are stored in two formats: secured text and plaintext.

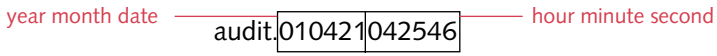
Secured audit logs

Secured audit log files are protected using a message authentication code (MAC) and stored on the Entrust/Authority host server in the \authdata\log\ folder. A MAC is calculated for each audit log and appended to the file and guarantees that an audit log has not been modified since it was created.

Secured audit log files are named as follows:

`audit.`*`date&time`*

For example:



By default, when a file exceeds 1 Mbyte, a new file is created for subsequent audit logs. Eventually there will be many audit log files and you must delete or archive them. Check with your organization's archival policy—if one exists—before you archive audit log files.

Error messages in manager.log and mgraudit.log

Error messages are also logged in the manager.log file, located in the \authdata\log\ folder. By default, the manager.log file rolls over every 1 Mbyte and keeps only the six most recent files of each type. When the manager.log file and the mgraudit.log file reach 1 Mbyte, the files are renamed manager.log-0 and mgraudit.log-0 respectively, and new manager.log and mgraudit.log files are created. If you want to keep more than six files, copy the older files to a different location. This prevents them from being deleted.

As the log files approach the default maximum file size, or the one you have specified in entmgr.ini, then you may see the lock files manager.log.lck and mgraudit.log.lck. They are created temporarily and exist until the log file reaches its maximum size and the next log file is created.

Modifying the default behavior of log files

You can control the default behavior of log files by modifying the Filesize and NumFiles settings in the entmgr.ini file. See "Customizing the initialization files" on page 483 for more information.

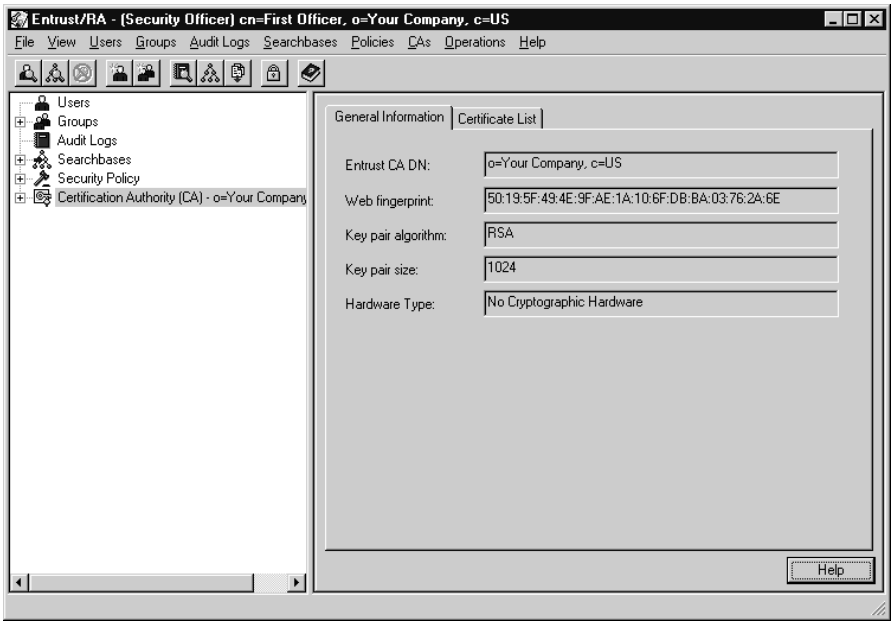
Viewing audit logs in Entrust/RA

You can view audit logs in Entrust/RA as described in the following procedure. Alternatively, you can view audit logs using the Entrust Master Control Command Shell. See "Command-line interface for Entrust/Authority Master Control" on page 341.

To view audit logs in Entrust/RA

1. Log in to Entrust/RA as an administrative user. See the "Setting up the Certification Authority (CA)" chapter in *Using Entrust/PKI 6.0 on Windows*.

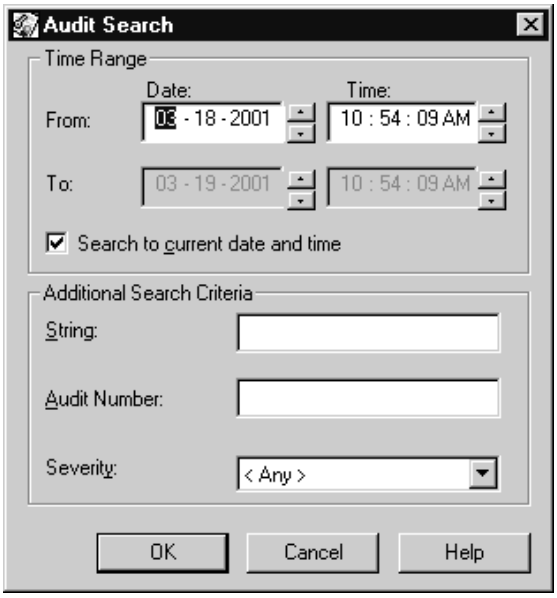
Entrust/RA appears.



Only certain administrative users can view audit logs. See Table 1 on page 18 for more information.

2 Click Audit Logs > Search.

The Audit Search dialog box appears.



3 Set the *Date* and *Time* in the *From* and *To* fields. Select *Search to current date and time* to include the most recent logs in your search.

You can choose one-day and one-minute increments or highlight the date and time and type new values.

4 Type a text string in the *String* field to find audit logs that include such text in any of the audit log fields (see Table 2 on page 23).

If you're not looking for a particular text string, use the wildcard (*) setting.

5 Type a log number in the *Audit Number* field, to find a particular event log number.

If you're not looking for a particular log number, use the wildcard (*) setting.

6 To search for logs based on a particular severity, choose *ALARM*, *Event*, or *Log* in the *Severity* drop-down list.

If you're not looking for a particular type of log, use the default < *Any* > setting.

7 Click *OK*.

A dialog box appears stating that the operation was completed successfully. The *Audit Log Entries* property page displays the audit logs.

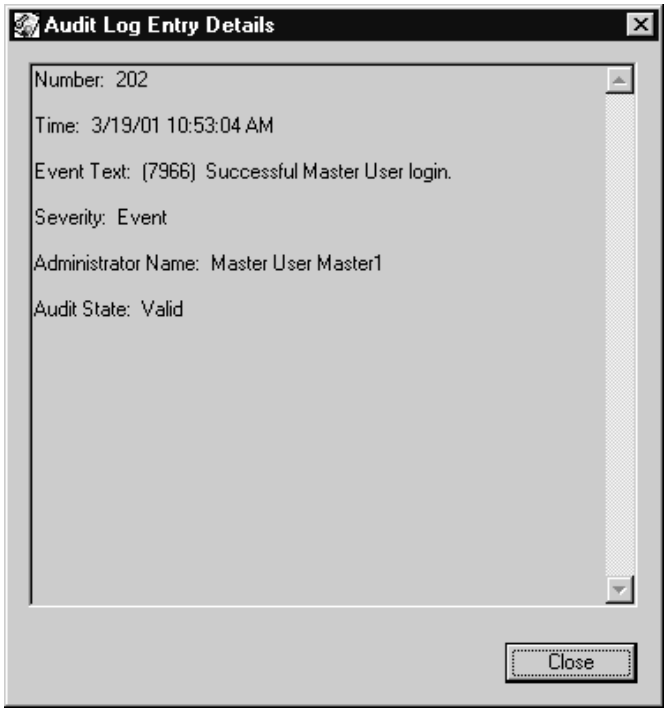
Viewing an audit log in Entrust/RA

You can view all the information about an audit log in the Audit Log Entry Details dialog box.

To view an audit log in detail in Entrust/RA

1. Search for a range of audit logs as described in "To view audit logs in Entrust/RA" on page 19.
2. Double-click an audit log.

   The Audit Log Entry Details dialog box appears.



You have now viewed an audit log in detail.

Description of audit log fields in Entrust/RA

Table 2 lists the log fields that appear in the *Audit Log Entries* property page. "Audit logs" on page 445 lists all audits and a brief description of each, and describes what actions you must take when certain audits occur.

Table 2: Description of audit log fields in Entrust/RA

Field	Description
Num	Displays the unique audit log number, which increments by one for each log, starting at 1.
Time	Displays the date and time when an audit event occurred. The time is based on the time zone of the machine the administrator is logged into.
Event Text	Displays a brief explanation of what event the audit log is related to, along with an audit identifying number. See "Audit logs" on page 445 for audit descriptions.
Severity	Displays one of three severity levels, as follows: • Log is the least severe level. • Event is the next severity level. • ALARM requires the attention of a Master User.
Administrator Name	Displays the DN of the administrative users that performed this action.
Target Name	Refers to the distinguished name (DN) of the entity to which the action, reported by the audit, was applied. If the action does not apply to a particular entity, this field is not displayed.

Table 2: Description of audit log fields in Entrust/RA (continued)

Field	Description
Extra	May show additional information about the audit.
Audit State	Shows whether an audit log is valid, according to a message authentication code (MAC), which guarantees that it has not been modified since the log was created. Audit states are as follows: • Valid. The audit log is valid. Note that all audit states other than *Valid* suggest tampering with the audit log file, or corrupt data. Any audit state other than *Valid* indicates a situation requiring investigation by a Master User. • Invalid. The audit log is corrupt. • Missing. The audit log is missing from an audit log file. • Out of Range. The audit log unique number is outside the range specified in the audit log file header. • Parse Error. The encoded audit log could not be decoded. This audit log is corrupt. • Out of Sequence. One or more audit logs were found to be out of sequence. For example, audit log 55 was found after audit log 60. • Bad File Header. The audit log file header could not be read; in this case none of the audit logs in the audit log file can be validated.

Sorting audit logs in Entrust/RA

After you search for audit logs, you can sort the audit logs by one of several list categories. For example, to sort the audit logs into groups of Logs, Events, and ALARMs, click *Severity.* The audit logs will display beginning with the most severe category, ALARM, and ending with the least severe category, Log.

To sort audit logs in Entrust/RA

1. Search for audit logs as described in "To view audit logs in Entrust/RA" on page 19.

The *Audit Log Entries* property page displays the audit logs.

Click a title bar button to sort the logs accordingly.

Audit Log Entries

Num	Time	Event Text
202	3/19/01 10:53:04 AM	(7966) Successful Master User login.
201	3/19/01 10:50:54 AM	(7822) Master User logged off.
200	3/19/01 10:45:52 AM	(7966) Successful Master User login.
199	3/19/01 10:33:47 AM	(7831) Database backup failed.
198	3/19/01 9:34:05 AM	(7831) Database backup failed.
197	3/19/01 9:16:56 AM	(7900) Authority revocation list (ARL) written to the Dire
196	3/19/01 8:34:13 AM	(7831) Database backup failed.
195	3/19/01 7:34:22 AM	(7831) Database backup failed.
194	3/19/01 6:34:40 AM	(7831) Database backup failed.
193	3/19/01 5:34:49 AM	(7831) Database backup failed.
192	3/19/01 4:34:57 AM	(7831) Database backup failed.
191	3/19/01 3:35:05 AM	(7831) Database backup failed.
190	3/19/01 2:59:54 AM	(7914) Entrust full data backup failed.
189	3/19/01 2:59:54 AM	(7829) Infrastructure data files backup failed.
188	3/19/01 2:59:54 AM	(7831) Database backup failed.
187	3/19/01 2:59:15 AM	(7776) User State History database table trimmed.
186	3/19/01 2:58:15 AM	(7900) Authority revocation list (ARL) written to the Dire
185	3/19/01 2:35:23 AM	(7831) Database backup failed.
184	3/19/01 1:59:48 AM	(7854) Database integrity check was successful.
183	3/19/01 1:59:48 AM	(7712) Audit trail integrity check passed.

2 Click a title bar button to sort the logs accordingly.

You have now sorted the audit logs.

Clearing audit logs in Entrust/RA

After you search, sort, and view audit logs in the *Audit Log Entries* property page, you can clear the property page to perform new searches. To do this, click Audit Logs > Clear Audit Logs List.

Saving audit logs in Entrust/RA

As an administrative user with sufficient permissions, you can save audit logs to a location on the machine hosting Entrust/RA. Audit logs are saved as text files (*.txt).

Files saved with the .txt extension can later be opened in a spreadsheet application where you can sort audit logs as you can in Entrust/RA. Note that .txt files are not secured. They are not encrypted and signed. Anyone can open them in a spreadsheet application.

To save audit logs in Entrust/RA

1 Repeat Step 1 through Step 3 as described in "To view audit logs in Entrust/RA" on page 19.

2 Click Audit Logs > Save As.

3 Select the location, a name, and an extension; or you can accept the default and click *OK*.

You have now saved an audit log.

Viewing audit logs in the Windows Event Viewer

If you log in to the server hosting Entrust/PKI 6.0 as a local system administrator, you can view audit logs in the Windows NT Event Viewer utility. You can choose the severity of the audit logs that appear in the viewer.

By default, audit logs are not sent to the Event Viewer. To send audit logs to this utility and to choose the severity of the logs sent, you must modify the entmgr.ini file as described below.

To send audit logs to the Event Viewer

1. Open the entmgr.ini file in a text editor.

   By default, the entmgr.ini file is located in the entmgrdata/manager folder. You can browse to this location or find the file using the Windows Find utility.

2. Under the [logging] section heading, type a number after the level= line to indicate the severity level of audit logs that you want to appear in the Event Viewer:

   - type 0 if you don't want to send any audit logs to the Event Viewer
   - type 1 to send only ALARM level audit logs to the Event Viewer
   - type 2 to send ALARM and Event level audit logs to the Event Viewer
   - type 3 to send all audit logs (Log, Event, and ALARM) to the Event Viewer

   In the Event Viewer, audit log severity levels are represented iconically. See Table 4 on page 30.

Note: Ignore the facility= line. This is used for UNIX logging only. It does not affect audit log reporting in Windows.

---

3. Save the file.

   You have now sent audit logs to the Event Viewer.

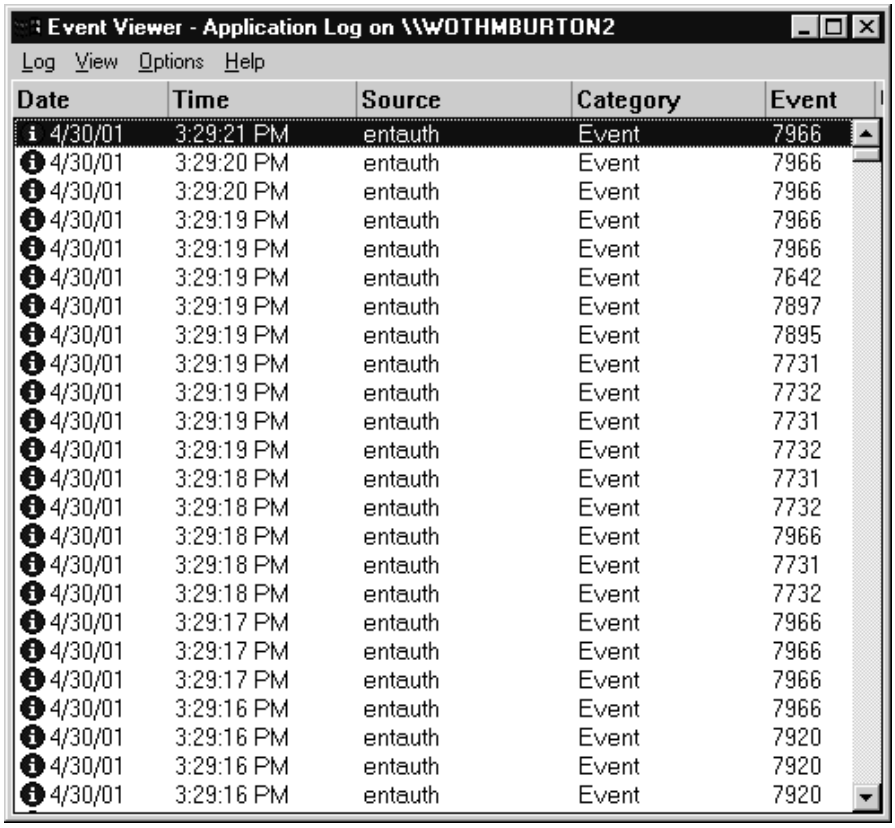
To open the Event Viewer

1. In the Windows taskbar, click Start > Programs > Windows Explorer.

   The Exploring window appears.

2. Click the drive letter containing the Windows folder.
3. Double-click the Windows folder.
4. Double-click the system32 folder.
5. Scroll down to the file named Eventvwr.exe and double-click it.

The Event Viewer window appears.



Date	Time	Source	Category	Event
4/30/01	3:29:21 PM	entauth	Event	7966
4/30/01	3:29:20 PM	entauth	Event	7966
4/30/01	3:29:20 PM	entauth	Event	7966
4/30/01	3:29:19 PM	entauth	Event	7966
4/30/01	3:29:19 PM	entauth	Event	7966
4/30/01	3:29:19 PM	entauth	Event	7642
4/30/01	3:29:19 PM	entauth	Event	7897
4/30/01	3:29:19 PM	entauth	Event	7895
4/30/01	3:29:19 PM	entauth	Event	7731
4/30/01	3:29:19 PM	entauth	Event	7732
4/30/01	3:29:19 PM	entauth	Event	7731
4/30/01	3:29:19 PM	entauth	Event	7732
4/30/01	3:29:18 PM	entauth	Event	7731
4/30/01	3:29:18 PM	entauth	Event	7732
4/30/01	3:29:18 PM	entauth	Event	7966
4/30/01	3:29:18 PM	entauth	Event	7731
4/30/01	3:29:18 PM	entauth	Event	7732
4/30/01	3:29:17 PM	entauth	Event	7966
4/30/01	3:29:17 PM	entauth	Event	7966
4/30/01	3:29:17 PM	entauth	Event	7966
4/30/01	3:29:16 PM	entauth	Event	7966
4/30/01	3:29:16 PM	entauth	Event	7920
4/30/01	3:29:16 PM	entauth	Event	7920
4/30/01	3:29:16 PM	entauth	Event	7920

6 Click Log > Application.

Under the Source column, look for "entauth". The Event Viewer does not describe events, but it does provide all the information you'll need to review system activity.

To view an audit log in detail, double-click an event. The Event Detail dialog box appears and displays complete event details.

You have now viewed an audit log in the Windows Event Viewer.

Description of audit log fields in the Event Viewer

Table 3 lists the log fields that appear in the Windows Event Viewer. Consult the Microsoft online Help for more information about the Event Viewer utility and the audit log fields in the table.

Table 3: Description of audit log fields in the Windows Event Viewer

Field	Description
Severity (icon)	Displays the severity of the event using icons. There are three severity types: • Information: akin to Log in the Audit Log Entries property page in Entrust/RA • Warning: akin to Event in the Audit Log Entries property page in Entrust/RA • Error: akin to ALARM in the Audit Log Entries property page in Entrust/RA
Date	Displays the date (month/day/year) when the event occurred.
Time	Displays the time (hour/minute/second) when the event occurred.
Source	Displays the name of the application that sent the event. Events originating from an Entrust application display "Entrust/Authority" in this field.
Category	Displays the classification of the event as defined by the source (above).
Event	Displays an identification number associated with the event.
User	Displays the name of the user logged in to the computer that sent the event.
Computer	Displays the name of the computer that sent the event.

Producing reports in Entrust/RA

Entrust/RA has a reporting feature that lets you create a list of all users in your Certification Authority (CA) domain, or lists of users in various states. Reports are saved in text files with .EtReport extensions in a folder of your choice. The information fields within the reports are delimited by tab characters. You can view these reports using any spreadsheet or text editor application.

Guard these reports carefully. They may list user setup information (reference numbers and authorization codes) that is used to create user profiles.

To create a user report

1. Log in to Entrust/RA as an administrative user (see the "Setting up the Certification Authority (CA)" chapter in *Using Entrust/PKI 6.0 on Windows*).
2. Click Operations > Create Report and click a report type.

   The Save As dialog box appears.
3. Choose a destination folder and type a name for your report. You should use the .EtReport filename extension.
4. Click *Save*.

   You have now created a user report which you can view using any spreadsheet application.

Explanation of report fields

Table 4 lists report fields as they appear in the report file, and contains a brief description of each. Note that not all fields may have values (for instance, a user who has not been disabled will not have a date under User Disabled). The report fields are separated by tab characters because the reports are meant to be imported into other applications (such as spreadsheet or text editing applications).

Table 4: Description of user report fields

Field	Description
Name	User's distinguished name (DN); for example, cn=Alice Gray,o=Your Company,c=US.
State	User's current state, which may be one of the following: Added, Activated, Deactivated, Revoked, Key Recovery, or DN Change. (You can create a report that lists only users in one of these states; see "To create a user report" on page 30.)
Upgraded by	The last user who performed encryption key update. The values are one of Client, Administrator, or Unknown.

Table 4: Description of user report fields (continued)

Field	Description
Reference #	Reference number from the user's activation codes. Make sure to *keep this information confidential* and that you give it to the user securely according to your organization's security policy. Remind the user to keep the information confidential until they use it to log in to Entrust/PKI 6.0. Depending on your organization's security policy, you may not be able to view this information.
Authorization	Authorization code from the user's activation codes. Make sure you *keep this information confidential* and that you give it to the user securely according to your organization's security policy. Remind the user to keep the information confidential until they use it to log in to Entrust/PKI 6.0. Depending on your organization's security policy, you may not be able to view this information.
Added	Date and time the user was added.
Activated	Date and time the user entered the activation codes and created a profile using an Entrust-Ready application (for example, Entrust/Entelligence or Entrust/RA in the case of administrative users).
Key Rec Init	Date and time a Security Officer or Administrator started a key recovery operation for the user.
Key Rec Done	Date and time the user completed a key recovery operation in Entrust/Entelligence or Entrust/RA (in the case of administrative users).
User Deactivated	Date and time a Security Officer or Administrator deactivated the user.
Deactivated Reason	Reason for deactivating the user. Entered automatically by Entrust/RA. For example, "User disabled by Administrator".
Previous status	User state prior to current state (see "State" above).
Enc Updt Pend	Whether or not encryption key pair update is pending for a user.
Signing Cert Desc	Describes the type of signing key pair (for example, 1024-bit DSA).
Signing Cert Issue	Date and time the user's current verification certificate was issued.
Signing Cert Expiry	Date and time the user's current verification certificate will expire.
Signing Cert Revoked	Date and time the user's most recent verification certificate was revoked.
Sign. Cert Revok. Reason	Reason for revoking the user's verification certificate. May be one of Superseded, Key Compromise, Affiliation Change, Unspecified, or Cessation of Operation. See the "Administering individual users" chapter in *Using Entrust/PKI 6.0 on Windows* for information about revocation reasons.
Encrypt Cert Desc	Describes the type of encryption key pair (for example, 1024-bit RSA).

Table 4: Description of user report fields (continued)

Field	Description
Encrypt Cert Issue	Date and time the user's current encryption certificate was issued.
Encrypt Cert Expiry	Date and time the user's current encryption certificate will expire.
Encrypt Cert Revoked	Date and time the user's most recent encryption certificate was revoked.
Encr. Cert Revok. Reason	Reason for revoking the user's encryption certificate. May be one of Superseded, Key Compromise, Affiliation Change, Unspecified, or Cessation of Operation. See the "Administering individual users" chapter in *Using Entrust/PKI 6.0 on Windows* for information about revocation reasons.
Pending DN	User's current distinguished name and user's new distinguished name.

Chapter 2

Other administrative tasks

Administering Entrust/PKI requires that you regularly perform certain tasks, such as reviewing audit logs or enabling users. There are, however, some important Entrust/PKI tasks that you perform with little regularity or very few times. For instance, you may occasionally change a Master User password, or you may recover the CA profile. This guide refers to such tasks as "other administrative tasks".

This chapter focuses on such tasks. In particular, it covers the following *other* administrative tasks:

- "Using the Directory Browser" on page 34
- "Showing and hiding the Entrust/RA toolbar" on page 44
- "Setting up Security Officers for key recovery" on page 45
- "Changing passwords and password rules" on page 48
- "Changing the number of authorizations for sensitive operations" on page 57
- "Setting a new user limit" on page 60
- "Setting Entrust/RA preferences" on page 62
- "Updating the time zone for database reporting" on page 72
- "Changing algorithms" on page 73
- "Assigning multiple processes to a subsystem" on page 75
- "Using a Combined CRL" on page 77
- "Specifying additional CRL distribution points" on page 83
- "Recovering the CA profile" on page 86
- "Exporting the CA certificate" on page 88
- "Viewing the contents of a CA certificate" on page 90

Using the Directory Browser

Directory Administrators and administrative users with sufficient permissions use the Directory Browser to perform Directory-related tasks, such as adding and deleting entries, and adding, changing, and deleting attributes.

Note: If you are using Microsoft Active Directory, Directory-related tasks are performed using Microsoft Active Directory tools.

In the procedures that follow, you will learn how to do the following:

- "Logging in to the Directory Browser" on page 34
- "Adding a Directory entry" on page 35
- "Finding a Directory entry" on page 37
- "Deleting a Directory entry" on page 38
- "About Directory attributes" on page 39
- "Adding an attribute to a Directory entry" on page 40
- "Replacing a Directory attribute value" on page 41
- "Deleting a Directory attribute or attribute value" on page 42

You can also use the Directory Browser to change your administrative user profile password, to change the Entrust Directory Administrator password (the password used to connect to the Directory), and to perform bulk operations. See "Changing an administrative user profile password" on page 52, "Changing the Entrust Directory Administrator password" on page 54, and the "Administering multiple users" chapter in *Using Entrust/PKI 6.0 on Windows* for more information on these topics.

Logging in to the Directory Browser

Directory Administrators and Administrators log in to the Directory Browser through the Start menu. However, Administrators have permission to open Entrust/RA; Directory Administrators do not.

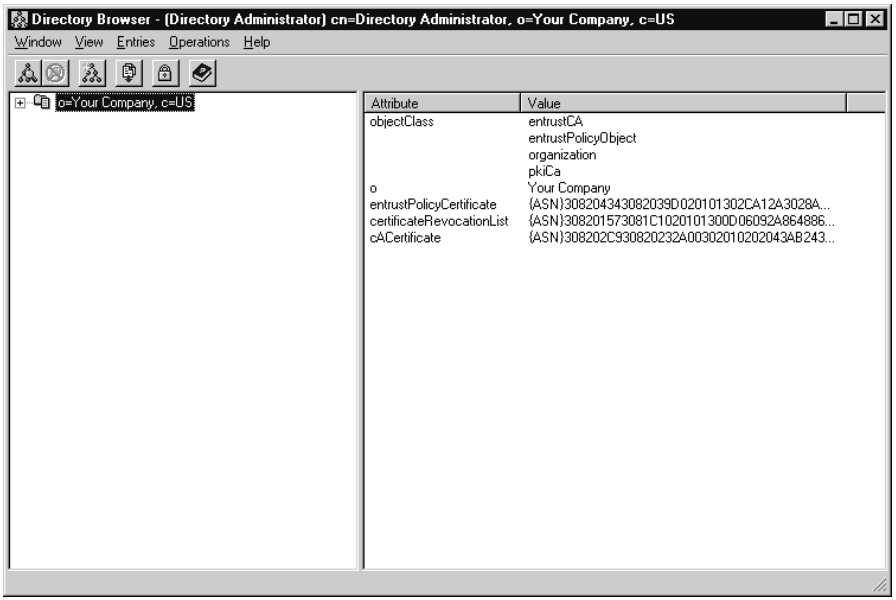
To log in to the Directory Browser

1. In the Windows taskbar, click Start > Programs > Entrust PKI > Entrust RA.

2 If you log in as an Administrator, Entrust/RA appears. Click the Directory Browser button on the toolbar to open the Directory Browser or click Operations > Directory Browser.

Click the Directory Browser button to open the Directory Browser.

If you log in as a Directory Administrator, the Directory Browser appears.



You have now logged in to the Directory Browser.

Adding a Directory entry

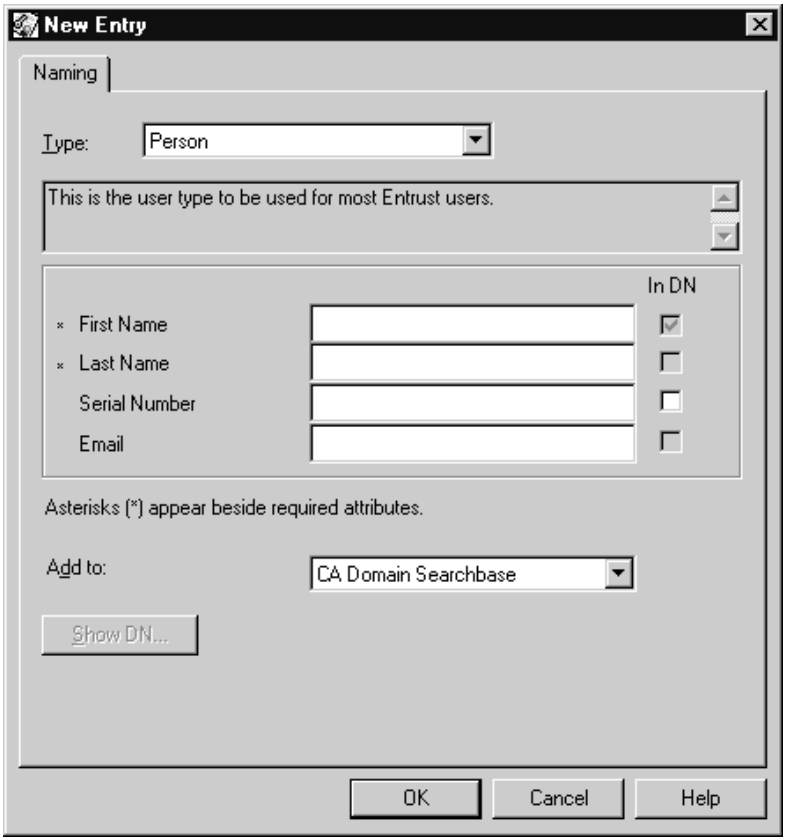
You can add entries to the Directory one at a time. For example, you might do this when a new employee joins your organization.

To add a Directory entry

1 Log in to the Directory Browser. See "Logging in to the Directory Browser" on page 34.

2 Click Entries > New Entry.

The New Entry dialog box appears.



3 In the *Naming* property page, *Person* appears in the *Type* drop-down list. If it does not, click the *Type* drop-down list and select *Person*.

4 Type the entry's first name in the *First Name* field. By default, this information is required.

An asterisk appears beside required fields. Make sure that you type information into all fields with asterisks.

Note: The information required in the *Naming* property page may differ from that described in this procedure if a Security Officer or Administrator has modified the template definition file. Contact a Security Officer if you have any questions about the appearance of this dialog box.

---

5 Type the entry's last name in the *Last Name* field. By default, this information is required.

6 Type the entry's serial number (for example, the employee number) in the *Serial Number* field. By default, this information is optional.

Depending on your organization, the serial number may not be the employee number. Check with a Security Officer if you're not sure what number to type here.

7 Type the entry's e-mail address in the *Email* field. By default, this information is optional.

8 In the *Add to* drop-down list, select the searchbase under which to add the entry.

By default, *CA Domain Searchbase* is listed first and is often the top-level searchbase in an organization. If you are uncertain about which searchbase to choose, check with a Security Officer. For more information, see the "Setting up the Certification Authority (CA)" chapter in *Using Entrust/PKI 6.0 on Windows*.

9 Click *OK*.

You have now added a Directory entry.

Finding a Directory entry

Follow the steps below to find entries in the Directory. To find users and entries using Entrust/RA, see the "Administering individual users" chapter in *Using Entrust/PKI 6.0 on Windows*.

When you search for entries with Directory attribute values that include special characters, the name you enter in the Find dialog box must match the Directory entry.

Optionally, you can use wildcards. Wildcards let you search for partial attributes. Add an asterisk (*) with a partial search string to find users that include the search string information. For example, type *dr* to find all entries named "Andrew" and "Drew". Type dr* to exclude "Andrew" and find only entries beginning with the letters "d" and "r".

You can control the number of entries that are returned in a Find. You do this in the Preferences dialog box in Entrust/RA. For details, see "To display warnings about operations" on page 62.

Canceling a Find operation

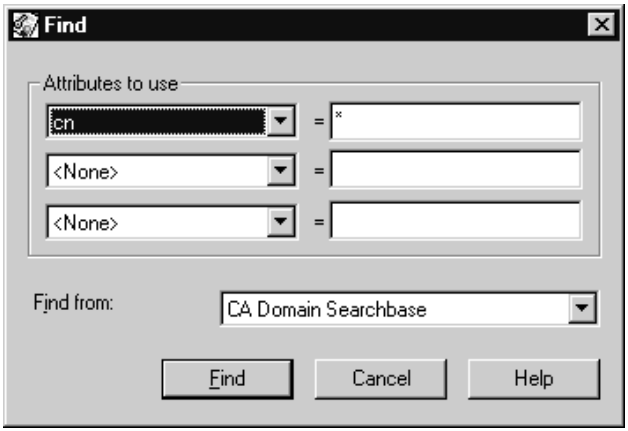
If you click the Cancel button in the toolbar before the Find operation is complete, only those entries that are found at that point display onscreen, and a message appears explaining what has happened. Entries are not found in any particular order. Therefore, if you notice results that range from A-Z after canceling a find, it does not mean that you have found every entry in the Directory that meets your search criteria.

To decrease the time it takes to find entries, specify narrow criteria for your search. Type accurate information in the *Attributes to use* fields.

To find a Directory entry

1. Log in to the Directory Browser. See "Logging in to the Directory Browser" on page 34.
2. Click Entries > Find.

   The Find dialog box appears.



3. Select Directory attributes in the *Attributes to use* drop-down lists.

   The list of Directory attributes in the drop-down lists is configured in your entrustra.ini file. See "Customizing the entrustra.ini file" on page 520.
4. Type the attribute values you want to search for in the *Attributes to use* fields. You can search for partial Directory attributes by adding an asterisk (*) with a partial string, for example, *Dr*.
5. Choose a searchbase in the *Find from* drop-down list to find entries in that searchbase. By default, *CA Domain Searchbase* is listed first and is often the top-level searchbase in an organization.
6. Click *Find*.

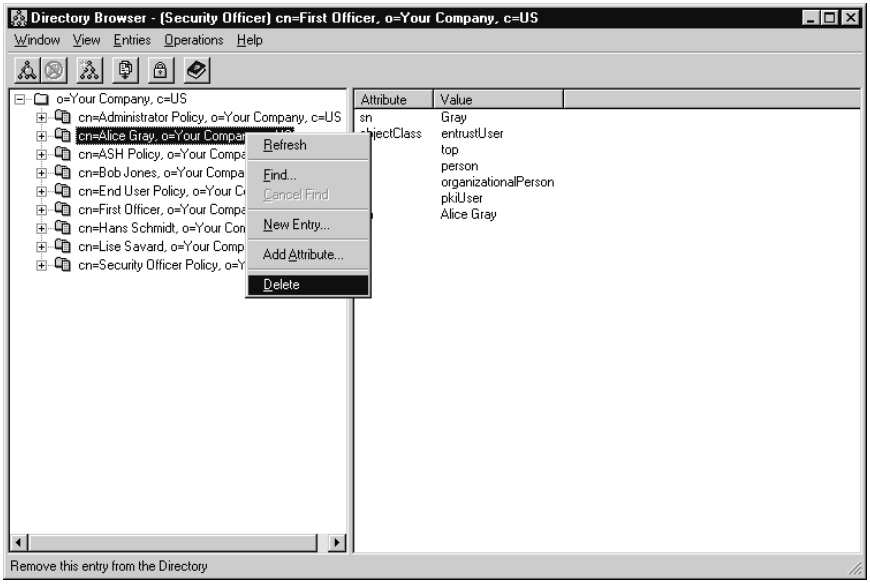
   The entries matching your search criteria appear in the tree view in the Directory Browser.

Deleting a Directory entry

Depending on your administrative permissions, you may be able to delete a Directory entry—for example, when an employee leaves your organization. When you delete a Directory entry, the corresponding user is not deleted from the Entrust/Authority database. Before deleting a Directory entry, you may want to deactivate, revoke, or archive the user.

To delete a Directory entry

1. Log in to the Directory Browser. See "Logging in to the Directory Browser" on page 34.
2. If you cannot see the entry you want to delete, search for the entry (see "Finding a Directory entry" on page 37).
3. In the Directory Browser, right-click the entry in the tree view, and click Delete in the pop-up menu.



4. If a confirmation dialog box appears, click *OK* if you're sure an Administrator has already deactivated the user. Contact an Administrator if you're not sure.

   You have now deleted a Directory entry.

About Directory attributes

You can add, change, and delete Directory attributes using the Directory Browser.

Directory attributes are not absolute and will vary depending on your organization's deployment of Entrust/PKI. You may need certain information before you can add, change, or delete an attribute. While you can enter any data in the name and value fields in the Add Attribute dialog box, the attributes permitted by your Directory schema vary. If you're not sure about which attributes you can safely add, change, or delete, contact an administrator with advanced knowledge of your Directory.

Adding an attribute to a Directory entry

You can add an attribute to a Directory entry for any of the following reasons:

- To include missing information. It may happen that the template definition file is changed to include or exclude an attribute.
- To re-add attributes that have been accidentally deleted.

You should be familiar with the conventions of your organization's Directory before you add an attribute to a Directory entry. Consult an administrator with advanced knowledge of your Directory if you have any questions about which attributes are valid and which are not.

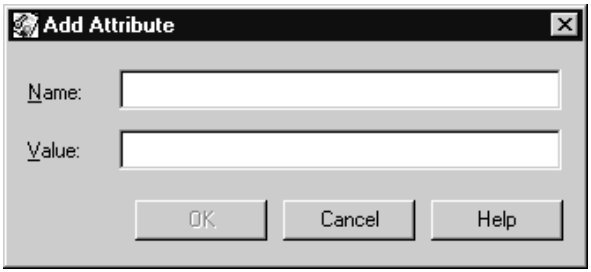
You can also add attributes to Directory entries in bulk. This is useful when adding a known value to all entries. For example, your organization may need to include employee phone numbers in its Directory entry information. While you can add this information to each entry individually, it is more efficient to do it in bulk. See the "Administering multiple users" chapter of *Using Entrust/PKI 6.0 on Windows*.

Note: When you add, change, or delete an attribute in the DN, you may have to make the same change to the user in the Entrust/Authority database. See the "Administering individual users" chapter in *Using Entrust/PKI 6.0 on Windows*.

To add an attribute to a Directory entry

1. In the Directory Browser, right-click an entry and click Add Attribute in the pop-up menu.

   The Add Attribute dialog box appears.



2. Type the attribute name in the *Name* field.
3. Type the attribute value in the *Value* field and click *OK*.

Note: Error checking prevents you from adding Directory attributes that are not supported in the Directory. You receive an error message when you enter an attribute that your Directory does not support.

You have now added an attribute to the Directory entry.

Adding a Directory attribute value

You can add additional values to an existing Directory attribute.

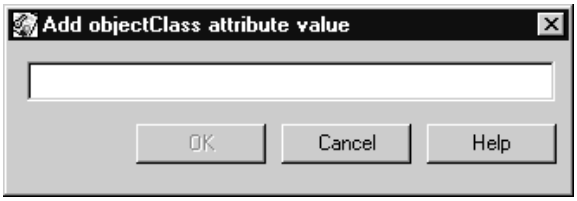
You should be familiar with the conventions of your organization's Directory before you add an attribute value to a Directory attribute. Consult an administrator with advanced knowledge of your Directory if you have any questions about which attribute values are valid and which are not.

Note: Error checking prevents you from adding Directory attribute values that are not supported in the Directory. You receive an error message when you enter an attribute value that your Directory does not support.

To add a Directory attribute value

1. In the Directory Browser, click an entry.
2. In the right pane, right-click an attribute value, and click Add in the pop-up menu.

   A dialog box appears in which you can add a Directory attribute value. This example shows one of the dialog boxes that can appear.



3. Type the attribute value and click *OK*.

   You have now added a Directory attribute value.

Replacing a Directory attribute value

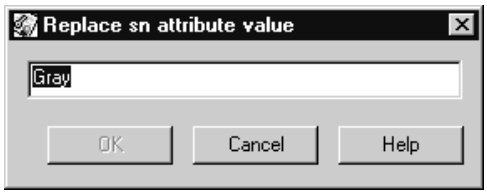
Before you replace a Directory attribute value, make sure that the attribute value is one that can be safely replaced. You should not replace attribute values that comprise a DN or that are required by your security policy.

Note: You will receive an error message when you try to change to an attribute value that your Directory does not support.

To replace a Directory attribute value

1. In the Directory Browser, click an entry.
2. In the right pane, right-click an attribute value, and click Replace in the pop-up menu.

   A dialog box appears in which you can replace Directory information. This example shows one of the dialog boxes that can appear.



3. Type the new information and click *OK*.

   You have now replaced an entry's Directory attribute value.

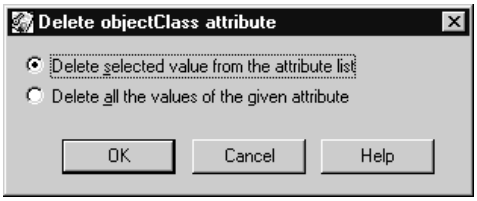
Deleting a Directory attribute or attribute value

Before you delete an attribute value, make sure that the attribute value is one that can be safely deleted. You should not delete vital attribute values that comprise a DN or that are required by your security policy.

To delete a Directory attribute or attribute value

1. In the Directory Browser, click an entry.
2. In the right pane, right-click a Directory attribute value and click Delete in the pop-up menu.

If the Directory attribute has multiple values, a message box appears asking you if you want to delete just the selected value or all of the values. To delete a single Directory attribute value, choose the first option and click *OK*.



If a Directory attribute has more than one value and you delete only one, only that value is deleted. If you delete all values, you have deleted the Directory attribute.

You have now deleted a Directory attribute value.

Showing and hiding the Entrust/RA toolbar

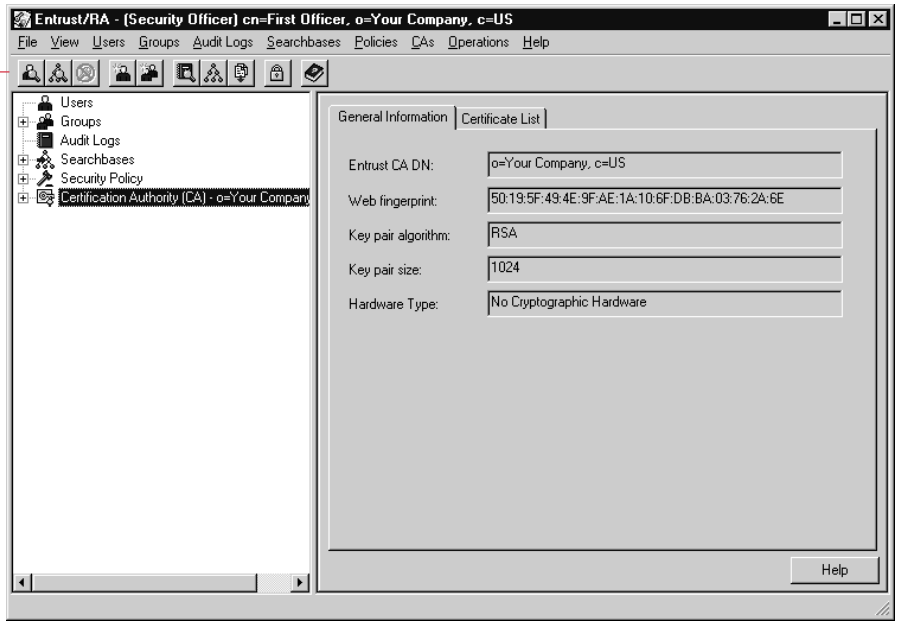
In Entrust/RA 6.0, you can show or hide the Entrust/RA toolbar. The toolbar includes buttons used to perform many common tasks, such as finding and adding users, performing audit log searches, and accessing online help.

To show and hide the Entrust/RA toolbar

1 Log in to Entrust/RA using your administrative profile. See the "Setting up the Certification Authority (CA)" chapter in *Using Entrust/PKI 6.0 on Windows*.

Entrust/RA appears.

The Entrust/RA window includes a toolbar with buttons used to perform many common tasks.



2 Click View > Toolbar.

A check mark appears beside the Toolbar menu item when the toolbar is visible. Toggle the check mark to show and hide the toolbar.

You have now shown or hidden the Entrust/RA toolbar.

Setting up Security Officers for key recovery

Whenever a Security Officer's profile is lost or that Security Officer forgets their password, that officer must be *recovered*.

Recovering a Security Officer is a two-stage process:

1. A Master User sets up the Security Officer for key recovery. This involves generating a new reference number and authorization code for the officer.
2. The Security Officer uses the new reference number and authorization code to recover the lost or corrupt profile.

The procedure in this section covers the first of these two steps.

Note: Security Officers can also be set up for key recovery in Entrust/RA. See the "Administering individual users" chapter in *Using Entrust/PKI 6.0 on Windows*.

To set up a Security Officer for recovery

1. Log in to Entrust/Authority Master Control as a Master User (for details, see the "Setting up the Entrust/PKI environment" chapter in *Using Entrust/PKI 6.0 on Windows*).
2. Click Configuration > Recover Officer.

The Recover Officers dialog box appears listing all Security Officers in your Certification Authority (CA).

3 Click the name of the Security Officer that you want to recover (if necessary, scroll through the list to find the Security Officer) and click *OK*.

A dialog box appears stating that you are about to put the selected Security Officer into key recovery and that the reference number and authorization code will be returned.

4 Enter your Master User password and click *OK*.

Entrust/Authority sets up the Security Officer for recovery and displays a dialog box with a message similar to the following:

"The officer has been set for recovery.

Reference number: 61644069

Authorization code: 67R8-3WZ9-NOCH"

5 Write down this reference number and authorization code, and click *OK*. Keep this information secure, because it's needed for the next step.

6 *In a secure manner*, give the reference number and authorization code to the Security Officer.

You have now set up a Security Officer for key recovery.

Note: The Security Officer uses this reference number and authorization code to finish the recovery process with Entrust/RA. Typically, a Security Officer starts Entrust/RA, clicks *Recover Profile*, and enters all the information needed. This is similar to creating a new profile (see the "Setting up the Certification Authority (CA)" chapter in *Using Entrust/PKI 6.0 on Windows*).

Changing passwords and password rules

Changing passwords regularly and often is an essential aspect of good security policy. Changing password rules may also be necessary to improve security; for instance, you may decide to modify the default password rules so that your organization's name cannot be used as a password.

This section describes how to perform the following password-related tasks:

- "Changing a Master User password" on page 48
- "Changing Master User password rules" on page 50
- "Changing an administrative user profile password" on page 52
- "Changing the CA Directory Access password" on page 53
- "Changing the Entrust Directory Administrator password" on page 54

Note: Password rules for Security Officers, Administrators, and end users are changed by Security Officers in user policies. See "Changing a user policy" on page 184.

Changing a Master User password

You should change your Master User password periodically to increase the security of your infrastructure. If you suspect a Master User password has been compromised, your Entrust infrastructure may be untrustworthy.

After you change the Master User password, stop and restart the Entrust/Authority service in order to pick up the password change.

Attention: Do *not* forget your Master User password. If one or two Master Users forget their passwords, the third Master User can recover the others. However, if all three Master Users lose or forget their passwords, your Entrust/PKI system will be rendered permanently and irrevocably inaccessible. See the "Setting up the Entrust/PKI environment" chapter in *Using Entrust/PKI 6.0 on Windows* for some suggestions on how to prevent this from happening.

Similarly, if multiple Master User authorizations are enabled, two Master Users are needed to change the password of the third. Losing or forgetting these two Master User passwords will make your Entrust/PKI system permanently and irrevocably inaccessible.

By default, a Master User password must obey the following rules, which are specified in the password.rules file (the default location for this file is the C:\entmgrdata\pwd folder):

- It must have at least one uppercase letter.
- It must have at least one number.
- It must have at least one lowercase letter.
- It must have at least 10 characters and no more than 64.
- Valid characters are

  A to Z, a to z, 0 to 9, and ~ ! @ # $ % ^ & * _ + - = | : ; < > ? , . /
- It must not be part of the profile folder path.
- It must not include the words "Entrust" or "Entrust Technologies".
- It must not be a person's name specified in the names.txt file. The default location for this file is the C:\Program Files\Entrust Authority\pwd folder.
- It must not be a keyboard sequence, a numeric sequence, or an alphabetic sequence.
- It must not be a dictionary word or combination of words specified in the words.sor file. The default location for this file is the C:\Program Files\Entrust Authority\pwd folder.

To change a Master User password

1. Log in to Entrust/Authority Master Control as a Master User (for details, see the "Setting up the Entrust/PKI environment" chapter in *Using Entrust/PKI 6.0 on Windows*).

   The Entrust/Authority Master Control window appears.
2. Click User > Change Password.

The Change Password dialog box appears.

Click the Master User whose password you want to change.

3 In the list, click the Master User whose password you want to change.

Master Users can change their own passwords, as well as the passwords of other Master Users.

4 Enter your new Master User password in the *New password* field.

5 Enter the same Master User password again in the *Confirm new password* field, and click *OK*.

A dialog box appears stating that you are about to change either your own password or that of another Master User.

6 Enter your old Master User password and click *OK*.

A dialog box appears confirming that the password was changed successfully.

7 Click *OK*.

You have now changed a Master User password.

Changing Master User password rules

To change password rules for Master Users, you need to modify one or more of the following files:

- password.rules, which contains the Master User password rules
- names.txt, which contains a list of people's names that cannot be used as passwords
- words.sor, which contains dictionary words that cannot be used as passwords

Depending on your organization's security policy, you may wish to modify these files to reflect rules, names, and words particular to your organization. For example, you may wish to specify that your organization name cannot be used as a password.

To modify the password rules file

1. Using any text editor, open the file "password.rules".

   The default location is C:\authdata\pwd.

2. Using the keywords "KEEP", "REJECT", "VARIABLE", and "STRING", create a one-line entry for each rule you wish to specify. The syntax is as follows (see the file itself for examples):

   ```
   keyword "brief explanation of rule, to appear in an error
   message if rule is violated" rule
   ```

   where "rule" is the word, number, or phrase you wish to include or exclude. If the rule is multi-character or a literal string, it must be enclosed in double quotes ("), for example, "Your Company". If "rule" is a number or range of characters, it must be enclosed in square brackets ([ ]). For example, [0-9] specifies the range of characters from (and including) 0 to 9.

   The caret (^) character specifies the logical inverse of the characters it precedes; that is, everything *but* the characters specified. For example, [^0-9] specifies all characters *except* those from (and including) 0 to 9.

3. Save the file and exit your text editor.

   You have now modified the password rules file.

   It's a good idea to test your password rules by trying to enter passwords that do and do not comply with your new rules.

To modify the names or words file

1. Using any text editor, open the file "names.txt" if you want to modify the list of excluded names, or "words.sor" if you want to modify the list of excluded dictionary words.

   The default location for these files is C:\Program Files\Entrust\Entrust Authority\pwd.

2. In alphabetical order, add to the list any names or words that cannot be used as passwords.

Note: You should not delete names or words from these files, because a name or a common or known word in a password reduces the security of that password.

---

3 Save the file and exit your text editor.

You have now modified the names or words file.

Changing an administrative user profile password

If you even *suspect* that your administrative user profile password has been revealed, change it immediately. For information on choosing and protecting passwords, look up "password security" in *Using Entrust/PKI 6.0 on Windows*.

If you are a Directory Administrator, you must use the Directory Browser to change your administrative user profile password.

If you have sufficient permissions to open and use Entrust/RA, you can also use Entrust/RA to perform this operation.

You can also use Entrust/Entelligence to change an administrative user profile password.

The following procedure describes how to change your administrative user profile password using Entrust/RA or the Directory Browser. For information on changing your password using Entrust/Entelligence, refer to the *Entrust/Entelligence 6.0 User Guide*.

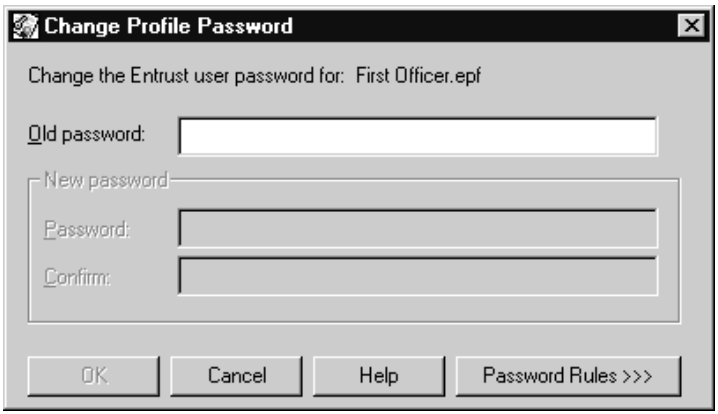
To change an administrative user profile password

1 If you are a Directory Administrator, log in to the Directory Browser. See "Logging in to the Directory Browser" on page 34.

If you have enough permissions and want to use Entrust/RA to change passwords, log in to Entrust/RA.

2 Click Operations > Change Password.

The Change Profile Password dialog box appears.



Note: Click *Password Rules* in the Change Profile Password dialog box to see the criteria your password must meet to make sure it's secure.

3. Enter your old password in the *Old password* field.
4. Under *New password*, enter a new password in the *Password* field.
5. Enter the same password again in the *Confirm* field and click *OK*.

   A dialog box appears confirming that the password was changed successfully.
6. Click *OK*.

   You have now changed your administrative user profile password.

Changing the CA Directory Access password

The CA uses the CA Directory Access password to authenticate itself to the Directory and write certificates and CRLs to it.

Only Master Users can change the CA Directory Access password. You should change your CA Directory Access password if you suspect it has been compromised. You should change your CA Directory Access password periodically.

The CA Directory Access password must be changed in two places, first in the Directory and then in Entrust:

1. To change the CA Directory Access password in the Directory, use one of your Directory tools. Consult the documentation for your Directory to find out how to do this.
2. To change the CA Directory Access password in Entrust, use Entrust/Authority Master Control. The procedure that follows describes how to do this.

To change the CA Directory Access password in Entrust

1. Log in to Entrust/Authority Master Control as a Master User (for details, see the "Setting up the Entrust/PKI environment" chapter in *Using Entrust/PKI 6.0 on Windows*).

   The Entrust/Authority Master Control window appears.
2. Click User > Change Password.

The Change Password dialog box appears.

3 In the list, click *CA Directory Access*.

In the *New password* field, enter your new CA Directory Access password.

4 In the *Confirm new password* field, enter the same CA Directory Access password again and click *OK*.

A dialog box appears stating that you are about to change the CA Directory Access password.

5 Enter your Master User password and click *OK*.

A dialog box appears confirming that the password was changed successfully.

6 Click *OK*.

Restart the Entrust/Authority service. This ensures that the new password takes effect.

You have now changed the CA Directory Access password in Entrust.

Changing the Entrust Directory Administrator password

Entrust/RA uses the Entrust Directory Administrator password to authenticate itself to the Directory to perform such tasks as adding or deleting users.

Security Officers can change the Entrust Directory Administrator password.

The Entrust Directory Administrator password must be changed in two places, first in the Directory and then in Entrust:

1. To change the Entrust Directory Administrator password in the Directory, use one of your Directory tools. Consult the documentation for your Directory to find out how to do this.
2. To change the Entrust Directory Administrator password in Entrust, use the Directory Browser. The procedure that follows describes how to do this.

To change the Entrust Directory Administrator password in Entrust

1. Log in to the Directory Browser. See "To log in to the Directory Browser" on page 34.
2. Click Operations > Change Directory Password.

   The Change Directory Password dialog box appears.

3. Enter the new password in the *Password* field.
4. Re-enter the same password in the *Confirm* field and click *OK*.

   A dialog box appears confirming that the password was changed successfully.
5. Click *OK*.

   You have now changed the Entrust Directory Administrator password in Entrust.

Changing password length

In Entrust/PKI 6.0, you can change the range for the password length setting to include a minimum of 6 or 7 characters. The default range for this setting is 8 to 20 characters. The minimum length must be at least 6 characters.

Note: Passwords with less than 8 characters are not supported in pre-4.0 clients. Entrust/Client 3.0 enforces the 8-character minimum.

To change the minimum password length

1. To change the range for password length setting to include a shorter minimum, you need to customize the policy certificate. To customize certificates, you edit the master.certspec file. Before you can edit this file, you need to export it. In Entrust/RA, click File > Certificate Definitions > Export.

Attention: Entrust Technologies recommends a minimum password length of 8 characters. Shorter passwords may significantly weaken end user security in certain environments. For further information on suitable environments for this setting, contact Entrust Support.

2. To locate this setting in the master.certspec file, search for the following occurrence of "password_length" in the "Password Rules Variables" section of the master.certspec file:

```
password_length=Integer,Password length (characters):,Minimum number of
characters required for passwords.,Range,8,20
```

3. Edit the value of the Range.

   For example, to set the minimum password length to 6 characters, revise as follows:

```
password_length=Integer,Password length (characters):,Minimum number of
characters required for passwords.,Range,6,20
```

4. Save your changes and import the revised master.certspec file. In Entrust/RA, click File > Certificate Definitions > Import.
5. To test your change, go back to Entrust/RA and try to change the policy setting for *Password length (characters).* Double-click *Security Policy* in the tree view, and then double-click *User Policies*. The available policy certificates appear below *User Policies.*

   If you increase the minimum password length to more than 8 characters, review your existing *User Policies*.

   For more information about flexible certificates and the master.certspec file, see *Administering Entrust/PKI 6.0 on Windows.*

Changing the number of authorizations for sensitive operations

In Entrust/PKI, certain operations require the authorization of one or more Security Officers or Administrators with sufficient permissions to perform the operations. Such operations are called sensitive operations, and the number of authorizations required for them is set as part of creating roles (see "Creating a role" on page 161).

For example, cross-certifying with other CAs is considered a sensitive Security Officer operation, and re-encrypting the database is considered a sensitive Master User task. See the "Setting up administrative users" chapter in *Using Entrust/PKI 6.0 on Windows* for details on how to authorize sensitive operations.

This section describes how to make the following changes to the number of authorizations required for sensitive operations:

- "Changing the number of Master User authorizations" on page 57
- "Resetting Security Officer authorizations to one" on page 59

Changing the number of Master User authorizations

During the initialization of Entrust/PKI, three Master Users must choose, enter, and confirm their passwords. This gives you the option of having two Master User authorizations for sensitive operations.

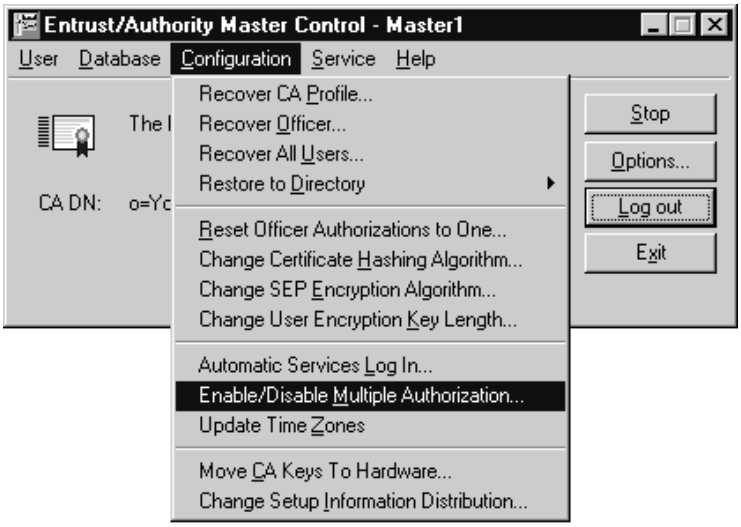
When Entrust/PKI is first installed, sensitive operations in Entrust/Authority Master Control require the authorization of only one Master User. However, if you enable multiple Master User authorizations, sensitive operations will require two Master User authorizations.

Attention: If multiple Master User authorizations are enabled, two Master Users are needed to change the password of the third. Losing or forgetting these two Master User passwords will make your Entrust/PKI system permanently and irrevocably inaccessible.

To enable multiple Master User authorization

1. Log in to Entrust/Authority Master Control as a Master User (for details, see the "Setting up the Entrust/PKI environment" chapter in *Using Entrust/PKI 6.0 on Windows*).

2 Click Configuration > Enable/Disable Multiple Authorization.



A dialog box appears stating that you are about to enable multiple Master User authorizations and that sensitive operations will need to be authorized by two Master Users.

3 Enter your Master User password and click *OK*.

You have now enabled multiple Master User authorization.

To disable multiple Master User authorization

1 Log in to Entrust/Authority Master Control as a Master User (for details, see the "Setting up the Entrust/PKI environment" chapter in *Using Entrust/PKI 6.0 on Windows*).

2 Click Configuration > Enable/Disable Multiple Authorization.

A dialog box appears stating that you are about to disable multiple Master User authorizations and that sensitive operations will need to be authorized by only one Master User.

3 You and another Master User must enter your passwords to authorize this operation. When this is done, click *OK*.

You have now disabled multiple Master User authorization.

Resetting Security Officer authorizations to one

Certain Security Officer operations in Entrust/RA require multiple authorizations. Sometimes, however, the Security Officers required to authorize an operation are not present, or several may forget their passwords at the same time. In such cases, you must reset the number of Security Officer authorizations to one so that one Security Officer can recover the others. (You can also recover each Security Officer in turn as described in "Setting up Security Officers for key recovery" on page 45.)

To reset Security Officer authorizations to one

1. Log in to Entrust/Authority Master Control as a Master User (for details, see the "Setting up the Entrust/PKI environment" chapter in *Using Entrust/PKI 6.0 on Windows*).
2. Click Configuration > Reset Officer Authorizations to One.

   A dialog box appears stating that you are about to reset to one the number of Security Officer authorizations required for sensitive operations.
3. Enter your Master User password and click *OK*.

   You have now reset to one the number of Security Officer authorizations required for sensitive operations.

Note: Security Officers can subsequently increase the number of required authorizations. For information on how to do this, see "Creating a role" on page 161.

Setting a new user limit

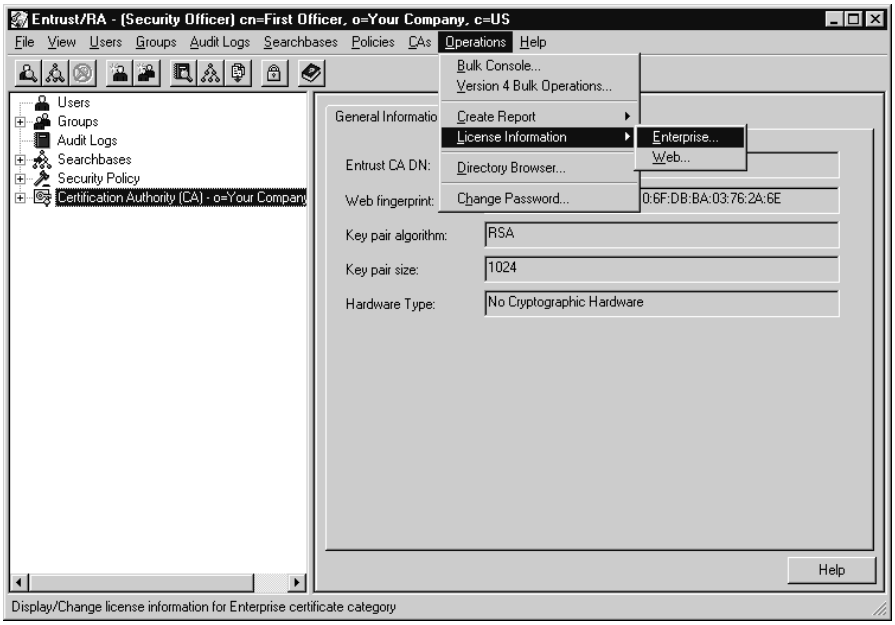
If you need to increase the total allowable number of users in your CA domain, you must first purchase additional licenses (Enterprise or Web) from your Entrust Technologies sales representative. You will then receive a new license card that includes your serial number, new user limit, and an authorization code. This section describes how to enter the license information to increase your user limits.

To set new user limits

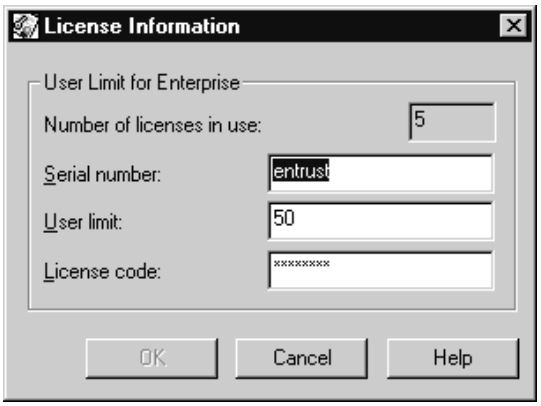
1. Log in to Entrust/RA as an administrative user (for details, see the "Setting up the Certification Authority (CA)" chapter in *Using Entrust/PKI 6.0 on Windows*).

   Entrust/RA appears.

2. To set a new user limit for Enterprise licenses, click Operations > License Information > Enterprise.



The License Information dialog box appears showing the number of licenses currently in use and your present user limit. This example shows the information about the Enterprise license.



3 Enter the information from your new license card in the *Serial number*, *User limit* and *License code* fields and click *OK*.

The Authorization Required dialog box appears.

4 Authorize the operation and click *OK*.

A dialog box appears confirming the change to the user limit.

5 Click *OK*.

6 If applicable, repeat Step 2 through Step 5 for the Web license category. The only difference is that for the Web license, the limit concerns the number of Web certificates issued, not the number of users.

7 Stop and restart the Entrust/Authority service. See "Setting up the Entrust/PKI Environment" in *Using Entrust/PKI 6.0 on Windows* for more information.

You have now set new user limits.

Setting Entrust/RA preferences

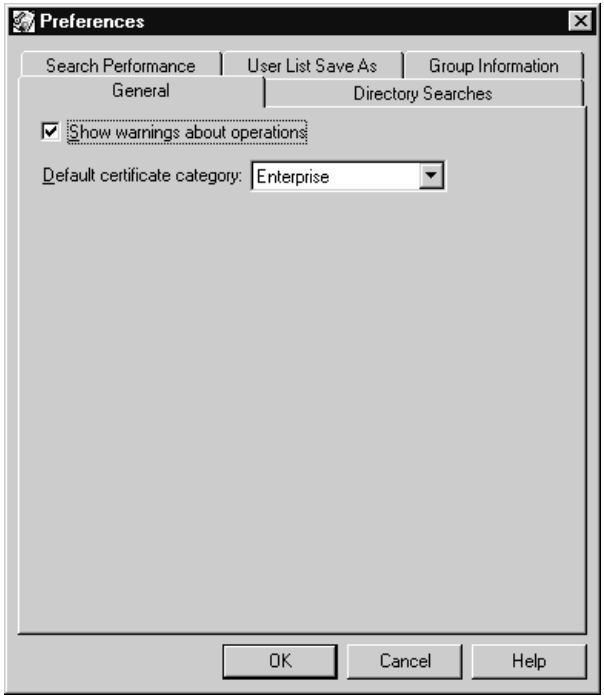
By choosing Preferences from the Entrust/RA File menu, you can set the following general Entrust/RA options:

- whether warning dialog boxes appear when you perform operations such as changing user types (see "To display warnings about operations" on page 62)
- what the default certificate category is (see "To specify a default certificate category" on page 63)
- how many users to retrieve during Directory searches (see "To specify how many users to retrieve in a Directory search" on page 64)
- what the default searchbase is (see "To specify a default searchbase" on page 65)
- how to modify search performance (see "To modify search performance" on page 66)
- how to specify information to save from a user list (see "To specify information to save from a user list" on page 68)
- how to modify the default view for displaying group membership (see "To modify the default view for displaying group membership" on page 69)

To display warnings about operations

1. Log in to Entrust/RA as a Security Officer (for details, see the "Setting up administrative users" chapter in *Using Entrust/PKI 6.0 on Windows*).
2. Click File > Preferences.

The Preferences dialog box appears, with the *General* property page visible.



3 Ensure *Show warnings about operations* is selected and click *OK*.

If you don't want such warnings to appear, clear *Show warnings about operations* and click *OK*.

Note: It's a good idea to leave *Show warnings about operations* selected unless you are very familiar with Entrust/PKI operations.

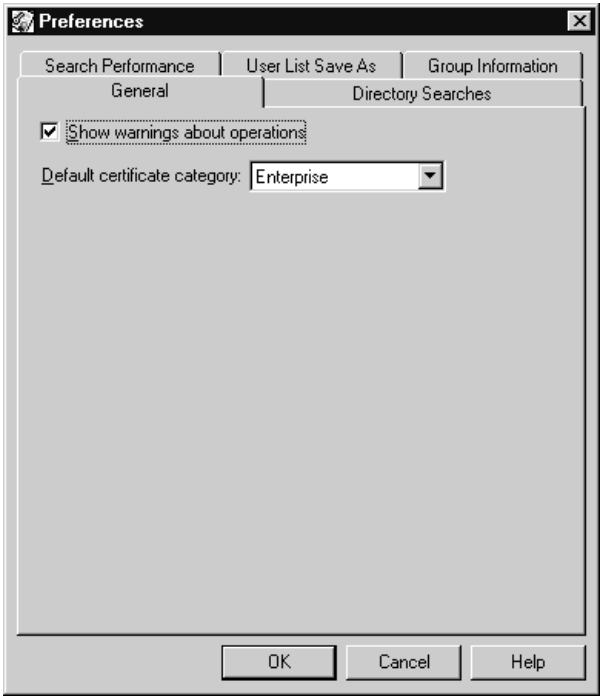
You have now set Entrust/RA to display warnings about operations.

To specify a default certificate category

1 Log in to Entrust/RA as a Security Officer (for details, see the "Setting up administrative users" chapter in *Using Entrust/PKI 6.0 on Windows*).

2 Click File > Preferences.

The Preferences dialog box appears, with the *General* property page visible.



3 In the *Default certificate category* drop-down list, choose a certificate category and click *OK*. The certificate category you choose will be the default category you use when creating users. For example, if you specify the Enterprise certificate category, this will be the default category when you create users.

If you did not purchase a Web license, *Enterprise* will be the only certificate category in the list.

You have now specified a default certificate category.

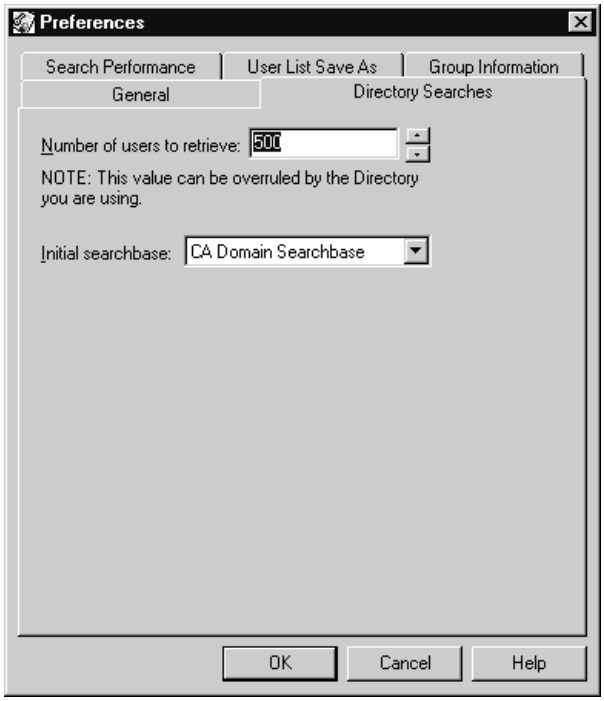
To specify how many users to retrieve in a Directory search

1 Log in to Entrust/RA as a Security Officer (for details, see the "Setting up administrative users" chapter in *Using Entrust/PKI 6.0 on Windows*).

2 Click File > Preferences.

The Preferences dialog box appears, with the *General* property page visible.

3 Click the *Directory Searches* tab.



4 Choose the maximum number of users to retrieve by either typing a value in the *Number of users to retrieve* field or by clicking the arrow buttons. Click *OK* when you're finished.

The larger the number you enter, the longer the wait to see search results if you select broad search parameters.

Note: The default maximum number of users to retrieve in a Directory search is 300, but the range is from 1 to 500. However, the number you specify here may be overruled by the Directory you are using if it has a lower retrieval limit.

You have now specified how many users to retrieve in a Directory search.

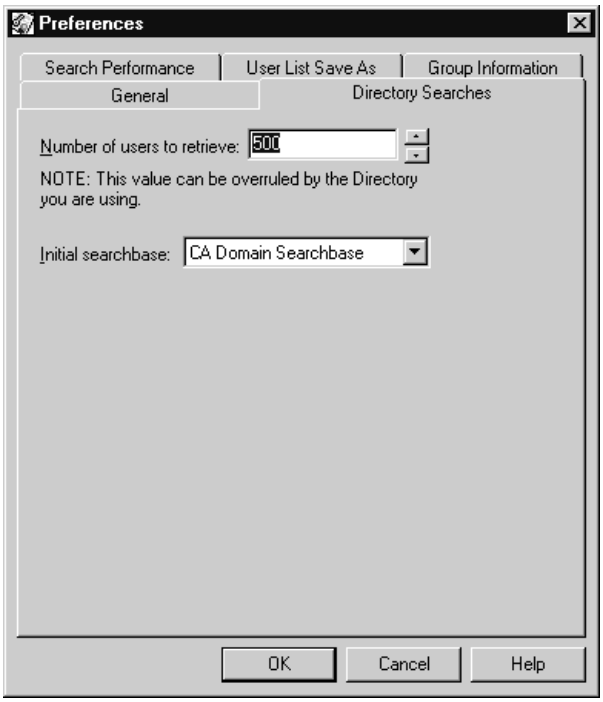
To specify a default searchbase

1 Log in to Entrust/RA as a Security Officer (for details, see the "Setting up administrative users" chapter in *Using Entrust/PKI 6.0 on Windows*).

2 Click File > Preferences.

The Preferences dialog box appears, with the *General* property page visible.

3 Click the *Directory Searches* tab.



4 Choose a searchbase in the *Initial searchbase* drop-down list (if you have other searchbases defined; otherwise, your searchbase is your CA) and click *OK*.

The initial searchbase is the default searchbase that will appear in Entrust/RA and Entrust/Entelligence when administrative users search for entries in the Directory.

You have now specified a default searchbase.

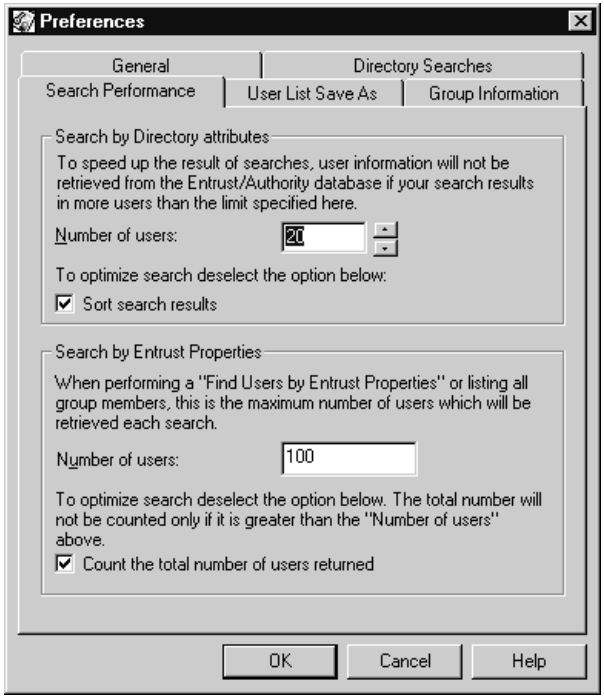
To modify search performance

1 Log in to Entrust/RA as a Security Officer (for details, see the "Setting up administrative users" chapter in *Using Entrust/PKI 6.0 on Windows*).

2 Click File > Preferences.

The Preferences dialog box appears, with the *General* property page visible.

3 Click the *Search Performance* tab.



4 In the *Number of users* field for *Search by Directory attributes*, enter the maximum number of users for which you want to immediately retrieve database information.

Because searching the Entrust/Authority database takes time, limiting the retrieval of database information to a specified number of users will help speed up your search. If your search finds more users than the number specified, user information won't be retrieved from the Entrust/Authority database immediately. Rather, it will be retrieved when you click on a user in the tree view.

For example, if the entry in the *Number of users* field is 20 and your search results in 100 users, all users found will immediately appear in the tree view, but database information will be retrieved only when you click on a user in the tree view.

5 In the *Number of users* field for *Search by Entrust Properties*, enter the maximum number of users for which database information will be retrieved.

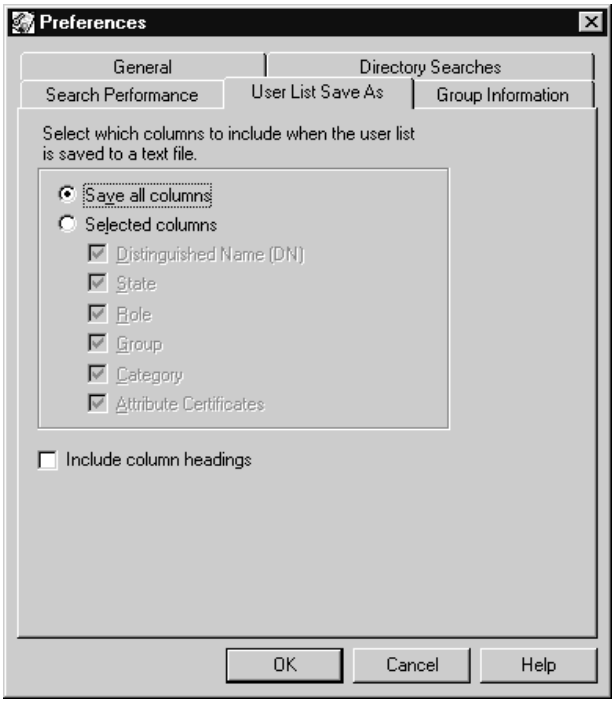
6 Click *OK*.

You have now modified search performance.

To specify information to save from a user list

1. Log in to Entrust/RA as a Security Officer (for details, see the "Setting up administrative users" chapter in *Using Entrust/PKI 6.0 on Windows*).
2. Click File > Preferences.

   The Preferences dialog box appears, with the *General* property page visible.
3. Click the *User List Save As* tab.
4. The User List Save As property page appears.



5. To save the entire user list to a text file, click *Save all columns* (the default choice).

   If you want to save information from specific columns only, click *Selected columns* and clear the check boxes of those columns you don't want to save to a text file. You can save information from the following columns in a user list to a text file:

   - Distinguished Name (DN)
   - State
   - Role
   - Group
   - Category

- Attribute Certificates

6 Select *Include column headings* if you want column headings included at the top of the text file. Saving the column headings along with the user list may make your text file easier to read later.

7 Click *OK*.

You have now specified the user list information to be saved to a text file.

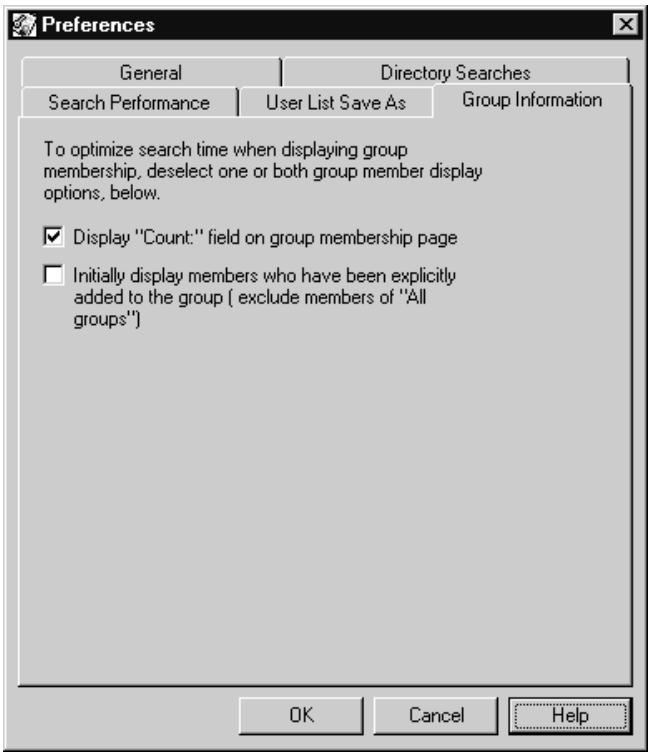
To modify the default view for displaying group membership

1 Log in to Entrust/RA as a Security Officer (for details, see the "Setting up administrative users" chapter in *Using Entrust/PKI 6.0 on Windows*).

2 Click File > Preferences.

The Preferences dialog box appears, with the *General* property page visible.

3 Click the *Group Information* tab.



The two settings on this property page are used to display or hide the *Count* and *Number you can view* fields on the *Group* property page of Entrust/RA (look up "viewing group properties" in *Using Entrust/PKI 6.0 on Windows*). If you have large numbers of users in your groups, you may experience slower access times

when displaying the *Group* property page. By turning off either or both of the *Count* and *Number you can view* fields, you can speed up the time it takes you to access the *Group* property page.

4 Select or clear the settings on the *Group Information* property page, as required.

You have now modified the default view for displaying group membership.

Modifying Entrust/RA to improve database searches

You may want to set up the following Entrust/RA preferences to improve database searches. See "Customizing the entmgr.ini file" on page 527 for ways to control user searching in large databases.

To modify Entrust/RA preferences to improve database searches

1. Log in to Entrust/RA as a Security Officer.
2. Click File > Preferences.
3. Click the *Search Performance* tab.
4. Clear *Count the total number of users returned*.
5. Increase the *Number of users* if you want more results returned in one page.
6. Click *OK*.
7. Click the *Group Information* tab.
8. Clear *Display "Count:" field on group membership page*, and click *OK*.

   You have now set up the Entrust/RA preferences for searching large databases.

Updating the time zone for database reporting

If you use third-party tools for database reporting, you need to update your time zone in Entrust/Authority

- if you accidentally installed Entrust/Authority before setting the correct time zone on a new server
- if you physically move the Entrust/Authority server to a different time zone

To update the time zone for database reporting

1. Stop all of the following services or sessions: the Entrust/Authority service, your Directory, Entrust/Authority Master Control, and any Entrust/RA sessions that are running. In addition, make sure that no application is accessing the Entrust/Authority service.

   Stopping these services and sessions ensures that all log entries after the time zone is changed refer to the new time zone.

2. Set the correct time zone on the Windows NT computer hosting Entrust/Authority.

3. Restart the Entrust/Authority service and your Directory.

   Ensure the Informix services have started (in the Windows taskbar, click Start > Settings > Control Panel, double-click the *Services* icon, and verify that the Informix services have started).

Note: In a Windows 2000 environment, double-click *Administrative Tools* before you double-click the *Services* icon.

---

4. Log in to Entrust/Authority Master Control as a Master User (for details, see the "Setting up the Entrust/PKI environment" chapter in *Using Entrust/PKI 6.0 on Windows*).

5. Click Configuration > Update Time Zones.

   A message appears, advising you to reboot your machine if you have not already done so.

6. Click *OK* if you have rebooted, *Cancel* otherwise.

   You have now updated the time zone for database reporting.

Changing algorithms

Entrust/Authority uses various cryptographic algorithms to secure the operations it performs. For example, it uses algorithms to encrypt data and to create digital signatures. You choose these and other algorithms when you install Entrust/PKI.

This section describes how to change the settings for certain algorithms:

- the hashing algorithm that Entrust/Authority uses when signing certificates
- the protocol encryption algorithm
- the encryption key length used by end users

Changing the certificate hashing algorithm

The certificate hashing algorithm is the algorithm Entrust/Authority uses for hashing as part of the process of creating certificate signatures. Algorithm choices are SHA-1 (the default) and MD5, with organization policy usually determining which one is chosen.

Very few conditions warrant a change in algorithm. You would make such a change if organization policy regarding this algorithm changes.

To change the certificate hashing algorithm

1. Log in to Entrust/Authority Master Control as a Master User (for details, see the "Setting up the Entrust/PKI environment" chapter in *Using Entrust/PKI 6.0 on Windows*).
2. Click Configuration > Change Certificate Hashing Algorithm.

   A dialog box appears prompting you to authorize the operation.
3. Enter your Master User password and click *OK*.

   You have now changed the certificate hashing algorithm. Repeat this procedure to return to SHA-1.

Changing the protocol encryption algorithm

Entrust's protocol encryption algorithm secures information about user encryption keys exchanged by Entrust-Ready applications and Entrust/PKI during key update and user initialization.

You have two choices for the protocol encryption algorithm: CAST-128 and Triple DES. Organization policy usually determines which algorithm is chosen.

You would make such a change if organization policy regarding this algorithm changes.

To change the protocol encryption algorithm

1. Log in to Entrust/Authority Master Control as a Master User (for details, see the "Setting up the Entrust/PKI environment" chapter in *Using Entrust/PKI 6.0 on Windows*).
2. Click Configuration > Change SEP Encryption Algorithm.

   The Select Algorithm dialog box appears.
3. Click the algorithm you want to use and click *OK*.

   The default is CAST-128. Triple DES is slower than CAST-128.

   A dialog box appears stating that you are about to change the algorithm used to encrypt protocol messages.
4. Enter your Master User password and click *OK*.

   You have now changed the protocol encryption algorithm.

Changing the length of the user encryption key

The user encryption key is used to create users' encryption key pairs. The options available are RSA-1024 and RSA-2048. The default on installation is RSA-1024.

If your company decides the default is no longer strong enough, you can change to the RSA-2048 algorithm. This change could, however, slow key generation considerably if your users have slow systems. It is unlikely that you would want to change from RSA-2048 to RSA-1024.

To change the length of the user encryption key

1. Log in to Entrust/Authority Master Control as a Master User (for details, see the "Setting up the Entrust/PKI environment" chapter in *Using Entrust/PKI 6.0 on Windows*).
2. Click Configuration > Change User Encryption Key Length.

   A dialog box appears asking you to authorize the operation.
3. Enter your Master User password and click *OK*.

   You have now changed the length of the user encryption key.

Assigning multiple processes to a subsystem

The Entrust/Authority service consists of a number of subsystems, and you can dedicate more than one process to three of them: PKIX-Certificate Management Protocol, Entrust Secure Exchange Protocol, and Administration. When you install Entrust/PKI, two processes are assigned to PKIX-Certificate Management Protocol, two to Entrust Secure Exchange Protocol, and four to Administration.

These default settings should suit most Entrust/PKI installations. But if your current settings are a bottleneck or you need to manage large numbers of Entrust users connecting at the same time, you may need to assign more processes to a subsystem.

You can assign up to nine processes using Entrust/Authority Master Control. You can assign additional processes using Entrust Master Control Command Shell (entsh) and the variable AshNumProcs in the [Login] section of the entmgr.ini file.

The following section presents some guidelines on assigning processes.

Guidelines for assigning processes

Changing the number of processes assigned to a subsystem is a serious matter. If the current setting is optimal, you could unintentionally degrade Entrust/PKI performance by changing the number of processes assigned to a subsystem. You should change your current setting only if the number of processes assigned is a bottleneck in your Entrust/PKI.

To find out whether current settings for the number of processes are the problem, systematically consider the alternatives:

- Verify that the systems on which Entrust/PKI is running are optimally configured.
- Check CPU utilization for clues about system bottlenecks. For example, if CPU utilization is high, the CA is being saturated—you need to add more or faster processors, not more processes.

If Entrust/PKI performance still isn't satisfactory, the number of processes is probably the problem. Monitoring system performance can help you determine which subsystem to add processes to. For information on how to do this, see "Monitoring Entrust/PKI" on page 432.

Before increasing the number of processes you assign to a subsystem, check that your Entrust/Authority server can handle the additional traffic. Also verify that the server has enough RAM to support more processes. The default settings for subsystem processes are associated with a recommended memory requirement of 128 Mbytes of RAM.

Note: Dedicating more processes than necessary to a subsystem is wasteful, since additional processes use memory even when they aren't in use. For example, a process assigned to the Administration subsystem uses memory even when Entrust/RA is idle.

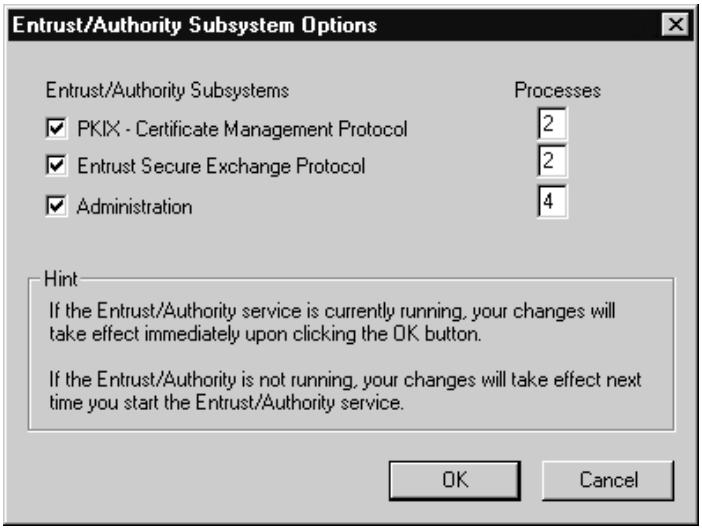
To change the number of processes assigned to a subsystem

1. Log in to Entrust/Authority Master Control (for details, see the "Setting up the Entrust/PKI environment" chapter in *Using Entrust/PKI 6.0 on Windows*).

   Entrust/Authority Master Control appears.

2. Click *Options*.

   The Entrust/Authority Subsystem Options dialog box appears.



3. In the *Processes* field that corresponds to the subsystem you want to change, type the number of processes you want to dedicate to the subsystem. You can assign a maximum of 9 processes to each subsystem.

   See the "Setting up the Entrust/PKI environment" chapter in *Using Entrust/PKI 6.0 on Windows* for more information on subsystems.

4. Click *OK*.

   You have now changed the number of processes assigned to a subsystem.

   You can also change the number of processes assigned to subsystems in the entmgr.ini file, or by using the Entrust/Authority command-line interface (see "To assign a specific number of processes to an Entrust/Authority subsystem" on page 359).

Using a Combined CRL

Certificate revocation lists (CRLs) identify certificates that should not be trusted, because they are no longer considered valid by the CA that issued them. When an application encrypts information for a recipient or verifies a signature, it checks an appropriate CRL to confirm that the certificate in question has not been revoked.

To allow for system scalability and more efficient processing, Entrust/Authority maintains standardized, partitioned CRLs at unique "distribution points" in the Directory. Each certificate contains a pointer to one or more CRL distribution points where applications can find the CRL for the certificate in question. For more information about CRLs and CRL distribution points, see the "About Entrust/PKI" chapter in *Using Entrust/PKI 6.0 on Windows*.

The use of partitioned CRLs at unique distribution points is an optional part of two standards:

- ITU-T Recommendation X.509 (1997)/ISO/IEC 9594-8:1997, *Information Technology - Open Systems Interconnection - The Directory: Authentication Framework*. This standard is jointly published and maintained by the International Telecommunication Union (ITU-T) and the International Organization for Standardization/International Electrotechnical Commission (ISO/IEC).
- IETF Request for Comments 2459, *Internet X.509 Public Key Infrastructure Certificate and CRL Profile*. This standard is published and maintained by the Internet Engineering Task Force (IETF).

Because the use of partitioned CRLs at unique distribution points is an optional part of these standards, it is not used by all clients that are compliant with the standards (for example, Windows 2000 clients). Because Entrust Technologies supports open standards-compliant systems, Entrust/Authority also supports clients that meet the minimum requirements of these standards. Entrust/Authority supports these clients by issuing a single CRL (referred to as a "Combined CRL").

Entrust/Authority creates and maintains the Combined CRL in the Entrust/Authority database. The Combined CRL includes all certificates issued by that CA that have been subsequently revoked. At regular intervals, Entrust/Authority publishes the Combined CRL to the CA entry in the Directory, where it can be accessed by clients that don't support the use of partitioned CRLs at unique distribution points. Entrust/Authority can also publish the Combined CRL to a file (see "Publishing the Combined CRL to a file" on page 80).

Note: If you are interoperating with Microsoft CAPI-enabled applications, there are important considerations to be aware of in terms of using Combined CRLs. For more information, check the Entrust/PKI 6.0 Release Notes for a listing of related white papers and where to find them.

Turning the Combined CRL on and off

Entrust/Authority does not create, maintain, and publish the Combined CRL by default. If you want to use a Combined CRL, you must turn it on.

You turn on the Combined CRL feature either before or after first-time initialization.

To turn on the Combined CRL before first-time initialization

1. Open the entmgr.ini file in a text editor.

   By default, the entmgr.ini file is located in the C:\authdata\manager folder, where "C:\" represents the drive where Entrust/Authority is installed.
2. Locate the [Advanced Settings] section.
3. Locate the UseCombinedCRL entry in this section.
4. Change the value of this entry to "1" as follows:

   ```
   UseCombinedCRL=1
   ```
5. Save and close the entmgr.ini file.

   You have now turned on the Combined CRL.

Turning on the Combined CRL after first-time initialization

During first-time initialization, Entrust/Authority removes the UseCombinedCRL entry from the entmgr.ini file and stores its setting in the Entrust/Authority database. Because this entry no longer exists in the entmgr.ini file, you must use another method to turn the Combined CRL on after first-time initialization.

After first-time initialization, you turn the Combined CRL on using Entrust Master Control Command Shell. The Entrust/Authority service must be stopped. See "Turning the use of Combined CRL on and off" on page 410. For information about stopping the Entrust/Authority service, see the "Setting up the Entrust/PKI environment" chapter in *Using Entrust/PKI 6.0 on Windows*.

Turning off the Combined CRL

By default, the Combined CRL feature is turned off. You only need to turn this feature off after you have turned it on. When you turn off the Combined CRL, Entrust/Authority stops maintaining it in the database and publishing it to the

Directory. Entrust/Authority does not remove the existing Combined CRL from the Directory.

To turn the Combined CRL off, use Entrust Master Control Command Shell (see "Turning the use of Combined CRL on and off" on page 410). The Entrust/Authority service must be stopped. For information about stopping the Entrust/Authority service, see the "Setting up the Entrust/PKI environment" chapter in *Using Entrust/PKI 6.0 on Windows*.

Changing the Combined CRL lifetime

The "lifetime" of the Combined CRL determines how often Entrust/Authority publishes a new Combined CRL to the Directory. By default, the Combined CRL has a lifetime of 24 hours.

You can increase and decrease the lifetime of the Combined CRL. For example, the Combined CRL can become large, which may increase how long it takes for clients to retrieve it from the Directory. You may want to decrease how often the Combined CRL is published to minimize the number of times that clients need to retrieve it from the Directory.

You change the lifetime of the Combined CRL either before or after first-time initialization.

To change the Combined CRL lifetime before first-time initialization

1. Open the entmgr.ini file in a text editor.

   By default, the entmgr.ini file is located in the C:\authdata\manager folder, where "C:\" represents the drive where Entrust/Authority is installed.

2. Locate the [Advanced Settings] section.
3. Locate the CombinedCRLLifetime entry in this section.
4. Change the value of this entry to the new lifetime (in hours). For example, the following entry specifies a lifetime of 36 hours:

   ```
   CombinedCRLLifetime=36
   ```

   You can set the value of this entry from 4 to 336 hours (two weeks). You can also set the value to "0" to specify that the lifetime of the Combined CRL is the same as the Security Policy lifetime (see the "Setting up the Certification Authority (CA)" chapter in *Using Entrust/PKI 6.0 on Windows*).

5. Save and close the entmgr.ini file.

   You have now changed the lifetime of the Combined CRL.

Changing the Combined CRL lifetime after first-time initialization

During first-time initialization, Entrust/Authority removes the CombinedCRLLifetime entry from the entmgr.ini file and stores its setting in the Entrust/Authority database. Because this entry no longer exists in the entmgr.ini file, you must use another method to change the Combined CRL lifetime after first-time initialization.

To change the Combined CRL lifetime after first-time initialization, use Entrust Master Control Command Shell (see "Changing the Combined CRL lifetime" on page 411). The Entrust/Authority service must be stopped. For information about stopping the Entrust/Authority service, see the "Setting up the Entrust/PKI environment" chapter in *Using Entrust/PKI 6.0 on Windows*.

Publishing the Combined CRL to a file

Entrust/Authority creates and maintains the Combined CRL in the Entrust/Authority database and publishes the Combined CRL to the CA entry in the Directory. Entrust/Authority can also publish the Combined CRL to a file. For example, you may want to make the Combined CRL information available from a Web site.

To publish the Combined CRL to a file

1. Open the entmgr.ini file in a text editor.

   By default, the entmgr.ini file is located in the C:\authdata\manager folder, where "C:\" represents the drive where Entrust/Authority is installed.

2. Locate the [CRL] section.

   If this section does not exist, add it after any other section in the entmgr.ini file.

3. Locate the CombinedCRLFile entry in this section.

   If the CombinedCRLFile entry does not exist, add it after any other entry in the [CRL] section.

4. Change the path and filename. For example, the following entry specifies a file named "CRLfile.crl" in the TEMP folder on the C: drive:

   ```
   CombinedCRLFile=C:\temp\CRLfile.crl
   ```

   You can also specify the name and location using UNC (Universal Naming Convention). The syntax is

   CombinedCRLFile=*host_name**share**name_of_file*

Note: If the file does not exist, Entrust/Authority creates the file and publishes the Combined CRL to it the next time you start the Entrust/Authority service. If the file already exists, Entrust/Authority waits until the current Combined CRL lifetime expires before publishing the new version.

5 Save and close the entmgr.ini file.

Note: You must restart the Entrust/Authority service after modifying entries in the entmgr.ini file.

You have now specified a file where Entrust/Authority publishes the Combined CRL.

Limiting the size of the published Combined CRL

Depending on the Directory you are using, problems may occur if Entrust/Authority tries to publish a very large Combined CRL. You can specify that when the Combined CRL exceeds a particular "threshold" (for example, 500,000 bytes), you do not want Entrust/Authority to publish it to the Directory.

When the Combined CRL exceeds the threshold, Entrust/Authority creates an audit. This audit warns you that the Combined CRL is no longer being published to the Directory. However, Entrust/Authority continues to maintain the Combined CRL in the database. As revoked certificates expire, the size of the Combined CRL in the database decreases. When the size is once again below the threshold, Entrust/Authority begins publishing the Combined CRL to the Directory.

To limit the size of the Combined CRL

1 Open the entmgr.ini file in a text editor.

By default, the entmgr.ini file is located in the C:\authdata\manager folder, where "C:\" represents the drive where Entrust/Authority is installed.

2 Locate the [CRL] section.

If this section does not exist, add it after any other section in the entmgr.ini file.

3 Locate the CombinedCRLThreshold entry in the [CRL] section, and type the threshold value.

If the CombinedCRLThreshold entry does not exist, add it after any other entry in the [CRL] section.

For example, the following entry specifies a threshold of 500,000 bytes:

```
CombinedCRLThreshold=500000
```

4 Save and close the entmgr.ini file.

Note: You must restart the Entrust/Authority service after modifying entries in the entmgr.ini file.

You have now limited the size of the Combined CRL that can be published to the Directory.

Specifying additional CRL distribution points

To allow for system scalability and more efficient processing, Entrust/Authority maintains standardized, partitioned certificate revocation lists (CRLs) at unique "distribution points" in the Directory. Each certificate contains a pointer to one or more CRL distribution points (CDPs) where applications can find the CRL for the certificate in question. For more information about CRLs and CRL distribution points, see the "About Entrust/PKI" chapter in *Using Entrust/PKI 6.0 on Windows*.

Entrust/Authority inserts CDP pointers in certificates when issuing them. Entrust/Authority specifies CDP entries using the DirectoryName form. Because not all clients understand this form, Entrust/Authority can also add CDP entries using the Uniform Resource Locator (URL) form. When a client receives a certificate, it checks the entries to find one it can interpret. The client checks the entries in the order they are listed.

If you want Entrust/Authority to insert CDP pointers using the URL form, you must provide the required URLs.

Specifying the CDP in URL form

You define each URL-type CDP using an entry in the entmgr.ini file. You define each entry in the [CDP] section of this file. You can define as many URL entries as you want.

To specify the CDP in URL form

1. Open the entmgr.ini file in a text editor.

   By default, the entmgr.ini file is located in the C:\authdata\manager folder, where "C:\" represents the drive where Entrust/Authority is installed.

2. Locate the [CDP] section.

   If this section does not exist, add it after any other section in the entmgr.ini file.

3. Type a CDP entry.

   The CDP entry uses this format:

   ```
   number=url
   ```

   where "number" is an integer (starting with "1"), and "url" is the URL. You can specify the following tokens when specifying the URL. Entrust/Authority

substitutes the appropriate strings for the tokens when inserting the CDP pointer in a certificate.

For this token...	Entrust/Authority substitutes the...
<server>	address and port of the LDAP server, separated by a colon (for example, "47.97.234.150:400") Note that if the default port number (that is, 389) is used, Entrust/Authority does not include the colon or the port number (for example, "47.97.234.150").
<ca>	DN of the CA, in its URL-encoded form. For example, if the DN of the CA is "o=Your Company,c=US", then the URL-encoded form is "o=Your%20Company,%20c=US".
<crl>	DN of the partitioned CRL for this certificate, in its URL-encoded form. For example, if the partitioned CRL for this certificate is located at "cn=CRL1,o=Your Company,c=US", then the URL-encoded form is "cn=CRL1,%20o=Your%20Company,%20c=US".
<semi>	semicolon character Note that the ";" representation of this character is not used. Within the entmgr.ini file, this representation indicates the start of a comment.
<serial>	serial number of the certificate being created
<domain>	DNS domain name (for example, "yourcompany.com")
<fullname>	host name concatenated with domain name (for example, "myhost.yourcompany.com")

Note: The tokens are case-insensitive.

For example, you can specify two CDP entries as follows:

```
1=ldap://<server>/<ca>?certificateRevocationList?base?
objectclass=cRLDistributionPoint
2=ftp://www.yourcompany.com/somefile.crl
```

4 Save and close the entmgr.ini file.

You have now specified the CDP entry using the URL form.

Note: You must restart the Entrust/Authority service after defining entries in the [CDP] section of entmgr.ini. This allows Entrust/Authority to begin inserting the CDP entries in new certificates.

Recovering the CA profile

The CA has its own profile just as users do. This profile is used by the Administration subsystem and for moving and archiving users. The name of this profile is ca.epf, and the profile is created during the initialization of Entrust/Authority. Two key pairs are associated with the CA profile:

- a signing key pair, called the CA User Signing Key Pair
- an encryption key pair, called the CA User Encryption Key Pair

Entrust/Authority also creates a random password for the CA profile during first-time initialization and stores the password securely in the Entrust/Authority database. You must recover the CA profile if it is damaged. There should be no reason to suspect that the CA profile password has been compromised since it is known only to Entrust/Authority and is stored securely in the Entrust/Authority database.

Recovering the CA profile is different from recovering an Entrust user. When you recover a user, Entrust/Authority generates a reference number and an authorization code. This information is securely communicated to the user, and the user must enter this information in an Entrust-Ready application to complete the recovery operation. By contrast, when you recover the CA profile, Entrust/Authority automatically creates a new profile with a new signing private key and verification public key certificate and generates a new random password; this all happens with no user intervention.

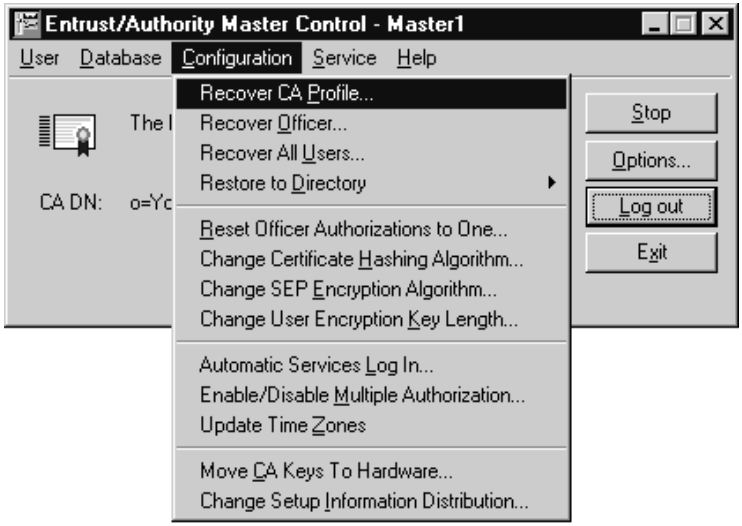
The CA profile key pair has a lifetime of 365 days and undergoes key update after 265 days. For more information on key updates, see the "About Entrust/PKI" chapter in *Using Entrust/PKI 6.0 on Windows*.

Note: If you recover the CA profile of a CA that exported its public keys to another CA, you should export the keys again.

To recover the CA profile

1. Log in to Entrust/Authority Master Control as a Master User (for details, see the "Setting up the Entrust/PKI environment" chapter in *Using Entrust/PKI 6.0 on Windows*).

2 Click Configuration > Recover CA Profile.



A dialog box appears asking if you are sure you want to recover the CA profile and advising you that there is no reason to perform this operation if the Entrust/Authority service is running now.

3 If you still wish to proceed, click *Yes*.

Once the CA profile is recovered, a dialog box appears confirming the recovery.

4 Click *OK*.

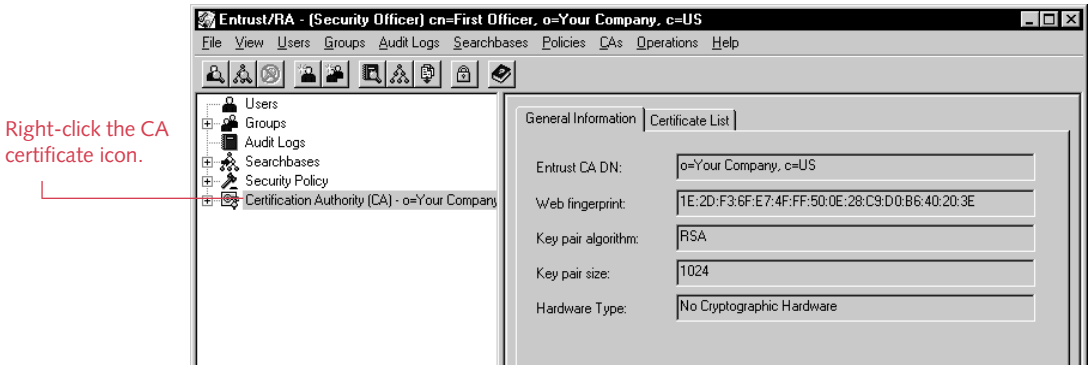
You have now recovered the CA profile.

Exporting the CA certificate

Some applications may require a CA certificate from other applications as a prerequisite for working with them. If an application you want Entrust/PKI to work with has this requirement, you can meet it by exporting a CA certificate to a file and submitting the file to the other application.

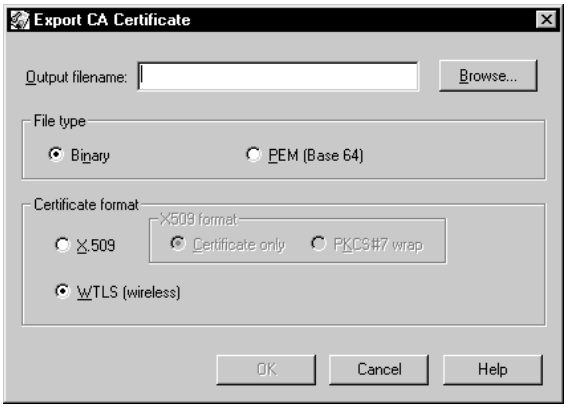
To export the CA certificate

1 In Entrust/RA, right-click the CA certificate icon in the tree view (for example, *Certification Authority (CA) - o=Your Company,c=US*).



2 Click Export CA Certificate in the pop-up menu.

3 In the Export CA Certificate dialog box, type a filename in the *Output filename* field, and click *Browse* to specify a location for the file.



4 Select either *Binary* or *PEM (Base 64)* for *File type*.

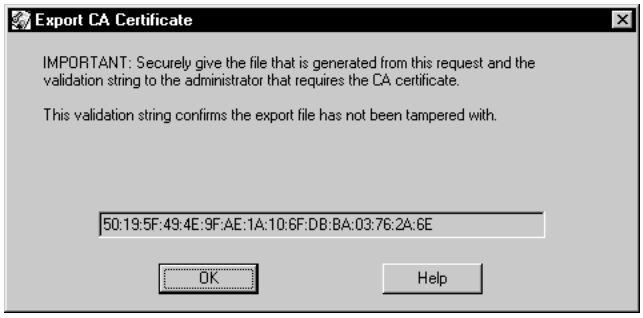
If you select *Binary*, an extension of .der is appended to the filename. If you select *PEM (Base 64)*, an extension of .pem is appended to the filename. The choice of

file formats is provided for interoperability with PKIs from other vendors. Entrust/PKI can read certificates exported in either file format.

5 Select *WTLS (wireless)* for *Certificate format* if you are exporting the CA certificate for use with WAP servers; otherwise choose *X.509*. If you are exporting the CA as an X.509 certificate, you must choose to export it either as raw certificate data (*Certificate only*), or you can export it using the PKCS #7 standard (*PKCS#7 wrap*). In a scenario where there is a hierarchy of CAs, the raw certificate data only contains information about the certificate you are exporting, whereas the PKCS #7 format includes the certificate chain, including any superior CAs, if the certificate you are exporting is from a subordinate CA.

6 Click *OK*.

When the CA certificate has been written to file, a dialog box appears displaying the validation string for the exported CA certificate. Later on, when the exported CA certificate is imported elsewhere (for example, on a WAP server, or by a Security Officer at another CA), this validation string is used for confirming that the exported CA certificate has not been altered.



7 Copy the validation string down on paper and click *OK*.

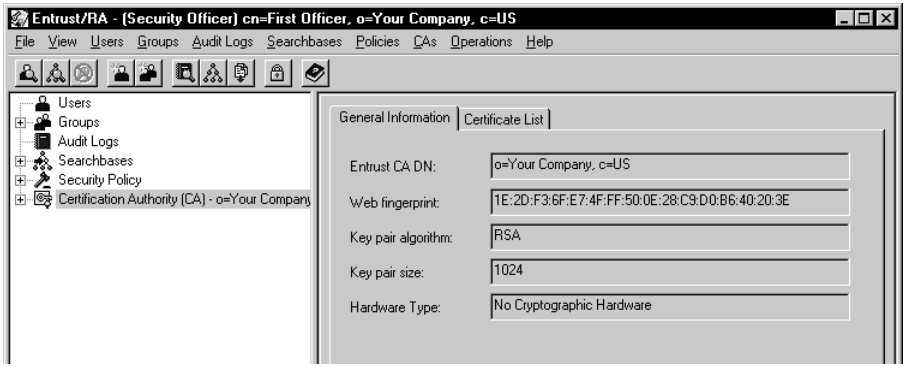
You have now exported the CA certificate.

Viewing the contents of a CA certificate

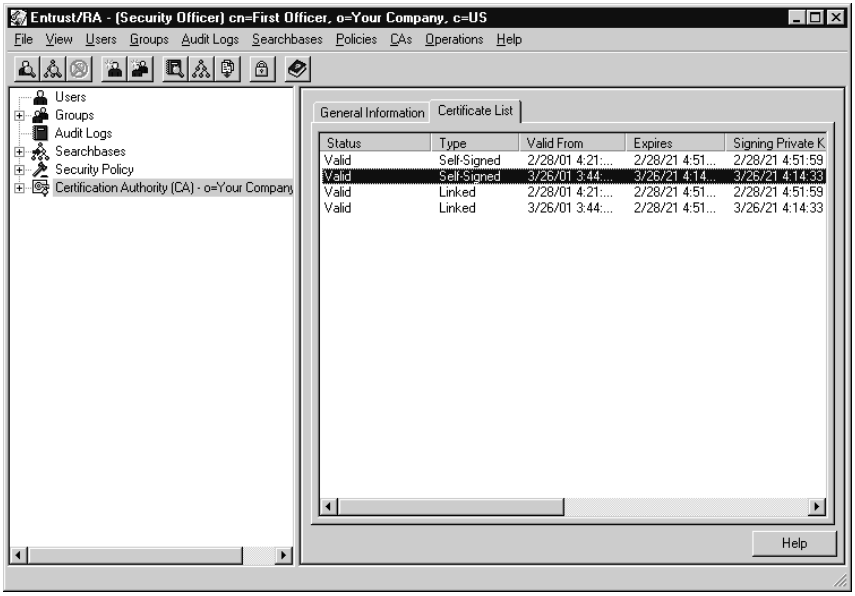
In Entrust/RA, you can view the contents of a CA certificate decoded in ASN1 format. When you view the contents of a CA certificate, you can see all the details of the actual certificate, some of which are not displayed anywhere else. For example, if you want to see all the extensions in a CA certificate, you can do so by viewing the certificate contents.

To view the contents of a CA certificate

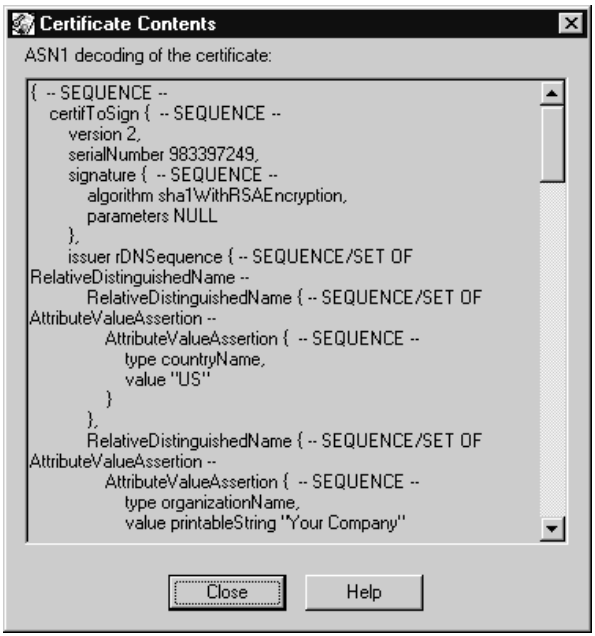
1 In Entrust/RA, click the CA certificate icon in the tree view (for example, *Certification Authority (CA) - o=Your Company,c=US*).



2 Click the *Certificate List* property page.



3 Double-click the CA certificate for which you want to view the contents, or right-click the CA certificate and click Certificate Contents in the pop-up menu. The Certificate Contents dialog box appears.



4 When you have finished viewing the contents, click *Close*.

You have now viewed the contents of a CA certificate.

Chapter 3

Using CA hardware

This chapter includes information about using CA hardware devices with Entrust/PKI 6.0.

A CA hardware device is used to store the CA key pair that is used by Entrust/PKI 6.0 to certify its users and to store a key that is used to protect the database. The CA key pair includes a signing private key and a verification public key. The CA signing private key is used to sign all certificates; the CA verification public key is used to verify users' certificates when interacting with Entrust/PKI. Storing the CA key pair on a separate, secure hardware device adds another layer of security to the Entrust/PKI model.

Note: If you plan to use a CA hardware device with Entrust/PKI 6.0, read also "Disaster recovery and high availability" on page 553.

This chapter includes the following sections:

- "Updating the CA hardware password" on page 94
- "Updating the CA key pair on hardware" on page 96
- "Moving the CA key pair to a PKCS #11 v1 hardware device" on page 97
- "Updating the CA key pair to a second hardware device" on page 98
- "Hardware-based database protection" on page 100

Updating the CA hardware password

A Master User is required to update a CA hardware password should the password change. The CA hardware password might be changed for several reasons, for example, if you add a new hardware device to the system during an upgrade using a new password.

If the CA hardware password is changed and not updated, Master Users cannot log in to Entrust/Authority Master Control and Entrust/Authority cannot perform signing operations. Consequently, the Master User must update the password in Entrust/Authority immediately after the password change occurs.

Changing the device password generates the following audit: 7649 CA Hardware password set. The password value is not revealed in the audit information.

To update the CA hardware password using Entrust/Authority Master Control

1. Log in to Entrust/Authority Master Control as a Master User.

   If the CA hardware password has been changed, a dialog box appears with the following message: "The CA hardware password is no longer valid. Please enter the valid password in the following dialog."

2. Click *OK*.

   The Hardware password dialog box appears.

3. Type the password and click *OK*.

Note: The *Hardware* field identifies the hardware device. Check this field before you enter the hardware password if Entrust/PKI 6.0 is configured to use multiple hardware devices (for example, to enable hardware-based database protection and to store the CA key pair).

The new password is validated and stored in the Entrust/Authority database.

You have now changed the CA hardware password using Entrust/Authority Master Control.

To update the CA hardware password using entsh

1. Start entsh and log in as a Master User.

   To start the Command Shell, click Start > Programs > Entrust PKI > Entrust Master Control Command Shell.

   If the CA hardware password has been changed, the following message appears:

   ```
   The CA hardware password is no longer valid. Please enter the
   valid CA hardware password:
   ```

2. Enter the CA hardware password.

   The following text appears:

   ```
   The CA hardware password has been updated. Please log in again.
   ```

   The new password is validated and stored in the Entrust/Authority database.

3. Log in to entsh.

   You have now changed the CA hardware password using entsh.

Updating the CA key pair on hardware

Entrust/PKI 6.0 lets you update the CA key pair on software or on hardware (that is, where the source and destination are the same). You can also update the CA key pair from the Entrust/Authority database to a hardware device, or from a hardware device to the Entrust/Authority database (that is, where the source and destination are different).

Attention: When you update the CA key pair, you actually create a new CA key pair. Do not update the CA key pair if Entrust users are using pre-5.0 client software. Additionally, Entrust does not support CA key pair update for subordinate CAs.

The entsh command "ca key update" performs the CA key pair update. This command displays all available devices (including software) to which you can update the CA key pair. The procedure describing the update operation is found in "Updating the CA key pair" on page 389.

Backing up and restoring the CA key pair

Depending on your CA hardware device, you should be able to backup and restore the CA key pair in the event of a hardware failure. CA hardware manufacturers provide different mechanisms for creating a backup copy of the CA key pair so that the key pair can be restored in the event of a hardware failure.

Consult the documentation provided with your CA hardware device for more information about how to back up, and later restore, the CA key pair.

Moving the CA key pair to a PKCS #11 v1 hardware device

This section describes how to move the CA key pair from software to a PKCS #11 v1 hardware device.

Note: You cannot move the CA key pair from software to a PKCS #11 v2 hardware device.

To move the CA key pair to a PKCS #11 v1 hardware device

1. Start Entrust Master Control Command Shell on your workstation (see "Starting Entrust Master Control Command Shell" on page 347).
2. Log in to Entrust/Authority Master Control as a Master User (see "Logging in to Entrust/Authority Master Control" on page 349).
3. At the prompt, enter

   ```
   ca key move
   ```

   A list of hardware devices appears.
4. Select the hardware device you want to use.

   The command shell prompts you for the device's Key Exchange Key (KEK).

   Enter the values requested.

Note: Moving the CA key pair to hardware is a sensitive operation. If Entrust/PKI 6.0 is enabled for multiple authorizations, two of the three Master Users must be present to authorize this operation.

You have now used Entrust Master Control Command Shell to move CA key pair to hardware.

Updating the CA key pair to a second hardware device

You can update the CA key pair from one CA hardware device to another. When you do so, you create a second, updated CA key pair. Note that the CA key pair now resides on both the original and the secondary device.

Note: Updating the CA key pair to hardware is a sensitive operation. If Entrust/PKI 6.0 is enabled for multiple authorizations, two of the three Master Users must be present to authorize this operation.

This procedure is not restricted to devices sharing a common PKCS #11 device version; you can update the CA key pair from a PKCS #11 version 1 device to a PKCS #11 version 2 device, or vice versa. In either case, the process involves updating the key to software first, as described in the procedure below.

Attention: When you update the CA key pair, you actually create a new CA key pair. Do not update the CA key pair if Entrust users are using pre-5.0 client software. Additionally, Entrust does not support CA key pair update for subordinate CAs.

To update the CA key pair to a second hardware device

1. If you have not already done so, review the guidelines for securing your server found in *Installing Entrust/PKI 6.0 on Windows*.
2. Verify that your installation of Entrust/PKI meets the following requirements:
   - you are running a root CA (with the CA key pair in a hardware device); subordinate CAs cannot update the CA key pair
   - you are using Entrust-Ready applications compatible with release 5.0 (or later)

   If you are running a root CA, but you are using Entrust-Ready applications compatible with releases before 5.0 (or non-Entrust applications that don't support key update)

   - upgrade to Entrust-Ready applications compatible with release 5.0 (or later) before proceeding to the next step.

     OR
   - use the command "ca keyrecover-all" in the command-line interface to Entrust/Authority (entsh) so that all users receive new certificates signed with the new CA key pair. When you recover all users, you must distribute new activation codes to every single user.

Attention: Consult a First Officer or Master User before you execute the "ca keyrecover-all" command. If your CA includes a large number of users, the process of re-activating each user and distributing reference numbers and authorization codes will require a considerable time investment.

3 Start the command-line interface to Entrust/Authority and log in as a Master User.

To start the command shell, click Start > Programs > Entrust PKI > Entrust Master Control Command Shell.

Note: If you choose to continue with the remaining steps in this procedure, note that your CA key pair will exist in software for a period of time.

4 At the prompt, enter the following command:

```
ca key update
```

Select software as the destination for the CA key pair. The services will restart.

5 Edit the entmgr.ini file by changing the path to the CRYPTOKI library to match the new hardware. For example, change the following entry:

```
[Entrust Settings]
CryptokiLibraryNT=full_path_to_the_CRYPTOKI_library
```

6 Ensure that the new CA hardware device is configured appropriately.

7 At the prompt, enter the following command:

```
ca key update
```

Select hardware as the destination for the CA key pair. The services will restart.

You have now updated the CA key pair to a second hardware device.

Hardware-based database protection

Entrust/PKI 6.0 provides enhanced database security with the addition of hardware-based database protection functionality. Hardware-based database protection works by storing a database key on a secured PKCS #11 v2 hardware device. This database key is used to secure sensitive information in the database and may be stored on the same hardware device that you may have set up to store the CA key pair, or on a second, separate device.

Note: Only a PKCS #11 v2 hardware device can store the database key. You cannot use a PKCS #11 v1 hardware device to store the database key.

CA hardware devices and redundancy

You must install and configure a PKCS #11 v2 hardware device in order to use hardware-based database protection. PKCS #11 v2 hardware devices are produced by several vendors. Entrust/PKI 6.0 works with many of them. For a complete list of vendors whose products are compatible with Entrust/PKI, visit www.entrust.com.

Attention: Entrust Technologies recommends that you back up your hardware device with a second CA hardware device for use in the event of a hardware failure.

If you don't back up your device with a second device, and any of the following events happen, you will not be able to access the database and Entrust/PKI will be inoperable:

- the hardware device fails
- the hardware token is lost (if applicable)
- the database key is lost or damaged

When to enable hardware-based database protection

You can enable hardware-based database protection immediately after you install and initialize Entrust/PKI 6.0 or at any time in the future.

When you enable database protection after first-time initialization, one backup of the database will exist on your system (llocated in the default folder used for backups). Entrust Technologies recommends that you delete this initial backup

and perform another backup after you enable hardware-based database protection.

If you enable database protection after Entrust/PKI has been running for a period of time, a number of backups will exist and you need to deal with these. Entrust Technologies recommends that you securely store all your backups and (if possible) delete those created before you enable hardware-based database protection.

Identifying CA hardware libraries when using separate devices

Complete the procedure below if you plan to use separate hardware devices to store the CA key pair and the database key.

Note: Skip this procedure if you are not using a hardware device to store the CA key pair, or if you use a hardware device that uses device tokens. You can use a single token to store the CA key pair and database key provided there is sufficient space on the token. If you are using more than one token, they need to be used simultaneously.

During first-time configuration, you can configure Entrust/PKI 6.0 to store the CA key pair on a PKCS #11 v2 hardware device (for example, "Device 1"). If you do so, the configuration utility updates the entmgr.ini file to include the path to the hardware device driver. To set up hardware-based database protection on a second hardware device (for example, "Device 2"), follow the next procedure (that is, "To identify CA hardware libraries when using separate devices" on page 101).

To identify CA hardware libraries when using separate devices

1. Create and open a new file called "ethardware.ini".
2. Open the entmgr.ini file in a text editor and copy the path to the hardware device driver located below the [Entrust Settings] section heading. For example:

```
[Entrust Settings]
CryptokiLibraryNT=D:\vt\virtualtoken.dll
```

3. After you copy the path to the hardware device driver from the entmgr.ini file, comment it out using a semicolon. (When Entrust/Authority processes the file, it ignores everything between the semicolon and the end of the line.) For example:

```
[Entrust Settings]
;CryptokiLibraryNT=D:\vt\virtualtoken.dll
```

4. Save and close the entmgr.ini file.

5 In the ethardware.ini file, create a section heading called "[CryptokiLibraries]" and paste the path to the hardware device driver from the entmgr.ini file below the heading. For example:

```
[CryptokiLibraries]
CryptokiLibraryNT=D:\vt\virtualtoken.dll
```

6 Below this line add the following line:

V2LibraryNT=*path to the second hardware device driver to be used for database protection*

where "path to the second hardware device driver to be used for database protection" may be, for example

```
d:\nfast\bin\cknfast.dll
```

7 Save the file to the Windows folder (for example, c:\winnt).

You have now identified the CA libraries when using separate hardware devices.

Enabling and disabling hardware-based database protection

A Master User will use Entrust Master Control Command Shell (or entsh) to enable, disable, and query hardware-based database protection.

To confirm that hardware-based database protection is enabled or disabled, see "To query hardware-based database protection" on page 104.

Note: The procedure described below may differ depending on the PKCS #11 v2 hardware device you have installed.

To enable hardware-based database protection

1 Start Entrust Master Control Command Shell on your workstation (see "Starting Entrust Master Control Command Shell" on page 347).

The Entrust Master Control Command Shell window opens.

2 Log in to Entrust/Authority Master Control as a Master User (see "Logging in to Entrust/Authority Master Control" on page 349).

3 At the prompt, enter

db hw-protection enable -alg *algorithm*

where "algorithm" is CAST-128, DES, or Triple-DES

The following message appears:

```
Select the destination of the database key
choose one of:
1. CAHdwareVendor01 SN: 99ERT-A7-00-1 SLOT: 897756
2. CAHdwareVendor02 SN: REM77Z28X SLOT: 1000000029
```

Note: If the hardware device does not appear in this list, confirm that it is correctly installed. Consult the documentation provided by the hardware manufacturer for details.

---

4 At the prompt, enter the value that corresponds to the destination device.

For example, to store the database key on CAHardwareVendor02, enter

```
2
```

If your CA hardware device requires the entry of a password, the following message appears:

```
Enter password for CA hardware:
```

5 Enter the password for the CA hardware.

The following message appears:

```
Hardware-based protection for database enabled.
```

You have now enabled hardware-based database protection.

To disable hardware-based database protection

1 Start Entrust Master Control Command Shell on your workstation (see "Starting Entrust Master Control Command Shell" on page 347).

The Entrust Master Control Command Shell window opens.

2 Log in to Entrust/Authority Master Control as a Master User (see "Logging in to Entrust/Authority Master Control" on page 349).

3 At the prompt, enter

```
db hw-protection disable
```

The following message appears:

```
Hardware-based protection for database disabled
```

You have now disabled hardware-based database protection.

Note: To use backup files created using hardware-based database protection, you need both the backup file and the hardware token or device. You cannot access these backups if you disable hardware-based database protection and remove the hardware component.

To query hardware-based database protection

1. Start Entrust Master Control Command Shell on your workstation (see "Starting Entrust Master Control Command Shell" on page 347).

   The Entrust Master Control Command Shell window opens.

2. Log in to Entrust/Authority Master Control as a Master User (see "Logging in to Entrust/Authority Master Control" on page 349).

3. At the prompt, enter

   ```
   db hw-protection query
   ```

   If hardware-based database protection is enabled, the following information appears:

   ```
   Hardware-based db protection ON
   Hardware Description CAVendor Corp. Ltd
   Hardware Serial Number 91SAM-34RR0-LU3
   Key Type CAST5
   Key Length 128
   ```

   If hardware-based database protection is disabled, information similar to the following appears:

   ```
   Hardware-based db protection OFF
   ```

   You have now queried hardware-based database protection.

Chapter 4

Managing Entrust/Authority data for Informix

The Entrust/Authority database stores important information about Entrust/PKI and its users. For example, the database may store CA signing keys and user encryption key-pair histories.

This chapter contains complete details on managing Entrust/Authority data for Informix, in the following sections:

- "Backing up Entrust/Authority data" on page 106
- "Validating the database" on page 126
- "Restoring data from backups" on page 127
- "Restoring Entrust/Authority data to the Directory" on page 133
- "Re-encrypting the database" on page 136
- "Changing the Informix database password" on page 139

Backing up Entrust/Authority data

Entrust/Authority data is stored in both the Entrust/Authority database and Entrust/Authority data files. The Entrust/Authority database stores important information about Entrust/PKI and its users. For example, the database may store CA signing keys and user encryption key-pair histories. Entrust/Authority data files include audit and error logs, archived users, Entrust/Authority .ini files, and environmental settings.

Because of the importance of Entrust/Authority data, Entrust Technologies recommends that you back up the database at regular, frequent intervals. If a problem occurs that affects the operation of the database, you will need backup files to restore Entrust/Authority.

For maximum protection of your data, Entrust Technologies strongly recommends that you adopt the following practices:

- Perform full backups of the Entrust/Authority database at regular intervals, at a minimum of once a day. To facilitate this process, Entrust/PKI lets you schedule backups that are initiated automatically. You can also initiate backups manually (for example, if you don't want to wait for a scheduled backup to occur).
- If you're using a Directory other than Microsoft Active Directory, back up the Directory when you back up the Entrust/Authority database. Entrust Technologies recommends you back up your Directory regularly. For information about backing up the Directory, see "Backing up the Directory" in Chapter 2 of *Using Entrust/PKI 6.0 on Windows*.
- Simulate a complete loss of data on a regular basis to ensure your backup and restore procedures work. That is, you should practice manually backing up and restoring Entrust/Authority data on a test machine. For information about restoring Entrust/Authority data, see "Restoring data from backups" on page 127.

This section provides information on

- "What is a backup?" on page 107
- "What types of backups does Entrust/Authority perform?" on page 107
- "Where is Entrust/Authority data backed up?" on page 110
- "Scheduling automatic backups with the graphical interface" on page 113
- "Changing the number of backups with the graphical interface" on page 117
- "Changing the backup directory with the graphical interface" on page 118
- "Configuring automatic backup settings from the command line" on page 120
- "Initiating manual backups" on page 122

- "Verifying full backups" on page 124
- "Archiving backups" on page 124

What is a backup?

A backup contains a copy of all Entrust/Authority data at a specific point in time. During an Entrust/Authority backup, the Entrust/Authority *database* and Entrust/Authority *data files* are backed up separately.

The Entrust/Authority database stores important information about Entrust/PKI and its users, including

- the CA signing key pair
- user DNs
- user encryption key-pair histories
- user information

The Entrust/Authority data files include

- audit and error logs
- archived users
- Entrust/Authority .ini files

What types of backups does Entrust/Authority perform?

An Entrust/Authority backup can be a "full backup" or an "incremental backup".

- A full backup is a copy of the Entrust/Authority database and data files at the time that the backup occurred.
- An incremental backup is a copy of the changes to the Entrust/Authority database since the previous backup (either full or incremental) occurred. An incremental backup is performed by backing up "transaction logs". A transaction log is a record of one change to the Entrust/Authority database. Whenever the Entrust/Authority database changes (for example, when you add a user or when a certificate is updated), the database creates a transaction log to record the change.

For details on transaction logs, see "About transaction logs" on page 108.

For details on the differences between full and incremental backups, see "Why does Entrust/Authority perform both full and incremental backups?" on page 108.

About transaction logs

When the Entrust/Authority database changes (for example, when you add a user or when a certificate is updated), the database creates and stores a transaction log. The transaction log provides a record of one change. As part of a backup (either full or incremental), the information in transaction logs is backed up and then deleted. The Entrust/Authority database creates new transaction logs to save changes that occur after completing the backup.

Because transaction logs exist only until their information has been backed up, the transaction logs that exist at any particular time provide a record of changes that have occurred since the previous backup. For example, suppose that Entrust/Authority performs an incremental backup at 12:00 p.m. After completing the backup, the Entrust/Authority database records changes to the database in transaction logs. At 1:00 p.m., Entrust/Authority performs an incremental backup, which saves the information in the transaction logs (that is, the changes that have occurred since 12:00 p.m.). Entrust/Authority then deletes the existing transaction logs. The Entrust/Authority database again begins recording changes to the database in new transaction logs. If a failure occurs at 1:45 p.m., the existing transaction logs contain only the changes that have not yet been backed up (that is, the changes that occurred since 1:00 p.m.).

Note: Depending on the cause of the failure, you may not be able to access or use the transaction logs. The failure may have deleted these logs or caused them to become corrupt.

When do backups occur?

By default, Entrust/Authority automatically performs a full backup between 3:00 a.m. and 5:00 a.m. every day. It automatically performs incremental backups every 60 minutes. This schedule ensures that the Entrust/Authority database and data files are backed up at frequent and regular intervals.

Entrust/PKI lets you schedule backups that are initiated automatically according to settings you specify. See "Scheduling automatic backups with the graphical interface" on page 113 for details on specifying when and how automatic backups occur.

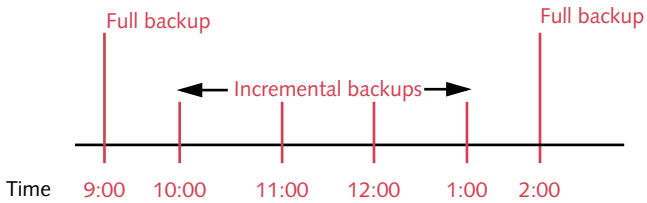
You can also initiate backups manually (for example, if you don't want to wait for a scheduled backup to occur). See "Initiating manual backups" on page 122 for details on initiating a manual backup.

Why does Entrust/Authority perform both full and incremental backups?

Depending on the size of the Entrust/Authority database, a full backup may require a significant amount of disk space, because it is a copy of the full

database. To ensure that changes to the Entrust/Authority database are backed up between full backups, without requiring the same amount of space for every backup, Entrust/Authority uses incremental backups. Because incremental backups save only changes to the database, incremental backups usually occupy significantly less space than full backups.

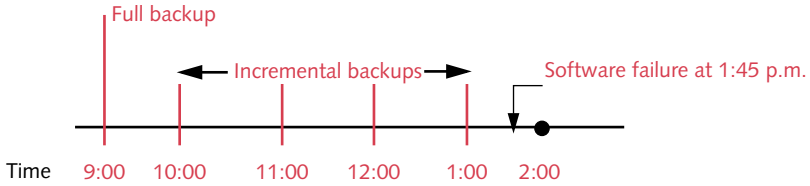
The following diagram shows how full backups and incremental backups work together to protect the Entrust/Authority database and data files. This diagram shows part of a backup schedule in which full and incremental backups occur between 9:00 a.m. and 2:00 p.m.



In this diagram, Entrust/Authority creates

- one full backup at 9:00 a.m. This backup is a copy of the database and data files as they exist at 9:00 a.m.
- one incremental backup at 10:00 a.m., 11:00 a.m., 12:00 p.m., and 1:00 p.m. Each of these backups is a copy of the changes to the database that occurred since the previous backup (that is, the changes that occurred during the previous hour).
- one full backup at 2:00 p.m. This backup is a copy of the database and data files as they exist at 2:00 p.m.

Suppose that a software failure occurs at 1:45 p.m., as shown in the following diagram:



To recover Entrust/Authority data to the time of the last backup (that is, the backup that occurred at 1:00), you use

- the full backup created at 9:00 to restore the entire Entrust/Authority database and data files

- the incremental backup created at 10:00 to restore the changes to the database that occurred between 9:00 and 10:00
- the incremental backup created at 11:00 to restore the changes to the database that occurred between 10:00 and 11:00
- the incremental backup created at 12:00 to restore the changes to the database that occurred between 11:00 and 12:00
- the incremental backup created at 1:00 to restore the changes to the database that occurred between 12:00 and 1:00

Depending on the cause of the failure, you may even be able to restore the changes to the database that occurred after the last backup and before the failure (that is, the changes that occurred between 1:00 p.m. and 1:45 p.m. and that were not backed up). These changes are recorded in transaction logs. For details on transaction logs, see "About transaction logs" on page 108.

Why are incremental backups so important?

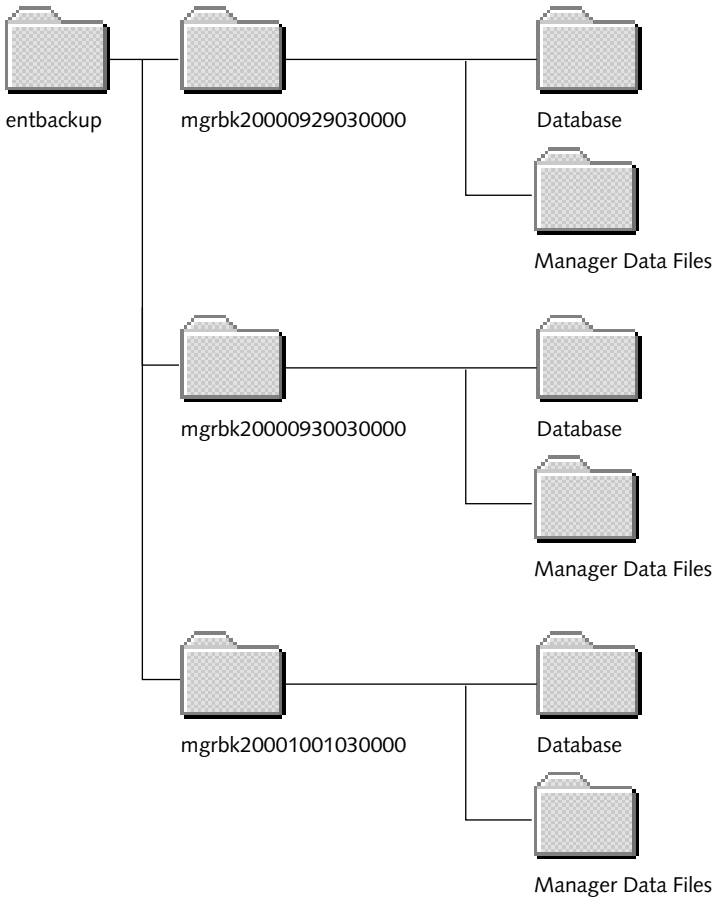
If you don't perform regular and frequent backups of the transaction logs through incremental backups, the space allocated for transaction logs in the Entrust/Authority database will become full and Entrust/Authority will stop making changes to the database. Entrust/Authority will begin making changes to the database again only after you perform an incremental backup. Entrust Technologies recommends that you schedule incremental backups to occur at least once an hour.

Where is Entrust/Authority data backed up?

Before you begin working with backups, you should know where they are stored and how they are organized.

Figure 1 on page 111 shows an example directory structure for full backups of Entrust/Authority data. In this example, three full backups have occurred at 3:00 a.m. on September 29, September 30, and October 1, 2000.

Figure 1: Sample directory structure for Entrust/Authority data backups



Backup directory

All backups are stored in the backup directory. The default path and name of the backup directory is

```
C:\entbackup
```

To change the name and location of the backup directory, see "Changing the backup directory with the graphical interface" on page 118.

Attention: If you are running multiple CAs, only one of your CAs should be configured to perform Entrust/Authority database backups; the other CAs back up only the Entrust/Authority data files. If you change this default configuration, your backups may fail and data loss may occur.

Contents of the backup directory

When Entrust/Authority performs a full backup, it creates a subdirectory in the backup directory. The name of each subdirectory identifies the date and time that Entrust/Authority performed the full backup. The format of the subdirectory name is as follows:

```
mgrbkyyyymmddhhmmss
```

where yyyy=year, mm=month, dd=day, hh=hour, mm=minute, and ss=seconds.

For example, if Entrust/Authority performs a full backup at 3:00 a.m. on September 29, 2000, Entrust/Authority creates a subdirectory named "mgrbk20000929030000" in the backup directory. The complete path to this example subdirectory is

```
C:\entbackup\mgrbk20000929030000
```

By default, Entrust/Authority keeps the seven most recent backups in the backup directory. You can increase or decrease the number of backups stored in the backup directory. For information about changing the number of backups stored online, see "Changing the number of backups with the graphical interface" on page 117.

How is each backup subdirectory organized?

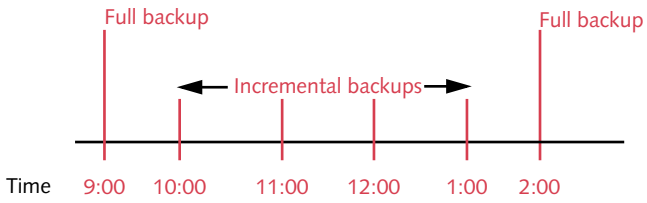
Each backup folder contains two subfolders: a Database subfolder and a Manager Data Files subfolder. These subfolders organize files created during backups. If the backup folder is named "mgrbk20000929030000", the complete paths to its subfolders are

```
C:\entbackup\mgrbk20000929030000\Database
C:\entbackup\mgrbk20000929030000\Manager Data Files
```

See Figure 1 on page 111 for a graphical representation of the backup directory structure.

The Database folder contains a copy of the Entrust/Authority database, and the Manager Data Files folder contains a copy of the Entrust/Authority data files. (To learn what is stored in the database and what is stored in the data files, see "What is a backup?" on page 107.)

Each backup subdirectory contains the files created for the full backup. Each backup subdirectory also contains files created for any incremental backups that have occurred after the full backup, but before the next full backup. For example, suppose that Entrust/Authority performs backups on September 29, 2000, as shown in the following example:



At 9:00 a.m., Entrust/Authority performs a full backup. Entrust/Authority creates a subdirectory in the backup directory named "mgrbk20000929090000" containing a Database folder and a Manager Data Files folder. The Database folder and Manager Data Files folder contain the backup files created for the full backup.

Entrust/Authority adds the backup files created for the incremental backups performed between 10:00 a.m. and 1:00 p.m. to the Database folder in the mgrbk20000929090000 subdirectory.

At 2:00 p.m., Entrust/Authority performs another full backup. Entrust/Authority creates a subdirectory in the backup directory named "mgrbk20000929140000" containing a Database folder and a Manager Data Files folder. The Database folder and Manager Data Files folder contain the backup files created for the full backup performed at 2:00 p.m.

Scheduling automatic backups with the graphical interface

By default, Entrust/Authority automatically performs a full backup at 3:00 a.m. every day, and incremental backups every 60 minutes. This schedule ensures that the Entrust/Authority database and data files are backed up at frequent and regular intervals. To learn why frequent and regular backups are crucial, see "What types of backups does Entrust/Authority perform?" on page 107.

You can change the schedule for automatic full and incremental backups. You can schedule

- the time and frequency of automatic full backups
- the frequency of automatic incremental backups

This section shows you how to schedule automatic backups with Entrust/Authority Master Control. To learn how to schedule automatic backups

from the command line, see "Configuring automatic backup settings from the command line" on page 120.

Scheduling automatic full backups to occur more than once a day with the graphical interface

If you want full backups to occur more frequently than once a day, you need to make sure the settings in the *Full backups never start before* and *Full backups never start after* fields do not cause a conflict with the *Full backup every* field. The *Full backup every* field controls how frequently backups are performed, while the *Full backups never start before* and *Full backups never start after* fields control when during the day backups may start.

If Entrust/Authority is unable to start a full backup before the time specified in the *Full backups never start after* field, the backup will not occur. If you set *Full backup every* to less than 24 hours, but keep the default settings for *Full backups never start before* and *Full backups never start after*, backups will not occur as expected.

For example, if you do not want full backups to occur between 9:00 a.m. and 5:00 p.m., but you wish full backups to occur every two hours between 5:00 p.m. and 9:00 a.m., set the fields as follows:

Full backup every: 2 hours

Full backups never start before: 17:00

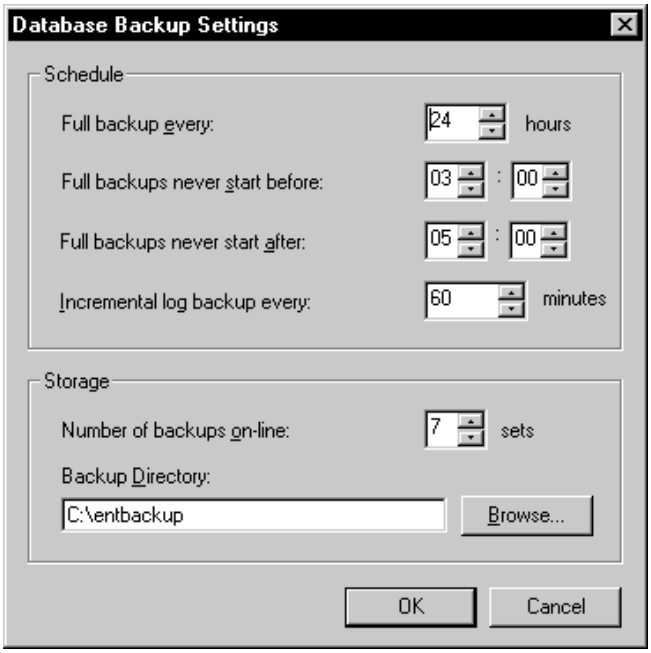
Full backups never start after: 09:00

Note: If you set *Full backups never start before* to 00:00 and *Full backups never start after* to 23:55, you ensure that full backups can occur throughout the day.

To change the schedule for automatic full backups with the graphical interface

1. Log in to Entrust/Authority Master Control as a Master User (see "Logging in to Entrust/Authority Master Control" in Chapter 2 of *Administering Entrust/PKI 6.0 on Windows).*
2. Click Database > Backup Settings.

The Database Backup Settings dialog box appears.



3 In the *Full backup every* field, type the interval (in hours) between automatic full backups. Entrust Technologies strongly recommends that you keep the default value of 24 hours or that you specify a value that is a multiple of 24 (such as 48 or 72).

If you set the *Full backup every* field to less than 24 hours (that is, you want full backups to occur more frequently than once a day), see "Scheduling automatic full backups to occur more than once a day with the graphical interface" on page 114.

Note: If you change the schedule of automatic full backups, Entrust Technologies strongly advises you to schedule full backups to occur at least once per day. The more time that has passed since the last full backup, the more data you risk losing if a problem occurs.

---

4 In the *Full backups never start before* field, type the time (in hours and minutes) when you want the first automatic full backup to start.

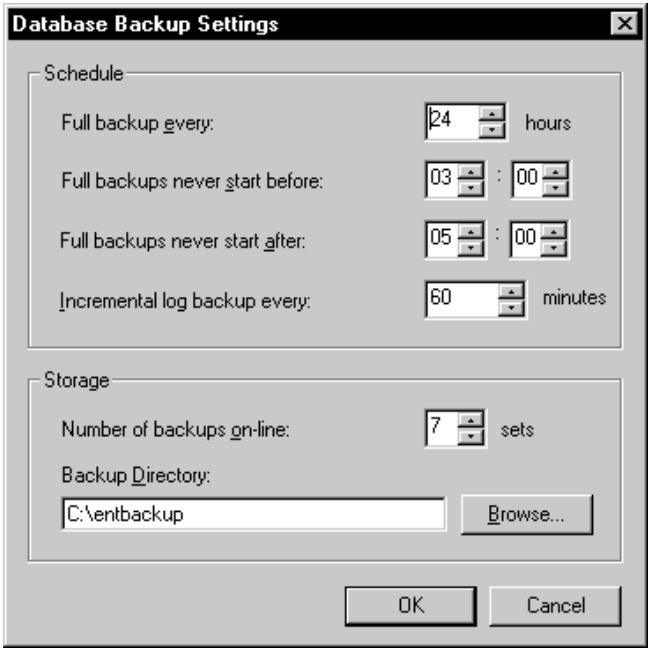
5 In the *Full backups never start after* field, type the time (in hours and minutes) after which automatic full backups should not start, and click *OK*.

You have now changed the schedule for automatic full backups. Changes take effect immediately.

To change the schedule for automatic incremental backups with the graphical interface

1. Log in to Entrust/Authority Master Control as a Master User (see "Logging in to Entrust/Authority Master Control" in Chapter 2 of *Administering Entrust/PKI 6.0 on Windows*).
2. Click Database > Backup Settings.

   The Database Backup Settings dialog box appears.



3. In the *Incremental log backup every* field, type the interval (in minutes) between automatic incremental backups, and click *OK*.

Note: Entrust Technologies strongly recommends that you schedule incremental backups at least once an hour. If you don't perform regular and frequent backups of the transaction logs through incremental backups, the space allocated for transaction logs in the Entrust/Authority database will become full and Entrust/Authority will stop making changes to the database.

You have now changed the schedule for automatic incremental backups. Changes take effect immediately.

Changing the number of backups with the graphical interface

Entrust/Authority creates a subdirectory in the backup directory for the files created during a full backup. Entrust/Authority also adds files created during incremental backups to this subdirectory. For information about backup directories and their contents, see "Where is Entrust/Authority data backed up?" on page 110.

By default, Entrust/Authority keeps the seven most recent backups online. For example, by default Entrust/Authority performs a full backup at 3:00 a.m. every day. If the first full backup is performed on September 29, 2000, the backup directory contains the following subdirectories after seven days:

```
C:\entbackup\mgrbk20000929030000
C:\entbackup\mgrbk20000930030000
C:\entbackup\mgrbk20001001030000
C:\entbackup\mgrbk20001002030000
C:\entbackup\mgrbk20001003030000
C:\entbackup\mgrbk20001004030000
C:\entbackup\mgrbk20001005030000
```

When seven subdirectories exist, Entrust/Authority deletes the oldest backup to make room for the new subdirectory when the next full backup occurs.

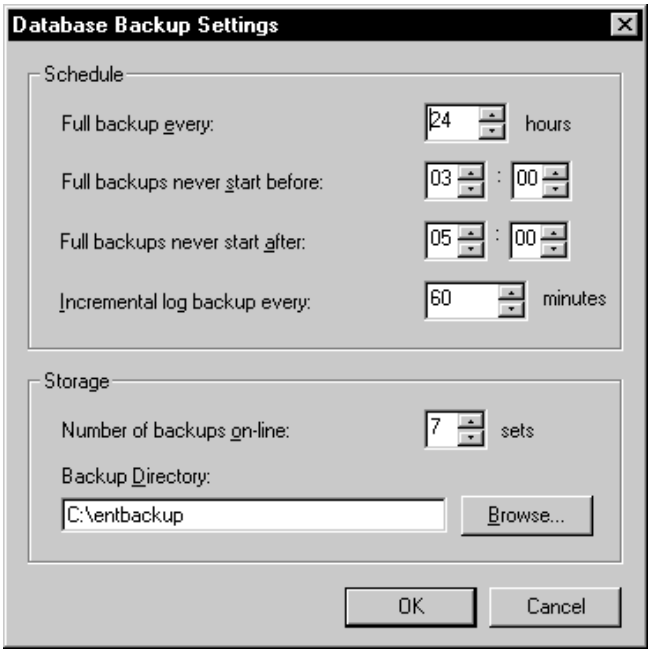
You can increase or decrease the number of online backups. As your database increases in size, you may want to reduce the amount of disk space required for online backups by decreasing the number of backups kept online. Or, you may want to increase the number of online backups to have more options available if you need to restore the database.

This section shows you how to change the number of online backups with Entrust/Authority Master Control. To learn how to change the number of online backups from the command line, see "Configuring automatic backup settings from the command line" on page 120.

To change the number of backups with the graphical interface

1. Log in to Entrust/Authority Master Control as a Master User (see "Logging in to Entrust/Authority Master Control" in Chapter 2 of *Administering Entrust/PKI 6.0 on Windows.*
2. Click Database > Backup Settings.

The Database Backup Settings dialog box appears.



3 In the *Number of backups on-line* field, type the number of backups you want Entrust/Authority to keep online in the backup directory, and click *OK*.

You have now changed the number of backups stored online. Changes to the backup settings take effect immediately.

Changing the backup directory with the graphical interface

Entrust/Authority stores backups in the backup directory (see "Backup directory" on page 111 for details on how backups are stored). You can change the name and location of the backup directory. Entrust/Authority stores the next backup in the new location.

If you change the name or location of the backup directory, Entrust Technologies strongly recommends the following:

- Put the backup directory on the Windows server that hosts Entrust/Authority, but on a different physical disk than the Entrust/Authority database. If a problem occurs with the disk that the database is located on, your backups are located on a different disk.
- Do not include spaces in the path to the backup directory.

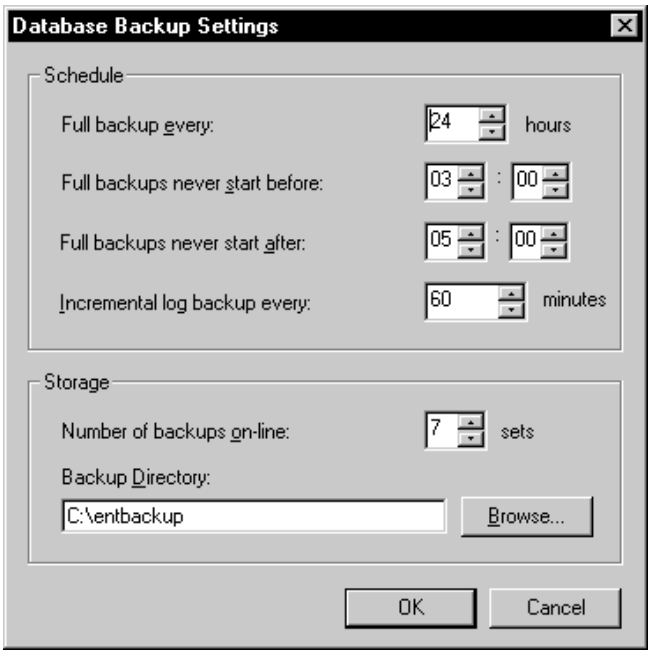
- Do not use more than 27 characters in the backup directory pathname. If you use more than 27 characters, the Entrust/Authority database will not recognize the backup directory pathname and the backup will fail.

This section shows you how to change the name and location of the backup directory with Entrust/Authority Master Control. To learn how to change the name and location of the backup directory from the command line, see "Configuring automatic backup settings from the command line" on page 120.

To change the name and location of the backup directory with the graphical interface

1. Log in to Entrust/Authority Master Control as a Master User (see "Logging in to Entrust/Authority Master Control" in Chapter 2 of *Administering Entrust/PKI 6.0 on Windows).*
2. Click Database > Backup Settings.

   The Database Backup Settings dialog box appears.



3. In the *Backup Directory* field, type the path and name of the backup directory, and click *OK*.

   You have now changed the name and location of the backup directory. Changes to the backup settings take effect immediately.

Configuring automatic backup settings from the command line

You can display and configure automatic backup settings for the Entrust/Authority database from the command line through Entrust Master Control Command Shell. For complete details on using the command line interface, see Chapter 15 of *Administering Entrust/PKI 6.0 on Windows*.

To display the settings for automatic backup of the Entrust/Authority database from the command line

1. Start Entrust Master Control Command Shell on your workstation (see "Starting Entrust Master Control Command Shell" in Chapter 15 of *Administering Entrust/PKI 6.0 on Windows*).
2. Log in to Entrust/Authority Master Control as a Master User (see "Logging in to Entrust/Authority Master Control" in Chapter 15 of *Administering Entrust/PKI 6.0 on Windows*).
3. At the prompt, enter

   ```
   db backup query
   ```

   Information similar to the following example appears, followed by the prompt:

   ```
   backup-directory           C:\entbackup
   keep-online                7
   incremental-backup-period  01:00
   full-backup-period         24:00
   not-before                 03:00
   not-after                  05:00
   ```

   You have now displayed the settings for automatic backup of the Entrust/Authority database.

To configure the settings for automatic backup of the Entrust/Authority database from the command line

1. Start Entrust Master Control Command Shell on your workstation (see "Starting Entrust Master Control Command Shell" in Chapter 15 of *Administering Entrust/PKI 6.0 on Windows*).
2. Log in to Entrust/Authority Master Control as a Master User (see "Logging in to Entrust/Authority Master Control" in Chapter 15 of *Administering Entrust/PKI 6.0 on Windows*).

3 At the prompt, enter the following, specifying a value for at least one of "-dir", "-keep", and "-timer":

```
db backup config <-dir backup-directory> <-keep n> <-timer
incremental-period full-period notbefore notafter>
```

where

- "-dir" refers to the backup directory for the database
- "-keep" refers to the number of backup sets to keep online
- "-timer" refers to the backup period, with "incremental-period" the incremental backup period, "full-period" the full backup period, "notbefore" the time before which the full backup must not start, and "notafter" the time after which the full backup must not start.

Note: If you want full backups to occur more frequently than once a day, see "Scheduling automatic full backups to occur more than once a day from the command line" on page 121.

For example, to specify that nine backup sets be kept online, that incremental backups occur every two hours, that full backups occur every 24 hours, and that backups not start before 3 a.m. or after 5 a.m., enter

```
db backup config -keep 9 -timer 02:00 24:00 03:00 05:00
```

The prompt appears.

You have now configured the settings for automatic backup of the Entrust/Authority database.

Scheduling automatic full backups to occur more than once a day from the command line

If you want full backups to occur more frequently than once a day, you need to make sure the values for the "notbefore" and "notafter" variables do not cause a conflict with the value for the "full-period" variable. The "full-period" variable controls how frequently backups are performed, while the "notbefore" and "notafter" variables control when during the day backups may start.

If Entrust/Authority is unable to start a full backup before the time specified by the "notafter" value, the backup will not occur. If you set "full-period" to less than 24 hours, but keep the default settings for "notbefore" and "notafter", backups will not occur as expected.

The syntax for scheduling the automatic backup timer is as follows:

```
db backup config -timer <incremental-period> <full-period>
<notbefore> <notafter>
```

For example, if you do not want full backups to occur between 9:00 a.m. and 5:00 p.m., but you wish full backups to occur every two hours between 5:00 p.m. and 9:00 a.m. (and incremental backups to occur every hour), set the variables as follows:

```
db backup config -timer 01:00 02:00 17:00 09:00
```

Note: If you set "notbefore" to 00:00 and "notafter" to 23:55, you ensure that full backups can occur throughout the day.

Initiating manual backups

Entrust/Authority lets you initiate backups manually. You can manually back up the database at any time without waiting for the next automatic backup. For example, if an automatic backup does not complete successfully, you should immediately back up the database manually. You can then resolve the problem that occurred during the automatic backup, with the knowledge that you have a complete backup of the data. You can perform both full and incremental backups manually.

Note: Entrust Technologies strongly recommends that you practice manually backing up and restoring the database on a test machine. That is, you should regularly simulate a complete loss and restore of data to ensure your backup and restore procedures work. For information on restoring Entrust/Authority data, see "Restoring data from backups" on page 127.

You should initiate a full backup manually

- before a major operation (such as upgrading Entrust/PKI or the operating system, physically moving the Entrust/Authority server or the disks where the database is stored, or re-encrypting the database)
- after your organization makes a change in the Security Policy property pages of Entrust/RA (for instance, if there is a change in the number of Security Officer or Administrator authorizations)
- after Security Officers or Administrators have their keys recovered

You should initiate an incremental backup manually after many transactions have occurred and you want to save them immediately instead of waiting for the next automatic backup (for example, when many Entrust users are added).

To initiate a full backup manually with the graphical interface

1. Log in to Entrust/Authority Master Control as a Master User. See "Logging in to Entrust/Authority Master Control" in Chapter 2 of *Administering Entrust/PKI 6.0 on Windows*.
2. Click Database > Backup.

   Entrust/Authority validates the database and then backs up the database and data files. Validating the database checks the integrity of the data to ensure that Entrust/Authority does not back up corrupt data.

   A dialog box appears when the backup is finished.
3. Click *OK*.

   You have now initiated a full backup manually.

To initiate a full backup manually from the command line

1. Start Entrust Master Control Command Shell on your workstation (see "Starting Entrust Master Control Command Shell" in Chapter 15 of *Administering Entrust/PKI 6.0 on Windows*).
2. Log in to Entrust/Authority Master Control as a Master User (see "Logging in to Entrust/Authority Master Control" in Chapter 15 of *Administering Entrust/PKI 6.0 on Windows*).
3. At the prompt, enter

   ```
   db backup full
   ```

   When the backup is complete, this message appears:

   ```
   full backup completed successfully
   ```

   The prompt appears. This command may take some time to execute.

Note: Entrust/Authority does not validate the database when it is backed up using Entrust Master Control Command Shell.

You have now initiated a full backup manually.

To initiate an incremental backup manually with the graphical interface

1. Log in to Entrust/Authority Master Control as a Master User. See "Logging in to Entrust/Authority Master Control" in Chapter 2 of *Administering Entrust/PKI 6.0 on Windows*.
2. Click Database > Transaction Log Backup.

   When the backup is complete, a dialog box confirms the backup.

3 Click *OK*.

You have now initiated an incremental backup manually.

To initiate an incremental backup manually from the command line

1 Start Entrust Master Control Command Shell on your workstation (see "Starting Entrust Master Control Command Shell" in Chapter 15 of *Administering Entrust/PKI 6.0 on Windows*).

2 Log in to Entrust/Authority Master Control as a Master User (see "Logging in to Entrust/Authority Master Control" in Chapter 15 of *Administering Entrust/PKI 6.0 on Windows*).

3 At the prompt, enter

```
db backup incremental
```

When the backup is complete, this message appears:

```
incremental backup completed successfully
```

The prompt appears.

You have now initiated an incremental backup manually.

Verifying full backups

If problems occur during a full backup, you may not be able to use it to restore the Entrust/Authority database. You should always verify each full backup (initiated automatically or manually) to ensure that it completed successfully. You should verify a full backup as soon as possible after it completes. For example, automatic full backups run daily at 3:00 a.m. by default. You should verify the backup every morning to ensure it completed successfully.

You verify a full backup using audit logs. Entrust/Authority writes messages to audit logs while running a backup. For information about viewing and managing audit logs, see "About audit logs" in Chapter 8 of *Administering Entrust/PKI 6.0 on Windows.*

If the messages in the audit logs show that problems occurred during a backup, you can use the information in the messages to determine their cause.

Archiving backups

If you need to restore from backup, you can use the backups stored in the backup directory on the UNIX server that hosts Entrust/Authority. However, you may not be able to access these backups if a major hardware or software failure makes the UNIX server that hosts Entrust/Authority inaccessible or inoperable. For

additional protection, you should have several archived backups available to you. In fact, Entrust Technologies recommends that you archive your backups daily. The more archived backups that are available to you, the more options you have for restoring data.

You archive a backup by copying it from the backup directory to other media (such as magnetic tape or another hard drive), clearly labelling the media, and storing it in a secure location.

Attention: Store the backup media in a secure facility in a physical location that is different from the location of the UNIX server that hosts Entrust/Authority (preferably off-site).

Validating the database

Validating the Entrust/Authority database involves checking the integrity of its data to determine whether it has been corrupted by disk errors or by someone tampering with the protected data.

Whenever a record in the Entrust/Authority database is retrieved for use, Entrust/Authority automatically validates the retrieved record.

During database validations, Entrust/Authority records information in audit logs. (For information on checking audit logs, see Chapter 8 in *Administering Entrust/PKI 6.0 on Windows*.) If the audit logs indicate that the data lacks integrity or if the validation does not complete successfully, you must restore the Entrust/Authority database from backup. See "Restoring data from backups" on page 127.

To manually validate the database with the graphical interface

1. Log in to Entrust/Authority Master Control as a Master User (for details, see "Logging in to Entrust/Authority Master Control" in Chapter 2 of *Administering Entrust/PKI 6.0 on Windows*).
2. Click Database > Validate.

   A progress dialog box appears while Entrust/Authority validates the database.

   When the validation is complete, a dialog box appears confirming the validation.
3. Click *OK*.

   You have now validated the database.

To manually validate the database from the command line

1. Start Entrust Master Control Command Shell on your workstation (see "Starting Entrust Master Control Command Shell" in Chapter 15 of *Administering Entrust/PKI 6.0 on Windows*).
2. Log in to Entrust/Authority Master Control as a Master User (see "Logging in to Entrust/Authority Master Control" in Chapter 15 of *Administering Entrust/PKI 6.0 on Windows*).
3. At the prompt, enter

   ```
   db integrity validate
   ```

   Information about the progress of the validation appears. When the validation is complete, a message similar to the following appears:

   ```
   100% complete; estimated time remaining 00:00:00 @ (0.0/s)
   ```

   The prompt appears.

   You have now validated the database.

Restoring data from backups

If you have a hardware or software failure, or some other problem occurs with your Entrust/Authority database, you can restore the database and data files if you have a backup. Restoring data from backup means bringing the data back to exactly the same state as when the backup was performed.

Note: To restore data from backup, you need to copy the data from your tape media to the local machine before beginning the restore operation. You cannot restore directly from tape media.

For complete details on backing up Entrust/Authority data, see "Backing up Entrust/Authority data" on page 106.

Since the Entrust/Authority database and data files are backed up in different formats, restoring the database and the data files involves two separate procedures.

There are five steps involved in restoring Entrust/Authority data from backup:

1. Decide which backup to restore from. See "Deciding which backup to use when you restore" on page 127.
2. Restore the Entrust/Authority data files. See "Restoring the Entrust/Authority data files" on page 128.
3. Restore the file "authauto.ini". See "Restoring the file authauto.ini" on page 129.
4. Restore the file "onconfig". See "Restoring the file onconfig" on page 130.
5. Restore the Entrust/Authority database. See "Restoring the Entrust/Authority database" on page 130.

Deciding which backup to use when you restore

If you have determined that you need to restore from backups, you need to decide which backup to restore from. By default, Entrust/Authority keeps seven online backups in the backup directory. How do you decide which backup to use when you restore?

You always restore Entrust/Authority data from a full backup, followed optionally by incremental backups.

Select the backup directory to restore from using the following criteria:

- Choose the most recent backup to avoid losing changes.
- Choose a backup that does not include any known corrupted data.

Once you have chosen the backup to restore from, record its complete pathname (for example, C:\entbackup\mgrbk20000929030000). You will need to enter it later during the restore process.

Archiving the backup directory

Once you have selected the backup to restore from, Entrust Technologies recommends that you archive the backup subdirectory before performing the restore operation. This will ensure that if problems occur during the restore process, you will still have access to the backup files. For details on archiving a backup, see "Archiving backups" on page 124.

Have I lost the changes that occurred since the last backup?

You may also be able to restore changes to the database that occurred after the last backup and before the failure. The Entrust/Authority database records changes to the database since the last backup in transaction logs. You can recover the transaction logs if all of the following conditions are met:

- The last full backup was successful and you have the backup files.
- The file "path\Ifmxdata\entrust_nt\rootdbs.000" exists and is not corrupt.
- The file "path\Ifmxdata\entrust_nt\logdbs.000" exists and is not corrupt.

If any of these conditions are not met, you can restore to the last full or incremental backup, but you cannot recover changes to the database that occurred between the last backup and the failure.

During the restore process, you will have an opportunity to restore the current transaction logs. See "Restoring the Entrust/Authority database" on page 130.

Restoring the Entrust/Authority data files

The first step in restoring from backup is to restore the Entrust/Authority data files.

Note: You must restore the Entrust/Authority data files *before* restoring the Entrust/Authority database.

To restore the Entrust/Authority data files

1. Log in to the Entrust/Authority server as the Windows local system administrator.
2. Stop the Entrust/Authority service (see the "Setting up the Entrust/PKI environment" chapter in *Using Entrust/PKI 6.0 on Windows*).
3. If you are using a third-party Directory, stop your Directory.
4. In the Windows taskbar, click Start > Settings > Control Panel.

The Control Panel window appears.

5 Double-click *Services*.

Note: In a Windows 2000 environment, double-click *Administrative Tools* before you double-click the *Services* icon.

---

6 Stop the Informix Dynamic Server 2000 service.

This service appears as Informix IDS 2000 in the Control Panel.

7 Stop the Informix Dynamic Server Message service.

8 Rename the directory "authdata" (for example, to "oldauthdata" or something similar). By default, the authdata folder is located in C:\authdata on the Entrust/Authority server.

Note: If you upgraded to Entrust/PKI 6.0, the "authdata" folder is called "entmgrdata".

---

9 Copy the Entrust/Authority data files folder from the selected backup to the drive containing "oldauthdata". The data files folder is named "Manager Data Files" (for example, C:\entbackup\mgrbk20000929030000\Manager Data Files).

10 Rename the Manager Data Files folder "authdata".

Note: If you upgraded to Entrust/PKI 6.0, rename the Manager Data Files folder "entmgrdata".

You have now restored the Entrust/Authority data files.

Restoring the file authauto.ini

After restoring the Entrust/Authority data files, you may need to restore the file "authauto.ini".

To determine whether you need to restore this file, check the entry "DbLoginName=" in the [Login] section of the file entmgr.ini. (The entmgr.ini file is located by default in the C:\authdata\manager folder on the Entrust/Authority server.) For more details on the entmgr.ini file, see "Customizing the entmgr.ini file" on page 527.

If the DbLoginName= entry is blank, you do not need to restore the authauto.ini file.

If the DbLoginName= entry has a value, you need to restore the authauto.ini file.

To restore the authauto.ini file

1. Log in to the Entrust/Authority server as the Windows local system administrator.
2. Set the database password in the DbLoginPw= entry under the [Login] section of the file entmgr.ini. See "Customizing the entmgr.ini file" on page 527.

   Once you have specified the database password, Entrust/Authority encrypts the plaintext DbLoginPw entry and moves it to the authauto.ini file (located by default in C:\authdata\manager\authauto.ini).
3. If you had enabled automatic services login, you must re-enable it. See "Enabling automatic startup" in the "Setting up the Entrust/PKI environment" in *Using Entrust/PKI 6.0 on Windows*.

   You have now restored the file "authauto.ini".

Restoring the file onconfig

After restoring the Entrust/Authority data files, you need to restore the file "onconfig" in the <path>\informix\etc directory. This file contains configurable parameters affecting the operation of the Entrust/Authority database.

To restore the onconfig file

1. Log in to the Entrust/Authority server as the Windows local system administrator.
2. Copy the file "onconfig" from this folder:

   ```
   C:\entbackup\mgrbkyyyymmddhhmmss\Manager Data Files
   ```

   to this folder:

   ```
   C:\informix\etc
   ```

   where "mgrbkyyyymmddhhmmss" is the backup folder you've chosen to restore from.

   You have now restored the file "onconfig".

Restoring the Entrust/Authority database

The following procedure assumes that you installed Entrust/PKI in the default installation folder. If you installed to a different folder, you must know the full path to it before you begin the following procedure. If you are unsure of any step, contact Entrust Support.

To restore the Entrust/Authority database

1. In the Windows taskbar, click Start > Programs > Command Prompt.

   The Command Prompt window appears.

2 In the Command Prompt window, navigate to the Entrust/PKI location, for example:

```
C:\Program Files\Entrust\Authority\Bin
```

3 Decide which parameters to use for entrestore.bat.

The command you use to restore the Entrust/Authority database is entrestore.bat". To decide which parameters to use with the entrestore.bat command, review the following table.

Use this parameter...	when you want to...
-i	use an interactive restore method which proceeds through a series of steps, allowing you to view a list of available backups and choose the one that you want to restore from
backupdir	restore from a full backup contained in a specific directory
-latest	restore from the most recent full backup
<*log_fileset*> or <-none>	limit the number of incremental backups that are restored along with the full backup Use the "log_fileset" variable to specify, using an integer, the number of incremental backups that you want restored. (If you specify "2", the *first* two incremental backups are restored.) Use the parameter "-none" to specify that you don't want any incremental backups restored. If you don't include this option (that is, you don't specify either the "log_fileset" variable or the "-none" parameter), all incremental backups in the selected backup directory are restored.

4 For examples of how to use entrestore.bat with parameters, see "Examples of using parameters with entrestore.bat" on page 131.

5 Enter the "entrestore" command with the appropriate options at the DOS command prompt.

Examples of using parameters with entrestore.bat

Some examples of using the entrestore.bat command with various parameters are shown below.

- The following example uses an interactive restore method which proceeds through a series of steps, allowing you to view a list of available backups and choose the one that you want to restore from.

  ```
  entrestore -i
  ```

- The following example restores from the most recent full backup and all incremental backups in the same backup subdirectory, using the "-latest" parameter:

  ```
  entrestore -latest
  ```

  The "-latest" parameter also gives you the opportunity to restore the current transaction logs during the restore process.

- The following example restores from the full backup in the backup subdirectory created on September 29, 2000. All incremental backups in this subdirectory are also restored. In this example, "mgrbk20000929123456" is the value of the variable "backupdir".

  ```
  entrestore mgrbk20000929123456
  ```

  When you use the "backupdir" variable, you do not need to include the full path for the backup directory if it is in the default backup location. If the backup directory is not in the default backup location, you need to include the full path.

- The following example restores from the full backup in the backup subdirectory created on September 29, 2000. The first two incremental backups created following the full backup are also restored.

  ```
  entrestore mgrbk20000929123456 2
  ```

- The following example restores from the full backup in the backup subdirectory created on September 29, 2000. No incremental backups in this subdirectory are restored.

  ```
  entrestore mgrbk20000929123456 -none
  ```

What's next?

After the restore operation executes, you should perform a full backup. Incremental backups do not begin until a full backup has been performed.

Restoring Entrust/Authority data to the Directory

The Directory stores all the certificates produced by Entrust/Authority. These include the CA verification certificates, policy certificates, the First Officer certificate, all cross-certificates, and all user certificates. It also stores CRLs and ARLs.

If certificate information is lost or becomes corrupt (for example, if someone accidently deletes or rearranges information in the Directory, or the hard drive fails), you can restore certificate information to the Directory.

Note: You should back up your Directory on a regular basis using your Directory backup tools. The procedure described in this section will restore only certificate information. If any other type of Directory information becomes corrupt or is lost, such as object classes, Directory entries, or Directory attributes, you can retrieve this information only from a backup generated by your Directory backup tools. For more information on backing up your Directory, see "Backing up the Directory" in Chapter 2 of *Administering Entrust/PKI 6.0 on Windows*.

You have two options when restoring Entrust/Authority data to the Directory: you can either restore all data, or you can restore only CA data. The option to restore all data restores all CA certificate information (for example, the First Officer certificate, policy certificates and cross certificates) and all user certificates. The option to restore CA data restores all CA certificate information, but does not restore any user certificates.

The restore to Directory operation is not a replacement for performing backups using a third-party Directory backup tool. If you need to restore data to the Directory, you should use the third-party backup tool.

Note: You should run the Restore to Directory command in Entrust/Authority Master Control *only* if you are unable to restore all data to the Directory with the third-party backup tool.

When you run the Restore to Directory command, Entrust/Authority retrieves DNs and certificate information from the Entrust/Authority database. Entrust/Authority compares the DNs retrieved from the database with the DNs in the Directory. If a DN is missing from the Directory (for example, if an entry has been lost from the Directory), Entrust/Authority writes the DN of the missing entry to this file in this location: authdata\log\restoreErrors.txt. You can set up bulk commands that will recreate the missing Directory entries by using the DNs in the restoreErrors.txt file. For more information on using bulk commands to add

users, see "Adding users in bulk to Entrust/PKI" in Chapter 6 of *Administering Entrust/PKI 6.0 on Windows.*

Restoring Entrust/Authority data to the Directory is CPU- and network-intensive. Depending on how many users there are in your CA domain, the restore operation can take a long time. If you do not need to restore any lost or corrupt user certificates, you can save a lot of time by restoring just the CA data. If you must restore user certificates, consider doing it during off-peak hours. Also, you can save time by restore only specific user certificates through bulk operations.

To restore Entrust/Authority data to the Directory with the graphical interface

1. Restore the Directory from backup using your third-party Directory backup tool. If you determine that you need to restore Entrust/Authority data to the Directory, continue to step 2.
2. Log in to Entrust/Authority Master Control as a Master User (for details, see "Logging in to Entrust/Authority Master Control" in Chapter 2 of *Administering Entrust/PKI 6.0 on Windows*).
3. Click Configuration > Restore to Directory.

   Click Restore All or Restore CA Data in the cascading menu, depending on whether you want to restore all data to the Directory (CA certificates and user certificates) or you want to restore just the CA data to the Directory (CA certificates only).

   A dialog box appears, asking you to confirm the restore to Directory operation.
4. Click *OK*. The Restore to Directory Progress dialog box appears.

   If you have chosen to restore all data and there are many users in your CA domain, the restore to Directory operation may take a long time. You can click *Cancel* at any point during the operation and resume from where you left off at another time.

   When the restore operation completes, a dialog box appears, indicating that all certificates have been successfully restored to the Directory.
5. Click *OK*.

   You have now restored Entrust/Authority data to the Directory.

To restore Entrust/Authority data to the Directory from the command line

1. Start Entrust Master Control Command Shell on your workstation (see "Starting Entrust Master Control Command Shell" in Chapter 15 of *Administering Entrust/PKI 6.0 on Windows*).

2. Log in to Entrust/Authority Master Control as a Master User (see "Logging in to Entrust/Authority Master Control" in Chapter 15 of *Administering Entrust/PKI 6.0 on Windows*).
3. At the prompt, enter either

   ```
   dsa restore ca
   ```

   or

   ```
   dsa restore all
   ```

   where "ca" indicates that you're restoring CA data to the Directory, and "all" indicates that you're restoring all certificates to the Directory.

   Information about the progress of the restore process appears, followed by the prompt. This command may take some time to execute if you are restoring all data.

   You have now restored Entrust/Authority data to the Directory.

Re-encrypting the database

The Entrust/Authority database is encrypted to protect its sensitive data, such as copies of users' decryption private keys. You may need to occasionally re-encrypt the database if your organization's procedures and policies requires it.

Re-encrypting the database involves generating new random values for

- the password for the Entrust profile used by the Administration subsystem. For information about this subsystem, see "The Entrust/Authority service" in Chapter 2 of *Administering Entrust/PKI 6.0 on Windows.*
- the keys used to encrypt the database.

When the new values for the keys have been generated, the keys are used to re-encrypt the database.

Attention: Re-encrypting the database is a serious and potentially dangerous operation that is usually not required. If the operation is interrupted for any reason, the Entrust infrastructure may be unusable and you will have to restore the database from backup. Moreover, the Entrust/Authority service must be stopped during a database re-encryption, which may take a considerable amount of time depending on the size of your database.

To re-encrypt the database with the graphical interface

1. Log in to Entrust/Authority Master Control as a Master User (for details, see "Logging in to Entrust/Authority Master Control" in Chapter 2 of *Administering Entrust/PKI 6.0 on Windows).*
2. Click Database > Re-encrypt.

   A dialog box appears stating that you are about to re-encrypt the database. Before you proceed, read the information in the dialog box.
3. If you are sure you want to re-encrypt the database, enter your Master User login password.
4. Click *OK*.

   A dialog box appears prompting you to back up the database.

Attention: You are strongly advised to click *Yes* to back up the database now. An unexpected error during re-encryption, such as a disk failure or a power outage, may render the database completely useless. In such a case, you will need to restore it from a backup copy of the database.

5. Click *Yes*.

The database is validated and backed up. A progress dialog box appears, and then closes when the backup ends.

The Entrust/Authority service stops and starts up again automatically.

A dialog box appears asking if you want to proceed with re-encrypting the database.

6 If you are absolutely sure you want to re-encrypt the database, click *OK*.

The re-encryption operation begins. This may take a while, depending on the size of the database and the speed of the server.

When the re-encryption is complete, a dialog box appears confirming the re-encryption.

7 Click *OK*.

You have now re-encrypted the database. All data in the database has been secured with new encryption keys.

Note: To have the most recent data available as a backup, you should now perform another backup.

To re-encrypt the database from the command line

1 Start Entrust Master Control Command Shell on your workstation (see "Starting Entrust Master Control Command Shell" in Chapter 15 of *Administering Entrust/PKI 6.0 on Windows*).

2 Log in to Entrust/Authority Master Control as a Master User (see "Logging in to Entrust/Authority Master Control" in Chapter 15 of *Administering Entrust/PKI 6.0 on Windows*).

3 At the prompt, enter

```
db re-encrypt
```

A message appears, warning you that re-encrypting the database is a serious undertaking and asking you to confirm that you want to proceed.

4 Enter

```
y
```

Another message appears, warning you that your Entrust/PKI will probably be unusable if re-encryption fails, and asking whether you would like to perform a backup now.

5 Enter

```
y
```

When the backup is complete, this message appears:

```
backup completed successfully
```

Re-encryption of the database starts. This may take some time to execute. When re-encryption is complete, the following message appears:

```
Database re-encryption completed successfully.
```

The prompt appears.

You have now re-encrypted the database.

Changing the Informix database password

As part of routine system maintenance, you may want to change the Informix database password.

To change the Informix password

1. In the Windows taskbar, click Start > Settings > Control Panel.

   The Control Panel window appears.

2. Double-click *Services* in the Control Panel window.

Note: In a Windows 2000 environment, double-click *Administrative Tools* before you double-click the *Services* icon.

   The Services control panel appears.

3. Select the Informix IDS 2000 - entrust_nt service and click *Startup*.

Note: Select the Informix Server Discovery Process for SNMP if you have one. If you are running Windows 2000, select the Logon tab.

---

4. The Service dialog appears and should show the "informix" account.
5. In the Service dialog, enter and confirm the new password.

   You have changed the password for the "informix" account. The new password will take effect when you next reboot your system.

Chapter 5

Customizing Entrust/PKI

You can tailor Entrust/PKI to meet the specific needs of your organization. For instance, you can create roles and user policies, and you can customize user types.

Roles define the abilities of a user to administer Entrust/PKI through Entrust/RA. Security Officers have the most administrative capabilities. For instance, they can create new roles to supplement the predefined roles of Security Officer, Administrator, Directory Administrator, End User, and Auditor. End Users have no administrative capabilities at all. For more information, see "Creating roles" on page 142.

Note: The AutoRA Administrator role is also predefined role. It is a fully customizable role. For information about the AutoRA Administrator role, see *Entrust/AutoRA 6.0 Guide for Windows NT*.

Roles are governed by default certificate settings called "user policy" certificates. Entrust/PKI provides three default user policy certificates: Security Officer Policy, Administrator Policy, and End User Policy. You can change any of these policies and create new ones through Entrust/RA. For more information on user policy certificates, see "Setting Entrust/PKI user policies" on page 167.

User types are categories of users that you can add to Entrust/PKI 6.0. Entrust/PKI provides two default user types: Person and Web Server. A user of type "Person" is for people who are administrative users and end users. A user of type "Web Server" is for a Web server. If you are using Entrust/Profile Server, you would also use the "Web Server" type. For each user, you can associate a role that defines the user's capabilities. You can also create new user types. For instance, you can create a user type for hardware or software devices that access Entrust/PKI, such as routers, or for contract employees from a particular agency. For more information, see "Modifying the New User dialog box" on page 189.

Creating roles

In addition to the predefined roles provided with Entrust/PKI, you can create new roles. (The predefined roles are "Security Officer", "Administrator", "Directory Administrator", "End User", "Auditor", and "AutoRA Administrator". For an overview of these roles, see the "About Entrust/PKI" chapter in *Using Entrust/PKI 6.0 on Windows*.) For each role you create, you can set permissions that specify which administrative operations the role allows, and whether those operations require authorization. You can also create end-user roles that have different user policies and no administrative permissions.

By narrowly defining the capabilities of a role you create, you can improve the security of your PKI. For example, you can create a role to administer only the groups or roles that you specify.

This section describes what administrative roles are, how to create them, and how to administer them. Read this section if you want to create new administrative roles.

You should also read the "Setting up the Certification Authority (CA)" chapter in *Using Entrust/PKI 6.0 on Windows*. Searchbases and groups are features in Entrust/PKI 6.0 that can help you clearly define which users a role can administer. For more information, see "When to create a role" on page 148.

Note: Administrative users (Security Officers, Administrators, and so on) can't modify the permissions of a role if they do not already have permissions that are identical to the role they want to modify. For example, a Security Officer with a specific set of permissions can't modify the permissions of a Security Officer with a different set of permissions. This is a security precaution. Another administrative user with corresponding permissions must make the required changes.

Overview of roles

A role lets you either administer Entrust/PKI 6.0 to some extent (as in the case of an administrative role), or not administer it at all (as in the case of an end-user role).

To create an administrative role, you need to understand what types of tasks an administrative user can do. For instance, an administrative user can begin cross-certification, view authorization codes and reference numbers, add users, add user groups, assign certificate types, create administrative roles, and so on.

Permission settings specify which administrative tasks each role allows. Table 5 on page 144 lists all administrative tasks and their corresponding default settings for the predefined roles of Security Officer, Administrator, Directory

Administrator, End User, and Auditor. While you can't change permission settings for predefined roles (with the exception of the AutoRA Administrator role which is not shown in the table), you can change their names, user policy, and number of authorizations. You can also copy the roles to use as a basis for creating new roles. (For more information on copying a role, see "Creating a role" on page 161.)

Note that in Table 5, the "End User" role has no administrative permissions. You can't give an end-user role administrative permissions. However, you can create end-user roles with different user policies.

One other Entrust/PKI role, "Master User", is administrative in the sense that it controls aspects of the installation and operation of Entrust/PKI. But Master Users have no permissions to administer users and can only log in to Entrust/Authority Master Control. For a general description of the Master User role, see the "About Entrust/PKI" chapter in *Using Entrust/PKI 6.0 on Windows*.

By default, only Security Officers can create roles. As a Security Officer, you can create a new role and give it permission to create roles. Then members of the new role can also create and administer roles.

You can rename the predefined roles in Entrust/RA. For example, you could change the name of the "Security Officer" role to "Senior Administrator".

When you create a role, you can specify that certain administrative tasks require authorizations to complete. For example, you can create a role in which to delete a group requires authorization. You can also specify that multiple authorizations are required. In Table 5, an "A" beside a permission indicates that the operation requires authorization. Note that some operations have the "requires authorization" setting, even though the role doesn't have permission to do the operation. In these cases, the requires authorization setting is a recommended default in case you copy the role and grant it the permission.

Copying an existing role (such as a predefined role) is a convenient way to create a basis for a new role. For example, you can copy the Administrator role and give it additional permissions to create, delete, and administer all groups.

Note: If you are looking for more information about the predefined AutoRA Administrator role, see *Entrust/AutoRA 6.0 Guide for Windows NT*.

When you change a role's permissions, all members of the role instantly inherit the new settings. If you want members of a role to have different permissions, you must create a new role for each set of permissions that you want to give.

Table 5: Default settings for predefined Entrust roles (excluding the AutoRA Administrator)

Setting (A=requires authorization)	Security Officer	Administrator	Directory Administrator	End User	Auditor
Default User Policy certificate	Security Officer Policy	Administrator Policy	Administrator Policy	End User Policy	Administrator Policy
Number of authorizations required for sensitive operations	1	1	1	—	1
Audit Logs					
View own logs	✓	✓			✓
View all logs	✓	✓			✓
Bulk & Reports					
Process bulk files	✓	✓	✓		
Create reports	✓	✓			✓
Certificates					
Administer all categories	✓	✓			✓
Administer selected categories			✓(None)	✓(None)	
Administer all types	✓	✓			✓
Administer selected types			✓(None)	✓(None)	
Certification Authority (CA): CA					
View CA certificates	✓	✓			✓
Update CA signing keys	✓A	A	A		A
Revoke CA keys	✓A	A	A		A
View list of imported CAs	✓				✓
Import/export CA public keys	✓				

Table 5: Default settings for predefined Entrust roles (excluding the AutoRA Administrator)

Setting (A=requires authorization)	Security Officer	Administrator	Directory Administrator	End User	Auditor
Certification Authority (CA): Cross-certification					
View	✓				✓
Initiate	✓A	A	A		A
Revoke	✓A	A	A		A
Complete	✓A	A	A		A
Certification Authority (CA): Subordinate					
View	✓				✓
Add subordinate CAs	✓	A	A		A
Revoke	✓	A	A		A
Directory					
Bind to Directory	✓	✓	✓		
Change Directory password	✓	✓			
View entries	✓	✓	✓		✓
Create, delete, modify entries	✓	✓	✓		
Groups					
View	✓	✓			✓
Rename	✓				
Create	✓				
Delete	✓				
Administer all groups	✓				✓
Administer any group to which they belong		✓			
Administer selected groups			✓(None)	✓(None)	
License information					
View	✓	✓			

Table 5: Default settings for predefined Entrust roles (excluding the AutoRA Administrator)

Setting (A=requires authorization)	Security Officer	Administrator	Directory Administrator	End User	Auditor
Modify	✓				
Policy OIDs					
Administer all OIDs	✓	✓			✓
Administer selected OIDs					
Roles					
View	✓	✓			✓
Modify	✓A	A	A		A
Create	✓A	A	A		A
Delete	✓				
Administer all roles	✓				✓
Administer selected roles		✓(End User)	✓(None)	✓(None)	
Searchbase					
View	✓	✓	✓		✓
Modify	✓				
Create	✓				
Delete	✓				
Administer all searchbases	✓	✓	✓		✓
Administer selected searchbases				✓(None)	
Security Policy					
View Security Policy	✓	✓			✓
Modify Security Policy	✓A	A	A		A
Export Certificate Specifications	✓	✓			✓
Import Certificate Specifications	✓				
Export User Templates	✓	✓	✓		✓

Table 5: Default settings for predefined Entrust roles (excluding the AutoRA Administrator)

Setting (A=requires authorization)	Security Officer	Administrator	Directory Administrator	End User	Auditor
Import User Templates	✓				
Force CRLs	✓A				
View User Policy	✓	✓			✓
Modify User Policy	✓A	A	A		A
Create User Policy	✓A	A	A		A
User Templates					
Administer all templates	✓	✓	✓		✓
Administer selected templates				✓(None)	
Users					
View	✓	✓			✓
Add	✓A	✓			
Reactivate	✓A	✓			
Deactivate/Remove	✓A	✓			
Change DN	✓A	✓			
Modify properties	✓A	✓			
Revoke certificates	✓A	✓			
Update key pairs	✓A	✓			
Set for key recovery	✓A	✓A	A		A
Cancel key recovery	✓A	✓			
Modify key update options	✓A				
View activation codes	✓	✓			
Modify OIDs	✓A				
Change user's role	✓A	A	A		A
Modify group membership	✓A	✓			

Table 5: Default settings for predefined Entrust roles (excluding the AutoRA Administrator)

Setting (A=requires authorization)	Security Officer	Administrator	Directory Administrator	End User	Auditor
Import new user	✓A	A	A		A
Export to another CA	✓A	A	A		A
Archive users	✓A	A	A		A
View archived users	✓				✓
Retrieve archived users	✓A	A	A		
Restore information to the Directory	✓A	✓			
Perform PKIX requests	✓	✓			
Create user profile	✓				
Recover user profile	✓				

When to create a role

You can create administrative and non-administrative or "end user" roles. Typically, create an administrative role when

- you need administrative users who have permissions other than what the predefined roles offer
- you want to restrict which groups a role can administer
- you want to restrict which searchbases a role can administer
- you want to restrict which policy OIDs a role can administer
- you want to restrict which roles a role can administer
- you want to restrict which certificate categories or types a role can administer
- you want to restrict which user type templates a role can assign to new users
- you want to change which operations are sensitive

Create an end-user role when you want end users to have different user policies. For instance, suppose you want end users who can roam, and those who can't. Each type of end-user requires different user policies. For details on how to assign user policies to a user role, see "Creating a role" on page 161. You cannot set administrative permissions for end-user roles.

Examples of roles you can create

The following are examples of some custom roles that you might find helpful for administering your PKI. For details on how to create a role, see "Creating a role" on page 161.

Example 1: creating administrative users with basic capabilities

"Basic" administrative users are those that have limited responsibilities. For example, they can only add users within a specific searchbase, or administer certain groups. Members of these roles need not be highly trusted people in your organization.

Table 6 lists permission settings for typical tasks that a basic administrative user could do. To combine tasks, combine the permissions for each task.

Table 6: Example permissions settings for basic administrative user tasks

Task	Description	Permission(s) required
View Security Policy	Can view property sheet for the *Security Policy* entry of the tree view, but can't view sub-entries in tree view.	In *Security Policy*, select *View Security Policy.*
View audit logs	Can view only the *Audit Logs* entry in the tree view, and can do only tasks that appear in the Audit Log menu.	In *Audit Logs*, select *View own logs* or *View all logs.*
View and add users	Can view and add certain kinds of users only.	In *Users: General*, select *View* and *Add*. Elsewhere, select the appropriate certificate categories and types, groups, roles, searchbases, and user templates that can be administered.
View, rename, create, and delete groups	Can administer groups, but can't view or administer group members.	In *Groups*, select *View*, *Rename*, *Create*, and *Delete*. Also select the group(s) that can be administered.
View, rename, create, and delete groups; modify group membership	Can administer groups. Can also see group members and change their group membership.	In *Groups*, select *View*, *Rename*, *Create*, and *Delete*. Also select the group(s) that can be administered. In *User: General*, select *View.* In *User: Advanced*, select *Modify group membership*.
View and modify Directory entries	Can view and modify entries in the Directory Browser.	In *Directory*, select *Bind to Directory* and *View entries*. In *Searchbase*, select *View* and select the searchbases that can be administered.

Table 6: Example permissions settings for basic administrative user tasks (continued)

Task	Description	Permission(s) required
View User Policy, no details	Can view *User Policy* in the tree view but not display each policy's property page.	In *Security Policy*, select *View User Policy*. In *Searchbase*, select the searchbases that can be administered.
View User Policy, with details	Can view *User Policy* in the tree view and display each policy's property page.	In *Security Policy*, select *View User Policy* and *Export Certificate Specification*. In *Searchbase*, select *all searchbases*.

Example 2: creating a regional administrative user role

Suppose you want to administer users according to their geographic region. In each region, you want one or more administrative users who can administer users in that region only. Geographically based administrative users can be helpful when, for instance, your users are dispersed around the world, or are merely in separate buildings at one site.

There are two ways you can create regional administrative users depending on how user entries are organized in your Directory Information Tree (DIT). One way uses searchbases; the other way uses groups. In both ways, you create one or more roles and restrict the users each role can administer.

If your DIT is organized along geographical lines (that is, users in each region are in a distinct branch of your DIT), use the searchbase method. First, create a searchbase for each region of users, then create a role to administer each searchbase. (For more information on searchbases, see the "Setting up the Certification Authority (CA)" chapter in *Using Entrust/PKI 6.0 on Windows*.) This method has trade-offs. The problem is, it may result in too many roles for you to administer conveniently. The advantage is, if you have a role for each searchbase, you can assign different capabilities to each region's administrative users.

If your DIT is not organized along geographical lines (that is, users in each region are scattered throughout the DIT), use the groups method. First, create a group for each region of users you wish to administer. (For more information on groups and how to create them, see the "Setting up the Certification Authority (CA)" chapter in *Using Entrust/PKI 6.0 on Windows*). Then, create just one role and set its groups permission to administer "any group to which they belong". Finally, when you add members to the new administrative role, also add them to one of the regional groups you created. The administrative users will only be able to administer users in the group to which they belong. The advantage of this method is you only need to create one role.

As a variation of the groups method, after you create regional groups, you can create a separate role to administer each group. However, as with the searchbase method, you trade simplicity for flexibility.

Example 3: creating end users with different user policies

You can create end users with different user policies. For example, suppose you want to create two sets of users: one that can roam, and one that can't.

First, create and name two sets of default certificate settings (called "user policies"). In one, allow roaming. In the other, don't. Then, create and name two end-user roles. Assign one policy certificate to the first role, and the second policy certificate to the second role. Finally, assign end users to each of the two new end-user roles.

For information on user policies and how to use them, see "Setting Entrust/PKI user policies" on page 167. For details on how to assign user policies to a user role, see "Creating a role" on page 161.

Viewing roles

Administrative users who have permission to view roles can see a list of roles under *Security Policy* in the tree view in Entrust/RA. The property page associated with a selected role, showing permission settings for the role, appears to the right of the tree view. (See Table 5 for a list of all administrative permissions.)

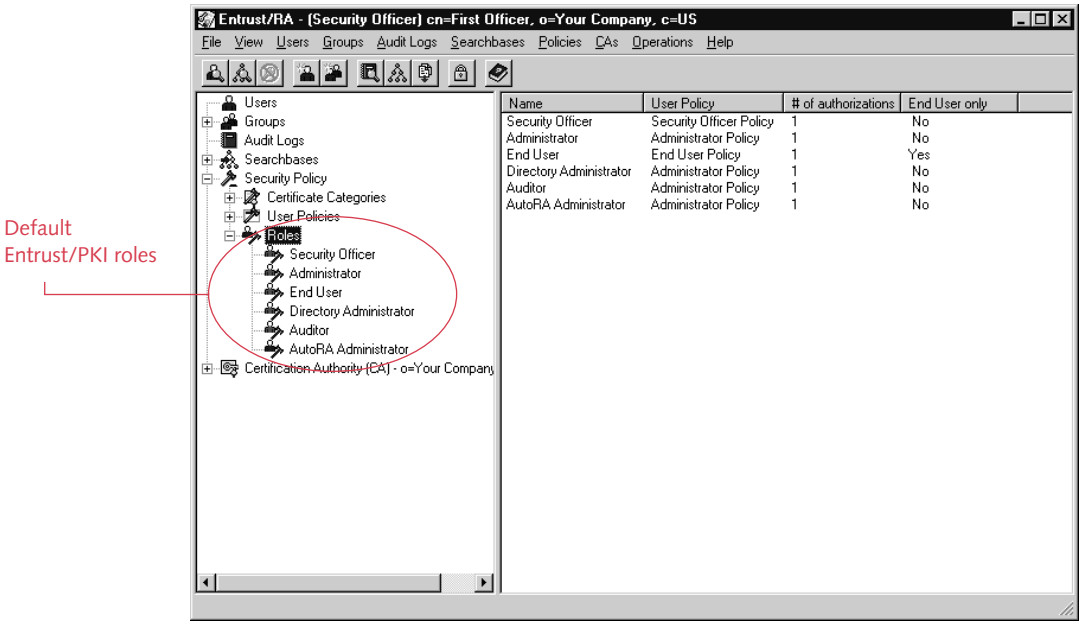
The following properties comprise a role's settings:

- role name
- number of authorizations required for sensitive operations
- default certificate settings (called a "user policy")
- administrative permissions (for roles that aren't end user)
- whether or not the role is end user

To change these settings, you must have sufficient permissions yourself, such as those of a Security Officer.

To view a role and its permissions

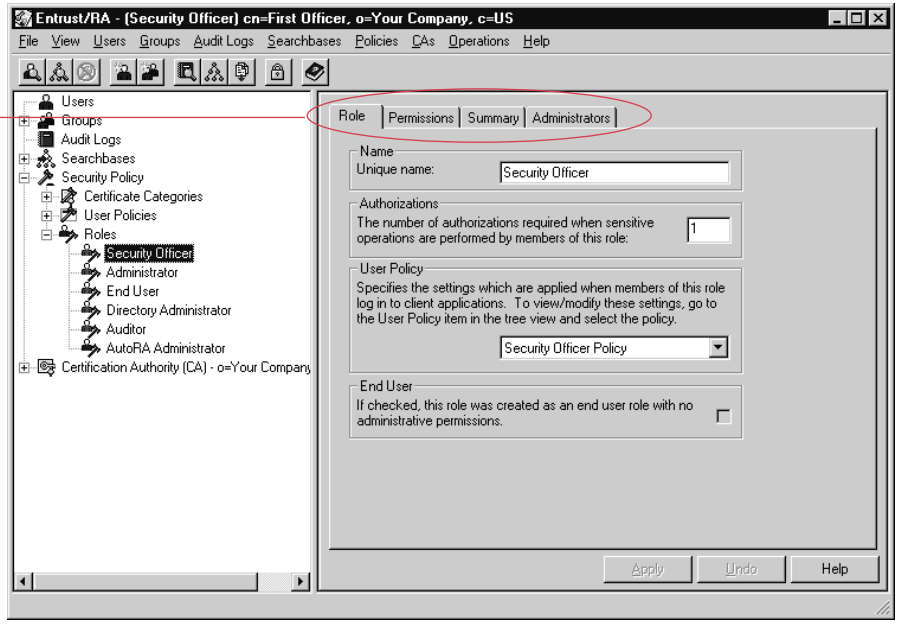
1 Click *Security Policy* in the tree view of Entrust/RA to expand it. If you have sufficient permissions, a *Roles* entry appears in the tree view. Click the *Roles* entry to expand it (a list of roles will appear).



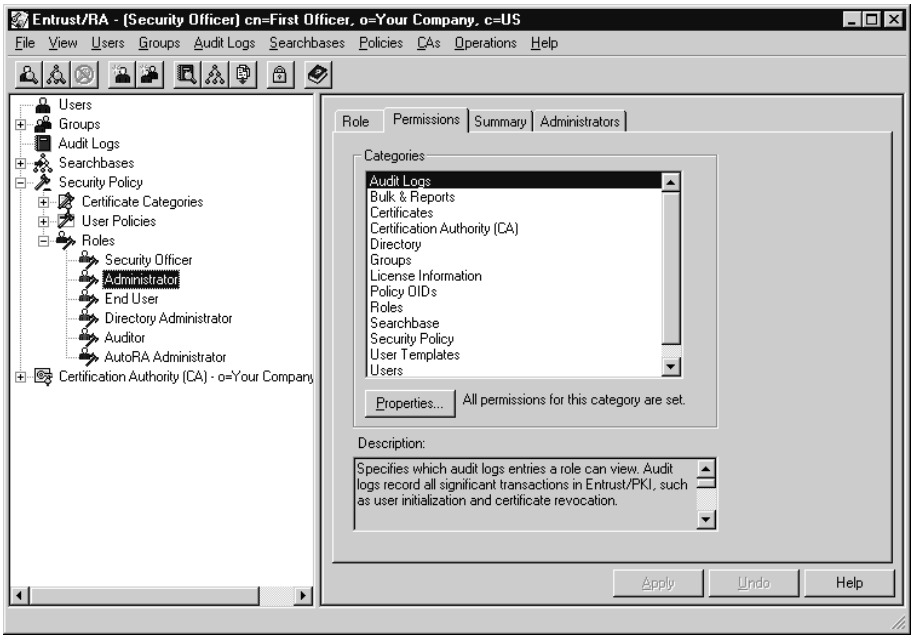
Roles with red icons are Entrust/PKI predefined roles. Regardless of whether they've been renamed (see "Creating a role" on page 161), predefined roles always appear at the top of the list and in the order of Security Officer, Administrator, End User, Directory Administrator, and Auditor. So, if you've renamed "Security Officer" to "Senior Administrator", it still appears first in the list.

Roles with blue icons are custom roles. They appear in the order in which they were created, beneath the predefined roles. (Note that the AutoRA Administrator role has a blue icon. It is a predefined role, but is fully customizable).

2 Click a role. Its property sheet appears on the right.



3 To view individual permissions settings, click the *Permissions* tab.

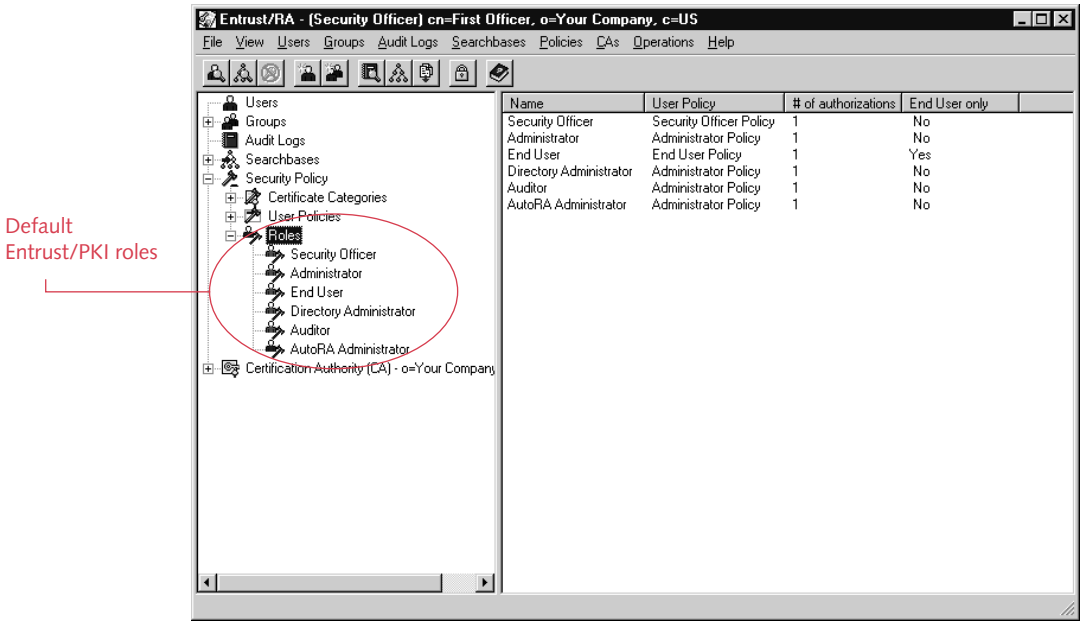


4 Then, double-click a permission category, or click a category and then click *Properties*. A dialog box like the following appears showing the permissions settings for the category you clicked.

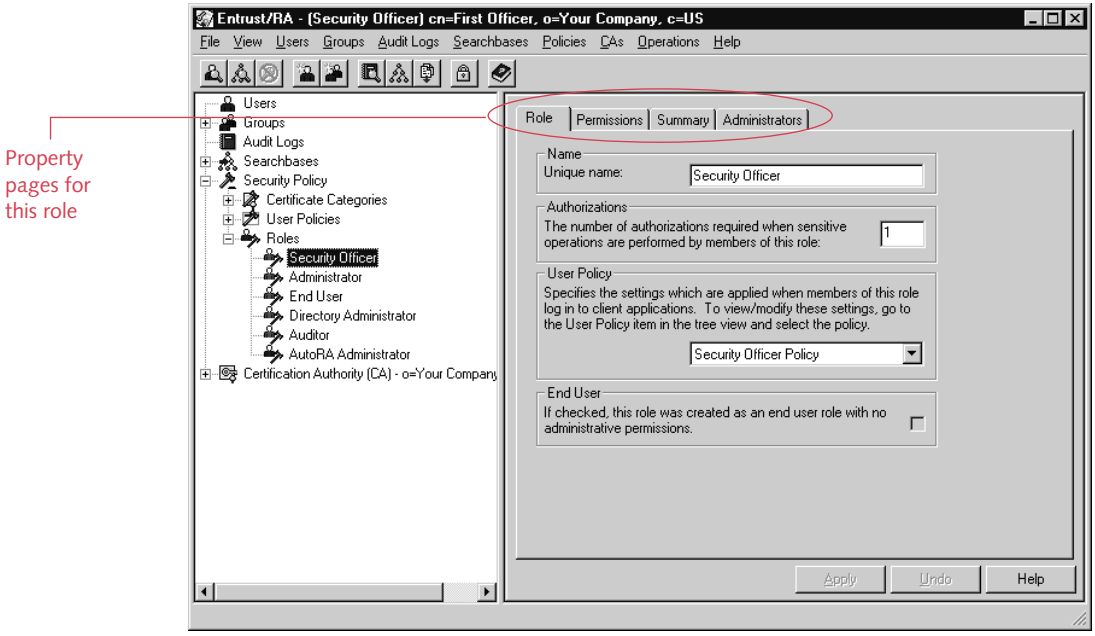
You have now viewed a role.

To view which administators can administer a role

1 Click *Security Policy* in the tree view of Entrust/RA to expand it. If you have sufficient permissions, a *Roles* entry appears in the tree view. Click the *Roles* entry to expand it (a list of roles will appear).



2 Click a role. Its property sheet appears on the right.



3 Click the *Administrators* property page to view all the administrative users who have permission to administer the selected role.

You have now viewed which administrative users can administer a role.

Viewing and saving a record of a role's settings in a file

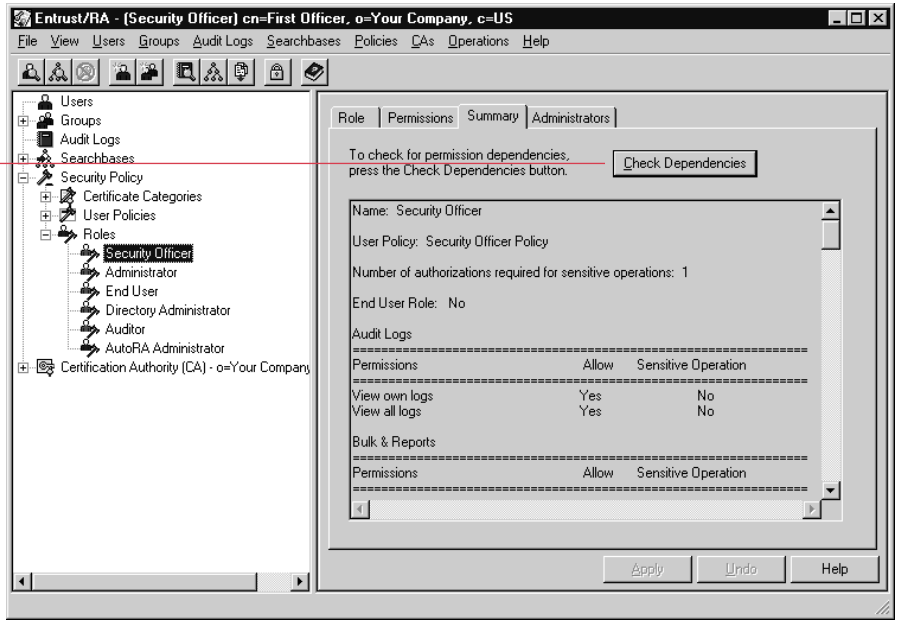
You can review a summary of a role's permissions in the Summary property page. You may want to review the role's capabilities or save them for future reference. An easy way to do this is by saving the role's settings in a plaintext file.

To save a record of a role's settings in a plaintext file

1 Display a role. For details, see "To view a role and its permissions" on page 152.

2 Click the role's *Summary* tab.

The *Summary* property page shows a summary of the role's permissions.



For more information on checking permission dependencies, see "Checking permission dependencies" on page 165.

3 Select and copy text on the *Summary* property page and paste it into a text editor.

4 In the text editor, save the file.

You have now saved a record of a role's settings in a plaintext file.

Descriptions of permission categories

Most permissions allow members of a role to do a specific administrative task, such as add a user to Entrust. Other permissions specify the scope of an operation, for instance, to which groups a role can add users. Also, for most permissions, an additional setting lets you specify whether an operation requires authorization to complete. In Table 5 on page 144, an "A" beside a permission indicates that the operation requires authorization for a given role. For example, changing the settings under *Security Policy* in the tree view of Entrust/RA requires authorization by Security Officers. For details about multiple authorizations, see the "Setting up administrative users" chapter in *Using Entrust/PKI 6.0 on Windows*.

In many cases, it makes sense to combine certain permissions with others. For instance, if you have permission to modify something, you should also have permission to view it. Table 6 on page 149 lists some combinations that provide basic administrative capabilities. You can create advanced capabilities by

combining the permissions for basic capabilities. You can also check a role for dependencies between permissions. For more information, see "Checking permission dependencies" on page 165.

The following sections describe individual permission categories.

Audit Logs

These permissions specify which audit logs you can view. Audit logs record all significant transactions that occur in Entrust/PKI, such as initializing users, revoking certificates, and so on.

If *View own logs* is selected, you can see only the logs you've generated yourself. This function lets you review your own actions, but no one else's.

If *View all logs* is selected, you can see all logs generated in Entrust/PKI.

Bulk and Reports

The *Process bulk files* permission lets you use the Bulk Console (in the menu bar, click Operations > Bulk Console) to process multiple users at one time. For example, if *Process bulk files* is selected, you can run a bulk script to add users to Entrust.

The *Create reports* permission lets you create reports on the Entrust/Authority database. A report can list such things as all added users, all users who have been set up for key recovery, and so on.

Certificates

These permissions specify which categories and types of certificates you can administer. For instance, if you can administer a certificate category or type, it means you can assign it to users. It also means you can view Entrust/PKI license information for that category, provided of course that you also have permission to administer license information. (See "License Information" on page 159 for more information.)

These permissions also determine how many of the settings under *Security Policy* in the tree view you can see. For instance, if you have permission to administer Web certificates only, you can only see the settings that relate to Web certificates.

If you designate no certificate categories and types for a role to administer, that role cannot view or modify any user entries and cannot view any *Security Policy* settings.

Certification Authority (CA)

These permissions specify which operations you can do that are related to your CA, to cross-certified CAs, and subordinate CAs. For instance, in the

cross-certification group of permissions, you can specify whether or not a role can initiate, revoke, or complete cross-certification with another CA. In the subordinate CA group of permissions, you can specify whether or not a role can add a subordinate CA.

Directory

These permissions specify whether you can bind to the Directory or change the Directory password. They also specify whether you can use the Directory Browser tool to create, delete, or modify Directory entries.

Groups

These permissions specify whether you can rename, create, or delete groups, and which groups you can administer. For example, if you can administer a particular group, you can assign users to that group.

Groups are discussed in detail in the "Setting up the Certification Authority (CA)" chapter in *Using Entrust/PKI 6.0 on Windows*.

License Information

These permissions specify whether you can view or modify Entrust/PKI license information (that is, the serial number and user limit) for all of your allowed certificate categories. For example, to change license information for Enterprise certificates, you must be able to administer the Enterprise certificate category. For more information, see "Certificates" on page 158.

License information appears by clicking Operations > License Information in the menu bar.

Policy OIDs

These permissions specify which encryption and verification policy object identifiers (OIDs) you can assign to roles and users, and whether you can create or delete OIDs.

For an explanation of OIDs and how to create them, see the "About Entrust/PKI" and the "Setting up the Certification Authority (CA)" chapters in *Using Entrust/PKI 6.0 on Windows*.

Roles

These permissions specify whether you can view, modify, create, or delete roles, and which roles you can administer.

Searchbase

These permissions specify whether you can modify, create or delete searchbases, and which searchbases of users you can administer. Since searchbases map to the structure of your Directory Information Tree (DIT), you can restrict a role to administering only users who appear in a certain part of the Directory.

Security Policy

These permissions specify which operations that are directly related to *Security Policy* in the tree view. These settings govern how members of your organization use Entrust/PKI to achieve security objectives.

User Templates

These permissions specify which user templates you can select to create new Entrust/PKI users. Templates define user types by specifying what fields appear in the New User dialog box and what information those fields require. Two default user templates are "Person" and "Web Server". If you have no user template available, you cannot create new users using the Entrust/RA graphical user interface; however, you can use bulk processing to do so.

For information on choosing a user template when creating a user, see the "Administering individual users" chapter in *Using Entrust/PKI 6.0 on Windows*. For details on how to create user type templates, see "Modifying the New User dialog box" on page 189.

Users

These permissions specify the operations you can do on individual users and administrative users. User permissions are divided into general and advanced categories.

General user permissions govern common user administration tasks like adding users, changing user DNs, and updating user key pairs. These tasks have few security implications and need not be the kinds of things that only Security Officers can do.

Advanced user permissions govern tasks that have greater security implications than the general tasks. For example, advanced tasks include modifying the set of default OIDs a user receives, changing a user's role, and importing users. These tasks are best suited for highly trusted people in your organization, like Security Officers.

Creating a role

To create a role, you must have sufficient permissions, such as those of a Security Officer. Creating a role is a two-stage process. First you add a role, then you customize its settings.

There are two ways to add a role. One way is to copy an existing role, such as one of the predefined roles. The other way is to start with a "blank" role. Use the first way when you want to create a role that's similar to an existing one. Use the second way when you want to create a role with only a few permissions, or when you want to minimize the chance of granting unintended permissions.

To create a role, see "To create a role by copying an existing role" on page 161 or "To create a new role" on page 162.

Once you've created a role, you customize its settings by changing its name, number of authorizations, user policy, and whether or not the role is for end users. You can also add administrative permissions (but not to an end-user role). To customize a role, see "To change a role's settings" on page 163.

Finally, you should check your role for permission dependencies and adjust the role's permissions accordingly. For more information, see "Checking permission dependencies" on page 165.

If you do not want the role to ever have administrative capabilities, select the end user check box.

Note: Once you select that a role is end user only and click *Apply*, you can't later make the role an administrative role and vice versa.

To create a role by copying an existing role

1. In the tree view, display the list of roles (for details, see "To view a role and its permissions" on page 152).
2. Right-click the role whose settings you wish to copy and click *Copy* in the pop-up menu.

   A copy of the role, with a blue icon, appears in the tree view. The property sheet shows the new role's default settings.
3. Customize the role's settings. For details, see "To change a role's settings" on page 163.
4. Once you've customized the role's permissions, click *Apply* to save them. Or, click *Undo* to cancel your changes.
5. Authorize the operation. For details on authorizing operations, see the "Setting up administrative users" chapter in *Using Entrust/PKI 6.0 on Windows*.

   You have now created a new role.

To create a new role

1. In the tree view, display the *Roles* entry.
2. Right-click *Roles* and click New in the pop-up menu.

   A role with the name *<New Role>* and a blue icon appears in the tree view. The property sheet shows the new role's default settings, which contain no permissions.

Note: If you start to create a new role and change your mind at this point, then you can back out of the operation by right-clicking on Roles again and selecting Refresh from the menu.

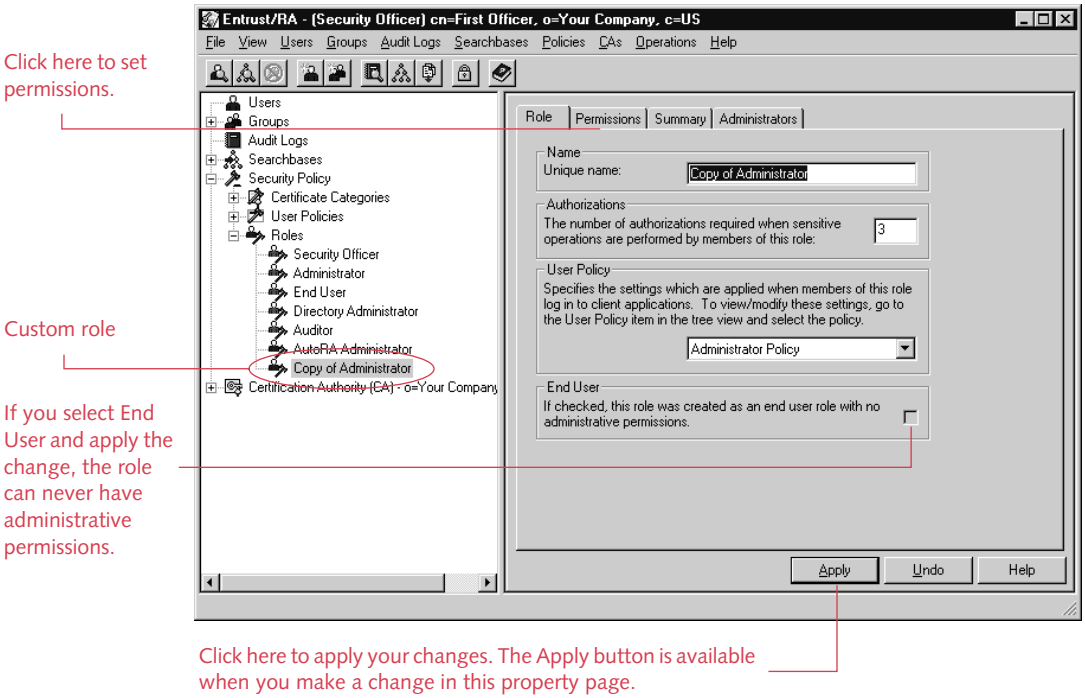
---

3. Set the role's permissions. For details, see "To change a role's settings" on page 163.
4. Once you've set the permissions you want, click *Apply* to save them. Or, click *Undo* to cancel your changes.
5. Authorize the operation. For details on authorizing operations, see the "Setting up administrative users" chapter in *Using Entrust/PKI 6.0 on Windows*.

   You have now created a role.

To change a role's settings

1 Click the role whose settings you want to change. (For details, see "To view a role and its permissions" on page 152.)



2 On the *Role* property page (click the *Role* tab), in the *Unique name* field, enter a unique name for the role. The name must contain at least one character that's not a tab or a space, and must not exceed 225 ASCII characters. For example, valid names are "Dept. 1268 Administrator" and "Senior Administrator's Role". You can rename a role at any time, provided you have the right permissions.

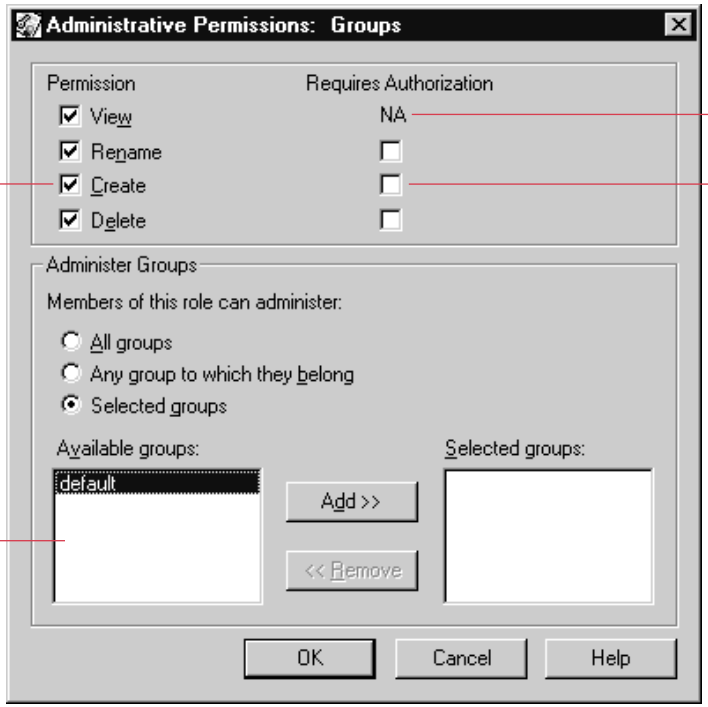
Under *Authorizations*, enter a number. You can set the number of authorizations to be greater than the total number of administrative users, but then there won't be enough administrative users to authorize operations. In such a case, to decrease the number of authorizations, a Security Officer must reset the value. If that's not possible (for instance, because there aren't enough Security Officers), a Master User must reset the number of authorizations for Security Officers, or set Security Officers to key-recovery mode in Entrust/Authority Master Control.

Under *User Policy*, select a user policy in the pull-down list. User Policies define a role's default certificate settings. You can customize default certificate settings and create new ones. For more information, see "Setting Entrust/PKI user policies" on page 167.

3 If you want to give the role permissions, click the *Permissions* tab.

To view permissions properties, either double-click a category or select a category and click *Properties.*

A dialog box like the following appears, showing settings for permissions in the category you selected.



Select a Permission check box to allow an operation.

"NA" means "not applicable": you can't make the operation sensitive.

Select a Requires Authorization check box to make an operation sensitive.

To allow the role to administer a group in this list, select the group and click Add to move it to the Selected list.

To allow an operation, select its *Permission* check box. To disallow an operation, clear its *Permission* check box. To make an operation require authorization to complete, select its *Requires Authorization* check box.

To set some permissions, such as which groups a role can administer, you must move items from an "available" list to a "selected" list. For example, to enable a role to administer selected groups only, click *Selected groups*. Then, in the *Available groups* list, select a group that you want the role to administer and click *Add*. The group appears in the *Selected groups* list. To disable a role from administering a particular group, select the group in the *Selected groups* list and click *Remove*.

Note that although you can set permissions independently of each other, it sometimes makes sense to combine certain permissions. For instance, if you set the permission to modify roles, you should also set the permission to view roles. Table 6 on page 149 lists combinations of permissions that give specific administrative capabilities. You should check for permission dependencies before

you apply your changes. For information on permission categories, see "Descriptions of permission categories" on page 157.

Click *OK*.

4 Repeat Step 3 for permissions in other categories.

5 Click *Apply* to save your changes or *Undo* to cancel.

Note: It's faster to click *Apply* once, after you've made all your changes, rather than each time you change a category of permissions.

6 Authorize the operation. For details on authorizing operations, see the "Setting up administrative users" chapter in *Using Entrust/PKI 6.0 on Windows*.

You have now changed a role's settings. You should now check the role for permission dependencies. See "Checking permission dependencies", next.

Attention: Changes to a role's settings take effect for individual role members the next time the members log in to Entrust/RA. Members that are currently logged in to Entrust/RA when you change their role's settings continue to have their old settings.

Checking permission dependencies

Although you can set permissions independently of each other, you should set some permissions only when you set certain other permissions. For example, if you set the permission to modify groups, you should also set the permission to view groups. Therefore, setting the modify groups permission *depends* on you settings the view groups permission. (Note that the converse is not necessarily true.) Other permissions also have dependencies.

When you set permissions, you can check if there are dependencies with other permissions. If there are dependencies, a dialog box tells you which additional permissions you should set.

Note that when you check permissions for the Directory Browser role, the message that appears recommends that you set other permissions. Ignore this message.

To check permission dependencies

1 Display a role. For details, see "To view a role and its permissions" on page 152.

2 Click the *Summary* tab on the role property page.

A summary of the role's permission settings appears.

3 Click the *Check Dependencies* button.

Entrust checks the role's permissions for dependencies and displays the results in a dialog box.

4 Add or remove permissions (whichever is appropriate) and check dependencies again. Ideally, you should have no dependencies. However, depending on the required capabilities of the role you're checking, one or more dependencies may be unavoidable.

You have now checked a role for its permission dependencies.

Note: If the *Show warnings about operations* setting in File > Preferences > General is turned on, dependencies will be checked automatically whenever you change a role's permissions and click *Apply*.

Setting Entrust/PKI user policies

Every user type in Entrust/PKI is governed by a set of policies. Each set of policies is defined in a certificate called a user policy. (User policies are sometimes also referred to as policy certificates.) When Entrust/PKI is first installed, there are three default user policies: the Security Officer Policy, the Administrator Policy, and the End User Policy. The Security Officer Policy defines a set of policies for Security Officers. The Administrator Policy defines a set of policies for Administrators, Auditors, and Directory Administrators. The End User Policy defines a set of policies for all end users.

Each user policy contains a list of settings that you configure by selecting or clearing check boxes, and by entering information into text fields. When you click in the check box or text field of a given setting, a brief explanation of the setting appears at the bottom of the property page. To find out more about a given policy setting, see "Overview of policy settings" on page 168.

You can change the value of any user policy setting through Entrust/RA. See "Changing a user policy" on page 184. The settings for each user policy are contained within a certificate. Like other certificates, all user policy certificates are stored in the Directory. When you make changes to a user policy, affected users receive an updated user policy certificate the next time they log in to an Entrust-Ready application.

You can also create or copy new user policies. For example, you may want to define a special set of policies for a new administrative role that you have created. You can create new user policies for the predefined roles as well. For example, you may not want Administrators, Directory Administrators, and Auditors all sharing the settings that are defined by the Administrator Policy. Instead, you may want to create new user policies and assign one to each of these roles. See "Creating a new user policy" on page 185.

All of the available settings in the user policies are defined in a special certificate specifications file. It is possible to create additional user policy settings by editing this file. It is strongly recommended that you do not alter the properties of any of the default settings. For more information, see "Customizing policy certificates" on page 306.

Overview of policy settings

This section describes all the policy settings that you can configure in the default user policies.

Note: Entrust recommends that you do not weaken the default password settings described below. Strong password rules are less susceptible to password attacks.

Password expires in (weeks)

The *Password expires in (weeks)* setting designates the length of time a user's Entrust password is valid. When the set amount of time has transpired, the password expires and the user is prompted to create a new password.

You can set password lifetimes from 0 to 52 weeks, or you can set the password lifetime to zero (0), which means the password does not expire. By default, *Password expires in (weeks)* is set to 0 (the password does not expire).

Password history

The *Password History* setting designates the number of unique passwords a user must create before an old password can be used again. If a user is creating a new password, and it matches any of the passwords stored in the password history, the user will be prompted to create a different password.

You can set the password history from 0 to 8 passwords. By default, *Password History* is set to 8 passwords.

Password length (characters)

The *Password length (characters)* setting designates the minimum number of characters that must appear in a user's password.

You can set the minimum password length from 8 to 20 characters. By default, *Password length (characters)* is set to 8 characters. Passwords with less than 8 characters are not supported in pre-4.0 clients. Entrust/Client 3.0 enforces the 8-character minimum.

Note: You can change the range for password length to include a shorter minimum length by customizing the policy certificate. See "Customizing policy certificates" on page 306.

Attention: Entrust Technologies recommends a minimum password length of 8 characters. Shorter passwords may significantly weaken end user security in certain environments. For further information on suitable environments for a lesser value, contact Entrust Support.

Password needs non-alphanumeric character

The *Password needs non-alphanumeric character* setting is a true-or-false setting that designates whether or not users must include a non-alphanumeric character in their passwords. Non-alphanumeric characters include, but are not limited to, the following: ! @ # $ % ^ & *.

By default, users do not require a non-alphanumeric character in their passwords (that is, the *Password needs non-alphanumeric* setting is unselected).

Password needs uppercase letter

The *Password needs uppercase letter* setting is a true-or-false setting that designates whether or not users must include an uppercase letter in their passwords.

By default, users require an uppercase letter in their passwords (that is, the *Password needs uppercase letter* setting is selected).

Password needs lowercase letter

The *Password needs lowercase letter* setting is a true-or-false setting that designates whether or not users must include a lowercase letter in their passwords.

By default, users require a lowercase letter in their passwords (that is, the *Password needs lowercase letter* setting is selected).

Password needs number

The *Password needs number* setting is a true-or-false setting that designates whether or not users must include a number in their passwords.

By default, the *Password needs number* setting is unselected.

Disable single login

The *Disable single login* setting is a true-or-false setting that designates whether or not end users can use the single login feature.

When the *Disable single login* setting is selected, users are required to log in to every Entrust-Ready application they launch. When the *Disable single login* setting is unselected, users can log in once to an Entrust-Ready application, and

have Entrust/PKI automatically secure each subsequent Entrust-Ready application they launch, without requiring the users to enter their Entrust password again.

By default, only End Users can use the single login feature (the *Disable single login* setting is unselected). Security Officers, Administrators, Directory Administrators, and Auditors are not allowed to use single login due to security reasons (that is, the *Disable single login* setting is selected in the Security Officer Policy and the Administrator Policy).

Login timeout (minutes)

The *Login timeout (minutes)* setting designates how many minutes of inactivity must pass before the user is automatically logged out of an Entrust-Ready application. You can set the *Login timeout (minutes)* setting from 0 to 300 minutes.

In some cases, you may not want Entrust-Ready applications to time out. For example, an Entrust-Ready application that prints cheques automatically will stop running if the application times out. If you want Entrust-Ready applications to not time out, set the *Login timeout (minutes)* setting to "0". Note, however, that setting *Login timeout (minutes)* to "0" does not apply to Entrust/RA. If *Login timeout (minutes)* is set to "0", Entrust/RA times out according to the defaults.

By default, *Login timeout (minutes)* is set to 10 minutes in the Security Officer Policy and the Administrator Policy, and in the End User Policy the default is set to 15 minutes.

Note: Users of Entrust-Ready applications can choose a timeout that is less than or equal to the *Login timeout (minutes)* setting specified in the End User Policy.

Symmetric encryption algorithms

The *Symmetric encryption algorithms* setting designates which algorithms users can employ for encrypting data. When configuring this setting, it is important to keep in mind which encryption algorithms are supported in the Entrust-Ready end-user application. You must include in the *Symmetric encryption algorithms* setting at least one algorithm that is supported by the Entrust-Ready end-user application. For example, if you specify IDEA-128 as the only allowed encryption algorithm, and the Entrust-Ready end-user application does not support IDEA-128, users will not be able to encrypt.

Enter any of the following algorithms *exactly as shown* into the *Symmetric encryption algorithms* text field: ALL, CAST-128, CAST-80, CAST-64, CAST-40, IDEA-128, AES-256, TRIPLE-DES, DES, RC2-128Compatible, RC2-40Compatible. (Use spaces instead of commas when entering values into the *Symmetric encryption algorithms* text field.)

The value *ALL* allows users to encrypt using any of the algorithms from this list. If the value *ALL* is specified, the default encryption algorithm is CAST-128.

If you enter more than one value into the *Symmetric encryption algorithms* field, the first value entered is taken to be the default. For example, if you enter the values "CAST-128 TRIPLE-DES IDEA-128" into the *Symmetric encryption algorithms* field, CAST-128 is set as the default encryption algorithm, and CAST-128, TRIPLE-DES, and IDEA-128 are the algorithms users are permitted to use for encrypting data. The default established here can be overridden by users, as long as the algorithm chosen by the user matches one of the permitted algorithms listed in the *Symmetric encryption algorithms* field.

Algorithm for digital signature

The *Algorithm for digital signature* setting designates which algorithms users can employ for creating the signing private key. When configuring this setting, it is important to keep in mind which signature algorithms are supported in the Entrust-Ready end-user application. You must include in the *Algorithm for digital signature* setting at least one algorithm that is supported by the Entrust-Ready end-user application. For example, if you specify ECDSA-192 as the algorithm, and the Entrust-Ready end user application does not support ECDSA-192, users will not be able to create signing private keys.

Using uppercase letters only, type one of the following algorithms in the *Algorithm for digital signature* field: RSA-1024, RSA-2048, DSA-1024, or ECDSA-192.

When you are creating a user policy for single-key-pair users, you must specify either RSA-1024 or RSA-2048 for this setting. If you choose an algorithm other than RSA-1024 or RSA-2048, users you create with this user policy will not be able to create their Entrust profile. Of the available algorithms, only RSA supports both encryption and signing. For more information, see "Number of key pairs" on page 180.

When you select *Allow CAPI Export*, the policy setting for *Algorithm for digital signature* can be only RSA-1024, RSA-2048, or DSA-1024. ECDSA-192 is not supported for CryptoAPI profile export. If you choose ECDSA-192, users you create with this user policy will ignore the *Allow CAPI Export* policy setting silently. See "Allow CAPI Export" on page 179.

Message in Entrust-Ready clients

The *Message in Entrust-Ready clients* setting allows you to present a message to users when they log in to an Entrust-Ready application. This setting is ideal for conveying important Entrust/PKI system administration information to end users, who will see the message as soon as they log in to an Entrust-Ready application.

By default, the *Message in Entrust-Ready clients* setting is blank.

Permit roaming

The *Permit roaming* setting designates whether or not users can download their profile from the Entrust/ProfileServer. When *Permit roaming* is selected, users can access Entrust-Ready applications by downloading their user profile from the Entrust/ProfileServer, regardless of where they are located or what machine they are using. When *Permit roaming* is unselected, users can only access Entrust-Ready applications with a user profile that is stored locally on disk or on a token. If the *Permit roaming* setting is unselected, the *Permit desktop* setting must be selected.

By default, the *Permit roaming* setting is selected. If you unselect it, you must select the *Permit desktop* setting (which is selected by default).

For more information, see the *Entrust/ProfileServer* guide.

Permit desktop

The *Permit desktop* setting designates whether or not users have a user profile that is stored locally on disk or on a token.

You may wish to unselect the *Permit desktop* setting for roaming users, because this reduces the risk of roaming users losing their locally stored user profile when they are working outside the office.

By default, the *Permit desktop* setting is selected. If you unselect it, you must select the *Permit roaming* setting (which is selected by default).

Enforce token usage

The *Enforce token usage* setting designates whether or not users must use a hardware token to access Entrust-Ready applications.

Enforcing token usage increases the level of access control to Entrust-Ready applications. When the *Enforce token usage* setting is selected, users must have possession of a hardware token, as well as knowledge of their Entrust password, in order to access Entrust-Ready applications. For example, you might want to enforce token usage for all administrative users who have access to the Entrust/RA application.

If the *Enforce token usage* setting is selected, the *Permit desktop* setting must also be selected. By default, the *Enforce token usage* setting is unselected.

If *Number of key pairs* is set to 1, *Enforce token usage* must be unselected. The use of hardware tokens is not supported for single-key-pair users. If you select this setting for a single-key-pair user, the user will not be able to create their Entrust profile. See "Number of key pairs" on page 180.

If *Allow PKCS#12 Export* is selected, *Enforce token usage* must be unselected. Users who use a hardware token cannot export their profiles in PKCS #12 format. See "Allow PKCS #12 Export" on page 178.

Allow personal address book use

The *Allow personal address book use* setting designates whether or not users can maintain personal trust relationships with individual users inside their CA domain, including users in cross-certified CAs. These relationships are maintained using a personal address book.

By default, the *Allow use of personal address book* setting is selected.

Allow CA personal address book use

The *Allow CA personal address book use* setting applies to the IPSecNegotiator Toolkit. This setting designates whether or not users can maintain personal trust relationships with other, uncross-certified CA domains. Allowing users to maintain a personal trust relationship with an uncross-certified CA domain means you allow your users to trust all the users in the other, uncross-certified CA domain. These relationships are maintained using a Certification Authority personal address book.

More information about the IPSec Negotiator Toolkit appears on the Entrust Support Extranet. To register, see www.entrust.com/xtranet.

By default, the *Allow use of CA personal address book* setting is unselected.

Permit Server Login usage

Enable this setting to permit users to create the encrypted password file for use with Server Login. Users who currently use this file will still be able to use Server Login as long as they don't change their computer or password.

By default, this setting is disabled.

Enforce identity usage

When enabled, this setting lets users create their Entrust profile using an identity device, for example, a BioMouse.

By default, this setting is disabled.

DN encoding format

The *DN encoding format* setting designates what format Entrust-Ready applications use to encode certain attributes of the DN. Only those attributes that are defined with the DirectoryString syntax (described in the X.520 standard) are encoded using the specified format. DN attributes are encoded, for instance, during S/MIME or PKIX-CMP operations.

When DN encoding takes place, if PrintableString is specified in the user policy, and the attribute values to be encoded fall within the PrintableString range of characters (that is, a-z A-Z 0-9 ' ( ) + , - . / : = ? space) then PrintableString is used. If a character falls outside this range, and TeletexString is specified, the

Entrust-Ready application attempts to encode using TeletexString (a character set that fits within the Latin-1 range). If TeletexString is not specified, or a character falls outside its range, the Entrust-Ready application attempts to encode using UTF8, if it is specified (UTF8 is a special encoding of Unicode, and is the preferred method for encoding format). If UTF8 is not specified, the Entrust-Ready application attempts to encode using BMPString, if it is specified (the BMPString format is another form of Unicode). If a character falls outside the range of PrintableString and TeletexString, and neither UTF8 nor BMPString is specified, the Entrust-Ready application reports an error.

The *DN encoding format* setting allows you to disable formats that are now deprecated, although you may need to maintain deprecated formats to remain compatible with legacy systems, for example. The formats that are currently deprecated are TeletexString and BMPString.

By default, the *DN encoding format* setting supports the PrintableString, TeletexString, and UTF8 encoding formats.

Perform Directory consistency check

Enable this setting for a Directory consistency check. By default, the *Perform directory consistency check* setting is enabled.

The *Perform directory consistency check* setting designates whether the DN in a certificate is checked against the DN of the Directory entry that contains that certificate. Each certificate has a field (the subject name field) that uses a DN to name the subject of the certificate. DNs are also used to designate the location of each entry in a Directory. Often, these two DNs have the same value. However, there are all kinds of situations where these two values could be different. For example, a local CA service provider may have a public Directory that lists all the organizations in your area, and you might find your certificate located in this Directory under cn=your name, ou=your company, ou=software companies, o=your town's public directory. This DN indicates the structure of the public Directory, and indicates the location of your entry in that Directory. However, your certificate was generated by your company's CA, and the DN in your certificate's subject name field would be different from the DN of your entry in the public Directory. For example, the subject name DN might be cn=your name, ou=marketing, o=your company, c=US. In a case like this, or in any other case where the DN of the Directory entry is different from the DN in the certificate subject name field, you must disable the *Perform directory consistency check* setting.

When the *Perform directory consistency check* setting is enabled, the DN of the Directory entry is compared against the DN of the certificate subject name. If, for a given user, the values of these two DNs are found to be different, then you cannot encrypt for that user.

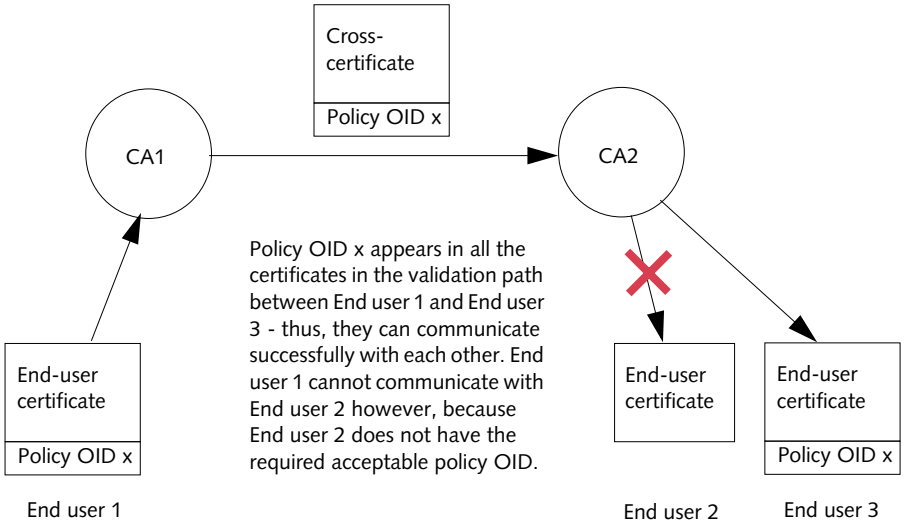
Enforce S/MIME

The *Enforce S/MIME* setting designates whether or not users are required to communicate using S/MIME (Secure Multipurpose Internet Mail Extensions) in Entrust/Express.

By default, the *Enforce S/MIME* setting is unselected.

Acceptable policy OIDs

The *Acceptable policy OIDs* setting allows you to enforce policies for users in your CA domain. You can use policy OIDs for just about anything, such as dictating how users communicate with each across different CA domains.



For example, a CA that you are cross-certified with might have user policies with varying grades of security for its different end-user roles. To be added to a role with a user policy of high security, users must go through a security check. Users who don't go through a security check are assigned to a role with a lower level of security in the user policy. Therefore, the policy of this CA is that for users to be highly trusted, they must undergo a security check. This policy can be represented numerically as an OID (that is, an object identifier). Let's say the OID for this policy is 2.6.7.1.47.98.2. An administrative user can include policy OID 2.6.7.1.47.98.2 in the encryption certificate, the verification certificate, or both certificates, of highly trusted users. The administrative user can then designate policy OID 2.6.7.1.47.98.2 as a policy that must be checked for. This is done by including the OID in the *Acceptable policy OIDs* setting of the user policy for highly trusted users. The user policies of lower security, on the other hand, would not have this OID included in the *Acceptable policy OIDs* setting.

You might decide that users belonging to a highly trusted role in your CA domain should only communicate with highly trusted users in the other CA domain. To enforce this measure for your highly trusted users, you would select the *Require policy* check box in the *End User Policy*, enter policy OID 2.6.7.1.47.98.2 in the *Acceptable policy OIDs* field, and use the certificate specification file to add policy OID 2.6.7.1.47.98.2 to your CA's cross-certificate. (The cross-certificate makes up a part of the validation path between users in different CA domains, so the cross-certificate must also contain the required policy OID.) Now users in your CA domain would only be able to communicate successfully with users in the other CA domain that also had policy OID 2.6.7.1.47.98.2 in their policy certificate.

By default, no policy OIDs are included in the *Acceptable policy OIDs* setting. To enforce checking of policy OIDs, the *Require policy* setting must be selected (see "Require policy" on page 176). Policy OIDs can have from 0 to 32 arcs. The arcs are the sets of numbers between the decimal points. For instance, our example policy OID, 2.6.7.1.47.98.2, has seven arcs. For more information on OIDs, see the "About Entrust/PKI" chapter in *Using Entrust/PKI 6.0 on Windows*.

For information on adding policy OIDs to cross-certificates, see "Customizing cross-certificates" on page 314.

Do not process policy mappings

The *Do not process policy mappings* setting disables policy mapping.

Policy mapping can take place between organizations that have the same policy, but use different policy OIDs to represent this policy. For example, Company A may have a policy that states highly trusted users are required to undergo a security check. CA domain A represents this policy using policy OID 2.6.7.1.47.98.2. CA domain B has the same policy, but represents it using policy OID 2.6.7.1.47.98.4. It is possible for CA domain A to map OID 2.6.7.1.47.98.2 to OID 2.6.7.1.47.98.4 so users in CA domain A can communicate with users in CA domain B. However, CA domain B may decide to disable policy mapping to ensure stricter security. Policy mapping can be selected or unselected using the *Do not process policy mappings* setting.

By default, the *Do not process policy mappings* setting is unselected.

Require policy

The *Require policy* setting works in conjunction with the *Acceptable policy OIDs* setting. If you want Entrust-Ready applications (such as Entrust/Entelligence) to check for and validate the list of acceptable policy OIDs, you must select the *Require policy* setting.

By default, the *Require policy* setting is unselected.

PKIX compliance

The *PKIX compliance* setting allows you to control the set of certificates relied upon by your users with respect to PKIX compliance. If the *PKIX compliance* setting is selected, checks are done on each certificate relied upon by your users. Should one or more of these checks fail, users cannot rely upon the certificate because it was not issued in compliance with PKIX.

By default, the *PKIX compliance* setting is unselected.

For more information on PKIX compliance in Entrust/PKI, see "PKIX and FPKI compliance" on page 603.

FPKI compliance

The *FPKI compliance* setting allows you to control the set of certificates relied upon by your users with respect to FPKI compliance. If the *FPKI compliance* setting is selected, checks are done on each certificate relied upon by your users. Should one or more of these checks fail, users cannot rely upon the certificate because it was not issued in compliance with FPKI.

By default, the *FPKI compliance* setting is unselected.

For more information on FPKI compliance in Entrust/PKI, see "PKIX and FPKI compliance" on page 603.

Do not process anyPolicy

The *Do not process anyPolicy* setting is used to control the handling of the "any-policy" certificate policy identifier. The extension consists of a single integer which indicates the number of additional certificates after which the any-policy policy identifier will be inhibited. As long as "any-policy" is not inhibited, the identifier is considered a match for any policy. When the "any-policy" policy identifier is inhibited, it is not considered a match for other policies when processing the certificate policies extension, unless the certificate in which it appears is self-issued. This setting can be used to inhibit "any-policy" at the beginning of certificate validation.

By default, this setting is false (unselected).

Policy Certificate expires in (days)

The *Policy Certificate expires in (days)* setting allows you to specify how long a user policy (which is defined in a policy certificate) is valid.

A user's policy certificate is refreshed every time the user logs in to Entrust/PKI. If any changes have been made to the policy certificate, these are picked up by the user when they log in.

Users who are working offline can continue to work offline for as long as their policy certificate is valid. The longer a policy certificate is valid, the longer users

associated with that policy certificate can work offline. At the same time, the longer a policy certificate is valid, the longer you are allowing users to work offline with potentially out-of-date user policies. A balance must be struck between allowing users to work offline and maintaining tight control of your user policies.

You can set the *Policy Certificate expires in (days)* setting from 1 to 3650 days. By default, all policy certificates expire after 30 days.

HTTP Proxy for CRL Requests

The HTTP Proxy is an http address through which HTTP requests for CRLs will be sent. It defines the HTTP proxy used for retrieving CRLS via HTTP.

For example, if this is set to http://crlserver.company.com/, and a certificate has a CRL Distribution point showing the CRL at http://crl.somewhereelse.com/crlfile.crl, then the Entrust client software will send a request to http://crlserver.company.com/ for http://crl.somewhereelse.com/crlfile.crl. If this setting is not present, the client software will send the request for crlfile.crl directly to http://crl.somewhereelse.com/.

Note that this setting is only available after the policy certificate is verified, meaning that the proxy setting may not actually be available for the first CRL requests.

By default, this setting is disabled (blank).

Allow PKCS #12 Export

The *Allow PKCS #12 Export* setting allows you to specify whether a user will be able to export their profile in PKCS #12 format.

Note that before a user can export their profile, one or both of their user certificates must also contain an extension allowing profile export. See "Allowing profile export" in *Using Entrust/PKI 6.0 on Windows* for details.

If *Allow PKCS #12 Export* is selected, *Enforce token usage* must be unselected. Users who use a hardware token cannot export their profiles in PKCS #12 format. See "Enforce token usage" on page 172.

Minimum PKCS #12 Hash Count

The *Minimum PKCS #12 Hash Count* setting allows you to specify the minimum number of times that the user-supplied password for the PKCS #12 file will be hashed during export. A value of 1 for this setting is very weak, 2000 is acceptable, and 10000 is highly secure.

Note that the default minimum hash count setting (2000) may prevent older browsers from reading the PKCS #12 file. If you want to enable users to specify a lower hash count so that older browsers can read the exported files, consider

setting the minimum hash count to 1. More recent browsers support either a hash count of 2000 or variable hash counts.

By default, this setting is 2000.

Allow CAPI Export

The *Allow CAPI Export* setting allows a user to export their profile to Microsoft CryptoAPI (Cryptographic Application Programming Interface). This feature allows a user to synchronize their profile with CryptoAPI so that the user's certificates and private keys can be imported into the Microsoft CryptoAPI store. The user's certificates and private keys will be imported to the CryptoAPI store for use with CryptoAPI-enabled applications (such as Microsoft Outlook Express and Microsoft Internet Explorer).

If this setting is enabled and *Allow unprotected CAPI Export* is unselected, the user will be prompted to select a level of protection for their exported keys.

If this setting is enabled, you can also enable the *Allow CAPI Key Export* (see "Allow CAPI Key Export" on page 180).

Note that before a user can export their profile, one or both of their user certificates must also contain an extension allowing profile export. See "Allowing profile export" in *Using Entrust/PKI 6.0 on Windows* for details.

When you select *Allow CAPI Export*, the policy setting for *Algorithm for digital signature* can be only RSA-1024, RSA-2048, or DSA-1024. ECDSA-192 is not supported for CryptoAPI profile export. If you choose ECDSA-192, users you create with this user policy will not be able to create their Entrust profile. See "Algorithm for digital signature" on page 171.

CryptoAPI profile export is supported for roaming users, but is not recommended in all situations. Roaming typically mandates a zero footprint on the client machine, while CryptoAPI is based on local storage of certificates and keys. CryptoAPI export will be executed whenever an Entrust user logs in using a roaming profile. The certificates and keys imported into the CryptoAPI store will remain once the user has logged out.

When deciding whether to enable CryptoAPI export for a roaming user, consider the following:

- If the roaming user consistently logs in to their personal Windows account on one or more machines, and the operating system is Windows NT or Windows 2000, then it is acceptable to allow CryptoAPI export for the user.
- If roaming users are sharing Windows accounts (in a kiosk situation or on home computers) or are using Windows 95, Windows 98 or Windows ME (where Windows accounts are not strongly protected), Entrust Technologies does not recommend allowing CryptoAPI export. In this case, the *Allow CAPI Export* setting should be disabled.

By default, this setting is disabled.

Allow unprotected CAPI Export

The *Allow unprotected CAPI Export* setting specifies that when a user exports their profile to Microsoft CryptoAPI (Cryptographic Application Programming Interface), they will export the keys unprotected. When this setting is selected, the user is not prompted to enter any further information during the export operation.

When this setting is selected, *Allow CAPI Export* must also be selected.

CryptoAPI profile export is not supported for roaming users.

By default, this setting is disabled.

Allow CAPI Key Export

The *Allow CAPI Key Export* setting allows you to specify whether Microsoft CryptoAPI (Cryptographic Application Programming Interface) can export a user's keys to another application. Enabling this setting only takes effect if the *Allow CAPI Export* setting is enabled (see "Allow CAPI Export" on page 179).

By default, this setting is disabled.

Number of key pairs

The *Number of key pairs* setting allows you to specify whether a user will have one or two key pairs in their Entrust profile.

When this setting is changed for an existing user, the change does not take effect when the user next logs in to an Entrust-Ready application. When changing this setting to "1" (that is, when changing a dual-key-pair user to a single-key-pair user), there are several ways that this change can take effect in the user's profile:

- Recover the user's keys (see "Recovering a user's keys" in *Using Entrust/PKI 6.0 on Windows*).
- Initiate a DN change for the user (see "Changing a DN" in *Using Entrust/PKI 6.0 on Windows*).
- Wait until the user's keys update automatically (see "Automatic key update" in *Using Entrust/PKI 6.0 on Windows*).

Before changing the user from a dual-key-pair user to a single-key-pair user, you can also revoke the user's encryption certificate or update the user's keys. This will ensure that the Entrust-ready application generates a new key pair for the user and requests a new dual-usage certificate from Entrust/Authority. See "Revoking a user's certificates" and "Updating a user's key pairs" in *Using Entrust/PKI 6.0 on Windows*.

When changing this setting to "2" (that is, when changing a single-key-pair user to a dual-key-pair user), there are several ways that this change can take effect in the user's profile:

- Recover the user's keys (see "Recovering a user's keys" in *Using Entrust/PKI 6.0 on Windows*).
- Initiate a DN change for the user (see "Changing a DN" in *Using Entrust/PKI 6.0 on Windows*).
- Wait until the user's keys update automatically (see "Automatic key update" in *Using Entrust/PKI 6.0 on Windows*).

When you set a user as a single-key-pair user, the policy setting for *Algorithm for digital signature* can be only RSA-1024 or RSA-2048. If you choose an algorithm other than RSA-1024 or RSA-2048, users you create with this user policy will not be able to create their Entrust profile. Of the available algorithms, only RSA supports both encryption and signing. See "Algorithm for digital signature" on page 171.

If *Number of key pairs* is set to 1, *Enforce token usage* must be unselected. The use of hardware tokens is not supported for single-key-pair users. If you select this setting for a single-key-pair user, the user will not be able to create their Entrust profile. See "Enforce token usage" on page 172.

By default, this setting is set to "2".

See "Adding single-key-pair users" in *Using Entrust/PKI 6.0 on Windows* for more details.

Prevent single login register

The *Prevent single login register* setting allows you to prevent or allow users from registering an Entrust-Ready application for Single Login.

When this setting is enabled, users are not allowed to manually register Entrust-Ready applications through the Entrust Security Warning dialog box. All Entrust-Ready applications have to be automatically registered for those users. See also "Delay single login register" on page 181.

By default, this setting is disabled.

Delay single login register

The *Delay single login register* setting allows you to delay users from registering an Entrust-Ready application for Single Login.

When this setting is enabled, users will need to wait 5 seconds before they can register the application for single login. This setting is useful if you want to ensure that users understand the implications of registering an application for Single Login before doing so. See also "Prevent single login register" on page 181.

By default, this setting is disabled.

Maximum bad login attempts

The *Maximum bad login attempts* setting allows you to configure the number of bad password attempts to allow before a suspension of a user profile is issued. You can also configure the time window in which to record bad password attempts (see "Login attempt window" on page 182).

By default, this setting is set to 0 (or disabled).

For more information on the this and the Maximum login attempts setting, see the "Password Attempt Management" section in the *Entrust Desktop Solutions Administrator's Guide*.

Login attempt window

The *Login attempt window* setting allows you to configure the time window in which to record bad password attempts as set in "Maximum bad login attempts" on page 182.

By default, this setting is set to 1 minute.

For more information on the this and the setting "Login timeout (minutes)" on page 170, see the "Password Attempt Management" section in the *Entrust Desktop Solutions Administrator's Guide*.

ICE settings signed

The *ICE settings signed* setting requires your ICE Administrator settings to be signed before they will be used.

By default, this setting is disabled.

ICE settings ignored

The *ICE settings ignored* setting instructs Entrust/ICE to ignore any ICE Administrator settings.

By default, this setting is disabled.

Enable the use of an ARL cache

The *Enable the use of an ARL cache* setting allows you to cache ARLs.

By default, this setting is enabled (ARLs are cached).

Enable the use of a CRL cache

The *Enable the use of a CRL cache* setting allows you to cache CRLs.

By default, this setting is enabled (CRLs are cached).

Enable the use of a XCert cache

The *Enable the use of a XCert cache* setting allows you to cache cross-certificates.

By default, this setting is enabled (cross-certificates are cached).

Enable the use of a Cert cache

The *Enable the use of a Cert cache* setting allows you to cache user certificates.

By default, this setting is enabled (user certificates are cached).

Secure Delivery SMTP

The *Express Secure Delivery* setting allows you to integrate users of Entrust/Express 6.0 with the ZixIt SecureDelivery service. Entrust/Express requires the SMTP address of the ZixIt server in order to send mail to the server and find a certificate in the Directory. Enter the SMTP address of the ZixIt server here.

By default, this setting is disabled (there is no server address listed).

Content Scanner SMTP

The *Content Scanner SMTP* setting allows you to use content scanning software with Entrust/Express. Entrust/Express can be configured to decrypt all encrypted outbound e-mail messages and scan them for content improprieties as dictated by your organization's content policy. In order to do so, the outbound message is automatically encrypted for the e-mail gateway and the gateway is added to the distribution list.

Each time an outbound message is copied to the default address, a window appears to inform the user that the message was copied to the address in question. The user can select a checkbox so that this window will not be displayed in the future.

In order to use this feature, your organization must purchase and set up a server-based content scanning application capable of handling encrypted messages.

By default, this setting is disabled.

For more information on the this and the Login attempt period setting (below), see the "Content Scanning" section in the *Entrust Desktop Solutions Administrator's Guide*.

Express Search Source Order

The *Express Search Source Order* setting allows you to define the default certificate search order. In Entrust/Express, the default certificate search order is:

- Certificate cache

- X.500 Directory (Directory)
- Entrust Address Book (EAB)
- Exchange Global Address List (GAL)

The certificate cache will always be searched first, but Entrust/Express 6.0 lets you allow users to change the search order of the other three sources.

By default, this setting is "Directory EAB GAL".

Searchbase Search Order

The *Searchbase Search Order* setting allows you to choose the searchbase sequence when client applications attempt to search for users in Entrust/Authority. Enter the friendly names of the searchbases and separate each with a semi-colon. For example, to search through the Finance searchbase first, then the Develpment searchbase, and, finally, the HR searchbase, enter the following text:

Finance;Development;HR

Leave this setting blank to search through all searchbases.

Changing a user policy

The following procedure describes how to change policy settings through the Entrust/RA application. You must be a Security Officer, or otherwise have the appropriate privileges, to change user policies.

Changes made to a user policy take effect as soon as you click *Apply* at the bottom of the given user policy property page. Users affected by changes to a user policy must log in to an Entrust-Ready application, after the changes are complete, to begin using the updated user policy.

Certain user policy settings (for example, *Number of key pairs* and *Algorithm for digital signature*) do not take effect when the user next logs in to an Entrust-Ready application. Rather, they take effect when the user's keys are updated. Key update occurs if you

- recover the user's keys (see "Recovering a user's keys" in *Using Entrust/PKI 6.0 on Windows*)
- initiate a DN change for the user (see "Changing a DN" in *Using Entrust/PKI 6.0 on Windows*)
- wait until the user's keys update automatically (see "Automatic key update" in *Using Entrust/PKI 6.0 on Windows*)

To modify a user policy

1. Log in to Entrust/RA as a Security Officer (see the "Setting up the Certification Authority (CA)" chapter in *Using Entrust/PKI 6.0 on Windows*).

Entrust/RA appears.

2 Double-click *Security Policy* in the tree view, and then double-click *User Policies*. The available policy certificates appear below *User Policies*.

3 Click the policy certificate you want to modify. All the policy certificate information that you can modify appears in the right pane. Make sure the *General Information* property page is selected.

4 Scroll through the *Policy Attributes* to find the settings you want to modify. Select, unselect, or enter the appropriate information for the settings you wish to modify. For more information on a particular setting, see "Overview of policy settings" on page 168.

5 Click *Apply* when you have finished making changes. If you want to view the ASN.1 decoding of the policy certificate and see the results of your changes, click the *Certificate Contents* property page.

You have now modified a policy certificate.

Creating a new user policy

The following procedure describes how to create new user policies through the Entrust/RA application. You must be a Security Officer, or otherwise have the appropriate privileges, to create a new user policy.

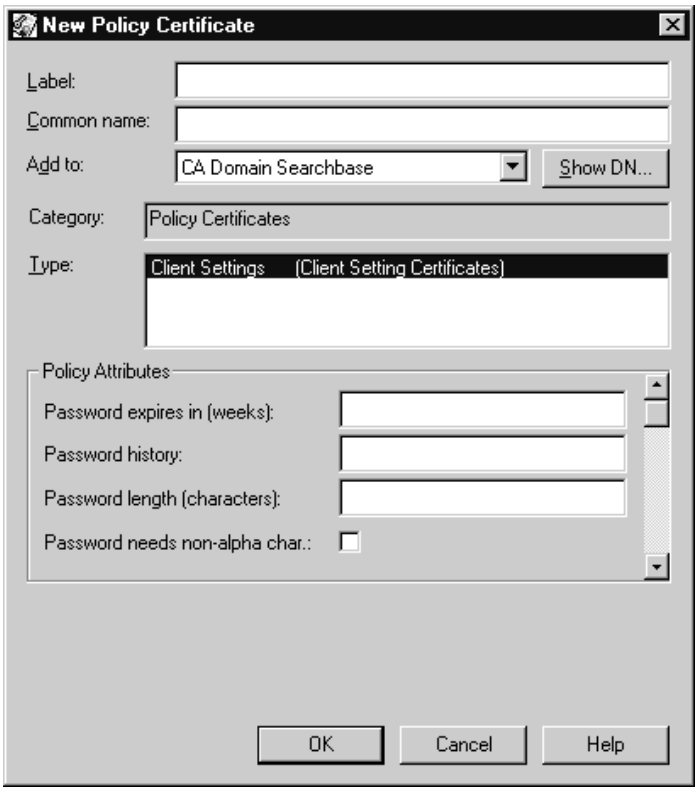
To create a new user policy

1 Log in to Entrust/RA as a Security Officer (see the "Setting up the Certification Authority (CA)" chapter in *Using Entrust/PKI 6.0 on Windows*).

Entrust/RA appears.

2 Double-click *Security Policy* in the tree view. Right-click *User Policies*, which appears below *Security Policy*, and then click New in the pop-up menu.

The New Policy Certificate dialog box appears.



3 In the *Label* field, enter a name that describes the new policy certificate. This is the label that will appear in the Entrust/RA tree view.

4 In the *Common name* field, enter the common name for the new policy certificate's Directory entry (that is, the value for cn=).

5 Select a searchbase in the *Add* to drop-down list. (For more information about searchbases, see the "Setting up the Certification Authority (CA)" chapter in *Using Entrust/PKI 6.0 on Windows*.)

6 In the *Type* text box, select the policy certificate you wish to use.

7 Scroll through the policy settings listed under *Policy Attributes* and select, clear1, or enter the appropriate information for the settings you wish to configure. For more information on a particular setting, see "Overview of policy settings" on page 168.

8 When you have finished configuring all the policy settings, click *OK*.

You have now created a new policy certificate.

Assigning default certificate settings to a role

User policies are assigned to different roles. You can have a single user policy assigned to numerous roles, or you can assign a different policy to each role. You cannot assign multiple user policies to a single role.

To assign a user policy to a role, see "Creating a role" on page 161.

Enforcing the use of custom policy certificates

By default, in Entrust/PKI 6.0, when a user logs in, the client software (such as Entrust/Entelligence) first looks for the user's policy certificate in the Directory. If the client cannot connect to the Directory (for example, the user is working offline), the client then looks for the user's .pch file, which is a locally stored cached copy of the user's policy certificate. If this cannot be found, the user logs in with default user policy settings (see "Overview of policy settings" on page 168 for a description of each policy setting, including the default for each).

If you wish to enforce the use of custom policy certificates that your organization has established (that is, you do not want users logging in using the default user policy settings), you can add an extension to user certificates that will require users to have a valid .pch file when they log in. This forces users to have a valid user policy from the Directory or from a cached .pch file. To enforce the use of policy certificates, you must edit the master.certspec file.

To enforce the use of policy certificates

1. Log in to Entrust/RA.
2. Click File > Certificate Definitions > Export.

   Save the file to a known location on disk.
3. Open the file in a text editor and add the following section:

```
[ent_default Verification Extensions]
;-------------------------------------------------------------
---------
; Require .PCH during login
;-------------------------------------------------------------
---------
require_policy_cert=2.16.840.1.114027.30.2,n,m,DER,0500
```

4. Click File > Save and close the file.
5. In Entrust/RA, click File > Certificate definitions > Import to import the updated master.certspec file.

Performing this procedure will add the extension to every verification certificate created with the Default certificate type. To add the extension to other certificate types, add the code above again but replace "ent_default" with the relevant identifier.

Note: This change only affects newly created verification certificates. If you add this extension to the master.certspec but don't update the users' verification certificates, they will still be able to log in offline with no .pch file. If there is a requirement to have different types of users in the same CA, (some of whom require the .pch file and some of whom don't), you must have two or more certificate types; at least one with an extension and one without.

Modifying the New User dialog box

The New User dialog box lets administrative users with sufficient permissions add new users to Entrust/PKI 6.0. By default, you can add two different types of users (otherwise known as "user types") in Entrust/RA: people and Web servers. But you can add other user types by modifying a template definition file, called "usertype.templates". To add a new user type from scratch, see "To add a new user type" on page 192.

The template definition file includes three other user types:

- Organizational Unit

  You can select this user type in the New Entry dialog box in the Directory Browser application only. By default, you will not see Organizational Unit in the New User dialog box in Entrust/RA. This user type is used in the creation of searchbases. For more information, see the "Setting up the Certification Authority (CA)" chapter in *Using Entrust/PKI 6.0 on Windows*.

- Hardware

  Activate this user type to add hardware devices such as VPN connectors or printers to the Entrust/PKI 6.0 system.

- Person 4.0 PKI

  Activate this user type if you want to include a user's e-mail address in their DN. This user type is included in Release 6.0 to allow administrative users performing a Change DN operation to keep the user's e-mail address in the DN (in Release 4.0, it was possible to include a user's e-mail address in their DN. This is a practice which is not recommended in Release 6.0).

You can activate the Hardware and Person 4.0 PKI user types so that they appear in the New User dialog box. To activate either user type, see "To activate the Hardware and Person 4.0 PKI user types" on page 197.

Who's responsible for modifying the template definition file?

A Security Officer or administrative user with sufficient permissions is responsible for exporting, editing, and importing the template definition. You can use any text editor to edit the file (for example, Notepad).

Note: Entrust Technologies recommends that you keep a secure backup of the template definition file. If you accidentally delete or misplace the backup, you can retrieve the file in its original form from the Entrust/PKI 6.0 CD-ROM.

Summary of steps

The following are the steps you will take to edit the template definition file:

1. Export the file from Entrust/RA to a location on your disk. See "Exporting the template definition file" on page 190.
2. Open the file in a text editor.
3. Add or activate a user type:
   - To add a new user type from scratch, you will add a user type and a new user type section. See "To add a new user type" on page 192.
   - To activate the Hardware or Person 4.0 User user types, you will add a new line of text in the user type section. See "To activate the Hardware and Person 4.0 PKI user types" on page 197.
4. Add attributes to the new user type section. See "Adding attributes" on page 194.
5. Save or rename the file.
6. Import the file into Entrust/RA. See "To import the template definition file" on page 198.
7. Test the file. See "To test the template definition file" on page 199.

   Make sure you correctly format the information you add in the template definition file. Entrust Technologies suggests that you model your new user types and attributes after existing user types and attributes to ensure that you don't omit any vital information.

Exporting the template definition file

If you are a Security Officer or an administrative user with sufficient permissions, you can export the template definition file to a location on your computer's disk. This is the first step in the process of adding or activating a user type.

Note: Entrust Technologies strongly recommends that you export a backup of the original usertype.templates file and store it in a safe location. This way you can restore from the backup if you encounter problems.

To export the template definition file

1. Log in to Entrust/RA. See the "Setting up the Certification Authority (CA)" chapter in *Using Entrust/PKI 6.0 on Windows*.

   Entrust/RA appears.
2. Click File > User Templates > Export.

   The Save As dialog box appears.

3 In the Save As dialog box, specify a filename and location for the usertype.templates file and click *Save*.

When you export the file, you should rename it in order to maintain a complete file history. If you don't rename the file, it is overwritten. Export the most recent file and store the backups on disk.

A dialog box appears stating that the file was exported successfully.

4 Click *OK*.

You have now exported the file for editing. When you finish editing it, import the file as described in "Importing the template definition file" on page 198.

Customizing the New User dialog box

Users types refer to all people and devices that you add to Entrust/PKI 6.0 and are defined by the conventions of your Directory (these conventions are known as the "Directory schema"). In other words, depending on the type of Directory your organization uses, you may be able to add certain types of users and not others. Consult a Directory Administrator if you have any questions about user types and the Directory schema. For general information about Directories, see the "About Entrust/PKI" chapter in *Using Entrust/PKI 6.0 on Windows*.

Adding a new user type is a two-step process:

- First add the new user type to the User Type Template List (see "To add a new user type" on page 192).
- Then, create a new section for the new user type (see "To add a new user type section" on page 192).

Instructions in the template definition file

The working section of the default usertype.templates file follows a series of instructions. All text on a line that begins with a semicolon (;) is instructional. Begin editing after the line reading "End of Instructions".

Directory validation

All new user types must be validated by your Directory schema. You cannot add arbitrary or unsupported user types. In the next procedure, "To add a new user type", adding the Internet User user type shows how you add a user type from scratch. If your Directory does not support this user type, however, it cannot be properly added. Before you proceed, find out what user types your Directory supports. If you are uncertain whether a user type is valid, consult a Directory Administrator or Security Officer.

To add a new user type

1. Open the template definition file in a text editor.
2. After the instructions section, under the heading "[User Type Template List]" increase the count= value by one.

   For example, to add a new user type—in this example, "Internet User"—change count=3 to count=4 (as Internet User is the fourth user type in this file).

```
[User Type Template List]
count=3
0=Person
1=Web Server
2=Organizational Unit
```

Increase the count value by 1, in this case from 3 to 4.

User types.

   The user type that appears first in the New User dialog box user type list is that which is assigned the 0 value in the [User Type Template List]. By default (as seen above) this is the Person user type (0=Person).

   The Organizational Unit user type is used to add searchbases. It is available only in the New Entry dialog box in the Directory Browser (you won't see this user type in the New User dialog box).

3. Still under the heading "[User Type Template List]" add the new user type to the numbered list.

   For example, add "3=Internet User" below "2=Organizational Unit".

```
[User Type Template List]
count=4
0=Person
1=Web Server
2=Organizational Unit
3=Internet User
```

Type the name of the new user type in the list below the last line. Use common English. You can include spaces and uppercase and lowercase letters.

   You have now completed the first step in adding a new user type. To continue, see "To add a new user type section" below.

To add a new user type section

1. Scroll to the bottom line (below the Hardware section) and type the section heading. Make sure you enter between two square brackets [ ], the exact text that appears after the = sign in the user type you added above.

```
[Internet User]
```

Type the new user type heading. Use common English. You can include spaces and uppercase and lowercase letters.

2 Below the heading, type the id number for the new user type.

```
[Internet User]
id=3
```

Type the user type ID number. This is the number assigned to the user type in the User Type Template List. The number in this example is "3".

You can insert the new user type section anywhere in the file as long as the id number is correct.

3 Type "count=" and the number of attributes you want to add to this user type.

Attributes refer to common user information, such as name, serial number, telephone number, and e-mail address. At this point, you should know how many and what type of attributes you want to add for the new user type.

If you plan to add one attribute for the new user type, type "count=1".

```
[Internet User]
id=3
count=1
```

Type the number of attributes the new user type will include. To add a single attribute, type "count=1".

4 Type "Structural Object Class=" and the name that represents this user.

You must know how the user type is represented in the Directory. The names are generally not in plain English. Using our example, Internet User is represented in the Directory as "inetOrgPerson". Note the lowercase and uppercase letters.

```
[Internet User]
id=3
count=1
Structural Object Class=inetOrgPerson
```

Type the proper structural object class definition for the user type. You must type the class name as it is known in the Directory. Ask a Directory Administrator if you need assistance.

5 This step is optional. If you want the new user type to be available only using the Directory Browser's New Entry dialog box, add "DirBrowserOnly=1" below Structural Object Class (as described in Step 4). Using this setting, the new user type will not appear in the New User dialog box in Entrust/RA. By default, this

line does not appear in any but the Organizational Unit type, as you do not create Entrust users of organizational units.

```
[Internet User]
id=3
count=1
Structural Object Class=inetOrgPerson
DirBrowserOnly=1
```

Add this line if you want this user type to appear only in the New Entry dialog box in the Directory Browser.

If you later decide that you want to include the new user type in the Entrust/RA New User dialog box (where you add users to Entrust), simply remove the DirBrowserOnly=1 line or change the 1 to 0.

6 Type "description=" and a description of the user type.

The description appears in the New User dialog box and helps administrative users understand the user type. The example used here is very short. Do not hesitate to add more descriptive, conversational text if you think administrative users will benefit from such a description.

```
[Internet User]
id=3
count=1
Structural Object Class=inetOrgPerson
description=Internet user
```

Type a description to help administrative users when they add new users.

You will now add attributes for the user type. See "To add an attribute for a new user type" on page 194.

Note: If you create a new user type and decide to remove the user type later, ensure that the user type you are deleting is not currently in use by any users. If a user exists and its type is deleted, then you may have problems diplaying information related to that particular user when you are using Entrust/RA.

Adding attributes

You can add as many different attributes for each user type as you want. There are many attribute types in the Directory. Use standardized attributes to ensure interoperability.

To add an attribute for a new user type

1 Open the usertype.templates file in a text editor and complete the procedure, "To add a new user type section" on page 192.

2 Under description=Internet user (in this example), type "0=". Attributes are listed, in order, beginning with "0".

```
[Internet User]
id=3
count=1
Structural Object Class=inetOrgPerson
description=Internet user
0=
```

Type "0=" at the beginning of the new attribute line. Type "1=" for the next attribute after this one.

3 Next, type the name of the attribute, followed by a comma. This is what will appear as the field label in the New User dialog box.

```
[Internet User]
id=3
count=1
Structural Object Class=inetOrgPerson
description=Internet user
0=Common Name,
```

This number represents the number of attributes.

Type the name of the attribute. Use common English. Then type a comma.

4 Next, type the attribute name as it is recognized in Directory language. You must know the name of the attribute used in the Directory. Using our example, "common name" is represented in the Directory as "cn". Then type a comma.

```
[Internet User]
id=3
count=1
Structural Object Class=inetOrgPerson
description=Internet user
0=Common Name,cn,
```

Type the attribute as it is represented in the Directory. In this example, "cn" stands for "common name", a frequently used Directory attribute. Then type a comma.

5 Next, type a value to do one of the following.

Value	State of In DN checkbox in New User dialog box	Adds information in the attribute field to the DN? Yes/No.
0	Enabled and unchecked	Yes, if check box checked. No, if unchecked.
1	Disabled and checked	Yes. The attribute is automatically added in the DN.

Value	State of In DN checkbox in New User dialog box	Adds information in the attribute field to the DN? Yes/No.
2	Disabled and unchecked	No. The attribute cannot be added in the DN.
3	Enabled and checked	Yes, if check box checked. No, if unchecked.

```
[Internet User]
id=3
count=1
Structural Object Class=inetOrgPerson
description=Internet user
0=Common Name,cn,1,
```

This is the number of attributes.

Type a value as described in the table above.

6 Next, type a "1" or "2" to include or exclude the attribute in the Directory. Type "0" to make this attribute optional in the Directory. Then type a comma.

```
[Internet User]
id=3
count=1
Structural Object Class=inetOrgPerson
description=Internet user
0=Common Name,cn,1,0,
```

Type "1" to include the attribute in the Directory, "2" to exclude the attribute, or "0" to make this attribute optional. Then type a comma.

7 Next, type "1" to make the value you enter for this attribute unique across all available searchbases in the Directory. Type "0" if you don't want to make this attribute unique.

For example, if you make the common name value unique, you cannot add two users to Entrust/PKI with the same common name.

```
[Internet User]
id=3
count=1
Structural Object Class=inetOrgPerson
description=Internet user
0=Common Name,cn,1,0,1
```

Type "1" to make the attribute value unique across all available searchbases in the Directory.

Note: When you add a user with a unique attribute value, Entrust searches the available searchbases in the Directory for this value to see if it currently exists. If this value is not found, it can be used. However, there is no way to guard against its reuse if it is later removed or changed using a third-party tool.

You have now added an attribute to the new user type. Repeat this procedure to add other attributes. Remember to increase count = to the number of attributes you want to add.

You have now added a new user type section to the template definition file.

8 Save the file using a new filename and import the file as described in "To import the template definition file" on page 198.

Determining the structure of the common name for the Person user type

In the template definition file, the Person user type includes an additional line in the Person section:

```
overrideCommonNameFormat=1
```

This override specifies how the cn (common name) value is formed (that is, showing the first name followed by the last name, or the other way around).

To form the cn as follows, enter 1:

"cn=First Name Last Name"

To form the cn as follows, enter 2:

cn="Last Name, First Name"

If the override value is undefined, missing, or a number other than 1 or 2 is entered, the cn will consist of the value entered in the First Name field.

Note: A cn comprises two names, but the label associated with each is irrelevant. If you add a customized user type, it may not be the First and Last name that you are expecting. It could be two others, such as "Given Name" or "Family Name". It depends on the user type you're adding.

To activate the Hardware and Person 4.0 PKI user types

1 Open the template definition file in a text editor.

2 After the instructions section, under the heading "[User Type Template List]" increase the count= value by one.

For example, to activate one of the user types, change count=3 to count=4.

```
[User Type Template List]
count=3
0=Person
1=Web Server
2=Organizational Unit
```

Increase the count value by 1, in this case, from 3 to 4.

3 Still under the heading "[User Type Template List]" add the new user type to the numbered list.

For example, add "3=Hardware" below "2=Organizational Unit".

```
[User Type Template List]
count=4
0=Person
1=Web Server
2=Organizational Unit
3=Hardware
```

Type "3=Hardware" to add the Hardware user type. To add the Person 4.0 User user type, type "3=Person 4.0 PKI".

4 Save the file using a new filename and import the file as described in "To import the template definition file" on page 198.

Sections for both the Hardware and Person 4.0 User user types are included in the usertype.templates file. This is all you need to do to add either user type in the New User dialog box.

Importing the template definition file

After you edit the template definition file, you must import it back into Entrust/RA.

Error checking

When you import the file, Entrust/RA and Entrust/Authority check the file for errors. If Entrust/RA finds errors, descriptions of the errors are written to a log file. Any errors found by Entrust/Authority are displayed in a dialog box onscreen.

To import the template definition file

1 Log in to Entrust/RA. See the "Setting up the Certification Authority (CA)" chapter in *Using Entrust/PKI 6.0 on Windows*.

2 Click File > User Type Templates > Import.

The Open dialog box appears.

3 Select usertype.templates in the Open dialog box and click *Open*.

The Save As dialog box appears.

4 Accept the default or specify a name and location for a log file that will list any errors found by Entrust/RA when the file is imported, and click *Save*.

An information dialog box appears, stating that the file was imported successfully.

5 Click *OK*.

You have now imported the template definition file.

Testing the template definition file

After you modify the template definition file and import it back into Entrust/RA, you should test the file against the New User dialog box to ensure that the resulting information is represented correctly.

To guarantee that administrative users can enter users in each user type and can enter the required information in the attribute fields, you should create a test entry in each user type, specifying the required data.

As noted in "Exporting the template definition file" on page 190, you should always keep a backup copy of the usertype.template file on disk in a known location. If you experience a problem that you cannot resolve, create a new usertype.template file based on the backup and try again.

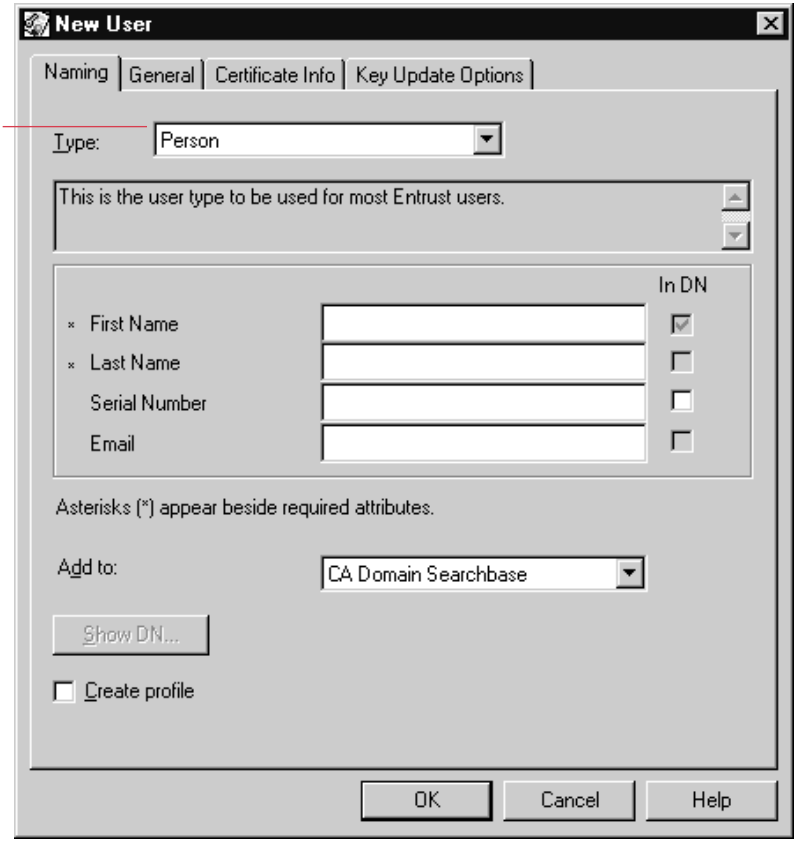
To test the template definition file

1 Export, edit, and import the usertype.template file. See "Modifying the New User dialog box" on page 189.

2 Click *Users* > *New User*.

The New User dialog box appears.

Click to confirm that the new user type is included in this drop-down list. By default, Person is listed first.



3 Click the *Type* drop-down list and select a user type.

Check that the description fields and attribute fields appear correctly.

4 Type attribute information in each attribute field for a test entry.

One attribute field should exist for every attribute added in the templates definition file.

5 Click *Show DN* to ensure that at least one attribute is listed. All attributes designated in the template definition file for inclusion in the DN should be seen in the DN.

6 Click *OK*.

7 Repeat Step 3 through Step 6 for each user type listed in the *Type* box.

8 Right-click each test entry in the tree view and click Properties in the pop-up menu. Review the information for each in the *General Information* property page.

If you are satisfied with the results of the test, you can let administrative users begin adding new users and devices to Entrust/PKI 6.0.

Chapter 6

Cross-certifying with other CAs

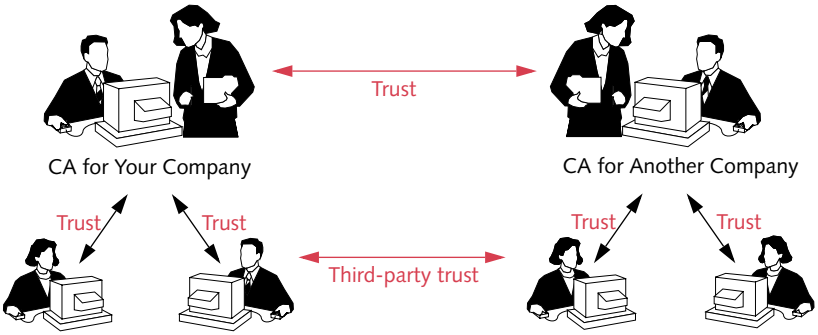
Cross-certification is a means for establishing third-party trust among users who belong to different Certification Authorities (CAs). This third-party trust means that users who belong to one CA can exchange secured data with users in a cross-certified CA. To understand how cross-certification works, you must understand the concepts of third-party trust and direct trust as they relate to network security (see the "About Entrust/PKI" chapter in *Using Entrust/PKI 6.0 on Windows*).

This chapter explains how cross-certification relates to trust and how cross-certificates establish trust among CAs. This chapter also describes the types of cross-certification you can perform between Entrust CAs, and how to choose which type of cross-certification you require.

This chapter does not describe how to cross-certify an Entrust CA with a CA from another vendor. Although this type of cross-certification is possible, both CAs must meet special requirements. If you want to perform this type of cross-certification, contact Entrust Support.

About cross-certification

Cross-certification is a process that allows two CAs to trust each other. When two CAs trust each other, their users indirectly trust each other. This trust among users from cross-certified CAs is a form of third-party trust. The following diagram shows the trust relationships that exist between cross-certified CAs.



Once trust is established via cross-certification, users can exchange secured information with each other, provided that searchbases have been created. A searchbase enables users who belong to a CA to locate users in a cross-certified CA. For more information about searchbases, see the "Setting up the Certification Authority (CA)" chapter in *Using Entrust/PKI 6.0 on Windows*.

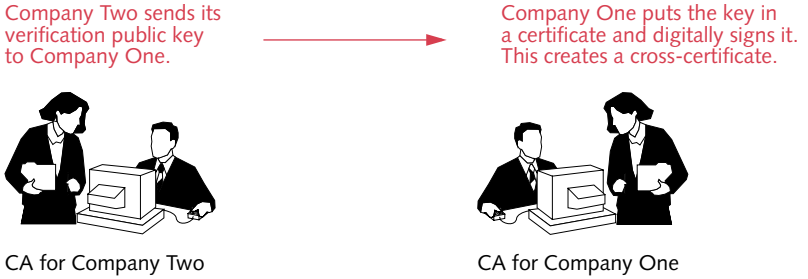
This chapter describes how to implement cross-certification. However, there is much more to cross-certification than simply implementation. For example, because cross-certification extends third-party trust, CAs must be completely comfortable with each other's security policies. To use passports as an analogy, one country is unlikely to trust another country's passports without first examining the policies used by that country to create and distribute their passports. For example, each country would want to understand, in detail, the process by which the other country verifies a citizen's identity before issuing them a passport.

As well as understanding the other CA's security policies, a CA should also learn which people have access to high-level security functions for the other CA. It is also likely that representatives of both CAs would sign a legal agreement before performing cross-certification. This agreement would state the required security policies in both domains and give signed assurance that these policies would be followed.

Establishing trust using cross-certification

Within Entrust/PKI, trust is established between CAs using cross-certificates. A cross-certificate contains one CA's verification public key, and is digitally signed by another CA using its own signing private key. The CA that signed the cross-certificate trusts the CA whose verification public key is in the cross-certificate.

Understanding how trust is established using cross-certificates is best shown by example. Suppose that Company One wants to trust Company Two. As shown in the following diagram, Company Two securely provides its verification public key to Company One. Company One creates a cross-certificate that contains Company Two's key and is digitally signed by Company One. Company One stores the cross-certificate in the CA entry of its Directory.

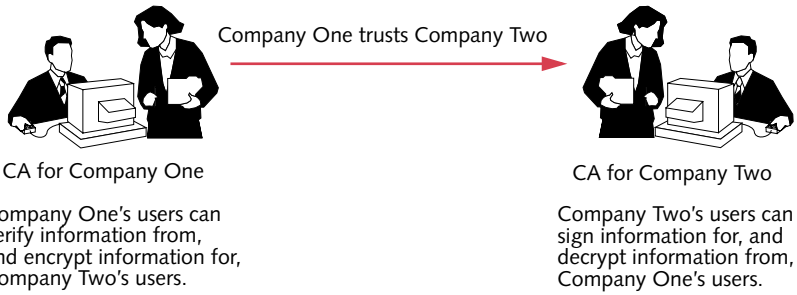


Cross-certificates are valid for 36 months, by default. Security Officers can change the lifetime of cross-certificates using the Administration Policy (see the "Setting up the Certification Authority (CA)" chapter in *Using Entrust/PKI 6.0 on Windows*). The minimum lifetime is 2 months and the maximum lifetime is 60 months. You must update cross-certificates before they expire (see "Updating a cross-certificate" on page 241) to ensure there is no interruption in service for users in the cross-certified CAs.

One-way trust versus two-way trust

Suppose that Company One trusts Company Two, but Company Two doesn't trust Company One. This relationship is referred to as "unilateral cross-certification" because trust is established in only one direction (that is, only Company One creates a cross-certificate). As shown in the following diagram, users who belong to Company One indirectly trust users who belong to

Company Two, which means that Company One's users can encrypt information for, and verify (authenticate) information from, Company Two's users.



Two-way trust is referred to as "mutual cross-certification" (that is, both CAs create a cross-certificate). For example, suppose that two departments are managed by their own CAs. Through mutual cross-certification, employees in both departments can encrypt inter-department memos for each other, and verify information from each other.

How Entrust-Ready applications use cross-certificates

When users in cross-certified CAs exchange secured data, Entrust-Ready applications, such as Entrust/Entelligence, validate user certificates using cross-certificates. You can see how applications use cross-certificates in the following example, where Bob (of Company One) uses Entrust/Entelligence to encrypt a file for Alice (of Company Two).

How Bob encrypts a file for Alice

1. Bob uses the Entrust/Entelligence searchbase feature to search for Alice in the Directory and retrieve her encryption public certificate.
2. Entrust/Entelligence checks that Company Two is still trusted using the cross-certificate held by Company One. Entrust/Entelligence verifies the cross-certificate signature, and checks that the cross-certificate has integrity and has not expired. Entrust/Entelligence also ensures that the cross-certificate has not been revoked by examining the authority revocation list (ARL) signed by Company One.
3. Entrust/Entelligence extracts Company Two's verification public key from the cross-certificate held by Company One.
4. Entrust/Entelligence validates Alice's encryption public certificate by validating Company Two's digital signature on the certificate, and checking the certificate's lifetime. In so doing, Entrust/Entelligence checks that the certificate has integrity, and also ensures that the certificate is not revoked by examining the certificate revocation list (CRL) that is issued by Company Two.

5 If the cross-certificate held by Company One is valid, and if Alice's encryption public certificate is valid, Bob is allowed to encrypt the file for Alice.

Methods for establishing trust

You can establish trust between CAs online or offline.

Online cross-certification occurs between CAs on servers that connect over a network via TCP/IP. For more information about cross-certifying CAs online, see "Cross-certifying online" on page 208.

Offline cross-certification occurs between CAs on servers that do *not* connect over a network via TCP/IP (for example, a network is not available or you do not want the servers connected over a network). For more information about cross-certifying CAs offline, see "Offline cross-certification" on page 218.

Cross-certifying online

Online cross-certification is a method of cross-certifying CAs on servers that are connected over a network. You can perform unilateral cross-certification online to establish one-way trust and you can perform mutual cross-certification online to establish two-way trust.

The information in this section assumes that the necessary trusting business relationships (such as signed legal agreements) have been established.

Online cross-certification requirements

Before you begin online cross-certification

- the CAs must be located on servers that are connected over a network. The servers must be connected via TCP/IP during the cross-certification process. Once cross-certification is complete, the network connection is no longer necessary.
- the Directories for both CAs must be connected. The Directories can be connected physically (for example, they could be two branches in one Directory) or logically (for example, they could be two physically distinct Directories that are "chained" together).

After you complete online cross-certification

- the Directories must remain connected to enable Entrust-Ready applications to verify the validity of user certificates in the Directory.
- the CAs must have searchbases to enable Entrust-Ready applications to locate users who belong to a cross-certified CA. For more information about searchbases, see the "Setting up the Certification Authority (CA)" chapter in *Using Entrust/PKI 6.0 on Windows*.

Performing online cross-certification

To perform online unilateral or mutual cross-certification, one CA follows a procedure to initiate cross-certification and the other CA follows a procedure to complete cross-certification. For mutual cross-certification, either CA can initiate cross-certification because trust will be established in both directions. However, because unilateral cross-certification establishes trust only one way, unilateral cross-certification must be initiated by the CA that wants to establish trust, and completed by the CA that is to be trusted.

During online cross-certification, the two CAs communicate with each other. For example, the CAs communicate when one CA provides its verification public key to the other CA for inclusion in a cross-certificate. CAs communicate using one of the following protocols:

- SEP (Secure Exchange Protocol), a proprietary protocol developed by Entrust Technologies. Entrust/PKI Release 6.0 and earlier supports SEP for mutual cross-certification only.
- PKIX-CMP (Public-Key Infrastructure X.509–Certificate Management Protocol), an industry-standard protocol defined by the IETF (Internet Engineering Task Force). For information about PKIX-CMP, contact the IETF and request RFC 2510. Entrust/PKI Release 6.0 supports PKIX-CMP for mutual and unilateral cross-certification.

You must know which protocol you are going to use before a CA initiates cross-certification. Because both CAs must use the same communication protocol, the protocol used depends on the release of Entrust/PKI that the CAs belong to and the type of cross-certification (either unilateral or mutual). Table 7 on page 209 describes when each protocol is used.

Table 7: Cross-certification protocols

Use this protocol...	if you are...
PKIX-CMP	mutually or unilaterally cross-certifying two CAs from Entrust/PKI Release 5.0 or newer You can use SEP to mutually cross-certify these CAs because Entrust/PKI Release 6.0 supports both PKIX-CMP and SEP. However, Entrust Technologies intends SEP to be used for backward compatibility only.
SEP	mutually cross-certifying two CAs, and one or both CAs are from Entrust/PKI Release 4.x (or earlier) You cannot unilaterally cross-certify two CAs online if one or both CAs are from Entrust/PKI Release 4.x (or earlier).

The following procedures describe how to

- initiate cross-certification
- obtain a password that will be used to complete cross-certification
- complete cross-certification

For clarity, these procedures refer to the CA that initiates cross-certification as "Company One" and to the CA that completes cross-certification as "Company Two".

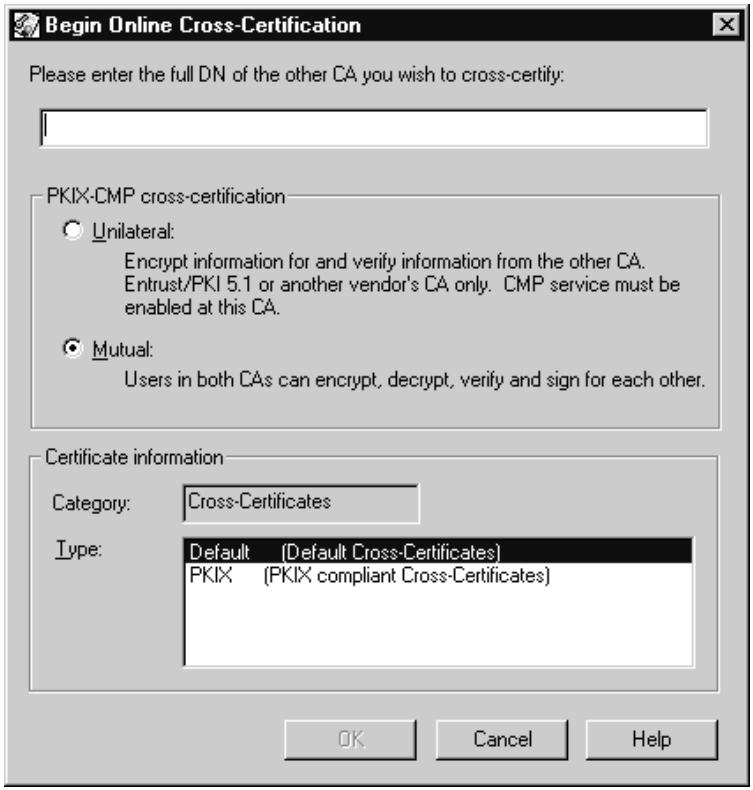
To initiate online cross-certification at Company One

1 Log in to Entrust/RA as a Security Officer or an administrative user with appropriate permissions for Company One (see the "Setting up the Certification Authority (CA)" chapter in *Using Entrust/PKI 6.0 on Windows*).

2 Click "+" beside the *Certification Authority* icon in the tree view of the Entrust/RA window.

3 Right-click the *Cross-Certified CAs* icon and click Online Cross-Certification > Begin Online Cross-Certification in the pop-up menu.

The Begin Online Cross-Certification dialog box appears.



4 In the Begin Online Cross-Certification dialog box, do the following:

- In the text box at the top of the dialog box, type the DN of the CA you are cross-certifying with (for example, o=Company Two,c=US).
- Click the type of cross-certification (either unilateral or mutual).

  The label of this group box is *PKIX-CMP cross-certification*, but SEP will be used if you are performing mutual cross-certification between two CAs and one or both CAs are from Entrust/PKI Release 4.x or earlier.
- Click the type of cross-certificate and click *OK*.

  Only the existing cross-certificate types appear. For information about cross-certificate types, see "Customizing certificates" on page 269.

The Authorization Requested dialog box appears.

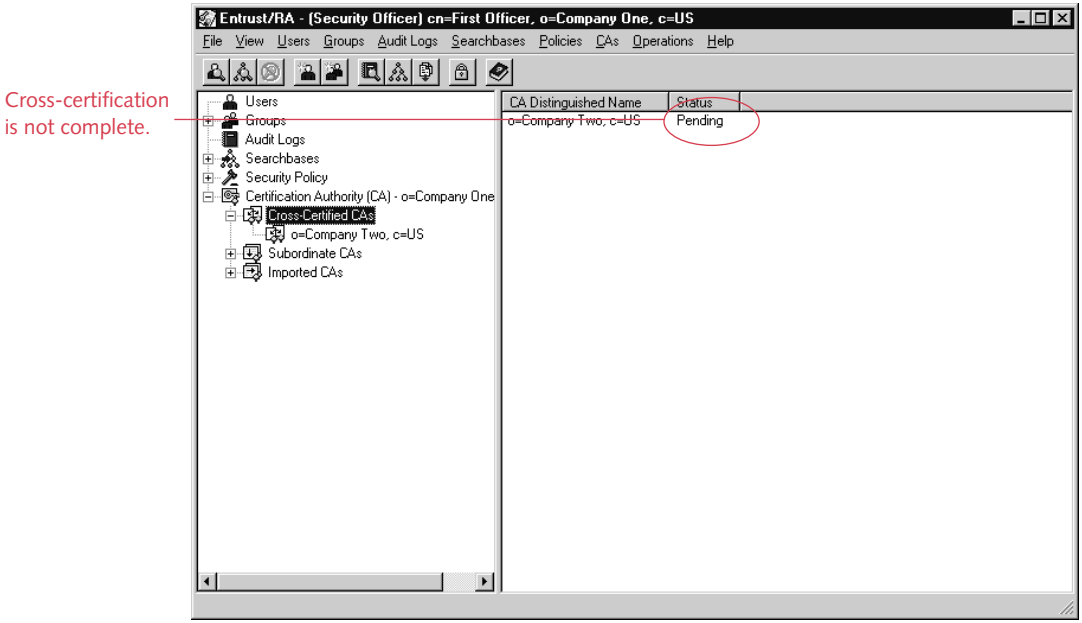
5 Authorize the operation and click *OK*.

A dialog box appears when the initiation is complete.

6 Click *OK*.

You have now initiated cross-certification.

The CA that will complete cross-certification (that is, "Company Two") appears in Company One's Entrust/RA window with a status of "Pending".



Note: When Company Two completes cross-certification, the status in Company One's Entrust/RA window changes from "Pending" to "Complete". You must right-click *Cross-Certified CAs* entry in the tree view to see this change.

You must provide the Security Officer for the CA that completes cross-certification with the

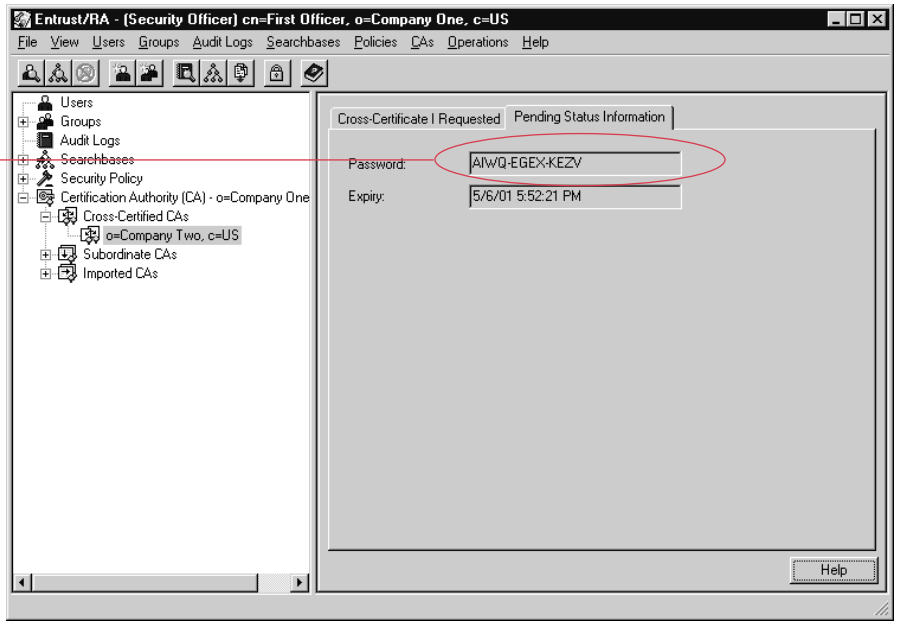
- TCP/IP address and port number of the CA that initiated cross-certification (that is, "Company One"). If PKIX-CMP is being used as the communication protocol for cross-certification, you can obtain this information from the "Authority" entry in the entrust.ini file (the entry will be in the format of "Authority=ipaddress+portnumber"). If SEP is being used as the communication protocol for cross-certification, you can obtain this information from the "Manager" entry in the entrust.ini file (the entry will be in the format of "Manager=ipaddress+portnumber").
- password used to secure the certification process. You must provide this password to the Security Officer in a highly secure manner (for example, in person).

To obtain the cross-certification password

1. Log in to Entrust/RA as a Security Officer or an administrative user with appropriate permissions for Company One (see the "Setting up the Certification Authority (CA)" chapter in *Using Entrust/PKI 6.0 on Windows*).
2. Click "+" beside the *Certification Authority* icon in the tree view of the Entrust/RA window.
3. Click "+" beside the *Cross-Certified CAs* icon.
4. Click the cross-certified CA.

   The *Cross-Certificate I Requested* property page appears.
5. Click *Pending Status Information*.

   The *Pending Status Information* property page shows a 12-character single-use password in the *Password* field.



Give this password to the Security Officer for the other CA.

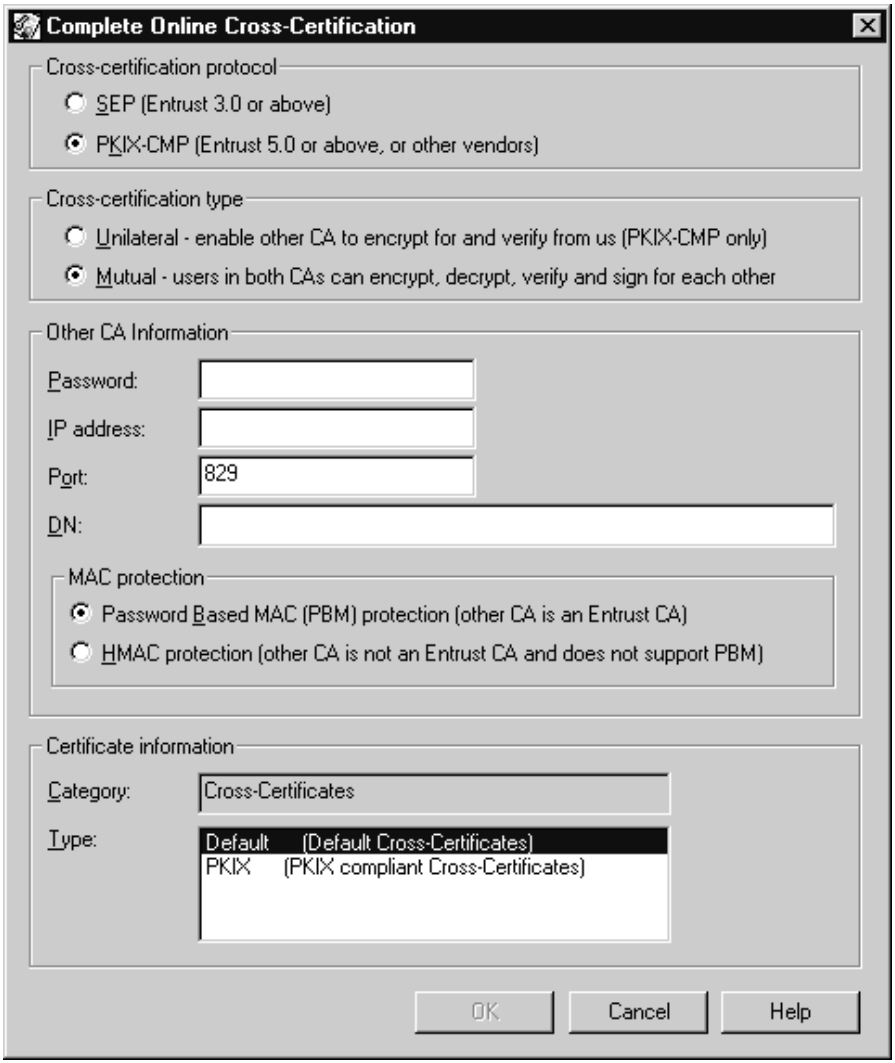
You have now obtained the cross-certification password.

Note: This password is valid for three days only. If the other CA does not complete cross-certification within three days, you must cancel the cross-certification and then repeat the initiation procedure.

To complete cross-certification online at Company Two

1. Ensure that you have received the following information from the Security Officer for the CA that initiated cross-certification:
   - TCP/IP address and port number of the CA that initiated cross-certification (that is, "Company One")
   - password used to secure the certification process
2. Log in to Entrust/RA as a Security Officer or an administrative user with appropriate permissions for Company Two (see the "Setting up the Certification Authority (CA)" chapter in *Using Entrust/PKI 6.0 on Windows*).
3. Click "+" beside the *Certification Authority* icon in the tree view of the Entrust/RA window.
4. Right-click the *Cross-Certified CAs* icon and click Online Cross-Certification > Complete Online Cross-Certification in the pop-up menu.

The Complete Online Cross-Certification dialog box appears.



5 In the Complete Online Cross-Certification dialog box, do the following:

- Select the communication protocol for cross-certification (see Table 7 on page 209).
- Select the type of cross-certification.
- In the *Password* field, type the cross-certification password that you received from the Security Officer for Company One (see "To obtain the cross-certification password" on page 213).

When you type the password, the hyphens (-) are optional. The hyphens are present in the password solely for readability.

- In the *IP address* field, type the TCP/IP address of Company One (for example, 97.12.33.87).
- In the *Port* field, type the port number of Company One.
- In the *DN* field, type the DN for Company One.
- In the *MAC protection* group box, choose *Password Based MAC (PBM) protection (other CA is an Entrust CA)* if you are cross-certifying with another Entrust CA. The message authentication code is used when authenticating between two Entrust CAs that are cross-certifying using PKIX-CMP.

  Choose *HMAC protection (other CA is not an Entrust CA and does not support PBM)* if you are cross-certifying with a CA that uses another vendor's PKI that requires this option. Check with the documentation of the other CA's PKI vendor for information.
- Click the type of certificate that you want to create and click *OK*.

  Only the existing cross-certificate types appear. For information about cross-certificate types, see "Customizing certificates" on page 269.

The Authorization Requested dialog box appears.

6 Authorize the operation and click *OK*.

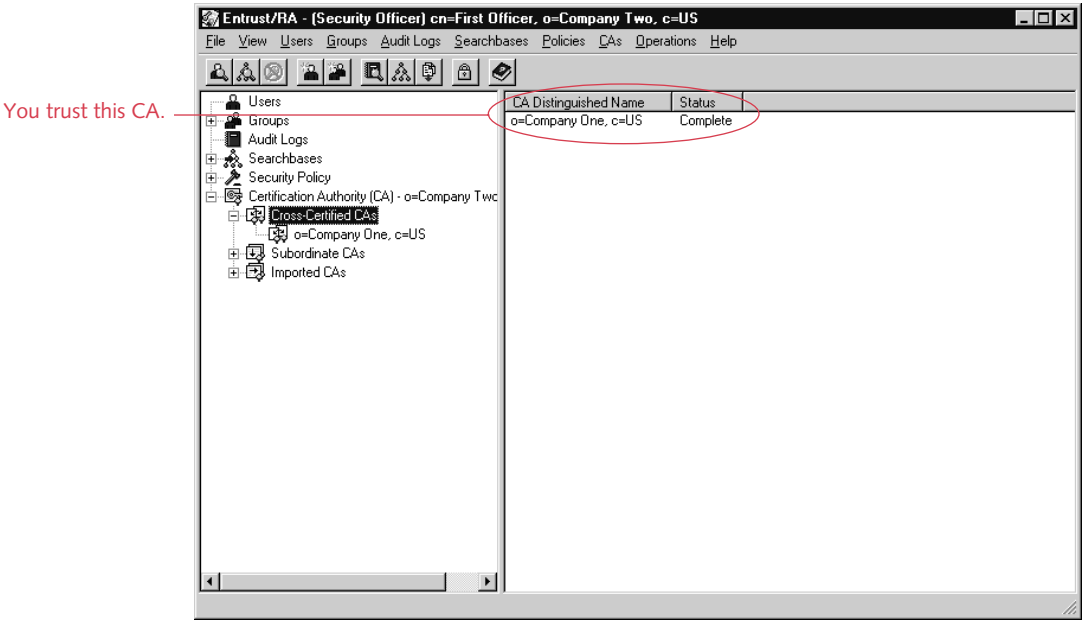
A dialog box appears when the cross-certification is complete.

7 Click *OK*.

You have now completed cross-certification.

Note: Cross-certificates are valid for 36 months, by default (see the "Setting up the Certification Authority (CA)" chapter in *Using Entrust/PKI 6.0 on Windows*). You must update cross-certificates before they expire (see "Updating a cross-certificate" on page 241) to ensure there is no interruption in service for users in the cross-certified CAs.

If Company One and Company Two performed mutual cross-certification, each CA appears in the other CA's Entrust/RA window with a status of "Complete". For example, Company One appears in the Entrust/RA window for Company Two.



If Company One and Company Two have performed unilateral cross-certification, Company Two appears in the Entrust/RA window for Company One with a status of "Complete". (Company One does not appear in the Entrust/RA window for Company Two because trust was established only one way).

You can click a cross-certified CA in the Entrust/RA window to display information about that CA. For details on displaying information about a cross-certified CA, see "Displaying information about cross-certified CAs" on page 233.

Offline cross-certification

Offline cross-certification is a method of cross-certifying CAs on servers that are not connected over a network via TCP/IP. You can perform unilateral cross-certification offline to establish one-way trust or you can perform mutual cross-certification offline to establish two-way trust.

The information in this section assumes that the necessary trusting business relationships (such as signed legal agreements) have been established.

Offline cross-certification requirements

Before you begin offline cross-certification, connect the Directories for both CAs. The Directories can be connected physically (for example, they could be two branches in one Directory) or logically (for example, they could be two physically distinct Directories that are "chained" together).

After you complete offline cross-certification,

- the Directories must remain connected to enable Entrust-Ready applications to verify the validity of user certificates in the Directories.
- the CAs must have searchbases to enable Entrust-Ready applications to locate users who belong to a cross-certified CA. For more information about searchbases, see the "Setting up the Certification Authority" chapter in *Using Entrust/PKI 6.0 on Windows*.

Performing offline cross-certification

To perform offline unilateral cross-certification, one CA follows a procedure to initiate cross-certification and the other CA follows a procedure to complete cross-certification. Because unilateral cross-certification establishes trust only one way, it is important that the CA that initiates cross-certification is the one that will be trusted by the CA that completes cross-certification.

To perform offline mutual cross-certification, the initiation and completion procedures are performed twice: once to establish trust in one direction (for example, from Company One to Company Two) and once to establish trust in the reverse direction (for example, from Company Two to Company One). Because trust will be established in both directions, either CA can initiate cross-certification first.

Note: For unilateral offline cross-certification, it is important to understand that the CA that initiates the cross-certification will be trusted by the CA that completes the cross-certification. This is different from online unilateral cross-certification.

In online unilateral cross-certification, if CA1 wants to trust CA2, then CA1 initiates the cross-certification. In offline unilateral cross-certification, if CA1 wants to trust CA2, then CA2 must initiate the cross-certification.

During the procedure to initiate cross-certification, the CA creates a certificate request (formatted as defined by the PKCS #10 standard) that contains its CA verification public key, saves it to a file, and generates a validation string. During the procedure to complete cross-certification, the other CA signs this certificate request after verifying the validation string.

The following procedures describe how to

- initiate cross-certification
- complete cross-certification

For clarity, these procedures refer to the CA that initiates cross-certification as "Company Two" and to the CA that completes cross-certification as "Company One". In the following example, Company One will sign a certificate request from Company Two. This will allow Company One to trust Company Two.

To initiate offline cross-certification at Company Two

1. Log in to Entrust/RA as a Security Officer or an administrative user with appropriate permissions for Company Two.
2. Click the "+" symbol beside the *Certification Authority* icon in the tree view of the Entrust/RA window.
3. Right-click the *Cross-Certified CAs* icon and click Offline Cross-Certification > Request Cross-Certificate for > Enterprise/Web in the pop-up menu.

The CA Cross-Certificate Request dialog box appears.

4 Choose a destination folder and type a name for your cross-certificate request file. You can choose the file format to be binary (an extension of .der is appended to the filename), or PEM (an extension of .pem is appended to the filename). The choice of formats is provided for interoperability with PKIs from other vendors. Entrust/PKI can read files in either format.

5 Click *Save*.

When the initiation is complete, a dialog box appears displaying the validation string. The Security Officer (or an administrative user with appropriate

permissions) at Company One will verify the validation string to confirm that the cross-certificate request file has not been tampered with.

Securely give this validation string to the Security Officer for the other CA.

6 Copy the validation string down on paper and click *OK*.

This validation string is created based on an MD5 hash of the PKCS #10 request.

7 Deliver the cross-certificate request (that is, the file with the .der or .pem extension) file to Company One in a secure manner.

You have now initiated offline cross-certification.

To complete offline cross-certification at Company One

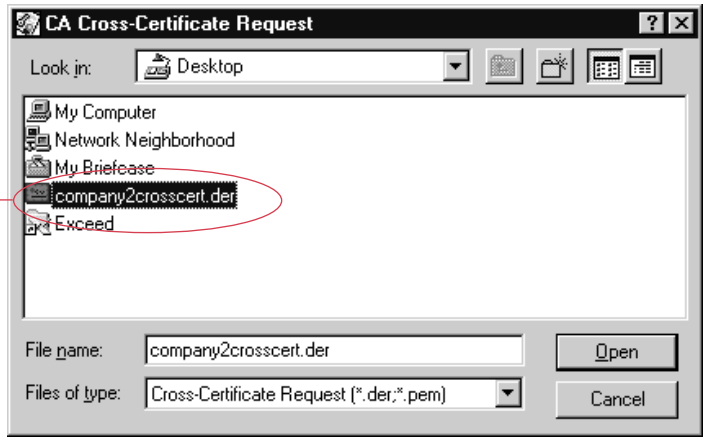
1 Log in to Entrust/RA as a Security Officer or an administrative user with appropriate permissions for Company One.

2 Click the "+" symbol beside the *Certification Authority* icon in the tree view of the Entrust/RA window.

3 Right-click the *Cross-Certified CAs* icon and click Offline Cross-Certification > Sign Cross-Certificate for > Enterprise/Web in the pop-up menu.

The CA Cross-Certificate Request dialog box appears.

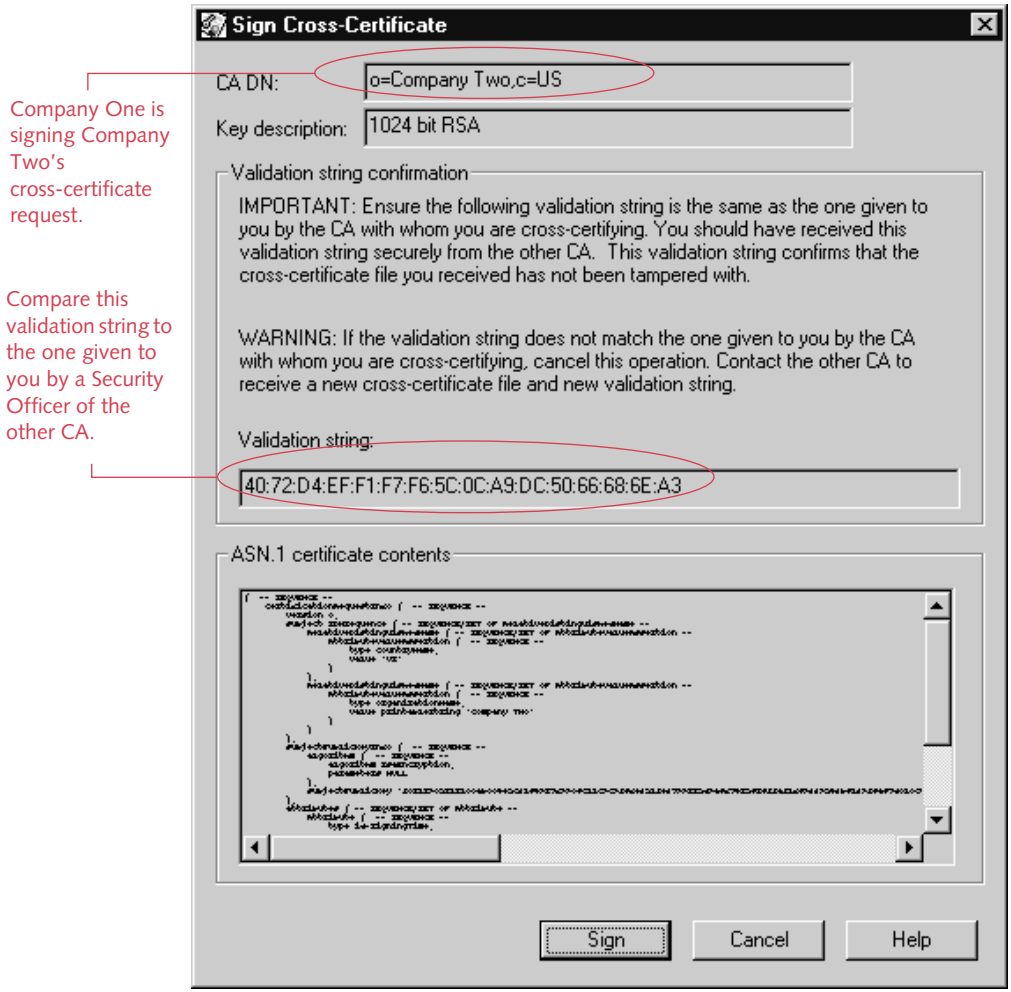
Click the file you received from the Security Officer for the other CA.



4 Locate and click the cross-certificate request file and click *Open*.

The cross-certificate request file was provided to you by a Security Officer of the CA you are cross-certifying with. The file has either a .der or .pem extension (for example, company2crosscert.der).

The Sign Cross-Certificate dialog box appears.



5 Compare the validation string in the dialog box to the validation string that a Security Officer of the other CA gave you.

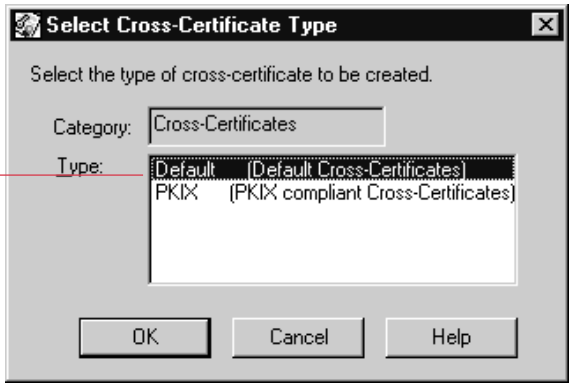
This validation string is created based on an MD5 hash of the PKCS #7 response.

If the validation strings do not match, click *Cancel*. The cross-certificate request file may have been tampered with. Ask the Security Officer of the other CA for a new cross-certificate request file.

6 If the validation strings match, click *Sign*.

The Select Cross-Certificate Type dialog box appears.

Click the type of cross-certificate you want to create.



Only the existing cross-certificate types appear. For information about cross-certificate types, see "Customizing certificates" on page 527.

7 Click the type of cross-certificate you want to create and click *OK*.

The Authorization Requested dialog box appears.

8 Authorize the operation and click *OK*.

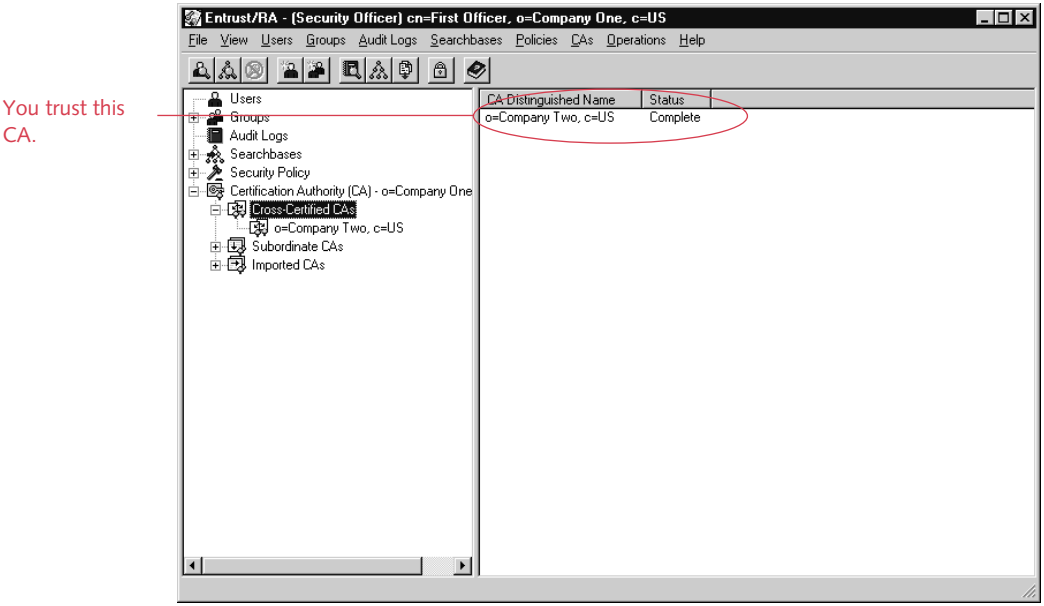
A dialog box appears when the cross-certification is complete.

9 Click *OK*.

You have now signed a cross-certificate.

Note: Cross-certificates are valid for 36 months, by default (see the "Setting up the Certification Authority" chapter in *Using Entrust/PKI 6.0 on Windows*). You must update cross-certificates before they expire to ensure there is no interruption in service for users in the cross-certified CAs (see "Updating a cross-certificate" on page 241).

The CA that initiated cross-certification appears in the Entrust/RA window under *Cross-Certified CAs*.



Importing the cross-certificate you requested

The cross-certificate you requested is the certificate that contains your CA's key and that is signed by the other CA, allowing the other CA to trust your CA. You can now import this certificate from another CA during offline cross-certification.

As with offline cross-certification, importing the cross-certificate you requested is a two step process: one CA writes the cross-certificate it signed to a file and delivers it securely to the other CA, and the other CA then imports this cross-certificate into their database, and optionally into their Directory. The procedures that follow describe an exchange between two CAs: Company One and Company Two.

To export the cross-certificate signed by Company One

1. Log in to Entrust/RA as a Security Officer or an administrative user with appropriate permissions for Company One.
2. Click the "+" symbol beside the *Certification Authority* icon in the tree view of the Entrust/RA window.
3. Click the "+" symbol beside the *Cross-Certified CAs* icon.

   Cross-certified CAs appear.
4. Right-click the CA of Company Two and click Write Cross-Certificate I Signed to file in the pop-up menu.

The Export Cross-Certificate dialog box appears.

5 Choose a destination folder and type a name for the cross-certificate file.

You can choose to export the certificate as raw certificate data, or you can export the certificate in the PKCS#7 format, as defined by the PKCS#7 standard. You can also choose the file format to be binary (an extension of .der is appended to the filename), or PEM (an extension of .pem is appended to the filename). The export options and the file formats are provided for interoperability with PKIs from other vendors. Entrust/PKI can read certificates exported in any combination of the available export options and file formats.

6 Click *Save*.

When the cross-certificate has been written to file, a dialog box appears displaying the validation string. The Security Officer at Company Two will verify

the validation string to confirm that the cross-certificate file has not been tampered with.

7 Copy the validation string down on paper and click *OK*.

8 Deliver the cross-certificate file (that is, the file with the .der or .pem extension) to Company Two in a secure manner.

You have now exported the cross-certificate signed by Company One.

To import the cross-certificate signed by Company One

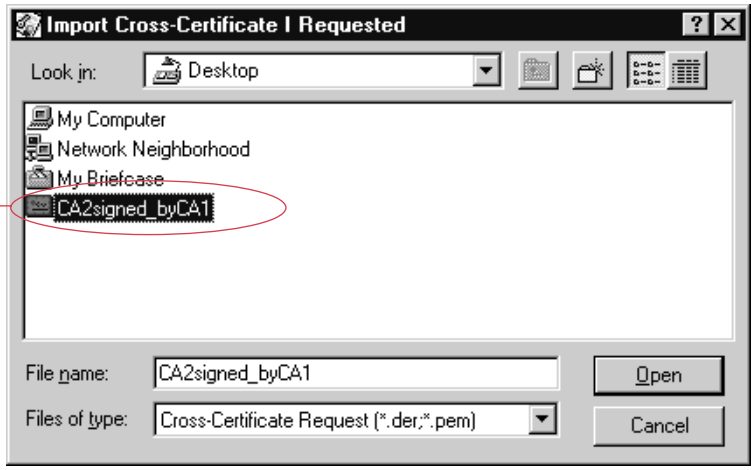
1 Log in to Entrust/RA as a Security Officer or an administrative user with appropriate permissions for Company Two.

2 Click the "+" symbol beside the *Certification Authority* icon in the tree view of the Entrust/RA window.

3 Right-click the *Cross-Certified CAs* icon and click Offline Cross-Certification > Import cross-certificate I requested in the pop-up menu.

The Import Cross-Certificate I Requested dialog box appears.

Click the cross-certificate file you received from the Security Officer of the other CA.

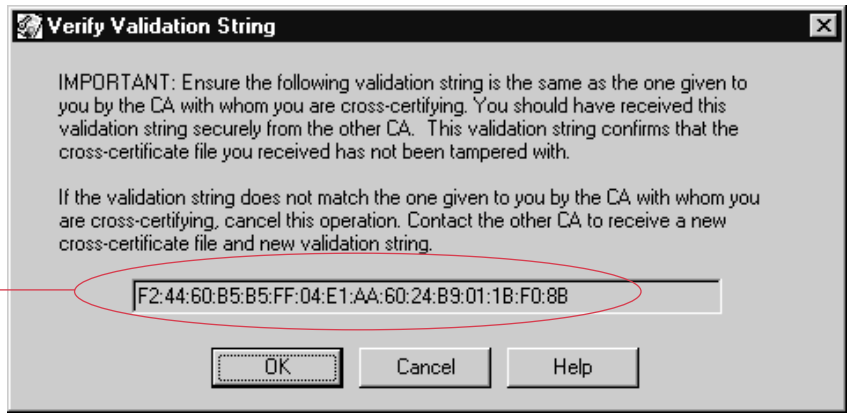


4 Locate and click the cross-certificate file from Company One and click *Open*.

The cross-certificate file was provided to you by a Security Officer of Company One. The file has a .der or a .pem extension (for example, CA2signed_byCA1.der).

The Verify Validation String dialog box appears.

Compare this validation string to the one given to you by a Security Officer of the other CA.



5 Compare the validation string in the dialog box to the validation string that a Security Officer of the other CA gave you.

If the validation strings do not match, click *Cancel*. The cross-certificate file may have been tampered with. Ask the Security Officer of the other CA for a new cross-certificate file.

6 If the validation strings match, click *OK*.

The Authorization Requested dialog box appears.

7 Authorize the operation and click *OK*.

When the importation is complete, a dialog box appears confirming the operation completed successfully.

8 Click *OK*.

You have now imported the cross-certificate you requested.

Note: Cross-certificates are valid for 36 months, by default (see "Setting up the Certification Authority" in *Administering Entrust/PKI 6.0 on Windows*). You must update cross-certificates before they expire (see "Updating a cross-certificate" on page 241) to ensure there is no interruption in service for users in the cross-certified CAs.

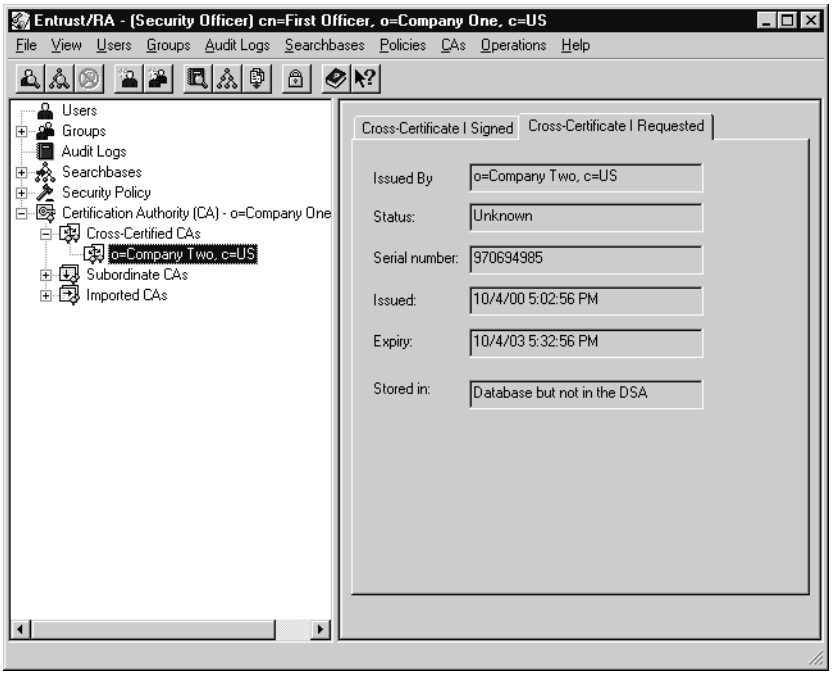
To add the cross-certificate you requested to the Directory

1 Log in to Entrust/RA as a Security Officer or an administrative user with appropriate permissions.

2 Click the "+" symbol beside the *Certification Authority* icon in the tree view of the Entrust/RA window.

3 Click the "+" symbol beside the *Cross-Certified CAs* icon.

The cross-certified CAs appear in the Entrust/RA window.



4 In the tree-view, right-click the CA that issued the certificate you want to add to the Directory, and click Add certificate I requested to Directory in the pop-up menu.

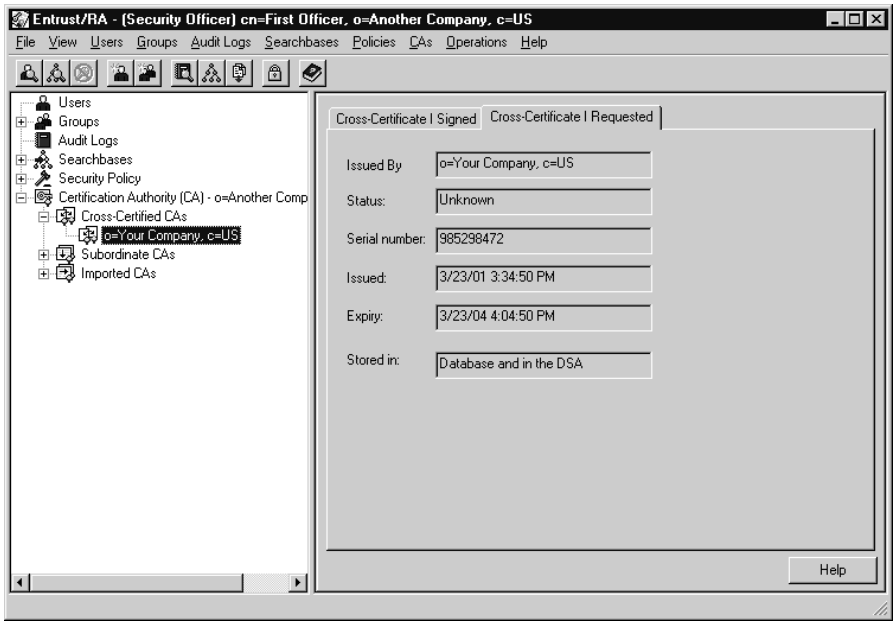
The Authorization Requested dialog box appears.

5 Authorize the operation and click *OK*.

When the cross-certificate has been added to the Directory, a dialog box appears, confirming that the operation completed successfully.

6 Click *OK*.

You have now added the cross-certificate you requested to the Directory. The status is updated in the *Cross-Certificate I Requested* property page.

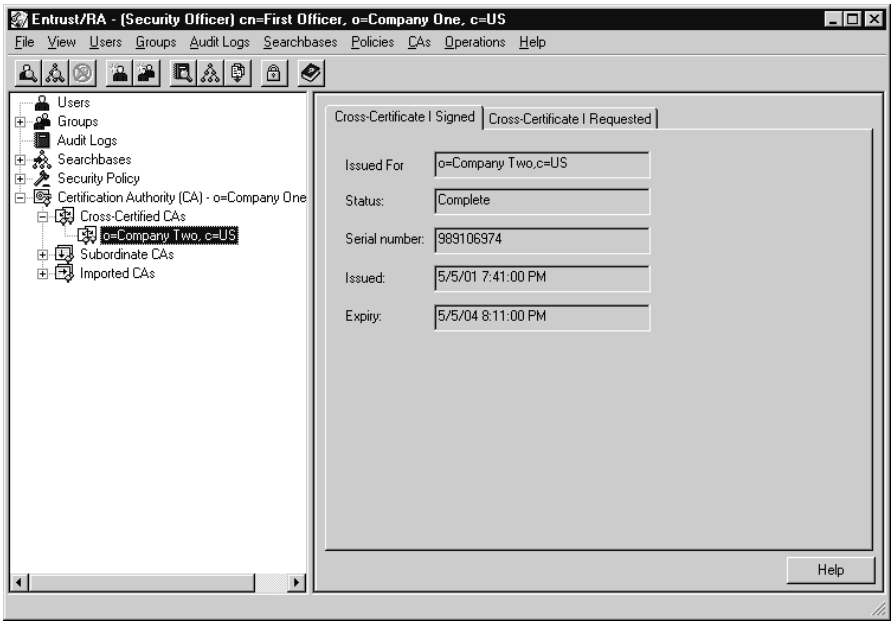


Displaying information about cross-certified CAs

You can display information about a CA that you are cross-certified with, and you can display information about a CA that is cross-certified with you.

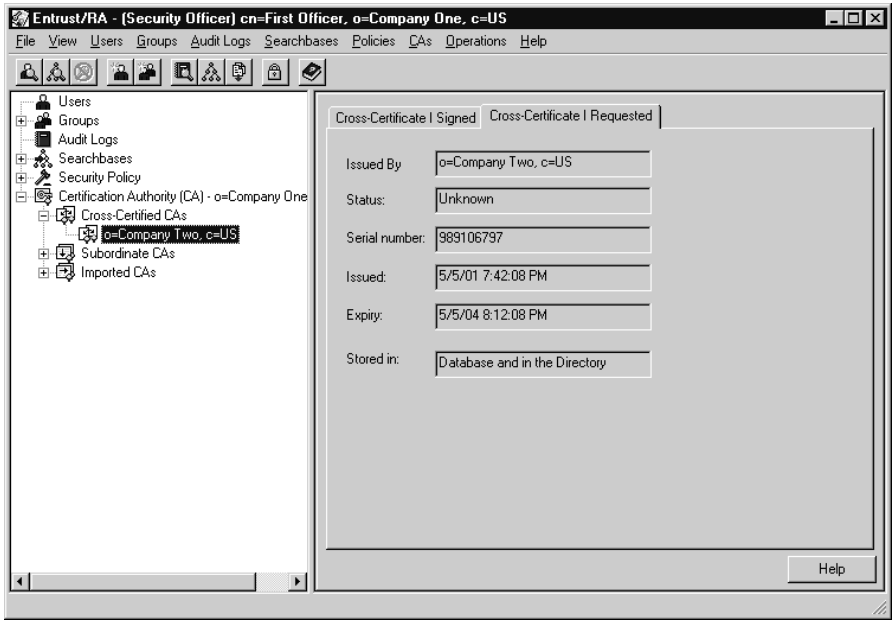
The *Cross-Certificate I Signed* property page contains information about a cross-certificate that you created and signed, and that contains another CA's verification public key. The *Cross-Certificate I Requested* property page contains information about a cross-certificate created and signed by another CA, which contains your verification public key.

For example, suppose that Company One and Company Two mutually cross-certify. The *Cross-Certificate I Signed* property page in the Entrust/RA window for Company One shows information about the cross-certificate that contains Company Two's verification public key. This cross-certificate was created and signed by Company One.



The *Cross-Certificate I Requested* property page in the Entrust/RA window for Company One shows information about the cross-certificate that contains Company One's verification public key. This cross-certificate was created and signed by Company Two. If you performed offline cross-certification, you must import the cross-certificate signed by the other CA before you can display information on the *Cross-Certificate I Requested* property page. For more

information on importing this cross-certificate, see "Importing the cross-certificate you requested" on page 226.



These property pages display the following information:

- *Issued For* and *Issued By* contains the DN of the CA.
- *Status* contains the state of the cross-certificate. The state *Complete* means the cross-certificate has been signed or received successfully. The state *Revoked* means the cross-certificate has been revoked and is no longer valid (see "Revoking a cross-certificate" on page 238). If you performed offline cross-certification and imported the "cross-certificate I requested", the status in the *Cross-Certificate I Requested* property page appears as *Unknown*, because this cross-certificate is an unmanaged certificate.
- *Serial number* contains the serial number of the cross-certificate assigned by Entrust/Authority.
- *Issued* contains the date and time that the cross-certificate was signed.
- *Expiry* contains the date and time that the cross-certificate will expire (see "Updating a cross-certificate" on page 241 for information on how to update a cross-certificate that is about to expire).
- *Stored in* appears just in the *Cross-Certificate I Requested* property page. It indicates whether or not you have removed from the Directory the cross-certificate you requested. You may no longer trust the other CA, but they may still trust you. In this case, when you remove this cross-certificate from the Directory, the status of *Stored in* reads *Database but not in the*

Directory. If you still trust the other CA, and have not removed from the Directory the cross-certificate you requested, the status of *Stored in* reads *Database and in the Directory*.

Note: This information does not apply to cross-certificates signed by your CA, so it does not appear in the *Cross-Certificate I Signed* property page.

To display information about cross-certified CAs

1. Log in to Entrust/RA as a Security Officer or an administrative user with appropriate permissions.
2. Click the "+" symbol beside the *Certification Authority* icon in the tree view of the Entrust/RA window.
3. Click the "+" symbol beside the *Cross-Certified CAs* icon.

   Cross-certified CAs appear.
4. Click the CA you want information about.

   The *Cross-Certificate I Signed* property page appears. If available, the *Cross-Certificate I Requested* tab also appears.

   You have now displayed information about cross-certified CAs.

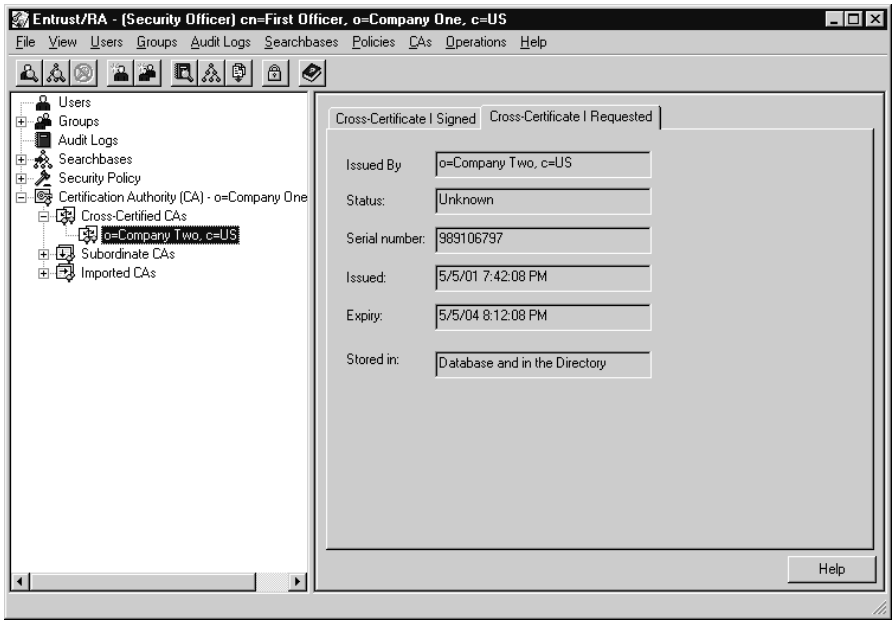
Removing the cross-certificate you requested from the Directory

You can only revoke cross-certificates signed by your CA. However, if another CA revokes a cross-certificate that contains your CA's key (that is, if they revoke the cross-certificate that they signed), you can update your *Cross-Certificate I Requested* property page to reflect this information. This is done by removing from your Directory the cross-certificate you requested. When you complete this operation, the status of the *Stored in* field in the *Cross-Certificate I Requested* property page will change from *Database and in the Directory* to *Database but not in the Directory*.

To remove the cross-certificate you requested from the Directory

1. Log in to Entrust/RA as a Security Officer or an administrative user with appropriate permissions.
2. Click the "+" symbol beside the *Certification Authority* icon in the tree view of the Entrust/RA window.
3. Click the "+" symbol beside the *Cross-Certified CAs* icon.

   The cross-certified CAs appear in the Entrust/RA window.



4 Right-click the CA that issued the certificate you want to remove from the Directory, and click Remove certificate I requested from Directory in the pop-up menu.

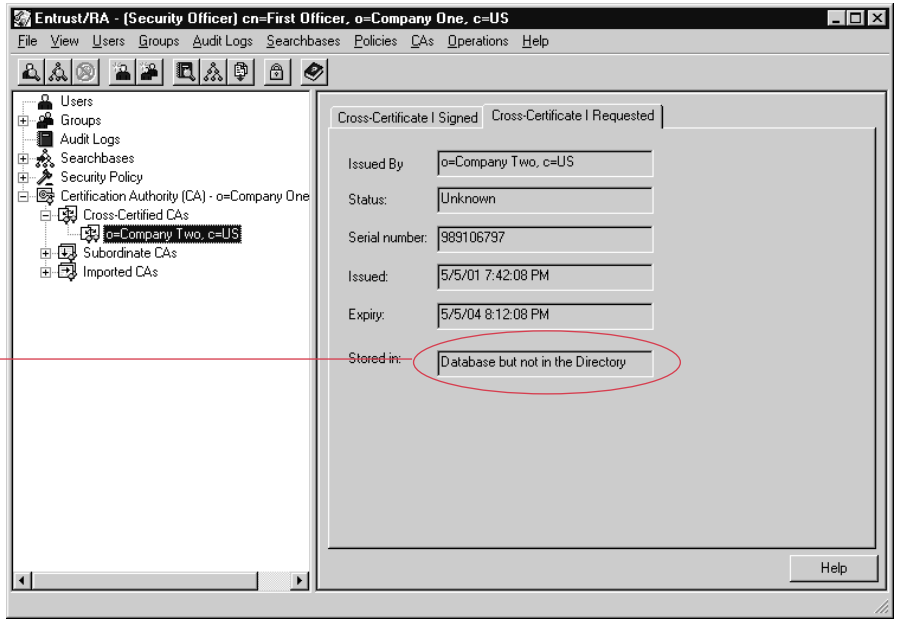
The Authorization Requested dialog box appears.

5 Authorize the operation and click *OK*.

When the cross-certificate you requested has been removed from the Directory, a dialog box appears, confirming that the operation completed successfully.

6 Click *OK*.

You have now removed the cross-certificate you requested from the Directory. The status is updated in the *Cross-Certificate I Requested* property page.



The cross-certificate you requested is no longer stored in your Directory.

Revoking a cross-certificate

Revoking a cross-certificate disables the extended third-party trust (or with unilateral cross-certification, one-way trust) previously established between two CAs. For example, suppose that Company One and Company Two mutually cross-certify. If Company One revokes the cross-certificate it signed for Company Two, users who belong to Company One can no longer decrypt data from and sign data for users who belong to Company Two. However, users in Company Two can still decrypt data from and sign data for users who belong to Company One.

You can only revoke cross-certificates signed by your CA. You revoke a cross-certificate for any of the following reasons:

- You no longer trust the other CA.
- Users in your CA no longer need to exchange secured files with users in the other CA.
- If a cross-certificate is about to expire and you want to cross-certify the CA again before the cross-certificate expires, you may want to revoke the cross-certificate before you cross-certify the other CA again.

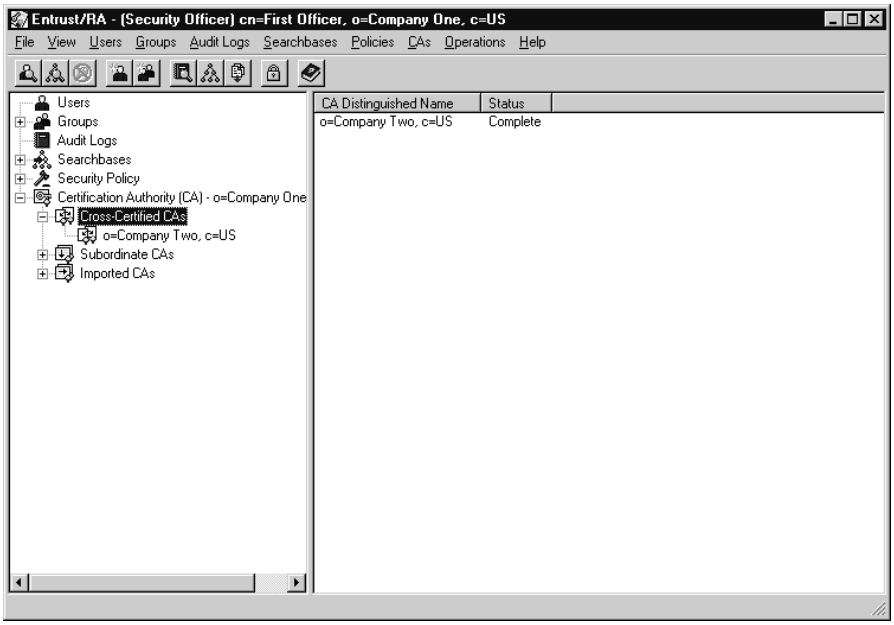
When you revoke a cross-certificate, you are prompted for a reason for the revocation.

Revoked cross-certificates are referenced in an authority revocation list (ARL) stored in the Directory. When users verify a certificate that was signed by a cross-certified CA, the Entrust-Ready application obtains a copy of the ARL from the Directory. By default, ARLs are updated every six hours, but are updated immediately when a cross-certificate is revoked to ensure there is no delay before users become aware of a revoked cross-certificate. (One exception is when the ARL is cached at the local Entrust-Ready application. In this case, the application would not update its ARL until its cached ARL expires.)

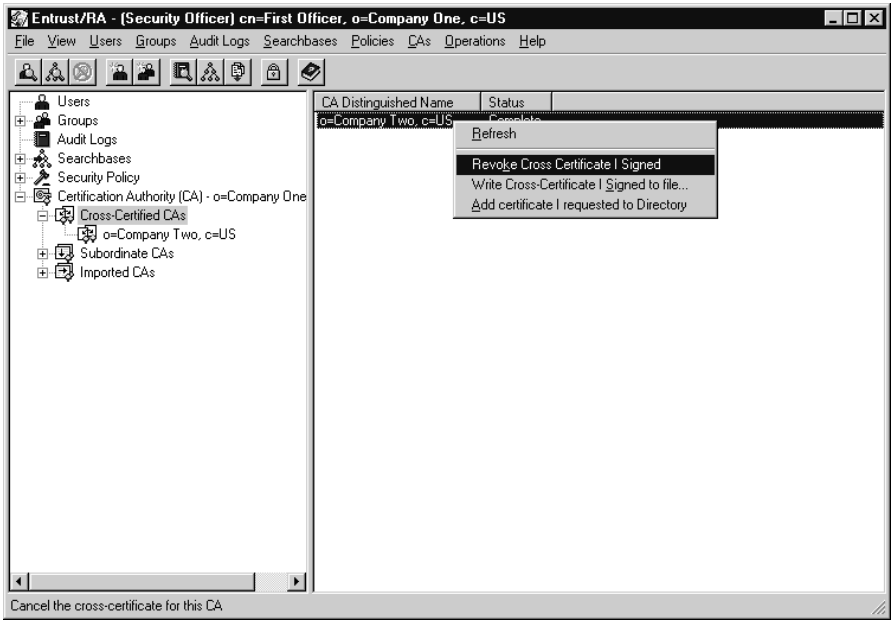
To revoke a cross-certificate

1. Log in to Entrust/RA as a Security Officer or an administrative user with appropriate permissions (see the "Setting up the Certification Authority (CA)" chapter in *Using Entrust/PKI 6.0 on Windows*).
2. Click "+" beside the *Certification Authority* icon in the tree view of the Entrust/RA window.
3. Click "+" beside the *Cross-Certified CAs* icon.

The cross-certified CAs appear in the Entrust/RA window.



4 Right-click the CA whose certificate you want to revoke and click Revoke Cross-Certificate I Signed in the pop-up menu.



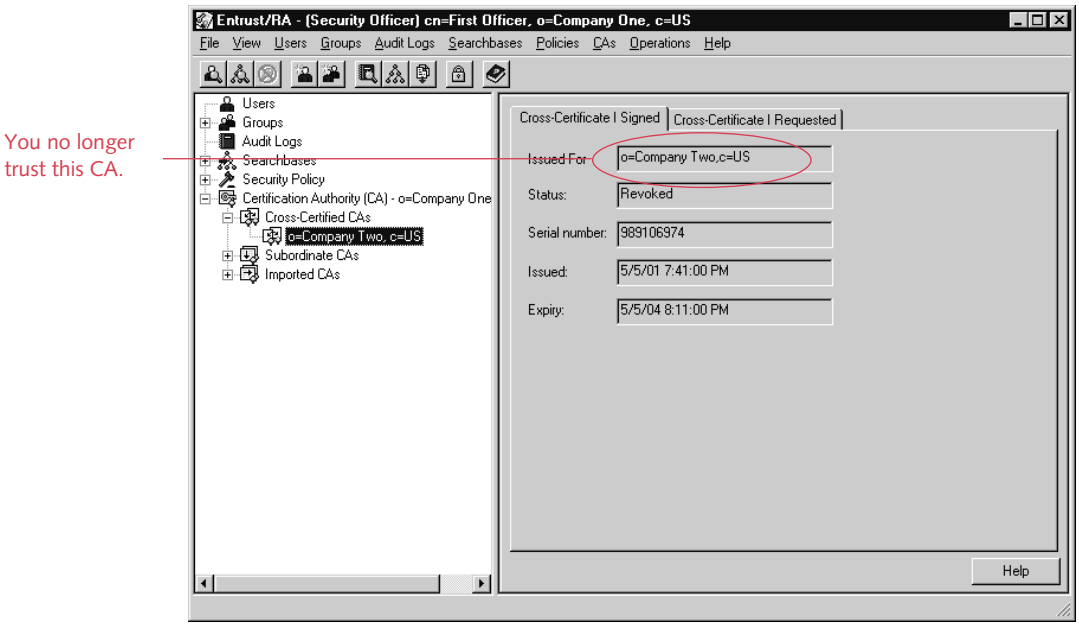
The Authorization Requested dialog box appears.

5 Authorize the operation and click *OK*.

A confirmation dialog box appears when the revocation is complete.

6 Click *OK*.

You have now revoked a cross-certificate. The status is updated in the *Cross-Certificate I Signed* property page.



Updating a cross-certificate

Cross-certificates are valid for 36 months, by default. Security Officers can change the lifetime of cross-certificates using the Administration Policy (see the "Setting up the Certification Authority (CA)" chapter in *Using Entrust/PKI 6.0 on Windows*). For both online and offline cross-certificates, the minimum lifetime is 2 months and the maximum lifetime is 60 months.

Unlike users' certificates, cross-certificates are never automatically updated. Therefore, Security Officers must keep track of cross-certificate lifetimes and update the cross-certificates before they expire to ensure there is no interruption in service for users in the cross-certified CAs.

Note: When you updated a cross-certificate, the old cross-certificate is automatically revoked.

To update cross-certificates

Follow the steps in the appropriate section:

- If you are performing mutual or unilateral online cross-certification, see "Performing online cross-certification" on page 208.
- If you are performing mutual or unilateral offline cross-certification, see "Performing offline cross-certification" on page 218.

You have now updated the cross-certificates.

Chapter 7

Creating a hierarchy of CAs

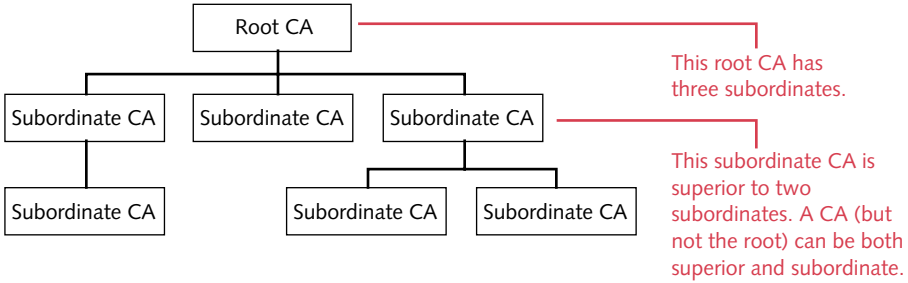
A hierarchy of CAs is way of arranging two or more CAs in a restrictive trust relationship. In a hierarchy, a CA at any level signs the CA certificates of the CAs immediately below it in the hierarchy. Trust derives from a single "trust anchor" called the root CA at the top of the hierarchy. Only the root CA can cross-certify with other CAs.

Creating a strict hierarchy of CAs is an advanced feature of Entrust/PKI that's best configured by experienced Security Officers. You should also have some Directory administration experience in creating new Directory branches and administrators, and chaining to other Directories.

About hierarchy of CAs

A hierarchy of CAs is a way of arranging two or more CAs into a restrictive, hierarchical trust relationship. A hierarchy is an inverted tree structure that contains superior and subordinate CAs. At the top of a hierarchy is a single "root" CA, which acts as the trust anchor for all CAs in the hierarchy. See Figure 2.

Figure 2: Example of a hierarchy of CAs, showing an inverted tree structure



The terms "superior" and "subordinate" refer to pairs of CAs that are adjacent vertically. A CA can have several subordinates, but it can have only one superior. Also, a CA can be both superior and subordinate. In Figure 2, the middle tier of CAs is subordinate to the root, but two CAs in the middle tier are in turn superior to other CAs.

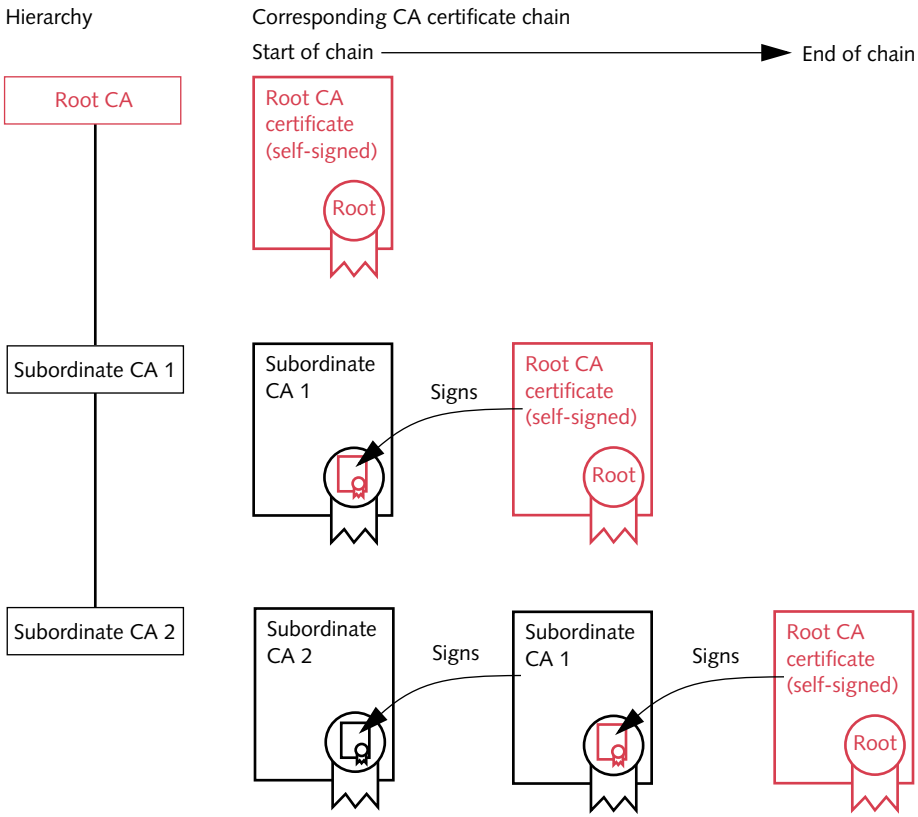
The distinguishing aspect of a hierarchy is that a superior CA signs the CA certificates of its immediate subordinates. Signing another CA's certificates is called adding a subordinate. The main difference between a CA that's in a hierarchy and a CA that's not, is where the CAs derive their trust.

A CA that's not in a hierarchy signs its own CA certificates. A standalone CA is its own trust anchor: technically, it's a root CA with no subordinates. By default, when you install Entrust/PKI, you install a root CA. Root CAs can cross-certify with each other to create a network of trust between peers.

A CA that's in a hierarchy has its certificates signed by its direct superior. A superior may be the root of a hierarchy, or some level of subordinate beneath the root. The pattern of superiors signing their subordinates' certificates eventually converges at the root, which signs its own certificates. Each subordinate is at the end of a certificate chain that begins with the root's certificate. In effect, all CAs and users in a hierarchy can trust each other, because they all share a trust anchor.

Figure 3 on page 245 shows a hierarchy of three CAs, and what the certificates look like for each CA. Note that a CA deeper in the hierarchy has a longer certificate chain.

Figure 3: Example of a CA certificate chain for a three-level hierarchy



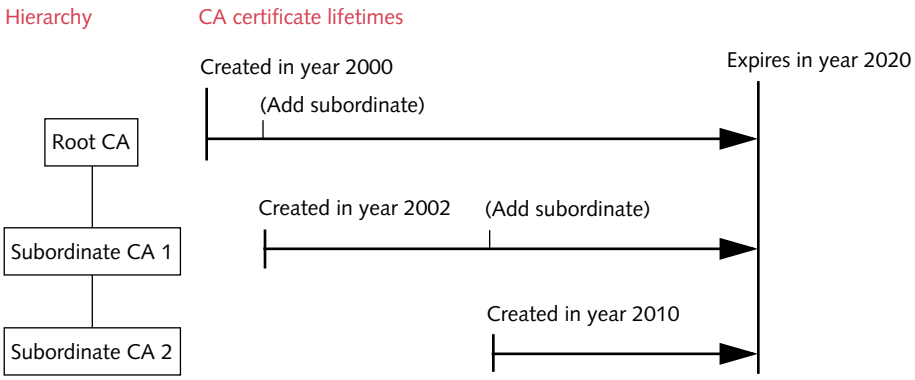
Implications of a hierarchy

The following are some implications of creating a CA in a hierarchy:

- You must decide when you install a CA whether it is to be a root or a subordinate. (The default is root.) The reason is, a CA initializes differently in each case. If it's a root, it creates its own keys and signs its own certificates. If it's a subordinate, it creates its keys then requests its superior to sign them.
- A subordinate CA can't cross-certify with other CAs. To do so may create unintended trust relationships. Since the purpose of a hierarchy is to establish trust from a single point, the root CA, only the root can cross-certify with other CAs.

- When you revoke a subordinate, by extension you revoke everything beneath the subordinate—including its own subordinate CAs.
- The effective lifetime of a subordinate CA's certificates decreases with each level in the hierarchy beneath the root. To understand these decreasing lifetimes, consider a root that has just created its CA certificate with a lifetime of twenty years. See Figure 4. Suppose that after two years, the root adds a subordinate and signs its certificates. The subordinate's certificates will expire in eighteen years, on the same date as the root's certificates. Now suppose that in another eight years, the subordinate adds a subordinate. The new subordinate's certificates will expire in only ten years, but also on the same date as the root's certificates.

Figure 4: Decreasing lifetimes of subordinate CA certificates



When to create a hierarchy of CAs

There are two basic reasons for using a hierarchical trust model (as opposed to a trust model based on mutual cross-certification). First, in a hierarchy, the trust anchor (root CA) controls the trust relationships of all CAs in the hierarchy. This is desirable when, for example, you don't want CAs in your organization to cross-certify with other CAs.

Another reason for using a hierarchical trust model is it's easy to distribute the trust anchor to other entities—especially to those that are Web based and aren't Entrust users. It's much easier to distribute the certificate of one CA (the root), rather than the certificates of several CAs, as would be necessary in a cross-certified trust model. In the latter model, there's also the issue of knowing which certificate to give an entity. It's easiest if all entities that are establishing trust with a group of CAs receive just one certificate—the root CA's.

The purpose of a hierarchy is to establish a single point of trust for two or more CAs. Therefore, create a hierarchy of CAs when

- you want control of all trust relationships of a group of CAs. That is, you want the group of CAs to derive all its trust from a single trust anchor. For instance, you may not want CAs in your organization to cross-certify with CAs outside your organization.
- you are issuing Web certificates predominately. Web browsers assume CAs are in a hierarchy. Browsers do not support normal cross-certification to establish trust.
- you need to join an existing hierarchy.

Creating a hierarchy of CAs

You create a hierarchy of CAs top-down by adding one or more subordinates to a root CA. Subordinates can in turn add subordinates, thereby extending the hierarchy. You cannot create a hierarchy by making an existing root CA subordinate to another CA.

How you create a hierarchy depends on where you are in the hierarchy, and what kind of CA you are, or plan to add (that is, Entrust/PKI or a third-party CA). There are three basic scenarios to consider:

- You want to add an online subordinate. To do this, you first create an installation password for the subordinate at the superior CA. Then you install the subordinate, using the password, in an online process. For instructions, see "Adding a subordinate" on page 248, and "To add an online subordinate" on page 249.
- You want to add an offline subordinate. To do this, you first install the subordinate. Then, at the superior, you sign the subordinate's CA certificates in an offline process. For instructions, see "Adding a subordinate" on page 248, and "To add an offline subordinate" on page 250.
- You want to become subordinate to a third-party superior. To do this, you first install the subordinate CA. Then you initialize it in an offline process, in which your superior signs your CA certificates. For instructions, see "Becoming subordinate to an Entrust or third-party superior CA" on page 252.

In the online process of adding a subordinate, the superior CA uses a network connection to communicate directly with the subordinate and establish trust.

In the offline process of adding a subordinate, you establish trust between the superior and subordinate CAs manually by exchanging request and response files (for instance, by ftp, e-mail, or disk).

Adding a subordinate

Before you add a subordinate, you must ensure that the subordinate and superior CAs can access each other's Directory entries.

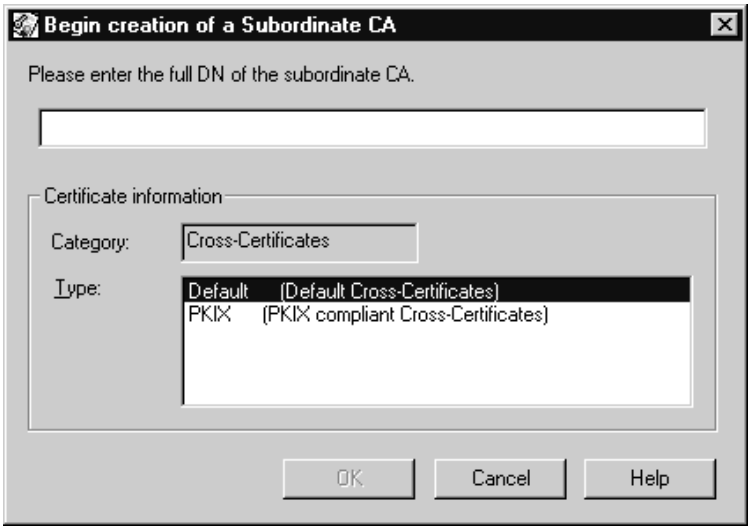
How you add the subordinate depends on whether it's an online or an offline subordinate. If it's an online subordinate, you first create an installation password for it, then install it using the password. If it's an offline subordinate, you first install it, then initialize it in an offline process.

See "To add an online subordinate" on page 249, or "To add an offline subordinate" on page 250, as required.

To add an online subordinate

1. Log in to the superior CA using Entrust/RA.
2. Right-click *Subordinate CAs* in the tree view (it appears under the *Certification Authority (CA)* entry) and click Begin creation of a subordinate CA > Online in the pop-up menu.

   The Begin creation of a Subordinate CA dialog box appears.



3. In the *Please enter the full DN of the subordinate CA* field, enter the full distinguished name (DN) of the subordinate CA you want to add, at the top of the dialog box. For example, enter o=Another Company,c=US.

   In the *Certificate information* group box, choose a type of certificate to issue to the subordinate CA. (You can't change the certificate category, because technically, subordinate CA certificates are a kind of cross-certificate.) If you want to create subordinate CAs that have customized certificates, you must define new cross-certificate types in the master.certspec file. For more information on defining cross-certificate types, see "Customizing cross-certificates" on page 314.

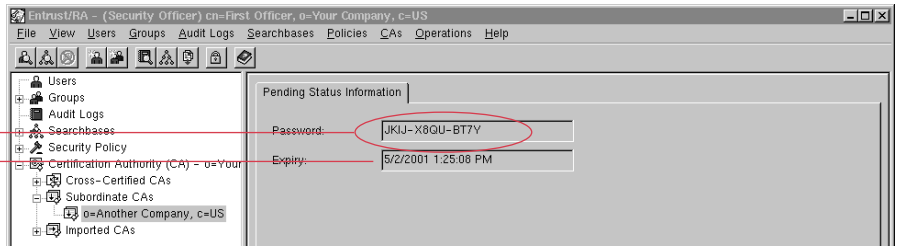
   Click *OK* and authorize the operation.

   The property page updates to show the pending status information for the subordinate CA you added. Note that the password expires in three days. If you

do not complete the add subordinate operation before the password expires, you must cancel the operation and start again.

Give this password, your IP address, your DN, and your key-pair algorithm to the subordinate.

Note that the password expires in three days.



4 Record the password, your CA DN, the IP address or DNS (Domain Name Server) name of your CA server, the port number of your CA server (as in, the value for the CMPlisten setting in the Comms section of the entmgr.ini file), and the key pair algorithm (RSA, DSA, or ECDSA) as well as the hashing algorithm (SHA1 or MD5) that your CA uses.

Give this information securely to the person who will install the subordinate CA. They need this information to configure the subordinate during first-time initialization.

5 Install the subordinate. During installation, choose "Subordinate" and "Online" when prompted, and during first-time initialization, enter the password you received from your superior. For more information on installing a CA, see *Installing Entrust/PKI 6.0 on Windows*.

You have now installed an online subordinate.

To view the completed operation, in Entrust/RA at the superior CA, you must click CAs > Refresh.

To add an offline subordinate

1 Install the subordinate CA, choosing "Subordinate" and "Offline" when you're prompted during the installation. For more information on installing a CA, see *Installing Entrust/PKI 6.0 on Windows*.

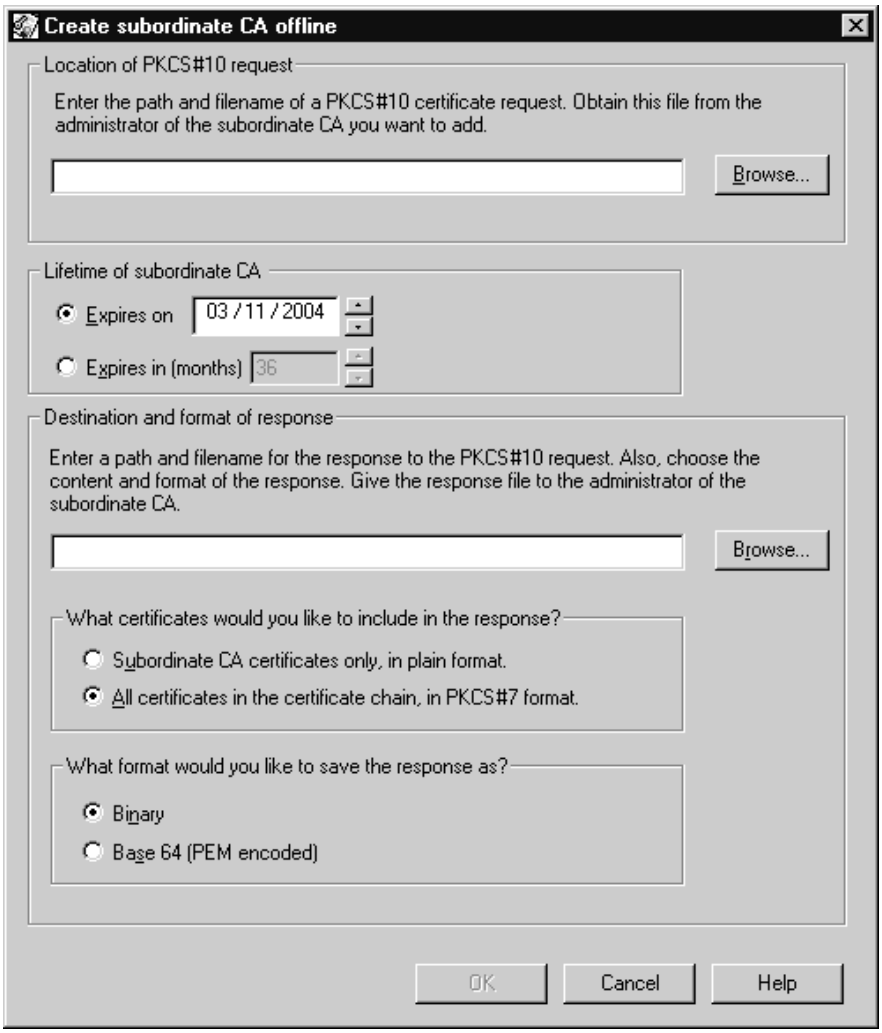
2 At the subordinate CA, generate a PKCS #10 request to sign the subordinate's CA certificates. See "To ask a superior CA to generate a subordinate CA certificate" on page 253. If you are adding a third-party subordinate CA, consult your third-party CA's documentation for instructions on creating this request.

Transmit the request (for instance, by e-mail, ftp, or on disk) to the superior CA.

3 Log in to Entust/RA as a Security Officer at the superior CA.

4 Click *Certification Authority (CA)* in the tree view of Entrust/RA to expand it. Right-click *Subordinate CAs*, and click Begin creation of a subordinate CA > Offline in the pop-up menu.

The Create subordinate CA offline dialog box appears.



In the *Location of PKCS #10 request* field, enter the path and filename of PKCS #10 request from the subordinate CA. (This request was created in Step 2.) Or, click *Browse* to locate the request.

In the *Lifetime of subordinate CA* group box, in the *Expires on* field, enter the date in the form (mm, dd, yyyy) on which you want this certificate to expire. Or,

in the *Expires in (months)* field, enter the number of months for which you want the certificate to remain valid.

Attention: Currently, when the subordinate CA's certificates expire, the CA will cease to be operational. The subordinate CA's keys will not update. The lifetime of the subordinate CA certificate should be chosen accordingly.

In the *Destination and format of response* field, enter the path and filename of the response file that you will create. Or, click *Browse* to choose a location. A response file contains the certificates of the subordinate CA, signed by the superior CA.

In the *What certificates would you like to include in the response?* group box, choose an option. Choose *Subordinate CA certificates only, in plaintext format* for a response that contains only the subordinate CA certificates, in a certificate signed by the superior. Choose *All certificates in the certificate chain, in PKCS #7 format* for a response that contains the subordinate's CA certificates, in a certificate signed by the superior, plus the certificates of all CAs in the chain up to and including the root. If you're installing an Entrust subordinate CA, choose *All certificates in the certificate chain, in PKCS #7 format*.

In the *What format would you like to save the response as?* group box, you can choose the file format to be *Binary* (an extension of .der is appended to the filename), or *Base 64 (PEM encoded)* (an extension of .pem is appended to the filename). The choice of formats is provided for interoperability with PKIs from other vendors. Entrust/PKI can read files in either format.

Click *OK*.

Entrust reads the request file and saves a response file that contains the signed certificates of the subordinate CA.

5 Transmit the response file (for instance, by ftp, e-mail, or disk) to the subordinate CA.

6 At the subordinate CA, import the response file. See "To import a PKCS #7 response into Entrust/Authority" on page 255. Aside from importing the response file, there may be other steps you also need to do if you are adding a third-party offline subordinate CA. Consult the documentation for your third-party CA for details.

You have now added an offline subordinate CA.

Becoming subordinate to an Entrust or third-party superior CA

Before you become subordinate to an Entrust or third-party CA, you must ensure that the two CAs can access each other's Directory entries. Consult with the

administrator of the superior CA to see if you should install your own Directory, or use an existing Directory (such as the superior CA's).

You must use Entrust Master Control Command Shell to perform offline initialization of a CA that is subordinate to a third-party or Entrust superior CA. This functionality is not available in the graphical user interface.

Offline initialization of a subordinate CA is a three-step process:

1. The subordinate CA asks the superior CA to generate a subordinate CA certificate. The subordinate CA sends this request as a message that conforms to a specific standard — Public-Key Cryptography Standards (PKCS) #10. PKCS #10 describes a syntax for certification requests. A PKCS #10 certification request *must* contain a distinguished name and a public key, and it *must* be signed by the entity that wants a certificate. This step is covered in "To ask a superior CA to generate a subordinate CA certificate" on page 253.
2. The superior CA transforms your request into an X.509 public-key certificate and sends that certificate, along with all the certificates needed to complete the path to the root CA, back to you. In Entrust/PKI, superior CAs send these certificates as messages conforming to the PKCS #7 standard. This standard defines a general syntax for messages that include cryptographic enhancements, such as digital envelopes or digital signatures. This step is covered in "To add an offline subordinate", starting at Step 3 on page 250.
3. The subordinate CA imports the PKCS #7 response into Entrust/Authority and completes the process of initializing the subordinate CA. This step is covered in "To import a PKCS #7 response into Entrust/Authority" on page 255.

   In Entrust/PKI 5.1 or newer, you can specify how your CA certificate request is encoded: binary DER (Distinguished Encoding Rules), PEM (Privacy Enhanced Mail), or PEM with header (includes BEGIN and END lines). The choice of formats is provided for interoperability with PKIs from other vendors. Entrust/PKI can read files in any of these formats.

To ask a superior CA to generate a subordinate CA certificate

1. Install the subordinate CA, choosing "Subordinate" and "Offline" when you're prompted during the installation. For more information on installing a CA, see *Installing Entrust/PKI 6.0 on Windows*.

   Note that when you install a subordinate CA offline, you create three Master Users (as when installing a CA in any other circumstance), but first-time initialization stops before the Entrust CA or First Officer profiles are created and before Entrust/Authority Master Control runs.
2. At the subordinate CA, launch the Entrust Master Control Command Shell and log in as a Master User.
3. At the prompt, enter

   ```
   subordinate make-subcert-req
   ```

This command generates a file containing a PKCS #10 request for CA certificates, as well as a verification string for the file.

The following message appears:

```
How should the PKCS#10 file be encoded?
choose one of:
1. Binary
2. PEM
3. PEM with header
4. Quit
[1] >
```

Accept the default, Binary, or select a different format, for example, enter the following to encode the PKCS #10 request for the certificate in PEM format:

```
2
```

If you select option 3 (PEM with header), the saved file will include "-----BEGIN CERTIFICATE REQUEST-----" and "-----END CERTIFICATE REQUEST-----" lines.

The following message appears:

```
Enter name for PKCS#10 file.
Default is <C:\authdata\pkcs10.pem>:
```

4 Enter a path and filename for the PKCS #10 request.

The following message appears:

```
Include extensions from CA category of certificate
specification (y/n) ? [n]
```

5 To include the extensions defined in the default CA certificate type in the PKCS #10 extensionRequest attribute, enter

```
y
```

Otherwise, accept the default value [n].

If you require extensions from the third-party CA that aren't sent with the certificate request by default, choosing "yes" here provides one way to request them.

For more information about certificate extensions, see "Customizing certificates" on page 269.

If your third-party CA is not an Entrust CA, make a note to check the user documentation for your third-party CA to confirm that the certificates being sent back from the superior CA have the following extensions in them:

- subjectKeyIdentifier

Note that the key Id format for the CA (as set by the advanced setting KeyIdMode) should match the format of the returned subjectKeyIdentifier.

- basicConstraints with "CA" field set to "true"

6 Entrust/Authority Master Control creates and saves the file. A verification string appears (for example, 8B:G6:7D:R8:33:2S:AC:9Y:3C:X0:09:R2:K5:65:4D:JW). Record this string.

This verification string is created based on an MD5 hash of the PKCS #10 request.

7 Send the PKCS #10 request to your superior CA (for instance, by ftp, e-mail, or disk).

8 Securely give the verification string you recorded in Step 5 to the administrator of the superior CA. The administrator needs this string to verify that the PKCS #10 request they receive is from you and is unaltered.

Attention: Give your verification string to the administrator securely and separately from your PKCS #10 request. Sending the request and its verification string together greatly increases the risk that undetectable tampering with the request will occur.

You have now asked a superior CA for a subordinate CA certificate.

Now proceed to Step 3 of "To add an offline subordinate" on page 250.

To import a PKCS #7 response into Entrust/Authority

1 Start Entrust Master Control Command Shell at the subordinate CA.

For more information about starting Entrust Master Control Command Shell, see "To start Entrust Master Control Command Shell from the Windows taskbar" on page 347.

2 Log in to Entrust Master Control Command Shell as a Master User.

3 At the prompt, enter

```
subordinate read-subcert
```

When prompted, enter the path and filename of the PKCS #7 response from the superior CA.

Entrust/Authority reads the response and displays a verification string (for example, 5Y:3C:X0:09:R2:K5:65:4D:QW:8B:G6:7D:R8:33:2S:AC).

This verification string is created based on an MD5 hash of the PKCS #7 response.

Entrust/Authority will recognize the format of the file as binary DER encoded, PEM, or PEM with header.

Entrust Master Control Command Shell also displays a prompt asking you if the verification string is correct.

4 Contact the superior's administrator and confirm that your verification string matches the original string at the superior.

If the verification strings don't match, the response you received may be corrupt. Enter "no" at the prompt and ask the superior's administrator to send you a new PKCS #7 response.

If the verification strings match, enter "yes" at the prompt. Entrust/Authority creates profiles for the CA and First Officer.

You have now imported a PKCS #7 response to a request for a CA certificate into Entrust/Authority.

Note: CA initialization completes after you import the PKCS #7 response. Profiles are created and you are prompted to back up the Entrust/Authority database.

Removing a subordinate CA from a hierarchy

You remove a subordinate from your hierarchy by revoking its CA certificate. The process is like that of revoking an end user or entity, but the effect is far greater. When you revoke a subordinate, you revoke a link in the chain of trust that extends to everything in the hierarchy beneath the subordinate CA. Consequently, the extent of your action may be considerable.

Revoking a subordinate CA's certificate is a more serious operation than revoking a cross-certified CA's certificates. When you revoke a cross-certificate, users in your CA can no longer trust users in the other CA. However, users in the other CA can trust still trust each other, and they can still trust users in your CA (unless the other CA revokes its cross-certificate to your CA).

When you revoke a subordinate CA's certificate, users everywhere in the hierarchy can no longer trust users in the hierarchy beneath the subordinate. For instance, users in the subordinate CA can no longer trust each other. Users in the hierarchy who are not in the revoked subordinate domain continue to trust each other because they share a trust anchor (the root CA).

Attention: Revoking a subordinate CA is permanent.

After you've revoked a subordinate CA, you can reinstall it using the same DN. For more information, see "Reinstalling a subordinate CA" on page 259.

When to revoke a subordinate CA

The circumstances for revoking a subordinate CA are rare. Revoke a subordinate CA when

- its CA signing key pair has been compromised.
- its Master Users have been compromised.
- it is issuing certificates in a manner that contravenes your organization's certification practice statement. For example, suppose your organization requires that a person be an employee for at least six months before they can become a Security Officer. If an administrator of the subordinate CA assigns a newly hired employee the role of Security Officer, that would contravene your certification practice statement.

When you revoke a subordinate CA, you are prompted to provide a reason for the revocation.

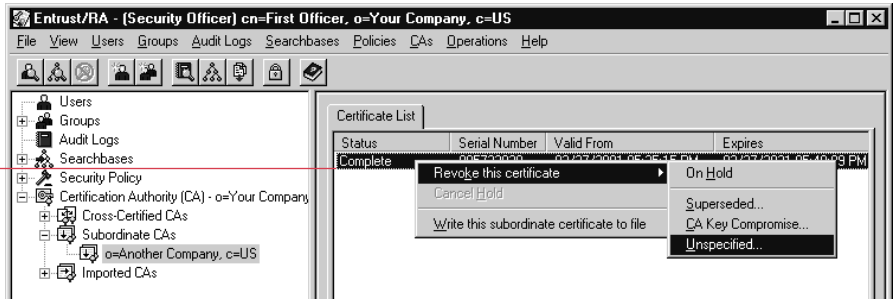
To revoke a subordinate CA's certificate at the superior CA

1. Log in to Entrust/RA as a Security Officer at the superior CA.
2. Click *Certification Authority (CA)* in the tree view of Entrust/RA to expand it, and click *Subordinate CAs* to expand it.

   Click the subordinate CA that you want to revoke.

   The *Certificate List* property page shows all certificates issued to this subordinate.
3. On the *Certificate List* property page, right-click the current certificate for the subordinate CA. In the pop-up menu, choose *Revoke this certificate* and one of the following revocation reasons:
   - Click *On Hold* if you want to suspend the activity of the subordinate CA on a temporary basis. From this state, you can later take the subordinate CA's certificate off hold (when you right-click the certificate of a subordinate CA that is on hold, the pop-up window will then give you the option to take the subordinate CA's certificate *Off Hold*), or you can revoke the certificate of the subordinate CA.
   - Click *Superseded* when you don't trust the certificate, but you know the key hasn't been compromised. For example, choose *Superseded* when you're revoking a subordinate because its administrative users (Master Users, Security Officers, and so on) have contravened your certificate practice statements.
   - Click *CA Key Compromise* when the subordinate's signing key pair has been compromised.
   - Click *Unspecified* when the other reasons don't apply.

Right-click a subordinate CA's certificate to revoke it.



4. Authorize the operation. For details on authorizing operations, see the "Setting up administrative users" chapter in *Using Entrust/PKI 6.0 on Windows*.

   You have now revoked a subordinate CA's certificate.

   If you want to reinstall the subordinate, see "Reinstalling a subordinate CA" on page 259.

Reinstalling a subordinate CA

There are two cases when you need to reinstall a subordinate CA:

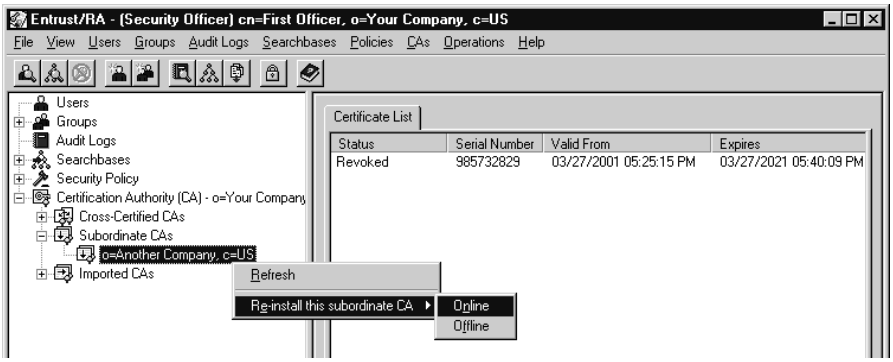
- You've revoked an Entrust or third-party subordinate for some reason, and you want to add it to your hierarchy again. (For more information on reasons why you'd revoke a subordinate, see "Removing a subordinate CA from a hierarchy" on page 257.)
- A third-party subordinate has updated its CA keys, so you need to update the subordinate's CA certificate. (Note that in Entrust/PKI 6.0, an Entrust subordinate CA can't update its keys.)

If the subordinate is an Entrust CA, you can reinstall it online. For instructions, see "To reinstall a subordinate CA online" on page 259.

If you are reinstalling or updating the certificate for an offline subordinate CA, see "To reinstall an offline subordinate CA" on page 260.

To reinstall a subordinate CA online

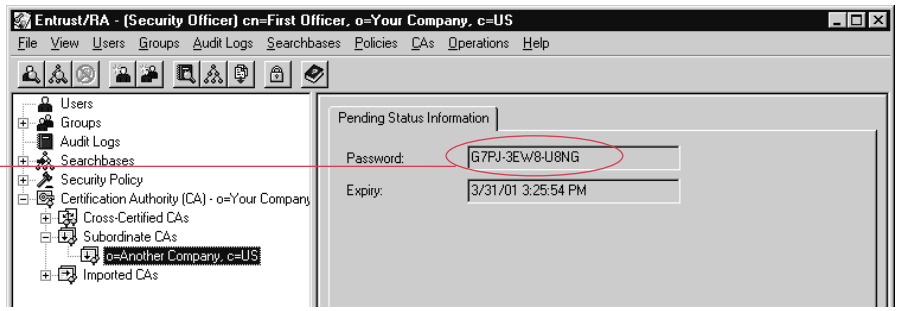
1. Log in to Entrust/RA as a Security Officer at the superior CA.
2. Click *Certification Authority (CA)* in the tree view to expand it, and click *Subordinate CAs* to expand it.
3. Right-click the subordinate CA you want to reinstall. In the pop-up menu, click Reinstall this subordinate CA > Online.



4. Authorize the operation. For details on authorizing operations, see the "Setting up administrative users" chapter in *Using Entrust/PKI 6.0 on Windows*.
5. Refresh the property page for the subordinate CA you are reinstalling by right-clicking Subordinate CAs and clicking Refresh in the pop-up menu.

The *Pending Status Information* property page appears, showing a new installation password for the subordinate CA.

Use this password to reinstall the subordinate CA.



6 Reinstall the subordinate CA, using the password on the *Pending Status Information* property page. For more information, see "To add an online subordinate" on page 249, and *Installing Entrust/PKI on Windows*.

When you've finished reinstalling the subordinate CA, its *Certificate List* property page at the superior CA shows its new certificate and any previous certificates. (To view the updated information, in Entrust/RA at the superior CA you must click CAs > Subordinate CAs > Refresh.)

You have now reinstalled an Entrust subordinate CA.

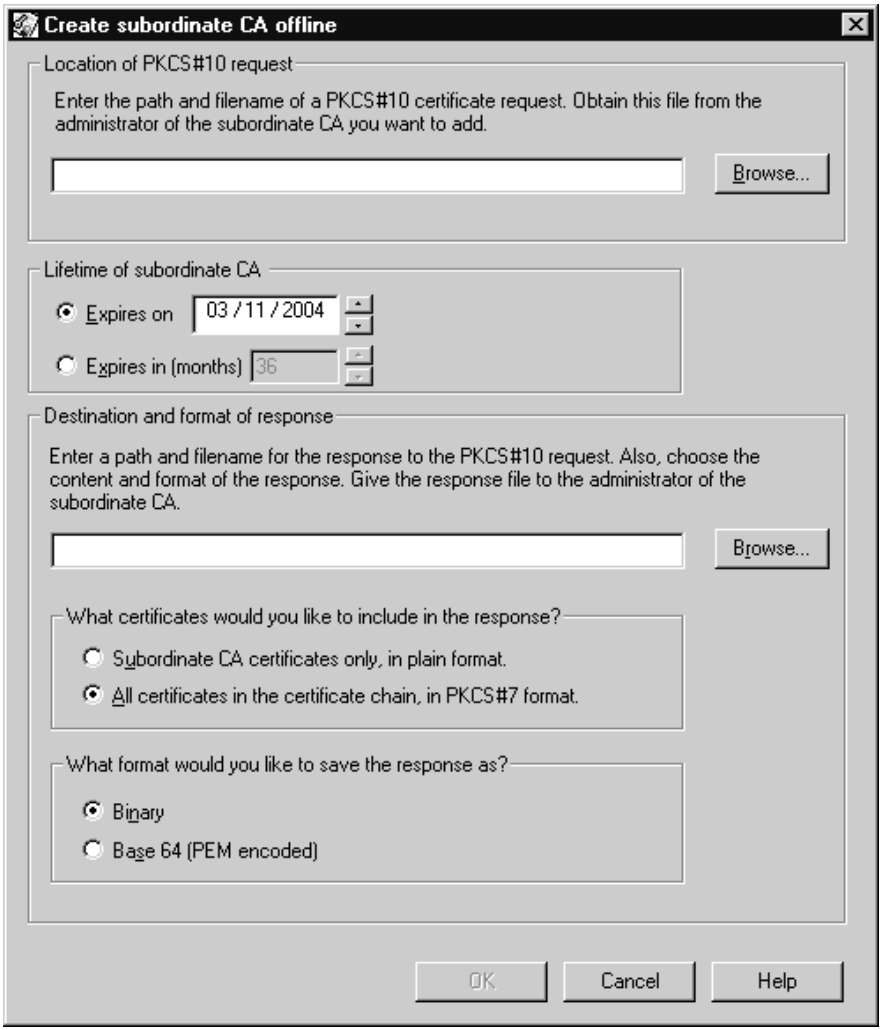
To reinstall an offline subordinate CA

1 At the offline subordinate CA, update the CA keys and create a PKCS #10 request to sign the CA certificates. Transmit the request (for instance, by e-mail, ftp, or on disk) to the superior CA.

2 At the superior CA, click *Certification Authority (CA)* in the tree view of Entrust/RA to expand it, and click the *Subordinate CAs* entry to expand it.

3 Right-click the subordinate CA you want to reinstall. In the pop-up menu, click Reinstall this subordinate CA > Offline.

The Create subordinate CA offline dialog box appears.



In the *Location of PKCS #10 request* field, enter the path and filename of the PKCS #10 request from the subordinate CA. (This request was created in Step 1.) Or, click *Browse* to locate the request.

In the *Destination and format of response* field, enter the path and filename of the response file that you will create. Or, click *Browse* to choose a location. A response file contains the CA certificates of the subordinate CA, signed by the superior CA.

In the *What certificates would you like to include in the response?* group box, choose an option. Choose *Subordinate CA certificates only, in plaintext format* for a response that contains only the subordinate CA certificates, in a certificate signed by the superior. Choose *All certificates in the certificate chain, in PKCS #7 format* for a response that contains the subordinate's CA certificates, in a certificate signed by the superior, plus the certificates of all CAs in the chain up to and including the root. If you're installing an Entrust subordinate CA, choose *All certificates in the certificate chain, in PKCS #7 format*.

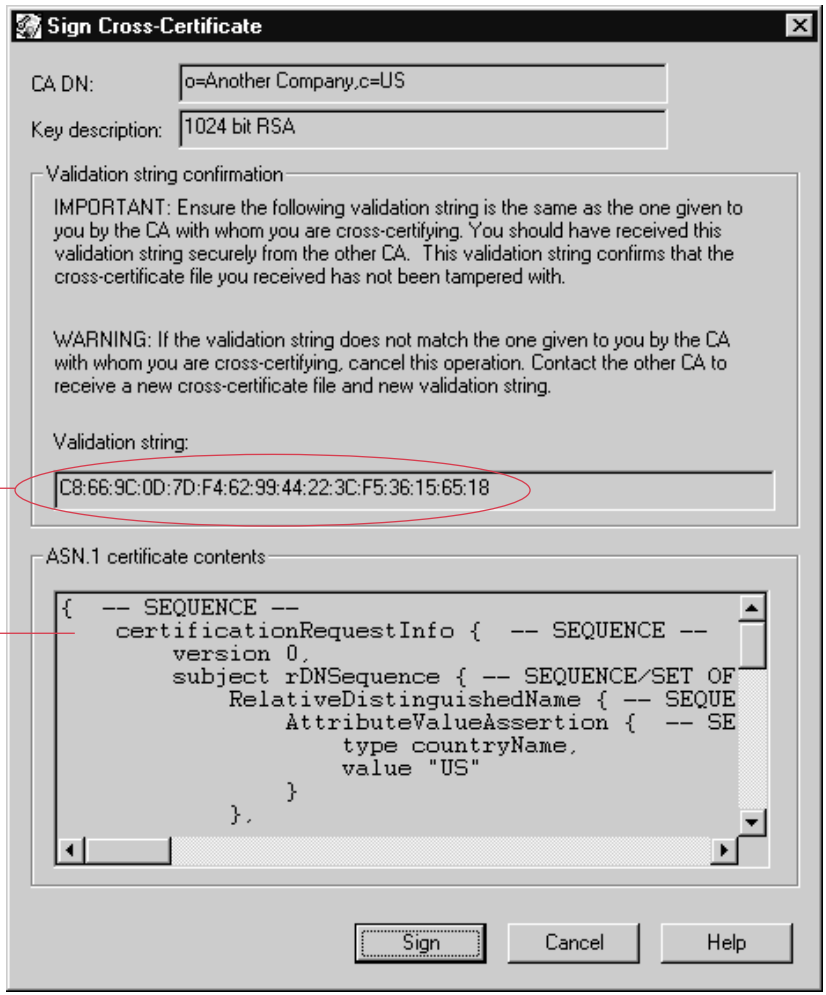
In the *What format would you like to save the response as?* group box, you can choose the file format to be *Binary* (an extension of .der is appended to the filename), or *Base 64 (PEM encoded)* (an extension of .pem is appended to the filename). The choice of formats is provided for interoperability with PKIs from other vendors. Entrust/PKI can read files in either format.

Click *OK*.

The Sign Cross-Certificate dialog box appears.

Record this value and give it to the subordinate CA administrator securely and separately from the response file. The subordinate CA administrator needs this string to validate the response file.

You can view details of the response certificate by scrolling through this window. You can copy and paste the contents of this window into a text editor.



4 Record the validation string and securely give it to the administrator of the subordinate CA. The administrator needs this string to verify that the response file they receive is from you and is unaltered. You can display this dialog box again by repeating Step 3.

Attention: Give your validation string to the administrator securely and separately from your response file. Sending the response and its validation string together greatly increases the risk of undetectable tampering with the response.

5 Click *Sign*.

6. Transmit the response file (for instance, by e-mail, ftp, or on disk) to the administrator of the subordinate CA.
7. At the subordinate CA, import the response file containing the subordinate CA certificate signed by the superior CA. Compare the validation string that is displayed when you import the response with the string you received from the superior CA administrator. The strings must match for you to accept the response.

   You have now reinstalled a third-party subordinate CA.

Updating a third-party subordinate CA's certificate

If a third-party subordinate CA updates its CA keys, you need to update the subordinate's CA certificate. (Note that in Entrust/PKI 6.0, a subordinate CA can't update its keys.)

The process of updating a third-party CA's certificate is the same as for reinstalling a third-party subordinate CA. For instructions, see "To reinstall an offline subordinate CA" on page 260.

Saving a CA certificate to file

You can save a subordinate CA's certificate to file. You do this when you want to view the certificate, or when a software application that needs the certificate needs it saved to file.

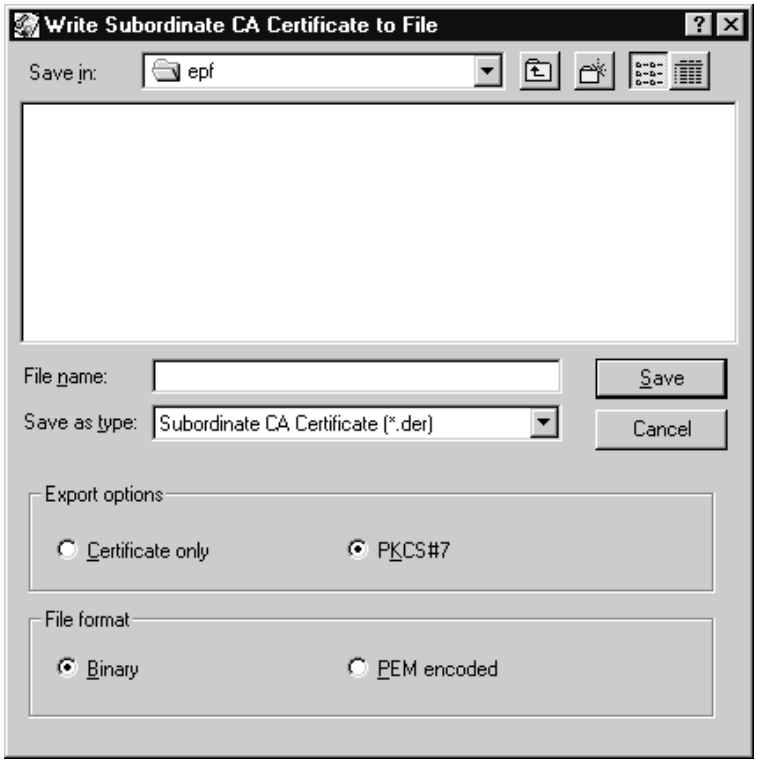
To save a CA certificate to file

1. Log in to Entrust/RA as a Security Officer at the superior CA.
2. Click *Certification Authority (CA)* in the tree view of Entrust/RA to expand it, and click *Subordinate CAs* to expand it.

   Click the subordinate CA whose certificate you want to save to file.

   The *Certificate List* property page shows all certificates issued to this subordinate.
3. On the *Certificate List* property page, right-click a certificate. In the pop-up menu, click *Write this Subordinate Certificate to file*.

   The Write Subordinate CA Certificate to File dialog box appears.



4. Choose a destination for the file.

   In the *File name* field, enter a name for the CA certificate file.

In the *Export options* group box, select an option. Choose *Certificate only* for a file that contains the subordinate's CA certificate only. Choose *PKCS #7* for a file that contains the subordinate's CA certificate, plus the certificates of all CAs in the chain up to and including the root.

In the *File format* group box, you can choose the file format to be *Binary* (an extension of .der is appended to the filename), or *PEM encoded* (an extension of .pem is appended to the filename). The choice of formats is provided for interoperability with PKIs from other vendors. Entrust/PKI can read files in either format.

5 Click *Save*.

You have now saved a subordinate CA's certificate to file.

Preparing the Directory

CAs in a hierarchy (and cross-certified CAs) need to access each other's Directory entries. These entries include certificate revocation lists (CRLs), authority revocation lists (ARLs), and encryption certificates. "Preparing the Directory" means ensuring that CAs in a hierarchy can access each other's Directory entries.

When you plan a hierarchy of CAs, you must decide how many Directories you will install. For instance, when you install Entrust/PKI, you have the option of installing a local Directory or using an existing Directory.

CAs that share a Directory can easily form a hierarchy or cross-certify with each other, because they can already access each other's Directory entries. CAs that don't share a Directory can't access each other's Directory entries (and, therefore, can't form a hierarchy or cross-certify with each other) unless you chain or use LDAP referrals to connect their Directories.

Chaining and referring Directories

If you choose to install one or more local Directories for CAs in your hierarchy, you must configure them so that the CAs can search across them all seamlessly for entries. One way to configure these Directories is by *chaining* them to each other. To understand chaining, suppose Directory A is chained to Directory B, and a client requests data from the Directory Service Agent (DSA) for Directory A. If DSA A can't find the requested data in Directory A, it passes the client's request to DSA B. If DSA B finds the data, it returns it to DSA A, which in turn sends it to the client. If Directory B is chained back to Directory A (called "two-way chaining"), then client requests to DSA B are similarly sent to DSA A.

Another way to configure Directories so that CAs can search across them all is to *refer* the Directories to each other. To understand referring, suppose Directory A is referred to Directory B, and a client requests data from the Directory Service Agent (DSA) for Directory A. If DSA A can't find the requested data in Directory A, it tells the client to request the data from DSA B. In other words, DSA A *refers* clients to DSA B if it can't fulfill requests locally. If Directory B is referred back to Directory A (called "two-way referring"), then DSA B can also refer client requests to DSA A. Referring is less efficient than chaining.

Note: Before you can install an Entrust root or subordinate CA, you must have a directory installed, running, and configured so CAs in a hierarchy or cross-certified CAs can access its Directory entries.

Chapter 8

Customizing certificates

Entrust/Authority certificates contain keys generated by Entrust/Authority and other information, such as the date that the certificate will expire. Entrust/PKI makes use of many different kinds of certificates, including Enterprise, Web, cross-certificates, policy, and CA certificates. If you are a Security Officer (or if you have the appropriate permissions as explained in "Creating roles" on page 142), you can customize certificates.

Attention: The default certificates meet the requirements of most organizations. The procedures described in this chapter for customizing certificates should be performed only by a highly trusted system administrator with advanced technical knowledge.

Typically you customize certificates to meet the certificate requirements of your organization while ensuring the authenticity of the certificate content. You can customize certificates by defining extra content and by excluding default content. Extra content may be of a fixed or variable nature. For example, you can use variable content to customize certificates on a per-user basis. You can define that a certificate issued to a user must include a value that represents the expenditures the user is allowed to make on behalf of the company (for example, under $500.00, under $2000.00, and so on).

You can also specify that you no longer want a particular type of certificate issued, and that you want content about users added to the Entrust/Authority database rather than to a certificate.

This chapter includes the following sections:

- "About certificates" on page 271
- "Working with the master.certspec file" on page 277
- "Customizing Enterprise and Web certificates" on page 282

- "Customizing policy certificates" on page 306
- "Customizing cross-certificates" on page 314
- "Customizing CA certificates" on page 325
- "Excluding certificate extensions" on page 328
- "Making a certificate type obsolete" on page 330
- "Customizing database fields" on page 332

About certificates

Entrust/Authority certificates contain keys generated by Entrust/Authority and other information, such as the date that the certificate will expire. Entrust/Authority certificates are issued by Entrust/Authority from several "certificate categories".

Entrust/Authority certificate categories

Entrust/Authority certificate categories determine the purpose of certificates issued from those categories.

The following table lists the Entrust/Authority certificate categories and the purpose of certificates issued from those categories.

Certificate category	Purpose of certificates
Enterprise	These certificates identify the owner of the certificate (that is, the user that Entrust/Authority issued the certificate to). The owner is an Enterprise user.
Web	These certificates identify the owner of the certificate (that is, the user that Entrust/Authority issued the certificate to). The owner is a Web user.
cross-certificate	These certificates establish direct trust between CAs, allowing the users who belong to each CA to indirectly trust each other.
policy	These certificates define what users who belong to a particular Entrust role can and cannot do with Entrust-Ready applications.
CA	These certificates store the verification public key of the CA's signing key pair.

For example, certificates issued from the Enterprise category are used to identify users and certificates issued from the cross-certificate category are used to establish trust between cross-certified CAs.

Certificates from different categories can be used for different purposes because of the content that Entrust/Authority inserts by default when issuing certificates. This default content is determined by the X.509 recommendation.

Adding X.509 content to certificates

The X.509 recommendation is a generally accepted definition of standard certificate content within the certification industry. Certificates that follow the X.509 recommendation are considered to be "X.509 compliant".

Entrust/Authority inserts content in accordance with the X.509 recommendation when issuing certificates. The X.509 recommendation specifies that the content included in a certificate depends on how the certificate will be used. This means that Entrust/Authority inserts different X.509 content for certificates issued from each certificate category.

The X.509 recommendation is maintained and published by the International Telecommunications Union (ITU). For information about this recommendation, contact the ITU and request *ITU-T Recommendation X.509*.

Note: This chapter assumes that you have at least a basic working knowledge of the X.509 recommendation. If you are not familiar with this recommendation, do not attempt to perform the procedures in this chapter.

Each certificate category contains one or more *types* of certificates. You use certificate types to *customize* certificates (for example, to define content for Entrust/Authority to insert in certificates, in addition to the X.509 content it inserts by default).

Dividing certificate categories into certificate types

Each certificate category contains one or more "certificate types". The following table lists the certificate categories and names their default certificate types.

This certificate category...	includes default type(s) named...
Enterprise	Default
Web	Default
cross-certificate	Default
policy	Client settings
CA	Default, Link

Certificates issued from a certificate category are based on a particular certificate type. A certificate type determines how Entrust/Authority customizes certificates issued for that type. Certificate types are defined using Entrust/Authority "certificate definitions".

Note: The certificate categories include at least one default certificate type. This means that you aren't required to define or modify any certificate types using certificate definitions.

Why customize certificates?

Typically you customize certificates to meet the certificate requirements of your organization while ensuring the authenticity of the certificate content. For example, you can customize certificates by adding information to them. Because the CA digitally signs certificates and you trust this signature, you can trust the integrity of the certificate contents (that is, only an administrative user with the appropriate permissions can change the contents). Had you chosen to add this information to the Directory directly, for example, you would not be assured of its authenticity. Because this information is not signed, and there is no requirement to locate the Directory on a physically secured server, anyone with access to the Directory can change the unsigned information in it.

Note: Certificate content is digitally signed; it is not encrypted. Do not add content that should be encrypted (that is, information such as employees' salaries that should be kept private) to certificates. If this information is related to users, you can add it to the Entrust/Authority database using database fields (see "Customizing database fields" on page 332).

Customizing content for certificates

Entrust/Authority certificate definitions specify how Entrust/Authority customizes certificates after inserting the other X.509 content.

You can work with certificate definitions to customize certificates for your organization. For example, suppose that you want the certificates that Entrust/Authority issues to Enterprise users to contain the department within your organization that the certificate owner belongs to. Using certificate definitions, you

- identify the certificate type that you want this content added to (in this example, the Default certificate type in the Enterprise certificate category)
- define the content (in this example, the department name) that Entrust/Authority is to insert in certificates of this type

You can customize all the certificates that Entrust/Authority issues. The customizations should reflect how the certificates will be used. For example, you can add identifying information about users to certificates issued from the Enterprise and Web certificate categories. However, this information is not useful

to add to certificates issued from the cross-certificate category because cross-certificates are not used to identify users.

Working with Entrust/Authority certificate definitions

The Entrust/Authority database stores Entrust/Authority certificate definitions. You can work with Entrust/Authority certificate definitions using the Entrust/Authority certificate definition file. This file is an ASCII (text) file.

This file is available before and after you initialize Entrust/PKI. Before initialization, the file is named "initial.certspec" and is stored in the following location, by default, on the Entrust/Authority server:

```
c:\Program Files\Entrust\Entrust Authority\etc\initial.certspec
```

Note: If you are running Entrust/PKI in an Identrus environment, see "Running Entrust/PKI in an Identrus environment" on page 589.

This is the only time that the file exists without being explicitly created by a Security Officer. This file is made available to you so that you can customize certificates before first-time initialization. For example, Entrust/Authority issues certificates for the CA and the First Officer during first-time initialization. Using the initial.certspec file, you can customize even the very first certificates issued by Entrust/Authority for your organization.

After initialization, by default the file is named "master.certspec" to distinguish it from the pre-initialized version. Note that this file does not exist until you export it from Entrust/RA. For more information, see "Creating the master.certspec file" on page 278.

Note: For the remainder of this chapter, references to "master.certspec" apply as well to "initial.certspec" unless noted otherwise.

Contents of the master.certspec file

This is an excerpt from the default master.certspec file.

```
[Certificate Categories]
;*********************************************************************
;* This section lists the categories to which certificate types may be *
;* assigned. Note that the order in which they are listed is not       *
;* important.                                                          *
;*                                                                     *
;* No changes to this section are required.                            *
;*********************************************************************
1=enterprise
2=web
3=set
4=xcert
5=policycert
6=cacert
.
.
.
```

The default master.certspec file is compliant with the X.509 recommendation. This file also meets the minimum requirements of the Internet Engineering Task Force (IETF) Request for Comments (RFC) 2459: *Internet X.509 Public Key Infrastructure Certificate and CRL Profile*.

The recommendations in RFC 2459 are based on the X.509 recommendation, but are more stringent. The default master.certspec file meets the minimum requirements of RFC 2459. It does not enforce the RFC 2459 suggestions for stricter requirements. However, you can edit the file to meet those suggestions if necessary.

You can work with the initial.certspec and master.certspec files if you are a Security Officer or if you have the appropriate permissions as explained in "Creating roles" on page 142. The steps to access and process Entrust/Authority certificate definitions are different for the initial.certspec and master.certspec files. The following steps provide a summary of the procedures for each file. These procedures are described in more detail in the remainder of this chapter.

To work with the initial.certspec file

1. Open the initial.certspec file for editing (see "Opening the master.certspec file" on page 279).
2. Edit the initial.certspec file as described in the following sections:
   - "Customizing Enterprise and Web certificates" on page 282
   - "Customizing policy certificates" on page 306

- "Customizing cross-certificates" on page 314
- "Customizing CA certificates" on page 325
- "Making a certificate type obsolete" on page 330
- "Customizing database fields" on page 332

3 Initialize Entrust/PKI (see *Installing Entrust/PKI 6.0 on Windows*).

During initialization, Entrust/Authority writes the Entrust/Authority certificate definitions in the initial.certspec file to the Entrust/Authority database and creates a file named "initial.log". If the definitions were not successfully written to the database, initial.log contains a list of errors that you must resolve before initial.log can be processed successfully.

To work with the master.certspec file

1 Log in to Entrust/RA as a Security Officer (see the "Setting up the Certification Authority (CA)" chapter in *Using Entrust/PKI 6.0 on Windows*).

2 Create the master.certspec file as described in "Creating the master.certspec file" on page 278.

Entrust/RA exports (that is, "retrieves" or "gets") Entrust/Authority certificate definitions from the Entrust/Authority database.

3 Open the master.certspec file for editing (see "Opening the master.certspec file" on page 279).

4 Edit the master.certspec file as described in the following sections:

- "Customizing Enterprise and Web certificates" on page 282
- "Customizing policy certificates" on page 306
- "Customizing cross-certificates" on page 314
- "Customizing CA certificates" on page 325
- "Making a certificate type obsolete" on page 330
- "Customizing database fields" on page 332

5 Process the master.certspec file (see "Processing changes to the master.certspec file" on page 280).

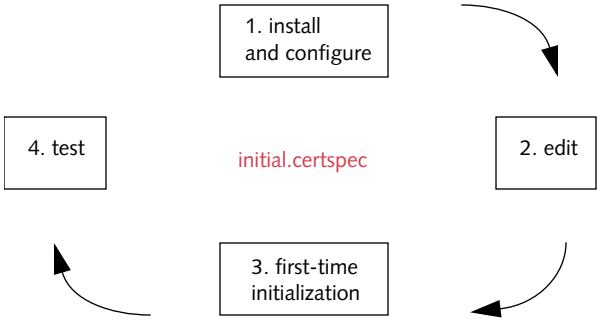
Entrust/Authority writes the Entrust/Authority certificate definitions in the master.certspec file to the Entrust/Authority database. Entrust/RA then deletes the master.certspec file.

Once the contents of the master.certspec file have been written into the Entrust/Authority database, Entrust/Authority begins using the revised Entrust/Authority certificate definitions immediately.

Working with the master.certspec file

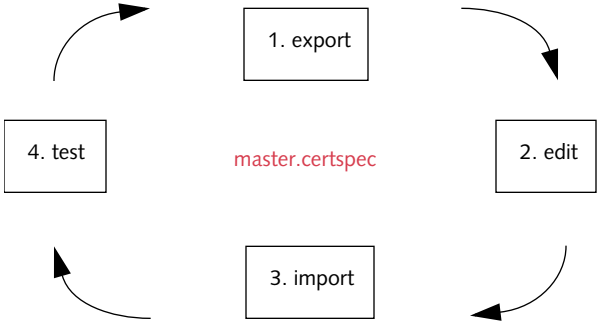
Before you can edit Entrust/Authority certificate definitions, you must create and open the master.certspec file. You can then edit this file to customize certificates issued by Entrust/Authority. After you have edited the master.certspec file, you process the file to write its contents to the Entrust/Authority database.

The following diagram summarizes the steps you can take to customize the certificate definitions in the initial.certspec file:



After you have completed first-time initialization of Entrust/PKI, you will need to export the certificate definition file before you can make any changes to it.

The following diagram summarizes the steps you can take to customize the certificate definitions in the master.certspec file:



The following table indicates where you can find information about editing the master.certspec file to customize certificates issued by Entrust/Authority.

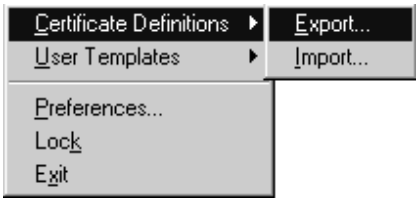
For information about...	See...
Enterprise or Web certificates	"Customizing Enterprise and Web certificates" on page 282
policy certificates	"Customizing policy certificates" on page 306
cross-certificates	"Customizing cross-certificates" on page 314
CA certificates	"Customizing CA certificates" on page 325

Creating the master.certspec file

Entrust/RA creates the master.certspec file automatically when you export Entrust/Authority certificate definitions from the Entrust/Authority database. You export these certificate definitions using Entrust/RA.

To create the master.certspec file

1. In Entrust/RA, click File > Certificate Definitions > Export.



   The Save As dialog box appears.

2. Type a filename and location for the file and click *Save*.

   By default, the file is named "master.certspec".

   When the master.certspec file has been successfully created, a dialog box appears.

3. Click *OK*.

   You have now created the master.certspec file.

Attention: As a Security Officer, you are responsible for the master.certspec file. Entrust Technologies does not recommend that you keep copies of this file. Keeping additional copies increases the risk of overwriting a newer version of the file with an older version.

Opening the master.certspec file

Once you have created the master.certspec file, you can open it in any text editor (for example, WordPad or Notepad). The following excerpt from the default master.certspec file shows the beginning of this file.

```
 [Certificate Categories]
;*********************************************************************
;* This section lists the categories to which certificate types may be *
;* assigned. Note that the order in which they are listed is not       *
;* important.                                                          *
;*                                                                     *
;* No changes to this section are required.                            *
;*********************************************************************
1=enterprise
2=web
3=set
4=xcert
5=policycert
6=cacert
.
.
.
```

Navigating the master.certspec file

The master.certspec file groups similar information into sections. You may modify the contents of several sections each time you edit the file.

To help you navigate the file, the beginning of each section is identified by its name enclosed in brackets. For example, the names of all the certificate categories are listed in the [Certificate Categories] section. When you need to work in a particular section, you can search for its section header using the search facility in the text editor.

Attention: Do *not* change the certificate categories listed in the [Certificate Categories] section. This section is reserved for future use, and *any* changes to it may cause errors when you write the certificate definitions to the Entrust/Authority database.

The following excerpt from the default master.certspec file shows the major sections it contains.

```
[Certificate Categories]
.
.
.
[Certificate Types]
.
.
.
[Extension Definitions]
.
.
.
[Advanced Settings]
.
.
.
[Database Field Definitions]
.
.
.
[Attribute Definitions]
.
.
.
[Variables]
.
.
.
[Obsolete Certificate Types]
.
.
.
```

Processing changes to the master.certspec file

Once you have finished making your changes to the master.certspec file, you process the file. When you process the file, Entrust/RA checks the file for errors. If Entrust/RA finds errors, descriptions of the errors are written to a log file that you specify (named "master.log" by default). The log file is never deleted by Entrust/RA.

If Entrust/RA doesn't find errors, it splits lines longer than 72 characters into two or more lines. Additional lines are preceded by "_continue_=". For example:

```
web_default=web,Default,Default Web Certificates (for browsers and secure
_continue_=e-mail)
```

Long lines are split by "_continue_=".

Once Entrust/RA has finished processing the file, Entrust/Authority checks the file for errors. If Entrust/Authority finds errors, descriptions of the errors are displayed in a dialog box. If Entrust/Authority doesn't find errors, it writes the contents of the file to the Entrust/Authority database and signals Entrust/RA to delete the master.certspec file. The master.log file contains a summary of the certificate types defined.

To process the master.certspec file

1. In Entrust/RA, click File > Certificate Definitions > Import.

   The Open dialog box appears.

2. Select the master.certspec file in the Open dialog box and click *Open*.

   The Save As dialog box appears.

3. Specify a name and location (or accept the default) for a log file for any errors found by Entrust/RA during the import, and click *Save*.

   A dialog box appears when the certificate definitions are imported.

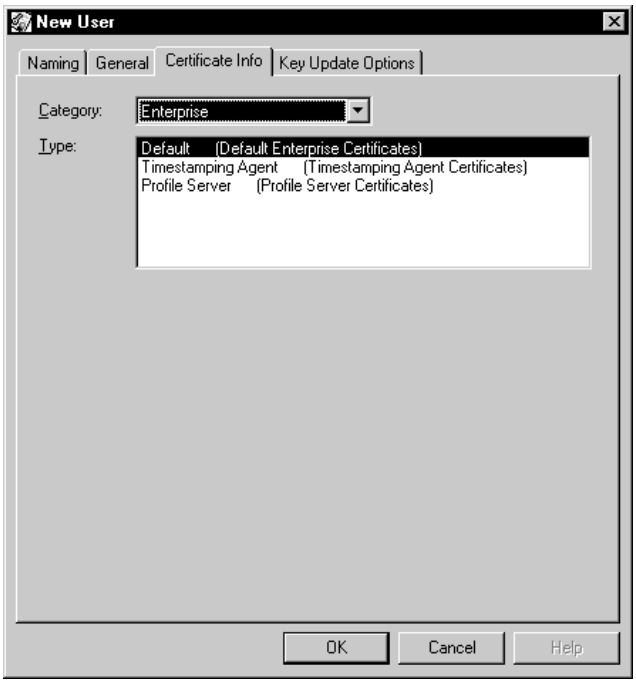
4. Click *OK*.

   You have now processed the master.certspec file.

Customizing Enterprise and Web certificates

Certificates issued from the Enterprise and Web certificate categories provide useful information that identifies the certificate owner (that is, the person the certificates are issued to). Certificates issued from these three certificate categories are referred to collectively as "user certificates". Entrust/Authority issues user certificates when a user logs in for the first time, when a user's keys need to be updated, and when a user recovers their keys.

You select the type of certificate that Entrust/Authority issues to a particular user in the Certificate property page of the New User dialog box in Entrust/RA. This example shows that the Default certificate type from the Enterprise category is assigned to this particular user. You can change the assigned certificate type by clicking another certificate type. For more information about the New User dialog box, see the "Administering individual users" chapter in *Using Entrust/PKI 6.0 on Windows*.



Within a certificate category, the difference among the types of certificates is the extra content that Entrust/Authority adds to the certificate, in addition to the other X.509 content. For example, the previous example shows three certificate types for the Enterprise certificate category. All three certificate types contain the

same X.509 content. However, the "Timestamp" certificate type is customized to include extra content related to the Entrust/Timestamp Server while the "ProfileServer" certificate type is customized to include extra content related to the Entrust/Profile Server.

Creating and modifying a user certificate type

If the certificate types listed in the New User dialog box don't meet your organization's requirements, you can create and modify certificate types. For example, you can create a certificate type in the Enterprise category that requires Entrust/Authority to insert the job title of the certificate owner when issuing certificates for this certificate type.

Note: If the default certificate types meet your organization's requirements, you don't need to modify the default certificate types or create new ones.

You create and modify certificate types in the Enterprise and Web certificate categories using the master.certspec file. This section assumes that you know how to create, open, and process the master.certspec file. If you don't know how to perform these tasks, see "Working with the master.certspec file" on page 277. In the master.certspec file, you can

- create new Enterprise and Web certificate types. After you create a certificate type and process the master.certspec file, the New User dialog box lists the new certificate type the next time you open it.
- modify the default Enterprise and Web certificate types. After you modify a certificate type and process the master.certspec file, Entrust/Authority begins issuing certificates according to your changes.

You create or modify a certificate type using a certificate type definition and one or more certificate extensions.

What is a certificate type definition?

The master.certspec file lists a certificate type definition for each certificate type in the Enterprise and Web certificate categories. A certificate type definition

- uniquely identifies the certificate type in the master.certspec file
- specifies the category that the certificate type belongs to
- provides the name and description for this certificate type when it is displayed in the New User dialog box

What is a certificate extension?

The master.certspec file may contain certificate extensions for each certificate type in the Enterprise and Web certificate categories. A certificate extension defines one value that Entrust/Authority inserts in certificates issued for this certificate type. The value provided by an extension can be of a fixed or variable nature.

Certificate extensions defined in the master.certspec file define the extra content for certificates. These certificate extensions don't define the X.509 content that Entrust/Authority inserts by default. However, you can use the master.certspec file to exclude these default certificate extensions.

Note: A certificate type may contain database field definitions in addition to certificate extensions. Database fields are used to add content to the Entrust/Authority database instead of certificates (see "Customizing database fields" on page 332). If a certificate type was created solely for the purpose of using database fields, the master.certspec file does not need to contain certificate extensions for that certificate type.

Using certificate type definitions and certificate extensions

Understanding how certificate type definitions and certificate extensions work together is best explained by example. Suppose that you want Enterprise users who work in branch offices to receive certificates that include their location (for example, "Boston", "Chicago", and so on).

To do this, you create two new certificate types in the Enterprise certificate category: one certificate type for users in the Boston office and one certificate type for users in the Chicago office. After creating the certificate types, you assign the appropriate certificate type in the New User dialog box to users in each branch office.

To create the certificate type for this example, you edit the master.certspec file to add the

- certificate type definition (see "Editing a certificate type definition" on page 285)
- certificate extension that specifies the name of the branch office (see "Editing certificate extensions" on page 287)

Editing a certificate type definition

Each certificate type for the Enterprise and Web certificate categories is represented by a certificate type definition (see "What is a certificate type definition?" on page 283) in the master.certspec file.

In the master.certspec file, certificate type definitions for all certificate types are located in the [Certificate Types] section. This example highlights the definitions for the default certificate types in the Enterprise and Web certificate categories:

```
[Certificate Categories]
.
.
.
[Certificate Types]



ent_default=enterprise,Default,Default Enterprise Certificates
web_default=web,Default,Default Web Certificates (for browsers and secure
e-mail)
xcert_default=xcert,Default,Default Cross-Certificates
xcert_pkix=xcert,PKIX,PKIX compliant Cross-Certificates
polcert_cliset=policycert,Client Settings,Client Setting Certificates
cacert_default=cacert,Default,Default Self Signed CA Certificates
cacert_link=cacert,Link,Self Issued Link CA Certificates
.
.
.
```

To add a certificate type definition

1. In the master.certspec file, locate the [Certificate Types] section header.
2. On a new line in the [Certificate Types] section, type the definition for the certificate type.

   You can add a new line anywhere in the [Certificate Types] section. Typically, however, you type definitions for new certificate types after the default ones.

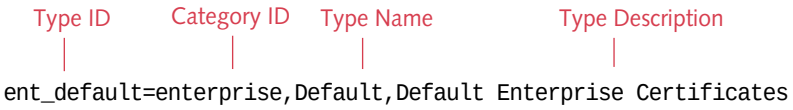
The following command specifies a definition for a certificate type. (Table 8 describes the parameters used in this command).

```
Type ID=Category ID,Type Name,Type Description
```

Table 8: Parameters in definitions for user certificate types

Parameter	Description
Type ID	Provides a unique representation for the certificate type in the master.certspec file (for example, "ent_default"). The characters may be either uppercase or lowercase. Entrust/Authority converts all characters to lowercase when processing the master.certspec file. There is no maximum length for the Type ID, although the first 20 characters must be unique. If the Type ID exceeds 20 characters, it is truncated to 20 characters when Entrust/Authority processes the master.certspec file. It's a good idea to identify the certificate category in the Type ID. For example, you can prefix the Type ID for Enterprise certificates with "ent_".
Category ID	Identifies the user certificate category that this certificate type belongs to (that is, "Enterprise" and "Web").
Type Name	Defines the name for this certificate type that appears in the list of certificate types in the New User dialog box. Note that the name cannot include commas (",") or semicolons (";").
Type Description	Defines the description for this certificate type that appears in the list of certificate types in the New User dialog box. Note that the description cannot include commas (",") or semicolons (";").

For example, here are the values for the parameters in the definition for the default Enterprise certificate type:



Note: You can modify the Type Name and Type Description parameters for the default certificate types. Be particularly careful making changes to the definition for the default Enterprise certificate type. Entrust/Authority uses this certificate type for first-time initialization. Changes to this certificate type affect system dependencies throughout Entrust/PKI.

3 Add a comment that describes the purpose of the certificate type when you add a definition.

You are not required to add a comment, but you may find them useful. Comments help you remember the purpose the next time you edit the file. To include a comment, type a semicolon (";"), followed by your comment. When Entrust/Authority processes the file, it ignores everything between the semicolon and the end of the line.

You have now added a certificate type definition.

Editing certificate extensions

Each certificate type for the Enterprise and Web certificate categories may contain certificate extensions (see "What is a certificate extension?" on page 284) in the master.certspec file.

For each certificate extension, you can specify whether you want its value included in encryption certificates, verification certificates, or both encryption and verification certificates for a certificate type. For example, suppose that you want Enterprise certificates to indicate whether the certificate owner is authorized to make purchases over $500.00 on behalf of the company. When a user submits a purchase request, the person who receives the request can check the user's certificate before approving the purchase. This type of content is useful in verification certificates, which are used to verify a user's identity. There is no need for encryption certificates to include this content because they aren't used to verify a user's identity.

In the master.certspec file, certificate extensions for all certificate types are located in the [Extension Definitions] section. The following example shows a certificate extension that defines extra content for verification certificates issued

for the certificate type identified as "ent_timestamp" in the [Certificate Types] section.

```
.
.
.
[Extension Definitions]
.
.
.
[ent_timestamp Verification Extensions]
;-----------------------------------------------------------------------
;- Timestamp Certificate Type                                          -
;-                                                                     -
;- This is a special certificate type that is to be used ONLY for      -
;- creating certificates used to secure Timestamping Agents.           -
;- Timestamping Agents' verification certificates include the extended -
;- key usage extension which contains the id-kp-timeStamping usage     -
;- identifier.                                                         -
;-----------------------------------------------------------------------
extKeyUsage=2.5.29.37,n,m,SeqOfObjectIdentifier,1.3.6.1.5.5.7.3.8
.
.
.
```

Associating a certificate extension with a certificate type

Within the [Extension Definitions] section, certificate extensions are organized according to their certificate type, and whether the value provided by the extension is to be included in encryption, verification, or both encryption and verification certificates for that certificate type. The organization is identified using subsection headers. For example, certificate extensions for

- verification certificates issued for the ent_default certificate type will be defined in the [ent_default Verification Extensions] section
- encryption certificates issued for the ent_default certificate type will be defined in the [ent_default Encryption Extensions] section
- verification and encryption certificates issued for the ent_default certificate type will be defined in the [ent_default Common Extensions] section

---

Note: Certificate extensions defined as "Common" extensions cannot also be defined as "Encryption" or "Verification" extensions.

To define a subsection header for a certificate extension

1. In the master.certspec file, locate the [Extension Definitions] section header.
2. On a new line in this section, type the subsection header.

   Subsection headers for certificate extensions use the following formats:

   ```
   [Type ID Verification Extensions]
   [Type ID Encryption Extensions]
   [Type ID Common Extensions]
   ```

   where "Type ID" represents the Type ID as defined in the [Certificate Types] section for the certificate type (for example, "ent_default").

   Subsection headers must appear in square brackets and are case-sensitive.

Note: You only define subsection headers for the kinds of certificate extensions you will define. For example, if you aren't going to define certificate extensions for encryption certificates for the ent_corporate certificate type, you don't need to create the [ent_corporate Encryption Extensions] subsection header.

Once you have created a subsection within the [Extension Definitions] section, you can define its certificate extensions.

Defining a certificate extension in a subsection

You define a certificate extension within a subsection of the [Extension Definitions] section of the master.certspec file by specifying information about the value provided by the extension, and the value itself.

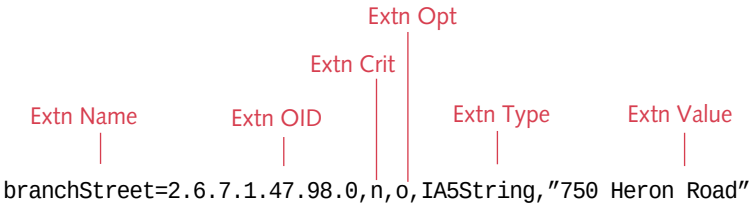
The value provided by a certificate extension can be of a fixed or variable nature. For example, one certificate extension may provide your organization's phone number. The phone number is a fixed value (that is, the value is the same in every certificate). A different certificate extension may provide the name of the department that the certificate owner belongs to. The department name is a variable value (that is, it is provided by the administrative user on a per-user basis when adding a user).

To define a certificate extension, type it in the appropriate subsection of the [Extension Definitions] section. Certificate extensions use the following syntax, which is described in the following topics:

```
Extn Name=Extn OID,Extn Crit,Extn Opt,Extn Type,Extn Value
```

Note: Entrust/Authority uses the information you provide to encode the extension as an X.509 certificate extension. You are not coding an X.509 extension yourself.

For example, the following defines a certificate extension named "branchStreet":



Note: The value "2.6.7.1.47.98.0" is a sample OID that is used for illustration purposes only.

Extn Name

The value of this parameter provides a unique representation for the certificate extension in the master.certspec file. For example, the Extn Name for the following certificate extension is "keyusage":

```
keyusage=2.5.29.15,n,m,BitString,101
```

Extn OID

The value of this parameter represents the object identifier (OID) of the certificate extension, in dotted decimal form. For example, the value "2.5.29.15" in the following certificate extension specifies the Extn OID:

```
keyusage=2.5.29.15,n,m,BitString,101
```

All extensions that you define must be registered and assigned object identifiers. If the extension is not registered, you must first obtain an OID for that extension by registering it. You obtain OIDs for extensions from your registration authority (for example, GTIS or ANSI).

If you use an OID in more than one certificate extension, you should use the same Extn Name for each of the certificate extensions.

Entrust/Authority recognizes some Extn OIDs, some of which you can define in the master.certspec file and some of which you cannot. Table 9 lists the Extn

OIDs and their corresponding Extn Names that Entrust/Authority recognizes, but that you cannot define in the master.certspec file.

Table 9: Recognized Extn OIDs that you can't use

OID	Extn Name
2.5.29.35	authorityKeyIdentifier
1.2.840.113533.7.65.0	entrustVersInfo
2.5.29.14	subjectKeyIdentifier
2.5.29.16	privateKeyUsagePeriod

Table 10 lists the Extn OIDs and Names that Entrust/Authority recognizes, and that you can define in the master.certspec file. You define them in the master.certspec file to change their default criticality parameter (see "Extn Crit" on page 292) or value parameter (see "Extn Value" on page 295).

Table 10: Recognized Extn OIDs that you can use

OID	Extn Name	Define Value?	Define Criticality?	Comments
2.5.29.17	subjectAltName	no	yes	Entrust/Authority looks at the criticality only. The type and value are ignored.
2.5.29.31	crlDistributionPoints	no	yes	Entrust/Authority looks at the criticality only. The type and value are ignored.
2.5.29.19	basicConstraints	yes	yes	You cannot set pathLenConstraint for user certificates. You must set the CA bit for the cross-certificate category.

Table 10: Recognized Extn OIDs that you can use (continued)

OID	Extn Name	Define Value?	Define Criticality?	Comments
2.5.29.32	certificatePolicies	yes	yes	If a definition of this extension is included in the certificate definition file, its value will override the Encryption OIDs or Verification OIDs set under *Security Policy* in Entrust/RA, or both.
2.5.29.15	keyUsage	yes	yes	The extension must be consistent with the section in which it is defined. You cannot use this extension in a [*Type ID* Common Extensions] section.

Extn Crit

The value of this parameter determines whether the value provided by the certificate extension is critical. The value of this parameter is "c" (critical) or "n" (non-critical). For example, the value "n" in the following certificate extension indicates that the value provided by the extension is non-critical:

```
keyusage=2.5.29.15,n,m,BitString,101
```

According to Recommendation X.509, when an application receives a certificate that includes an extension that the application does not know how to interpret, the application checks whether the extension is critical. The application must reject the certificate if the extension is marked as critical.

Extn Opt

The value provided by this parameter determines whether the value provided by the certificate extension is mandatory. The value of this parameter is "o" (optional) or "m" (mandatory).

This parameter is important for certificate extensions that provide variable values (that is, values provided by the administrative user when enabling users). When Entrust/Authority tries to issue a certificate that includes a variable certificate extension and the value for that certificate extension is missing (for example, the administrative user did not provide a value when enabling the user), Entrust/Authority checks whether the extension is mandatory. Entrust/Authority issues the certificate only if the certificate extension is optional. Entrust/Authority does not issue a certificate if the certificate extension is mandatory.

Be careful about how you set the optionality for new certificate extensions with variables. If the extensions belong to types of certificates that are already being issued to users, Entrust/Authority will try to include the extension the next time that the certificates for these users are issued (for example, when the certificates are updated or when the users are set up for key recovery). If you set the extension as "mandatory", Entrust/Authority cannot issue the certificate. The value of the extension has not been provided by the administrator for the existing users. You must re-enable each user and provide values for the new mandatory extension.

Note: Bulk commands provide a convenient method of performing operations for multiple users. For information about bulk commands, see the "Administering multiple users" chapter in *Using Entrust/PKI 6.0 on Windows*.

Extn Type

The value provided by this parameter determines the data type of the value provided by the certificate extension. For example, the data format for the following certificate extension is "BitString":

```
keyusage=2.5.29.15,n,m,BitString,101
```

Table 11 lists and describes the values that you can specify for the Extn Type parameter.

Table 11: Description of data formats

Use this data format... (single value)	(ASN.1 sequence)	when the value provided by the extension is...
BitString	SeqOfBitString	a string of bits listed in binary notation, with the least significant bit first (for example, "1", "10100", "0110")
BMPString	SeqOfBMPString	a string of characters that are allowed for the corresponding ASN.1 type
Boolean	SeqOfBoolean	a value of "1" for TRUE or "0" for FALSE
DER	DER	in its ASN.1 DER-encoded representation (see "Extn Value" on page 295)
GeneralizedTime	SeqOfGeneralizedTime	a string containing a "coordinated universal time" or Greenwich Mean Time (GMT) value as specified for the corresponding ASN.1 type (for example, "9709051200Z")
IA5String	SeqOfIA5String	(Same as BMPString.)

Table 11: Description of data formats (continued)

Use this data format... (single value)	(ASN.1 sequence)	when the value provided by the extension is...
Integer	SeqOfInteger	an integer with a zero, positive or negative value of any magnitude (for example, "0", "58", "-5")
NumericString	SeqOfNumericString	(Same as BMPString.)
ObjectIdentifier	SeqOfObjectIdentifier	an object identifier, in dotted decimal form (for example, "2.5.29.15")
OctetString	SeqOfOctetString	a string of byte values listed in hexadecimal notation with two characters per octet and leading zeroes preserved (for example, "1A2B", "0C", "090B0D0F11"). No spaces are allowed between octets.
PrintableString	SeqOfPrintableString	(Same as BMPString.)
Real	SeqOfReal	a real number with a zero, positive or negative value of any magnitude (for example, "0", "1.813", "-163.875", "1.345e-4")
UTCTime	SeqOfUTCTime	a string containing a "coordinated universal time" or Greenwich Mean Time (GMT) value as specified for the corresponding ASN.1 type (for example, "9709051200Z")
UTF8String	SeqOfUTF8String	(Same as BMPString.)

If the certificate extension provides a value of a variable nature, you will define the variable (see "Editing a variable" on page 297). The data format that you specify for the Extn Type parameter determines the data format that you can specify for the variable when you define it. Table 12 lists the data formats that you can specify for the "Extn Type" and the corresponding variable types.

Table 12: Data formats for extension types and their variable types

Data type for certificate extension	Data type for variable
BitString	BitString, Boolean
BMPString	TextString
Boolean	Boolean
DER	OctetString
GeneralizedTime	DateTime
IA5String	TextString

Table 12: Data formats for extension types and their variable types (continued)

Data type for certificate extension	Data type for variable
Integer	Integer
NumericString	TextString
ObjectIdentifier	ObjectIdentifier
OctetString	OctetString
PrintableString	TextString
Real	Real
SeqOfBitString	BitStringList, BooleanList
SeqOfBMPString	TextStringList
SeqOfBoolean	BooleanList
SeqOfGeneralizedTime	DateTimeList
SeqOfIA5String	TextStringList
SeqOfInteger	IntegerList
SeqOfNumericString	TextStringList
SeqOfObjectIdentifier	ObjectIdentifierList
SeqOfOctetString	OctetStringList
SeqOfPrintableString	TextStringList
SeqOfReal	RealList
SeqOfUTCTime	DateTimeList
SeqOfUTF8String	TextStringList
UTCTime	DateTime
UTF8String	TextString

Extn Value

This parameter identifies the value that Entrust/Authority is to include in certificates. For example, the value for the following certificate extension is "101":

```
keyusage=2.5.29.15,n,m,BitString,101
```

Extensions or attributes with sequence types can also specify multiple constant values. Each value should be space-separated. For example, the extendedKeyUsage extension could be specified as:

```
extKeyUsage=2.5.29.37,n,m,SeqOfObjectIdentifier,1.2.840.113533.7.74.1
1.2.840.113533.7.74.2
```

Entrust/Authority adds the value to the specified certificate extensions just before issuing the certificate. Entrust/Authority adds the value in its ASN.1 DER-encoded representation. The ASN.1 recommendation for data structures is defined in *Recommendation X.680: Abstract Syntax Notation One (ASN.1)* and is maintained by the ITU-T. To represent ASN.1 structures in a machine-readable form, Entrust/Authority encodes them using the Distinguished Encoding Rules (DER). These rules are just one of the methods defined in *Recommendation X.690: ASN.1 Encoding Rules* for encoding ASN.1 data structures. The X.690 recommendation is also maintained by the ITU-T.

For all of the formats listed in Table 11 except the DER format, Entrust/Authority can convert the value provided by the Extn Value parameter to its ASN.1 DER-encoded format before inserting the value in a certificate.

For a fixed value that Entrust/Authority can convert to its ASN.1 DER-encoded format, you simply type the value. For example, for the following certificate extension, Entrust/Authority inserts the ASN.1 DER-encoded representation of the fixed value "101" in certificates:

```
keyusage=2.5.29.15,n,m,BitString,101
```

If the fixed value is one of the text string types, enclose it in quotation marks and do not include semicolons (";"). For example:

```
branchStreet=2.5.29.15,n,o,IA5String,"750 Heron Road Ottawa"
```

Within the quotation marks, you can use the following characters, as long as you identify them within the quotation marks. You identify these characters using an "escape sequence" (that is, the backslash ("\") character) to identify them as part of the value:

- \ ("\\")
- " ("\"")
- < ("\<")
- > ("\>")
- line break ("\n")
- tab ("\t")

For example, this certificate extension adds the value "Apples\Oranges" to the certificate:

```
code=2.5.29.15,n,m,IA5String,"\"Apples\\Oranges\""
```

Note: Text string data types include: NumericString, PrintableString, IA5String, BMPString, UTF8String, SeqOfNumericString, SeqOfPrintableString, SeqOfIA5String, SeqOfBMPString, and SeqOfUTF8String.

For a value that Entrust/Authority cannot convert to its ASN.1 DER-encoded format (that is, a value that cannot be represented by one of the data formats listed in Table 11 other than "DER"), you must provide the ASN.1 DER-encoded representation of the value. The representation must contain two characters per octet, without spaces between octets, and the representation must preserve leading zeroes.

Note: If you want to include a complex value in a certificate extension and you do not know how to convert the value to its ASN.1 DER-encoded representation, contact the ITU-T for *Recommendation X.690: ASN.1 Encoding Rules*. Converting values to their ASN.1 DER-encoded representations is a complex topic that goes beyond the scope of this documentation.

For any variable value, type the variable name enclosed in angle brackets ("<>"). For example, the following certificate extension uses a variable named "branchstreet":

```
branchStreet=2.5.29.15,n,o,IA5String,"<branchstreet>"
```

If you use a variable, you must define it in the [Variables] section of the master.certspec file (see "Editing a variable" on page 297).

Editing a variable

Certificate extensions define a value (of a fixed or variable nature) that Entrust/Authority is to insert in certificates. A certificate extension that provides a variable value specifies a variable name for its Extn Value parameter (see "Extn Type" on page 293). The variable must be defined in the [Variables] section of the master.certspec file.

To define a variable

1. Locate the [Variables] section in the master.certspec file.

2 On a new line in this section, define the variable using the following syntax. (Table 13 describes the parameters used in variable definitions.)

```
Var ID=Var Type,Var Label,Var Desc,[Range | OneOf],Value Spec
1,<Value Spec 2>,...
```

Table 13: Parameters in variable definitions

Parameter	Description
Var ID	Names the variable. The name must be unique and must match the name used in the Extn Value parameter of the certificate extension as defined in the [Extension Definitions] section. The characters may be either uppercase or lowercase. Entrust/Authority converts all characters to lowercase when processing the master.certspec file. There is no maximum length for the Var ID, although the first 20 characters must be unique. If the Var ID exceeds 20 characters, it is truncated to 20 characters when Entrust/Authority processes the master.certspec file.
Var Type	Identifies the data type of the variable. If the value provided for the variable is a single value, the following data types are valid: • Boolean • Integer • BitString • OctetString • Real • ObjectIdentifier • TextString • DateTime

Table 13: Parameters in variable definitions (continued)

Parameter	Description
Var Type (continued)	To specify a variable with multiple values, use one of the following types: • BooleanList • IntegerList • BitStringList • OctetStringList • RealList • ObjectIdentifierList • TextStringList • DateTimeList Note that the value you specify for the Var Type parameter must correspond to the value specified for the Extn Type parameter of the certificate extension (see "Extn Type" on page 293).
Var Label	Defines the prompt that appears in Entrust/RA dialog boxes (for example, "User's phone number:").
Var Desc	Defines the help description that appears in Entrust/RA dialog boxes (for example, the values that are valid for this variable).
Range or OneOf	Indicates whether the value provided by this parameter must lie between a minimum and maximum range or whether the value must be one of a list of valid values. The parameter you choose determines the number of Value Spec n parameters you must specify.
Value Spec n	For a range of values, you must provide the minimum and maximum values using two Value Spec n parameters. Use Value Spec 1 for the minimum number or length and Value Spec 2 for the maximum number or length. For the OneOf parameter, you specify each value that can be selected using one Value Spec n parameter. The Value Spec n parameter cannot include commas, semicolons, or spaces.

For example, the following variable definition specifies that the administrative user must choose a value between 0 and 52 weeks:

```
password_lifetime=Integer,Password expires in (weeks):,Number
of weeks that passwords remain valid. (0 = never
expire),Range,0,52
```

The following variable definition specifies that the administrative user must choose a combination of the four values specified (that is, all the algorithms, SHA-1, MD5, or MD2):

```
allowed_hashing_algs=TextStringList,Hashing
algorithms:,Algorithm(s) which may be used to hash a user's
data.,OneOf,"ALL","SHA1","MD5","MD2"
```

Setting up default values

In Entrust/PKI 5.1 or newer, you can set up a default value for a variable specified in an extension or attribute in the master.certspec or initial.certspec file. The default value is used to encode user or attribute certificates when no variable is available. This means that a new extension or attribute can be added to a certificate type without a new value to be added to all of the users (or policies) using that certificate type.

You may want to set up default values for the variable definitions if you plan to use any of the following user policies:

- permit_server_login
- enforce_identity_use
- http_proxy_setting

The default value can be specified in the [Variables] section of the master.certspec or initial.certspec file as follows:

```
<Var ID>=<Var Type>,<Var Label>,<Var Description>,[<Default
Value>,]<Range | OneOf>,<Value 1,[<Value 2>,...]
```

The default value can also be specified in the [Default Variable Values] section of the entmgr.ini file as follows:

```
<Var ID>=<Default Value>
```

If the [Default Variable Values] section does not appear in the entmgr.ini file, you must add it the first time you add a default value, as follows:

```
[Default Variable Values]
<Var ID>=<Default Value>
```

The [Default Variable Values] section can go anywhere in the entmgr.ini file.

Note: Older versions of Entrust/RA Toolkit and Entrust/RA (that is, released prior to the 5.1 release) cannot interpret the default variable syntax in certificates. If you are using older versions of Entrust/RA Toolkit or Entrust/RA, you should define default variable values in the entmgr.ini file, as required. For more information about this alternate method, see "Customizing the initialization files" on page 483.

Once a default value is established, you can then add new extensions or attributes with new variables to a certificate from Entrust/RA without having to modify certificates that already exist.

To add a default value to a variable definition

1. To add a default value to a variable definition, you need to customize the certificate. To customize certificates, you edit the master.certspec file. Before you can edit this file, you need to export it. In Entrust/RA, click File > Certificate Definitions > Export.
2. Place the default value before the Range or OneOf field. Note that the default is optional, and if the default value is not specified it implies that no default value is available.

   In this example:

   ```
   login_message=TextString,Message in Entrust-Ready
   clients:,Message text to appear in Entrust-Ready
   application.,Hello World,Range,0,80
   ```

   the words "Hello World" are the default value.
3. Save your changes and import the revised master.certspec file. In Entrust/RA, click File > Certificate Definitions > Import.

   When the master.certspec (or initial.certspec) file is imported, any default values are validated by Entrust/RA against the variable definition. If the value is invalid you will see an error message.

   You have now added a default value to a variable definition.

Example of user certificate types

The following excerpt from an example master.certspec file shows the additions to the master.certspec file to add an organization's address to Enterprise certificates issued for the ent_corporate certificate type.

```
[Certificate Categories]
.
.
.
[Certificate Types]
.
.
.
ent_corporate=enterprise,Corporate,Corporate Enterprise Certificates
;
[Extension Definitions]
.
.
.
[ent_corporate Common Extensions]
branchStreet=2.5.29.15,n,o,IA5String,"1000 Innovation Drive Ottawa"
.
.
.
[Advanced Settings]
.
.
.
[Database Field Definitions]
.
.
.
[Attribute Definitions]
.
.
.
[Variables]
.
.
.
[Obsolete Certificate Types]
.
.
.
```

Entrust/Timestamp and Entrust/Profile Server types

The master.certspec file includes a certificate type for Entrust/Timestamp and a certificate type for Entrust/Profile Server. These certificate types are enabled by default. The following excerpt from the master.certspec file shows these certificate type definitions.

```
[Certificate Categories]
.
.
.
[Certificate Types]
.
.
.
; ----------------------------------------------------------------------
; Timestamp Certificate Type
; ----------------------------------------------------------------------
ent_timestamp=enterprise,Timestamping Agent,Timestamping Agent Certificates
;
; ----------------------------------------------------------------------
; Profile Server Certificate Type
; ----------------------------------------------------------------------
ent_profsrvr=enterprise,ProfileServer,ProfileServer Certificates
.
.
.
```

If you do not use Entrust/Timestamp or Entrust/Profile Server, you can define these certificate types as obsolete in the master.certspec file or comment them out before initialization in the initial.certspec file. For information about defining certificate types as obsolete, see "Making a certificate type obsolete" on page 330.

Sample certificate types

The master.certspec file includes two sample Web certificate types:

- web_server that you may want to use to secure a Web server
- web_codesign that you may want to use to generate certificates to verify that you signed specific applications if you are a vendor

These certificate types have been provided because they are used for common security tasks.

If you do not use these certificate types, you can define them as obsolete in the master.certspec file or comment them out before initialization in the initial.certspec file. For information about defining certificate types as obsolete, see "Making a certificate type obsolete" on page 330.

```
[Certificate Categories]
.
.
.
[Certificate Types]
.
.
.
; ----------------------------------------------------------------------
; Sample Web Certificate Types
; Following are example certificate types for web servers and code signing.
; ----------------------------------------------------------------------
web_server=web,Web Server,Sample Web Server Certificates
web_codesign=web,Code Signing,Sample Code Signing Certificate

[Extension Definitions]
.
.
.
[web_server Verification Extensions]
;----------------------------------------------------------------------
;- Sample Web Server Certificate Type                                 -
;-                                                                    -
;- This server certificate type is designed for use by SSL servers.   -
;----------------------------------------------------------------------
; ----------------------------------------------------------------------
; <ASN1> keyUsage::=BIT STRING {digitalSignature(0), keyEncipherment(2)}
; ----------------------------------------------------------------------
keyusage=2.5.29.15,n,m,BitString,101
;
; ----------------------------------------------------
; <ASN1> netscapeCertType::=BIT STRING {sSLServer(1)}
; ----------------------------------------------------
netscapecerttype=2.16.840.1.113730.1.1,n,m,BitString,01
```

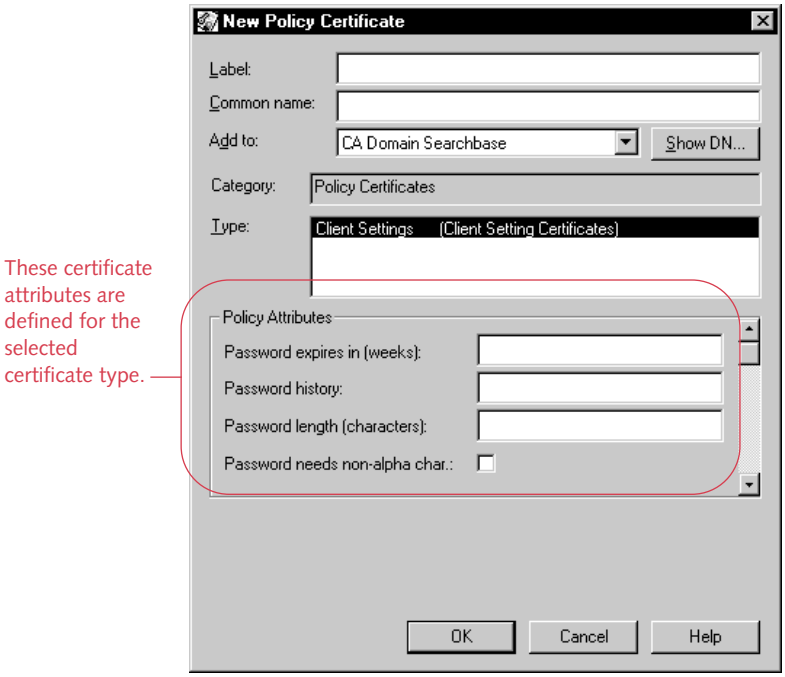
```
[web_codesign Verification Extensions]
;-----------------------------------------------------------------------
;- Sample Code Signing Certificate Type                                -
;-                                                                     -
;- The web_codesign certificate type is designed to verify code signed -
;- either by Netscape or Microsoft code signing technologies.          -
;-----------------------------------------------------------------------
; ---------------------------------------------------
; <ASN1> keyUsage::=BIT STRING {digitalSignature(0)}
; ---------------------------------------------------
keyusage=2.5.29.15,n,m,BitString,1
;
; -------------------------------------------------------
; <ASN1> netscapeCertType::=BIT STRING {objectSigning(3)}
; -------------------------------------------------------
netscapecerttype=2.16.840.1.113730.1.1,n,m,BitString,0001
.
.
.
```

Customizing policy certificates

Certificates issued from the policy category define user policies. User policies determine what users who belong to a particular Entrust role can and cannot do within Entrust/PKI (see "Creating a new user policy" on page 185). Entrust/Authority issues policy certificates when you create or modify a user policy.

You create or modify a user policy using the New Policy Certificate dialog box in Entrust/RA. In this dialog box, you choose the type of policy certificate that you want Entrust/Authority to issue. Currently one certificate type, named "Client Settings", is available.

The certificate type you choose determines the attributes that appear in the lower portion of the dialog box. This example shows attributes that appear when the Client Settings certificate type is selected.



After you provide values for the certificate attributes and click *OK*, Entrust/Authority issues the policy certificate, including the attributes you have defined. Once certificate attributes have been included in a policy certificate, they are referred to as "user policies".

Creating and modifying a certificate type

If the certificate type listed in the New Policy Certificate dialog box doesn't meet your organization's requirements, you can add attributes to this certificate type. You cannot remove or modify the predefined attributes. You can also define new certificate types in the policy certificate category.

Note: Entrust Technologies does not intend for you to create certificate types in the policy certificate category for use with Entrust-Ready applications. Entrust Technologies has provided this feature to enable you to create policy certificates for your own purposes.

You create and modify certificate types in the policy certificate category using the master.certspec file. This section assumes that you know how to create, open, and process the master.certspec file. If you don't know how to perform these tasks, see "Working with Entrust/Authority certificate definitions" on page 274. In the master.certspec file, you can

- create new policy certificate types. After you create a new certificate type and process the master.certspec file, the New Policy Certificate dialog box lists the new certificate type the next time you open it.
- add attributes to the default policy certificate type. After you modify the default policy certificate type and process the master.certspec file, Entrust/Authority begins issuing certificates according to your changes.

You create or modify a certificate type using a certificate type definition and one or more certificate attributes.

What is a certificate type definition?

The master.certspec file contains a certificate type definition for each certificate type in the policy certificate category. A certificate type definition

- uniquely identifies the certificate type in the master.certspec file
- specifies the certificate category that the certificate type belongs to
- specifies the name and description for this certificate type when it is displayed in the New Policy Certificate dialog box

What is a certificate attribute?

The master.certspec file contains one or more certificate attributes for each certificate type in the policy certificate category. A certificate attribute defines one value that Entrust/Authority inserts in certificates issued for this certificate type. The value provided by an attribute can be of a fixed or variable nature.

Editing a certificate type definition

Each certificate type for the policy certificate category is represented by a certificate type definition (see "What is a certificate type definition?" on page 307).

In the master.certspec file, certificate type definitions for all certificate types are located in the [Certificate Types] section. This example highlights the definition for the default certificate type in the policy certificate category.

```
[Certificate Categories]
.
.
.
[Certificate Types]




; ---------------------------------------------------------------------
; Default Certificate Types
;
; These type definitions must not be removed. Their <Type Name>
; and <Type Description> fields may be changed, however.
; ---------------------------------------------------------------------
ent_default=enterprise,Default,Default Enterprise Certificates
web_default=web,Default,Default Web Certificates (for browsers and secure
e-mail)
set_cardholder=set,Cardholder,Cardholder Certificates
set_merchant=set,Merchant,Merchant Certificates
set_gateway=set,Gateway,Payment Gateway Certificates
set_ca=set,CA,CA Certificates
xcert_default=xcert,Default,Default Cross-Certificates
xcert_pkix=xcert,PKIX,PKIX compliant Cross-Certificates
polcert_cliset=policycert,Client Settings,Client Setting Certificates
cacert_default=cacert,Default,Default Self Signed CA Certificates
cacert_link=cacert,Link,Self Issued Link CA Certificates
.
.
```

To add a certificate type definition

1. In the master.certspec file, locate the [Certificate Types] section header.
2. On a new line in the [Certificate Types] section, type the definition for the certificate type.

   You can add a new line anywhere in the [Certificate Types] section. Typically, however, you type definitions for new certificate types after the default ones.

The following command specifies a definition for a certificate type. (Table 14 describes the parameters used in definitions for policy certificate types, except for the Category ID, which is always "policycert".)

```
Type ID=policycert,Type Name,Type Description
```

Attention: Do not modify the Type ID or Category ID parameters for the default certificate type. Changes to these parameters may cause serious problems with your user policies.

Table 14: Parameters in policy certificate type definitions

Parameter	Description
Type ID	Provides a unique representation for the certificate type in the master.certspec file (for example, "polcert_cliset"). The characters may be either uppercase or lowercase. Entrust/Authority converts all characters to lowercase when processing the master.certspec file. There is no maximum length for the Type ID, although the first 20 characters must be unique. If the Type ID exceeds 20 characters, it is truncated to 20 characters when Entrust/Authority processes the master.certspec file. It's a good idea to identify the certificate category in the Type ID. For example, you can prefix the Type ID for policy certificate types with "polcert_".
Type Name	Defines the name for this certificate type that appears in the list of certificate types in the New User Policy dialog box. Note that the name cannot include commas (",") or semicolons (";").
Type Description	Defines the description for this certificate type that appears in the list of certificate types in the New User Policies dialog box. Note that the description cannot include commas (",") or semicolons (";").

For example, here are the values for the parameters that specify the definition for the default policy certificate type:

```
Type ID        Category ID      Type Name            Type Description
polcert_cliset=policycert,Client Settings,Client Setting Certificates
```

3 Add a comment that describes the purpose of the certificate type when you add a definition.

You are not required to add a comment, but you may find them useful. Comments help you remember the purpose the next time you edit the file. To include a comment, type a semicolon ("; "), followed by your comment. When Entrust/Authority processes the file, it ignores everything between the semicolon and the end of the line.

You have now added a certificate type definition.

Editing certificate attributes

Each certificate type for the policy certificate category contains certificate attributes (see "What is a certificate attribute?" on page 307) in the master.certspec file.

In the master.certspec file, certificate attributes for all policy certificate types are located in the [Attribute Definitions] section. Within the [Attribute Definitions] section, certificate attributes are organized according to their certificate type. The organization is identified using subsection headers. For example, certificate attributes for the polcert_cliset certificate type are defined in the [polcert_cliset Attributes] subsection of the [Attribute Definitions] section.

To define a subsection header for a certificate attribute

1 In the master.certspec file, locate the [Attribute Definitions] section header.

2 On a new line in this section, type the subsection header.

Subsection headers for certificate attributes use the following format:

[*Type ID* **Attributes]**

where "Type ID" represents the Type ID as defined in the [Certificate Types] section for the certificate type (for example, [polcert_cliset Attributes]).

Subsection headers must appear in square brackets and are case-sensitive.

Once you have created a subsection within the [Attribute Definitions] section, you can define its certificate attributes.

Defining a certificate attribute in a subsection

You define a certificate attribute within a subsection of the [Attribute Definitions] section of the master.certspec file by specifying information about the value provided by the attribute, and the value itself.

The value provided by a certificate attribute can be of a fixed or variable nature. For example, one certificate attribute for the default policy certificate type defines the lifetime of certificates. The lifetime is a variable value (that is, it is provided by the administrative user on a per-policy basis when creating the user policy).

Attention: Do not modify the certificate attributes that have been defined for the default policy certificate type. Changes may cause serious problems with your user policies.

To define a certificate attribute, type it in the appropriate subsection of the [Attribute Definitions] section. Certificate attributes use the following syntax, which is described in the following topics:

```
Attr Name=Attr OID,Attr Type,Attr Value
```

Note: Entrust/Authority uses the information you provide to encode it as an X.509 certificate attribute. You are not coding an X.509 attribute yourself.

For example, the following defines a certificate attribute named "policy_cert_lifetime":

Attr Name | Attr OID | Attr Type | Attr Value

```
policy_cert_lifetime=1.2.840.113533.7.77.13,Integer,<policy_cert_lifetime>
```

Note: The value "1.2.840.113533.7.77.13" is an OID that has been registered for this attribute.

Attr Name

The value of this parameter provides a unique representation for the certificate attribute in the master.certspec file. For example, the Attr Name for the following certificate attribute is "allowed_hashing_algs":

```
allowed_hashing_algs=1.2.840.113533.7.77.4,SeqOfUTF8String,"<al
lowed_hashing_algs>"
```

Attr OID

The value of this parameter represents the object identifier (OID) of the certificate attribute, in dotted decimal form. For example, the value "2.840.113533.7.77.4" in the following certificate attribute specifies the OID:

```
allowed_hashing_algs=1.2.840.113533.7.77.4,SeqOfUTF8String,"<al
lowed_hashing_algs>"
```

All attributes that you define must be registered and assigned object identifiers. If the attribute is not registered, you must first obtain an OID for that attribute by registering it. You obtain OIDs for attributes from your registration authority (for example, GTIS or ANSI).

If you use an OID in more than one certificate attribute, you should use the same Attr Name for the certificate attributes.

Attr Type

The value of this parameter identifies the data format of the value provided by the certificate attribute. For example, the data format for the following certificate attribute is "SeqOfUTF8String":

```
allowed_hashing_algs=1.2.840.113533.7.77.4,SeqOfUTF8String,"<al
lowed_hashing_algs>"
```

All the data formats that are valid for certificate extensions are also valid for certificate attributes. Table 11 lists and describes the values that you can specify for the Attr Type parameter. One additional data format is valid for certificate attributes. This data format is named "PasswordRules" and its values include a set of Entrust-defined password rules.

If the certificate attribute provides a value of a variable nature, you will define the variable (see "Editing a variable" on page 297). The data format that you specify for the Attr Type parameter determines the data format that you can specify for the variable when you define it. Table 12 on page 294 lists the data formats that you can specify for the "Extn Type" and the corresponding variable types. The same formats and variable types are available for "Attr Type". One additional format is available for "Attr Type". This data type is named "PasswordRules", and its corresponding variable data types are Integer and Boolean.

Attr Value

The value of this parameter identifies the value that Entrust/Authority is to include in certificates for this attribute. For example, the value for the following certificate attribute is provided by a variable named "allowed_hashing_algs":

```
allowed_hashing_algs=1.2.840.113533.7.77.4,SeqOfUTF8String,"<al
lowed_hashing_algs>"
```

Entrust/Authority adds the value to certificates just before issuing them. Entrust/Authority adds the value in its ASN.1 DER-encoded representation. The ASN.1 recommendation for data structures is defined in *Recommendation X.680: Abstract Syntax Notation One (ASN.1)* and is maintained by the ITU-T. To represent ASN.1 structures in a machine-readable form, Entrust/Authority encodes them using the Distinguished Encoding Rules (DER). These rules are just one of the methods defined in *Recommendation X.690: ASN.1 Encoding Rules* for encoding ASN.1 data structures. The X.690 recommendation is also maintained by the ITU-T.

For all of the formats listed in Table 11 except the DER format, Entrust/Authority can convert the value provided by the Attr Value parameter to its ASN.1 DER-encoded format before inserting the value in a certificate.

Note: If you want to include a complex value in a certificate attribute and you do not know how to convert the value to its ASN.1 DER-encoded representation, contact the ITU-T for *Recommendation X.690: ASN.1 Encoding Rules*. Converting values to their ASN.1 DER-encoded representations is a complex topic that goes beyond the scope of this documentation.

Editing a variable for a certificate attribute

If you define a certificate attribute that names a variable (instead of a fixed value), you must define the variable. Variables are defined in the [Variables] section of the master.certspec file.

You define variables for certificate attributes in the same way that you define variables for certificate extensions. For information, see "Editing a variable" on page 297.

Customizing cross-certificates

Certificates issued from the cross-certificate category enable CAs to establish direct trust. Entrust/Authority issues cross-certificates during the cross-certification process (see "Cross-certifying with other CAs" on page 203).

During the cross-certification process, you are prompted to choose a cross-certificate type. The cross-certificate type you choose determines the extra content that Entrust/Authority adds to the certificate, in addition to the standard X.509 content.

The dialog box that prompts you to choose a certificate type depends on the cross-certification method you are using, and whether you are initiating or completing cross-certification. For any of the dialog boxes, you can change the certificate type that is used to generate a cross-certificate by clicking another certificate type.

Creating and modifying a cross-certificate type

If the certificate types listed in the cross-certification dialog boxes don't meet your organization's requirements, you can create and modify certificate types. For example, you can create a certificate type in the cross-certificate category that specifies that only a specific subgroup of users in the cross-certified CA may be trusted.

Note: If the default certificate type meets your organization's requirements, you don't need to modify it or create new certificate types.

You create and modify certificate types in the cross-certificate category using the master.certspec file. This section assumes that you know how to create, open, and process the master.certspec file. If you don't know how to perform these tasks, see "Working with the master.certspec file" on page 277. In the master.certspec file, you can

- create new cross-certificate types. After you create a certificate type and process the master.certspec file, the cross-certification dialog boxes list the new certificate type the next time they appear.
- modify the default cross-certificate types. After you modify a certificate type and process the master.certspec file, Entrust/Authority begins issuing certificates according to your changes.

You create or modify a certificate type using a certificate type definition and one or more certificate extensions.

What is a certificate type definition?

The master.certspec file contains a certificate type definition for each certificate type in the cross-certificate category. A certificate type definition

- uniquely identifies the certificate type in the master.certspec file
- specifies the category that the certificate type belongs to
- provides the name and description for this certificate type when it is displayed in the cross-certification dialog boxes

What is a certificate extension?

The master.certspec file may contain certificate extensions for each certificate type in the cross-certificate category. A certificate extension defines one value that Entrust/Authority inserts in certificates issued for this certificate type. The value provided by an extension for a cross-certificate type must be of a fixed nature.

Certificate extensions defined in the master.certspec file define the extra content for certificates. These certificate extensions don't define the X.509 content that Entrust/Authority inserts by default. However, you can use the master.certspec file to exclude certificate extensions that add X.509 content.

Using certificate type definitions and certificate extensions

To create a new certificate type within the cross-certificate category, you edit the master.certspec file to add the

- certificate type definition (see "Editing a certificate type definition" on page 315)
- certificate extension (see "Editing certificate extensions" on page 318)

Editing a certificate type definition

Each certificate type for the cross-certificate category is represented by a certificate type definition (see "What is a certificate type definition?" on page 315) in the master.certspec file.

In the master.certspec file, certificate type definitions for all certificate types are located in the [Certificate Types] section. This example highlights the definition for the default certificate type in the cross-certificate category:

```
[Certificate Categories]
.
.
.
[Certificate Types]



; ---------------------------------------------------------------------
; Default Certificate Types
;
; These type definitions must not be removed. Their <Type Name>
; and <Type Description> fields may be changed, however.
; ---------------------------------------------------------------------
ent_default=enterprise,Default,Default Enterprise Certificates
web_default=web,Default,Default Web Certificates (for browsers and secure
e-mail)
set_cardholder=set,Cardholder,Cardholder Certificates
set_merchant=set,Merchant,Merchant Certificates
set_gateway=set,Gateway,Payment Gateway Certificates
set_ca=set,CA,CA Certificates
xcert_default=xcert,Default,Default Cross-Certificates
polcert_cliset=policycert,Client Settings,Client Setting Certificates
cacert_default=cacert,Default,Default Self Signed CA Certificates
cacert_link=cacert,Link,Self Issued Link CA Certificates
.
.
```

To add a certificate type definition

1. In the master.certspec file, locate the [Certificate Types] section header.
2. On a new line in the [Certificate Types] section, type the definition for the certificate type.

   You can add a new line anywhere in the [Certificate Types] section. Typically, however, you type definitions for new certificate types after the default ones.

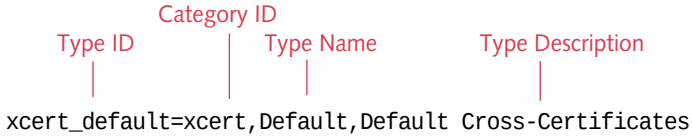
The following command specifies a definition for a cross-certificate type. (Table 15 describes the parameters used in cross-certificate type definitions, except for the Category ID, which is always "xcert".)

```
Type ID=xcert,Type Name,Type Description
```

Table 15: Parameters in definitions for cross-certificate types

Parameter	Description
Type ID	Provides a unique representation for the certificate type in the master.certspec file (for example, "xcert_default"). The characters may be either uppercase or lowercase. Entrust/Authority converts all characters to lowercase when processing the master.certspec file. There is no maximum length for the Type ID, although the first 20 characters must be unique. If the Type ID exceeds 20 characters, it is truncated to 20 characters when Entrust/Authority processes the master.certspec file. It's a good idea to identify the certificate category in the Type ID. For example, you can prefix the Type ID for cross-certificates with "xcert_".
Type Name	Defines the name for this certificate type that appears in the list of certificate types in the cross-certification dialog boxes. Note that the name cannot include commas (",") or semicolons (";").
Type Description	Defines the description for this certificate type that appears in the list of certificate types in the cross-certification dialog boxes. Note that the description cannot include commas (",") or semicolons (";").

For example, here are the values for the parameters that specify one of the default cross-certificate type definitions:



3 Add a comment that describes the purpose of the certificate type when you add a definition.

You are not required to add a comment, but you may find them useful. Comments help you remember the purpose the next time you edit the file. To include a comment, type a semicolon (";"), followed by your comment. When Entrust/Authority processes the file, it ignores everything between the semicolon and the end of the line.

You have now added a certificate type definition.

Editing certificate extensions

Each certificate type for the cross-certificate category contains certificate extensions (see "What is a certificate extension?" on page 315) in the master.certspec file.

The [Extension Definitions] section contains one subsection for certificate extensions that belong to a particular cross-certificate type.You define the subsections in the [Extension Definitions] section for certificate extensions for cross-certificate types just as you define subsections for certificate extensions for user certificate types. For example, certificate extensions for the xcert_default certificate type are defined in the [xcert_default Verification Definitions] section of the [Extension Definitions] section. For information about creating subsections within the [Extension Definitions] section, see "Associating a certificate extension with a certificate type" on page 288.

Note: Certificate extensions for cross-certificate types are included only in verification certificates. There is no reason to include these extensions in encryption certificates, which are used to encrypt information.

Once you have created a subsection within the [Extension Definitions] section, you can define its certificate extensions.

Defining a certificate extension in a subsection

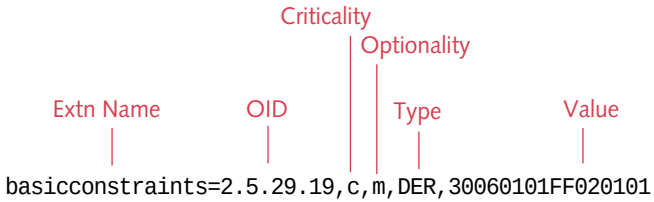
You define a certificate extension within a subsection of the [Extension Definitions] section of the master.certspec file by specifying information about the value provided by the extension, and the value itself.

To define a certificate extension, type it in the appropriate subsection of the [Extension Definitions] section. Certificate extensions use the following syntax:

```
Extn Name=Extn OID,Extn Crit,Extn Opt,Extn Type,Extn Value
```

Note: Entrust/Authority uses the information you provide to encode it as an X.509 certificate attribute. You are not coding an X.509 extension yourself.

For example, the following defines a certificate extension named "basicconstraints". The OID "2.5.29.19" has been registered for this certificate extension.



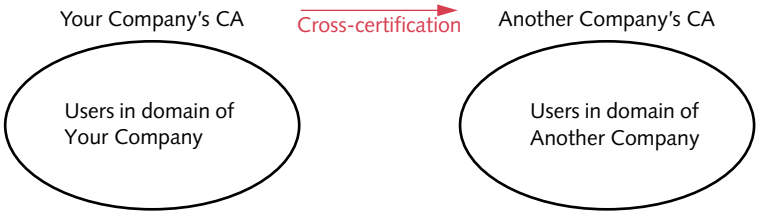
Note: You cannot include variable values in certificate extensions for cross-certificate types. You can include only fixed values.

For information about creating certificate extensions, see "Defining a certificate extension in a subsection" on page 289.

Applications of certificate extensions in cross-certificates

The value provided by a certificate extension may define a trust limitation between cross-certified CAs. By default, cross-certification enables all users in cross-certified domains to trust each other. For example, Figure 5 shows that Your Company's CA is cross-certified with Another Company's CA. The users who belong to Your Company trust the users who belong to Another Company.

Figure 5: Cross-certification of CAs



This default extension of trust is appropriate in many cases. For example, if you have users in different locations (for example, several branch offices), you may use one CA to manage each location. Because you are using different CAs only to manage your user base, you can cross-certify the CAs and you do not need to change the trust extended by default. All users in all CAs can trust each other.

There may be cases, however, when you want to cross-certify CAs, but you do not want to extend trust to all users in the cross-certified CAs. You may want to

control which users belonging to a cross-certified CA are trusted. For example, Your Company may want to trust only a subgroup of users who belong to Another Company.

Within Entrust/PKI, you can limit trust across cross-certified CAs in three ways. You can limit the trust extended to users in the domain of

- one cross-certified CA based on criteria about user DNs
- one cross-certified CA based on criteria about policy settings in certificates
- a series of two or more cross-certified CAs based on the number of consecutive CAs you want trust extended to

---

Note: Trust relationships between cross-certified CAs are defined by two trust models: the hierarchical trust model (see "Creating a hierarchy of CAs" on page 243) and the network trust model (see "Cross-certifying with other CAs" on page 203). When limiting trust between cross-certified CAs, the methods you use to identify trusted users are the same for both models of cross-certification. However, the relationships between the CAs are different.

Limiting trust using distinguished names

You can limit third-party trust based on user distinguished names (DNs). For users in the domain of a cross-certified CA, you specify that

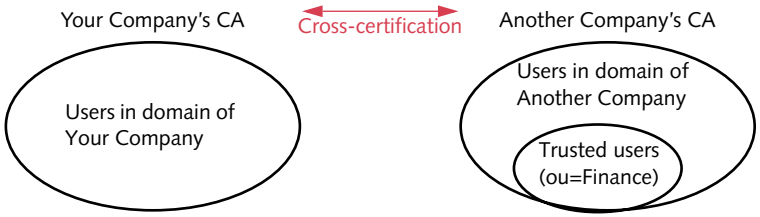
- only the users whose DNs meet criteria you specify are trusted
- the users whose DNs meet criteria you specify are not trusted

For example, suppose that Another Company identifies the department that a user belongs to in the organizational unit (ou) of the user's DN:

- John Smith works in the Finance department and is assigned the user DN "cn=John Smith, ou=Finance, o=Another Company, c=US"
- Alice Jones works in the Sales department and is assigned the user DN "cn=Alice Jones, ou=Sales, o=Another Company, c=US"

If Another Company cross-certifies with Your Company, you can limit third-party trust to only the users in the Finance department. Figure 6 shows how you can limit trust between CAs within the network trust model using user DNs. This example shows that users in Your Company trust users whose DNs include "ou=Finance" in Another Company's CA. All other users in the domain of Another Company are not trusted.

Figure 6: Trust limited by user DN



For the network cross-certification example shown in Figure 6, you can also limit third-party trust to all users in the domain of Another Company, except for users in the Finance department.

You can also use user DNs to limit trust between CAs within the hierarchical trust model. For example, you can specify that users should or should not trust users in the other domain whose DNs include "o=Another Company, c=US".

Limiting trust using policy settings

You can limit third-party trust based on policy settings in certificates. Policy settings used in certificates are created by administrative users and identified using object identifiers (OIDs). OIDs provide a standard mechanism for uniquely identifying things, or "objects", such as certificate policies, encryption algorithms, and Directory attributes. For information about OIDs, see the "About Entrust/PKI" chapter in *Using Entrust/PKI 6.0 on Windows*.

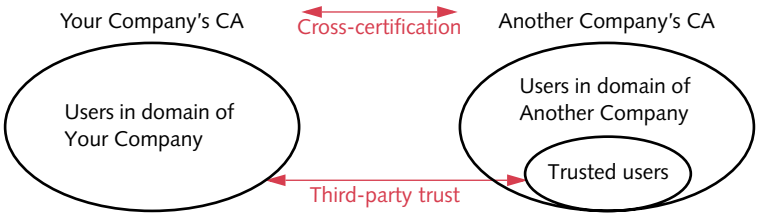
For users in the domain of a cross-certified CA, you specify that only users and CAs in the path with certificates that contain a policy setting configured to a specific value are trusted. All other users in the domain are not trusted.

For example, suppose that Another Company includes one of the following policy settings in certificates issued to users: "High assurance", "Medium assurance", or "Low assurance". These policy settings indicate how a user proved their identity to the CA before the certificate was issued:

- John Smith identified himself to the CA over the telephone, giving his employee number. His certificate includes the policy setting "Low assurance".
- Alice Jones identified herself to the CA in person, showing her passport. Her certificate includes the policy setting "High assurance".

If Your Company cross-certifies with Another Company, you can limit third-party trust to the users with certificates in which the Assurance policy setting is configured to either "Medium assurance" or "High assurance". Figure 7 shows that trust is limited to only users with "High assurance".

Figure 7: Trust limited by policy settings



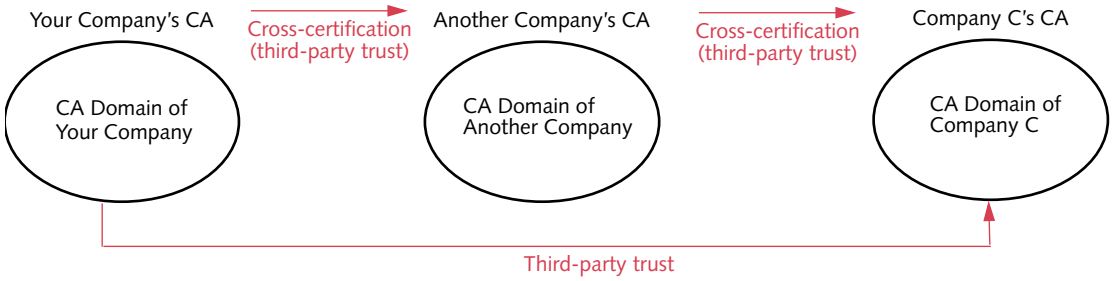
You can also use policy settings to limit trust between CAs within the hierarchical trust model. For example, you can specify that users who belong to one CA cannot trust users who belong to another CA if those users hold certificates with the policy setting "Low assurance".

Note: If a CA that you are cross-certifying uses a different OID to identify the policy setting, you "map" the OIDs. In the cross-certificates, you identify the OID that you are using to represent the policy setting and the OID that the other CA is using to represent the same policy setting.

Limiting trust using CA domains

By default, users in the domain of a cross-certified CA also trust all users in all domains across a series of two or more cross-certified CA domains. For example, as shown in Figure 8, suppose that your organization is cross-certified with Another Company, and Another Company is cross-certified with Company C. By default, all the users in the domain for your organization's CA trust all the users in the domain for Another Company's CA and all the users in the domain for Company C's CA.

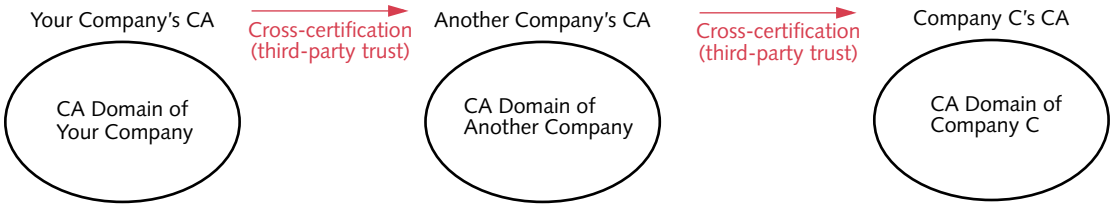
Figure 8: Default trust across domains of cross-certified CAs



In some cases, extending third-party trust to all users in the domains of cross-certified CAs may be appropriate. For example, if a different CA manages each department in your organization and all the CAs are cross-certified, you may want third-party trust to exist for all users in all the CA domains.

In other cases, you may want to limit third-party trust to only the users in CA domains for a maximum number of cross-certified CAs. For example, you may want to trust Another Company, but you do not want that trust to extend to Company C. You can limit third-party trust such that all the users in the domain for your organization's CA trust all the users in the domain for Another Company's CA, but not any of the users in the domain for Company C's CA (as shown in Figure 9).

Figure 9: Limited trust across domains of cross-certified CAs



You can also limit trust between CAs within the hierarchical trust model. For example, you can specify that a CA cannot trust one or more lower-level CAs.

Examples of cross-certificate types

The following excerpt from an example master.certspec file shows the additions to the master.certspec file to limit the trust extended to a cross-certified CA.

Note: The OIDs used in these examples have been registered for these certificate extensions.

```
[Certificate Categories]
.
.
.
[Certificate Types]
.
.
.
; ----------------------------------------------------------------------
; Your Company's certificate types
; ----------------------------------------------------------------------
xcert_sample=xcert,sample,Sample Cross-Certificates
.
.
.
[Extension Definitions]
;
[xcert_sample Verification Extensions]

; BasicConstraints with cA = TRUE, pathLenConstraint = 1
basicconstraints=2.5.29.19,c,m,DER,30060101FF020101

; Certificate policies with policyIdentifier = 1.1.1.1
certificatepolicies=2.5.29.32,c,m,DER,300730050603290101

; Policy mappings mapping 1.1.1.1 to 1.1.1.2
policymappings=2.5.29.33,n,m,DER,300C300A06032901010603290102

; Policy constraints with requireExplicitPolicy = 0
policyconstraints=2.5.29.36,c,m,DER,3003800100
.
.
.
```

Note: If you are trying to define pathlen=0, the DER representation is 30060101ff020100.

Customizing CA certificates

Certificates issued from the CA certificate category are used for two purposes:

- CA verification certificates contain the CA verification public key used to verify the CA signature on users' certificates.
- CA link certificates provide a connection between CA key pairs once they have been updated.

Entrust/Authority issues a CA verification certificate when you initialize Entrust/PKI after installation. Entrust/Authority issues CA link certificates when CA key pairs are updated. For information about CA verification and CA link certificates, see the "Maintaining the Certification Authority" chapter in *Using Entrust/PKI 6.0 on Windows*.

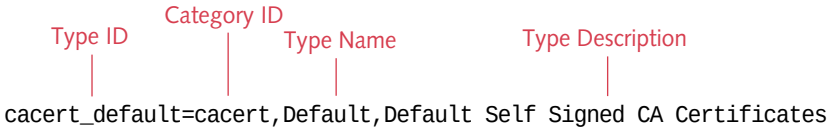
You can define additional content that you want Entrust/Authority to include in CA certificates. Entrust/Authority includes this additional content when issuing certificates for the certificate type.

Note: You cannot create new CA certificate types. You can only define the additional content that you want Entrust/Authority to include in certificates issued for the existing types.

The certificate types in the CA certificate category that Entrust/Authority issues are defined in the master.certspec file. The following excerpt from the default master.certspec file shows the definitions for the predefined CA certificate types.

```
[Certificate Categories]
.
.
.
[Certificate Types]
.
.
.
ent_default=enterprise,Default,Default Enterprise Certificates
web_default=web,Default,Default Web Certificates (for browsers and secure
e-mail)
set_cardholder=set,Cardholder,Cardholder Certificates
set_merchant=set,Merchant,Merchant Certificates
set_gateway=set,Gateway,Payment Gateway Certificates
set_ca=set,CA,CA Certificates
xcert_default=xcert,Default,Default Cross-Certificates
xcert_pkix=xcert,PKIX,PKIX compliant Cross-Certificates
polcert_cliset=policycert,Client Settings,Client Setting Certificates
cacert_default=cacert,Default,Default Self Signed CA Certificates
cacert_link=cacert,Link,Self Issued Link CA Certificates
.
.
.
```

For example, here are the values for the parameters that specify the definition for the predefined CA verification certificate type:

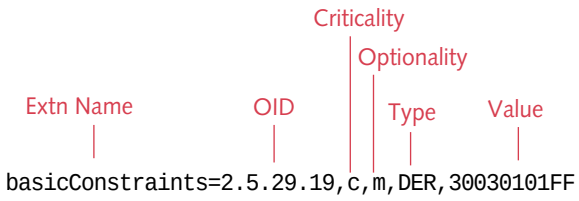


Extra content for CA certificates is defined using certificate extensions. The cacert_default certificate type does not have any predefined certificate extensions.

The cacert_link certificate type has two predefined certificate extensions. These extensions are required for certificates to comply with RFC 2549. The following excerpt from the master.certspec file shows the certificate extensions.

```
[Certificate Categories]
.
.
.
[Certificate Types]
.
.
.
cacert_link=cacert,Link,Self Issued Link CA Certificates
.
.
.
[Extension Definitions]
;
[cacert_link Verification Extensions]
basicConstraints=2.5.29.19,c,m,DER,30030101FF;BasicConstraints w/cA = TRUE
certificatePolicies=2.5.29.32,n,m,DER,300830060604551d2000;
.
.
.
```

Here are the values for the parameters that specify the default certificate extension for the CA link certificate type definition:



For information about defining certificate extensions for CA certificates, see "Editing certificate extensions" on page 287.

Excluding certificate extensions

Certificates issued by Entrust/Authority are compliant with the X.509 recommendation (see "Adding X.509 content to certificates" on page 272). This recommendation specifies certain certificate extensions that are required for X.509-compliant certificates. Entrust/Authority inserts the values provided by these X.509 extensions in certificates when they are issued.

Because Entrust/Authority automatically inserts the values provided by X.509 certificate extensions, they are not defined in the master.certspec file. Changes to these certificate extensions may cause certificates to no longer be compliant with the X.509 recommendation. However, you can specify that you want Entrust/Authority to exclude certain X.509 certificate extensions. Entrust/Authority does not insert the values provided by these extensions when issuing certificates.

Note: If you exclude *any* of the extensions from the X.509 recommendation, your certificates may no longer be compliant with this recommendation.

In the Enterprise, Web, and cross-certificate categories, you can exclude these certificate extensions from any certificate type:

- crlDistributionPoints
- privateKeyUsagePeriod
- basicConstraints

In the CA certificate category, you can exclude these certificate extensions for the cacert_default certificate type:

- crlDistributionPoints
- privateKeyUsagePeriod
- basicConstraints
- authorityKeyIdentifier
- keyUsage

In the CA certificate category, you can exclude these certificate extensions for the cacert_link certificate type:

- crlDistributionPoints
- privateKeyUsagePeriod
- basicConstraints
- keyUsage

The flags used in the master.certspec file to represent X.509 certificate extensions are

- noAuthorityKeyId
- noKeyUsage
- noBasicConstraints
- noCRLDistPoints
- noPrivateKeyUsage

To exclude an X.509 certificate extension

1. Locate the [Advanced settings] section of the master.certspec file.
2. In the [Advanced settings] section, type a section header using the following format:

   [*Type ID* **Advanced]**

   where "Type ID" is the identifier of the certificate type as defined in the [Certificate Types] section. For example, the following header identifies certificate extensions to exclude from the ent_default certificate type:

   ```
   [ent_default Advanced]
   ```

3. On a new line in this section, type the entry and set its value to "1".

   For example, the following example shows that the certificate extension represented by the crlDistributionPoints flag is to be excluded from certificates issued for the ent_default certificate type:

   ```
   [ent_default Advanced]
   noCRLDistPoints=1
   ```

Making a certificate type obsolete

If your organization no longer uses a certificate type, you cannot delete it even if no users are enabled with that type. Instead, you may make the certificate type obsolete.

You make a certificate type obsolete by defining its replacement certificate type. If Entrust/Authority receives a request to issue a certificate for a type that has been defined as obsolete (for example, when a certificate must be updated), Entrust/Authority uses the replacement certificate type.

You can make a certificate type obsolete if you (or another Security Officer) created it. You can also make the Entrust/Timestamp certificate type, the Entrust/Profile Server certificate type, and the sample Web certificate types obsolete.

Attention: Once a certificate type has been made obsolete, this action cannot be undone.

Defining a replacement certificate type

You make a certificate type obsolete by defining its replacement certificate type in the master.certspec file. For information about creating, opening and processing the master.certspec file, see "Working with Entrust/Authority certificate definitions" on page 274.

When deciding which certificate type will replace an obsolete type, consider the effect of the optionality of certificate extensions that provide variable values. If the replacement certificate type includes a certificate extension with a variable value that is not optional (that is, it is mandatory), Entrust/Authority cannot issue the certificate (unless a value for the variable has already been provided). The administrative user must re-enable each user and provide values for the mandatory certificate extension in the replacement certificate type.

To make a certificate type obsolete

1. In the master.certspec file, locate the [Obsolete Certificate Types] section.
2. Define a certificate type as obsolete using the following syntax:

   ```
   Obsolete Type ID=Replacement Type ID
   ```

   where

   - "Obsolete Type ID" identifies the certificate type that you want to make obsolete (as defined in the [Certificate Types] section)

- "Replacement Type ID" identifies the certificate type (as defined in the [Certificate Types] section) that you want to replace the obsolete one

For example, the following excerpt from an example master.certspec file shows that the sample certificate type, named "web_server", is obsolete. The default Web certificate type, named "web_default", replaces the "web_server" certificate type.

```
[Certificate Categories]
.
.
.
[Obsolete Certificate Types]
web_server=web_default
.
.
.
```

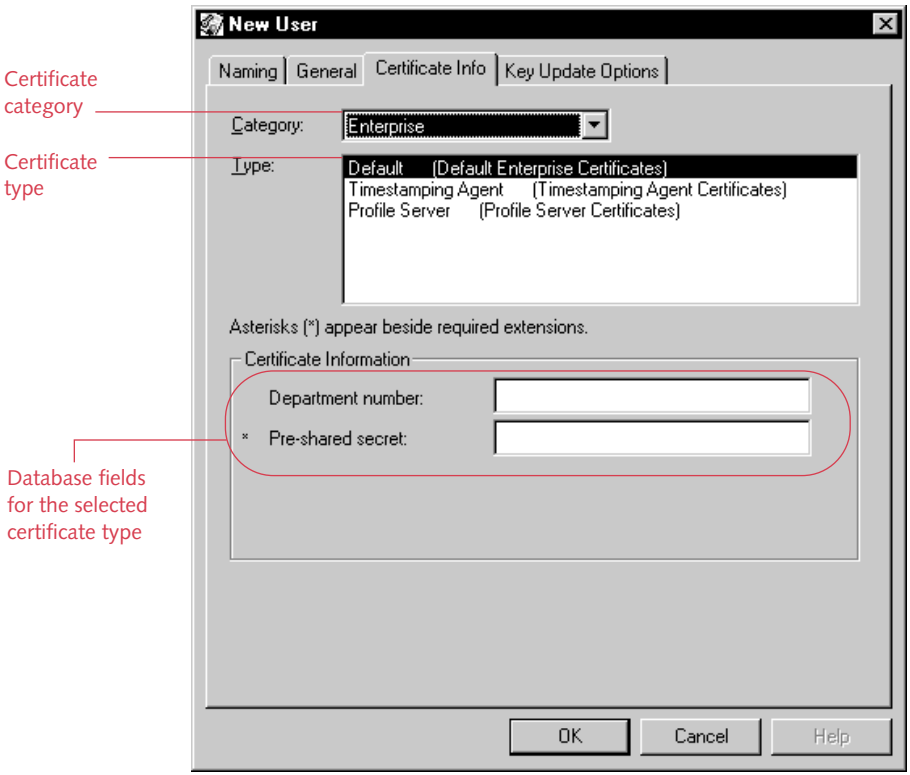
Note: It's a good idea to include a comment that describes why you made a certificate type obsolete. Comments help you remember the reason the next time you edit the file. To include a comment, type a semicolon (";"), followed by your comment. When Entrust/Authority processes the file, it ignores everything between the semicolon and the end of the line.

Once you have defined a certificate type as obsolete, you can remove its certificate extensions or certificate attributes, and variables from the master.certspec file. To remove certificate extensions or certificate attributes, and variables, for an obsolete certificate type, select the appropriate lines and delete them.

Customizing database fields

You can add custom information about users to the Entrust/Authority database. You add this information using "database fields".

Database fields are associated with user certificate types (that is, certificate types in the Enterprise and Web certificate categories). Entrust/RA prompts for the information for a database field when an administrative user enables a user using that certificate type. The database fields associated with a certificate type are listed in the New User dialog box.



Database fields are useful for storing information about users when you want to limit the people who have access to that information. For example, you can use database fields to store a "secret" provided by the user. The user must tell this secret to an administrative user before obtaining sensitive information via an unsecured method of communication (for example, the telephone). The information provided for the database fields is stored *only* in the Entrust/Authority database. Although database fields are associated with particular certificate types, the information is *not* inserted in a certificate.

You can create database fields if you are a Security Officer (or if you have the appropriate permissions, as explained in "Creating roles" on page 142). You create database fields using the master.certspec file.

Note: This section assumes that you know how to create, open, and process the master.certspec file. If you don't know how to perform these tasks, see "Working with Entrust/Authority certificate definitions" on page 274.

Defining a database field

You define database fields for a particular user certificate type. If the certificate type doesn't exist in the master.certspec file, you must define it. For information about defining a certificate type, see "Editing a certificate type definition" on page 285.

You define database fields in the [Database Field Definitions] section of the master.certspec file. A subsection header identifies the certificate type that database fields are associated with. For example, the following section header in the [Database Fields] section identifies database fields associated with the ent_default certificate type:

```
[ent_default Database Fields]
```

Each database field definition uses the following format. (Table 16 describes the parameters used in database field definitions.)

```
field_id=secrecy,optionality,<variable_id>
```

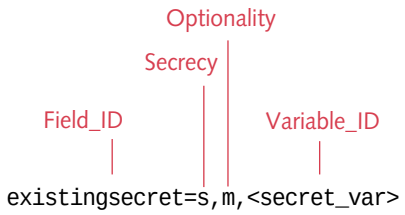
Table 16: Parameters in database field definitions

Parameter	Description
field_id	Identifies the field within the master.certspec file.
secrecy	Indicates whether the field is secret. The value is "s" (secret) or "n" (not secret). Values for secret database fields are encrypted in the Entrust/Authority database. Values for non-secret database fields are published in a database table named "CustomUserInfo". Each non-secret value is stored in its own row in this table, along with the user ID that corresponds to the user who is assigned this value, and the field_id. Using a third-party reporting tool, you can generate reports that include values for non-secret database fields.

Table 16: Parameters in database field definitions (continued)

Parameter	Description
optionality	Indicates whether a value must be provided for the field in the New User dialog box. The value is "o" (optional) or "m" (mandatory). An administrative user is required to enter values only for fields that are mandatory. These fields are identified by an asterisk ("*") in the New User dialog box.
variable_id	Names the variable that represents the value. You include only one variable for each database field. You must define the variable in the [Variables] section of the master.certspec file. You define variables for database fields just as you define variables for certificate extensions and certificate attributes. For information about defining variables, see "Editing a variable" on page 297.

For example, the following defines a database field named "existingsecret":



Example of database fields

The following example shows the additions to the master.certspec file to create three database fields for Enterprise certificates issued for the ent_default certificate type:

- a preshared secret that must remain confidential
- charge-back information that can be included in a report
- a comment pertaining to the user that can be included in a report

The comment information is optional, but the preshared secret and the chargeback information are required.

The following excerpt from the master.certspec file shows the information you enter to define these database fields:

```
[Certificate Categories]
.
.
.
[Certificate Types]
.
.
.
ent_default=enterprise,Default,Default Enterprise Certificates
.
.
.
[Database Field Definitions]

[ent_default Database Fields]
existingsecret=s,m,<secret_var>
chargeback=n,m,<chargeback_var>
comment=n,o,<comment_var>
.
[Variables]
secret_var=TextString,Pre-shared Secret,Secret information which identifies the
user.,Range,1,20

chargeback_var=TextString,Charge-back Dept.,Department to be charged for this
user's certificates.,Range,6,6

comment_var=TextString,Comment,Miscellaneous information about the
user.,Range,1,50
.
.
.
```

Removing X.509 certificate extensions

The size of a certificate depends on the names and extensions defined in the certificate. You can reduce the size of certificates by removing default extensions in the certificate definitions file.

You can modify of all types of X.509 certificates, including CA, cross-certification, and user certificates. This will be especially valuable when you want to reduce your certificate storage requirements, for example, if you plan to deploy WAP (Wireless Application Protocol) servers and devices.

X.509 certificates

There are six default extensions that Entrust/PKI 6.0 includes in every X.509 certificate (and two additional extensions in CA certificates, see "X.509 CA certificates" on page 337). Table 17 lists the six extensions and indicates which of the extensions you can remove.

Table 17: Removable X.509 certificate extensions

Extension	Removable?
Basic Constraints	Yes
Private Key Usage	Yes
Authority Key Identifier	Yes
Subject Key Identifier	Yes
CRL Distribution Points	Yes
Entrust Version Information	Yes (not recommended)[1]

1. Entrust recommends that you do not remove the Entrust Version Information extension for Enterprise certificates. This extension contains information needed to update a user's keys.

X.509 CA certificates

In CA certificates, there are eight extensions—two in addition to the six listed in Table 17 on page 336. Table 18 lists the eight extensions found in CA certificates and indicates which of the default extensions you can remove.

Table 18: Removable CA X.509 certificate extensions

Extension	Removable?
Basic Constraints	Yes
Private Key Usage	Yes
Authority Key Identifier	Yes
Subject Key Identifier	Yes
CRL Distribution Points	Yes
Entrust Version Information	Yes (not recommended)[1]
Netscape Certificate Type	Yes
Key Usage	Yes

1. Entrust recommends that you do not remove the Entrust Version Information extension for Enterprise certificates. This extension contains information needed to update a user's keys.

Removing default extensions from an X.509 certificate

You will edit the certificate definitions file to remove default extensions from an X.509 certificate.

To remove default extensions from an X.509 certificate

1. In Entrust/RA, click File > Certificate Definitions > Export.

   By default, the certificate definitions file is called "master.certspec"; however, this file may have been renamed when Entrust/PKI 6.0 was installed.

   Save the file to a known location on disk.
2. Open the file in a text editor.
3. To remove an extension from the existing ent_default or the cacert_default certificate types, scroll down to the [Advanced Settings] section and remove the comment (";") that precedes the extension.

4 If you want to remove an extension from a new certificate type, you must add additional information in that section.

Add the following text to remove the corresponding extension.

Extension	To remove, add this line
Basic Constraints	noBasicConstraints=1
Private Key Usage	noPrivateKeyUsage=1
Authority Key Identifier	noAuthorityKeyId=1
CRL Distribution Points	noCRLDistPoints=1
Entrust Version Information	noEntrustVersInfo=1
Netscape Certificate Type	noNetscapeCertType=1
Key Usage	noKeyUsage=1
Subject Key Identifier	noSubjectKeyId=1

For example:

```
[wap_default Advanced]

noBasicConstraints=1
noCRLDistPoints=1
noKeyUsage=1
```

5 Click File > Save and close the file.

6 In Entrust/RA, click File > Certificate Definitions > Import to import the updated file.

Make sure that you select the same file that you just edited.

You have now removed default extensions from an X.509 certificate.

Attention: By default, "noSubjectKey" does not appear in the initial.certspec or master.certspec file. This setting is intended for use with Entrust-Ready applications built on Release 5.1 (or newer) Entrust toolkits or for use with Entrust/Entelligence 5.1 or newer. If you are working with older clients, do not add this advanced setting to the initial.certspec or master.certspec file.

Turning off revocation checking for foreign certificates

Entrust clients (such as Entrust/RA and Entrust/Entelligence) rely on the Directory for revocation information in the form of revocation lists. When the subordinate CA certificate is issued by the superior CA, Entrust clients retrieve the revocation list issued by the superior CA to determine whether a subordinate CA certificate is revoked.

In certain environments, the clients may not be able to obtain the revocation list. This may happen for either of the following reasons:

- The superior CA does not publish revocation lists to the Directory.
- The superior CA's Directory is not accessible to the clients.

Since the clients do not have access to revocation lists issued by the superior CA, the clients will not be able to validate the subordinate CA certificate and hence will not be able to validate certificate chains that contain the subordinate CA certificate.

Revocation checking for foreign certificates can be turned off. From an end-entity's viewpoint, foreign certificates are those that are not issued by the CA that issued its certificates.

To turn off revocation checking

1. To turn off revocation checking, you need to add an attribute to the policy certificate. To customize the policy certificate, you edit the master.certspec file (or the initial.certspec file). Before you can edit this file, you need to export it. In Entrust/RA, click File > Certificate Definitions > Export.
2. Place this setting in the [polcert_cliset Attributes] section:

```
;
;---------------------------------------------
; Disable RL Checking for Foreign Certificates
;---------------------------------------------
check_rls=1.2.840.113533.7.77.18,Boolean,0
```

3. Save your changes and import the revised master.certspec file. In Entrust/RA, click File > Certificate Definitions > Import.
4. Once you have successfully imported the master.certspec file, test your change.

   After you have successfully imported the master.certspec file, Entrust/Authority should issue the following audit if the services are running:

```
[-07732 User Policy updated.]
```

The detailed audit message will list all the settings that went into the policy certificate and one of them should be

```
check_rls id-Entrust-SecureNetworks.77.18 010100
```

You have now turned off revocation checking.

Chapter 9

Command-line interface for Entrust/Authority Master Control

In Entrust/PKI, you can use a tool called Entrust Master Control Command Shell to perform Master User operations at the command line. You can use Entrust Master Control Command Shell to do everything from logging in to Entrust/Authority Master Control to setting encryption algorithms.

Entrust Master Control Command Shell consists of a command-line version of every task in its graphical counterpart, plus a few Entrust/RA and some utility commands (monitoring Entrust/PKI performance, for instance, or changing advanced settings).

As well, Entrust Master Control Command Shell has a number of commands that can only be executed within the shell, and not from the graphical counterpart.

In addition to giving you command-line options, Entrust Master Control Command Shell

- enhances infrastructure security by providing an alternative to the X11 window system on computers with a UNIX operating system
- enables some automated administration

Entrust Master Control Command Shell is installed when you install Entrust/PKI 6.0.

For a quick reference to the "entsh" command set, see also the *Entrust/Master Control Command Shell Quick Reference*.

This chapter contains the following sections:

- "Before you begin" on page 343
- "Initializing Entrust/PKI" on page 345

- "Starting and stopping Entrust Master Control Command Shell" on page 347
- "Logging in and out" on page 349
- "Changing Master User passwords" on page 351
- "Starting and stopping the Entrust/Authority service" on page 352
- "Enabling and disabling Entrust/Authority subsystems" on page 357
- "Administering the database" on page 360
- "Administering the Directory" on page 368
- "Setting the security policy" on page 373
- "Maintaining the CA" on page 381
- "About revocation lists" on page 395
- "Modifying behavior for certificate and revocation list extensions" on page 402
- "Using a Combined CRL" on page 410
- "Establishing compatibility with Entrust-Ready applications" on page 413
- "Adding information to user entries" on page 415
- "Encoding DNs" on page 417
- "Offline initialization of a subordinate CA" on page 419
- "Creating and recovering Security Officers" on page 423
- "Accessing certificate definition files" on page 430
- "Monitoring Entrust/PKI" on page 432
- "Getting help on Entrust Master Control Command Shell" on page 438
- "Entrust Master Control Command Shell security classes" on page 442
- "Command summary" on page 443

Before you begin

Here are some things you should know about Entrust Master Control Command Shell before you start to use it:

- Entrust Master Control Command Shell functions like a standard MS-DOS or UNIX shell. It's a command-line interface program you use to tell your computer what to do.
- You usually need to start Entrust Master Control Command Shell *and* log in to Entrust/Authority Master Control before you can run command shell commands. However, if autologin is enabled, you may not have to log in (see "Starting commands automatically" on page 354).
- You must be logged in to Windows as the local system administrator to run Entrust Master Control Command Shell.
- Entrust Master Control Command Shell has two modes: interactive mode and batch mode.
- Entrust Master Control Command Shell organizes commands into groups, called command sets ("db" for the database command set, for example, and "service" for the service command set). To run a command, you type the name of the command set it's in, followed by the name of the command, and press the Enter key. For example, the command to start the Entrust/Authority service is "start", and it is in the command set "service". You therefore enter "service start" to start the Entrust/Authority service.
- Commands have arguments, and these arguments follow the rules for Tcl scripting. For example, you use curly braces to group words into a single argument (see the "Administering multiple users" chapter in *Using Entrust/PKI 6.0 on Windows*).
- Online help is available. You can get syntax help on all command shell commands, as well as help on command sets. If you make a syntax error when running a command, Entrust Master Control Command Shell will suggest proper syntax.
- Error messages are logged using Entrust error codes. If you make a non-syntax error in a command, you will get an Entrust error message and description.
- You can use Entrust Master Control Command Shell to get detailed information on everything that happens when Entrust/Authority is running (see the procedures "To view audit logs in Entrust/RA" on page 19 and "To view an audit log in detail in Entrust/RA" on page 22).
- Commands are grouped into four security classes. The security class a command belongs to determines the number and type of authorizations needed to run the command. Class 0 commands are the most restricted, and Class 3 commands are the least restricted. The section "Entrust Master

Control Command Shell security classes" on page 442 describes the four classes in detail and lists all Entrust Master Control Command Shell commands by security class.

Initializing Entrust/PKI

If you have installed and configured a root CA, but you haven't yet initialized Entrust/PKI, you can use Entrust Master Control Command Shell to do so.

Initializing Entrust/PKI is the final step in the installation of Entrust/Authority. During this step, three Master Users and the First Officer choose, enter, and confirm their passwords. Once they have done this, Entrust/Authority is ready for use.

The settings for 10 variables appear in the [Advanced Settings] section of the configuration file "entmgr.ini" *before* you initialize Entrust/PKI (the entmgr.ini file is in the authdata folder). These variables are referred to as "advanced configuration variables" in this chapter.

Note: In new Entrust/PKI installations, the default "authdata" folder is created when you configure Entrust/Authority. However, this folder can have any name you choose.

You can set the value of advanced configuration variables before or after initializing Entrust/PKI. However, once you initialize Entrust/PKI, the [Advanced Settings] section is deleted from the entmgr.ini file. The settings are stored in the Entrust/Authority database, and you can change them only through Entrust Master Control Command Shell. For more information on these variables, see "Listing advanced configuration variables" on page 364.

Note: Two other Master Users (in addition to yourself) and one First Officer must all be present to enter their own passwords when you initialize Entrust.

To initialize Entrust/PKI

1. Start Entrust Master Control Command Shell on your workstation (see "Starting Entrust Master Control Command Shell" on page 347).
2. At the prompt, enter

   ```
   init
   ```

   You are prompted for the password for Master1.
3. All three Master Users and the First Officer now privately choose, enter, and confirm their user passwords.

   Master User and First Officer passwords must obey the rules set out in the "Setting up the Entrust/PKI environment" chapter of *Using Entrust/PKI 6.0 on Windows*.

A few minutes after the Master Users and First Officer enter and confirm their passwords, the following messages appear:

```
Initialization complete
creating CA profile...
creating First Officer profile...
You are logged in to Master Control
```

You are prompted to back up the Entrust/Authority database. It is strongly recommended that you perform this backup.

You have now initialized Entrust/PKI.

Starting and stopping Entrust Master Control Command Shell

This section describes how to start and stop Entrust Master Control Command Shell on your workstation.

Note: You must be logged in to Windows as the local system administrator to run Entrust Master Control Command Shell. If you aren't, you won't be able to log in to Entrust/Authority Master Control or run any of its commands.

Starting Entrust Master Control Command Shell

You usually need to start Entrust Master Control Command Shell *and* log in to Entrust/Authority Master Control before you can run command shell commands. However, if autologin is enabled, you may not have to log in (see "Starting commands automatically" on page 354).

This section explains how to start Entrust Master Control Command Shell. The section "Logging in to Entrust/Authority Master Control" on page 349 shows you how to log in to Entrust/Authority Master Control at the command line.

You can start Entrust Master Control Command Shell from the Programs menu . For more information, see "To start Entrust Master Control Command Shell from the Windows taskbar" on page 347.

To start Entrust Master Control Command Shell from the Windows taskbar

To start the command shell, in the Windows taskbar, click Start > Programs > Entrust PKI > Entrust Master Control Command Shell.

The Entrust Master Control Command Shell window appears. The window presents copyright information about Entrust/PKI 6.0, information on getting help in Entrust Master Control Command Shell, and the default Entrust Master Control Command Shell prompt (entsh$).

Stopping Entrust Master Control Command Shell

Once you've performed all the Entrust Master Control Command Shell operations you want to, you should log out of Entrust/Authority Master Control and stop Entrust Master Control Command Shell.

This section describes how to stop Entrust Master Control Command Shell. The section "Logging out of Entrust/Authority Master Control" on page 349 tells you how to log out of Entrust/Authority Master Control at the command line.

To stop Entrust Master Control Command Shell

1. Log out of Entrust/Authority Master Control if you haven't already done so (see "Logging out of Entrust/Authority Master Control" on page 349).
2. At the prompt, enter the following:

   ```
   exit
   ```

   What happens next depends on how you started Entrust Master Control Command Shell. For example, if you started Entrust Master Control Command Shell from the Programs menu in Windows , the Entrust Master Control Command Shell window closes.

   You have now stopped Entrust Master Control Command Shell.

Logging in and out

This section describes how use Entrust Master Control Command Shell to log in to and out of Entrust/Authority Master Control.

Logging in to Entrust/Authority Master Control

You usually need to log in to Entrust/Authority Master Control before you can perform Master User tasks at the command line. However, if autologin is enabled, you may not have to log in (see "Starting commands automatically" on page 354).

If you try to log in while your Entrust/Authority Master Control session is still active, the message "You are logged in to Master Control" appears.

To log in to Entrust/Authority Master Control

1. Start Entrust Master Control Command Shell on your workstation (see "Starting Entrust Master Control Command Shell" on page 347).
2. At the prompt, enter

   ```
   login
   ```

   Entrust Master Control Command Shell prompts you for your Master User name.
3. Enter your Master User username (Master1, Master2, or Master3).

   Entrust Master Control Command Shell prompts you for your Master User password.
4. Enter your Master User password.

   The message "You are logged in to Master Control" appears, and the prompt changes to specify both the CA DN and your Master User username.

   You have now logged in to Entrust/Authority Master Control.

Logging out of Entrust/Authority Master Control

When you've finished all the Master User operations you need to perform at the command line, log out of Entrust/Authority Master Control. This is a standard security measure.

To log out of Entrust/Authority Master Control

To log out of Entrust/Authority Master Control, enter

```
logout
```

The prompt "entsh$" appears.

Changing Master User passwords

You should change your Master User password periodically to ensure the security of your infrastructure. If you suspect that someone has discovered the password or that your Entrust profile is corrupt, change the password immediately and take any other steps your organization requires in such situations.

To change a Master User password

1. Start Entrust Master Control Command Shell on your workstation (see "Starting Entrust Master Control Command Shell" on page 347).
2. Log in to Entrust/Authority Master Control as a Master User (see "Logging in to Entrust/Authority Master Control" on page 349).
3. At the prompt, enter

   passwd *user*

   where "user" is the username of the Master User whose password you are changing.

Note: Master Users can change their own Master User password at any time, as well as the passwords of other Master Users. This makes it possible to recover Master Users who forget their password. (Depending on your security policy, two Master User authorizations may be needed when you are changing a Master User password other than your own.)

Entrust Master Control Command Shell prompts you to enter a new Master User password.

4. Enter the new Master User password.

   Entrust Master Control Command Shell prompts you to confirm the new Master User password.
5. Enter the new Master User password again.

   Entrust Master Control Command Shell prompts you for your current password.
6. Enter your current password.

   A message that the password has changed appears, followed by the prompt.

   You have now changed a Master User password.

Starting and stopping the Entrust/Authority service

The Entrust/Authority service consists of several subsystems. You can use Entrust Master Control Command Shell to configure seven of them:

- the Entrust Secure Exchange Protocol (SEP) subsystem
- the Key Generator subsystem
- the Automatic Backup subsystem
- the Database Integrity Check subsystem
- the CRL and Maintenance subsystem
- the Administration Service Handler (ASH) subsystem
- the Public-Key Infrastructure (X.509) - Certificate Management Protocol subsystem (also called PKIX-CMP)

Note: You can change only three of these subsystems in the Entrust/Authority Master Control graphical user interface: Entrust SEP, ASH, and PKIX-CMP.

This section describes how to perform the following tasks related to the Entrust/Authority service:

- "Displaying the status of the Entrust/Authority service" on page 352
- "Starting the Entrust/Authority service" on page 353
- "Stopping the Entrust/Authority service" on page 354
- "Starting commands automatically" on page 354

See the "Setting up the Entrust/PKI environment" chapter of *Using Entrust/PKI 6.0 on Windows* for more information on the Entrust/Authority service.

Displaying the status of the Entrust/Authority service

You can use Entrust Master Control Command Shell to find out the current status of all seven subsystems listed above. When you ask for this information, Entrust Master Control Command Shell tells you the status of *all seven subsystems*. If you want to know the status of only one subsystem, you can display the status of an individual subsystem (see "Enabling and disabling Entrust/Authority subsystems" on page 357).

In telling you the status of the Entrust/Authority subsystems, Entrust Master Control Command Shell tells you whether a subsystem is enabled or disabled, whether it is up or down, and how many processes are dedicated to it.

To display the status of the Entrust/Authority service

1. Start Entrust Master Control Command Shell on your workstation (see "Starting Entrust Master Control Command Shell" on page 347).
2. Log in to Entrust/Authority Master Control as a Master User (see "Logging in to Entrust/Authority Master Control" on page 349).
3. At the prompt, enter

   `service status`

   After a short interval, a list of all the Entrust/Authority subsystems and their current status appears, followed by the prompt. If all seven subsystems are enabled and running, for example, a message similar to the following would appear:

```
sep     Entrust SEP               enabled  up  2 processes
keygen  Key Generator             enabled  up  1 processes
backup  Automatic Backup          enabled  up  1 processes
integ   Database Integrity Check  enabled  up  1 processes
amb     CRL and Maintenance       enabled  up  1 processes
ash     Admin Service Handler     enabled  up  4 processes
cmp     PKIX-CMP                  enabled  up  2 processes
```

   You have now displayed the status of the Entrust/Authority service.

Starting the Entrust/Authority service

You don't need to start the Entrust/Authority service if all the tasks you're going to perform are Master User tasks. However, if you want to perform client or administrative operations, you must start the Entrust/Authority service.

Entrust/Authority subsystems are either enabled or disabled, and subsystems cannot start unless they are enabled. You have to start all enabled subsystems at once—you cannot start one subsystem at a time.

Note: Subsystems start automatically if they're enabled and any other Entrust/Authority subsystems are running.

How you start the Entrust/Authority subsystems therefore depends on what subsystems you want to start and whether any of the subsystems you want to start are disabled.

To start the Entrust/Authority service

1. Start Entrust Master Control Command Shell on your workstation (see "Starting Entrust Master Control Command Shell" on page 347).

2 Log in to Entrust/Authority Master Control as a Master User (see "Logging in to Entrust/Authority Master Control" on page 349).

3 Check the status of the Entrust/Authority service to make sure all the Entrust/Authority subsystems you want to start are enabled (see "To display the status of the Entrust/Authority service" on page 353).

4 At the prompt, type

```
service start
```

After a short interval, the prompt appears.

You have now started the Entrust/Authority service.

Stopping the Entrust/Authority service

Stopping the Entrust/Authority subsystems requires serious consideration. It's usually associated with making a major change in your Entrust/Authority installation. For example, if you plan to restore from backup, you must stop the Entrust/Authority subsystems before doing so. You usually keep Entrust/Authority running during routine operations.

The only subsystems you can stop are subsystems that are already enabled. Furthermore, you must stop subsystems as a group—you cannot stop them one at a time. See "Enabling and disabling Entrust/Authority subsystems" on page 357 for more information on enabling and disabling subsystems.

To stop the Entrust/Authority service

1 Start Entrust Master Control Command Shell on your workstation (see "Starting Entrust Master Control Command Shell" on page 347).

2 Log in to Entrust/Authority Master Control as a Master User (see "Logging in to Entrust/Authority Master Control" on page 349).

3 At the prompt, enter

```
service stop
```

After a short interval, the prompt appears.

You have now stopped the Entrust/Authority service.

Starting commands automatically

In Entrust/PKI, some commands can be performed without a Master User password, that is, without logging in. This feature is called "autologin".

Here are the commands you can run automatically if you enable autologin in Entrust Master Control Command Shell:

- service start (see "To start the Entrust/Authority service" on page 353)
- db backup full (see "To perform a full backup of the Entrust/Authority database" on page 361)
- db backup incremental (see "To perform an incremental backup of the Entrust/Authority database" on page 361)
- db backup query (see "To display the settings for automatic backup of the Entrust/Authority database" on page 362)
- db backup config (see "To configure the settings for automatic backup of the Entrust/Authority database" on page 362)
- db integrity validate (see "To manually validate the integrity of the Entrust/Authority database" on page 365)
- db integrity query (see "To display the settings for automatic validation of the Entrust/Authority database" on page 366)
- db integrity config (see "To configure the settings for automatic validation of the Entrust/Authority database" on page 366)
- db lsaudit (see "To get a summary of your audit logs" on page 432 and "To view audit logs" on page 433)

These commands are all "safe" commands, and they are useful for automating tasks. This puts them in Class 2 of the Entrust Master Control Command Shell grouping of commands by security class (see "Entrust Master Control Command Shell security classes" on page 442).

While autologin can be useful, it increases risk to your infrastructure because the Master User password is not required.

Before enabling autologin, your organization should carefully consider both your security requirements and your requirement to keep Entrust/PKI up and running. Entrust Technologies recommends that those who require maximum security not use this feature.

You can use the command shell to display the status of autologin, to enable autologin, or to disable autologin.

To display the status of autologin

1. Start Entrust Master Control Command Shell on your workstation (see "Starting Entrust Master Control Command Shell" on page 347).
2. Log in to Entrust/Authority Master Control as a Master User (see "Logging in to Entrust/Authority Master Control" on page 349).
3. At the prompt, enter

   ```
   service autologin query
   ```

After a short interval, "enabled" or "disabled" appears, followed by the prompt.

You have now displayed the status of autologin.

To enable autologin

1. Start Entrust Master Control Command Shell on your workstation (see "Starting Entrust Master Control Command Shell" on page 347).
2. Log in to Entrust/Authority Master Control as a Master User (see "Logging in to Entrust/Authority Master Control" on page 349).
3. At the prompt, enter

   ```
   service autologin enable
   ```

   After a short interval, the prompt appears.

   You have now enabled autologin.

To disable autologin

1. Start Entrust Master Control Command Shell on your workstation (see "Starting Entrust Master Control Command Shell" on page 347).
2. Log in to Entrust/Authority Master Control as a Master User (see "Logging in to Entrust/Authority Master Control" on page 349).
3. At the prompt, enter

   ```
   service autologin disable
   ```

   After a short interval, the prompt appears.

   You have now disabled autologin.

Enabling and disabling Entrust/Authority subsystems

Entrust/Authority has a number of subsystems. You can enable or disable these seven:

- the Entrust Secure Exchange Protocol (SEP) subsystem
- the Key Generator subsystem
- the Automatic Backup subsystem
- the Database Integrity Check subsystem
- the CRL and Maintenance subsystem
- the Administration Service Handler (ASH) subsystem
- the Public-Key Infrastructure (X.509) - Certificate Management Protocol subsystem (also called PKIX-CMP)

This section describes how to perform the following tasks associated with enabling and disabling subsystems:

- "To display the status of an Entrust/Authority subsystem" on page 357
- "To enable an Entrust/Authority subsystem" on page 358
- "To disable an Entrust/Authority subsystem" on page 358
- "To assign a specific number of processes to an Entrust/Authority subsystem" on page 359

Note: This section describes how to assign a specific number of processes to a subsystem. It does not tell you how to determine whether additional processes are needed. That topic is covered in the section "Assigning multiple processes to a subsystem" on page 75.

To display the status of an Entrust/Authority subsystem

1. Start Entrust Master Control Command Shell on your workstation (see "Starting Entrust Master Control Command Shell" on page 347).
2. Log in to Entrust/Authority Master Control as a Master User (see "Logging in to Entrust/Authority Master Control" on page 349).
3. At the prompt, enter

   `service config` *subsys* **`query`**

   where "subsys" refers to the subsystem whose status you want to know. All seven subsystems are listed in Step 3 on page 353.

For example, if you wanted to know the status of the Entrust SEP subsystem, you would enter the following:

```
service config sep query
```

A message appears, telling you whether the subsystem is enabled or disabled. The message also tells you the number of processes assigned to the subsystem. The prompt appears.

You have now displayed the status of an Entrust/Authority subsystem.

To enable an Entrust/Authority subsystem

1. Start Entrust Master Control Command Shell on your workstation (see "Starting Entrust Master Control Command Shell" on page 347).
2. Log in to Entrust/Authority Master Control as a Master User (see "Logging in to Entrust/Authority Master Control" on page 349).
3. At the prompt, enter

   service config *subsys* **enable**

   where "subsys" refers to the subsystem you want to enable. All seven subsystems are listed in Step 3 on page 353.

   For example, if you wanted to enable the Entrust SEP subsystem, you would enter the following:

   ```
   service config sep enable
   ```

   The prompt appears.

   You have now enabled an Entrust/Authority subsystem.

To disable an Entrust/Authority subsystem

1. Start Entrust Master Control Command Shell on your workstation (see "Starting Entrust Master Control Command Shell" on page 347).
2. Log in to Entrust/Authority Master Control as a Master User (see "Logging in to Entrust/Authority Master Control" on page 349).
3. At the prompt, enter

   service config *subsys* **disable**

   where "subsys" refers to the subsystem you want to disable. All seven subsystems are listed in Step 3 on page 353.

   For example, if you wanted to disable the Entrust SEP subsystem, you would enter the following:

   ```
   service config sep disable
   ```

The prompt appears.

You have now disabled an Entrust/Authority subsystem.

To assign a specific number of processes to an Entrust/Authority subsystem

1. Start Entrust Master Control Command Shell on your workstation (see "Starting Entrust Master Control Command Shell" on page 347).
2. Log in to Entrust/Authority Master Control as a Master User (see "Logging in to Entrust/Authority Master Control" on page 349).
3. At the prompt, enter

   `service config` *subsys* **`procs`** *n*

   where "subsys" refers to the subsystem you want to assign processes to and "n" refers to the number of processes you want to assign.

   For example, if you wanted to assign six processes to the Entrust SEP subsystem, you would enter the following:

   ```
   service config sep procs 6
   ```

Note: Three subsystems—Automatic Backup, Database Integrity Check, and CRL and Maintenance—require only one process. Disable a subsystem if you no longer require it.

The prompt appears.

You have now assigned a specific number of processes to an Entrust/Authority subsystem.

Administering the database

You can use Entrust Master Control Command Shell to maintain, verify, configure, and back up the Entrust/Authority database. See the following sections for more information:

- "Re-encrypting the database" on page 360
- "Backing up the Entrust/Authority database" on page 361
- "Updating the time-zone information for reports" on page 363
- "Listing advanced configuration variables" on page 364
- "Validating the integrity of the database" on page 365

Re-encrypting the database

Re-encrypting the database involves generating new random values for the keys used to encrypt the database and then using these keys to re-encrypt the database. Re-encrypting the database is potentially very dangerous. If the operation is interrupted for any reason, the Entrust infrastructure may be unusable and you will have to restore the database from backup.

To re-encrypt the database

1. Start Entrust Master Control Command Shell on your workstation (see "Starting Entrust Master Control Command Shell" on page 347).
2. Log in to Entrust/Authority Master Control as a Master User (see "Logging in to Entrust/Authority Master Control" on page 349).
3. At the prompt, enter

   db re-encrypt

   A message appears, warning you that re-encrypting the database is a serious undertaking and asking you to confirm that you want to proceed.
4. Enter

   y

   Another message appears, warning you that your Entrust/PKI will probably be unusable if re-encryption fails and asking whether you would like to perform a backup now.
5. Enter

   y

   When the backup is complete, this message appears:

   `backup completed successfully`

Re-encryption of the database starts. This may take some time to execute. When re-encryption is complete, the following message appears:

```
Database re-encryption completed successfully.
```

The prompt appears.

You have now re-encrypted the database.

Backing up the Entrust/Authority database

To help ensure that backups occur at frequent and regular intervals, Entrust/Authority automatically runs full and incremental backups. By default, Entrust/Authority automatically runs a full backup at 3:00 a.m every 24 hours, and an incremental backup every 60 minutes.

You can, however, run manual full or incremental backups. You can also find out what the current backup settings are and change them if you want to.

To perform a full backup of the Entrust/Authority database

1. Start Entrust Master Control Command Shell on your workstation (see "Starting Entrust Master Control Command Shell" on page 347).
2. Log in to Entrust/Authority Master Control as a Master User (see "Logging in to Entrust/Authority Master Control" on page 349).
3. At the prompt, enter

   ```
   db backup full
   ```

   When the backup is complete, this message appears:

   ```
   full backup completed successfully
   ```

   The prompt appears. This command may take some time to execute.

   You have now performed a full backup of the Entrust/Authority database.

To perform an incremental backup of the Entrust/Authority database

1. Start Entrust Master Control Command Shell on your workstation (see "Starting Entrust Master Control Command Shell" on page 347).
2. Log in to Entrust/Authority Master Control as a Master User (see "Logging in to Entrust/Authority Master Control" on page 349).
3. At the prompt, enter

   ```
   db backup incremental
   ```

Information about the progress of the backup appears. When the backup is complete, this message appears:

```
incremental backup completed successfully
```

The prompt appears. This command may take some time to execute.

You have now performed an incremental backup of the Entrust/Authority database.

To display the settings for automatic backup of the Entrust/Authority database

1. Start Entrust Master Control Command Shell on your workstation (see "Starting Entrust Master Control Command Shell" on page 347).
2. Log in to Entrust/Authority Master Control as a Master User (see "Logging in to Entrust/Authority Master Control" on page 349).
3. At the prompt, enter

   db backup query

   Information similar to the following appears, followed by the prompt:

```
backup-directory                C:\entbackup
keep-online                     7
incremental-backup-period       01:00
full-backup-period              24:00
not-before                      03:00
not-after                       05:00
```

You have now displayed the settings for automatic backup of the Entrust/Authority database.

To configure the settings for automatic backup of the Entrust/Authority database

1. Start Entrust Master Control Command Shell on your workstation (see "Starting Entrust Master Control Command Shell" on page 347).
2. Log in to Entrust/Authority Master Control as a Master User (see "Logging in to Entrust/Authority Master Control" on page 349).
3. At the prompt, enter a command similar to the following, specifying a value for at least one of "-dir", "-keep", and "-timer":

   db backup config -dir *<backup-directory>* -keep *<n>* -timer *<incremental-period>* *<full-period>* *<notbefore>* *<notafter>*

where

- "-dir" refers to the backup directory for the database. Note that in keeping with Tcl syntax, you must use forward slashes in the backup directory path; to use forward slashes, you must enclose the path in curly braces or use double backslashes, as in the following two examples:

  {C:\entbackup}
  C:\\entbackup

- "-keep" refers to the number of backup sets to keep online
- "-timer" refers to the backup period, with "incremental-period" the incremental backup period, "full-period" the full backup period, "notbefore" the time before which the full backup must not start, and "notafter" the time after which the full backup must not start

For example, to specify that nine backup sets be kept online, that incremental backups occur every two hours, that full backups occur every 24 hours, and that backups not start before 3 a.m. or after 5 a.m., enter

```
db backup config -keep 9 -timer 02:00 24:00 03:00 05:00
```

The prompt appears.

You have now configured the settings for automatic backup of the Entrust/Authority database.

Updating the time-zone information for reports

If you use third-party tools for database reporting, you need to update your time zone in Entrust/Authority if you

- accidentally installed Entrust/Authority before setting the correct time zone on a new server
- have physically moved the Entrust/Authority server to a different time zone

When you update time-zone information, Entrust/Authority recalculates and updates your time-zone offset from Greenwich Mean Time.

To update time-zone information

1. Start Entrust Master Control Command Shell on your workstation (see "Starting Entrust Master Control Command Shell" on page 347).
2. Log in to Entrust/Authority Master Control as a Master User (see "Logging in to Entrust/Authority Master Control" on page 349).
3. At the prompt, enter

   ```
   db reports-timezone
   ```

   The prompt appears.

You have now used Entrust Master Control Command Shell to update time-zone information for database reporting.

Listing advanced configuration variables

The settings for several variables appear in the [Advanced Settings] section of the configuration file "entmgr.ini" *before* you initialize Entrust/PKI (the entmgr.ini file is in the authdata folder). These variables are referred to as "advanced configuration variables" in this chapter.

You can set the value of advanced configuration variables before or after initializing Entrust/PKI. However, once you initialize Entrust/PKI, the [Advanced Settings] section is deleted from entmgr.ini. The settings are stored in the Entrust/Authority database, and you can change them *only* through Entrust Master Control Command Shell.

The variables are as follows:

- CompatLevel. This variable specifies the version of Entrust-Ready applications that Entrust/Authority Release 6.0 is compatible with.
- UserCertBasicConst. This variable specifies whether user certificates include the basicContraints extension.
- UserCertPrivateKeyUse. This variable specifies whether user certificates include the privateKeyUsage extension.
- CaCertPrivateKeyUse. This variable specifies whether CA certificates include the privateKeyUsage extension.
- KeyIdMode. This variable specifies how key identifiers are formatted.
- CrlUseDefRevokeReason. This variable specifies whether CRL/ARL revocation lists include the revocation reason extension if the administrative user doesn't specify a revocation reason.
- UseCombinedCRL. This variable specifies whether Entrust/Authority creates and maintains a Combined certificate revocation list (CRL) in the CA entry in the Directory.
- CombinedCRLLifetime. This variable specifies the lifetime of the combined CRL (in hours) if a Combined CRL exists.
- OffLineARL. This variable specifies the lifetime and next expiry date for offline ARLs, if the root CA is offline.
- AuditCRLWrite. This variable can suppress the logs that are issued for the "nearing expiration" reason.
- MSCompatibility. A setting that specifies whether Entrust/PKI is compatible with Microsoft CryptoAPI-enabled applications.
- UserObjectClasses. This variable provides a comma-delimited list of object classes that should be added to a user entry when the entry is enabled.

- DNEncoding. This variable specifies how attributes are encoded.

For information on how to change settings for each of these variables, see the following sections:

- "Establishing compatibility with Entrust-Ready applications" on page 413 for information on CompatLevel
- "Modifying behavior for certificate and revocation list extensions" on page 402 for information on UserCertBasicConst, UserCertPrivateKeyUse, CaCertPrivateKeyUse, KeyIdMode, and CrlUseDefRevokeReason
- "Using a Combined CRL" on page 410 for information on UseCombinedCRL and CombinedCRLLifetime
- "Adding information to user entries" on page 415 for information on UserObjectClasses
- "Encoding DNs" on page 417 for information on DNEncoding

To list all advanced configuration variables

1. Start Entrust Master Control Command Shell on your workstation (see "Starting Entrust Master Control Command Shell" on page 347).
2. Log in to Entrust/Authority Master Control as a Master User (see "Logging in to Entrust/Authority Master Control" on page 349).
3. At the prompt, enter

   ```
   db get all
   ```

   The name, value, and description of each advanced configuration variable appear, followed by the prompt.

   You have now listed all advanced configuration variables.

Validating the integrity of the database

Validating the Entrust/Authority database checks the integrity of its data to determine whether it has been corrupted by disk errors or by someone tampering with the protected data. Entrust/Authority automatically validates data every time a record in the database is used.

You can use Entrust Master Control Command Shell to manually validate the database, to find out what the settings for automatic validation are, and to change the settings for automatic validation.

To manually validate the integrity of the Entrust/Authority database

1. Start Entrust Master Control Command Shell on your workstation (see "Starting Entrust Master Control Command Shell" on page 347).

2 Log in to Entrust/Authority Master Control as a Master User (see "Logging in to Entrust/Authority Master Control" on page 349).

3 At the prompt, enter

```
db integrity validate
```

Information about the progress of the validation appears. When the validation is complete, a message similar to the following appears:

```
100% complete; estimated time remaining 00:00:00 @ (0.0/s)
```

Then the prompt appears.

You have now validated the integrity of the Entrust/Authority database.

To display the settings for automatic validation of the Entrust/Authority database

1 Start Entrust Master Control Command Shell on your workstation (see "Starting Entrust Master Control Command Shell" on page 347).

2 Log in to Entrust/Authority Master Control as a Master User (see "Logging in to Entrust/Authority Master Control" on page 349).

3 At the prompt, enter

```
db integrity query
```

A message similar to the following appears, indicating how often database validation occurs (measured by the number of hours between validations), the earliest validation can start, and the latest it can start:

```
period      24:00
not-before  02:00
not-after   05:00
```

The prompt appears.

You have now displayed the settings for automatic validation of the Entrust/Authority database.

To configure the settings for automatic validation of the Entrust/Authority database

1 Start Entrust Master Control Command Shell on your workstation (see "Starting Entrust Master Control Command Shell" on page 347).

2 Log in to Entrust/Authority Master Control as a Master User (see "Logging in to Entrust/Authority Master Control" on page 349).

3 At the prompt, enter a command similar to the following:

`db integrity config -timer` *`period notbefore notafter`*

where

- "period" refers to how often database validation occurs (measured by the number of hours between validations)
- "notbefore" refers to the earliest time at which validation can start
- "notafter" refers to the latest time at which validation can start

For example, to have the Entrust/Authority database validated every 24 hours, with the earliest starting time 3 a.m. and the latest starting time 5 a.m., enter

```
db integrity config -timer 23:00 03:00 05:00
```

The prompt appears.

You have now configured the settings for automatic validation of the Entrust/Authority database.

Enabling and disabling hardware-based database protection

Refer to "Hardware-based database protection" on page 100 for more information about the following commands:

- db hw-protection disable
- db hw-protection enable
- db hw-protection query

Administering the Directory

You can use Entrust Master Control Command Shell to perform the following Directory-related tasks:

- "Restoring CA data or all certificates to the Directory" on page 368
- "Writing CRLs and ARLs to the Directory" on page 369
- "Setting the CA or Directory Administrator password" on page 370
- "Changing the Directory Administrator DN" on page 371

Restoring CA data or all certificates to the Directory

The Directory stores the latest encryption certificate produced by Entrust/Authority. These include

- CA certificates
- policy certificates
- the First Officer certificate
- all cross-certificates
- certificate revocation lists (CRLs)
- authority revocation lists (ARLs)
- latest encryption certificates

If certificate information is lost or becomes corrupt (for example, if someone accidentally deletes or rearranges information in the Directory, or the hard drive fails) you can restore certificate information to the Directory.

You have two options when restoring Entrust/Authority data to the Directory: you can either restore CA data, or you can restore all certificates. The option to restore CA data restores all CA certificate information and CRLs but does not restore any user certificates. It does, however, restore the certificates of the First Officer and the CA profile. The option to restore all certificates restores all CA certificate information and CRLs and all user certificates.

Note: The procedure described in this section does not restore all data to the Directory. Full restoration involves restoring the Directory from backup and *then* restoring CA data or all certificates.

To restore CA data to the Directory

1. Start Entrust Master Control Command Shell on your workstation (see "Starting Entrust Master Control Command Shell" on page 347).
2. Log in to Entrust/Authority Master Control as a Master User (see "Logging in to Entrust/Authority Master Control" on page 349).
3. At the prompt, enter

   ```
   dsa restore ca
   ```

   where "ca" indicates that you are restoring CA data to the Directory.

   Information about the progress of the restore process appears, followed by the prompt. This command may take some time to execute.

   You have now restored CA data to the Directory.

To restore all certificates to the Directory

1. Start Entrust Master Control Command Shell on your workstation (see "Starting Entrust Master Control Command Shell" on page 347).
2. Log in to Entrust/Authority Master Control as a Master User (see "Logging in to Entrust/Authority Master Control" on page 349).
3. At the prompt, enter

   ```
   dsa restore all
   ```

   where "all" indicates that you're restoring all certificates to the Directory.

   Information about the progress of the restore process appears, followed by the prompt.

   You have now restored all certificates to the Directory.

Writing CRLs and ARLs to the Directory

A certificate revocation list (CRL) is a certificate containing the serial numbers of public key certificates that have been revoked, as well as the reason for each revocation. An authority revocation list (ARL) is a list of the serial numbers of CA public key certificates (including cross-certificates) that have been revoked.

CRLs and ARLs are stored in the Directory. The following procedure explains how to manually write current CRLs and ARLs to the Directory.

To write CRLs and ARLs to the Directory

1. Start Entrust Master Control Command Shell on your workstation (see "Starting Entrust Master Control Command Shell" on page 347).

2 Log in to Entrust/Authority Master Control as a Master User (see "Logging in to Entrust/Authority Master Control" on page 349).

3 At the prompt, enter the following:

```
dsa pushrls
```

The prompt appears.

You have now written CRLs and ARLs to the Directory.

Setting the CA or Directory Administrator password

The CA uses the CA password to authenticate itself to the Directory and write certificates and CRLs to it. Entrust/RA uses the Directory Administrator password to authenticate itself to the Directory to perform such tasks as adding or deleting users.

See *Installing Entrust/PKI 6.0 on Windows* for more information on creating these entries and passwords.

You can use Entrust Master Control Command Shell to change either of these passwords.

Note: Unless you are running Entrust/Directory 4.0 or earlier, you will need to use one of your Directory tools to change your Directory password before using the command shell to change either the CA or the Directory Administrator password.

To change the CA password

1 Start Entrust Master Control Command Shell on your workstation (see "Starting Entrust Master Control Command Shell" on page 347).

2 Log in to Entrust/Authority Master Control as a Master User (see "Logging in to Entrust/Authority Master Control" on page 349).

3 At the prompt, enter the following:

```
dsa passwd cadn
```

The following message appears:

```
Enter new LDAP password for CA DN:
```

4 Enter the LDAP (Lightweight Directory Access Protocol) password for the CA.

The following message appears, followed by the prompt:

```
dsa password changed
```

You have now changed the CA password.

To change the Directory Administrator password

1. Start Entrust Master Control Command Shell on your workstation (see "Starting Entrust Master Control Command Shell" on page 347).
2. Log in to Entrust/Authority Master Control as a Master User (see "Logging in to Entrust/Authority Master Control" on page 349).
3. At the prompt, enter the following:

   ```
   dsa passwd diradmin
   ```

   A message appears, asking you for the new LDAP (Lightweight Directory Access Protocol) password for the Directory Administrator:

   ```
   Enter new LDAP password for DirAdmin
   ```

4. Enter the new Directory Administrator password.

   The following message appears, followed by the prompt:

   ```
   dsa password changed
   ```

   You have now changed the Directory Administrator password.

Note: If you are running Entrust/Directory 4.0 or earlier, you can use this command with the ldap parameter to change the Directory Administrator password both in the Directory and in Entrust. First run the command "dsa passwd -ldap" to change the password in Entrust/Directory, and then continue with the remaining steps in this procedure.

Changing the Directory Administrator DN

The Directory Administrator DN is set during the installation of Entrust/PKI, when the DN entry is created in the Directory. If you change Directories or decide to reorganize your current Directory structure, you need to change your Directory Administrator DN.

You can use Entrust Master Control Command Shell to change the Directory Administrator DN.

Note: The Directory Administrator DN must be changed in two places: in the Directory and in Entrust. Before using the command shell to change it, use one of your Directory tools to change the DN in the Directory.

To change the Directory Administrator DN

1. Start Entrust Master Control Command Shell on your workstation (see "Starting Entrust Master Control Command Shell" on page 347).
2. Log in to Entrust/Authority Master Control as a Master User (see "Logging in to Entrust/Authority Master Control" on page 349).
3. At the prompt, enter

   dsa diradmin *dn*

   where "dn" is the DN of the Directory Administrator of the Directory you want to connect to.

   A message appears, asking you for the LDAP password for the new Directory Administrator.
4. Enter the requested LDAP password.

   The prompt appears.

   You have now changed the Directory Administrator DN.

Setting the security policy

You can use Entrust Master Control Command Shell to perform the following security-related tasks:

- "Changing the certificate hashing algorithm" on page 373
- "Changing the protocol encryption algorithm" on page 374
- "Changing the length of the user encryption key" on page 375
- "Changing the number of Master User authorizations" on page 377
- "Distributing setup information" on page 378

Changing the certificate hashing algorithm

The certificate hashing algorithm is the algorithm Entrust/Authority uses for hashing as part of the process of creating certificate signatures. Algorithm choices are SHA-1 (the default) and MD5. Organization policy usually determines which one is chosen.

Very few conditions warrant a change in algorithm. You would make such a change if organization policy regarding this algorithm changes.

You can use Entrust Master Control Command Shell to display your current algorithm and to change it.

To display the current certificate hashing algorithm

1. Start Entrust Master Control Command Shell on your workstation (see "Starting Entrust Master Control Command Shell" on page 347).
2. Log in to Entrust/Authority Master Control as a Master User (see "Logging in to Entrust/Authority Master Control" on page 349).
3. At the prompt, enter

   ```
   policy hasher query
   ```

   The name of the current certificate hashing algorithm appears, followed by the prompt. For example, if MD5 is the current certificate hashing algorithm, the following message appears, followed by the prompt:

   ```
   md5
   ```

   You have now displayed the current certificate hashing algorithm.

To change the certificate hashing algorithm

1. Start Entrust Master Control Command Shell on your workstation (see "Starting Entrust Master Control Command Shell" on page 347).

2 Log in to Entrust/Authority Master Control as a Master User (see "Logging in to Entrust/Authority Master Control" on page 349).

3 At the prompt, enter

```
policy hasher algorithm
```

where "algorithm" refers to the certificate hashing algorithm you want to change to.

What happens next depends on whether the current and requested algorithms are different or the same:

- If the current and requested algorithms are different, a message appears asking you to confirm the change. For example, if your current certificate hashing algorithm is MD5 and you ask to change to SHA-1, the following message appears:

  ```
  change hasher to SHA-1 (y/n) ? [y]
  ```

  Accept the default and go to Step 4.

- If the current and requested algorithms are the same, a message appears saying that the algorithm was not changed. For instance, if MD5 is *already* the certificate hashing algorithm and you ask to change to MD5, the following message appears:

  ```
  The hashing algorithm was not changed because it is already set
  to md5
  ```

4 The prompt appears.

You have now changed the certificate hashing algorithm.

Changing the protocol encryption algorithm

Entrust's protocol encryption algorithm secures information about user encryption keys exchanged by Entrust-Ready applications and Entrust/PKI during key update and user initialization.

You have two choices for the protocol encryption algorithms: CAST-128 and Triple DES. Organization policy usually determines which algorithm is chosen.

You would make such a change if organization policy regarding this algorithm changes.

You can use Entrust Master Control Command Shell to display your current algorithm and to change it.

To display the current protocol encryption algorithm

1 Start Entrust Master Control Command Shell on your workstation (see "Starting Entrust Master Control Command Shell" on page 347).

2 Log in to Entrust/Authority Master Control as a Master User (see "Logging in to Entrust/Authority Master Control" on page 349).

3 At the prompt, enter

```
policy proto-enc query
```

The name of the current protocol encryption algorithm appears, followed by the prompt. For example, if CAST-128 is the current protocol encryption algorithm, the following message appears, followed by the prompt:

```
cast-128
```

You have now displayed the current protocol encryption algorithm.

To change the protocol encryption algorithm

1 Start Entrust Master Control Command Shell on your workstation (see "Starting Entrust Master Control Command Shell" on page 347).

2 Log in to Entrust/Authority Master Control as a Master User (see "Logging in to Entrust/Authority Master Control" on page 349).

3 At the prompt, enter

policy proto-enc *algorithm*

where "algorithm" refers to the SEP encryption algorithm you want to change to.

The prompt appears.

You have now changed the SEP encryption algorithm.

Changing the length of the user encryption key

The user encryption key is used to create users' encryption key pairs. The algorithm options available for this key are RSA-1024 and RSA-2048. The default on installation is RSA-1024.

If your company decides the default is no longer strong enough, you can change to the RSA-2048 algorithm. This change will slow key generation considerably.

You can use Entrust Master Control Command Shell both to find out the current algorithm and to change it.

To display the current length of the user encryption key

1 Start Entrust Master Control Command Shell on your workstation (see "Starting Entrust Master Control Command Shell" on page 347).

2 Log in to Entrust/Authority Master Control as a Master User (see "Logging in to Entrust/Authority Master Control" on page 349).

3 At the prompt, enter the following:

```
policy user-enc-keylen query
```

Entrust Master Control Command Shell returns one of two values: 1024 or 2048.

The prompt appears.

You have now displayed the current length of the user encryption key.

To change the length of the user encryption key

1 Start Entrust Master Control Command Shell on your workstation (see "Starting Entrust Master Control Command Shell" on page 347).

2 Log in to Entrust/Authority Master Control as a Master User (see "Logging in to Entrust/Authority Master Control" on page 349).

3 At the prompt, enter

policy user-enc-keylen *length*

where "length" refers to the key length you want to change to.

What happens next depends on whether the current and requested key length are different:

- If the current and requested key length are different, a message appears stating that this will restart the services and asking if you want to continue. For instance, if your current user encryption key length is 1024 and you want to change to 2048, the following message appears:

  ```
  This will restart the services. proceed (y/n) ? [y]
  ```

  Accept the default and go to Step 4.

- If the current and requested key length are the same, a message appears saying that the algorithm length was not changed. For instance, if 2048 is *already* the user encryption key length and you ask to change to 2048, the following message appears:

  ```
  The user encryption key length was not changed because it
  already is set to 2048
  ```

4 The prompt appears.

You have now changed the length of the user encryption key.

Changing the number of Master User authorizations

During the initialization of the Entrust infrastructure, three Master Users must enter their passwords. This allows you to choose between one or two Master Users to authorize sensitive operations, depending on your organization's policy.

When Entrust/PKI is first installed, sensitive operations in Entrust/Authority Master Control require the authorization of only one Master User. However, you can specify that two or more Master User authorizations are needed for sensitive operations. This is called multiple Master User authorization.

You can use Entrust Master Control Command Shell to find out whether multiple Master User authorization is enabled; you can also use the command shell to enable or disable multiple Master User authorization.

To display the status of multiple Master User authorization

1. Start Entrust Master Control Command Shell on your workstation (see "Starting Entrust Master Control Command Shell" on page 347).
2. Log in to Entrust/Authority Master Control as a Master User (see "Logging in to Entrust/Authority Master Control" on page 349).
3. At the prompt, enter the following:

   ```
   policy auths query
   ```

   Either "enabled" or "disabled" appears, followed by the prompt.

   You have now displayed the status of multiple Master User authorization.

To enable multiple Master User authorization

1. Start Entrust Master Control Command Shell on your workstation (see "Starting Entrust Master Control Command Shell" on page 347).
2. Log in to Entrust/Authority Master Control as a Master User (see "Logging in to Entrust/Authority Master Control" on page 349).
3. At the prompt, enter the following:

   ```
   policy auths enable
   ```

   The prompt appears.

   You have now enabled multiple Master User authorization.

To disable multiple Master User authorization

1. Start Entrust Master Control Command Shell on your workstation (see "Starting Entrust Master Control Command Shell" on page 347).

2 Log in to Entrust/Authority Master Control as a Master User (see "Logging in to Entrust/Authority Master Control" on page 349).

3 At the prompt, enter the following:

```
policy auths disable
```

The following message appears:

```
This command must be authorized by an additional master user
Enter user name
```

4 Enter the name of a second Master User.

You are prompted for the password of the second Master User.

5 Enter the password of the second Master User.

The prompt appears.

You have now disabled multiple Master User authorization.

Distributing setup information

Before users can be enabled in Entrust/PKI, a reference number and authorization code must be used to create a profile. These are called "activation codes" or "setup information".

Setup information can be written to a file and distributed to users, and the information in the file can be encrypted with a password you specify. You can enable or disable the distribution and encryption of this setup information. You can also find out whether setup information is currently being encrypted.

To display whether setup information is being encrypted

1 Start Entrust Master Control Command Shell on your workstation (see "Starting Entrust Master Control Command Shell" on page 347).

2 Log in to Entrust/Authority Master Control as a Master User (see "Logging in to Entrust/Authority Master Control" on page 349).

3 At the prompt, enter the following:

```
policy authcode-dist query
```

A message appears, telling you whether or not setup information is currently being encrypted.

The prompt appears.

You have now displayed whether setup information is currently being encrypted.

To enable the encryption of setup information

1. Start Entrust Master Control Command Shell on your workstation (see "Starting Entrust Master Control Command Shell" on page 347).
2. Log in to Entrust/Authority Master Control as a Master User (see "Logging in to Entrust/Authority Master Control" on page 349).
3. At the prompt, enter the following:

   `policy authcode-dist enable`

   A message appears, warning you that this will restart the services and asking if you want to proceed. Select the default to continue.

   The following message appears:

   `encrypt distributed authcodes (y/n) ? [y]`

4. Select the default.

   The following message appears, asking you for the password for encrypting authorization codes:

   `Enter new password for encrypting authcodes:`

Note: If you enter "n", a message appears, telling you that setup information can be distributed in plaintext (that is, unencrypted).

---

5. Enter the password you want to use to encrypt authorization codes.

   The command shell asks you to confirm this password:

   `Confirm new password for encrypting authcodes:`

6. Enter the same password again.

   The prompt appears.

   You have now enabled the encryption of setup information.

To disable the encryption of setup information

1. Start Entrust Master Control Command Shell on your workstation (see "Starting Entrust Master Control Command Shell" on page 347).
2. Log in to Entrust/Authority Master Control as a Master User (see "Logging in to Entrust/Authority Master Control" on page 349).
3. At the prompt, enter the following:

   `policy authcode-dist disable`

After a short interval, the prompt appears.

You have now disabled the encryption of setup information.

Maintaining the CA

You can use Entrust Master Control Command Shell to perform the following CA-related Master User tasks:

- "Recovering the CA profile" on page 381
- "Resetting the certificate serial number counter" on page 382
- "Displaying a list of CA certificates" on page 382
- "Exporting a CA certificate to a file" on page 383
- "Changing the lifetime of the next CA certificate" on page 385
- "Changing the CA key pair algorithm" on page 386
- "Updating the CA key pair" on page 389
- "Revoking the CA certificate" on page 392
- "Recovering all users" on page 394

Recovering the CA profile

The CA has its own profile just as users do. This profile is used by the Administration subsystem and for moving and archiving users. Entrust/Authority creates a CA profile with a random password during first-time initialization and stores the password securely in the Entrust/Authority database. You must recover the CA profile if its .epf file is damaged or lost.

When you recover a CA profile, Entrust/Authority automatically creates a new profile with a new signing private key and verification public key certificate and generates a new random password; this all happens with no user intervention. The CA profile key pair has a lifetime of 365 days and undergoes key update after 265 days.

For more information on updating keys, see the "About Entrust/PKI" chapter in *Using Entrust/PKI 6.0 on Windows*.

To recover the CA profile

1. Start Entrust Master Control Command Shell on your workstation (see "Starting Entrust Master Control Command Shell" on page 347).
2. Log in to Entrust/Authority Master Control as a Master User (see "Logging in to Entrust/Authority Master Control" on page 349).
3. At the prompt, enter the following command:

   ```
   recover ca
   ```

   After a short interval, the prompt appears.

   You have now recovered the CA profile.

Resetting the certificate serial number counter

Whenever you restore the Entrust/Authority database to the last full or incremental backup, your serial number counter is automatically resynchronized. If you perform a manual restore (that is, if you restore the database using any method other than Entrust Technologies' "entrestore" command), you need to manually resynchronize the serial number counter.

To reset the certificate serial number counter

1. Start Entrust Master Control Command Shell on your workstation (see "Starting Entrust Master Control Command Shell" on page 347).
2. Log in to Entrust/Authority Master Control as a Master User (see "Logging in to Entrust/Authority Master Control" on page 349).
3. At the prompt, enter the following command:

   `ca resynch-next-cert-serial`

   After a short interval, the prompt appears.

   You have now reset the certificate serial number counter.

Displaying a list of CA certificates

You can display a list of all CA certificates that have been issued, as well as obtain detailed information about a specific CA certificate.

To display a list of CA certificates

1. Start Entrust Master Control Command Shell on your workstation (see "Starting Entrust Master Control Command Shell" on page 347).
2. Log in to Entrust/Authority Master Control as a Master User (see "Logging in to Entrust/Authority Master Control" on page 349).
3. At the prompt, enter the following command:

   `ca cert list`

   A list of all CA certificates is displayed. An example list follows:

```
Serial        Type    Issue Date            Expiry Date            Revoked
973001127     CA      2000/10/31 08:35:29   2020/10/31 09:05:29
```

   You have now displayed a list of all CA certificates.

To obtain detailed information about a CA certificate

1. Start Entrust Master Control Command Shell on your workstation (see "Starting Entrust Master Control Command Shell" on page 347).
2. Log in to Entrust/Authority Master Control as a Master User (see "Logging in to Entrust/Authority Master Control" on page 349).
3. At the prompt, enter the following command:

   ca cert get *sernum*

   where "sernum" is the serial number of the CA certificate for which you want information.

   Information about the CA certificate appears. Example information follows:

```
Serial Number:          973001127
Type:                   CA
Issued By:              o=Your Company,c=US
Subject:                o=Your Company,c=US
Issue Date:             2000/10/31 08:35:29
Expiry Date:            2020/10/31 09:05:29
Key Expiry Date:        2020/10/31 09:05:29
Private Key Type:       1024 bit RSA
Authority Key Id:       90838783018FAD464AD05055EC07D95CE624766
Subject Key Id:         90838783018FAD464AD05055EC07D95CE624766
Revoked:                no
```

   You have now obtained information about a specific CA certificate.

Exporting a CA certificate to a file

Some applications may require a CA certificate from other applications as a prerequisite for working with them. If an application you want Entrust/PKI to

work with has this requirement, you can meet it by exporting a CA certificate to a file and submitting the file to the other application.

Table 19: Command-line interface command for exporting a CA certificate

Command	Parameters	Description
ca cert export	-binary | -pem {-x509 -cert | -pkcs7} | -wtls [-overwrite] <file>	Choose either binary or pem, depending on which file format you want the CA certificate to be written in. The choice of file formats is provided for interoperability with PKIs from other vendors. Entrust/PKI can read certificates exported in either file format. Choose wtls if you are exporting the CA certificate for use with WAP servers, otherwise choose x509. If you are exporting the CA as an x509 certificate, you must choose to export it either as raw certificate data (cert), or you can export it using the PKCS#7 standard (pkcs7). In a scenario where there is a hierarchy of CAs, the raw certificate data only contains information about the certificate you are exporting, whereas the PKCS#7 format includes the certificate chain, including any superior CAs, if the certificate you are exporting is from a subordinate CA. Include the overwrite option if you want to overwrite an existing file. Specify a filename for the exported CA certificate. If you are exporting the certificate in binary format, include an extension of .der at the end of the file name. If you are exporting the certificate in PEM format, include an extension of .pem at the end of the filename.

To export a CA certificate to a file

1. Start Entrust Master Control Command Shell on your workstation (see "Starting Entrust Master Control Command Shell" on page 347).

2. Log in to Entrust/Authority Master Control as a Master User (see "Logging in to Entrust/Authority Master Control" on page 349).
3. At the prompt, enter the "ca cert export" command.

   For example, the following command writes the CA certificate to a binary file as an x509 certificate in PKCS#7 format:

```
ca cert export -binary -x509 -pkcs7 "c:/program files/entrust/cacert.der"
```

   Note that the path to the filename is surrounded by double quotes. This allows you to have spaces and special characters in your path. Also, it is important that you use forward slashes in your path rather than backslashes, because backslashes may be interpreted as part of an escape sequence (such as "\n" for newline, or "\t" for tab, and so on).

   The following example writes a CA certificate to a PEM file as a WTLS certificate:

```
ca cert export -pem -wtls cacert.pem
```

   If you do not specify a path, as in this case, the file is saved by default to the authdata directory.
4. When the CA certificate has been written to file, a message appears, affirming that the operation was successful. The message also displays the validation string for the exported CA certificate. Later on, when the exported CA certificate is imported elsewhere (for example, on a WAP server, or by a Security Officer at another CA), this validation string is used for confirming that the exported CA certificate has not been altered.
5. Copy the validation string down on paper.

   You have now exported a CA certificate to a file.

Changing the lifetime of the next CA certificate

If you are going to generate a new CA certificate, you can specify its lifetime and private key usage period before you generate it. You can also display the lifetime and private key usage period for the current CA certificate.

Note: The CA certificate lifetime is measured in months, and the CA private key usage period is always a percentage of the CA certificate lifetime.

To display the lifetime and private key usage period of the CA certificate

1. Start Entrust Master Control Command Shell on your workstation (see "Starting Entrust Master Control Command Shell" on page 347).

2 Log in to Entrust/Authority Master Control as a Master User (see "Logging in to Entrust/Authority Master Control" on page 349).

3 At the prompt, enter the following command:

```
ca cert query
```

The lifetime and private key usage period of the current CA certificate are displayed. An example display follows:

```
CA certificate lifetime (in months)                        240
CA private key usage (% of CA certificate lifetime)        100
```

You have now displayed the lifetime and private key usage period of the current CA certificate.

To change the lifetime and private key usage period of the next CA certificate

1 Start Entrust Master Control Command Shell on your workstation (see "Starting Entrust Master Control Command Shell" on page 347).

2 Log in to Entrust/Authority Master Control as a Master User (see "Logging in to Entrust/Authority Master Control" on page 349).

3 At the prompt, enter the following command:

```
ca cert config (-lifetime value | -period value)+
```

where "lifetime" is measured in months, and "period" is a percentage of the CA certificate lifetime. You can configure either "lifetime" or "period", or you can configure both in one command (as indicated by the "+"). The default value for the CA certificate lifetime is 240 months, and the default value for the CA private key usage period is 100%.

Note: The "ca cert config" command affects only the *next* CA certificate you generate.

You have now changed the lifetime and private key usage period of the next CA certificate to be generated.

Changing the CA key pair algorithm

You can change the CA key pair algorithm used to create the Certification Authority (CA) key pair using Entrust/Authority Master Control. You might change the CA key pair algorithm, for example, if you plan to generate a new CA key pair.

For more information about the CA key pair, see "Updating the CA key pair" on page 389.

The ECDSA CA key pair algorithm

In addition to a choice of RSA and DSA algorithms, you can also create an elliptic curve key to create the CA key pair. ECDSA keys and the digitial signatures are shorter than RSA keys and signatures and offer comparable cryptographic strength, making elliptic-curve cryptography ideal for low-bandwidth environments.

When you select ECDSA, you must also choose the curve you want to use. These are your choices:

- default
  The default curve used when you select ECDSA is a 192-bit elliptic curve (see "To change the CA key pair algorithm" on page 388).
- named curve
  If your organization requires the use of another elliptic curve that is listed in the X.962 standard, you can do so as well (see "To choose a named ECDSA elliptic curve" on page 388).

  For a complete list of all supported named curves, see Table 11 on page 607.
- custom curve
  You can also use non-standard, or custom, elliptic curves using the format described in X.962 (see "To choose a custom ECDSA elliptic curve" on page 388

Note: To view information about an ECDSA elliptic curve (such as curve name and other characteristics), see "To write ECDSA curve data to a file" on page 389.

To display the CA key pair algorithm

1. Start Entrust Master Control Command Shell on your workstation (see "Starting Entrust Master Control Command Shell" on page 347).
2. Log in to Entrust/Authority Master Control as a Master User (see "Logging in to Entrust/Authority Master Control" on page 349).
3. At the prompt, enter the following command:

   ```
   ca key query
   ```

   The algorithm of the current CA key pair is displayed, for example:

   ```
   CA key pair algorithm RSA-1024
   ```

You have now displayed the CA key pair algorithm.

To change the CA key pair algorithm

1. Start Entrust Master Control Command Shell on your workstation (see "Starting Entrust Master Control Command Shell" on page 347).
2. Log in to Entrust/Authority Master Control as a Master User (see "Logging in to Entrust/Authority Master Control" on page 349).
3. At the prompt, enter the following command:

   ca key config -alg *algorithm*

   where "algorithm" is either RSA-1024, RSA-2048, DSA-1024, or ECDSA

Note: ECDSA uses a default 192-bit curve. To change the curve type, see "To choose a named ECDSA elliptic curve" on page 388 and "To choose a custom ECDSA elliptic curve" on page 388.

You have now changed the CA key pair algorithm.

To choose a named ECDSA elliptic curve

1. Start Entrust Master Control Command Shell on your workstation (see "Starting Entrust Master Control Command Shell" on page 347).
2. Log in to Entrust/Authority Master Control as a Master User (see "Logging in to Entrust/Authority Master Control" on page 349).
3. At the prompt, enter

   ca key ec-params set-named *name of curve*

   where "name of curve" is a named curve listed in the X.962 standard

   For example:

   ```
   ca key ec-params set-named x962t239k1
   ```

   You have now chosen a named ECDSA elliptic curve.

To choose a custom ECDSA elliptic curve

1. In a text editor (such as Notepad) create a text file (using an .ini extension with the filename) and save the file to a known location on disk.

   The filename is arbitrary, but choose a memorable, intuitive filename (for example, customcurve001.ini).
2. Add a section in the file as follows:

   ```
   [Custom EllipticCurve Parameters]
   ```

```
ECDSA=ecParameters
ecParameters=curve value
```

where "curve value" is the hex-encoded DER encoding of the ECParameters defined in the X.962 standard

3 Save the file.

4 Start Entrust Master Control Command Shell on your workstation (see "Starting Entrust Master Control Command Shell" on page 347).

5 Log in to Entrust/Authority Master Control as a Master User (see "Logging in to Entrust/Authority Master Control" on page 349).

6 At the prompt, enter

```
ca key ec-params load filename
```

where "filename" is the name of the .ini file you created in Step 1

You have now chosen a custom ECDSA elliptic curve.

To write ECDSA curve data to a file

1 Start Entrust Master Control Command Shell on your workstation (see "Starting Entrust Master Control Command Shell" on page 347).

2 Log in to Entrust/Authority Master Control as a Master User (see "Logging in to Entrust/Authority Master Control" on page 349).

3 At the prompt, enter

```
ca key ec-params write <-force> filename
```

where "force" is an optional parameter indicating that an existing initialization file will be overwritten, and "filename" is the name of the initialization file in which to list the curve name and parameter details.

For example:

```
ca key ec-params write ecdsa.txt
```

You have now written the ECDSA curve data to a file.

Updating the CA key pair

The Certification Authority (CA) includes a signing key pair made up of a signing private key and a verification public key. The CA signing private key is used to sign all certificates, including all user certificates. Every time users interact within the Entrust/PKI system, the CA signature on the users' certificates is verified using the CA verification public key. This system of signing and verifying ensures the authenticity and integrity of every user certificate signed by the CA, and is what allows users to trust each other in the Entrust/PKI environment.

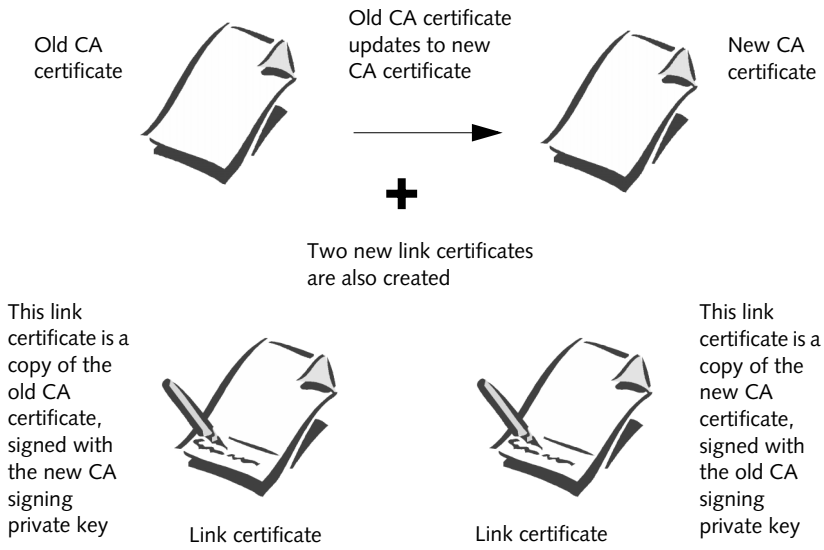
It is important to maintain the integrity of the CA key pair. Maintaining the integrity of the CA key pair might include, at times, updating the CA key pair.

You must update the CA key pair

- if you want to change the type or strength of the algorithm used by the CA to sign certificates
- if you want to change the lifetimes of the CA keys
- if the CA signing private key has been compromised or corrupted

Link certificates

When you update the CA keys, the new CA verification public key certificate is signed by the old CA signing private key, and the old CA verification public key certificate is signed by the new CA signing private key. This "cross-signing" of certificates produces two new certificates, called link certificates. These link certificates provide a connection between the old CA and the new CA. Link certificates allow users who trust the old CA signature to now trust the new CA signature.



Link certificates are created every time you update the CA key pair. They allow users to verify old CA signatures, no matter how many times the CA keys get updated. When a signature is verified, the most recent CA keys are examined first, and then the link certificates for each previous CA are verified until the CA key pair that originally created the signature is reached and verified.

Note: If you are interoperating with Microsoft CAPI-enabled applications, there are important considerations to be aware of when performing CA key update. For more information, check the Entrust/PKI 6.0 Release Notes for a listing of related white papers and where to find them.

Updating the CA key pair on software and hardware

You can update the CA key pair on software or on hardware (that is, where the source and destination are the same).

Attention: If you are supporting Entrust-Ready applications that are older than Release 5.0, it is extremely important that you not update the CA signing key pair, or that you upgrade these applications to Release 5.0 or newer before updating the CA signing key pair. Entrust-Ready applications that are older than Release 5.0 will cease to function if you update the CA signing key pair.

You can also update the CA signing private key from the Entrust/Authority database to a hardware device, or from a hardware device to the Entrust/Authority database (that is, where the source and destination are different).

Note: "Updating" the CA signing private key differs from "moving" it. Updating the CA signing private key creates a new, updated key. Moving the CA signing private key literally moves it from one location to another. For more information, see "Moving the CA key pair to a PKCS #11 v1 hardware device" on page 97.

To update the CA key pair

1. Start Entrust Master Control Command Shell on your workstation (see "Starting Entrust Master Control Command Shell" on page 347).
2. Log in to Entrust/Authority Master Control as a Master User (see "Logging in to Entrust/Authority Master Control" on page 349).
3. At the prompt, enter the following command:

```
ca key update
```

The following is an example of the information that will appear onscreen with an installed hardware device (if you do have not installed a hardware device the second option will not appear):

```
Select the destination for the new CA key
choose one of:
```

```
1. Software
2. CAHardware Corp. Ltd SN : 0D7F-777S-AEE5 SLOT : 4948476543
```

4 Enter "1" to update the key in software or "2" to update the keys in hardware.

If you update the key from hardware to software, no password is required.

If you update the keys from software to hardware, you are prompted for the device password (if required).

If you enter 2, for example, the following information appears on screen:

```
The services will be stopped and the CA key updated.
Do you wish to continue? (y/n) ? [y]
```

5 Press Enter to continue.

6 The following information appears, for example, if you are using a hardware device which requires the CA password:

```
The CA hardware password has been updated.
Please log in again.
```

7 Log in to Entrust/Authority Master Control as a Master User (see "Logging in to Entrust/Authority Master Control" on page 349).

You have now updated the CA key pair.

Attention: If you are logged in to Entrust/RA when you update the CA key pair on the command line, you must log out and log in again for the operation to take effect and to prevent errors in Entrust/RA.

Revoking the CA certificate

If a CA's signing key pair has been compromised, you should revoke the CA certificate. The "ca cert revoke" command revokes *all* related CA and link certificates.

Attention: Revoking the CA certificate is a very serious operation and should be carried out with extreme caution. If you decide to revoke the CA certificate, any user certificates signed by that CA before it was revoked will no longer be valid.

When the current CA certificate is revoked, the CA signing private key is automatically updated. You are not prompted for a location to store the new CA key: if the previous CA key had been stored in software, the new key will be

stored in software; if the previous key had been stored in a hardware device, the new key will be stored in the same hardware device.

Note: This command can now be executed only from Entrust Master Control Command Shell. In Entrust/PKI 5.0 it was possible to update the CA signing private key or revoke the CA certificate from Entrust/RA.

To revoke the CA certificate

1. Start Entrust Master Control Command Shell on your workstation (see "Starting Entrust Master Control Command Shell" on page 347).
2. Log in to Entrust/Authority Master Control as a Master User (see "Logging in to Entrust/Authority Master Control" on page 349).
3. At the prompt, enter the following command:

   `ca cert revoke` *`sernum`*

   where "sernum" is the serial number of the CA certificate you want to revoke. See Table 20 for optional parameters you can use with the "ca key revoke" command.

Table 20: Optional parameters for use with the "ca cert revoke" command

Use this optional parameter...	when you want to...
<-reason unspecified | cacompromise | superseded>	specify a reason for revoking the CA certificate. If you do not specify a reason, it defaults to "unspecified".
<-text *value*>	add a comment. If you do not add a comment, "-text" defaults to an empty string.
<-time *dd-mm<-yy<yy>>. hh:mm<:ss>>*	specify a time when the certificate was last known to be uncompromised. The time is only written to the CRL if the reason specified is "cacompromise". If you do not specify a value for "-time", it defaults to the issue time of the certificate. When specifying the time, you specify the day and month (dd-mm), a two-digit or four-digit year (yy<yy>), and, optionally, the hour (hh), minute (mm) and second (ss).

You have now revoked the CA certificate.

Recovering all users

If you have revoked the CA certificate (see "Revoking the CA certificate" on page 392), you may need to recover users.

If many users' certificates have been signed by the CA key pair you are revoking, these users' certificates will not be valid once the CA key pair has been revoked. Users must be issued new certificates signed by the current valid CA. To achieve this, a Master User must log in to Entrust/Authority Master Control and run the Recover All Users command.

This is a very serious operation. When you recover all users, you must distribute new activation codes to every single user. Carrying out this task requires finding a secure channel for distributing activation codes. Probably the most secure method of distribution is to the deliver the activation codes to each user in person.

See the "Maintaining the Certification Authority" chapter in *Using Entrust/PKI 6.0 on Windows* for more information on recovering all users.

To recover all users

1. Start Entrust Master Control Command Shell on your workstation (see "Starting Entrust Master Control Command Shell" on page 347).
2. Log in to Entrust/Authority Master Control as a Master User (see "Logging in to Entrust/Authority Master Control" on page 349).
3. At the prompt, enter the following command:

   ```
   ca keyrecover-all
   ```

   A message appears, advising you that this will put all users into the key-recovery state and asking you to confirm that you want to proceed.
4. Enter

   ```
   y
   ```

   After a short interval, a list of all users appears, along with their reference numbers and authorization codes. The prompt appears.

   You have now recovered all users.

About revocation lists

A certificate revocation list (CRL) is a signed, time-stamped certificate containing the serial numbers of public key certificates that have been revoked, and the reason for each revocation (if requested).

An authority revocation list (ARL) is a signed, time-stamped list of the serial numbers of CA public key certificates (including cross-certificates) that have been revoked.

A new CRL is created for approximately every 750 user certificates. A new ARL is created for every 100 CA certificates.

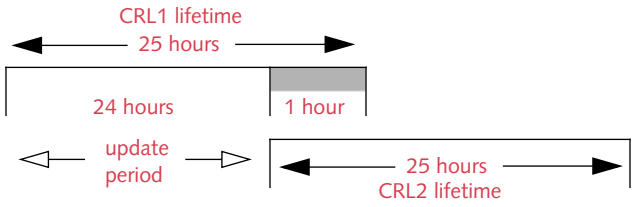
Note: If you choose to interoperate with Microsoft CryptoAPI-enabled applications, this will affect how CRLs are issued after CA key update and how the CRL distribution point appears in certificates. This choice is made when you configure Entrust/PKI, or if you change an [Advanced] setting in the entmgr.ini file prior to first-time initialization, or any time after first-time initialization with the entsh command "db set MsCompatibility 1". For more information regarding how the CRLs and CDPs are affected, check the Entrust/PKI 6.0 Release Notes for a listing of related white papers and where to find them.

Lifetimes and update periods

A revocation list's update period specifies how often it is updated by Entrust/PKI. The update period for ARLs is 6 hours. The update period for CRLs defaults to 24 hours, and can be set in the security policy to a value of 4 to 48 hours, in one hour increments.

The lifetime of the revocation list (RL) is not equal to the update period; it is one hour longer than the update period so it can be updated *before* it expires. For example, if the CRL update period is 24 hours, the CRL lifetime will be 25 hours, as shown in Figure 10

Figure 10: CRL lifetimes



When CRL1 expires (at the 25-hour mark), or if you "push" the CRL (force it to issue at a particular point in time),

- expired revocations are removed if necessary.
- the current list, CRL1, is written to the Directory

You may want to reissue CRLs if revocation has occurred or if certificate hold is canceled.

Exporting revocation lists

If you set up Entrust/PKI in autologin mode, you won't need to log in again to run the "rl" commands.

For more information about revocation lists, refer to *Administering Entrust/PKI 6.0 on Windows,* Table 21 lists the command-line interface commands relating to revocation lists.

Table 21: Command-line interface commands for revocation lists (RLs)

Command	Parameters	Description
rl export	[-pem | -bin] [-fn <fn>] (-crl <#>) | (-arl <#>) | (-cert <sernum>) | -combined	Return the given CRL or ARL as PEM or binary. The default format is PEM. If a filename is specified, save the data to the file. Otherwise, if the format is PEM, return it to the screen. If a certificate serial number is specified, return the RL if the certificate is on a CRL.
rl count	-arl | -crl	The number that appears is the number of the latest CRL or ARL.

To export a revocation list

1. Start Entrust Master Control Command Shell on your workstation.
2. Log in to Entrust Master Control Command Shell as a Master User.

   For more information about logging in as a Master User, refer to *Administering Entrust/PKI 6.0 on Windows*.
3. At the prompt, enter the "rl export" command.

   For example, the following command will save the combined CRL in PEM format to the file "CRL.pem":

```
rl export -pem -fn CRL.pem -combined
```

To access the number of certificate revocation lists

1. Start Entrust Master Control Command Shell on your workstation.
2. Log in to Entrust Master Control Command Shell as a Master User.

   For more information about logging in as a Master User, refer to *Administering Entrust/PKI 6.0 on Windows*.
3. At the prompt, enter

   ```
   rl count -crl
   ```

   The number that appears is the number of the latest CRL.

Note: The count of the CRL starts at 1. For CRLs, 0 is reserved for the combined CRL.

To access the number of authority revocation lists

1. Start Entrust Master Control Command Shell on your workstation.
2. Log in to Entrust Master Control Command Shell as a Master User.

   For more information about logging in as a Master User, refer to *Administering Entrust/PKI 6.0 on Windows*.
3. At the prompt, enter

   ```
   rl count -arl
   ```

   The number that appears is the number of the latest ARL.

Note: The count of the ARL starts at 1.

Example of how the "rl" commands could be used in a script

The following "entsh" script exports all CRLs to a file:

```
set fp [open "crl.out" "w"]
for {set i 1} {$i<=[rl count -crl]} {incr i} {
   puts $fp [rl export -crl $i]
}
close $fp
```

Suppressing audit logs issued for the "nearing expiration" reason

In a large system (for example, if you have one million users) you may want to suppress these audits to reduce the size of the audit trail. These audits occur when a CRL is nearing expiration and is re-issued and written to the Directory. This is a normal operating condition.

To suppress audit logs issued for the "nearing expiration" reason

1. Start Entrust Master Control Command Shell on your workstation.

   For more information about starting Entrust Master Control Command Shell, refer to *Administering Entrust/PKI 6.0 on Windows*.

2. Log in to Entrust Master Control Command Shell as a Master User.

   For more logging in as a Master User, refer to *Administering Entrust/PKI 6.0 on Windows*.

3. At the prompt, enter

   ```
   db set AuditCRLWrite 0
   ```

To enable audit logs issued for the "nearing expiration" reason

1. Start Entrust Master Control Command Shell on your workstation.
2. Log in to Entrust Master Control Command Shell as a Master User.

   For more logging in as a Master User, refer to *Administering Entrust/PKI 6.06.0 on Windows*.

3. At the prompt, enter

   ```
   db set AuditCRLWrite 1
   ```

Changing the ARL lifetime

Offline root CAs are used to protect the root CA key by physically isolating the PKI workstation and taking it off the network. The offline root CA does not run most of the time and the only users belonging to it are administrative users. The offline root CAs may include subordinate CAs or may be cross-certified with one or more CAs.

If your root CA is offline, you may want to set ARLs (and the complete CRL if it is used) to have a longer lifetime so that the offline root CA Master User does not have to access the offline root CA workstation as frequently.

To change the ARL lifetime before first-time initialization, use the advanced setting in the entmgr.ini file. For more information, see *Installing Entrust/PKI 6.0 on Windows*.

Attention: If you disable or shorten the lifetime of offline ARLs, it does not guarantee that clients will use the newly issued ARLs. If clients have cached the previous ARL, they may not pick up a new one until the previous one expires.

Table 22: Command-line interface commands for ARLs

Command	Parameters	Description
db get OffLineARL	[*LL*][-*YYMMDD*]	View the Offline ARL setting. "LL" is the lifetime in months from 1 to 12. The default is 0 (feature disabled), and leading zeros are not required. "-YYMMDD" is the optional next expire date in year/month/day order. Leading zeros are required. The revocation lists expire at 00:00 GMT on that day. If the optional "-YYMMDD" is not included, the setting is modified to include the current day. For example if today is March 4, 2000 and '7' is the OffLineARL value, it changes to 7-000304. As the ARLs are updated, the setting changes to show the next expiry date. The LL (months) setting never changes unless it is set manually by a Master User. The "-YYMMDD" value must fall within the lifetime months. If today is March 4, 2000 and you set the OfflineARL setting to 6-001204, you'll receive an error. If you pick the last day of the month the day may change depending on your lifetime. If your day is set to 31 and the next month only has 30 days, the day changes to 30. All ARLs and the optional combined CRL are issued when the OffLineARL setting is changed or disabled. When the OffLineARL setting is disabled, all the ARLs are reissued with their default 6 hour lifetime.
db set OffLineARL	[*LL*][-*YYMMDD*]	Set or reset the OffLineARL setting to a "LL" month lifetime, when the first lifetime expires on "-YYMMDD".
offlineARLs	none	Update offline ARLs before they expire.

Examples

In the following examples, the current day is March 4, 2000 and the combined CRL is enabled.

To display the OffLineARL setting

Enter the following command:

```
$ db get OffLineARL
```

In this example:

```
Name: OffLineARL
Value: 7-001010
```

They are due to expire at 00:00 GMT on October 10. The lifetime is 7 months.

To reset the OffLineARL setting

To reset the OffLineARL setting to a 7-month lifetime, when the first lifetime expires on March 10, 2000 00:00 GMT, enter the following example command:

```
$ db set OffLineARL 7-000310
```

The audits show the ARL (there is only one) and the combined CRL are issued.

```
[-07900 Authority revocation list (ARL) written to the
Directory.] : Log : : updated = 'Sat Mar 04 18:19:38 2000',
expires = 'Thu Mar 09 19:00:00 2000', distribution point =
'cn=CRL1,o=polar bear inc,c=US'

[-07943 Certificate revocation list (CRL) written to the
Directory.] : Log : : updated = 'Sat Mar 04 18:19:39 2000',
expires = 'Thu Mar 09 19:00:00 2000', distribution point =
'o=polar bear inc,c=US'
```

To update the offline ARLs

Enter the following command:

```
$ offlineARLs
```

In this example:

```
Offline ARLs updated successfully, next expiration is 'Mon Oct
09 20:00:00 2000'. Check audit logs for details.
```

The update was successful. The audits show that the ARL (only one exists) and the combined CRL are issued.

```
[-07900 Authority revocation list (ARL) written to the
Directory.] : Log : : updated = 'Sat Mar 04 18:22:03 2000',
expires = 'Mon Oct 09 20:00:00 2000', distribution point =
'cn=CRL1,o=polar bear inc,c=US'

[-07943 Certificate revocation list (CRL) written to the
Directory.] : Log : : updated = 'Sat Mar 04 18:22:03 2000',
expires = 'Mon Oct 09 20:00:00 2000', distribution point =
'o=polar bear inc,c=US'
```

To disable offline ARLs

Enter the following command:

```
$ db set OffLineARL 0
```

The audit trail shows that the ARL (only one exists) and the combined CRL are issued with their default update periods 6 hours and 24 hours respectively.

```
[-07900 Authority revocation list (ARL) written to the
Directory.] : Log : : updated = 'Sat Mar 04 18:56:52 2000',
expires = 'Sun Mar 05 01:56:52 2000', distribution point =
'cn=CRL1,o=polar bear inc,c=US'

[-07943 Certificate revocation list (CRL) written to the
Directory.] : Log : : updated = 'Sat Mar 04 18:56:52 2000',
expires = 'Sun Mar 05 19:56:52 2000', distribution point =
'o=polar bear inc,c=US'
```

Modifying behavior for certificate and revocation list extensions

This section describes advanced settings that modify Entrust/PKI default behavior for some certificate and revocation list extensions. These settings are useful if you have particular certificate extension requirements. If you do not have any special requirements, ignore this section and use the defaults. For more information on extensions, see the X.509 recommendation of the International Telecommunication Union or "Customizing certificates" on page 269.

You can do the following with this group of advanced settings:

- exclude the privateKeyUsage extension from user certificates
- exclude the privateKeyUsage extension from CA certificates
- exclude the basicConstraints extension from user certificates
- set the format of key identifier extensions
- exclude the reasonCode extension from revocation lists

Note: All of the advanced settings described in this section can be set before first-time initialization (see "Customizing the initialization files" in *Installing Entrust/PKI on Windows*). You may want to change them before first-time initialization because the CA certificate and the initial Security Officer certificates are created during first-time initialization.

Excluding the privateKeyUsage extension from user certificates

The privateKeyUsage extension is by default included in all user verification certificates. This extension indicates the date the private portion of a public-key pair expires. This setting can be used to not include this extension. However, if this extension is received during a PKIX-CMP request, it will be included in the certificate. This applies to all user certificates.

If you need to exclude the privateKeyUsage extension from a particular group of user certificates, you can use an advanced FCS setting to do this (see "Excluding certificate extensions" on page 328).

To display whether the privateKeyUsage extension is excluded from user certificates

1. Start Entrust Master Control Command Shell on your workstation (see "Starting Entrust Master Control Command Shell" on page 347).

2 Log in to Entrust/Authority Master Control as a Master User (see "Logging in to Entrust/Authority Master Control" on page 349).

3 At the prompt, enter

```
db get UserCertPrivateKeyUse
```

Detailed information about the advanced variable "UserCertPrivateKeyUse" appears. The variable name appears, as does its current value and a general description of the value. This description lists all possible values for the variable. If the privateKeyUsage extension is included in user certificates, the following would appear:

```
Name        : UserCertPrivateKeyUse
Value       :1
Description: Include the privateKeyUsage extension in user
certificates. '0' = No, '1' = Yes (default)
```

You have now displayed whether the privateKeyUsage extension is excluded from user certificates.

To change the setting for the privateKeyUsage extension in user certificates

1 Start Entrust Master Control Command Shell on your workstation (see "Starting Entrust Master Control Command Shell" on page 347).

2 Log in to Entrust/Authority Master Control as a Master User (see "Logging in to Entrust/Authority Master Control" on page 349).

3 At the prompt, enter

db set UserCertPrivateKeyUse *value*

where "value" indicates whether you want to use the privateKeyUsage extension in user certificates.

If you want to use the privateKeyUsage extension, for example, enter

```
db set UserCertPrivateKeyUse 1
```

The prompt appears.

4 Stop and restart the Entrust/Authority service (see "Starting and stopping the Entrust/Authority service" on page 352).

Your change takes effect.

You have now changed the setting for the privateKeyUsage extension in user certificates.

Excluding the privateKeyUsage extension from CA certificates

The privateKeyUsage extension is by default included in CA certificates and cross-certificates. The extension indicates the date the private portion of a public-key pair expires. This setting can be used to not include this extension. This applies to all CA certificates.

If you need to exclude the privateKeyUsage extension from only cross-certificates or the CA certificate, you can use an advanced FCS setting to do this (see "Excluding certificate extensions" on page 328).

To display whether the privateKeyUsage extension is excluded from CA certificates

1. Start Entrust Master Control Command Shell on your workstation (see "Starting Entrust Master Control Command Shell" on page 347).
2. Log in to Entrust/Authority Master Control as a Master User (see "Logging in to Entrust/Authority Master Control" on page 349).
3. At the prompt, enter

   ```
   db get CaCertPrivateKeyUse
   ```

   Detailed information about the advanced variable "CaCertPrivateKeyUse" appears. The variable name appears, as does its current value and a general description of the value. This description lists all possible values for the variable. If the privateKeyUsage extension is *not* included in CA certificates, the following would appear:

   ```
   Name        : CaCertPrivateKeyUse
   Value       :0
   Description: Include the privateKeyUsage extension in CA
   certificates. '0' = No, '1' = Yes (default)
   ```

   You have now displayed whether the privateKeyUsage extension is excluded from CA certificates.

To change the setting for the privateKeyUsage extension in CA certificates

1. Start Entrust Master Control Command Shell on your workstation (see "Starting Entrust Master Control Command Shell" on page 347).
2. Log in to Entrust/Authority Master Control as a Master User (see "Logging in to Entrust/Authority Master Control" on page 349).

3 At the prompt, enter

```
db set CaCertPrivateKeyUse value
```

where "value" indicates whether you want to use the privateKeyUsage extension in CA certificates.

If you want to use the privateKeyUsage extension, for example, enter

```
db set CaCertPrivateKeyUse 1
```

The prompt appears.

4 Stop and restart the Entrust/Authority service (see "Starting and stopping the Entrust/Authority service" on page 352).

Your change takes effect.

You have now changed the setting for the privateKeyUsage extension in CA certificates.

Excluding the basicConstraints extension from user certificates

The basicConstraints extension is by default included in all user certificates. The extension indicates if the certificate is a user certificate or a CA certificate. When the extension is not present, the certificate is also assumed to be a user certificate. This setting can be used to not include this extension. This applies to all user certificates.

If you need to exclude the basicConstraints extension from a particular group of user certificates, you can use an advanced FCS setting to do this (see "Excluding certificate extensions" on page 328).

To display whether the basicConstraints extension is excluded from user certificates

1 Start Entrust Master Control Command Shell on your workstation (see "Starting Entrust Master Control Command Shell" on page 347).

2 Log in to Entrust/Authority Master Control as a Master User (see "Logging in to Entrust/Authority Master Control" on page 349).

3 At the prompt, enter

```
db get UserCertBasicConst
```

Detailed information about the advanced variable "UserCertBasicConst" appears. The variable name appears, as does its current value and a general description of the value. This description lists all possible values for the variable.

If the basicConstraints extension is included in user certificates, the following would appear:

```
Name        : UserCertBasicConst
Value       :1
Description: Include the basicConstraints extension in user
certificates. '0' = No, '1' = Yes (default)
```

You have now displayed whether the basicConstraints extension is excluded from user certificates.

To change the setting for the basicConstraints extension in user certificates

1. Start Entrust Master Control Command Shell on your workstation (see "Starting Entrust Master Control Command Shell" on page 347).
2. Log in to Entrust/Authority Master Control as a Master User (see "Logging in to Entrust/Authority Master Control" on page 349).
3. At the prompt, enter

   `db set UserCertBasicConst` *`value`*

   where "value" indicates whether you want to use the basicConstraints extension in user certificates.

   If you don't want to use the basicConstraints extension, for example, enter

   ```
   db set UserCertBasicConst 0
   ```

   The prompt appears.
4. Stop and restart the Entrust/Authority service (see "Starting and stopping the Entrust/Authority service" on page 352).

   Your change takes effect.

   You have now changed the setting for the basicConstraints extension in user certificates.

Setting the format of key identifier extensions

Key identifiers specify what key pairs or public keys are associated with a certificate. For example, the authorityKeyIdentifier extension specifies the key pair that signed a certificate, and the subjectKeyidentifier extension identifies the public key in the certificate. These extensions are defined in the X.509 recommendation of the International Telecommunication Union.

Key identifiers should be derived from the public key, and they must be unique. Key identifiers are essentially hashes of bit strings or parts of such hashes combined with other data.

The Entrust default method of calculating the key identifiers is a 20-byte SHA-1 hash of the encoding of the subjectPublicKey including the tag, length, and number of unused bits.

Two other methods are specified in RFC 2459, the X.509 recommendation (*Internet X.509 Public Key Infrastructure Certificate and CRL Profile*):

- The keyIdentifier consists of the 160-bit SHA-1 hash of the value of the BIT STRING subjectPublicKey (excluding the tag, length, and number of unused bits).
- The keyIdentifier consists of a four-bit type field with the value 0100 followed by the least significant 60 bits of the SHA-1 hash of the value of the BIT STRING subjectPublicKey.

The format of these identifiers is important if you are interoperating with another CA. Your CA should use the same key identifier format as the other CA.

Attention: You must set this variable correctly *before* you start Entrust/Authority for the first time. Otherwise, your certificates will not have the proper key identifier format. Changing the format affects only new certificates. For more information, see the "Customizing the initialization files" appendix in *Installing Entrust/PKI 6.0 on Windows*.

To display how key identifier extensions are formatted

1. Start Entrust Master Control Command Shell on your workstation (see "Starting Entrust Master Control Command Shell" on page 347).
2. Log in to Entrust/Authority Master Control as a Master User (see "Logging in to Entrust/Authority Master Control" on page 349).
3. At the prompt, enter

   `db get KeyIdMode`

   Detailed information about the variable "KeyIdMode" appears: The variable name appears, as does its current value and a general description of the value. This description lists all possible values for the variable. If the current value of the variable is "0", the following information would appear:

```
Name        : KeyIdMode
Value       :0
Description: Key identifier format. '0' = Normal, '1' = RFC2459
compliant (160 bit for all), '2' = RFC2459 compliant (160 bit
for RSA,DSA,60 bit for EC), '3' = 60 bit for all.
```

You have now displayed how key identifier extensions are formatted.

To change how key identifier extensions are formatted

1. Start Entrust Master Control Command Shell on your workstation (see "Starting Entrust Master Control Command Shell" on page 347).
2. Log in to Entrust/Authority Master Control as a Master User (see "Logging in to Entrust/Authority Master Control" on page 349).
3. At the prompt, enter

   db set KeyIdMode *value*

   where "value" refers to the key identifier format you want to use.

   If you want to use format 2, for instance, enter

   ```
   db set KeyIdMode 2
   ```

   The prompt appears.
4. Stop and restart the Entrust/Authority service (see "Starting and stopping the Entrust/Authority service" on page 352).

   Your change takes effect.

   You have now changed how key identifier extensions are formatted.

Excluding the reasonCode extension from revocation lists

The reasonCode revocation list extension is associated with both certificate revocation lists (CRLs) and authority revocation lists (ARLs). The extension is defined in X.509 and is included with each revocation. This extension indicates why the revocation was done. By default Entrust/PKI includes the reasonCode extension for each revocation. One of the revocation reasons is "unspecified". Some applications may want to exclude the reasonCode extension if it is set to "unspecified" because excluding the revocation reason is effectively the same as not specifying it. Excluding the extension results in smaller CRLs and ARLs.

To display whether the reasonCode extension is excluded from revocation lists

1. Start Entrust Master Control Command Shell on your workstation (see "Starting Entrust Master Control Command Shell" on page 347).
2. Log in to Entrust/Authority Master Control as a Master User (see "Logging in to Entrust/Authority Master Control" on page 349).
3. At the prompt, enter

   db get CrlUseDefRevokeReason

Detailed information about the advanced variable "CrlUseDefRevokeReason" appears. The variable name appears, as does its current value and a general description of the value. This description lists all possible values for the variable. If the reasonCode *is* included in revocation lists, the following would appear:

```
Name        : CrlUseDefRevokeReason
Value       :1
Description: Include reasonCode in revocation lists if the
certificate is revoked with the 'unspecified reason'. '0' = No,
'1' = Yes (default).
```

You have now displayed whether the reasonCode extension is excluded from revocation lists.

To change the setting for the reasonCode extension in revocation lists

1. Start Entrust Master Control Command Shell on your workstation (see "Starting Entrust Master Control Command Shell" on page 347).
2. Log in to Entrust/Authority Master Control as a Master User (see "Logging in to Entrust/Authority Master Control" on page 349).
3. At the prompt, enter

   db set CrlUseDefRevokeReason *value*

   where "value" indicates whether you want to include a reason code in revocation lists.

   If you want to use a reason code in revocation lists, enter

   ```
   db set CrlUseDefRevokeReason 1
   ```

   If you *don't* want to use a reason code in revocation lists, enter

   ```
   db set CrlUseDefRevokeReason 0
   ```

   The prompt appears.
4. Stop and restart the Entrust/Authority service (see "Starting and stopping the Entrust/Authority service" on page 352).

   Your change takes effect.

   You have now changed the setting for the reasonCode extension in revocation lists.

Using a Combined CRL

Certificate revocation lists (CRLs) identify certificates that are no longer trusted or valid in the CA domain. Entrust/Authority maintains multiple CRLs at unique "distribution points" in the Directory. Each certificate contains a pointer to one or more CRL distribution points where applications can find the CRL for the certificate in question.

A Combined CRL includes all the revoked certificates that have not yet expired. At regular intervals, Entrust/Authority publishes the Combined CRL to the CA entry in the Directory. Some clients reference the Combined CRL in the Directory to check whether a certificate has been revoked. Entrust/Authority can also publish the Combined CRL to a file (see "To publish the Combined CRL to a file" on page 426).

You can use Entrust Master Control Command Shell to do the following:

- specify whether Entrust/Authority creates and maintains a Combined CRL
- specify the lifetime of a Combined CRL (if one exists)

For more information on using Combined CRLs, see "Using a Combined CRL" on page 77.

Turning the use of Combined CRL on and off

Once you have initialized Entrust/PKI, the *only way* you can turn the use of Combined CRL on and off is by using Entrust Master Control Command Shell. You can also use the command shell to tell you whether Entrust/Authority is currently maintaining a Combined CRL.

For information on how to turn the use of Combined CRL on and off before initialization, see "To turn on the Combined CRL before first-time initialization" on page 78.

To display whether Entrust/Authority maintains a Combined CRL

1. Start Entrust Master Control Command Shell on your workstation (see "Starting Entrust Master Control Command Shell" on page 347).
2. Log in to Entrust/Authority Master Control as a Master User (see "Logging in to Entrust/Authority Master Control" on page 349).
3. At the prompt, enter

   ```
   db get UseCombinedCRL
   ```

   Detailed information about the advanced variable "UseCombinedCRL" appears. The variable name appears, as does its current value and a general description of

the value. This description lists all possible values for the variable. If Entrust/Authority uses a Combined CRL, the following appears:

```
Name        : UseCombinedCRL
Value       :1
Description: Create a combined CRL in CA entry. '0' = No
(default), '1' = Yes
```

You have now displayed whether Entrust/Authority is maintaining a Combined CRL.

To turn the use of Combined CRL on or off

1. Start Entrust Master Control Command Shell on your workstation (see "Starting Entrust Master Control Command Shell" on page 347).
2. Log in to Entrust/Authority Master Control as a Master User (see "Logging in to Entrust/Authority Master Control" on page 349).
3. At the prompt, enter

   db set UseCombinedCRL *value*

   where "value" indicates whether or not you want Entrust/Authority to maintain a Combined CRL.

   If you want to use a Combined CRL, enter

   ```
   db set UseCombinedCRL 1
   ```

   If you *don't* want to use a Combined CRL, enter

   ```
   db set UseCombinedCRL 0
   ```

   The prompt appears.
4. Stop and restart the Entrust/Authority service (see "Starting and stopping the Entrust/Authority service" on page 352).

   Your change takes effect.

   You have now turned the use of Combined CRL on or off.

Changing the Combined CRL lifetime

If you are using a Combined CRL, you may want to change its lifetime. The default lifetime is 24 hours.

To display information on the lifetime of a Combined CRL

1. Start Entrust Master Control Command Shell on your workstation (see "Starting Entrust Master Control Command Shell" on page 347).

2 Log in to Entrust/Authority Master Control as a Master User (see "Logging in to Entrust/Authority Master Control" on page 349).

3 At the prompt, enter

```
db get CombinedCRLLifetime
```

Detailed information about the advanced variable "CombinedCRLLifetime" appears. The variable name appears, as does its current value and a general description of the value. This description lists all possible values for the variable. If Entrust/Authority uses a Combined CRL and its lifetime is 24 hours, the following appears:

```
Name        : CombinedCRLLifetime
Value       :24
Description: Combined CRL lifetime in hours, used when
UseCombinedCRL is set. Value from 4 to 336 hours or 0 to use
the CRL policy lifetime.
```

You have now displayed information on the lifetime of a Combined CRL.

To change the lifetime of a Combined CRL

1 Start Entrust Master Control Command Shell on your workstation (see "Starting Entrust Master Control Command Shell" on page 347).

2 Log in to Entrust/Authority Master Control as a Master User (see "Logging in to Entrust/Authority Master Control" on page 349).

3 At the prompt, enter

db set CombinedCRLLifetime *value*

where "value" indicates the desired lifetime of the Combined CRL in hours.

For instance, to set the lifetime of a Combined CRL to 96 hours, enter

```
db set CombinedCRLLifetime 96
```

The prompt appears.

4 Stop and restart the Entrust/Authority service (see "Starting and stopping the Entrust/Authority service" on page 352).

Your change takes effect.

You have now changed the lifetime of a Combined CRL.

Establishing compatibility with Entrust-Ready applications

When you install Entrust/Authority, by default it is configured to work with version 5 Entrust-Ready applications. If you later need to change this information, you *must* use the command shell to make the change.

To display which versions of Entrust-Ready applications Entrust/Authority is compatible with

1. Start Entrust Master Control Command Shell on your workstation (see "Starting Entrust Master Control Command Shell" on page 347).
2. Log in to Entrust/Authority Master Control as a Master User (see "Logging in to Entrust/Authority Master Control" on page 349).
3. At the prompt, enter

   ```
   db get Compatlevel
   ```

   Detailed information about the variable "CompatLevel" appears. The variable name appears, as does its current value and a general description of the value. This description lists all possible values for the variable. If Entrust/Authority is compatible only with version 5 Entrust-Ready applications, the following appears:

   ```
   Name        : CompatLevel
   Value       :5
   Description: Client Version Compatibility Level.  Permitted
   values are '3', '4', '5' supporting compatibility with 3.0, 4.0
   or only 5.0 clients and above.
   ```

Note: The compatLevel does not prevent you from encoding the DN with any UTF8 value.

You have now displayed which versions of Entrust-Ready applications Entrust/Authority is compatible with.

To change which versions of Entrust-Ready applications Entrust/Authority is compatible with

1. Start Entrust Master Control Command Shell on your workstation (see "Starting Entrust Master Control Command Shell" on page 347).
2. Log in to Entrust/Authority Master Control as a Master User (see "Logging in to Entrust/Authority Master Control" on page 349).

3 At the prompt, enter

```
db set CompatLevel value
```

where "value" indicates the oldest version of Entrust-Ready applications you want Entrust/Authority to be compatible with.

For instance, to set Entrust/Authority to work with versions 3, 4, and 5 of Entrust-Ready applications, enter

```
db set CompatLevel 3
```

The prompt appears.

4 Stop and restart the Entrust/Authority service (see "Starting and stopping the Entrust/Authority service" on page 352).

Your change takes effect.

You have now changed which versions of Entrust-Ready applications Entrust/Authority is compatible with.

Adding information to user entries

The type of information that can be stored in a user entry in the Directory depends on the object classes associated with that entry. Object classes may be either structural or auxiliary:

- Structural object classes define the type of entry and may restrict how the entry is named and where it is located in the Directory Information Tree (DIT). A Directory entry's structural object class is defined when the entry is created.
- Auxiliary object classes allow you to add attributes to user entries in the Directory.

When a user is activated in Entrust/PKI, the object class "entrustUser" is added to the user entry, as are any auxiliary object classes you have specified be added to the entry. The UserObjectClasses variable specifies which auxiliary object classes to add to user entries.

You can use Entrust Master Control Command Shell to either modify the list of object classes to add when users are enabled in Entrust/PKI or to view the list of object classes currently added when users are enabled. The object classes you add depend on what applications you will be using with the Directory and how your Directory is configured. Contact your Directory Administrator to determine what object classes to specify when a user is activated in Entrust/PKI.

Note: If you use Entrust Master Control Command Shell to add auxiliary object classes to user entries, it will affect only users enabled after you make that change.

To display the object classes associated with user entries

1. Start Entrust Master Control Command Shell on your workstation (see "Starting Entrust Master Control Command Shell" on page 347).
2. Log in to Entrust/Authority Master Control as a Master User (see "Logging in to Entrust/Authority Master Control" on page 349).
3. At the prompt, enter

```
db get UserObjectClasses
```

Detailed information about the advanced variable "UserObjectClasses" appears. The variable name appears, as does its current value and a general description of the value. This description lists all possible values for the variable. If user entries

currently include the object classes associated with a PKI user when a user is enabled, the following appears:

```
Name        : UserObjectClasses
Value       :entrustUser,pkiUser
Description: A comma delimited list of object classes which
should be added to user entries when enabled,or 'none' if no
additional object classes are desired.
```

You have now displayed the object classes associated with user entries.

To change the object classes associated with user entries

1. Start Entrust Master Control Command Shell on your workstation (see "Starting Entrust Master Control Command Shell" on page 347).
2. Log in to Entrust/Authority Master Control as a Master User (see "Logging in to Entrust/Authority Master Control" on page 349).
3. At the prompt, enter

   db set UserObjectClasses *value*

   where "value" is a comma-delimited list of the object classes you want to add.

   For instance, to add the object classes associated with Entrust users, enter

   ```
   db set UserObjectClasses pkiUser
   ```

   And if you do not want to add any object classes, enter

   ```
   db set UserObjectClasses none
   ```

   The prompt appears.
4. Stop and restart the Entrust/Authority service (see "Starting and stopping the Entrust/Authority service" on page 352).

   Your change takes effect.

   You have now changed the object classes associated with user entries.

Encoding DNs

For software applications to work together, data values must be encoded in such a way that other applications can "understand" them. In Entrust/PKI, distinguished name (DN) attributes must be encoded so that other products can decode them.

Entrust/PKI supports four character sets for encoding such attributes:

- PrintableString. This character set consists of a subset of the ASCII character set.
- TeletexString. This uses the ITU character set known as T.61. Entrust/PKI uses the Latin1 character set, ISO 8859-1, but tags it as TeletexString. The TeletexString option should be used only for backward compatibility and interoperability. It should *not* be used for certificates for new users.
- BMPString. This is an ASN.1 string that uses a subset of the multi-byte Universal Character Set (UCS) with two bytes per character. This encoding format is equivalent to Unicode.
- UTF8String. This character set is a transformation format of UCS. This is the preferred encoding format.

Entrust/PKI uses the following sequence of rules to determine which character set to use to encode these attributes:

1. If PrintableString is allowed and values in the attributes are within that character set, use PrintableString.
2. Otherwise, if TeletexString is allowed and values in the attributes are within the Latin1String character set, use TeletexString.
3. Otherwise, if UTF8String is allowed, use that character set.
4. Otherwise, if BMPString is allowed, use that character set.
5. Otherwise, return an error.

The DNEncoding variable specifies all the character sets that can be used to encode these attributes. The default value for this variable is "printable teletex utf8", which means that any of the character sets PrintableString, TeletexString, and UTF8String may be used to encode attribute values.

If you want to change how attributes are encoded, be sure you verify all your interoperability requirements first. If you don't, you risk losing interoperability with third-party products. For example, if you don't configure the encoding properly, a third-party product may not be able to decode the values properly.

For optimal compatibility with Release 3.0 or 4.0 software, use PrintableString or TeletexString.

To display information on how attributes are encoded

1. Start Entrust Master Control Command Shell on your workstation (see "Starting Entrust Master Control Command Shell" on page 347).
2. Log in to Entrust/Authority Master Control as a Master User (see "Logging in to Entrust/Authority Master Control" on page 349).
3. At the prompt, enter

   db get DNEncoding

   Detailed information about the variable "DNEncoding" appears. The variable name appears, as does its current value and a general description of the value. This description lists all possible values for the variable. If the current value of the variable is "UTF8", the following information would appear:

   ```
   Name        : DNEncoding
   Value       :utf8
   Description: DN Encoding.  Value is a space separated list of
   the values printable, teletex, bmp, and utf8.
   ```

   You have now displayed information on how attributes are encoded.

To change how attributes are encoded

1. Start Entrust Master Control Command Shell on your workstation (see "Starting Entrust Master Control Command Shell" on page 347).
2. Log in to Entrust/Authority Master Control as a Master User (see "Logging in to Entrust/Authority Master Control" on page 349).
3. At the prompt, enter

   db set DNEncoding *value*

   where "value" refers to the character set or sets you want to use to encode attributes.

   If you want to use the PrintableString character set, for instance, enter

   ```
   db set DNEncoding printable
   ```

   The prompt appears.
4. Stop and restart the Entrust/Authority service (see "Starting and stopping the Entrust/Authority service" on page 352).

   Your change takes effect.

   You have now changed how attributes are encoded.

Offline initialization of a subordinate CA

You must use Entrust Master Control Command Shell to perform offline initialization of a CA that is subordinate to a third-party or Entrust superior CA. This functionality is not available in the graphical user interface.

Offline initialization of a subordinate CA is a three-step process:

1. The subordinate CA asks the superior CA to generate a subordinate CA certificate. The subordinate CA sends this request as a message that conforms to a specific standard — Public-Key Cryptography Standards (PKCS) #10. PKCS #10 describes a syntax for certification requests. A PKCS #10 certification request *must* contain a distinguished name and a public key, and it *must* be signed by the entity that wants a certificate. This step is covered in "To ask a superior CA to generate a subordinate CA certificate" on page 419.
2. The superior CA transforms your request into an X.509 public-key certificate and sends that certificate, along with all the certificates needed to complete the path to the root CA, back to you. In Entrust/PKI, superior CAs send these certificates as messages conforming to the PKCS #7 standard. This standard defines a general syntax for messages that include cryptographic enhancements, such as digital envelopes or digital signatures. This step is covered in "To add an offline subordinate", starting at Step 3 on page 250.
3. The subordinate CA imports the PKCS #7 response into Entrust/Authority and completes the process of initializing the subordinate CA. This step is covered in "To import a PKCS #7 response into Entrust/Authority" on page 421.

This section describes Step 1 and Step 3. How Step 2 occurs is beyond the scope of the command shell.

For more information on CA hierarchies and adding subordinate CAs, see "Creating a hierarchy of CAs" on page 243.

In Entrust/PKI 5.1 or newer, you can specify how your CA certificate request is encoded: binary DER (Distinguished Encoding Rules), PEM (Privacy Enhanced Mail), or PEM with header (includes BEGIN and END lines). The choice of formats is provided for interoperability with PKIs from other vendors. Entrust/PKI can read files in any of these formats.

To ask a superior CA to generate a subordinate CA certificate

1. Start Entrust Master Control Command Shell at the subordinate CA.
2. Log in to Entrust Master Control Command Shell as a Master User.
3. At the prompt, enter

```
subordinate make-subcert-req
```

This command generates a file containing a PKCS #10 request for CA certificates, as well as a verification string for the file.

The following message appears:

```
How should the PKCS#10 file be encoded?
choose one of:
1. Binary
2. PEM
3. PEM with header
4. Quit
[1] >
```

Accept the default, Binary, or select a different format, for example, enter the following to encode the PKCS #10 request for the certificate in PEM format:

```
2
```

If you select option 3 (PEM with header), the saved file will include "-----BEGIN CERTIFICATE REQUEST-----" and "-----END CERTIFICATE REQUEST-----" lines.

The following message appears:

```
Enter name for PKCS#10 file.
Default is <C:\authdata\pkcs10.pem>:
```

4 Enter a path and filename for the PKCS #10 request.

The following message appears:

```
Include extensions from CA category of certificate
specification (y/n) ? [n]
```

5 To include the extensions defined in the default CA certificate type in the PKCS #10 extensionRequest attribute, enter

y

Otherwise, accept the default value [n].

If you require extensions from the third-party CA that aren't sent with the certificate request by default, choosing "yes" here provides one way to request them.

For more information about certificate extensions, see "Customizing certificates" on page 269.

If your third-party CA is not an Entrust CA, make a note to check the user documentation for your third-party CA to confirm that the certificates being sent back from the superior CA have the following extensions in them:

- subjectKeyIdentifier

  Note that the key Id format for the CA (as set by the advanced setting KeyIdMode) should match the format of the returned subjectKeyIdentifier.

- basicConstraints with "CA" field set to "true"

6 Entrust/Authority Master Control creates and saves the file. A verification string appears (for example, 8B:G6:7D:R8:33:2S:AC:9Y:3C:X0:09:R2:K5:65:4D:JW). Record this string.

This verification string is created based on an MD5 hash of the PKCS #10 request.

7 Send the PKCS #10 request to your superior CA (for instance, by ftp, e-mail, or disk).

8 Securely give the verification string you recorded in Step 5 to the administrator of the superior CA. The administrator needs this string to verify that the PKCS #10 request they receive is from you and is unaltered.

Attention: Give your verification string to the administrator securely and separately from your PKCS #10 request. Sending the request and its verification string together greatly increases the risk that undetectable tampering with the request will occur.

You have now asked a superior CA for a subordinate CA certificate.

To import a PKCS #7 response into Entrust/Authority

1 Start Entrust Master Control Command Shell at the subordinate CA.

For more information about starting Entrust Master Control Command Shell, see "To start Entrust Master Control Command Shell from the Windows taskbar" on page 347.

2 Log in to Entrust Master Control Command Shell as a Master User.

3 At the prompt, enter

```
subordinate read-subcert
```

When prompted, enter the path and filename of the PKCS #7 response from the superior CA.

Entrust/Authority reads the response and displays a verification string (for example, 5Y:3C:X0:09:R2:K5:65:4D:QW:8B:G6:7D:R8:33:2S:AC).

This verification string is created based on an MD5 hash of the PKCS #7 response.

Entrust/Authority will recognize the format of the file as binary DER encoded, PEM, or PEM with header.

Entrust Master Control Command Shell also displays a prompt asking you if the verification string is correct.

4 Contact the superior's administrator and confirm that your verification string matches the original string at the superior.

If the verification strings don't match, the response you received may be corrupt. Enter "no" at the prompt and ask the superior's administrator to send you a new PKCS #7 response.

If the verification strings match, enter "yes" at the prompt. Entrust/Authority creates profiles for the CA and First Officer.

You have now imported a PKCS #7 response to a request for a CA certificate into Entrust/Authority.

Note: CA initialization completes after you import the PKCS #7 response. Profiles are created and you are prompted to back up the Entrust/Authority database.

Creating and recovering Security Officers

Most tasks associated with Security Officers are normally done by in Entrust/RA. For example, administrative users with appropriate permissions use Entrust/RA to create Security Officers and to recover them.

But if you need to perform such tasks and cannot log in to Entrust/RA, using Entrust Master Control Command Shell to do them is your only option. Here are the Security Officer tasks you can use Entrust Master Control Command Shell to perform:

- "Creating a Security Officer" on page 423
- "Recovering a Security Officer" on page 424
- "Changing the number of Security Officer authorizations" on page 425
- "Reissuing a Security Officer's policy certificate" on page 427
- "Listing Security Officers" on page 428

Note: While you could do all of these tasks at the command line all the time, they are not intended to be used that way. Rather, they have been included in Entrust Master Control Command Shell as recovery commands.

Creating a Security Officer

Every installation of Entrust/PKI must have at least one Security Officer. Security Officers administer sensitive Entrust/PKI operations, and most organizations have two or more Security Officers. Organizations create Security Officers when they first set up Entrust/PKI, but others can be created later when needed.

Creating a Security Officer is a two-stage process:

1. Entrust/Authority Master Control generates a reference number and authorization code for a new Security Officer.
2. The Security Officer uses the reference number and authorization code to create a profile. The Security Officer may use Entrust/RA or Entrust/Entelligence to do this. Once the profile is created, the Security Officer can carry out Entrust/PKI tasks.

To create a Security Officer

1. Start Entrust Master Control Command Shell on your workstation (see "Starting Entrust Master Control Command Shell" on page 347).

2. Log in to Entrust/Authority Master Control as a Master User (see "Logging in to Entrust/Authority Master Control" on page 349).
3. At the prompt, enter

   officer create *dn*

   where "dn" is the distinguished name (DN) of the Security Officer you are creating. For instance, to create a Security Officer with the common name "Second Officer" who belongs to the organization "Your Company", in the country "US", you would enter

   ```
   officer create {cn=Second Officer,o=Your Company,c=US}
   ```

Note: If the Security Officer's common name contains any spaces, you *must* enclose the DN of the Security Officer you are creating in curly brackets or double quotation marks when you enter this command, because Entrust Master Control Command Shell commands use Tcl syntax. See the "Administering multiple users" chapter in *Using Entrust/PKI 6.0 on Windows* for an introduction to this syntax.

   The reference number and authorization code appear for the Security Officer you are creating, followed by the prompt.

4. Write down this reference number and authorization code, and keep the information secure. You need it for the next step.
5. *In a secure manner*, give this reference number and authorization code to the Security Officer. The Security Officer uses this information in Entrust/RA to create their profile (see the "Setting up the Certification Authority (CA)" chapter in *Using Entrust/PKI 6.0 on Windows* for how to use Entrust/RA to do this).

   You have now created a Security Officer.

Recovering a Security Officer

Whenever a Security Officer's profile is lost or becomes corrupt, that Security Officer must be *recovered*.

Recovering a Security Officer is a two-stage process:

1. Entrust/Authority Master Control generates a new reference number and authorization code for the Security Officer.
2. The Security Officer uses the new reference number and authorization code to recover the lost or corrupt profile.

To recover a Security Officer

1. Start Entrust Master Control Command Shell on your workstation (see "Starting Entrust Master Control Command Shell" on page 347).
2. Log in to Entrust/Authority Master Control as a Master User (see "Logging in to Entrust/Authority Master Control" on page 349).
3. At the prompt, enter

   `officer recover` *dn*

   where "dn" is the distinguished name of the Security Officer you are recovering. For instance, to recover a Security Officer with the common name "Second Officer" who belongs to the organization "Your Company", in the country "US", you would enter

   ```
   officer recover {cn=Second Officer,o=Your Company,c=US}
   ```

Note: If the Security Officer's common name contains any spaces, you *must* enclose the DN of the Security Officer you are recovering in curly brackets or double quotation marks when you enter this command, because Entrust Master Control Command Shell commands use Tcl syntax. See the "Administering multiple users" chapter in *Using Entrust/PKI 6.0 on Windows* for an introduction to this syntax.

   A reference number and authorization code appear for the Security Officer you are recovering.

4. Write down this reference number and authorization code, and keep the information secure. It's needed for the next step.
5. *In a secure manner*, give this reference number and authorization code to the Security Officer. The Security Officer uses this information to recover their profile with Entrust/RA (see the "Administering individual users" chapter in *Using Entrust/PKI 6.0 on Windows* for how to use Entrust/RA to do this).

   You have now recovered a Security Officer.

Changing the number of Security Officer authorizations

From time to time, you may need to change the number of Security Officer authorizations required for sensitive operations (see "Changing the number of authorizations for sensitive operations" on page 57 for a discussion of sensitive operations).

You might want to increase security by requiring two or more Security Officer authorizations for these tasks. By contrast, if a task requires multiple Security

Officer authorizations and only one Security Officer is available, you may need to change the number of Security Officer authorizations to one.

Note: While you can set the number of authorizations to one in both Entrust Master Control Command Shell and the corresponding graphical user interface, you can increase the number of authorizations only in the command shell.

Whenever Entrust Master Control Command Shell is running, you can find out how many Security Officer authorizations are required for sensitive operations.

To display the number of Security Officer authorizations required for sensitive operations

1. Start Entrust Master Control Command Shell on your workstation (see "Starting Entrust Master Control Command Shell" on page 347).
2. Log in to Entrust/Authority Master Control as a Master User (see "Logging in to Entrust/Authority Master Control" on page 349).
3. At the prompt, enter

   ```
   officer auths query
   ```

   Entrust Master Control Command Shell displays a number that tells you how many Security Officer authorizations are required for sensitive operations. For instance, if two Security Officers are needed to authorize sensitive operations, the following appears:

   ```
   2
   ```

   The prompt appears.

   You have now displayed the number of Security Officer authorizations required for sensitive operations.

To change the number of Security Officer authorizations required for sensitive operations

1. Start Entrust Master Control Command Shell on your workstation (see "Starting Entrust Master Control Command Shell" on page 347).
2. Log in to Entrust/Authority Master Control as a Master User (see "Logging in to Entrust/Authority Master Control" on page 349).
3. At the prompt, enter

   `officer auths set` *n*

   where "n" is the number of Security Officer authorizations required for sensitive operations.

For example, to require three Security Officer authorizations for sensitive operations, enter

```
officer auths set 3
```

The prompt appears.

You have now changed the number of Security Officer authorizations required for sensitive operations.

Reissuing a Security Officer's policy certificate

The [polcert_cliset Attributes] section of the certificate definition file defines about 30 different attributes that go into a Security Officer's certificate: attributes for password rules, for client settings, for application settings, and for business controls. Each attribute references one or more variables in the [Variables] section of the same file. These variables in turn can have different values, called "variable values". For a discussion of all the variable values, which are called "client-side settings", see "Overview of policy settings" on page 168.

Note: Before you initialize Entrust/PKI, the certificate definition file is called "initial.certspec". After you initialize Entrust/PKI, the default name of the file is "master.certspec". For more information on this file, see "About certificates" on page 271.

Settings for one or more variable values may inadvertently lock you out of Entrust/RA. For example, if the Security Officer policy certificate contains an incorrect value for the variable bcpolicyoids (acceptable policy OIDs), Security Officers will be unable to log in. The incorrect client-side setting must be reset, but the only administrative user with enough permissions to do that is locked out! When this happens, the only choice you have is to use the command shell to change client-side settings and reissue the policy certificate with the new settings.

You can use Entrust Master Control Command Shell to display client-side settings for a Security Officer and to change those settings.

To display client-side settings for a Security Officer

1. Start Entrust Master Control Command Shell on your workstation (see "Starting Entrust Master Control Command Shell" on page 347).
2. Log in to Entrust/Authority Master Control as a Master User (see "Logging in to Entrust/Authority Master Control" on page 349).
3. At the prompt, enter

```
officer client-setting query clientsetting
```

where "clientsetting" is the variable whose value you want to know. The variables you can display are those in the [polcert_cliset Attributes] section of the certificate definition file. For instance, if you wanted to know the value of the variable "signature_key_algs", you would enter

```
officer client-setting query signature_key_algs
```

The current value of the requested variable appears, followed by the prompt. For instance, if the current value of the variable "signature_key_algs" is RSA-1024, this would appear, followed by the prompt:

```
RSA-1024
```

To display the values of all client-side settings, enter

```
officer client-setting query all
```

You have now displayed a client-side setting for a Security Officer.

To change a client-side setting for a Security Officer

1. Start Entrust Master Control Command Shell on your workstation (see "Starting Entrust Master Control Command Shell" on page 347).
2. Log in to Entrust/Authority Master Control as a Master User (see "Logging in to Entrust/Authority Master Control" on page 349).
3. At the prompt, enter

```
officer client-setting set clientsetting n
```

where "clientsetting" is the variable whose value you want to set and "n" is the value you want it set to. The variables you can set are those in the [polcert_cliset Attributes] section of the certificate definition file. For instance, if you wanted to set the variable "signature_key_algs" to RSA-1024, you would enter

```
officer client-setting set signature_key_algs RSA-1024
```

The prompt appears.

You have now changed a client-side setting for a Security Officer.

Listing Security Officers

You can use Entrust Master Control Command Shell to list all the Security Officers in your CA. When you do this, the command shell returns the DNs of every Security Officer in your CA, including those who do not yet have profiles.

To list Security Officers

1. Start Entrust Master Control Command Shell on your workstation (see "Starting Entrust Master Control Command Shell" on page 347).
2. Log in to Entrust/Authority Master Control as a Master User (see "Logging in to Entrust/Authority Master Control" on page 349).
3. At the prompt, enter

   ```
   officer list
   ```

   A list of all Security Officers appears, showing the common name, organization, and country of each Security Officer in your CA. Then the prompt appears.

   You have now listed all the Security Officers in your CA.

Accessing certificate definition files

You can modify or create certificate types in Entrust/PKI by editing a file called a certificate definition file. Usually, Security Officers use Entrust/RA to perform these tasks. But if you need to perform such tasks and cannot log in to Entrust/RA, your only choice is to use Entrust Master Control Command Shell to do them.

Here are the tasks related to certificate definition files that you can perform in the command shell:

- writing certificate specifications in the Entrust/Authority database to a certificate definition file
- importing the specifications in the certificate definition file into the Entrust/Authority database

---

Note: You could perform these two tasks at the command line all the time. However, they are not intended to be used that way. Rather, they have been included in Entrust Master Control Command Shell as recovery commands.

To write the certificate specifications in the Entrust/Authority database to a certificate definition file

1. Start Entrust Master Control Command Shell on your workstation (see "Starting Entrust Master Control Command Shell" on page 347).
2. Log in to Entrust/Authority Master Control as a Master User (see "Logging in to Entrust/Authority Master Control" on page 349).
3. At the prompt, enter

   `fcs export` *`filename`*

   where "filename" refers to the name of the certificate definition file you are writing the specifications to.

   After a short interval, the prompt appears.

   You have now written the certificate specifications in the Entrust/Authority database to a certificate definition file.

Note: You can create and edit certificate definition files as often as you want. These files exist solely for the purpose of being edited. Each time you create a certificate definition file, it contains the current certificate specifications in the Entrust/Authority database.

To import the specifications in the certificate definition file into the Entrust/Authority database

1. Start Entrust Master Control Command Shell on your workstation (see "Starting Entrust Master Control Command Shell" on page 347).
2. Log in to Entrust/Authority Master Control as a Master User (see "Logging in to Entrust/Authority Master Control" on page 349).
3. At the prompt, enter

   `fcs import` *`filename`*

   where "filename" refers to the name of the certificate definition file containing the specifications you want to import into the Entrust/Authority database.

   You have now imported the specifications in the certificate definition file into the Entrust/Authority database.

Monitoring Entrust/PKI

Monitoring Entrust/PKI is an essential part of ensuring that the system is running as effectively and efficiently as possible. You can use Entrust Master Control Command Shell to perform the following monitoring tasks:

- "Checking audit logs" on page 432
- "Using performance counters" on page 434

Checking audit logs

Audit logs record all transactions that occur when you run Entrust/PKI, such as administrative user logins and logouts, user initializations, key updates, and any system failures. You can view audit logs to troubleshoot the system and monitor user activity.

While you usually view audit logs in Entrust/RA, you can also view them in Entrust Master Control Command Shell. Use the command shell if Entrust/RA isn't handy or you simply want a high-level summary of your audit logs.

You can use Entrust Master Control Command Shell to get either a summary of your audit logs or a complete description of every transaction:

- The summary of your audit logs tells you how many transactions occurred in a given time period. It also gives you summary information about transaction severity, showing you how many transactions of each level of severity occurred in a given time period. Transactions are classified as logs, events, or alarms (see "Description of audit log fields in the Event Viewer" on page 29).
- The complete description of every transaction provides detailed information on every transaction that occurred in a given time period. Among other things, this description gives you, for every transaction, an audit log number, the date and time the audit event occurred, a brief explanation of the event the audit log is related to, and the severity of the transaction.

See "About audit logs" on page 18 for more information on audit logs.

To get a summary of your audit logs

1. Start Entrust Master Control Command Shell on your workstation (see "Starting Entrust Master Control Command Shell" on page 347).
2. Log in to Entrust/Authority Master Control as a Master User (see "Logging in to Entrust/Authority Master Control" on page 349).

3 At the prompt, enter

```
db lsaud -s start time -e end time
```

where "start time" refers to the start time and "end time" refers to the end time for the date range your log summary will cover.

For instance, if you want to specify a start time of 15:55 and an end time of 16:05, you would enter

```
db lsaud -s 15:55 -e 16:05
```

A report such as the following appears, followed by the prompt:

```
Valid audits            62     Log            3
MAC failure             0     Event          59
Missing record          0     Alarm          0
Out of range            0
Out of sequence         0
Parse failure           0
File header bad         0

          TOTAL         62     TOTAL          62



Audit Status Report for Date Range
Thu Jul 22 15:55:00 1999 -- Thu Jul 22 14:05:00 1999
```

Note: Use the 24-hour clock for your start and end times. If you don't specify an end time, the command shell defaults to the current time. Use the hh:mm[:ss] time format. Similarly, if you don't specify a date, the command shell defaults to the current date. You may specify "yesterday" or a specific day such as "mon". You may also use the dd-mm[-yy[yy]] date format. If no year is specified, the command shell defaults to the current year.

You have now obtained a summary of your audit logs.

To view audit logs

1 Start Entrust Master Control Command Shell on your workstation (see "Starting Entrust Master Control Command Shell" on page 347).

2 Log in to Entrust/Authority Master Control as a Master User (see "Logging in to Entrust/Authority Master Control" on page 349).

3 At the prompt, enter

db lsaud -d -s *start time* **-e** *end time*

where "d" refers to obtaining a complete description, "start time" refers to the start time, and "end time" refers to the end time for the date range your log view will cover.

For instance, if you want a complete description of activity between a start time of 15:55 and an end time of 16:05, you would enter

```
db lsaud -d -s 15:55 -e 16:05
```

Information such as the following appears, followed by the prompt:

```
180 Thu Jul 22 16:08:22 1999 7966 Master user login Successful Master user
login.1 Event Master User Master1 0 Valid
```

You have now obtained detailed information on your audit logs.

Note: You may also write audit logs to a file using the -f <outfile> parameter.

Using performance counters

Entrust Master Control Command Shell has over 30 counters you can use to monitor Entrust/PKI. These performance counters can help you plan and manage system capacity for your Entrust/PKI.

The following entsh command lists all the command shell counters and their descriptions:

```
estat describe all
```

You can use Entrust Master Control Command Shell to give you information on the current value of all these counters.

Here are some guidelines on how to interpret this information:

- Transactions are counters or metrics associated with a time value (average time and maximum time).
- Gauges are counters or metrics that are not associated with a time value.
- The data presented refer either to the period since the services were last stopped or to the period since performance counters were last reset to zero.

Note: Entrust Master Control Command Shell allows you to change the number of processes assigned to four subsystems: the Entrust Secure Exchange Protocol (SEP) subsystem, the Key Generator subsystem, the Administration Service Handler (ASH) subsystem, and the Public-Key Infrastructure - Certificate Management Protocol (PKIX-CMP) subsystem. Entrust/Authority Master Control allows you to change the number of processes assigned to every subsystem but the Key Generator subsystem. You can make this change only in the command shell.

- If CPU utilization is low and client machines are often staying connected to Entrust/PKI servers for long periods of time without actually using them, values for the performance counters "cmp_cliwait" or "sep_cliwait" tend to be high. In such cases, increasing the number of processes for the affected subsystem should improve Entrust/PKI performance.
- If Entrust/PKI is running on multiple processors and the average value of the performance counter "keywait" is high (greater than 1,000 milliseconds), increasing the number of processes assigned to the Key Generator subsystem will allow you to exploit these processors and reduce the keywait delay.
- When you are trying to solve a performance problem with your Entrust/PKI, you may want to clear your performance counters to get a new troubleshooting baseline.

You can use Entrust Master Control Command Shell to perform the following tasks related to performance counters:

- obtain a description of all performance counters
- obtain a description of a specific performance counter
- display the current value of all counters
- display changes in the value of counters
- clear all counters

Note: You don't need to be logged in to the command shell to run commands related to performance counters.

To obtain a description of all performance counters

1. Start Entrust Master Control Command Shell on your workstation (see "Starting Entrust Master Control Command Shell" on page 347).
2. At the prompt, enter

   ```
   estat describe all
   ```

   All performance counters are listed, along with a description of each.

You have now obtained a description of all performance counters.

To obtain a description of a specific performance counter

1. Start Entrust Master Control Command Shell on your workstation (see "Starting Entrust Master Control Command Shell" on page 347).
2. At the prompt, enter

   estat describe *counter*

   where "counter" refers to the performance counter you want to get a description of. For example, if you wanted a description of the performance counter "keygen", you would enter

   `estat describe keygen`

   A description of the counter appears.

   You have now obtained a description of a specific performance counter.

To display performance counters

1. Start Entrust Master Control Command Shell on your workstation (see "Starting Entrust Master Control Command Shell" on page 347).
2. At the prompt, enter

   estat print

   Information about all Entrust/PKI performance counters appears.

   If you want information on all counters and gauges to appear regularly, enter

   estat print -cont=*n*

   where "n" refers to the desired interval, measured in seconds. For example, if you want information to appear every 5 seconds, you would enter

   `estat print -cont=5`

   Information about all Entrust/PKI performance counters appears regularly, at the specified interval.

   You have now displayed performance counters.

Note: To end the display of performance counters, press Ctrl-C. Pressing Ctrl-C will close the command shell window.

To display changes in performance counters

1. Start Entrust Master Control Command Shell on your workstation (see "Starting Entrust Master Control Command Shell" on page 347).
2. At the prompt, enter

   estat print -*counter* **-cont=***n*

   where "counter" is either "dsitrans" for transactions or "dsigauges" for gauges, and "n" refers to the desired interval.

   For example, if you want information on changes in transactions to appear every 10 seconds, enter

   ```
   estat print -dsitrans -cont=10
   ```

   You have now displayed changes in performance counters.

Note: To end the display of changes in performance counters, press Ctrl-C. Pressing Ctrl-C will close the command shell window.

To clear performance counters

1. Start Entrust Master Control Command Shell on your workstation (see "Starting Entrust Master Control Command Shell" on page 347).
2. At the prompt, enter

   estat clear

   The Entrust Master Control Command Shell prompt appears.

   You have now cleared performance counters.

Getting help on Entrust Master Control Command Shell

You can use Entrust Master Control Command Shell to get different types of help:

- information on command names and syntax
- descriptions of error codes
- version information

Note: You don't need to be logged in to Entrust Master Control Command Shell to perform any of the commands described in this section.

Getting help on command names and syntax

Getting help on command names and syntax is useful if you forget the name of a command you need or want to know more about a specific command. You can get help on all Entrust Master Control Command Shell commands, on a specific command set, or on a specific command.

There are 11 command sets in Entrust Master Control Command Shell:

- ca, for commands related to the CA
- db, for commands related to the database
- dsa, for commands related to the Directory
- estat, for commands related to performance counters
- fcs, for commands related to certificate definition files
- officer, for commands related to Security Officers
- policy, for commands related to policy
- rl, for commands related to revocation lists
- service, for commands related to the Entrust/Authority service
- subordinate, for commands related to subordinate CAs
- util, for commands related to error codes

To get help on all Entrust Master Control Command Shell commands

1. Start Entrust Master Control Command Shell on your workstation (see "Starting Entrust Master Control Command Shell" on page 347).

2 At the prompt, enter "help" or "?".

A list of all the Entrust Master Control Command Shell commands appears, along with a brief description of what each command does. The commands are listed in alphabetical order.

The prompt appears.

You have now obtained help on all Entrust Master Control Command Shell commands.

To get help on a specific command set

1 Start Entrust Master Control Command Shell on your workstation (see "Starting Entrust Master Control Command Shell" on page 347).

2 At the prompt, enter

```
help CommandSetName
```

where "CommandSetName" refers to the name of the command set. For example, to get help on the db command set, enter

```
help db
```

A list of all the commands in the command set appears, along with a brief description of what each command does.

The prompt appears.

You have now obtained help on a specific command set.

To get help on a specific command

1 Start Entrust Master Control Command Shell on your workstation (see "Starting Entrust Master Control Command Shell" on page 347).

2 At the prompt, enter

```
help CommandName
```

where "CommandName" refers to the name of the command. If the command consists of more than one word, all the words in the command must be in quotes. For example, to get help on the command "db get", enter

```
help "db get"
```

A description of the command appears, along with a summary of its syntax.

The prompt appears.

You have now obtained help on a specific command.

Obtaining a description of an error code

The error codes you see when you run Entrust Master Control Command Shell are Entrust error codes. You can use Entrust Master Control Command Shell to obtain a full description of a given error code. Most of the error codes you see when you run Entrust Master Control Command Shell are described in more detail in the appendix "Audit logs" on page 445.

To obtain a description of an error code

1. Start Entrust Master Control Command Shell on your workstation (see "Starting Entrust Master Control Command Shell" on page 347).
2. At the prompt, enter

   `util errinfo -`*`code`*

   where "code" refers to the error code you want information on. Be sure to include the hyphen before the error code.

   A message describing the error appears, followed by the prompt.

   For example, if you wanted to get a description of error code -7968, you would enter

   ```
   util errinfo -7968
   ```

   When you request information about error code -7968, the following message appears, followed by the prompt:

   ```
   (-7968) Invalid Master User login attempt.
   ```

   You have now obtained a description of an error code.

Displaying the command shell version

You can check which version of Entrust Master Control Command Shell you are running any time after you start the shell. You do not need to be logged in to Entrust/Authority Master Control to do this.

To find out which version of the command shell you are running

1. Start Entrust Master Control Command Shell on your workstation (see "Starting Entrust Master Control Command Shell" on page 347).

2 At the prompt, enter

```
version
```

A message similar to the following appears, specifying the shell's version number, the Entrust/Authority build number, and the Entrust/Authority build date and time:

```
Entrust/Authority Shell, version 6.0(287) @ April 20 2001
03:31:15
```

You have now found out which version of Entrust Master Control Command Shell you are running.

Entrust Master Control Command Shell security classes

Entrust Master Control Command Shell commands are grouped into four security classes:

- Class 0 commands cause a policy change or update that may be irreversible. These are the most restricted commands. For example, creating a Security Officer is a Class 0 command, as is changing the certificate hashing algorithm.
- Class 1 commands require access to the Entrust/Authority database, but they do not cause an irreversible change. Stopping the Entrust/Authority services and changing a Master User password are both Class 1 commands.
- Class 2 commands are non-harmful commands. These commands are also useful for automating tasks. Class 2 commands include starting the Entrust/Authority service and viewing audit logs.
- Class 3 commands are non-harmful commands, and they do not require access to the Entrust/Authority database. These commands are the least restricted. Examples of Class 3 commands include displaying performance counters and finding out which version of the command shell you are running.

The security class a command belongs to determines the number and type of authorizations needed to run the command:

- Class 0 commands require one additional Master User password if policy has been set to require multiple Master User authorizations. Otherwise, the command shell issues a warning and a yes/no prompt.
- Class 1 commands require that you be logged in to an active Entrust Master Control Command Shell session. Otherwise, you will be prompted to log in.
- Class 2 commands require either that autologin be enabled or that you be logged in to an active Entrust Master Control Command Shell session.
- Class 3 commands require no authorization.

Command summary

Many Entrust Master Control Command Shell commands have one or more options, and several require more than one authorization. The number and type of authorizations required depends on the security class the command belongs to (see "Entrust Master Control Command Shell security classes" on page 442).

The *Entrust Master Control Command Shell Quick Reference* lists all the tasks you can perform with Entrust Master Control Command Shell *except* those associated with advanced configuration variables. It also lists the command you enter to perform each task, and the security class the command belongs to.

Table 23 summarizes the command shell tasks associated with advanced configuration variables.

Table 23: Advanced configuration variables, commands, and security classes

Task	Command	Security class
Modifying default behavior for certificate and revocation list extensions		
• displaying or changing whether the privateKeyUsage extension is excluded from user certificates	db get/set UserCertPrivateKeyUse	1
• displaying or changing whether the privateKeyUsage extension is excluded from CA certificates	db get/set CaCertPrivateKeyUse	1
• displaying or changing whether the basicConstraints extension is excluded from user certificates	db get/set UserCertBasicConst	1
• displaying or changing how authorityKeyIdentifier and subjectKeyIdentifier key extensions are formatted	db get/set KeyIdMode	1
• displaying or changing whether the reasonCode extension is excluded from revocation lists	db get/set CrlUseDefRevokeReason	1
Using a Combined CRL		
• displaying or changing whether Entrust/Authority maintains a Combined CRL	db get/set UseCombinedCRL	1
• displaying or changing the lifetime of a Combined CRL	db get/set CombinedCRLLifetime	1

Table 23: Advanced configuration variables, commands, and security classes (continued)

Task	Command	Security class
Changing the ARL lifetime		
• displaying or changing lifetime and next expiry date for offline ARLs, if the root CA is offline	db get/set OffLineARL	1
Establishing compatibility with Entrust-Ready applications		
• displaying or changing which Entrust-Ready versions are compatible	db get/set CompatLevel	1
Adding information to user entries		
• displaying or changing information on the user's object classes	db get/set UserObjectClasses	1
Encoding DNs		
• displaying or changing how attributes are encoded	db get/set DNEncoding	1
Compatibility with with Microsoft CryptoAPI-enabled applications	db get/set MsCompatibility	1
Suppress the logs that are issued for the "nearing expiration" reason	db get/set AuditCRLWrite	1

Appendix A

Audit logs

The following table lists all Entrust audit logs and briefly explains what they mean. The following abbreviations appear throughout the audit logs:

- CRL—certificate revocation list
- ARL—authority revocation list
- DN—distinguished name

An audit record that refers to a "user" applies to all Entrust users.

An audit record may have one of three severity ratings. In order of increasing importance, they are: Log, Event, and ALARM.

Audit	Severity	Audit log message and information
7641	ALARM	License low. The percentage of remaining licenses available is less than the threshold set in the "EntLicenseThreshold" setting in the entmgr.ini file (see "Customizing the entmgr.ini file" on page 527).
7642	Event	PKI status. A "heartbeat" is output at regular intervals (hourly, by default). The heartbeat includes information on the version of Entrust/PKI that is installed, the platform on which it is running, the type of CA hardware used (if applicable), which database is installed, and information on Enterprise and Web user licenses. The heartbeat interval is set with the EntDailyAuditNotBefore, EntDailyAuditNotAfter, and EntDailyAuditPeriod settings in the entmgr.ini file (see "Customizing the entmgr.ini file" on page 527).

Audit	Severity	Audit log message and information
7643	Event	PKIX certificate request with long certificate lifetime allowed. A certificate request contains a certificate lifetime that exceeds the security policy limit and the FCS entry for the certificate type contains the "noTruncateUserLifetime" flag. Without the FCS flag, the lifetime is truncated to the maximum allowed lifetime.
7644	Event	MasterKey table upgraded. The internal format of a database table has been upgraded from one version of Entrust/Authority to another.
7645	Event	Hardware-based database protection turned off. A Master User disabled hardware-based database protection.
7646	Event	Hardware-based database protection turned on. A Master User enabled hardware-based database protection. The database (and its backups) now cannot be unlocked without the hardware device.
7647	Event	Certificate removed from hold. The certificate suspension has been canceled. The certificate is trusted.
7649	Event	CA hardware password set. The CA hardware password has been set in the command-line interface or from Entrust/Authority Master Control.
7651	Event	Cross-certificate signed by an external Certification Authority (CA) successfully removed from the Directory. The cross-certificate signed by another CA for this CA has been successfully removed from the crossCertificatePairs attribute in the CA entry.
7652	Event	Cross-certificate signed by an external Certification Authority (CA) successfully written to the Directory. The cross-certificate signed by another CA for this CA has been successfully written to the crossCertificatePairs attribute in the CA entry.
7653	Event	Cross-certificate signed by an external Certification Authority (CA) imported successfully. A Security Officer or Administrator with appropriate privileges has imported a cross-certificate signed by another CA for this CA into the database.
7654	ALARM	An invalid entBackupAlarmTime value specified in entmgr.ini. The value for the backup interval needs to be in one of the following formats: HH:MM (number of hours and minutes) or an integer (number of days).

Audit	Severity	Audit log message and information
7655	ALARM	External backup not detected within configured time. A backup did not occur during the interval specified in entBackupAlarmTime.
7656	Event	External backup detected. A backup did occur during the interval specified in entBackupAlarmTime.
7657	Event	User attribute certificate reissued. Attribute certificates can be reissued automatically by Entrust/Authority. Otherwise, event 7661 is logged.
7658	Event	User attribute certificate renamed. An Entrust/RA administrative user with sufficient privileges has renamed a user attribute certificate. This event only occurs if the user's DN is changed.
7659	Event	All attribute certificates for user deleted. An Entrust/RA administrative user with sufficient privileges has deleted all attribute certificates for the specified user. This event only occurs when all attribute certificates are removed as a single operation, for example, when the user is archived or when an export operation is completed.
7660	Event	User attribute certificate deleted. An Entrust/RA administrative user with sufficient privileges has deleted a user attribute certificate.
7661	Event	User attribute certificate changed. An Entrust/RA administrative user with sufficient privileges has modified a user attribute certificate.
7662	Event	User attribute certificate created. An Entrust/RA administrative user with sufficient privileges has created a new user attribute certificate.
7663	ALARM	A cross-certificate was issued but cannot be stored locally. The cross certificate issued cannot be stored in the database. Consequently, you cannot manage the certificate and it cannot be revoked. Manual effort must be made to ensure that the certificate is not posted to any Directory, and the CA that received the cross-certificate does not use it, post it, or distribute it to end users.
7664	Event	Successfully automatically revoked a cross-certificate. A cross certificate was issued and there was a subsequent error. The cross certificate was revoked. If the error 7663 is also present, disregard it. The certificate was successfully revoked and poses no security threat.

Audit	Severity	Audit log message and information
7665	ALARM	Attempt to automatically revoke a cross-certificate failed. The cross certificate that was issued may be stored in the database (unless the audit log 7663 is also present). It may be possible for an administrative user with sufficient privileges to revoke the cross certificate.
7666	Event	Certificate Definition specified the entrustUserRole attribute in the subjectDirectoryAttributes extension. The Certificate Definition defined the subjectDirectoryAttributes extension which included the entrustUserRole attribute. Entrust/Authority will replace this attribute with its own value.
7667	ALARM	CMP cross-certification failed. A PKIX-CMP Cross-Certification operation failed. Look in the manager.log file for more information.
7668	Event	CMP user certificate request failed. Entrust/Entelligence 5.0 or later using PKIX-CMP, or another PKIX-CMP user, has requested another certificate and a failure has occurred. The manager.log file will contain any errors associated with this failure. The end user's DN is included in the audit log.
7669	Event	An administrator requested that all changed CRLs be issued to the Directory. An administrative user with sufficient privileges has issued all changed CRLs from Entrust/RA. The number of CRLs issued is included with the audit log. For each CRL issued, audit log 7943 will precede this audit log.
7670	Event	Automatic login has been disabled. A Master User has disabled automatic services login. For the services to start, a Master User must enter their password.
7671	Event	Automatic login has been enabled. A Master User has enabled automatic services login to allow the services to be started unattended as part of system startup.
7673	ALARM	Directory operation—rename entry failed. An administrative user with sufficient privileges failed to successfully rename a Directory entry.
7674	Event	Directory operation—successfully renamed Directory entry. An administrative user with sufficient privileges successfully renamed a Directory entry.

Audit	Severity	Audit log message and information
7675	Event	Directory operation—Directory Administrator password changed in the Directory. An administrative user successfully changed their password.
7676	Event	Directory operation—Certification Authority (CA) Directory password changed in the Directory. The CA Directory password was successfully changed.
7677	Event	Client attempted to use an unsupported protocol. A client connected to Entrust/Authority and attempted to use a known, but unsupported protocol to make a request. The only protocol that falls into this category is the SEP protocol, which is disabled for subordinate CAs. The DN of the requestor is included with this audit log, so that they can be contacted, and her client configuration can be updated. The requestor may be using a pre-5.0 release of the Entrust client software that only supports SEP. The requestor may be configured to use SEP instead of CMP.
7678	ALARM	Administrator trying to access ASH does not have a local verification certificate. The administrative user attempting access does not have the correct credentials and is not recognized by this CA. The DN of the administrative user is included with the audit log. Contact the administrative user whose DN is in the audit log. If that administrative user has not been trying to log in, this audit log may indicate a break-in attempt.
7679	ALARM	Cross-certificate issued by an external Certification Authority (CA) is expired. See audit log 7864. If this audit log occurs, users at the CA who issued the certificate will not trust users at your CA.
7680	ALARM	Cross-certificate issued at this Certification Authority (CA) is expired. See audit log 7864. If this audit log occurs users at your CA will not trust users at the CA for whom the certificate is issued.
7681	ALARM	Subordinate Certification Authority (CA) certificate is expired. See audit log 7864. If this audit log occurs the subordinate CA cannot operate.
7682	Event	Cross-certificate issued by an external Certification Authority (CA) is nearing the end of its lifetime. See audit log 7864.
7683	Event	Cross-certificate issued at this Certification Authority (CA) is nearing the end of its lifetime. See audit log 7864.

Audit	Severity	Audit log message and information
7684	ALARM	Subordinate Certification Authority (CA) certificate is nearing the end of its lifetime. This audit log occurs when there is less than 28, 21, 14, 7 and 1 day left in the certificate lifetime. The certificate issuer distinguished name, subject distinguished name and expiration date are included with the audit log. This audit log is to warn an administrative user that the certificate should be updated.
7685	ALARM	Directory Browser—delete Directory entry failed. An attempt to delete a Directory entry has failed.
7686	Event	Directory Browser—successfully deleted a Directory entry. An attempt to delete a Directory entry has succeeded.
7687	ALARM	Directory Browser—add Directory entry failed. An attempt to add a new Directory entry has failed.
7688	Event	Directory Browser—successfully added a Directory entry. A new Directory entry has been added using the Directory Browser tool.
7689	ALARM	Directory Browser—delete Directory attribute value failed. An attempt to delete a Directory attribute value from an existing Directory attribute has failed.
7690	Event	Directory Browser—successfully deleted a Directory attribute value. A Directory attribute value has been deleted from an existing Directory attribute.
7691	ALARM	Directory Browser—delete Directory attribute failed. An attempt to delete a Directory attribute from an existing Directory entry has failed.
7692	Event	Directory Browser—successfully deleted a Directory attribute. A Directory attribute has been deleted from an existing Directory entry.
7693	ALARM	Directory Browser—replace Directory attribute value failed. An attempt to replace an existing Directory attribute value has failed.
7694	Event	Directory Browser—successfully replaced a Directory attribute. A Directory attribute value has been replaced with a new attribute.
7695	ALARM	Directory Browser—add Directory attribute failed. An attempt to add an attribute to an existing Directory entry has failed.

Audit	Severity	Audit log message and information
7696	Event	Directory Browser—successfully added a Directory attribute. A Directory attribute has been added to an existing Directory entry.
7697	ALARM	Certification Authority (CA) certificate is revoked. A subordinate CA checks its certificate (issued by its superior CA) and determines that it has been revoked.
7698	ALARM	Error checking Certification Authority (CA) certificate for revocation. The certificate may be revoked. A subordinate CA is unable to determine the revocation status of its certificate. This occurs when one or more of the superior CAs fail to publish their revocation lists or if their Directories are unavailable. The subordinate CA will assume that its certificate has not been revoked. However, the Security Officers of the subordinate CA should contact the superior CAs and ascertain that neither the subordinate nor any of the superior CAs have been revoked.
7699	Event	Subordinate Certification Authority (CA) canceled. An administrative user has canceled a subordinate CA that was previously initialized. The authorization code that was created for it is deleted, and the subordinate cannot be installed; however, the subordinate CA can be re-added.
7700	Event	Subordinate Certification Authority (CA) revoked. An administrative user has revoked the certificate of a subordinate CA. The subordinate CA and all certificates it has issued are invalidated, including user certificates, cross-certificates, and subordinate CA certificates.
7701	Event	Subordinate Certification Authority (CA) initialized. An administrative user has added a subordinate CA and an authorization code has been generated. This authorization code is used when the subordinate CA is installed.
7702	Event	A message confirming the new Certification Authority (CA) certificate was sent to the superior CA. The CA has accepted the certificate that was issued to it by the superior CA. This is the CA certificate used for issuing certificates and CRLs.
7703	Event	The new Certification Authority (CA) certificate was received in the response from the superior CA. The CA has received a response to a certificate request that was sent earlier. The request was accepted by the superior CA and a certificate was included in the response. Were the request denied, the subordinate CA would have received an error.

Audit	Severity	Audit log message and information
7704	Event	A certificate request was sent to the superior Certification Authority (CA). The CA has requested a CA certificate from the superior CA. This is done by a CA in a hierarchy when it is subordinate to another CA. This request is sent during installation of the subordinate when the subordinate is configured to communicate with its superior using the online protocol (PKIX-CMP).
7705	Log	A new Certification Authority (CA) key pair has been generated. The CA has generated a new key pair for its own use. This keypair is used for signing and verifying certificates and CRLs. This is done during the installation of a new CA or when a CA performs a CA key rollover.
7706	Event	Offline subordinate Certification Authority (CA) setup is complete. A Master User has imported your CA certificate from a PKCS #7 file received from a superior CA. Your CA is a subordinate CA and is now fully set up. Users can be created.
7707	Event	Offline subordinate Certification Authority (CA) certificate request created. Your CA is a subordinate CA and has created a certificate request to ask the superior CA to create your CA certificate. This request is a file in the PKCS #10 format and is passed to the superior CA offline. The offline certificate request file is created by a Master User using the entsh command-line application.
7708	ALARM	Combined certificate revocation list (CRL) size larger than specified threshold. The Combined CRL has stopped being published to the Directory since it is larger than the specified threshold in entmgr.ini. The threshold purpose is to protect directories against large entries. If your Directory can handle the size of the Combined CRL you can re-enable publishing to the Directory by increasing the size of the threshold or removing the threshold entry from entmgr.ini. The Combined CRL is always maintained in the Entrust/Authority database regardless of its size and of the threshold value.
7709	Event	Certification Authority (CA) Certificate generated for subordinate CA. The CA has generated a certificate for the subordinate CA. This is done when the subordinate CA is being installed.
7710	Log	User import from another Certification Authority (CA) has completed. A user who has been imported from another CA has logged in and completed the process. The user's current DN and the user's DN at the other CA are included in the audit log. An administrative user at the user's other CA can now complete the export operation and archive the user if they want.

Audit	Severity	Audit log message and information
7711	ALARM	Audit log integrity check failed. Audit log integrity check has failed. The audit log integrity is checked each time an integrity check is done. Such checks are done periodically or can be done manually with the entsh command-line interface.
7712	Log	Audit log integrity check passed. Audit log integrity check has passed. The audit log integrity is checked each time an integrity check is done. Such checks are done periodically or can be done manually with the entsh command-line interface.
7713	Log	The Certification Authority (CA) key is nearing its expiry. The signing private key of your subordinate CA is nearing expiry. This audit log is generated if the age of the signing private key exceeds 70% of its lifetime. You should install a newer release of Entrust/Authority and roll over the keys of your subordinate CA.
7714	ALARM	The Certification Authority (CA) key has expired. The CA signing private key of a subordinate CA has expired. You need to obtain a new key. Contact your superior CA and request activation codes. Then create new keys in Entrust/Authority Master Control.
7715	Log	Key update requested by client. A key update request has been received by a client using the PKIX-CMP protocol. If the client is an Entrust client, the reason for the key update is included with the audit log.
7716	ALARM	Restore First Officer failed. Unable to restore the First Officer entry and certificate to the Directory after a Restore To Directory operation. Restoring the First Officer may fail for the following reasons: • The First Officer DN entry is missing from the policy section in the entmgr.ini file. • The DN listed for the First Officer in the entmgr.ini file is not a valid DN or does not correspond to any user in the Entrust/Authority database. • The First Officer entry is not present in the Directory and creation of the entry failed. This may occur if the parent of the First Officer entry does not exist or some other Directory problem has occurred. • The First Officer certificate could not be recovered. The preceding log file entries in the manager.log should clarify the cause of the failure.

Audit	Severity	Audit log message and information
7717	Event	First Officer Restored. The First Officer entry and certificate have been restored to the Directory after a Restore To Directory operation.
7718	Log	Certification Authority (CA) certificate written to the Directory. The CA certificate has been written to the Directory. This audit log occurs when the CA certificate has been successfully written to the Directory during a Restore to Directory operation.
7719	Event	Authority advanced setting changed. An advanced setting has been changed by a Master User using the entsh application. The audit log includes the name, description and the old and new value for the setting.
7721	Event	File deletion after user retrieval failed. An error occurred removing user archive data from permanent storage after a user was salvaged. This is not a serious problem, but it may indicate invalid file permissions.
7724	ALARM	Error occurred revoking a Certification Authority (CA) key. This audit log is generated if an error occurs revoking the CA key. The key identifier (in hexadecimal form) is included with this audit log.
7725	Event	A Certification Authority (CA) key has been revoked. This audit log is generated when a CA key is revoked. CA key revocation is triggered from Entrust/RA. The key identifier (in hexadecimal form) is included with this audit log.
7726	ALARM	Certification Authority (CA) key update failed. This log is generated when an attempt to roll over CA keys fails.
7727	Event	The CA private signing private key has been updated. A new Certification Authority (CA) signing private key pair was created. This audit log is generated when the CA keys roll over. This can be triggered three different ways: 1. manually from Entrust/RA 2. automatically if the private CA signing private key is nearing its expiry 3. automatically if the current CA signing private key is revoked

Audit	Severity	Audit log message and information
7728	Event	An optional extension was omitted from the certificate. This event is generated during certificate creation for a user who is missing information required to generate one or more optional extensions. This situation most commonly arises when an administrative user opts not to enter optional information on the Certificate Information property page of the New User dialog box.
7729	Event	User extension variable values changed. This audit log indicates that an administrative user has changed one or more values included in a user's certificate information.
7730	Event	Policy Certificate removed from Directory. A policy certificate was removed from the Directory. A policy certificate will only be removed from the Directory if its DN is changed.
7731	Event	Policy Certificate written to Directory. A policy certificate was written to the Directory. The audit log includes the DN, identifier, name, and FCS Certificate.
7732	Event	Policy Certificated updated. A policy certificate was updated because it was about to expire, the certificates were restored to Directory or the CA key was updated. The audit log includes the DN, identifier, name and FCS certificate type.
7733	Event	Policy Certificate changed. A policy certificate was changed by an administrative user. The audit log will include the DN, identifier, name and FCS Certificate. The attributes and values are included with the audit log.
7734	Event	Policy Certificate created. A policy certificate was created. The audit log will include the DN, identifier, name and FCS Certificate. The attributes and values are included with the audit log.
7735	Event	Default policy certificates created. The default policy certificates have been created during first-time initialization.
7740	Event	External Certification Authority (CA) user public key deleted. An administrative user has deleted a CA user public key that was previously imported. The DN of the CA whose key is included with the audit log.
7741	Event	External Certification Authority (CA) user public key imported. An administrative user has imported the CA user public key from another CA. The audit log includes the DN of the other CA.

Audit	Severity	Audit log message and information
7742	Event	Certification Authority (CA) user public key exported. The CA user public key has been exported to a file by an administrative user. This file is transferred to another CA to allow users to be moved between the CAs. This is only necessary if the CAs do not have a trust link through either cross-certification or strict hierarchy. The hash of the CA user public key file, the serial number of the CA user certificate, and the expiration date of the CA user encryption public key is included with the audit log.
7743	Event	User salvaged from permanent storage. An administrative user has salvaged a user from permanent storage. The user is now an Entrust user in the disabled state. The user's DN is included with the audit log.
7744	Event	User archived to permanent storage. An administrative user has archived a user to permanent storage. The user will no longer appear as an Entrust user. The user's DN is included with the audit log.
7745	Event	User imported and set for key recover. A user has been imported by an administrative user and set to the key recover state. The DN of the user, the user's old DN and old CA DN are included with the audit log.
7746	Event	User imported and set active. A user has been imported by an administrative user and set to the active state. The DN of the user, the user's old DN and old CA DN are included with the audit log.
7747	Event	User export canceled. An administrative user has canceled the export of a user either in the export or export hold state. The DN of the user is included with the audit log.
7748	Event	User exported to another Certification Authority (CA). The export of a user from one CA to another has been completed by an administrative user. The user is no longer supported at this CA and is now in the export state. The DN of the user is included with the audit log.
7749	Event	User put in export hold state. The first stage of exporting a user to another CA has been done by an administrative user. The export hold state means the user can still operate at the current CA until the new CA imports the user. The DN of the user is included with the audit log.

Audit	Severity	Audit log message and information
7750	Event	Searchbase changed. A searchbase has been changed. The audit log will include the name, DN and identifier of the searchbase and will also indicate if the searchbase is an end user or administrator searchbase.
7751	Event	Searchbase created. A searchbase has been created. The audit log will include the name, DN and identifier of the new searchbase and will also indicate if the searchbase is an end user or administrator searchbase.
7752	Event	Searchbase deleted. A searchbase has been deleted. The audit log will include the name and identifier of the deleted searchbase.
7753	Event	User templates modified. The user templates information has been modified.
7754	Event	User templates initialized. The user templates information has been initialized from the templates.ini file during first-time initialization.
7755	Event	User authorization code lifetime changed. The users's custom authorization code lifetime has been changed. This cannot be done with the Entrust/RA user interface. It is only available through the Entrust/RA Toolkit. The user's DN old lifetime value, and new lifetime value are included with the audit log. The custom authorization lifetime is intended for use by the automated user registration application.
7756	Event	A user's authorization code has been refreshed by an administrator. The DN of the user is included with the audit log. Only the authorization code is changed; the reference number is not changed.
7757	Event	Delete all group members. All members of a group have been removed. The audit log will include the name and group identifier of the group. In addition, the number of users successfully deleted and the number of users that could not be deleted from the group will be displayed.
7758	Event	Copy all group members. The members of a group have been copied to another group. The audit log will include the names of the source and target groups. As well, a count of the number of users successfully copied and the number of users that could not be copied will be displayed.

Audit	Severity	Audit log message and information
7759	Event	Group(s) of user have changed. The groups to which a user belongs have changed. The audit log will include the DN of the user as well as a list of groups to which the user has been added and a list of groups from which the user has been deleted.
7760	Event	Role of user has changed. The role of a user has been changed. The audit log will include the DN of the user being changed as well as the name and role identifier of the new role.
7761	ALARM	Administrator does not have access to perform operation. The audit log will include information describing the current administrative user, what operation was attempted and why the administrative user did not have access to perform the operation.
7762	Event	Group deleted. A group has been deleted. The name and group identifier of the deleted group are shown in the audit log.
7763	Event	Role deleted. A role has been deleted. The name and role identifier of the deleted role is shown in the audit log.
7764	Event	Group changed. The name of a group has been changed. The old and new names of the group as well as the group identifier display in the audit log.
7765	Event	Role changed. A role has been changed. Multiple audit logs are written and display the various changes to role attributes. Each audit log message contains the name and role identifier of the role being changed.
7766	Event	Group created. A group has been created. The audit log contains the name and group identifier of the new group.
7767	Event	Role created. A role has been created. Multiple audit logs are written displaying the various attributes of the new role. Each audit log will contain the name and role identifier of the new role.
7768	Event	Initial roles created. The default roles are created during first-time initialization. The name and role identifiers of the default roles are provided in the audit log.

Audit	Severity	Audit log message and information
7769	Event	Initial group created. The initial group is created during first-time initialization. The name and group identifier of the group are recorded in the audit log.
7770	ALARM	Free disk space is critically low. Entrust won't start if disk space falls below the level set in the entmgrMinFree parameter in the entmgr.ini file. If a value for entmgrMinFree has not been specified, the default is 5 Mbytes.
7771	Event	A Master User acknowledged that the system clock has been advanced significantly. The system clock has been moved forward. A Master User has confirmed that this is the desired activity. After receiving the acknowledgment, PKIX-CMP subsystem must be started within one hour or the Master User will have to acknowledge the time change again. You cannot turn the time back significantly.
7772	ALARM	Notary information in the policy certificate could not be updated. An error occurred reading the information. The reason for the failure is included with the audit log.
7773	Event	Notary information in the policy certificate updated. A Master User changed information in the enttimestamp.ini file or created a new file with new information. This file is read each time the Administrative Service is started. See the *Entrust/Timestamp 4.0 User Guide* for details.
7774	Event	CMS: Insufficient privileges to do operations on this user. This message applies to Certificate Distribution Agent (CDA) requests. CDA uses Entrust/CMS to request certificates for other users. CDAs are always Security Officers or Administrators. No CDA operations are allowed on Security Officers. A CDA must be a Security Officer to do an operation on an Administrator. The CDA name and user name are included with the audit log and a message is returned to the CDA.
7775	Event	Offline cross-certificate requested. A Security Officer has created a PCKS #10 offline cross-certificate request. The Security Officer name and main CA DN are included with the audit log.

Audit	Severity	Audit log message and information
7776	Event	User State History database table trimmed. The user state history table has been automatically trimmed. This table is used for creating reports of user state changes. Trimming this table ensures that it does not take up too much space. This is done periodically by the PKIX-CMP subsystem. Trimming does not result in lost information. All events are recorded in the audit logs, which should be archived. Trimming is controlled by the entUserStateHistTrim parameters in the entmgr.ini file.
7777	Event	Certification Authority (CA) key transferred from software to hardware. A CA's main signing private key that was created by software and stored encrypted in the Entrust/Authority database has been transferred from software to hardware. This can be done after upgrading from a previous release of Entrust/Authority, or if the hardware is obtained after installing Release 6.0. The key transfer is done by a Master User using Entrust/Authority Master Control. All systems have a main CA signing private key. There is one audit log for each key moved. The names of the CA and the Master User are included in the audit log. For all subsequent additional CAs created, the keys are created in hardware. These keys cannot be moved from hardware back to software.
7779	Event	Revocation request rejected. A Certificate Distribution Agent (CDA) has requested the revocation of the certificates for one or more users and the request has been rejected. An explanation of the failure is returned to the CDA and is included with the audit log. The audit log also includes the name of the CDA and, if applicable, one or more user-to-be-revoked names.
7783	ALARM	Cryptographic hardware not found. An infrastructure process was unable to contact the cryptographic hardware. This means the system cannot operate. Refer to the *Entrust/PKI 6.0 Cryptographic Hardware Guide* for instructions on the hardware you are using.
7784	Event	Password for Setup Information Distribution changed. The password used to protect the files written for authorization code distribution has been changed. A Master User performs this activity using Entrust/Authority Master Control. The name of the Master User is included with the audit log.

Audit	Severity	Audit log message and information
7785	Event	State of setup information distribution changed. The setup information (reference number and authorization code) distribution state has been changed by a Master User using Entrust/Authority Master Control. The audit log contains the old state and the new state. There are four possible states: distribution is off/security is off (no encryption for files on disk), distribution is on/security is off (no encryption), distribution is on/security is on, distribution is off/security is on (no files are written to disk, but when the distribution feature is turned on the files are encrypted when they are written to disk on the Entrust/Authority machine).
7787	Event	Web security policy changed. A Security Officer has changed with Web-specific security policy. A description of the change and the Security Officer's name is included with the audit log.
7788	Event	Enterprise security policy changed. A Security Officer has changed the Enterprise-specific security policy. A description of the change and the Security Officer's name are included in the audit log.
7790	Event	User's flexible certificate type changed. A Security Officer or Administrator has changed certificate category of a user. The user's name and administrative user's name is included in the audit log.
7791	Event	User category changed. A Security Officer or Administrator has changed the user category of a user. A user's category can only be changed if they are enabled and they have not yet completed initialization. Administrators and Security Officers are always in the enterprise category. The certificate category cannot be changed. The administrative user's name and the user's name are included in the audit log.
7792	Event	Flexible certificate specification (FCS) definition file changed. A Security Officer has changed the flexible certificate definition file using Entrust/RA. There will be one audit log of this type for each change to the flexible certificate definition file. The Security Officer name is included in the audit log.
7800	Event	IBM 4755 encrypted Certification Authority (CA) signing private key read from shared memory. This message appears if you storing the CA key on an IBM 4755 hardware device. Applicable to the AIX operating system only. This audit log appears when the system is first installed and each time a process is started.

Audit	Severity	Audit log message and information
7801	Event	Unsupported PKIX inner message. An Entrust/CMS message containing PKIX information not supported by Entrust/Authority has been received. A description of the failure is included with the audit log.
7803	ALARM	Free disk space is getting low (time to clean up?). In the entmgr.ini file, the parameter entmgrWarnFree determines when Entrust prints a warning about disk space being low. If no one has provided an entmgrWarnFree parameter in the .ini file, the default is to issue a warning when 20 Mbytes or less of space is free. Note that Entrust will not start if disk space is below the amount specified in the entmgrMinFree parameter in the entmgr.ini file. See ALARM 7770.
7804	Event	CMS update request error. An Entrust/CMS request to update a user's certificate has failed. A description of the failure is included with the audit log.
7805	Event	CMS key recover request error. An Entrust/CMS request to recover a user's keys has failed. A description of the failure is included with the audit log.
7806	Event	CMS encryption certificate request missing. A request was issued to initialize an Entrust/CMS user, providing the user with an encryption certificate. However, this request itself is missing an encryption certificate request. The request indicated that the encryption certificate request was present.
7807	Event	CMS preliminary request message error. An Entrust/CMS or Entrust/Entelligence 5.0 or later application has requested preliminary information for a user. That request has failed. A description of the failure is included with the audit log.
7808	ALARM	CMS init user request error. An Entrust/CMS request to create a user identity has failed. A description of the failure is included with the audit log.
7810	Event	CMS reply error. Entrust/Authority experienced an internal error when formatting an Entrust/CMS reply. A description of the error is included with the audit log.
7811	Event	CMS invalid PKIX version received. An Entrust/CMS message has been received with a version that is not supported. The version in the message is included with the audit log.

Audit	Severity	Audit log message and information
7812	ALARM	CMS message processing error. An Entrust/CMS message has been received but cannot be used. The reason for the error is included with the audit log.
7813	Event	CMS request adjusted. Entrust/Authority adjusted a setting in an Entrust/CMS request. The request is still processed. If a single request has more than one adjustment, there is an audit log for each one. Each audit log includes a description of the change. A list of changes is also returned to the originator of the Entrust/CMS request.
7814	ALARM	CMS request refused. An Entrust/CMS request was refused. If available, the name of the user being operated on and other information is included with the audit log.
7815	Event	Directory restore canceled. Restoration of Entrust data to the Directory has been canceled with no harm done.
7816	ALARM	Certificate found in Directory issued by this Certification Authority (CA) invalid. When replacing a user's encryption public key certificate (such as during key update or during recovery of Directory data), if an invalid certificate is found in the user's entry that is issued by this CA, it is removed from the Directory.
7817	Log	Cross-certificate pairs written to Directory. Either a cross-certificate operation was performed or Directory data was recovered by a Master User or Security Officer.
7818	Event	Secure Exchange Protocol (SEP) encryption algorithm changed. The SEP algorithm has been changed by a Master User in Entrust/Authority Master Control. The change description is included in the audit log.
7819	ALARM	Certificate signature hash changed. The certificate hashing algorithm for the system has been changed by a Master User in Entrust/Authority Master Control. The change description is included in the audit log.
7820	Event	Expired revoked certificate re-added to revocation list. Removal of expired certificates from CRLs or ARLs is an option that can be set by Security Officers. If the removal setting is changed from Yes to No, expired certificates that have been removed previously are re-added—not immediately when the setting is changed, but the next time the CRL or ARL is updated.

Audit	Severity	Audit log message and information
7821	Event	Entrust full data backup canceled. The full data backup has been canceled with no harm done. The incomplete backup data should be discarded.
7822	Log	Master User logged off. A Master User has logged off from Entrust/Authority Master Control.
7823	Log	Admin user logged off. An Administrator or Security Officer using Entrust/RA has disconnected from Entrust/Authority. The IP address of the machine on which they were running Entrust/RA is included in the audit log.
7824	Event	License information changed. A Security Officer has changed the license information for the Entrust installation. The user category, old license limit, and new license limit are included with the audit log.
7826	Event	Database re-encryption completed successfully. This audit log will appear with audit logs 7851 (Administration subsystem password change), 7955 (Entrust master key), and 7956 (CA master key).
7827	ALARM	Database re-encryption failed. Re-encryption of the Entrust/Authority database failed. This audit log will state whether recovery from backup is mandatory or not.
7828	Event	Database integrity check canceled. A database integrity check was canceled with no harm done. Any errors found before cancellation will be included in the audit.
7829	ALARM	Infrastructure data files backup failed. Look in manager.log for details.
7830	Log	Infrastructure data files backup completed successfully. The infrastructure data files have been backed up successfully either manually by a Master User in Entrust/Authority Master Control, or automatically by the PKIX-CMP subsystem.
7831	ALARM	Database backup failed. Entrust/Authority database backup failed. Look in manager.log for details.
7832	Log	Incremental database backup completed successfully. Backup of the Entrust/Authority database was completed successfully either manually by a Master User in Entrust/Authority Master Control, or automatically by the PKIX-CMP subsystem.

Audit	Severity	Audit log message and information
7833	ALARM	Directory database backup failed. The Directory backup failed. Look in manager.log for details.
7834	Log	Directory database backup completed successfully. The Directory has been successfully backed up either manually by a Master User in Entrust/Authority Master Control, or automatically by the PKIX-CMP subsystem.
7835	ALARM	Integrity failure. An integrity failure has been detected in the Entrust/Authority database. The table and row that have failed integrity are included in the audit log.
7836	ALARM	Infrastructure certificate serial number counter adjusted. The certificate serial number counter has been reset by a Master User using Entrust/Authority Master Control. This avoids duplicate certificate serial numbers if there is a chance that some data may have been lost when the Entrust/Authority database is restored from backup.
7840	Event	User alternate name changed. An end user's alternate name (usually an e-mail address) for insertion in certificates has been changed by an Administrator or Security Officer. The end user's DN, old e-mail address, and new e-mail address are included in the audit log.
7841	Event	User verification certificate policy changed. The end user's verification certificate policy list has been changed by an Administrator or Security Officer. The end user's DN is included in the audit log, plus the changes to the list.
7842	Event	User encryption certificate policy changed. The end user's encryption certificate policy list has been changed by an Administrator or Security Officer. The end user's DN is included in the audit log, plus the changes to the list.
7843	Event	User private signature key lifetime changed. The lifetime of an end user's signing private key (set as a percentage of the verification public key certificate lifetime) has been changed by an Administrator or Security Officer.
7848	Event	Administration subsystem profile recovered. The Administration subsystem profile has been recovered automatically by Entrust/Authority Master Control when it starts and discovers that the verification public key certificate in the Administration subsystem profile has expired. The profile can also be recovered manually by a Master User in Entrust/Authority Master Control.

Audit	Severity	Audit log message and information
7849	Event	Directory Administrator password changed. The Directory Administrator's password has been changed by an Administrator or Security Officer.
7851	Event	The Administration subsystem profile password has been changed. The password to unlock the profile used by the Administration subsystem has been changed. This only occurs during a Re-encrypt Database operation done by a Master User in Entrust/Authority Master Control. This audit log will appear with audit logs 7955, 7956, and 7828.
7854	Log	Database integrity check was successful. The integrity check performed by a Master User was successful.
7855	ALARM	Database integrity check failed. An integrity check of the Entrust/Authority database has failed. Other audit logs indicating what failed is listed before this audit log. Check the manager.log as well. Contact Entrust Support; a recovery from the last backup may be necessary.
7856	ALARM	A signing private key update request has been received from a user with automatic key update disabled. The update request will be refused by the PKIX-CMP subsystem. Contact Entrust Support if this audit log occurs. The end user's DN is included in this audit log.
7857	ALARM	An encryption key update request has been received from a user with automatic key update disabled. The update request will be refused by the PKIX-CMP subsystem. Contact Entrust Support if this audit log occurs. The end user's DN is included in this audit log.
7858	Event	Automatic key update has been enabled for the user. An Administrator or Security Officer has enabled automatic key update for a user. The end user's DN is included in the audit log.
7859	Event	Automatic key update has been disabled for the user. An Administrator or Security Officer has disabled automatic key update for a user. The end user's DN is included in the audit log.
7860	Event	User signing private key lifetime property changed. An Administrator or Security Officer has changed the signing private key lifetime for a user. The user's DN is included in the audit log.

Audit	Severity	Audit log message and information
7861	Event	User encryption key lifetime property changed. An Administrator or Security Officer has changed the encryption public key certificate lifetime for a user. The user's DN is included in the audit log.
7862	Event	The user verification expire date has been changed. An Administrator or Security Officer has changed the expiry date of a user's verification public key certificate. The end user's DN is included in the audit log.
7863	Event	The user signing/encryption expire date has been changed. An Administrator or Security Officer has changed the expiry date of a user's signing or encryption key. The end user's DN is included in the audit log.
7864	ALARM	Failed to receive signing private key update acknowledgment from Entrust/Entelligence. The new end user signing certificate has been revoked. The PKIX-CMP subsystem revoked the end user's verification public key certificate because Entrust/Entelligence did not acknowledge receiving it. The user's DN, the certificate's serial number, and the certificate's validity period are included in this audit log. Contact the user immediately. If Entrust/Entelligence picked up the new verification public key certificate, the user will have to be set up for key recovery. The manager.log file will give the reason for the failure.
7865	ALARM	Failed to receive encryption key update acknowledgment from Entrust/Entelligence. An Entrust/Entelligence encryption key pair update operation failed after cryptographic information was sent to Entrust/Entelligence. The newly created encryption public key certificate and decryption private key are archived by the PKIX-CMP subsystem. If the key update failed at Entrust/Entelligence, and Entrust/Entelligence tries the encryption key pair update again, it will receive the encryption key pair that was just archived by the PKIX-CMP subsystem. The manager.log file will give the reason for the failure.

Audit	Severity	Audit log message and information
7866	ALARM	Failed to receive key recover acknowledgment from Entrust/Entelligence. The new user certificate has been revoked. A Recover operation failed after cryptographic information was sent to Entrust/Entelligence. The PKIX-CMP subsystem revoked the newly created end user verification public key certificate and encryption public key certificate (if one was created). The user's DN, the certificate's serial number, and the certificate's validity period are included in this audit log. Contact the user immediately. If the user received no errors during the Entrust/Entelligence Recover User operation, set the user up for key recovery. Ask the user to delete their existing profile (.epf file) and to repeat the Recover User operation. If this audit log recurs, the manager.log file will give the reason for the failure. Contact Entrust Support if the problem cannot be resolved.
7867	ALARM	Failed to receive user initialization acknowledgment from Entrust/Entelligence. The new user verification certificate will be revoked. A new user initialization failed after cryptographic information was sent to Entrust/Entelligence. The PKIX-CMP subsystem revoked the newly created user verification public key certificate and encryption public key certificate. The user's DN, the certificate's serial number, and the certificate's validity period are included in this audit log. Contact the user immediately. If the user received no errors during the Entrust/Entelligence Create New User operation, set the user up for key recovery. Ask the user to delete their existing profile (.epf file) and to perform the Recover User operation. If this audit log recurs, the manager.log file will give the reason for the failure. Contact Entrust Support if the problem cannot be resolved.
7873	ALARM	An error occurred cleaning up a data file backup file. A data file backup file could not be deleted. Look in the manager.log file for details.
7874	Event	A data file failure was successfully corrected. A data file write error was successfully corrected.
7875	ALARM	An error occurred cleaning a data file failure. A data file failure could not be corrected. Look in the manager.log file for details.
7876	ALARM	An error occurred writing to the database. The Entrust/Authority database is corrupt. Look in the manager.log file for details.
7877	ALARM	The data file has a corrupted entry. An Entrust data file has a corrupt entry. Please contact Entrust Support.

Audit	Severity	Audit log message and information
7878	ALARM	The data file is missing an entry. An Entrust data file is missing an entry. Please contact Entrust Support.
7881	Event	Distinguished name change for user completed by the client application. The client application has completed the DN change operation. The end user's DN is included in the audit log.
7882	Event	Distinguished name change for a user approved. A Security Officer or Administrator has approved an end user DN change. The Security Officer or Administrator name and the end user's DN are included in the audit log.
7883	Event	Distinguished name change for a user canceled. A user had been set up for a DN change but this was canceled by a Security Officer or an Administrator before the client application completed the DN change. The Security Officer's or Administrator's name and the end user's DN are included in the audit log.
7884	ALARM	An invalid certificate serial number was received from the client application. This audit log indicates that the client application is not synchronized with Entrust/Authority. Set the user up for key recovery. The end user's DN is included in the audit log.
7885	ALARM	Protocol verification certificate validation failed. The current protocol verification certificate is not valid. Please contact Entrust Support.
7886	ALARM	Certification Authority (CA) certificate validation failed. The CA certificate is not valid. Please contact Entrust Support.
7887	ALARM	An Entrust data file is missing. The name of the missing file is included in the audit log. Please contact Entrust Support.
7888	Event	New authority revocation list created. The current ARL space is reserved for future certificate revocations and a new ARL is created. The name of the entry where the ARL is stored (distribution point) is included in the audit log.
7889	Event	New certificate revocation list created. The current ARL space is reserved for future certificate revocations and a new ARL is created. The name of the entry where the CRL is stored (that is, its distribution point) is included in the audit.

Audit	Severity	Audit log message and information
7890	ALARM	Attribute has invalid extension. A CRL or certificate has an invalid extension. An example of an extension is a CRL distribution point extension which is contained in certificates and indicates where the CRL for that certificate is located. The extension name and attribute type are included in the audit log. Please contact Entrust Support.
7891	ALARM	Attribute is missing an extension. A CRL or certificate is missing an extension. An example of an extension is a CRL distribution point extension which is contained in certificates and indicates where the CRL for that certificate is located. The extension name and attribute type are included in the audit log. Please contact Entrust Support.
7892	ALARM	Internal audit log error. An internal error has occurred in the audit log. Please contact Entrust Support.
7895	Event	Certification Authority (CA) policy certificate updated. The certificate is updated either due to a user searchbase change done by a Security Officer (with audit log 7900), or because it is about to expire and is updated by the PKIX-CMP subsystem. This audit log always appears with audit log 7899.
7896	ALARM	Certification Authority (CA) policy certificate validation failed. The certificate is not valid. This may be due to corrupt data. Please contact Entrust Support.
7897	Event	Certification Authority (CA) policy certificate written to the Directory. This audit log occurs with audit logs 7900 and 7897 when a user searchbase is modified. It appears by itself when the Directory data is recovered back to the Directory. It appears with audit log 7987 when the CA attribute certificate is updated by the PKIX-CMP subsystems because it is close to expiring.
7898	Event	Entrust searchbase information updated. Searchbase information is updated. There are two searchbase lists: one for users (the user searchbase list) which is placed in the CA attribute certificate, and one for Security Officers and Administrators when using Entrust/RA (the admin searchbase list). Security Officers can modify both lists; Administrators can modify only the admin searchbase list. The type of list modified is included in the audit log. If the user searchbase list is modified, this audit log will appear with audit logs 7899 and 7897.

Audit	Severity	Audit log message and information
7899	Event	Certification Authority (CA) policy certificate created. The first certificate is created. The CA policy certificate is stored in the CA Directory entry and contains Directory searchbase information for Entrust/Entelligence. This audit log occurs only during the first-time start-up of Entrust/Authority Master Control.
7900	Log	Authority revocation list written to the Directory. The DN of the issuer (which is always the CA) is included in the audit log.
7901	ALARM	The authority revocation list at the Directory is expired. This is the same as for the CRL. See audit log number 7932.
7902	ALARM	Remote authority revocation list validation failed. The ARL at a remote CA is not valid. Inform a Security Officer in the remote CA that the ARL validation failed.
7903	ALARM	Authority revocation list validation failed. The signature on the ARL stored in the Entrust/Authority database is not valid. This could be the result of corrupt data. Please contact Entrust Support.
7904	ALARM	Cross-certificate validation failed. A cross-certificate issued by your CA or another CA is not valid.
7905	Event	An invalid cross-certification authorization code was entered. A Security Officer has entered an invalid authorization code during an Complete Cross-Certification operation. If a number of these audit logs occurs, it indicates that an unauthorized operation is being attempted.
7906	Event	Cross-certification transaction failed. An Accept Cross-Certification operation has failed. Look in the manager.log file for more information.
7907	Event	Reverse cross-certification completed. Your Certification Authority (CA) has signed a reverse certificate with its CA signing private key which contains the remote CA verification public key, and your CA has included this certificate in its cross-certificate pairs. This audit log occurs at both the local and remote CA when the Security Officer(s) for one CA perform an Accept Cross-Certification operation. This audit log includes the DN of the remote CA, and always appears with audit log 7910.

Audit	Severity	Audit log message and information
7908	Event	Forward cross-certification completed. A remote Certification Authority (CA) has signed a forward certificate with its CA signing private key which contains your CA verification public key, and your CA has included this certificate in its cross-certificate pairs. This audit log occurs at both the local and remote CA when the Security Officer(s) for one CA perform an Accept Cross-Certification operation. This audit log always appears with audit log 7909.
7909	Event	Reverse cross-certificate revoked. The reverse cross-certification is canceled (see audit log 7912). The reverse certificate which is signed with your CA's signing private key and contains the remote CA's verification public key is placed on the ARL.
7910	Event	Reverse cross-certification canceled. This audit log occurs under the same conditions and time as audit log number 7911. The difference is that the local Certification Authority (CA) revokes its reverse certificate if cross-certification is already set up between the two CAs.
7911	Event	Forward cross-certification canceled. This audit log occurs when • a Revoke Cross Certificate operation is performed by the Security Officer(s). If the operation has not yet been completed, only the authorization code information is removed which makes it impossible for the operation to be completed by the remote Certification Authority (CA). • a Complete Cross-Certification operation is performed by the Security Officer(s) when the cross-certification has already been completed with the remote CA. This audit log will appear at both the remote and local CA. The forward certificates are replaced by new ones. This audit log includes the DN of the remote CA and always appears with audit log 7912.
7912	Event	Reverse cross-certification initialized. A Begin Cross-Certification operation is performed by the Security Officer(s). This audit log includes the DN of the remote Certification Authority (CA).
7913	Event	Forward cross-certification initialized. A Begin Cross-Certification operation is performed by the Security Officer(s). This audit log includes the DN of the remote Certification Authority (CA).

Audit	Severity	Audit log message and information
7914	ALARM	Entrust full data backup failed. Check the manager.log file for error information.
7915	Log	Entrust full data backup completed successfully. The Entrust/Authority database, data files, and (optionally) the Directory data have backed up successfully. Backups are run automatically by the PKIX-CMP subsystem or manually by a Master User in Entrust/Authority Master Control.
7916	Event	Certification Authority (CA) Directory password changed. The CA Directory password stored in the Directory has been changed by a Master User in Entrust/Authority Master Control. The name of the Master User and the CAs DN are included in the audit log.
7919	Event	Entrust subsystem stopped. Windows NT only. A Key Management Assistant process has stopped. A process identifier is included with the audit log, along with optional text indicating if the process stopped because of an error.
7920	Event	Entrust subsystem started. Windows NT only. A Key Management Assistant process has started. A process identifier is included with the audit log.
7921	Event	Master Control stopped. The Entrust/Authority Master Control application has stopped. If the process was terminated by the application because of an error, the error text will be included in the audit log.
7922	Event	Master Control started. The Entrust/Authority Master Control application has started. The process ID is included in the audit log.
7929	ALARM	Security policy validation failed. The security policy cannot be read by Entrust/Authority Master Control, the PKIX-CMP subsystem or Administration subsystem. This is a serious event and could mean that data is corrupt. Please contact Entrust Support.
7930	Event	Security policy has been modified. One or more Security Officers have modified a security policy item. The old value, new value and user name of the Security Officer(s) who performed the modification are included in the audit log. If more than one item is modified at a time, there will be one audit log per item.

Audit	Severity	Audit log message and information
7931	Log	Expired certificate removed from revocation list. An Entrust/Entelligence user encryption public key certificate or verification public key certificate that was on the CRL, or a cross-certificate on the ARL, has been removed because the certificate has reached the end of its validity period. The serial number of the certificate is included in the audit log.
7932	ALARM	The certificate revocation list at the Directory is expired. One of the CRLs stored in the Directory is out of date. This occurs if the PKIX-CMP subsystem is stopped for a time exceeding the CRL lifetime (which can be set anywhere from 4 to 48 hours, in 1-hour increments), or if the Directory service is not running. In either case, the PKIX-CMP subsystems will immediately issue a new CRL and place it in the Directory. The Directory entry in which the CRL is stored and the CRL expiry date are included in the audit log. See the manager.log file for possible error messages.
7933	ALARM	Certificate revocation list validation failed. The signature on the CRL stored in the Entrust/Authority database is not valid. This may be the result of corrupt data. Please contact Entrust Support.
7939	Event	User encryption certificate revoked. An end user's latest encryption certificate has been revoked by a Security Officer or an Administrator using Entrust/RA. The certificate revoked was the last encryption public key certificate that Entrust/Entelligence has retrieved. If a Security Officer or an Administrator has updated an end user's encryption key pair and Entrust/Entelligence has not yet retrieved it, the certificate in the Directory is not revoked but the previous certificate (which Entrust/Entelligence has retrieved) is revoked. The certificate serial number and end user DN are included in the audit log.
7941	Event	User verification certificate revoked. A user's latest verification public key certificate has been revoked by a Security Officer or an Administrator using Entrust/RA. The certificate's serial number and end user's DN are included in the audit log.
7943	Log	Certificate revocation list written to the Directory. The CRL is written to the Directory. The DN of the issuer (which is always that of the CA) is included in the audit log.
7944	ALARM	Write to Directory failed. An attempt to write data to the Directory has failed. Explanatory text, if available, may be included in the audit log and in the manager.log file.

Audit	Severity	Audit log message and information
7945	ALARM	Read from Directory failed. An attempt to read data from the Directory has failed. Explanatory text, if available, may be included in the audit log and in the manager.log file.
7946	ALARM	Directory not responding. Entrust/Authority Master Control, the PKIX-CMP subsystem, the SEP subsystem, or Administration subsystem cannot contact the Directory.
7947	ALARM	Entrust first time initialization failed. The first-time initialization of the Entrust/Authority Master Control application failed. See the manager.log file for details of the errors.
7948	Event	Entrust first-time initialization complete. The first-time initialization of the Entrust/Authority Master Control application completed successfully.
7949	ALARM	User certificate validation failed. An end user encryption public key certificate or verification public key certificate is not valid. Information about the type of certificate and the user's DN is included in the audit log. Please contact Entrust Support.
7950	Log	Certificate recovered back to Directory. When a Master User does a Recover Directory operation, this audit log appears for each certificate that is written back to the Directory. The certificate owner's DN is included in the audit log.
7951	Event	Protocol verification certificate created. The protocol verification certificate has been created. This audit log occurs during the initial start of the Entrust/Authority Master Control application and when the PKIX-CMP subsystem automatically updates the protocol signing private key pair.
7952	Event	Certification Authority (CA) certificate created. The CA certificate has been created. This audit log occurs only during the initial start of Entrust/Authority Master Control.
7953	Event	Entrust master key created. This audit log occurs only during the initial start of Entrust/Authority Master Control.
7954	Event	Certification Authority (CA) master key created. This audit log occurs only during the initial start of Entrust/Authority Master Control.

Audit	Severity	Audit log message and information
7955	Event	Entrust master key updated. The Entrust master key was updated by a Master User in Entrust/Authority Master Control, who re-encrypted the database. The name of the Master User who performed the operation is included in the audit log. Will appear with audit logs 7956, 7853, and 7828.
7956	Event	Certification Authority (CA) master key updated. The CA master key has been updated by a Master User in Entrust/Authority Master Control, who re-encrypted the database. The name of the Master User who performed the operation is included in the audit log. Will appear with audit logs 7955, 7853, and 7828.
7957	Event	Protocol signing private key pair updated. The protocol signing private key pair has been updated automatically by the PKIX-CMP subsystem.
7958	Event	Protocol signing private key pair created. The protocol signing private key has been created. This audit log only occurs during the initial start of Entrust/Authority Master Control.
7960	Event	Certification Authority (CA) signing private key pair created. The CA signing private key pair has been created. This audit log only occurs at the initial start of Entrust/Authority Master Control.
7963	Event	Master User password reset. A Master User's password has been reset. This Master User's name and that of the Master User who performed the operation are included in the audit log.
7964	Event	Master User password change. A Master User has changed their own password. The name of the Master User is included in the audit log.
7965	Log	Successful Administrator login. An Administrator or Security Officer using Entrust/RA has connected to Entrust/Authority. The IP address of the machine on which the administrative user is running Entrust/RA is included in the audit log.
7966	Event	Successful Master User login. A Master User has successfully started Entrust/Authority Master Control, Administration subsystem, or the PKIX-CMP subsystem. The name of the Master User is included in the audit log.

Audit	Severity	Audit log message and information
7968	Event	Invalid Master User login attempt. A non-existent Master User name and/or invalid password was entered when running Entrust/Authority Master Control. The audit logs should be monitored for excessive events of this type as this may indicate that an unauthorized person is trying to use Entrust. The name that was entered is included in the audit log.
7972	Event	Master User created. A Master User has been created by Entrust/Authority Master Control during first-time installation of the infrastructure. The name of the Master User is included in the audit log.
7976	Event	Certificate lifetime exceeds Certification Authority (CA) signing private key lifetime. The intended expiry date of a newly created certificate exceeds the expiry date of the CA signing private key (that is, the verification certificate validity period). A certificate is valid for no longer than the key it is signed with. For example, if the Entrust/Authority CA certificate has a fixed validity period of 20 years and you have been running Entrust for 16 years and have set the Entrust/Entelligence user verification certificate validity to 60 months (5 years). When a verification certificate is created, the expiry date on this certificate should be 1 year later than the expiry date of the CA certificate. Since this is not allowed, the expiry date of the verification certificate is set to equal the expiry date of the CA certificate. Information about the dates is included in the audit log.
7977	Log	User verification certificate created. A verification public key certificate has been created for an Entrust/Entelligence user. This occurs when • an Entrust/Entelligence user is initialized • an Entrust/Entelligence user performs a Recover User operation • Entrust/Entelligence does a signing private key pair update The end user's DN is included in the audit log.

Audit	Severity	Audit log message and information
7978	Log	User encryption certificate created. A encryption public key certificate has been created for an Entrust/Entelligence user. This occurs when • an Entrust/Entelligence user is initialized • Entrust/Entelligence performs an encryption key pair update • an administrative user manually updates an Entrust/Entelligence user's encryption key pair The end user's DN is included in the audit log.
7979	Event	User key recovery canceled. The key recovery that has been set up for a user was canceled by an administrative user.
7980	ALARM	User key recovery failed. An Entrust/Entelligence user has attempted to do key recovery and a failure has occurred. The most likely reason for failure is due to the password or authorization number being entered incorrectly at the Entrust/Entelligence. The manager.log file will contain any errors associated with this failure. The end user's DN is included in the audit log.
7981	Event	User key recovery completed. An Entrust/Entelligence user has successfully recovered the encryption key pair history. The end user DN is included in the audit log.
7982	Event	User key recovery initialized. An Entrust/Entelligence user in the active state has been set up for key recovery by an administrative user. The end user's DN is included in the audit log.
7983	ALARM	User signing private key pair update failed. Entrust/Entelligence has attempted a signing private key pair update but failed. The manager.log file will contain detailed error messages. The end user's DN is included in the audit log.

Audit	Severity	Audit log message and information
7985	Event	User signing private key pair updated. Entrust/Entelligence has done a successful signing private key pair update. This transaction is transparent to the Entrust/Entelligence user. The end user's DN and the reason for the update are included in the audit log. The various reasons are described here: • Close to expiration. Entrust/Entelligence has done a signing private key pair update because the current key pair is nearing its expiry date. • Pending DN change. Entrust/Entelligence has done a signing private key pair update because the user's DN has changed.
7986	ALARM	User encryption key pair update failed. Entrust/Entelligence has attempted an encryption key pair update but failed. The manager.log file will contain detailed error messages. The end user's DN is included in the audit log.
7987	Event	User encryption key pair update done when previous key pair had expired. Entrust/Entelligence has successfully updated its encryption key pair after the previous encryption key pair expired. The end user's DN is included in the audit log.

Audit	Severity	Audit log message and information
7988	Event	User encryption key pair updated. Entrust/Entelligence has done a successful encryption key pair update. This transaction is transparent to the Entrust/Entelligence user. The end user's DN and reason for the update are included in the audit log. The various reasons are described here: • Close to expiration or revoked. Entrust/Entelligence has done an encryption key pair update because the current key pair it has is revoked or nearing expiration. • Pending DN change. Entrust/Entelligence has done an encryption key pair update because the end user's DN has changed. • Retrieval of previous key pair. Entrust/Entelligence has done an encryption key pair update but the PKIX-CMP subsystem found that the client possessed the second-to-last encryption key pair. In this case, the PKIX-CMP subsystem does not update the pair but sends the latest encryption key pair to Entelligence. This occurs if Entrust/Entelligence has done an encryption key pair update and the user is not using their latest profile. • Initiated by administrator. A Security Officer or an Administrator has forced the update of a user's encryption key pair. • Update pending. Entrust/Entelligence is doing an encryption key pair update, and the key has already been updated by a Security Officer or an Administrator. • Automatic update changes. A Security Officer or an Administrator has enabled or disabled automatic key update or changed the signing private key/encryption public key expiry date or the verification public key expiry date.

Audit	Severity	Audit log message and information
7989	Event	User deactivated. An Entrust/Entelligence user has been deactivated by a Security Officer or an Administrator. This means that the latest encryption public key certificate for that user is removed from the Directory and the Entrust user license count is decreased by one. The end user's DN and the reason for deactivating the user are included in the audit log. A user can be deactivated by a Security Officer or an Administrator or may be deactivated because of three consecutive failures during key update operations. The various reasons are described here: • Deactivated by administrative user • Deactivated because of errors during user initialization • Deactivated because of errors during key recover • Deactivated because of errors during encryption key update • Deactivated because of errors during signing private key update
7990	Event	User reactivated. An Entrust/Entelligence user who was previously deactivated has been reactivated. This means that the latest encryption public key certificate for the user has been replaced in the Directory and the Entrust user license count is increased by one. The end user's DN is included in the audit log.
7991	ALARM	User initialization failed. An Entrust/Entelligence or PXIX-CMP user has attempted to create a profile and a failure has occurred. The most likely reason for failure is due to the password or authorization number being entered incorrectly at Entrust/Entelligence. The manager.log file will contain any errors associated with this failure. The end user's DN is included in the audit log.
7992	Event	User initialized. An Entrust/Entelligence user has successfully created a profile. The end user's DN is included in the audit log.
7993	Log	User removed. An previously-added Entrust/Entelligence user is then deactivated by a Security Officer or an Administrator. The user's data files are removed from the Entrust/Authority database. This is of no consequence because the encryption key pair has not yet been retrieved by Entrust/Entelligence for use. The end user's DN is included in the audit log.
7994	Event	User added. An Entrust/Entelligence user has been added and activated. An administrative user can activate the end user at the same time as adding the end user, or at a later time.

Audits logged for revocation lists

Audits are logged for revocation lists (RLs), and the revocation list is written to the Directory, if any of the following events occur:

- a certificate is revoked under the following circumstances:

  An administrative user has revoked a user certificate and "automatic writing of CRLs when a certificate is revoked" is enabled.
- restore to Directory
- new revocation list created
- administrative user publishing changed CRLs
- forced write due to CA key revocation
- nearing expiration

  In a large system you may want to suppress these audits to reduce the size of the audit trail. These audits occur when a CRL is nearing expiration and is re-issued and written to the Directory. This is a normal operating condition.
- expired
- offline ARL setting changed
- offline ARLs pushed
- combined CRL setting changed

  This will only occur if the offline ARL setting is enabled.
- certificate hold canceled

For example, the following audit is logged when an ARL is written to the Directory because it has expired:

```
874 : Wed Aug 16 14:53:14 2000 : 7900 : Authority revocation list (ARL) written to the
Directory. : Log : : : updated = 'Wed Aug 16 14:53:14 2000', expires =
'Wed Aug 16 21:53:14 2000', distribution point = 'cn=CRL1,o=polar bear inc,c=CA ,
reason for write = 'expired' :
```

Appendix B

Customizing the initialization files

An initialization file contains entries that determine how an application should run. Entrust/PKI uses three initialization files: the entrust.ini file, the entrustra.ini file, and the entmgr.ini file.

The installation program creates the initialization files and saves them in the Entrust root folder on the Entrust/Authority server. The installation program writes some entries to these files based on your choices during installation, and includes other entries by default.

You can customize how Entrust-Ready applications run by adding, removing, and modifying entries. Once you have finished customizing these files, you distribute them to an appropriate location as described in this chapter.

This chapter describes each of the initialization files used by Entrust/PKI and defines each entry in the files. This chapter contains information about

- "Note on default values" on page 484
- "Customizing the entrust.ini file" on page 485
- "Customizing the entrustra.ini file" on page 520
- "Customizing the entmgr.ini file" on page 527

Note on default values

A default value is provided for each entry in this chapter. A default value in this chapter means one of two things:

- the value which is placed in the initialization file during installation or configuration of Entrust/PKI
- the value which will be used if the entry is *not* present in the initialization file

If this chapter indicates that an entry has no default value, this means one of two things:

- the entry has no effect if it is not present in the initialization file.
- you must manually specify a value for the entry. For example, an entry requiring you to enter an IP address has no default value.

Customizing the entrust.ini file

The entrust.ini file defines important information about how Entrust-Ready applications and Entrust/RA run. The installation program creates this file and saves it in the C:\authdata\manager folder, by default, on the Entrust/Authority server.

Entrust Technologies recommends that you customize and distribute two versions of the entrust.ini file: one version for end users and one version for administrative users. Some entries in the entrust.ini file are required only to run Entrust/RA and are not required in the entrust.ini file distributed to end users.

Table 1 lists each entry in the entrust.ini file for end users and its default value (if any), and describes how the entry is used. An example entrust.ini file appears in "Example of an entrust.ini file" on page 515. Table 2 on page 517 describes the additional entries for administrative users.

Table 1: Entries in the entrust.ini file for all Entrust users

Entry	Description
;	A marker that indicates that the current line contains a comment. When using the entrust.ini file, Entrust-Ready applications ignore everything on a line starting with a semicolon. You can include as many comments as you want in an entrust.ini file. For example, you can add a comment for each entry to describe why you set the entry to a particular value.
[Entrust Settings]	A section header for entries that define general Entrust profile settings.
AcceptOnlyOAEP	An entry that determines the padding technique (either PKCS #1 version 1.5 or PKCS #1 version 2) used to unwrap a decryption key. The padding technique defined by PKCS #1 version 2 increases the security of the key, and is known as Optimal Asymmetric Encryption Padding (OAEP). The value is "0" (allow both padding techniques) or "1" (allow only PKCS #1 version 2 padding techniques). You must set this entry to "0" if you are using Entrust/PKI 4.0 with later releases of Entrust/PKI. Note that the AcceptOnlyOAEP entry is the same as the ProcessOAEPOnly entry. However, the AcceptOnlyOAEP entry applies only to applications based on EntrustSession. These applications may use different settings from, or may be from a different release of EntrustSession than, other applications. Default: 0

Table 1: Entries in the entrust.ini file for all Entrust users (continued)

Entry	Description
ArlCacheEnabled	An entry that determines whether authority revocation lists (ARLs) are stored in a cache. Entrust-Ready applications use this cached information, if it is valid, in subsequent sessions. For example, this information may be used in offline sessions. Certificates from the cache are validated every time they are used. The value is "0" (no) or "1" (yes). Default: 1
Authority	An entry that specifies the address and port number used by Entrust/PKI 5.0 (or later) applications to connect to Entrust/Authority. These applications communicate with Entrust/Authority using the version of PKIX-CMP in Entrust/PKI 5.0 (or later). This entry is set during installation. The value is either an IP address or DNS specification, and the port number (for example, Authority=1.2.3.4+829). This entry doesn't support multiple values. Only Entrust/PKI 5.0 (or later) applications that support the current version of PKIX-CMP can use this address and port. As well, an Entrust profile from Release 4.0 that is updated or recovered using PKIX-CMP cannot be converted for use with SEP. If this entry is not defined, applications connect to Entrust/Authority using the address and port number specified by the Manager entry. Note that PKIX-CMP features are not available through the address and port defined by the Manager entry. Note that if you are using both Entrust/PKI 5.0 (or later) applications and Entrust/PKI 4.0 (or earlier) applications, you also require the Manager entry (see the Manager entry in this table). Default: No default
AuthTimeout	An entry that specifies the maximum time, in seconds, to wait for SEP exchanges with Entrust/Authority to complete. This entry isn't used by Entrust/PKI 5.0 or later. However, don't delete this entry until all Entrust-Ready applications have been upgraded to Release 5.0 or later. Default for applications based on CMS toolkit: 45 Default for other applications: No default

Table 1: Entries in the entrust.ini file for all Entrust users (continued)

Entry	Description
BannerFile	An entry that specifies the name of a file that contains a bitmap image. This image is displayed to the user when they log in. The filename must use the extension ".bmp" (for example, logo.bmp). The location of this file depends on how you plan to distribute the entrust.ini file. This entry isn't used by Entrust/PKI 5.0 or later. However, don't delete this entry until all Entrust-Ready applications have been upgraded to Release 5.0 or later. Default: No default
CertificateCache Enabled	An entry that determines whether user certificates are stored in a cache (a file named "profile name.cch" where "profile name" varies). Entrust-Ready applications use this cached information in subsequent sessions. For example, this information may be used in offline sessions. Certificates from the cache are validated every time they are used. The value is "0" (no) or "1" (yes). Default: 1
CertificateCacheSize	An entry that specifies the number of user certificates that are stored in a cache if caching is enabled (that is, "CertificateCacheEnabled=1"). This entry isn't used by Entrust/PKI 5.0 or later. However, don't delete this entry until all Entrust-Ready applications have been upgraded to Release 5.0 or later. Default: 50
ClientSocketTimeout	An entry that specifies the length of time, in seconds, that PKIX-CMP connections can be idle before they time out. Default: 45
ClientType	An entry that specifies whether the user is working with the "regular" Entrust/PKI or Entrust/Lite. The value is "Heavy" or "Lite". Releases of Entrust/PKI beginning with 5.0 don't support Entrust/Lite. This entry must be set to Heavy. Default: Heavy
ConnectTimeout	An entry that specifies the maximum time, in seconds, to wait for a Directory connection to be established. This entry isn't used by Entrust/PKI 5.0 or later. However, don't delete this entry until all Entrust-Ready applications have been upgraded to Release 5.0 or later. Default: 30

Table 1: Entries in the entrust.ini file for all Entrust users (continued)

Entry	Description
CrlCacheEnabled	An entry that determines whether CRL information is stored in a cache (that is, a file named "profile name.crl" where "profile name" varies). Entrust-Ready applications use this cached information in subsequent sessions. For example, this information may be used in offline sessions. Certificates from the cache are validated every time they are used. The value is "0" (no) or "1" (yes). Default: 1
CrossCertARLCache Enabled	An entry that determines whether authority revocation lists (ARLs) from cross-certified CAs are stored in a cache (that is, a file named "profile name.arl" where "profile name" varies). Entrust-Ready applications use this cached information in subsequent sessions. For example, this information may be used in offline sessions. Certificates from the cache are validated every time they are used. This entry isn't used by Entrust/PKI 5.0 or later. However, don't delete this entry until all Entrust-Ready applications have been upgraded to Release 5.0 or later. The value is "0" (no) or "1" (yes). Default: 1
CrossCertARLCache Size	An entry that specifies the maximum number of ARLs from cross-certified CAs that are stored in a cache if caching is enabled (that is, "CrossCertARLCacheEnabled=1"). This entry isn't used by Entrust/PKI 5.0 or later. However, don't delete this entry until all Entrust-Ready applications have been upgraded to Release 5.0 or later. There is no maximum value. The minimum value is "1". A value of "0" is treated as "1". Default: 8
CrossCertCACache Enabled	An entry that determines whether cross-certificates are stored in a cache (that is, a file named "profile name.xcc" where "profile name" varies). Entrust-Ready applications use this cached information in subsequent sessions. For example, this information may be used in offline sessions. Certificates from the cache are validated every time they are used. The value is "0" (no) or "1" (yes). Default: 1

Table 1: Entries in the entrust.ini file for all Entrust users (continued)

Entry	Description
CrossCertCACacheSize	An entry that specifies the maximum number of cross-certificates that are stored in a cache if caching is enabled (that is, "CrossCertCACacheEnabled=1"). There is no maximum number. This entry isn't used by Entrust/PKI 5.0 or later. However, don't delete this entry until all Entrust-Ready applications have been upgraded to Release 5.0 or later. Default: 5
CrossCertCRLCache Enabled	An entry that determines whether certificate revocation lists (CRLs) from cross-certified CAs are stored in a cache (that is, a file named "profile name.crl" where "profile name" varies). Entrust-Ready applications use this cached information in subsequent sessions. For example, this information may be used in offline sessions. Certificates from the cache are validated every time they are used. Entrust/Entelligence caches CRLs from both its own and cross-certified CAs. This entry isn't used by Entrust/PKI 5.0 or later. However, don't delete this entry until all Entrust-Ready applications have been upgraded to Release 5.0 or later. The value is "0" (no) or "1" (yes). Default: 1
CrossCertCRLCache Size	An entry that specifies the maximum number of CRLs from cross-certified CAs that are stored in a cache if caching is enabled (that is, "CrossCertCRLCacheEnabled=1"). This entry isn't used by Entrust/PKI 5.0 or later. However, don't delete this entry until all Entrust-Ready applications have been upgraded to Release 5.0 or later. The minimum value is "0" (which is treated as "1") and there is no maximum value. Default: 8
CrossCertDebug	An entry that determines whether attempts by Entrust/Entelligence to locate a certificate for the target cross-certified CA are recorded. If recorded, results are written to a file named "xcert.txt". This file is located on the drive where the Entrust/Entelligence binaries are installed. If the file exists, results are appended to it. If the file doesn't exist, it is created. The value is "0" (no) or "1" (yes). Default: 0

Table 1: Entries in the entrust.ini file for all Entrust users (continued)

Entry	Description
CrossCertMaximum Depth	An entry that specifies the maximum number of levels that an Entrust-Ready application traverses in the Directory when looking for a certificate for the target cross-certified CA. In most cases, a value of 2 or 3 is appropriate. There is no maximum value for this entry. Default: No default
CryptokiLibrary95	An entry that specifies the path to the CRYPTOKI library (the software.dll for a hardware token) on Windows 95. This entry is used only if a hardware token is used, and is written to the entrust.ini file when the token's driver is installed. Default: No default
CryptokiLibraryNT	An entry that specifies the path to the CRYPTOKI library (the software.dll for a hardware token) on Windows NT. This entry is used only if a hardware token is used, and is written to the entrust.ini file when the token's driver is installed. Default: No default
CryptokiV2Library	An entry that specifies the path to the CRYPTOKI library (the software.dll for a hardware token) on Windows 2000. This entry is used only if a hardware token is used, and is written to the entrust.ini file when the token's driver is installed. Default: No default
DefaultContext_time _req	An entry that determines the length of time, in seconds, that the security context remains valid. This entry is used only for applications based on EntrustSession. The value "0" means that the security context remains valid forever. Default: 0
DefaultCredentials _time_req	An entry that determines the length of time, in seconds, that credentials remain valid. This entry is used only for applications based on EntrustSession that don't use the single login feature. The value "0" means that credentials remain valid forever. This setting was increased to 604800 (seven days) in Release 5.1. Default: 0
DefaultProfileLocation	An entry that specifies the path to the Entrust profile for this user. This path can be modified during the Create User operation. Default: No default

Table 1: Entries in the entrust.ini file for all Entrust users (continued)

Entry	Description
DirectoryPassword	An entry that specifies a password (encrypted) that an Entrust-Ready application uses to connect to the Directory instead of connecting anonymously. Default: No default
EncryptWith	An entry used to specify the symmetric algorithm used for regular encryption operations by users. After a new user is created and a profile is initialized, this algorithm can be changed using the Entrust Options dialog box in Entrust/Entelligence. Note that only the CAST-128 symmetric algorithm can be used to initialize the Entrust profile when a new user is created. The value is one of the following: • CAST (CAST-128 or the strongest available CAST algorithm) • CAST-128 • CAST-80 • CAST-64 • CAST-40 • DES • Triple DES • RC2-128 Compatible • RC2-40 Compatible • IDEA Note that this symmetric algorithm may also be specified by the "allowed symmetric encryption algorithms" user policy (see "Setting Entrust/PKI user policies" on page 167). If this symmetric algorithm is defined in both the entrust.ini file and the user's profile, the value in the entrust.ini file is ignored, and the value in the user's profile is used. Default: CAST

Table 1: Entries in the entrust.ini file for all Entrust users (continued)

Entry	Description
EncryptWithOAEP	An entry that determines the padding technique (either PKCS #1 version 1.5 or PKCS #1 version 2) that is allowed when wrapping RSA keys for encryption operations. The padding technique defined by PKCS #1 version 2 increases the security of the key, and is known as Optimal Asymmetric Encryption Padding (OAEP). The value is "0" (use PKCS #1 version 1.5) or "1" (use PKCS #1 version 2). You must set this entry to "0" if you are using Entrust/PKI Release 4.0 applications with later releases of Entrust/PKI applications. Note that the EncryptWithOAEP entry is the same as the GenerateOAEP entry. However, the EncryptWithOAEP entry applies only to applications based on EntrustSession. These applications may use different settings from, or may be from a different release of EntrustSession than, other applications. Default: 0
GenerateOAEP	An entry that determines the padding technique (either PKCS #1 version 1.5 or PKCS #1 version 2) that is allowed when wrapping RSA keys for encryption operations. The padding technique defined by PKCS #1 version 2 increases the security of the key, and is known as Optimal Asymmetric Encryption Padding (OAEP). The value is "0" (use PKCS #1 version 1.5) or "1" (use PKCS #1 version 2). You must set this entry to "0" if you are using Entrust/PKI Release 4.0 applications with later releases of Entrust/PKI applications. Note that the GenerateOAEP entry is the same as the EncryptWithOAEP entry. However, the EncryptWithOAEP entry applies only to applications based on EntrustSession. These applications may use different settings from, or may be from a different release of EntrustSession than, other applications. Default: 0
IdentityLibraryNT=full_path_to_identity_library	An entry used to specify the identity library. This setting applies only if you are using an identity device. Specify the full path to the identity library (dll). If you install Entrust/Toolkit Server Login, then the entrust.ini file will be updated to include an identity library entry. For more information, see the Entrust/Toolkit Server Login Release Notes for release 5.1. These release notes can be obtained from the Entrust Support web site.

Table 1: Entries in the entrust.ini file for all Entrust users (continued)

Entry	Description
InstallDir	An entry that specifies the root folder of the installation directory for an Entrust-Ready application. This entry is written to the entrust.ini file during installation and is used for reference for support and re-installation. Default: No default
LastWorkOffline	An entry that specifies whether the user has selected the "Remember last setting for Work Offline" option in the Entrust Options dialog box. If the user deselects this option, the LastWorkOffline setting is removed from the entrust.ini file. Default: No default
LastProfile to LastProfile4	An entry that specifies the path to the last five Entrust profiles used on the workstation. Entrust-Ready applications update these entries automatically when a user logs in. These entries are subsequently used to populate the pull-down menus from which users can select profiles for login. Default: No default
LDAPProxyClient	An entry that indicates whether an Entrust-Ready application needs to receive a certificate revocation list (CRL) in-band when establishing a connection with EntrustSession. This entry applies only to Entrust-Ready applications using EntrustSession Release 5.0 or later. This entry is used only during an offline login (that is, a login that takes place when an Entrust-Ready application cannot access the Directory). The value is "0" (no) or "1" (yes). Default: 1
LDAPProxyServer	An entry that indicates whether the initiator of an EntrustSession connection should send the certificate revocation list (CRL) in-band when establishing a connection with EntrustSession, in case the receiver of the connection is offline (that is, the Directory is inaccessible to the receiver). This entry applies only to Entrust-Ready applications using EntrustSession Release 5.0 or later. The value is "0" (no) or "1" (yes). Default: 1

Table 1: Entries in the entrust.ini file for all Entrust users (continued)

Entry	Description
MailDll	An entry that specifies the type of mail client for mail-enabled Entrust-Ready applications (that is, "MailEnabled=1"). This entry isn't used by Entrust/PKI 5.0 or later. However, don't delete this entry until all Entrust-Ready applications have been upgraded to Release 5.0 or later. The value is "VIM" or "MAPI". If you specify "VIM", you must specify a post office name for VIM mail clients (see the entry "PostOffice" on page 495). Default: No default
MailEnabled	An entry that determines whether an Entrust-Ready application is mail-enabled. This information is changed in the Entrust Options dialog box in Entrust/Entelligence. This entry isn't used by Entrust/PKI 5.0 or later. However, don't delete this entry until all Entrust-Ready applications have been upgraded to Release 5.0 or later. The value is "0" (no) or "1" (yes). Default: 0
Manager	An entry that specifies the IP address and port number used by Entrust-Ready applications to connect to SEP for communications with Entrust/Authority. If you are using only Entrust/PKI 5.0 or later Entrust-Ready applications and the "Authority" entry is defined in the entrust.ini file, the Manager entry is ignored. If no value is defined for this entry or for the "Authority" entry, Entrust-Ready applications use the address and port specified in the "Server" entry. The value of this entry is either an IP address or DNS specification, and the port number (for example, "Manager=1.2.3.4+709"). This entry does not support multiple values. Default: No default
MaximumCrossCert Nodes	An entry that determines the maximum number of CAs to check to find a path to the target cross-certified CA. There is no maximum value for this entry. Default: 20

Table 1: Entries in the entrust.ini file for all Entrust users (continued)

Entry	Description
PostOffice	An entry that specifies the post office name for VIM mail clients if an Entrust-Ready application is mail-enabled (see MailEnabled entry). This entry isn't used by Entrust/PKI 5.0 or later. However, don't delete this entry until all Entrust-Ready applications have been upgraded to Release 5.0 or later. Default: No default
ProcessOAEPOnly	An entry that determines the padding technique (either PKCS #1 version 1.5 or PKCS #1 version 2) used to unwrap a decryption key. The padding technique defined by PKCS #1 version 2 increases the security of the key, and is known as Optimal Asymmetric Encryption Padding (OAEP). The value is "0" (allow both padding techniques) or "1" (allow only PKCS #1 version 2 padding technique). You must set this entry to "0" if you are using Entrust/PKI 4.0 with later releases of Entrust/PKI. Note that the AcceptOnlyOAEP entry is the same as the ProcessOAEPOnly entry. However, the AcceptOnlyOAEP entry applies only to applications based on EntrustSession. These applications may use different settings from, or may be from a different release of EntrustSession than, other applications. Default: 0
ProfileServer	An entry that specifies the IP address and port number of the Profile Server for roaming login. This entry is only necessary when using Entrust/Roaming. The value is either an IP address or DNS specification, and port number (for example, "ProfileServer=1.2.3.4+640"). This entry doesn't support multiple entries. Default IP address: No default Default port: 640
ProfileServerDN	An entry that specifies the distinguished name of the Profile Server (for example, "ProfileServerDN=cn=ProfileServer, o=Your Company, c=US"). The value for this entry must exist in the Directory specified by the Server entry. This entry is only necessary when using Entrust/Roaming. Default: No default

Table 1: Entries in the entrust.ini file for all Entrust users (continued)

Entry	Description
ProfileServerKeyType	An entry that determines the default symmetric key for all communications with the Profile Server. The Profile Server must be configured to allow communication with this symmetric key type or the Profile Server will reject all communication attempts. This entry is only necessary when using Entrust/Roaming. The value is one of the following: • CAST-128 • Triple DES • IDEA Default: CAST-128
ProgDir	An entry that specifies the folder where the Entrust/Entelligence executable is located. This entry is used by Microsoft Word and Microsoft Excel macros.The value is written to the entrust.ini file during installation. Default: No default
Recipients	An entry that specifies the Windows coordinate entries, where "Recipients" is the dialog box in Entrust/Entelligence and "438 261" (for example) is the last location the window appeared. Written to the entrust.ini file when the window is opened. These coordinates are dependent on the user's display resolution set for their monitor. This entry isn't used by Entrust/PKI 5.0 or later. However, don't delete this entry until all Entrust-Ready applications have been upgraded to Release 5.0 or later. Default: No default
RoamingIDField	An entry that specifies the Directory entry that is searched for the user's roaming ID. For example, the entry "RoamingIDField=phoneNumber" means that the roaming user ID is the same as the user's phone number Directory entry. This entry is only necessary when using Entrust/Roaming. Default: No default
RoamSearchBase	An entry that specifies the Directory location where searches for a user's entries are performed. This entry acts in the same way as the SearchBase entry (for example, "RoamSearchBase=o=Your Company,c=US"). This entry is only necessary when using Entrust/Roaming. Default: No default

Table 1: Entries in the entrust.ini file for all Entrust users (continued)

Entry	Description
SearchAttributeOrder	An entry that allows you to specify the order of attributes Entrust/Express uses when searching for users' certificates. For example, "mail" is an alias for the "rfc822Mailbox=" attribute, and "email" is an alias for the "msMailID=" attribute. When "mail" is used, the address is expected to be a standard RFC822 SMTP address of the form user@domain, with no other information. When "email" or any other mail attribute is used, the address is expected to be of the form "<TYPE>:<address>", where "<TYPE>" is one of the standard address types (for example, "EX" for Microsoft Exchange, "OPENMAIL" for HP OpenMail), and the address format follows that of the mail system type. Default: mail;email
SearchBase	An entry that specifies the name of the default searchbase. This value is written to the entrust.ini file during installation. If multiple searchbases are created, only the default searchbase will appear in this entry. Additional searchbases are defined in the policy certificate, not in the entrust.ini file. Default: No default
SearchBaseOrder	An entry that allows you to choose the searchbase sequence when client applications attempt to search for users in Entrust/Authority. Enter the DN of the searchbases and separate each with a semicolon. Note: For ease of maintenance, the preferred method for specifying searchbase sequence is via the user policy setting "Searchbase Search Order" (see "Overview of policy settings" on page 168). The user policy setting overrides the SearchBaseOrder setting. The SearchBaseOrder setting only has an effect if the user policy does not contain this setting (that is, if you are using a release of Entrust/PKI prior to Release 6.0). Default: Entrust/Express will loop through all searchbases.
SearchTimeout	An entry that specifies the maximum time, in seconds, to wait for a Directory operation to complete. This entry isn't used by Entrust/PKI 5.0 or later. However, don't delete this entry until all Entrust-Ready applications have been upgraded to Release 5.0 or later. Default: 180

Table 1: Entries in the entrust.ini file for all Entrust users (continued)

Entry	Description
SEPByProxy	An entry that determines whether a proxy Entrust-Ready application can create a profile for a remote Entrust-Ready application that doesn't have online access to Entrust/Authority. For information about using proxy applications and other related settings, see the documentation for EntrustSession. This entry applies only to Entrust-Ready applications using EntrustSession 4.0 or earlier. These applications use the SEP protocol for communicating with Entrust/Authority. This entry doesn't apply to Release 5.0 (or later) Entrust-Ready applications, which communicate using the PKIX-CMP protocol. The value is "0" (yes) or "1" (no). Default: 0
SEPVersion	An entry that specifies the version of SEP used for backward compatibility in proxied environments (see the SEPByProxy entry). This entry applies only to Entrust-Ready applications for EntrustSession 4.0 or earlier. These applications use the SEP protocol for communicating with Entrust/Authority. This entry doesn't apply to Release 5.0 (or later) Entrust-Ready applications, which communicate using the PKIX-CMP protocol. The value is "V3.1.0.D", "4.0 domestic version of SEP", "V4.0.0.D", or "V5.0.0.D". Default: V3.1.0.D

Table 1: Entries in the entrust.ini file for all Entrust users (continued)

Entry	Description
Server	An entry that specifies the address and port number used by Entrust-Ready applications to connect to an LDAP-compliant Directory (or LDAP servers fronting a Directory). The value of this entry is an IP address or a DNS specification, and port number. It can also be set to "<none>" (without the quotation marks) to work offline. This entry is written to the entrust.ini file during installation of Entrust/PKI. If you have multiple LDAP servers, you can specify more than one address and port number. Separate each address with a comma (for example, "Server=1.2.3.4+389, 1.2.3.6+389"). If the first address doesn't respond within the time specified by the DirectoryOperationTimeLimit entry, the Entrust-Ready application tries to connect using the second address, and so on. Once the Entrust-Ready application has connected to an LDAP-compliant Directory using this address and port, the application doesn't try to connect to any other LDAP-compliant Directory for the duration of the session. If no value is defined for the Manager entry, Entrust-Ready applications use the address and port specified in the Server entry. When the applications try to send the key management message to the LDAP server, errors occur. Default: <none>

Table 1: Entries in the entrust.ini file for all Entrust users (continued)

Entry	Description
SigningKey	An entry that specifies the default digital signature algorithm for users. This entry is used only by Entrust-Ready applications that use SEP (that is, Entrust-Ready applications prior to Release 5.0 and Entrust-Ready Release 5.0 or later applications that use SEP). Entrust-Ready applications check this entry before user creation, key recovery, or automatic update of the signing key pair. The value is one of the following: • RSA (RSA 1024-bit key) • RSA-2048 (RSA 2048-bit key) • DSA (DSA 1024-bit key) • ECDSA (ECDSA 192-bit key) Entrust-Ready applications prior to Release 5.0 can use only "RSA" and "DSA". Note that this digital signature algorithm may also be specified by the "algorithm for digital signatures" user policy (see "Setting Entrust/PKI user policies" on page 167). If this algorithm is defined in both the entrust.ini file and the user policy, Entrust-Ready applications use the value specified by the user policy. Default: RSA
TraceFile	An entry that determines whether logging is performed. This setting only applies to applications based on EntrustSession. The value is "0" (no) or "1" (yes). Default: 0
V21Compatible	An entry that determines whether only algorithm OIDs supported by version 2.1 will be sent. This setting only applies to applications based on EntrustSession. The value is "True" (version 2.1 compatibility) or "False". The values "True" and "False" are not case-sensitive. Default: False
Version	An entry that specifies the release of the software and its platform. This entry is written to the entrust.ini file during installation. This entry isn't used by Entrust/PKI 5.0 or later. However, don't delete this entry until all Entrust-Ready applications have been upgraded to Release 5.0 or later. Default: No default

Table 1: Entries in the entrust.ini file for all Entrust users (continued)

Entry	Description
[Directory Connection Settings]	A section header for entries that affect connections by Entrust-Ready Release 5.0 (or later) applications to the Directory. The entries in this section don't affect Entrust-Ready applications prior to Release 5.0.
ASN1Attributes	An entry that lets you specify the list of ASN.1-encoded attributes. Attributes on this list will be interpreted as ASN.1-encoded attributes (which may be either in ASCII format or binary format). These values will not be decoded as UTF8 and will be converted from indefinite length to definite length, if necessary. The value is a list of attribute names, each of which is separated by a comma. Default: userCertificate, CACertificate, authorityRevocationList, certificateRevocationList, crossCertificatePair, attributeCertificate, entrustPolicyCertificate, deltaRevocationList, x509AttributeCertificate, attributeCertificateRevocationList, aACertificate, crossPrivilegeCertificate, attributeAuthorityRevocationList, attributeDescriptorCertificate, attributeCertificateAttribute.
AuthMethod	An entry that instructs an application to perform NTLM authentication to the directory using the Windows credentials of the user who is currently logged in. This setting is added to the entrust.ini file during configuration of Entrust/PKI 6.0 with Active Directory. Default: WinAuth
AVASorting	An entry that can be used to enable sorting of attribute value assertions (AVAs) within a relative distinguished name (RDN) when a distinguished name (DN) is displayed in Entrust/RA. By default DNs displayed in Entrust/RA are either formatted according to the ASN.1 distinguished encoding rules (DER) or as they are returned by the Directory, which may result in AVAs appearing in different orders when an RDN contains more than one AVA. For example, if an RDN contains cn and serialNumber (cn=Jones + serialNumber=1234), the RDN may be displayed with either cn first or serialNumber first, depending on the values for each attribute. To enable AVA sorting, set this entry to "1". Default: 0

Table 1: Entries in the entrust.ini file for all Entrust users (continued)

Entry	Description
AVASortOrder	An entry that can be used to configure the order in which attribute value assertions (AVAs) are sorted in a relative distinguished name (RDN) when AVA sorting is enabled. By default, if AVA sorting is enabled, the attributes are sorted alphabetically. To specify a custom sorting order, add the entry AVASortOrder to the entrust.ini file used by Entrust/RA and specify a comma-separated list of attributes in the order in which you would like them to appear in the RDN. If a custom sorting order is specified and additional attributes which are not listed in the AVASortOrder entry occur in an RDN, they will be placed at the end of the AVA list in alphabetical order. AVA sorting must be enabled otherwise this setting will be ignored. In the following example, the AVA sort order is serial number followed by common name: [Directory Connection Settings] AVASortOrder=serialNumber,cn Default: No default
BadHostsLogFile	An entry that specifies the IP addresses and port numbers of the servers defined by the Server entry (see "Server" on page 499) that were unavailable when contacted. The servers are listed in the file specified by this entry, along with the time that the Entrust-Ready application tried to connect, plus the interval specified by the RetryTime entry. For example, if an application could not connect at 8:00 and the RetryTime entry specifies an interval of two hours, then no application will attempt to contact this server before 10:00. Default: No default
DeleteOldRDN	An entry that allows you to choose to keep or delete the old attributes in the entry during a DN change. Set this to 1 to set the default to "always delete old RDN" or 0 to set the default to "don't delete old RDN". This entry applies only to Entrust/RA Toolkit and not to Entrust/RA; Entrust/RA contains a setting allowing you to specify whether to keep or delete old attributes. Default: 0

Table 1: Entries in the entrust.ini file for all Entrust users (continued)

Entry	Description
DirectoryConnectTime Limit	An entry that specifies the length of time, in seconds, to wait for a connection operation to the LDAP Directory to succeed. If no response from the Directory is received within this time period, the attempt to connect is aborted and the user will log in without access to the Directory. Default: 0 (use the system default connect time limit)
DirectoryOperation TimeLimit	An entry that specifies the time, in seconds, that the Directory has to respond when an Entrust-Ready application attempts to connect to the host server. Once an Entrust-Ready application has connected to a server, it will not attempt to connect to any other server for the duration of that session. For Entrust/PKI releases prior to 5.0, the time-out applied only to search operations. For Release 5.0 and later, the time-out applies to all operations (for example, add, modify, and delete operations). This entry may also be specified in the [Entrust Settings] section using the DirectorySearchTimeLimit entry. If the entry is specified in both sections, the value in the [Entrust Settings] section is used. Default: 180
DirectoryProtocol	An entry that specifies the version of the LDAP protocol used by Entrust-Ready applications to connect to the Directory. The value is "LDAPv2" or "LDAPv3". This entry is set when you configure Entrust/Authority. This entry is also included in the entmgr.ini file, and it may have a different value there. Default: LDAPv2
DirectorySearchSize Limit	An entry that specifies the maximum number of entries displayed from a Directory search initiated by Entrust/Entelligence. The value "0" specifies no maximum. This entry may also be specified in the [Entrust Settings] section. If the entry is specified in both places, the value in the [Entrust Settings] section is used. Default: 300

Table 1: Entries in the entrust.ini file for all Entrust users (continued)

Entry	Description
DirectoryServerLimit	An entry that allows you to specify the maximum number of seconds the server will wait during a search call. This time limit is used only for searches, and if this parameter is omitted, the DirectoryOperationTimeLimit is used in its place (see "DirectoryOperation TimeLimit" on page 503). The user must still wait a maximum time specified in DirectoryOperationTimeLimit, but DirectoryServerLimit compensates for clock drift among servers during a search. If you set a value for DirectoryServerLimit, set its value higher than DirectoryOperationTimeLimit, or else it will have no effect. This entry does not affect Entrust-Ready applications prior to Release 5.1. The valid range is 0-32767, in seconds (where 0 is used when an infinite time-out is desired). Default: If this entry is not included in the entrust.ini file, it defaults to the value set for DirectoryOperationTimeLimit.
DNSAttempts	An entry that specifies how many times to poll a Domain Name Service (DNS) server to determine if it is down. A DNS server is responsible for translating domain names to IP addresses (for example, "www.yourcompany.com=231.11.222.4"). To look up a name, a computer must determine which DNS server to contact. When a DNS server responds to polling, the server is assumed to be available. Polling the DNS servers before resolving host names speeds up online login operations. When the user attempts to log in online, the DNS server is polled for the number of times specified by the DNSAttempts entry. If the DNS server doesn't respond, it is assumed to be unavailable. This condition is common on laptop computers with docking stations. To turn off this feature, set this entry to "0". DNS servers will not be polled. Default: 0

Table 1: Entries in the entrust.ini file for all Entrust users (continued)

Entry	Description
FilterAttributeOptions	An entry that determines whether option tags are included in attribute names in LDAP version 2 or version 3 Directory searches. If tags are stripped, the attributes are combined into the same value list. Possible values are • "0": do not strip any attribute options from the attribute description; pass them up to the application intact • "1": filter out all the attribute options; attribute descriptions will be merged into the general attribute type and passed to the application without the option tags • "2": filter out only the ;binary option; all other options will be passed to the application Default: 2
Ldapv2UseOptions	An entry that determines whether options should be used in LDAP version 2 Directories. The value is one of the following: • 0 (no) • 1 (use the default option ";binary" for the Directory) • 2 (use configured attribute options) • 3 (use both the default and configured options) Default: 0
Ldapv3PreventBind	An entry that prevents the bind and unbind operations from being performed only for anonymous simple bind operations. LDAP v3 has the ability to perform anonymous operations without first executing the bind operation on a new connection. Entrust-Ready applications Release 6.0 and later can take advantage of this feature so that redundant bind/unbind operations need not be performed. If Ldapv3PreventBind is set to 1 (active) and an anonymous simple bind is performed, the bind and unbind will not be performed on that connection. By contrast, if a simple bind with password (or any non-anonymous bind) is required, the bind and unbind will be performed (the setting will have no effect in this case). This setting has no effect when operating in LDAP v2 (the bind and unbind will always be performed). Setting this entry to "1" enables this feature. If the entry is absent from the entrust.ini file, the feature is disabled. Default: 0

Table 1: Entries in the entrust.ini file for all Entrust users (continued)

Entry	Description
OpaqueAttributes	An entry that allows the user to specify a list of binary attributes that should not be decoded as ASN.1 or UTF8. When one of these attributes is received in a search, the value will be converted to HEX and prefixed with "{BIN}" for display. The value is a list of attribute names, each of which is separated by a comma. Default: entrustRoamingProfile, entrustRoamFileEncInfo, entrustRoamingPAB, entrustRoamingRecipList, entrustRoamingSLA, entrustRoamingCAPAB, entrustRoamingPRV, entrustRoamingEOP, objectGUID, objectSID.
ReferralAuthMethods	An entry that allows you to control the authentication method used when an LDAP referral is chased. The value should be a comma-separated list containing any of the following authentication methods, in your preferred order: simple (name and password), anonymous, WinAuth (Windows authentication). In the following example: [Directory Connection Settings] ReferralAuthMethods=anonymous,simple the first preference when authenticating to the referred LDAP server is anonymous authentication, and the second is simple name and password authentication. Both methods would be accepted. If the value is set to WinAuth, the first preference when authenticating to the referred LDAP server is Windows Authentication, and the second is simple name and password authentication. Default: anonymous

Table 1: Entries in the entrust.ini file for all Entrust users (continued)

Entry	Description
ReferralDepthLimit	An entry that specifies the maximum hop count for processing LDAP referrals and search continuations. The hop count is the number of referrals that the client will follow before "giving up" on the search. The effective setting, if this entry is missing from the entrust.ini file, is "0", which means that the clients will not follow LDAP referrals for search operations. Entrust-Ready applications (Release 5.0 or later) are capable of chasing LDAP referrals, on Search operations only, if the ReferralDepthLimit setting is configured to a value greater than 0. Note that the application must also be configured to use LDAPv3 as the Directory access protocol (referral chasing in LDAPv2 is not supported). To do this, the DirectoryProtocol setting in the entrust.ini file must be set to LDAPv3. For example, consider a situation where ReferralDepthLimit=2 and a search operation to server A is performed. Server A returns 3 referrals to servers B, C and D. Server B then returns a referral to E and E returns a referral to F. The search to server F will not be performed and the search will return a "Depth Limit exceeded" error. Default: 0
RetryTime	An entry that determines the time (in seconds) to wait before retrying a host listed in an entry in the BadHostsLogFile entry. This entry is used only when there are multiple entries in the Server entry and one of them fails. As well, a path and filename must be specified for the BadHostsLogFile entry. To turn off this feature, set the RetryTime entry to "0". Default: 0
SessionlessConnect	An entry that eliminates persistent connections for Entrust-Ready applications Release 6.0 and later. Entrust-Ready applications Release 6.0 and later do not require persistent connections. Note that this does not apply to Entrust/Authority servers. Setting this entry to "1" enables this feature. If the entry is absent from the entrust.ini file, the feature is disabled. Default: 0
UseBinaryOption Defaults	An entry that determines whether default LDAPv3 ";binary" options are turned off. The value is "0" (off) or "1" (on). Default: 1

Table 1: Entries in the entrust.ini file for all Entrust users (continued)

Entry	Description
[X500 Search Data Structure]	A section header for entries that define the attributes that can be used as criteria when searching the Directory using Entrust's Address Book. (See [X500 Group Data Structure] for information about defining which attributes are returned as the results of a search.) The format of each entry in the [X500 Search Data Structure] is data*n=attribute,label,width* where • "n" is a number starting from "0" (for example, "data0", "data1", "data2", and so on) • "attribute" is the valid name of the X.500 attribute (for example, "sn" or "cn") • "label" is the description of the attribute that is displayed on-screen (type "N/A" to omit a label) • "width" is the number of characters to allocate for entering the criteria on-screen (for example, "5") Note that previous releases included a "d" parameter that determined whether to display the attribute on-screen. For example, the following entry specifies that a user can search the Directory based on a last name (for example, search for entries where sn=smith). The text "Last Name" appears in the Address Book, and the user can enter criteria that contains a maximum of 30 characters. [X500 Search Data Structure] data0=sn,Last Name,30 You can specify any number of entries in this section. For simple searches, only the first two attributes specified are available as search criteria. For advanced searches, all the attributes specified are available as search criteria. The following example shows that three attributes (that is, "sn", "cn", and "email") may be used as search criteria. The "sn" and "cn" attributes can be used for simple and advanced searches. The "email" attribute can be used only for advanced searches. [X500 Search Data Structure] data0=sn,Last Name,30 data1=cn,Name,30 data2=email, Email Address, 60

Table 1: Entries in the entrust.ini file for all Entrust users (continued)

Entry	Description
[X500 Search Data Structure] (continued)	Defaults: data0=sn,Last Name,24 data1=cn,Common Name,24 data2=mail,eMail,64
[X500 Group Data Structure]	A section header for entries that define the attributes that are returned as the results of a Directory search using Entrust's Address Book. (See [X500 Search Data Structure] for information about defining which attributes are used as search criteria.) The format of each entry in the [X500 Group Data Structure] is data*n*=*attribute,label,width* where • "n" is a number starting from "0" (for example, "data0", "data1", "data2", and so on) • "attribute" is the valid name of the X.500 attribute (for example, "sn" or "cn") • "label" is the description of the attribute that is displayed on-screen (type "N/A" to omit a label) • "width" is the number of characters to allocate for entering the criteria on-screen (for example, "5") Note that previous releases included a "d" parameter that determined whether to display the attribute on-screen. For example, the following entry specifies that search results are displayed in "Last Name". If users search for "smi", anyone with a last name beginning with "smi" is returned in the search results. [X500 Group Data Structure] data0=surname,Last Name,30

Table 1: Entries in the entrust.ini file for all Entrust users (continued)

Entry	Description
[X500 Group Data Structure] (continued)	You can specify any number of entries in this section. For simple searches, only the attribute specified by the first entry is returned. For advanced searches, attributes specified by all of the entries are returned. For example, the following entries specify three attributes (that is, "cn", "rfc822Mailbox", and "serialNumber"). The "cn" attribute is returned for both simple and advanced searches. The "rfc822Mailbox" and "serialNumber" attributes are returned only for advanced searches. [X500 Group Data Structure] data0=cn,Name,30 data1=rfc822Mailbox, email name, 60 data2=serialNumber, serial number, 30 Default: data0=cn,Name,30
[X500 Display Data Structure]	A section header for entries that define which attributes defined in the [X500 Group Data Structure] section are to be displayed. Entrust-Ready applications (Release 5.0 or later) that use the Entrust Address Book (such as Entrust/Entelligence and Entrust/Express) and Entrust/Entelligence Release 4.0 don't use this section and its entries. There is no limit to the number of entries in this section. The format of each entry is data*n=attribute,label,width* where • "n" is a number starting from "0" (for example, "data0", "data1", "data2", and so on) • "attribute" is the valid name of the X.500 attribute (for example, "sn" or "cn") • "label" is the description of the attribute that is displayed on-screen (type "N/A" to omit a label) • "width" is the number of characters to allocate for entering the criteria on-screen (for example, "5") Note that previous releases included a "d" parameter that determined whether to display the attribute on-screen. Defaults: data0=cn,Name,30 data1=sn,Last Name,30

Table 1: Entries in the entrust.ini file for all Entrust users (continued)

Entry	Description
[X500 Sorting Data Structure]	A section header for entries that define which attributes defined in the [X500 Group Data Structure] section can be used to sort. There is no limit to the number of entries in this section. The format of each entry is data*n*=*attribute,label,width* where • "n" is a number starting from "0" (for example, "data0", "data1", "data2", and so on) • "attribute" is the valid name of the X.500 attribute (for example, "sn" or "cn") • "label" is the description of the attribute that is displayed on-screen (type "N/A" to omit a label) • "width" is the number of characters to allocate for entering the criteria on-screen (for example, "5") Default: data0=sn,N/A,30
[OIDAlias]	A section header for the AliasName entry.
AliasName	An entry that specifies the customized OID aliases that reside in your Directory. This entry is for customized installations. Default: No default
[OIDTable]	A section header for the AlgName entry.
AlgName	An entry that specifies the customized OIDs that your organization uses. This entry is for customized installations. If you define an OID in this section, you must enter an X500 attribute syntax in the [X500AttrSyntax] section. There is no limit to the number of OIDs you can define. Each OID must be on a separate line in the entrust.ini file. Default: No default
[X500AttrSyntax]	A section header for the X500AttributeType entry.

Table 1: Entries in the entrust.ini file for all Entrust users (continued)

Entry	Description
X500AttributeType	An entry that specifies your organization's customized OIDs using X500 attribute syntax. This entry is for customized installations. If you specify an X500 attribute in this section, you must define an OID in the [OIDTable] section. There is no limit to the number of X500 attributes you can define. Each X500 attribute must be on a separate line in the entrust.ini file. Default: No default
[PEMAlgNames]	A section header for the PEMAlgorithmName entry.
PEMAlgorithmName	An entry that specifies the definitions for the customized algorithms in the Directory. This entry is for customized installations. This entry isn't used by Entrust/PKI 5.0 or later. However, don't delete this entry until all Entrust-Ready applications have been upgraded to Release 5.0 or later. Default: No default
[DOS Extensions]	A section header for mapping entries to the entries in the [MacTypes] section. Encrypted files are regularly transferred between the PC and Macintosh platforms. For the *Launch* operation in the Decrypting and Verifying dialog box in Entrust/Entelligence to work correctly, the applications between the two platforms must be mapped. For example, the [DOS Extensions] entry "CSV=10" is mapped to the [MacTypes] entry "10=MS Excel 4.0 Macrosheet". For example: [DOS Extensions] BAT=34 C=36 TXT=15 CMD=34 These entries aren't used by Entrust/PKI 5.0 or later. However, don't delete these entries until all Entrust-Ready applications have been upgraded to Release 5.0 or later. Default: See the entrust.ini file.

Table 1: Entries in the entrust.ini file for all Entrust users (continued)

Entry	Description
[MacTypes]	A section header for mapping entries to the entries in the [DOS Extensions] section. Encrypted files are regularly transferred between the PC and Macintosh platforms. For the *Launch* operation in the Decrypting and Verifying dialog box in Entrust/Entelligence to work correctly, the applications between the two platforms must be mapped. For example, the [DOS Extensions] entry "CSV=10" is mapped to the [MacTypes] entry "10=MS Excel 4.0 Macrosheet". For example: [Mac Types] 01=XCEL,XLC3; MS Excel 3.0 Chart 02=XCEL,XLS3; MS Excel 3.0 Spreadsheet 03=XCEL,XLM3; MS Excel 3.0 Macrosheet 04=XCEL,XLW3; MS Excel 3.0 Workspace These entries aren't used by Entrust/PKI 5.0 or later. However, don't delete this entry until all Entrust-Ready applications have been upgraded to Release 5.0 or later. Default: See the entrust.ini file.
[Login Strings]	A section header for the Message entry.
Message	An entry that specifies a message that appears at the bottom of the Entrust Login dialog box. You can enter a string such as "Contact your Entrust Administrator at 555-1234." You can also define a message using the "Message of the day" user policy (see "Setting Entrust/PKI user policies" on page 167). This message appears only when a specific set of circumstances exist, and it overrides the message defined by the Message entry in the entrust.ini file. Default: <None>

Table 1: Entries in the entrust.ini file for all Entrust users (continued)

Entry	Description
[Attribute Options]	A section header for entries that can be used to create specialized instances of an attribute type. The attribute options defined by these entries are an LDAPv3 feature. For example, the option ";binary" is used to indicate that the attribute's values are DER-encoded binary values. Other types of options can be used to create language tags or any other type of special qualifier. The default behavior is to automatically use ";binary" in LDAPv3 when the following attributes are referenced: userCertificate, attributeCertificate, cACertificate, certificateRevocationList, authorityRevocationList, crossCertificatePair, entrustPolicyCertificate, and x509AttributeCertificate. Also, only the ";binary" attribute options will automatically be stripped regardless of the protocol version. This behavior can be customized using AttributeName entries.
AttributeName	An entry that specifies a list of options for an attribute. An example option is "binary". The format of this entry is *attributeName=option <option> <option>....* where each option is separated by a space and "attributeName" is one of the following: • userCertificate • attributeCertificate • CACertificate • certificateRevocationList • authorityRevocationList • crossCertificatePair • x509AttributeCertificate • attributeCertificateAttribute Default: No default
[FIPS Mode]	A section header for entries that are used when running in FIPS 140-1 mode. These entries are provided by Entrust Support. This section applies to Windows installations only. There is no limit to the number of entry pairs in this section. One pair is provided as an example below (see "Etfile32Auth" on page 515 and "Etfile32Name" on page 515). Other pairs of entries in this section provide authorization strings and engine names used when FIPS Mode is enabled. If the necessary entries are not inserted by the application's installation procedure, contact Entrust Support.

Table 1: Entries in the entrust.ini file for all Entrust users (continued)

Entry	Description
Etfile32Auth	An entry that specifies a string that contains a MAC of etfile32.dll, along with an algorithm identifier. If this entry is present, the engine runs in FIPS mode and checks its fingerprint against the MAC specified in this entry. Default: No default
Etfile32Name	An entry that specifies the name of the EntrustFile Toolkit engine (for example, "Etfile32Name=etfile32"). Default: No default
FipsMode	An entry that specifies whether FIPS 140-1 Mode is enabled or disabled. Setting this entry to "1" enables FIPS mode. Default: 0

Example of an entrust.ini file

```
[Entrust Settings]
ClientSocketTimeout=240
Authority=1.2.3.4+829
Manager=1.2.3.4+709
Server=1.2.3.4+389
DefaultProfileLocation=
EncryptWith=Cast
ClientType=Heavy
MaximumCrossCertNodes=20
CrossCertCACacheSize=5
ArlCacheEnabled=1
CrossCertCRLCacheSize=8
CrossCertCACacheEnabled=1
CrossCertARLCacheSize=8
CrossCertDebug=0
CrlCacheEnabled=1
CertificateCacheEnabled=1
CertificateCacheSize=50
SearchBase=o=Your Company,c=US
SigningKey=RSA
DefaultCredentials_time_req=43200
DefaultContext_time_req=3600
ProgDir=
InstallDir=

[Directory Connection Settings]
DirectoryProtocol=LDAPV3
DirectorySearchSizeLimit=300
```

```
DirectoryOperationTimeLimit=180

[X500 Search Data Structure]
data0=sn,Last Name,24
[X500 Group Data Structure]
data0=cn,Name,30,d

[X500 Display Data Structure]
data0=cn,Name,30,d
data1=sn,N/A,30

[X500 Sorting Data Structure]
data0=sn,N/A,30

[OIDAlias]
; AliasName=AlgName

[OIDTable]
; AlgName=Numeric Object Identifier

[X500AttrSyntax]
; X500AttributeType=X500SyntaxName

[PEMAlgNames]
; PEM Algorithm Name=X.500 Algorithm Name

[DOS Extensions]
BAT=34
C=36
TXT=15
CMD=34
.
.
.
IPK=43

[Mac Types]
01=XCEL,XLC3; MS Excel 3.0 Chart
02=XCEL,XLS3; MS Excel 3.0 Spreadsheet
03=XCEL,XLM3; MS Excel 3.0 Macrosheet
.
.
.
43=IMgr,IPkg; Informed Filler Package

[ASH Information]
ASHServer=1.2.3.4
ASHPort=710
ASHdn=cn = ASH Service, o=Your Company,c=US
```

```
[Login Strings]
Message=<None>

[FIPS Mode]
etadmapiAuth=DES-MAC,64,C7EAA250F7EA8013:CAST3-MAC,64,64,C4285C5D172
3EED6
entrustraAuth=DES-MAC,64,EF2A099462C4A1AB:CAST3-MAC,64,64,43A8410626
F539BE
FipsMode=1
EntadminAuth=DES-MAC,64,EF2A099462C4A1AB:CAST3-MAC,64,64,43A8410626F
539BE
entadminName=entrustRA
EtAdmapiName=etadmapi
```

Table 2: Entries in the entrust.ini file for administrative users

Entry	Description
[Entrust Settings]	A section header for entries that define general Entrust profile settings.
DefaultContext_time _req	An entry that specifies the length of time, in seconds, that the security context remains valid. Entrust/RA uses this entry to secure a session with the Administration subsystem. Note that this entry is also included in the entrust.ini file for end users, but is used differently in each file. The value "0" means that the security context remains valid forever. Default: 0
DefaultCredentials _time_req	An entry that specifies the length of time, in seconds, that credentials remain valid. Entrust/RA uses this entry is to secure a session with Administration subsystem. The value "0" means that credentials remain valid forever. Note that this entry is also included in the entrust.ini file for end users, but is used differently in each file. Default: 0
[Directory Connection Settings]	A section header for entries that affect the connection to the Directory. The entries in this section don't affect Entrust-Ready applications prior to Entrust/PKI 5.0.
LDAPModifyOperation	An entry that determines whether an LDAP "modify" operation is performed as a replacement or performed as two steps (a deletion and an addition). The value is "Replace" or "AddAndDelete". Default: No default

Table 2: Entries in the entrust.ini file for administrative users (continued)

Entry	Description
LDAPRename Operation	An entry that specifies how an LDAP "rename" operation is performed. In Entrust/PKI 5.0 and later, non-leaf entries cannot be renamed. The value is "ChangeParent", "ChangeRDN", or "none" (in this case, the DN change procedure that was in place before Release 5.0 is used). Default: ChangeParent
[ASH Information]	A section header for entries for the Administration subsystem.
ASH	An entry that specifies one or more values, where each value comprises an IP address (or DNS specification) and port number (for example, "ASH=host1+710). When attempting to connect to the Administration subsystem, Entrust/RA first tries to connect using the IP address (or DNS specification) defined by the "ASHServer" entry and the port defined by the "ASHPort" entry. If Entrust/RA can't connect using these values (for example, they have not been defined), it tries to connect using the values specified by the "ASH" entry. Each value defined for the "ASH" entry is separated by a comma (for example, "ASH=host1+710, host2+710,10.10.10.10+710"). Entrust/RA tries each value, in the order listed, until it successfully connects to the Administration subsystem. If Entrust/RA cannot connect using any value, an error is displayed. This entry is required to run Entrust/RA. It should never be included in the entrust.ini file for end users. Default: No default
ASHdn	An entry that specifies the DN of the Administration subsystem. This entry is written to the entrust.ini file during installation, and is required to run Entrust/RA. This entry should never be included in the entrust.ini file for end users. Default: No default
ASHPort	An entry that specifies the port number (for example, ASHPort=710) for the IP address (or DNS specification) defined by the "ASHServer" entry. This entry is written to the entrust.ini file during installation, and is required to run Entrust/RA. This entry should never be included in the entrust.ini file for end users. Default: No default

Table 2: Entries in the entrust.ini file for administrative users (continued)

Entry	Description
ASHServer	An entry that specifies an IP address or DNS specification (for example, ASHServer=11.11.11) for the Administration subsystem. When attempting to connect to the Administration subsystem, Entrust/RA first tries to connect using the IP address (or DNS specification) defined by the "ASHServer" entry and the port defined by the "ASHPort" entry. If Entrust/RA can't connect using these values (for example, they have not been defined), it tries to connect using the values specified by the "ASH" entry. This entry is written to the entrust.ini file during installation, and is required to run Entrust/RA. This entry should never be included in the entrust.ini file for end users. Default: No default
ASHTimeout	An entry that specifies the time, in seconds, that an Entrust-Ready or EntrustSession application will wait before its connection attempt times out. If a time-out occurs, all Administration subsystems may be busy. This entry is used by Entrust/RA. It should never be included in the entrust.ini file for end users. Default: 60
DebugFile	An entry that allows you to control whether Entrust/RA records various debug information. To cause Entrust/RA to record debug information, add this entry and specify the path to the debug file name. This entry is used by Entrust/RA. It should never be included in the entrust.ini file for end users. Default: No default
PKIVersion	An entry that improves the performance and memory requirements of the EADMSLogin function in the Entrust/RA Toolkit 5.0. This entry specifies that the etadmintk4.dll is not to be loaded. This entry is optional. Note that if PKIVersion=4, Entrust/RA Toolkit applications will first try etadmintk4.dll to see if the function called is available in the version 4.0 library. If it is not available, the application will access the function in the version 5.0 etadmintk.dll library. Default: 5

Customizing the entrustra.ini file

The entrustra.ini file contains entries that configure Entrust/RA. The installation program creates this file in the Windows folder (for example, C:\Winnt), by default, on the Entrust/Authority server.

You customize and distribute the entrustra.ini file to administrative users (for example, Security Officers, Administrators, and Directory Administrators). For information about distributing this file, see the chapter on distributing Entrust/RA and Entrust-Ready applications in *Installing Entrust/PKI 6.0 on Windows*.

You customize the entrustra.ini file by modifying its entries. Table 3 lists each entry in the entrustra.ini file and its default value (if any), and describes how the entry is used. "Example of an entrustra.ini file" on page 526 shows a typical entrustra.ini file.

Table 3: Entries in the entrustra.ini file

Entry	Description
;	A marker that indicates that the current line contains a comment. When using the entrustra.ini file, Entrust/RA ignores everything on a line starting with a semicolon. You can include as many comments as you want in an entrustra.ini file. For example, you can add a comment for each entry to describe why you set the entry to a particular value.
[Recent EPFs]	A section header for up to ten entries that identify recently used Entrust profiles (.epf).
0, 1, 2, 3, 4, 5, 6, 7, 8, 9	Entries that identify the most recently used Entrust profiles (.epf files). Each entry (0 to 9) identifies one profile, where the most recently used profile is "0". For example: 0=c:\entrust\authdata\manager\epf\Business Officer.epf 1=c:\entrust\authdata\manager\epf\Finance Officer.epf . . . 9=c:\entrust\authdata\manager\epf\Inventory Officer.epf The profile names in this list (for example, "Business Officer", "Finance Officer", and "Inventory Officer") appear in the drop-down lists in the Entrust/RA Login dialog box and the Authorizations Required dialog box. Default: No default
[Directory Search Criteria]	A section header for entries related to attribute names used for searches in the Directory Browser.

Table 3: Entries in the entrustra.ini file (continued)

Entry	Description
AttributeName*n*	Entries that specify attribute names. These names appear in the drop-down lists in the Find Directory dialog box under the Entries menu of the Directory Browser. For example, "AttributeName0=cn" means that the Directory Browser can search on the common name of the user. You must configure this information directly in the entrustra.ini file; it cannot be changed from Entrust/RA. The order of the attribute names is important; AttributeName0 is the entry that appears at the top of the drop-down list. There is no limit to the number of AttributeName*n* entries. Default: AttributeName0
count	An entry that specifies the number of AttributeName*n* entries. Default: 6
[User List]	A section header for entries that determine the default settings in the *Search Performance* property page and the *Directory Searches* property page of the Preferences dialog box. This dialog box appears when you click File > Preferences in Entrust/RA.
GetAuthorityInfo	An entry that specifies the default number of users Entrust/RA retrieves during a Directory user search by Directory attributes. The value appears as the *Number of users* under *Search by Directory Attributes* on the *Search Performance* property page of the Preferences dialog box. There is no maximum value. Default: 20
MaxNumberFor DatabaseSearch	An entry that specifies the default number of users Entrust/RA retrieves during a Directory user search by Entrust properties. The value appears as the *Number of users* under *Search by Entrust Properties* on the *Search Performance* property page of the Preferences dialog box. The value is an integer between 1 and 500. Default: 300
Users	An entry that specifies the number of users Entrust/RA retrieves during a Directory user search. The value appears as the *Number of users to retrieve* in the *Directory Searches* property page of the Preferences dialog box. The value is an integer between 1 and 500. Default: 300

Table 3: Entries in the entrustra.ini file (continued)

Entry	Description
[General Options]	A section header for entries that determine the default settings in the *General* property page of the Preferences dialog box. This dialog box appears when you click File > Preferences in Entrust/RA.
ChangeDNFormat	An entry that allows you to configure Entrust/RA to preselect one of *Keep old entry in Directory* or *Rename existing Directory entry* for all Change DN operations. If you set ChangeDNFormat=0 or if ChangeDNFormat does not appear at all in the entrustra.ini file, Entrust/PKI uses the values from the DirectoryProtocol and LDAPRenameOperation entries in the entrust.ini file. For more information about these entries and the entrust.ini file, refer to *Administering Entrust/PKI 6.0 on Windows*. If you set ChangeDNFormat=1, you will not see the radio buttons in the Change DN dialog box in Entrust/RA. You will keep the existing entry, if the user is not a roaming user. (If the user is a roaming user, the value of this entry will be ignored.) If you set ChangeDNFormat=2, you will not see the radio buttons in the Change DN dialog box in Entrust/RA. Your entry will be renamed if the Directory settings allow the entry to be renamed. If the Directory settings do not allow the entry to be renamed, this value will be interpreted by Entrust/RA as though it were set to ChangeDNFormat=0. Default: 0
DefaultCategory	An entry that determines the default certificate category when adding users. The value is "Enterprise", "Web". The value appears in the *Default Certificate Category* drop-down list in the *General* property page of the Preferences dialog box. Default: Enterprise
ReadOnlyDirectoryBrowser	An entry that determines whether the Directory Browser should be launched in read-only mode from Entrust/RA. When this entry is set to 1, the following operations are not available from the main menu, the toolbar or popup menus in the Directory Browser: • Add entry • Delete entry • Add attributes • Bulk • Change Directory Administrator password Default: 0, except when Active Directory is used, in which case the default is 1

Table 3: Entries in the entrustra.ini file (continued)

Entry	Description
Warnings	An entry that determines whether warning messages appear during certain operations in Entrust/RA. For example, when you click *Apply* when adding a user, a warning message appears telling you that you cannot restore the user's previous settings. The value is "0" (no) or "1" (yes). The value determines the setting of the *Show warnings about operations* check box in the *General* property page of the Preferences dialog box. Default: 1
[DirAdmin Defaults]	A section header for entries that determine the default settings in the *Directory Searches* property page of the Preferences dialog box. This dialog box appears when you click File > Preferences in Entrust/RA.
Root Searchbase	An entry that names the CA's default searchbase. This value determines under what entry searches are performed in the Directory. The Root Searchbase value can be any of the existing searchbases. This value also determines the default setting of the *Initial Searchbase* drop-down list in the *General* property page of the Preferences dialog box. Default: No default
[Bulk Console Defaults]	A section header for entries that are used when using Entrust/PKI 5.0 (and later) bulk operations.
InputPath	An entry that specifies the default location shown in the *Input* field of the Bulk Console window when choosing an .entra bulk file in Entrust/RA. Default: No default
OutputPath	An entry that specifies the default location shown in the *Output* field of the Bulk Console window when choosing an .entra bulk file in Entrust/RA. Default: No default
[Entrust ini override]	A section header for entries that identify which entrust.ini file should be read when a user logs in to Entrust/RA using a particular Entrust profile. For example, you can specify that one entrust.ini file should be read when a user logs in to Entrust/RA using the "Finance Officer" profile and another entrust.ini file should be read when a user logs in to Entrust/RA using the "Business Officer" profile. You must add this section and its entries manually.

Table 3: Entries in the entrustra.ini file (continued)

Entry	Description
profile name	Entries that map a profile name to an entrust.ini file. For example: c:\entrust\authdata\manager\epf\Business Officer.epf=X:\Entrust\profiles\business_officer\entrust.ini c:\entrust\authdata\manager\epf\Finance Officer.epf=Y:\Entrust\profiles\finance_officer\entrust.ini
[Settings]	A section that keeps the values of window settings for Entrust/RA and Directory Browser from session to session. The current values are written to the entrustra.ini file each time you close Entrust/RA and Directory Browser. These entries aren't included in the entrustra.ini file during installation. They are written to the file the first time that you close Entrust/RA and the Directory Browser.
Admin Main Window Settings	An entry for settings related to the Entrust/RA window. This entry determines • the location of the bar that divides the Entrust/RA window into a left pane and a right pane. • the left, top, right, and bottom position of the window. • whether the toolbar is visible. The value is "0" (no) or "1" (yes). • whether the status bar is visible. The value is "0" (no) or "1" (yes). The values always appear in the order that the settings are listed above. For example, the value of the first setting always represents the location of the bar that divides the Entrust/RA window into two panes. The value of the last setting always determines whether the status bar is visible. The following example shows the default values for this entry: Admin Main Window Settings=250,0,0,750,520,1,1

Table 3: Entries in the entrustra.ini file (continued)

Entry	Description
DirAdmin Window Settings	An entry for settings related to the Directory Browser window. This entry determines • the location of the bar that divides the Directory Browser window into a left pane and a right pane. • the left, top, right, and bottom position of the window. • whether the toolbar is visible. The value is "0" (no) or "1" (yes). • whether the status bar is visible. The value is "0" (no) or "1" (yes). The values always appear in the order that the settings are listed above. For example, the value of the first setting always represents the location of the bar that divides the Directory Browser window into two panes. The value of the last setting always determines whether the status bar is visible. The following example shows the default values for this entry: DirAdmin Window Settings=292,25,25,775, 525,1,1

Example of an entrustra.ini file

```
[Recent EPFs]
0=C:\authdata\manager\epf\First Officer.epf
1=
2=
3=
4=
5=
6=
7=
8=
9=
Default to MRU=1

[Directory Search Criteria]
count=6
AttributeName0=cn
AttributeName1=sn
AttributeName2=objectClass
AttributeName3=organizationalUnitName
AttributeName4=serialnumber
AttributeName5=mail


[User List]
Users=500
GetAuthorityInfo=50

[General Options]
Restore on Startup=0
Warnings=0
Default Category=Enterprise

[DirAdmin Defaults]
Root Searchbase=CA Domain Searchbase

[Settings]
Admin Main Window Settings=250,11,7,738,527,1,1
DirAdmin Window Settings=292,46,20,796,520,1,1
```

Customizing the entmgr.ini file

The entmgr.ini file specifies important configuration data for Entrust/Authority. The installation program creates this file in the C:\authdata\manager folder, by default, on the Entrust/Authority server.

Some entries are included in the entmgr.ini file only until you initialize Entrust/PKI. For information about these entries, see *Installing Entrust/PKI 6.0 on Windows*.

Because only Entrust/Authority uses this file, you don't need to distribute it. However, you may want to customize it by modifying its entries. Table 4 lists each entry in the entmgr.ini file and its default value (if any), and describes how the entry is used. "Example of an entmgr.ini file" on page 547 shows a typical entmgr.ini file.

Table 4: Entries in the entmgr.ini file

Entry	Description
;	A marker that indicates that the current line contains a comment. When using the entmgr.ini file, Entrust-Ready applications ignore everything on a line starting with a semicolon. You can include as many comments as you want in an entmgr.ini file. For example, you can add a comment for each entry to describe why you set the entry to a particular value.
[Entrust Settings]	A section header for entries that define general Entrust/Authority settings.
CryptokiLibraryNT	An entry that specifies the file path to the CRYPTOKI library (the software .dll for the Entrust/Authority hardware device) on Windows NT. This entry is used only if hardware is used to store the CA private keys. The entry is written during installation if hardware is chosen and must be added manually if the CA keys are initially installed in software and then moved to hardware. Default: No default
CryptokiV2Library	An entry that specifies the file path to the CRYPTOKI library (the software .dll for the Entrust/Authority hardware device) on Windows 2000. This entry is used only if hardware is used to store the CA private keys. The entry is written during installation if hardware is chosen and must be added manually if the CA keys are initially installed in software and then moved to hardware. Default: No default

Table 4: Entries in the entmgr.ini file (continued)

Entry	Description
[Comms]	A section header for entries that affect the Administration subsystems.
ASHlisten	An entry that specifies the port on which the Administration subsystem listens for Entrust-Ready applications attempting to connect to the server. This entry is set during installation. Default: 710
CMPlisten	An entry that indicates the port on which the PKIX-CMP subsystem listens for key management requests. This entry is set during installation. Default: 829
DirectoryProtocol	An entry that specifies the version of the LDAP protocol Entrust/Authority uses to connect to the Directory. This entry is set during installation. The value is "LDAPv2" or "LDAPv3". If Entrust/Authority cannot connect using this value, the connection fails. Default: LDAPv2
ListenTimeout	An entry that determines the period of time in seconds that Entrust/Authority will wait for a client response to SEP and PKIX-CMP requests. Default: 240 seconds
Mgrlisten	An entry that indicates the port on which the SEP subsystem listens for key management requests. This entry is set during installation. Default: 709
ProtocolOverride	An entry that determines whether a subordinate CA accepts or denies SEP requests. When a SEP request is denied, an audit is generated that contains the DN of the user that made the request. The value is "SEP-ON" (accept) or "SEP-OFF" (deny). Default: No default
Server	An entry that specifies the IP address and port of the Directory server used by Entrust/Authority. This entry is set during installation. Default: No default

Table 4: Entries in the entmgr.ini file (continued)

Entry	Description
ServerType	An entry that specifies whether the installation communicates with the Directory in binary mode or ASCII mode. This entry is set during installation. The value is "External" (binary mode), "EntrustSLAPD" (ASCII mode), or "EntrustLDAP" (ASCII mode). Default: External
[Login]	A section header for entries that correspond to general options in Entrust/Authority Master Control.
AshDisabled	An entry that specifies whether the Administration subsystem starts when the Entrust/Authority service is started. The value is "True" (no) or "False" (yes). Default: False
AshNumProcs	An entry that determines the number of processes created for the Administration subsystem. For small systems only one or two processes are required. For large systems, more processes may be necessary to improve performance or to manage large numbers of simultaneous Entrust users. The minimum value is "1" and the maximum value is "256". Default: 4
AutoRestartIdleTime	An entry that specifies the length of time that must pass without a request before the Administration, CMP, and SEP subsystems restart. The value is specified as a number of days (for example, "AutoRestartIdletime=1" indicates one day) or as hours and minutes with a maximum of 24 hours (for example, "AutoRestartIdletime=00:15" indicates fifteen minutes). The AutoRestartIdletime entry works together with the following entries: AutoRestartBefore, AutoRestartNotAfter, and AutoRestartPeriod. Note that prior to Entrust/PKI Release 5.0, this entry was named "EntAdmSrvIdleTime". The AutoRestartIdleTime entry replaces the EntKeySrvIdleTime, EntKeyMgrIdleTime, and EntAdmSrvIdleTime entries. Default: 0

Table 4: Entries in the entmgr.ini file (continued)

Entry	Description
AutoRestartNotBefore, AutoRestartNotAfter	Entries that define the earliest and latest times that the Administration, CMP, and SEP subsystems restart. By default, these subsystems restart between 3:00 a.m. and 5:00 a.m. The values for these entries are specified as hours and minutes with a maximum of 24 hours (for example, "AutoRestartNotBefore=14:02" indicates two minutes after 2 p.m.). The AutoRestartNotBefore and AutoRestartNotAfter entries work together with the AutoRestartIdleTime and AutoRestartPeriod entries. Note that prior to Entrust/PKI Release 5.0, the AutoRestartNotBefore entry was named "EntAdmSrvStartTimeout" and the AutoRestartNotAfter entry was named "EntAdmSrvEndTimeout". Defaults: AutoRestartNotBefore=00:00 AutoRestartNotAfter=23:59
AutoRestartPeriod	An entry that specifies how often subsystem processes restart. For maintenance purposes, the subsystem processes must restart regularly. By default, the subsystem processes restart once per day. The value is specified as a number of days (for example, AutoRestartPeriod=1 indicates one day) or as hours and minutes with a maximum of 24 hours (for example, "AutoRestartPeriod=00:05" indicates five minutes). The AutoRestartPeriod entry works together with the AutoRestartNotBefore, AutoRestartNotAfter, and AutoRestartIdleTime entries. Note that prior to Entrust/PKI 5.0, this entry was named "EntAdmSrvTimeout". The AutoRestartPeriod entry replaces the EntKeySrvTimeout and EntKeyMgrTimeout entries. Default: 1
CMPDisabled	An entry that specifies whether the PKIX-CMP subsystem starts when the Entrust/Authority service is started. The value is "True" (no) or "False" (yes). Default: False

Table 4: Entries in the entmgr.ini file (continued)

Entry	Description
CMPNumProcs	An entry that determines the number of processes created for the PKIX-CMP subsystem. For small systems only one or two processes are required. For large systems, more processes may be necessary to improve performance or to manage large numbers of simultaneous Entrust users. The minimum value is "1" and the maximum value is "256". Default: 2
DbLoginName	An entry that defines the user ID (for example, "Informix") used to connect to the Entrust/Authority database. This entry is set during installation when using an Oracle database. It is also set during installation when using an Informix database for Entrust/PKI on UNIX prior to Release 5.0, and Entrust/PKI on Windows prior to Release 6.0. If this entry is not set, the identity of the current UNIX or Windows user ID is used to connect to the database. Default: No default
DbLoginPw	An entry that defines the password used to connect to the Entrust/Authority database. This entry is set during installation when using an Oracle database. It is also set during installation when using an Informix database for Entrust/PKI on UNIX prior to Release 5.0, and Entrust/PKI on Windows prior to Release 6.0. If this entry is set and no DbLoginPw entry exists in the authauto.ini file, the DbLoginPw entry is encrypted and moved to the authauto.ini file. If this entry is not set, the identity of the current UNIX or Windows user ID is used to connect to the database. To change this password, enter the new password here; Entrust/Authority automatically encrypts the plaintext password. Default: No default
DisableDbBackup	An entry that is used for multiple CAs only. This entry is set for only one of the CAs by the installation program, and causes the automatic database backup at the end of the installation program not to occur. The value is "0" (no) or "1" (yes). Default: 1

Table 4: Entries in the entmgr.ini file (continued)

Entry	Description
EntBackupBackupsTo KeepOnline	An entry that specifies the number of online backups of the database to store. The minimum value is "1" and the maximum value depends on the amount of space available for backups. Default: 7
EntBackupNotBefore, EntBackupNotAfter	Entries that define the earliest and latest time that an automatic database backup occurs. By default, an automatic backup occurs between the hours of 3:00 a.m. and 5:00 a.m. The values for these entries are specified as hours and minutes with a maximum of 24 hours (for example, "EntBackupNotBefore=14:02" indicates two minutes after 2 p.m). The EntBackupNotBefore and EntBackupNotAfter entries work together with the EntBackupPeriod entry. Note that prior to Entrust/PKI Release 5.0, the EntBackupNotBefore entry was named "EntBackupStartTime" and the EntBackupNotAfter entry was named "EntBackupNotAfter". Defaults: EntBackupNotBefore=03:00 EntBackupNotAfter=05:00
EntBackupPeriod	An entry that specifies how often an automatic database backup occurs. By default, an automatic backup occurs once per day, which is the frequency recommended by Entrust Technologies. The value is specified as a number of days (for example, "EntBackupPeriod=1 indicates one day) or as hours and minutes with a maximum of 24 hours (for example, EntBackupPeriod=00:05 indicates five minutes). You can change this value by modifying the backup settings in Entrust/Authority Master Control. The new value that you specify is written to the entmgr.ini file. The EntBackupPeriod entry works together with the following entries: EntBackupNotBefore and EntBackupNotAfter. Note that prior to Entrust/PKI 5.0, this entry was named "EntBackupFrequency". Default: 24:00

Table 4: Entries in the entmgr.ini file (continued)

Entry	Description
EntCrlCheckPeriod	An entry that determines how often Entrust/Authority checks whether CRLs need to be written to the DSA. By default, Entrust/Authority performs this check every 30 minutes. The value is specified as a number of days (for example, "EntCrlCheckPeriod=1" indicates one day) or as hours and minutes with a maximum of 24 hours (for example, "EntCrlCheckPeriod=00:05" indicates five minutes). Default: 00:30 (every 30 minutes)
EntDailyAuditNotBefore, EntDailyAuditNotAfter	Entries that define the earliest and latest time that daily audits occur. By default, daily audits occur between 12:00 midnight and 11:59 p.m. The values for these entries are specified as hours and minutes with a maximum of 24 hours (for example, "EntDailyAuditNotBefore=14:02" indicates two minutes after 2 p.m). The EntDailyAuditNotBefore and EntDailyAuditNotAfter entries work together with the EntDailyAuditPeriod entry. Defaults: EntDailyAuditNotBefore=00:00 EntDailyAuditNotAfter=23:59
EntDailyAuditPeriod	An entry that specifies how often daily audits occur. By default, an incremental backup is performed once per hour. The value is specified as a number of days (for example, "EntDailyAuditPeriod=1" indicates one day) or as hours and minutes with a maximum of 24 hours (for example, "EntDailyAuditPeriod=00:05" indicates five minutes). This value is written to the entmgr.ini file when you modify the backup settings using Entrust/Authority Master Control. The EntDailyAuditPeriod entry works together with the EntDailyAuditNotBefore and EntDailyAuditNotAfter entries. Default: 01:00

Table 4: Entries in the entmgr.ini file (continued)

Entry	Description
EntIncBackupNot Before, EntIncBackupNotAfter	Entries that define the earliest and latest time that incremental backups occur. By default, incremental backups occur between 12:00 midnight and 11:59 p.m. The values for these entries are specified as hours and minutes with a maximum of 24 hours (for example, "EntIncBackupNotBefore=14:02" indicates two minutes after 2 p.m). The EntIncBackupNotBefore and EntIncBackupNotAfter entries work together with the EntIncBackupPeriod entry. Defaults: EntIncBackupNotBefore=00:00 EntIncBackupNotAfter=23:59
EntIncBackupPeriod	An entry that specifies how often incremental backups occur. By default, an incremental backup is performed once per hour. The value is specified as a number of days (for example, "EntIncBackupPeriod=1" indicates one day) or as hours and minutes with a maximum of 24 hours (for example, "EntIncBackupPeriod=00:15" indicates fifteen minutes). This value is written to the entmgr.ini file when you modify the backup settings using Entrust/Authority Master Control. The EntIncBackupPeriod entry works together with the EntIncBackupNotBefore and EntIncBackupNotAfter entries. Default: 01:00
EntIntegrityNotBefore, EntIntegrityNotAfter	Entries that define the earliest and latest time that automatic integrity checks occur. By default, automatic integrity checks occur between 2:00 a.m. and 5:00 a.m. The values for these entries are specified as hours and minutes with a maximum of 24 hours (for example, "EntIntegrityNotBefore=14:02" indicates two minutes after 2 p.m). The EntIntegrityNotBefore and EntIntegrityNotAfter entries work together with the EntIntegrityPeriod entry. Defaults: EntIntegrityNotBefore=02:00 EntIntegrityNotAfter=05:00

Table 4: Entries in the entmgr.ini file (continued)

Entry	Description
EntIntegrityPeriod	An entry that specifies how often automatic integrity checks occur. By default, an automatic integrity check is performed once every 24 hours. The value is specified as a number of days (for example, "EntIntegrityPeriod=1" indicates one day) or as hours and minutes with a maximum of 24 hours (for example, "EntIntegrityPeriod=00:05" indicates five minutes). This value is written to the entmgr.ini file when you modify the backup settings using Entrust/Authority Master Control. The EntIntegrityPeriod entry works together with the EntIntegrityNotBefore and EntIntegrityNotAfter entries. Note that prior to Entrust/PKI 5.0, this entry was named "EntBackupDoIntegrityCheck". As well, the frequency of automatic integrity checks in Entrust/Authority 4.0 was controlled by the "backup" entry. Default: 24:00
EntLicenseThreshold	An entry that causes an audit log to be generated during the daily audit when the percentage of available Entrust licenses falls below the specified threshold. The value for EntLicenseThreshold is an integer expressing a percentage. If the percentage of available licenses left is lower than the threshold specified, Audit log 7641 is generated. Setting EntLicenseThreshold to 0 means that Audit log 7641 will never be generated. Setting this entry to a value of 100 or more means that Audit log 7641 would be generated every time the daily audit occurs. See also "EntDailyAuditPeriod" on page 533. Default: 0
EntMgrMinFree	An entry that provides a value for the minimum required amount of free disk space in megabytes. If there is less than that number of megabytes of free disk space on the hard drive where Entrust/Authority is installed, Entrust/Authority will not start. An audit log is produced when space falls below the value of the EntMgrWarnFree entry. Default: 5

Table 4: Entries in the entmgr.ini file (continued)

Entry	Description
EntMgrWarnFree	An entry that provides a value for the minimum number of megabytes of free disk space before a warning appears. If there is less than that number of megabytes of free disk space on the hard drive where Entrust/Authority is installed, messages will be added to the audit log when any Entrust/Authority subsystem or Entrust/Authority Master Control is started. Default: 20
entUserAttrCertReissue NotBefore	An entry that specifies when the periodic process will begin to check for attribute certificates that are about to expire. Default: 00:00
entUserAttrCertReissue NotAfter	An entry that specifies when the periodic process will stop checking for attribute certificates that are about to expire. Default: 23:55
entUserAttrCertReissue Period	An entry that specifies the period for checking for attribute certificates that are about to expire. By default, the periodic process checks when the PKI starts and will check every 20 hours. Note that the entUserAttrCertReissuePeriod should be less than the Reissue Delta (in the [User Attribute Cert] section) to ensure that certificates are reissued before they expire. Default: 20:00

Table 4: Entries in the entmgr.ini file (continued)

Entry	Description
EntUserStateHistTrim IdleTime	An entry that specifies the length of time that must pass without a request before Entrust/Authority can begin trimming the UserStateHist database table. The purpose of this parameter is to ensure higher priority work is not interrupted. By default, the UserStateHist table is trimmed after 1 minute of inactivity. The UserStateHist table is maintained in the Entrust/Authority database. "Trimming" this table discards data that is older than a specified number of days, which helps to minimize the size of the database. All events are recorded in the audit trails, which should be archived. The frequency should be determined according to a trade-off between how busy Entrust/Authority is and how often you wish to remove old entries and trim down the database. The value is specified as hours and minutes with a maximum of 24 hours (for example, "EntUserStateHistTrimIdleTime=14:02" indicates fourteen hours and two minutes). Default: 00:01
EntUserStateHistTrim Max	An entry that specifies the maximum number of days to keep data in the UserStateHist database table. If you change the value of this entry, ensure that you keep enough historical data for reporting purposes. Default: 186
EntUserStateHistTrim NotBefore, EntUserStateHistTrim NotAfter	Entries that specify the earliest and latest time of day that Entrust/Authority can begin trimming the UserStateHist table. The values for these entries are specified as hours and minutes with a maximum of 24 hours (for example, "EntUserStateHistTrimNotBefore=14:02" indicates two minutes after 2 p.m). For example, using the default values, the UserStateHist table can be trimmed between the hours of 3:00 a.m. and 10:00 a.m. The EntUserStateHistTrimNotBefore and EntUserStateHistTrim NotAfter entries work together with the EntUserStateHistTrimPeriod entry. Note that prior to Entrust/PKI 5.0, this entry was named "EntUserStateHistTrimStartTime". Defaults: EntUserStateHistTrimNotBefore=03:00 EntUserStateHistTrimNotAfter=10:00

Table 4: Entries in the entmgr.ini file (continued)

Entry	Description
EntUserStateHistTrim Period	An entry that specifies how often Entrust/Authority trims the UserStateHist table. By default, Entrust/Authority trims the UserStateHist table once per day. The value is specified as a number of days (for example, "EntUserStateHistTrimPeriod=1" indicates one day) or as hours and minutes with a maximum of 24 hours (for example, "EntUserStateHistTrimPeriod=12:00" indicates twelve hours). The EntUserStateHistTrimPeriod entry works together with the EntUserStateHistTrimNotBefore and EntUserStateHistTrimNotAfter entries. Note that prior to Entrust/PKI 5.0, this entry was named "EntUserStateHistTrimFrequency". Default: 24:00
KeyCacheSize	An entry that defines the number of RSA keys that will be pre-generated and stored in the key cache to improve burst CMP/SEP performance. Default: 10
ManagerBackup	An entry that specifies the location of online backups. This entry is set during installation. Default: c:\entbackup
MaxRestarts	An entry that specifies the number of times that any subsystem can fail to start successfully before all services are shut down. Default: 10
OfflineSuperior	By default, Entrust/PKI periodically checks its own CA certificate. If the superior CA's revocation list is unavailable (for example, if the root CA is offline), the validation will fail and the CA will issue an audit. While these audits do not cause any operational problems, they do clutter the audit logs. To prevent this audit log from being generated, set OfflineSuperior to "1". Default: 0

Table 4: Entries in the entmgr.ini file (continued)

Entry	Description
ReLoadCAKeysPeriod	An entry that specifies how often Entrust/Authority checks whether CA keys are reloaded. The value is specified as a number of days (for example, "ReLoadCAKeysPeriod=1" indicates one day) or as hours and minutes with a maximum of 24 hours (for example, "ReLoadCAKeysPeriod=00:05" indicates five minutes). Default: 00:30
SEPDisabled	An entry that specifies whether the Entrust Secure Exchange Protocol (SEP) subsystem starts when the Entrust/Authority service is started. The value is "True" (no) or "False" (yes). Default: False
SEPNumProcs	An entry that determines the number of processes created for the Entrust Secure Exchange Protocol (SEP) subsystem. For small systems only one or two processes are required. For large systems, more processes may be necessary to improve performance or to manage large numbers of simultaneous Entrust users. Default: 2

Table 4: Entries in the entmgr.ini file (continued)

Entry	Description
userSearchExplicit WildcardPrefix	An entry that affects the behavior of the Search By Entrust Properties feature of Entrust/RA and any Entrust/RA Toolkit searches. When this setting is set to false, the name entered in the distinguished name (DN) field is treated as if it was prefixed with an asterisk. For example, when userSearchExplicitWildcardPrefix=false, Bob Jo is equivalent to *Bob Jo* and you wouldn't have to type the "cn=" to search through all DNs to find "Bob Jones" and "Bob Johnson". In this example, the search would return all users with the first name Bob and a last name starting with Jo. If this option were set to true, and you wanted to find Bob Jones and Bob Johnson, you'd need to search on cn=Bob Jo Use of this setting is highly recommended since it can speed up searches by DN when set to true. See also "Modifying Entrust/RA to improve database searches" on page 71. Default: False
userSearchOneScreen	An entry that, when set to true, causes only one page of results to be returned for an Entrust/RA search. This option also affects the order in which users are returned. When set to false, users are returned in the order in which they were added to the database, and this ordering is used to facilitate subsequent page searches. However, this ordering also requires sorting. Lack of free disk space can restrict the ability of the search to be optimized. Setting this option to true avoids the ordering requirement and permits more efficient searches. See also "Modifying Entrust/RA to improve database searches" on page 71. Default: False

Table 4: Entries in the entmgr.ini file (continued)

Entry	Description
Watchdog	An entry that specifies the length of time, in seconds, that a request has to respond to internal check before it is terminated and restarted. This entry protects against Entrust Services or Entrust subsystem implementation errors. Default: 300
[Object Class]	This section contains entries for use with Active Directory related to object classes.
OrgRoleFirstOfficer	An entry that configures Entrust/Authority to create the First Officer as an organizationalRole entry instead of an organizationalPerson (the default object class). Active Directory defines organizationalPerson as an abstract object class, so organizationalPerson cannot be instantiated. This entry is added by default to the entrust.ini file during configuration of Entrust/PKI 6.0 with Active Directory only. Default: 1 when using Active Directory; 0 for other Directories
[Policy]	This section contains security-related settings. Some settings are present only after installation and are moved to the database and removed from this section after Entrust/Authority is started for the first time.
IgnorePKCS10 Signature	An entry that determines whether Entrust/Authority attempts to verify digital signatures when processing a PKCS #10 message to create a cross-certificate. Default: 0
UseEntrustPolicyObject OC	An entry that specifies whether the entrustPolicyObject and pmiUser auxiliary object classes are used. Active Directory does not support the use of auxiliary object classes to dynamically extend the schema of an entry. Normally, prior to writing an entrustPolicyCertificate to the Directory, the object class entrustPolicyObject is added to the entry, and prior to writing an attribute certificate, the pmiUser object class is added. If Active Directory is used, UseEntrustPolicyObjectOC should be set to "0" to disable both of these write operations. This entry is added by default to the entrust.ini file during configuration of Entrust/PKI 6.0 with Active Directory only. Default: 0 when using Active Directory, 1 for other Directories
[Authorization]	A section header for entries about Entrust license information.

Table 4: Entries in the entmgr.ini file (continued)

Entry	Description
SerialNumber	An entry that specifies the serial number for Enterprise licences. This serial number is located in your licensing information that you received from your Entrust sales representative. This information is changed through Entrust/RA. See "Setting a new user limit" on page 60. This entry is set during installation. Default: No default
UserLimit	An entry that specifies the number of Enterprise licences. Each Enterprise user requires one license. This number is located in the licensing information that you received from your Entrust sales representative. This information is changed through Entrust/RA. See "Setting a new user limit" on page 60. This entry is set during installation. Default: No default
Key	An entry that specifies the license key that authorizes your Enterprise user limit. This information is changed through Entrust/RA. See "Setting a new user limit" on page 60. This entry is set during installation. Default: No default
WebSerialNumber	An entry that specifies the serial number for Web licences. This serial number is located in the licensing information that you received from your Entrust sales representative. This information is changed through Entrust/RA. See "Setting a new user limit" on page 60. This entry is set during installation. Default: No default
WebUserLimit	An entry that specifies the number of Web licences. Each Web user requires one license. This number is located in the licensing information that you received from your Entrust sales representative. This information is changed through Entrust/RA. See "Setting a new user limit" on page 60. This entry is set during installation. Default: No default
WebKey	An entry that specifies the license key that authorizes your Web user limit. This information is changed through Entrust/RA. See "Setting a new user limit" on page 60. This entry is set during installation. Default: No default
[database]	A section header for entries that specify information about the Entrust/Authority database.

Table 4: Entries in the entmgr.ini file (continued)

Entry	Description
dbname	An entry that specifies the name of the Entrust/Authority database. The value is taken from the ODBC Windows Control Panel. This entry is set during installation. Default: entrustv5_odbc
[Superior]	A section header for information for subordinate CAs. If the CA is not being set up as a subordinate, the entire section and associated entries are not present. If this section is present, it is removed after first-time startup of Entrust/Authority.
SuperiorAlgID	This entry specifies the signing algorithm that must be obtained from the superior CA to indicate how the subordinate CA certificate is to be signed. This entry exists only when your subordinate CA will be set up online. The values depend on the superior CA. If the superior CA is an Entrust CA, the values is "sha1WithRSAEncryption#NULL", "dsa-with-sha1X957", or "md5WithRSAEncryption#NULL". Default: No default
SuperiorCAAvailability	An entry that indicates whether a subordinate CA will be set up online (using PKIX-CMP) or offline (using PKCS #10 and PKCS #7). This entry is set during installation. The value is "online" or "offline". Default: online
SuperiorDN	An entry that specifies the DN of the superior CA. This entry exists only if you are using a strict hierarchy and the subordinate CA will be set up online. This must be a valid distinguished name. This entry is set during installation. Default: No default
SuperiorManager	An entry that specifies the IP address and port of the superior CA (for example, "SuperiorManager=1.2.3.4+355"). This entry exists only if you are using a strict hierarchy and the subordinate CA will be set up online. Default: No default
[Logging]	A section header for entries that are used to duplicate audit logs that are sent to the Windows Event Viewer.

Table 4: Entries in the entmgr.ini file (continued)

Entry	Description
Facility	This entry controls which local facility will be used to identify the audit logs sent to syslog. The value is an integer from 0 to 7. Note that this entry is for UNIX only. Windows users should not change this entry. Default: 0
Filesize	An entry that specifies the maximum size, in kilobytes, of the Entrust/Authority log files: manager.log, mgraudit.log, and audit.<date>. Entrust/Authority errors are written to the manager.log file until this file exceeds the size specified by Filesize. Manager.log is then renamed to manager.log-0 and a new manager.log file is started. When this file in turn reaches the maximum specified file size, manager.log-0 is renamed to manager.log-1, and manager.log is renamed to manager.log-0. This process continues up to the number of files specified in NumFiles (see "NumFiles" on page 545). Once the number of files reaches the value of NumFiles, the oldest file is deleted each time a new manager.log file is created. The audit.<date> files are never automatically deleted. The minimum value for Filesize is 100 Kbytes. Audits are written in a readable form to mgraudit.log. Audits in a tamper-proof binary form are written to audit.<date>. Default: 1024
Level	An entry that determines whether audit logs are displayed in the Windows Event Viewer. If audit logs are enabled, this entry also specifies the type of audit logs (that is, alarms, events, or logs) that are sent to the Windows Event Viewer. The value is one of the following: • 0 (send no audit logs to Event Viewer) • 1 (send alarms to Event Viewer) • 2 (send alarms and events to Event Viewer) • 3 (send alarms, events, and logs to Event Viewer) Default: 0

Table 4: Entries in the entmgr.ini file (continued)

Entry	Description
NumFiles	An entry that determines the maximum number of manager.log files that will be retained. Once the number of files reaches the value of NumFiles, the oldest file is deleted each time a new manager.log file is created. See also "Filesize" on page 544. The minimum value for NumFiles is 1. Default: 5
[ASH Information]	A section header for entries that configure the Entrust/Authority service.
Session Disconnect	An entry that indicates whether the Entrust/Authority service should disconnect from the Entrust/RA when it has been idle for the time period specified by the "Session Timeout" entry. The value is "0" (no) or "1" (yes). Default: 1
Session Timeout	An entry that indicates the time, in seconds, that the Entrust/Authority service will wait before it disconnects from a commented client. This entry is valid only if "Session Disconnect=1". Default: 120
[Auto]	A section header for an entry for automatic login information.
autoFile	An entry that determines the location of a file that contains automatic login information. Use this entry if a cluster is being used to implement high availability. The file should be stored in a private location on each machine and should not be mirrored. Autologin should be enabled separately on each machine. You can specify an absolute or relative pathname. Relative pathnames are interpreted relative to the location of the "authdata" folder (by default, C:\authdata). Default: C:\authdata\manager\authauto.ini
[Default Variable Values]	A section containing default values for user policies. For more information, see "Setting up default values" on page 300.
[User Attribute Cert]	A section header for entries controlling how user attribute certificates are written to the Directory.
Attribute Name	An entry that specifies the attribute to which attribute certificates are written in the Directory. Default: attributeCertificateAttribute

Table 4: Entries in the entmgr.ini file (continued)

Entry	Description
Object Class	An entry that determines what auxiliary object class should be added to the user's Directory entry to permit the inclusion of attribute certificates in the user's entry. If the UseEntrustPolicyObjectOC entry in the [Policy] section is set to 0, the object class will not be added to the entry. Default: pmiUser
Reissue Block	An entry that determines the number of attribute certificates renewed in a block. To ensure that the periodic process does not spend all of its time reissuing attribute certificates, it reissues the certificates in blocks. After each block, it will perform other periodic functions (such as reissuing CRLs) before reissuing the next block of attribute certificates. Default: 100
Reissue Delta	An entry that determines when attribute certificates are reissued. By default, attribute certificates are reissued when they are within 24 hours of expiring. The value can be specified as hours and minutes (hh:mm) or days (dd). Note that the Reissue Delta should be greater than the value for entUserAttrCertReissuePeriod (see "entUserAttrCertReissuePeriod" on page 536) to ensure that certificates are reissued before they expire. Default: 24:00
[CDP]	A section header for entries that configure CRL distribution points. To allow for system scalability and more efficient processing, Entrust/Authority maintains standardized, partitioned certificate revocation lists (CRLs) at unique "distribution points" in the Directory. Each certificate contains a pointer to one or more CRL distribution points (CDPs) where applications can find the CRL for the certificate in question. To learn how to specify additional CRL distribution points, see "Specifying additional CRL distribution points" on page 83.

Table 4: Entries in the entmgr.ini file (continued)

Entry	Description
[Reports]	A section header for entries related to formatting of user reports.
Time Format	An entry that allows you to customize the way dates and times are formatted in the user reports generated by Entrust/RA. This entry does not affect any the format of any other dates or times in Entrust/RA. If you require a date format like the following: 05.09.2000 16:12:13 then set Time Format=%m.%d.%Y %H:%M:%S In this example, <format> is a string made up of the following components: %Y for year %m for month %d for day %H for hour %M for minute %S for seconds You can change the order of the time components, and you do not have to include all of them in the <format>. If you don't specify the time format, the following format will be used: Tue Sep 05 16:12:13 2000 As well, the <format> can include any of the conversion specifications from the C function strftime().

Example of an entmgr.ini file

```
[Comms]
server=yourcompany+389
Mgrlisten=709
ASHlisten=710
CMPlisten=829
serverType=External
DirectoryProtocol=LDAPV3

[login]
sepDisabled=false
sepDeactivated=false
cmpDisabled=false
ashDisabled=false
sepNumProcs=2
cmpNumProcs=2
```

```
ashNumProcs=4
watchdog=300
maxRestarts=10
autoRestartPeriod=1
autoRestartNotBefore=00:00
autoRestartNotAfter=23:59
entbackupPeriod=24:00
entbackupNotBefore=03:00
entbackupNotAfter=05:00
entincbackupPeriod=01:00
entincbackupNotBefore=00:00
entincbackupNotAfter=23:59
entintegrityPeriod=24:00
entintegrityNotBefore=02:00
entintegrityNotAfter=05:00
reloadCAKeyskPeriod=00:30
entCrlCheckPeriod=00:30
entPCertReissuePeriod=24:00
entCaKeyCheckPeriod=24:00
entUserStateHistTrimMax=186
entUserStateHistTrimPeriod=24:00
entUserStateHistTrimNotBefore=03:00
entUserStateHistTrimNotAfter=05:00
entbackupBackupsToKeepOnline=7
disableDbBackup=false
ManagerBackup=C:\entbackup
CADN=o=Your Company,c=US
entbackupLastUse=2

[policy]
FirstOfficerDN=cn=First Officer,o=Your Company,c=US

[Authorization]
SerialNumber=yrco
UserLimit=20
Key=aielclyn
WebSerialNumber=yrco
WebUserLimit=50
WebKey=aielclyn

[database]
dbname=CA1_odbc

[logging]
level=0
facility=0

[FIPS Mode]
FipsMode=1
```

```
entmasterAuth=DES-MAC,64,033226D803CC9DC5:CAST3-MAC,64,64,5FAEFF0F976FCE
_continue_=58
entauthsrvAuth=DES-MAC,64,090F24286E8846B3:CAST3-MAC,64,64,4EB7A43794776
_continue_=6A4
ENTAUT~1Auth=DES-MAC,64,090F24286E8846B3:CAST3-MAC,64,64,4EB7A437947766A
_continue_=4
entshAuth=DES-MAC,64,C17410BE491BABE7:CAST3-MAC,64,64,42EFE37D95CAC245
entsh-euroAuth=DES-MAC,64,83EB57BA5B8DC980:CAST3-MAC,64,64,4BDF904FCB405
_continue_=F3D
```

Appendix C

Navigating the log files

The following table lists Entrust log files, provides their default locations, and briefly explains what they mean.

Default location	Log	Explanation
C:\authdata\log	certspec.log	Generated when the certificate definition file is imported using the entsh command "fcs import". Also generated when changes are merged into the certificate definition file during upgrade of Entrust/Authority.
	initial.log	Generated during first-time initialization, when the certificate definition file (initial.certspec) is processed.
	manager.log	The main Entrust/Authority error log file, showing Entrust/Authority errors. To control the size and number of manager.log files, see the entries "Filesize" and "Numfiles" under "Customizing the entmgr.ini file" on page 527.
	mgraudit.log	A clear-text version of the audit logs. To control the size of the mgraudit.log file, see the entry "Filesize" under "Customizing the entmgr.ini file" on page 527.

Default location	Log	Explanation
C:\entbackup\mgrbk*yyyymmddhhmmss*\Manager Data Files	authbkup.log	Generated if an error occurs during backup of Entrust/Authority data files. If the entbackup directory does not exist, authbkup.log will be created in the root of the drive where Entrust/Authority is installed.
C:\entbackup\mgrbk*yyyymmddhhmmss*\Database	bkupfull.log bkupfullfile.log	Generated during full database backup. For Informix databases only.
	bkuplog.log bkuplogfile.log	Generated during incremental database backup. For Informix databases only.
C:\Program Files\Entrust\Entrust Authority\Tools\dvt	etdircheck.log	Generated by the Directory Verification Tool.
	etdircheckdetails.log	Generated by the Directory Verification Tool.
C:\Informix	install.log	Generated during Informix installation or upgrade.
C:\Program Files\Entrust\Entrust Authority	installmgr.log	Generated during Entrust/Authority installation or upgrade.

Appendix D

Disaster recovery and high availability

This appendix covers disaster recovery planning—how to minimize loss of Entrust/Authority data and services in the event of a disaster—including details on remote replication or disaster recovery sites as part of a disaster recovery plan.

This appendix also provides information to help you determine the availability requirements for your Entrust/PKI system, as well as guidelines on planning, designing, configuring and maintaining a highly available Entrust/PKI.

Since each organization's needs are different, you will need to plan and design solutions for disaster recovery and availability that work for your organization. While case studies and examples are provided here, this appendix does not attempt to be an exhaustive treatment of disaster recovery and availability. Rather, the focus is on providing guidelines and pointing out additional resources to help you determine your availability and disaster recovery needs and design a system to meet those needs. This appendix includes the following sections:

- "Overview" on page 554
- "Identifying and assessing risks" on page 557
- "Disaster recovery strategies" on page 561
- "Disaster-proofing your Entrust/PKI" on page 564
- "Ensuring service availability" on page 566
- "Ensuring data availability" on page 569
- "Determining Entrust/PKI availability requirements" on page 572
- "Configuring Entrust/Authority in a high availability system" on page 576
- "Ensuring a successful Entrust/PKI failover" on page 582
- "Case studies of highly available Entrust/PKI systems" on page 583

Overview

This section provides an overview of disaster recovery planning and high availability.

What is disaster recovery planning?

A Disaster Recovery Plan (DRP) identifies the critical applications, systems, data, and other resources of the organization. It also lists the priorities in which these systems will be restored in case of catastrophic failure, who is responsible for each activity, where it will be done, and how it will be done. A DRP should be prepared and tested in anticipation of an event, and not after the event has occurred.

There are two overall aspects of disaster recovery planning:

- a contingency management strategy to be used following a minor disruption situation
- a disaster recovery strategy to be used following a major disaster situation

Each strategy is composed of a number of phases:

- Response Phase

  This includes the initial activities after a disruption or disaster occurs, before a formal contingency or disaster declaration is made. If a declaration occurs, a decision is made whether to proceed into a contingency management phase (for minor disruptions) or into the resumption phase (for more serious disruptions or disasters).

- Contingency Management Phase

  This includes those activities, after a formal contingency declaration is made, to bring the primary system back to full production at the primary facility. If delays cause the system to fail being recovered within the contingency management window, then it may be necessary to move to the next phase—resumption.

- Resumption Phase

  This includes those activities, after a formal disaster declaration is made, to bring the initial, critically required services back into production, at the backup facility.

- Recovery Phase

  This includes those activities, after initial resumption is complete, to bring the additional, important services back into production, at the backup facility.

- Restoration Phase

  This includes those activities required to bring the primary system back to a state whereby repatriation can occur.

- Repatriation Phase

  This includes those activities required to move system operations from the backup facility back to the primary facility. This requires a great deal of precision to ensure that real-time transaction data is not lost during the transfer.

What is high availability?

High availability is one component of a contingency management strategy that can address minor disruptions in services through failover and redundancy. High availability is fundamentally a risk-management process aimed at ensuring specific levels of user service at a manageable cost.

It is critical to address both *data availability* and *service availability* in your contingency management strategy. Three factors affecting data and service availability are people, processes and product (technology). In other words, availability depends on trained people who follow specific processes using highly reliable products and technology.

Availability must be measured at the point of user interaction with any given business process. Fault-tolerant systems are generally defined as achieving 99.999% availability, translating to about five minutes of downtime per year. Such systems are very expensive. Highly available systems, by contrast, are generally ranked in the 99% to 99.9% availability range and are less expensive to implement than fault-tolerant systems. This range of availability translates into about nine hours to 3.65 days of downtime per year.

Resources

This section provides references to documents containing overviews of high availability, overviews of disaster recovery, and glossaries of critical terms. Note that high availability/disaster recovery software and hardware vendor Web sites can also be excellent resources.

- Dowd, Kevin, et al. *High Performance Computing: RISC Architectures, Optimizations and Benchmarks*. O'Reilly & Associates, 1998.
- Marcus, Evan, and Hal Stern. *Blueprints for High Availability: Designing Resilient Distributed Systems*. John Wiley & Sons, 2000.
- Missbach, Michael and Uwe M. Hoffmann. *SAP hardware solutions: servers, storage, and networks for mySAP.com.* Prentice Hall Professional Technical Reference, 2000.

  Includes information on zero-down-time backup, remote database mirroring, clustering, and other failover solutions.
- Stone, Brad, and Julie Symons. *UNIX Fault Management: A Guide for System Administrators.* Prentice Hall Professional Technical Reference, 2000.

- http://www.highavailabilitycenter.com

  The most comprehensive Web site for high availability, from IMEXResearch.com.
- http://www.imexresearch.com

  The parent site for highavailabilitycenter.com and clusteringcenter.com.
- http://www.dhbrown.com

  The Web site of the research and consulting firm D. H. Brown Associates, containing overviews and analyses of high availability and disaster recovery technologies.
- http://www.wiley.com/compbooks

  Wiley Computer Publishing's Web site.
- http://www.webopedia.com

  Short, clear definitions of terms related to computer technology.

Identifying and assessing risks

Planning a system that meets your organization's specific needs involves identifying risks, assessing impact and establishing preventive measures.

Identifying risks to your Entrust/PKI and assessing their impact is the essential first step in determining disaster recovery and availability requirements. Risk analysis involves identifying the most probable threats to your organization and analyzing the degree to which your organization may be vulnerable to these threats.

Establishing preventive measures may involve a disaster recovery plan, high availability architecture, or both. For example, you may find that from a disaster recovery perspective, your risks can be managed with a plan that involves remote replication, while your service and data availability needs are met with a standard architecture.

Once you've performed a risk/cost analysis, the information in the next two sections will help you determine your requirements with respect to availability and disaster recovery.

This sample worksheet identifies risk sources, risk events, and the probability and seriousness of those events.

Table 5: Sample worksheet for assessing the potential risk of outages[1]

Risk source	Event	Probability of event (1-10)	Seriousness (1-10)
End user	Absence through illness		
	Employee turnover, lack of training		
	Sabotage		
	Hacking		
	Incorrect data entry		
	Mistaken transaction		
	Fraudulent transaction		
	Desktop component failure		
Network	Hub/router/switch component failure		
	Power outage to communications rack		
	Telecommunication failure		
	Unauthorized access		
Server	Application or database failure		
	App/database interoperability problem		
	Component failure—disk, controller, processor, OS, and so on		
	Administrative—failed upgrade, reconfiguration, poor capacity planning		
	Environmental failure—power, air conditioning, and so on		
Building incidents	Fire, power loss, flooding, lightning strike		
	Security breach, theft, damage		
Locality incidents	Terrorist event leading to security shutdown		
	Flood, hurricane, power, earthquake		

1. IBM Corporation. "Arriving at the upside of uptime: How people, processes and technology work together to build high availability computing solutions for e-business". December 1999.

This sample worksheet helps you calculate downtime.

Table 6: Sample worksheet for calculating downtime[1]

Cause	Frequency per year	Incident duration in hours	Annual downtime in hours
Planned			
Backup (copying data to tape or archive)			
Maintenance (replacing hardware, installing software patches, bug fixes)			
Upgrades (adding/enhancing hardware/software)			
Unplanned			
Hardware (failures—CPU, memory, disk, NIC, and so on)			
Software (failures—BIOS, firmware, drivers, OS, and so on)			
Human error (process and procedural errors, mistakes)			
Environmental (fire, power outages, other events)			
Total			

1. From Hewlett-Packard's high availability Web site, http://www.softwareproducts.hp.com/highavailability.

This sample worksheet helps calculate the costs of downtime.

Table 7: Sample worksheet for calculating the costs of downtime[1]

	Number of users	User cost per hour	Total hours of downtime	Percent burdened	System cost per hour	Total cost per occurrence	Total
Lost revenue							
Idle labor							
Other							
	Average # of users						
				Total			

1. From Hewlett-Packard's high availability Web site, http://www.softwareproducts.hp.com/highavailability.

Resources

This section provides references to resources on risk analysis and contingency management planning.

- IBM Corporation. "Arriving at the upside of uptime: How people, processes and technology work together to build high availability computing solutions for e-business". December 1999.
- *Disaster Recovery Journal*. A quarterly magazine dedicated to business continuity, since 1987.
- http://www.drj.com

  The Web site for the *Disaster Recovery Journal*. This is by far the most informative disaster recovery site available. Features articles, listings of disaster recovery associations, technology, events, a glossary, and an extensive list of links.
- Wold, Geoffrey H. and Robert F. Shriver. "Risk analysis techniques: The risk analysis process provides the foundation for the entire recovery planning effort". On the Web site of the *Disaster Recovery Journal*, http://www.drj.com.

Disaster recovery strategies

Most businesses depend heavily on technology and automated systems, and their disruption for even a few days could cause severe financial loss. The continued operations of an organization depend on management's awareness of potential disasters, their ability to develop a plan to minimize disruptions of mission critical functions, and the capability to recover operations expediently and successfully. The risk analysis process provides the foundation for the entire recovery planning effort.

A primary objective of disaster recovery planning is to protect the organization in the event that all or part of its operations and/or computer services are rendered unusable.

This section provides an overview of the strategies and technologies for ensuring quick and reliable recovery from disaster, including

- a sound backup and restore strategy
- remote replication, or disaster recovery sites

Note: To ensure that you develop a sound disaster recovery strategy, you may want to consult a database expert.

Creating a backup and restore strategy

The backbone to any disaster recovery plan is a sound backup and restore strategy. In this area especially, process is at least as important as product—technology by itself cannot protect your organization from data loss in the event of a disaster; rather, the right technology must be coupled with a stringent procedure to ensure that critical data loss does not occur.

A critical part of any backup and restore strategy is regular testing.

For an overview of the technologies for protecting data, see "Ensuring data availability" on page 569.

Using remote replication

Remote replication involves maintaining a mirror image of data at the logical volume level, at one or more physically separate sites. These sites are also known as disaster recovery sites. If a disaster occurs at the primary site, "failover" (either automatic or manual) occurs to the disaster recovery site, allowing critical data to be available to the business operation.

Disk-level vs. database-level replication

There are two main methods for remote replication: *disk-level* and *database-level*. Database-level replication is the preferred method, because it guarantees database consistency. Disk-level replication can involve the risk of data corruption or inconsistency.

After resumption, the integrity of the database should be validated to ensure the success of the replication. If a primary site failure occurs during real-time disk-level replication to the secondary site, database integrity errors may result. In this case, performing an integrity validation is the only way to ensure the data's validity.

When choosing your replication method, consider the impact of a replication failure on your organization.

Real-time vs. delayed replication

Data replication to the secondary site (whether disk-level or database-level) can be either *real-time* or *delayed*.

Real-time data replication can be either synchronous or asynchronous, and the data must be secured during transfer (for example, by a VPN).

Delayed data replication can occur by physically transporting a disk or tape from the primary to the secondary site.

The frequency of replication determines how much data would be lost in the event of a failover to the secondary site.

Resources

This section provides references to documents on disaster recovery, backup and restore, and remote replication.

- http://www.isaca.org/jrnlhome.htm

  The *IS Audit and Control Journal* Web site contains some great articles on disaster recovery under the "Article Index" link.

- http://www.weyerhaeuser.com/disaster-recovery

  Weyerhaeuser Disaster Recovery Services' Web page.

- *Disaster Recovery Journal*. A quarterly magazine dedicated to business continuity, since 1987.

- http://www.drj.com

  The Web site for the *Disaster Recovery Journal*. This is by far the most informative disaster recovery site available. Features articles, listings of disaster recovery associations, technology, events, a glossary, and an extensive list of links. Click the "New to DR" link for useful information for people new to disaster recovery issues.

- http://www.markencom.com/backup.htm

  The Marken Communications Web site contains white papers, industry position papers, and articles.

- http://www.drie.org

  The Disaster Recovery Information Exchange Web site includes some good links, plus information on organized business continuity planning, information systems security, and physical security.

- http://www.ContingencyPlanning.com

  This is the Web site for *Contingency Planning and Management* magazine. Check out the "Knowledge Base" Link at http://www.ContingencyPlanning.com/webguide_search.cfm and enter "Disaster Recovery" for a wealth of information.

- http://www.continuityplanner.com

  This Web site's purpose is to help develop an online community of business continuity planners.

- http://www.disaster-resource.com

  Resources for disaster recovery and business continuity.

Groups dedicated to disaster recovery issues

Two of the major organizations are

- Association of Contingency Planners (ACP)

  http://www.acp-international.com

- Disaster Recovery Information Exchange (DRIE)

  http://www.drie.org

There are also many independent groups. The Web site of the *Disaster Recovery Journal* (http://www.drj.com) contains an extensive list of disaster recovery associations.

Disaster-proofing your Entrust/PKI

Contingency and disaster planning for the Entrust/PKI system should be completed and documented as part of your organization's overall Disaster Recovery Plan. This section includes

- information on issues involved in backup and restore for Entrust/PKI
- information on enhanced database protection and remote replication
- an example Entrust/PKI architecture incorporating a disaster recovery site

Backup and restore for Entrust/PKI

Once you understand the major issues involved in data backup and restore, you need to consider the issues related to backup and restore specific to Entrust/PKI, as well as other Entrust server components you are deploying.

If you are using an Informix database, a first step in considering these issues is to read "Managing Entrust/Authority data for Informix" on page 105. If you are using an Informix database, you may find that Entrust's automatic backup system, coupled with daily tape backups stored offsite, meets your needs. However, if you have determined that this backup system is insufficient, or if you are using a database other than the Informix database, planning and executing an alternate system is necessary.

Enhanced database protection and remote replication

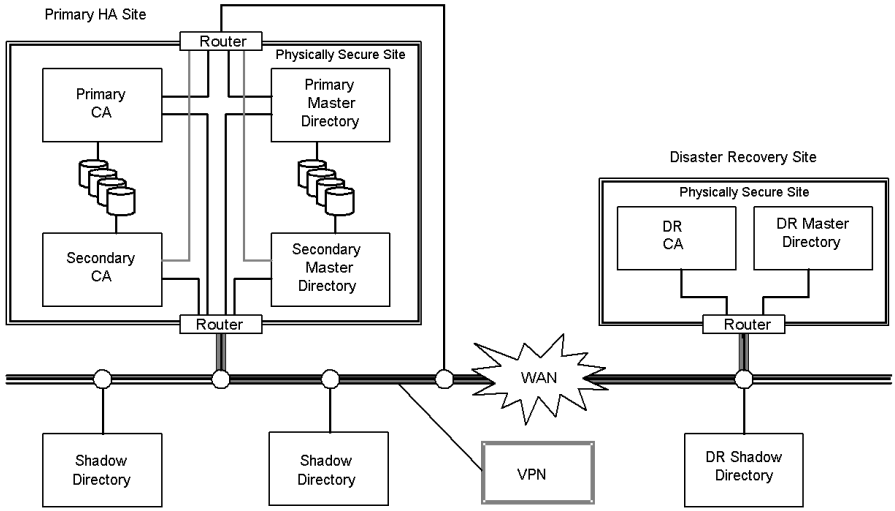
Entrust/PKI 6.0 provides enhanced database security with the addition of hardware-based database protection functionality. This functionality can be used to increase the security of real-time remote replication of the Entrust/Authority database. If hardware-based database protection is in place, Master User keying information is never exposed or transmitted remotely. If you do not implement hardware-based database protection, you must use a VPN to secure the connection.

For more information on enhanced database protection, see "Hardware-based database protection" on page 100.

Example architecture using a disaster recovery site

Figure 1 shows a sample Entrust/PKI architecture incorporating a disaster recovery site.

Figure 1: Sample Entrust/PKI architecture incorporating disaster recovery



Ensuring a successful transfer to the disaster recovery site

Once it has been determined that the HA site has experienced a catastrophic failure and a transfer of PKI services to the disaster recovery site is required, the following steps need to be performed:

1. Set the DNS server to point to the disaster recovery CA and the disaster recovery Master Directory.
2. Start the Directory.
3. Start the database services.
4. Start the Entrust/Authority services using Entrust/Authority Master Control or Entrust Master Control Command Shell.
5. If you are using one or more shadow Directories, ensure that they are properly shadowing the Master Directory.

Ensuring service availability

This section provides an overview of some of the technologies for ensuring service availability.

Clustering

Clusters are groups of two or more computers, or "nodes", designed to behave as if they were a single unit. A node is a single instance of a UNIX or Windows server—it can be a stand-alone server or a domain within a stand-alone server. Clusters are ideal for mission-critical systems on a network because one node can automatically take over for a failed node in the same cluster and prevent the application, system, or network from going down at crucial moments.

Third-party clustering software is available for both UNIX and Windows. Microsoft's Cluster Server (MSCS) is an example of Windows clustering software, while IBM's HA-CMP is available for AIX. Veritas produces clustering software for both UNIX and Windows. On UNIX, SUN Cluster, HP MC/ServiceGuard, and Legato Cluster are a few of the available cluster software packages.

Load balancing

Load balancing distributes processing and communications activity evenly across a computer network so that no single device is overwhelmed. Load balancing is especially important for networks where it's difficult to predict the number of requests that will be issued to a server. Load balancing can be used for two or more shadow Directories—if one Directory starts to become overloaded, requests are forwarded to another server with more capacity.

For Web serving, one approach is to use a rotating DNS (domain name system)—to route each request in turn to a different server host address in a DNS table, round-robin fashion.

Load balancing can be implemented with hardware, software, or a combination of both.

Hardware redundancy

Hardware redundancy ensures that additional hardware devices are available to replace failed devices. Redundant network interface cards, for example, can increase service availability; if one NIC fails, network activity can automatically be redirected through another NIC so that users do not experience any application downtime.

Power source protection

Protecting your power source is critical in ensuring service availability. Separate power sources can be valuable for long power outages, while an uninterruptible power source (UPS) can help protect against shorter power outages, sags, and surges.

A UPS sits between a power supply (for example, a wall outlet) and a device (for example, a computer) to prevent undesired features of the power source from adversely affecting the performance of the device. A UPS has internal batteries to guarantee that continuous power is provided to the equipment even if the power supply stops providing power for short periods.

In addition, some UPS or UPS–software combinations can

- automatically shut down equipment during long power outages
- restart equipment after a long power outage
- monitor and log the status of the power supply
- provide alarms on certain error conditions
- provide short circuit protection

Monitoring a highly available system

Monitoring—to prevent service failure and to recognize failure as quickly as possible—is an integral part of any high availability system. Most system management packages, and some system software included with the operating system, provide a means of monitoring the health of components in the network and alerting the administrator in the event of a problem—for example, if a disk drive is full, or a cooling system has failed. Good features to look for in a system management package include

- proactive alerting of a problem
- the ability to preset performance ranges for components and provide notification if the component is operating outside the levels set
- the ability to automatically correct repetitive problems without human intervention
- the ability to reroute traffic around a trouble spot

Cluster management software, for example, can be used to monitor the health of each node in a cluster and quickly respond to failures in a way that minimizes or eliminates application downtime. A process running on both high availability servers can monitor the servers, sending and receiving "heartbeats" between the servers. Lack of a heartbeat after a specified interval and number of retries may trigger a failover. The monitoring process also keeps track of which servers are active.

High availability software typically monitors for process existence, not functionality. A process may exist but may not be functioning properly, so that users are unable to complete tasks. It is possible to expand monitoring routines so that they test process functionality in addition to process existence.

In addition, regular testing of the system can help prevent failure.

Resources

This section provides references to information on technologies and strategies for clustering and uninterruptible power sources.

- http://www.clusteringcenter.com

  IMEX Research's Web site on clustering.
- http://www.uga.edu/~ucns/lans/docs/ups.html

  A very thorough FAQ page on uninterruptible power sources.

Ensuring data availability

This section provides an overview of some of the technologies for ensuring data availability.

Disk mirroring

Disk mirroring is a technique in which data is written to two duplicate disks simultaneously. This way if one of the disk drives fails, the system can instantly switch to the other disk without any loss of data or service. Disk mirroring is used commonly in on-line database systems where it's critical that the data be accessible at all times.

Disk mirroring is very effective if availability is a prime concern. When a disk is mirrored, it shares the controller with the server's primary disk, with each update being written to both disks. When the primary disk fails, users are transparently switched to the mirrored disk, with all files intact (up to the last backup).

If a file is corrupted or erased on a mirrored set of drives, both copies will be completely unusable.

RAID

RAID is short for Redundant Array of Independent (or Inexpensive) Disks, a category of disk drives that employ two or more drives in combination for availability and performance.

There are a number of different RAID levels. The three most common are 0, 3, and 5:

- Level 0: Provides data striping (spreading out blocks of each file across multiple disks), but no redundancy. This improves performance but does not deliver increased availability.
- Level 1: Provides disk mirroring.
- Level 3: Same as Level 0, but also reserves one dedicated disk for error correction data. It provides good performance and increased availability.
- Level 5: Provides data striping at the byte level and also stripe error correction information. With less storage overhead than RAID 1, this option results in excellent performance and high availability.

RAID levels 0 and 1 are often combined. RAID 0+1 provides mirrored striping: one side of the mirror is built, then mirrored to the other copy. Loss of any member of the stripe causes loss of the entire side of the mirror. RAID 1+0, by contrast, provides striped mirroring; this is a better option, since following loss of a single disk, further failures of any other disk except for its mirror can be tolerated.

RAID systems do not eliminate or reduce the need for backup. However, they maintain uptime of the network when performance is an important consideration, as RAID systems are able to survive single or multiple drive failure with no degradation to performance or data loss.

Intelligent disk arrays

An intelligent disk array is an array of disk units and controllers supporting connections from multiple "computers" or multiple redundant connections for each computer, or both. Intelligent disk arrays usually have large RAM caches. The management of the data (using RAID 1, for example) is transparent to the user. Intelligent disk arrays provide features such as high performance, availability, scalability, and connectivity. EMC's Symmetrix and Hitachi's Freedom Data Networks are two lines of intelligent disk arrays.

The Storage Area Network (SAN) is one form of intelligent disk array. A SAN is a data communications platform which combines LAN networking models with the core building blocks of server performance and mass storage capacity. SAN eliminates the bandwidth bottlenecks and scalability limitations imposed by previous SCSI bus-based architectures.

A SAN's architecture works in a way that makes all storage devices available to all servers on a LAN or WAN. As more storage devices are added to a SAN, they too will be accessible from any server in the larger network. In this case, the server merely acts as a pathway between the end user and the stored data. Because stored data does not reside directly on any of a network's servers, server power is utilized for business applications.

Resources

This section provides references to documents on technologies and strategies for ensuring data availability.

- http://www.acnc.com/raid.html

  Contains extensive information on RAID, with diagrams of each level.
- http://www.iftraid.com/support/fqna.html

  A FAQ Web page on RAID issues.
- http://www.raid-advisory.com/

  Home page of the RAID Advisory Board.
- http://www.storage-area-networks.com

  Includes a flash presentation on SANs, with a case study on using a SAN for a disaster recovery site.

- Thornburgh, Ralph H. and Barry Schoenborn. *Storage Area Networks: Designing and Implementing a Mass Storage System*. Prentice Hall Professional Technical Reference, 2000.

Determining Entrust/PKI availability requirements

Planners must determine which Entrust/PKI components need to be available, for which purposes and business processes, and during which time periods. The answers to these questions determine an organization's availability requirements.

Determining your Entrust/PKI availability requirements involves these steps:

1. Understand the functions of each of the core components in Entrust/PKI. See "About Entrust/PKI" in *Using Entrust/PKI 6.0 on Windows* to learn about the core PKI components.

   In addition, you will need to understand the functions of any other Entrust server components you will be deploying (such as Entrust/AutoRA, Entrust/ProfileServer, or Entrust/TruePass). Consult the product documentation for these products for a description of their functions.

2. Determine the availability requirements of each of the client applications that will access Entrust/PKI. See "Determining availability requirements for client applications" on page 572.

3. Understand what operations are available and unavailable if various Entrust/PKI core components fail. See "Understanding available operations in various scenarios" on page 573.

4. Determine your organization's availability requirements for each of the core Entrust/PKI components, as well as other Entrust core components your organization will be deploying. See "Determining availability requirements for components" on page 574.

5. If your analysis revealed a need for the Directory to be highly available, consider using one or more shadow Directories. See "Using shadow Directories with Entrust/PKI" on page 574.

Determining availability requirements for client applications

Once you understand the functions of the Entrust/PKI core components, you need to determine the availability requirements of each of the client applications that will access Entrust/PKI. Consider when users will be accessing each system, and whether your users inhabit multiple time zones.

Below is a sample worksheet for determining availability requirements over an eight-hour period.

Table 8: Sample worksheet for determining availability requirements for client applications

Monday	06:00	07:00	08:00	09:00	10:00	11:00	12:00	13:00	14:00
E-mail client									
Web client									
VPN client									
...									

Understanding available operations in various scenarios

Now that you have determined your availability requirements for Entrust/PKI operations, you need to understand how the availability of those operations is affected by various failure scenarios. Table 9 lists Entrust/PKI operations and whether they are available in various failure scenarios.

Note: If your organization is deploying other Entrust server components (such as Entrust/AutoRA, Entrust/ProfileServer, or Entrust/TruePass), these components will also affect the availability of Entrust/PKI operations.

Table 9: Availability of Entrust/PKI operations in various failure scenarios

Operation	Entrust/Authority failure	Directory failure
user login, logout	only until cached CRLs/ARLs expire	in offline mode only
file encryption	only until cached CRLs/ARLs expire	only if other users' certificates are in the certificate cache
file decryption	only until cached CRLs/ARLs expire	in offline mode only
change profile password	only until cached CRLs/ARLs expire	in offline mode only
user creation/ initialization	no	no
user updates (certificate management)	no	no
certificate revocation	no	no

Table 9: Availability of Entrust/PKI operations in various failure scenarios (continued)

Operation	Entrust/Authority failure	Directory failure
CRL and ARL creation	no	no
security policy modification	no	no

Determining availability requirements for components

Once you have determined when Entrust/PKI operations must be available to users and administrative users and how those operations are affected by various failure scenarios, the final step in creating your availability profile is to determine availability requirements for Entrust/PKI components.

Note: If your organization is deploying other Entrust server components (such as Entrust/AutoRA, Entrust/ProfileServer, or Entrust/TruePass), your availability profile must also include availability requirements for these components.

Below is a sample worksheet for determining availability requirements over an eight-hour period. See "Using shadow Directories with Entrust/PKI" on page 574 for a description of shadow Directories.

Table 10: Sample worksheet for determining availability requirements for Entrust/PKI core components

Monday	06:00	07:00	08:00	09:00	10:00	11:00	12:00	13:00	14:00
Entrust/Authority									
Master Directory									
Shadow Directory 1									
Shadow Directory 2									
Entrust/AutoRA									
Entrust/ProfileServer									
Entrust/TruePass									

Using shadow Directories with Entrust/PKI

For security and availability purposes, the functions of the Directory can be split into a Master Directory and one or more shadow Directories.

Note: While it is not strictly necessary to split the functions of the Directory into Master and shadow functions, since Entrust users regularly access the Directory, Entrust Technologies recommends the use of shadow Directories.

In an architecture that includes shadow Directories, the Master Directory is accessed primarily by Entrust/Authority. Entrust/Authority writes end user public encryption certificates, CRLs and ARLs to the Master Directory; the shadow Directories receive this information. Entrust users have no access to the Master Directory—they retrieve all information from the shadow Directories.

Entrust users access a shadow Directory to retrieve their certificates and associated CRLs at login. After login, users access a shadow Directory to retrieve public encryption certificates and CRLs for other user entries, as well as retrieve cross-certificates and ARLs for other CAs in certificate chains.

In this scenario, Entrust users access a shadow Directory regularly to use Entrust-Ready applications such as Entrust/Direct. Several shadow Directories may be listed in each user's entrust.ini file (to learn how to customize users' entrust.ini files, see "Customizing the entrust.ini file" on page 485). The user's Entrust software will attempt to use the first shadow Directory on the line, but if that Directory fails, the Entrust software will attempt to use the next one.

Configuring Entrust/Authority in a high availability system

This section provides information on configuring Entrust/Authority in a high availability system.

Note: Entrust's Professional Services group can help your organization plan and deploy your Entrust architecture in a high availability configuration. For more information on Entrust Professional Services, go to http://www.entrust.com/services.

Installing on a shared drive versus local drives

There are two options when installing Entrust/Authority in a high availability system: installing Entrust/Authority on a shared disk, or installing it locally on each machine.

Installing Entrust/Authority on a shared disk means that you have to perform only one installation, and if you decide to install a patch release or an upgrade, only one upgrade must be performed.

Installing Entrust/Authority locally on each machine means that if you want to install a patch release or upgrade the system in future, you can install the patch or perform the upgrade on the secondary machine and then manually fail over to the secondary machine while upgrading the primary machine.

Using multiple shadow Directories

If you use multiple shadow Directories, users' entrust.ini files must contain the host names of all shadow Directories. The Directory host name is specified in the "Server=" line of the [Entrust Settings] section of the entrust.ini file.

If more than one host name is listed on the Server= line, the user's Entrust software will attempt to use the first shadow Directory on the line. If that Directory fails, the Entrust software will attempt to use the next one on the Server= line, and so on.

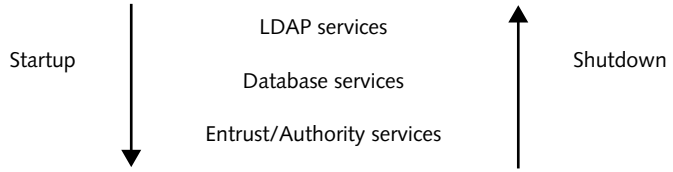
To learn how to customize the entrust.ini file, see "Customizing the entrust.ini file" on page 485.

Configuring high availability software for use with Entrust/PKI

If you are using Entrust/PKI in a clustered environment, you must ensure that the high availability software package is configured to start and stop Entrust/PKI components in the proper order.

Figure 2 shows the proper start and stop sequence for Entrust/PKI services.

Figure 2: Start and stop sequence for Entrust/PKI services



In addition, all Entrust/PKI components must be started only after all other system and network-level components have been started. Likewise, all Entrust/PKI components should be stopped before any system and network-level components are stopped.

For instructions on how to start and stop Entrust/Authority services, see "Starting and stopping Entrust/Authority" in *Using Entrust/PKI 6.0 on Windows*.

Note: If you are using CA hardware, contact Entrust Professional Services for help on configuration issues.

Monitoring a highly available Entrust/PKI

Monitoring the highly available system is critical to preventing service failure and to recognizing service failure as quickly as possible.

There are two levels of service monitoring:

- Test routines that periodically test for the presence of certain processes.
- Test routines that monitor whether users are actually able to access the system to perform tasks. Such a "health check" can provide an added check to determine whether a given service is functioning as required.

Such test routines are paired with restart routines that will attempt to restart services in the event of service failure.

Entrust/PKI processes to monitor

The following are processes that should be monitored in an Entrust/PKI installation:

- Entrust/Authority services. The Entrust test routine for UNIX, for example, tests for the presence of the entmon process; if this process is not found, the Entrust/Authority services are offline.
- The Directory. Consult your Directory documentation for the proper process to monitor.
- The database. Consult your database documentation for the proper process to monitor.

Other processes to monitor

It is also important to monitor processes related to the network interface, disk space, and the health of the file system.

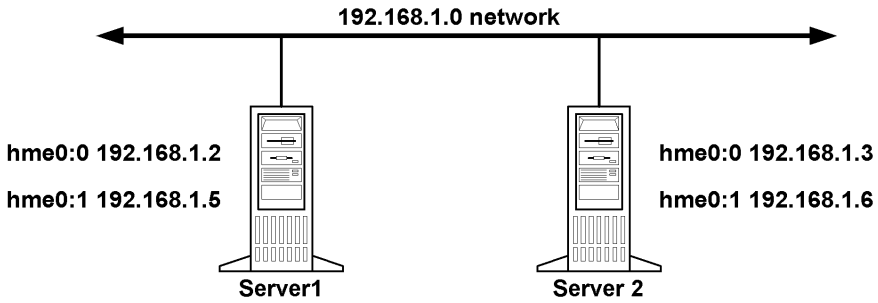
Using multiple network interface cards and virtual interfaces

Every host must have at least one IP address. A client uses a host name or IP address to access a server. This server may be a cluster of machines. Services that move between hosts in a high availability cluster need a virtual address. In the case of Entrust/PKI, if you are using a high availability cluster to host the Entrust/Authority services, each node in the cluster will have its own IP address, but the cluster as a whole needs a virtual IP address so that Entrust-Ready clients will continue to be able to access Entrust/Authority in the event of a failover to a secondary node in the cluster.

When using a virtual IP address for Entrust/Authority, every Entrust user's entrust.ini file must contain this virtual IP address. While Entrust/Authority does in fact have a physical IP address, the outside world sees only the virtual address.

Figure 3 shows how a virtual IP address works.

Figure 3: Using virtual IP addresses

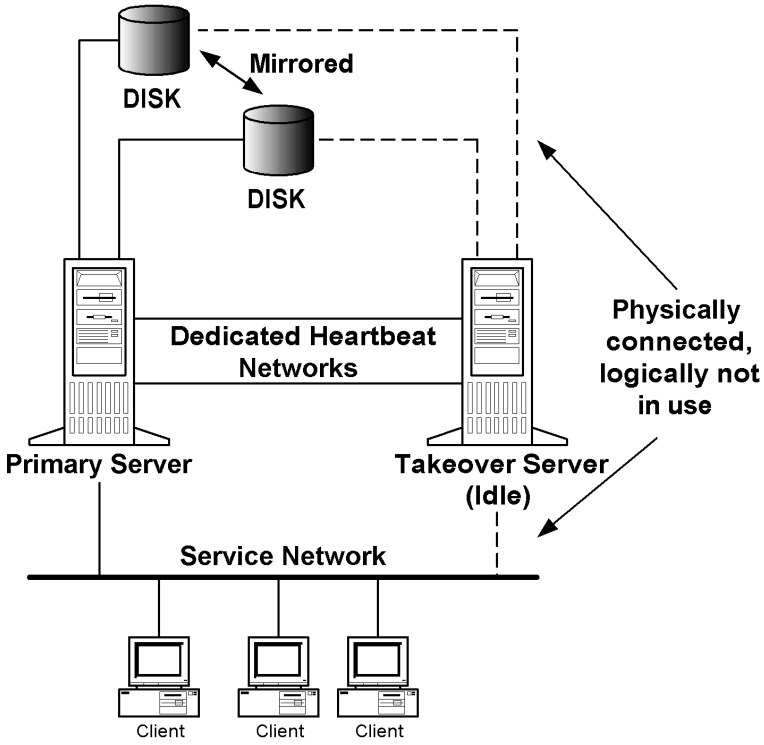


The server hosting Entrust/Authority can have multiple NICs, ensuring that if one network path fails, the other can take over with no failure of services.

If you use multiple NICs for Entrust/Authority, you must identify the production NIC and assure that the system is using the production NIC to access the production LAN (as opposed to the heartbeat, for example).

It is also critical to ensure that Entrust/Authority is communicating with shared disk resources, whether over SCSI or a SAN (Storage Area Network). Figure 4 shows a high availability configuration for networks.

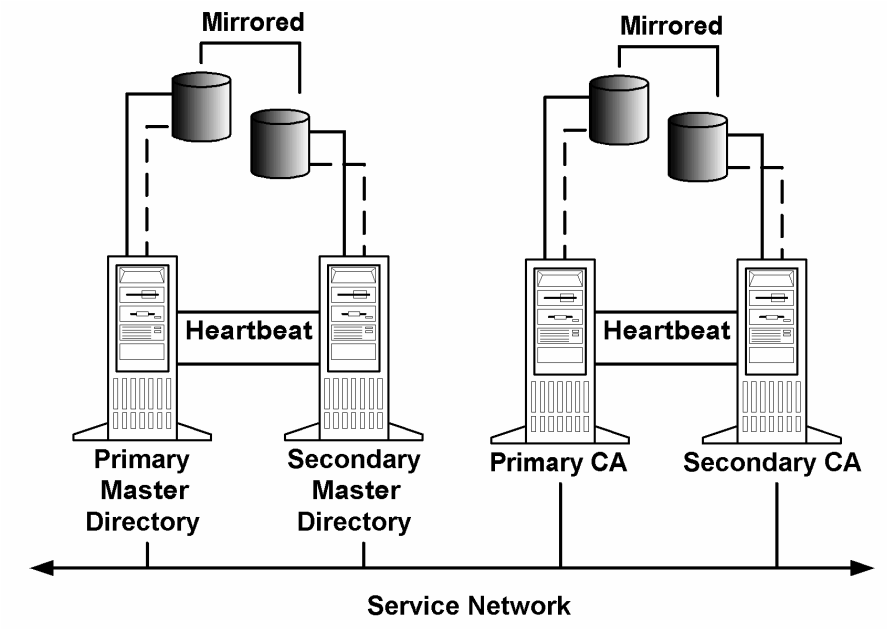
Figure 4: Networks configured for high availability



Sample Entrust/PKI architecture incorporating high availability

Figure 5 shows a sample Entrust/PKI architecture incorporating high availability clusters and RAID.

Figure 5: Sample Entrust/PKI architecture incorporating high availability



Ensuring a successful Entrust/PKI failover

In a high availability system in which additional installations of Entrust/PKI core components are available to take over from the primary installations, a switch from the primary to a secondary component is called a failover (or takeover). This section includes information on the circumstances in which a failover would be necessary, the impact of such a failover on Entrust users, and how to ensure a successful failover of Entrust/PKI.

Why a failover may be necessary

There are three events that can cause a failover:

- The primary node in the cluster stops sending out its heartbeat.
- A monitoring agent detects an application failure that requires failover.
- An operator manually requests a failover for maintenance reasons, such as an upgrade.

What happens during failover?

The following steps occur in the failover process:

1. The high availability services are stopped on the failed node (if possible).
2. Ownership of the shared disks, which were previously defined as being logically owned by the failed node, is migrated to the failover node. This effectively ensures that the failover node has full access to all the committed data that was previously owned by the failed node. These devices are then exported to the public network.
3. The network addresses of the failed node is migrated to the failover node.
4. The high availability services start up on the failover node.
5. Monitoring of the high availability services begins on the failover node.

   Maintenance should begin on the failed node to determine the cause of the failure. It should be brought up as soon as possible to make the system highly available again.

Attention: After failover to the secondary node, the high availability software should not fail back to the primary node. This ensures that services don't "ping-pong" between nodes.

Case studies of highly available Entrust/PKI systems

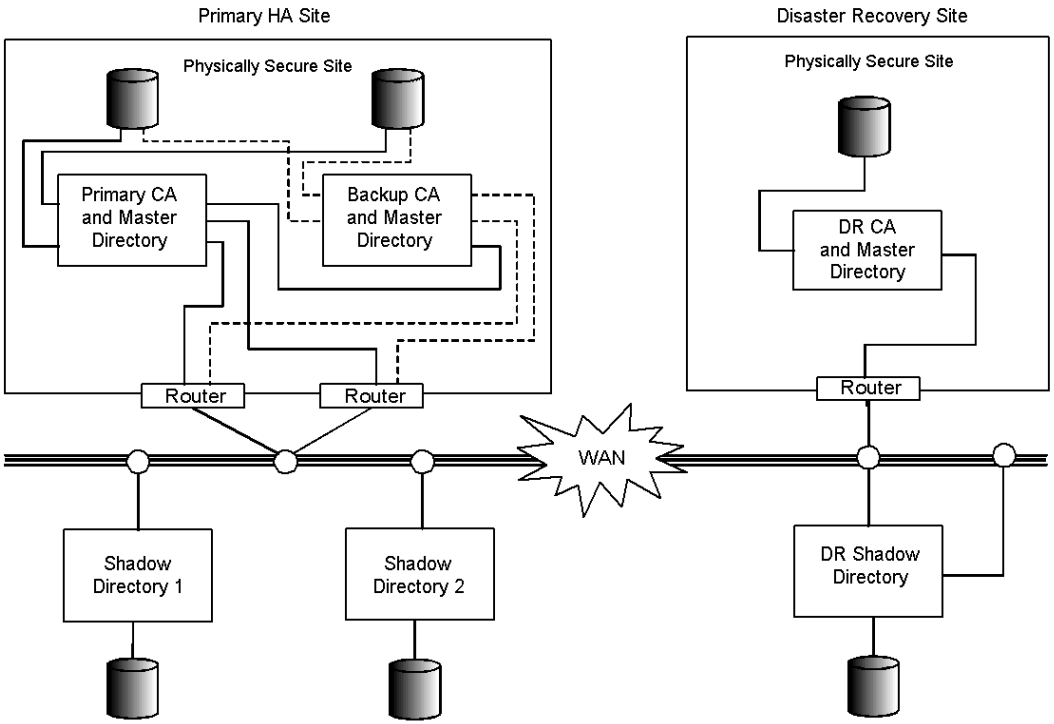
This section provides case studies from Entrust/PKI customers who have implemented high availability and/or disaster recovery sites.

Case Study 1

This customer determined that their highest priority was users' access to the Directory. However, the customer also required the ability to create and recover users at any time. This meant that both the CA and the Directory had to be highly available.

Figure 6 illustrates this customer's PKI architecture.

Figure 6: Customer's PKI architecture for Case Study 1



Master and shadow Directories

For security purposes, the customer decided to implement Master and shadow Directories. Only the CA accesses the Master Directory. Two shadow Directories provide high availability access to users. The shadow Directories are kept synchronized with the Master Directory using shadowing. Users' entrust.ini files contain the hostnames of both shadow Directories, allowing Entrust clients to automatically fail over from one Directory to another in the event of a shadow Directory failure.

High availability cluster

To ensure availability of the CA and the Master Directory, two servers were configured to work in a cluster using high availability software from Veritas. The two servers were placed in an asymmetric configuration. The primary server is used to provide CA and Master Directory services; the secondary server is continuously running, but exists solely to take over the CA and Master Directory services from the primary server in case of primary server failure. The high availability software monitors the processes on the primary server.

Note: Entrust Technologies recommends that you install your Directory on a server separate from Entrust/Authority, the Entrust/Authority database, and the backup files for both Entrust/Authority and the database.

The high availability cluster resides in a physically secure site.

Entrust/Authority program files will be installed on a shared disk array.

High availability software

The customer chose to use Veritas products to ensure file system integrity and recovery as well as fully automatic failover management, including fault detection, isolation, and recovery.

Because Entrust/Authority must provide services from a single network address or host name, if a failure occurs on the primary server hosting Entrust/Authority and the secondary server takes over, the customer's high availability software was configured to perform a logical network address failover using TCP/IP.

The customer chose to

- execute the test routines which monitor the Entrust, Informix, and Directory processes every 15 seconds
- attempt to restart the service on the primary server if the service was found to be not running for two consecutive tests
- halt the primary server and restart all services on the secondary server if the service failed after two restart attempts

Disaster recovery site

The customer's disaster recovery site includes the CA and Master Directory installed but inactive, residing in a physically secure site, as well as a shadow Directory which is installed but inactive.

This customer's disaster recovery site is a cold site, to be started manually once a disaster had been declared in accordance with their disaster recovery policies and procedures.

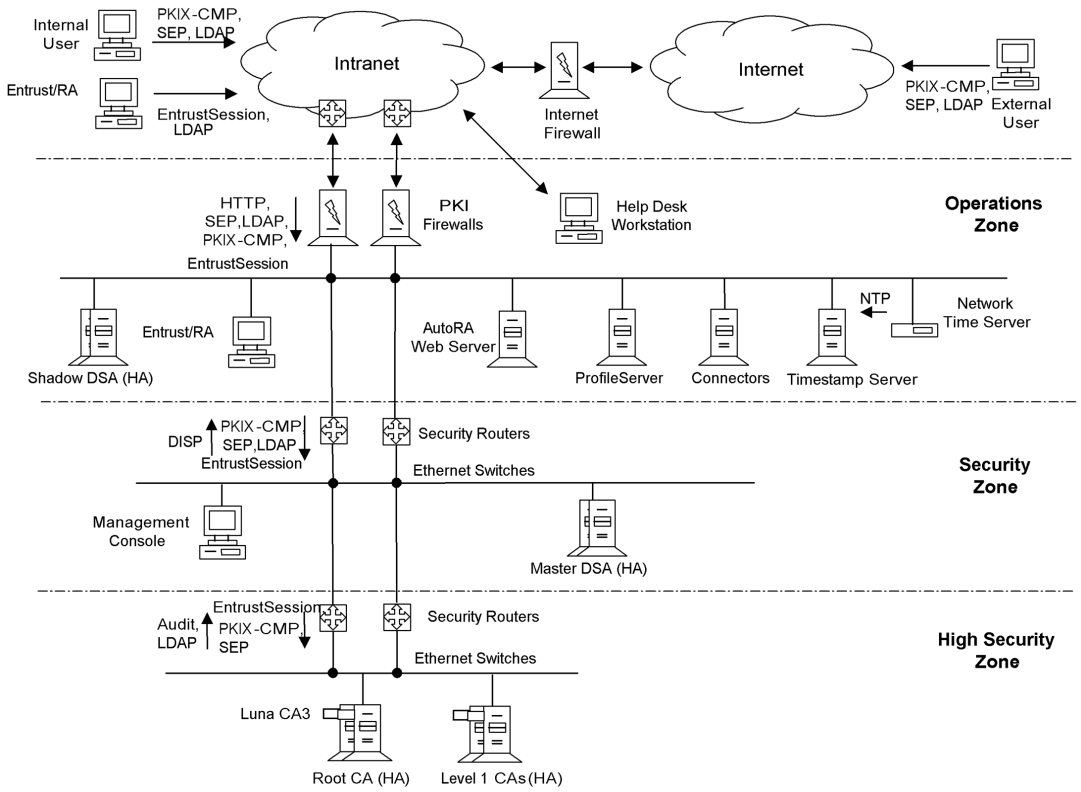
The customer chose to use rdist, a UNIX utility for distributing files, to duplicate the CA files to the disaster recovery site. Duplication of files occurs nightly. This remote backup strategy required a secure connection between the two sites, such as a virtual private network (VPN) relying on Entrust certificates for authentication.

Case Study 2

This customer's PKI will provide secure e-mail, automated registration, time stamping and roaming login capabilities. In addition to the core Entrust/PKI components, this customer chose to deploy Entrust/ProfileServer, Entrust/Timestamp and Entrust/AutoRA.

Figure 7 illustrates this customer's PKI architecture.

Figure 7: Customer's PKI architecture for Case Study 2



This customer requires a root CA as well as one subordinate CA, on separate platforms. The root CA provides public key generation and management services to the subordinate CA. The subordinate CA and the Master Directory must support at least one million users, and potentially up to five million.

All mission-critical components are duplicated for high availability; no single points of failure exist except for the firewalls, and the customer planned to investigate a corporate redundant firewall strategy.

This customer's PKI architecture is separated into three zones: High Security, Security, and Operations.

High Security zone

The root and subordinate CAs, cluster administration workstations, a firewall and an Ethernet switch are installed in the High Security zone.

The primary and secondary servers hosting the root CA are configured in a high availability cluster, as are the primary and secondary servers hosting the

subordinate CA. The customer chose to use HP's MC/ServiceGuard high availability software to manage the clusters.

The customer will use RAID 1+0 for both the root and subordinate CAs' data storage requirements.

A firewall is used to provide a second layer of access control to the components in the High Security zone (the root and subordinate CAs). The firewall is configured to allow only those protocols required for PKI management services.

A fast Ethernet switch is used at the data link layer to switch Ethernet packets between the various platforms in the High Security zone. The switch supports high availability, non-blocking configurations with redundant inter-switch links.

Security zone

The Master Directory, a firewall, and the network management console are located in the Security zone.

The Master Directory holds the master copy of the PKI directory and will periodically shadow critical portions to the shadow Directory (using DISP). The Master Directory is used as the trusted repository for all PKI certificates and is only accessed by the root and subordinate CAs, as well as Entrust administrative users using Entrust/RA. All PKI user access for certificates, CRLs, and ARLs is directed to the shadow Directory.

The network management console is used as a central repository for audit logs from the CAs and as a monitoring station for security alarms from the firewalls. It is not used for remote management of the components in the High Security zone.

A firewall is used to provide a layer of access control to the components in the High Security zone (the root and subordinate CAs). The firewall is configured to allow only those protocols required for PKI management services.

Operations zone

The Operations zone contains an Entrust/RA workstation, the Entrust/AutoRA Web server, and a help desk workstation. Entrust/ProfileServer and Entrust/Timestamp are also located in the Operations zone.

Backup strategy

This customer determined that the automatic backup schedule provided by Entrust/Authority for the Entrust/Authority database and data files was sufficient to meet their backup needs. All backups are stored in a separate backup partition on the primary server hosting the root CA. The customer's network is used to provide a daily secure backup to a remote location. The Master Directory is backed up using the third-party Directory backup tool, and this backup is also copied daily using the customer's network.

The remote backup copies of the Entrust/Authority database and data files can be used in case of a local PKI failure. The remote backup can also be used to recreate the PKI in a backup location in case of a catastrophic failure.

Each CA stores its private keys on a Luna CA3 cryptographic token. Clones of each token are made, and one copy is stored offsite.

Disaster recovery site

A remote facility acts as the disaster recovery site and backup site for the customer's PKI. This site contains sufficient equipment to duplicate the production environment for the core PKI. The root and subordinate CAs are hosted on one machine at this facility. The disaster recovery site also contains a fast Ethernet switch, a Master Directory containing the master copy of the PKI, and a firewall to provide a layer of access control to the components in the disaster recovery PKI.

Appendix E

Running Entrust/PKI in an Identrus environment

This chapter includes the following sections:

- interoperability of Entrust/PKI and Identrus components (see "Entrust/PKI in the Identrus environment" on page 590)
- installation of Entrust/PKI in an Identrus environment (see "Installing Entrust/PKI in the Identrus environment" on page 592)
- method of changing the value that represents the URL of an OCSP (Online Certificate Status Protocol) responder in certificates (see "Changing the URL of an OCSP responder" on page 602)

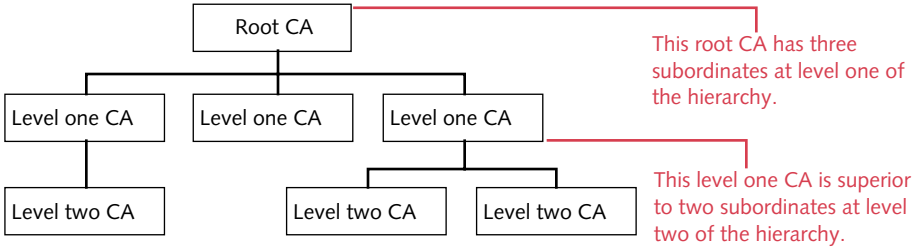
Entrust/PKI in the Identrus environment

The Identrus environment is comprised of components including CAs, OCSP responders, and smartcards. The following sections discuss the interoperability of Entrust/PKI with other components in the Identrus environment.

Interoperability with the root CA

CAs are organized hierarchically in the Identrus environment. A hierarchy is an inverted tree structure that contains a single "root" CA and one or more levels of subordinate CAs below the root CA.

The following diagram shows a hierarchy of CAs:



Entrust/PKI supports CA hierarchies. The Entrust CA (named "Entrust/Authority") can be installed as subordinate CA at either level-one or level-two of the Identrus hierarchy.

Note: This chapter describes how to install Entrust/PKI when Entrust/Authority will operate as a level one CA. If you are running a level one CA and you want to add a level two CA below it, contact Entrust Support.

Interoperability with OCSP responders

In the Identrus environment, information about certificate revocations is available from OCSP responders. By default, Entrust/Authority looks for revocation information in the Directory, and the Directory of its superior CA.

To support interoperability within the Identrus environment, Entrust/Authority inserts the URL of the OCSP responder in certificates. This URL is included as a DER-encoded value. For information about providing the URL as a DER-encoded value to Entrust/Authority, see "Specifying the value that represents the OCSP responder URL" on page 594.

Note: The procedures that follow describe how to set up Entrust/PKI to work with Valicert's OCSP responders. If you are using an OCSP responder from another vendor, check with that vendor to see how it interoperates with Entrust/PKI.

Interoperability with smartcards

Identrus requires that the certificates generated by a CA can be stored on smartcards. To support interoperability within the Identrus environment, this release of Entrust/PKI supports smartcards through the use of Entrust/Web Connector.

Note: The process of loading certificates on smartcards is specific to each smartcard vendor. Consult with the vendor of the smartcards you plan to use to determine how this process will work.

Installing Entrust/PKI in the Identrus environment

Follow the procedures listed in *Installing Entrust/PKI 5.1 on Windows NT* to install Entrust/PKI within the Identrus environment. As well, follow the information in the next two sections, which describe additional information for installations in the Identrus environment.

Note: Before you install Entrust/PKI in the Identrus environment, see "Configuring the entmgr.ini file" on page 592 and "Changes to the certificate definition file" on page 597.

Providing Identrus-specific input for installation prompts

During the Entrust/PKI installation, you are prompted for information. Three of these prompts require specific input for Identrus installations:

When you are prompted to...	Do this...
choose whether this is a "Root Certificate Authority" or a "Subordinate Certificate Authority"	Choose "Subordinate Certificate Authority".
provide the DN of the superior CA	Type the DN of the Identrus root CA.
choose whether the subordinate CA will communicate with the superior CA "online" or "offline"	Choose "offline".

Configuring the entmgr.ini file

This section describes the changes you need to make to the entmgr.ini file.

Note: You must make the following changes to the entmgr.ini file after you have finished configuring Entrust/PKI, but before performing first-time initialization (that is, before running Entrust/Authority Master Control for the first time).

Identifying the certificate definition file

By default, Entrust/PKI reads information from a certificate definition file named "initial.certspec" during first-time initialization. For this release, Entrust/PKI must read information from a file named "identrus.certspec".

There is a line in the entmgr.ini file that specifies which certspec file Entrust/PKI will use during first-time initialization. Replace "initial.certspec" with "identrus.certspec", so that the line appears as follows:

```
InitialCertDefnFile=c:\Program Files\Entrust\Entrust
Authority\etc\identrus.certspec
```

For information about initial.certspec and identrus.certspec, see "Creating Identrus certificate types" on page 598.

Disabling audit warnings

By default, Entrust/PKI generates an audit warning when it cannot access the root CA's Directory. Because the Identrus root CA's Directory is not accessible, these audit warnings would be generated frequently. Although the warnings don't cause a problem, there is no reason to clutter the audit logs with them.

You can add the following entry in the [Login] section of the entmgr.ini so that Entrust/PKI will not issue these audit warnings:

```
offlineSuperior=1
```

Note: Configuring this setting is optional. Entrust/PKI will work whether this setting is configured or not.

Setting the key identifier algorithm

By default, Entrust/PKI generates key identifiers in a manner that's different from that specified by Identrus.

You must modify the KeyIdMode entry in the [Advanced Settings] section of the entmgr.ini file so that Entrust/PKI will generate key identifiers using the Identrus method, as follows:

```
KeyIdMode=1
```

Note: This is a required setting.

Using a Combined CRL

By default, Entrust/PKI uses only partitioned CRLs. To support interoperability with Valicert's OCSP responder, Entrust/Authority can be configured to also provide a Combined CRL.

You can modify the UseCombinedCRL entry in the [Advanced Settings] section of the entmgr.ini file so that Entrust/PKI will generate a Combined CRL, as follows:

```
UseCombinedCRL=1
```

For more information about Combined CRLs, see "Using a Combined CRL" in *Administering Entrust/PKI 6.0 on Windows*.

Note: If you are using Valicert's OCSP responders, this setting is required, in order to issue and maintain a Combined CRL for interoperability with Valicert's OCSP responders. If you are using an OCSP responder from another vendor, check with that vendor to see how it interoperates with Entrust/PKI.

Specifying the value that represents the OCSP responder URL

Within the Identrus environment, revocation information is provided by OCSP responders. Before you perform first-time initialization, you must specify the URLs of the OCSP responders that will provide revocation information.

You specify the URLs in the identrus.certspec file. Their value is represented in the identrus.certspec file as a DER-encoded value.

To generate the DER-encoded value

1. Open your web browser and access the following URL:

   `http://www.entrust.com/pkitools/aiaencoder.html`

2. As prompted, enter the URLs of the OCSP responders that will provide revocation information.

   The utility places the URLs in an extension (the AIA) and then DER-encodes the extension.

   Once you have received the DER-encoded value, you can specify the value in the identrus.certspec file.

To insert the DER-encoded value in the identrus.certspec file

1. When you have finished configuring Entrust/PKI, and before you perform first-time initialization (that is, before running Entrust/Master Control for the first time), open the identrus.certspec file in a text editor.

   By default, this file is stored in the folder named "c:\Program Files\Entrust\Entrust Authority\entmgrdata\manager".

2. Search for this line:

   ```
   aia=<enter der value here>
   ```

3. Replace the placeholder with the DER-encoded value.

   The placeholder appears in the identrus.certspec file twice. Ensure you replace both instances of the placeholder with the DER-encoded value.

4. Proceed with the first-time initialization of Entrust/PKI.

Note: If you don't provide the DER-encoded value, or if the value you provide is wrong, an error occurs during first-time initialization. If you get this error, edit the identrus.certspec file and then restart first-time initialization as described in *Installing Entrust/PKI 5.1 on Windows NT*.

Providing timely revocation information to an OCSP responder

You should ensure that revocation information is available to an OCSP responder as soon as possible. Entrust/PKI provides two ways of doing this:

- by configuring a setting in the central security policy. This setting causes Entrust/Authority to publish a new revocation list every time a certificate is revoked. For information about configuring this setting in the central security policy, see "Determining Security Policy Settings" in *Using Entrust/PKI 6.0 on Windows*.
- by explicitly choosing to issue a new revocation list each time a certificate is revoked. For information about explicitly choosing to issue a new revocation list, see the "Suspending a certificate" chapter in *Using Entrust/PKI 6.0 on Windows*.

Providing information to your smartcard vendor

In the Identrus environment, certificates are stored on smartcards. While the process of providing certificate information to your smartcard vendor (or to the

Smartcard Management System if you are using one) varies among vendors, the information required by the vendor is the same. You obtain this information using Entrust/Web Connector and Entrust/RA.

The procedures in this section use both Entrust/Web Connector and Entrust/RA. The information in this section is not intended to describe how to use these products. This information is only intended to describe their role in generating information for your smartcard vendor.

To prepare Entrust/Web Connector

1. Copy the DER-encoded PKCS#7 file returned to you from the root CA, and rename the copy "xcert.der".
2. Put the xcert.der file in the following location on the computer that is running Entrust/Web Connector:

   `Program Files\Entrust\Web-Connector\certs\`

Follow the next procedures to obtain the information required by your smartcard vendor (or the Smartcard Management System if you are using one) for each Identrus certificate that will be stored on a smartcard.

Note: Because all Identrus users require two Identrus certificates (that is, a utility certificate and an identity certificate), you must perform the following procedures for each certificate type.

To generate Identrus user certificates

1. Start Entrust/RA.
2. Create a new user, and choose the Web category and an Identrus certificate type within that category.
3. Record the reference number and authorization code generated by Entrust/RA for the new user.

To obtain the information for the smartcard vendor

1. Obtain a PKCS#10 Certificate Request in Base 64 (PEM) format from the smartcard vendor.

Note: This request must use the DN assigned to the user when generating Identrus user certificates (see "To generate Identrus user certificates" on page 596).

---

2. Run the Entrust/Web Connector application by opening your web browser and accessing the following URL:

   ```
   http://web_server_IP_address_or_
   name/cda-cgi/clientcgi.exe?action=start
   ```

3. Click Web Site Administrators.
4. Click Retrieve a site certificate for your Web server.
5. Type the reference number generated when you created the user (see "To generate Identrus user certificates" on page 596).
6. Type the authorization code generated when you created the user (see "To generate Identrus user certificates" on page 596).
7. In the Options field, select Displayed as PEM encoding of certificate in raw DER.
8. Paste the content of the Base 64 (PEM) encoded PKCS#10 Certificate Request in the box at the bottom of the page.
9. Click Submit Request.

   Entrust/Web Connector displays the certificate information for the requested certificate.
10. Copy the certificate information, and paste it into a document that you can send to the smartcard vendor.
11. Click Web Site Administrators.
12. Click Retrieve cross-certificate(s) for your Web server.

    Entrust/Web Connector displays the certificate information for the Level One CA.
13. Copy the certificate information, and paste it into a document that you can send to the smartcard vendor.
14. Click Web Site Administrators.
15. Click Retrieve the CA certificate for your Web server.

    Entrust/Web Connector displays the certificate information for the Identrus Root CA.
16. Copy the certificate information, and paste it into a document that you can send to the smartcard vendor.
17. Send the three documents that contain the information generated by Entrust/Web Connector to the smartcard vendor.

Changes to the certificate definition file

A certificate definition file is read by Entrust/PKI during first-time initialization, and provides information about certificates generated by Entrust/Authority. For

information about certificate definition files, see *Administering Entrust/PKI 6.0 on Windows.*

Because you changed the entmgr.ini file to point to identrus.certspec (see "Configuring the entmgr.ini file" on page 592), the certificate definition file read during first-time initialization is the identrus.certspec file. This section describes the changes to this file for this release.

Note: You are not required to edit the file to make these changes. The following sections are provided only for your information.

Creating Identrus certificate types

The Identrus certificate profile differs from the default Entrust/Authority certificate profile. Identrus users have two types of certificates:

- utility certificates used primarily for secure communications
- identity certificates used to provide digital signatures that are required as evidence of contractual commitment

In the identrus.certspec file, you will find the information necessary create the Identrus certificate types during first-time initialization. The following example shows the entries that create the Identrus certificate types:

```
.
.
.
[Certificate Types]
.
.
.
; ---------------------------------------------------------------
; Identrus end-entity certificates
; ---------------------------------------------------------------
identrus_identity=web,Identity cert,Identrus Identity Certificate
identrus_utility=web,Utility cert,Identrus Utility Certificate

[Extension Definitions]
.
.
.
[identrus_identity Verification Extensions]
; ---------------------------------------------------------------
; Identrus Identity Certificate Type
; ---------------------------------------------------------------
keyusage=2.5.29.15,n,m,BitString,11
certificatepolicies=2.5.29.32,n,m,DER,3082018d30......76e65722e
.
```

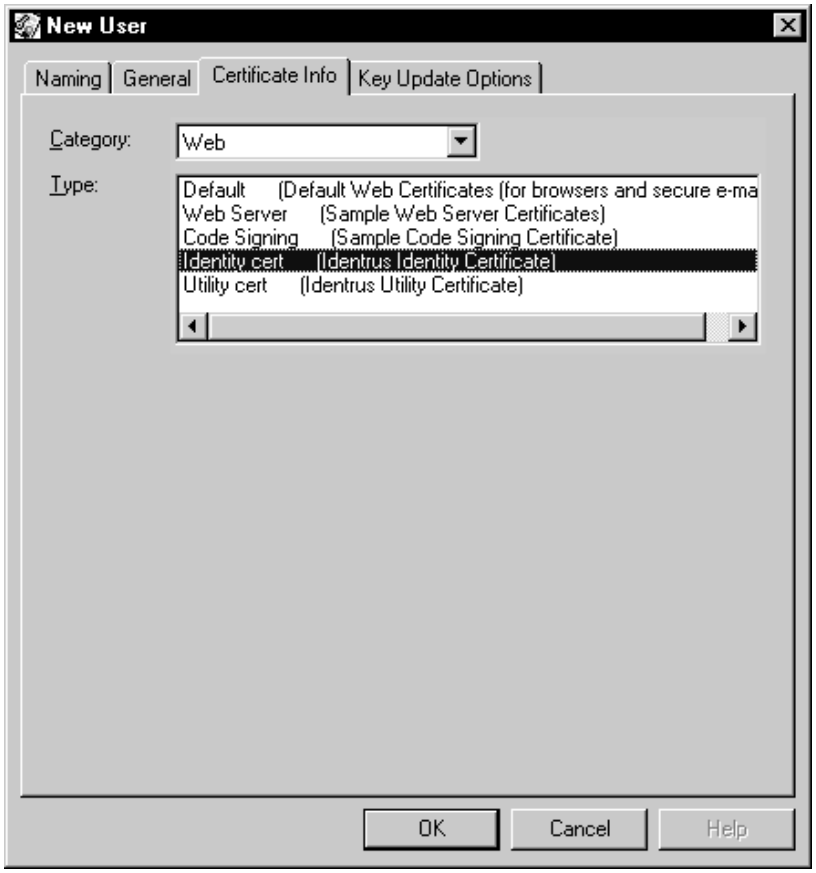
```
.
.
aia=1.3.6.1.5.5.7.1.1,n,m,DER,<place der encoding here>

[identrus_utility Verification Extensions]
;
-----------------------------------------------------------------
; Identrus Utility Certificate Type
; -----------------------------------------------------------------
keyusage=2.5.29.15,n,m,BitString,10111
certificatepolicies=2.5.29.32,n,m,DER,3082018d30......76e65722e
.
.
.
aia=1.3.6.1.5.5.7.1.1,n,m,DER,<place der encoding here>

[Advanced Settings]
.
.
.
; The Identrus certificate profile states that end-entity
certificates
; should not contain a basicConstraints extension.
[identrus_identity Advanced]
noBasicConstraints=1

[identrus_utility Advanced]
noBasicConstraints=1
.
.
.
```

This information is used by Entrust/Authority during first-time initialization to create the Identrus certificate types. Note that when adding users within Entrust/RA, you associate them with these certificate types. The Identrus certificate types appear in the New User dialog box that appears when you add a user:



Disabling revocation list checking

In the Identrus environment, OCSP responders manage information about revoked certificates. However, by default, Entrust/Authority expects to find information about its own certificate revocation status in the root CA's Directory. For this reason, Entrust/PKI turns off this check during first-time initialization, and it remains turned off.

For this release of Entrust/PKI, the identrus.certspec file contains the following entry that turns off revocation list checking:

```
;----------------------------------------
;Disable RL Checking for Foreign Certificates
;----------------------------------------
check_rls=1.2.840.113533.7.77.18,Boolean,0
```

Changing the URL of an OCSP responder

Certificates generated by Entrust/Authority contain a DER-encoded value that represents the URL of an OCSP responder. The URL represented by the DER-encoded value points to the OCSP responder that provides appropriate certificate revocation information. You provided the DER-encoded value in the identrus.certspec file, and this value was read during first-time initialization (see "Specifying the value that represents the OCSP responder URL" on page 594 and "Changes to the certificate definition file" on page 597).

Under special circumstances, you may need to change the URL for an OCSP responder after first-time initialization (for example, if you have been running a pilot project and you want to change the URL before going live). If you change the URL for an OCSP responder after first-time initialization, you must change the DER-encoded value that Entrust/Authority inserts in certificates. You change this value by editing the certificate definition file.

Note: Be aware that any certificates issued before you change the DER-encoded value will include the previous DER-encoded value. Only new certificates will include the changed DER-encoded value.

To change the DER-encoded value

1. Obtain the DER-encoded value for the new URL from:

   `http://www.entrust.com/pkitools/aiaencoder.html`

2. Export the certificate specification file. For information, see "Customizing certificates" in *Administering Entrust/PKI 6.0 on Windows*.
3. Open the certificate specification file in any text editor.
4. Search for the line that begins with "aia=", and contains the existing DER-encoded value.
5. Replace the existing DER-encoded value with the new value.

   The DER-encoded value appears in the certificate definition file twice. Ensure you replace the both instances of the existing value with the new value.
6. Save and close the certificate definition file.
7. Import the certificate specification file. For information, see "Customizing certificates" in *Administering Entrust/PKI 6.0 on Windows*.

Appendix F

PKIX and FPKI compliance

This appendix includes the section "PKIX and FPKI compliance checks" on page 604.

PKIX and FPKI compliance checks

The following sections outline the various checks that are performed when you select the *PKIX compliance* check box or the *FPKI compliance* check box in the New Policy Certificate dialog box. In all descriptions that follow, the self-signed certificate is not checked.

Strict PKIX compliance checking

When the *PKIX compliance* check box is selected in the New Policy Certificate dialog box, Entrust/Authority checks that

- the Entrust/Authority key identifier extension is present in all certificates, is non-critical, and contains a key identifier field.
- if the basic constraints extension is present and indicates that the subject is a CA, the subject key identifier extension is present and non-critical.
- the basicConstraints extension is present in all certificates except possibly the last certificate in the chain. It's also present in all certificates containing a key usage extension with the keyCertSign bit set. The basicConstraints extension, if present, is critical.
- the basic constraints extension, if present, has cA set to true.
- if the key usage extension is present and the keyCertSign bit is set, the certificate contains the basic constraints extension with cA=true.
- if the private key usage period extension is present, it's non-critical and contains at least one field (either notBefore or notAfter).
- the policy mappings extension, if present, is non-critical.
- the policy mappings extension appears only in certificates with a basic constraints extension having cA=true.
- the name constraints extension appears only in CA certificates, and is set to critical.
- the policy constraints extension appears only in CA certificates, and, if present, contains at least one of requireExplicitPolicy or inhibitPolicyMapping.

Strict FPKI compliance checking

When the *FPKI compliance* check box is selected in the New Policy Certificate dialog box, Entrust/Authority checks that

- the authority key identifier extension is present in all certificates except the self-signed certificate, and is non-critical.

- the authority key identifier extension, if present, contains a key identifier field, but no authorityCertIssuer or authorityCertSerialNumber.
- the subject key identifier extension is present in all certificates and is non-critical.
- the key usage extension is present in all certificates except the self-signed certificate.
- the key usage extension, if present, is critical and is set to one of the allowed combinations.
- if the private key usage period extension is present, it's non-critical and contains at least one field (either notBefore or notAfter).
- the certificate policies extension is present in all certificates except possibly the self-signed certificate.
- the policy mappings extension, if present, is non-critical.
- the policy mappings extension appears only in certificates with a basic constraints extension having cA=true.
- the basic constraints extension is present in all certificates. In the self-signed certificate it's non-critical. In all other certificates, it's critical. In all but possibly the last certificate in the chain, cA must be true.
- the name constraints extension appears only in CA certificates, doesn't appear in the self-signed certificate, and, if present, is set to critical.
- the policy constraints extension appears only in CA certificates and must be critical. It doesn't appear in the self-signed certificate.
- the CRL distribution points extension doesn't appear in the self-signed certificate.

References

For detailed information about the subject matter in the preceding sections, consult the following documents:

- *ITU-T Rec X.509 (2000) - Information Technology - Open Systems Interconnection - The Directory, Public-Key and Attribute Certificate Frameworks.* This document can be purchased at http:/www.itu.int/home/.
- *IETF RFC 2459: Internet X.509 Public Key Infrastructure Certificate and CRL Profile.* Currently, this document is available at http://www.rfceditor.org/rfc/rfc2459.txt.

Another excellent reference book is *Understanding Public-Key Infrastructure: Concepts, Standards, and Deployment Consideration*, Carlisle Adams and Steve Lloyd; Macmillan Technical Publishing, 1999; ISBN: 1-57870-166-x.

Appendix G

Named elliptic curves

Entrust/PKI 6.0 lets you choose an elliptic curve key to create the CA key pair. You can choose the default curve (a 192-bit elliptic curve), a custom curve, or you can use a named curve from the table below.

Note: Table 11 includes named curves from the 1998 and 2000 versions of the X9.62 standard. Release 5.0.x software supports only those curves listed in the 1998 standard. Therefore, if you attempt to use a 5.0.x client with Entrust/PKI 6.0 that uses one of the curves from the 2000 standard for its CA signing key, you will receive an error. For example, if you use Entrust/Entelligence 5.0.2 with Entrust/PKI 6.0 CA using the x962t163k1 named curve listed in the 2000 standard. There are four exceptions to this rule (these curves appear in both the 1998 and 2000 standards): x962p192r1, x962p256r1, prime192v1, prime256v1. Entrust 5.0.x clients can process x962p192r1 and x962p256r1 curves.

Table 11 lists the name and standard for all supported named curves in Entrust/PKI 6.0. For more information about elliptic curves and how to change the CA key pair algorithm, see "Changing the CA key pair algorithm" on page 386.

Table 11: Named curves supported in Entrust/PKI 6.0.

Named curve	Defined in the following standard
c2pnb163v1	X9.62-1998 J.4.1 example 1
c2pnb163v2	X9.62-1998 J.4.1 example 2
c2pnb163v1	X9.62-1998 J.4.1 example 3
c2tnb191v1	X9.62-1998 J.4.3 example 1

Table 11: Named curves supported in Entrust/PKI 6.0. (continued)

Named curve	Defined in the following standard
c2tnb191v2	X9.62-1998 J.4.3 example 2
c2tnb191v3	X9.62-1998 J.4.3 example 3
c2onb191v4	X9.62-1998 J.4.3 example 4
c2onb191v5	X9.62-1998 J.4.3 example 5
c2tnb239v1	X9.62-1998 J.4.5 example 1
c2tnb239v2	X9.62-1998 J.4.5 example 2
c2tnb239v3	X9.62-1998 J.4.5 example 3
c2onb239v4	X9.62-1998 J.4.5 example 4
c2onb239v5	X9.62-1998 J.4.5 example 5
c2tnb359v1	X9.62-1998 J.4.8 example 1
c2tnb431w1	X9.62-1998 J.4.10 example 1
prime192v1[1]	X9.62-1998 J.5.1 example 1
prime192v2	X9.62-1998 J.5.1 example 2
prime192v3	X9.62-1998 J.5.1 example 3
prime239v1	X9.62-1998 J.5.2 example 1
prime239v2	X9.62-1998 J.5.2 example 2
prime239v3	X9.62-1998 J.5.2 example 3
prime256v1[2]	X9.62-1998 J.5.3 example 1
x962t163k1	X9.62-2000 J.4.1 example 1
x962t163r1	X9.62-2000 J.4.1 example 2
x962t163r2	X9.62-2000 J.4.1 example 3
x962t193r1	X9.62-2000 J.4.2 example 1
x962t193r2	X9.62-2000 J.4.2 example 2
x962t233k1	X9.62-2000 J.4.3 example 1
x962t233r1	X9.62-2000 J.4.3 example 2
x962t239k1	X9.62-2000 J.4.4 example 1
x962t283k1	X9.62-2000 J.4.5 example 1
x962t283r1	X9.62-2000 J.4.5 example 2

Table 11: Named curves supported in Entrust/PKI 6.0. (continued)

Named curve	Defined in the following standard
x962t409k1	X9.62-2000 J.4.6 example 1
x962t409r1	X9.62-2000 J.4.6 example 2
x962t571k1	X9.62-2000 J.4.7 example 1
x962t571r1	X9.62-2000 J.4.7 example 2
x962p160k1	X9.62-2000 J.5.1 example 1
x962p160r1	X9.62-2000 J.5.1 example 2
x962p160r2	X9.62-2000 J.5.1 example 3
x962p192k1	X9.62-2000 J.5.2 example 1
x962p192r1[3]	X9.62-2000 J.5.2 example 2
x962p224k1	X9.62-2000 J.5.3 example 1
x962p224r1	X9.62-2000 J.5.3 example 2
x962p256k1	X9.62-2000 J.5.4 example 1
x962p256r1[4]	X9.62-2000 J.5.4 example 2
x962p384r1	X9.62-2000 J.5.5 example 1
x962p521r1	X9.62-2000 J.5.6 example 1

1. This curve is the same as x962p192r1. These curves appear in both versions of the standard.
2. This curve is the same as x962p256r1. These curves appear in both versions of the standard.
3. This curve is the same as prime192v1. These curves appear in both versions of the standard.
4. This curve is the same as prime256v1. These curves appear in both versions of the standard.

Appendix H

Server security

This appendix includes a checklist that describes a number of security issues for you to consider when you are running Entrust/PKI 6.0.

You can also consult the ISS software (and brochure) on the CD "ISS System Scanner 1.7.1 for Solaris & HP-UX and ISS Internet Scanner v6.0 for Microsoft Windows (19991116)".

ISS software provides extensive, detailed information about how to secure your system. You can also find more information at the following web site:

```
http://www.iss.net/securing_e-business/security_products/
security_assessment/system_scanner/
```

This appendix includes the section "Security checklist" on page 612.

Security checklist

You should secure all Entrust/PKI 6.0 and backup servers against unauthorized physical and network access. Table 12 on page 612 describes how you can improve the security of your Windows servers that will host Entrust/PKI 6.0.

For more information, see "Securing your environment and data" in *Installing Entrust/PKI 6.0 on Windows*.

Table 12: Security checklist for Windows servers

Security issue	Done? Y/N
Restrict physical access to the servers that will host Entrust/PKI 6.0. For example, put the servers in a locked room to which only a few of your most highly trusted administrative users have access. Set up an audit log of users who visit this room.	
Place servers by firewalls or behind double firewalls.	
Dedicate the Entrust/Authority server primarily to Entrust/PKI 6.0. You might run other applications on this server, such as billing, monitoring, activation code distribution, or intrusion detection software. As much as possible, avoid applications that are network-dependent (such as DNS and/or NIS). Don't allow remote open database connectivity (ODBC) access to the Entrust/Authority server.	
Install Entrust/PKI 6.0 on a drive whose file system is configured as NTFS (New Technology File System), not FAT (File Allocation Table). In fact, the Entrust/Authority database requires NTFS; Informix won't install on a drive that uses the FAT file system.	
Disable inbound remote logins (for example, through telnet or rlogin), and delete unused program files such as ftp.exe, rasdial.exe, and telnet.exe.	
Arrange for a formal security audit of physical and system security. Consider setting up a software tool (for example, Internet Security Scanner) to go through your system and look for vulnerabilities.	
Disable NetBIOS-over-TCP/IP (NBT).	
Disable the Guest account and delete all other user accounts. Only the Windows administrator account should remain. The Site Planner should own this account.	
Change the Windows administrator username from "administrator" to something less obvious, such as "Iou$1Db0Nso0%" ("I owe you money one day but nothing now so no interest"), and choose an equally cryptic but memorable password. Then, attackers must guess the administrator username also, not just the administrator password. Don't use the example administrator username provided above.	
Set password security options to specify minimum and maximum password age, password uniqueness, and password length.	

Table 12: Security checklist (continued)for Windows servers (continued)

Security issue	Done? Y/N
Enable the event-auditing system, and audit all failed operations and low-frequency successes.	
If your Directory resides on the same server as Entrust/Authority, block all TCP/IP ports (including ftp on port 21 and telnet on port 23) except 389, 710, and 829, which are the default ports used by Entrust/PKI. If your Directory resides on a different server, lock port 389 as well. If your Entrust/PKI 6.0 installation needs to be compatible with Entrust/PKI 4.0 or earlier, don't block port 709.	
The preceding recommendations will help you improve your server security. For current security strategies, consult Microsoft or industry publications such as http://www.developer.earthweb.com/dlink.index-jhtml.72.950.-.0.jhtml. This site contains a list of online resources for network security. *Microsoft Windows NT 3.5 Guidelines for Security, Audit, and Control*, Citibank NA, Coopers & Lybrand, The Institute of Internal Auditors, and Microsoft; Microsoft Press, Redmond, 1994; ISBN 1-556-15814-9 (this book applies to later versions of Windows NT as well).	
FIPS 140-1 level 2 specifies that it must not be possible to open or remove any removable covers or panels from your CPU without leaving detectable signs, such as a broken label or label residue on the enclosure. To operate in FIPS 140-1 level 2 compliant mode, Administrators (or Security Officers) must apply tamper-evident labels to the CPU casing as shown in the following diagram. The labels included in your Entrust/PKI 6.0 package have been successfully tested and comply with FIPS 140-1 level 2 requirements for physical security (tamper evidence).	

Index

A

Acceptable policy OIDs policy setting 175
AcceptOnlyOAEP
  in entrust.ini file 485, 495
accessing certificate definition files 430
activated report field 31
activation codes
  in reports 31
Add Attribute dialog box 40
added report field 31
Added state
  on a report 30
adding
  database fields 332–335
  Directory attribute values 41
  Directory attributes 39
  Directory entry attributes 40
  information to CA certificates 325–327
  information to cross-certificates 314–324
  information to policy certificates 306–313
  information to user certificates 282–305
  Release 4.0-compatible users 191
  user types to the Directory Browser 193
  users
    changing default settings for 189
adding users
  using templates 189
Admin Main Window Settings, in entrustra.ini file 524
administer groups
  permissions to 142
Administration Policy
  lifetime of cross-certificates 205
Administration Service
  password processing 136
administration subsystem
  .ini file settings 545
  disconnect 545
  listen port 528
  time-out 545
Administration susbsystem
  increasing number of processes 75, 529
administrative permissions
  and searchbases 142
  roles 142
  viewing 151, 152
administrative tasks 33
Administrator
  changing your profile password 52
  does not have access message 458
  does not have certificate message 449
administrator
  roles 142
Advanced Settings section
  in entmgr.ini 78, 79
AIA extension, in Identrus environment 594
aiaencoder.html 594
Alarm severity rating 445
alarms
  7641 445
  7654 446
  7655 447
  7663 447
  7664 447
  7665 448
  7666 448
  7667 448
  7674 448
  7678 449
  7679 449
  7680 449
  7681 449
  7684 450
  7685 450
  7687 450
  7689 450
  7691 450
  7693 450
  7695 450
  7697 451
  7698 451
  7705 452
  7708 452
  7711 453
  7713 453
  7714 453
  7715 453
  7716 453
  7718 454
  7724 454
  7726 454
  7761 458
  7770 459
  7772 459
  7783 460
  7803 462
  7808 462
  7812 463
  7814 463
  7816 463
  7817 463
  7819 463

7822 464
7823 464
7827 464
7829 464
7830 464
7831 464
7832 464
7833 465
7834 465
7835 465
7836 465
7855 466
7856 466
7857 466
7865 467
7866 467, 468
7867 468
7873 468
7875 468
7876 468
7877 468
7878 469
7884 469
7885 469
7886 469
7887 469
7890 470
7891 470
7892 470
7896 470
7900 471
7901 471
7902 471
7903 471
7904 471
7914 473
7915 473
7929 473
7931 474
7932 474
7933 474
7943 474
7944 474
7945 475
7946 475
7947 475
7949 475
7950 475
7965 476
7977 477
7978 478
7980 478
7983 478
7985 479
7986 479
7987 479
7988 480
7991 481
AlgName, in entrust.ini file 511
algorithm
changing
certificate hashing 73
SEP encryption 73
choosing 74
ECDSA CA key pair algorithm 387
length of user encryption key 74
Algorithm for digital signature policy setting 171
AliasName
in entrust.ini file 511
Allow CA personal address book use policy setting 173
Allow CAPI Export policy setting 179
Allow CAPI Key Export policy setting 180
Allow personal address book use policy setting 173
Allow PKCS #12 Export policy setting 178
Allow unprotected CAPI Export policy setting 180
archiving
backups 124
ARL 206, 238
cache size 488
caching 486, 488
changing the lifetime 398
defined 395
expiry message 471
validation failed message 471
writing to the Directory 369
ARLCacheEnabled, in entrust.ini file 486
ASH Information section 545
in entrust.ini file 518
ASH, in entrust.ini file 518
AshDisabled, in entmgr.ini file 529
ASHdn, in entrust.ini file 518
ASHlisten, in entmgr.ini file 528
AshNumProcs, in entmgr.ini file 529
ASHPort, in entrust.ini file 518
ASHServer, in entrust.ini file 519
AshSrvIdleTime
in entmgr.ini file 529
AshSrvIdleTime, in entmgr.ini file 529
AshSrvNotAfter, in entmgr.ini file 530
AshSrvNotBefore, in entmgr.ini file 530
AshSrvPeriod, in entmgr.ini file 530
ASHTimeout, in entrust.ini file 519
ASN1Attributes, in entrust.ini file 501
attribute
default values for variables 300
invalid extension message 470
missing extension message 470

attribute certificates
automatic reissue
timer controls 536
attribute values
deleting 42
AttributeName
entrust.ini file 514
in entrustra.ini file 521
AttributeOptions section, in entrust.ini file 514
attributes
adding
to Directory entries 40
to the template definitions file 194
deleting 42
Directory
adding and deleting 39
Audit Log Entry Details dialog box 22
audit logs 18–29
and MAC 18
archiving policy 18
checking 432
clearing the Audit Log Entries property page 25
definition 18
description of fields
in Entrust/RA 23
in the Windows Event Viewer 29
error messages in manager.log and mgraudit.log 19
levels sent to Event Viewer 544
listing 445, 551
location on disk 18
maximum file size 18
permission to view 158
revocation lists 482
saving, in Entrust/RA 25
secured audit logs 18
sending, to the Windows Event Viewer 27
sorting, in Entrust/RA 24
states 24
suppressing 398
using to verifying backups 124
viewing
administrative permissions 18
audit log details in Entrust/RA 22
in Entrust/RA 19
in the Windows Event Viewer 28
audit record. See audit logs, logs
Audit Search dialog box 21
audit trail
audit trail integrity check failed message 453
AuditCRLWrite 398
Auditor
pre-defined role 143
audits logged for revocation lists 482
authauto.ini file
restoring 129
AuthMethod
in entrust.ini file 501
authority revocation list. *See* ARL
Authority, in entrust.ini file 486
authorityKeyIdentifier extension 406
authorization code
Security Officer's
recovery 46
authorization report field 31
Authorization Requested dialog box 211, 216, 224, 230, 231, 237, 240
Authorization section
in entmgr.ini file 541
authorizations
administrator
for sensitive operations 57
enabling Master Users 57
recovering Security Officers 45
resetting number of Security Officers to one 59
Security Officer 142
setting number of Master Users 57, 377
AuthTimeout, in entrust.ini file 486
Auto section
in entmgr.ini file 545
autoFile
in entmgr.ini file 545
autologin 396
automatic
backups
Entrust/Authority database 108, 113–116
reissue of attribute certificates
timer controls 536
availability. See high availability
AVASorting
in entrust.ini file 501

B

backing up
automatic, frequency setting 532
Entrust/Authority data 106
Entrust/Authority database 361
configure settings 120, 362
configure settings with command line 120
display settings 362
display settings with commmand line 120
how to do automatically 113
how to do manually 122
backing up and restoring the CA key pair 96
backup
automatic backup settings 532

backup and restore
and disaster recovery planning 561
backup directories
about 111
changing name and location 118
changing number kept online 117
contents 112
Database folder 112
format of name 112
number kept online 112
backup directory
changing 118
contents 112
default location 111
number kept online 117
backup files
incremental backups 534
organization 112
backup folders
contents 112
backups
archiving 124
automatic 108
scheduling 113–116
automatic backup settings 532
backup canceled message 464
backup failed message 464
changing number kept online 117
default schedule 108, 113
default settings 108, 113
directories. See backup directories
Entrust/Authority data files 106
Entrust/Authority database 106
error on cleanup 468
frequency 106, 108, 113
full 108–110
full data backup failed message 473
how to do automatically 113
how to do manually 122
how to schedule 106, 108, 113
increasing room for 117
incremental 108–110
location 111–113
manual 122–124
name of directories 112
number kept online 112
restoring changes not backed up 108
restoring from 127–132
scheduling 106, 108, 113
space required 108
transaction logs 108, 128
verifying 124
BadHostsLogFile, in entrust.ini file 502
BannerFile, in entrust.ini file 487
basicConstraints 255, 421
basicConstraints extension 405
Begin Online Cross-Certification dialog box 210
Binary 254, 420
binding
CA Directory entry to Entrust/Authority 53
bulk commands
permission to generate 158
Bulk Console Defaults section, in entrustra.ini file 523

C

CA
attribute certificate validation message 470
CA key compromise 394
certificate
verification failed 469
certificate is revoked message 451
certificate may be revoked message 451
cross-certifying 203–241
default searchbase 523
Directory Access passwords 53
how to change 53
establishing trust with other CAs 203–241
exporting the CA certificate 383
key has expired message 453
key pair 389
key reload period 539
key revocation error message 454
key update failed message 454
maintaining, via command line 381
permissions to administer 158
personal address book 173
profile recovery 381
recovering CA profile 381
revoking the CA signing key 394
signing key pair 389
do not update for pre-5.0 clients 391
subordinate CA
certificate expired message 449
certificate nearing expiry message 450
SEP requests 528
subordinate CAs
.ini file settings 543
superior CA
asking for a certificate 253, 419
superior CAs
.ini file settings 543
CA certificate
exporting 88
CA certificate category 271, 272

CA certificates
adding information to 325–327
customizing 274, 325–327
CA Cross-Certificate Request dialog box 220, 222
CA hardware devices
identifying libraries 101
redundancy 100
using to store the CA key pair and database key 100
CA hierarchy
in Identrus environment 590
CA key pair
moving to a PKCS #11 v1 hardware device 97
updating 389
updating from one device to another 98
updating on hardware 96
ca keyrecover-all command 394
CA link certificate
issuing 325
CA link certificates
customizing 325
ca move-to-hardware command 387
ca resynch-next-cert-serial command 382
CA verification certificate
customizing 325
issuing 325
CAPI profile export. See profile export
CAs
cross-certification of 204
CAST-128 algorithm 74
categories
for CA certificates 271
for cross-certificates 271
for Enterprise certificates 271
for policy certificates 271
for Web certificates 271
of certificates 271
categories. See certificate categories 272
CDP section
in entmgr.ini file 546
certificate
categories 271
extensions 271
request 254, 420
types 271
selecting during cross-certification 224
certificate attributes
for policy certificate types 307
policy certificate types 310
certificate categories
about 271
CA 271, 272
cross-certificate 271, 272
default types 272
dividing into types 272
Enterprise 271, 272
policy 271, 272
Web 271, 272
certificate definition files
accessing 430
certificate extension
authorityKeyIdentifier 406
basicConstraints 405
modifying default behavior 402
privateKeyUsage 402, 404
subjectKeyIdentifier 406
certificate extensions
cross-certificate types 318
defining
in master.certspec file 271, 332
excluding 328
for user certificate types 284, 315
removing 336
removing X.509 certificate extensions 337
user certificate types 287
certificate type. See also user certificate types
certificate types 282
about 272
adding different kinds of content 273
certificate attributes 307
certificate extensions 284, 315
Client settings 272
comprised of 307
Default 272
definition of 272
definitions 283, 307, 315
describing in cross-certification dialog boxes 317
describing in New User dialog box 286
describing in New User Policy dialog box 309
Entrust/Profile Server 303
Entrust/Timestamp 303
excluding extensions 284, 315
Link 272
making obsolete 330
names in New User dialog box 286
naming in cross-certification dialog boxes 317
naming in New User Policy dialog box 309
New User dialog box 282, 283
New User Policy dialog box 306, 307
sample Web 303
selecting during cross-certification 216
using, policy 306
certificate types. See also user certificate types
certificate, revocation reason 408
CertificateCacheEnabled, in entrust.ini file 487
CertificateCacheSize, in entrust.ini file 487

certificates
about 271–276
added to revocation list message 463
adding content 273
adding information to users' 273
and their types 272
assigning a policy certificate to a role 187
CA certificate is revoked message 451
CA certificate may be revoked message 451
cache size 487
caching 487
certificate is invalid message 463
cross. *See* cross-certificates
customizing 269
data type 293, 295, 313
default types 272
defaults
overriding 522
digitally signed 273
encryption policy changed message 465
end user revoked message 468
end user validation failed message 475
exporting CA certificate 383
hashing algorithm 73, 373
changing 73
invalid serial number 469
issued from certificate categories 271
keyUsage extension 291
link 390
listed in Combined CRL 77, 81
making obsolete 330
optionality 292
permission to administer 158
policy, adding to 306
reasons for customizing 273
recommendation for content 272
revoked certificate message 467
serial number counter message 465
serial number, resetting 382
signature changed message 463
types of 272
user, for Identrus 596
verification failed 469
verification policy changed message 465
Chaining Directories 268
Change Directory Password dialog box 55
Change Master User or Directory Password dialog 50
Change Profile Password dialog box 52
changing
a role's settings 163
Administrator profile password 52
CA Directory password 53
certificate hashing algorithm 73
client-side settings, via command line 428
default settings
for adding a user 189
Directory Administrator password 54
Entrust/Authority database password 139
Entrust/RA preferences 68
Master User password 48, 351
password rules 50
Security Officer authorizations 425
SEP encryption algorithm 73
changing the CA key pair algorithm at the command line 386
checklist, security 612
clearing audit logs in Entrust/RA 25
Client settings certificate type 272
client-side settings 300, 545
changing 428
displaying 427
ClientSocketTimeout, in entrust.ini file 487
ClientType, in entrust.ini file 487
clustering
and service availability 566
CMPDisabled
in entmgr.ini file 530
CMPlisten
in entmgr.ini file 528
CMPNumProcs
in entmgr.ini file 531
CMS init user request error 462
CMS message processing error 463
CMS request refused 463
Combined CRL 77–82
and initialization 78
changing lifetime 79
creating 78
publishing to file 80
setting threshold 81
turning on and off 78
using 77, 594
CombinedCRLfile, in entmgr.ini 80
CombinedCRLLifetime
in entmgr.ini 79
CombinedCRLLifetime, in entmgr.ini 79
CombinedCRLThreshold, in entmgr.ini 81
command line 341–444
adding information to users 415
AdditionalUserObjectClasses 415
administering the database 360
administering the Directory 368
audit logs, checking 432
autologin 356
backing up the database 361
ca keyrecover-all 394
ca move-to-hardware 387
ca resynch-next-cert-serial 382

changing certificate hashing algorithm 373
changing client-side settings 428
changing Directory attributes 418
changing Master User authorizations 377
changing Master User passwords 351
changing product compatibility 413
changing protocol encryption algorithm 374
changing Security Officer authorizations 425
changing the CA key pair algorithm 386
changing user encryption key length 375
changing user object classes 416
choosing a custom ECDSA elliptic curve 388
choosing a named ECDSA elliptic curve 388
combined CRLs
  lifetime 411
  turning on and off 411
command summary 443
configuration variables 364
configure settings for database validation 366
creating Security Officers 423
db backup config command 121
db backup full command 123
db backup incremental command 124
db backup query command 120
db get DNEncoding 418
db get KeyIdMode 407
db get UseCombinedCRL 410
db get UserCertBasicConst 405
db integrity validate command 126
db re-encrypt command 137
db set UserCertBasicConst 406
Directory Administrator
  DN, changing 371
disable automatic login 356
disabling Entrust/Authority subsystems 358
display encoding of Directory attributes 418
display Entrust/Authority subsystem status 357
display product compatibility 413
display status of automatic login 355
display user object classes 415
displaying client-side settings 427
displaying the CA key pair algorithm 387
distributing start-up information to users 378
dsa diradmin command 372
dsa passwd command 370, 371
dsa restore all command 135, 369
dsa restore command 135, 369
enabling Entrust/Authority subsystems 357, 358
Entrust/PKI performance counters 434
exit command 348
exporting the CA certificate 383
fcs export 430
fcs import 431
get CrlUseDefRevokeReason 408
get UserCertPrivateKeyUse 403
getting started 343
help 438
init command 345
initializing Entrust/PKI before using 345
listing Security Officers 428
logging in 349
logging out 349
maintaining the CA 381
monitoring Entrust/PKI 432
officer auths query command 426
officer auths set command 426
officer create command 424
officer recover command 425
policy authcode-dist 379
policy auths disable 378
policy auths enable 377
policy auths query 377
policy hasher query command 373
policy protocol query command 375
policy proto-enc command 375
policy user-enc-keylen 376
pushrls command 370
recover CA 381
recovering keys for all clients 394
recovering Security Officers 423, 424
recovering the CA profile 381
re-encrypting the database 360
resetting certificate serial number 382
restoring CA data to the Directory 368
revoking the CA signing key 394
security classes 442
security policy settings 373
show settings for database validation 366
show status of Entrust/Authority service 352
starting 347
starting commands automatically 354
starting Entrust/Authority 352
starting Entrust/Authority service 353
stopping Entrust/Authority 352
stopping Entrust/Authority service 354
stopping the command shell 347
subordinate read-subcert 255, 421
superior CA
  request certificate 253, 419
to change the CA key pair algorithm 388
updating the time zone 363
using combined CRLs 410
validating the database 365
writing CRLs and ARLs to the Directory 369
writing the ECDSA elliptic curve algorithm to a file 389
Command Line Utilities window 130, 131
comment character 485, 520, 527

comments
in entmgr.ini file 527
in entrust.ini file 485
in entrustra.ini file 520
common name (cn)
how it is formed 197
structure for Person user type 197
Comms section, in entmgr.ini file 528
Compatlevel command 413, 414
Complete Online Cross-Certification dialog box 215
configuring
entmgr.ini file 527
entrust.ini 485
entrustra.ini file 520
number of administrator authorizations
for sensitive operations 57
subsystems 357
configuring Entrust/Authority for high availability 576
ConnectTimeout
in entrust.ini file 487
Content scanner SMTP policy setting 183
corruption
data file 468
count
in entrustra.ini file 521
creating
cross-certificate types 314
hierarchy of CAs
online 243
policy certificate types 307
reports 30
roles 142, 161
Security Officers
via command line 423
user certificate types 283
user reports 30
CRL
combined CRL larger than threshold message 452
defined 395
expired message 474
push 396
validation failed message 474
CRL distribution points 546
CRL section
in entmgr.ini 80, 81, 83
CrlCacheEnabled
in entrust.ini file 488
CRLs
caching 488, 489
combined CRLs 77–82, 410
lifetime 411
turning on and off 411
how often checked 533
partitioned 77, 83, 546
standards 77
writing to the Directory 369
CrossCertARLCacheEnabled
in entrust.ini file 488
CrossCertARLCacheSize
in entrust.ini file 488
CrossCertCACacheEnabled
in entrust.ini file 488, 489
CrossCertCACacheSize
in entrust.ini file 489
CrossCertCRLCacheEnabled
in entrust.ini file 489
CrossCertCRLCacheSize
in entrust.ini file 489
CrossCertDebug
in entrust.ini file 489
cross-certificate category 271, 272
Cross-Certificate I Requested property page 213, 232, 237
Cross-Certificate I Signed property page 240
cross-certificate types
applications of certificate extensions 319
editing definitions 315
examples 323
cross-certificates
adding information to 314–324
cache size 489
caching 488, 489
creating types of 314
customizing 314–324
definition 205
expired message 449
expiring 238
function of 314
how to revoke 238
how to update 241
lifetime 205, 217, 224, 230
mutual versus unilateral cross-certification 206
path checking 494
receiving verification public key 208
revoking 238
updating 205, 217, 224, 230, 241
used by applications 206
uses of 205
validation failed message 471
verifying validity 208, 218, 541
written to Directory message 463
cross-certification 203–241
description 204
displaying information about CAs 233
how to revoke cross-certificates 238
how to update cross-certificates 241
legal agreements 204
methods 207, 208

mutual 206, 218
example 206
of CAs 204
offline
description 207, 218
how to complete 221
how to initiate 219
method 207, 218
mutual 218
requirements 218
unilateral 218
online
choosing type 211
completing 208
description 207, 208
how to complete 214
how to initiate 209
how to obtain password 213
initiating 208
method 207, 208
mutual 208
protocols 208, 209
requirements 208
unilateral 208
revoking cross-certificates 203, 238
status
Complete 212
Pending 212
Revoked 240
Stored in
Database but not in the DSA 232, 237
unilateral 205
updating 241
updating cross-certificates 205, 217, 224, 230, 241
validating certificates 206
cross-certification dialog box
assigning certificate types 317
CrossCertMaximumDepth, in entrust.ini file 490
CryptoAPI profile export 179
supported algorithms 171
Cryptographic hardware not found 460
CryptokiLibrary95, in entrust.ini file 490
CryptokiLibraryNT
in entmgr.ini file 527
in entrust.ini file 490
CryptokiV2Library
in entmgr.ini file 527
in entrust.ini file 490
custom ECDSA curves 387
customizing
configuration data for Entrust/PKI clients 485
entmgr.ini file 527
entrust.ini file 485
Entrust/Authority 527
Entrust/PKI 141
Entrust/RA configuration data 520
entrustra.ini file 520
template definition file 189
the New User dialog box 189
user roles 142
See also roles

D

data file
cleanup error 468
corrupted entry 468
is missing 469
missing entry 469
data files
restoring from backup 128
data0
in entrust.ini file 511
database
backing up 106, 361
backup failed message 464
Entrust/Authority
maintaining 105, 360–363
integrity check canceled message 464
integrity check failed message 466
re-encrypting 360
restoring from backups 130
validating 365
configure settings for 366
display settings for 366
write error 468
Database Backup Settings dialog box 114–119
database fields
customizing 332–335
in New User dialog box 332
Database folder
contents 112
in backup folder 112
database key 100
database protection 100
database section 542
database-level replication 562
db backup config command 121, 122, 362
db backup full command 123, 361
db backup incremental command 124, 361
db backup query command 120, 362
db get AdditionalUserObjectClasses command 415
db get Compatlevel command 413
db get CrlUseDefRevokeReason command 408
db get DNEncoding command 418
db get KeyIdMode command 407
db get OffLineARL 399

db get UseCombinedCRL command 410
db get UserCertBasicConst command 405
db get UserCertPrivateKeyUse command 403
db integrity config command 366
db integrity query command 366
db integrity validate command 126, 366
db re-encrypt command 137, 360
db set Compatlevel command 414
db set OffLineARL 399
db set UserCertBasicConst command 406
dbloginname, in entmgr.ini file 531
dbloginpw, in entmgr.ini file 531
dbname, in entmgr.ini file 543
deactivated reason report field 31
Deactivated state
  on a report 30
DebugFile
  in entrust.ini file 519
default backup schedule 108, 113
Default certificate type 272
default ECDSA curve 387
default values
  variable definitions 300
DefaultCategory
  in entrustra.ini file 522
DefaultContext_time_req
  in entrust.ini file 490, 517
DefaultCredentials_time_req
  in entrust.ini file 490, 517
DefaultProfileLocation
  in entrust.ini file 490
defaults
  changing settings
    for adding a user 189
Delay single login register policy setting 181
delayed replication 562
deleting
  a user's Directory entry 39
  certificates 330
  Directory attribute values 42
  Directory attributes 39, 42
dialog boxes
  Add Attribute 40
  assigning certificate types 299
  Audit Log Entry Details 22
  Audit Search 21
  Authorization Requested 211, 216, 224, 230, 231, 237, 240
  Begin Online Cross-Certification 210, 211
  CA Cross-Certificate Request 220, 222
  Change Directory Password 55
  Change Profile Password 52
  Complete Online Cross-Certification 215
  cross-certification 317
  Database Backup Settings 114–119
  Find (Directory Browser) 38
  New User 200, 282, 283, 286
  New User Policy 306, 307, 309
  Operation Complete 216, 231, 237
  Operation Completed Successfully 211
  Preferences 68
  Save As 190
  Select Cross-Certificate Type 224
  Sign Cross-Certificate 223
DirAdmin Defaults section
  in entrustra.ini file 523
DirAdmin Window Settings, in entrustra.ini file 525
Directory
  address and port number 499
  administering via command line 368
  Administrator DN, changing
    Certification Authority 371
  Administrator password
    changing 54
  ascii or binary mode 529
  attributes
    changing 418
    displaying how encoded 418
  backups synchronized with database backups 106
  chaining 268
  command line
    Directory passwords, changing 370
  connection settings 501, 517
  connection time-out 503
  connectivity
    offline cross-certification 218
    online cross-certification 208
  CRL expired message 474
  database backup failed message 465
  deleting an entry 39
  finding users 37
  IP address and port 528
  not responding message 475
  operation alarm 448
  options 505, 507
  passwords, changing
    Certification Authority 53, 370
  permission to administer 159
  protocol used 528
  protocol used to connect 503
  publishing Combined CRL 77, 79, 81
  read failure message 475
  restore canceled message 463
  restoring CA data via command line 368
  restoring Entrust/Authority data to 133
  search
    number of users to retrieve 64
  search option tags 505

search settings 508, 509, 510, 511
search size limit 503
searchbases for cross-certification 204, 205, 206, 208
searches
number of users 65
storing cross-certificates 205
using shadow Directories 574
write failure message 474
writing CRLs and ARLs to Directory 369
Directory Administrator DN, changing 371
Directory attributes
adding and deleting 39
Directory Browser 34–43
.ini file settings 520
add directory attribute failed message 450
add entry failed message 450
adding attribute values 41
adding attributes 40
adding user types to 193
delete attribute failed message 450
delete attribute value failed message 450
delete entry failed message 450
deleting attribute values 42
deleting attributes 42
Directory Browser button on the Entrust/RA toolbar 35
finding entries 37
replace directory attribute failed message 450
replacing attribute values 41
Directory Connection Settings section
in entrust.ini file 501, 517
Directory entries
adding 35
adding attribute values 41
adding attributes 40
deleting 38
finding 37
replacing attribute values 41
Directory Protocol, in entmgr.ini file 528
Directory Search Criteria section, in entrustra.ini file 520
DirectoryOperationTimeLimit, in entrust.ini file 503
DirectoryPassword, in entrust.ini file 491
DirectoryProtocol, in entrust.ini file 503
DirectorySearchSizeLimit, in entrust.ini file 503
DirectoryServerLimit, in entrust.ini file 504
Disable single login policy setting 169
DisableDbBackup, in entmgr.ini file 531
disabling
Master Users 58
disabling hardware-based database protection 103
disaster recovery 553–588
and enhanced database protection 564
backup and restore strategy 561
case studies 583
ensuring successful transfer of services 565
example architecture 564
for Entrust/PKI 564
remote replication 561
disk arrays
intelligent 570
disk errors
and corruption of Entrust/Authority database 126
disk mirroring 569
disk space
and backups 117
disk space too low 459, 462
disk-level replication 562
distinguished name (dn) 30
distribution points
for certificate revocation lists 77, 83, 546
DN Change state, on a report 30
DN encoding format policy setting 173
dn. See distinguished name
DNs
encoding format policy 173
DNSAttempts, in entrust.ini file 504
Do not process anyPolicy policy setting 177
Do not process policy mappings policy setting 176
DOS Extensions section, in entrust.ini file 512
dsa diradmin command 372
dsa passwd command 370, 371
dsa pushrls command 370
dsa restore all command 135, 369
dsa restore ca command 135, 369

E

ECDSA CA key pair algorithm
overview 387
ECDSA curves
custom curves 387
default curve 387
named curves 387
editing
templates definition file 191
e-mail
user e-mail message 465
Enable the use of a Cert cache policy setting 183
Enable the use of a CRL cache policy setting 182
Enable the use of a XCert cache policy setting 183
Enable the use of an ARL cache policy setting 182
enabling and disabling hardware-based database protection 102
enabling hardware-based database protection 102
encr. cert revok. reason report field 32
encrypt cert desc report field 31
encrypt cert expiry report field 32
encrypt cert issue report field 32

encrypt cert revoked report field 32
encryption
algorithms, setting 170
of Entrust/Authority database
backup after 122
re-encryption 136
EncryptWith, in entrust.ini file 491
EncrytWithOAEP
in entrust.ini file 492
EncrytWithOAEP, in entrust.ini file 492
end updt pend report field 31
end user
pre-defined non-administrative role 142
Enforce identity usage policy setting 173
Enforce S/MIME policy setting 175
Enforce token usage policy setting 172
enforce_identity_use 300
.entadmin bulk file 523
EntBackupBackupsToKeepOnline, in entmgr.ini file 532
entbackupFrequency
in entmgr.ini file 532
entbackupFrequency, in entmgr.ini file 532
EntBackupNotAfter
in entmgr.ini file 532
EntBackupNotAfter, in entmgr.ini file 532
EntBackupNotBefore
in entmgr.ini file 532
EntBackupNotBefore, in entmgr.ini file 532
EntBackupPeriod, in entmgr.ini file 532
EntCrlCheckPeriod, in entmgr.ini file 533
EntDailyAuditNotAfter, in entmgr.ini file 533
EntDailyAuditNotBefore, in entmgr.ini file 533
EntDailyAuditPeriod, in entmgr.ini file 533
Enterprise certificate category 271, 272
Enterprise certificates
adding database fields 332–335
adding information to 282–305
assigning types 282
creating types of 283
customizing 282–305
function of 282
EntIncBackupNotAfter, in entmgr.ini file 534
EntIncBackupNotBefore, in entmgr.ini file 534
EntIncBackupPeriod, in entmgr.ini file 534
EntIntegrityNotAfter
in entmgr.ini file 534
EntIntegrityNotBefore, in entmgr.ini file 534
EntIntegrityPeriod
in entmgr.ini file 535
EntLicenseThreshold, in entmgr.ini file 535
entmasterIdletime, in entmgr.ini file 537
entmgr.ini file 527
Advanced Settings section 78, 79
ASH Information section 545
AshDisabled 529
ASHlisten 528
AshNumProcs 529
AshSrvIdleTime 529
AshSrvNotAfter 530
AshSrvNotBefore 530
AshSrvPeriod 530
Attribute Name 545
Authorization section 541
Auto section 545
autoFile 545
CDP section 546
changing Combined CRL lifetime 79
CMPDisabled 530
CMPlisten 528
CMPNumProcs 531
CombinedCRLFile 80
CombinedCRLLifetime 79
CombinedCRLThreshold 81
command line configuration 364
commenting 527
Comms section 528
CRL section 80, 81, 83
CryptokiLibraryNT 527
CryptokiV2Library 527
customizing 527
database section 542
dbloginname 531
dbloginpw 531
dbname 543
Default Variable Values 545
Directory Protocol 528
DisableDBBackup 531
EntBackupBackupsToKeepOnline 532
entbackupFrequency 532
EntBackupNotAfter 532
EntBackupNotBefore 532
EntBackupPeriod 532
EntCrlCheckPeriod 533
EntDailyAuditNotAfter 533
EntDailyAuditNotBefore 533
EntDailyAuditPeriod 533
EntIncBackupNotAfter 534
EntIncBackupNotBefore 534
EntIncBackupPeriod 534
EntIntegrityNotAfter 534
EntIntegrityNotBefore 534
EntIntegrityPeriod 535
EntLicenseThreshold 535
entmasterIdletime 537
entmgrMinFree 535
entmgrWarnFree 536
Entrust Settings section 527
entUserAttrCertReissueNotAfter 536

entUserAttrCertReissueNotBefore 536
entUserAttrCertReissuePeriod 536
entUserStateHistTrimFrequency 538
EntUserStateHistTrimIdleTime 537
entUserStateHistTrimMax 537
entUserStateHistTrimNotAfter 537
entUserStateHistTrimNotBefore 537
entUserStateHistTrimPeriod 538
IgnorePKCS10Signature 541
Key 542
Level 544
ListenTimeout 528
Logging section 543
Login section 529
ManagerBackup 538
MaxRestarts 538
Mgrlisten 528
NumFiles 545
OfflineSuperior 538
OrgRoleFirstOfficer 541
Policy section 541
ProtocolOverride 528
publishing Combined CRL to file 80
Reissue Block 546
Reissue Delta 546
ReLoadCAKeysPeriod 539
Reports section 547
SEPDisabled 539
SEPNumProcs 539
SerialNumber 542
Server 528
serverType 529
Session Disconnect 545
Session Timeout 545
setting threshold for Combined CRL 81, 83
Superior section 543
SuperiorAlgID 543
SuperiorCAAvailability 543
SuperiorDN 543
SuperiorManager 543
Time Format 547
UseCombinedCRL 78
UseEntrustPolicyObjectOC 541
User Attribute Cert 545
UserLimit 542
userSearchExplicitWildcardPrefix 540
userSearchOneScreen 540
using Combined CRL 78
Watchdog 541
WebKey 542
WebSerialNumber 542
WebUserLimit 542
entmgrMinFree, in entmgr.ini file 535
entmgrWarnFree, in entmgr.ini file 536
entrestore.sh command 131–132
Entrust
initialization of 57
profile
default location 490
Entrust ini override section
in entrustra.ini file 523
Entrust Master Control Command Shell
Combined CRL, using 78, 80
Entrust profile
encryption 491
most recently used 520
number displayed at login 493
Entrust Settings section
in entmgr.ini file 527
in entrust.ini file 485, 517
entrust.ini
configuring 485
customizing 485
entrust.ini file
AcceptOnlyOAEP 485, 495
AlgName 511
AliasName 511
ArlCacheEnabled 486
ASH 518
ASH Information section 518
ASHdn 518
ASHPort 518
ASHServer 519
ASHTimeout 519
ASN1Attributes 501
Attribute Options section 514
AttributeName 514
AuthMethod 501
Authority 486
AuthTimeout 486
AVASorting 501
AVASortOrder 502
BadHostsLogFile 502
BannerFile 487
CertificateCacheEnabled 487
CertificateCacheSize 487
ClientSocketTimeout 487
ClientType 487
commenting 485
configuring for multiple shadow Directories 576
ConnectTimeout 487
CrlCacheEnabled 488
CrossCertARLCacheEnabled 488
CrossCertARLCacheSize 488
CrossCertCACacheEnabled 488, 489
CrossCertCACacheSize 489
CrossCertCRLCacheEnabled 489
CrossCertCRLCacheSize 489

CrossCertDebug 489
CrossCertMaximumDepth 490
CryptokiLibrary95 490
CryptokiLibraryNT 490
CryptokiV2Library 490
customizing 485
data0 510, 511
DebugFile 519
DefaultContext_time_req 490, 517
DefaultCredentials_time_req 490, 517
DefaultProfileLocation 490
Directory Connection Settings section 501, 517
DirectoryOperationTimeLimit 503
DirectoryPassword 491
DirectoryProtocol 503
DirectorySearchSizeLimit 503
DirectoryServerLimit 504
DNSAttempts 504
DOS Extensions section 512
EncryptWith 491
EncryptWithOAEP 492
Entrust Settings 492
Entrust Settings section 485, 517
Entrust/RA version 485
Etfile32Auth 515
Etfile32Name 515
example 515
FilterAttributeOptions 505
FIPS Mode section 514
FipsMode 515
for administrative users 485
for end users 485
GenerateOAEP 492
IdentityLibraryNT 492
InstallDir 493
LastProfile 493
LDAPModifyOperation 517
LDAPProxyClient 493
LDAPProxyServer 493
LDAPRenameOperation 518
LdapV2UseOptions 505
Ldapv3PreventBind 505
Login Strings section 513
MacTypes section 513
MailDll 494
MailEnabled 494
Manager 494
MaximumCrossCertNodes 494
Message 513
OIDAlias section 511
OIDTable section 511
OpaqueAttributes 506
PEMAlgNames section 512
PEMAlgorithmName 512
PKIVersion 519
PostOffice 495
ProcessOAEPOnly 495
ProfileServer 495
ProfileServerDN 495
ProfileServerKeyType 496
ProgDir 496
Recipients 496
ReferralAuthMethods 506
ReferralDepthLimit 507
RetryTime 507
RoamingIDField 496
RoamSearchBase 496
SearchBase 497
SearchBaseOrder 497
SearchTimeout 497
SEPByProxy 498
SEPVersion 498
Server 499
SessionlessConnect 507
SigningKey 500
TraceFile 500
UseBinaryOptionDefaults 507
V21Compatible 500
Version 500
X500 Display Data Structure section 510
X500 Group Data Structure section 509, 510
X500 Search Data Structure section 508, 509
X500 Sorting Data Structure section 511
x500AttributeType 511, 512
X500AttrSyntax section 511
Entrust/Authority
adding content to certificates 273
address and port number 486
Combined CRL 77
changing lifetime 79
publishing to file 80
setting threshold 81
turning on and off 78
disabling subsystems 357, 358
display subsystem status 357
displaying status 352
enabling subsystems 357, 358
issuing CA certificates 325
issuing certificates 269
minimum required disk space 535
settings 527
starting 352
stopping 352
timeout 528
Entrust/Authority data
backing up 106
restoring from backups 127
restoring to Directory 133

Entrust/Authority data files 107
backing up 112–113
restoring 128
Entrust/Authority database
adding user information 332
archiving backups 124
automatic backup schedule 106, 113
backing up 106
backup directories 111
backup directory
changing 118
backups, changing number kept online 117
changing password 139
Combined CRL threshold 81
corruption of 126
Entrust/PKI certificate definitions 274
how to re-encrypt 136
how to validate 126
integrity of 126
maintaining 106
manual backup 122–124
recording changes to 108
re-encryption 136
dangers of 136
failures 136
restoring from backups 127–132
restoring on NT 130
scheduling backups 113
simulating data loss 106
using Combined CRL 77
validating 126
manually 126
verifying backups 124
Entrust/Authority host
settings in entrust.ini file 483, 485, 520, 527
Entrust/Authority Master Control
command line 341–444
adding information to users 415
administering the database 360
administering the Directory 368
audit logs, checking 432
autologin 356
backing up the database 361
changing certificate hashing algorithm 373
changing client-side settings 428
changing directory attributes 418
changing Master User authorizations 377
changing Master User passwords 351
changing product compatibility 413
changing protocol encryption algorithm 374
changing Security Officer authorizations 425
changing user encryption key length 375
changing user object classes 416
combined CRL lifetime 411
combined CRLs 411
command summary 443
configuration variables 364
configure backup settings 120, 362
configure settings for database validation 366
creating Security Officers 423
Directory Administrator DN, changing 371
Directory passwords, changing 370
disable automatic login 356
disabling Entrust/Authority subsystems 358
display backup settings 120, 362
display client-side settings 427
display encoding of Directory attributes 418
display Entrust/Authority subsystem status 357
display product compatibility 413
display status of automatic login 355
display user object classes 415
distributing start-up information to users 378
enabling Entrust/Authority subsystems 357, 358
exit command 348
exporting the CA certificate 383
getting started 343
help 438
init command 345
initializing Entrust/PKI before using 345
listing Security Officers 428
logging in 349
logging out 349
maintaining the CA 381
monitoring Entrust/PKI 432
officer auths query command 426
officer auths set command 426
officer create command 424
officer recover command 425
perform full backup manually 123
perform incremental backup manually 124
recovering Security Officers 423, 424
recovering the CA profile 381
re-encrypt the database 137
re-encrypting the database 360
requesting subordinate CA certificate 253, 419
resetting certificate serial number 382
restore Entrust/Authority data to the Directory 134
restoring CA data to the Directory 368
revoking the CA signing key 394
security policy settings 373
setting all clients up for key recovery 394
show settings for database validation 366
show status of Entrust/Authority service 352
starting 347
starting commands automatically 354
starting Entrust/Authority 352
stopping 347
stopping Entrust/Authority 352

updating the time zone 363
using combined CRLs 410
validate the database manually 126
validating the database 365
writing CRLs and ARLs to the Directory 369
command shell security classes 442
Entrust/PKI performance counters 434
login settings 529
Entrust/Authority service
stopping for Combined CRL 78, 79, 80
Entrust/Entelligence
certificate search log 489
location of executable 496
Entrust/PKI certificate definitions
about 273
changing 275, 276
location of 274
Entrust/Profile Server, certificate type 303
Entrust/RA
security context 517
validity of credentials 517
Entrust/RA preferences
default certificate category 62
default searchbase 62
Directory searches 62
modifying search performance 62
modifying the default view for group membership 62
warnings 62
Entrust/Timestamp
certificate type 303
Entrust/Web Connector
in Identrus environment 596
entrustra.ini
customizing 520
entrustra.ini file 38
0 to 9 520
Admin Main Window Settings 524
AttributeName 521
Bulk Console Defaults section 523
ChangeDNFormat 522
commenting 520
count 521
customizing 520
DefaultCategory 522
DirAdmin Defaults section 523
DirAdmin Window Settings 525
Directory Search Criteria section 520
Entrust ini override section 523
General Options 522
General Options section 522
GetAuthorityInfo 521
InputPath 523
MaxNumberForDatabaseSearch 521
OutputPath 523
profile name 524
ReadOnlyDirectoryBrowser 522
Recent EPFs section 520
Root Searchbase 523
Settings section 524
User List section 521
Users 521
Warnings 523
EntrustSession 485, 490, 492, 493, 495, 498, 500, 519
entsh 347
entUserStateHistTrimFrequency, in entmgr.ini file 538
EntUserStateHistTrimIdleTime, in entmgr.ini file 537
entUserStateHistTrimMax, in entmgr.ini file 537
entUserStateHistTrimNotAfter, in entmgr.ini file 537
entUserStateHistTrimNotBefore, in entmgr.ini file 537
entUserStateHistTrimPeriod, in entmgr.ini file 538
Etfile32Auth, in entrust.ini file 515
Etfile32Name, in entrust.ini file 515
.EtReport files 30
Event severity rating 445
Event Viewer
displaying audit logs in 27
events
7642 445
7643 446
7644 446
7645 446
7646 446
7647 446
7649 446
7651 446
7652 446
7653 446
7656 447
7657 447
7658 447
7659 447
7660 447
7661 447
7662 447
7668 448
7669 448
7670 448
7671 448
7673 448
7675 449
7676 449
7677 449
7682 449
7683 449
7686 450
7688 450
7690 450
7692 450

7694 450
7696 451
7699 451
7700 451
7701 451
7702 451
7703 451
7704 452
7706 452
7707 452
7709 452
7717 454
7719 454
7721 454
7725 454
7727 454
7728 455
7729 455
7730 455
7731 455
7732 455
7733 455
7734 455
7735 455
7740 455
7741 455
7742 456
7743 456
7744 456
7745 456
7746 456
7747 456
7748 456
7749 456
7750 457
7751 457
7752 457
7753 457
7754 457
7755 457
7756 457
7757 457
7758 457
7759 458
7760 458
7762 458
7763 458
7764 458
7765 458
7766 458
7767 458
7768 458
7769 459
7771 459
7773 459
7774 459
7775 459
7776 460
7777 460
7779 460
7784 460
7785 461
7787 461
7788 461
7790 461
7791 461
7792 461
7800 461
7801 462
7804 462
7805 462
7806 462
7807 462
7810 462
7811 462
7813 463
7815 463
7818 463
7820 463
7821 464
7824 464
7826 464
7828 464
7840 465
7841 465
7842 465
7843 465
7848 465
7849 466
7851 466
7854 466
7858 466
7859 466
7860 466
7861 467
7862 467
7863 467
7874 468
7881 469
7882 469
7883 469
7888 469
7889 469
7895 470
7897 470
7898 470
7899 471
7905 471

7906 471
7907 471
7908 472
7909 472
7910 472
7911 472
7912 472
7913 472
7916 473
7919 473
7920 473
7921 473
7922 473
7930 473
7939 474
7941 474
7948 475
7951 475
7952 475
7953 475
7954 475
7955 476
7956 476
7957 476
7958 476
7960 476
7963 476
7964 476
7966 476
7968 477
7972 477
7976 477
7979 478
7981 478
7982 478
7989 481
7990 481
7992 481
7994 481
excluding certificate extensions 328
exit command 348
exporting the CA certificate
general 88
in entsh 383
Express Search Source Order policy setting 183
extension
AIA 594
default values for variables 300
extensionRequest 254, 420
extensions
certificate 271

F

failover
ensuring success 582
fcs export command 430
fcs import command 431
fields, user report 30
files
creating master.certspec 278
initial.certspec 274, 275
master.certspec 274, 276
names.txt 50
opening master.certspec 279
password.rules 50
processing master.certspec 280
words.sor 50
Filesize, in entmgr.ini file 544
FilterAttributeOptions
in entrust.ini file 505
Find dialog box (Directory Browser) 38
finding
Directory Browser entries 37
users in the Dirtectory 37
FIPS 140-1 level 2 613
FIPS Mode section, in entrust.ini file 514
FipsMode, in entrust.ini file 515
firewalls 612
First Officer
restore failed message 453
first-time initialization 86, 277
failure message 475
flexible
password rules 167
flexible certificates
criticality 292
data type 293, 295, 312, 313
keyUsage extension 291
found users
limiting number of 62
FPKI compliance policy setting 177
full backups
backup directories created 112
default schedule 108
example of 109
initiating manually 122
number kept online 117
scheduling 113
verifying 124
when to perform manually 122

G

General Options section, in entrustra.ini file 522
GenerateOAEP, in entrust.ini file 492

GetAuthorityInfo, in entrustra.ini file 521
groups
permission to administer 159

H

hardware
.dll for CRYPTOKI library 527
tokens 172
and number of key pairs 172
and PKCS #12 profile export 172
CryptokiLibrary 490
Hardware user type
in templates definition file 191
hardware-based database protection 100
devices 100
disabling 103
enabling 102
enabling and disabling 102
identifying CA libraries 101
querying 104
redundancy 100
when to enable 100
hash count for PKCS #12 profile export 178
help
on Entrust/Authority Master Control command line 438
hierarchy
CA, in Identrus environment 590
Identrus environment 590
hierarchy of CAs
creating online 243
hierarchy, creating a hierarchy of CAs 244
high availability 553–588
and multiple NICs 578
and service monitoring 567
and shadow Directories 576
and virtual interfaces 578
case studies 583
configuring Entrust/PKI 576
data 569
determining availability requirements 572
failure scenarios 573
monitoring services 577
of Entrust/PKI 572
overview 554
sample Entrust/PKI architecture 580
services 566
HMAC, specifying during online cross-certification 216
HTTP Proxy for CRL Requests policy setting 178
http_proxy_setting 300

I

ICE policy settings 182
identifying CA libraries when using multiple hardware devices 101
Identrus 589
identrus.certspec 593
IETF
RFC 2459, "Internet X.509 Public Key Infrastructure Certificate and CRL Profile" 275
IETF Request for Comments 2459
certificate revocation lists 77
IgnorePKCS10Signature
in entmgr.ini file 541
including
Hardware user type
in templates definition file 191
increasing
number of processes
for Administration subsystem 75, 529
for PKIX-CMP subsystem 75, 531
incremental backups
and transaction logs 108
backup files 113
default schedule 108
example of 109
initiating manually 122
scheduling 113
when to perform manually 122
Informix database. See Entrust/Authority database
Informix, database password 139
.ini files
configuring 483
entmgr.ini 527
entrust.ini 485
entrustra.ini 520, 527
init command 345
initial.certspec
file 277
location of 274
working with 275
initialization 57, 86
changing Entrust/PKI certificate definitions 274
Combined CRL lifetime 79
customizing CA certificates 274
failure message 475
turning on Combined CRL 78, 79
using Combined CRL 78
initialization file
definition 483
InputPath, in entrustra.ini file 523
InstallDir, in entrust.ini file 493
installing Entrust/Authority on a shared drive 576

integrity
automatic check frequency 535
failure message 465
how often checked 534
intelligent disk arrays 570
internal audit error 470
International Telecommunications Union
X.509 recommendation 272
interoperability
OCSP responders 590
ISO/IEC Recommendation
certificate revocation lists 77
ISS software 611
issuing
ARLs 206
ITU-T, X.509 recommendation 275
certificate revocation lists 77

K

Key
in entmgr.ini file 542
key
CA key compromise 394
disabled signing key message 466
encryption key update failed 479
encryption key update message 466, 467
end user key update failed 478
in entmgr.ini file 542
recovery
setting all clients up 394
signing key update message 467
key rec done report field 31
Key Rec Init field 31
Key Recovery state
on a report 30
keys
recovery
backup after 122
specifying what signing algorithm to use 171
user encryption keys 374, 375

L

LastProfile
in entrust.ini file 493
LastWorkOffline
in entrust.ini file 493
LDAP
timeout 487
LDAP v3, attribute options 514
LDAPModifyOperation, in entrust.ini file 517
LDAPProxyServer, in entrust.ini file 493
LDAPProxyTimeout, in entrust.ini file 493
LDAPRenameOperation, in entrust.ini file 518
LDAPV2UseOptions, in entrust.ini file 505
LDAPv3PreventBind, in entrust.ini file 505
legal agreements
for cross-certification 204
Level
in entmgr.ini file 544
license
information changed message 464
permission to view or modify license information 159
user limits, setting 60
lifetime
of cross-certificates 205, 217, 224, 230
limiting
number of found users 62
limits
setting user limit 60
Link certificate type 272
link certificates 390
list
of Security Officers in your domain 46
ListenTimeout, in entmgr.ini file 528
listing
of audit logs 445, 551
load balancing 566
Log severity rating 445
logging in
to Entrust/Authority Master Control command line 349
logging out
of Entrust/Authority Master Control command line 349
Logging section, in entmgr.ini file 543
login
automatic 356
bitmaps
see BannerFile 487
displaying login messages 513
logging in from any workstation 172
message displayed at login 171
one-time login to all Entrust-Ready applications 169
restricting where users log in 172
roaming login 172
timeout 170
Login attempt window policy setting 182
Login section
in entmgr.ini file 529
Login Strings section, in entrust.ini file 513
Login timeout (minutes) policy setting 170
logout
disable automatic 356
display status of automatic login 355
logs
7712 452, 453
7993 481

M

MAC, specifying during online cross-certification 216
MacTypes section, in entrust.ini file 513
MailDll, in entrust.ini file 494
MailEnabled, in entrust.ini file 494
Manager Data Files folder
  in backup folder 112
Manager, in entrust.ini file 494
manager.log.lck 19
Manager_Data_Files. See Entrust/Authority data files
ManagerBackup, in entmgr.ini file 538
manual backups
  initiating 122–124
  when to perform 122
Master and shadow Directories 575
Master User
  authorizations
    setting 57, 377
  changing CA Directory Access password 53
  changing password 48
  disabling multiple authorizations 58
  enabling multiple authorizations 57
  passwords
    rules 48
  when to enter passwords 57
master.certspec
  contents 275
  creating 278
  defining certificate extensions 271, 332
  file 277
  opening 279
  post-initialization 274
  processing 280
  working with 276, 277–281
Maximum bad login attempts policy setting 182
MaximumCrossCertNodes, in entrust.ini file 494
MaxNumberForDatabaseSearch, in entrustra.ini file 521
MaxRestarts, in entmgr.ini file 538
Message in Entrust-Ready clients policy setting 171
Message, in entrust.ini file 513
messages
  0 severity rating 445, 551
  1 severity rating 445, 551
  2 severity rating 445, 551
  backups 124
  severity ratings 445, 551
  See also audit logs, alarms, events
mgraudit.log.lck 19
Mgrlisten, in entmgr.ini file 528
Minimum PKCS #12 hash count policy setting 178
mirroring
  disk 569
modifying cross-certificate types 314
modifying policy certificate types 307
modifying user certificate types 283
monitoring Entrust/PKI processes 578
moving servers or disks 122
moving the CA key pair
  to a PKCS #11 v1 hardware device 97
mutual cross-certification
  description 206
  example 206
  offline 208, 218
  online 208
    choosing 211
  protocols 208, 209

N

name report field 30
named ECDSA curves 387
names.txt file 50
network
  online cross-certification 208
New User dialog box
  assigning certificate types 282
  customizing 189
  database fields 332
  describing certificate types 286
  displaying certificate types 283
  exporting the template definition file 189
  importing the template definition file 189
  name of certificate types 286
New User Policy dialog box
  assigning certificate types 306, 307, 309
noSubjectKey 338
Notary information message 459
NTFS 612
number
  of found users
    limiting 62
  of users
    retrieved in Directory search 64
Number of key pairs policy setting 180
number of processes
  increasing
    for Administration subsystem 75, 529
    for PKIX-CMP subsystem 75, 531
    for Secure Exchange Protocol subsystem 75, 539
NumFiles, in entmgr.ini file 545

O

Object Clas 546
objects in Entrust 321
obsolete certificate types 330

OCSP 589
officer auths query command 426
officer auths set command 426
officer client-setting command 428
officer create command 424
officer list command 429
officer recover command 425
Officer, Security. *See* Security Officer
offline cross-certification. See cross-certification, offline 207
offline root CA
  revocation 339
offlineARLs 399
OIDAlias section, in entrust.ini file 511
OIDs
  in user policies 175
OIDTable section, in entrust.ini file 511
onconfig file
  restoring 130
online
  creation of hierarchy of CAs 243
Online Certificate Status Protocol 589
online cross-certification. See cross-certification, online 207
OpaqueAttributes, in entrust.ini file 506
Operation Complete dialog box 216, 231, 237
Operation Completed Successfully dialog box 211
options
  changing Entrust/RA 62
OrgRoleFirstOfficer, in entmgr.ini file 541
OutputPath, in entrustra.ini file 523
overrideCommonNameFormat 197
overview of policy settings 168

P

partitioned CRLs 77, 83, 546
passwd command 351
password
  length 56
  rules 56
Password expires in (weeks) policy setting 168
Password History policy settings 168
Password length (characters) policy setting 168
Password needs lowercase letter policy setting 169
Password needs non-alphanumeric character policy setting 169
Password needs number policy setting 169
Password needs uppercase letter policy setting 169
password.rules file 50
passwords
  binding the CA to Entrust/Authority 54
  changing
    administrator 52
    Administrator profile 52
    CA Directory Access 53
    Directory Administrator 54
    Entrust/Authority database 139
    Master User 48
  Directory Access password 54
  for Administrators 52
  for Master User 49
  modifying password rules 50
  names.txt file 50, 51
  password expiry in user policy 168
  password history in user policies 168
  password.rules file 50
  requiring an uppercase character 169
  requiring lowercase characters 169
  requiring numbers 169
  rules
    flexible 167
    for Master User 49
  setting minimum length 168
  special characters in 49, 169
  words.sor file 50
    changing 51
PEM 254, 420
  with header 254, 420
PEMAlgNames section, in entrust.ini file 512
PEMAlgorithmName
  in entrust.ini file 512
PEMAlgorithmName, in entrust.ini file 512
pending DN report field 32
Pending Status Information property page 213
Perform Directory consistency check policy setting 174
performance
  performance counters 434
permission
  and roles
    viewing 151
permissions
  administrative
    viewing 152
  to administer groups 142
Permit desktop policy setting 172
Permit roaming policy setting 172
Permit Server Login usage policy setting 173
permit_server_login 300
personal address book 173
physical access 612
PKCS #1 492, 495
PKCS #1 padding 485
PKCS #10 219, 254, 419

PKCS #11 v1
moving the CA key pair 97
PKCS #11 v2
devices that use a password 94
storing the database key 100
PKCS #12 profile export 178
and token usage 172
PKIVersion
in entrust.ini file 519
PKIX compliance policy setting 177
PKIX-Certificate Management Protocol subsystem
timeout 487
PKIX-CMP
cross-certification failed message 448
listen port 528
PKIX-CMP susbsystem
increasing number of processes 75, 531
policy authcode-dist enable command 379
policy auths disable command 378
policy auths enable command 377
policy auths query command 377
policy certificate category 271, 272
Policy certificate expires in (days) policy setting 177
policy certificate types
editing definitions 308
policy certificate. See also User Policies
policy certificates
adding information to 306–313
creating types of 307
customizing 306–313
definition of 306
policy hasher query command 373
policy OIDs
permission to administer 159
policy protocol query command 375
policy proto-enc command 375
Policy section
in entmgr.ini file 541
Policy settings
Symmetric encryption algorithms 170
policy settings
Acceptable policy OIDs 175
Algorithm for digital signature 171
Allow CA personal address book use 173
Allow CAPI Export 179
Allow CAPI Key Export 180
Allow personal address book use 173
Allow PKCS#12 Export 178
Allow unprotected CAPI Export 180
Content Scanner SMTP 183
Delay single login register 181
Disable single login 169
DN encoding format 173
Do not process anyPolicy 177
Do not process policy mappings 176
Enable the use of a Cert cache 183
Enable the use of a CRL cache 182
Enable the use of a XCert cache 183
Enable the use of an ARL cache 182
Enforce identity usage 173
Enforce S/MIME 175
Enforce token usage 172
Express Search Source Order 183
FPKI compliance 177
HTTP Proxy for CRL Requests 178
ICE settings ignored 182
ICE settings signed 182
Login attempt window 182
Login timeout (minutes) 170
Maximum bad login attempts 182
Message in Entrust-Ready clients 171
Minimum PKCS#12 Hash Count 178
Number of key pairs 180
overview 168
Password expires in (weeks) 168
Password History 168
Password length (characters) 168
Password needs lowercase letter 169
Password needs non-alphanumeric character 169
Password needs number 169
Password needs uppercase letter 169
Perform Directory consistency check 174
Permit desktop 172
Permit roaming 172
Permit Server Login usage 173
PKIX compliance 177
Policy Certificate expires in (days) 177
Prevent single login register 181
Require policy 176
Searchbase Search Order 184
Secure Delivery SMTP 183
Symmetric encryption algorithms 170
policy user-enc-keylen command 376
PostOffice
in entrust.ini file 495
power source protection 567
preferences
changing Entrust/RA 62, 68
search 71
Preferences dialog box 68
Prevent single login register policy setting 181
Previous status report field 31
privateKeyUsage extension 402, 404
processes
increasing number of
for Administration subsystem 75, 529
for PKIX-CMP subsystem 75, 531
for Secure Exchange Protocol subsystem 75, 539

ProcessOAEPOnly
in entrust.ini file 495
product compatibility 413
profile
CA profile recovery 381
profile name
in entrustra.ini file 524
Profile Server
administrator DN 495
IP address and port 495
symmetric key type 496
profiles
default location 490
encryption 491
most recently used 520
number displayed at login 493
ProfileServer, in entrust.ini file 495
ProfileServerDN, in entrust.ini file 495
ProfileServerKeyType, in entrust.ini file 496
ProgDir, in entrust.ini file 496
property pages
Cross-Certificate I Requested 213, 232, 237
Cross-Certificate I Signed 240
Pending Status Information 213
protecting the database 100
protocol encryption algorithm 374
ProtocolOverride, in entmgr.ini file 528
protocols for online cross-certification 208
Public-Key Infrastructure X.509-Certificate Management Protocol
description 209
publishing Combined CRL 77
push
CRL 396

Q

querying hardware-based database protection 104

R

RAID
and data availability 569
read this first 15
ReadOnlyDirectoryBrowser
in entrustra.ini file 522
real-time replication 562
reasonCode extension 408
Recent EPFs section, in entrustra.ini file 520
Recipients, in entrust.ini file 496
recommendations
X.509 272, 275
records
audit. *See* audit logs
recover CA command 381
Recover Officers dialog 46
recovering
Security Officers 45
via command line 424
redundancy
using CA hardware 100
re-encrypting
the database 57
re-encryption
of Entrust/Authority database 136–138
dangers of 136
reference # report field 31
reference number
for Security Officer 46
Release 4.0-compatible users
adding 191
ReLoadCAKeysPeriod, in entmgr.ini file 539
remote replication 561
and enhanced database protection 564
removing X.509 certificate extensions 336
replacing
Directory attribute values 41
replication
and disaster recovery 561
report fields, described 30
reports
creating 30
fields, explanation of 30
how to create 30
security of 30
Reports section
in entmgr.ini file 547
Require policy policy setting 176
resetting
number of Security Officers to one 59
responders
OCSP 590
restoring
authauto.ini file 129
Entrust/Authority data files 128
Entrust/Authority data to Directory 133
Entrust/Authority database 130
from archived backups 124
onconfig file 130
restoring from backups
Entrust/Authority data 127–132
Entrust/Authority data files 128
RetryTime, in entrust.ini file 507
revocation list extension
modifying default behavior 402
reasonCode 408

revocation lists 395
accessing the number 397
exporting 396
lifetimes and update periods 395
Revoked state, on a report 30
revoking
certificates
from multiple clients 77
written to a Combined CRL 77
cross-certification 203
risk assessment 557
roaming login 172
Directory search 496
RoamingIDField, in entrust.ini file 496
RoamSearchBase, in entrust.ini file 496
roles
administrative permisssions 142
administrator 142
and permissions
viewing 151
appropriate combinations of permissions 157
assigning a policy certificate to a role 187
changing a role setting 163
checking permission dependencies 165
creating 142, 161
by copying 161
from scratch 162
debugging role definitions 165
defaults for Entrust user roles 144
permission to administer 159
pre-defined 142
sample customized roles 149
viewing 151
writing role settings to text file 156
root CA
in Identrus environment 590
Root Searchbase
in entrustra.ini file 523
rotating DNS 566
rules
modifying for passwords 50
password
Master User 48

S

S/MIME 175
sample Web certificate types 303
SANs 570
saving audit logs in Entrust/RA 25
scheduling automatic backups 113
defaults 108, 113
search preferences 71
SearchAttributeOrder, in entrust.ini file 497
Searchbase Search Order policy setting 184
SearchBase, in entrust.ini file 497
SearchBaseOrder, in entrust.ini file 497
searchbases
and administrative permissions 142
and cross-certification 204
default 66, 497
initial 66
permission to administer 160
required for offline cross-certification 218
required for online cross-certification 208
searches, setting preferences for improved 71
searching
Directory search settings 523
for audit logs in Entrust/RA 20
Search Performance settings 521
setting number of users to find during a search 65, 66, 67
SearchTimeout
in entrust.ini file 497
Secure Delivery SMTP policy setting 183
Secure Exchange Protocol
algorithm changed message 463
description 209
encryption algorithm, changing 73
Secure Exchange Protocol susbsystem
increasing number of processes 75, 539
secured audit logs 18
securing servers 611
security
of passwords 52
of reports 30
policy, setting 373
security checklist 612
Security Officer
authorizations
for sensitive operations 142
listing 428
listing Officers in your domain 46
recovering 45
resetting number to one 59
security policy
and cross-certification 204
changing password rules 51
permission to administer 160
validation failed message 473
Select Cross-Certificate Type dialog box 224
sensitive operations
number of administrator authorizations
configuring 57
Security Officer authorizations 142
SEP. See Secure Exchange Protocol
SEPByProxy, in entrust.ini file 498

SEPDisabled, in entmgr.ini file 539
SEPNumProcs, in entmgr.ini file 539
SEPVersion, in entrust.ini file 498
SerialNumber, in entmgr.ini file 542
Server
  in entmgr.ini file 528
  in entrust.ini file 499
server security 611
serverType
  in entmgr.ini file 529
service autologin disable command 356
service autologin enable command 356
service autologin query command 355
service config command 357, 358
service start command 354
service status command 353
service stop command 354
Session Disconnect, in entmgr.ini file 545
Session Timeout, in entmgr.ini file 545
SessionlessConnect
  in entrust.ini file 507
setting
  user policies 167
setting new user limit 60
settings
  role
    changing 163
Settings section
  in entrustra.ini file 524
setup information
  in reports 30
shadow Directories
  using with Entrust/PKI 574, 576
Show warnings about operations 63
Sign Cross-Certificate dialog box 223
signing cert desc report field 31
signing cert expiry report field 31
signing cert issue report field 31
signing cert revok reason report field 31
signing cert revoked report field 31
signing key
  algorithm used 171
  default algorithm 500
signing private key
  signing cross-certificates 205
SigningKey
  in entrust.ini file 500
single key pair user
  changing to dual key pair 180
single key pair users
  and token usage 172
  changing from dual key pair 180
  policy settings 180
  supported algorithms 171
single login 169
smartcards
  in Identrus environment 590, 591
sorting audit logs 24
special characters
  in passwords 49
standards
  and Combined CRL 77
starting
  commands automatically 354
  Entrust/Authority 352
  Entrust/Authority Master Control command line 347
  Entrust/Authority service 353
  read this first 15
start-up package
  distributing 378
state report field 30
stopping
  Entrust Master Control Command Shell 347
  Entrust/Authority 352
  Entrust/Authority service 354
storage area networks 570
subject key identifier 336, 337, 338
subjectKeyIdentifier 254, 420
subjectKeyIdentifier extension 406
subordinate CA 244
subordinate read-subcert command 255, 421
subsystems
  assigning multiple processes 75
  configuring 357
summary of steps
  working with initial.certspec 275
  working with master.certspec 276
superior CA 244
Superior section
  in entmgr.ini file 543
SuperiorAlgID
  in entmgr.ini file 543
SuperiorCAAvailability
  in entmgr.ini file 543
SuperiorDN
  in entmgr.ini file 543
SuperiorManager
  in entmgr.ini file 543
Symmetric encryption algorithms policy setting 170

T

tasks
  administrative 33
template definition file
  customizing 189

templates
to add users 189
templates definition file
adding a new user type 192
adding attributes 194
backing up 189
deleting 189
editing 189, 191
exporting 190
importing 198
including Hardware user type 191
instructions in the file 191
modifying 189
permission to administer 160
testing 199
troubleshooting 199
validating 191
testing templates definition file 199
third-party
trust
among users 204
establishing through cross-certification 203
threshold, setting for Combined CRL 81
Time Format
in entmgr.ini file 547
time zone, updating 363
timer controls
automatic reissue of attribute certificates 536
timing-out of Entrust/RA 170
to add new user types to the Directory Browser 193
to move the CA key pair to a PKCS #11 hardware device 97
to remove default extensions from an X.509 certificate 337
to update the CA hardware password
using Entrust/Authority Master Control 94
using entsh 95
tokens 172
TraceFile, in entrust.ini file 500
transaction logs 108
restoring from 128
triple DES algorithm 74
trust
across CA domains 203
among users 204
establishing through cross-certification 203
indirect between users 204
one-way versus two-way 205, 208, 218
revoking between CAs 238
through cross-certification 204
types
of certificates 271, 282

U

unilateral cross-certification
CA in Entrust/RA window 217
description 205
offline 218
online 208, 218
choosing 211
protocols 208, 209
uninterruptible power sources 567
Universal Naming Convention
naming Combined CRL 80
updating
CA key pair 389
updating the CA hardware password 94
updating the CA key pair
between hardware devices 98
updating the CA key pair on hardware 96
upgraded by report field 30
upgrading
Entrust/PKI
backup after 122
operating system
backup after 122
UPS 567
UseBinaryOptionDefaults
in entrust.ini file 507
UseCombinedCRL
entmgr.ini 78
UseEntrustPolicyObjectOC, in entmgr.ini file 541
user
initialization failed message 481
user certificate types
assigning 282, 314
editing definitions 285
example 302
user certificates, defined 282
user deactivated report field 31
user encryption key length 375
user limit
changing 60
User List section
in entrustra.ini file 521
user lists
specifying information to save 68
user policies 167, 300, 545
changing 184
creating 185
overview 167
overview of settings 168
setting 167
user template file 189
user templates. See templates definition file

user type
default user type in New User dialog box 192
deleting 194
display only in Directory Browser 193
UserLimit, in entmgr.ini file 542
users
activating the Hardware and Person 4.0 PKI user types 197
adding
backup after 122
changing default settings for 189
adding a new user type 192
adding information to users
via command line 415
adding using templates 189
changing object classes 416
display user object classes 415
finding in the Directory 37
indirect trust 203
limiting
number of found users 62
modifying the new user dialog box 189
number (limit) of 60
number of
retrieved in Directory search 64
permission to administer 160
setting new limit 60
Users, in entrustra.ini file 521
userSearchExplicitWildcardPrefix, in entmgr.ini file 540
userSearchOneScreen, in entmgr.ini file 540
using CA hardware 93
Using the Directory Browser 34–44

V

V21Compatible, in entrust.ini file 500
Valicert 591
validating
Entrust/Authority database 123, 126
validation string
for offline cross-certification 220, 227
variable definitions
default values 300
variables
defining for certificate extensions and attributes 297
for certificate attributes 313
verification key
in cross-certificate 205
provided during cross-certification 208
verifying backups 124
Version
in entrust.ini file 500
viewing
administrative permissions 151, 152
audit log details 22
audit logs 19
in Entrust/RA 18
in the Windows Event Viewer 27
roles 151
roles and permissions 151
virtual IP addresses
for Entrust/Authority 578

W

Warnings
in entrustra.ini file 523
warnings
show/suppress warnings 523
Watchdog, in entmgr.ini file 541
Web certificate category 271, 272
Web certificate types, sample 303
Web certificates
adding database fields 332–335
adding information to 282–305
assigning types 282
creating types of 283
customizing 282–305
function of 282
WebKey
in entmgr.ini file 542
WebSerialNumber
in entmgr.ini file 542
WebUserLimit
in entmgr.ini file 542
when to enable hardware-based database protection 100
Windows 2000
Combined CRL 77
retrieving Combined CRL 79
Windows Event Viewer
description of audit log fields 27
opening 27
sending audit logs to 27
words.sor file 50

X

X.509 certificate extensions
listed 337
removing 337
X.509 certificates
removing certificate extensions 336
X500 Display Data Structure section, in entrust.ini file 510
X500 Group Data Structure section, in entrust.ini file 509, 510

X500 Search Data Structure section, in entrust.ini file 508, 509
X500 Sorting Data Structure section, in entrust.ini file 511
X500AttributeType
    in entrust.ini file 512
X500AttributeType, in entrust.ini file 511
X500AttrSyntax section, in entrust.ini file 511