PGP WDE for Linux

User's Guide

Version Information

PGP Whole Disk Encryption for Linux User's Guide. Version 10.1.1. Released January 2011.

Copyright Information

Trademark Information

Licensing and Patent Information

The IDEA cryptographic cipher described in U.S. patent number 5,214,703 is licensed from Ascom Tech AG. The CAST-128 encryption algorithm, implemented from RFC 2144, is available worldwide on a royalty-free basis for commercial and non-commercial uses. PGP Corporation has secured a license to the patent rights contained in the patent application Serial Number 10/655,563 by The Regents of the University of California, entitled Block Cipher Mode of Operation for Constructing a Wide-blocksize block Cipher from a Conventional Block Cipher. Some third-party software included in PGP Universal Server is licensed under the GNU General Public License (GPL). PGP Universal Server as a whole is not licensed under the GPL. If you would like a copy of the source code for the GPL software included in PGP Universal Server, contact *PGP Support* (*https://support.pgp.com*). PGP Corporation 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.

Acknowledgments

This product includes or may include:

-- The Zip and ZLib compression code, created by Mark Adler and Jean-Loup Gailly, is used with permission from the free Info-ZIP implementation, developed by zlib (http://www.zlib.net). -- Libxml2, the XML C parser and toolkit developed for the Gnome project and distributed and copyrighted under the MIT License found at http://www.opensource.org/licenses/mit-license.html. Copyright © 2007 by the Open Source Initiative. -- bzip2 1.0, a freely available high-quality data compressor, is copyrighted by Julian Seward, © 1996-2005. -- Application server (http://jakarta.apache.org/), web server (http://www.apache.org/), Jakarta Commons (http://jakarta.apache.org/commons/license.html) and log4j, a Java-based library used to parse HTML, developed by the Apache Software Foundation. The license is at www.apache.org/licenses/LICENSE-2.0.txt. -- Castor, an open-source, data-binding framework for moving data from XML to Java programming language objects and from Java to databases, is released by the ExoLab Group under an Apache 2.0-style license, available at http://www.castor.org/license.html. -- Xalan, an open-source software library from the Apache Software Foundation that implements the XSLT XML transformation language and the XPath XML query language, is released under the Apache Software License, version 1.1, available at http://xml.apache.org/xalan-j/#license1.1. -- Apache Axis is an implementation of the SOAP ("Simple Object Access Protocol") used for communications between various PGP products is provided under the Apache license found at http://www.apache.org/licenses/LICENSE-2.0.txt. -- mx4j, an open-source implementation of the Java Management Extensions (JMX), is released under an Apache-style license, available at http://mx4j.sourceforge.net/docs/ch01s06.html. -- jpeglib version 6a is based in part on the work of the Independent JPEG Group. (http://www.ijg.org/) -- libxslt the XSLT C library developed for the GNOME project and used for XML transformations is distributed under the MIT License http://www.opensource.org/licenses/mit-license.html. -- PCRE Perl regular expression compiler, copyrighted and distributed by University of Cambridge. ©1997-2006. The license agreement is at http://www.pcre.org/license.txt. -- BIND Balanced Binary Tree Library and Domain Name System (DNS) protocols developed and copyrighted by Internet Systems Consortium, Inc. (http://www.isc.org) -- Free BSD implementation of daemon developed by The FreeBSD Project, © 1994-2006. -- Simple Network Management Protocol Library developed and copyrighted by Carnegie Mellon University © 1989, 1991, 1992, Networks Associates Technology, Inc, © 2001- 2003, Cambridge Broadband Ltd. © 2001- 2003, Sun Microsystems, Inc., © 2003, Sparta, Inc, © 2003-2006, Cisco, Inc and Information Network Center of Beijing University of Posts and Telecommunications, © 2004. The license agreement for these is at http://net-snmp.sourceforge.net/about/license.html. -- NTP version 4.2 developed by Network Time Protocol and copyrighted to various contributors. -- Lightweight Directory Access Protocol developed and copyrighted by OpenLDAP Foundation. OpenLDAP is an open-source implementation of the Lightweight Directory Access Protocol (LDAP). Copyright © 1999-2003, The OpenLDAP Foundation. The license agreement is at http://www.openldap.org/software/release/license.html. Secure shell OpenSSH developed by OpenBSD project is released by the OpenBSD Project under a BSD-style license, available at http://www.openbsd.org/cgi-bin/cvsweb/src/usr.bin/ssh/LICENCE?rev=HEAD. -- PC/SC Lite is a free implementation of PC/SC, a specification for SmartCard integration is released under the BSD license. -- Postfix, an open source mail transfer agent (MTA), is released under the IBM Public License 1.0, available at http://www.opensource.org/licenses/ibmpl.php. -- PostgreSQL, a free software object-relational database management system, is released under a BSD-style license, available at http://www.postgresql.org/about/licence. -- PostgreSQL JDBC driver, a free Java program used to connect to a PostgreSQL database using standard, database independent Java code, (c) 1997-2005, PostgreSQL Global Development Group, is released under a BSD-style license, available at http://jdbc.postgresql.org/license.html. -- PostgreSQL Regular Expression Library, a free software object-relational database management system, is released under a BSD-style license, available at http://www.postgresql.org/about/licence. -- 21.vixie-cron is the Vixie version of cron, a standard UNIX daemon that runs specified programs at scheduled times. Copyright © 1993, 1994 by Paul Vixie; used by permission. -- JacORB, a Java object used to facilitate communication between processes written in Java and the data layer, is open source licensed under the GNU Library General Public License (LGPL) available at http://www.jacorb.org/lgpl.html. Copyright © 2006 The JacORB Project. -- TAO (The ACE ORB) is an open-source implementation of a CORBA Object Request Broker (ORB), and is used for communication between processes written in C/C++ and the data layer. Copyright (c) 1993-2006 by Douglas C. Schmidt and his research group at Washington University, University of California, Irvine, and Vanderbilt University. The open source software license is available at http://www.cs.wustl.edu/~schmidt/ACE-copying.html. -- libcURL, a library for downloading files via common network services, is open source software provided under a MIT/X derivate license available at http://curl.haxx.se/docs/copyright.html. Copyright (c) 1996 - 2007, Daniel Stenberg. -- libuuid, a library used to generate unique identifiers, is released under a BSD-style license, available at http://thunk.org/hg/e2fsprogs/?file/fe55db3e508c/lib/uuid/COPYING. Copyright (C) 1996, 1997 Theodore Ts'o. -- libpopt, a library that parses command line options, is released under the terms of the GNU Free Documentation License available at http://directory.fsf.org/libs/COPYING.DOC. Copyright © 2000-2003 Free Software Foundation, Inc. -- gSOAP, a development tool for Windows clients to communicate with the Intel Corporation AMT chipset on a motherboard, is distributed under the gSOAP Public License version 1.3b, available at http://www.cs.fsu.edu/~engelen/license.html. -- Windows Template Library (WTL) is used for developing user interface components and is distributed under the Common Public License v1.0 found at http://opensource.org/licenses/cpl1.0.php. -- The Perl Kit provides several independent utilities used to automate a variety of maintenance functions and is provided under the Perl Artistic License, found at http://www.perl.com/pub/a/language/misc/Artistic.html. -- rEFIt - libeg, provides a graphical interface library for EFI, including image rendering, text rendering, and alpha blending, and is distributed under the license found at http://refit.svn.sourceforge.net/viewvc/*checkout*/refit/trunk/refit/LICENSE.txt?revision=288. Copyright (c) 2006 Christoph Pfisterer. All rights reserved. -- Java Radius Client, used to authenticate PGP Universal Web Messenger users via Radius, is distributed under the Lesser General Public License (LGPL) found at http://www.gnu.org/licenses/lgpl.html. -- Yahoo! User Interface (YUI) library version 2.5.2, a Web UI interface library for AJAX. Copyright (c) 2009, Yahoo! Inc. All rights reserved. Released under a BSD-style license, available at http://developer.yahoo.com/yui/license.html. -- JSON-lib version 2.2.1, a Java library used to convert Java objects to JSON (JavaScript Object Notation) objects for AJAX. Distributed under the Apache 2.0 license, available at http://json-lib.sourceforge.net/license.html. -- EZMorph, used by JSON-lib, is distributed under the Apache 2.0 license, available at http://ezmorph.sourceforge.net/license.html. -- Apache Commons Lang, used by JSON-lib, is distributed under the Apache 2.0 license, available at http://commons.apache.org/license.html. -- Apache Commons BeanUtils, used by JSON-lib, is distributed under the Apache 2.0 license, available at http://commons.apache.org/license.html. -- SimpleIni is an .ini format file parser and provides the ability to read and write .ini files, a common configuration file format used on Windows, on other platforms. Distributed under the MIT License found at http://www.opensource.org/licenses/mit-license.html. Copyright 2006-2008, Brodie Thiesfield. -- uSTL provides a small fast implementation of common Standard Template Library functions and data structures and is distributed under the MIT License found at http://www.opensource.org/licenses/mit-license.html. Copyright (c) 2005-2009 by Mike Sharov <msharov@users.sourceforge.net>. -- Protocol Buffers (protobuf), Google's data interchange format, are used to serialize structure data in the PGP SDK. Distributed under the BSD license found at http://www.opensource.org/licenses/bsd-license.php. Copyright 2008 Google Inc. All rights reserved.

Additional acknowledgements and legal notices are included as part of the PGP Universal Server.

Export Information

Export of this software and documentation may be subject to compliance with the rules and regulations promulgated from time to time by the Bureau of Export Administration, United States Department of Commerce, which restricts the export and re-export of certain products and technical data.

Limitations

The software provided with this documentation is licensed to you for your individual use under the terms of the End User License Agreement provided with the software. The information in this document is subject to change without notice. PGP Corporation does not warrant that the information meets your requirements or that the information is free of errors. The information may include technical inaccuracies or typographical errors. Changes may be made to the information and incorporated in new editions of this document, if and when made available by PGP Corporation.

Unsupported Third Party Products

By utilizing third party products, software, drivers, or other components ("Unsupported Third Party Product") to interact with the PGP software and/or by utilizing any associated PGP command or code provided by to you by PGP at its sole discretion to interact with the Unsupported Third Party Product ("PGP Third Party Commands"), you acknowledge that the PGP software has not been designed for or formally tested with the Unsupported Third Party Product, and therefore PGP provides no support or warranties with respect to the PGP Third Party Commands or the PGP software's compatibility with Unsupported Third Party Products. THE PGP THIRD PARTY COMMANDS ARE PROVIDED "AS IS," WITH ALL FAULTS, AND THE ENTIRE RISK AS TO SATISFACTORY QUALITY, PERFORMANCE, ACCURACY, AND EFFORT IS WITH YOU. TO THE MAXIMUM EXTENT PERMITTED BY APPLICABLE LAW, PGP DISCLAIMS ALL REPRESENTATIONS, WARRANTIES, AND CONDITIONS, WHETHER EXPRESS OR IMPLIED, INCLUDING ANY WARRANTIES OR CONDITIONS OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE, TITLE, NONINFRINGEMENT, QUIET ENJOYMENT, AND ACCURACY WITH RESPECT TO THE PGP THIRD PARTY COMMANDS OR THE PGP SOFTWARE'S COMPATIBILITY WITH THE UNSUPPORTED THIRD PARTY PRODUCT.

Contents

1 Introduction

This guide tells you how to use PGP Whole Disk Encryption for Linux.

In This Chapter

About PGP Whole Disk Encryption for Linux

Thank you for using PGP Whole Disk Encryption for Linux, a software product from PGP Corporation that locks down the entire contents of your Linux system using PGP Whole Disk Encryption (WDE) technology.

For more information about PGP WDE, see the:

- *PGP Desktop User's Guide*
- *PGP WDE Quick Start Guide*
- *PGP WDE Data Sheet* (available via the PGP WDE page on the PGP Corporation website)

PGP Whole Disk Encryption for Linux gives you access to PGP WDE functionality using a command-line interface.

The encryption algorithm used by PGP Whole Disk Encryption for Linux is AES-256. The hashing algorithm is SHA-1. You cannot change these.

Warning: Once you unlock a disk, its files are available to you—as well as anyone else who can physically use your system. Your files are unlocked until you lock them again by shutting down your system.

Important Terms

Understanding the following terms will help make it easier to use PGP Whole Disk Encryption for Linux:

- **PGP Whole Disk Encryption (PGP WDE):** a technology that encrypts the entire contents of a disk; boot disks, partitions, and non-boot disks such as USB thumb drives can all be whole disk encrypted.
- **PGP Whole Disk Encryption for Linux:** a software product from PGP Corporation that brings PGP WDE technology to the Linux platform, allowing you to lock down the entire contents of your Linux system.
- **command line:** the interface to PGP Whole Disk Encryption for Linux functionality. All PGP Whole Disk Encryption for Linux commands and options are accessed via the command-line interface.
- **passphrase user:** a user who can authenticate to an encrypted disk using a passphrase.
- **public-key user:** a user who can authenticate to an encrypted disk using the passphrase to the corresponding private key.
- **encrypt:** the process of "scrambling" data so that it is not usable unless you properly authenticate.
- **decrypt:** the process of "unscrambling" encrypted data.
- **master boot record (MBR):** software on a disk that is "in front" of the partition table; that is, it is implemented during the startup process *before* the operating system itself. The instructions in the MBR tells the system how to boot.
- **instrument:** a part of the process of whole disk encrypting a disk/partition where the Linux MBR is replaced with the PGPMBR.
- **PGPMBR:** an MBR from PGP Corporation that implements the PGP BootGuard. Once a disk is instrumented, even if it is not fully encrypted, subsequent startups will bring up PGP BootGuard.
- **PGP BootGuard:** the screen that appears after instrumenting a disk that requires proper authentication for the boot process to continue. If proper authentication is *not* provided, the boot process will not continue; the operating system will not load and the system will not be usable.
- **uninstrument:** removing the PGPMBR and replacing it with the original Linux MBR (which was saved when the disk was instrumented).
- **whole disk recovery token (WDRT):** an additional passphrase for a whole disk encrypted disk that is passed to the appropriate PGP Universal Server if the disk is part of a PGP Universal-managed environment.
- **PGP Universal Server:** a management console for securing data from PGP Corporation.

- **managed user:** someone using PGP Whole Disk Encryption for Linux in a PGP Universal Server-managed environment. Managed users receive policies and settings from their PGP Universal Server.
- **enroll:** the process of a user in a PGP Universal Server-managed environment contacting their PGP Universal Server so that they can receive applicable policies and settings.
- **standalone user:** someone using PGP Whole Disk Encryption for Linux with no associated PGP Universal Server. Standalone users establish their own policies and settings.
- **recovery:** the process of restoring access to a disk/partition that has been whole disk encrypted but now cannot be decrypted.

Audience

This User's Guide is for anyone who is going to be using PGP Whole Disk Encryption for Linux to perform PGP WDE functions on their Linux system.

System Requirements

The system requirements for PGP Whole Disk Encryption for Linux are:

- Ubuntu 8.04 and 9.04 (32-bit versions) and Red Hat Enterprise Linux/CentOS 5.2 and 5.3 (32-bit versions), Ubuntu 8.04 and 9.04 (64-bit versions), Red Hat Enterprise Linux 5.2 and 5.3 (64-bit versions)

Note: CentOS is free, open source software based on Red Hat Enterprise Linux. For the purposes of supporting PGP Whole Disk Encryption for Linux, the two are functionally equivalent.

- 512 MB of RAM
- 64 MB hard disk space

Using PGP Whole Disk Encryption for Linux in a PGP Universal Server-Managed Environment

If you are using PGP Whole Disk Encryption for Linux in a PGP Universal Server-managed environment, your PGP Universal administrator may have enabled or disabled certain features. For example, you may be required to encrypt your drive immediately after enrolling with your PGP Universal Server.

If you have any questions about features that may be have been automatically enabled or disabled, contact your PGP Universal administrator.

2 Installing and Uninstalling

This section describes how to install and uninstall PGP Whole Disk Encryption for Linux.

In This Chapter

Installing

The PGP Whole Disk Encryption for Linux installer is a bsx (Bash Self-eXtracting) file.

You must have root privileges to install.

> **Note:** The installer file may have a slightly different filename than shown in the procedure below depending on the platform you are installing onto.

To install PGP Whole Disk Encryption for Linux

1. Download the installer file, called `pgp_desktop_10.0.1_linux_ub9.04_i386.bsx` for Ubuntu 9.04, to a known location on your system.
2. Begin the installation process using either of the following methods:

   a Make the file an executable (using `chmod +x [filename]`), then use `./[filename]` **Enter** to begin the installation.

   or

   b Begin the installation via a shell: `bash [filename]` **Enter**
3. Follow the on-screen instructions.
4. Reboot your system when the installation is complete.

Uninstalling

Use the built-in uninstaller for the version of Linux you are using to uninstall PGP Whole Disk Encryption for Linux. You must have root privileges to uninstall.

Warning: You must decrypt any whole disk encrypted drives before uninstalling PGP Whole Disk Encryption for Linux or removing any packages.

The packages that are installed are: pgp-libs, pgpwde, pgp-release, and kmod-pgpwde.

3 Licensing

This section describes how to license PGP Whole Disk Encryption for Linux.

You must *license* PGP Whole Disk Encryption for Linux if you are using it standalone; that is, you are *not* in a PGP Universal Server-managed environment.

You do not need to *enroll* PGP Whole Disk Encryption for Linux if you are using it standalone; that is only required for PGP Universal Server-managed environments.

Note: As PGP Whole Disk Encryption for Linux will not operate normally until licensed, you should license it immediately after installation.

In This Chapter

Overview

PGP Whole Disk Encryption for Linux requires a valid license to operate. This section describes how to license your copy of PGP Whole Disk Encryption for Linux.

PGP Whole Disk Encryption for Linux supports the following licensing scenarios:

- Using a License Number**.** This is the normal method to license PGP Whole Disk Encryption for Linux. You must have your license information and a working connection to the Internet.
- Through a Proxy Server. If you connect to the Internet through a proxy server, use this method to license PGP Whole Disk Encryption for Linux. You must have your license information and the appropriate proxy server information.

The licensing command is `--license-authorize.`

Once PGP Whole Disk Encryption for Linux is correctly installed and licensed on your system, you can encrypt your drive. See The Encryption Process for complete information.

--license-authorize

Use `--license-authorize` to license PGP Whole Disk EncryptionLinux.

The usage format is:

```
pgpwde --license-authorize --license-name <name> --
license-number <number> [--license-email
<emailaddress>] [--license-organization <org>]
```

Where:

- `--license-authorize` is the command to license PGP Whole Disk Encryption for Linux.
- `--license-name` is the option to specify the user.

  `<name>` is your name or a descriptive name.
- `--license-number` is the option to enter a license number.

  `<number>` is a valid license number for PGP Whole Disk Encryption for Linux.
- `--license-email` is the option to enter an email address.

  `<emailaddress>` is a valid email address.
- `--license-organization` is the option to enter an organization.

  `<org>` is the name of your organization.

If you decide not to enter a license email, you may see a warning message but your license will authorize.

Example:

```
pgpwde --license-authorize --license-name "Alice
Cameron"
--license-number "aaaaa-bbbbb-ccccc-ddddd-eeeee-fff"
--license-email "acameron@example.com"
--license-organization "Example Corporation"
```

(When entering this text, it all goes on a single line.)

Licensing via a Proxy Server

If the Internet access of the system hosting PGP Whole Disk Encryption for Linux is via an HTTP proxy connection, you can still license your copy of PGP Whole Disk Encryption for Linux directly; you simply need to add the necessary proxy information.

Use `--license-authorize` to license PGP Whole Disk Encryption for Linux via a proxy server.

The usage format is:

```
pgpwde --license-authorize --license-name <name> --
license-number <number> [--license-email
<emailaddress>] [--license-organization <org>] [--
proxy-server <proxyserver>] [--proxy-username
<proxyusername>] [--proxy-passphrase <proxypass>]
```

Where:

- `--license-authorize` is the command to license PGP Whole Disk Encryption for Linux.
- `--license-name` is the option to specify the user.

  `<name>` is your name or a descriptive name.
- `--license-number` is the option to enter a license number.

  `<number>` is a valid license number for PGP Whole Disk Encryption for Linux.
- `--license-email` is the option to enter an email address.

  `<emailaddress>` is a valid email address.
- `--license-organization` is the option to enter an organization.

  `<org>` is the name of your organization.
- `--proxy-server` is the command to go through a proxy server to access the Internet.

  `<proxyserver>` is the appropriate proxy server.
- `--proxy-username` is the command to specify a user on the proxy server when authentication is required.

  `<proxyusername>` is a valid username on the specified proxy server.
- `--proxy-passphrase` is the option to specify the passphrase of the specified user when authentication is required.

  `<proxypass>` is the passphrase for the specified user on the proxy server.

Example:

```
pgpwde --license-authorize --license-name "Alice
Cameron"
--license-number "aaaaa-bbbbb-ccccc-ddddd-eeeee-fff"
--license-email "acameron@example.com"
--license-organization "Example Corporation"
--proxy-server "proxyserver.example.com"
--proxy-username "acameron"
--proxy-passphrase 'a_cameron1492sailedblue'
```

(When entering this text, it all goes on a single line.)

4 Enrolling

This section describes how to enroll PGP Whole Disk Encryption for Linux.

You must enroll PGP Whole Disk Encryption for Linux if you are using it in a PGP Universal Server-managed environment.

You do not need to *license* PGP Whole Disk Encryption for Linux in a PGP Universal Server-managed environment, as the license is included in the installer.

Note: As PGP Whole Disk Encryption for Linux will not operate normally until you enroll, you should enroll immediately after installation.

In This Chapter

Overview

You must enroll with a PGP Universal Server before you can use any PGP Whole Disk Encryption for Linux features in a PGP Universal Server-managed environment.

When enrollment is complete, PGP Whole Disk Encryption for Linux will receive policies and settings from its PGP Universal Server. It will also send information to the PGP Universal Server that can be seen by the PGP Universal administrator.

Note: You must initiate enrollment on your own. You will not be prompted to do so.

Enrollment uses LDAP credentials. The username and passphrase required for both enrolling and checking enrollment status are the username and passphrase of the user on the LDAP server.

If enrollment is unsuccessful, contact your PGP Universal administrator for assistance.

You can check the enrollment status of a client using the `--check-enroll` command. When successful, this command will note that the client is enrolled and will download the latest policies and settings. If unsuccessful, this means that the client must enroll again because of a change of policies or settings on the PGP Universal Server.

Once PGP Whole Disk Encryption for Linux is correctly installed on your system and you have enrolled, you can encrypt your drive. Refer to The Encryption Process for complete information.

--enroll

Use `--enroll` to enroll PGP Whole Disk Encryption for Linux.

Entering a username and passphrase on the command line are optional. If you do not enter them, you will be prompted for them.

Note: `--enroll` is preceded by **pgpenroll** instead of the usual **pgpwde**.

The usage format is:

```
pgpenroll --enroll [--username <user>] [--passphrase
<phrase>]
```

Where:

- `--enroll` is the command to enroll with a PGP Universal Server.
- `--username` specifies a username for an operation (optional).

  `<user>` is the username (on the LDAP server) of the user being enrolled.
- `--passphrase` specifies the passphrase for an operation (optional).

  `<phrase>` is the passphrase (on the LDAP server) of the user being enrolled.

Examples:

-
  ```
  pgpenroll --enroll --username "Alice Cameron"
  --passphrase 'Frodo@Baggins22'
  ```

  This example shows user Alice Cameron enrolling PGP Whole Disk Encryption for Linux. The username and passphrase she is using are her credentials on her organization's LDAP server.

-
  ```
  pgpenroll --enroll
  ```

  This example shows a user enrolling PGP Whole Disk Encryption for Linux. Because the username and passphrase are not supplied on the command line, the enrolling user will be prompted for them.

--check-enroll

Use `--check-enroll` to check the enrollment status of a client.

Note: `--check-enroll` is preceded by **pgpenroll** instead of the usual **pgpwde**.

If the enrollment check fails, contact your PGP Universal administrator for instructions.

The usage format is:

```
pgpenroll --check-enroll [--username <user>]
[--passphrase <phrase>]
```

Where:

- `--enroll` is the command to check the enrollment status of a client.
- `--username` specifies a username (on the LDAP server) for an operation.

  `<user>` is the username of the user whose enrollment status is being checked.
- `--passphrase` specifies the passphrase for an operation.

  `<phrase>` is the passphrase (on the LDAP server) of the user whose enrollment status is being checked.

Example:

```
pgpenroll --check-enroll --username "Alice Cameron"
--passphrase 'Frodo@Baggins22'
```

This example shows the enrollment status of Alice Cameron being checked.

5 The Command-Line Interface

This section describes the command-line interface used by PGP Whole Disk Encryption for Linux .

In This Chapter

Overview

PGP Whole Disk Encryption for Linux uses a command-line interface.

Note: Versions of PGP Whole Disk Encryption for other platforms support both a graphical user interface and a command line interface. PGP Whole Disk Encryption for Linux has only a command-line interface.

You enter a valid command at the command prompt and press **Enter**. PGP Whole Disk Encryption for Linux responds based on what you entered: with success (if you entered a valid command) or with an error message (if you entered an invalid or incorrectly structured command).

All PGP Whole Disk Encryption for Linux commands have a *long form*: the text "pgpwde", a space, two hyphens "--", the command name, and options (if appropriate).

For example:

```
$pgpwde --help [Enter]
```

is the command to display the built-in help information. It has no options.

(The command prompt, $ in the above example, and [Enter] will no longer be shown in examples; only the necessary commands and options will be shown.)

A few commands also have a *short form*: either one hyphen and then a single letter or two hyphens and two letters.

For example:

`-h` for help instead of `--help`

`--aa` for administrative authorization instead of `--admin-authorization`

You can mix long forms and short forms in a single command.

Short forms are noted where appropriate.

Scripting

PGP Whole Disk Encryption for Linux commands can easily be inserted into scripts for automating common tasks, such as encrypting a disk or getting information about an encrypted disk.

PGP Whole Disk Encryption for Linux commands can easily be added to scripts written with scripting languages such as Perl or Python.

WDE-ADMIN Active Directory Group

If you are an administrator of PGP Whole Disk Encryption for Linux clients in a PGP Universal environment and using Active Directory, you can create a special Active Directory group to allow you to run commands on your managed PGP Whole Disk Encryption for Linux clients without knowing the passphrase of a user on the encrypted disk.

This special Active Directory group, which *must* be called WDE-ADMIN, must be a security group, not a distribution group.

Using the `--admin-authorization` option is useful for running administrative tasks in an enterprise.

Refer to the *PGP Universal Administrator's Guide* for more information about creating and using the WDE-ADMIN Active Directory group.

Passphrases

For consistency, all example passphrases in this guide are shown in single quotation marks ('). Putting passphrases between single quotation marks ensures that reserved characters and spaces are interpreted correctly.

If you do not use any reserved characters or spaces in your passphrases, then you do not have to enclose them in single quotation marks.

On Windows systems, for example, if you have a space in a passphrase, you must enclose the passphrase in single or double quotation marks when you enter it. Also, double quotation marks (") as part of the passphrase must be escaped with a preceding double quotation mark.

For example, if you want to use

Thomas "Stonewall" Jackson

as your passphrase, you would have to enter it as

'Thomas ""Stonewall"" Jackson'

on the command line. You need the quotation marks at the beginning and end for the spaces and you need to escape each double quotation mark used in the passphrase with another double quotation mark.

Note: If you are having problems entering certain characters in your passphrases, check the information about how to handle reserved characters for the operating system or shell interpreter you are using.

--interactive

You can use `--interactive` whenever you could use a command that requires a passphrase be entered on the command line. If you do, you will be prompted to enter a valid passphrase on a separate line.

Using `--interactive` makes using PGP Whole Disk Encryption for Linux more secure by preventing passphrases from being entered in the clear on the command line. When you use `--interactive`, the characters you enter are *not* displayed.

Note: `--interactive` is also used in a different way when configuring local self recovery. See *Local Self Recovery* for more information.

6 Before You Encrypt

When you encrypt an entire disk using PGP Whole Disk Encryption for Linux, every sector is encrypted using a symmetric key. This includes all files including operating system files, application files, data files, swap files, free space, and temp files.

On subsequent reboots, PGP Whole Disk Encryption for Linux prompts you for the correct passphrase. As long as you correctly authenticate to your PGP Whole Disk Encryption for Linux-encrypted disk (after you enter the correct passphrase at the PGP BootGuard screen), your files are available. When you shut down your system, the disk is protected against use by others.

Before encrypting your disk with PGP Whole Disk Encryption for Linux, there are some important things to do:

- Ensure the health of the hard disk.
- Choose the encryption options to use.
- Make sure to maintain power throughout encryption.

In This Chapter

Ensure Disk Health

PGP Corporation deliberately takes a conservative stance when encrypting drives, to prevent loss of data. It is not uncommon to encounter Cyclic Redundancy Check (CRC) errors while encrypting a hard disk.

If PGP Whole Disk Encryption for Linux encounters a hard drive or partition with bad sectors, it will, by default, pause the encryption process. This pause allows you to remedy the problem before continuing with the encryption process, thus avoiding potential disk corruption and lost data.

To avoid disruption during encryption, PGP Corporation recommends that you start with a healthy disk by correcting any disk errors prior to encrypting.

As best practices, before you attempt to encrypt your drive:

- use a third-party scan disk utility that has the ability to perform a low-level integrity check and repair any inconsistencies with the drive that could lead to CRC errors.

Choose Encryption Options

There are several options you can use during the encryption process itself:

- `--dedicated-mode`: Uses maximum computer power to encrypt faster; your system is less responsive during encryption.
- `--fast-mode`: Skips unused sectors, so encryption of the disk is faster.
- `--safe-mode`: Allows encryption to be resumed without loss of data if power is lost during encryption; encryption takes longer.

These options are also described with the `--encrypt` command.

Maintain Power Throughout Encryption

Because encryption is a CPU-intensive process, encryption cannot begin on a laptop computer that is running on battery power. The computer *must* be on AC power. Do not remove the power cord from the system before the encryption process is over.

Regardless of the type of computer you are working with, your system must not lose power, or otherwise shut down unexpectedly, during the encryption process, unless you use the `--safe-mode` option. Even if you are using the `--safe-mode` option, it is still better *not* to lose power during the encryption process.

If loss of power during encryption is a possibility—or if you do not have an uninterruptible power supply for your computer—be sure to use the `--safe-mode` option.

7 The Encryption Process

This section describes the two methods for whole disk encrypting a drive.

In This Chapter

Overview

To PGP Whole Disk Encrypt a drive requires several things: the drive must be instrumented, there must be at least one authorized user on the drive, and the drive must be encrypted.

There are two ways to PGP Whole Disk Encrypt a drive:

- **using a single command, --secure:** this one command does all three of the above actions. It instruments the drive, creates an authorized user, and encrypts the drive. This command is most useful when you have just installed PGP Whole Disk Encryption for Linux and thus have not instrumented any drives, created any authorized users, or encrypted any drives.
- **using multiple commands:** for scenarios where you do not need all three things required to PGP Whole Disk Encrypt at drive, or if you just prefer using individual commands, you can use `--instrument`, `--add-user`, and finally `--encrypt` to PGP Whole Disk Encrypt a drive.

Using --secure

The `--secure` command instruments the drive, creates an authorized user, and encrypts the drive, all using a single command.

Note: PGP Whole Disk Encryption for Linux must be correctly installed and licensed before you can use `--secure`.

Refer to Disk Operation for more information about the `--secure` command.

To PGP Whole Disk Encrypt a drive using a single command

1. Access a command prompt on your system.
2. Enter the text for the `--secure` command on a single line.

   For example:

   ```
   pgpwde --secure --disk 0 --username "Alice Cameron"
   --passphrase 'Frodo*1*Baggins22' --all --fast-mode
   ```

3. Press **Enter**. PGP Whole Disk Encryption for Linux begins to PGP Whole Disk Encrypt the drive.

You can check the progress of the encryption process using the `--status` command. Run the command and check the highwater mark; it will continue to get larger as the encryption process continues.

Using Individual Commands

For scenarios where you do not need to instrument a drive, add a user, and encrypt the drive all at the same time or if you just prefer using individual commands, you can run the three needed commands individually.

The three commands and the order in which you need to run them are:

- `--instrument:` replaces the Linux MBR with the PGPMBR.
- `--add-user:` adds an authorized user to the drive.
- `--encrypt:` encrypts the drive.

To PGP Whole Disk Encrypt a drive using individual commands

1. Access a command prompt on your system.
2. Enter the text for the `--instrument` command on a single line, then press **Enter**.

   For example:

   ```
   pgpwde --instrument --disk 0
   ```

   This example instruments the boot drive. You can use the `--status` command to make sure the drive was instrumented.

3. Enter the text for the `--add-user` command on a single line, then press **Enter**.

   For example:

   ```
   pgpwde --add-user --disk 0 --username "Alice Cameron"
   --passphrase 'Frodo@Baggins22'
   ```

   This example adds a user named Alice Cameron to the boot drive. You can use the `--verify-user` command to make sure the user was created.

4 Enter the text for the `--encrypt` command on a single line, then press **Enter**.

For example:

```
pgpwde --encrypt --disk 0 --passphrase
'Frodo@Baggins22' --all --safe-mode
```

This example encrypts all partitions of the boot drive in safe mode.

You can check the progress of the encryption process using the `--status` command. Run the command and check the highwater mark; it will continue to get larger as the encryption process continues.

8 The PGP BootGuard Screen

This section describes actions you can take at the PGP BootGuard screen.

In This Chapter

Overview

Your computer boots up in a different way once you use PGP Whole Disk Encryption for Linux to protect the boot disk—or a secondary fixed disk—on your system. On power-up, the first thing you see is the PGP BootGuard log-in screen asking for your passphrase. When you properly authenticate, PGP Whole Disk Encryption for Linux decrypts the disk.

When you use a PGP WDE-encrypted disk, it is decrypted and opened automatically as needed. With most modern computers, after the disk is completely encrypted, there is no noticeable slowdown of your activities.

Once you unlock a disk or partition, its files are available to you—as well as anyone else who can physically use your system. Your files are unlocked until you lock them again by shutting down your computer.

When you shut down a system with an encrypted boot disk or partition, or if you remove an encrypted removable disk from the system, all files on the disk or partition remain encrypted and fully protected—data is never written to the disk or partition in an unencrypted form. Proper authentication (passphrase, token, or private key) is required to make the files accessible again.

On the PGP BootGuard log-in screen you can:

- Authenticate an encrypted boot or secondary disk or partition on the system.
- View information about the disks or partitions on your system.
- Authenticate if you have forgotten your passphrase.
- Choose your keyboard layout.

Authenticating

The PGP BootGuard log-in screen prompts you for the proper passphrase for a protected disk or partition for one of two reasons:

- If your boot disk or partition is protected using PGP Whole Disk Encryption for Linux, you must authenticate correctly for your system to start up. This is required because the operating system files that control system startup are encrypted, and must be decrypted before they can be used to start up the system.
- If a secondary fixed disk or partition is protected using PGP Whole Disk Encryption for Linux, you can authenticate at startup so that you don't have to authenticate later when you need to use files on the secondary disk or partition. Because the files on the secondary (non-boot) disk or partition are not required for startup, you are not required to authenticate at startup.

Note: The PGP BootGuard log-in screen accepts the authentication information from any user configured for an encrypted disk or partition. For example, if you have two users configured for a boot disk or partition and two different users configured for a secondary fixed disk or partition on the same system, *any* of the four configured users can use their passphrase to authenticate on the PGP BootGuard log-in screen at startup, even the two users configured on the secondary disk or partition.

▸ To authenticate at the PGP BootGuard log-in screen

1. Start or restart the system that has a disk or partition protected by PGP Whole Disk Encryption for Linux. On startup, the PGP BootGuard log-in screen is displayed.
2. Type a valid passphrase and press **Enter**.

   Caution: The PGP BootGuard log-in screen assumes you are using one of the supported keyboard layouts when you type your passphrase. If you used a different keyboard layout to create the passphrase for a disk or partition protected by PGP Whole Disk Encryption for Linux, you could have problems authenticating because the mappings between the keyboard layouts may be different.

   To see the characters you type, press **Tab** before you begin typing.
3. If you entered a valid passphrase, the PGP BootGuard log-in screen goes away and the system boots normally.

   If you typed an invalid passphrase, an error message is displayed. Try typing the passphrase again.

Authenticating if You Have Forgotten Your Passphrase

If you have forgotten your passphrase and cannot authenticate to the PGP BootGuard screen, you can authenticate using local self recovery if you have previously configured it.

Note: Local self recovery *must* be configured in advance.

See Local Self Recovery for information about using the command line or a text file to configure the local self recovery questions.

▸ To authenticate at the PGP BootGuard screen using local self recovery

1. On the PGP BootGuard screen, use the arrow keys to select **Forgot Passphrase** in the lower right corner, then press **Enter**. A new screen appears, showing the first local self recovery question.
2. Enter the answer to the first question, then press **Enter**. The second question appears.
3. Enter the answer to the second question, then press **Enter**. The third question appears.
4. Enter the answer to the third question, then press **Enter**. The fourth question appears.
5. Enter the answer to the fourth question, then press **Enter**. The fifth and last question appears.
6. Enter the answer to the fifth question, then press **Enter**.

   If you entered three or more of the questions correctly, the PGP BootGuard screen goes away and the system boots normally.

   If you did not enter three or more questions correctly, you are given another chance.

If you subsequently remember your original passphrase, you can continue using it. Using local self recovery does not remove your passphrase.

If you do not believe you will ever remember your original passphrase, you can change your passphrase after authenticating to PGP BootGuard using the `--recovery-change-passphrase` command. This means that you do not have to continue using the local self recovery questions to authenticate to PGP BootGuard. Using this command does remove your original passphrase, so it will not work if you remember it later.

Choosing a Keyboard

The PGP BootGuard screen lets you change your keyboard layout.

Note: Different keyboard layouts can have different mappings between characters, potentially causing problems when you enter your passphrase to authenticate. Select the keyboard layout that most closely maps to the keyboard you are using, then make sure to use that same layout each time you authenticate.

▸ To select a keyboard layout at the PGP BootGuard screen

1. On the PGP BootGuard screen, use the arrow keys to select **Keyboard** in the lower right corner, then press **Enter**. A list of supported keyboard layouts is displayed.
2. Use the arrow keys to select the desired keyboard layout, then press **Enter**. The text under the list of supported keyboard layouts changes to reflect the new keyboard layout.
3. Press **Tab** to move focus to the **Go Back** command, then press **Enter**.

9 Generic Commands

PGP Whole Disk Encryption for Linux generic commands are:

- `--help (-h)`, which shows basic help information for PGP Whole Disk Encryption for Linux.
- `--version`, which shows version information for PGP Whole Disk Encryption for Linux.

In This Chapter

--help (-h)

The `--help` command provides a brief description of the commands and options available in PGP Whole Disk Encryption for Linux.

The long form usage format is:

```
pgpwde --help
```

The short form usage format is:

```
pgpwde -h
```

Example:

```
pgpwde --help

PGP WDE command line tool.

Usage: pgpwde --action [--options]

ACTIONS

-h  --help        Print this help
```

and so on.

This example shows the response to the `--help` command.

--version

The `--version` command displays information about the version of PGP Whole Disk Encryption for Linux you are using.

The usage format is:

```
pgpwde --version
```

Example:

```
pgpwde --version

PGP WDE, Version 10. 1

Copyright (C) 2010 PGP Corporation

Request sent to Version was successful
```

This example shows the response to the `--version` command.

10 Disk Information Commands

PGP Whole Disk Encryption for Linux includes several commands that provide information about the disks on a system and their status:

- `--enum`: Tells you about the disks on the system, including disk designation.
- `--status`: Gives you PGP WDE information about a disk on the system.
- `--show-config`: Gives you PGP BootGuard information about a disk on the system.
- `--info`: Gives you general information about a disk on the system.

In This Chapter

--enum

The `--enum` command displays disk designations (for example, Disk 0 as the boot disk), which is used in other PGP Whole Disk Encryption for Linux commands.

The usage format is:

```
pgpwde --enum
```

Where:

`--enum` displays information about the disks on your system.

Example:

- `pgpwde --enum`

```
Total number of installed fixed/removable storage
device (excluding floppy and CDROM): 1

Disk 0 has 2 online volumes
```

```
  volume /dev/hda1 is on partition 1 with offset 63
  volume /dev/hda5 is on partition 5 with offset
64260063
Request sent to Enumerate was successful
```

This example shows that the system has one disk, Disk 0, with two online volumes. Disk 0 is the boot disk in most cases.

--info

The `--info` command provides general status information for the specified disk.

Note: Use the `--status` command for PGP WDE-specific information about a disk.

Information you can see about a disk using `--info` includes:

- model information.
- total number of sectors on the disk.

The usage format is:

```
pgpwde --info --disk <number>
```

Where:

- `--disk` specifies the disk to which the operation applies.
- `<number>` is the disk number on the system.

Examples:

- `pgpwde --info --disk 0`

  ```
  Disk information for disk 0.
  Model Number: ST910021AS
  Total number of sectors on disk: 192426569
  Request sent to Display disk information was successful
  ```

  This example shows the model number and sectors for a boot disk.

- `pgpwde --info --disk 1`

  ```
  Disk information for disk 1.
     Model Number: SanDisk U3 Titanium USB 2.18
     Total number of sectors on disk: 4001425
  Request sent to Display disk information was successful
  ```

  This example shows the model number and sectors for a USB thumb drive.

--show-config

The `--show-config` command displays information about how PGP BootGuard is configured on an encrypted disk.

No information displays if the command is run on a disk that is not encrypted by PGP WDE.

The usage format is:

```
pgpwde --show-config --disk <number>
```

Where:

- `--disk` specifies the disk to which the operation applies.
- `<number>` is the disk number on the system.

Examples:

- `pgpwde --show-config --disk 0`

```
         Login Message:
Display Startup Screen: No
     Use Audio Prompts: No
          User lockout: Disabled
    Allow user decrypt: Yes
Show configuration information completed
```

  This example shows the PGP BootGuard information for a boot disk that is encrypted. An error displays if the disk is not encrypted.

--status

The `--status` command provides PGP WDE-specific status information for the specified disk.

(Use the `--info` command for general information about a disk.)

Information you can see about a disk using `--status` includes:

- whether or not the disk is instrumented.
- whether or not the disk is whole disk encrypted.
- the number of sectors on the disk.
- the highwater mark (the number of encrypted sectors on the disk).

Note: If you are encrypting or decrypting a disk, and you want to check progress, you can run `--status` periodically and check the high water mark; this number increases as encryption progresses or decreases as decryption progresses.

The usage format is:

```
pgpwde --status --disk <number>
```

Where:

- `--disk` is the option specifying to which disk on the system the information applies.
- `<number>` is the disk number on the system.

Examples:

- `pgpwde --status --disk 0`

  ```
  Disk disk0 is instrumented by bootguard.
      Current key is valid.
  Whole disk encrypted
     Total sectors: 192426569  highwatermark: 192426569
  Request sent to Disk status was successful
  ```

  In this example, Disk 0 is instrumented by PGP BootGuard, the current key used for authentication is valid, the disk is encrypted, the total number of sectors on the disk is 192426569, and the high water mark (the number of sectors encrypted) is 192426569.

- `pgpwde --status --disk 1`

  ```
  Disk disk 1 is not instrumented by bootguard.
  Request sent to Disk status was successful
  ```

  In this example, disk 1 is *not* instrumented by PGP BootGuard.

11 Boot Bypass Commands

The boot bypass feature lets you reboot a system one or more times without having to authenticate at the PGP BootGuard screen.

Caution: Using the boot bypass feature weakens the protection provided by PGP Whole Disk Encryption. Pay extra attention to the physical security of systems when a bypass restart count exists. Use the `--remove-bypass` command to remove any unnecessary remaining bypass restarts.

Boot bypass is generally used for remote deployment or upgrade scenarios when one or more reboots is required; patch management, for example.

By default, boot bypass is disabled for a system. You must use the `--add-bypass` command to enable bypass restarts.

Note: All three boot bypass commands apply to the boot disk only, even if you specify another disk on the command line.

The Boot Bypass commands are:

- `--add-bypass`: Enables the specified disk for boot bypass.
- `--check-bypass`: Checks to see if the specified disk is enabled for boot bypass.
- `--remove-bypass`: Removes boot bypass from a disk where it is enabled.

In This Chapter

--add-bypass

The `--add-bypass` command lets you enable one or more bypass restarts for a system.

You can set the number of bypass restarts for a system to any value from 0 to 4,294,967,295. Setting the count to 0 (zero) disables bypass restarts. Setting the count from 1 to 4,294,967,295 allows that many bypass restarts, as long as the count is not higher than the preference set on the PGP Universal Server.

In a PGP Universal-managed environment, the PGP administrator can establish a preference on the PGP Universal Server that limits the number of bypass restarts that can be established using `--add-bypass`.

The preference is called **wdeMaximumBypassRestarts**. Setting the preference to 0 (zero) disables boot bypass. Setting the preference to a value from 1 to 4,294,967,295 allows that many bypass restarts. If the preference does not exist on the PGP Universal Server, the value is set to 1, allowing one bypass restart for each system.

If you enter a number on the command line that is greater than the preference on the PGP Universal Server, an error will be returned. The error message does not display the value configured for the preference.

The usage format is:

```
pgpwde --add-bypass --disk <number> --count
<bypassrestarts> --admin-authorization | --admin-
passphrase <phrase>
```

Where:

- `--disk` specifies the disk to which the operation applies.

  `<number>` is the disk number on the system.

- `--count` specifies that bypass restarts are being configured the boot disk on the system.

  `<bypassrestarts>` is the desired number of bypass restarts.

- `--admin-authorization` specifies that the command is being performed by a member of the WDE-ADMIN Active Directory group.

- `--admin-passphrase` specifies that the passphrase of an authorized user on the encrypted disk will be used to authenticate.

  `<phrase>` is the passphrase of an authorized user on the disk.

Example:

-
  ```
  pgpwde --add-bypass --disk 0 --count 4 --admin-
  passphrase 'bilbo@baggins42'
  ```

  This example shows that four bypass restarts was added to the boot disk on the system using the passphrase of an authorized user on the disk.

--check-bypass

The `--check-bypass` command tells you if boot bypass is configured for the specified boot disk. If configured, it also displays the original and remaining bypass restart counts.

The usage format is:

```
pgpwde --check-bypass --disk <number> –admin-authorization | –
-admin-passphrase <phrase>
```

Where:

- `--disk` specifies the disk to which the operation applies.
- `<number>` is the disk number on the system.
- `--admin-authorization` specifies that the command is being performed by a member of the WDE-ADMIN Active Directory group.
- `--admin-passphrase` specifies that the passphrase of an authorized user on the encrypted disk will be used to authenticate.
- `<phrase>` is the passphrase of an authorized user on the disk.

Examples:

- `pgpwde --check-bypass --disk 0 --admin-passphrase 'bilbo@baggins42'`

  This example shows that Disk 0 is configured for boot bypass via the presence of the "Bypass User."
- `pgpwde --check-bypass --disk 0 --admin-passphrase 'bilbo@baggins42'`

  This example shows that Disk 0 is **not** configured for boot bypass.

--remove-bypass

The `--remove-bypass` command removes boot bypass from the system, including the original and remaining bypass restart counts.

The usage format is:

```
pgpwde --remove-bypass --disk <number> –admin-authorization |
--admin-passphrase <phrase>
```

Where:

- `--disk` specifies the disk to which the operation applies.
- `<number>` is the disk number on the system.

- `--admin-authorization` specifies that the command is being performed by a member of the WDE-ADMIN Active Directory group.
- `--admin-passphrase` specifies that the passphrase of an authorized user on the encrypted disk will be used to authenticate.
- `<phrase>` is the passphrase of an authorized user on the disk.

Example:

-
  ```
  pgpwde --remove-bypass --disk 0 --admin-passphrase
  'bilbo@baggins42'
  ```

  ```
  Remove bypass completed
  ```

  This example shows the removal of boot bypass from a disk.

12 Disk Operation

The disk operation commands are:

- `--decrypt`: Decrypts the specified disk.
- `--encrypt`: Encrypts the specified disk.
- `--resume`: Resumes a halted encrypt or decrypt process.
- `--secure:` Encrypts a disk to a specified user and passphrase.
- `--stop`: Halts an encrypt or decrypt process.

In This Chapter

--decrypt

The `--decrypt` command starts the process of decrypting an encrypted disk.

If the disk is still being encrypted, you need to stop the encryption process using `--stop` before you can begin to decrypt it.

If you begin to decrypt an encrypted disk, you can pause the decrypt and then re-start the decrypt process, but you cannot stop the decrypt and then encrypt just the portion that was decrypted. If you being to decrypt an encrypted drive, you must *fully* decrypt it *before* you can re-encrypt it.

Note: If you are decrypting a disk, and you want to check progress, you can run `--status` periodically and check the lowwater and highwater marks.

The usage format is:

```
pgpwde --decrypt --disk <number> –admin-authorization |
--passphrase <phrase> --all --partition <partnumber>
```

Where:

- `--decrypt` specifies that the disk is to be decrypted.
- `--disk` specifies the disk to which the operation applies.
- `<number>` is the disk number on the system.
- `--admin-authorization` specifies that the command is being performed by a member of the WDE-ADMIN Active Directory group.
- `--passphrase` specifies the passphrase for an operation.
- `<phrase>` is the passphrase of an authorized user on the disk.
- `--all` specifies that all partitions should be decrypted.
- `--partition` specifies that only the listed partition should be decrypted.
- `<partnumber>` is the partition to be decrypted.

Example:

-
  ```
  pgpwde --decrypt --disk 0 --all --passphrase
  "Frodo*1*Baggins22"
  ```

  This example shows all partitions of a boot disk being decrypted.

--encrypt

The `--encrypt` command begins the process of whole disk encrypting a disk.

To use the `--encrypt` command, the drive to be encrypted must be instrumented and have at least one configured user. The –secure command instruments the drive, adds a user, and encrypts the drive using just one command.

Once the encryption process has started, you can stop it using `--stop`.

Three options are available for encrypting:

- `--dedicated-mode`: Uses maximum computer power to encrypt faster; your system is less responsive during encryption.
- `--fast-mode`: Skips unused sectors, so encryption of the disk is faster.
- `--safe-mode`: Allows encryption to be resumed without loss of data if power is lost during encryption; encryption takes longer.

The usage format is:

```
pgpwde --encrypt --disk <number> --passphrase <phrase>
| --keyid <keyid> --all --partition <partnumber>
--dedicated-mode --fast-mode --safe-mode
```

Where:

- `--encrypt` specifies that the disk is to be encrypted.

- `--disk` specifies the disk to which the operation applies.
- `<number>` is the disk number on the system.
- `--passphrase` specifies the passphrase for an operation.
- `<phrase>` is the passphrase of an authorized user on the disk.
- `--keyid` specifies a user by key ID for an operation.
- `<keyid>` is the key ID of an authorized user on the disk.
- `--all` specifies that all partitions should be decrypted.
- `--partition` specifies that only the listed partition should be encrypted.
- `<partnumber>` is the partition to be encrypted.
- `--dedicated-mode` specifies that dedicated mode (uses maximum computer power to encrypt faster) be used in the encryption process.
- `--fast-mode` specifies that fast mode (skipping unused sectors) be used in the encryption process.
- `--safe-mode` specifies that safe mode (encryption can be resumed without loss of data if power is lost) be used in the encryption process.

Example:

-
  ```
  pgpwde --encrypt --disk 0 --passphrase
  'Frodo*1*Baggins22' --safe-mode --all
  ```

  This example shows encryption of all partitions of a boot disk being started using safe mode. Authentication is provided by an authorized user.

--resume

The `--resume` command resumes a stopped process, either encrypting or decrypting a disk.

The usage format is:

```
pgpwde --resume --disk <number> --passphrase <phrase>
```

Where:

- `--resume` specifies that a stopped process is to be resumed.
- `--disk` specifies the disk to which the operation applies.
- `<number>` is the disk number on the system.
- `--passphrase` specifies the passphrase for an operation.
- `<phrase>` is the passphrase of an authorized user on the disk.

Example:

- `pgpwde --resume --disk 0 --passphrase 'Frodo@Baggins44'`

This example shows a stopped process being resumed on the boot disk.

--secure

The `--secure` command encrypts a disk to a specified user and passphrase.

It does three things that can also be done separately: it instruments the disk, adds a passphrase user, and encrypts the disk.

The usage format is:

```
pgpwde --secure --disk <number> --username <name>
--passphrase <phrase> --keyid <keyid> --all --partition
<partnumber> --dedicated-mode --fast-mode --safe-mode
```

Where:

- `--secure` specifies that: a disk is to be instrumented, a passphrase user created, and the disk encrypted.
- `--disk` specifies the disk to which the operation applies.
- `<number>` is the disk number on the system.
- `--username` specifies a passphrase user on the disk is to be created.
- `<name>` is the name of the passphrase user being created.
- `--passphrase` specifies a passphrase is to be created.
- `<phrase>` is the passphrase of the user being added to the disk.
- `--keyid` specifies a user by key ID for an operation.
- `<keyid>` is the key ID of an authorized user on the disk.
- `--all` specifies that all partitions should be encrypted.
- `--partition` specifies that only the listed partition should be encrypted.
- `<partnumber>` is the partition to be encrypted.
- `--dedicated-mode` specifies that dedicated mode (uses maximum computer power to encrypt faster) be used in the encryption process.
- `--fast-mode` specifies that fast mode (skipping unused sectors) be used in the encryption process.
- `--safe-mode` specifies that safe mode (encryption can be resumed without loss of data if power is lost) be used in the encryption process.

Example:

-
  ```
  pgpwde --secure --disk 0 --username "Alice Cameron"
  --passphrase 'Frodo*1*Baggins22' --all --safe-mode
  ```

  This example shows a boot disk being secured (instrumented and encrypted, with a new passphrase user).

--stop

The `--stop` command stops the current process, either encrypting or decrypting a disk.

The usage format is:

```
pgpwde --stop --disk <number>
```

Where:

- `--stop` specifies the current process is to be stopped.
- `--disk` specifies the disk to which the operation applies.
- `<number>` is the disk number on the system.

Example:

- `pgpwde --stop --disk 0`

  This example shows the encryption or decryption process on disk 0 being stopped.

13 Disk Management

The disk management commands are:

- `--auth`: Lets you authenticate to an encrypted disk.
- `--instrument`: Installs PGP WDE configuration information on specified disk.
- `--uninstrument`: Removes WDE configuration from specified disk.

In This Chapter

--auth

The `--auth` command lets you authenticate to an encrypted disk.

The usage format is:

```
pgpwde --auth --disk <number> --passphrase <phrase>
```

Where:

- `--auth` specifies you are authenticating to an encrypted disk.
- `--disk` specifies the disk to which the operation applies.
- `<number>` is the disk number on the system.
- `--passphrase` specifies the passphrase for an operation.
- `<phrase>` is the passphrase of an authorized user on the disk.

Example:

- `pgpwde --auth --disk 0 --passphrase 'Sam&Gamgee44'`

  This example shows a user on an encrypted disk authenticating to the boot disk, disk 0.

--instrument

The `--instrument` command replaces the Linux MBR with the PGPMBR.

Instrumenting the disk or partition is the first step in the process of securing a disk; it is followed by adding a passphrase user and then encrypting the disk. These three actions can be done individually, in that order, or all at once using the `--secure` command.

The usage format is:

```
pgpwde --instrument --disk <number>
```

Where:

- `--instrument` specifies that a disk or partition is to be instrumented.
- `--disk` specifies the disk to which the operation applies.
- `<number>` is the disk number on the system.

Example:

- `pgpwde --instrument --disk 0`

  This example shows a boot disk being instrumented.

--uninstrument

The `--uninstrument` command replaces the PGPMBR with the original (saved) Linux MBR. The removes the requirement to authenticate at the PGP BootGuard screen when starting the system.

Uninstrumenting a disk is normally done as part of the decryption process, so this command is not normally used on its own.

Caution: You can only uninstrument a disk that has been instrumented but nothing else. You cannot uninstrument an encrypted disk.

The usage format is:

```
pgpwde --uninstrument --disk <number>
```

Where:

- `--uninstrument` specifies specifies that a disk or partition is to be uninstrumented.
- `--disk` specifies the disk to which the operation applies.
- `<number>` is the disk number on the system.

Example:

- `pgpwde --uninstrument --disk 0`

  This example shows a boot disk being uninstrumented.

14 User Management Commands

The user management commands are:

- `--add-user`: Adds user to disk or group.
- `--change-passphrase`: Changes passphrase of specified user or group.
- `--change-userdomain`: Changes authentication domain of specified user or group.
- `--list-user`: Lists authorized users on an encrypted disk.
- `--remove-user`: Removes user from specified disk or group.
- `--verify-user`: Verifies passphrase of user or group.

In This Chapter

--add-user

The `--add-user` command adds an authorized user to the encrypted disk.

The usage format is:

```
pgpwde --add-user --disk <number> --domain-name
<domain> --passphrase <phrase> --username <user> --
admin-passphrase <pass> | --recovery-token <string>
```

Where:

- `--add-user` specifies that you are adding an authorized user to a disk.
- `--disk` specifies the disk to which the operation applies.
- `<number>` is the disk number on the system.

- `--username` specifies a username for an operation.
- `<user>` is the username of the user being added.
- `--domain-name` specifies the name of the domain to which the user authenticates. The default is the login domain.
- `<domain>` is the domain to which the user authenticates.
- `--passphrase` specifies the passphrase for an operation.
- `<pass>` is the passphrase the user being added will use to authenticate.
- `--username` specifies a username for an operation.
- `<user>` is the username of the user being added.
- `--admin-passphrase` specifies that the passphrase of an authorized user on the encrypted disk will be used to authenticate the adding of the new user account.
- `<phrase>` is the passphrase of an authorized user on the disk.
- `--recovery-token` specifies that the disk's recovery token (WDRT) will be used for authentication.
- `<string>` is the WDRT string.

Example:

-
  ```
  pgpwde --add-user --disk 0 --username "Alice Cameron" -
  -passphrase 'Frodo@Baggins22' --admin-passphrase
  'Sam&Gamgee44'
  ```

  This example shows a new passphrase user, Alice Cameron, being added to a boot disk with a passphrase of Frodo@Baggins22. The passphrase (Sam&Gamgee44) of an existing user on the disk is used to authenticate.

-
  ```
  pgpwde --add-user --disk 0 --username "Alice Cameron" -
  -domain EXAMPLECORP --passphrase 'Frodo@Baggins22' --
  admin-passphrase 'Sam&Gamgee44'
  ```

  This example shows a new user, in domain EXAMPLECORP, being added to a boot disk.

--change-passphrase

The `--change-passphrase` command lets you change the passphrase of a passphrase user on an encrypted disk.

The usage format is:

```
pgpwde --change-passphrase --disk <number> --username
<user> --new-passphrase <newpass> --passphrase <phrase>
```

Where:

- `--change-passphrase` specifies that you are changing the passphrase of a passphrase user.
- `--disk` specifies the disk to which the operation applies.
- `<number>` is the disk number on the system.
- `--username` specifies the existing user whose passphrase is being changed.
- `<user>` is the username of the existing user whose passphrase is being changed.
- `--new-passphrase` specifies that you are changing an existing passphrase to a new passphrase.
- `<newpass>` is the text of the new passphrase.
- `--passphrase` specifies the existing passphrase.
- `<phrase>` is the passphrase that is being changed.

Example:

-
  ```
  pgpwde --change-passphrase --disk 0 --username "Alice
  Cameron" --new-passphrase 'Sam&Gamgee44' --passphrase
  'Frodo@Baggins22'
  ```

  This example shows an existing passphrase user on an encrypted disk changing their passphrase.

--change-userdomain

The `--change-userdomain` command lets you change the user domain to which an authorized user authenticates.

This command is useful for organizations going through a domain migration.

The usage format is:

```
pgpwde --change-userdomain --disk <number> --new-domain
<domain> --username <user>
```

Where:

- `--change-userdomain` specifies that you are changing the user domain to which an authorized user on the drive authenticates.
- `--disk` specifies the disk to which the operation applies.
- `<number>` is the disk number on the system.
- `--new-domain` specifies the new domain to which the user will authenticate.
- `<domain>` is the name of the new authentication domain.

- `--username` specifies a username for the operation.
- `<user>` is the username of an existing user who is being removed.

Example:

-
  ```
  pgpwde --change-userdomain --disk 0 --new-domain
  EXAMPLECORP --username "Alice Cameron"
  ```

  This example shows the authentication domain of user Alice Cameron being changed to `EXAMPLECORP`.

--list-user

The `--list-user` command lists those users who are authorized users on the specified encrypted disk.

The usage format is:

```
pgpwde --list-user --disk <number>
```

Where:

- `--list-user` specifies that you are listing authorized users on a disk.
- `--disk` specifies the disk to which the operation applies.
- `<number>` is the disk number on the system.

Example:

-
  ```
  pgpwde --list-user --disk 0
  ```

  ```
  Total of 1 users:

      User 0: Name: Alice Cameron Type: Symmetric domain:
  EXAMPLECORP

  System Record Information:

     Serial Number: 1

           Disk ID: EXAMPLECORP.MSHOME.Alice Cameron.

         Disk UUID: 32eca196-7d16-4f83-9159-f7228af85594

        Group UUID: 32eca196-7d16-4f83-9159-f7228af85594

  Request sent to List users on disk was successful
  ```

  This example shows the users who can authenticate to the specified boot disk.

--remove-user

The `--remove-user` command removes a user who is currently authorized on the encrypted disk.

The usage format is:

```
pgpwde --remove-user --disk <number> --username <user>
--admin-passphrase <pass>
```

Where:

- `--remove-user` specifies that you are removing an authorized user on the disk.
- `--disk` specifies the disk to which the operation applies.
- `<number>` is the disk number on the system.
- `--username` specifies a username for the operation.
- `<user>` is the username of an existing user who is being removed.
- `--admin-passphrase` specifies that the passphrase of an authorized user on the encrypted disk will be used to authenticate the removal of the user.
- `<phrase>` is the passphrase of an authorized user on the disk.

Example:

-
  ```
  pgpwde --remove-user --disk 0 --username "Alice
  Cameron" --admin-passphrase 'Sam&Gamgee44'
  ```

  This example shows user Alice Cameron being removed from the boot disk.

--verify-user

The `--verify-user` command verifies the passphrase of a user who is an authorized user on an encrypted disk.

The usage format is:

```
pgpwde --verify-user --disk <number> --passphrase
<phrase> --username <user> | --keyid <keyid>
```

Where:

- `--verify-user` specifies that you are verifying the passphrase of an authorized user.
- `--disk` specifies to which disk on the system the information applies.
- `<number>` is the disk number on the system.

- `--passphrase` specifies the passphrase for an operation.
- `<phrase>` is the passphrase of an authorized user on the disk.
- `--username` specifies a username for an operation.
- `<user>` is the username of an authorized user account on the disk.
- `--keyid` specifies a user by key ID for an operation.
- `<keyid>` is the key ID of an authorized user on the disk.

Example:

-
  ```
  pgpwde --verify-user --disk 0 --passphrase
  'Frodo@Baggins44' --username "Alice Cameron"
  ```

  ```
  Successfully verified user Alice Cameron
  ```

  This example shows passphrase user Alice Cameron's passphrase being verified via her username.

-
  ```
  pgpwde --verify-user --disk 0 --passphrase
  'Frodo@Baggins44' --keyid 0x12345678
  ```

  ```
  Successfully verified user Alice Cameron
  ```

  This example shows PGP key user Alice Cameron's passphrase being verified via the key ID of her PGP key.

15 PGP BootGuard Customization Commands

PGP Whole Disk Encryption for Linux includes commands for modifying the default PGP BootGuard screen.

The PGP BootGuard customization commands are:

- `--set-background`: Lets you specify a custom PGP BootGuard screen background.
- `--set-language`: Lets you specify a language for the PGP BootGuard display and keyboard.
- `--set-sound`: Enables or disables audio prompts on the PGP BootGuard screen.
- `--set-start`: Lets you specify a custom PGP BootGuard startup screen background.
- `--set-text`: Lets you specify a text message for the PGP BootGuard authentication screen.

In This Chapter

--set-background

The `--set-background` command lets you specify a custom background image for the PGP BootGuard authentication screen.

Custom background images must be created according to the following specifications:

- XPM files only.
- Image size of 640 by 480.

- Palette of 15 colors only, including black (one color is reserved for fonts). You do not have to use all 15 colors in the image.
- 8-bit RGB only (cannot be 16-bit RGB). You can verify you are using 8 bit by looking at the XPM header using a text editor: 8-bit values appear as #285A83 (one hex triplet), 16-bit values appears as #28285A5A8383 (two hex triplets).

Note: If you specify an image that does not meet these requirements, a default text-only screen will be used.

Graphics applications that support the XPM file format include Graphic Converter on Mac OS X, GIMP on Mac OS X/FreeBSD and UNIX/LINUX, and the Convert command on Linux.

The new background image will display when the PGP BootGuard authentication screen next appears.

The usage format is:

```
pgpwde --set-background --disk <number> --image <file>
```

Where:

- `--disk` specifies the disk to which the operation applies.
- `<number>` is the disk number on the system.
- `--image` specifies the image file to use as the custom background.
- `<file>` is the name of the XPM file.

Example:

- `pgpwde --set-background --disk 0 --image "corplogo.xpm"`

  This example shows an image file, corplogo.xpm, being set as the background image for the PGP BootGuard authentication screen.

--set-language

The `--set-language` command lets you specify the languages that will be used by PGP BootGuard for display and for the keyboard.

You can specify one language and one display from the list of supported languages. You are not required to use the same language for both.

Options not specified are not changed. So if you specify a new language for text, the existing keyboard setting is not changed. The response to the `--set-language` command shows both the previous settings and the new settings, for both display and keyboard.

Changes will take effect on the next system startup.

The usage format is:

```
pgpwde --set-language --disk <number> --display <view>
--keyboard <type>
```

Where:

- `--disk` specifies the disk to which the operation applies.
- `<number>` is the disk number on the system.
- `--display` specifies the language to be used for viewing.
- `<view>` is desired language ID for the display: **default** (keep existing language), **de**, **en**, **es**, **fr**, or **jp**.
- `--keyboard` specifies the language to be used for typing text.
- `<type>` is the desired language for the keyboard: **default** (keep existing language), **de**, **en**, **en-gb**, **es**, **fr**, or **jp**.

Example:

- `pgpwde --set-language --disk 0 --display jp --keyboard jp`

  ```
  Boot language is set to Keyboard=en   Display=en

  Boot language now set to Keyboard=jp  Display=en
  ```

  This example shows Japanese being specified for both display and keyboard in PGP BootGuard.

--set-sound

The `--set-sound` command lets you enable or disable the use of audio clues for actions that occur during the PGP BootGuard authentication process. Audio clues are disabled by default.

Audio clues can help vision-impaired users more easily navigate the PGP BootGuard authentication process.

When enabled, the system will play audible tone combinations during the PGP BootGuard authentication process. Each tone combination begins with a middle sound and is followed by either a higher tone, another middle tone, or a lower tone.

The three combinations are:

- **Ready for passphrase/pin entry:** When the system is first ready for passphrase/pin entry, the middle-middle tone combination plays.
- **Successful authentication:** If the authentication attempt was successful, the middle-high tone combination plays. The system then continues booting.

- **Unsuccessful authentication:** If the authentication attempt was unsuccessful, the middle-low tone combination plays. The PGP BootGuard authentication screen displays and the passphrase field is cleared for another authentication attempt.

The tone combinations cannot be customized; you can only decide whether to enable audio clues or disable them.

Changes will take effect on the next system startup.

The usage format is:

```
pgpwde --set-sound --disk <number> --beep | --no-beep
```

Where:

- `--disk` specifies the disk to which the operation applies.
- `<number>` is the disk number on the system.
- `--beep` enables audio clues.
- `--no-beep` disables audio clues.

Example:

-
  ```
  pgpwde --set-sound --disk 0 --beep
  Accessibility Sounds set to [ON]
  ```

  This example shows audio clues being enabled.

--set-start

The `--set-start` command lets you display a custom startup image for PGP BootGuard that appears *before* the authentication screen. Press any key to make the startup screen disappear.

Custom startup images must be created according to the following specifications:

- XPM files only.
- Image size of 640 by 480.
- Palette of 15 colors only, including black (one color is reserved for fonts). You do not have to use all 15 colors in the image.
- 8-bit RGB only (cannot be 16-bit RGB). You can verify you are using 8 bit by looking at the XPM header using a text editor: 8-bit values appear as #285A83 (one hex triplet), 16-bit values appears as #28285A5A8383 (two hex triplets).

Graphics applications that support the XPM file format include Graphic Converter on Mac OS X, GIMP on Mac OS X/FreeBSD and UNIX/LINUX, and the Convert command on Linux.

The new startup image will display on the next system startup (unless bBoot bypass is used).

The usage format is:

```
pgpwde --set-start --disk <number> --image <file>
```

Where:

- `--disk` specifies the disk to which the operation applies.
- `<number>` is the disk number on the system.
- `--image` specifies the image file to use as the startup screen.
- `<file>` is the name of the XPM file.

Example:

- `pgpwde --set-start --disk 0 --image "corpsplash.xpm"`

  This example shows an image file, corpsplash.xpm, being set as the PGP BootGuard startup image.

--set-text

The `--set-text` command lets you specify text that will display when the PGP BootGuard screen appears.

You can disable the display of text by entering no text where the message would go.

You can enter one line of text, up to 80 characters (including spaces). The default text is: "Forgot your passphrase? Please contact your IT department or Security Administrator."

Note: Text must go in quotation marks or only the text up to the first space will display. The quotation marks do not display.

Changes will take effect on the next system startup.

The usage format is:

```
pgpwde --set-text --disk <number> --message <text>
```

Where:

- `--disk` specifies the disk to which the operation applies.
- `<number>` is the disk number on the system.
- `--message` specifies new text for the PGP BootGuard screen.
- `<text>` is the text you want to display. If left empty, no text will display.

Examples:

-
  ```
  pgpwde --set-text --disk 0 --message "You must change
  your login passphrase monthly."
  ```

  This example shows a new text message for the PGP BootGuard screen.

-
  ```
  pgpwde --set-text --disk 0 --message
  ```

  This example shows the display of text for the PGP BootGuard screen being disabled.

16 Recovery Token Commands

In PGP Universal-managed environments with the appropriate policy, Whole Disk Recovery Tokens (WDRTs) are created automatically when a disk, partition, or removable disk is whole disk encrypted. They are sent to the PGP Universal Server managing security for the disk or partition when they are created.

WDRTs can be used to access the disk or partition in case the passphrase or authentication token is lost.

Once a WDRT is used, it cannot be used again. A new WDRT must be generated for the system. All new WDRTs are also automatically sent to the PGP Universal Server managing the disk when the new WDRT is created.

Because the first WDRT for a system is created automatically, the only command related to WDRTs is to create a new WDRT.

The recovery token commands are:

- `--new-wdrt`: Creates a new WDRT after use.

In This Chapter

--new-wdrt

The `--new-wdrt` command creates a new WDRT (recovery token) when the previous WDRT has been used.

The usage format is:

```
pgpwde --new-wdrt --disk <number> --admin-authorization
| --admin-passphrase <phrase> --recovery-token <string>
```

Where:

- –new-wdrt specifies the creation of a new WDRT.
- `--disk` specifies the disk to which the operation applies.
- `<number>` is the disk number on the system.

- `--admin-authorization` specifies that the command is being performed by a member of the WDE-ADMIN Active Directory group.
- `--admin-passphrase` specifies that the passphrase of an authorized user on the encrypted disk will be used to authenticate the adding of the new user account.
- `<phrase>` is the passphrase of an authorized user on the disk.
- `--recovery-token` specifies that a recovery token (WDRT) will be created to replace the used one.
- `<string>` is the WDRT string.

Example:

-
  ```
  pgpwde --new-wdrt --disk 0 --admin-passphrase
  'bilbo@baggins44' --recovery-token 'Gandalf-
  Bilbo+Merry=OneRing'
  ```

  This example shows a new WDRT being created.

17 Local Self Recovery

Local self recovery lets you authenticate to PGP BootGuard even if you have forgotten your passphrase.

> **Note:** Local self recovery only works if you configure it *before* you lose your passphrase; PGP Corporation recommends configuring it immediately after licensing PGP Whole Disk Encryption for Linux if you plan on using it.

When you configure local self recovery, you create five security questions; three must be answered correctly to authenticate to PGP BootGuard.

> **Note:** If you are using PGP Whole Disk Encryption for Linux in a PGP Universal Server-managed environment, your PGP Universal Server administrator may have disabled the option for local self recovery. Your administrator may also have specified that local self recovery be configured during enrollment. In this case, you are prompted to enter the security questions as as you set up PGP Whole Disk Encryption for Linux.

The local self recovery commands are:

- `--recovery-configure`: Configures the local self recovery feature.
- `--recovery-questions`: Displays local self recovery questions.
- `--recovery-verify`: Verifies existing local self recovery questions and answers.
- `--recovery-remove`: Removes existing local self recovery questions and answers.
- `--recovery-change-passphrase`: Changes a lost passphrase.

In This Chapter

--recovery-configure

The `--recovery-configure` command configures local self recovery.

You can configure the required five questions and answers in either of two ways:

- text files: you create two text files; one text file with five questions, each on separate lines, and a second text file with five answers to those questions, again each on a separate line.
- interactively (`--interactive)`: You will be prompted for five questions and their corresponding answers.

You can also use `--interactive` to have PGP Whole Disk Encryption for Linux interactively prompt for a passphrase. Simply do not enter a passphrase on the command line.

Note: Text files and `--interactive` are mutually exclusive. Use one method or the other.

You will need to be able to correctly answer three of the five questions if you forget your passphrase and need to authenticate to PGP BootGuard using `--recovery-verify`.

The usage format is:

```
pgpwde --recovery-configure --user <username> --
passphrase <phrase> [--disk <disknumber>] [--questions-
file <questions>] [--answers-file <answers>]
[--interactive]
```

Where:

- –recovery-configure specifies that you are configuring local self recovery.
- `--user` specifies which user account is being used.
- `<username>` is the name of the user account.
- `--passphrase` specifies the passphrase for an operation.
- `<phrase>` is the passphrase for specified user account.
- `--disk` specifies disk on the system for which local self recovery is being configured.
- `<disknumber>` is the disk number on the system. Disk 0, the boot disk, is the default.
- `--questions-file` specifies the five questions will be in a text file.
- `<questions>` is the path to the text file with the five questions, each on its own line.
- `--answers-file` specifies the five answers will be in a text file.

- `<answers>` is the path to the text file with the five answers, each on its own line.
- `--interactive` specifies you will be prompted for the five questions and answers.

Examples:

-
  ```
  pgpwde --recovery-configure --user "Alice Cameron"
  --passphrase 'bilbo#baggins+Frodo' --disk 0
  --interactive
  ```

  This example shows local self recovery being configured for user Alice Cameron using interactive questions and answers.

-
  ```
  pgpwde --recovery-configure --user "Alice Cameron"
  --passphrase 'bilbo#baggins+Frodo' --disk 0
  --questions-file "/home/user/docs/questions.txt"
  --answers-file "/home/user/docs/answers.txt"
  ```

  This example shows local self recovery being configured for user Alice Cameron with the five questions and answers in the specified text files.

--recovery-questions

The `--recovery-questions` command displays *configured* local self recovery questions.

Note: `--recovery-questions` only shows existing questions. You cannot modify or add questions using this command.

The usage format is:

```
pgpwde --recovery-questions --user <username> [--disk
<disknumber>]
```

Where:

- `--recovery-questions` specifies that you are configuring local self recovery.
- `--user` specifies which user account is being used.
- `<username>` is the name of the user account.
- `--disk` specifies disk on the system for which local self recovery is being configured.
- `<disknumber>` is the disk number on the system. Disk 0, the boot disk, is the default.

Example:

-
  ```
  pgpwde --recovery-questions --user "Alice Cameron"
  --disk 0
  ```

This example displays the configured local self recovery questions for user Alice Cameron.

--recovery-verify

The `--recovery-verify` command lets you verify the configured local self recovery questions and answers. You can answer the five questions either using a text file or interactively.

> **Note:** You cannot modify the local self recovery questions using `--recovery-verify`.

To authenticate to PGP BootGuard using the configured local self recovery questions and answers, see Recovering a Lost Passphrase.

The usage format is:

```
pgpwde --recovery-verify --user <username> [--disk
<disknumber>] [--answers-file <answers>]
[--interactive]
```

Where:

- `--recovery-verify` specifies that you are verifying existing local self recovery questions and answers.
- `--user` specifies which user account is being used.
- `<username>` is the name of the user account.
- `--disk` specifies the disk on the system for which the command is being performed.
- `<disknumber>` is the disk number on the system. Disk 0, the boot disk, is the default.
- `--answers-file` specifies the five answers will be in a text file.
- `<answers>` is the path to the text file with the five answers, each on its own line.
- `--interactive` specifies you will be prompted for the five answers and questions.

Example:

-
  ```
  pgpwde --recovery-questions --user "Alice Cameron"
  --disk 0 --answers-file "/home/user/docs/answers.txt"
  ```

  This example shows user Alice Cameron verifying configured local self recovery questions and answers using the file answers.txt.

--recovery-remove

The `--recovery-remove` command removes *configured* local self recovery questions and answers.

The usage format is:

```
pgpwde --recovery-remove --user <username> --passphrase
<phrase> [--disk <disknumber>]
```

Where:

- `--recovery-remove` specifies that you are removing configured local self recovery questions and answers.
- `--user` specifies which user account is being used.
- `<username>` is the name of the user account.
- `--passphrase` specifies the passphrase for an operation.
- `<phrase>` is the passphrase for specified user account.
- `--disk` specifies disk on the system for which local self recovery is being removed.
- `<disknumber>` is the disk number on the system. Disk 0, the boot disk, is the default.

Example:

-
  ```
  pgpwde --recovery-remove --user "Alice Cameron"
  --passphrase 'bilbo#baggins+Frodo' --disk 0
  ```

  This example removes configured local self recovery questions and answers for user Alice Cameron.

--recovery-change-passphrase

The `--recovery-change-passphrase` command lets you create a new passphrase when you have forgotten your existing passphrase and authenticated to PGP BootGuard using local self recovery.

Note: PGP Corporation recommends creating a new passphrase as soon as you authenticate to PGP BootGuard after forgetting your passphrase and authenticating using local self recovery.

The usage format is:

```
pgpwde --recovery-change-passphrase --user <username>
[--disk <disknumber>] --new-passphrase <newpass>
[--answers-file <answers>]
```

Where:

- `--recovery-verify` specifies that you are authenticating to PGP BootGuard.
- `--user` specifies which user account is being used.
- `<username>` is the name of the user account.
- `--disk` specifies the disk on the system for which the command is being performed.
- `<disknumber>` is the disk number on the system. Disk 0, the boot disk, is the default.
- `--new-passphrase` specifies the five answers will be in a text file.
- `<newpass>` is the path to the text file with the five answers, each on its own line.
- `--answers-file` specifies the five answers will be in a text file.
- `<answers>` is the path to the text file with the five answers, each on its own line.

Example:

-
  ```
  pgpwde --recovery-change-passphrase --user "Alice
  Cameron" --disk 0 --new-passphrase
  'Bilbo%Baggins$Underhill' --answers-file
  "/home/user/docs/answers.txt"
  ```

  This example shows user Alice Cameron authenticating to PGP BootGuard using the answers in the file answers.txt.

Authenticating if you Have Forgotten Your Passphrase

If you have forgotten your passphrase and cannot authenticate to the PGP BootGuard screen, you can authenticate using local self recovery if you have previously configured it.

Note: Local self recovery *must* be configured in advance.

See Local Self Recovery for information about using the command line or a text file to configure the local self recovery questions.

To authenticate at the PGP BootGuard screen using local self recovery

1. On the PGP BootGuard screen, use the arrow keys to select **Forgot Passphrase** in the lower right corner, then press **Enter**. A new screen appears, showing the first local self recovery question.
2. Enter the answer to the first question, then press **Enter**. The second question appears.

3 Enter the answer to the second question, then press **Enter**. The third question appears.

4 Enter the answer to the third question, then press **Enter**. The fourth question appears.

5 Enter the answer to the fourth question, then press **Enter**. The fifth and last question appears.

6 Enter the answer to the fifth question, then press **Enter**.

If you entered three or more of the questions correctly, the PGP BootGuard screen goes away and the system boots normally.

If you did not enter three or more questions correctly, you are given another chance.

If you subsequently remember your original passphrase, you can continue using it. Using local self recovery does not remove your passphrase.

If you do not believe you will ever remember your original passphrase, you can change your passphrase after authenticating to PGP BootGuard using the `--recovery-change-passphrase` command. This means that you do not have to continue using the local self recovery questions to authenticate to PGP BootGuard. Using this command does remove your original passphrase, so it will not work if you remember it later.

18 Options

This section lists and describes the options you can use with PGP Whole Disk Encryption for Linux.

In This Chapter

Overview

PGP Whole Disk Encryption for Linux supports the following options:

- `--admin-authorization`: Specifies that the command is authorized by member of the WDE-ADMIN Active Directory group.
- `--admin-passphrase`: Specifies the passphrase of an existing PGP WDE user.
- `--all`: Specifies the use of partition mode encryption on **all** partitions.
- `--auto-start`: Starts encryption immediately.
- `--base-disk`: Specifies the disk number of the original group.
- `--beep`: Enables beep when PGP BootGuard screen appears.
- `--count`: Specifies the number of bypass restarts being configured.
- `--dedicated-mode`: Specifies that dedicated mode be used.
- `--disk (-d)`: Specifies the number of the target disk. Zero (0) is boot disk.
- `--display`: Specifies the PGP BootGuard display language.
- `--domain-name`: Specifies the user authentication domain.
- `--fast-mode`: Specifies that fast mode be used.
- `--image`: Specifies an image file to be used.
- `--keyboard`: Specifies the PGP BootGuard keyboard language.
- `--keyid`: Specifies the key ID of a PGP key.
- `--license-email:` Specifies an email address for the license holder.
- `--license-name:` Specifies the person to whom PGP Whole Disk Encryption for Linux is licensed.
- `--license-number:` Specifies a valid license number for PGP Whole Disk Encryption for Linux.
- `--license-organization:` Specifies the organization of the license holder.
- `--message`: Specifies custom message for PGP BootGuard screen.
- `--new-domain`: Specifies a new domain for a user.
- `--new-passphrase`: Specifies a new passphrase for an existing user.
- `--no-beep`: Disables beep when PGP BootGuard screen appears.
- `--partition`: Specifies a partition for an operation.
- `--passphrase (-p)`: Specifies a passphrase for an operation.
- `--recovery-token`: Specifies a whole disk recovery token.
- `--safe-mode`: Specifies that safe mode be used.
- `--username (-u)`: Specifies a username for an operation.

"Secure" Options

The descriptions of some options in PGP Whole Disk Encryption for Linux mention that they are "secure," as in "This option is not secure". In this context, "secure" means that the option’s argument is saved in non-pageable memory (when that option is available to applications). Options that are not "secure" are saved in normal system memory.

--admin-authorization

Specifies that the operation is authorized by a member of the WDE-ADMIN Active Directory group. In other words, by an administrator of PGP WDE clients in a PGP Universal-managed environment.

No passphrase is required on the command line when using this option. Instead, the administrator will be authenticated against the WDE-ADMIN group when the option is used.

This option can be shortened to `--aa`.

Example:

-
  ```
  pgpwde --add-user --disk 0 --username "Alice Cameron"
  --passphrase 'Frodo@Baggins22' --admin-authorization
  --recovery-token 'Gandalf-Bilbo+Merry=OneRing'
  ```

  This example shows a new passphrase user being added to a boot disk with a recovery token by a member of the WDE-ADMIN Active Directory group.

--admin-passphrase

Specifies that the passphrase being used is that of an authorized user of the encrypted disk.

This option can be shortened to `--ap`.

Example:

-
  ```
  pgpwde --add-user --disk 0 --username "Alice Cameron"
  --passphrase 'Frodo@Baggins22' --admin-passphrase
  'Sam&Gamgee44'
  ```

This example shows a new passphrase user being added to a boot disk. The passphrase of an existing user on the disk is used to authenticate.

--all

Specifies that all partitions should be encrypted.

Example:

- `pgpwde --encrypt --disk 0 --passphrase 'Frodo*1*Baggins'` **--all**

  This example shows encryption of a boot disk being started. All partitions are to be encrypted.

--answers-file

Specifies the path to a text file with five answers, each on a new line of the file.

Example:

- `pgpwde --recovery-configure --user "Alice Cameron" --passphrase 'bilbo#baggins+Frodo' --disk 0 --questions-file "/home/user/docs/questions.txt"` **--answers-file "/home/user/docs/answers.txt"**

  This example shows local self recovery being configured for user Alice Cameron with the five questions and answers in the specified text files.

--auto-start

Specifies whether or not encryption should begin immediately. Options are `Yes` or `No`. The default is `No`.

Example:

- `pgpwde --verify-user` **--auto-start Yes** `--base-disk 0 --disk 1 --passphrase 'Sam&Gamgee44' --username "Jose Medina"`

  This example shows disk 1 on the system being added to the encrypted disk group. Encryption will begin immediately.

--beep

Specifies that audio clues for actions that occur during the PGP BootGuard authentication process should be enabled.

The default is audio clues are disabled.

Example:

- `pgpwde --set-sound --disk 0` **--beep**

  ```
  Accessibility Sounds set to [ON]
  ```

  This example shows audio clues being enabled.

--count

Specifies the number of bypass restarts being configured for the boot disk on a system.

Only works with the `--add-bypass` command.

Valid values for `--count` are 0 through 4,294,967,295.

Setting `--count` to 0 disables the boot bypass feature on the system.

In a PGP Universal-managed environment, a preference constrains what values are valid for `--count` on the command line; you cannot set a value on the command line that is higher than the value set in the preference.

--dedicated-mode

Specifies that Dedicated Mode should be used for the encryption process. Dedicated Mode uses maximum computer power to encrypt faster; your system is less responsive during encryption.

Example:

- `pgpwde --encrypt --disk 0 --passphrase 'Frodo*1*Baggins22'` **--dedicated-mode**

  This example shows encryption of a boot disk being started using Dedicated Mode.

--disk (-d)

Specifies the disk to which the operation applies.

Example:

`pgpwde --info` **--disk 0**

This example shows general information being requested for disk 0.

--display

Specifies the display language for PGP BootGuard.

Example:

- `pgpwde --set-language --disk 0` **--display jp** `--keyboard jp`

  ```
  Boot language is set to Keyboard=en   Display=en
  Boot language now set to Keyboard=jp  Display=en
  ```

  This example shows Japanese being specified for both display and keyboard in PGP BootGuard.

--domain-name

Specifies an authentication domain. The default is the login domain.

Example:

- `pgpwde --add-user --disk 0 --username "Alice Cameron"` **--domain EXAMPLECORP** `--passphrase 'Frodo@Baggins22' --admin-passphrase 'Sam&Gamgee44'`

  This example shows a new user, in domain EXAMPLECORP, being added to a boot disk.

--fast-mode

Specifies that Fast Mode should be used for the encryption process. Fast mode skips unused sectors, so encryption of the disk is faster.

Example:

- `pgpwde --encrypt --disk 0 --passphrase 'Frodo*1*Baggins'` **--fast-mode**

  This example shows encryption of a boot disk being started using fast mode.

--image

Specifies an XPM file to use for an operation.

Example:

- `pgpwde --set-background --disk 0` **--image "corplogo.xpm"**

  This example shows an image file, corplogo.xpm, being set as the background image for the PGP BootGuard authentication screen.

--interactive

Specifies that questions and answers should be asked and answered interactively, as opposed to coming from text files.

Example:

- `pgpwde --recovery-questions --user "Alice Cameron" --disk 0` **--interactive**

This example shows user Alice Cameron verifying configured local self recovery questions and answers interactively.

--keyboard

Specifies the keyboard language for PGP BootGuard.

Example:

- `pgpwde --set-language --disk 0 --display jp` **--keyboard jp**

  ```
  Boot language is set to Keyboard=en   Display=en
  ```

  ```
  Boot language now set to Keyboard=jp  Display=en
  ```

  This example shows Japanese being specified for both display and keyboard in PGP BootGuard.

--keyid

Specifies the key ID of a PGP key.

Example:

- `pgpwde --verify-user --disk 0 --passphrase 'Frodo@Baggins44'` **--keyid 0x12345678**

```
Successfully verified user Alice Cameron
```

This example shows PGP key user Alice Cameron's passphrase being verified via the key ID of her PGP key.

--license-email

Specifies an email address for the license holder.

Example:

-
  ```
  pgpwde --license-authorize --license-name "Alice
  Cameron"
  --license-number "aaaaa-bbbbb-ccccc-ddddd-eeeee-fff"
  --license-email "acameron@example.com"
  --license-organization "Example Corporation"
  ```

  This example shows the license holder's email address being entered during licensing.

--license-name

Specifies the person to whom PGP Whole Disk Encryption for Linux is licensed.

Example:

-
  ```
  pgpwde --license-authorize --license-name "Alice Cameron"
  --license-number "aaaaa-bbbbb-ccccc-ddddd-eeeee-fff"
  --license-email "acameron@example.com"
  --license-organization "Example Corporation"
  ```

  This example shows the license holder's name being entered during licensing.

--license-number

Specifies a valid license number for PGP Whole Disk Encryption for Linux.

Example:

-
  ```
  pgpwde --license-authorize --license-name "Alice
  Cameron"
  --license-number "aaaaa-bbbbb-ccccc-ddddd-eeeee-fff"
  --license-email "acameron@example.com"
  --license-organization "Example Corporation"
  ```

  This example shows the license number being entered during licensing.

--license-organization

Specifies the organization of the license holder.

Example:

- `pgpwde --license-authorize --license-name "Alice Cameron" --license-number "aaaaa-bbbbb-ccccc-ddddd-eeeee-fff" --license-email "acameron@example.com"` **--license-organization "Example Corporation"**

  This example shows the organization of the license holder being entered during licensing.

--message

Specifies text for the PGP BootGuard screen.

Example:

- `pgpwde --set-text --disk 0` **--message 'You must change your login passphrase monthly.'**

  ```
  Custom message Updated
  ```

  ```
  Set custom authentication screen text completed
  ```

  This example shows a new text message for the PGP BootGuard screen.

--new-domain

Specifies a new authentication domain for an authorized user.

Example:

- `pgpwde --change-userdomain --disk 0` **--new-domain EXAMPLECORP** `--username "Alice Cameron"`

  This example shows the authentication domain of user Alice Cameron being changed to `EXAMPLECORP`.

--new-passphrase

Specifies the new passphrase when a passphrase user is changing their passphrase.

Example:

-
  ```
  pgpwde --change-passphrase --disk 0 --username "Alice
  Cameron" --new-passphrase 'Sam&Gamgee44' --passphrase
  'Frodo@Baggins22'
  ```

  This example shows an existing passphrase user on an encrypted disk changing their passphrase.

--no-beep

Specifies that audio clues for actions that occur during the PGP BootGuard authentication process should be disabled.

The default is audio clues are disabled.

Example:

-
  ```
  pgpwde --set-sound --disk 0 --no-beep
  ```

  ```
  Accessibility Sounds set to [OFF]
  ```

  This example shows audio clues being enabled.

--partition

Specifies that only the listed partition should be encrypted.

Example:

-
  ```
  pgpwde --decrypt --disk 0 --passphrase
  'Frodo*1*Baggins22' --partition 3
  ```

  This example shows partition 3 on the boot disk being decrypted.

--passphrase (-p)

Specifies the passphrase of an authorized user on an encrypted disk.

Example:

-
  ```
  pgpwde --add-user --disk 0 --username "Alice Cameron"
  --passphrase 'Frodo@Baggins22' --admin-passphrase
  'Sam&Gamgee44'
  ```

  ```
  Add user completed
  ```

  This example shows a new passphrase user being added to a boot disk with a passphrase of Frodo@Baggins22. In this example, `--passphrase` is being used to specify the passphrase that the new user of the encrypted disk will use to access it.

--questions-file

Specifies the path to a text file with five questions, each on a new line of the file.

Example:

-
  ```
  pgpwde --recovery-configure --user "Alice Cameron"
  --passphrase 'bilbo#baggins+Frodo' --disk 0 --questions-file
  "/home/user/docs/questions.txt" --answers-file
  "/home/user/docs/answers.txt"
  ```

This example shows local self recovery being configured for user Alice Cameron with the five questions and answers in the specified text files.

--recovery-token

Specifies that a recovery token (WDRT) be created.

Example:

```
pgpwde --add-user --disk 0 --username "Alice Cameron"
--passphrase 'Frodo@Baggins22' --admin-passphrase
'Sam&Gamgee44' --recovery-token 'Gandalf-Bilbo+Merry=OneRing'
```

This example shows a new passphrase user being added to a boot disk with an associated recovery token.

--safe-mode

Specifies that Safe Mode should be used for the encryption process. Safe Mode allows encryption to be resumed without loss of data if power is lost during encryption; encryption takes longer.

Example:

- `pgpwde --encrypt --disk 0 --passphrase 'Frodo*1*Baggins22'` **--safe-mode**

  This example shows encryption of a boot disk being started using safe mode.

--username

Identifies an authorized user of an encrypted disk by their username.

Example:

- `pgpwde --change-passphrase --disk 0` **--username "Alice Cameron"** `--new-passphrase 'Sam&Gamgee44' --passphrase 'Frodo@Baggins22'`

  This example shows an existing passphrase user on an encrypted disk changing their passphrase. They are identified by their username.

A Quick Reference

This section lists and briefly describes all PGP Whole Disk Encryption for Linux commands and options.

In This Chapter

Commands

General

--help (-h)	Shows basic help information for PGP Whole Disk Encryption for Linux.
--version (-V)	Shows PGP Whole Disk Encryption for Linux version information.
--license-authorize	Licenses PGP Whole Disk Encryption for Linux.
--enroll	Enrolls PGP Whole Disk Encryption for Linux with a PGP Universal Server.
--check-enroll	Checks enrollment status of PGP Whole Disk Encryption for Linux.

Disk Information

--enum	Lists system disks and volumes.
--info	Lists general system disk information.
–show-config	Displays PGP BootGuard configuration information.
–status	Displays PGP WDE-related status of disk.

User Management

–add-user	Adds user to disk.
--change-passphrase	Changes passphrase of specified user.
--change-userdomain	Changes authentication domain of specified user.
--list-user	Lists authorized users on an encrypted disk.
--remove-user	Removes user from specified disk.
--verify-user	Verifies passphrase of user.

Disk Management

--auth	Authenticates to an encrypted disk.
--instrument	Installs WDE configuration information on specified disk.
--uninstrument	Removes WDE configuration from specified disk.

Disk Operation

--decrypt	Decrypts the specified disk.
--encrypt	Encrypts the specified disk.
--resume	Resumes halted encrypt or decrypt process.
--secure	Encrypts a disk to a specified user and passphrase.
–stop	Halts encrypt or decrypt process.

Boot Bypass Commands

–add-bypass	Sets disk for one-time authentication bypass.
–check-bypass	Checks disk to see if authentication bypass is set.
--remove-bypass	Removes authentication bypass from disk.

PGP BootGuard Customization Commands

--set-background	Sets custom PGP BootGuard screen background.
--set-language	Sets PGP BootGuard display and keyboard languages.
--set-sound	Sets PGP BootGuard audio prompt.
–set-start	Sets custom PGP BootGuard startup screen background.
--set-text	Sets PGP BootGuard authentication screen text message.

Recovery Token

--new-wdrt	Creates a new WDRT after use.

Local Self Recovery

--recovery-configure	Sets up the local self recovery feature.
–recovery-questions	Displays configured local self recovery questions.
–recovery-verify	Verifies configured local self recovery questions.
–recovery-remove	Removes configured local self recovery questions and answers.
–recovery-change-passphrase	Changes a user passphrase via local self recovery.

Options

The PGP Whole Disk Encryption for Linux options are:

- `--admin-authorization`: Specifies that the command is authorized by member of the WDE-ADMIN Active Directory group.
- `--admin-passphrase`: Specifies the passphrase of an existing PGP WDE user.
- `--all`: Specifies the use of partition mode encryption on all partitions.
- `--answers-file:` Specifies the path to a text file with five answers.
- `--auto-start`: Starts encryption immediately.
- `--beep`: Enables beep when PGP BootGuard screen appears.
- `--count`: Specifies the number of bypass restarts being configured.
- `--dedicated-mode`: Specifies that dedicated mode be used.
- `--disk (-d)`: Specifies the number of the target disk. Zero (0) is boot disk.
- `--display`: Specifies the PGP BootGuard display language.
- `--domain-name`: Specifies the user authentication domain.
- `--fast-mode`: Specifies that fast mode be used.
- `--image`: Specifies an image file to be used.
- `--interactive:` Specifies questions and answers be asked and answered interactively, not from text files.
- `--keyboard`: Specifies the PGP BootGuard keyboard language.
- `--keyid`: Specifies the key ID of a PGP key.
- `--license-email:` Specifies an email address for the license holder.
- `--license-name:` Specifies the person to whom PGP Whole Disk Encryption for Linux is licensed.
- `--license-number:` Specifies a valid license number for PGP Whole Disk Encryption for Linux.
- `--license-organization:` Specifies the organization of the license holder.
- `--message`: Specifies custom message for PGP BootGuard screen.
- `--new-domain`: Specifies a new domain for a user.
- `--new-passphrase`: Specifies a new passphrase for an existing user.
- `--no-beep`: Disables beep when PGP BootGuard screen appears.

- `--partition`: Specifies a partition for an operation.
- `--passphrase (-p)`: Specifies a passphrase for an operation.
- `--proxy-passphrase:` Specifies the passphrase of the specified user on the proxy server.
- `--proxy-server:` Specifies a proxy server to go through to license PGP Whole Disk Encryption for Linux.
- `--proxy-username:` Specifies a user on the proxy server.
- `--questions-file:` Specifies the path to a text file with five questions.
- `--recovery-token`: Specifies a whole disk recovery token.
- `--safe-mode`: Specifies that safe mode be used.
- `--username (-u)`: Specifies a username for an operation.

B Troubleshooting

This section describes how PGP Whole Disk Encryption for Linux can be used to troubleshoot problems you might encounter when whole disk encrypting drives.

In This Chapter

Overview

The troubleshooting tips in this appendix assume:

- PGP Whole Disk Encryption for Linux is correctly installed on the system.
- The software is licensed to support PGP Whole Disk Encryption for Linux.

Before troubleshoot problems with PGP Whole Disk Encryption for Linux, PGP Corporation recommends checking existing resources for information about the issue you are experiencing:

- The *PGP Whole Disk Encryption for Linux Release Notes* include the latest information available about PGP Whole Disk Encryption for Linux, including system requirements and known incompatibilities.
- The *PGP Desktop User's Guide* includes more information about how to prepare a drive for encryption, how to encrypt it, and how to use it after encryption.

Encryption Does Not Begin

While the vast majority of drives can be encrypted without a problem, on some occasions you may find a drive where the encryption process does not start.

Perform the following steps:

1 Review the *PGP Whole Disk Encryption for Linux Release Notes* for issues that could be blocking encryption.

  Potential issues include unsupported operating systems and software incompatibilities. If any issues are found, make the appropriate changes and then attempt encryption again.

If encryption still will not begin, you can use PGP Whole Disk Encryption for Linux to learn more information.

1 First, determine the boot drive on the system using the `--enum` command.

```
pgpwde --enum
```

  The response will be something like:

```
Total number of installed fixed/removable storage
device (excluding floppy and CDROM): 1

Disk 0 has 1 online volumes:

  volume C is on partition 2 with offset 80325

Enumerate disks completed
```

  This example shows that the system has one disk, Disk 0, which is drive letter C and is the boot disk. You now know:

  - The boot drive can be whole disk encrypted, as it is Disk 0. Only boot disks that are Disk 0 can be whole disk encrypted.
  - That Disk 0 is the boot disk (which you need to know for subsequent commands).

2 Next, check the status of the boot drive using the `--status` command.

```
pgpwde --status --disk 0

Disk disk 0 is not instrumented by bootguard.

Disk status completed
```

  This example shows the response for a disk that is not whole disk encrypted; that is, the disk is not instrumented by PGP BootGuard.

  If a disk is encrypted or even partially encrypted, the response would be something like:

```
pgpwde --status --disk 0

Disk disk 0 is instrumented by bootguard.

    Current key is valid.

Whole disk encrypted

   Total sectors: 192426569  highwater mark: 192426569

Disk status completed
```

This response or something similar would mean that the encryption process started but then stopped again. For information on dealing with a drive where encryption does not finish, refer to Encryption Does Not Finish.

If the problem continues, you will need to get further assistance.

- The PGP Support forums are user community forums hosted by PGP Corporation and monitored by PGP Corporation personnel. Check the PGP Whole Disk Encryption forums for more information.

  To access the PGP Support forums, please visit *PGP Support* (*http://forum.pgp.com*).

- The PGP Support Knowledge Base and PGP Technical Support may also be able to assist you with your issue.

  To access the PGP Support Knowledge Base or request PGP Technical Support, please visit *PGP Support Portal Web Site* (*https://support.pgp.com*). **Note that you may access portions of the PGP Support Knowledge Base without a support agreement; however, you must have a valid support agreement to request PGP Technical Support.**

Encryption Does Not Finish

Once encryption has started, most drives finish encryption normally. On some occasions, however, the encryption process may stop on its own. The cause is generally a problem with the drive being encrypted.

If the system being encrypted loses power during the process, encryption will automatically stop. Depending on whether or not you were using the Safe Mode option (`--safe-mode`), you have two options:

- If you were using Safe Mode, simply get the system back up and restart encryption. It should resume near the point where power was lost.
- If you were *not* using Safe Mode, get the system back up, decrypt the portion of the drive that was encrypted, and then restart encryption.

The best practice for a drive where encryption stopped automatically is to decrypt the partially encrypted drive, check it for problems, then start encryption again. Be sure to **fully decrypt** any drive on which encryption was started before checking it for problems.

Note: Refer to the *PGP Desktop User's Guide* for extensive information about preparing a drive for encryption.

If encryption stops before finishing (without losing power), perform the following steps:

1. Decrypt the portion of the drive that was encrypted.
2. When the drive is fully decrypted, check the status of the boot drive using the `--status` command.

```
pgpwde --status --disk 0

Disk disk 0 is not instrumented by bootguard.

Disk status completed
```

This example shows the response for a disk that has been fully decrypted.

If the response to the `--status` command shows the drive still partially encrypted, make sure the drive is fully decrypted.

3. Next, check the health of the drive; make the changes necessary to ensure the health of the drive.
4. Review the *PGP Whole Disk Encryption for Linux Release Notes* for issues that could be affecting encryption. If any applicable issues are found, make the appropriate changes.
5. When all changes have been made, reboot the system.
6. Begin the encryption process again.

If the problem continues, you will need to get further assistance:

- The PGP Support forums are user community forums hosted by PGP Corporation and monitored by PGP Corporation personnel. Check the PGP Whole Disk Encryption forums for more information.

  To access the PGP Support forums, please visit *PGP Support* (*http://forum.pgp.com*).

- The PGP Support Knowledge Base and PGP Technical Support may also be able to assist you with your issue.

  To access the PGP Support Knowledge Base or request PGP Technical Support, please visit *PGP Support Portal Web Site* (*https://support.pgp.com*). **Note that you may access portions of the PGP Support Knowledge Base without a support agreement; however, you must have a valid support agreement to request PGP Technical Support.**

Problems at PGP BootGuard

On rare occasions, a drive may successfully encrypt but PGP BootGuard may prevent access to the system.

Most cases involving problems at the PGP BootGuard screen involve entering the passphrase correctly.

It's easy to spot a problem involving entering your passphrase: you enter what you believe is the correct passphrase and press **Enter**; PGP BootGuard displays an error message instead of giving you access to your system.

If you cannot successfully enter your passphrase at the PGP BootGuard screen, perform the following steps:

1. Carefully re-enter your passphrase. You may have typed it incorrectly.

   To see the characters you are typing, press **Tab** then enter your passphrase.

2. Make sure **Caps Lock** is off, unless your passphrase is all capital letters.

3. Make sure you are using the correct keyboard layout. If the wrong keyboard layout is selected, you may inadvertently be typing the wrong characters.

   Select **Keyboard** on the main PGP BootGuard screen and press **Enter**. Available keyboard layouts are displayed; the selected keyboard layout is shown under the list. Select **Go Back** and press **Enter** to return to the main PGP BootGuard screen.

   Refer to the *PGP Desktop User's Guide* for more information about supported keyboard layouts.

4. If there are other configured users for the drive, try the passphrases of these users.

5. If you determine that you have forgotten your passphrase and you configured local self recovery, you can attempt to recover your passphrase.

6. If you are in an enterprise environment, contact your PGP Universal administrator for instructions.

If the problem continues, you will need to get further assistance.

- The PGP Support forums are user community forums hosted by PGP Corporation and monitored by PGP Corporation personnel. Check the PGP Whole Disk Encryption forums for more information.

  To access the PGP Support forums, please visit *PGP Support* (*http://forum.pgp.com*).

- The PGP Support Knowledge Base and PGP Technical Support may also be able to assist you with your issue.

  To access the PGP Support Knowledge Base or request PGP Technical Support, please visit *PGP Support Portal Web Site* (*https://support.pgp.com*). **Note that you may access portions of the PGP Support Knowledge Base without a support agreement; however, you must have a valid support agreement to request PGP Technical Support.**