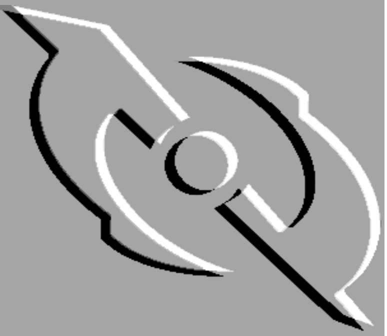


PGP Certificate Server for NT

Administrator’s Guide

Version 2.0

PGP Certificate Server for NT, Version 2.0

8-98. Printed in the United States of America.

Portions of this software may use public key algorithms described in U.S. Patent numbers 4,200,770, 4,218,582, 4,405,829, and 4,424,414, licensed exclusively by Public Key Partners; the IDEA(tm) cryptographic cipher described in U.S. patent number 5,214,703, licensed from Ascom Tech AG; and the Northern Telecom Ltd., CAST Encryption Algorithm, licensed from Northern Telecom, Ltd. IDEA is a trademark of Ascom Tech AG. Network Associates Inc. may have patents and/or pending patent applications covering subject matter in this software or its documentation; the furnishing of this software or documentation does not give you any license to these patents. The compression code in PGP is by Mark Adler and Jean-Loup Gailly, used with permission from the free Info-ZIP implementation. LDAP software provided courtesy University of Michigan at Ann Arbor, Copyright © 1992-1996 Regents of the University of Michigan. All rights reserved. This product includes software developed by the Apache Group for use in the Apache HTTP server project (http://www.apache.org/). Copyright © 1995-1997 The Apache Group. All rights reserved. See text files included with the software or the PGP web site for further information.

Network Associates, Inc.
3965 Freedom Circle
Santa Clara, CA 95054
http://www.nai.com

(408) 988-3832 main

info@nai.com

LIMITED WARRANTY

Limited Warranty. Network Associates warrants that for sixty (60) days from the date of original purchase the media (for example diskettes) on which the Software is contained will be free from defects in materials and workmanship.

Customer Remedies. Network Associates' and its suppliers' entire liability and your exclusive remedy shall be, at Network Associates' option, either (i) return of the purchase price paid for the license, if any, or (ii) replacement of the defective media in which the Software is contained with a copy on nondefective media. You must return the defective media to Network Associates at your expense with a copy of your receipt. This limited warranty is void if the defect has resulted from accident, abuse, or misapplication. Any replacement media will be warranted for the remainder of the original warranty period. Outside the United States, this remedy is not available to the extent Network Associates is subject to restrictions under United States export control laws and regulations.

Warranty Disclaimer. To the maximum extent permitted by applicable law, and except for the limited warranty set forth herein, THE SOFTWARE IS PROVIDED ON AN "AS IS" BASIS WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED. WITHOUT LIMITING THE FOREGOING PROVISIONS, YOU ASSUME RESPONSIBILITY FOR SELECTING THE SOFTWARE TO ACHIEVE YOUR INTENDED RESULTS, AND FOR THE INSTALLATION OF, USE OF, AND RESULTS OBTAINED FROM THE SOFTWARE. WITHOUT LIMITING THE FOREGOING PROVISIONS, NETWORK ASSOCIATES MAKES NO WARRANTY THAT THE SOFTWARE WILL BE ERROR-FREE OR FREE FROM INTERRUPTIONS OR OTHER FAILURES OR THAT THE SOFTWARE WILL MEET YOUR REQUIREMENTS. TO THE MAXIMUM EXTENT PERMITTED BY APPLICABLE LAW, NETWORK ASSOCIATES DISCLAIMS ALL WARRANTIES, EITHER EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO IMPLIED WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE, AND NONINFRINGEMENT WITH RESPECT TO THE SOFTWARE AND THE ACCOMPANYING DOCUMENTATION. SOME STATES AND JURISDICTIONS DO NOT ALLOW LIMITATIONS ON IMPLIED WARRANTIES, SO THE ABOVE LIMITATION MAY NOT APPLY TO YOU. The foregoing provisions shall be enforceable to the maximum extent permitted by applicable law.

Table of Contents

Chapter 2. Configuration 23

Preface

Who Should Read This Guide

This guide describes how to install, configure, operate, and maintain a PGP Certificate Server. The guide is for System Administrators or others who are responsible for setting up and running the Server. The Certificate Server allows PGP users to submit and retrieve keys according to the policies enforced at your site.

What's in This Guide

Chapter 1 ***The PGP Certificate Server***

Describes the PGP Certificate Server's features and explains how the Server works.

Chapter 2 ***Configuration***

Describes how to configure the PGP Certificate Server.

Chapter 3 ***Operation and Maintenance***

Describes how to run and maintain the Server.

Chapter 4 ***Monitoring and Logging***

Describes how to monitor Server usage and how to interpret the log files.

Conventions Used in this Guide

The following sections explain the conventions used in this manual to delineate and emphasize important terms, concepts, and instructions.

Typographical Conventions

New terms, variables, commands, and code samples appear in a different style or font to help distinguish them from the surrounding text.

- New terms are shown in *italics* and are generally defined in context or, if necessary, are elaborated on in greater detail in the Glossary. Variables are also shown in *italics*, for example, http://<*www.company.com*>/certserver/default.htm

- Commands are shown in **bold** to indicate information that appears on the screen.
- Code samples are shown in `Courier font (this is an example of Courier)`.

Special Advisements

The following special advisements are used to call your attention to information that requires consideration.

NOTE: Notes give supplemental information that emphasizes a concept or explains a caveat regarding the current topic of discussion.

TIP: Tips give specific guidelines you should follow or precautions you should take when carrying out a specific task.

WARNING: Alerts are warnings about conditions or procedures that could result in unwanted consequences unless specific measures are observed.

For More Information

There are several ways to find out more about Network Associates and its products.

Customer Service

To order products or obtain product information, contact the Network Associates Customer Care department.

You can contact Customer Care at one of the following numbers Monday through Friday between 6:00 A.M. and 6:00 P.M. Pacific time.

Phone	(408) 988-3832
Fax	(408) 970-9727

Or write to:

Network Associates, Inc.
3965 Freedom Circle
Santa Clara, CA 95054
U.S.A.

Technical Support

Network Associates is famous for its dedication to customer satisfaction. We have continued this tradition by making our site on the World Wide Web a valuable resource for answers to technical support issues. We encourage you to make this your first stop for answers to frequently asked questions, for updates to Network Associates software, and for access to Network Associates news and encryption information.

World Wide Web	http://www.nai.com

Technical Support for your PGP product is also available through these channels:

Phone	(970) 522-2952
Fax	(408) 970-9727
Email	PGPSupport@pgp.com

To provide the answers you need quickly and efficiently, the Network Associates technical support staff needs some information about your computer and your software. Please have this information ready before you call:

- PGP product name
- PGP product version
- Computer platform and CPU type
- Amount of available memory (RAM)
- Operating system and version and type of network
- Content of any status or error message displayed on screen, or appearing in a log file (not all products produce log files)
- Email application and version (if the problem involves using PGP with an email product, for example, the Eudora plug-in)

Your Feedback is Welcome

We continually improve our product documentation and welcome customer feedback. If you would like to provide input, please send email to us at the following address:

tns_documentation@nai.com

Related Reading

Here are some documents that you may find helpful in understanding cryptography:

Non-Technical and beginning technical books

- “*Cryptography for the Internet*,” by Philip R. Zimmermann. Scientific American, October 1998. This article, written by PGP’s creator, is a tutorial on various cryptographic protocols and algorithms, many of which happen to be used by PGP.
- “*Privacy on the Line*,” by Whitfield Diffie and Susan Eva Landau. MIT Press; ISBN: 0262041677. This book is a discussion of the history and policy surrounding cryptography and communications security. It is an excellent read, even for beginners and non-technical people, and contains information that even a lot of experts don't know.
- “*The Codebreakers*,” by David Kahn. Scribner; ISBN: 0684831309. This book is a history of codes and code breakers from the time of the Egyptians to the end of WWII. Kahn first wrote it in the sixties, and published a revised edition in 1996. This book won't teach you anything about how cryptography is accomplished, but it has been the inspiration of the whole modern generation of cryptographers.
- *“Network Security: Private Communication in a Public World,”* by Charlie Kaufman, Radia Perlman, and Mike Spencer. Prentice Hall; ISBN: 0-13-061466-1. This is a good description of network security systems and protocols, including descriptions of what works, what doesn’t work, and why. Published in 1995, it doesn’t have many of the latest technological advances, but is still a good book. It also contains one of the most clear descriptions of how DES works of any book written.

Intermediate books

- “*Applied Cryptography: Protocols, Algorithms, and Source Code in C*,” by Bruce Schneier, John Wiley & Sons; ISBN: 0-471-12845-7. This is a good beginning technical book on how a lot of cryptography works. If you want to become an expert, this is the place to start.
- “*Handbook of Applied Cryptography*,” by Alfred J. Menezes, Paul C. van Oorschot, and Scott Vanstone. CRC Press; ISBN: 0-8493-8523-7. This is the technical book you should read after Schneier’s book. There is a lot of heavy-duty math in this book, but it is nonetheless usable for those who do not understand the math.

- "*Internet Cryptography,*" by Richard E. Smith. Addison-Wesley Pub Co; ISBN: 0201924803. This book describes how many Internet security protocols work. Most importantly, it describes how systems that are designed well nonetheless end up with flaws through careless operation. This book is light on math, and heavy on practical information.
- "*Firewalls and Internet Security: Repelling the Wily Hacker,*" by William R. Cheswick and Steven M. Bellovin. Addison-Wesley Pub Co; ISBN: 0201633574. This book is written by two senior researchers at AT&T Bell Labs and is about their experiences maintaining and redesigning AT&T's Internet connection. Very readable.

Advanced books

- "*A Course in Number Theory and Cryptography,*" by Neal Koblitz. Springer-Verlag; ISBN: 0-387-94293-9. An excellent graduate-level mathematics textbook on number theory and cryptography.
- "*Differential Cryptanalysis of the Data Encryption Standard*," by Eli Biham and Adi Shamir. Springer-Verlag; ISBN: 0-387-97930-1. This book describes the technique of differential cryptanalysis as applied to DES. It is an excellent book for learning about this technique.

The PGP Certificate Server

1

This chapter describes the PGP Certificate Server's features and explains how the Server works. This guide assumes that you are the System Administrator.

General Features

The PGP Certificate Server allows users to submit and retrieve *keys* from a database. A key is a digital code used in conjunction with a cryptographic algorithm to encrypt, sign, decrypt, and verify email messages and files. Information encrypts differently with different keys. The Server uses a set of user-defined policies to control key submission and retrieval.

Server features include the following:

- Automated installation and configuration of the Server through easy-to-use scripts and a Web-based Configuration/Monitoring wizard.
- Single point-of-control user interface to start and stop the Server and monitor other activities.
- Flexible key retrieval that supports searches on multiple key attributes, such as the key type, key ID, creation date, and so on.
- Authentication safeguards that limit access to restricted Server functions (includes access controls and signature verification).
- *PGP Replication Engine* that allows you to replicate database entries to multiple Servers. The databases on these Servers are automatically updated to reflect the contents of the database on the primary Server.

How it Works

The PGP Certificate Server is designed to run on the Windows NT platform. The Server is based on the *Lightweight Directory Access Protocol (LDAP)*, a global directory model. LDAP provides a standard method to manage the submittal and retrieval of keys stored in a centralized database**.** The Server includes a Replication Engine to propagate the contents of the master database to multiple Servers, if required.

The Server enforces the certificate policy established during configuration. The certificate policy identifies the criteria that the Server uses to enforce the acceptance or rejection of keys. The certificate policy also identifies how keys are retrieved by users.

Some older versions of PGP only support access to key Servers over the Web. For these versions, the Server includes a CGI interface that supports the HTTP protocol. This also allows PGP clients to access the Server through firewalls without an LDAP proxy.

What is a Certificate?

A digital *certificate*, called a certificate throughout this document, is information included with a person's public key that helps others verify that a key is genuine or valid. A digital certificate consists of three things:

- A public key.
- Certificate information (identifies information about the user, such as name, user ID, and so on).
- A digital signature.

Installation and Configuration

Installation of the Server is performed using a simple-to-use *installation wizard*. The wizard makes sure all the necessary software components are loaded in the proper sequence and stored in the appropriate directories.

Configuration of the Server is performed using a Web-based *Configuration/Monitoring wizard*. The wizard helps you set up the Server to meet the requirements of your site. This configuration method should address the needs of most sites. However, if you need to change the Server's configuration, you can edit the Server's configuration file.

Operation

You control the Server's major functions (for example, starting and stopping the Server) from the Server's graphical console. The console also allows you to monitor Server activity.

When the Server is running, it responds to user requests to add, search for, and retrieve keys.

The Server uses two sets of criteria to accept or reject keys:

- Configuration parameters in the configuration file
- User access level

Submitting Keys

When a key is submitted to the Server, the Server checks to see if the key meets the criteria established during configuration. The Server can enforce the following checks:

- Key is signed by the appropriate entities. Required signatures are identified during configuration.
- Signatures or User IDs associated with a key are approved for submittal. Authorized signatures are identified during configuration. Note that you can strip or remove all unauthorized signatures or User IDs from the key before storing it on the Server (see Chapter 2 for details).

If the key passes these checks, the Server accepts the key. If the key does not pass the policy requirements, the key is rejected and a copy of the key is placed in a *pending bucket.* You can examine the key and decide if the key should be allowed on the Server.

Retrieving Keys

When a key is placed on the Server, PGP users can retrieve the key to encrypt data and verify digital signatures.

All users can use the standard LDAP search and retrieval functions to access keys. Here are some of the attributes you can use in your search:

- email address
- User name (both first and last names)
- Key IDs
- PGP key type, size, revocation status (that is, if the key's owner has revoked the key because it is old or compromised).
- Creation and expiration dates

All users use the same interface to access keys. As System Administrator, your authority level, established during Server configuration, allows you to add, disable, and delete keys from the Server.

For more information on how to configure these settings, See Chapter 2, Configuration.

Importing and Exporting Keys

As System Administrator, you can import and export keys. Use these features to distribute large numbers of keys. You can import both PGP *keyrings* and *ASCII-armored key files* from any client machine that has proper access to the Server using the LDAP protocol. (A keyring is a set of keys. ASCII-armored key file is binary information encoded using a standard printable, 7-bit ASCII character set.) You can also export keys to any client machine from the machine running the Server.

Replication of the Database to Other Servers

The Replication Engine is a robust replication mechanism used to propagate the contents of a primary or master Server's database to one or more slave Servers. A replication daemon monitors the master Server and updates the slave Servers' databases whenever a change occurs on the master Server.

You identify the slave Servers when you run the Configuration/Monitoring wizard. The wizard stores this information in the master Server's configuration file.

Not all installations will use the master-slave Server configuration. A variety of server configuration models are described in Chapter 3, "Operation and Maintenance."

Monitoring Usage and Activity

The statistics collected by the Server and the Replication Engine allow you to monitor usage and track various activities. During configuration you identify the type of activities that you want to track and the level of detail you want the Server to record.

There are several ways to find out how the Server and Replication Engine are performing:

- Monitor the activity in real-time (click the Monitor tab from the console or use the Configuration/Monitoring wizard's monitoring features). Use the wizard when you do not have direct access to the machine where the Server resides.
- Consult the *Access Log File* (stores a more complete record of activities).
- Check the *Events panel* (displays any errors that occur while the Server or Replication Engine is running).
- Consult the *NT Event Log* (lists all of the errors that have occurred over a longer period of time).

Installing the Server Software

See the PGP Installation Guide for instructions.

Removing the Software

To remove the Server:

1. Run the uninstall program, accessible from the Start menu.

NOTE: The uninstall program does not delete all files from the Server directory (by default, the Server directory is C:\Program Files\Network Associates\pgpcertd). After you run uninstall, remove any unwanted files and directories.

Configuration 2

Configuring the Server

When you install the Server, the configuration settings are set to default values that should work for most sites. However, there are a few settings that you may want to modify before starting the Server. You specify the values for these settings through the Web-based Configuration/Monitoring wizard. These values are stored in the pgpcertd.cfg configuration file. You can edit this file with your favorite text editor, if required.

NOTE: If you installed the Replication Engine, use the configuration file described in this section, pgpcertd.cfg, to configure the Replication Engine. (The setup program's default setting includes the installation of the Replication Engine.)

Setting Up the Configuration/Monitoring Wizard

The Configuration/Monitoring wizard (wizard) is a quick and easy-to-use interface used to configure the Server and Replication Engine. To use the wizard, a Web server must be running on the machine where the Server is installed.

The wizard is a CGI script that can be hosted by any Web server that supports the standard CGI protocols. For example, the wizard works with the following server products:

- Netscape's FastTrack Server 3.01
- O'Reilly Software's WebSite Professional 2.0
- Microsoft's Internet Information Server 2.0 (IIS)
- Apache Server 1.3

The wizard is shipped in two formats:

- Windows executable file
- Perl 5.x script

In most cases you will use the executable file. However, in some cases you may need to use the raw Perl script.

During Server installation, if the installation program locates the Microsoft IIS, the installation program automatically adds support for that Web server, and the executable version of the wizard script is automatically set up for you. However, if you are using another Web server, you must modify the server's configuration file by adding the appropriate aliases or mappings for the Server's Web documents and CGI scripts.

NOTE: If you can place the Web documents and script files in directories that your Web server can access, you can avoid making these configuration changes

To configure the alias for the Web documents (based on the default installation paths), enter the following lines in the Web server's configuration file or wherever your server configures aliases:

```
/certserver/  c:\Program Files\Network Associates\PGPcertd\web\htdocs\
/certserver/docs/  c:\Program Files\Network Associates\PGPcertd\docs\
```

To configure the alias for the CGI scripts, enter the following lines in the Web server's configuration file:

```
/certserver/cgi-bin/  c:\Program Files\Network Associates\PGPcertd\web\cgi-bin\
```

NOTE: For complete information on how to modify the Web server's configuration, consult the Web server's documentation.

In some cases, you may need to use the Perl script to set up your Web server so it can host the wizard. To use the Perl script, you need a Perl interpreter for your Web server, and you should be familiar with both Perl and the implementation of Perl CGI scripts. If you do not have Perl, you can download a copy from the Web and install it on your system from any of the following sites:

http://www.activestate.com/download.htm

http://www.perl.org

http://www.perl.com

Using the Configuration/Monitoring Wizard

You can access the Configuration/Monitoring wizard by entering one of the following URLs in the location field of any Web browser:

http://<hostname>[:<port>]/certserver/default.htm

http://<hostname>[:<port>]/certserver/

http://<hostname>[:<port>]/certserver/cgi-bin/cs.exe

For example, when you type the following entry in your browser's location field, the browser displays a page that contains several useful Certificate Server links, including a link to the Configuration/Monitoring wizard:

http://<*www.company.com*>/certserver/default.htm

When you type the following entry in your browser's location field, the browser brings you directly to the Configuration/Monitoring wizard (that is, if your Web server is configured to run on the non-standard port 8080):

http://<*www.company.com*>:8080/certserver/cgi-bin/cs.exe

When you access the wizard, read the introductory information to learn how the wizard works, and then follow the on-screen prompts to progress through the various configuration settings. The configuration information that you supply is stored in a configuration file. If you prefer not to use the Web-based wizard or want to make some quick adjustments to a few configuration settings, you can use a text editor to edit this file. For information on the name and location of this file, see the following section.

The information provided for each step in the configuration process is fairly explicit. However, if you need a more detailed explanation, the following section describes each of the configuration settings.

Examining and Editing the Configuration File

All of the configuration values are stored in a configuration file, pgpcertd.cfg. The file is normally stored in the following default location:

C:\Program Files\Network Associates\PGPcertd\etc\pgpcertd.cfg

NOTE: If your configuration file is corrupted or deleted, you can use the master configuration file, pgpcertd-Master.cfg, to restore the original settings.

You can edit the configuration file at any time. The changes take effect the next time you start the Server. To restart the Server after you edit the file, press the Re-start button on the Server's console.

The following table includes a brief description of each configuration setting. More complete information for a setting is located on the page noted in the right column.

Table 2-1. Configuration Settings

Setting	Purpose	Page #
AccessLogFile	Identifies the file where access statistics are logged.	page 29
AccessLogDetails	Controls the level of statistics recorded in the Access Log File.	page 28
Allow	Defines level of access for users.	page 29
AllowSigID	Identifies keys that are allowed when TrimSigs is turned on.	page 35
CacheEntries	Identifies the number of the database entries cached by the Server.	page 34
CycleLogDay	Controls when the Access Log File is cycled (archived).	page 31
CycleLogTime	Controls the time of day that cycling of the Access Log File occurs.	page 32
CycleLogKeep	Controls the number of old Access Log Files that the Server retains.	page 32
DBCacheSize	Controls the database cache size in bytes.	page 34
DefaultAccess	Defines default access.	page 32
Directory	Identifies where the database files are located.	page 34
IdleSyncTimeout	Directs the Server to save the database cache to disk after the Server has remained idle for a specified number of seconds.	page 35
LogLevel	Controls the level of information recorded in the System Log File.	page 33
Mode	Identifies the file permissions associated with the database.	page 35
MustSigID	Identifies the signatures a key must have to pass the policy requirement.	page 36
PolicyFailures	Controls if rejected keys are sent to the pending bucket or returns an error.	page 36
Port	Identifies the port to listen to for regular LDAP connections.	page 33

Table 2-1. Configuration Settings

Setting	Purpose	Page #
PublicKeyRing	Identifies the file that contains the Server's TLS key and any keys specified by an Allow Keyid, MustSigID, or AllowSigID configuration value.	page 40
PrivateKeyRing	Identifies the PGP private keyring file that contains the private portion of the Server's TLS key.	page 39
RandSeedFile	Name of file to use to store persistent pseudo random seed.	page 40
ReadOnly	Controls read/write access to database entries.	page 35
Replica	Identifies the location where the database contents are to be replicated.	page 38
ReplicationSecureKeyID	Identifies the key ID of the keypair to use as the key to authenticate the client side of all LDAPS connections.	page 38
RepLogFile	Identifies the log file where changes are recorded for replication.	page 39
SecureMode	Controls if Secure Mode is required, optional, or disabled.	page 40
SecurePort	Identifies the port to listen to for TLS connections.	page 40
ServerSecureKeyID	Identifies the key ID of the keypair to use as the Server's LDAPS key.	page 34
SizeLimit	Identifies the maximum number of matches returned for a query.	page 34
TimeLimit	Identifies the maximum number of seconds allocated for a query.	page 34
TrimPhotoIDs	Instructs the Server to remove PhotoIDs from submitted keys.	page 36
TrimSigs	Instructs the Server to remove unauthorized signatures from submitted keys.	page 37
TrimUsers	Instructs the Server to remove unsigned user IDs from submitted keys.	page 37

Note the following:

- Configuration setting keywords are not case-sensitive.
- Comments can be included by preceding a line with the pound sign (#).

- Blank lines are ignored.
- Long lines can be split by continuing the configuration value on the next line (continue the value with one or more spaces or tabs).
- If a configuration value, such as a filename, contains a space, it must be enclosed in double quotes (“).

General Configuration Settings

This section describes the general configuration settings for the following:

- Users that have access to the Server.
- How Server statistics are logged.
- Other settings that affect how the database responds to queries.

NOTE: When you establish access controls, there are two levels of access of concern. First, you use the “Allow” configuration setting to define the type of access a user or group of users has to various Server functions. Second, you use the “MustSigID” configuration setting to restrict the keys that can be stored on the Server by requiring them to be signed by a given key ID.

AccessLogDetails *<type>*

Items to Log in Access Log

Controls the type of Server operations recorded in the Access Log File. You must list each operation that you want recorded in the Access Log File, and you must separate operations with a space. Table 2-2 describes valid values for AccessLogDetails.

Table 2-2. Valid Values for AccessLogDetails

Value	Description
none	No information is recorded in the access log.
bind	Records bind operations.
unbind	Records unbind operations.
abandon	Records all abandon operations. This setting is on by default.
add	Records all add operations. This setting is on by default.
modify	Records all modify operations. This setting is on by default.
search	Records all search operations. This setting is on by default.

Table 2-2. Valid Values for AccessLogDetails

Value	Description
delete	Records all delete operations. This setting is on by default.
ldap	Records all LDAP operations that are not normally handled by the Certificate Server.
all	Record all of the operations listed above.

AccessLogFile *<filename>* Access Log File

Identifies the relative path or absolute pathname for the Access Log File. By default, there is no Access Log File. If you want one, use this setting to name the file.

Allow <who> <access> Allow access by

Identifies users that have a specific kind of access to the Server.

<*who*> Parameter This entity is

The <*who*> parameter identifies a user or group of users with a specific IP address, hostname, or keyID (32 or 64-bit):

Allow ip <*IP Address*> <*access*>

Allow host <*Hostname*> <*access*>

Allow keyid <*KeyID*> <*limited access level*>

Where

- <*IP Address*> is the dotted decimal IP address (for example, 127.0.0.1).
- <*Hostname*> is the server's TCP/IP hostname (for example, certserver.pgp.com).
- <*KeyID*> is the 32 or 64 bit KeyID of a PGP key. When a keyid is specified for the <*who*> parameter, only the add and delete access settings are valid.

The first parameter (ip, host, and keyid) identifies the method used to identify the user, and the second parameter is the IP address, hostname, or key IDs. For example, to identify a user or group of users by their IP addresses, enter "ip" followed by the appropriate IP addresses.

Use wildcard characters to include a range of users that fit a given criteria.

<*access*> Parameter

Access Permitted

The <*access*> parameter identifies the level of access granted to the users identified in the following manner:

1. By an ip or host <*who*> parameter:

   Allow <*access*>

   Where

   - <*access*> is none, compare, search, read, add, delete, or all.

2. By the keyid <*who*> parameter:

   Allow <*who*> <*KeyID access*>

   Where

   - <*KeyID access*> is Delete or GroupUpdate.

A <*KeyID access*> of Delete means that a request strongly authenticated with the specified KeyID will be deleted or disabled. The special value "self" can be used in place of an actual KeyID if deletes and disables to the key's owner are allowed.

A <*KeyID access*> of GroupUpdate allows an administrator running PGPKeys to send the administrator's list of groups to the Server to replace the list of groups that is already there. The administrator must be the owner of the key specified by the indicated keyid.

Table 2-3 describes the levels of access, which are hierarchically accumulative (that is, each level of access automatically includes all of the permissions granted by the lower levels of access in the hierarchy).

Table 2-3. Descriptions of Access Levels

Access Level	**Description**
none	Denies all access to the specified user.
compare	If the value is known, it can be compared against the value in the database.
search	Allows the designated users to search the contents of the database if searching from an LDAP client (searching from a PGP client requires read access).
read	Allows the specified user to query and retrieve keys from the Server. The following example gives read access to all users: **allow ip * read**

Table 2-3. Descriptions of Access Levels

Access Level	Description
add	Allows the specified user to query and retrieve keys and to add new keys to the Server. The following example gives read access and add access to all users who reside at pgp.com: **allow host *.pgp.com add**
delete	Allows the user to retrieve, add, and delete keys from the Server. Users with “delete” permission can delete keys from the Server if they are using a key with a signature authorized to perform this operation. The following example gives read, add, and signed deletes to the users at the designated address: **allow ip 205.180.136.115 delete** Although users with “delete” permission can perform signed deletions, they are not authorized to perform LDAP deletes. See note below.
all	Allows the specified users to perform all of the above functions. They can also use the standard LDAP functions (add, delete and modify) to manipulate data stored in the database. This setting is not normally used with the Server, but is provided for those sites that intend to build their own LDAP front-end to access the Server’s directory.

NOTE: Delete authority requires two configuration changes. You must allow the host or IP to perform deletes (use an “allow host” or “allow ip” line), and you must indicate what PGP key must sign the delete request (use an “allow keyid” line).

NOTE: The permission granted by the first “allow host” or “allow IP” line that is encountered takes precedence over all subsequent lines. This means that once you grant a certain type of permission to a user, any subsequent permissions that conflict with the initial level of permission are ignored. To avoid any conflicts, place the most specific items first. For example, you should define complete host names (admin.pgp.com) before partial host names (*.pgp.com).

CycleLogDay *<frequency>* Day to Cycle Log

This setting controls when and if the Access Log File is cycled (archived). For more information about Access Log File cycling, see “Access Log File Cycling” on page 77.

- To cycle the Access Log File weekly, enter the day you want cycling to occur: **Monday**, **Mon**, **Tuesday**, **Tues**, **Wednesday**, **Wed**, **Thursday**, **Thurs**, **Friday**, **Fri**, **Saturday**, **Sat**, **Sunday**, or **Sun**.
- To cycle the Access Log File every day of the week, enter **daily**.
- To disable cycling, enter **never** (the Access Log File continues to grow in size).
- Defaults to “never”.

CycleLogKeep *<number>* Logs to Keep

Use this setting to control the number of old Access Log Files that the Server retains. When the number of old logs in the Access Log File directory exceeds the value for <*number*>, the Server deletes the required number of log files until the number of log files matches the value for <*number*>.

The value for <*number*> can be between 0 to 99. If you enter 0, the Access Log File is truncated when CycleLogDay and CycleLogTime occurs, and the data is not archived. Defaults to 10.

CycleLogTime *<time>* Time to Cycle Log

This setting, which controls the time of day that cycling of the Access Log File occurs, uses a 24 hour clock (military time).

<*time*> is in the following format: HH:MM. HH is between 00 and 23, and MM is between 00 and 59. Defaults to 23:59.

DefaultAccess none | compare | search | read | add | delete | all Default Access

Identifies the default level of access granted to all users who are not covered by the access permissions specified with the “Allow” setting. Table 2-4 describes valid values for DefaultAccess:

Table 2-4. Valid Values for DefaultAccess

Value	Description
none	Denies all access to default users.
compare	If the value is known, it can be compared against the value in the database.
search	Allows default users to search the contents of the database.
read	Allows default users to query and retrieve keys from the Server.

Table 2-4. Valid Values for DefaultAccess

Value	Description
add	Allows default users to query and retrieve keys and to add new keys to the Server.
delete	Allows default users to retrieve, add, and delete keys from the Server. Users with “delete” permission can delete keys from the Server if they are using a key with a signature authorized to perform this operation. Although users with “delete” permission can perform signed deletions, they are not authorized to perform LDAP deletes.
all	Allows default users to perform all of the above functions. They can also use the standard LDAP functions (add, delete, and modify) to manipulate data stored in the database.

LogLevel *<level>* Logging Level

Identifies the degree of information recorded in the Event Log File (note that this is not the same as the Access Log File). You can view the contents of the Event Log File to find out how the Server is performing. There are four levels of access, and they are hierarchically accumulative (that is, each level of logging details automatically includes all of the details provided by the lesser levels). Table 2-5 describes valid values for the LogLevel setting.:

Table 2-5. Values for LogLevel

Value	Description
error	Logs all error messages.
warning	Logs all errors and warning messages.
info	Logs all errors, warnings, and informational messages.
verbose	Logs all messages, including LDAP specific information.

Since the logging is output to the Event Log File, each entry generated by the Server has a source of “PGPCERTD.” This distinguishes these messages from those generated by other processes.

Port *<Port>* Port

Where <*Port*> is the port to listen to for regular LDAP connections. Valid values are from 1 to 65534. This defaults to port 389, the well-known port for LDAP. The port numbers for the Port and SecurePort configuration settings must be different, and no other program can use either of those ports.

ServerSecureKeyID *<KeyID>*

Identifies the key ID of the keypair to use as the Server's LDAPS key. This key must be in the keyring files specified by the PublicKeyRing and PrivateKeyRing configuration values. If this is not specified, the first public/private keypair found in the keyring is used.

Where *<KeyID>* is either a 32 or 64 bit PGP KeyID. The KeyID must have the prefix 0x which is followed by a hexidecimal value. For example, 0x9615A02DBBE1E0E2.

SizeLimit *<size>* Size Limit

Identifies the maximum number of matches to return for a given search operation. The default is 500 entries.

TimeLimit *<seconds>* Time Limit

Identifies the maximum number of seconds, in real time, that the Server will spend processing a client search request. If the request is not fulfilled in the allotted time, a message is sent to the client indicating that the request has timed out. The default value is 300 seconds (5 minutes).

Database Configuration Settings

CacheEntries *<number of entries>* Cache Entries

Identifies the number of entries (that is, keys and their associated user IDs) that are cached by the Server. The default cache size is 50 entries.

DBCacheSize *<size>* DB Cache Size

Identifies the size, in bytes, of the in-memory cache associated with the database. Increasing the database cache size uses up additional memory but can dramatically improve performance, especially when modifying database entries. The default size is 1,000,000.

Directory *<path>* Directory

Identifies the relative or fully qualified path to the directory where the database files and associated index entries are stored. There is no default value. If a filename includes blank spaces, the name must be enclosed in quotes.

WARNING: There must be a value for this setting, or the Server will not start.

IdleSyncTimeout *<seconds>* Idle Sync Timeout

Identifies the number of seconds the Server can remain idle before any new entries in the database cache are saved to disk. After the time-out expires, the contents of the current cache are examined to see if any new entries have been added, and then this information is saved to disk. The default is 10 seconds.

Mode *<file permissions>* Mode

Identifies the file permissions associated with newly created database files. This value can be expressed in octal (preceded with a zero), hexadecimal (preceded with 0x), or in decimal format. The default file permissions are 0600 (gives read and write access by the file's owner).

ReadOnly on | off Access Mode

Controls if clients can read and write entries to the database, or if they are restricted to read-only access. This setting is useful when replicating data to multiple Servers, and you want to grant the ability to search for and retrieve entries, but prevent users from adding or modifying entries. When read-only mode is turned on, any attempt by a client to write to the database results in an "unwilling to perform" error message. By default, the read-only setting is turned off, which means clients have read and write access to the Server.

Certificate Policy Configuration Settings

The certificate policy configuration settings define the policy requirements for your site. Use these settings to identify which signatures must be on a key before the Server will accept the key, and which signatures are allowed to remain on a submitted key. For information on gathering key IDs, see "Extracting Key IDs for Configuration Purposes" on page 43.

AllowSigID *<keyID>* Policy Configuration Keys Submitted...

Lists the 32 or 64-bit key IDs for signatures that are considered allowable when the TrimSigs setting is turned on. When trimming signatures, only the owner's signature and those listed by the MustSigID and AllowSigID settings are allowed to remain on the key. All other signatures are trimmed from the key before it is placed on the Server. You can place multiple AllowSigID lines in the configuration file and each are treated with equal significance.

NOTE: Before you start the Server, make sure all of the required signatures are stored on the Server or in the Server's public keyring (see "SecurePort <Port> Secure Port" on page 40).

MustSigID *<keyID>*

Policy Configuration

Identifies the 32 or 64-bit key IDs for required signatures on a client key. To require multiple signatures, list each of the required signatures on a single line. To require at least one of two or more signatures on a key, list each of the optional keys on a separate line. For example:

MustSigID 0x1234567812345678 0x12345678

In this case, the key must be signed by both keys before it is accepted by the Server. Let us look at another example:

MustSigID 0xabcdef0123456789

MustSigID 0xfedcba987654321

In this case, the key must be signed by at least one of the keys in order to pass the policy requirement.

NOTE: Before you start the Server, make sure all of the required signatures are stored on the Server or in the Server's public keyring (see "SecurePort <Port> Secure Port" on page 40).

PolicyFailures pending | error

Policy

Allows you to specify if keys rejected due to policy failure are sent to the pending bucket for further evaluation, or if they are tossed with an accompanying error message. If set to "pending," the key is stored in the pending bucket. If set to "error," the key is ignored and an error message is generated. The "error" setting is useful for sites that do not want to maintain a pending bucket. The default setting is "pending."

TrimPhotoIDs yes | no

Trim Photo IDs

Allows you to remove a PhotoID from a key before the key is stored on the Server. When this setting is turned on (that is, set to yes), PhotoIDs, which can be quite large, are removed from keys. Use this setting to reduce the size of the data stored by the Server. The default setting is "no."

TrimSigs yes | no

Remove Unallowed Signatures

Allows you to remove unauthorized signatures from a key before it is stored on the Server. When this setting is turned on (that is, set to yes), all signatures except the owner's and those listed by the MustSigID and AllowSigID settings are trimmed from the key. The default setting is "no."

NOTE: Do not use this setting unless the MustSigID or AllowSigID setting is used.

TrimUsers yes | no

Remove Unallowed User IDs

Allows you to remove unauthorized user IDs from a key before it is stored on the Server. When this setting is turned on (that is, set to yes), only user IDs that still have a signature (not counting the self-signature) are kept on the key. All other user IDs are trimmed. The default setting is "no."

Certificate Policy Configuration Matrix

The following matrix is designed to help you understand the ramifications of using these settings in combination with one another.

Table 2-6. Certificate Policy Configuration Matrix

MustSigID	AllowSigID	TrimUserID	TrimSigs	Server Results
Not set	Any or no value	No	No	The Server accepts all keys regardless of how they are signed, and performs no trimming.
Set	Any or no value	No	No	The Server accepts any certificate with at least one User ID signed with a key in the MustSigID list. No trimming is performed.
Set	Any or no value	No	Yes	The Server accepts any certificate with at least one user ID signed with a key in the MustSigID list. All User IDs are accepted, but only the owner's signature and those in the AllowSigID and MustSigID lists are retained; all other signatures are removed from the key.
Set	Any or no value	Yes	Yes	The Server accepts any certificate with at least one User ID signed with a key in the MustSigID list. Only User IDs signed by a key listed in the MustSigID or AllowSigID lists are accepted; all other user IDs are trimmed. Only the owner's signature and those in the AllowSigID and MustSigID lists are retained; all other signatures are removed from the key.

A key may be revoked if the key is compromised or old. If a key is revoked, and the key has a signature from a MustSigID, the key still passes policy and is allowed in the database. This is so that revoked signatures can propagate to clients that already have the key with the positive signature on it. You can disable the key if this behavior is not desired.

Replication Engine Configuration Settings

If you plan to support replication (database entries stored on a master Server are mirrored on other slave Servers on the network), you must identify the other Servers. The Replication Engine, PGPrepd, can then replicate the required data and transfer the data to the replication Servers.

NOTE: When you use the Replication Engine, you identify the Servers that will hold the replicated database information, and the name of the replication log file. If you do not specify both of these configuration settings, the replication will not work. Note that the Replication Engine uses the same configuration file as the Certificate Server.

The following are the relevant configuration settings for replication:

Replica [<*protocol*>://] *<hostname>* or *<IPaddress>* [:*<port>*] Hosts to Replicate Database to

Identifies the protocol, hostname or IP address, and an optional port number for the slave machine that must be updated whenever a change in the contents of the master database occurs. Valid protocols are LDAP, LDAPS, and HTTP. If the protocol http is used, the replica Server must be running an MIT style public key server. When replicating to more than one Server, list multiple Servers on the same line, or create a separate entry in the configuration file for each Server. If you do not specify a port number, the default port for that protocol is used. The protocol defaults to ldap.

ReplicationSecureKeyID *<KeyID>*

Specifies the key ID of the keypair to use as the key to authenticate the client side of all LDAPS connections. This key must be in the keyring files specified by the PublicKeyRing and PrivateKeyRing configuration values. If this is not specified, the first public/private keypair found in the keyring is used.

Where <KeyID> is either a 32 or 64 bit PGP KeyID. The KeyID must have the prefix 0x which is followed by a hexidecimal value. For example, 0x9615A02DBBE1E0E2.

RepLogFile *<filename>*

Replication Log File

Gives the fully qualified path for the log file that records all changes to the database on the master Server. The Replication Engine program consults this file to identify the data that must be replicated to the slave Servers.

When you set up a replication scheme for your Server, note that the replication updates the Servers based on any changes that occur after the replication is implemented. If your master Server contains existing entries, you must export the entries to the other Servers to ensure that they are in each Server's database.

For details on how to export data from one Certificate Server to another, see Chapter 3, "Operation and Maintenance."

NOTE: If a filename includes one or more spaces, you must enclose the entire name in quotes.

Secure Mode Configuration Settings

Use the following configuration settings to operate the Server in *Secure Mode*. Use Secure Mode to perform administrative functions such as the deletion of keys. When the Server is in Secure Mode (that is, the value for the SecureMode setting in the configuration file is Required), the Server cannot start unless it can successfully provide secure access by *Transport Layer Security (TLS)*. TLS is a protocol based on SSL that provides encrypted and authenticated communications.

For more information about Secure Mode, see page 67.

PrivateKeyRing *<filename>*

Private Key Ring

Where <*filename*> is the fully qualified pathname to a valid PGP private keyring file. If a relative pathname is used, it is relative to the directory from which the Server was started. Use PGP to create this file. To enable support for TLS, the private portion of the Server's TLS key must be in this keyring.

PublicKeyRing *<filename>*

Public Key Ring

Where <*filename*> is the fully qualified pathname to a valid PGP public keyring file. If a relative pathname is used, it is relative to the directory from which the Server was started.

The Server looks in this keyring file for keys specified by the KeyID configuration setting. Any key used as the Server's TLS key or specified by an 'Allow Keyid', 'MustSigID', or 'AllowSigID' configuration value can be located in this file.

RandSeedFile <*filename*>

Name of file to use to store persistent pseudo random seed. Defaults to Windows NT directory (for example, C:\WinNT\randseed.rnd).

SecureMode <*Mode*>

Secure Mode

Where <*Mode*> is Disabled, Required, or Optional.

- Disabled - Turns off Secure Mode.
- Required - The Server cannot start unless it can successfully provide secure access (Transport Layer Security (TLS)). When this setting is used, there must be a server key in the PublicKeyRing and PrivateKeyRing keyring files, and the operator must enter the passphrase for the secret server key each time the Server starts.
- Optional - If the Server is started in 'auto-start' mode (-s command line argument), TLS is not enabled. If the Server is not in auto-start mode, TLS is enabled and the operator must enter a passphrase for the Server's key before the Server starts.

SecurePort <*Port*>

Secure Port

Where <*Port*> is the port to listen to for TLS connections. Valid values are from 1 to 65534. This setting defaults to port 636, the well-known port for LDAP over TLS (LDAPS).

Verifying Validity of New Configuration Settings

After you change values for the Server's configuration settings, use the Server's Check Configuration feature to verify that your changes are valid.

IMPORTANT: Verification can only be performed while the Server is not running. To check the configuration, you can start another instance of the Control panel.

To verify that your changes are valid:

a. Start the Server's console (for details, see "Starting the Server's Console" on page 45).

b. Select the "Check Configuration" button on the console's Control panel. If there are any configuration errors, the errors are displayed on the Events panel.

HTTP Support for PGP 5.0 Clients

PGP version 5.0 supports adding and searching keys from MIT-style key servers. This style of key server is based on HTTP. The Server uses LDAP as a communication protocol between the client and the server. To allow existing PGP 5.0 clients to access the Server, an HTTP-to-LDAP gateway is included. The HTTP gateway consists of a series of CGI scripts that require access to a Web server.

To set up the HTTP gateway, you must take the following steps:

- Select the machine where the Web server will reside
- Select the port number the HTTP gateway will listen to
- Select an existing or dedicated Web server

MIT-style key servers act as limited Web servers. By default, they listen to port 11371 rather than the standard port of 80. For PGP 5.0 clients to work with the default port, the HTTP gateway must be set up to run on a Web server listening to port 11371. However, a simpler method to make this work is to set up the HTTP gateway on a Web server running on port 80 and set up the PGP 5.0 clients to access that port.

The following sections describe both of these methods.

Scenario 1: Existing Web server, HTTP gateway on port 80

If you are setting up the PGP 5.0 clients to access the key server on port 80, an existing Web server may be able to also handle the HTTP gateway CGI scripts. The only requirements are that the machine must be running Sun Solaris or Windows NT and it must have TCP/IP access to the machine running the Server.

To accomplish this, you must move two CGI scripts to a directory where your Web Server can run scripts, and an alias (or mapping) must be set up to point to the CGI scripts. The two scripts are listed below:

C:\Program Files\Network Associates\PGPcertd\web\cgi-bin\add.exe

C:\Program Files\Network Associates\PGPcertd\web\cgi-bin\lookup.exe

These CGI scripts must be accessible from a browser using the following URLs:

http://<*your.host.com*>/pks/add

http://<*your.host.com*>/pks/lookup

You must add two aliases that do the following mappings:

/pks/add C:\Program Files\Network Associates\PGPcertd\web\cgi-bin\add.exe

/pks/lookup C:\Program Files\Network Associates\PGPcertd\web\cgi-bin\lookup.exe

NOTE: These aliases cause problems for some Web servers (including Microsoft's Internet Information Server). This is because the entire name is aliased and the script name does not have an extension.

NOTE: Please see your Web server's documentation to find out where to place CGI scripts on your Web server and how to create aliases.

The HTTP gateway requires a configuration file that tells the gateway the URL of the Server that it must use to satisfy requests. The name of the file must be pgpmit.cfg and it must be in the Windows NT directory. For example:

C:\WinNT\pgpmit.cfg

The contents of the file should appear in the following format:

url ldap://<*hostname*>:<*port number*>

In the following example, the Server is running on the host *certserver.company.com* on port 389:

url ldap://<*certserver.company.com*>:389

If the port number is not included, it is assumed to be 389. You must also change the key server preference for the PGP 5.0 clients. In the PGP 5.0 client, the key server preferences must be changed to the hostname of the machine running the Web server, and to port 80.

Scenario 2: Existing Web server, HTTP gateway on port 11371

This is similar to scenario 1. The main difference is that since the Web server is running on port 11371 instead of the default port of 80, the Web server is most likely a dedicated gateway. Follow the instructions that appear in scenario 1, but set the port for the server to 11371.

Extracting Key IDs for Configuration Purposes

During the configuration process, you must identify the 64-bit key IDs for the MustSigID, AllowSigID, and Allow keyID configuration settings. The pgpkeyid utility parses all of these IDs automatically. Use the following commands:

```
pgpkeyid [-e] -k <keyring>

pgpkeyid [-e] -a <asciiarmor>
```

Table 2-7. Command Line Switches for the pgpkeyid Command

Switch	Description
-e	Lists the key ID of the encryption portion of the DSS/Diffie-Hellman key. If you do not use this switch, you receive the signing portion (DSS) of the key.
-k	Parses the key IDs from a PGP keyring.
-a	Parses the key IDs from an ASCII-armored key file.

The following is an example of the command line used to list all of the encryption keys in a keyring file:

```
pgpkeyid -e -k keyring.pgp > keyring.new
```

Operation and Maintenance 3

This chapter describes how to run the Server and perform various maintenance duties.

Starting the Server’s Console

NOTE: To run the console you must have NT Administrator access privileges.

After you install the software, make the required configuration changes, and verify the new configuration, use the Server’s console to start and stop the Server and to monitor the Server’s activities. (You can also use the console’s Control panel to change the Server’s port, configuration file, and database location. For details, see “The Control Panel” on page 45.)

To start the PGP Certificate Server Console, choose **Start:Programs:PGP Certificate Server:PGP Certificate Server**. The Server’s console displays the Control panel.

Use the tabs at the top of the console to access the Monitor and Events panels. The Monitor panel shows what the Server is doing. The Events panel shows how the Server is performing.

NOTE: Chapter 4 describes the Monitor and Events panels in detail.

The Control Panel

Use the controls on the Control panel to perform the following tasks:

NOTE: Before you start the Server, you can change the Server’s port, secure port, and configuration file.The first time you run the console, the Server uses the defaults for these settings.

- Identify the Server’s port numbers (Port) - Use the Default port or enter a different port address. When you check “Default,” the software uses the port number found in the configuration file; if the configuration file does not contain a port number, the software uses the default port 389 for the protocol in use.

- Identify the Server's Secure port (Secure Port) - Use the default port or enter a different address. When you check "Default," the software uses the port number found in the configuration file; if the configuration file does not contain a port number, the software uses the default port, 636.
- Identify the Server's configuration file (Configuration File) - Identifies the name and location of the Server's configuration file.
- Create the Server's initial database or create a database in a new directory (Settings - Create Database) - Causes the Server to consult the configuration file's "directory" setting to identify where to create new database files. The only time you must select this option is when you initially create your database or when you create a database in a new directory.
- Tell the Server not to perform signature verification (No Signature Verification) - Use this option when you are adding a large number of keys from one Server to another, and the signatures are known to be valid. This option speeds the process by eliminating signature checking (the Server assumes that all signatures are valid).
- Check the Server's configuration file for validity (Check Configuration) - Checks the validity of the values in the configuration file without starting the Server. If the program detects configuration problems, the console automatically advances to the Events panel and displays details about the problem. You may want to use this mode after you make configuration changes to make sure that all of the configuration values are valid.
- Start the Server
- Stop the Server
- Restart the Server
- Server Help

Starting the Certificate Server from the Console

To start the Server, click the Start button on the Control panel. Note that the Start button becomes a Stop button.

Starting the Server From the Command Line

You can also start the Server from the NT command line. When you start the Server in this manner, you specify the Server's run-time parameters, which are reflected in the controls on the console. The command line method offers a debug mode useful for resolving Server problems. The following is the command with the appropriate command line switches:

```
pgpcertd [-a] [-c] [-n] [-s] [-t <port>] [-f <file>] [-p <port>] [-d <level>]
```

NOTE: The first time you start the Server you must create the database. As a result, you must, at the very minimum, enter the following: **pgpcertd -n.**

The following paragraphs describe each of the command line switches.

Table 3-1. Server Command Line Switches

Command Line Switch	**Description**
-a	Instructs the Server to assume that all signatures have already passed the policy requirements. All other policy checks are enforced. Use this option to copy a large number of verified keys from one Server to another.
-c	Checks the current configuration file for accuracy. Does not start the Server. When you make configuration changes, use this option to verify that the new configuration values are valid.
-n	Causes the Server to consult the “directory” setting in the configuration file to determine where to create new database files. You must use this switch the first time you run the Server and when you change the location of the database file. If you do not, an error occurs.
-s	Automatically starts the Server after the console loads.
-t <*port*>	Identifies the port number that the Server listens to for Secure Mode. Defaults to port 636. If -t is not present, the Server uses the value for SecurePort found in the configuration file. -t -1 (minus one) disables LDAPS and Secure Mode.
-f <*file*>	Identifies the configuration file that the Server uses. Defaults to ..\etc\pgpcertd.cfg.
-p <*port*>	Identifies the port number that the Server listens to for client requests and certificate submittals. Defaults to port 389.

Table 3-1. Server Command Line Switches

<table>
<tr><th>Command Line Switch</th><th>Description</th></tr>
<tr><td>-d *<level>*</td><td>Turns on the debug mode and provides a level of information based on the level you select. The following are the debug levels:
<table>
<tr><td>Debug Level</td><td>Description</td></tr>
<tr><td>1</td><td>Trace</td></tr>
<tr><td>2</td><td>Packets</td></tr>
<tr><td>4</td><td>Arguments</td></tr>
<tr><td>8</td><td>Connections</td></tr>
<tr><td>16</td><td>Data encodings</td></tr>
<tr><td>32</td><td>Search filters</td></tr>
<tr><td>64</td><td>Configuration</td></tr>
<tr><td>128</td><td>Access</td></tr>
<tr><td>256</td><td>Statistics</td></tr>
<tr><td>512</td><td>Statistics (more)</td></tr>
<tr><td>1024</td><td>Shell</td></tr>
<tr><td>2048</td><td>Parsing</td></tr>
<tr><td>8192</td><td>PGP Errors</td></tr>
<tr><td>65535</td><td>All</td></tr>
</table>
NOTE: This switch is primarily used for debugging purposes. Do not use this switch unless you are very familiar with this process or you are consulting with a Technical Support Engineer.</td></tr>
</table>

Verifying that the Server is Running

After you start the Server, the Start button changes to a Stop button. Click the tab for the Monitor panel. The Monitor panel shows the activities that are taking place on the Server. To check on the Server's performance and to see if any errors have occurred, click the tab for the Events panel.

For more information on configuring and examining the NT Event Log file, see Chapter 4, "Monitoring and Logging."

Running the Server as an NT Service

In addition to starting the Server from the Start menu or the command line, you can also run the Server as an NT service, Pgpsrv. This feature allows the Server to continue to run when you log out of your NT account.

Verifying the Service’s Entry in the Windows Registry

The Server installation procedure modifies the Windows registry. As a result, you should verify the Server entries in the registry before you run the Server as an NT service. You can examine the registry by running the registry editor, REGEDT32.EXE.

The Server installation procedure creates a subkey, Parameters, under HKEY_LOCAL_MACHINE\SYSTEM\CurrentControlSet\Services\Pgpsrv. Several values are set under the Parameters subkey. You may need to change the value for AppParameters, type REG_SZ, which specifies the startup parameters for the application. For example:

-p 389 -s -f c:\Program~1\Network~1\PGPcertd\etc\pgpcertd.cfg

The path of the configuration file cannot include spaces in the file and directory names. If the path contains names with spaces or names that are greater than eight characters in length, enclose the full path of the file in quotes. Names greater than eight characters should be replaced by their DOS shorthand (see previous example).

Starting the Service

The installation program automatically installs the Server as a service but does not start it (the service will not start if configured to start automatically at boot time). To start the service, follow the instructions below.

To start the Server as an NT service:

1. Display the Control Panel (Start:Settings:Control Panel).
2. Double click the Services icon.
3. Select the entry for Pgpsrv in the Services box.

NOTE: The service is configured for Manual (Startup Type) and System Account, Allow Service to Interact with Desktop (Log On As). These settings, which can be viewed by double clicking on the Pgpsrv entry, should not be changed. The service cannot be configured for Automatic (Startup Type) because the Server application has a GUI that comes up immediately when the Server is started.

4. Click the **Start** button to start the service.

WARNING: Once the service is started, any user who can log into the NT machine can modify the Server's GUI. However, the service can only be controlled by a user with administrative privileges.

Stopping the Service

Stop the service from the NT machine's Control Panel (Start:Settings:Control Panel, select Services, select Pgpsrv, and click Stop). Note that it is a good idea to stop the PGP Certificate Server before stopping the service. (To stop the Server, click the Stop button on the Server's console). When you stop the Server before you stop the service, the application can cleanup before the service exits.

Uninstall the Server as an NT Service

The Server's uninstall procedure uninstalls the Pgpsrv service automatically. If you need to uninstall the service manually, follow the steps below.

To uninstall the Server as an NT service:

1. Bring up a DOS command prompt.
2. Type delsrv at the prompt.

Running the Replication Engine's Console

NOTE: To run the PGP Replication Engine Console, you must have NT Administrator access privileges.

The Replication Engine's console allows you to start the Replication Engine and specify options that control how the Replication Engine functions.

To run the PGP Replication Engine Console:

Choose **Start:Programs:PGP Certificate Server:PGP Replication Engine**.

The PGP Replication Engine console displays the Control panel. Use the Control panel to start and stop the Replication Engine and to set initialization options. Use the tabs at the top of the console to access the Monitor panel (shows the activities taking place on the Replication Engine) and the Events panel (shows how well the Replication Engine is working).

NOTE: Complete details regarding the Monitor and Events panels appear in Chapter 4.

Preparing to Start the Replication Engine

Before you start the Replication Engine, you can identify the following directory and files:

- The configuration file that you want the Replication Engine to use
- The file where the replication database entries are stored
- The directory where the Replication Engine's temporary files are stored

The first time you run the console, these settings are set to their default values. The following paragraphs describe these settings:

Table 3-2. Directory and Files used by Replication Engine

Directory or File	Description
Configuration File	Identifies the configuration file you want the Replication Engine to use. By default, the Replication Engine uses the Certificate Server's configuration file, pgpcertd.cfg. This file contains all of the configuration settings that affect the Replication Engine.
Replication Log File	Identifies the replication log file you want the Replication Engine to use. This option overrides a value in the configuration file. During normal operation, the Replication Engine continuously monitors the replication log file for more entries. When you select the "Run Once" option, the Replication Engine looks at the replication log file once only.
Temporary File Directory	Identifies the directory you want the Replication Engine to use as a temporary file directory. By default, the Replication Engine uses the values in the TEMP or TMP environmental variables to identify the temporary file directory. If these variables are not defined, the files are stored in the current directory.

NOTE: If you shut down the Replication Engine (pgprepd) and, when you restart the Replication Engine, you do not want to continue to process the database entries where you left off, remove all of the pgprepd related files from the temporary directory.

NOTE: The temporary files used by the Replication Engine can become quite large. Make sure they are stored on a partition that is large enough to hold this data. Use the “Temporary File Directory” option (or -t in the command line) to explicitly designate where these temporary files are stored.

Starting the Replication Engine from the Console

To start the Replication Engine from the Replication Engine’s console, click the Start button on the Control panel. Notice that the Start button turns into a Stop button.

Starting the Replication Engine from the Command Line

You can also start the Replication Engine from the NT command line. When you use this method, the Replication Engine runs with the settings you enter on the command line, and the values displayed on the Replication Engine’s console reflect those values:

pgprepd [-f <*file*>] [-t <*directory*>] [-r <*file*>] [-c] [-o] [-d<*level*>] [-s]

Table 3-3 describes the Replication Engine’s command line switches.

NOTE: When you start the Replication Engine from the command line, you can run the Replication Engine in a debug mode useful for resolving problems. For details, see Table 3-3.

Table 3-3. Replication Engine Command Line Switches

Command Line Switches	Description
-f <*file*>	Identifies the configuration file that you want the Replication Engine to use. By default, the Replication Engine uses the Server’s configuration file, pgpcertd.cfg). This file contains all of the configuration settings that affect the Replication Engine.
-t <*directory*>	Identifies the directory for the Replication Engine’s temporary files.
-r <*file*>	Identifies the log file you want the Replication Engine to use. This option overrides the value in the configuration file.

Table 3-3. Replication Engine Command Line Switches

Command Line Switches	Description
-c	Checks the current configuration file for accuracy. Does not start the Replication Engine. When you make configuration changes, use this option to verify that the new configuration values are valid.
-o	During normal operation, the Replication Engine continuously monitors the replication log file for new entries. When you use the -o option, the Replication Engine looks at the replication log file once only. When you use this option, you must also use the -r option.
-d *<level>*	Turns on debug mode and gives you information based on the level you choose. The levels are the same as those for the Certificate Server, pgpcertd.
-s	Automatically starts the Engine when the console starts.

Verifying that the Replication Engine is Running

When you start the Replication Engine, the Start button on the Control panel changes to a Stop button. At this point, you can advance to the Monitor panel to see the activities the Replication Engine is processing. You can also advance to the Events panel to see how well the Replication Engine is working.

To find out how to configure and examine the NT Event Log file, see Chapter 4, “Monitoring and Logging.”

Using the Replication Engine

If your installation is large or your users are in a number of different locations, a single Certificate Server may not meet your users demand for keys. As a result, you may require a Certificate Server on a number of systems. When you install a Certificate Server on multiple systems, the Replication Engine synchronizes the databases. The Replication Engine runs on the same system as the Certificate Server, and sends new and modified keys to other Certificate Servers. You can run the Replication Engine on any of the systems where a Certificate Server resides.

Certificate Servers are either Master Servers or Slave Servers.

- Master Servers perform all Server functions. A Certificate Server, replication log (a log where new and modified keys and destination Servers are recorded), and a Replication Engine reside on each master Server.
- Slave Servers perform all Server functions except adds, deletions, and disables. A Certificate Server resides on each slave Server.

You can add slave or master Servers to the same physical location or remote locations. Multiple master Servers (Servers that can perform all Server functions), are considered peers.

To learn more about the replication process, see See "How the Replication Process Works" on page 60.

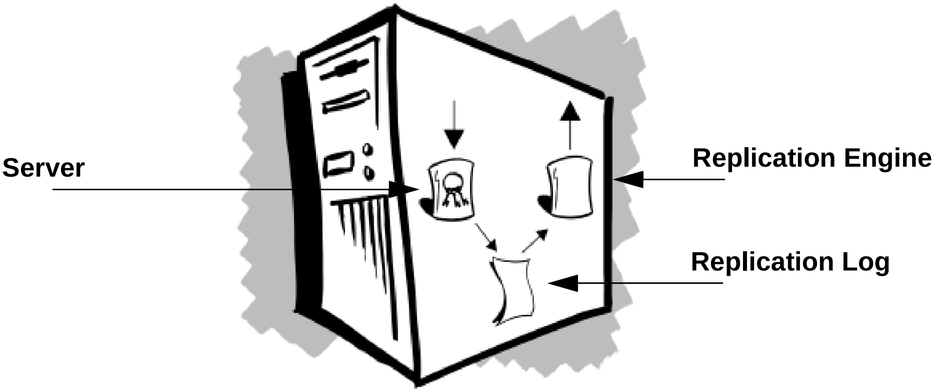


Figure 3-1. Certificate Server Components: Server, Replication Log, and Replication Engine

---

❑ **NOTE:** Requests for deletions and disables must be strongly authenticated using LDAPS or a Signed Request over LDAP. In either case, the Server that is receiving the replication must be set up to allow deletions from the key that strongly authenticated the deletion or disable request. In other words, if deletions or disables are to succeed, the master and slave Servers must have the same `Allow keyid 0x????` delete lines in their configuration files.

Server Configurations

Before you install additional Servers, consider the locations that the Servers will service and the load in each location. The following section describes a few typical scenarios.

Server Configuration Models

The following Server configuration models represent only a few of the potential Server configurations. Each of the models identify the Server relationships (master, slaves, and peers), and the efficiency and tolerance rating for the models.

efficiency - A model with high efficiency has a minimum number of replications. As a result, it uses less network bandwidth, CPU load, and Server load. The fewer the replications, the higher the efficiency. Efficiency is high, medium, or low.

tolerance - A model with high tolerance has a large number of redundant replications. The more redundancy a model has built into it, the higher its tolerance. An example of a configuration with low tolerance is a single Server; if that Server goes down or has a problem, there is no backup. Tolerance is high, medium, or low.

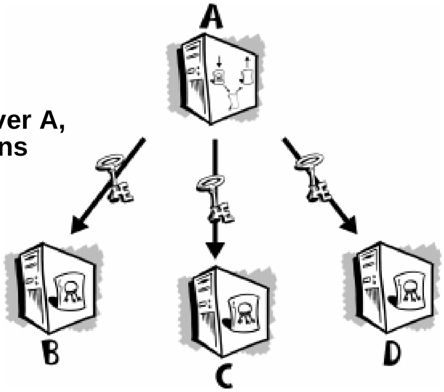
NOTE: When a Server replicates a modified key, the whole key is replicated, not just the changes.

Master Slave Model (high efficiency, medium tolerance)

Use for load balancing or distribution in a single organization. Can also use to ensure 100% availability. This model consists of one master Server (Server A) with multiple slave or backup Servers (Server B, Server C, Server D, and so on). All Servers are available for user requests to update local keyrings, but only the master, Server A, can perform adds, deletions, and disables.

When this model is used, all users can access Server A, but they may not be able to access each of the slave Servers (accessible Servers appear in the PGPkeys Search Window). For example, each of the slave Servers may service a specific division, and the users in that division may access Server A and their division’s slave Server only. In another example, some users may know about two Servers, and be told to use one as their primary Server, and the other as their backup Server.

Tolerance is medium because there are backup systems for queries but only one system for adds, deletions, and disables. Efficiency is medium because this configuration requires the minimum number of replications.



One Master Server, Server A, distributes all replications to all other Servers

Figure 3-2. Master-Slave Model

Star Model (medium efficiency, medium tolerance)

Use for load balancing and 100% availability. This model consists of one master Server, Server A, that distributes all replications to all other Servers (Server B, Server C, Server D, and so on). All of the Servers can receive adds, deletions, and disables. However, there is no direct communication between Server B, Server C, and Server D. These Servers send adds, deletions, and disables to Server A, and Server A replicates these changes to the other Servers.

Efficiency is medium because there is two-way replication. (In this case if Server B receives an add, Server B sends the add to Server A, Server A sends the add back to Server B, and also sends the add to Server C and Server D).

This model is easier to use than the Master-Slave model, because you can use any of the Servers for any purpose.

Let’s look at an example. Server A is in New York, Server B is in Australia, Server C is in San Francisco, and Server D is in Germany.

If you use the Master-Slave model, all adds, deletions, and disables must occur in New York. You can perform other Server functions locally, but you must perform your adds, deletions, and disables remotely.

If you use the Star model, you can perform all Server functions locally; the delay in Server synchronization is minimal (that is, limited to the time it takes to update changes via the Replication Engine).

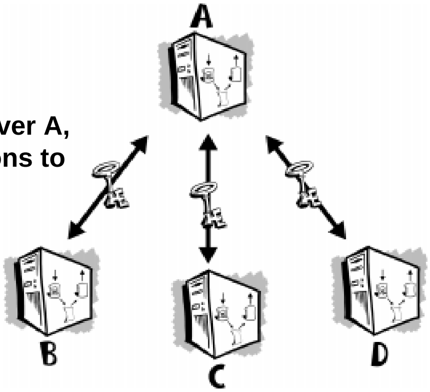

One Master Server, Server A, distributes all replications to all other Servers
A
B
C
D

Figure 3-3. Star Model

Ring Model (high efficiency, low tolerance)

This model is similar to the Star model. However, this model has a high efficiency rate. Users can perform adds, deletions, disables, and queries on all Servers.

The advantage to using this model rather than the Star model is that it limits the number of replications. If you do an add on Server B, Server B sends the add to Server C, Server C sends the add to Server D, Server D sends the add to Server A, and Server A sends the add to Server B. Server B recognizes that the key matches an existing key in its database, and does not replicate the add.

Each Server's configuration file has a replica line that identifies the Servers that receive replications from that Server. In this example, the replica line on each Server would contain the name of just one Server, the next Server in the ring.

Note that you can create hybrids of this model. For example, in addition to the existing replications, you could modify the configuration to include replications between Server A and Server C. You might also modify this configuration to include bidirectional replications between servers (for example, Server B replicates to Server C, and Server C replicates to Server B). This modification ensures that all servers are updated even if one link or server goes down.

Tolerance in this model is low because if any one of the Servers go down, replication cannot complete the loop to all Servers. Whereas in the Star model, if Server C goes down, Server A, Server B, and Server D continue to operate normally. Only Server C would be out of synchronization.

Each Server replicates to the next Server in the ring

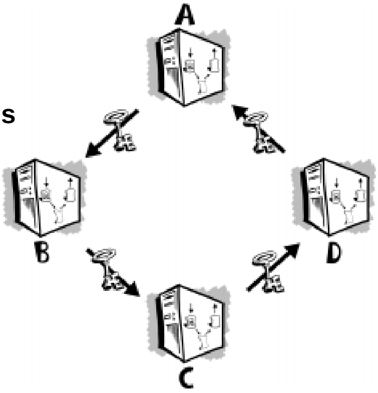


Figure 3-4. Ring Model

Fully Connected Model (low efficiency, high tolerance)

This model is similar to the Star and Ring models. You can perform all Server functions, including adds, deletions, and disables, on all of the Servers. The advantage in this model is that if any of the Servers go down, the other Servers continue to operate normally.

In the Star model, if System A goes down, System B, System C, and System D are on their own, and do not receive any replications during System A's down time. The Star model cannot give you 100% availability unless Server A is up all the time.

In the Fully Connected Model, if Server A goes down, Servers B, C, and D continue to work normally. The problem inherent in this model is that anything more than a small number of Servers becomes very inefficient. Each time you add a single key to a Server, you get multiple redundant replications (for example, if there are 4 servers, each add generates at least 12 replications). The more servers you add, the more replications you generate, and the less efficient this model becomes.

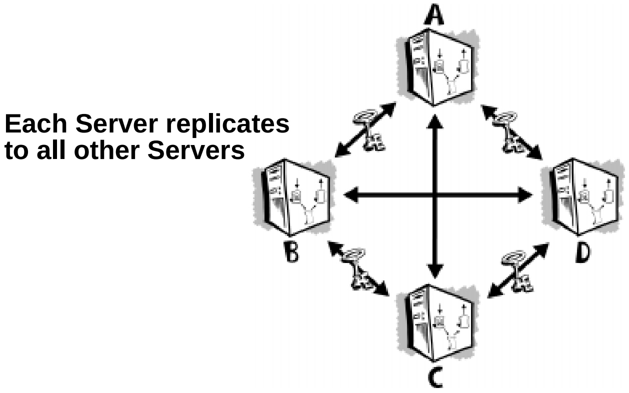


Figure 3-5. Fully Connected Model

Examples of Different Server Configurations

International company with offices in the U.S. and Europe - A Server in the U.S. handles all Server requests that originate in the U.S., and a Server in Europe handles all Server requests that originate in Europe. The Replication Engine maintains the two databases. When a key is added to either of the Servers, the Replication Engine copies the key to the other Server. The two Servers are peers.

Domestic company with one large location, four divisions - One Server is installed in each division and handles all Server requests that originate in that division. The Replication Engine on each Server replicates new or modified keys to the other three Servers. The four Servers are peers. This configuration is primarily used for load balancing.

Domestic company with one large location - Users throughout the office submit keys to Server A, and Server A replicates all new or modified keys to Server B, Server C, and Server D. This configuration allows the company to distribute the load of Server requests and maintain a low bandwidth replication model.

Small domestic company, one location - Server A, the master Server, replicates to Server B, the slave Server. If Server A fails, users can automatically go to Server B for their requests. This configuration might be used to ensure 100% Server availability (fault tolerance).

How the Replication Process Works

With the exception of Servers B, C, and D in the Master-Slave model, all servers run a Replication Engine and replicate adds, deletions, and disables to the servers that appear in the replica line of their configuration file. Note that each Replication Engine must be started on each server, even in a master-slave relationship. If the server's Replication Engine is not started, it can receive adds from other servers, but it cannot forward adds to other servers.

Let's take a closer look at how replication occurs between three Servers. Server A is the master server, and Server B and Server C are slave servers. The slave Servers, Servers B and C, can perform all Server functions except adds, deletions, and disables. The master Server, Server A, can perform all Server functions, including adds, deletions, and disables.

- When Server A receives a new or modified key, Server A checks the replica line in its configuration file to identify the Servers that it replicates or sends new or modified keys to. The names of Server B and Server C appear in Server A's configuration file.
- Server A places copies of the new or modified key in a queue (that is, its replication log or replog) for Servers B and C.
- Server A's Replication Engine sees that there are keys in the queue, and looks for the machines that it must send the keys to. Server A finds the key designated for Server B, and sends it to Server B. Server A finds the key designated for Server C, and sends it to Server C. Since Servers B and C are slave Servers, they do not replicate the keys to other Servers.

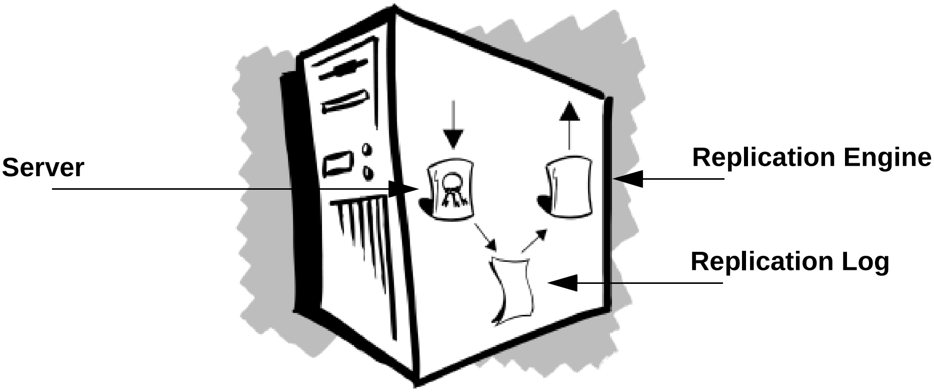


Figure 3-6. The Certificate Server's Components

When does replication occur?

Replication occurs almost instantaneously, as long as the Replication Engine and target Server are running.

- If the Replication Engine is not running, the target Server is updated as soon as the Replication Engine is restarted.
- If the target Server is not running, the Replication Engine continuously looks for the machine. When the target Server becomes available, it sends the keys that are queued for that Server.

Note that if the target Server and Replication Engine are both down, the replication information is not lost.

How and where is the replication log (replog) maintained?

If there is a master Server, the master Server maintains the replication log (replog). The master Server receives new or modified keys, writes them to its own database, places the keys in the replog, and sends the keys to the slave Servers. The slave Servers do not maintain a replog, but they do have a database.

If all Servers are peers, each Server maintains a replog and database; the database is updated by the replication process.

Can two Servers replicate to each other?

What happens if Server A replicates to Server B, and Server B replicates to Server A? Does the Replication Engine get caught in an infinite loop?

When a new or modified key is added to Server A, Server A adds the key to the database and replog. The Replication Engine sees the key in the replog, knows that it replicates keys to Server B, and sends the key to Server B. Server B adds the key to its database and replog.

Server B’s Replication Engine sees the new key in the replog, knows that it replicates keys to Server A, and sends the key to Server A. When Server A receives the key, it sees that it is not a new key, and it does not write the key to the database or replog.

Another scenario might be that Server B has a different version of the key in its database. Server A sends a new key to Server B, and Server B has the key, but it is different. Server B merges the keys, and sends the key back to Server A. Server A sees that the key is different, and merges the keys in its database, and sends the key back to Server B. Server B sees that it is not a new key, and it does not write the key to the database or replog.

If a Server is offline, how and when is its database updated?

When a Server is offline, the Replication Engine on other Servers continue to place new or modified keys for the target Server in the replog. When the Server is back online, the Replication Engines on the other Servers automatically update the Server’s database.

If the master Server is offline, databases are not updated until the master Server is back online.

Adding A New Server

To add a new Server to your configuration:

1. Install the Certificate Server software on the system that will run the new Server (see PGP’s Global Installation Guide for details).
2. Edit the configuration files.
    - If the new Server is to replicate changes to one or more Server’s, add the hostnames and optional port number of the Servers to the replica line of the new Server’s configuration file.
    - If one or more Servers is to replicate changes to the new Server, add the new Server to the replica line of the configuration file for those Servers.
3. Export the existing database to the new Server.

NOTE: Before you use the pgpexport command, stop Server A or configure it for Read Only mode.

a. Login to the machine where the existing Server resides (Server A).

b. Run pgpexport; supply the pathname of the directory that contains the current database (for example, C:\Program Files\Network Associates\PGPcertd\data). pgpexport creates an export file on Server A.

c. Run pgpimport on Server A; specify ldap://<Server B>

For information on pgpimport, see page 68.

Stopping the Replication Engine from the Console

To stop the Replication Engine, click the Stop button on the Control panel.

Setting Up Secure Mode

The Server includes a Secure Mode that you can use to perform deletions and other administrative tasks. When the Server is in Secure Mode, the Server cannot start unless it can successfully provide secure access by Transport Layer Security (TLS). TLS is a protocol based on SSL that provides encrypted and authenticated communications.

The Server is shipped with public and private portions of a passphrase-less evaluation key for a default user ID. Use this key only for evaluation purposes. You must create your own key for the Server.

The following sections describe how to start the Server in Secure Mode using the Server evaluation key, and how to create your own key for the Server.

To start the Server in Secure Mode using the Server evaluation key:

1. Select the PGP Certificate Server from your system's Start menu (Start:Programs:PGP Certificate Server:PGP Certificate Server). The Server displays its console and Control panel.
2. Deselect the Disabled check box in the **Secure Port** field on the Server's Control panel.
3. Press the Start button on the Server's Control panel. The Server displays a dialog box (PGP Enter Passphrase for Selected Key), with **Signing key** and **Passphrase of signing key** fields. The **Signing key** field displays the Server evaluation key (PGP Certificate Server Untrusted Evaluation Key), which does not require a passphrase.
4. Press the **OK** button. The titlebar for the Server's console displays the following text:

   ```
   PGPcertd running -389:636
   ```

   389 is the regular Server port, and 636 is the secure Server port. You are now in Secure Mode.
5. Stop the Server (click the **Stop** button on the console) and exit the Server's console (click the **X**, top right corner of the console).

To create your own key, you must perform the following tasks:

- Display the Server's keyring in PGPkeys
- Create a new key for the Server
- Delete the evaluation key for the Server

☐ **NOTE:** To create a new key, must have PGP for Email and Files version 5.5 or later.

Display the Server's keyring in PGPkeys:

1. Select PGPkeys from your system's Start menu (Start:Programs:PGP:PGPkeys).
2. Select **Preferences** from the PGPkeys **Edit** menu.
3. Select the **Files** tab. This panel displays your public and private keyring files.
    - If you are on a system dedicated to the PGP Certificate Server, go to step 4 on page 64 to set the Server's keyring as the default for PGPkeys.
    - If you are on your own workstation, PGPkeys displays your personal keyring. In this case, follow the remaining steps in this section to display the Server's keyring in PGPkeys.

    Record the values in these fields; you will need this information after you create the new Server keys. The normal path for your public and private keyring files, pubring.pkr and secring.skr, is as follows:

    C:\Program Files\Network Associates\PGP60\PGP60Admin

☐ **NOTE:** The PGP60Admin directory only applies if you are the Administrator.

4. To display the Server's keyring file in PGPkeys, do the following:
    - Press the **Browse** button in the **Public Key Ring File** field. PGPkeys displays the **Select Public Keyring File** file location box. Locate the Server's public keyring file, PGPcertd-pubring.pkr. The default directory for this file is as follows:

      C:\Program Files\Network Associates\PGPcertd\etc

      Click the **OK** button.

- Press the **Browse** button in the **Private Key Ring File** field. PGPkeys displays the Select Private Keyring File file location box. Locate the Server's private keyring file, PGPcertd-secring.skr. The default directory for this file is as follows:

  C:\Program Files\Network Associates\PGPcertd\etc

  Click the **OK** button.

5. Click the **OK** button on the PGP Preferences screen. PGPkeys displays the Server's keyring. The only key on this keyring is the PGP Certificate Server Untrusted Evaluation Key. The following section tells you how to create a new key pair for the Server to replace the evaluation key.

Create a new key for the Server:

1. Select **New Key** from the PGPkeys Key menu. The Server displays the Key Generation wizard. Read the first screen and click the **Next** button.

   Each of the following steps represents a new wizard screen.

2. Enter the full name and email address for the Server's new key pair. In the **Full name** field, enter <*company name*> Certificate Server, for example, Acme Certificate Server (the name can include spaces). In the **Email address** field, enter the email address for the Server administrator, for example, admin@acme.com. Click the **Next** button.

3. Select the type of key you want to generate. **Diffie-Hellman/DSS** is recommended and preselected. To generate a Diffie-Hellman/DSS key pair, click the **Next** button. To generate an RSA key pair, select **RSA** and click the **Next** button.

4. Select the size of the key pair you want to generate. **2048 bits** is recommended and preselected. To generate a 2048 bit key pair, click the **Next** button. To generate a different size key pair, select the size and click the **Next** button.

5. Select when you want the key pair to expire. **Key pair never expires** is preselected. It is recommended that you select **Key pair expires on** and select a date one or two years into the future. Click the **Next** button.

6. Enter a passphrase for the key pair's private key. The passphrase should be at least 8 characters long and should contain non-alphabetic characters. The Hide Typing box is checked. Enter a passphrase in the **Passphrase** field; enter the passphrase a second time in the **Confirmation** field. Click the **Next** button.

The wizard generates your new key pair. If you have selected a large key pair and you are on a slow machine, this process may take several minutes.

7. The wizard discusses sending your key to the Server. Leave the checkbox on this screen unchecked (**Send my key to the root server now**), and click the **Next** button.

8. Click the **Finish** button. The wizard adds the new key to the Server's keyring in PGPkeys. You have successfully created a new key for the Server.

Delete the evaluation key from the Server's keyring:

1. Delete the evaluation key (**PGP Certificate Server Untrusted Evaluation Key**) for the Server's keyring. To do so, select the key in the PGPkeys window, and select the **Delete** option from the PGPkeys Edit menu.

If you are on a system dedicated to PGP, you have completed the process.

If you are on your personal workstation, follow the steps below to display your own personal keyring in PGPkeys:

1. Select **Preferences** from the PGPkeys **Edit** menu.

2. Select the **Files** tab. This panel displays the Server's public and private keyring files.

3. To display your keyring file in PGPkeys, do the following:

   - Press the **Browse** button in the **Public Key Ring File** field. PGPkeys displays the **Select Public Keyring File** file location box. Locate your public keyring file, pgpcertd-pubring.pkr (you recorded this information earlier). The default directory for this file is as follows:

     C:\Program Files\Network Associates\PGP60\PGP60Admin

     Click the **OK** button.

   - Press the **Browse** button in the **Private Key Ring File** field. PGPkeys displays the **Select Private Keyring File** file location box. Locate your private keyring file, pgpcertd-secring.skr (you recorded this information earlier). The default directory for this file is as follows:

     C:\Program Files\Network Associates\PGP60\PGP60Admin

     Click the **OK** button.

4. Click the **OK** button on the PGP Preferences screen. PGPkeys displays your keyring.

Using Secure Mode

Use Secure Mode to perform administrative functions such as the deletion of keys. You can use Secure Mode for all interactions with the Server. For example, you can use Secure Mode to perform searches privately.

To see an example of Secure Mode, use the evaluation key shipped with the Server. For real Server installations, create a new key pair for the Server (see page 63 for details). Thereafter, use the new key pair you create.

To use Secure Mode, follow the steps below:

1. Start the Server from the Start menu. The Server displays the Control panel.
2. Deselect the **Disabled** check box in the **Secure Port** field.
3. Click the Server's **Start** button. The Server displays the PGP Enter Passphrase for Selected Key window.
4. Enter the passphrase of the Server's signing key and click **OK**.

The Server is now in Secure Mode.

Administering the Server

When the Server is running, the Server responds to client requests and requires very little attention. The Administrative version of the PGP system software allows you to submit and retrieve certificates from the Server. In addition to adding and retrieving certificates, you can also disable or remove certificates from the Server.

To prevent unauthorized users from performing these operations, access is restricted to users with the proper authority. Access is regulated in the following ways:

- Password authentication
- Secure access via TLS and LDAPS
- Signature authentication (user submits a request, and the software checks the user's signature to make sure the user has permission to perform the operation)

Use the "Allow" configuration setting to control user access to the add, delete, and disable features.

When a key does not pass authentication, the key is rejected and a copy of the key is sent to the pending bucket. As System Administrator, you must periodically review the keys in the pending bucket. If the keys are valid, you can sign them and re-submit them to the PGP Certificate Server. If the keys are invalid, you can delete them.

Resolving Keys in the Pending Bucket

As System Administrator, it is your responsibility to identify the keys rejected due to policy infractions. The pending bucket is an area on the server that holds keys rejected by the key Server. To access the pending bucket, select the "Pending" box when performing a key search**.**

For complete details on how to search for and manage keys, consult the PGP Security Officer's Guide.

Importing and Exporting Keys

This section describes how to import and export keys.

Importing Keys to the Certificate Server

When you set up your Server, you may want to import keys from an existing keyring file. You can import keys from any machine that has add privileges and access to the Certificate Server. Select Start:Programs:PGP Certificate Server:PGP Import. Use the following import command to send all keys in a file to a Server:

```
pgpimport [-d] <file> ldap://<hostname>[:<port>]
```

Where <*file*> is the name of the file that you want to send to a Server, <*hostname*> is the name of the target Server, and <*port*> is the optional LDAP port number for the target Server. The LDAP port number is required only if the machine does not use the default LDAP port of 389. Use the -d option to delete the imported file after the import runs to completion.

Exporting Keys from the Certificate Server

Use the Server's export feature to export keys from one Server to another or to a backup device. To export the contents of the Server, log on to the machine where the database is located (to perform the export you must have read access to the database). Select Start:Programs:PGP Certificate Server:PGP Export. The following is the export command:

```
pgpexport [-v|-i |-l | -1] <directory> > <file>
```

The following paragraphs describe each of these switches:

Table 3-1. Switches for Export Command

Switch	Description
-v	Enables verbose mode. When verbose mode is enabled, the Replication Engine exports the following values to the standard error device, stderr: the key ID of each key you export, and a running total of the number of keys exported. By default, the verbose mode is turned off.
-i	Instructs the export process to ignore any errors that are encountered during the exportation of the certificate from the database. This option is useful when you know that there are errors, but you still need to perform the export.
<*directory*>	Identifies the directory where the Server database files are located. If you do not explicitly specify a directory, the current directory location is assumed.
<*file*>	By default, the exported output is sent to the standard output device. To save the keys to a file, you must redirect the output to a named file.
-l	pgpexport checks if the key database is already in use. If so, it aborts. To override this and have pgpexport run even if the database is in use, use the -l option.
-1	Indicates that the exported keyblock should be PGP Certificate Server version 1.x compatible. New features are removed from the keys before exporting the keys.

Monitoring and Logging

4

This chapter describes how to monitor Server and Replication Engine activities and check the log files.

Monitoring Operations

While the Server or the Replication Engine is running, you can consult the Monitor and Events panels to find out how well the programs are performing. The following paragraphs describe these panels on the Server and the Replication Engine.

Monitoring the Server

The Monitor panel gives information about the keys submitted and retrieved from the Server. Use the “Auto Update” option on the Monitor panel to update this information automatically.

The Events panel gives useful information about the Server’s performance. For a more detailed look at the errors that occur, consult the NT Event Log. The NT Event Log lists Server errors as well as other errors that may have occurred on the machine.

For a more in-depth chronological view of all the requests the Server has processed, examine the Access Log File. The level of information is controlled by the “AccessLogDetails” setting in the configuration file.

Events Panel Not Displaying Information

If the Events panel is not displaying information, the NT event log may be full. To correct this problem, use the steps that follow.

To display information on the Events Panel:

1. Run the Event Viewer (Start:Programs:Administrative Tools:Event Viewer).
2. Select **Log Settings...** from the Log menu.
3. In the **Change Settings for ... Log** field, select Application.
4. In the **Event Log Wrapping to...** field, select the Overwrite Events as Needed option.

Monitoring the Replication Engine

The Monitor panel gives statistics that tell you how the replication is progressing on each of the slave machines where you are replicating information.

The Events panel gives you useful information about the Replication Engine's performance. For a more detailed look at the errors that occur, consult the NT Event Log. The NT Event Log lists Server errors as well as other errors that may have occurred on the machine.

The following sections describe how to use the consoles and wizard to monitor Server and Replication Engine activity, and identify the underlying sources of information that the software uses to report this information.

Monitoring Certificate Server Activity

To monitor Server requests and activities, click the Monitor tab on the Server console.

Statistics

Up Time

Length of time the Server has been running.

Ops Completed

Number of client operations the Server has processed since the Server started the current run. This number is based on the following operations:

- Client connections opened
- Client connections closed
- Searches
- Adds

Current Connections

Number of active connections on the Server. Each connection consists of multiple operations. The number is typically low since most client connections are generally short-lived.

Total Connections

Total number of connections made to the Server since it was started. Under most circumstances, each connection is equivalent to one client session (for example, a client performing a search).

Bytes Transmitted

Number of bytes transmitted from the Server to the client since the Server was started.

Entries Served

Total number of certificates returned to clients through searches since the Server was started.

Client List

The lower section of the panel displays information on all clients currently connected to the Server. This information includes the hostname or IP address for each client, and the time and date the connection was established.

Monitoring Certificate Server Events and Errors

Use the Events panel on the console to see how the Server is performing.

The Events panel displays all Server events and errors. Entries are ordered by their date and time. When the Events panel is selected, it displays all of the messages generated since the Server was last started. Click the Refresh button to update the listing.

Monitoring Replication Engine Activity

To view the statistics concerning the Replication Engine, click the Monitor tab on the Replication Engine's console.

The Monitor panel displays the functions that the Replication Engine is processing. Information includes the address of the host machine and the port it is connected to. To update this information automatically, use the "Auto Update" option on the Monitor panel.

Status

Displays the current status for the Replication Engine: Up (Server available), Down (Server unavailable), or Untried (Server not tried).

Queue Size

Displays the number of replication operations that are currently lined up for processing.

Last Update Time

Displays the time that the last successful replication occurred for the selected host.

Total Replications

Displays the total number of successful replications that have occurred since the Replication Engine was last started.

Monitoring the Replication Engine’s Events and Errors

To see how well the Replication Engine is performing, click the Events tab from the console.

The Events panel displays all the Replication Engine’s events and errors that have occurred since the Server was last started. They are ordered by date and time. Click the Refresh button to update the listing.

Monitoring Remotely with the Wizard

Most event monitoring and analysis is performed from the console. To perform these functions remotely from another system, use the Web-based Configuration/Monitoring wizard.

To use the wizard:

1. Load the Configuration/Monitoring wizard with your preferred Web browser. The URL varies based on the location and port number of your Web Server, but should look like this:

   http://<hostname>:8080/certserver/cgi-bin/cs.exe

2. To check the status, select the Status tab or icon shown by the wizard.
3. To see the information in the Access Log File, click the Logs tab.

The following sections describe the Access Log File and NT Event Log.

Examining the Access Log File

The Access Log File contains entries for each request that the Server has processed. The level of information in the log file is controlled by the “AccessLogDetails” configuration setting. Log entries appear in the following format:

operation session time result IP host type-specific

Table 4-1. Values in Log Entries

Value	Description

operation	A three letter code describing the type of request submitted to the Server. The following codes may appear in the Access Log File:
	BND = Bind operation
	UBD = Unbind operation
	ABN = Abandon
	SRC = Search operation
	ADD = Add certificate operation
	MOD = Modify entry operation
	DLT = Delete operation

session	A connection identifier that ties together multiple operations done over the same connection.
time	The time the request was generated. The time, given in Universal Time Coordinates (GMT), is expressed using the following format: YYYY-MM-DD HH:MM:SS
result	An LDAP result code in decimal format. For more information on the meaning of these codes, see the LDAP documentation.
IP	The IP address, in dotted decimal format, of the client making the request.
host	The host address of the client making the request. If the host address cannot be resolved, a dash appears in this field.

Table 4-1. Values in Log Entries

Value	Description
type-specific	The type-specific information for the specified operation. The following are the returned values (values vary depending on the type of operation): BND The LDAP bind name enclosed in double quotes. A dash is used if the name cannot be resolved (normal case). UBD None ABN None ADD The ID of the certificate that was added to the Server. On signed requests, the ID of the certificate of the signer is also shown. A dash is used if there is no signer. MOD The ID of the modified certificate. On signed requests, the ID of the certificate of the signer is also shown. A dash is used if there is no signer. SRC The number, in decimal, for the matches or hits returned by a search operation, followed by a dash (-). Note that the trailing dash may not always be present. DLT The ID of all deleted certificates, shown in double quotes, and, if the request requires a signature, the ID for the certificate of the signer. A dash indicates that the value is not applicable

NOTE: If double quotes or back-slashes exist in a value that is always enclosed in double quotes, they are escaped with back-slashes.

Sample Access Log File Entries

The following are typical entries that appear in the Access Log File:

```
SRC 0 1998-05-26 23:46:21 0 171.169.27.75 xyz.nai.com 1 -

SRC 0 1998-05-26 23:46:21 0 171.169.27.75 xyz.nai.com 3 -

SRC 7 1998-05-26 01:16:03 0 171.169.27.75 xyz.nai.com 1 -

ADD 7 1998-05-26 01:16:03 0 171.169.27.75 xyz.nai.com 0x27535B7C40C97D40 -

SRC 0 1998-05-27 01:31:24 0 171.169.17.15 abc.nai.com 1 -

SRC 0 1998-05-27 01:31:24 0 171.169.17.15 abc.nai.com 13 -
```

Access Log File Cycling

Periodically, the Server copies the contents of the Access Log File to a new file, removes the contents of the Access Log File, and begins to record new access information in the empty Access Log File. This action, called cycling, prevents the Access Log File from becoming too large, and allows administrators to process the logs without interfering with Server operations.

The Access Log File settings in the configuration file control how frequently the Server cycles the entries in the Access Log File, and how many cycled files are retained on the Server. For more information, see See “General Configuration Settings” on page 28.

Naming Convention for Cycled Files

The cycled files are named by inserting the date, in the format YYYYMMDD, between the filename and extension of the Access Log File:

<filename>.<YYYYMMDD>.<extension>

For example, if the name of the Access Log File is cert.log, a cycled file created on April 30, 1998, would be named cert.19980430.log. If the Access Log File name does not have an extension, the date becomes the extension. For example, cert.19980430.

Cycled files are kept in the same directory as the Access Log File. If the Server attempts to cycle data and a file with today’s date already exists, the Server assumes that the log files have already cycled, and no additional cycling occurs.

Retention Period for Cycled Files

The maximum number of cycled files retained by the Server is controlled by the CycleLogKeep configuration setting. Each time the Server cycles, the Server counts the cycled files in the AccessLogFile directory and compares the total number of cycled files to the value for CycleLogKeep. When the number of cycled files exceeds the value for CycleLogKeep, the Server deletes enough old cycled files to satisfy the limit set by CycleLogKeep.

Note that the Server counts only those files that use the naming scheme for the current AccessLogFile, and that the date in the file name identifies the age of the file.

For details on the CycleLogKeep configuration setting, see page 32.

Examining the NT Event Log

In addition to the operational information provided in the Access Log File, other information is sent to the NT Event Log. The level of information that appears in the NT Event Log is controlled by the “LogLevel” configuration setting.

All entries in the NT Event Log that are associated with the Server have a source of “pgpcertd.” This distinguishes these messages from those generated by other processes.

LDAP Error Messages

This section lists the LDAP error messages found in the Access Log File. Each entry includes a brief description of the condition that generated the error:

Table 4-2. LDAP Error Messages in the Access Log File

LDAP Error Message	Description
0 (0x00) LDAP_SUCCESS	The request was successful.
1 (0x01) LDAP_OPERATIONS_ERROR	An unexpected Server error was encountered. See the Server Error Log for more information.
2 (0x02) LDAP_PROTOCOL_ERROR	The client accessing the Server is not following the proper protocol.
3 (0x03) LDAP_TIMELIMIT_EXCEEDED	The time limit for a single operation was exceeded. Only partial results were returned. If this occurs on a query operation, try again with a more restrictive query.
4 (0x04) LDAP_SIZELIMIT_EXCEEDED	The query operation matched more than the allowed number of entries. Only partial results were returned. Try the operation again with more restrictive query criteria.
5 (0x05) LDAP_COMPARE_FALSE	The comparison operation returned false.
6 (0x06) LDAP_COMPARE_TRUE	The comparison operation returned true.
7 (0x07) LDAP_STRONG_AUTH_NOT_SUPPORTED	Certain types of strong authentication are not supported.
8 (0x08) LDAP_STRONG_AUTH_REQUIRED	The operation performed requires a signed PGP request.

Table 4-2. LDAP Error Messages in the Access Log File

LDAP Error Message	Description
9 (0x09) LDAP_PARTIAL_RESULTS	This Server could not satisfy the entire request. You may need to contact a referral Server.
16 (0x10) LDAP_NO_SUCH_ATTRIBUTE	This requested attribute is not available.
17 (0x11) LDAP_UNDEFINED_TYPE	This error cannot be returned by the Server.
18 (0x12) LDAP_INAPPROPRIATE_MATCHING	This error cannot be returned by the Server.
19 (0x13) LDAP_CONSTRAINT_VIOLATION	Due to Server policies, all User IDs and signatures were trimmed from the certificate. Nothing was left of the certificate to add.
20 (0x14) LDAP_TYPE_OR_VALUE_EXISTS	An attempt to add the same type or value was made.
21 (0x15) LDAP_INVALID_SYNTAX	An invalid keyblock was received by the Server. See the Server's Error Log for more details.
32 (0x20) LDAP_NO_SUCH_OBJECT	Attempted to reference an object that does not exist.
33 (0x21) LDAP_ALIAS_PROBLEM	This error cannot be returned by the Server.
34 (0x22) LDAP_INVALID_DN_SYNTAX	The distinguished name does not have a valid syntax.
35 (0x23) LDAP_IS_LEAF	This error cannot be returned by the Server.
36 (0x24) LDAP_ALIAS_DEREF_PROBLEM	This error cannot be returned by the Server.
48 (0x30) LDAP_INAPPROPRIATE_AUTH	The authorization used for this request was invalid and unnecessary.
49 (0x31) LDAP_INVALID_CREDENTIALS	The certificate that you attempted to add does not contain the signatures required to pass policy. The certificate may have been placed in the pending bucket.

Table 4-2. LDAP Error Messages in the Access Log File

LDAP Error Message	Description
50 (0x32) LDAP_INSUFFICIENT_ACCESS	You do not have sufficient authority to perform the requested operation. The PGP signer of the request may not be authorized or the host may not have authority to the Server.
51 (0x33) LDAP_BUSY	This error cannot be returned by the Server.
52 (0x34) LDAP_UNAVAILABLE	The entry is not available or the PGP certificate has been disabled.
53 (0x35) LDAP_UNWILLING_TO_PERFORM	This operation is not supported.
54 (0x36) LDAP_LOOP_DETECT	This error cannot be returned by the Server.
64 (0x40) LDAP_NAMING_VIOLATION	The submitted certificate did not match the certificate in the database. The certificate ID may have collided with the ID of another key.
65 (0x41) LDAP_OBJECT_CLASS_VIOLATION	This error cannot be returned by the Server.
66 (0x42) LDAP_NOT_ALLOWED_ON_NONLEAF	An attempt to delete a non-leaf entry was made.
67 (0x43) LDAP_NOT_ALLOWED_ON_RDN	This error cannot be returned by the Server.
68 (0x44) LDAP_ALREADY_EXISTS	The submitted certificate exists in the database and it has not changed. Or, a regular LDAP add was made and the entry already exists.
69 (0x45) LDAP_NO_OBJECT_CLASS_MODS	This error cannot be returned by the Server.
70 (0x46) LDAP_RESULTS_TOO_LARGE	This error cannot be returned by the Server.
80 (0x50) LDAP_OTHER	This error cannot be returned by the Server.
81 (0x51) LDAP_SERVER_DOWN	The client detected that the Server was not accessible. The host or port number may not be correct, the network may be having problems, or the Server may be down.

Table 4-2. LDAP Error Messages in the Access Log File

LDAP Error Message	Description
82 (0x52) LDAP_LOCAL_ERROR	The client LDAP software had a problem.
83 (0x53) LDAP_ENCODING_ERROR	The data received by the Server was incorrect or corrupted.
84 (0x54) LDAP_DECODING_ERROR	The data sent by the Server was incorrect or corrupted.
85 (0x55) LDAP_TIMEOUT	This error cannot be returned by the Server.
86 (0x56) LDAP_AUTH_UNKNOWN	This error cannot be returned by the Server.
87 (0x57) LDAP_FILTER_ERROR	This error cannot be returned by the Server.
88 (0x58) LDAP_USER_CANCELLED	The operation was aborted by the user.
89 (0x59) LDAP_PARAM_ERROR	The certificate received by the Server is invalid. The keyblock may be missing.
90 (0x5a) LDAP_NO_MEMORY	A memory allocation problem occurred.

Glossary

ASCII-armored text	Binary information that has been encoded using a standard, printable, 7-bit ASCII character set, for convenience in transporting the information through communication systems. In the PGP program, ASCII armored text files are given the default filename extension, and they are encoded and decoded in the ASCII radix-64 format.
authentication	The determination of the origin of encrypted information through the verification of someone's digital signature or someone's public key by checking its unique fingerprint.
certificate	A unique digital code used to encrypt, sign, decrypt, and verify email messages and files. In traditional PGP parlance, certificates are generally referred to as keys.
certify	To sign another person's public key.
certifying authority	One or more trusted individuals who are assigned the responsibility of certifying the origin of keys and adding them to a common database.
conventional encryption	Encryption that relies on a common passphrase instead of public key cryptography. The file is encrypted using a session key, which encrypts using a passphrase that you will be asked to choose
decryption	A method of unscrambling encrypted information so that it becomes legible again. The recipient's private key is used for decryption.
digital signature	See signature.
encryption	A method of scrambling information to render it unreadable to anyone except the intended recipient, who must decrypt it to read it.
fingerprint	A uniquely identifying string of numbers and characters used to authenticate public keys. This is the primary means for checking the authenticity of a key.
introducer	A person or organization who is allowed to vouch for the authenticity of someone's public key. You designate an introducer by signing their public key.

key	A digital code used to encrypt and sign and decrypt and verify emailemail messages and files. Keys come in key pairs and are stored on keyrings.
key escrow	A practice where a user of a public key encryption system surrenders their private key to a third party thus permitting them to monitor encrypted communications.
key fingerprint	A uniquely identifying string of numbers and characters used to authenticate public keys. For example, you can telephone the owner of a public key and have him or her read the fingerprint associated with their key so you can compare it with the fingerprint on your copy of their public key to see if they match. If the fingerprint does not match, then you know you have a bogus key.
key ID	A legible code that uniquely identifies a key pair. Two key pairs may have the same user ID, but they will have different Key IDs.
key pair	A public key and its complimentary private key. In public-key cryptosystems, like the PGP program, each user has at least one key pair.
keyring	A set of keys. Each user has two types of keyrings: a private keyring and a public keyring.
LDAP	An acronym for the Lightweight Directory Access Protocol which specifies how directory services are provided through a standard query interface.
message digest	A compact “distillate” of your message or file checksum. It represents your message, such that if the message were altered in any way, a different message digest would be computed from it
meta-introducer	A trusted introducer of trusted introducers.
passphrase	A series of keystrokes that allow exclusive access to your private key which you use to sign and decrypt email messages and file attachments.
plaintext	Normal, legible, un-encrypted, unsigned text.
private key	The secret portion of a key pair-used to sign and decrypt information. A user's private key should be kept secret, known only to the user.
private keyring	A set of one or more private keys, all of which belong to the owner of the private keyring.

public key	One of two keys in a key pair-used to encrypt information and verify signatures. A user’s public key can be widely disseminated to colleagues or strangers. Knowing a person’s public key does not help anyone discover the corresponding private key.
public keyring	A set of public keys. Your public keyring includes your own public key(s).
public-key cryptography	Cryptography in which a public and private key pair is used, and no security is needed in the channel itself.
sign	To apply a signature.
signature	A digital code created with a private key. Signatures allow authentication of information by the process of signature verification. When you sign a message or file, the PGP program uses your private key to create a digital code that is unique to both the contents of the message and your private key. Anyone can use your public key to verify your signature.
SLAPD	An LDAP implementation developed at the University of Michigan which defines the actual functions used to access information (certificates) from a centralized server.
SLURPD	An LDAP extension that allows the contents of a database to be replicated from master to slave servers.
text	Standard, printable, 7-bit ASCII text.
trusted	A public key is said to be trusted by you if it has been certified by you or by someone you have designated as an introducer.
trusted introducer	Someone who you trust to provide you with keys that are valid. When a trusted introducer signs another person’s key, you trust that their keys are valid, and you do not need to verify their keys before using them.
user ID	A text phrase that identifies a key pair. For example, one common format for a user ID is the owner's name and email address. The user ID helps users (both the owner and colleagues) identify the owner of the key pair.
verification	The act of comparing a signature created with a private key to its public key. Verification proves that the information was actually sent by the signer, and that the message has not been subsequently altered by anyone else.

web of trust	A distributed trust model used by PGP to validate the ownership of a public key where the level of trust is cumulative, based on the individuals’ knowledge of the introducers.

Index