MICRO FOCUS

ZENworks® 2017
Full Disk Encryption Policy Reference

December 2016

Legal Notice

For information about legal notices, trademarks, disclaimers, warranties, export and other use restrictions, U.S. Government rights, patent policy, and FIPS compliance, see https://www.novell.com/company/legal/.

Third-Party Materials

All third-party trademarks are the property of their respective owners.

Contents

About This Guide

This *ZENworks Full Disk Encryption Policy Reference* provides information about deploying and managing Disk Encryption policies.

Audience

This guide is written for the ZENworks Full Disk Encryption administrators.

Feedback

We want to hear your comments and suggestions about this manual and the other documentation included with this product. Please use the User Comments feature at the bottom of each page of the online documentation.

Additional Documentation

ZENworks Full Disk Encryption is supported by other documentation (in both PDF and HTML formats) that you can use to learn about and implement the product. For additional documentation, see the ZENworks documentation website.

1 Concepts

ZENworks Full Disk Encryption uses disk encryption and pre-boot authentication to protect the data on a device's local fixed volumes when the device is powered off or in hibernation mode. The disk encryption and pre-boot authentication settings are applied to the device through a Disk Encryption policy.

The following sections introduce the Disk Encryption policy concepts you need to understand to successfully implement disk encryption on the managed devices in your ZENworks Management Zone.

Disk Encryption

ZENworks Full Disk Encryption provides software-based encryption for standard hard disks and supports hardware-based encryption used with self-encrypting hard disks.

Standard Hard Disks

Standard hard disks are any 3.5 or 2.5 inch IDE, SATA, or PATA disks that do not include a hardware encryption chip.

With standard hard disks, ZENworks Full Disk Encryption provides sector-based encryption of the entire disk or selected volumes (partitions). All files on a volume are encrypted, including any temporary files, swap files, or operating system files. Because all files are encrypted, the data cannot be accessed when booting the computer from external media such as a CD-ROM, floppy disk, or USB drive.

You can choose the industry-standard encryption algorithm (AES, Blowfish, DES, or DESX) and key length that best meets your organizations requirements.

NOTE: The cryptographic module used in ZENworks Full Disk Encryption to encrypt standard hard drives is *not* Federal Information Processing Standard (FIPS) 140-2 certified. However, the cryptographic module implements standards consistent with FIPS 140-2 Level 1 certification.

Self-Encrypting Hard Disks

Self-encrypting hard disks are disks that perform their own encryption via a hardware encryption chip.

ZENworks Full Disk Encryption supports self-encrypting hard disks that are compliant with the *Trusted Computing Group OPAL 2.0* specification. The two modes of support are:

- **Pre-boot authentication with software-based encryption:** This is supported on *ALL* OPAL 2.0 compliant drives.

  Pre-boot authentication is the process of authenticating a user to a device before the device boots to the primary operating system. Using ZENworks pre-boot authentication (ZENworks PBA) in conjunction with Windows login greatly enhances drive security. Software-based encryption adds a second layer of encryption to the drive's native hardware encryption.

For more information about ZENworks pre-boot authentication, see “Pre-Boot Authentication” on page 8.

- **Pre-boot authentication with drive locking:** ZENworks supports drive locking on *SOME* OPAL 2.0 compliant drives. The support is limited because of variations in the way drive manufacturers implement the OPAL 2.0 specification related to drive locking.

  When using this mode, drive locking is initiated during ZENworks PBA initialization. After user authentication occurs through the ZENworks PBA, the drive is unlocked until it is powered off. Only the native hardware encryption is used; ZENworks does not apply software-based encryption in this mode.

  For a list of known drive-locking compatible and incompatible drives, see *ZENworks Full Disk Encryption Self-Encrypting Drive Support*. For information about how to test a drive for drive-locking compatibility, see *ZENworks Full Disk Encryption Self-Encrypting Drive Compatibility Testing*.

NOTE: When upgrading a device with an OPAL drive from a ZENworks 11.3.x to an 11.4.x or later version, any existing Full Disk Encryption policies on the device and the Full Disk Encryption Agent must be temporarily removed prior to the upgrade. See “Full Disk Encryption policy fails on Opal devices during version upgrade” in *ZENworks 2017 Troubleshooting Full Disk Encryption* for more information.

Pre-Boot Authentication

ZENworks Full Disk Encryption protects a device’s data when the device is powered off or in hibernation mode. As soon as someone successfully logs in to the Windows operating system, the encrypted volumes are no longer protected and the data can be freely accessed. To provide increased login security, you can use ZENworks Pre-Boot Authentication (PBA).

The ZENworks PBA is a Linux-based component. When the Disk Encryption policy is applied to a device with a standard hard disk, a 100 MB partition containing a Linux kernel and the ZENworks PBA is created on the hard disk. When the policy is applied to a device with a self-encrypting hard disk, the Linux kernal and ZENworks PBA are installed to the disk’s datastore memory.

During normal operation, the device boots to the Linux partition and loads the ZENworks PBA. As soon as the user provides the appropriate credentials (user ID/password or smart card), the PBA terminates and the Windows operating system boots, providing access to the encrypted data on the previously hidden and inaccessible Windows drives.

The Linux partition is hardened to increase security, and the ZENworks PBA is protected from alteration through the use of MD5 checksums and uses strong encryption for authentication keys.

ZENworks Pre-Boot Authentication is strongly recommended. If you don’t use the ZENworks PBA, encrypted data is protected only by Windows authentication.

For more information about ZENworks Pre-Boot Authentication, see the *ZENworks Full Disk Encryption PBA Reference*

Policy Assignments

You can assign Disk Encryption policies to devices and device folders. You cannot assign a Disk Encryption policy to device groups, users, user groups, or user folders.

A device can apply only one Disk Encryption policy. When multiple Disk Encryption policies are assigned to a device through different objects (the device, group, or folder), the Full Disk Encryption Agent must determine a single *effective policy* to enforce on the device. Effective policies are discussed in “Effective Policy” on page 9.

Effective Policy

Disk Encryption policies can be assigned to devices and device folders. This means that a device could have both direct assignments (on the device) and inherited assignments (from its membership in folders). However, only one policy is the effective policy that is applied to the device.

The effective policy is determined by where the assignment occurs in the ZENworks management hierarchy, using the following order of precedence:

1. Device
2. Device Folder

In general, the policy “closest” to the device becomes the effective policy. This means that a policy assigned to the device precedes a policy assigned to one of the folders in which the device resides.

The order of precedence also takes into account that each level of the hierarchy includes multiple sublevels. For example, if a device resides in a subfolder of the **Workstations** root folder, it might inherit assignments from both folders. The following table expands the levels to show the complete order of precedence:

Level	Order of Precedence	Example	Details
Device	1. First policy listed 2. Second policy listed 3. Third policy listed	1. Policy B 2. Policy A	When two or more Disk Encryption policies are assigned directly to the device, the effective policy is determined by which one is listed first (at the top) in the device’s **Assigned Policies** list in ZENworks Control Center. In the example, Policy B is the effective policy because it precedes Policy A.

Level	Order of Precedence	Example	Details
Folder	1. Device folder a. First policy listed b. Second policy listed 2. Parent folder a. First policy listed b. Second policy listed 3. Root folder a. First policy listed b. Second policy listed	1. Device Folder a. Policy I b. Policy H 2. Parent Folder a. Policy K 3. Root folder a. Policy R b. Policy S	When a device does not have a direct Disk Encryption policy assignment, it inherits its effective policy from the device folder structure. The effective policy is determined by selecting the first policy in the **Assigned Policies** list of the folder closest to the device object. In the example, the effective policy is Policy I because it is the first policy in the folder where the device resides. If the device's folder did not have any Disk Encryption policy assignments, the effective policy would be Policy K.

Policy Versioning

A Disk Encryption policy can have multiple versions. Only one version, called the *published* version, is active at any one time.

When you change the published version of a policy, a Sandbox version is created. The published version remains active until you publish the Sandbox version, at which time the Sandbox version becomes active as the new published version. All old versions are retained until you delete them.

For information about publishing different versions of a policy, see "Publishing Policies" on page 38.

2 Policy Deployment

The following sections provide information about deploying Disk Encryption policies to devices in your ZENworks Management Zone.

Deployment Best Practices

As you deploy Disk Encryption policies to devices, we strongly recommend that you implement the practices in the sections below to achieve the best results.

Encryption Recommendations

The following recommendations apply to the encryption settings for a Disk Encryption policy:

- **Local Fixed Volumes:** You can encrypt all volumes or selected volumes. If possible, encrypt all volumes. If you specify volumes, the drive volumes must be the same an all target devices (for example, C: on all devices).
- **Encryption:** Use the default algorithm (AES) and key length (256) unless your organization requires a different algorithm and key length. For fastest initial encryption of a device, enable the **Encrypt only the used sectors of the drive** option. After initial deployment, additional data written to the disk is automatically encrypted.
- **Reboot Behavior:** Force a reboot but provide a reasonable time out before the reboot. Provide a custom message with the reboot. Run Windows check disk during the reboot to ensure disk integrity.

Pre-Boot Authentication Recommendations

The following recommendations apply to the pre-boot authentication settings for a Disk Encryption policy:

- **Single Sign-On:** Activate single sign-on. This enables credentials to be entered one time at the PBA login screen and passed to both the Windows login and ZENworks login.
- **User ID/Password Authentication:** If you enable user ID/password authentication, the following recommendations apply:
  - Populate the PBA Users list with IT administrators and key personnel that should always have access to the data on the device.
  - Enable user capturing so that the ZENworks PBA captures the credentials of the first user to log in to Windows after the policy is applied. The captured credentials can be used to log in to the PBA and Windows.

- **Smart Card Authentication:** If you enable smart card authentication, the following recommendations apply:
  - A Disk Encryption policy can specify only one smart card reader and one PKCS#11 provider. If you have devices with different readers or providers, create different policies for the devices.
  - Enable smart card user capturing so that the ZENworks PBA captures the smart card credentials of the first user to log in after the policy is applied. The captured credentials can be used to log in to the PBA and Windows.
- **Reboot Behavior:** Force a reboot but provide a reasonable time out before the reboot. Provide a custom message with the reboot. Be aware that encryption of the target volumes does not start until this final reboot occurs.
- **Lockout:** Don't use lockout settings unless your organization requires it. Leave the PBA keyboard layout set to auto detect so that the layout is determined the Windows operating system locale.

Image Devices Before Applying Policies

ZENworks Full Disk Encryption encrypts data. As with any encryption product, you should image target devices prior to performing encryption. With ZENworks Full Disk Encryption, this means that you should image a device before assigning a Disk Encryption policy to it. If encryption or pre-boot authentication fails on a device, you can apply the image to restore the device.

For information about using ZENworks Configuration Management to image devices, see the *ZENworks 2017 Preboot Services and Imaging Reference*.

Test Policies Before Assigning Them to Production Devices

Before applying a policy to production devices, apply the policy to test devices that have the same hardware configurations as the production devices.

This practice is essential if the policy installs ZENworks Pre-Boot Authentication to devices. After successful pre-boot authentication occurs, the ZENworks PBA must configure the BIOS settings correctly for Windows. With some hardware configurations, the standard boot method and Linux kernel configuration used by the PBA might not work, resulting in hardware that does not function correctly or is not recognized by Windows. In some cases, the device might not boot to Windows.

As part of the Disk Encryption policy, you can customize the DMI (Direct Media Interface) file to provide compatibility for hardware configurations that are not supported. This is a discovery process that can require multiple customization attempts. The easiest way to recover from a failed attempt is to reimage the device (see Image Devices Before Applying Policies).

For information about testing devices, see "Testing a Disk Encryption Policy" on page 26.

Control the Policy Assignments

A Disk Encryption policy can be assigned to devices or to device folders. A device inherits any Disk Encryption policies assigned to the folders in which the device resides. It then applies the policy that is closest to it. For example, if a policy is assigned to a device and another policy is assigned to the device's parent folder, the device applies the policy assigned to it and ignores the folder-assigned policy. For more information, see "Effective Policy" on page 9.

Because of the system requirements and hardware support considerations for Full Disk Encryption, we strongly recommend that folder assignments be used with caution. Before assigning a Disk Encryption policy to a folder, you should ensure that all devices within the folder (and subfolders) can support the policy. If a device cannot, you can move the device to another folder or assign an appropriate Disk Encryption policy directly to the device.

The same policy can be applied to devices with standard hard disks and devices with self-encrypting hard disks. With self-encrypting hard disks, the Full Disk Encryption Agent ignores the encryption settings and only applies the pre-boot authentication settings.

Do Not Encrypt SCSI or RAID Hard Disks

Encryption of SCSI and RAID hard disks is not supported. If you target a SCSI or RAID hard disk for encryption, the device becomes unbootable. To recover the device, you must use a use a bootable operating system CD (or Windows PE or Bart PE) to delete the ZENworks (NAC) partition created when the policy was applied.

If possible, you should disable or uninstall the Full Disk Encryption Agent on devices with SCSI or RAID hard disks. For instructions, see "Uninstalling the Full Disk Encryption Agent" in the *ZENworks Full Disk Encryption Agent Reference*.

Make Sure the ERI File is Uploaded to the ZENworks Server

After the Disk Encryption policy is applied to a device, the Full Disk Encryption Agent generates an emergency recovery information (ERI) file. If the agent has contact with the ZENworks Server, the file (and its auto-generated password) is uploaded to the server. Otherwise, the agent retries the upload every 5 minutes until it is successful.

The ERI file and password are required to recover the device if a problem occurs. Without the ERI, the device's encrypted data is lost. You should ensure that the agent has network access to the ZENworks Server so that the file is uploaded as soon as possible. To verify that the file is uploaded, use ZENworks Control Center to view the **Emergency Recovery Information** list for the device (ZENworks Control Center > Devices > device object > Emergency Recovery).

Remove and reapply the policy after adding a new disk drive or volume

When you apply a Full Disk Encryption policy to a device, you have the option to encrypt all local fixed volumes or specify the volumes that will be encrypted. Once the policy is applied, the specified volumes are encrypted.

If you add a new disk drive to the device, or you want to specify another volume on the device for encryption, the policy must be removed, including disk decryption, and then be reapplied to recognize the new volumes. If the existing policy is not set to encrypt all local fixed volumes, you need to edit the Local Fixed Volumes setting in the policy to recognize the new volumes before reapplying the policy and encrypting the drives.

Creating a Disk Encryption Policy

The Disk Encryption policy lets you configure both full disk encryption and pre-boot authentication for a device.

When applied to a device with a standard hard disk, the Disk Encryption policy can provide both full disk encryption and pre-boot authentication. When applied to a device with a self-encrypting hard disk, the full disk encryption settings are ignored and only the pre-boot authentication is enforced.

The following sections explain how to create a new Disk Encryption policy by using the Create New Policy Wizard.

In addition to using the wizard, you can create policies by:

- Copying an existing Disk Encryption policy. All original system requirements, details, and settings are copied to the new policy. You can then make any desired modifications to the new policy. See "Copying a Policy" on page 39.
- Creating a Sandbox version of an existing Disk Encryption policy and then publishing it as a new policy. For information, see "Publishing a Sandbox Version" on page 38.

Creating a Policy

To create a Disk Encryption policy by using the Create New Policy Wizard:

1. In ZENworks Control Center, click **Policies** to display the Policies page.
2. In the Policies panel, click **New** > **Policy** to launch the Create New Policy wizard.
3. On the Select Platform page, select **Windows**, then click **Next**.
4. On the Select Policy Category page, select **Windows Full Disk Encryption Policies**, then click **Next**.
5. On the Select Policy Type page, select **Disk Encryption Policy**, then click **Next**.
6. On the Define Details page, specify a name for the policy, select the folder in which to place the policy, then click **Next**.

   The name must be unique among all other policies located in the selected folder. For additional requirements, see "Naming Objects in ZENworks Control Center" in the *ZENworks 2017 Control Center Reference*.
7. Proceed with the wizard to define the details of the policy. Refer to the following sections for detailed information about each page of information you must supply:
   - "Configure Disk Encryption - Volumes, Algorithm, and Emergency Recovery" on page 15
   - "Configure Disk Encryption - Admin Password and Encryption Initialization" on page 17
   - "Configure Pre-Boot Authentication Methods" on page 19
   - "Configure Pre-Boot Authentication - Reboot and Lockout" on page 22
   - "Configure Pre-Boot Authentication - Hardware Compatibility" on page 24

8 After you have defined the details listed above and are at the Summary page, review the information to make sure it is correct. If it is incorrect, click the **Back** button to revisit the appropriate wizard page and make changes. If it is correct, select either of the following options (if desired), then click **Finish** to create the policy.

- **Create as Sandbox:** Select this option to create the policy as a Sandbox version. The Sandbox version is isolated from devices until you publish it. For example, you can assign it to devices, but it is applied only after you publish it. You can also use the Sandbox version to test the policy on devices you've designated as test devices. For information, see "Testing a Disk Encryption Policy" on page 26.
- **Define Additional Properties:** Select this option to display the policy's property pages. These pages let you define system requirements that must be met before the policy can be assigned to a device, assign the policy to devices, and add the policy to policy groups.

9 To test the policy before assigning it to devices, see "Testing a Disk Encryption Policy" on page 26.

10 To assign the policy to devices, see "Assigning a Disk Encryption Policy" on page 26.

Configure Disk Encryption - Volumes, Algorithm, and Emergency Recovery

ZENworks Full Disk Encryption supports encryption of IDE, SATA, and PATA hard disks. Encryption of SCSI hard disks is not supported; encrypting a SCSI drive can cause the device to become unbootable.

The information in this section assumes that you are on the Configure Disk Encryption - Volumes, Alogorithm and Emergency Recovery page of the Create New Disk Encryption Policy wizard. If you are not, see "Creating a Policy" on page 14 for instructions about how to get there.

The Volumes, Alogorithm and Emergency Recovery page lets you specify which disk volumes on a device to encrypt and the algorithm to use for the encryption. In addition, you can choose whether or not to allow users to create Emergency Recovery Information (ERI) files that can be used to regain access to encrypted volumes if a problem occurs with the device.

Local Fixed Volumes

Any of a device's local fixed disk volumes can be encrypted. Removable disks, such as thumb drives, cannot be encrypted. Neither can non-local disks, such as network drives.

- **Encrypt all local fixed volumes:** Select this option to encrypt all volumes.
- **Encrypt specific local fixed volumes:** Select this option to limit encryption to specific volumes. To specify a volume, click **Add**, then select the drive letter assigned to the volume. If a volume that you specify does not exist on a device to which the policy is assigned, or the specified volume is not a local fixed volume, no encryption of the specified volume takes place.

After the policy is applied, encryption of the target volumes is performed sequentially, one volume at a time. A maximum of 10 volumes are encrypted, even if the device has more than 10.

Encryption Settings

Encryption is the process of converting plain-text data into cipher text that can then be decrypted back into its original plain text. An encryption algorithm, also known as a cipher, is a set of steps that determines how an encryption key is applied to the plain-text data to encrypt and decrypt the text.

The following settings determine the algorithm that is used to encrypt the selected fixed volumes, and the length of the encryption key that is used in the encryption process.

- **Algorithm:** Select one of the following encryption algorithms:
  - **AES:** The AES (Advanced Encryption Standard) algorithm is a symmetric-key encryption standard adopted by the U.S. government. AES has a 128-bit block size with key lengths of 128, 192, and 256 bits.

    AES provides the highest security coupled with fast encryption speed. This algorithm is the optimal choice for most users.
  - **Blowfish:** The Blowfish algorithm is a symmetric-key block cipher. It has a 64-bit block size with key lengths of 32 to 448 bits. It is a strong, fast, and compact algorithm.
  - **DES:** The DES (Data Encryption Standard) algorithm is a symmetric-key encryption standard that uses a 56-bit key.

    Because of its 56-bit key size, DES is not as secure as AES or Blowfish. DES keys have been broken in less than 24 hours.
  - **DESX:** The DESX algorithm is a variant of the DES algorithm. It uses a 128-bit key.
- **Key Length:** Select a key length. Key lengths vary depending on the encryption algorithm you select. We recommend that you choose the maximum key length for the algorithm. Doing so provides the highest security with no significant performance loss.
- **Encrypt only the used sectors of the drive:** During initial encryption of a fixed disk volume, all of the sectors are encrypted unless you select this option. If you select this option, only the sectors that contain data are encrypted. Additional sectors are encrypted as they are used.

  Encrypting all sectors (used and unused) greatly increases the initial encryption time. You should only encrypt unused sectors if you are concerned about unauthorized users possibly recovering previously deleted files from the unused (and unencrypted) sectors.
- **Block 1394 (FireWire) port:** The 1394 interface provides direct memory access, or DMA. Direct access to system memory can compromise security by providing read and write access to stored sensitive data, including encryption and authentication data used by ZENworks Full Disk Encryption. Select this option to prevent direct access to memory through the 1394 port.
- **Enable software encryption of Opal compliant self-encrypting drives:** When enabled, this option does the following to OPAL 2.0 compliant self-encrypting drives:
  - Prevents the ZENworks Pre-Boot Authentication (PBA) mechanism from initiating the drive's locking feature. This allows the ZENworks PBA to work with *ALL* OPAL 2.0 compliant self-encrypting drives, not just the drives that are known to be drive-locking compatible with ZENworks Full Disk Encryption.
  - Applies software encryption to the drive, adding a second layer of encryption to the drive's already hardware-encrypted contents.

  NOTE: This setting is automatically applied to enforced encryption policies when upgrading from ZENworks 11.3.x to 11.4.x or later versions. However, you must remove the policy and Full Disk Encryption Agent during the upgrade process. For more information, see Full Disk Encryption policy fails on Opal devices during version upgrade.

- **Enable encryption lockdown:** Prevents drive decryption when a Full Disk Encryption policy is removed from a device, unless this setting is disabled before the policy is removed.

Once a policy is enforced on a device with encryption lockdown enabled, it can be disabled in one of three different ways:

- Click the Disk Encryption policy in the Policies page, go to **Details** > **Disk Encryption**, deselect **Enable encryption lockdown** in Encryption Settings, and click **Apply**.
- Select the check box for the device on the Devices page that has encryption lockdown enabled, and select **FDE-Force Device to Decrypt** in the Quick Tasks drop-down menu.
- Use the **Decrypt Drives** command from the ZENworks Full Disk Encryption Agent Commands feature on the device itself in the **ZENworks Agent** > **ZENworks Full Disk Encryption** dialog box.

Emergency Recovery Information (ERI) Settings

An Emergency Recovery Information (ERI) file is required to regain access to encrypted volumes if a problem occurs with the device. When the policy is applied to a device, or the policy changes, an ERI file is automatically created and uploaded to the ZENworks Server. You can also enable users to manually create ERI files and store them locally.

- **Allow user to create ERI files:** Select this option to enable users to create ERI files. This is done through the ZENworks Full Disk Encryption Agent's About box.
- **Require user to provide a strong password when creating an ERI file:** The ERI file is password-protected to ensure that no unauthorized users can use it to gain access to the encrypted device. The user enters the password when creating the file. Select this option to force the user to provide a password for the file that meets the following requirements:
  - Seven or more characters
  - At least one of each of the four types of characters:
    - uppercase letters from A to Z
    - lowercase letters from a to z
    - numbers from 0 to 9
    - at least one special character ~ ! @ # $ % ^ & * ( ) + { } [ ] : ; < > ? , . / - = | \ ”

  For example: qZG@3b!
- **Use common password for system-generated ERI files:** When this option is selected, all system-generated ERI files will use the password that is specified in this setting.

Configure Disk Encryption - Admin Password and Encryption Initialization

The information in this section assumes that you are on the Configure Disk Encryption - Admin Password and Encryption Initialization page of the Create New Disk Encryption Policy wizard. If you are not, see “Creating a Policy” on page 14 for instructions about how to get there.

The Admin Password and Encryption Initialization page lets you specify an Administrator password for the ZENworks Full Disk Encryption Agent and determine when a device is rebooted to initiate the encryption of the device’s volumes.

Admin Password

The Administrator password enables access to the Administrator options in the Full Disk Encryption Agent. These options help you see the current status of the agent and view the assigned Disk Encryption policy, as well as troubleshoot problems with the agent or policy.

To set the password, click **Set**, specify the password, then click **OK**.

If you ever need to allow a user to access the Administrator options, we recommend that you use the Password Key Generator utility to generate a password key. The key, which is based on the FDE Admin password, functions the same as the FDE Admin password but can be tied to a single device or user and can have a usage or time limit.

The Password Key Generator utility is accessible under the **Configuration Tasks** list in the left navigation pane.

Reboot Options

When the Disk Encryption policy is applied to a device, the device's disks cannot be encrypted until the device reboots and loads the Full Disk Encryption Agent's encryption drivers.

- **Reboot Behavior:** Select one of the following:
  - **Force device to reboot immediately:** Reboots the device immediately after the Disk Encryption policy is applied.
  - **Do not reboot device:** Does not force a reboot after the Disk Encryption policy is applied. The user must initiate a reboot before disk encryption can occur.
  - **Force device to reboot within *XX* minutes:** Reboots the device within the specified number of minutes after the Disk Encryption policy is applied. Providing a reboot delay can give the user time to save work prior to the reboot. The default delay is 5 minutes.
- **Display predefined message to user before rebooting:** If you selected the **Do not reboot device** option or the **Force device to reboot within XX minutes** option, you can display a message to the user. The **Force device to reboot immediately** option does not support a message.

  Select this option to display the following message:

  ```
  ZFDE Policy Enforcement

  Your ZENworks Administrator has assigned a Disk Encryption policy to your computer. To enforce the policy, your computer must be rebooted.
  ```
- **Override predefined message with custom message:** This option is available only after you select the **Display predefined message to user before rebooting** option. It lets you override the predefined message with your own custom message. Select the option, then specify a title for the message window and the text to include in the message body.

CheckDisk Options

We strongly recommend that you run Windows CheckDisk with Repair during the reboot. The disk check and repair is performed on the system volume (C: drive), ensuring that system and partition records are error-free prior to encrypting the target volumes.

This option is selected by default. If you are sure that the target volumes are in perfect condition (for example, the disks are new), you can select the **Do not run Windows check disk** option.

NOTE: This setting does not apply to Windows XP. On Windows XP, CheckDisk is run if it is needed regardless of the setting.

Configure Pre-Boot Authentication Methods

The information in this section assumes that you are on the Configure Pre-Boot Authentication Methods page of the Create New Disk Encryption Policy wizard. If you are not, see “Creating a Policy” on page 14 for instructions about how to get there.

Encrypted data is available only after a user successfully authenticates to Windows on a device. If Windows authentication is not sufficient for your security requirements, you can enable ZENworks Pre-Boot Authentication (PBA) to add another layer of access protection.

The ZENworks PBA is a Linux-based component. When the Disk Encryption policy is applied to a device with a standard hard disk, a 100 MB partition containing a Linux kernel and the ZENworks PBA is created on the hard disk. When the policy is applied to a device with a self-encrypting hard disk, the Linux kernel and ZENworks PBA are installed to the disk’s datastore memory.

During normal operation, the device boots to the Linux partition and loads the ZENworks PBA. As soon as the user provides the appropriate credentials (user ID/password or smart card), the PBA terminates and the Windows operating system boots, providing access to the encrypted data on the previously hidden and inaccessible Windows drives.

The Linux partition is hardened to increase security, and the ZENworks PBA software is protected from alteration through the use of MD5 checksums and strong encryption for authentication keys.

ZENworks Pre-Boot Authentication

Select this option to enable the ZENworks PBA. This adds an additional layer of access protection before the standard Windows login.

Authentication Methods

These settings let you configure the methods that can be used for authenticating to a device’s encrypted disks. If you have enabled ZENworks Pre-Boot Authentication, you must select at least one of the methods.

- **Enable user ID/password authentication:** Select this option to enable users to authenticate via a user ID and password. If you select this option, you must configure the settings in the **User ID/Password Authentication Settings** section.
- **Enable smart card authentication:** Select this option to enable users to authenticate via a smart card. If you select this option, you must configure the settings in the **Smart Card Authentication Settings** section.
- **Default Authentication Method:** If you enable both the user ID/password and smart card authentication methods, you must select the default method. Both methods are available to a user during pre-boot authentication, but the default method is presented if the user does not select a method within the allotted time.
- **Activate Single Sign-On for ZENworks PBA and Windows Login:** Select this option to activate single sign-on for the PBA and Windows login. The user logs in to the PBA and the PBA handles the login to the Windows operating system. Single sign-on applies to both authentication methods (user ID/password or smart card).

User ID/Password Authentication Settings

If you selected **Enable user ID/password authentication** as one of the supported authentication methods, configure the following settings:

- **During PBA login, show user name of last successful logged-in user:** Select this option to prepopulate the **User ID** field of the PBA login screen with the username of the last user who logged in to the PBA. This is convenient for the device's primary user, but weakens security by providing unauthorized users with a valid user ID.
- **Create PBA account for first user who logs in to Windows after the policy is applied (User Capturing):** Select this option to automatically capture the credentials of the first user to authenticate after the policy is applied. During the first reboot after the policy is applied, the Windows login is displayed and the PBA captures the credentials provided for the Windows login. During subsequent reboots, the PBA login is displayed and accepts the captured credentials.

  Captured credentials exist only on the device where they are captured. The credentials are not stored with this policy.

  If a device has multiple users, the PBA captures only the first user to log in after the policy is applied. You can capture additional users by using the **FDE - Enable Additive User Capturing** quick task for the device. When the quick task is applied to a device, it activates the user capturing mode for the next reboot. To use the quick task, select the device in **Devices** > **Workstations**, then click **Quick Tasks** > **FDE - Enable Additive User Capturing**.

  NOTE: The PBA captures the credentials of the first user to authenticate after reboot, whether the credentials are user ID/password or smart card. The PBA login screen allows the user to switch between user ID/password login and smart card login. If a device supports both types of login, you should make sure the device's user logs in with the user ID/password and not the smart card. Otherwise, the smart card credential is captured and the user cannot log in via the user ID/password. This becomes a problem if you have not enabled smart cards as an authentication method (see Authentication Methods) because the user cannot log in.

- **Allow access for the following users:** User capturing is the recommended way to create a PBA account for a device's users. However, you can enable this option and use the **PBA Users** list to define PBA user accounts.

  All accounts that you add to the **PBA Users** list are created on all devices to which the policy is applied. Because of this, the **PBA Users** list is a good way to give Administrators access to each of the devices. For example, if you have a common Windows Administrator account that you use across devices, you can add the Windows Administrator as a PBA user. You can then log in to both the PBA and Windows on a device by using the Administrator account and password.

  To add a PBA user account, click **Add**, then fill in the following fields:

  - **Replace password if user already exists in PBA:** When the policy is applied, if the user you are adding matches an existing PBA user (for example, a user added by a previously applied Disk Encryption policy), the existing user account is retained, including the existing password. Select this option to replace the existing password with the one you specify in this dialog box.
  - **User Name:** Specify a user name for the PBA user. If single sign-on is active, this user name must be the same as the Windows user name. If single sign-on is not active, the user name does not need to match the Windows user name.
  - **Domain:** Specify a domain name for the PBA user. If single sign-on is active, this must be the Windows domain name or workgroup name. If single sign-on is not active, this field is optional. You can leave it blank or use it as another component to further distinguish the PBA user name.

- **Password:** Specify a password for the PBA user. If single sign-on is active, this must be the Windows password. If single sign-on is not active, you can specify any password.

- **Remove existing users from PBA if not in this list:** Select this option to remove any user accounts from the PBA that are not listed in the **PBA Users** list. Because captured users do not display in the list, they are also removed.

Smart Card Authentication Settings

If you selected **Use smart card authentication** as one of the supported authentication methods, configure the smart card settings.

- **Smart Card Reader:** Select the card reader used by the devices to which this policy will be assigned.
- **PKCS#11 Provider:** Select the PKCS #11 provider used by the devices to which this policy will be assigned.
- **Create PBA account for first smart card user who logs in to the ZENworks PBA after the policy is applied (User Capturing):** Select this option to automatically capture the credentials of the first user to authenticate after the policy is applied. During the first reboot after the policy is applied, the PBA login screen is displayed and the user is prompted for the smart card. The PBA captures the smart card credentials (certificate and PIN). During subsequent reboots, the PBA login accepts the captured smart card credentials.

  If a device has multiple users, the PBA captures only the first user to log in after the policy is applied. You can capture additional users by using the **FDE - Enable Additive User Capturing** quick task for the device. When the quick task is applied to a device, it activates the user capturing mode and creates a PBA account for the next user who logs in. To use the quick task, select the device in **Devices** > **Workstations**, then click **Quick Tasks** > **FDE - Enable Additive User Capturing**.

  NOTE: The PBA captures the credentials of the first user to authenticate after reboot, whether the credentials are smart card or user ID/password. The PBA login screen allows the user to switch from smart card login to user ID/password login, but you should make sure the device's user logs in with the smart card and not the user ID/password. Otherwise, the user ID/password credential is captured and the user cannot log in via the smart card. This becomes a problem if you have not enabled user ID/password as an authentication method (see Authentication Methods) because the user cannot log in.

- **Allow certificate content to be used for authentication:** User capturing is the recommended way to create a PBA account for smart card users because it accurately captures the smart card certificate information. If you don't enable user capturing, you must manually define certificates that can be used for authentication. If you do enable user capturing, you can still manually define additional certificates that allow access.

  To define a certificate, click **Add**, fill in the following fields, then click **OK** to add the certificate to the list:

  - **Certificate Name:** Specify a name to identify the certificate in this policy. This is simply a display name and does not need to match the certificate file name or any other certificate property.
  - **Certificate Content:** Open the certificate in a text editor, then cut and paste the contents of the certificate into this box. You must use an X.509 certificate (`*.cer`; base64-encoded).

- **Remove existing certificates from PBA if not in this list:** Select this option to remove any certificates from the PBA that are not listed in the **Certificates** list. Because captured certificates do not display in the list, they are also removed.

- **Allow certificate key usages to be used for authentication:** In addition to enabling user capturing or defining the certificates that can be used for authentication, you need to further identify the certificates via key usages (this setting) or labels (the **Allow certificate labels to be used for authentication** setting). This adds a second layer of security to the certificate authentication.
  - **Key Usages:** Key usages define the purposes for which a certificate's public key can be used, such as Data Encipherment or Digital Signature. You can view a certificate's key usages by using Microsoft Certificate Manager (available as a snap-in to Microsoft Management Console).

    To add a certificate's key usages to the list, click **Add**, select the desired usages (Shift-click or Ctrl-click to select multiple usages), click the arrow to move the selected items to the **Selected List** box, then click **OK**.

    If you add more than one key usage, the PBA evaluates the key usages against the certificates in the order the usages are listed, from top to bottom. You can use **Move Up** and **Move Down** to change the order of the key usages in the list.
  - **Match policy:** The match policy determines how many of the defined key usages must be contained in the smart card's certificate in order for the match to be made and authentication to take place. Select one of the following options:
    - **Any:** The certificate must contain at least one of the listed key usages.
    - **All:** The certificate must contain all of the listed key usages.
    - **None:** The certificate cannot contain any of the listed key usages. This option lets you use the Key Usages list as an exclusion list (blacklist) rather than an inclusion list (whitelist).
- **Allow certificate labels to be used for authentication:** In addition to enabling user capturing or defining the certificates that can be used for authentication, you need to further identify the certificates via labels (this setting) or key usages (the **Allow certificate key usages to be used for authentication** setting). This adds a second layer of security to the certificate authentication.

  A certificate label is a property defined within the certificate. You need to use the PKCS #11 middleware provider software to view the certificate label.

  To add a certificate label to the list, click **Add**, specify the label (case-sensitive), then click **OK**.

  If you add more than one label, the PBA attempts to match the first label in the list to a certificate on the authenticating smart card. If no match occurs, the second label is tested, then the third label, and so on until a match occurs or authentication fails. You can determine the order of the labels in the list by selecting a label and clicking **Move Up** or **Move Down** to reposition it in the list.

Configure Pre-Boot Authentication - Reboot and Lockout

The information in this section assumes that you are on the Configure Pre-Boot Authentication - Reboot and Lockout page of the Create New Disk Encryption Policy wizard. If you are not, see "Creating a Policy" on page 14 for instructions about how to get there.

The Reboot and Lockout page lets you determine when the device is rebooted after initialization of the ZENworks PBA; the first pre-boot authentication does not occur until the device reboots. It also lets you specify the number of times a user can enter the incorrect PBA login information before being locked out.

Reboot Options

Both the ZENworks PBA and the Full Disk Encryption Agent's encryption drivers are initialized the first time the device reboots after the Disk Encryption policy is applied. However, the ZENworks PBA requires an additional reboot to facilitate user capturing (if enabled) or authentication of a predefined user. In addition, encryption of the target volumes does not begin until this reboot occurs.

The following options let you specify how you want this second reboot to occur:

- **Reboot Behavior:** Select one of the following:
  - **Force device to reboot immediately:** Reboots the device immediately after the PBA is initialized.
  - **Do not reboot device:** Does not force a reboot after the PBA is initialized. The user must initiate a reboot before user capturing or predefined user authentication can occur.
  - **Force device to reboot within *XX* minutes:** Reboots the device within the specified number of minutes after the PBA initializes. The default delay is 5 minutes.
- **Display predefined message to user before rebooting:** If you selected the **Do not reboot device** option or the **Force device to reboot within XX minutes** option, you can display a message to the user. The **Force device to reboot immediately** option does not support a message.

  Select this option to display the following message:

  ```
  ZFDE Policy Enforcement

  Your ZENworks Administrator has assigned a Disk Encryption policy to your
  computer. To enforce the policy, your computer must be rebooted.
  ```
- **Override predefined message with custom message:** This option is available only after you select the **Display predefined message to user before rebooting** option. It lets you override the predefined message with your own custom message. Select the option, then specify a title for the message window and the text to include in the message body.

Lockout Settings

The Lockout settings apply to the ZENworks PBA login.

- **Enable lockout for failed logins:** Select this option to enable the PBA to lock out users based on failed login attempts, then configure the following settings:
  - **Maximum Number of Failed Logins:** Specify the maximum number of failed logins to allow before the lockout is enforced (the default is 10). When the maximum number of failed logins is reached, the device is locked. A PBA override must be performed to access the device and reset the failed login count. See "PBA Override" in the *ZENworks Full Disk Encryption PBA Reference* for more information.
  - **Failed Logins after which Login is Delayed:** Specify the number of failed logins to allow before delaying subsequent logins (the default is 3). When the specified number of failed logins is reached, each failed login attempt results in a 2 minute delay before the next attempt can be made. Make sure to specify a number that is less than the one entered in the **Maximum Number of Failed Logins** field.

  For example, using the defaults of 10 and 3 for the two settings, 10 failed logins are allowed before lockout, but after the third failed login all subsequent login attempts are delayed by 2 minutes.
- **PBA Keyboard Layout:** Select the keyboard layout used for authentication.

Configure Pre-Boot Authentication - Hardware Compatibility

The information in this section assumes that you are on the Configure Pre-Boot Authentication - Hardware Compatibility page of the Create New Disk Encryption Policy wizard. If you are not, see “Creating a Policy” on page 14 for instructions about how to get there.

After pre-boot authentication occurs, the BIOS settings must be correctly set for Windows. With older or unusual hardware configurations, the standard ZENworks PBA boot method and Linux kernel configuration used to provide the BIOS settings might not work, resulting in hardware that does not function correctly or is not recognized by Windows.

The Hardware Compatibility page provides support for older or unusual hardware configurations. These configurations might include the following:

- Hardware that does not function correctly or is no longer recognized under Windows after successful pre-boot authentication.This failure occurs because not all of the BIOS settings can be correctly handled and set for Windows.
- New hardware that is not yet natively supported.
- Poorly programmed BIOS implementations.

This hardware compatibility support applies only to software encrypted disks; self-encrypting hard disks are not supported. In addition, some devices might not support the boot methods or Linux kernel configurations used to provide hardware compatibility.

About Hardware Compatibility

Hardware compatibility is enabled through the use of two alternative boot methods and an alternative Linux kernel that supplies ACPI (Advanced Configuration and Power Interface) support. These alternative boot methods and kernel are defined through the use of a DMI (Direct Media Interface) file. The predefined file includes the following default setting:

```
[default]
KICKSTART=FAST
```

This default setting uses the standard boot method (KICKSTART=FAST) and no alternative Linux kernel. It is applied to all hardware configurations unless a configuration is explicitly defined in the file.

The predefined file also includes explicit settings for hardware configurations with known issues. For example:

```
[FUJITSU SIEMENS,LIFEBOOK C1110]
DMI_SYS_VENDOR=FUJITSU SIEMENS
DMI_PRODUCT_NAME=LIFEBOOK C1110
KICKSTART=BIOS
```

This setting applies to the Fujitsu Siemens Lifebook C1110 laptop. It applies a different boot method (KICKSTART=BIOS) that involves rebooting the computer a second time so that the BIOS hardware settings can be passed to Windows. It does not use an alternative Linux kernel configuration.

The following example uses both an alternative boot method and Linux kernel configuration:

```
[LENOVO, 417152U]
DMI_SYS_VENDOR=LENOVO
DMI_PRODUCT_NAME=417152U
KICKSTART=KEXEC
KERNEL=/boot/bzImage-acpi
```

This setting applies to the Lenovo ThinkPad T420s laptop. It applies a different boot method (KICKSTART=KEXEC) that is similar to KICSTART=BIOS but does not require a second reboot. It uses an alternative kernel configuration that enables ACPI support.

Discovering Hardware Information

Before you can add a hardware configuration to the DMI file, you must know the hardware configuration. ZENworks provides a utility, `DMICONFIG`, to discover this information.

1. Go to the device whose hardware configuration you want to discover.
2. Open a command shell (run as Administrator) and run `c:\windows\nac\sbs\dmiconfig dump`.
3. Write down the configuration lines that were dumped to the screen.

Editing the DMI File

If you are adding a hardware configuration, make sure you have the configuration information (see Discovering Hardware Information).

On the Hardware Compatibility page of the Create New Wizard Policy:

1. Click **Edit**.
2. Add the hardware information.
3. Add the KICKSTART line with the method you want to use:
   - **KICKSTART=FAST:** This is the standard method used by the ZENworks PBA.
   - **KICKSTART=BIOS:** This method is for systems that have unusual hardware configurations that are not supported by the standard FAST method. This method reboots the computer a second time so that the BIOS hardware settings can be passed to Windows.
   - **KICKSTART=KEXEC:** This method is similar to KICKSTART=BIOS but does not require a second reboot.
4. If you want to boot the computer using the alternative Linux kernel (with ACPI support), add the following line:

   `KERNEL=/boot/bzImage-acpi`
5. Include the following kernel parameters if needed:

   `KERNEL_PARAM=irqpol`

   `KERNEL_PARAM=`pci=snb-enable-ahci-to-legacy

   If both parameters are used, specify them on the same line:

   `KERNEL_PARAM=`irqpol pci=snb-enable-ahci-to-legacy
   - **irqpoll:** Alters the way that the kernel handles interrupts. This is useful if the PBA kernel log shows messages stating that an interrupt occurred.
   - **pci=snb-enable-ahci-to-legacy:** ZENworks AHCI mode kernel option that switches the chipset to ATA mode prior to performing the soft reset and booting to Windows. This parameter fixes many instances where the chipset is in AHCI mode and the soft reset fails to boot Windows.
6. Click **OK** to save your changes.

Testing a Disk Encryption Policy

To ensure that a Disk Encryption policy provides the results that you expect, we recommend that you test it on one or more devices before distributing it to all intended devices.

The best way to test a policy is to apply a Sandbox version of the policy to a test device. The following sections explain how to do this:

- “Designating Test Devices” on page 26
- “Assigning the Policy to Test Devices” on page 26

IMPORTANT: The Disk Encryption policy is supported only on devices that use the standard BIOS. Windows devices that use UEFI BIOS are not supported; if a Disk Encryption policy is assigned to a Windows UEFI device, the policy is not applied.

Designating Test Devices

You can designate any managed device in your ZENworks Management Zone as a test device. When a policy is assigned to a test device, the Sandbox version of the policy is applied, not the Published version. If no Sandbox version exists, the Published policy is applied.

To designate a managed device as a test device:

1. In ZENworks Control Center, click **Devices**.
2. In the **Devices** list, select the check box next to the target device, then click **Action** > **Set as Test**.

Assigning the Policy to Test Devices

1. In ZENworks Control Center, click **Policies** to display the Policies page.
2. Click the Disk Encryption policy you want to assign to test devices.
3. Assign the policy to test devices:
   - **3a** Click the **Relationships** tab.
   - **3b** In the Device Assignments panel, click **Add**, browse for and select the test devices, then click **OK**.
   - **3c** Select **Device Only** as the policy conflict resolution, then click **Next**.
   - **3d** Select **Enforce policies immediately on all assigned devices**, then click **Finish**.
4. Go to a test device and verify that the policy has been applied and is being enforced as expected.

Assigning a Disk Encryption Policy

You must assign a Disk Encryption policy to devices or device folders. The policy cannot be assigned to device groups, users, user groups, or user folders. Before assigning a Disk Encryption policy to devices, make sure you have reviewed “Deployment Best Practices” on page 11.

IMPORTANT: The Disk Encryption policy is supported only on devices that use the standard BIOS. Windows devices that use UEFI BIOS are not supported; if a Disk Encryption policy is assigned to a Windows UEFI device, the policy is not applied.

1. In ZENworks Control Center, click the **Policies** tab.
2. In the **Policies** list, select the check box next to the Disk Encryption policy (or policy group containing the Disk Encryption policy) you want to assign.
3. Click **Action** > **Assign to Device**.
4. Browse for and select the devices, device groups, and device folders to which you want to assign the policy:
   - **4a** Click next to a folder (for example, the `Workstations` folder or `Servers` folder) to navigate through the folders until you find the device, group, or folder you want to select.

     If you are looking for a specific item, such as a Workstation or a Workstation Group, you can use the **Items of type** list to limit the types of items that are displayed. If you know the name of the item you are looking for, you can use the **Item name** box to search for the item.
   - **4b** Click the underlined link in the **Name** column to select the device, group, or folder and display its name in the **Selected** list box.
   - **4c** Click **OK** to add the selected devices, folders, and groups to the **Devices** list.
5. Click **Next** to display the Finish page.
6. Review the information and, if necessary, use the **Back** button to make changes to the information.
7. If you want the policies to be immediately enforced on all the assigned devices, select **Enforce Policies Immediately on all Assigned Devices**.
8. Click **Finish**.

   The policies or policies groups are assigned to the selected devices, device groups, and device folders. You can view the assignments on the Relationships page of the policies or policy groups.

Policy Enforcement Workflow

After a policy is assigned to a device, the policy enforcement workflow on the device depends on the device type (standard or self-encrypting):

- “Standard Hard Disk” on page 28
- “Standard Hard Disk with Pre-Boot Authentication” on page 29
- “Self-Encrypting Hard Disk” on page 30

WARNING: When applying a full disk encryption policy, ensure that the encryption process is not interrupted prematurely with a power change on the disk drive(s); otherwise, all data on the disk can be lost due to disk corruption. You can check the encryption status on the device by accessing **Full Disk Encryption** > **About** in the ZENworks Agent.

Disk corruption due to power change has only been noted on secondary drives, but it may also be applicable to primary drives. For this reason, the following precautions are strongly recommended before applying a full disk encryption policy to a device:

- If possible, select the AES algorithm when configuring the full disk encryption policy.

Selecting the AES algorithm should preclude disk corruption from occurring in the event of a power-down during encryption. However, the additional precautions are best practices that will reduce the risk of possible disk corruption.

- Pre-configure devices receiving the policy so that power options are set to never automatically shut off, hibernate, or sleep.
- Inform all device users of the need to keep their devices running during the encryption process, to include avoiding *Sleep* and *Hibernation* options.

Standard Hard Disk

The following process occurs after a Disk Encryption policy (encryption only, no pre-boot authentication) is assigned to a device with standard hard disks:

1. The next time the ZENworks Agent refreshes it receives the Disk Encryption policy.
2. The ZENworks Full Disk Encryption Agent applies the policy to the device.
3. The device reboots according to the disk encryption reboot setting in the policy. During the reboot, the following occurs:
   - A CheckDisk occurs if the **Run Windows check disk with repair** option is enabled in the policy. On Windows XP, the operation is performed if needed even if the option is not enabled.
   - A 100 MB ZENworks partition is created. This partition is used for storage of Full Disk Encryption files and the Emergency Recovery Information (ERI) file.
   - The Full Disk Encryption drivers are initialized.
   - The user is prompted to log in to Windows.
4. The target disk volumes, as specified in the policy, are encrypted.

   Depending on the number of volumes and amount of data to be encrypted, encryption can take some time. If the device is rebooted during the encryption process, the process restarts where it left off prior to the reboot.

You can view the ZENworks Full Disk Encryption About Box to monitor the encryption process:

Standard Hard Disk with Pre-Boot Authentication

The following process occurs after a Disk Encryption policy (encryption and pre-boot authentication) is assigned to a device with standard hard disks:

1. The next time the ZENworks Agent refreshes it receives the Disk Encryption policy.
2. The ZENworks Full Disk Encryption Agent applies the policy to the device.
3. The device reboots according to the disk encryption reboot setting in the policy. During the reboot, the following occurs:
   - A CheckDisk occurs if the **Run Windows check disk with repair** option is enabled in the policy. On Windows XP, the operation is performed if needed even if the option is not enabled in the policy.
   - A 100 MB ZENworks partition is created. This partition is used for storage of encryption files, the Emergency Recovery Information (ERI) file, and the ZENworks PBA Linux kernel.
   - The Disk Encryption drivers and the ZENworks PBA are initialized.
   - The user is prompted to log in to Windows.

4. After successful Windows login, the device reboots according to the PBA reboot setting for the policy. During the reboot, the following occurs:
   - If user capturing *is* enabled, the user receives an informational prompt and then the Windows login is displayed. When the user logs in (either with userID/password or smartcard), the ZENworks PBA captures the credentials. On subsequent reboots, the user is presented with the ZENworks PBA login and must provide the captured credentials.
   - If user capturing *is not* enabled, the user is prompted to enter credentials at the PBA login screen. The user must enter valid credentials for a PBA user or smartcard defined in the policy. If single-sign on **is not** enabled, the Windows login is then displayed and the user must enter valid Windows credentials to log in.
5. After successful login, the target disk volumes, as specified in the policy, are encrypted.

   Depending on the number of volumes and amount of data to be encrypted, this can take some time. If the device is rebooted during the encryption process, the process restarts where it left off prior to the reboot.

   You can view the ZENworks Full Disk Encryption About Box to monitor the encryption process:

Self-Encrypting Hard Disk

The following process occurs after a Disk Encryption policy is assigned to a device with self-encrypting hard disks:

1. The next time the ZENworks Agent refreshes it receives the Disk Encryption policy.
2. The ZENworks Full Disk Encryption Agent applies the policy to the device.
3. ZENworks creates a 128 MB *MBR shadow* and copies the ZENworks PBA Linux kernel to it.
4. ZENworks initiates a forced shutdown of the device after the time period specified by the PBA **Force device to reboot within xx minutes** setting in the policy. If another setting (either **Force device to reboot immediately** or **Do not reboot device**) is configured as the PBA reboot option, the setting is ignored and the forced shutdown occurs after 5 minutes.

This is a hard shutdown, not a reboot. The user must power on the device after the shutdown.

5. At startup, the user receives a ZENworks Full Disk Encryption informational prompt and then the Windows login is displayed.

   During this initialization process, User Capturing and Single Sign-On are enabled regardless of the policy settings. After this one-time initialization process, the PBA enforces the User Capturing and Single Sign-On settings configured in the policy.

6. When the user logs in to Windows (either with userID/password or smartcard), the ZENworks PBA captures the credentials.

   On subsequent reboots, the user is presented with the ZENworks PBA login and can provide the captured credentials or any credentials predefined in the policy's PBA User list or Certificates list.

3 Policy Management

The following sections explain how to perform common management tasks for existing Disk Encryption policies. For information about creating Disk Encryption policies, see Chapter 2, “Policy Deployment,” on page 11.

Editing a Policy’s Details

After creating a policy, you can make changes to the policy’s details. Changing a policy’s details creates a Sandbox version of the policy. For the changes to be applied, you must publish the Sandbox version.

To edit a policy’s details:

1. In ZENworks Control Center, click **Policies** to display the Policies page.
2. In the **Policies** list, click the policy you want to edit.
3. Click the **Details** tab.
4. Make the desired changes.

   For information about the policy’s details, click the Help button in ZENworks Control Center or see “Appendix - Disk Encryption Policy Settings” on page 49.
5. Click **Apply** to save the changes.
6. To publish the changes, click **Publish**, then follow the wizard prompts.

   For more information about publishing changes to a policy, see “Publishing Policies” on page 38.

Defining a Policy’s System Requirements

You can define requirements, such as operating system, total memory, and processor speed, that a device must meet for the policy to be applied to it.

You define requirements through the use of filters. A filter is a condition that must be met by a device in order for the policy to be applied. For example, you can add a filter to specify that the device must have exactly 512 MB of RAM in order for the policy to be applied, and you can add another filter to specify that the hard drive be at least 20 GB in size.

To create system requirements for a policy:

1. In ZENworks Control Center, click the **Policies** tab.
2. Click the policy to display the policy’s Summary page.
3. Click the **Requirements** tab.
4. Click **Add Filter**, select a filter condition from the drop-down list, then fill in the fields.

   As you construct filters, you need to know the conditions you can use and how to organize the filters to achieve the desired results. For more information, see “Filter Conditions” on page 34 and “Filter Logic” on page 37.
5. (Optional) Add additional filters and filter sets.
6. Click **Apply** to save the settings.

Creating or changing system requirements creates a Sandbox version of the policy. For the requirements to be applied, you must publish the Sandbox version.

7 To publish the Sandbox version, click **Publish**, then follow the wizard prompts.

For more information about publishing the Sandbox version of a policy, see “Publishing Policies” on page 38.

Filter Conditions

You can choose from any of the following conditions when creating a filter:

- **Architecture:** Determines the architecture of Windows running on the device, either 32-bit or 64-bit. The condition you use to set the requirement includes a property, an operator, and a property value. The possible operators are equals (=) and does not equal (<>). For example, if you set the condition to architecture = 32, the device's Windows operating system must be 32-bit to meet the requirement.
- **Bundle Installed:** Determines if a specific bundle is installed. After specifying the bundle, the two conditions you can use to set the requirement are **Yes** and **No**. If you select **Yes**, the specified bundle must already be installed to meet the requirement. If you select **No**, the bundle must not be installed.
- **Connected:** Determines if the device is connected to a network. The two conditions you can use to set the requirement are **Yes** and **No**. If you select **Yes**, the device must be connected to the network to meet the requirement. If you select **No**, it must not be connected.
- **Connection Speed:** Determines the speed of the device's connection to the network. The condition you use to set the requirement includes an operator and a value. The possible operators are equals (**=**), does not equal (**<>**), is greater than (**>**), is greater than or equal to (**>=**), is less than (**<**), and is less than or equal to (**<=**). The possible values are bits per second (**bps**), kilobits per second (**Kbps**), megabits per second (**Mbps**), and gigabits per second (**Gbps**). For example, if you set the condition to `>= 100 Mbps`, the connection speed must be greater than or equal to 100 megabits per second to meet the requirement.
- **Disk Space Free:** Determines the amount of free disk space on the device. The condition you use to set the requirement includes a disk designation, an operator, and a value. The disk designation must be a local drive map (for example, c: or d:). The possible operators are equals (**=**), does not equal (**<>**), is greater than (**>**), is greater than or equal to (**>=**), is less than (**<**), and is less than or equal to (**<=**). The possible values are bytes (**Bytes**), kilobytes (**KB**), megabytes (**MB**), and gigabytes (**GB**). For example, if you set the condition to `c: >= 80 MB`, the free disk space must be greater than or equal to 80 megabytes to meet the requirement.
- **Disk Space Total:** Determines the amount of total disk space on the device. The condition you use to set the requirement includes a disk designation, an operator, and a value. The disk designation must be a local drive map (for example, c: or d:). The possible operators are equals (**=**), does not equal (**<>**), is greater than (**>**), is greater than or equal to (**>=**), is less than (**<**), and is less than or equal to (**<=**). The possible values are bytes (**Bytes**), kilobytes (**KB**), megabytes (**MB**), and gigabytes (**GB**). For example, if you set the condition to `c: >= 40 GB`, the total disk space must be greater than or equal to 40 gigabytes to meet the requirement.
- **Disk Space Used:** Determines the amount of used disk space on the device. The condition you use to set the requirement includes a disk designation, an operator, and a value. The disk designation must be a local drive map (for example, c: or d:). The possible operators are equals (**=**), does not equal (**<>**), is greater than (**>**), is greater than or equal to (**>=**), is less than (**<**), and is less than or equal to (**<=**). The possible values are bytes (**Bytes**), kilobytes (**KB**), megabytes (**MB**), and gigabytes (**GB**). For example, if you set the condition to `c: <= 10 GB`, the used disk space must be less than or equal to 10 gigabytes to meet the requirement.

- **Environment Variable Exists:** Determines if a specific environment variable exists on the device. After specifying the environment variable, the two conditions you can use to set the requirement are **Yes** and **No**. If you select **Yes**, the environment variable must exist on the device to meet the requirement. If you select **No**, it must not exist.
- **Environment Variable Value:** Determines if an environment variable value exists on the device. The condition you use to set the requirement includes the environment variable, an operator, and a variable value. The environment variable can be any operating system supported environment variable. The possible operators are **equal to**, **not equal to**, **contains**, and **does not contain**. The possible variable values are determined by the environment variable. For example, if you set the condition to `Path contains c:\windows\system32`, the Path environment variable must contain the `c:\windows\system32` path to meet the requirement.
- **File Date:** Determines the date of a file. The condition you use to set the requirement includes the file name, an operator, and a date. The file name can be any file name supported by the operating system. The possible operators are **on**, **after**, **on or after**, **before**, and **on or before**. The possible dates are any valid dates. For example, if you set the condition to `app1.msi on or after 6/15/07`, the `app1.msi` file must be dated 6/15/2007 or later to meet the requirement.
- **File Exists:** Determines if a file exists. After specifying the file name, the two conditions you can use to set the requirement are **Yes** and **No**. If you select **Yes**, the specified file must exist to meet the requirement. If you select **No**, the file must not exist.
- **File Size:** Determines the size of a file. The condition you use to set the requirement includes the file name, an operator, and a size. The file name can be any file name supported by the operating system. The possible operators are equals (**=**), does not equal (**<>**), is greater than (**>**), is greater than or equal to (**>=**), is less than (**<**), and is less than or equal to (**<=**). The possible sizes are designated in bytes (**Bytes**), kilobytes (**KB**), megabytes (**MB**), and gigabytes (**GB**). For example, if you set the condition to `doc1.pdf <= 3 MB`, the `doc1.pdf` file must be less than or equal to 3 megabytes to meet the requirement.
- **File Version:** Determines the version of a file. The condition you use to set the requirement includes the file name, an operator, and a version. The file name can be any file name supported by the operating system. The possible operators are equals (**=**), does not equal (**<>**), is greater than (**>**), is greater than or equal to (**>=**), is less than (**<**), and is less than or equal to (**<=**).

  Be aware that file version numbers contain four components: Major, Minor, Revision, and Build. For example, the file version for `calc.exe` might be 5.1.2600.0. Each component is treated independently. For this reason, the system requirements that you set might not provide your expected results. If you do not specify all four components, wildcards are assumed.

  For example, if you set the condition to `calc.exe <= 5`, you are specifying only the first component of the version number (Major). As a result, versions 5.0.5, 5.1, and 5.1.1.1 also meet the requirement.

  However, because each component is independent, if you set the condition to `calc.exe <= 5.1`, the `calc.exe` file must be less than or equal to version 5.1 to meet the requirement.
- **IP Segment:** Determines the device's IP address. After specifying the IP segment name, the two conditions you can use to set the requirement are **Yes** and **No**. If you select **Yes**, the device's IP address must match the IP segment. If you select **No**, the IP address must not match the IP segment.
- **Logged On To Primary Workstation:** Determines whether the user is logged on to his or her primary workstation. The two conditions you can use to set the requirement are **Yes** and **No**. If you select **Yes**, the user must be logged on to his or her primary workstation to meet the requirement. If you select **No**, and no user is logged on to the workstation, the requirement is not met. However, if a user other than the primary user is logged on to the workstation, the requirement is met.

- **Memory:** Determines the amount of memory on the device. The condition you use to set the requirement includes an operator and a memory amount. The possible operators are equals (**=**), does not equal (**<>**), is greater than (**>**), is greater than or equal to (**>=**), is less than (**<**), and is less than or equal to (**<=**). The memory amounts are designated in megabytes (**MB**) and gigabytes (**GB**). For example, if you set the condition to `>= 2 GB`, the device must have at least 2 gigabytes of memory to meet the requirement.
- **Novell Client Installed:** Determines if the device is using the Novell Client for its network connection. The two conditions you can use to set the requirement are **Yes** and **No**. If you select **Yes**, the device must be using the Novell Client to meet the requirement. If you select **No**, it must not be using the Novell Client.
- **Operating System - Windows:** Determines the architecture, service pack level, type, and version of Windows running on the device. The conditions you use to set the requirement includes a property, an operator, and a property value. The possible properties are **architecture**, **service pack**, **type**, and **version**. The possible operators are equals (**=**), does not equal (**<>**), is greater than (**>**), is greater than or equal to (**>=**), is less than (**<**), and is less than or equal to (**<=**). The property values vary depending on the property. For example, if you set the condition to `architecture = 32`, the device's Windows operating system must be 32-bit to meet the requirement.

  Be aware that operating system version numbers contain four components: Major, Minor, Revision, and Build. For example, the Windows 2000 SP4 release's number might be 5.0.2159.262144. Each component is treated independently. For this reason, the system requirements that you set might not provide your expected results.

  For example, if you specify **Operating System - Windows** in the first field, **Version** in the second field, **>** in the third field, and **5.1 -Windows XP Versions** in the last field, you are specifying only the first two components of the version number: Major (Windows) and Minor (5.0). As a result, for the requirement to evaluate to true, the OS must be at least 5.1 (Windows XP). Windows 2003 is version 5.2, so specifying > 5.1 also evaluates to True.

  However, because each component is independent, if you specify the version = 5.1, Windows XP SP2 evaluates to False because the actual version number might be 5.1.2159.262144. You can specify the version >= 5.1 to make the requirement evaluate as True because the actual revision component is greater than 0.

  When you select the OS version from the drop-down, the Major and Minor components are populated. The Revision and Build components must be typed manually.
- **Primary User Is Logged In:** Determines if the device's primary user is logged in. The two conditions you can use to set the requirement are **Yes** and **No**. If you select **Yes**, the primary user must be logged in to meet the requirement. If you select **No**, the user must not be logged in.
- **Processor Family:** Determines the device's processor type. The condition you use to set the requirement includes an operator and a processor family. The possible operators are equals (**=**) and does not equal (**<>**). The possible processor families are **Pentium**, **Pentium Pro**, **Pentium II**, **Pentium III**, **Pentium 4**, **Pentium M**, **WinChip**, **Duron**, **BrandID**, **Celeron**, and **Celeron M**. For example, if you set the condition to `<> Celeron`, the device's processor can be any processor family other than Celeron to meet the requirement.
- **Processor Speed:** Determines the device's processor speed. The condition you use to set the requirement includes an operator and a processor speed. The possible operators are equals (**=**), does not equal (**<>**), is greater than (**>**), is greater than or equal to (**>=**), is less than (**<**), and is less than or equal to (**<=**). The possible processor speeds are hertz (**Hz**), kilohertz (**KHz**), megahertz (**MHz**), and gigahertz (**GHz**). For example, if you set the condition to `>= 2 GHz`, the device's speed must be at least 2 gigahertz to meet the requirement.
- **Registry Key Exists:** Determines if a registry key exists. After specifying the key name, the two conditions you can use to set the requirement are **Yes** and **No**. If you select **Yes**, the specified key must exist to meet the requirement. If you select **No**, the key must not exist.

- **Registry Key Value:** Determines if a registry key value exists on the device. The condition you use to set the requirement includes the key name, the value name, an operator, a value type, and a value data. The key and value names must identify the key value you want to check. The possible operators are equals (**=**), does not equal (**<>**), is greater than (**>**), is greater than or equal to (**>=**), is less than (**<**), and is less than or equal to (**<=**). The possible value types are **INT_TYPE** and **STR_TYPE**. The possible value data is determined by the key, value name, and value type.
- **Registry Key and Value Exists:** Determines if a registry key and value exists. After specifying the key name and value, the two conditions you can use to set the requirement are **Yes** and **No**. If you select **Yes**, the specified key and value must exist to meet the requirement. If you select **No**, the key and value must not exist.
- **Service Exists:** Determines if a service exists. After specifying the service name, the two conditions you can use to set the requirement are **Yes** and **No**. If you select **Yes**, the service must exist to meet the requirement. If you select **No**, the service must not exist.
- **Specified Devices:** Determines if the device is one of the specified devices. After specifying the devices, the two conditions you can use to set the requirement are **Yes** and **No**. If you select **Yes**, the device must be included in the specified devices list to meet the requirement (an inclusion list). If you select **No**, the device must not be included in the list (an exclusion list).

Filter Logic

You can use one or more filters to determine whether the policy should be applied to a device. A device must match the entire filter list (as determined by the logical operators that are explained below) for the policy to be applied to the device.

There is no technical limit to the number of filters you can use, but there are practical limits, such as designing a filter structure that is easy to understand and organizing the filters so that you do not create conflicting filters.

Filters, Filter Sets, and Logical Operators

You can add filters individually or in sets. Logical operators, either **AND** or **OR**, are used to combine each filter and filter set. By default, filters are combined using **OR** (as determined by the **Combine Filters Using** field) and filter sets are combined using **AND**. You can change the default and use **AND** to combine filters, in which case filter sets are automatically combined using **OR**. In other words, the logical operator that is to combine individual filters (within in a set) must be the opposite of the operator that is used between filter sets.

You can easily view how these logical operators work. Click both the **Add Filter** and **Add Filter Set** options a few times each to create a few filter sets, then switch between **AND** and **OR** in the **Combine Filters Using** field and observe how the operators change.

As you construct filters and filter sets, you can think in terms of algebraic notation parentheticals, where filters are contained within parentheses, and sets are separated into a series of parenthetical groups. Logical operators (**AND** and **OR**) separate the filters within the parentheses, and the operators are used to separate the parentheticals.

For example, “(u AND v AND w) OR (x AND y AND z)” means “match either uvw or xyz.” In the filter list, this looks like:

```
u AND
v AND
w
OR
x AND
y AND
z
```

Nested Filters and Filter Sets

Filters and filter sets cannot be nested. You can only enter them in series, and the first filter or filter set to match the device is used. Therefore, the order in which they are listed does not matter. You are simply looking for a match to cause the policy to be applied to the device.

Publishing Policies

A policy can include multiple versions:

- **Published version:** The currently active version of the policy. This version is applied to any assigned devices.
- **Old versions:** Previously published versions that are not currently active.
- **Sandbox version:** A version that is currently being worked on and has not yet been published as the active version. The Sandbox version is not applied to assigned devices until it is published. A Sandbox version can be applied to devices that are designated as test devices. For more information, see “Testing a Disk Encryption Policy” on page 26.

The following sections explain how to republish an old version and publish a Sandbox version:

Republishing an Old Version

To republish an older version of a policy:

1. In ZENworks Control Center, click **Policies** to display the Policies page.
2. In the **Policies** list, click the policy for which you want to publish a previous version.
3. In the **Displayed Versions** list, select the version you want to publish.
4. Click **Create Sandbox**.
5. (Optional) Make changes to the Sandbox version.
6. Click **Publish**, then follow the wizard prompts.

Publishing a Sandbox Version

When you publish a Sandbox version of a policy, you have the option to publish it as a new version of the current policy or as a completely new policy.

1. In ZENworks Control Center, click **Policies** to display the Policies page.
2. In the **Policies** list, click the policy for which you want to publish a previous version.
3. In the **Displayed Versions** list, select **Sandbox**.
4. Click **Publish** to display the Publish Wizard.

5 If you want to publish the Sandbox version as a new version of the current policy, select **Publish as new version**, then click **Finish**.

or

If you want to publish the Sandbox version as a new policy, select **Publish as new policy**, fill in the new policy information, then click **Next** and follow the prompts to assign the policy to users and devices before clicking **Finish** to create the new policy.

Renaming, Copying, and Moving Policies

The following sections provide information to help you rename, copy, and move existing policies in your ZENworks system.

Renaming a Policy

If necessary, you can change a policy's name. Renaming a policy does not affect its assignments. However, it must be republished for the name change to be reflected on devices.

1 In ZENworks Control Center, click the **Policies** tab.

2 Select the check box next to the policy you want to rename, then click **Edit** > **Rename**.

3 In the **Name** field, type the new name.

4 Select the **Publish changed display name immediately** check box to make the change immediately available to devices.

This increments the published policy version and ensures that devices see the name change when the next device refresh occurs. If you do not select this check box, a Sandbox version of the policy is created; the change is not available on devices until after you publish the Sandbox version.

5 Click **OK**.

Copying a Policy

You can copy a policy to create a new policy. All of the policy's system requirements, details, and settings are copied to the new policy. The relationships (device assignments, user assignments, and policy groups) are not copied.

1 In ZENworks Control Center, click the **Policies** tab.

2 Select the check box next to the policy you want to copy, then click **Edit** > **Copy**.

3 In the **Name** field, type the name for the new policy.

4 Click **OK**.

Moving a Policy

You can move a policy from one folder in the **Policies** list to another. Moving a policy does not affect the policy's direct assignments to users and devices. It does, however, affect any assignments inherited from its current folder hierarchy.

1 In ZENworks Control Center, click the **Policies** tab.

2 Select the check box next to the policy you want to move, then click **Edit** > **Move**.

3 Browse for and select the destination folder, then click **OK**.

Enabling and Disabling Policies

A Disk Encryption policy can either be enabled or disabled. When a device receives an enabled policy, the Full Disk Encryption Agent applies the policy. When a device receives a disabled policy, the agent ignores the policy.

By default, a Disk Encryption policy is enabled during creation of the policy. The following sections explain how to disable a policy and enable it again.

Disabling a Policy

When you disable a policy that is currently assigned to devices, the policy is removed after the next device refresh. With the Disk Encryption policy, this causes the disks to be unencrypted and the ZENworks PBA to be removed.

When you assign a disabled policy devices, it is not applied until you enable it.

1. In ZENworks Control Center, click the **Policies** tab.
2. Select the check box next to the policy that you want to disable.
3. Click **Action > Disable**.

   In the **Policies** list, the **Enabled** status for the selected policy is changed to **No**.

Enabling a Policy

The Full Disk Encryption Agent does not apply disabled policies that are assigned to the device. To have the policy applied, you must enable it:

1. In ZENworks Control Center, click the **Policies** tab.
2. Select the check box next to the policy that you want to enable.
3. Click **Action > Enable**.

   In the **Policies** list, the **Enabled** status for the selected policy is changed to **Yes**. The policy is applied at the next device refresh.

Replicating Policies to Content Servers

If you have multiple ZENworks Servers or Satellites functioning as content servers, you can choose to replicate a policy to all content servers or selected content servers. If a policy is not replicated to a content server, the policy is not available to any devices that connect to that content server for their policies.

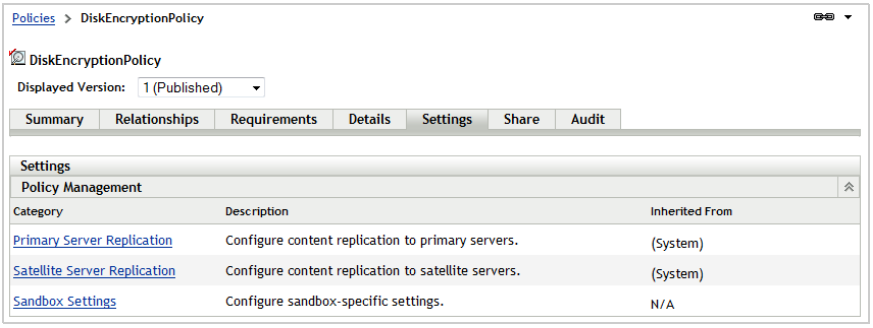
A policy inherits its content replication settings from its policy folder hierarchy or from the Management Zone. If you do not want it to use the inherited replication settings, you can override the settings on the policy.

The following instructions explain how to override the content replication settings for an individual policy. For information about configuring content replication settings on a policy folder or the Management Zone, see “Content”in the *ZENworks 2017 Primary Server and Satellite Reference*.

To define the replication settings for a policy:

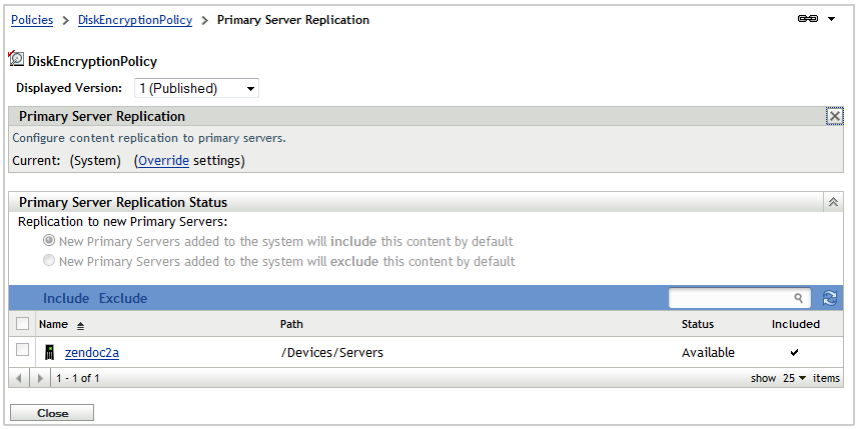
1. In ZENworks Control Center, click the **Policies** tab.
2. In the **Policies** list, click the policy to display its properties.

3 Click the **Settings** tab.



4 Configure the content replication settings for the Primary Servers:

4a In the Policy Management panel, click **Primary Server Replication**.



4b Click **Override Settings** to activate the Primary Server Replication Status panel.

4c Select whether or not the policy is replicated to new Primary Servers added to the system.

4d In the list of existing Primary Servers, select the servers that you want to receive the policy, then click **Include**.

A check mark appears in the **Included** column for the selected servers.

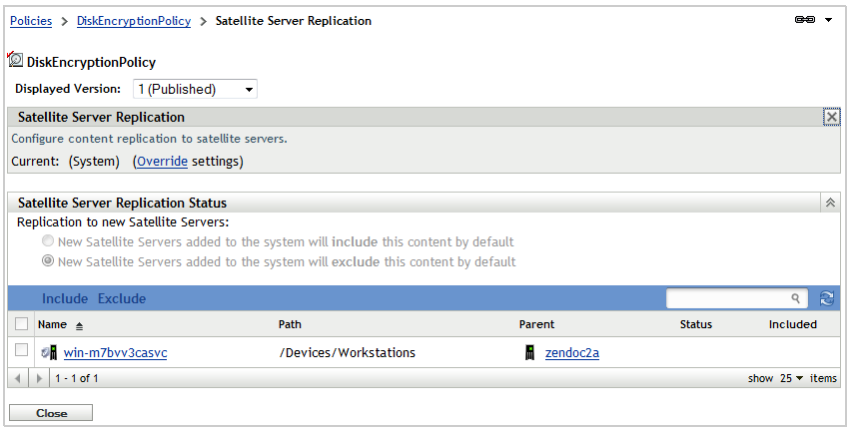
4e In the list of existing Primary Servers, select the servers that you don't want to receive the policy, then click **Exclude**.

The **Included** column is left blank to indicate that the servers are not included in the replication of this policy.

4f Click **OK** to save the changes.

5 Configure the content replication settings for Satellites:

5a In the Policy Management panel, click **Satellite Server Replication**.



5b Click **Override Settings** to activate the Satellite Server Replication Status panel.

5c Select whether or not the policy is replicated to new Satellite Servers added to the system.

5d In the list of existing Satellite Servers, select the servers that you want to receive the policy, then click **Include**.

A check mark appears in the **Included** column for the selected servers.

5e In the list of existing Satellite Servers, select the servers that you don't want to receive the policy, then click **Exclude**.

The **Included** column is left blank to indicate that the servers are not included in the replication of this policy.

5f Click **OK** to save the changes.

The policy's content replication settings are used only by the ZENworks system and do not affect the actual policy. Therefore, changing the replication settings does not require you to republish the policy to assigned devices and users.

Managing Policy Groups

If you have multiple policies that you always want assigned together, you can create a policy group and add the policies as group members. Then, rather than assigning the individual policies, you can assign the policy group.

A policy can be a member of more than one policy group. For example, assume that you have 10 policy groups to accommodate the unique needs of various groups within your organization. However, all organizations require the same Disk Encryption policy, so you add that policy to all of the policy groups.

The sections that follow provide instructions for managing policy groups.

You assign and remove policy groups for devices the same way that you assign and remove policies. For information, see "Assigning a Disk Encryption Policy" on page 26 and "Removing Policy Assignments From Devices" on page 46.

Creating Policy Groups

To create a policy group:

1. In ZENworks Control Center, click **Policies** in the navigation menu.
2. Click **New** > **Policy Group**.
3. Fill in the fields:

   Group Name: Provide a name for the policy group. The name must be different than the name of any other item (policy, group, folder, and so forth) that resides in the same folder. The name you provide displays in ZENworks Control Center.

   For more information, see "Naming Objects in ZENworks Control Center" in the *ZENworks 2017 Control Center Reference*.

   Folder: Type the name or browse to and select the ZENworks Control Center folder where you want the policy to reside. The default is `/policies`, but you can create additional folders to organize your policies.

   Description: Provide a short description of the policy group's contents. This description displays in ZENworks Control Center.
4. Click **Next** to display the Add Group Members page, then add the policies you want to be members of the group:
   - **4a** Click **Add** to display the Select Members dialog box.

     Because you are adding policies to the group, the Select Members dialog box opens with the `Policies` folder displayed.
   - **4b** Click next to a folder to navigate through the folders until you find the policy you want to select.

     If you know the name of the policy you are looking for, you can use the **Item name** box to search for the item. You can add only policies to the group. You cannot add other policy groups to the group.
   - **4c** Click the underlined link in the **Name** column to select the policy and display its name in the **Selected** list box.
   - **4d** (Optional) Repeat Step 4b and Step 4c to select additional policies.
   - **4e** Click **OK** to add the selected policies.
5. Click **Next** to display the Summary page, review the information and, if necessary, use the **Back** button to make changes to the information.
6. (Optional) Select the **Define Additional Properties** option to display the group's properties page after the group is created. You can then configure additional policy group properties, such as assigning the policy group to devices and users.
7. Click **Finish** to create the group.

Adding Policies to Existing Groups

To add a policy to an existing policy group:

1. In ZENworks Control Center, click the **Policies** in the navigation menu.
2. Click the policy group to display its properties.
3. In the Members panel, click **Add** to display the Select Members dialog box.

   Because you are adding policies to the group, the Select Members dialog box opens with the `Policies` folder displayed.

4 Click next to a folder to navigate through the folders until you find the policy you want to select.

   If you know the name of the policy you are looking for, you can use the **Item name** box to search for the item. You can add only policies to the group. You cannot add other policy groups to the group.

5 Click the underlined link in the **Name** column to select the policy and display its name in the **Selected** list box.

6 (Optional) Repeat Step 4 and Step 5 to select additional policies.

7 Click **OK** to add the selected policies to the **Members** list.

8 Click **OK** to save the policy group.

Renaming Policy Groups

You can rename a policy group. Renaming a group does not affect the group's assignments.

1 In ZENworks Control Center, click the **Policies** tab.

2 In the **Policies** list, select the check box next to the policy group you want to rename.

3 Click **Edit**, then click **Rename**.

4 Type the new name in the **Name** field, then click **OK**.

Moving Policy Groups

You can move a policy group from one folder in the **Policies** list to another. Moving a group does not affect the group's assignments.

1 In ZENworks Control Center, click the **Policies** tab.

2 In the **Policies** list, select the check box next to the policy group you want to move.

3 Click **Edit**, then click **Move**.

4 Select the destination folder for the policy group, then click **OK**.

Deleting Policy Groups

Deleting a policy group does not delete its policies. It does remove all assignments of the policy group.

1 In ZENworks Control Center, click the **Policies** tab.

2 In the **Policies** list, select the check box next to the policy group.

3 Click **Delete**, then click **OK** to confirm the deletion.

4 Policy Removal

The following sections provide information for removing policy assignments and deleting policies:

Removal Best Practices

The following sections provide a best practice approach to removing Disk Encryption policies that have been deployed to devices.

WARNING: When removing a full disk encryption policy, ensure that the decryption process is not interrupted prematurely with a power change on the disk drive(s); otherwise, all data on the disk can be lost due to disk corruption. You can check the decryption status on the device by accessing **Full Disk Encryption** > **About** in the ZENworks Agent.

Disk corruption due to power change has only been noted on secondary drives, but it may also be applicable to primary drives. For this reason, the following precautions are strongly recommended before removing a full disk encryption policy from a device:

- Pre-configure devices receiving the policy so that power options are set to never automatically shut off, hibernate, or sleep.
- Inform all device users of the need to keep the device running during the decryption process, to include avoiding *Sleep* and *Hibernation* options.

  This precaution is for user actions that are not a part of the reboot process that is required for decryption and policy removal.

Remove policy assignments before deleting a policy

Deleting a policy automatically removes the policy assignments. However, we recommend that you remove policy assignments before you delete a policy to see if the policy removal has any negative effects on the device. If so, the policy is still available to reassign.

Remove policy assignments before uninstalling the ZENworks Agent or ZENworks Full Disk Encryption Agent

Before uninstalling the ZENworks Agent or uninstalling or disabling the ZENworks Full Disk Encryption Agent from a device, remove the Full Disk Encryption policy assignment and refresh the device so that the device is decrypted and the ZENworks PBA (if installed) is removed.

Ensure that the device has a current ERI file

An Emergency Recovery Information (ERI) file enables you to recover the encrypted disk information if problems occur during the removal of the Disk Encryption policy. Verify that the device from which you are removing the policy has a current ERI file.

1. In ZENworks Control Center, click **Devices** > **Workstations**.
2. Click the device to display its details.

3 Click the **Emergency Recovery Information** tab.

The device's ERI files are displayed in the list. If there are no ERI files, or you are not sure if the ERI file is the most current, go back to the **Workstations** list, select the check box next to the device, then click **Quick Tasks** > **FDE - Force Device to Send ERI File to Server**. Wait for the task to complete and then verify that the ERI file is displayed in the device's ERI list.

Tell the device user that the policy is being removed

When you remove a Disk Encryption policy from a device, the encrypted disks must be decrypted, the encryption drivers removed, and the ZENworks PBA removed. This takes some time and requires multiple reboots of the device. We recommend that you make the user aware of what to expect.

Removing Policy Assignments From Devices

When a policy is assigned to an device or device folder, the assignment is reflected as a *relationship* in the policy's properties and in the device or device folder properties. You can edit the relationships for either the policy or the object to remove the assignment.

The following sections provide instructions for two common assignment removal scenarios.

Removing Policy Assignments From a Single Object

The following instructions explain how to remove policy assignments from device or device folder.

1 In ZENworks Control Center, click the device or device folder from which you want to remove policy assignments.

For device folders, you need to click **Details** next to the folder name rather than click the name.

2 Click **Relationships**.

3 In the Assigned Policies panel, click the **Direct** tab to ensure that it is active.

The **Direct** tab displays all policies that are assigned directly to the object. Direct assignments are the only assignments you can remove for the object.

4 Select the check box next to the assignments you want to remove, then click **Remove**.

Removing a Policy Assignment From Multiple Objects

The following instructions explain how to remove a single policy assignment from multiple devices or device folders at the same time.

1 In ZENworks Control Center, click the **Policies** tab.

2 In the **Policies** list, click the policy for which you want to remove assignments.

3 Click **Relationships.**

4 In the Device Assignments panel, select the check boxes next to the devices and device folders that you no longer want the policy assigned to, then click **Remove**.

Deleting Policies

When you delete a Disk Encryption policy, all assignments of the policy to devices are removed. At the next refresh, devices will begin to decrypt any encrypted volumes and remove the ZENworks PBA.

1. In ZENworks Control Center, click the **Policies** tab.
2. Select the check box next to the policy (or policies) that you want to delete.
3. Click **Delete**.

Deleting Versions of a Policy

When you make changes to a policy and publish the changes, the policy version is incremented (for example, from version 1 to version 2). The old version is retained in case you want to use it as the basis for a new version of the policy.

If you don't want to keep older versions of a policy, you can delete them. Doing so does not delete the currently published policy and does not affect the policy's assignments.

To delete a version of a policy:

1. In ZENworks Control Center, click the **Policies** tab.
2. Double-click the policy to display its property pages.
3. In the **Displayed Version** field, select the version you want to delete.
4. Click **Delete Selected Version**.

   The selected version is deleted and the published version is displayed.

A Appendix - Disk Encryption Policy Settings

The following sections provide information about the Details settings for the Disk Encryption policy.

For help creating a Disk Encryption policy, see “Creating a Disk Encryption Policy” on page 14.

Disk Encryption

ZENworks Full Disk Encryption supports encryption of standard hard disks of type IDE, SATA, and PATA. Encryption of SCSI and RAID hard disks is not supported.

The Disk Encryption page shows the fixed disk volumes that are encrypted and the algorithm to use for the encryption. In addition, you can choose whether or not to allow users to create Emergency Recovery Information (ERI) files that can be used to regain access to encrypted volumes if a problem occurs with the device.

Local Fixed Volumes

Displays the fixed disk volumes that are encrypted when the policy is applied to a device. You cannot change these settings for an existing policy.

Any of a device’s local fixed disk volumes can be encrypted. Removable disks, such as thumb drives, cannot be encrypted. Neither can non-local disks, such as network drives.

- **Encrypt all local fixed volumes:** Select this option to encrypt all volumes.
- **Encrypt specific local fixed volumes:** Select this option to limit encryption to specific volumes. To specify a volume, click **Add**, then select the drive letter assigned to the volume. If a volume that you specify does not exist on a device to which the policy is assigned, or the specified volume is not a local fixed volume, no encryption of the specified volume takes place.

After the policy is applied, encryption of the target volumes is performed sequentially, one volume at a time. A maximum of 10 volumes are encrypted, even if the device has more than 10.

Encryption Settings

Displays the encryption settings to be applied to the device. The only setting you can change is the **Block 1394 (FireWire) port** setting:

Encryption is the process of converting plain-text data into cipher text that can then be decrypted back into its original plain text. An encryption algorithm, also known as a cipher, is a set of steps that determines how an encryption key is applied to the plain-text data to encrypt and decrypt the text.

The following settings determine the algorithm that is used to encrypt the selected fixed volumes, and the length of the encryption key that is used in the encryption process.

- **Algorithm:** Select one of the following encryption algorithms:
  - **AES:** The AES (Advanced Encryption Standard) algorithm is a symmetric-key encryption standard adopted by the U.S. government. AES has a 128-bit block size with key lengths of 128, 192, and 256 bits.

    AES provides the highest security coupled with fast encryption speed. This algorithm is the optimal choice for most users.
  - **Blowfish:** The Blowfish algorithm is a symmetric-key block cipher. It has a 64-bit block size with key lengths of 32 to 448 bits. It is a strong, fast, and compact algorithm.
  - **DES:** The DES (Data Encryption Standard) algorithm is a symmetric-key encryption standard that uses a 56-bit key.

    Because of its 56-bit key size, DES is not as secure as AES or Blowfish. DES keys have been broken in less than 24 hours.
  - **DESX:** The DESX algorithm is a variant of the DES algorithm. It uses a 128-bit key.
- **Key Length:** Select a key length. Key lengths vary depending on the encryption algorithm you select. We recommend that you choose the maximum key length for the algorithm. Doing so provides the highest security with no significant performance loss.
- **Encrypt only the used sectors of the drive:** During initial encryption of a fixed disk volume, all of the sectors are encrypted unless you select this option. If you select this option, only the sectors that contain data are encrypted. Additional sectors are encrypted as they are used.

  Encrypting all sectors (used and unused) greatly increases the initial encryption time. You should only encrypt unused sectors if you are concerned about unauthorized users possibly recovering previously deleted files from the unused (and unencrypted) sectors.
- **Block 1394 (FireWire) port:** The 1394 interface provides direct memory access, or DMA. Direct access to system memory can compromise security by providing read and write access to stored sensitive data, including encryption and authentication data used by ZENworks Full Disk Encryption. Select this option to prevent direct access to memory through the 1394 port.
- **Enable software encryption of Opal compliant self-encrypting drives:** When enabled, this option does the following to OPAL 2.0 compliant self-encrypting drives:
  - Prevents the ZENworks Pre-Boot Authentication (PBA) mechanism from initiating the drive's locking feature. This allows the ZENworks PBA to work with *ALL* OPAL 2.0 compliant self-encrypting drives, not just the drives that are known to be drive-locking compatible with ZENworks Full Disk Encryption.
  - Applies software encryption to the drive, adding a second layer of encryption to the drive's already hardware-encrypted contents.

Emergency Recovery Information (ERI) Settings

An Emergency Recovery Information (ERI) file is required to regain access to encrypted volumes if a problem occurs with the device. When the policy is applied to a device, or the policy changes, an ERI file is automatically created and uploaded to the ZENworks Server. You can also enable users to manually create ERI files and store them locally.

- **Allow user to create ERI files:** Select this option to enable users to create ERI files. This is done through the ZENworks Full Disk Encryption Agent's About box.

- **Require user to provide a strong password when creating an ERI file:** The ERI file is password-protected to ensure that no unauthorized users can use it to gain access to the encrypted device. The user enters the password when creating the file. Select this option to force the user to provide a password for the file that meets the following requirements:
  - Seven or more characters
  - At least one of each of the four types of characters:
    - uppercase letters from A to Z
    - lowercase letters from a to z
    - numbers from 0 to 9
    - at least one special character ~ ! @ # $ % ^ & * ( ) + { } [ ] : ; < > ? , . / - = | \ ”

  For example: qZG@3b!
- **Use common password for system-generated ERI files:** When this option is selected, all system-generated ERI files will use the password that is specified in this setting.

Disk Encryption Reboot Control

This page lets you specify an Administrator password for the ZENworks Full Disk Encryption Agent and determine when a device is rebooted to initiate the encryption of the device’s volumes.

Admin Password

The Administrator password enables access to the Administrator options in the Full Disk Encryption Agent. These options help you see the current status of the agent and view the assigned Disk Encryption policy, as well as troubleshoot problems with the agent or policy.

To set the password, click **Set**, specify the password, then click **OK**.

If you ever need to allow a user to access the Administrator options, we recommend that you use the Password Key Generator utility to generate a password key. The key, which is based on the FDE Admin password, functions the same as the FDE Admin password but can be tied to a single device or user and can have a usage or time limit.

The Password Key Generator utility is accessible under the **Configuration Tasks** list in the left navigation pane.

Reboot Options

When the Disk Encryption policy is applied to a device, the device’s disks cannot be encrypted until the device reboots and loads the Full Disk Encryption Agent’s encryption drivers.

- **Reboot Behavior:** Select one of the following:
  - **Force device to reboot immediately:** Reboots the device immediately after the Disk Encryption policy is applied.
  - **Do not reboot device:** Does not force a reboot after the Disk Encryption policy is applied. The user must initiate a reboot before disk encryption can occur.
  - **Force device to reboot within *XX* minutes:** Reboots the device within the specified number of minutes after the Disk Encryption policy is applied. Providing a reboot delay can give the user time to save work prior to the reboot. The default delay is 5 minutes.

- **Display predefined message to user before rebooting:** If you selected the **Do not reboot device** option or the **Force device to reboot within XX minutes** option, you can display a message to the user. The **Force device to reboot immediately** option does not support a message.

  Select this option to display the following message:

  ```
  ZFDE Policy Enforcement

  Your ZENworks Administrator has assigned a Disk Encryption policy to your computer. To enforce the policy, your computer must be rebooted.
  ```

- **Override predefined message with custom message:** This option is available only after you select the **Display predefined message to user before rebooting** option. It lets you override the predefined message with your own custom message. Select the option, then specify a title for the message window and the text to include in the message body.

CheckDisk Options

We strongly recommend that you run Windows CheckDisk with Repair during the reboot. The disk check and repair is performed on the system volume (C: drive), ensuring that system and partition records are error-free prior to encrypting the target volumes.

This option is selected by default. If you are sure that the target volumes are in perfect condition (for example, the disks are new), you can select the **Do not run Windows check disk** option.

NOTE: This setting does not apply to Windows XP. On Windows XP, CheckDisk is run if it is needed regardless of the setting.

Pre-Boot Authentication

ZENworks Pre-Boot Authentication (PBA) provides increased authentication security for devices.

The ZENworks PBA is a Linux-based component. When the Disk Encryption policy is applied to a device with a standard hard disk, a 100 MB partition containing a Linux kernel and the ZENworks PBA is created on the hard disk. When the policy is applied to a device with a self-encrypting hard disk, the Linux kernal and ZENworks PBA are installed to the disk's datastore memory.

During normal operation, the device boots to the Linux partition and loads the ZENworks PBA. As soon as the user provides the appropriate credentials (user ID/password or smart card), the PBA terminates and the Windows operating system boots, providing access to the encrypted data on the previously hidden and inaccessible Windows drives.

The Linux partition is hardened to increase security, and the ZENworks PBA is protected from alteration through the use of MD5 checksums and uses strong encryption for authentication keys.

ZENworks Pre-Boot Authentication

During creation of the policy, ZENworks Pre-Boot Authentication was either enabled or disabled. You cannot change this setting for the policy.

If ZENworks Pre-Boot Authentication is disabled, none of the remaining settings on the page apply and are therefore disabled.

Authentication Methods

These settings let you configure the methods that can be used for authenticating to a device's encrypted disks. If you have enabled ZENworks Pre-Boot Authentication, you must select at least one of the methods.

- **Enable user ID/password authentication:** Select this option to enable users to authenticate via a user ID and password. If you select this option, you must configure the settings in the **User ID/Password Authentication Settings** section.
- **Enable smart card authentication:** Select this option to enable users to authenticate via a smart card. If you select this option, you must configure the settings in the **Smart Card Authentication Settings** section.
- **Default Authentication Method:** If you enable both the user ID/password and smart card authentication methods, you must select the default method. Both methods are available to a user during pre-boot authentication, but the default method is presented if the user does not select a method within the allotted time.
- **Activate Single Sign-On for ZENworks PBA and Windows Login:** Select this option to activate single sign-on for the PBA and Windows login. The user logs in to the PBA and the PBA handles the login to the Windows operating system. Single sign-on applies to both authentication methods (user ID/password or smart card).

User ID/Password Authentication Settings

If you selected **Enable user ID/password authentication** as one of the supported authentication methods, configure the following settings:

- **During PBA login, show user name of last successful logged-in user:** Select this option to prepopulate the **User ID** field of the PBA login screen with the username of the last user who logged in to the PBA. This is convenient for the device's primary user, but weakens security by providing unauthorized users with a valid user ID.
- **Create PBA account for first user who logs in to Windows after the policy is applied (User Capturing):** Select this option to automatically capture the credentials of the first user to authenticate after the policy is applied. During the first reboot after the policy is applied, the Windows login is displayed and the PBA captures the credentials provided for the Windows login. During subsequent reboots, the PBA login is displayed and accepts the captured credentials.

  Captured credentials exist only on the device where they are captured. The credentials are not stored with this policy.

  If a device has multiple users, the PBA captures only the first user to log in after the policy is applied. You can capture additional users by using the **FDE - Enable Additive User Capturing** quick task for the device. When the quick task is applied to a device, it activates the user capturing mode for the next reboot. To use the quick task, select the device in **Devices** > **Workstations**, then click **Quick Tasks** > **FDE - Enable Additive User Capturing**.

  NOTE: The PBA captures the credentials of the first user to authenticate after reboot, whether the credentials are user ID/password or smart card. The PBA login screen allows the user to switch between user ID/password login and smart card login. If a device supports both types of login, you should make sure the device's user logs in with the user ID/password and not the smart card. Otherwise, the smart card credential is captured and the user cannot log in via the user ID/password. This becomes a problem if you have not enabled smart cards as an authentication method (see Authentication Methods) because the user cannot log in.

- **Allow access for the following users:** User capturing is the recommended way to create a PBA account for a device's users. However, you can enable this option and use the **PBA Users** list to define PBA user accounts.

  All accounts that you add to the **PBA Users** list are created on all devices to which the policy is applied. Because of this, the **PBA Users** list is a good way to give Administrators access to each of the devices. For example, if you have a common Windows Administrator account that you use across devices, you can add the Windows Administrator as a PBA user. You can then log in to both the PBA and Windows on a device by using the Administrator account and password.

  To add a PBA user account, click **Add**, then fill in the following fields:

  - **Replace password if user already exists in PBA:** When the policy is applied, if the user you are adding matches an existing PBA user (for example, a user added by a previously applied Disk Encryption policy), the existing user account is retained, including the existing password. Select this option to replace the existing password with the one you specify in this dialog box.
  - **User Name:** Specify a user name for the PBA user. If single sign-on is active, this user name must be the same as the Windows user name. If single sign-on is not active, the user name does not need to match the Windows user name.
  - **Domain:** Specify a domain name for the PBA user. If single sign-on is active, this must be the Windows domain name or workgroup name. If single sign-on is not active, this field is optional. You can leave it blank or use it as another component to further distinguish the PBA user name.
  - **Password:** Specify a password for the PBA user. If single sign-on is active, this must be the Windows password. If single sign-on is not active, you can specify any password.
- **Remove existing users from PBA if not in this list:** Select this option to remove any user accounts from the PBA that are not listed in the **PBA Users** list. Because captured users do not display in the list, they are also removed.

Smart Card Authentication Settings

If you selected **Use smart card authentication** as one of the supported authentication methods, configure the smart card settings.

- **Smart Card Reader:** Select the card reader used by the devices to which this policy will be assigned.
- **PKCS#11 Provider:** Select the PKCS #11 provider used by the devices to which this policy will be assigned.
- **Create PBA account for first smart card user who logs in to the ZENworks PBA after the policy is applied (User Capturing):** Select this option to automatically capture the credentials of the first user to authenticate after the policy is applied. During the first reboot after the policy is applied, the PBA login screen is displayed and the user is prompted for the smart card. The PBA captures the smart card credentials (certificate and PIN). During subsequent reboots, the PBA login accepts the captured smart card credentials.

  If a device has multiple users, the PBA captures only the first user to log in after the policy is applied. You can capture additional users by using the **FDE - Enable Additive User Capturing** quick task for the device. When the quick task is applied to a device, it activates the user capturing mode and creates a PBA account for the next user who logs in. To use the quick task, select the device in **Devices** > **Workstations**, then click **Quick Tasks** > **FDE - Enable Additive User Capturing**.

NOTE: The PBA captures the credentials of the first user to authenticate after reboot, whether the credentials are smart card or user ID/password. The PBA login screen allows the user to switch from smart card login to user ID/password login, but you should make sure the device's user logs in with the smart card and not the user ID/password. Otherwise, the user ID/password credential is captured and the user cannot log in via the smart card. This becomes a problem if you have not enabled user ID/password as an authentication method (see Authentication Methods) because the user cannot log in.

- **Allow certificate content to be used for authentication:** User capturing is the recommended way to create a PBA account for smart card users because it accurately captures the smart card certificate information. If you don't enable user capturing, you must manually define certificates that can be used for authentication. If you do enable user capturing, you can still manually define additional certificates that allow access.

  To define a certificate, click **Add**, fill in the following fields, then click **OK** to add the certificate to the list:

  - **Certificate Name:** Specify a name to identify the certificate in this policy. This is simply a display name and does not need to match the certificate file name or any other certificate property.
  - **Certificate Content:** Open the certificate in a text editor, then cut and paste the contents of the certificate into this box. You must use an X.509 certificate (`*.cer`; base64-encoded).
- **Remove existing certificates from PBA if not in this list:** Select this option to remove any certificates from the PBA that are not listed in the **Certificates** list. Because captured certificates do not display in the list, they are also removed.
- **Allow certificate key usages to be used for authentication:** In addition to enabling user capturing or defining the certificates that can be used for authentication, you need to further identify the certificates via key usages (this setting) or labels (the **Allow certificate labels to be used for authentication** setting). This adds a second layer of security to the certificate authentication.
  - **Key Usages:** Key usages define the purposes for which a certificate's public key can be used, such as Data Encipherment or Digital Signature. You can view a certificate's key usages by using Microsoft Certificate Manager (available as a snap-in to Microsoft Management Console).

    To add a certificate's key usages to the list, click **Add**, select the desired usages (Shift-click or Ctrl-click to select multiple usages), click the arrow to move the selected items to the **Selected List** box, then click **OK**.

    If you add more than one key usage, the PBA evaluates the key usages against the certificates in the order the usages are listed, from top to bottom. You can use **Move Up** and **Move Down** to change the order of the key usages in the list.
  - **Match policy:** The match policy determines how many of the defined key usages must be contained in the smart card's certificate in order for the match to be made and authentication to take place. Select one of the following options:
    - **Any:** The certificate must contain at least one of the listed key usages.
    - **All:** The certificate must contain all of the listed key usages.
    - **None:** The certificate cannot contain any of the listed key usages. This option lets you use the Key Usages list as an exclusion list (blacklist) rather than an inclusion list (whitelist).
- **Allow certificate labels to be used for authentication:** In addition to enabling user capturing or defining the certificates that can be used for authentication, you need to further identify the certificates via labels (this setting) or key usages (the **Allow certificate key usages to be used for authentication** setting). This adds a second layer of security to the certificate authentication.

A certificate label is a property defined within the certificate. You need to use the PKCS #11 middleware provider software to view the certificate label.

To add a certificate label to the list, click **Add**, specify the label (case-sensitive), then click **OK**.

If you add more than one label, the PBA attempts to match the first label in the list to a certificate on the authenticating smart card. If no match occurs, the second label is tested, then the third label, and so on until a match occurs or authentication fails. You can determine the order of the labels in the list by selecting a label and clicking **Move Up** or **Move Down** to reposition it in the list.

Pre-Boot Authentication Reboot Control

This page lets you determine when the device is rebooted after initialization of the ZENworks PBA; the first pre-boot authentication does not occur until the device reboots. It also lets you specify the number of times a user can enter the incorrect PBA login information before being locked out.

Reboot Options

Both the ZENworks PBA and the Full Disk Encryption Agent's encryption drivers are initialized the first time the device reboots after the Disk Encryption policy is applied. However, the ZENworks PBA requires an additional reboot to facilitate user capturing (if enabled) or authentication of a predefined user. In addition, encryption of the target volumes does not begin until this reboot occurs.

The following options let you specify how you want this second reboot to occur:

- **Reboot Behavior:** Select one of the following:
  - **Force device to reboot immediately:** Reboots the device immediately after the PBA is initialized.
  - **Do not reboot device:** Does not force a reboot after the PBA is initialized. The user must initiate a reboot before user capturing or predefined user authentication can occur.
  - **Force device to reboot within *XX* minutes:** Reboots the device within the specified number of minutes after the PBA initializes. The default delay is 5 minutes.
- **Display predefined message to user before rebooting:** If you selected the **Do not reboot device** option or the **Force device to reboot within XX minutes** option, you can display a message to the user. The **Force device to reboot immediately** option does not support a message.

  Select this option to display the following message:

  ```
  ZFDE Policy Enforcement

  Your ZENworks Administrator has assigned a Disk Encryption policy to your computer. To enforce the policy, your computer must be rebooted.
  ```
- **Override predefined message with custom message:** This option is available only after you select the **Display predefined message to user before rebooting** option. It lets you override the predefined message with your own custom message. Select the option, then specify a title for the message window and the text to include in the message body.

Lockout Settings

The Lockout settings apply to the ZENworks PBA login.

- **Enable lockout for failed logins:** Select this option to enable the PBA to lock out users based on failed login attempts, then configure the following settings:
  - **Maximum Number of Failed Logins:** Specify the maximum number of failed logins to allow before the lockout is enforced (the default is 10). When the maximum number of failed logins is reached, the device is locked. A PBA override must be performed to access the device and reset the failed login count. See "PBA Override" in the *ZENworks Full Disk Encryption PBA Reference* for more information.
  - **Failed Logins after which Login is Delayed:** Specify the number of failed logins to allow before delaying subsequent logins (the default is 3). When the specified number of failed logins is reached, each failed login attempt results in a 2 minute delay before the next attempt can be made. Make sure to specify a number that is less than the one entered in the **Maximum Number of Failed Logins** field.

  For example, using the defaults of 10 and 3 for the two settings, 10 failed logins are allowed before lockout, but after the third failed login all subsequent login attempts are delayed by 2 minutes.
- **PBA Keyboard Layout:** Select the keyboard layout used for authentication.

DMI Settings

After pre-boot authentication occurs, the BIOS settings must be correctly set for Windows. With older or unusual hardware configurations, the standard ZENworks PBA boot method and Linux kernel configuration used to provide the BIOS settings might not work, resulting in hardware that does not function correctly or is not recognized by Windows.

This page provides support for older or unusual hardware configurations. These configurations might include the following:

- Hardware that does not function correctly or is no longer recognized under Windows after successful pre-boot authentication.This failure occurs because not all of the BIOS settings can be correctly handled and set for Windows.
- New hardware that is not yet natively supported.
- Poorly programmed BIOS implementations.

This hardware compatibility support applies only to software encrypted disks; self-encrypting hard disks are not supported. In addition, some devices might not support the boot methods or Linux kernel configurations used to provide hardware compatibility.

About Hardware Compatibility

Hardware compatibility is enabled through the use of two alternative boot methods and an alternative Linux kernel that supplies ACPI (Advanced Configuration and Power Interface) support. These alternative boot methods and kernel are defined through the use of a DMI (Direct Media Interface) file. The predefined file includes the following default setting:

```
[default]
KICKSTART=FAST
```

This default setting uses the standard boot method (KICKSTART=FAST) and no alternative Linux kernel. It is applied to all hardware configurations unless a configuration is explicitly defined in the file.

The predefined file also includes explicit settings for hardware configurations with known issues. For example:

```
[FUJITSU SIEMENS,LIFEBOOK C1110]
DMI_SYS_VENDOR=FUJITSU SIEMENS
DMI_PRODUCT_NAME=LIFEBOOK C1110
KICKSTART=BIOS
```

This setting applies to the Fujitsu Siemens Lifebook C1110 laptop. It applies a different boot method (KICKSTART=BIOS) that involves rebooting the computer a second time so that the BIOS hardware settings can be passed to Windows. It does not use an alternative Linux kernel configuration.

The following example uses both an alternative boot method and Linux kernel configuration:

```
[LENOVO, 417152U]
DMI_SYS_VENDOR=LENOVO
DMI_PRODUCT_NAME=417152U
KICKSTART=KEXEC
KERNEL=/boot/bzImage-acpi
```

This setting applies to the Lenovo ThinkPad T420s laptop. It applies a different boot method (KICKSTART=KEXEC) that is similar to KICSTART=BIOS but does not require a second reboot. It uses an alternative kernel configuration that enables ACPI support.

Discovering Hardware Information

Before you can add a hardware configuration to the DMI file, you must know the hardware configuration. ZENworks provides a utility, `DMICONFIG`, to discover this information.

1. Go to the device whose hardware configuration you want to discover.
2. Open a command shell (run as Administrator) and run `c:\windows\nac\sbs\dmiconfig dump`.
3. Write down the configuration lines that were dumped to the screen.

Editing the DMI File

If you are adding a hardware configuration, make sure you have the configuration information (see Discovering Hardware Information).

On the Hardware Compatibility page of the Create New Wizard Policy:

1. Click **Edit**.
2. Add the hardware information.
3. Add the KICKSTART line with the method you want to use:
   - **KICKSTART=FAST:** This is the standard method used by the ZENworks PBA.
   - **KICKSTART=BIOS:** This method is for systems that have unusual hardware configurations that are not supported by the standard FAST method. This method reboots the computer a second time so that the BIOS hardware settings can be passed to Windows.
   - **KICKSTART=KEXEC:** This method is similar to KICKSTART=BIOS but does not require a second reboot.
4. If you want to boot the computer using the alternative Linux kernel (with ACPI support), add the following line:

```
KERNEL=/boot/bzImage-acpi
```

5 Include the following kernel parameters if needed:

```
KERNEL_PARAM=irqpol
```

```
KERNEL_PARAM=pci=snb-enable-ahci-to-legacy
```

If both parameters are used, specify them on the same line:

```
KERNEL_PARAM=irqpol pci=snb-enable-ahci-to-legacy
```

- **irqpoll:** Alters the way that the kernel handles interrupts. This is useful if the PBA kernel log shows messages stating that an interrupt occurred.
- **pci=snb-enable-ahci-to-legacy:** ZENworks AHCI mode kernel option that switches the chipset to ATA mode prior to performing the soft reset and booting to Windows. This parameter fixes many instances where the chipset is in AHCI mode and the soft reset fails to boot Windows.

6 Click **OK** to save your changes.