Protegrity File Protector Guide 9.1.0.0

Created on: Aug 8, 2024

Notice

Copyright

Table of Contents

Chapter 1

Introduction to this Guide

This guide provides information about the overview, system requirements, installing, configuring, and using the File Protector.

1.1 Sections contained in this Guide

The guide is broadly divided into the following sections.

- Section *1 Introduction to this Guide* defines the purpose and scope for this guide. In addition, it explains how information is organized in this guide.
- Section *2 Overview of the File Protector* provides an overview of the File Protector for Windows platform including the general architecture, features, key components, and modules.
- Section *3 Installing and Uninstalling the File Protector* describes the procedures to install and uninstall the File Protector.
- Section *4 Upgrading the File Protector to v9.0.0.0* describes the procedures to upgrade the File Protector to v9.0.0.0.
- Section *5 Setting the Configuration Files* describes about the configuration files of the File Protector and their usage.
- Section *6 Managing the dfpshell* describes about the *dfpshell*, which is the system administrator shell for the File Protector. It is a privileged mode of operations for managing the File Protector.
- Section *7 Licensing* describes about the Licensing of the File Protector.
- Section *8 Using the Policy Management* describes about creating and managing the policies of the File Protector using the ESA.
- Section *9 Commands Overview* lists all the File Protector commands and their usage.
- Section *10 Using the File Protector* describes the usage of the supported features in the File Protector.
- Section *11 Backup and Restore the Protected Data* describes how to backup and restore protected data.
- Section *12 Metering* describes the Metering feature in the File Protector.
- Section *13 Troubleshooting* lists the common errors, permission restrictions, and problems that the users may encounter while working with the File Protector.
- Section *14 Use Cases for the File Protector* explains Use Cases of the File Protector.
- Section *15 Scenarios for Backup and Restore* describes scenarios and procedures to backup and restore the File Protector files and directories on Windows, including File Encryption, and Access Control.
- Section *16 Glossary* contains the list of terms and abbreviations used in this guide.

1.2 Accessing the Protegrity documentation suite

This section describes the methods to access the *Protegrity Documentation Suite* using the *My.Protegrity* portal.

1.2.1 Viewing product documentation

The **Product Documentation** section under **Resources** is a repository for Protegrity product documentation. The documentation for the latest product release is displayed first. The documentation is available in the HTML format and can be viewed using your browser. You can also view and download the *.pdf* files of the required product documentation.

1. Log in to the *My.Protegrity* portal.
2. Click **Resources** > **Product Documentation**.
3. Click a product version.
   The documentation appears.

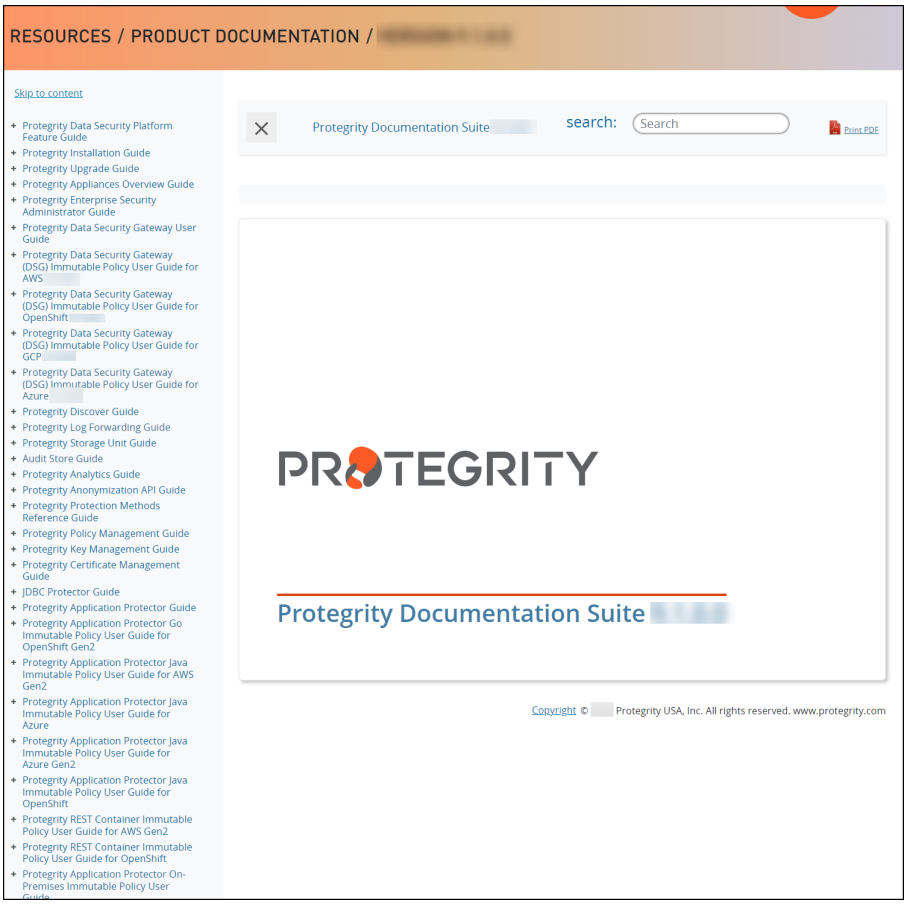


Figure 1-1: Documentation

4. Expand and click the link for the required documentation.
5. If required, then enter text in the **Search** field to search for keywords in the documentation.
   The search is dynamic, and filters results while you type the text.
6. Click the **Print PDF** icon from the upper-right corner of the page.
   The page with links for viewing and downloading the guides appears. You can view and print the guides that you require.

1.2.2 Downloading product documentation

This section explains the procedure to download the product documentation from the *My.Protegrity* portal.

1. Click **Product Management** > **Explore Products**.
2. Select **Product Documentation**.

The **Explore Products** page is displayed. You can view the product documentation of various Protegrity products as per their releases, containing an overview and other guidelines to use these products at ease.

3. Click **View Products** to advance to the product listing screen.
4. Click the **View** icon ( ) from the **Action** column for the row marked **On-Prem** in the **Target Platform Details** column.

   If you want to filter the list, then use the filters for: **OS**, **Target Platform**, and **Search** fields.
5. Click the icon for the action that you want to perform.

Chapter 2

Overview of the File Protector

The Protegrity File Protector provides a transparent solution for encrypting and protecting sensitive files. The File Protector can protect directories and sub-directories in real-time. As the File Protector operates at the file-system level, it ensures that the sensitive data is in protected form, whenever the file is accessed. The File Protector solution enables the applications and processes to transparently encrypt and decrypt files and directories. In this case, no modifications are required for the applications and processes. User can protect files or directories using the File Protector command set. After the protection is configured, the File Protector automatically allows access when the required policy is loaded for the application or user.

2.1 Architecture

This section discusses the architecture and the components of the File Protector.

The following diagram illustrates the architecture of the File Protector.

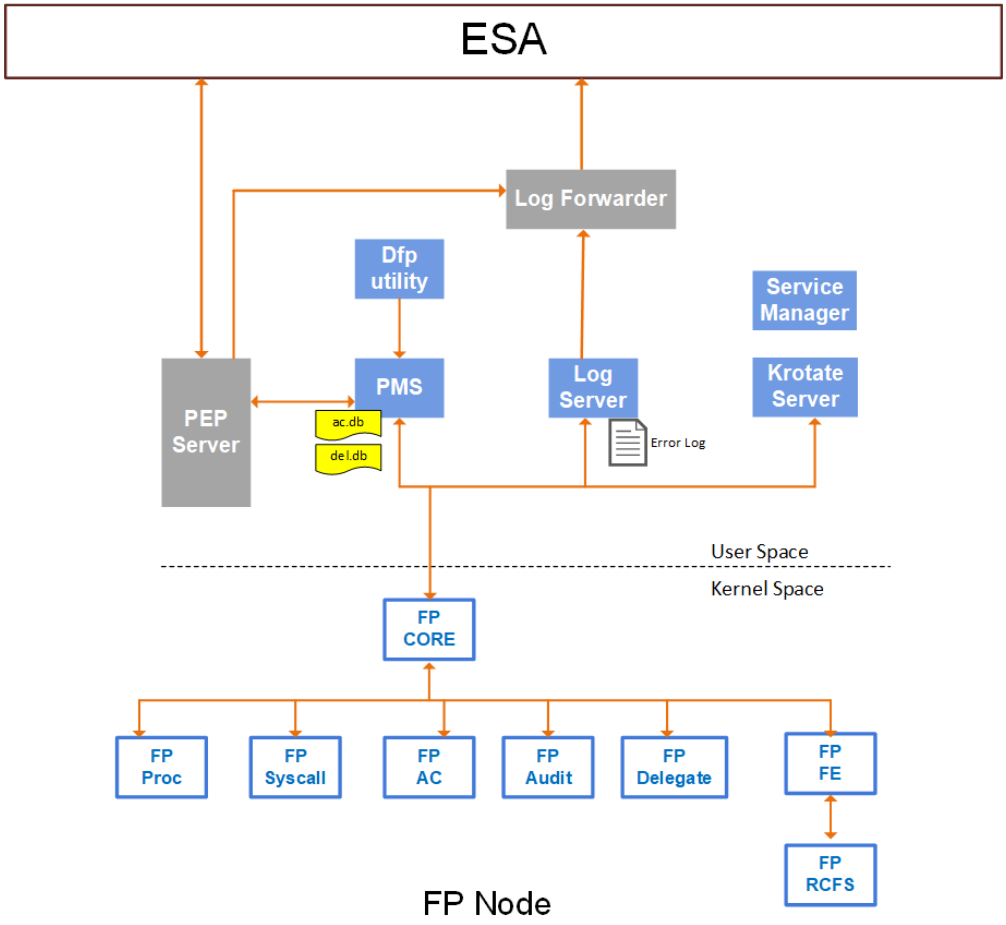


Figure 2-1: Architecture of the File Protector

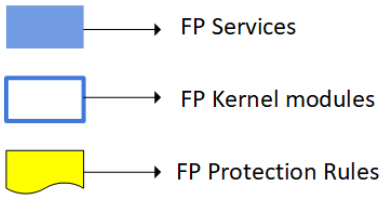


Figure 2-2: Legend for the Architecture

> **Note:** The File Protector design is based on the mini filter architecture of Windows. It taps the File System level calls and *process infrastructure*.

The key components of the File Protector can be classified into Services and Kernel modules.

Services

The following are the services of the File Protector:

- **PEP Server**

  The PEP server is the communication agent between the ESA and the File Protector. It is responsible for accepting the policy that is deployed from the Hub Controller. The PEP server is installed on the File Protector node and provides the data element key for the file encryption.

- **Log Forwarder**

The Log Forwarder is responsible for collecting log or audit information from the Log server and the PEP server and sending it to the ESA.

- **Log Server**

  The log server is responsible for collecting log or audit information locally. The log or audit information is lost if this service is not running.

- **PMS**

  The Policy Management Server (PMS) reads the policy information from the PEP server and provides it to the Kernel modules.

- **Krotate Server**

  The Key rotate (Krotate) server manages the key rotation on the protected files.

Modules

The following are the modules of the File Protector:

- **FP-Core**

  Manages requests from the user space services to the kernel modules.

- **FP-Proc**

  Policy management module maintains the policy cache.

- **FP-Syscall**

  Syscall module maintains the process level information required to manage the delegation.

- **FP AC**

  Supports the Access Control feature for protecting files and directories that contains sensitive data.

- **FP Audit**

  Audit modules relay audit information from the kernel to the user space.

- **FP Delegate**

  Delegation-related configurations are managed by this module.

- **FP FE**

  Supports the File Encryption feature that provides file-level encryption for files and directories containing sensitive information.

- **FP RCFS**

  Redirect Cache File System (RCFS) module enhances the file encryption performance.

2.2 Features

This section discusses the features of the File Protector.

Access Control

The Access Control settings protects a file or directory. The file protections apply only to the designated file. The directory protections apply to all the files available in a directory. The File Protector automatically enforces protection after the user has protected a file or directory. The File Protector requires that data elements be used to unprotect the protected resource from that point forward.

File Encryption

The File Encryption feature protects files and directories transparently on leading file systems and operating systems, regardless of the storage medium, such as hard drives, mounted volumes or shared from the Common Internet File System (CIFS) servers.

Delegation

Using the File Protector delegation feature, you can associate policies with applications, programs, processes, or users. When a program or process has been delegated, it can access protected and encrypted files based on the data element. A delegated user is able to access the protected files or directories upon logging in.

Key Rotation

The File Protector encryption feature provides the key rotation functionality, which automatically replaces the encryption key for the specified encrypted files. The Key rotation feature re-encrypts the encrypted files with the new active key using the same data element.

Audit Logging

The File Protector monitors the security operations and audit logs.

Chapter 3

Installing and Uninstalling the File Protector

This section describes how to install and uninstall the File Protector 9.1.0.0 on a Windows platform.

The minimum system requirements that must be met before installing the File Protector are explained in the section *System Requirements*

This section includes information about the following topics:

- Installing the Log Forwarder
- Installing the PEP Server
- Installing the File Protector
- Uninstalling the File Protector

3.1 System Requirements for Installation

The File Protector is packed as a set of executable files.

Before installing the File Protector, ensure that your configuration meets the minimum requirements. The following table lists the system requirements.

Table 3-1: System Requirements

System Component	Requirement
Operating System	• Windows Server 2016 • Windows Server 2019
CPU	P4 2.4 GHz or higher
Memory	Minimum 2 GB
Hard Disk	Minimum 1 GB free space

3.1 Defender Configuration

It is recommended that you disable the Windows Defender Protection or Windows real-time protection when running with File Protector.

> **Note:**
> The Windows Defender intervenes with the File Protector's encryption operation. If the Windows Defender is not disabled, then it can lead to inconsistencies in the encrypted data.

- To disable real-time protection using gpedit, follow these steps:
  1. Open the Local Group Policy Editor (type *gpedit.msc* in the search box).
  2. Navigate to **Computer Configuration** > **Administrative Templates** > **Windows Components** > **Microsoft Defender Antivirus**. Turn off the **Real-time Protection**.
  3. Restart the computer.
- To disable Microsoft Defender using group policy method, follow these steps:
  1. Open Local Group Policy Editor (type *gpedit.msc* in the search box).
  2. Navigate to **Computer Configuration** > **Administrative Templates** > **Windows Components**. Turn off the **Microsoft Defender Antivirus**.
  3. Restart the computer.
- To disable Defender Antivirus Protection in Windows Security, follow these steps:
  1. Navigate to **Start** > **Settings** > **Update & Security** > **Windows Security** > **Virus & threat protection** > **Manage settings**. Turn off the **Microsoft Defender Antivirus**.
  2. Turn off the **Real-time protection**.

3.2 Installing the Log Forwarder

The Log Forwarder is responsible for collecting log or audit information from the Log server and the PEP server and sending it to the ESA.

> **Note:** For more information about Log Forwarder, refer to section *Installing the Log Forwarder* in the *Protegrity Installation Guide 9.1.0.0*.

➤ To install the Log Forwarder on Windows:

1. Extract the `FileProtector_<OS>-64_x86-64_<version>.zip` file.
   The following files are extracted:
   - `PepServerSetup_<OS>_x64_<version>.exe`
   - `FileProtector_<OS>-64_x86-64_AccessControl_<version>.exe`
   - `FileProtector_<OS>-64_x86-64_FileEncryption_<version>.exe`
   - `LogforwarderSetup__<OS>_x64_<version>.exe`
   - `INSTALL.txt`
2. Run the *LogforwarderSetup_<OS>_<version>.exe* file from the created directory.
   The **Log Forwarder Setup Wizard** appears.

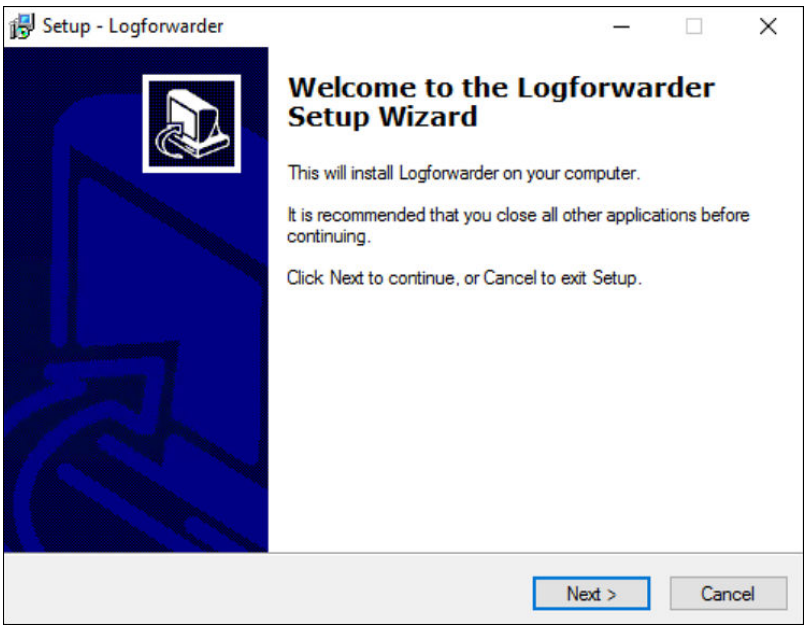


Figure 3-1: Log Forwarder Setup Wizard

3. Click **Next**.
4. Select the number of audit stores needed.

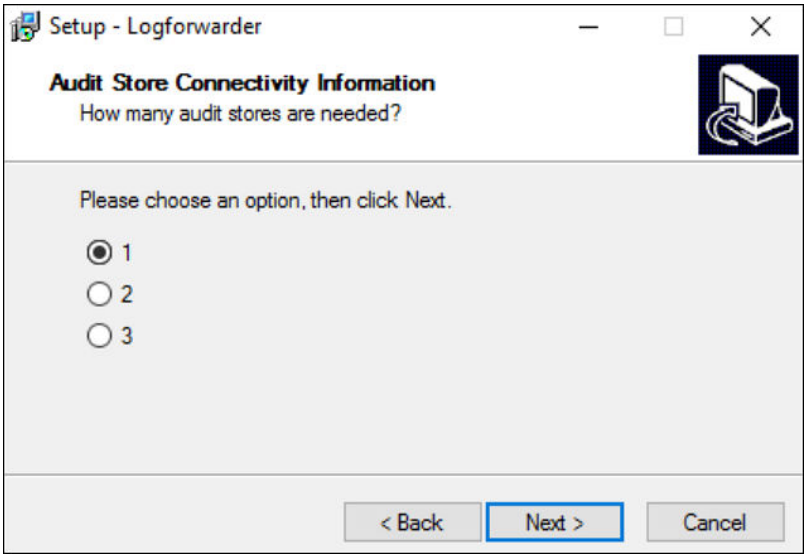


Figure 3-2: Audit Store Connectivity Information

5. Click **Next**.
6. Enter the Audit store Endpoint or Port number, where the Audit store is configured.

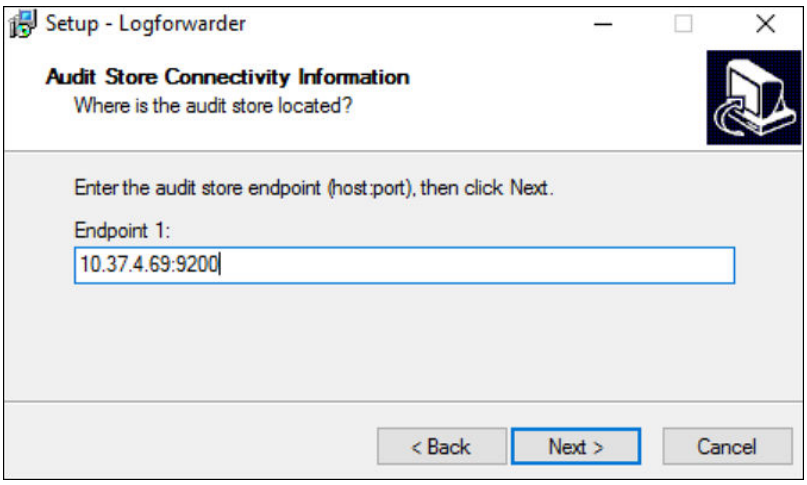


Figure 3-3: Audit Store Endpoint

7. Click **Next**.
   The *Select Destination Location* screen appears.
8. Set the installation directory for the Logforwarder to *C:\Program Files\Protegrity\fluent-bit*.
9. Click **Next**.
   The *Ready to Install* screen appears.
10. Click **Install**.
11. From the **Completing the Logforwarder Setup Wizard** screen, click **Finish** to complete the installation and exit. The directories are created under the installation directory that was defined and the installation files are installed in these directories.

    By default, the Logforwarder service should run automatically.

12. Perform the following steps to start the Log Forwarder.
    a. From the Windows Start Menu, search and select *Services*.
    b. Navigate to the *Logforwarder* service.
    c. Right-click the service and click **Start**.

Silent Mode of Installation

You can also execute the Log Forwarder installer without any manual intervention, which is also known as the Silent mode of installation. The following parameters must be provided to execute the installer in the Silent mode.

Table 3-2: Parameter List for Silent Installation

Parameter	Description
-endpoint1, -endpoint2, -endpoint3	Audit Store IP address and the Port number where the Log forwarder listens for logs **Note:** The default port number is *9200*. **Note:** The parameters -endpoint2 and -endpoint3 are optional.

Parameter	Description
-dir	Installation directory of the Log Forwarder, which is optional. If the installation directory is not specified, then the installation path is the default directory, which is the `..\Protegrity\fluent-bit` directory.
-pepdir	Installation directory of the PEP server, which is optional. If the installation directory is not specified, then the installation path is the default directory, which is the `..\Protegrity` directory.

At the command prompt, type the following command from the installer directory.

```
.\LogforwarderSetup_Windows_x64.exe -endpoint1 <ip address and port number> [-endpoint2 <ip
address and port number>] [-endpoint3 <ip address and port number>]
```

If you want to install the Log Forwarder and the PEP server in a directory other than the default directory, then you can add the *-dir* parameter to the command to specify the Log Forwarder installation directory and the *-pepdir* parameter to the command to specify the PEP server installation directory. The following snippet displays a sample command.

```
.\LogforwarderSetup_Windows_x64.exe -endpoint1 <ip address and port number> [-endpoint2 <ip
address and port number>] [-endpoint3 <ip address and port number>] -dir <Log Forwarder
installation directory> -pepdir <PEP server installation directory>
```

3.3 Installing the PEP Server

The PEP Server is the communication agent between ESA and the File Protector. It is responsible for accepting the policy that is deployed from the ESA.

> **Note:** For more information about PEP Server, refer to section *Installing and uninstalling PEP Server* in the *Protegrity Installation Guide 9.1.0.0*.

Before you begin

Before setting up the PEP Server, ensure that the following pre-requisites are met:

- Your environment meets the minimum requirements as mentioned in the section *PEP Server Pre-Installation Preparation* in the *Protegrity Installation Guide 9.1.0.0*.
- The ESA is up and running and the *Admin-Server* and *HubController* services are in running status to enable downloading the certificates automatically.

To install the PEP Server on Windows:

1. Run the `PepServerSetup_<OS>_x64_<version>.exe` file.

   The **PEP Server Setup Wizard** appears.

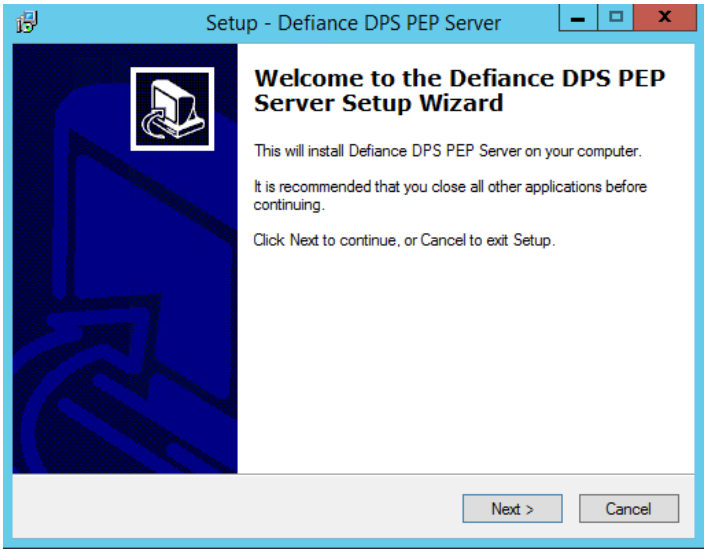


Figure 3-4: PEP Server Setup Wizard Screen

2. Click **Next** to begin installation.
   The **ESA Connectivity Information** screen appears.

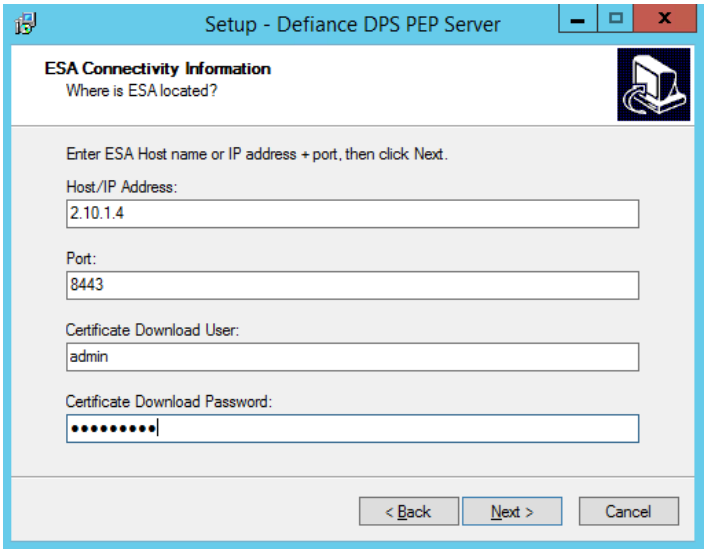


Figure 3-5: ESA Connectivity Information Screen

3. Enter the ESA Host name or IP Address in the **Host/IP Address** text box.
4. Enter the port in the **Port** text box.
   The ESA host will be included in the `pepserver.cfg` file.
5. Enter the user name for downloading certificates in the **Certificate Download User** text box.
6. Enter the password for downloading ESA certificates in the **Certificate Download Password** text box.

> **Note:** The following directories are automatically created:
> `..\Protegrity\Defiance DPS`: This is the default installation directory which includes the `\bin` and `\data` directories.
>
> `\bin`: This directory includes the executable files of the PEP Server.
>
> `\data`: This directory includes the configuration files of the PEP Server.

7. Click **Next**.
   The **Select Destination Location** screen appears.

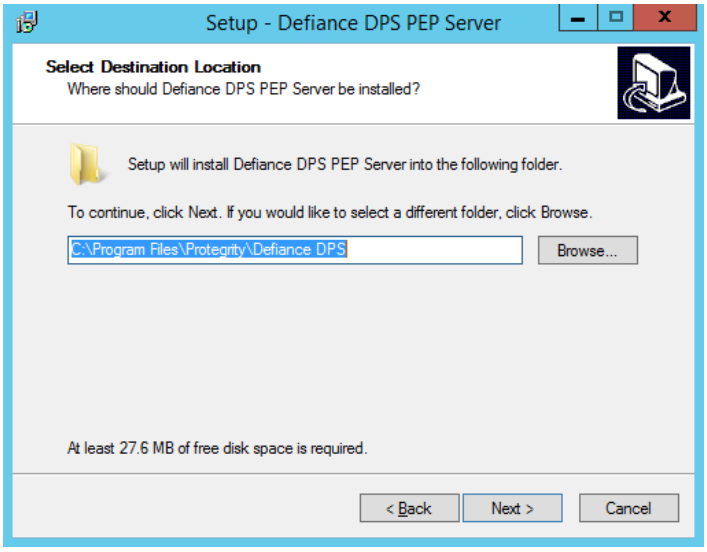


Figure 3-6: Select Destination Location Screen

8. Browse to the directory to which you want to install the Protegrity PEP Server or leave the default location (recommended).
9. Click **Next**.
   The **Ready to Install** screen appears.

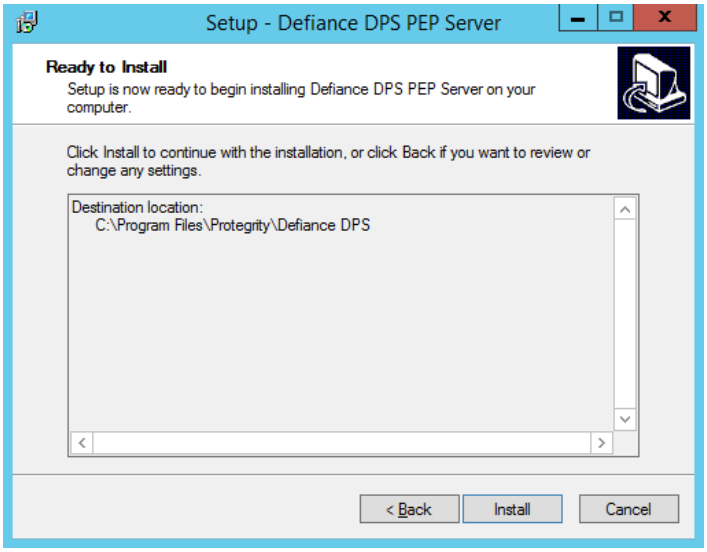


Figure 3-7: Ready to Install Screen

10. Click **Install**.
    The **Completing the Defiance DPS PEP Server Setup Wizard** screen appears.

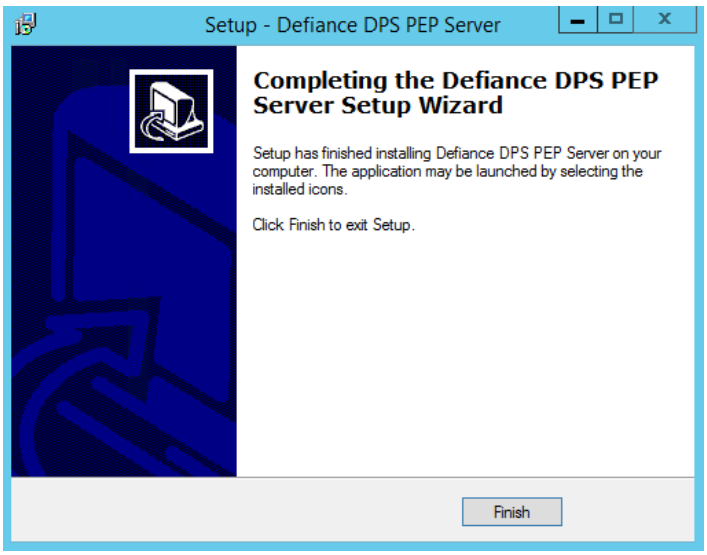


Figure 3-8: Completing the PEP Server Setup Wizard

11. Click **Finish** to complete the installation and exit.

> **Note:**
> The directories are created under the installation directory that you specified, and the File Protector files are installed in the directories.

> **Note:**
> To verify the status of the PEP server, navigate to **Windows** > **Start** > **Services**.

3.4 Installing the File Protector

This section describes how to install the File Protector Access Control, and File Encryption features.

Before installing the File Protector, ensure that the following prerequisites are met:

- The Windows Defender Protection or Windows real-time protection is disabled.
- The Logforwarder is installed and running.
- The PEP server is installed and running.
- The ESA is installed and running and the *Admin-Server* and *HubController* services are in running status.
- The ESA authentication files are present in the `..\Protegrity\Defiance DPS` directory.
- The Administrator privileges are available on the machine where you intend to install the File Protector.
- Ensure that the node is added to a Data Store.

You can select either of the two methods to install the File Protector.

- *Using the GUI*
- *Using the Command Line*

3.4.1 Using the GUI

This section describes how to install the File Protector using the GUI.

3.4.1.1 Installing the Access Control Feature

The Access Control feature protects the files and directories on the file leading File Systems and Operating Systems. This section describes the steps to install the Access Control feature.

To install the Access Control Feature:

1. Run the *FileProtector_<OS>-64_x86-64_AccessControl_<version>.exe* file.
   The **FP Access Control Setup Wizard** appears.

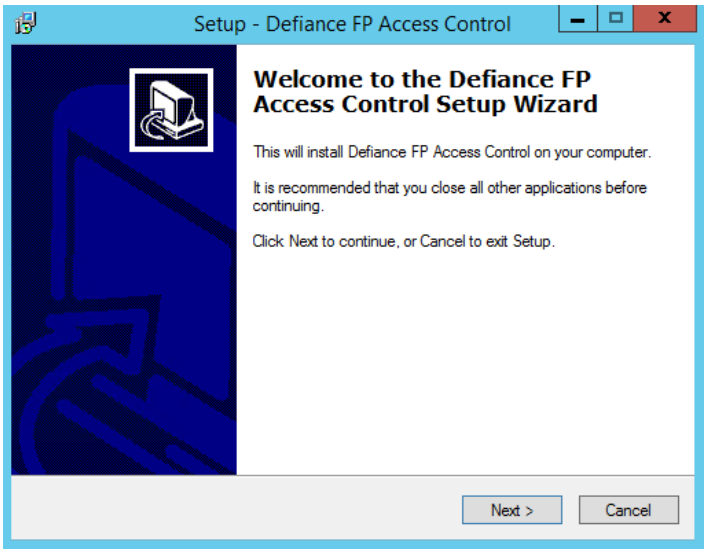


Figure 3-9: FP Access Control Setup Wizard Screen

2. Click **Next** to begin installation.
   The **Select Destination Location** screen appears.

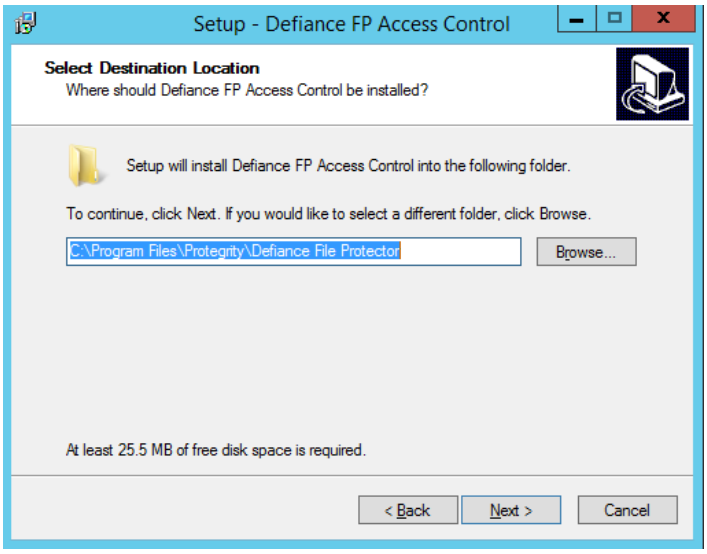


Figure 3-10: Select Destination Location Screen

3. Browse to the directory to which you want to install the FP Access Control, or leave the default location (recommended).
4. Click **Next**.
   The **Ready to Install** screen appears.

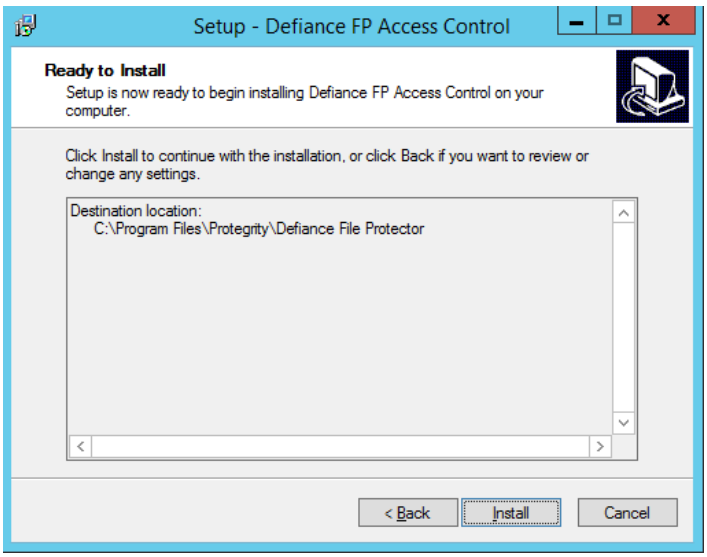


Figure 3-11: Ready to Install Screen

5. Click **Install**.

   The **Set Up Dfpshell Pass Phrase** console appears.

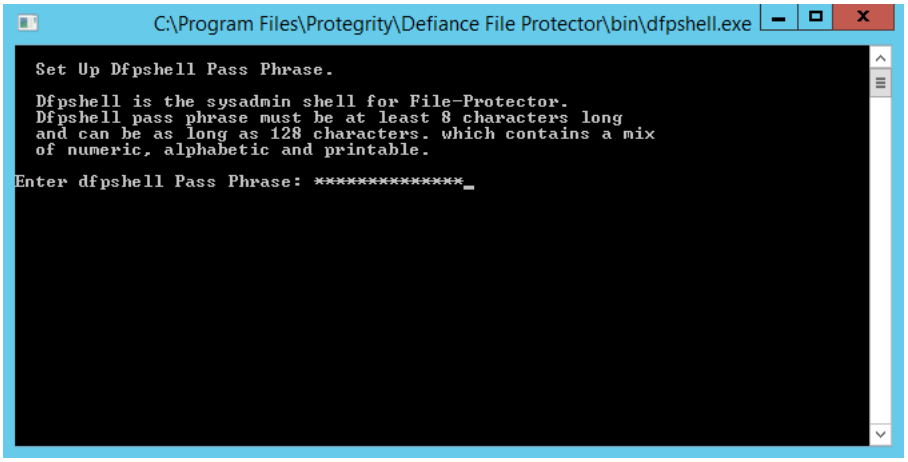


Figure 3-12: Set Up Dfpshell Pass Phrase Console

6. Type the *dfpshell* password.

> **Note:**
> The *dfpshell* password must meet the following criteria:
> - Should be a minimum 8 characters in length
> - Should contain a mix of numeric, alphabetic, and printable characters

7. Press **Enter**.
8. Type the *dfpshell* password again for verification.
9. Press **Enter**.

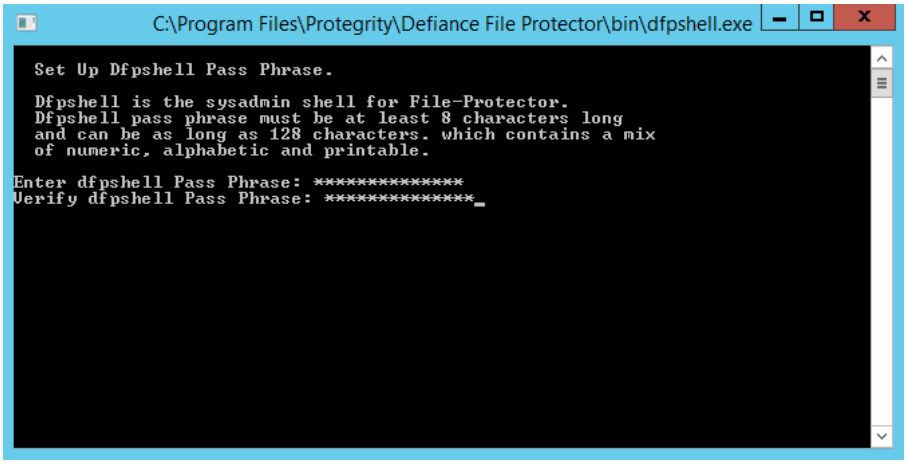


Figure 3-13: Verify the dfpshell password

After successful authentication of the *dfpshell* password, the setup is complete.

The **Completing the Defiance FP Access Control Setup Wizard** screen appears.

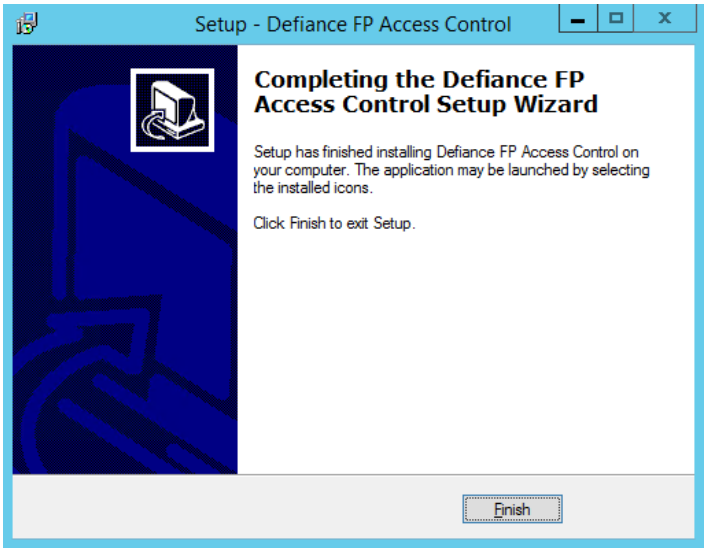


Figure 3-14: Completing the FP Access Control Setup Wizard

10. Click **Finish** to complete the installation and exit.

3.4.1.2 Installing the File Encryption Feature

The File Encryption feature allows File-level transparent encryption on the sensitive files and directories. This section describes the steps to install the File Encryption feature.

Before you begin
Ensure that the following prerequisites are met:

- The Windows Defender Protection or Windows real-time protection is disabled.
- The Log Forwarder is installed and running.
- The PEP server is installed and running.
- The ESA is installed and running and the *Admin-Server* and *HubController* services are in running status.
- The ESA authentication files are present in the `..\Protegrity\Defiance DPS` directory.
- The Administrator privileges are available on the machine where you intend to install the File Protector.

- The Access Control feature is installed and the *dfpshell* password is configured.
- The Policy Management Service (PMS) is up and running.

To install the File Encryption Feature:

1. Run the `FileProtector_<OS>-64_x86-64_FileEncryption_<version>.exe` file.
   The console appears to verify the *dfpshell* password.

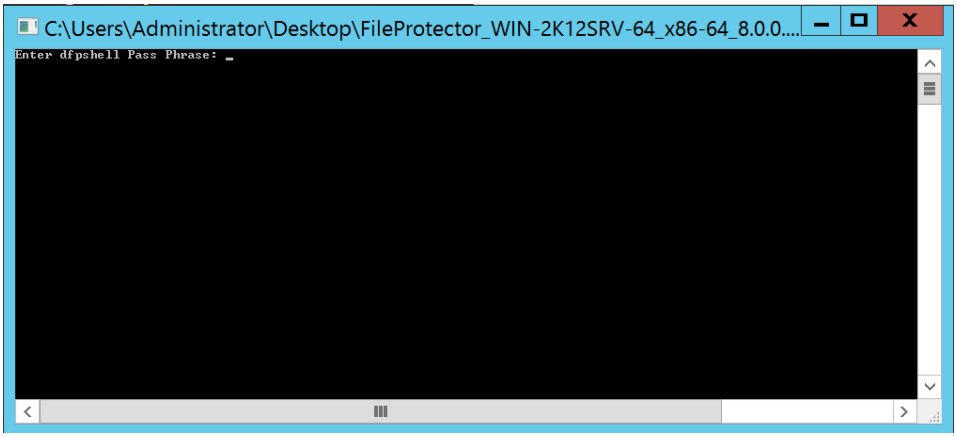


Figure 3-15: Verify the dfpshell Password

2. Type the *dfpshell* password.
3. Press **Enter**.
   The **FP File Encryption Setup Wizard** appears.

Figure 3-16: FP File Encryption Setup Wizard

4. Click **Next** to begin installation.
   The **Select Destination Location** screen appears.

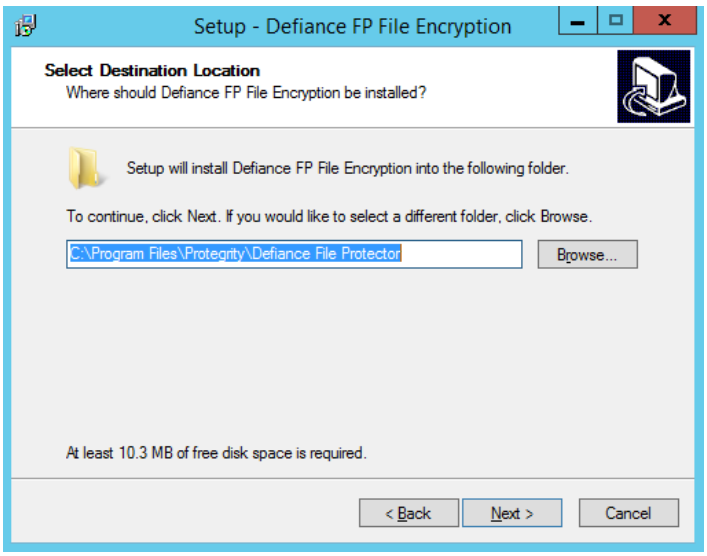


Figure 3-17: Select Destination Location Screen

5. Browse to the directory to which you want to install the FP Access Control, or leave the default location (recommended).
6. Click **Next**.
   The **Ready to Install** screen appears.

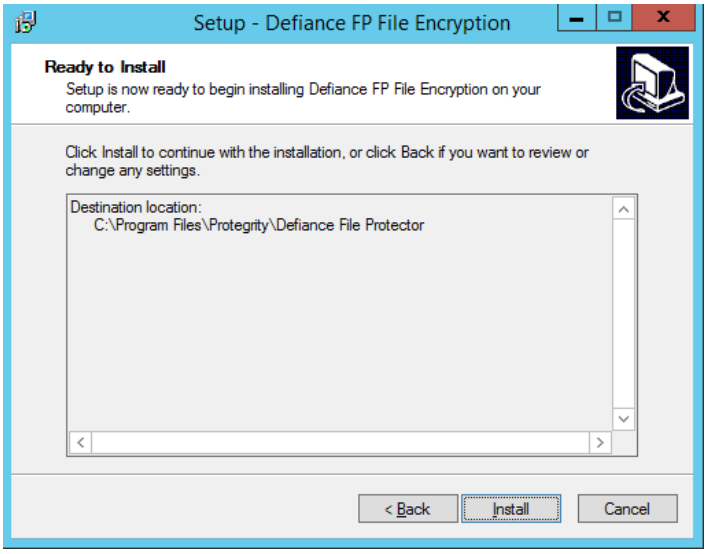


Figure 3-18: Ready to Install Screen

7. Click **Install**.
   The **Completing the Defiance FP File Encryption Setup Wizard** screen appears.

Figure 3-19: Completing the Defiance FP File Encryption Setup Wizard

8. Click **Finish** to complete the installation and exit.

3.4.2 Using the Command Line

This section describes the steps to install the File Protector using command line options. You can specify the parameters to install the File Protector without the interactive inputs.

The following table describes the options available for installing the File Protector using command line:

> **Note:**
> Run `-help` to list the usage of the installation parameters.

Table 3-3: Installation Parameters

Parameters	Description
`/silent`	Answers all the interactive questions when installing the File Protector features except the password. It specifies `C:\Program Files` as the default installation directory for the File Protector.
`/dfpshell-password <password>`	Specifies the *dfpshell* privilege password in the command for the installation of File Protector components.
`/peprestart`	Defines whether to restart the PEP server during the installation of the Access Control. This option is used for installing the Access Control feature.
`[/dir <install path> ]`	Specifies the absolute parent directory where you want to install the File Protector. This option is only for installing the Access Control feature.

To silently install File Protector:

1. Extract the `FileProtector_<OS>-64_x86-64_<version>.zip` file.
   The following files are extracted:
   - `PepServerSetup_<OS>_x64_<version>.exe`
   - `FileProtector_<OS>-64_x86-64_AccessControl_<version>.exe`
   - `FileProtector_<OS>-64_x86-64_FileEncryption_<version>.exe`

- *LogforwarderSetup_<OS>_x64_<version>.exe*
- *INSTALL.txt*

2. Install the *LogforwarderSetup_<OS>_x64_<version>.exe* file.
3. Install the *PepServerSetup_<OS>_x64_<version>.exe* file.
4. Run the following script to install the Access Control feature.
   FileProtector_<OS>-64_x86-64_AccessControl_<version>.exe /silent /dfpshell-password <password>
5. Run the following script to install the File Encryption feature.
   FileProtector_<OS>-64_x86-64_FileEncryption_<version>.exe /silent /dfpshell-password <password>

The File Protector is installed in the following path:
C:\Program Files\Protegrity\Defiance File Protector

3.5 Uninstalling

You can uninstall the File protector using the standard Windows Programs and Files uninstall facility.

While uninstalling the File Protector, ensure that the following prerequisites are met:

- No *dfp* command process is running in the background and the *dfpshell* is not loaded.
- The Administrator privileges are available on the machine where the File Protector is installed.

You can select either of the two methods to uninstall the File Protector.

- *Using the GUI*
- *Using the Command Line*

3.5.1 Using the GUI

This section describes the steps to uninstall the File Protector using the GUI.

> **Note:**
> As the features are interdependent, the Access Control feature must be uninstalled after you have completed uninstalling the File Encryption features.

To uninstall the File Protector:

1. Navigate to the installed directory and then go to **Protegrity** > **Defiance File Protector**.
2. Double click the *uninstFileEnc.bat* file to uninstall the File Encryption feature.
   A prompt for the *dfpshell* password appears.
3. Enter the *dfpshell* password.
4. Press **ENTER**.
   The **FP File Encryption Uninstall** dialog box appears.

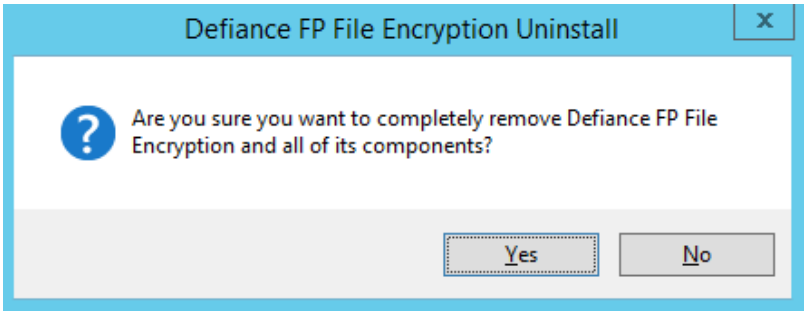


Figure 3-20: FP File Encryption Uninstall Dialog box

5. Click **Yes** to uninstall the File Protector File Encryption feature.
   The **FP File Encryption Uninstall** dialog box appears with a message *Defiance FP File Encryption was successfully removed from your computer.*

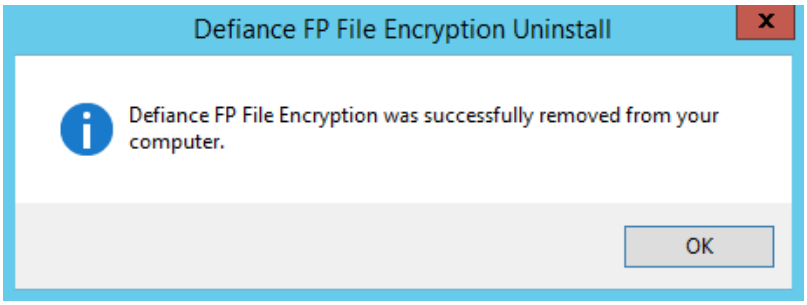


Figure 3-21: FP File Encryption Uninstall Dialog box

6. Click **OK** to exit.
7. Navigate to the installed directory and then go to **Protegrity** > **Defiance File Protector**.
8. Double click the `uninstAccessControl.bat` file to uninstall the Access Control feature.
   A prompt for the *dfpshell* password appears.
9. Enter the *dfpshell* password.
10. Press **ENTER**.
    The **FP Access Control Uninstall** dialog box appears.

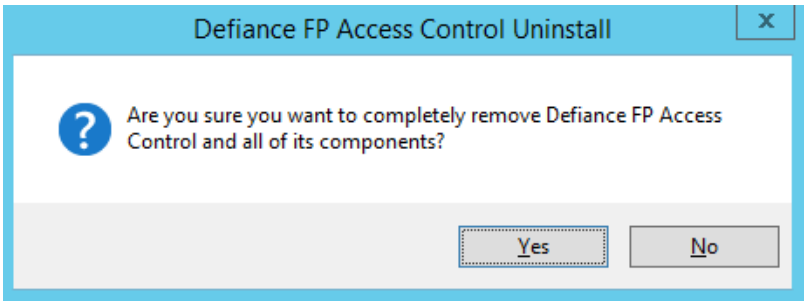


Figure 3-22: FP Access Control Uninstall Dialog box

11. Click **Yes** to uninstall the File Protector Access Control feature.
    The **FP Access Control Uninstall** dialog box appears with a message *Defiance FP Access Control uninstall complete.*

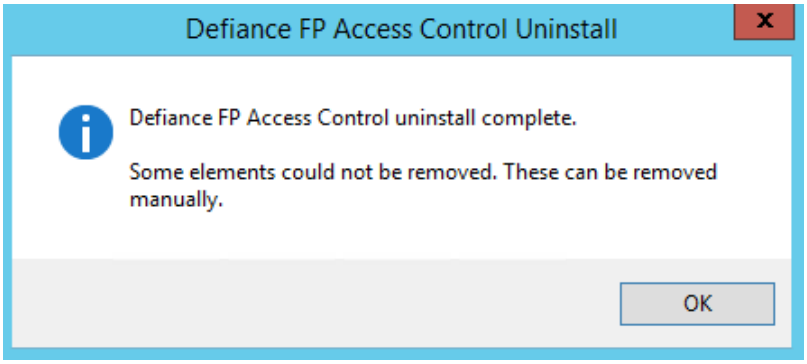


Figure 3-23: FP Access Control Uninstall Dialog box

> **Note:** Navigate to the `Programs Files/Protegrity/Defiance File Protector` directory to delete the following items manually:
>
> - `uninstFileEnc.bat` file
> - `uninstAccessControl.bat` file

12. Click **OK** to exit.

3.5.2 Using the Command Line

This section describes the steps to uninstall the File Protector silently.

➤ To silently uninstall the File Protector:

1. Navigate to the installation directory where you have installed the File Protector and then navigate to **Protegrity** > **Defiance File Protector**.
2. Run the following script to uninstall the File Encryption feature.
   ```
   uninstFileEnc.bat /silent /dfpshell-password <password>
   ```
3. Run the following script to uninstall the Access Control feature.
   ```
   uninstAccessControl.bat /silent /dfpshell-password <password>
   ```
4. Follow the message that is prompted to restart your machine.

Chapter 4

Upgrading the File Protector

This section describes how to upgrade the File Protector on a Windows platform.

If you already have a File Protector installed, then you can upgrade from the current version to the latest version. The upgrade procedure varies depending on the existing version of the File Protector.

4.1 Upgrading the File Protector from v7.x to v9.1.0.0

This section describes how to upgrade the File Protector on a Windows platform from File Protector v7.x to File Protector v9.1.0.0.

Before you begin
Ensure that the following prerequisites are met:

- Ensure that the ESA is upgraded to v9.1.0.0 and configured successfully.

  > **Note:**
  > For more information about upgrading the ESA, refer to section *Upgrade Paths to ESA v9.1.0.0* in the *Protegrity Data Security Platform Feature Guide 9.1.0.0*.

- Ensure that the PEP server is upgraded and configured successfully.

  > **Note:**
  > For more information about upgrading the PEP server, refer to section *Installing the PEP Server*.

- The *dfpshell* privileges are available.
- When performing the upgrade, ensure that the other Data Security Platform components, such as, the ESA, the PEP servers, among others, are compatible with your specific File Protector version.
- Ensure that you are able to add a node successfully.
- Ensure that the data store key is active.

To upgrade the File Protector from v7.x to v9.1.0.0:

1. Check the status of the Windows Defender. If Windows Defender is enabled, then perform the steps mentioned in the following Note. If the Windows Defender is disabled, then continue with step 2.

   Note:
   For more information about Defender Configuration, refer to section *System Requirements for Installation*.

   Note:
   Please contact Protegrity support to get access to the utility that can identify the files for backup before initiating the upgrade process.

   If the Windows Defender is enabled, then perform the following steps.

   1. Run the Powershell script provided to get the filepath.
   2. Disable the Windows Defender.
   3. Get a checksum of all the files and save it as a reference.
   4. Copy all the encrypted files in a different directory as a backup.
   5. Restart the system.
   6. Compare the checksum value of all the original files (not applicable for the backup files).
   7. If the checksum value changes after the restart, then remove it and restore the file from the backup.

2. If there are protected volumes, then run the commands mentioned in the following Note. If there are no protected volumes, then continue with step 3.

   Note:
   The Volume Encryption feature is not supported from v9.0.0.0.

   Note:
   If there are protected volumes, then run the following commands to unprotect all the protected volume drives.

dfp volume unprotect [- y] [-p <policy>] ***[--backup <temp backup path>]*** ***[--prev_exec <prev script>]*** ***[--post_exec <post script>]*** ***<-l <drive letters list file>|<drive letter> [<drive letter2> ...]>***	Unprotects the Volume Protected drives, including removing the Volume Encryption and Access Control.

3. Uninstall the Volume module.
4. Restart the node as the volume module is being uninstalled.
5. Stop the PEP server by running the following command from the command line.
   net stop pepserver
6. Uninstall the PEP server.
7. Extract the *FileProtector_<OS>-64_x86-64_<version>.zip* file.

The following files are extracted:

- *PepServerSetup_<OS>_x64_<version>.exe*
- *FileProtector_<OS>-64_x86-64_AccessControl_<version>.exe*
- *FileProtector_<OS>-64_x86-64_FileEncryption_<version>.exe*
- *LogforwarderSetup_<OS>_x64_<version>.exe*
- *INSTALL.txt*

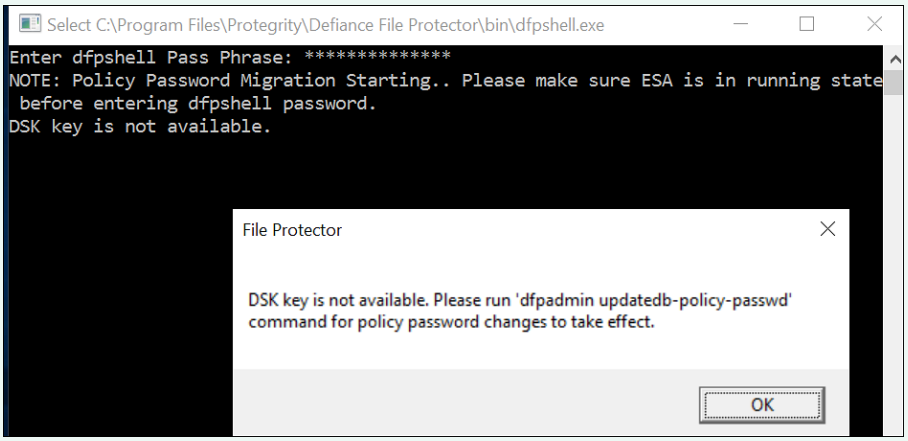
8. From the installation package, run the following file to install the Log Forwarder server.
LogforwarderSetup_<OS>_x64_<version>.exe

> **Note:**
> Make sure that the Log Forwarder is in the running state.
>
> To run the Log Forwarder from the Task Manager:
>
> 1. Navigate to the **Task Manager** > **Services**.
> 2. Right-click on the **Logforwarder service** and click **Start**.

9. From the installation package, run the following file to install the PEP server.
PepServerSetup_<OS>_x64_<version>.exe

> **Note:**
> Make sure that the PEP server is in the running state.
>
> Perform the following steps to run the Log Forwarder from the Task Manager:
>
> 1. Navigate to the **Task Manager** > **Services**.
> 2. Right-click on the **Protegrity PEP server** and click **Start**.

10. From the installation package, run the following file to upgrade the Access Control feature.
FileProtector_<OS>-64_x86-64_AccessControl_<version>.exe
11. From the installation package, run the following file to upgrade the File Encryption feature.
FileProtector_<OS>-64_x86-64_FileEncryption_<version>.exe

> **Note:**
> 1. If the prompt for *DSK key is not available* displays, then click **OK** to finish the upgrade.
>
> 
>
>
> *Figure 4-1:*
>
> 2. Run the following command for policy password changes to take effect.

```
dfpadmin database -o updatedb-policy-passwd -p policy_name <policy_password>
```

12. Run the following command to verify that all the File Protector services are running.
    ```
    dfpadmin service all status
    ```
13. Run the following command to update all the configuration files.
    ```
    dfpadmin update
    ```
14. Run the following command to check if all the delegated programs/users are in an active state.
    ```
    dfp delegate status
    ```
15. Run the following command to verify the version of the File Protector installed.
    ```
    dfp version
    ```
16. Run the following command to verify the version of the Access Control installed.
    ```
    dfp ac version
    ```
17. Run the following command to verify the version of the File Encryption feature installed.
    ```
    dfp fe version
    ```
18. Run the following command to verify that the protections made in File Protector v7.x are still available.
    ```
    dfpshell dfpadmin status
    ```

The File Protector is upgraded to v9.1.0.0.

4.2 Upgrading the File Protector from v9.0.0.0 to v9.1.0.0

This section describes how to upgrade the File Protector on a Windows platform from File Protector v9.0.0.0 to File Protector v9.1.0.0.

Before you begin
Ensure that the following prerequisites are met:

- Ensure that the ESA is upgraded to v9.1.0.0 and configured successfully.

  > **Note:**
  > For more information about upgrading the ESA, refer to section *Upgrade Paths to ESA v9.1.0.0* in the *Protegrity Data Security Platform Feature Guide 9.1.0.0*.

- Ensure that the PEP server is upgraded and configured successfully.

  > **Note:**
  > For more information about upgrading the PEP server, refer to section *Installing the PEP Server*.

- The *dfpshell* privileges are available.
- When performing the upgrade, ensure that the other Data Security Platform components, such as, the ESA, the PEP servers, among others, are compatible with your specific File Protector version.
- Ensure that you are able to add a node successfully.
- Ensure that the data store key is active.

➤ To upgrade the File Protector from v9.0.0.0 to v9.1.0.0:

1. Check the status of the Windows Defender. If Windows Defender is enabled, then perform the steps mentioned in the following Note. If the Windows Defender is disabled, then continue with step 2.

   Note:
   For more information about Defender Configuration, refer to section *System Requirements for Installation*.

   Note:
   Please contact Protegrity support to get access to the utility that can identify the files for backup before initiating the upgrade process.

   If the Windows Defender is enabled, then perform the following steps.

   1. Run the Powershell script provided to get the filepath.
   2. Disable the Windows Defender.
   3. Get a checksum of all the files and save it as a reference.
   4. Copy all the encrypted files in a different directory as a backup.
   5. Restart the system.
   6. Compare the checksum value of all the original files (not applicable for the backup files).
   7. If the checksum value changes after the restart, then remove it and restore the file from the backup.

2. Stop the PEP server by running the following command from the command line.
   net stop pepserver
3. Uninstall the PEP server.
4. Extract the *FileProtector_<OS>-64_x86-64_<version>.zip* file.
   The following files are extracted:
   - *PepServerSetup_<OS>_x64_<version>.exe*
   - *FileProtector_<OS>-64_x86-64_AccessControl_<version>.exe*
   - *FileProtector_<OS>-64_x86-64_FileEncryption_<version>.exe*
   - *LogforwarderSetup_<OS>_x64_<version>.exe*
   - *INSTALL.txt*
5. From the installation package, run the following file to install the Log Forwarder server.
   LogforwarderSetup_<OS>_x64_<version>.exe

   Note:
   Make sure that the Log Forwarder is in the running state.

   To run the Log Forwarder from the Task Manager:

   1. Navigate to the **Task Manager** > **Services**.
   2. Right-click on the **Logforwarder service** and click **Start**.

6. From the installation package, run the following file to install the PEP server.
   PepServerSetup_<OS>_x64_<version>.exe

   Note:
   Make sure that the PEP server is in the running state.

> Perform the following steps to run the Log Forwarder from the Task Manager:
>
> 1. Navigate to the **Task Manager** > **Services**.
> 2. Right-click on the **Protegrity PEP server** and click **Start**.

7. From the installation package, run the following file to upgrade the Access Control feature.
   FileProtector_<OS>-64_x86-64_AccessControl_<version>.exe
8. From the installation package, run the following file to upgrade the File Encryption feature.
   FileProtector_<OS>-64_x86-64_FileEncryption_<version>.exe
9. Run the following command to verify that all the File Protector services are running.
   dfpadmin service all status
10. Run the following command to update all the configuration files.
    dfpadmin update
11. Run the following command to check if all the delegated programs/users are in an active state.
    dfp delegate status
12. Run the following command to verify the version of the File Protector installed.
    dfp version
13. Run the following command to verify the version of the Access Control installed.
    dfp ac version
14. Run the following command to verify the version of the File Encryption feature installed.
    dfp fe version
15. Run the following command to verify that the protections made in File Protector v9.0.0.0 are still available.
    dfpshell dfpadmin status

The File Protector is upgraded to v9.1.0.0.

Chapter 5

Setting the Configuration Files

The File Protector configuration files contain parameter settings that are required for the File Protector services.

The following table describes the configuration files for the File Protector.

Configuration Files	Description
fe_disallow. conf	Contains the list of file paths that are not allowed to be encrypted using FE.
ac_disallow.conf	Contains the list of file paths that are not allowed to be protected using AC.
audit.conf	Contains the default configuration settings for audits.
dfp_ldap.conf	Configures the LDAP for managing the *dfpshell* account.
fe. conf	Enables or disables the redirect cache for files encrypted using FE.
key_rotation.conf	Contains configuration related to the key rotation functionality for the encrypted files and specifies the time interval of key rotation.
path_name_info.conf	Configures the representation of the protected path.
policy_management_server.conf	Configures the Policy Management Server (PMS) related settings.
pepserver.cfg	Configures the Log Server related settings.

Note: After updating any configuration file, ensure that you run the following command to update the configuration changes.

```
dfpadmin update
```

5.1 *fe_disallow.conf* File

In this configuration file, you can list the file and directory source paths that must not be encrypted by the `dfp fe protect` or the `dfp fe protect` commands. By default, the source paths of the system file are added in this configuration file to prevent protection of system files.

To add a new source path in the *fe_disallow.conf* file, update and save the new path manually in the file.

> **Note:** You must configure the *fe_disallow.conf* file before protecting the data with AC.

The following snippet displays a sample of the configuration settings. The usage of the *fe_disallow.conf* file is described in the comments within the *fe_disallow.conf* file.

```
#
# File Protector file encryption disallow configuration file.
# The following paths are disallowed to protect by 'dfp fe protect'.
#
# e.g.
#    C:\Windows
#    C:\Windows\System32
#
C:\Windows
```

> **Caution:** The system file paths should not be deleted. If you delete the system file paths, then the system files are available for protection by the File Protector commands. If you protect the system files, then this disrupts the operation of the system.

5.2 *ac_disallow.conf* File

In this configuration file, you can list the file and directory source paths that must not be protected by the ***dfp ac protect*** and ***dfp fe protect*** commands. By default, the system file source paths are added in this configuration file to prevent protection of system files.

To add a new source path in the *ac_disallow.conf* file, update and save the new path manually in the file.

> **Note:** You must configure the *ac_disallow.conf* file before protecting the data with AC.

The following snippet displays a sample of the configuration settings. The usage of the *ac_disallow.conf* file is described in the comments within the *ac_disallow.conf* file.

```
#
# File Protector access control disallow configuration file.
# The following paths are disallowed to protect by 'dfp ac protect'.
#
# e.g.
#    C:\Windows
#    C:\Windows\System32
#
C:\Windows
```

> **Caution:** The system file paths should not be deleted. If you delete the system file paths, then the system files are available for protection by the File Protector commands. If you protect the system files, then this disrupts the system operation.

5.3 *audit.conf* File

The File Protector provides an *audit.conf* file located in the *<Installation path of the File Protector>/protegrity/defiance file protector/data* directory to configure the events listed in the following table.

Table 5-1: Default Configuration Settings

Configuration Parameters	Options	Default Value	Description
Audit Configuration Items	LOAD_POLICY	{ALL}/SF	Load policy action
	DFPSHELL	{ALL}/SF	*dfpshell* is used
	UPDATE	{ALL}/SF	File Protector update or upgrade
	UNINSTALL	{ALL}/SF	File Protector uninstall
	KEY_ROTATION_ADD	{ALL}/SF	Add key rotation
	KEY_ROTATION_DEL	{ALL}/SF	Delete key rotation
	PROTECT	{ALL}/SF	To mask protect commands
	UNPROTECT	{ALL}/SF	To mask unprotect commands

The following snippet displays a sample of the configuration settings. The usage of the `audit.conf` file is described in the comments within the `audit.conf` file.

```
#
# FPNG audit configuration file.
# This configuration file configs user audit attributes.
#
#  Due to some audit settings
# (LOAD_POLICY, DFPSHELL, UPDATE, UNINSTALL)
#  cannot get by Data Element. so these operations audit settings could be changed
#  by this configuration file.
#
# Overview:
# LOAD_POLICY: [<user/FS>, ...]              Log load policy for user.
# DFPSHELL: [<user/FS>, ...]                 Log run dfpshell, chage dfpshell passwd
operation.
#
# Format Description:
# <event>: [<user/condition> ...]
# Where,
# 'event':       one of LOAD_POLICY, DFPSHELL
#               See section above for the meaning of each manifest symbols.
# 'user':       login-id or '{ALL}' for all users in the system.
# 'conditionsk': Any combinations of letter F, S.
#               F stands for Failure,
#               S stands for Success,
#
# Examples:
# LOAD_POLICY: guest/F                For user 'guest', log failed load policy operation
# UNINSTALL: {ALL}/SF                 For all users  log 'uninstall' success or failure.
# DFPSHELL: Mary/S, herry/S           For Mary and herry log success  run dfpshell.
#
# Any characters after '#' is comments.
# This is audit default setting configuration
LOAD_POLICY: {ALL}/SF
DFPSHELL: {ALL}/SF
UPDATE: {ALL}/SF
UNINSTALL: {ALL}/SF
EXPORT: {ALL}/SF
IMPORT: {ALL}/SF
KEY_ROTATION_ADD: {ALL}/SF
KEY_ROTATION_DEL: {ALL}/SF
PROTECT: {ALL}/SF
UNPROTECT: {ALL}/SF
```

5.4 *dfp_ldap.conf* File

This *dfp_ldap.conf* configuration file configures the LDAP for managing the *dfpshell* account.

The File Protector uses LDAP for managing the *dfpshell* account. The *dfpshell* privileges can be loaded without providing the *dfpshell* password when the File Protector is connected to the LDAP server.

The *dfp_ldap.conf* file is located in the `<Installation path of the File Protector>/protegrity/defiance file protector/data` directory. An overview of each configuration setting can be found in comments of the `dfp_ldap.conf` file.

```
[ldap]
#######################################
## DFP LDAP Client configuration    ##
#######################################
#-------------------------------------------------------------
# LDAP host to connect to, default is localhost
# [example] host = localhost

# LDAP port to connect to (default 389 for TLS).
# [example] port = 389

# Use TLS when communicating with the LDAP host(yes or no)
# [example] usetls=yes

# LDAP search base that holds users, default is ou=people,dc=esa,dc=protegrity,dc=com.
# [example] user-base-dn = ou=people,dc=esa,dc=protegrity,dc=com

# timeout: communication timeout in seconds, default is 10.
# [example] timeout = 10
```

5.5 *fe.conf* File

For frequent read and write operations on FE encrypted files, the File Protector provides the redirect cache feature (High Performance Mode) to enhance the read and write operations on the FE encrypted files. This *fe.conf* configuration file enables or disables the redirect cache for the FE encrypted files.

> **Note:**
> By default, the High Performance Mode is enabled and there are three parameter settings in the `fe.conf` file which has default values as well. The three parameter settings are only available when the High Performance Mode is enabled.
>
> When the High Performance Mode is enabled, the File Protector supports application level memory map or mmap syscall read or write on the FE encrypted files.

The following snippet displays a sample of the configuration settings. The usage of the `fe.conf` file is described in the comments within the `fe.conf` file.

```
#
# Redirect Cache Setting
#
#   on - enable the redirect cache for encrypted files
#  off - disable the redirect cache for encrypted files
#
FE_REDIRECT_CACHE_SWITCH=on

#
# The Number of Running paged Threads
#
# Note:
#   The range is from 0 to 16.
#   "0" means the number of paged threads is as same as the number of CPUs.
#
PAGED_THREADS=0

#
# paged Queue Buffer Size Limit Setting (in Megabytes)
```

```
# It represents the upper limit of all buffer size in paged queue.
#
# Note:
#   The paged queue buffer size range is from 1 to 65536 (MB).
#   The higher, the more dirty pages could be processed at a time.
#   Also it could bring more pressure to system.
#
PAGED_QUEUE_BUFFER_LIMIT=16

#
# The paged Work Flow Control.
# The CPU resource usage percentage of paged daemon.
#
# Note:
#   The CPU usage range is from 0 to 100.
#   The higher, the more CPU resources utilized.
#   "0" means utilizing the availabe CPU resources.
#
PAGED_CPU_USAGE=0
```

5.5.1 Configuring the Redirect Cache

The File Protector allows you to enable and disable the Redirect Cache for File Encryption.

➤ To configure the Redirect Cache:

1. Navigate to the installed directory `\Protegrity\Defiance File Protector\data` and configure the cache for the whole system in the `fe.conf` file.
2. Run the following command to configure the cache for a specific file or directory.
   `dfp fe set`
   For more information about the **`dfp fe set`** command, refer to section *Configuring Cache Settings for Encrypted File or Directory*.

5.5.1.1 Configuring the Redirect cache for the whole system

This section describes the steps to configure the Redirect cache for the whole system.

➤ To configure the Redirect cache for the whole system:

1. Set the value of the *FE_REDIRECT_CACHE_SWITCH* attribute to *on*.
2. Set the value of the *PAGED_THREADS* attribute between *0* and *16*.
3. Set the value of the *PAGED_QUEUED_BUFFER_LIMIT* attribute between *1* and *65536 (MB)*.
4. Set the valued of the *PAGED_CPU_USAGE* attribute between *0* and *100*.
5. Run the following command to update the changes.
   `dfpadmin update`

5.5.1.2 Configuring the Redirect cache for specific encrypted files

This section describes the steps to configure the Redirect cache for specific encrypted files.

➤ To configuring the Redirect cache for specific encrypted files:

1. Ensure that the *FE_REDIRECT_CACHE_SWITCH* attribute in the `fe.conf` file is set to *on*.
2. Run the following command to configure the redirect cache on specific files or directories.

   ```
   dfp fe set [-r] [-o cache=on/off] <files/directories>
   ```

   For more information about the **`dfp fe set`** command, refer to section *Configuring Cache Settings for Encrypted File or Directory*.

5.6 *key_rotation.conf* File

The `key_rotation.conf` configuration file is used to set the parameters for scheduling the key rotation. The `key_rotation.conf` file is located in the `<Installation path of the File Protector>/protegrity/defiance file protector/data` directory.

The following snippet displays a sample of the configuration settings. The usage of the `key_rotation.conf` file is described in the comments within the `key_rotation.conf` file.

```
#
# File Protector File-Key-Rotation Configuration File
#
#####################################################################
##
## Notes:
## 1. TIME:  The time to do the key rotation
##       - Format
##             [Minute]  [Hour]  [Day]  [Month]  [Week]
##               *         *       *       *        *
##               -         -       -       -        -
##               |         |       |       |        |
##               |         |       |       |        +----- day of week (0 - 6) (Sunday=0)
##               |         |       |       +------- month (1 - 12)
##               |         |       +---------- day of month (1 - 31)
##               |         +------------ hour (0 - 23)
##               +-------------- min (0 - 59)
##
##       - options:
##           -Month: 1~12
##           -   Day: 1~31
##           - Week: 0~6   (0 means Sunday)
##           -  Hour: 0~23  (0 means 12 a.m.)
##           -Minute: 0~59
##
##     e.g.
##     30  21  *  *  *   ( means 21:30 every night )
##     0  0  *  *  1-5   ( means 00:00 pm from Monday to Friday )
##     0  6  *  *  6,0   ( means 6:00 am from Saturday to Sunday )
##
#####################################################################
##
##  [Minute]  [Hour]  [Day]  [Month]  [Week]
0  0  *  *  *

##
## Times of retrying key rotation when the expired file was busy.
## NOTE: '0' means ignore and skip
##
KEY_ROTATE_RETRY_TIMES=3

##
## The interval (in seconds) between key rotation retries.
##
KEY_ROTATE_RETRY_INTERVAL=60
```

5.7 *path_name_info.conf* File

The *path_name_info.conf* configuration file is used to configure the representation of the protected path. The *path_name_info.conf* file is located in the *<Installation path of the File Protector>/protegrity/defiance file protector/data* directory.

The following snippet displays a sample of the configuration settings. The usage of the *path_name_info.conf* file is described in the comments within the *path_name_info.conf* file.

```
#
# Path Name Information Configuration
#

[ General Settings ]
#
#  Path Name Info Type of Local Volumes
#  Options:
#    0 -- Full Path Name
#    1 -- <Volume Name> + <Path on the Volume >
#    2 -- <FPDID> + <Path on the Volume>
#
LOCAL_PATH_NAME_INFO_TYPE=2

#
#  Path Name Info Type of Networking/Remote Share Volumes
#  Options:
#    0 -- Full Path Name
#    1 -- <IP Address> + <Path on the Share Volume>
#
REMOTE_PATH_NAME_INFO_TYPE=1

[ Path Name Info Volume List ]
#
#  Allow Users to Specify the Path Name Info Type for Specific Volumes.
#  Options:
#     on -- Allowed
#    off -- Disallowed
PATH_NAME_INFO_VOLUME_LIST=off

#
#  Specify the Path Name Info Type for Specific Volumes.
#  Note:
#   * The List will be Effective Only If PATH_NAME_INFO_VOLUME_LIST was "on".
#   * The Format of Each Line Setting:
#       <Volume Name>=<Path Name Info Type>
#   * The Value of Path Name Info Types for Local Volumes:
#       0 -- Full Path Name
#       1 -- <Volume Name > + <Path on the Volume>
#       2 -- <FPDID> + <Path on the Volume>
#   * The Value of Path Name Info Types for Networking/Remote Share Volumes:
#       0 -- Full Path Name
#       1 -- <IP Address> + <Path on the Share Volume>
#   * Maximum supported 512 volumes. The other volumes besides the first 512 ones will be
ignored.
#  e.g.
#        /dev/sda2=2
#        192.168.2.100:/export/home=0
#        \Device\HarddiskVolume2=2
#        \\192.168.2.237\test=0
#

#
# The End
#
```

5.8 *policy_management_server.conf* File

The Policy management server is used to configure the PMS service related settings including the *dfpshell* timeout value.

> **Note:** The time interval is specified in the `policy_management_server.conf` file located in the `<Installation path of the File Protector>/protegrity/defiance file protector/data` directory. The default *dfpshell* timeout interval is 20 minutes.

The following snippet displays a sample of the configuration settings. The usage of the `policy_management_server.conf` file is described in the comments within the `policy_management_server.conf` file.

```
#
# Policy Management Server Configuration
#

[Server Configuration]
#
# policy management server IP address
#
SERVER_IP_ADDR=127.0.0.1

#
# policy management server listening port
#
SERVER_LISTEN_PORT=15312

#
# policy management server listening threads
#
SERVER_LISTEN_THREADS=1

#
# policy management server connection type
#  Options:
#    ssl -- use SSL connection.
#    tcp -- use encrypted TCP/IP connection.
#
SERVER_CONNECT_TYPE=ssl

#
# the parent directory where pepserver installed
# it should have "Defiance DPS" sub directory
# for example:
#     PEP_INSTALL_DIR=C:\Program Files\Protegrity
#
PEP_INSTALL_DIR=

#
# the directory where DataBase Encryption Key localed
# it should have "kekup.bin", "master.key", "repository.key" sub files
# for example:
#     DFP_DB_ENC_KEY_DIR=C:\Program Files\Protegrity\Defiance DPS\data
#
DFP_DB_ENC_KEY_DIR=


[Client Configuration]
#
# client connecting server IP address
#
CLIENT_CONNECT_IP=127.0.0.1
#
# client connecting server port
#
CLIENT_CONNECT_PORT=15312
#
# client connection type
```

```
#  Options:
#    ssl -- use SSL connection.
#    tcp -- use encrypted TCP/IP connection.
#
CLIENT_CONNECT_TYPE=ssl


#
#  dfpshell privilege timeout interval in minutes
#  0 means disable timeout check
#
PRIVILEGE_TIMEOUT_INTERVAL=20

#
# The End
#
```

5.9 Configuring Log Server Settings File

The logging section of the *pepserver.cfg* file is used to configure the Log server service-related settings.

> **Note:** The default location of the `pepserver.cfg` file is `PEP Server/defiance-core/data` directory.

The following snippet displays a sample of the logging configuration settings. The usage of the `pepserver.cfg` file is described in the comments within the `pepserver.cfg` file.

```
# ---------------------------------
# Logging configuration
# ---------------------------------
[logging]

# Logging level for pepserver application logs: OFF - No logging, SEVERE, WARNING, INFO,
CONFIG, ALL
level = ALL

# Set the output type for protections logs. Set to either tcp or stdout.
# tcp    = (default) Logs are sent to fluent-bit using tcp
# stdout = Logs are sent to stdout
output = tcp

# Fluentbit host and port values (mostly localhost) where logs will be forwarded from the
protector.
host = 127.0.0.1
port = 15780

# In case that connection to the fluentbit is lost, set how logs must be handled.
# This setting is only for the protector logs and not application logs, sent from pepserver
# drop = (default) Protector throws logs away if connection to the fluentbit is lost
# error = Protector returns error without protecting/unprotecting data if connection to the
fluentbit is lost
mode = drop

# Intervall in seconds, on how often we send logs from protector to logforwarder. ( Default 1
sec )
# It can be set to a maximum of 86400 ( i.e. 24 hours ).
#logsendinterval = 1
```

Chapter 6

Managing the *dfpshell*

The *dfpshell* provides the File Protector administrator privileges in the shell. It is a privileged mode of operations for the management of theFile Protector that requires users to log on using the *dfpshell* password.

You will be prompted to set up the *dfpshell* password, when you install the File Protector initially.

The *dfpshell* password must meet the following criteria:

- Should be a minimum 8 characters in length
- Should contain a mix of numeric, alphabetic, and printable characters

The *dfpshell* password is required for configuring File Protector protection operations. If you run the File Protector commands without the *dfpshell* privileges, then the following error message appears.

```
ERROR: file protector privilege is needed!
```

> **Note:** For more information about the commands that require *dfpshell* privileges, refer to section *File Protector Commands Overview*.

The File Protector enables you to set the *dfpshell* password, reset the password, and activate or deactivate the *dfpshell* mode.

If you want to manage a network share, then login to the *dfpshell* directly on the machine that hosts the shared directory.

The following table lists and describes the syntax of the *dfpshell* commands.

Table 6-1: dfpshell Commands

Commands	Description
`dfpshell`	Activates the *dfpshell* mode.
`dfpshell -t`	Checks if the current session has the *dfpshell* privileges. If the current process has the required privileges, then the following message appears: `Has privilege!` If the current process does not have the required privileges, then the following message appears:

Commands	Description
	`INFO: No privilege!`
`dfpshell -c`	Changes the *dfpshell* password. The `-c` option changes the privilege key and the password. The command verifies the existing key and prompts for the new key.

6.1 Changing the *dfpshell* Password

This section describes the steps to change the *dfpshell* password. You can change the *dfpshell* password at any time.

To change the *dfpshell* password:

1. Type the following command and press ENTER.
   `dfpshell -c`
   A prompt for the existing *dfpshell* password appears.
2. Enter the current *dfpshell* password and then press ENTER.
   A prompt to enter the new *dfpshell* password appears.
3. Enter the new *dfpshell* password and then press ENTER.
4. Enter the new *dfpshell* password again for verification and then press ENTER.

The *dfpshell* password is updated.

6.2 Activating the *dfpshell* Mode

This section describes the steps to activate the *dfpshell* mode. You can activate and deactivate the *dfpshell* mode based on your requirements to access its privileges.

To activate the *dfpshell* mode:

1. Type the following command and press ENTER.
   `dfpshell`
   A prompt for the *dfpshell* password appears.
2. Enter the *dfpshell* password and then press ENTER.
3. Type the following command to know the current status of the *dfpshell* privileges and then press ENTER.
   `dfpshell -t`
   The following message appears:
   `Has privilege!`

> **Note:** If you do not have the *dfpshell* privileges, then the following message appears.

```
ERROR: No privilege.
```

4. Type the following command to start a new command prompt and then press ENTER.
   ```
   dfp start -n
   ```
5. Type the following command to deactivate the *dfpshell* mode and then press ENTER.
   ```
   exit
   ```

6.3 Recovering the *dfpshell* Active Password

This section describes the steps to recover the *dfpshell* password.

To recover the *dfpshell* password:

1. Restart the system to safe mode.
2. Remove the `.syslock_v2` file from the `C:\Windows\protegrity` directory.
3. In a multi-user mode, type the following command to create the new *dfpshell* password.
   ```
   dfpshell
   ```

6.4 Managing the *dfpshell* for LDAP Users

The File Protector uses LDAP for managing the *dfpshell* account. The *dfpshell* privileges can be loaded without providing the *dfpshell* password when the File Protector is connected to the LDAP server. The *dfpshell* privileges should be added to the LDAP users permissions.

> **Note:** The File Protector supports only the Open LDAP Server, Active Directory, Novell LDAP Server, and Solaris LDAP Server.

The LDAP configuration file, `dfp_ldap.conf`, is located in the `\data` directory. The following code snippet is an example of the `dfp_ldap.conf` configuration file.

```
[ldap]
#######################################
## DFP LDAP Client configuration  ##
#######################################
#-------------------------------------------------------------
# LDAP host to connect to, default is localhost
# [example] host = localhost

# LDAP port to connect to (default 389 for TLS).
# [example] port = 389

# Use TLS when communicating with the LDAP host(yes or no)
# [example] usetls=yes

# LDAP search base that holds users, default is ou=people,dc=esa,dc=protegrity,dc=com.
# [example] user-base-dn = ou=people,dc=esa,dc=protegrity,dc=com

# timeout: communication timeout in seconds, default is 10.
# [example] timeout = 10
```

To configure LDAP to be used by the File Protector:

1. On the LDAP server that you want to configure, to connect to the File Protector, add a new businessCategory *pty_role:dfp_sec_admin* to the LDAP user *ldapu*.
2. On the File Protector, configure the LDAP server settings.
   Refer to the sample code snippet of the `dfp_ldap.conf` configuration file to configure the LDAP server settings.
3. Save your changes.
4. Type the following command and then press ENTER.
   ```
   dfpshell
   ```
5. Enter the LDAP user (*ldapu*) credentials.

> **Note:** If any error appears due to the configuration file and the `dfpshell` command fails, then run the following command to load the *dfpshell* privileges, using the *dfpshell* password.
> ```
> dfpshell -quiteload
> ```

6.5 Managing the *dfpshell* Timeout

If an active *dfpshell* is not used for a long time, then it times out and you need to login again to continue. The time interval to log out is set in the `policy_management_server.conf` configuration file located in the `..\Protegrity\Defiance File Protector\data` directory. The default *dfpshell* timeout interval is 20 minutes.

The following code snippet is an example of the configuration file `policy_management_server.conf` that allows you to configure the *dfpshell* timeout.

```
#
# dfpshell privilege timeout interval in minutes
# 0 means disable timeout check
#
PRIVILEGE_TIMEOUT_INTERVAL=20
```

➤ To configure the *dfpshell* timeout:

1. Navigate to the `policy_management_server.conf` file located in the `..\Protegrity\Defiance File Protector\data` directory.
2. Set the value of the *PRIVILEGE_TIMEOUT_INTERVAL* parameter based on your requirement.
   ```
   PRIVILEGE_TIMEOUT_INTERVAL=15
   ```
3. Restart the *policy_management_server* services.
   a. Type the following command to disable the *policy_management_server* services and then press ENTER.
      ```
      dfpadmin service pms off
      ```
   b. Type the following command to enable the *policy_management_server* services and then press ENTER.
      ```
      dfpadmin service pms on
      ```
   c. Type the following command to check the status of the *policy_management_server* services and then press ENTER.
      ```
      dfpadmin service pms status
      ```
4. Type the following command to load the *dfpshell* privileges and then press ENTER.
   ```
   dfpshell
   ```
   The timeout countdown begins. If the loaded *dfpshell* is inactive for 15 minutes, then the *dfpshell* privilege times out. If there is any activity during the configured 15 minutes, such as running a File Protector command or executing a program within the loaded *dfpshell* session, then the timeout countdown restarts.

5. Set the value of the *PRIVILEGE_TIMEOUT_INTERVAL* parameter to *0* to disable the *dfpshell* timeout, so that the *dfpshell* does not expire.

   ```
   PRIVILEGE_TIMEOUT_INTERVAL=0
   ```

Chapter 7

Licensing

The status of the Protegrity Data Security Platform License and the terms of your license agreement with Protegrity determines the File Protector features and functionality.

A Protegrity license can be in the following states:

- Valid
- Expired
- Invalid

If the license is valid, then you have all read and write permissions for all the File Protector operations.

If the license is expired or invalid, then your permissions are determined by the following points:

- The license agreement with Protegrity
- The policy enforcement and management status after the license has expired

> **Note:** For more information about licensing, refer to *Protegrity Data Security Platform Licensing Guide 9.0.0.0*.

7.1 Checking License Validity

The File Protector provides the *dfpadmin* commands to check the license validity.

➤ To check validity of the File Protector license:

1. Run the following command to verify whether the license is valid.
   ```
   dfpadmin license check
   ```
2. Press ENTER.
   - If the license of the File Protector is valid, then the following message appears.
     ```
     File Protector License is OK!
     ```

- If the license of the File Protector is invalid, then the following message appears.
  Error: File Protector license is invalid
- If the license of the File Protector is expired, then the following message appears.
  Error: File Protector license is expired

7.2 Checking License Status

This section describes how to check the status of the File Protector license.

To view the detailed status of the File Protector license:

1. Run the following command to view the status of the license.
   dfpadmin license status
2. Press ENTER.
   The following license details appear.
   - License Status
   - Valid Date
   - Last Valid Date

```
C:\Users\Administrator>dfpadmin license status
================================================
    LICENSE STATUS
------------------------------------------------
      License State : OK
         Valid Date : From 2018-05-28 10:46:21 to 2106-02-07 11:58:15
    Last Valid Date : 2018-10-09 13:29:00
```

7.3 Operations Allowed in case of Invalid or Expired File Protector License

If the File Protector license is invalid or expired, no errors are generated when you perform some file protection operations.

If the File Protector license is expired or invalid, then the following operations are allowed :

- Access protected data
- Access encrypted data
- Run delegated programs
- Delegate processes
- Unprotect protected data
- Decrypt encrypted data
- Undelegate programs or processes

7.4 Operations Denied in case of Invalid or Expired File Protector License

If the File Protector license is invalid or expired, then you cannot perform some file protection operations.

If the license of the File Protector is invalid or expired, then you cannot perform the following tasks:

- Encrypt a new file or directory

- Protect a new file or directory
- Delegate a new program, process, or user

Chapter 8

Using the Policy Management

You can create and manage the policies of the File Protector using ESA. A File Protector policy stores one or multiple data elements. Each File Protector policy is protected by a password that is defined by the Security Administrator at the ESA.

Note: You cannot use ENTER which ends the password entry.

If you entered a wrong password, then perform the following steps:

1. Press *ENTER*.
   A confirmation prompt appears for re-trying the password entry.
2. Enter the correct password.
3. Press *ENTER*.

The mismatched password entries can cause confirmation to fail. You must enter your password again in case of the mismatched password entries.

If you have multiple PEP servers in one data store or different data stores in the ESA, then you should have the same data elements with same keys unless you change the key of data element and redeploy to PEP servers.

8.1 Deploying a Policy

This section describes the steps to deploy a policy of the File Protector.

To deploy the policy:

1. On the ESA Web UI, navigate to **Policy** > **All Policies**.
   The **Deploy** pane appears.
2. Select a data store.
3. Click **Deploy**.
   An information message box confirming the deployment of the policy appears.
4. Click **Ok**.
   The message box closes and state of the policy is modified to **Deployed**.

5. Run the following command to display the product information.
   ```
   dfp info
   ```
   The following information appears:
   - Product version
   - Available policy list on the PEP server
   - Data elements of the current process

8.2 Loading a Policy

This section describes the steps to load a policy of the File Protector.

➤ To load the policy:

After deploying policy, in the protector, run the following command to load the policy in the process.

```
dfp start -p <role>@<policy>
```

or,

```
dfp start –p <policy name>
```

> **Note:** You can load multiple policies on the same terminal.

8.3 Removing the Loaded Policies

The File Protector starts when a policy is loaded on the terminal, and it stops automatically when a program does not have the loaded policy. This section describes the steps to remove the loaded policies for a program.

➤ To remove the loaded policies for a program:

1. Run the following command to close the current terminal and start a new terminal with policy not loaded .
   ```
   dfp start -n
   ```
2. Run the following command to close the policies.
   ```
   exit
   ```

Chapter 9

Commands Overview

This section provides an overview of the commonly used File Protector commands. You can run the commands and perform all the File Protector functions using the Command Line Interface (CLI).

9.1 *dfp* Commands

This section provides an overview of the commonly used *dfp* commands.

The following screen lists the basic *dfp* commands.

```
C:\Windows\protegrity>dfp
Usage:
    dfp start -p <role>@<policy>
    dfp info
    dfp proc [-l] [ <PID> ]
    dfp delegate [-f] [-o <options>] -e <program> <role>@<policy> [<passwd>]
    dfp delegate [-f] [-o <options>] [-r] -p <PID> <role>@<policy> [<passwd>]
    dfp delegate [-f] -u <username> <role>@<policy> [<passwd>]
    dfp undelegate -e <program>
    dfp undelegate [-o <options>] [-r] -p <PID>
    dfp undelegate -u <username>
    dfp delegate cleanup [-y] [path-wildcard]
    dfp delegate status
    dfp delegate help
    dfp move <src> <dest>
    dfp ac protect [-f] -d <data element> <file>
    dfp ac protect [-f] [-r] [-i] -d <data element> <folder>
    dfp ac unprotect <file>
    dfp ac unprotect [-r] <path>
    dfp ac cleanup [-y] [path-wildcard]
    dfp ac stat <file or folder>
    dfp ac status
    dfp ac help
    dfp fe stat [-l] [-r] <file or folder>
    dfp fe protect [-f] -d <data element> [-o <options>] <file>
    dfp fe protect [-f] [-r] -d <data element> [-o <options>] <folder>
    dfp fe unprotect <file>
    dfp fe unprotect [-r] <folder>
    dfp fe dump <file>
    dfp fe set -o <options> <file>
    dfp fe set [-r] -o <options> <folder>
    dfp fe help
    dfp fe krotate add [-f] [-r] [-p <policy> <-|passwd>] <path ...>
    dfp fe krotate del [-r] <path ...>
    dfp fe krotate cleanup [-y] [path-wildcard]
    dfp fe krotate status
    dfp fe krotate help
    dfp file protect [-f] -d <data element> [-o <options>] <file>
    dfp file protect [-f] [-r] -d <data element> [-o <options>] <folder>
    dfp file unprotect <file>
    dfp file unprotect [-r] <folder>
    dfp file stat [-l] <file>
    dfp file stat [-l] [-r] <folder>
    dfp file help

    dfp export [-r] -p <role>@<policy> -d <dataelement> <src> <dest folder>
    dfp import [-r] -p <role>@<policy> <src> <dest folder>
    dfp version
    dfp help
```

Figure 9-1: Basic dfp commands

The following table describes the File Protector *dfp* commands.

> **Note:** Most of the File Protector commands require *dfpshell* privileges. The *dfpshell Privilege* column mentions whether a command requires the *dfpshell* privileges.

Table 9-1: Basic Windows File Protector Commands

Command	DFPShell Privilege	Description
`dfp start -p <role>@<policy>`	No	Loads the policy keys for the user in the current process.
`dfp info`	No	Displays product information such as product version, available policy list on the PEP server, and the data elements available in the current process.
`dfp proc [-l] [ <PID> ]`	No	Displays the information of the data elements available for a specified process. By default, it is the current process.

Command	DFPShell Privilege	Description
		If you specify the **-l** option, then the detailed information on data elements such as access mask, success audit mask, failure audit mask, no access operation, and data element name with the corresponding policy is displayed. The **<PID>** option displays the data elements of the specified PID process.
dfp delegate [-f] [-o <options>]-e <program> <role>@<policy>	Yes	Delegates a program with policy that become available when the program starts. For **[-o <options>]** option, the value is *parent=<dataelement name>*.
dfp delegate [-f] [-o <options>] [-r] -p <PID> <role>@<policy>	Yes	Delegates a process with policy. For **[-o <options>]** option, the value is *always-effective*.
dfp delegate [-f] -u <username> <role>@<policy>	Yes	Delegates a user with policy which become available when the user logs in to the machine.
dfp undelegate –e <program>	Yes	Removes delegation for the program.
dfp undelegate [-o <options>] [-r] -p <PID>	Yes	Removes process delegation.
dfp undelegate -u <username>	Yes	Removes user delegation.
dfp delegate cleanup [-y] [path-wildcard]	Yes	Cleans up the invalid delegations from the Delegation list.
dfp delegate status	Yes	Displays the Delegated Program List and Delegated User List. • The Delegated Program List displays information, such as program path, delegated policy, and specified parent data element information for all delegated programs. • The Delegated User List displays the user name and delegated policy information.
dfp delegate help	No	Displays all Delegation commands. For more information about Delegation commands, refer to section *Using Delegation*.
dfp delegate import	Yes	Imports the Delegation rules. For more information about this command, refer to section *Scenarios of File Protector Backup and Restore*.
dfp delegate export	Yes	Exports the Delegation rules.

Command	DFPShell Privilege	Description
		For more information about this command, refer to section *Scenarios of File Protector Backup and Restore*.
`dfp delegate sync`	Yes	Synchronizes the delegation rules from the current delegation setting to the `delegate.db` file.
`dfp delegate update`	Yes	Updates the delegation rules from the `delegate.db` file to the current delegation setting.
`dfp move <src> <dest>`	Yes	Moves a protected file or directory. Protection settings remains the same on the `dest` file when it is moved from the source path.
`dfp ac protect [-f] -d <data element> <file>`	Yes	Protects files. The **`-d`** option lets you specify the data elements. The **`-f`** option forces protection with new data element when the file is already protected by another data element.
`dfp ac protect [-f] [-r] [-i] -d <data element> <folder>`	Yes	Protects directories. If you specify the **`-i`** option, then new unprotected sub-directories and sub-files inherit protection from the parent directory. If you specify the **`-r`** option, then directories are protected recursively. If the **`-d`** option is used, then you can choose the data elements. If you specify the **`-f`** option, then protection is forced with new data element when the file is already protected by another data element.
`dfp ac unprotect <file>`	Yes	Removes the protection permissions from the files. The required key must be available while unprotection.
`dfp ac unprotect [-r] <folder>`	Yes	Removes the protection permissions from the directories. The required key must be available while unprotection. If you specify the **`-r`** option, then the directories are unprotected recursively.
`dfp ac cleanup [-y] [path-wildcard]`	Yes	Cleans up the invalid AC protections in the AC Protections list.
`dfp ac stat <file or folder >`	Yes	Prints the protection status of files and directories.
`dfp ac status`	Yes	Displays the following information of all the protected files and directories. • Complete protection list on the system • Protection status

Command	DFPShell Privilege	Description
		• Related data element
`dfp ac help`	No	Displays all the Access Control commands and options including explanations for the commands. For more information about the Access Control commands, refer to section *Using Access Control*.
`dfp ac import`	Yes	Separately imports Access Control protection rules. For more information about this command, refer to section *Scenarios of File Protector Backup and Restore*.
`dfp ac export`	Yes	Separately exports Access Control protection rules. For more information about this command, refer to section *Scenarios of File Protector Backup and Restore*.
`dfp ac sync`	Yes	Synchronizes Access Control protection rules from current access control setting to `ac.db` file.
`dfp ac update`	Yes	Synchronizes Access Control protection rules from the current Access Control setting to the `ac.db` file.
`dfp fe stat [-l] [-r] <file or directory>`	Yes	Shows the status of the encrypted files and directories. The `-l` option lets you display status of the cache setting. If you specify the `-r` option, then the status of the recursively encrypted directories is displayed.
`dfp fe protect [-f] -d <data element> <file>`	Yes	Encrypts a file. The `-f` option forces encryption on a specified file.
`dfp fe protect [-f] [-r] -d <data element> <folder>`	Yes	Encrypts files and directories. The `-f` option forces encryption on a specified directory. The `-r` option lets you encrypt directories and files recursively.
`dfp fe unprotect <file>`	Yes	Decrypts the files. The required key must be available.
`dfp fe unprotect [-r] <folder>`	Yes	Decrypts the directories. The required key must be available. The `-r` option lets you decrypt directories and files recursively.
`dfp fe dump <file>`	Yes	Displays detailed information on the encrypted files and directories.

Command	DFPShell Privilege	Description
dfp fe set –o <option> <file>	Yes	Configures the Cache settings for an encrypted file. For **<options>**, you can set the value as *cache=on/off*.
dfp fe set [-r] –o <option> <folder>	Yes	Configures the Cache settings for the encrypted directories. For **<options>**, you can set the value as *cache=on/off*. The **-r** option lets you encrypt directories and files recursively.
dfp fe help	No	Displays all File Encryption commands.
dfp fe krotate add [-f] [-r] [-p <policy>] <path ...>	Yes	Adds key rotation configuration for the encrypted files and directories.
dfp fe krotate del [-r] <path ...>	Yes	Removes the encrypted file or directory from the key rotation configuration list.
dfp fe krotate cleanup [-y] [path-wildcard]	Yes	Removes the decrypted or deleted files or directories from the key rotation list.
dfp fe krotate status	Yes	Displays the key rotation status for the specified encrypted files.
dfp fe krotate help	No	Displays all the file encryption key rotate commands and options explanations for the commands.
dfp file protect [-f] -d <data element> <file>	Yes	Protects files, including adding Access Control and File Encryption for the files. The **-f** option forces file protection on the protected file.
dfp file protect [-f] [-r] -d <data element> <folder>	Yes	Protects directories, including adding Access Control and File Encryption for the directories. The **-f** option forces file protection on the specified protected directory. The **-r** option lets you protect directories recursively.
dfp file unprotect <file>	Yes	Unprotects files. The required key must be available.
dfp file unprotect [-r] <folder>	Yes	Unprotects directories. The required key must be available. The **-r** option lets you unprotect directories and files recursively.
dfp file stat [-l] <file>	Yes	Displays the status of the protected files.

Command	DFPShell Privilege	Description
		The `-l` option lets you display status of the cache setting.
`dfp file stat [-l] [-r] <folder>`	Yes	Displays the status of the protected directories. The `-l` option lets you display status of the cache setting. The `-r` option lets you display status of the protected directories recursively.
`dfp export [-r] -p <role>@<policy> -d <dataelement> <src> <dest folder>`	Yes	Exports files to old VPD or DFP locked files using the policy.
`dfp import [-r] -p <role>@<policy> <src> <dest folder>`	Yes	Imports old VPD or DFP locked file to FP v7.1 using the policy.
`dfp version`	No	Displays the File Protector *dfp* version.
`dfp help`	No	Displays the File Protector *dfp* commands help.

9.2 *dfpadmin* Commands

This section provides an overview of the commonly used *dfpadmin* commands.

The following screen lists the basic *dfpadmin* commands.

```
C:\Users\Administrator>dfpadmin
Usage:
    dfpadmin status
    dfpadmin update
    dfpadmin service <service name> [ on | off | status ]
    dfpadmin service all [ on | off | status ]
    dfpadmin database -o updatedb-policy-passwd -p <role>@<policy> [passwd]
    dfpadmin database -u updatedb-policy-passwd-migrate
    dfpadmin pathnameinfo [ update | status ]
    dfpadmin fpdid [ update | status ]
    dfpadmin fpdid create [ -o <output file> ] <mount point>
    dfpadmin fpdid create -o <output file> random
    dfpadmin fpdid delete <mount point>
    dfpadmin fpdid import <mount point> <import file>
    dfpadmin fpdid export <mount point> <export file>
    dfpadmin start FPNG
    dfpadmin stop FPNG
    dfpadmin module
    dfpadmin module <module name> [ plug | unplug | status ]
    dfpadmin backup <exported file>
    dfpadmin restore <imported file>
    dfpadmin version
    dfpadmin license check
    dfpadmin license status
    dfpadmin help
```

Figure 9-2: dfpadmin Help Screen

The following table describes the File Protector *dfpadmin* commands used from the File Protector command line.

> **Note:** Most of the File Protector commands require *dfpshell* privileges. The *dfpshell Privilege* column mentions whether a command requires the *dfpshell* privileges.

Table 9-2: Windows File Protector dfpadmin Commands

Command	DFPShell Privilege	Description
`dfpadmin status`	No	Displays the following status information: • modules • service • access control list • delegation list • available policies • path name info setting • `FPDID` file list • memory report • license status • file encryption-supported cipher-list about the File Protector
`dfpadmin update`	Yes	Updates all the configuration files when the new settings are applied.
`dfpadmin service <service name> [ on | off | status ]`	Yes	Starts or stops a service, or displays its status. The `<service name>` includes the following services: • pms • log • krotate **Note:** Process turned off by this command will remain off after reboot. To start the process, run the command `dfpadmin service name on`.
`dfpadmin service all [ on | off | status ]`	Yes	Starts or stops all services, or displays all statuses.
`dfpadmin database -o updatedb-policy-passwd -p <role>@<policy> [passwd]`	Yes	When you change the policy password and deploy it again in the ESA, execute this command to update the `delegate.db`, and `krotate.db` files for the policy password changes to take effect. During the FP upgrade, if the DSK is not found, then this command is used to get the encrypted password value.
`dfpadmin database -o updatedb-policy-passwd-migrate`	Yes	After upgrading the File Protector, execute this command to update the `delegate.db`, and `krotate.db` files.

Command	DFPShell Privilege	Description
dfpadmin database -o updatedb-policy-passwd-dsk	Yes	After the new DSK is deployed from the ESA, execute this command to update the *delegate.db*, and *krotate.db* files.
dfpadmin pathnameinfo [update | status]	Yes	Updates the current setting of the path name information to take effect, or displays the current setting.
dfpadmin fpdid [update | status]	Yes	Updates the *FPDID* (File Protector Device ID) list, or displays the current status of the *FPDID* setting.
dfpadmin fpdid create [-o <output file>] <mount point>	Yes	Creates the *.FileProtectorDID* file for the specified volume. ***-o <output file>*** By default, the newly created *FPDID* file is stored in the specified volume, but users can return the *FPDID* as output to a specified file.
dfpadmin fpdid create -o <output file> random	Yes	Creates the *FPDID* file for a random device. The output file must be specified. The random device means an unspecified import device to which a random *FPDID* file for a volume is assigned. The randomly created *FPDID* file is used to import to any volume.
dfpadmin fpdid delete <mount point>	Yes	Deletes the *.FileProtectorDID* file from the specified volume.
dfpadmin fpdid import <mount point> <import file>	Yes	Imports the *.FileProtectorDID* file to the specified volume. If the *.FileProtectorDID* file exists, then it is replaced by the *.FileProtectorDID* file.
dfpadmin fpdid export <mount point> <export file>	Yes	Exports the *.FileProtectorDID* file to the specified volume.
dfpadmin start FPNG	Yes	Loads the File Protector modules and starts the services.
dfpadmin stop FPNG	Yes	Stops the File Protector services and unloads the modules.
dfpadmin module	No	Displays information about the kernel modules.
dfpadmin module <module name> [plug | unplug | status]	Yes	The ***Plug*** option creates all the dependencies of the kernel module. The ***Unplug*** option removes all the dependencies of the kernel module. The ***Status*** option displays all the information for the kernel module.

Command	DFPShell Privilege	Description
`dfpadmin backup <exported file>`	Yes	Exports the Access Control settings and delegates the settings to the specified destination file.
`dfpadmin restore <imported file>`	Yes	Imports the Access Control setting and delegates the settings from the destination file.
`dfpadmin version`	No	Displays the File Protector *dfpadmin* version.
`dfpadmin license check`	No	Checks the validity of the File Protector license.
`dfpadmin license status`	No	Displays information about the status of the File Protector license. The following items are listed: • License Status • License State • Valid Date • Last Valid Date
`dfpadmin help`	No	Displays the help for the File Protector *dfpadmin* commands.

Chapter 10

Using the File Protector

This section describes about the usage of the supported features in the File Protector.

The File Protector supports the following features:

- *Access Control*
- *File Encryption*
- *Delegation*
- *Key Rotation*
- *Audit Logging*

10.1 Using Access Control

The File Protector provides the Access Control feature to protect files and directories. This feature prevents unauthorized access to files and directories containing sensitive data. The contents of the files and directories are visible to the authorized users or programs that have access to load a policy or delegated with the required policy.

The File Protector adds an additional layer of security on top of the native file system permissions that you configure.

> **Caution:** Some program files are critical for the File Protector to function accurately and must not be protected. Avoid protecting the File Protector program installation directory files and Windows system files. Do not protect or modify these critical files in any way.

10.1.1 Protecting a File

This section describes the steps to protect a file containing sensitive information.

Before you begin
Ensure that the following prerequisites are met:

- The *dfpshell* privileges are available.
- The required policy is deployed.

➤ To protect a file:

1. Run the following command to load the required policy.
   dfp start –p policy
   A prompt for the policy password appears.
2. Enter the policy password.
3. Create a sample text file using a text editor.
4. Save the sample file and exit from the text editor.
5. Run the following command to protect the sample file.
   dfp ac protect [-f] -d <data element> <file>
   The File Protector protects the file with the specified data element and the following message appears.
   Protecting path <file path> by data element <data element> is succesful!
6. Run the following command to view the contents of the file.
   type <file>
   You can access the protected files based on the usage of the data element of an access mask.

   > **Note:** If the required access mask is not available then the following message appears.
   > *Permission Denied*

7. Run the following command to access a new shell with no data elements.
   dfp start -n
8. Run the following command to ensure that no data elements are loaded in the process.
   dfp info
9. Run the following command to view the contents of the file.
   type <file>
   As no data elements are loaded in the process, the following message appears.
   Permission Denied
10. Run the following command to verify the protection status of the file.
    dfp ac stat <file>
    The following message appears:
    Protected by <dataelement name>

    > **Note:** If a file is not protected, then the following message appears.
    > *Not Protected*

11. Run the following command to view the current protection status of the file.
    dfp ac status
    This following results appear:
    - Complete protection list on the system
    - Protection status
    - Related data element

The result of the ***dfp ac status*** command appears.

```
PS C:\Users\Administrator> dfp ac status
Enter dfpshell Pass Phrase: **************
Access Control List:
  ACTIVE    1: T:\Test_backuprestore\HOME\Fe_AC_dir\sub_dir1  <protected by aes1-rcwd>
```

12. Run ***exit*** to return to the shell.

10.1.2 Unprotecting a Protected File

This section describes the steps to unprotect a protected file.

Before you begin
Ensure that you have the *dfpshell* privileges available.

To unprotect a protected file:

1. Run the following command to revert a file to its unprotected state.
   dfp ac unprotect <file>
2. Press Enter.
   The following message appears.
   Removing the protection on path <file path> is succesful!

   After the file is unprotected, you can access that file based on the OS permissions.

10.1.3 Protecting a Directory

The Access Control feature provides options for protecting the directories containing sensitive information, recursively.

Before you begin
Ensure that the following prerequisites are met:

- The *dfpshell* privilege is available.
- The required policy is deployed.

To protect a directory:

1. Run the following command to load the required policy.
   dfp start –p policy
   A prompt for the policy password appears.
2. Enter the policy password.
3. Create a sample directory.
4. Run the following command to protect the sample directory.
   dfp ac protect [-f] [-r] [-i] -d <data element> <directory>
   The File Protector protects the directory recursively with the specified data element and the following message appears.

```
Recursively protecting path <directory path> by data element <data element> is
succesful!
```

5. Run the following command to view the contents of the directory.
   ```
   dir <directory>
   ```
6. Run the following command to verify the protection status of the directory.
   ```
   dfp ac stat <directory>
   ```
   The following message appears:
   ```
   Protected by <dataelement name>
   ```

> **Note:** If a directory is not protected, then the following message appears.
> ```
> Not Protected
> ```

7. Run the following command to view the current protection status of the directory.
   ```
   dfp ac status
   ```
   The result of the `dfp ac status` command appears.
   ```
   PS C:\Users\Administrator> dfp ac status
   Enter dfpshell Pass Phrase: **************
   Access Control List:
     ACTIVE    1: T:\Test_backuprestore\HOME\Fe_AC_dir\sub_dir1  <protected by aes1-rcwd>
   ```
8. Run `exit` to return to the shell.

10.1.4 Unprotecting a Protected Directory

This section describes the steps to unprotect a protected directory.

Before you begin
Ensure that you have the *dfpshell* privileges available.

To unprotect a protected directory:

1. Run the following command to revert a protected directory to its unprotected state.
   ```
   dfp ac unprotect [-r] <directory>
   ```

> **Note:** Unprotecting a directory (even when the protection is inherited), unprotects only the specified directory.
> To apply the unprotect operation recursively, use the `-r` option.

2. Press Enter.
   The following message appears.
   ```
   Recursively removing the protection on path <directory path> is succesful!

   Recursively removing the protection on path <file path> is succesful!
   ```

   The File Protector stops protecting the directory, making it visible to anyone who has the OS permissions.

10.1.5 Cleaning Up the Inactive AC Protection List

The File Protector provides options for cleaning up invalid AC protections in the status list. If any protect entity is deleted, then that entity is displayed as **INACTIVE** status.

Before you begin
Ensure that the following prerequisites are met:

- The *dfpshell* privileges are available.
- The required policy is deployed.

To clean up an invalid AC protection:

1. Run the following command to clean up the invalid AC protections.
   dfp ac cleanup
2. Press Enter.
   The command only cleans up the invalid AC protections, such as the *INACTIVE AC* protections list and the files or directories that have non-existent AC protections.
   The following snippet displays the sample result of the ***dfp ac cleanup*** command.

```
[root@labrh7 mount]# dfp ac cleanup
Enter dfpshell Pass Phrase: *
INFO: Acess Control Cache Cleaned Up Successfully !!
```

10.1.6 Moving Protected and Encrypted Files and Directories

To move a file or directory without losing protection and encryption, use ***dfp move*** command. This operation needs different permissions on source and destination.

The following tables provide more details about the ***dfp move*** command.

> **Note:** Moving a file or directory using the ***mv*** system command will result in the loss of protection and encryption.

Table 10-1: Source and Destination in different mount points

Source					**Target Existing**				
Status	Permission Needed				Status	Permission Needed			
	Read	Write	Delete	Create		Read	Write	Delete	Create
Not Protected					Not Protected				
Not Protected					Protected				
Protected	R		D		Protected		W		
Protected	R		D		Not Protected		W		
Source					**Target Not Existing**				
Status	Permission Needed				Status	Permission Needed			
	Read	Write	Delete	Create		Read	Write	Delete	Create
Not Protected					Not Protected				
Not Protected					Protected		W		C
Protected	R	D			Protected		W		C
Protected	R	D			Not Protected				

10.1.7 Managing the Encryption Output Setting

The Output Setting is only available for FE encrypted files or directories, for example, FE encrypted only, or AC plus FE encrypted files or directories, and the data element loaded. The Output Setting is used to specify how to output the content of those files that are FE encrypted, according to the Access Mask of data element. You can specify the Output Setting of data element in ESA by selecting either one of these - Exception and Protected Value.

Exception returns `Permission Denied` message when with data element to read files that are both FE encrypted and AC protected.

Protected Value returns ciphertext when the required data element is available to read FE encrypted, or AC plus FE protected files.

The Output Setting is available only when the Read access of Access Mask is not selected. If the Read access in Access Mask is not selected, then by default, it will be the Exception.

There are three scenarios for specifying the Output Setting:

1. When you do not select Read but select Write in Access Mask, there is only Exception type to select in Output Setting.
2. When you do not select Read and Write in Access Mask, you can select Exception or Protected Value in Output Setting.
3. If you do not have Read and Write access on the protected data file, then you cannot operate on the readable ciphertext of files and save the changes back to the files. This ensures the data integrity of files by preventing the files from becoming corrupt with wrong data.

The following table describes the scenarios of returns based on Output Setting:

Table 10-2: Returns on protected and encrypted files based on Output Setting

Access specified in Access Mask	Specified Output Setting	With data element to operate…	Protected and Encrypted File
Read	N/A	Read	Success
No Read	Exception		Permission Denied
No Read and No Write	Exception		Permission Denied
	Protected Value		Ciphertext

Table 10-3: Returns on encrypted files based on Output Setting

Access specified in Access Mask	Specified Output Setting	With data element to operate…	Protected and Encrypted File
Read	N/A	Read	Plaintext
No Read and No Write	Exception		Plaintext
	Protected Value		Ciphertext

If you specify Protected Value as Output Setting, then with data element you can back up the ciphertext of your FE encrypted files. You can create policy and role by adding data elements with no Read and Write access, configuring Protected Value as the Output Setting in the data elements. Finally, you can load the policy with the specified data elements, FE encrypt your files with the relevant commands, and back up the files.

Caution: You can regularly backup the FE protected files and directories. As part of this task, you can deploy a policy, in the ESA. You must ensure that for a particular policy, under Permissions, clear the Unprotect, Protect, and Re-Protect check boxes. Also, under the Output tab, select Protected Value.

10.2 Using File Encryption

The File Protector provides a File-level encryption method for files and directories that include highly sensitive information. You can encrypt and decrypt your files and directories recursively.

The File Encryption feature in the File Protector supports 3DES and AES (128-bit and 256-bit) encryption. The following file systems are supported on the Windows platform:

- NTFS
- FAT32

The File Encryption feature operates on top of the underlying file system to encrypt or decrypt the files and directories transparently using the standard ciphers. The user, program, or process that have the required policy loaded, can access the encrypted files. If the policy is not loaded, then the user, program, or process get a `Permission denied` error message on accessing the encrypted files.

For more information about loading a policy, refer to the section *Deploying a Policy*.

The File Encryption feature can be used with the Access Control feature to implement encryption and access control together on your files and directories.

> **Note:** The *dfpshell* privilege is required to execute the encryption and decryption commands.

10.2.1 Understanding the Encrypted Files Permissions

This section provides information about the permissions for file encryption.

When you only encrypt files and directories, you will have the permissions described in the following table:

Table 10-4: Permissions for encrypted files or directories

Open...	Permission	Encrypted File	Encrypted Directory
With Data Element loaded	Read	Yes	Yes
	Write	Yes	Yes
	Delete	Yes	Yes
Without Data Element loaded	Read	Ciphertext	Yes
	Write	No	Yes
	Delete	Yes	No

10.2.2 Encrypting a File

This section describes the steps to encrypt a file containing sensitive information.

Before you begin

Ensure that the following prerequisites are met:

- The *dfpshell* privileges are available.
- The required policy is deployed.

To encrypt a file:

1. Run the following command to load the required policy.
   ```
   dfp start -p policy
   ```
2. Enter the policy password.
3. Create a sample text file using a text editor.

4. Save the sample file and exit from the text editor.
5. Run the following command to encrypt the sample file.
   dfp fe protect [-f] [-o cache=on/off] –d <data element> <file>

   > **Caution:** Ensure that you avoid encrypting binary files as they might become unusable for execution.

   The File Protector encrypts the file with the specified data element and the following message appears.
   Encrypting path <file path> by data element <data element> is succesful!
6. Run the following command to view the contents of the file.
   type <file>
   The contents of the file are visible in cleartext format since the policy is loaded.
7. Run the following command to access a new terminal with no data elements.
   dfp start -n
8. Run the following command to ensure that no data elements are loaded in the process.
   dfp info
9. Run the following command to view the contents of the file.
   type <file>
   The contents of the file appears in ciphertext format as no data elements are loaded in the process.
10. Run the following command to verify the status of the encrypted file and cache setting.
    dfp fe stat [-l] <file>
    The following message appears:
    INFO:<file>: encrypted by <dataelement name>, cache setting=<ON/OFF>.
11. Run the following command for detailed information about the encrypted file.
    dfp fe dump <file>
    The result of the ***dfp fe dump*** command appears.

```
PS C:\Users\Administrator>dfp fe dump test.txt
Enter dfpshell Pass Phrase: **************
File Name: C:\Users\Administrator\test.txt
File Type: File Protector Encrypted File Format
File Version: 9
File Size: 8208
Header Size: 720
Ext Address: 0
Ext Size: 0
Data Address: 8192
Data Size: 7
Padding Size: 9
Encryption Time: 2021-12-21 10:53:20
Flags: 0x1
Data Element Name: aes1-rcwd
Data Element Key ID: 135
Redirect Cache Status: ON
```

12. View the file in any text editor.
    The contents of the file appears in ciphertext format indicating that the file is encrypted.
13. Run ***exit*** to return to the terminal.

10.2.3 Decrypting an Encrypted File

This section describes the steps to decrypt an encrypted file.

Before you begin
Ensure that you have the *dfpshell* privilege.

To decrypt an encrypted file:

1. Run the following command to revert a file to its decrypted state.
   ```
   dfp fe unprotect <file>
   ```
2. Press Enter.
   The following message appears.
   ```
   Removing the encryption on path <file path> is succesful!
   ```

   When a file is decrypted, you can view the contents of the file in cleartext format.

10.2.4 Encrypting a Directory

The File Encryption feature provides options for encrypting the directories containing sensitive information, recursively.

Before you begin
Ensure that the following prerequisites are met:

- The *dfpshell* privileges are available.
- The required policy is deployed.

To encrypt a directory:

1. Run the following command to load the required policy.
   ```
   dfp start –p policy
   ```
   A prompt for the policy password appears.
2. Enter the policy password.
3. Create a sample directory.
4. Run the following command to encrypt the sample directory.
   ```
   dfp fe protect [-f] [-r] [-o cache=on/off] –d <data element> <directory>
   ```
   The File Protector encrypts the directory recursively with the specified data element. The following message appears.
   ```
   Encrypting path <directory path> by data element <data element> is succesful!
   ```
5. Run the following command to view the contents of the directory.
   ```
   dir <directory>
   ```

   Note: The `.dfplock` file is the symbol for an FE encrypted directory and is created every time the directory is encrypted using FE.
   - The `.dfplock` file is displayed only when the policy is not loaded.
   - The `.dfplock` file cannot be normally deleted, even if you have the *dfpshell* privilege.
   - When the directory is unprotected, the `.dfplock` file is automatically deleted.
6. Run the following command to verify the status of the encrypted directory and cache setting.
   ```
   dfp fe stat [-l] <directory>
   ```
   The following message appears:
   ```
   INFO:<directory>: encrypted by <dataelement name>, cache setting=<ON/OFF>.
   ```

7. Run the following command for detailed information about the encrypted directory.
   dfp fe dump <directory>
   The result of the ***dfp fe dump*** command appears.

```
PS C:\delegate> dfp fe dump .\foldfe
Enter dfpshell Pass Phrase: **************
               File Name: C:\delegate\foldfe
               File Type: File Protector Encrypted File Format
            File Version: 9
               File Size: 8208
             Header Size: 720
             Ext Address: 0
                Ext Size: 0
            Data Address: 8192
               Data Size: 12
            Padding Size: 4
         Encryption Time: 2018-09-03 23:56:17
                   Flags: 0x1
       Data Element Name: aes1-rcwd
     Data Element Key ID: 2
   Redirect Cache Status: ON
```

8. Run ***exit*** to return to the terminal.

10.2.5 Decrypting an Encrypted Directory

This section describes the steps to decrypt an encrypted directory.

Before you begin
Ensure that you have the *dfpshell* privileges.

➤ To decrypt an encrypted directory:

1. Run the following command to revert an encrypted directory to its decrypted state.
   dfp fe unprotect [-r] <directory>
2. Press Enter.
   The following message appears.
   Recursively removing the encryption on path <directory path> is succesful!

   Recursively removing the encryption on path <file path> is succesful!

   The encrypted directory is decrypted.

10.2.6 Configuring Cache Settings for Encrypted File or Directory

You can configure the Redirect Cache File System (RCFS) settings for encryption.

➤ To configure the Cache settings for an Encrypted File or Directory:

1. Run the following command to enable the cache.
   dfp fe set [-r] -o cache=on <file or directory>
   Where:

`-o` is used to enable or disable the cache.

`-r` is used to encrypt directories with FE recursively.

2. Press **ENTER**.
   A prompt for the *dfpshell* appears.
3. Enter the *dfpshell* password.
4. Press **ENTER**.

> **Note:** By default, the cache setting is enabled. Run the command `fp fe set [-r] -o cache=off <file or directory>` to disable the cache.

10.2.7 Configuring the Disallow Encryption Configuration File

The Disallow Encryption Configuration File allows you to list the specific files and directories that should be excluded from the encryption process.

The Disallow Encryption Configuration file is located at the following installed location: `/protegrity/fileprotector/data/fe_disallow.conf`.

The system files are added to the Disallow Encryption Configuration file to prevent encryption of the system files by default.

The following snippet provides an overview of each of the configuration settings. The overview of the configuration settings are provided in the `fe_disallow.conf` file:

```
#
# File Protector file encryption disallow configuration file.
# The following paths are disallowed to protect by 'dfp fe protect'.
#
# e.g.
#    C:\Windows
#    C:\Windows\System32
#
C:\Windows
```

> **Caution:** Ensure that you do not remove the default system file paths.
>
> If you do so, then the system files become available for encryption by File Protector commands, and may disrupt the operation of your system.

10.2.8 Using Access Control on Encrypted Files

This section describes the process of using Access Control on encrypted files and directories.

10.2.8.1 Understanding the Encrypted and Protected Files Permissions

Access Control checks the permissions on encrypted files and directories. To control access on encrypted files and directories, you need to use `dfp ac protect` for your encrypted files and directories.

When you encrypt and protect files and directories, you have the permissions described in the following table:

Table 10-5: Permissions for encrypted and protected file or directories

Open...	Permission	Encrypted and Protected File	Encrypted and Protected Directories
With Data Element loaded	Read	Depends on the Access Mask of Data Element	Depends on the Access Mask of Data Element
	Write		

Open...	Permission	Encrypted and Protected File	Encrypted and Protected Directories
	Delete		
Without Data Element loaded	Read	Permission Denied	Permission Denied
	Write	Permission Denied	Permission Denied
	Delete	Permission Denied	Permission Denied

The protection data element and encryption data element should be the same.

10.2.8.2 Encrypting and Protecting a File

This section describes the steps to protect an encrypted file containing sensitive information.

Before you begin
Ensure that the following prerequisites are met:

- The *dfpshell* privileges are available.
- The required policy is deployed.

To encrypt and protect a file:

1. Type the following command to load the required policy and then press ENTER.
   ```
   dfp start –p policy
   ```
   A prompt for the policy password appears.
2. Type the policy password and then press ENTER.
3. Create a sample file containing some data using a text editor.
4. Save the sample file and exit from the text editor.
5. Type the following command to encrypt the sample file and then press ENTER.
   ```
   dfp fe protect –d <data element> <file>
   ```
   > **Caution:** Ensure that you avoid encrypting binary files as they might become unusable if encrypted.

   The File Protector encrypts the file with the specified data element and the following message appears.
   ```
   Encrypting path <file path> by data element <data element> is succesful!
   ```
6. Type the following command to protect the encrypted file and then press ENTER.
   ```
   dfp ac protect –d <data element> <file>
   ```
   The File Protector protects the encrypted file with the specified data element and the following message appears.
   ```
   Protecting path <file path> by data element <data element> is succesful!
   ```
7. Type the following command to view the contents of the file and then press ENTER.
   ```
   type <file>
   ```
   The contents of the file are visible in cleartext format since the policy is loaded.
   > **Note:** If the data elements are loaded in the process and you have the read, write, and delete permissions, then you can read, write, and delete the protected file.
8. Type the following command to access a new terminal with no data elements and then press ENTER.
   ```
   dfp start -n
   ```
9. Type the following command to ensure that no data elements are loaded in the process and then press ENTER.

```
dfp info
```

10. Type the following command to view the contents of the file and then press ENTER.
    ```
    type <file>
    ```
    As no data elements are loaded in the process, the following error message appears.
    ```
    Permission Denied
    ```

10.2.8.3 Encrypting and Protecting a Directory

This section describes the steps to protect an encrypted directory containing sensitive information, recursively.

Before you begin
Ensure that the following prerequisites are met:

- The *dfpshell* privileges are available.
- The required policy is deployed.

To encrypt and protect a directory:

1. Create a sample directory.
2. Type the following command to encrypt the sample directory and then press ENTER.
   ```
   dfp fe protect [-r] –d <data element> <directory>
   ```
   The File Protector encrypts the required directory with the specified data element and the following message appears.
   ```
   Encrypting path <directory path> by data element <data element> is succesful!
   ```
3. Type the following command to protect the encrypted directory and then press ENTER.
   ```
   dfp ac protect [-r] –d <data element> <directory>
   ```
   The File Protector protects the encrypted directory with the specified data element and the following message appears.
   ```
   Protecting path <directory path> by data element <data element> is succesful!
   ```
4. Create sample files and directories within the protected directory.
5. Type the following command to view the contents of the directory and then press ENTER.
   ```
   dir <directory>
   ```
6. You can edit and delete the files and directories as per the access mask permissions of the data element.
7. Type the following command to access a new terminal with no data elements and then press ENTER.
   ```
   dfp start -n
   ```
8. Type the following command to ensure that no data elements are loaded in the process and then press ENTER.
   ```
   dfp info
   ```
9. Type the following command to view the contents of the directory and then press ENTER.
   ```
   dir <directory>
   ```
   The following message appears.
   ```
   Permission denied.
   ```

10.2.8.4 Encrypting and Protecting a File using the *dfp file* Command

This section describes the steps to protect an encrypted file containing sensitive information using the *dfp file* command.

Before you begin
Ensure that the following prerequisites are met:

- The *dfpshell* privileges are available.
- The required policy is deployed.

To protect an encrypted file:

1. Type the following command to load the required policy and then press ENTER.
   ```
   dfp start –p policy
   ```
   A prompt for the policy password appears.
2. Type the policy password and then press ENTER.
3. Create a sample file containing data using a text editor.
4. Save the sample file and exit from the text editor.
5. Type the following command to protect the sample file and then press ENTER.
   ```
   dfp file protect -d <data element> <file>
   ```
   The File Protector protects the required file with the specified data element and the following message appears.
   ```
   Path <file path> was succesfully protected by data element <data element> <access control, encryption>!
   ```
6. Type the following command to view the contents of the file and then press ENTER.
   ```
   type <file>
   ```
   The contents of the file are visible in cleartext format since the policy is loaded.

   > **Note:** If the data elements are loaded in the process and you have the read, write, and delete permissions, then you can read, write, and delete the protected file.
7. Type the following command to access a new terminal with no data elements and then press ENTER.
   ```
   dfp start -n
   ```
8. Type the following command to ensure that no data elements are loaded in the process and then press ENTER.
   ```
   dfp info
   ```
9. Type the following command to view the contents of the file and then press ENTER.
   ```
   type <file>
   ```
   As no data elements are loaded in the process, the following error message appears.
   ```
   Permission Denied
   ```

10.2.8.5 Encrypting and Protecting a Directory using the *dfp file* Command

This section describes the steps to protect an encrypted directory containing sensitive information using the *dfp file* command.

Before you begin
Ensure that the following prerequisites are met:

- The *dfpshell* privileges are available.
- The required policy is deployed.

To protect an encrypted directory:

1. Create a sample directory.
2. Type the following command to protect the sample directory and then press ENTER.
   ```
   dfp file protect [-f] [-r] -d <data element> <directory>
   ```
   The File Protector protects the required directory with the specified data element and the following message appears.
   ```
   Path <directory path> was succesfully protected by data element <data element>
   <access control, encryption>!
   ```
3. Create sample files and directories within the protected directory.
4. Type the following command to view the contents of the directory and then press ENTER.
   ```
   dir <directory>
   ```
5. You can edit and delete the files and directories as per the access mask permission of the data element.
6. Type the following command to access a new terminal with no data elements and then press ENTER.
   ```
   dfp start -n
   ```
7. Type the following command to ensure that no data elements are loaded in the process and then press ENTER.
   ```
   dfp info
   ```
8. Type the following command to view the contents of the directory and then press ENTER.
   ```
   dir <directory>
   ```
   The following message appears.
   ```
   Permission denied.
   ```

10.3 Using Delegation

The ability to authorize programs with a given policy is known as delegation. Using the File Protector, you can grant access to the protected files by loading the required policy. When a process or program is delegated, it can perform security operations on the data as it loads the policy automatically.

10.3.1 Delegating and Undelegating a Program

When a delegated program starts, the policy becomes available for the executable process. Any child process, which is created by a delegated program, inherits policies from the parent process.

After delegation, whenever the program starts, it automatically accesses the data elements to access the protected files. The only access restrictions are the standard system permissions.

> **Note:** The program delegation uses the absolute path of the binaries. If you change the absolute path for any delegated program, then the program delegation is lost and you must reconfigure the delegation.

10.3.1.1 Delegating a Program

This section describes the steps to delegate a program. Application binaries or executables can be delegated using the `dfp delegate` command.

Before you begin
Ensure that the following prerequisites are met:

- The *dfpshell* privileges are available.
- The required policy is deployed.

To delegate a program:

1. Run the following command to configure program delegation.
   ```
   dfp delegate [-f] -e <program> <role@policy>
   ```
   where:
   - `-f` forces delegation of the policy to the running process ID, if the process has already been delegated
   - `-e` specifies the program that needs to be delegated

   Note: The programs need to be restarted after running the `dfp delegate` command.
2. Press ENTER.
   The program is delegated.

10.3.1.2 Undelegating a Program

You can undelegate a delegated program using the `dfp undelegate` command.

Before you begin
Ensure that the *dfpshell* privileges are available.

To undelegate a program:

1. Run the following command to undelegate a program.
   ```
   dfp undelegate -e <program>
   ```
2. Press ENTER.
   The previously delegated program is undelegated.

   Note: The running instance of a delegated program continues to access the protected files until it is restarted.

10.3.2 Delegating and Undelegating a Process

The running processes can be delegated with data elements from a specific policy, so that it can access protected and encrypted directories or files. This type of configuration is useful for running applications or services.

Note: The delegated running process ID (PID) loses its access to the data elements when the processes end or restart.

10.3.2.1 Delegating a Process

You can delegate a process using the *<PID>* values of the running process and associating a policy with it for enforcing a delegation.

Before you begin
Ensure that the following prerequisites are met:

- The *dfpshell* privileges are available.
- The required policy is deployed.

To delegate a process:

1. Run the following command to enforce delegation for the process referred by the *<PID>*.
   ```
   dfp delegate [-f] [-r] -p <PID> <role>@<policy>
   ```
   where:
   - `-f` - Forces delegation of the policy to the running *PID*, if the process has already been delegated
   - `-r` - Provides recursive delegation of child processes
   - `-p` - Specifies the *PID* that needs to be delegated
2. Press ENTER.
   The process is delegated.

10.3.2.2 Undelegating a Process

You can undelegate a process using the *<PID>* values of the running process.

Before you begin
Ensure that the *dfpshell* privileges are available.

To undelegate a process:

1. Run the following command to undelegate a process.
   ```
   dfp undelegate [-r] -p <PID>
   ```
2. Press ENTER.
   The process is undelegated.

10.3.3 Delegating and Undelegating a User

The *OS* users can be delegated with data elements from a specific policy, so that the newly created processes for the *OS* users can access the protected and encrypted directories or files. This type of configuration is useful for the specific *OS* user to get permissions, such as when the *OS* user logs into the system, starts applications or services, that the *OS* user is configured to run.

When you are delegating the *OS* user, a warning message appears that prompts you to continue or abort the delegation. The warning message states the risk associated with user delegation.

The following code snippet displays the warning message related to the delegation of the *OS* user.

```
C:\Users\Administrator>dfp delegate -u new_user policy_fe
WARNING: once you delegate the user, Administrator users like administrator or root would be
able to execute a program/process via the delegated user to get the delegated poli
cy permissions.
Aware of the risk, do you still want to continue? [Y|N] (Y)
Y
Enter the policy password : **********
Delegate user <LAB12KR2-0\new_user> by policy <policy_fe> successfully!
```

10.3.3.1 Delegating a User

You can delegate the system users using the `dfp delegate` command. The policy is loaded automatically for the delegated users after login.

Before you begin
Ensure that the following prerequisites are met:

- The *dfpshell* privileges are available.
- The required policy is deployed.

To delegate a user:

1. Run the following command to delegate a user.
   ```
   dfp delegate [-f] -u <username> <role>@<policy>
   ```
   where:
   - `-f` - Forces delegation of the policy to the user, if the user is already delegated. After you delegate the user, the newly created processes for the user include the specified policy.
   - `-u` - Specifies the username that needs to be delegated

   > **Note:** The user needs to re-login after executing the `dfp delegate` command, as data elements are loaded only upon re-login.
   >
   > If you are delegating the system user, then the user should exist.

2. Press ENTER.
   The user is delegated.

10.3.3.2 Undelegating a User

You can undelegate a delegated user using the `dfp undelegate` command.

Before you begin
Ensure that the *dfpshell* privileges are available.

To undelegate a user:

1. Run the following command to undelegate a user.
   ```
   dfp undelegate -u <username>
   ```
2. Press ENTER.
   The previously delegated user is undelegated.

> **Note:** After undelegating the user, if you do not terminate or close the previous process, then all the previous processes linked to the user continue to retain the delegated data elements.
>
> For example, a delegated user logs on to a machine where the current processes have the delegated data element. If you undelegate the user without terminating or closing the current login, then the current login continues to remain delegated.

10.3.4 Reviewing the Delegation Status

The **`delegate status`** command lists the active delegated programs, users, and processes, and specifies the policy in the delegated list.

Before you begin
Ensure that the *dfpshell* privileges are available.

To view the status of the delegation:

1. Run the following command to view the status of the delegation.
   `dfp delegate status`
2. Press ENTER.
   The following code snippet displays the result of the **`delegate status`** command.

```
C:\Users\Administrator>dfp delegate status
Delegated Program List:
  ACTIVE    1: C:\Windows\System32\notepad.exe  <policy_fe>
Delegated User List:
  ACTIVE    1: LAB12KR2-0\new_user  <policy_fe>
  ACTIVE    2: LAB12KR2-0\shrima  <policy1>
```

> **Note:** If any program, user, or process is not delegated, then the empty status message appears.
>
> ```
> C:\Users\Administrator>dfp delegate status
> Enter dfpshell Pass Phrase: *
> Delegated Program List: empty
> Delegated User List: empty
> ```

10.3.5 Removing an Invalid Delegation

This section describes the steps to remove the invalid delegations from the delegation cache, such as the inactive delegations list and the files or directories that have invalid delegations.

Before you begin
Ensure that the *dfpshell* privileges are available.

To clean up the invalid delegation:

1. Run the following command to clean up the invalid delegation.
   `dfp delegate cleanup`
2. Press ENTER.
   The following code snippet displays the result of the **`delegate cleanup`** command.

```
[root@labcos64-64 -]# dfp delegate cleanup
Enter dfpshell Pass Phrase: *
INFO: Delegation Cache Cleaned Up Successfully !!
```

10.4 Using Key Rotation

The File Protector encryption feature provides the key rotation functionality, which automatically replaces the encryption key for the specified encrypted files. The Key rotation feature re-encrypts the encrypted files with the new active key using the same data element.

You can specify the encrypted files or directories for key rotation and configure the time interval after which the key must be rotated.

10.4.1 Understanding the Key Rotation Status

The following table describes different statuses of Key Rotation:

Status	Description
Valid	The encryption key is valid and key rotation is not required.
Invalid	The status lists *Invalid* for the encrypted file or directory in the following cases: • The user specifies the key rotation configuration for an encrypted file or directory, and then tries to force encryption on the encrypted file or directory with another data element. • The user decrypts or deletes the encrypted file or directory.
Expired	The encryption key has expired and rotation is required.
Rotating	The file is processing key rotation.

10.4.2 Adding the Key Rotation Configuration

This section describes the steps to add the key rotation configuration for encrypted files or directories.

Before you begin
Ensure that the *dfpshell* privileges are available.

To add Key rotation configuration for encrypted files or directories:

1. Run the following command to add key rotation configuration for encrypted files or directories.
   ```
   dfp fe krotate add [-f] [-r] [-p <policy>] <path ...>
   ```
   Where:
   - `-f` - Adds new key rotation configuration for the encrypted files or directories (optional).
   - `-r` - Adds key rotation configuration recursively for the specified encrypted directories (optional).
   - `-p <policy>` - Specifies the policy name.
   - `<path ...>` - Specifies the path of the encrypted files or directories.

   > **Note:**
   > You can only add the encrypted files or directories for key rotation.

   A prompt for the policy password appears.
2. Enter the policy password.

10.4.3 Configuring Key Rotation

This section enables you to configure key rotation. You can select to rotate the keys at a specific time interval.

For example, consider that you have a file or directory encrypted by data element ***de1*** in the ***policy1***. For this, you might want to configure the key rotation to occur at 9 am every day. If the encrypted file is busy, then you can attempt the key rotation four times every 50 seconds.

Before you begin
Ensure that the *dfpshell* privileges are available.

To configure Key Rotation:

1. Run the following command to add key rotation for the encrypted files or directories.
   dfp fe krotate add [-f] [-r] -p policy1 - /<source path or file protector mount point>
2. Edit the *key_rotation.conf* file to add the following configuration.
   This configuration file follows the following format:

```
00 09 * * *
key_rotate_retry_times=4
key_rotate_retry_interval=50
```

3. Run the following command to ensure that the key rotation service is running.
   dfpadmin service all status
4. Run the following command to verify the configuration status for the key rotation.
   dfp fe krotate status
5. Change the key of ***de1*** in ESA and again deploy the policy to the File Protector.
6. Run the following command at 9:00 am to view the changed key rotation status of the encrypted files or directories.
   dfp fe krotate status

10.4.4 Displaying the Key Rotation Status

This section describes the steps to display the key rotation status for the encrypted files and directories.

Before you begin
Ensure that the *dfpshell* privileges are available.

To display the Key rotation status for the specified encrypted files and directories:

1. Run the following command to display the key rotation status.
   dfp fe krotate status
2. Press Enter.
   This command displays the following key rotation status information for the encrypted files or directories:
   - Status
   - Index
   - Path

- DE (Policy)
- Encrypted KeyID
- Last Rotate Date

The following snippet provides a sample result for the ***dfp fe krotate status*** command.

```
PS C:\Users\Administrator> dfp fe krotate status
Enter dfpshell Pass Phrase: **************
File Protector File-Krotate Setting Status:

  Status     Index  Path                       DE(Policy)              KeyID
[LastRotateDate]
-------------------------------------------------------------------------------------
  [Unknown]  4.     FPDID00000g9rMZ001PbfV3PrN rcwd(policy_access)      0      [-]
                    pJN?H:\Users\shrima\fe_new

  [Valid]    5.     C:\Users\Administrator\fil aes1-rcwd(policy_fe)     671    [-]
                    e

PS C:\Users\Administrator>
```

10.4.5 Deleting the Key Rotation Configuration

This section describes the steps to delete the key rotation configuration for the specified encrypted files or directories.

Before you begin
Ensure that the *dfpshell* privileges are available.

➤ To delete the Key rotation configuration for the specified encrypted files or directories:

1. Run the following command to delete the key rotation configuration.
   dfp fe krotate del [-r] <path ...>
   Where:

   -r - recursively deletes key rotation configuration for the specified encrypted directories (optional).

2. Press Enter.
   This command deletes the key rotation configuration.

10.4.6 Removing an Invalid Entry of Key Rotation

This section describes the steps to remove the decrypted or deleted files or directories entries from the key rotation list.

Before you begin
Ensure that the *dfpshell* privileges are available.

➤ To remove an invalid entry for Key rotation:

1. Run the following command to remove an invalid key rotation.
   dfp fe krotate cleanup [-y] [path-wildcard]
   Where:

`-y` - Applies *yes* for all interactive questions during cleaning up the key rotations.

> **Note:** If any command option is not provided, then this command provides a prompt to remove all the invalid key rotation.

2. Press Enter.
   The invalid key rotation is removed.

10.5 Using Audit Logging

The File Protector monitors the security operations and audit logs.

An audit log is triggered when you perform the following tasks:

- Protect and unprotect files and directories with AC
- Encrypt and decrypt files and directories with FE
- Read or write AC protected files or directories
- Read or write FE encrypted files
- Delegate or undelegate a program
- Execute a delegated program
- Delete an AC protected or FE encrypted file or directory
- Delete a `.dfplock` file and an encrypted file or directory in an FE encrypted directory
- Load a policy
- Enter a privileged terminal
- Export or import DFP protected files into the File Protector
- Add or delete the key rotation configuration
- Create, Write, or Delete the File Protector installation directory with *dfpshell*

An audit logging operation includes success and failure operations.

The following events are generated for the audit configuration:

Events	Description
OPEN_W	Open an FE encrypted and AC protected file for writing
OPEN_R	Open an FE encrypted and AC protected file for reading
FE_PROTECT	FE encrypt file or directory
FE_UNPROTECT	FE decrypt file or directory
AC_PROTECT	AC protect file or directory
AC_UNPROTECT	AC unprotect file or directory
CDEL	Delegate a program
UDEL	Undelegate a program
RMVF	Remove FE encrypted file or directory
USDEL	Delegate the user
USUDEL	Undelegate the user
LOAD_POLICY	Load the policy for the user
DFPSHELL	Login the *dfpshell* and change the *dfpshell* password operation
KEY_ROTATION_ADD	Add the key rotation
KEY_ROTATION_DEL	Delete the key rotation

10.5.1 Configuring the Log Server Logging Modes

The *pepserver.cfg* file configures the Log server service settings.

To configure the Log server Logging modes:

1. Navigate to the *..Protegrity\defiance-core\data* directory.
2. Open the *pepserver.cfg* file.
3. Update the following fields with the respective values in the *pepserver.cfg* file.

```
# In case that connection to the fluentbit is lost, set how logs must be handled.
# This setting is only for the protector logs and not application logs, sent from
pepserver
# drop = (default) Protector throws logs away if connection to the fluentbit is lost
# error = Protector returns error without protecting/unprotecting data if connection to
the fluentbit is lost
mode = drop
```

> **Note:** Here, the *mode* denotes how the logs will be handled under error conditions.
>
> - **Drop Mode**:
> - In this mode, the File Protector will drop the logs in case of any errors.
> - **Error Mode**:
> - In this mode, the File Protector will save the logs in case of failure and try to send it later.
> - If the Logforwarder service is not running, or the Logforwarder is running under error conditions, then the logs are stored at the default location in the *..\Protegrity\dfperrorlogd\errorlog* directory.

For more information about each configuration parameter in the *pepserver.cfg* file, refer to the section *Configuring Log Server Settings File*.

4. Run the following command to update the new remote log setting of the File Protector.

```
dfpadmin update
```

After successful updation, the following message appears.

```
Updating File Protector settings is successful!
```

Chapter 11

Backup and Restore the Protected Data

The Backup and Restore for files and directories protected by the File Protector can be achieved by following the recommended steps in the *Appendix Scenarios for Backup and Restore*. For File Protector protected files, and directories, you should backup according to the type of protection configured using Access Control, or File Encryption.

For more information about backing up and restoring the files, or directories, refer to section *Scenarios of File Protector Backup and Restore*.

The File Protector can identify the File Protector protected files automatically. Even if you run the same or different File Protector versions, even if the OS is the same or different, the format of the files protected by the File Protector does not change. You can follow the required procedures to perform the restore.

Chapter 12

Metering

The Metering feature counts the number of successful protect, unprotect, and reprotect operations per file.

The ESA, which is connected to the File Protector in the production environments, collates the total count of successful protect, unprotect, and reprotect operations per file, as reported by the File Protector. As part of Protegrity Prime, these counts need to be shared with Protegrity by generating the Metering report, from the ESA Web UI.

The pricing model for Protegrity Prime customers is derived from these reported counts containing the number of successful protect, unprotect, and reprotect operations performed on each file.

The following table describes how the Metering count is calculated for various scenarios of the File Protector.

Table 12-1: Metering Count Estimation

Protect Count	Unprotect Count	Re-protect Count	Note
The metering count for protect operation is incremented based on the total number of files that are protected by the File Protector.	The metering count for unprotect operation is incremented based on the total number of files that are unprotected by the File Protector.	The metering count for re-protect operation is incremented based on the total number of files that are re-protected by the File Protector.	• The metering count for protect, unprotect, and re-protect operations are considered individually for FE and AC protection. • Extra Unprotect metering count is generated as a read operation while decrypting a file.

12.1 Generating the Metering Report

The Metering report, which is available on the ESA, can be generated using the ESA Web UI.

> **Note:** Ensure that you are assigned the *Custom Business Manager* role to generate a metering report.

To generate the Metering Report:

1. On the ESA Web UI, navigate to **Settings** > **Licenses**.
2. Click **Download Report**.
   The Metering report is created after collating all successful protect, unprotect, and reprotect operations on each file that are reported by the File Protector.

The Protegrity Metering report includes the information as per the following table.

Attribute	Description
Description	The description provided when generating the report
Hostid	The ESA host ID assigned as a part of the licensing requirement
Created	The timestamp for the report specifying the date and time of the report creation
Metadata	The metadata information includes the following attributes: **hostname**: ESA host name **ip**: ESA IP address **platform**: ESA platform **version**: ESA platform version
Integrity	A check to determine if any modifications are done to the ESA repository where the counts are stored and the report is generated
Date	The date and month for which the collective counts are recorded
Node information	The metadata information includes the following attributes: **uid**: Unique identifier for the node **hostname**: Node host name **ip**: Node IP **platform**: Node platform **version**: Node version (PEP version)
Protect	The details that are sent to the ESA for all successful protect operations per file **metering**: The total count of protected files till date **delta**: The delta indicating the count of protected files for the node in a month
Unprotect	The details that are sent to the ESA for all unprotect operations per file **metering**: The total count of unprotected files till date **delta**: The delta indicating the count of protected files for the node in a month
Reprotect	The details that are sent to the ESA for all reprotect operations across all protectors **metering**: The total count of operations till date **delta**: The delta indicating the count of operations for the node in a month
Signature	The signature for the Metering report that can help validating if the report has been tampered

A sample of the Metering report is provided in the following snippet:

```
{
"description" : "",
"hostid" : "",
"created" : "",
"metadata" : {
"hostname" : "",
```

```
"ip" : "",
"platform" : "",
"version" : ""
},
"integrity" : "ok",
"dates" : [ {
"date" : "2018-03"
}, {
"nodes" : [ {
"uid" : "",
"metadata" : {
"hostname" : "",
"ip" : "",
"platform" : "",
"version" : ""
},
"protect" : {
"metering" :,
"delta" :
},
"unprotect" : {
"metering" :,
"delta" :
},
"reprotect" : {
"metering" :,
"delta" :
}
} ]
} ],
"signature" : ""
}
```

Chapter 13

Troubleshooting

13.1 File Protector Common Errors

Table 13-1: File Protector Common Errors

Error /Problem	This may happen because…	Recovery
Failed to deploy a File Protector policy to the File Protector machine, which has the PEP server running	• The PEP server credentials are incorrect. • The Firewall blocks the port *15600*. • The difference between the system times of the File Protector machine and the ESA is large.	• Get the correct PEP server credentials. • Configure the Firewall to open the port *15600* and try to deploy again. • Change the system time of the File Protector or the ESA machine to make them the same or close to each other and try to deploy again.
The following error appears when deploying the policy for the File Protector. `Post deploy application failed: Executed process terminated with error`	• The File Protector services might be off. **Note:** You can check the services status by running the command `dfpadmin services all status` • The File Protector might be turned off. **Note:** You can check the eight FP modules by running the command `fltmc`	• If all the FP services are off, then run the command `dfpadmin services all on` to start them. • If all FP services are not off, then run the command `dfpadmin services all off` and execute the command `dfpadmin services all on`. • If the FP is turned off, then stop and start the FP by running the `dfpadmin FPNG stop` and `dfpadmin FPNG start` commands.
The following error appears when deploying the policy for the File Protector. `Message unpack error Attribute data store name not found for table DPT_ENFORCEMENT_POINT`	The version of ESA appliance is not compatible with the PEP server that is installed with File Protector.	Ensure that the version of the PEP server is compatible with the ESA appliance.
Failed to start the PEP server even if the credentials are correct. The following error appears. `Failed to create shared memory:Invalid`	The ESA host name or IP address were not specified when installing the PEP server.	Change the url under the *[policymanagement]* section in the *pepserver.cfg* file located in the `C:\Program Files\Protegrity\Defiance DPS\data\` directory to the active ESA IP. The same IP should be used in the *[audit]* section.

Error /Problem	This may happen because…	Recovery
`argument (Code=-1 : Invalid parameter!)`		
You cannot list the policy by using the **`dfp policy status`** command	• Not specify the ESA host name or IP address when installing the PEP server. • The PMS connection failed.	Restart the PEP server. If you still cannot get the policy list, then restart the PMS service too.
You cannot execute the File Protector **`dfp`**/ **`dfpadmin`** commands by prompting the `command not found` error	If the File Protector was not successfully installed, then it could happen that the **`dfp`** and related commands are not set in the environment variable.	Add the `C:\Program Files\Protegrity\Defiance File Protector\bin` directory to the environment variable.
You cannot get the File Protector log events on the ESA when configuring the *remote* parameter located in the *log.conf* file	• The configuration in the *log.conf* file located in the `C:\Program Files\Protegrity\Defiance File Protector\data\` directory is incorrect. • The ESA credentials are incorrect.	Update the configuration in the *log.conf* file.
The key of the configured file or directory is not rotated during the configured time	The configuration in the *key_rotation.conf* file located in the `C:\Program Files\Protegrity\Defiance File Protector\data` directory is incorrect. New key in the data element is not created and deployed in the Policy management in ESA.	Update the configuration in the *key_rotation.conf* file. Create the new key in the data element and deploy the policy again.
On a Windows system, the status of protection becomes invalid for a shared file or directory after restarting the system.	The shared mount point is unmounted after restarting the system.	Remount the shared mount point and execute the **`dfp ac update`** command to make the protections of the shared file or directory active.
After updating a policy in the Policy Management on the ESA and deploying it to the File Protector, you do not get this policy updated in the existing delegations	The policy with which the delegation was done was not refreshed.	1. Restart the PEP server. 2. Run the **`dfpadmin update`** command. 3. Run the **`dfpadmin database -o updatedb-policy-passwd -p <role>@<policy> [passwd]`** command.
The following error message appears when running the *dfp* commands. *ERROR: failed to connect the policy management server <127.0.0.1:15312>!*	The PMS could not service the command. It could be busy in serving another command.	Check if the PMS service is running or stopped. 1. Run the **`dfpadmin service pms off`** command 2. Run the **`dfpadmin service pms on`** command If the issue still persists, then check the Event Viewer logs from PMS.
Failure in performing key rotation on the FE encrypted network shared files and directories.	The logon account of the key rotate service has no permissions on the network shared files and directories.	Change the logon account or the user permissions of the network shared files to ensure that the key rotate service has the permissions on the FE encrypted network shared files and directories.
There are a lot of invalid AC protections in the AC status list, so it takes a long time to list all the AC statuses.	The files or directories are created temporarily and are then removed.	Run the **`dfp ac cleanup [-y] [path-wildcard]`** command to clean up the invalid AC protections.
After running the **`dfp`** command, the following error appears. `ERROR: file protector privilege timed out!`	This is caused by the dfpshell timeout feature. If the dfpshell session is idle for a certain period, then it automatically logs out.	The timeout setting can be changed by changing the *PRIVILEGE_TIMEOUT_INTERVAL* parameter in the *policy_management_server.conf* file located in the `C:\Program`

Error /Problem	This may happen because…	Recovery
		`Files\Protegrity\Defiance File Protector\data\` directory.
In File Protector, if the OS admin removes a delegated user from the system (for example: user1, UID=101), and later creates a new user with the same user ID (for example: user2, UID=101), then this new user (user2) gets delegated.	In File Protector, the OS admin reuses the available IDs for a new user and these IDs are used for identifying the delegated users. If a new user is assigned the same ID as a delegated user that was removed from the OS, then this new user gets delegated.	Ensure that you perform one of the following steps: • Undelegate the user before it is removed from the OS. • Modify the new user ID to another value.
Application delegation is not working although delegation status is *ACTIVE*.	This can occur if delegation application was running before FP services were started or policy was not available on PMS.	Perform the following steps to overcome this issue: 1. Run the command `dfp info` and check if the required policy is available. 2. Restart the delegated service.
The FP Services cannot start automatically if stopped from the command-prompt even after restart.		To start the FP services, run the command `dfpadmin service pms` manually from the command prompt. **Note:** If FP services are turned OFF from command prompt, then turn it ON from command prompt only. Turning it ON from the Task Manager will not work.
The Logforwarder service does not start automatically after installation.		To start the Logforwarder service, run the command `net start logforwarder`. Alternatively, navigate to **taskmanager** > **services**. Right-click on the **logforwarder service** and click **Start**.

Chapter 14

Use Cases for the File Protector

This section explains prominent Use Cases of the File Protector.

14.1 File Protector for the Local File System

The following section is useful for the users who want to install the File Protector in the Local File System for protecting files and directories.

14.1.1 Use Case: Encrypting Files and Directories on a Local File System

This Use Case describes the steps to protect files and directories containing sensitive information.

Before you begin
Ensure that the *dfpshell* privilege is available.

➤ To encrypt files and directories on a local file system:

1. Configure the Protector node with the data store in the ESA and deploy the data store keys.
   For more information about configuring data stores, refer the section *Working with Data Stores* in *Protegrity Policy Management Guide 9.1.0.0*.
2. Create the policies on the ESA.
   For more information about creating policies, refer the section *Creating and Deploying Policies* in the *Protegrity Policy Management Guide 9.1.0.0*.
3. Deploy the policies on the Protector node.
   For more information about deploying policies, refer the section *Creating and Deploying Policies* in the *Protegrity Policy Management Guide 9.1.0.0*.
4. Install the File Protector on a system.
   For more information about installing the File Protector, refer to the section *Installing the File Protector*.
5. Type the following command to encrypt the required file or directory and then press ENTER.
   ```
   dfp fe protect –d <data element> <file or directory>
   ```

If any user, program, or process are required to access the encrypted path then delegation is required.

For more information about delegating the user, program, or process, refer to the section *Using Delegation*.

14.1.2 Use Case: Protecting Files and Directories on a Local File System

This Use Case describes the steps to protect files and directories containing sensitive information.

> **Note:** The commands discussed in this Use Case can be used to protect encrypted files and directories.

Before you begin
Ensure that the *dfpshell* privilege is available.

➤ To protect files and directories on a local file system:

1. Configure the Protector node with the data store in the ESA and deploy the data store keys.
   For more information about configuring data stores, refer the section *Working with Data Stores* in the *Protegrity Policy Management Guide 9.1.0.0*.
2. Create the policies on the ESA.
   For more information about creating policies, refer the section *Creating and Deploying Policies* in the *Protegrity Policy Management Guide 9.1.0.0*.
3. Deploy the policies on the Protector node.
   For more information about deploying policies, refer the section *Creating and Deploying Policies* in the *Protegrity Policy Management Guide 9.1.0.0*.
4. Install the File Protector on a system.
   For more information about installing the File Protector, refer to the section *Installing the File Protector*.
5. Type the following command to protect the required file or directory and then press ENTER.
   ```
   dfp ac protect –d <data element> <file or directory>
   ```

If any user, program, or process are required to access the encrypted path then delegation is required.

For more information about delegating the user, program, or process, refer to the section *Using Delegation*.

14.1.3 Use Case: Encrypting and Protecting Files and Directories on a Local File System

This Use Case describes the steps to protect encrypted files and directories containing sensitive information using the *dfp file* command.

Before you begin
Ensure that the *dfpshell* privilege is available.

➤ To protect and encrypt files and directories on a local file system:

1. Configure the Protector node with the data store in the ESA and deploy the data store keys.
   For more information about configuring data stores, refer the section *Working with Data Stores* in the *Protegrity Policy Management Guide 9.1.0.0*.
2. Create the policies on the ESA.

For more information about creating policies, refer the section *Creating and Deploying Policies* in the *Protegrity Policy Management Guide 9.1.0.0*.

3. Deploy the policies on the Protector node.

   For more information about deploying policies, refer the section *Creating and Deploying Policies* in the *Protegrity Policy Management Guide 9.1.0.0*.

4. Install the File Protector on a system.

   For more information about installing the File Protector, refer to the section *Installing the File Protector*.

5. Type the following command to protect the required file or directory and then press ENTER.

   ```
   dfp file protect -d <data element> <file or directory>
   ```

If any user, program, or process are required to access the encrypted path then delegation is required.

For more information about delegating the user, program, or process, refer to the section *Using Delegation*.

14.2 File Protector for the Common Internet File System (CIFS) or Server Message Block (SMB)

The following section is useful for the users who want to protect files and directories that are shared using the Common Internet File System (CIFS) or Server Message Block (SMB).

This section describes the following use cases for CIFS/SMB:

- Protecting shared files and directories on a CIFS/SMB Client
- Protecting shared files and directories on a CIFS/SMB Server
- Protecting shared files and directories on a CIFS/SMB Client and Server

14.2.1 Use Case: Protecting Files and Directories on a CIFS or SMB Client

In this scenario, the File Protector is installed on the CIFS Client to protect files located in the CIFS shared path on the server. In this case, multiple clients connect to the CIFS server through the CIFS. Each client that needs to access the protected data should have the File Protector installed.

The following steps provides an overview of protecting files and directories when the File Protector is installed on the CIFS client.

1. Configure the CIFS/SMB client node with the data store added to ESA.
2. Deploy the data store keys.
3. Create and deploy the policies on the CIFS client.
4. Mount the CIFS shared path on the CIFS client.
5. Protect the required files and directories using the File Protector.

The following figure displays the encrypted data flow between the CIFS server and clients.

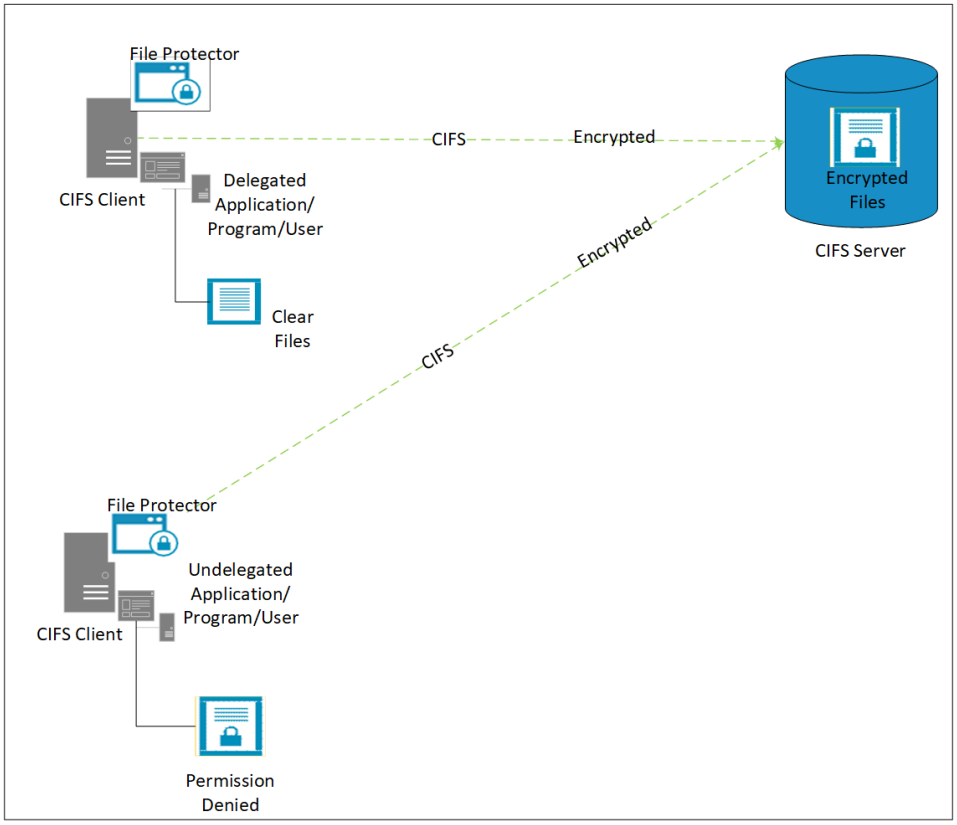


Figure 14-1: Use Case 1: File Protector Installed on the CIFS Client

Before you begin
Ensure that the *dfpshell* privileges are available.

➤ To encrypt files and directories on the CIFS Client:

1. Configure the CIFS/SMB client node with the data store in the ESA and deploy the data store keys.
   For more information about configuring data stores, refer the section *Working with Data Stores* in *Protegrity Policy Management Guide 9.1.0.0*.
2. Create the policies on the ESA.
   For more information about creating policies, refer the section *Creating and Deploying Policies* in the *Protegrity Policy Management Guide 9.1.0.0*.
3. Deploy the policies on the CIFS client.
   For more information about deploying policies, refer the section *Creating and Deploying Policies* in the *Protegrity Policy Management Guide 9.1.0.0*.
4. Install the File Protector on the CIFS Client.
   For more information about installing the File Protector, refer to the section *Installing the File Protector*.
5. Type the following command to mount the CIFS shared path on the CIFS client and then press ENTER.
   ```
   net use devicename: \\<computername or IP-Address>\<sharename>[\volume]
   ```
   For example, `net use T: \\10.10.135.87\test_cifs`

   Alternatively, perform the following steps to mount the CIFS shared path on the CIFS client using the UI.

a. Open **File Explorer**.
b. Select **This PC**.
c. Click **Computer** in the ribbon menu at the top.
d. Navigate to **Map Network drive**.
e. Select **Map network drive**.
The **Map Network Drive** window appears.
f. Select the drive letter you want to use for the network directory, from the **Drive** list.
g. In the **Folder** box, type the path to the shared network directory.
The format is `\\hostname\sharename`, where *hostname* is the name of the machine or the IP address and *sharename* is the name of the shared directory that is to be mounted on the client.
Alternatively, Click **Browse** to select the directory you want to map.
h. Click **Finish**.

6. Type the following command to protect files or directories in the CIFS shared path and then press ENTER.
```
dfp file protect [-r] -d <data element> <file or directory>
```

> **Note:** To access a file operation, the program, process, or user must be delegated.
>
> On the File Protector, only a delegated user, process, or program can run file operations on the encrypted files. The files are stored in an encrypted format on the server and always remain encrypted on the network. If any undelegated program, process, or user tries to access the protected file or directory path, then the following error message appears.
>
> ```
> Permission denied
> ```

7. Type the following command to delegate programs and then press ENTER.
```
dfp delegate -e <program> <policyName>
```
A prompt for the policy password appears.

> **Note:** For more information about delegating the user, program, or process, refer to the section *Using Delegation*.

8. Enter the policy password.

14.2.2 Use Case: Protecting Files and Directories on a CIFS or SMB Server

In this scenario, the File Protector is installed on the CIFS server.

The following steps provides an overview of protecting files and directories when the File Protector is installed on the CIFS server.

1. Create a CIFS shared directory on the CIFS server.
2. Create the CIFS configuration setup on the server side for sharing data with the CIFS clients.
3. Protect the files and directories using the File Protector and delegate the CIFS services with the required policy.

The following figure displays the encrypted data flow within the CIFS server and clients.

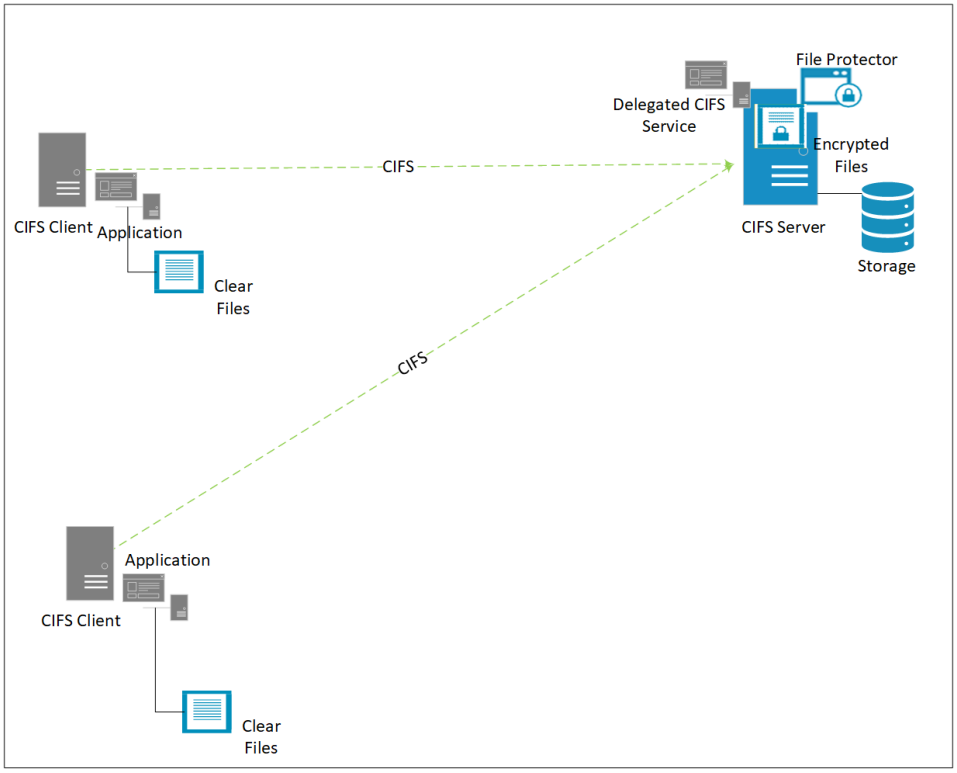


Figure 14-2: FP installed on the CIFS Server

Before you begin

Ensure that the following prerequisites are met:

- The File Protector is installed on the CIFS server.
- The File Protector is not installed on the CIFS client.
- The *dfpshell* privileges are available.

To encrypt files and directories on the CIFS server:

1. Configure the CIFS server with the data store in the ESA and deploy the data store keys.

   For more information about configuring data stores, refer the section *Working with Data Stores* in *Protegrity Policy Management Guide 9.1.0.0*.
2. Create the policies on the ESA.

   For more information about creating policies, refer the section *Creating and Deploying Policies* in the *Protegrity Policy Management Guide 9.1.0.0*.
3. Deploy the policies on the CIFS server.

   For more information about deploying policies, refer the section *Creating and Deploying Policies* in the *Protegrity Policy Management Guide 9.1.0.0*.
4. Install the File Protector on the CIFS server.

   For more information about installing the File Protector, refer to the section *Installing the File Protector*.
5. Type the following command to mount the CIFS shared path on the CIFS server and then press ENTER.

   ```
   net use devicename: \\<computername or IP-Address>\<sharename>[\volume]
   ```

   For example, `net use T: \\10.10.135.87\test_cifs`

Alternatively, perform the following steps to mount the CIFS shared path on the CIFS server using the UI.

a. Open **File Explorer**.

b. Select **This PC**.

c. Click **Computer** in the ribbon menu.

d. Navigate to **Map Network drive**.

e. Select **Map network drive**.
The **Map Network Drive** window appears.

f. Select the drive letter you want to use for the network directory, from the **Drive** list.

g. In the **Folder** box, type the path to the shared network directory.
The format is `\\hostname\sharename`, where *hostname* is the name of the machine or the IP address and *sharename* is the name of the shared directory that is to be mounted on the client.
Alternatively, Click **Browse** to select the directory you want to map.

h. Click **Finish**.

6. Run the following command to protect files or directories using the CIFS server on the File Protector.
```
dfp file protect -d <data element> <file or directory>
```
7. Run the following command to delegate the driver file `srv.sys` in the system directory.
```
dfp delegate -e C:\Windows\System32\Drivers/srv.sys <role>@<policy>
```
A prompt for the policy password appears.
8. Enter the policy password.
9. Run the following command to delegate another driver file `srv2.sys` in the system directory.
```
dfp delegate -e C:\Windows\System32\Drivers/srv2.sys <role>@<policy>
```
A prompt for the policy password appears.
10. Enter the policy password.

> **Note:** Restart of the server is required after undelegating the above services.

The files are stored in encrypted format on the server and in clear format on the network.

14.2.3 Use Case: Protecting Files and Directories on a CIFS or SMB Client and CIFS or SMB Server

In this scenario, the File Protector is installed on the CIFS server and CIFS clients.

The following steps provide an overview of protecting files and directories when the File Protector is installed on the CIFS server and CIFS clients.

1. Create a CIFS shared directory on the CIFS server.
2. Protect the files and directories in the shared path using the File Protector.
3. Mount the shared path on the CIFS client machines.

The following figure displays the encrypted data flow within the CIFS server and clients.

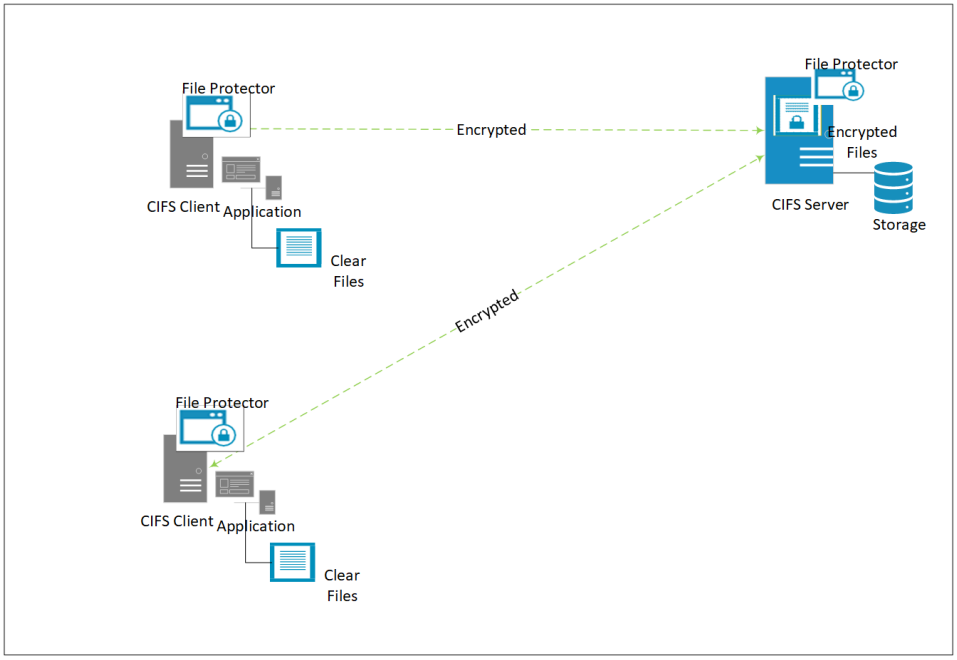


Figure 14-3: File Protector Installed on the CIFS client and the CIFS server

Before you begin

Ensure that the following prerequisites are met:

- The File Protector is installed on the CIFS server and clients.
- The server and the clients should have the same policy deployed.
- The *dfpshell* privileges are available.

To encrypt files and directories on the CIFS server and clients:

1. Configure the CIFS server with the data store in the ESA and deploy the data store keys.

   For more information about configuring data stores, refer the section *Working with Data Stores* in *Protegrity Policy Management Guide 9.1.0.0*.
2. Create the policies on the ESA.

   For more information about creating policies, refer the section *Creating and Deploying Policies* in the *Protegrity Policy Management Guide 9.1.0.0*.
3. Deploy the policies on the CIFS server.

   For more information about deploying policies, refer the section *Creating and Deploying Policies* in the *Protegrity Policy Management Guide 9.1.0.0*.
4. Install the File Protector on the CIFS server.

   For more information about installing the File Protector, refer to the section *Installing the File Protector*.
5. Create a CIFS shared directory on the CIFS server.
6. Type the following command to mount the CIFS shared path on the CIFS clients and then press ENTER.

   ```
   net use devicename: \\<computername or IP-Address>\<sharename>[\volume]
   ```

   For example, `net use T: \\10.10.135.87\test_cifs`

   Alternatively, perform the following steps to mount the CIFS shared path on the CIFS clients using the UI.

   a. Open **File Explorer**.

b. Select **This PC**.

c. Click **Computer** in the ribbon bar.

d. Navigate to **Map Network drive**.

e. Select **Map network drive**.
The **Map Network Drive** window appears.

f. Select the drive letter that you want to use for the network directory, from the **Drive** list.

g. In the **Folder** box, type the path to the shared network directory.
The format is `\\hostname\sharename`, where *hostname* is the name of the machine or the IP address and *sharename* is the name of the shared directory that is to be mounted on the client.
Alternatively, click **Browse** to select the directory you want to map.

h. Click **Finish**.

7. Type the following command to protect files or directories in the CIFS shared path and then press ENTER.
```
dfp file protect [-r] -d <data element> <file or directory>
```

> **Note:** To access a file operation, the program, process, or user must be delegated.
>
> On the File Protector, only a delegated user, process, or program can run file operations on the encrypted files. The files are stored in an encrypted format on the server and always remain encrypted on the network. If any undelegated program, process, or user tries to access the protected file or directory path, then the following error message appears.
>
> ```
> Permission denied
> ```

8. Type the following command to delegate programs and then press ENTER.
```
dfp delegate -e <program> <policyName>
```
A prompt for the policy password appears.

> **Note:** For more information about delegating the user, program, or process, refer to the section *Using Delegation*.

9. Enter the policy password.

> **Note:** The Access Protection is local to node on which the file is created and protected.
>
> When a file is encrypted using FE, then the contents of the file appear as ciphertext. In case of a directory that is encrypted using FE, a `.dfplock` file is created in the directory.
>
> When a file or directory is encrypted and protected, then the following error message appears.
>
> ```
> Access is Denied
> ```

14.3 Protecting Files and Directories on the SFTP Server

In this scenario, the File Protector is installed on the SFTP server.

The following figure displays the encrypted data flow within the SFTP service and clients.

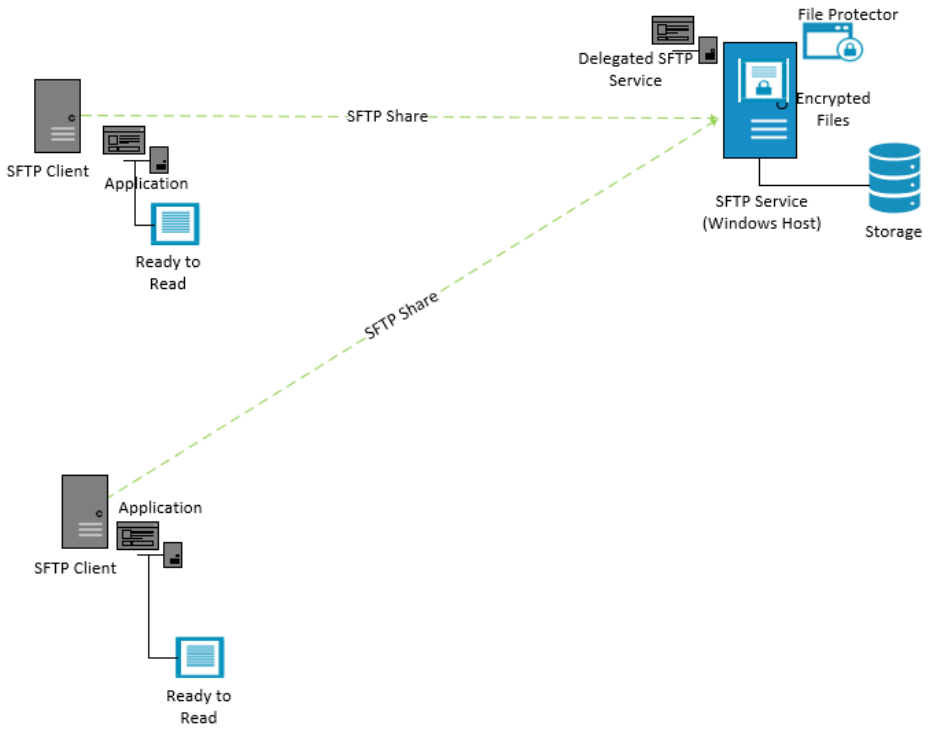


Figure 14-4: FP installed on the SFTP Server

Before you begin

Ensure that the following prerequisites are met:

- The File Protector is installed on the SFTP server.
- The *dfpshell* privileges are available.

To encrypt files and directories on the SFTP server:

1. Configure the SFTP service and client node with the data store added to the ESA.

   For more information about configuring data stores, refer the section *Working with Data Stores* in *Protegrity Policy Management Guide 9.1.0.0*.
2. Deploy the data store keys.

   For more information about creating policies, refer the section *Creating and Deploying Policies* in the *Protegrity Policy Management Guide 9.1.0.0*.
3. Create and deploy the policies on the SFTP service.

   For more information about deploying policies, refer the section *Creating and Deploying Policies* in the *Protegrity Policy Management Guide 9.1.0.0*.
4. Protect the required files and directories using the File Protector with the SFTP service.
5. Delegate the SFTP service by Program delegation.

   ```
   dfp delegate -e <sftp service> <policyName>
   ```
6. Restart the SFTP service.
7. Download the FE encrypted file on the client node.

   This file can be read in the clear with a delegated user or application.

The files are stored in encrypted format on the server and in clear format on the network.

Appendix A

Scenarios for Backup and Restore

This section contains scenarios and procedures to backup and restore the File Protector 9.1.0.0 files and directories on Windows, including File Encryption, and Access Control configurations.

The following are the prerequisites for File Protector backup and restore:

- The backup application should be running on local machine, and backup files and directories should be encrypted or protected.
- The **Machine1**, where the latest File Protector v9.1.0.0 is installed, contains the following:
  - a recursive FE encrypted directory
  - a recursive or inheritable AC protected directory
  - a recursive or inheritable FE and AC protected directory (directories can be FE and AC protected, either with a single File Protect command or by separate FE and AC commands)
- The **Machine1** has a mounted CIFS directory, which is provided by **Machine2** without the File Protector installed or **Machine3** with the same File Protector v9.1.0.0 installed.

  Each of the directory includes the following:

  - a recursive FE encrypted directory
  - a recursive or inheritable AC protected directory
  - a recursive or inheritable FE and AC protected directory

15.1 FE Encrypted Directory

This section describes the backup and restore for the scenario of recursive encrypted directory on a local and remote CIFS mount.

> **Note:** The IT Admin or a user is assigned the task to back up the encrypted files and directories regularly. Deploy a policy for the user in the ESA, where *Read*, *Write*, or *Protect* protection boxes are deselected for the data element and Protected Value in Output setting is set.

15.1.1 Backup on a Local Disk

This section describes the steps to backup an encrypted directory on a local disk.

You can backup the directory with **ciphertext**, without any data element loaded or backup agent delegated. For example, there is an encrypted directory, `fe` which has to be backed up.

➤ To backup an encrypted directory:

1. On **Machine1**, ensure there is no data element loaded and the local backup application is not delegated.
2. Use the local backup application to backup the directory `fe` on the local disk **Machine1**, naming the backup directory as `fe_bak`. This requires to backup the **ciphertext** of the encrypted files in the backed up directory `fe_bak`.

15.1.2 Restoring

This section describes the steps for restoring the backed up directory on a local disk.

As the backed up directory `fe_bak` is still on **Machine1**, you can just restore the directory in local.

The backed up directory `fe.bak` can be restored on either the source location or a target location.

There are two ways to restore the backup based on the changes made to the source directory `fe` after backup.

15.1.2.1 Restoring to the source location

This section describes the steps for restoring the backed up directory to the source location.

➤ To restore to source location:

1. The source directory `fe` is not changed after performing a backup.
   a. On **Machine1**, ensure that there are no data elements loaded, and the local restore application is not delegated. The source directory `fe` is renamed to `fe_ren`.
   b. Restore the backed up directory `fe_bak` to the source location and rename the directory `fe_bak` to `fe` on **Machine1**.
   c. Delete the renamed directory `fe_ren`.
2. The source directory `fe` is changed after performing a backup.
   a. In **Machine1**, ensure that there are no data elements loaded, and the local restore application is not delegated. **Machine1** renames the source directory `fe` to `fe_ren`.
   b. Restore the backed up directory `fe_bak` to the source location on **Machine1**.
   c. Load the data element used to FE protect the directory with all access masks.
   d. Copy all the files or directories under the restored directory `fe` to the renamed directory `fe_ren`.
   e. Delete the restored directory `fe_ren` and rename it back to `fe`.

15.1.2.2 Restoring to a target location

This section describes the steps for restoring the backed up directory to a target location.

To restore on a different target location:

1. On **Machine1**, ensure there are no data elements loaded, and the local restore application is not delegated.
2. Restore the backed up directory `fe_bak` to a different target location on **Machine1**.

15.1.3 Backup on a CIFS or SMB Mount

This section describes the steps to backup an encrypted directory on a CIFS or SMB mount.

Even if the CIFS mount may be shared by a machine that may or may not have FP installed, may have a different FP version, or different kernel of OS, you can still back it up with **ciphertext** without any data element loaded or backup agent delegated on local.

To backup an encrypted directory on CIFS or SMB mount:

1. On **Machine1**, ensure that there are no data elements loaded, and the local backup application is not delegated.
2. Use the local backup application to backup the directory `fe` to a target location which could be local, CIFS mounted as required.

The backed up directory is named `fe_bak` and the **ciphertext** of encrypted files are backed up in the `fe_bak` directory.

15.1.4 Restoring

This section describes the steps for restoring the backed up directory on CIFS or SMB mount.

There are two options for restoring the backups. Follow either of the two steps mentioned below:

To restore an encrypted directory:

1. If the backed up directory `fe_bak` is located in the **Machine1**, then you can just restore the directory in local.

   Follow the steps explained in the section *Restore* for *Local Disk*.
2. If the backed up directory is located in CIFS mount, then you have to mount the CIFS mount to **Machine1** and then restore it.
   a. In **Machine1**, ensure you get the backed up directory.
   b. Ensure that there are no data elements loaded, and the local restore application is not delegated.
   c. Restore the backed up directory in **Machine1** to a target location.

The target location can be a local or CIFS mount. The location may or may not be the same as the source directory `fe`, depending on your requirement.

15.2 AC Protected Directory

This section describes backup and restore for recursive and inheritable AC protected directories on a local and remote CIFS mount.

15.2.1 Backup on a Local Disk

This section describes the steps to backup directories protected using Access Control on a local disk.

The AC protected directories can be backed up only in **cleartext**. The File Protector does not support back up of AC protected files with **ciphertext**. For example, there is an AC protected directory named `ac` that should be backed up.

➤ To backup an AC protected directory :

1. On **Machine1**, perform all the AC protection procedures.
2. Run the following command to back up the `ac.db` file in the installed directory `/protegrity/fileprotector/data/`.
   ```
   dfp ac export <exported ac.db file>
   ```
3. Delegate the local backup application with the policy that AC protects the directory, or load the required policy.

   > **Note:**
   > Ensure that you include the data element having *Read* access control mask in the required policy.

4. Backup the directory in **Machine1** and name the directory as `ac_bak` using the local backup application. The content of the backup directory is in **cleartext**.
5. Run the command to back up the `delegate.db` file in the installed directory `/protegrity/fileprotector/data`.
   ```
   dfp delegate export <exported delegate.db file>
   ```

15.2.2 Restoring

This section describes the steps for restoring the backed up directory on a local disk.

The AC protections on the AC protected files and directories only take effect on the local machine. This means when the absolute path of the AC protected files and directories are changed, they will become clear files and directories. So you have to restore the backed up directory to the source location to keep the AC protection as is. Another option is to restore the directory to a new AC protected directory to ensure that the restored directory is AC protected.

There are two ways to restore the backed up AC protected directory according to the restored location. You can follow either of the two steps mentioned below.

➤ To restore the backed up AC protected directory:

1. Restore to the source location.

a. In **Machine1**, with data element to restore the backed up directory ***ac_bak*** to the source location in the local disk.

> **Note:** You can load the required policy to have the data element or delegate the data element to your restore application. Ensure that the data element has the *Read*, *Write*, or *Create* access control mask.

b. Run the following command to restore the backed up file `ac.db` to the installed directory `Protegrity/Defiance File Protector/data/`.

```
dfp ac import [-f] <imported ac.db file>
```

c. Run the following command to restore the backed up file `delegate.db` to the installed directory `Protegrity/Defiance File Protector/data/`.

```
dfp delegate import [-f] <imported delegate.db file>
```

> **Note:** When using the `-f` option to restore the backed up file `ac.db`, it recovers the current `ac.db` file in the installed directory `/Protegrity/Defiance File Protector/data/`. This means the new created AC protections in the current `ac.db` file are removed. The restored `ac.db` file only includes the previous backed up AC protections when you backed up the `ac.db` file.
>
> If you do not use the `-f` option, then the backed up AC protections will be restored and the current AC protections will be as is.

2. Restore to another new AC protected directory.
   a. On **Machine1**, AC protect another directory `newac`, which is recursive and inheritable.
   b. Using the required data element to restore the backed up directory `ac_bak` to the new AC protected directory `newac`.

15.2.3 Backup on a CIFS or SMB Mount

This section describes the steps to backup a directory protected with Access Control on CIFS or SMB mount.

This can only back up the **cleartext** for AC protected directory.

To backup an AC protected directory on CIFS or SMB mount:

1. Run the following command to back up the `ac.db` file in the installed directory `/Protegrity/Defiance File Protector/data/` on **Machine1**, after completing all the AC protections on the network.

```
dfp ac export <exported ac.db file>
```

2. Delegate the local backup application with the required policy that AC protects the directory, or load the proper policy.

> **Note:** Ensure that the required policy includes the data element which has the *Read* access control mask.

3. Use the local backup application to backup the directory named `ac_bak` on **Machine1**. It will require backing up in **cleartext**.
4. Run the following command to backup the `delegate.db` file in the installed directory `/Protegrity/Defiance File Protector/data/`.

```
dfp delegate export <exported delegate.db file>
```

15.2.4 Restoring

This section describes the steps for restoring the backed up directory on a CIFS or SMB mount.

Since the AC protection of the directory which is stored in the CIFS mount only takes effect on the local machine, which means the directories and files are clear in the CIFS server that shares the CIFS mount no matter the CIFS server has FP installed or not.

And when the absolute path of the AC protected directories and files are changed, they will become clear directories and files. So you have to restore the backed up directory to the source location to keep the AC protection intact. Or else restore the directory to another new AC protected directory to ensure the restored directory is AC protected.

Depending on the restore location, there are two ways to restore the backed up AC protected directory.

15.2.4.1 Restoring to the source location

This section describes the steps for restoring the backed up directory on CIFS or SMB mount.

➤ To restore to the source location:

1. On **Machine1**, with data element to restore the backed up directory *ac_bak* to the source location in the CIFS mount.

   > **Note:** You can load the required policy to have the data element or delegate the data element to your restore application. Ensure that the data element has the *Read*, *Write*, or *Create* access control mask.

2. Run the following command to restore the backed up file *ac.db* to the installed directory */Protegrity/Defiance File Protector/data/*.

   ```
   dfp ac import [-f] <imported ac.db file>
   ```

3. Run the following command to restore the backed up file *delegate.db* to the installed directory */Protegrity/Defiance File Protector/data/*.

   ```
   dfp delegate import [-f] <imported delegate.db file>
   ```

15.2.4.2 Restoring on a target location

This section describes the steps for restoring the backed up directory on CIFS or SMB mount.

➤ To restore on a target location:

1. Protect another directory *newac*, recursively and with inheritable AC protect, in **Machine1** in the CIFS mount.
2. Using data element restore the backed up directory *ac.bak* to this new AC protected directory, *newac*.

15.3 Encrypted and Protected Directory

This section describes the backup and restore of recursive and inheritable encrypted directories, that are protected using Access Control, on a local and remote CIFS mount.

15.3.1 Backup on a Local Disk

This section describes the steps to backup encrypted directories protected using Access Control on a local disk.

You can load the required policy or delegate the local backup application with a specified data element to back up the **ciphertext** for FE or AC protected files in the FE or AC protected directory. For example, there is an FE or AC protected directory named *feac* which has to be backed up.

➤ To backup an encrypted directory protected using Access Control:

1. Create a role for the data element that has to encrypt and protect the directory using Access Control.
2. In the role, configure the data element with no *Unprotect*, *Protect*, and *Manage Protection* permissions.
3. Configure *Protected Value* as the Output setting.
4. On **Machine1**, complete all the FE and AC protection (or File Protection), then run the following command to back up the `ac.db` file in the installed directory `Protegrity/Defiance File Protector/data/` directory.
   ```
   dfp ac export <exported ac.db file>
   ```
5. Delegate the role from the local backup application or load the required policy through the role.
6. In local **Machine1**, use the local backup application to backup the `feac` directory, naming the backed up directory as `feac_bak`.
   The **ciphertext** of encrypted files is backed up in `feac_bak` directory.
7. Run the following command to backup the `delegate.db` file in the installed `/Protegrity/Defiance File Protector/data/` directory.
   ```
   dfp delegate export <exported delegate.db file>
   ```

15.3.2 Restoring

This section describes the steps for restoring the backed up directory on a local disk.

There are three ways to restore the backed up directory `feac_bak` depending on the restored location.

15.3.2.1 Restoring to a different location

This section describes the steps for restoring the backed up directory on a local disk.

There are two ways to restore to the source directory `feac` (with or without changes after backup). You can follow either of the two steps mentioned below.

➤ To restore the backed up directory:

1. The source directory `feac` is not changed after backup.
   a. Create a new directory `temp` in the source directory `feac` location, in **Machine1**.
   b. Via role to load the data element or delegate the role to the restore application.
   c. Use the restore application to restore the backed up directory `feac_bak` to the new directory `temp`.
   d. Load the data element which FE and AC protected the directory with all access masks.
   e. Rename the source directory `feac` to `feac_ren`.
   f. Move the restored directory `feac` located in the target directory `temp` to the source directory `feac` location.
   g. Delete the renamed directory `feac_ren`.
   h. Run the following command to restore the backed up `ac.db` file to the installed directory `/Protegrity/Defiance File Protector/data/`.
      ```
      dfp ac import [-f] <imported ac.db file>
      ```
   i. Run the following command to restore the backed up `delegate.db` file to the installed directory `/Protegrity/Defiance File Protector/data/`.
      ```
      dfp delegate import [-f] <imported delegate.db file>
      ```
2. The source directory `feac` is changed after backup.
   a. In **Machine1**, in the source directory `feac` location, create a new directory `temp`.
   b. Ensure that there is no data element loaded and the local restore application is not delegated.

c. Restore the backed up directory *feac_bak* to the new directory *temp*.

d. Load the data element that AC protected the encrypted directory with all access masks.

e. Copy the restored directory *feac* located in the new directory *temp* to the source directory *feac* location.

f. Run the following command to restore the backed up *ac.db* file to the installed directory */Protegrity/Defiance File Protector/data/*.

```
dfp ac import [-f] <imported ac.db file
```

g. Run the following command to restore the backed up *delegate.db* file to the installed directory */Protegrity/Defiance File Protector/data/*.

```
dfp delegate import [-f] <imported delegate.db file>
```

15.3.2.2 Restoring with system single mode

This section describes the steps for restoring the backed up directory on a local disk.

You need to have the process to enter the system's single mode. Even if the source directory *feac* has changed or not after the backup, follow the steps to restore.

To restore the backed up directory:

1. Enter the system's single mode.
2. Restore the backed up directory *feac_bak* to the source location.
3. Boot back to multi users' mode.
4. Restore the backed up files *ac.db* and *delegate.db* to the installed directory */Protegrity/Defiance File Protector/data/* separately by command ***dfp ac import [-f] <imported ac.db file>*** and ***dfp delegate import [-f] <imported delegate.db file>***.

15.3.3 Backup on a CIFS or SMB Mount

This section describes the steps to backup an encrypted directory protected with Access Control on CIFS or SMB mount.

Even if the CIFS mount is shared with a machine that may or may not have the FP installed, different version of FP, or different kernel of OS, you can back up it with ciphertext with a specified data element loaded or with a backup agent delegated with the data element in local.

To backup an encrypted directory protected with Access Control on CIFS or SMB Node:

1. Create a role for the data element which is going to FE and AC protect the directory. In the role, configure the data element with no Unprotect, Protect and Manage Protection permissions and configure Protected Value as the Output setting.
2. In **Machine1**, delegate the role to the local backup application or load the proper policy via role.

Use the local backup application to backup the directory *feac* to a target location, which could be local or the CIFS mount as required by you, and name the backed up directory *feac_bak*. This means backing up **ciphertext** of encrypted files in the backed up directory *feac_bak*.

15.3.4 Restoring

This section describes the steps for restoring the backed up directory on CIFS or SMB node.

If the backed up directory *feac_bak* is located in the **Machine1**, then you can just restore the directory in local. If it is located in the CIFS mount, then you have to mount the CIFS mount to Machine1 and restore it.

➤ To restore an encrypted directory protected with Access Control:

1. In **Machine1**, ensure that you get the backed up directory.
2. Via role to load the data element or delegate the role to the restore application.
3. The **Machine1** uses the restore application to restore the backed up directory to a target location.

> **Note:** The target location can be local or CIFS mount, can be the same as source directory *fe* location or not, which depends on you. According to the restore location, refer to the steps in this table for the local restore since they are mostly the same.
>
> However, note that the method using system single mode is not available for the CIFS mount as there is no network in system single mode.

Appendix B

Glossary

AC

Access Control

AES

Advanced Encryption Standard

CIFS

Common Internet File System

DFPSHELL

Defiance File Protector Shell

ESA

Enterprise Security Administrator

FAT

File Allocation Table

FE

File Encryption

FP

File Protector

FPDID

File Protector Device ID

GUID

Globally Unique Identifier

KEK

Key Encryption Key

NTFS

New Technology File System

PEP Server

Protection Enforcement Point Server

PMS

Process Management Service

RCFS

Redirect Cache File System

RMS

Remote Management Service

SMB

Server Message Block

UID

Unique Identifier