Lenovo

Centrally Managing Access to Self-Encrypting Drives in Lenovo System x Servers

Understand self-encrypting drive technology and centralized key management systems

Centralized key management using IBM Security Key Lifecycle Manager

Manage and troubleshoot your SED-based server

Comprehensive guide for implementing a managed solution for SED drives

Ryan Bradley

Angelo Parisi

International Technical Support Organization

Centrally Managing Access to Self-Encrypting Drives in Lenovo System x Servers

March 2015

SG24-8247-00

Note: Before using this information and the product it supports, read the information in “Notices” on page vii.

Last update on March 2015

This edition applies to Version 2.5 of IBM Security Key Lifecycle Manager. For the latest levels of supported firmware for hardware components and drivers refer to Chapter 2, “Supported systems and sample configuration” on page 17.

© Copyright International Business Machines Corporation 2015. All rights reserved.
Note to U.S. Government Users Restricted Rights -- Use, duplication or disclosure restricted by GSA ADP Schedule Contract with IBM Corp.

Contents

© Copyright IBM Corp. 2015. All rights reserved.

Notices

Lenovo may not offer the products, services, or features discussed in this document in all countries. Consult your local Lenovo representative for information on the products and services currently available in your area. Any reference to a Lenovo product, program, or service is not intended to state or imply that only that Lenovo product, program, or service may be used. Any functionally equivalent product, program, or service that does not infringe any Lenovo intellectual property right may be used instead. However, it is the user's responsibility to evaluate and verify the operation of any other product, program, or service.

Lenovo may have patents or pending patent applications covering subject matter described in this document. The furnishing of this document does not give you any license to these patents. You can send license inquiries, in writing, to:

Lenovo (United States), Inc.
1009 Think Place - Building One
Morrisville, NC 27560
U.S.A.
Attention: Lenovo Director of Licensing

LENOVO PROVIDES THIS PUBLICATION “AS IS” WITHOUT WARRANTY OF ANY KIND, EITHER EXPRESS OR IMPLIED, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF NON-INFRINGEMENT, MERCHANTABILITY OR FITNESS FOR A PARTICULAR PURPOSE. Some jurisdictions do not allow disclaimer of express or implied warranties in certain transactions, therefore, this statement may not apply to you.

This information could include technical inaccuracies or typographical errors. Changes are periodically made to the information herein; these changes will be incorporated in new editions of the publication. Lenovo may make improvements and/or changes in the product(s) and/or the program(s) described in this publication at any time without notice.

The products described in this document are not intended for use in implantation or other life support applications where malfunction may result in injury or death to persons. The information contained in this document does not affect or change Lenovo product specifications or warranties. Nothing in this document shall operate as an express or implied license or indemnity under the intellectual property rights of Lenovo or third parties. All information contained in this document was obtained in specific environments and is presented as an illustration. The result obtained in other operating environments may vary.

Lenovo may use or distribute any of the information you supply in any way it believes appropriate without incurring any obligation to you.

Any references in this publication to non-Lenovo Web sites are provided for convenience only and do not in any manner serve as an endorsement of those Web sites. The materials at those Web sites are not part of the materials for this Lenovo product, and use of those Web sites is at your own risk.

Any performance data contained herein was determined in a controlled environment. Therefore, the result obtained in other operating environments may vary significantly. Some measurements may have been made on development-level systems and there is no guarantee that these measurements will be the same on generally available systems. Furthermore, some measurements may have been estimated through extrapolation. Actual results may vary. Users of this document should verify the applicable data for their specific environment.

© Copyright IBM Corp. 2015. All rights reserved.

Trademarks

Lenovo, the Lenovo logo, and For Those Who Do are trademarks or registered trademarks of Lenovo in the United States, other countries, or both. These and other Lenovo trademarked terms are marked on their first occurrence in this information with the appropriate symbol (® or ™), indicating US registered or common law trademarks owned by Lenovo at the time this information was published. Such trademarks may also be registered or common law trademarks in other countries. A current list of Lenovo trademarks is available on the Web at http://www.lenovo.com/legal/copytrade.html.

The following terms are trademarks of Lenovo in the United States, other countries, or both:

Advanced Settings Utility™
BladeCenter®
Dynamic System Analysis™
Lenovo®
NeXtScale™
Lenovo(logo)®
ServeRAID™
ServerProven®
System x®
ToolsCenter™

The following terms are trademarks of other companies:

Intel, and the Intel logo are trademarks or registered trademarks of Intel Corporation or its subsidiaries in the United States and other countries.

Linux is a trademark of Linus Torvalds in the United States, other countries, or both.

Internet Explorer, Microsoft, Windows, Windows Server, and the Windows logo are trademarks of Microsoft Corporation in the United States, other countries, or both.

Other company, product, or service names may be trademarks or service marks of others.

Preface

Data security is one of the paramount requirements for organizations of all sizes. Although many companies invested heavily in protection from network-based attacks and other threats, few effective safeguards are available to protect against potentially costly exposures of proprietary data that results from a hard disk drive being stolen, misplaced, retired, or redeployed.

Self-encrypting drives (SEDs) can satisfy this need by providing the ultimate in security for data-at-rest and can help reduce IT drive retirement costs in the data center. Self-encrypting drives are also an excellent choice if you must comply with government or industry regulations for data privacy and encryption.

To effectively manage a large deployment of SEDs in Lenovo® System x® servers, an organization must rely on a centralized key management solution. This Lenovo Press book explains the technology behind SEDs and demonstrates how to deploy a key management solution that uses IBM Security Key Lifecycle Manager and properly setup your System x servers.

The team who wrote this book

This document is produced by the following subject matter experts working in the Lenovo offices in Morrisville, NC, USA.

Ryan Bradley is an IT Consultant with Lenovo System x Enterprise Solution Services (xESS), formerly known as Lab Based Services (LBS). After starting eight years ago at IBM in Tools Center development, Ryan now has more than four years experience designing, implementing, and providing skills transfer on IBM hardware, software, cloud, and management solutions for clients. His areas of expertise include System x, Flex, and BladeCenter® hardware, and virtualization, system networking, and system storage.

Angelo Parisi is a Certified I/T Specialist with the Lenovo System x Client Technical Sales (CTS) group. He started his career at IBM in 1995 with the Business Partner Support group. Several years later, he moved to the newly formed x86 Server team where he works today. Currently, he is the team lead for the North American Region where he tends to some of the largest IBM accounts in his territory. With over a decade of experience working with large enterprise customers, Angelo has experience with large-scale and distributed systems, which he uses as a regular presenter at IBM Tech Edge events.

Thanks to the following people for their contributions to this project:

Axel Buecker
Andy Ehrenzeller
Luis Giron
David Watts
Lenovo

W. Craig Johnston
IBM

© Copyright IBM Corp. 2015. All rights reserved.

Comments welcome

Your comments are important to us!

We want our books to be as helpful as possible. Send us your comments about this book or in one of the following ways:

- Use the online feedback form found at the web page for this document:

  `http://lenovopress.com/sg248247`

- Send your comments in an email to:

  `comments@lenovopress.com`

Do you have the latest version?

We update our books and papers from time to time, so check whether you have the latest version of this document by clicking the **Check for Updates** button on the front page of the PDF. Pressing this button will take you to a web page that will tell you if you are reading the latest version of the document and give you a link to the latest if needed. While you're there, you can also sign up to get notified via email whenever we make an update.

1

Technology primer

In this chapter, we provide a primer for the two technologies that can empower an organization to use self-encrypting drive (SED) technology and combine it with a centralized encryption key lifecycle management solution for their System x servers.

We begin by describing the SED drive technology and how it is implemented in the System x servers. Next, we review the IBM Security Key Lifecycle Manager (SKLM) solution with which you can centrally manage your drive encryption keys (and more). We then describe a set of typical deployment scenarios for encrypted disks.

This chapter includes the following topics:

- Self-encrypting drive technology
- IBM Security Key Lifecycle Manager
- Deployment scenarios
- Conclusion

© Copyright Lenovo 2015. All rights reserved.

1.1 Self-encrypting drive technology

Data security is a growing requirement for organizations of all sizes. Although many companies invested heavily to protect themselves from network-based attacks and other threats, few effective safeguards are available to protect against potentially costly exposures of proprietary data that results from a hard disk drive (HDD) being stolen, misplaced, retired, or redeployed.

SEDs can satisfy this need by providing the ultimate security for data-at-rest. SEDs also can help reduce IT drive retirement costs in the data center. When combined with the compatible RAID controllers, the Serial Attached SCSI (SAS) SEDs in System x servers can deliver superb performance per watt with a cost-effective, secure solution for organizations of all sizes. Self-encrypting drives are also an excellent choice if you must comply with government or industry regulations for data privacy and encryption.

SAS SEDs have the following characteristics and capabilities:

- Interface speeds of 6 Gbps and 12 Gbps
- Rotational speeds of 7,200 RPM, 10,000 RPM, and 15,000 RPM
- Single hard disk drive capacities of 146 GB, 300 GB, 600 GB, 900 GB, 1.2 TB, 1.8 TB, 2 TB, 4 TB, or 6 TB
- Support for Native Command Queuing (NCQ)
- Support for Self-Monitoring, Analysis, and Reporting Technology (SMART)
- 2.5-inch and 3.5 inch form-factor available
- Hot-swap HDDs
- Encrypt data dynamically at the drive level with no performance effect
- Provide instant secure erasure (cryptographic erasure, so data is no longer readable)
- Enable auto-locking to secure data if a drive is misplaced or stolen while in use

When the SED is in normal use, its owner does not need to maintain authentication keys (that is, credentials or passwords) to access the data on the drive. The SED encrypts data that is being written to the drive and decrypts data that is being read from it, all without requiring an authentication key from the owner.

SEDs eliminate the need to overwrite, destroy, or store retired drives. When it is time to retire or repurpose the drive, the owner sends a command to the drive to perform a cryptographic erasure. The process is nearly instantaneous, regardless of the capacity of the drive. Cryptographic erasure replaces the encryption key that is inside the encrypted drive, which makes it impossible to ever use the deleted key to decrypt the encrypted data.

SEDs reduce IT operating expenses by reducing asset control challenges and disposal costs. Data security with SEDs helps ensure compliance with privacy regulations without hindering IT efficiency.

The use of an SED when auto-lock mode is enabled requires securing the drive with an authentication key. When secured in this manner, the drive's data encryption key is locked whenever the drive is powered down. When the SED is switched off or becomes unplugged, it automatically locks the drive's data. When the SED is powered on again, it requires authentication before it can unlock the encryption key and read any data on the drive. This requirement protects against mis-placement and theft.

The hardware encryption engine on the drives matches the SAS port's maximum speed and encrypts all data with no performance degradation. This performance scales linearly and automatically with each drive that is added to the system. No processor cycles from the host are necessary, and I/O operations occur without interruption.

ServeRAID™ M Series controllers offer SED support with any RAID 5 upgrade (with or without cache memory); therefore, no other licensing is required.

For more information, see the *Self-Encrypting Drives for System x*, TIPS0761, which is available at this website:

http://lenovopress.com/tips0761

1.1.1 Benefits of SED technology

The threat of data exposure increased over time. Although most current protection efforts focus around securing the transmission of data, the abilities of protecting data-at-rest changed little. Software-based encryption strategies have a serious effect on performance and require careful consideration of the operating system environment on which they are implemented. Any change in the operating system (including service packs) can result in having to retest the entire solution or wait for the provider to certify new environments.

Driven by the current state of cybercrime, government legislation and industry privacy requirements to safeguard data are on the rise in many countries. This safeguarding of data includes not only data transmission but also the disposal of data when storage media fails or is retired from active use. In a time when organizations try to drastically reduce their IT budgets on an annual basis, physically destroying or degaussing devices are not only costly but are not supported by the drive manufacturers.

Alternative methods (such as multi-pass data overwrite) are unsuitable in this age of rapidly increasing storage capacities. Although a 4.51 GB drive might take only a couple of hours, today's multi-terabyte drives can take days. Also, if the drive fails, there is no mechanism to destroy the data in a warranty-approved manner.

SEDs protect confidential or proprietary information that is stored locally on the server by encrypting the data with an AES-based cypher before it is physically written to the media. By performing this encryption at the last step before writing the data with a dedicated AES processor, SED drives provide scalable performance. This performance is because each drive has a dedicated AES processor. This configuration removes the encumbrance of encryption from being handled by a single processor on the RAID controller or adding CPU usage at the operating system level. As drives are added to the system for capacity or performance growth, each new drive includes its own AES processor. Also, by having clear data access from the operating system to the HDD, there is no requirement for operating system-specific support for the encryption. This configuration protects the organization's investment by not limiting them to specific operating builds or new releases of agents to support the encryption. No unique steps are required to install an operating system on a server that is using SED drive technology.

In summary, SED drives reduce the vulnerability of data-at-rest to potentially costly exposures of proprietary data that result from hard disk drive theft, misplacement, or improper drive disposal. There is no need for time-consuming wiping of drives that take even longer to complete as the capacity of the storage devices increase.

1.1.2 Certification standards

The encryption capabilities of SED drives are implemented in a way that meets or exceeds the requirements for federal government standards to Federal Information Processing Standard (FIPS) level 2. This certification is the result of extensive testing by federal security specialists and it is a testament to the strength of the encryption that is used on the device.

Specifically, the SED drives are validated according to the Trusted Computing Group Enterprise SSC Revision 1.0 standard.

Trusted Computing Group Enterprise SSC Revision 1.0

Encryption/FIPS — FIPS 140-2 Validated SEDs were certified by the US National Institute of Standards and Technology (NIST) and Canadian Communications Security Establishment (CSE) as meeting the Level 2 security requirements for cryptographic modules as defined in the Federal Information Processing Standards (FIPS) 140-2 Publication.

For more information about FIPS compliance, see this website:

`http://www.seagate.com/tech-insights/fips-140-2-standard-and-self-encrypting-drive-technology-master-ti/`

For more information about FIPS specification, see this website:

`http://csrc.nist.gov/groups/STM/cmvp/standards.html`

1.1.3 How SED drives work

With SED technology, the configured storage capacity is presented to the operating system as regular block level storage, as with any typical disk-based storage media. By using standard file management tools, it is not possible to differentiate between an encrypted volume and an unencrypted volume because all of the encryption occurs at the individual drive level within the hardware.

After the encrypted data is read from the spinning disk inside the drive, it is then decrypted in the drive controller and sent to the RAID adapter as clear unencrypted data. This process allows standard RAID drivers to be used with complete transparency at the operating system level.

Regardless of whether a drive was secured with the management tools, the data is always encrypted on the physical disk. At the time of manufacture, each SED is configured with a random AES key that is used to encrypt all data that is written. This key is referred to as the *Media Encryption Key* (MEK) and is stored in a hidden section of the disk, which is often referred to as *Band 0* or the *Global Band*.

If the drive was not secured by a RAID adapter, it has access to this AES key at start and loads this key at power-on to read and write data to the disk platters. In this mode, it functions as any normal hard disk that you are familiar with, it just happens to be encrypting and decrypting the data in real time. Because all data that is sent from and returned to the controller is unencrypted, there is no change to the standard installation procedures for an operating system.

The storage of the MEK key is shown Figure 1-1.

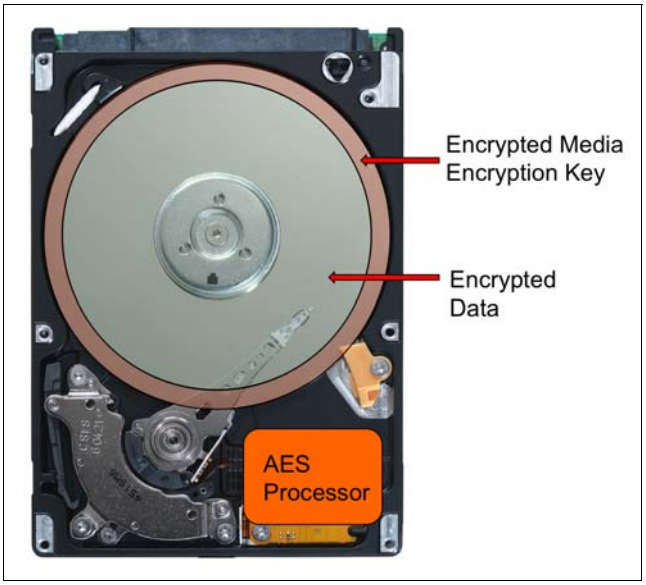

Figure 1-1 SED disk usage

After the drive is configured as part of an array or virtual drive, the management tool can be used to secure the volume. This process encrypts the drive-based AES MEK with another key that is managed by the RAID adapter or a dedicated key management server. This managed key, whether managed by the RAID controller or an external key management server, is referred to as the *Key Encryption Key* (KEK). In this scenario, the hard disk drive cannot access the MEK to decrypt the data that is stored on its platters unless it is paired with a RAID adapter that passes the correct KEK on to the drives at boot time. It is this key encryption that prevents the drive from being accessed if the drive fails.

Figure 1-2 shows how this exchange of keys occurs when the server is powered on.

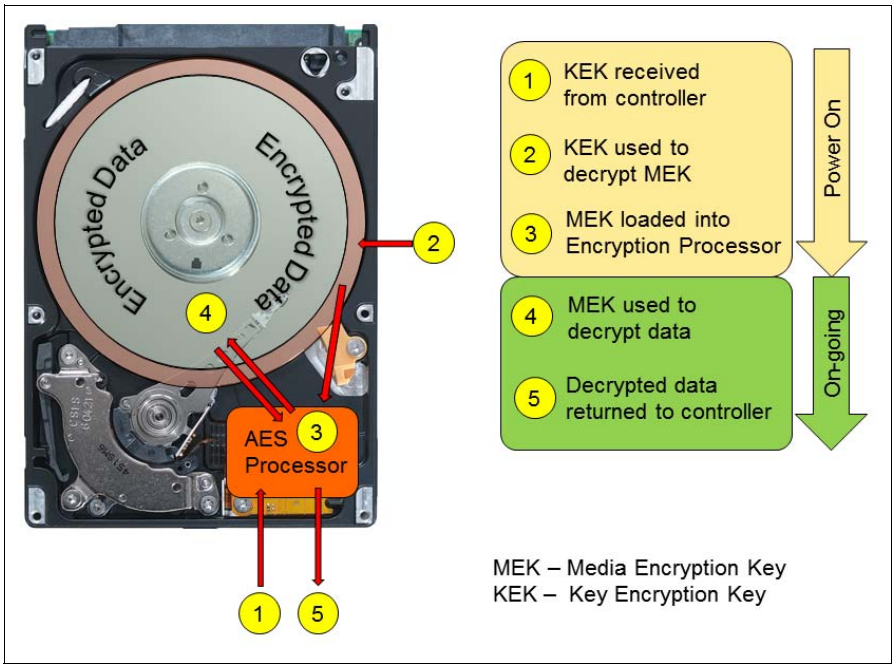

Figure 1-2 Secured SED drive boot process

If the drives are removed or some change occurs where the drives cannot obtain the KEK that is used to encrypt the data, the drives can no longer read the data and the disk is referred to as *cryptographically sanitized*. This status is functionally equivalent to, or exceeds, the data disposal capability of a three-pass data destruction tool. If the drives are reconnected to the same system or the key is restored to a new RAID adapter (as in the case of a service call), the drives can regain access to the data.

The important concept behind SEDs relative to controller- or operating system-based encryption is that because the encryption occurs at the last stage in the write process or the first stage of a read process, all data that is flowing in and out of the drive is clear or decrypted. Therefore, there is no effect on how the data is used or what operating system is employed because there are no specific drivers or agents that are required above the standard operating system driver for the RAID adapter in use. If the appropriate RAID adapter driver is supported by the intended operating system, no extra testing or configuration is required to make the storage available for use.

Figure 1-3 shows the components of the data flow (which are visualized as *Customer name*) that are encrypted versus decrypted.

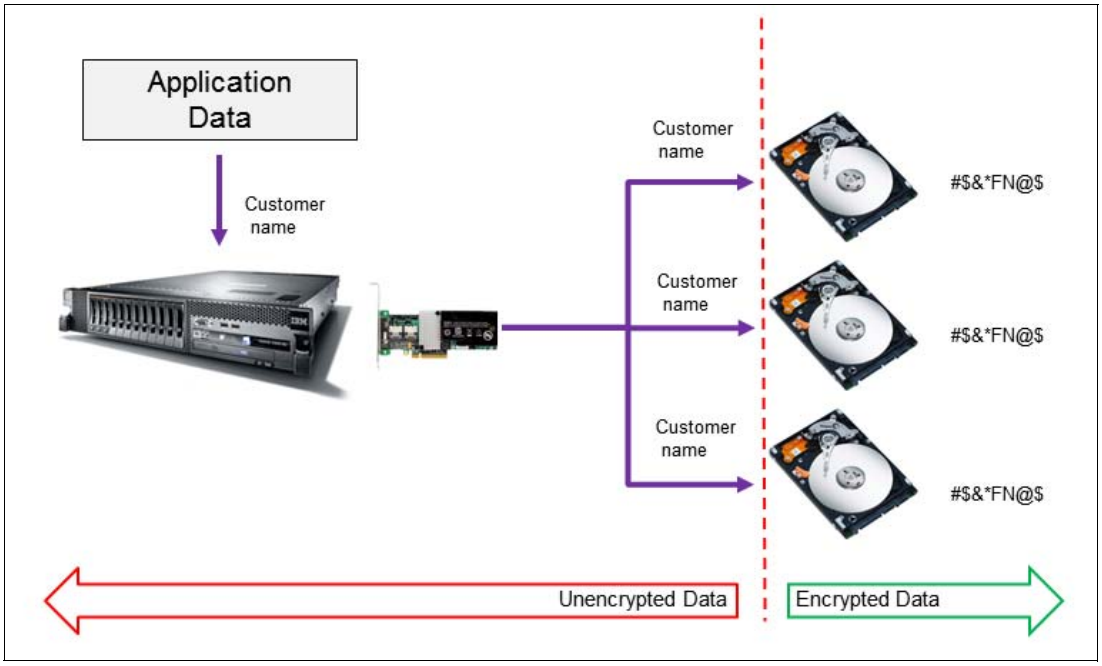

Figure 1-3 SED encryption

Data removal

Equally important as how the data is encrypted is how the data on a disk is destroyed for disposal, resale, or redeployment. Standard methods of multi-pass overwrite are too time-consuming to be viable given the increase in capacity of current storage devices. The alternatives of degaussing or physical destruction are not economical alternatives because they void any warranty that is associated with the device, are not supported by the vendors for data disposal, and destroy any resale value of the device.

Because SEDs are always encrypting the data that is written to the physical media with the MEK, they support a function that is called *Secure Instant Erase*. This function is a standards-approved method to destroy the data on the device by randomizing the encryption keystore (MEK) on the drive. Because the drive does not have the valid MEK that is required to decrypt the data, this state renders all of the data on the device invalid instantly, regardless

of the capacity of the disk. This data disposal method is referred to as *cryptographic sanitization*.

This cryptographic sanitization is an FIPS and warranty approved method for data disposal.

1.2 IBM Security Key Lifecycle Manager

You can use SKLM to create, back up, and manage the lifecycle of keys and certificates that an organization uses. You can manage encryption of symmetric keys, asymmetric key pairs, and certificates. SKLM provides a graphical user interface, command-line interface, and REST interface to manage keys and certificates.

SKLM waits for and responds to key generation or key retrieval requests that arrive through TCP/IP communication. This communication can be from a tape library, tape controller, tape subsystem, device drive, or tape drive.

This book focuses on the use of SKLM with System x servers and SEDs.

SKLM provides the following features:

- Manage symmetric keys, asymmetric key pairs, and X.509 V3 certificates.
- Manage the creation and lifecycle of keys, which contain metadata on their intended usage.
- Provide protected backup of critical data for disaster recovery. For example, on distributed systems, backup includes cryptographic key data (actual keys and certificates that are managed), metadata about the keys, and configuration files.

1.2.1 SKLM components

The SKLM solution on distributed systems includes the SKLM server, WebSphere Application Server, and DB2.

The WebSphere Application Server runs a Java virtual machine that provides the runtime environment for the application code. The application server provides communication security, logging, messaging, and web services.

For more information about SKLM, see this website:

`http://www.ibm.com/software/products/en/key-lifecycle-manager`

1.2.2 Keys overview

An encryption key often is a random string of bits that is generated specifically to scramble and unscramble data. Encryption keys are created by using algorithms that are designed to ensure that each key is unique and unpredictable. The longer the key that is constructed this way, the more difficult it is to break the encryption code.

SKLM uses two types of encryption algorithms: symmetric algorithms and asymmetric algorithms. Symmetric, or secret key encryption, uses a single key for encryption and decryption. Symmetric key encryption is used to encrypt large amounts of data efficiently.

Advanced Encryption Standard (AES) keys are symmetric keys that can be three different key lengths (128, 192, or 256 bits). AES is the encryption standard that is recognized and

recommended by the US government. The 256-bit keys are the longest keys that are allowed by AES. By default, SKLM generates 256-bit AES keys.

Asymmetric, or public and private encryption, uses a pair of keys. Data that is encrypted by using one key can be decrypted by using only the other key in the public and private key pair. When an asymmetric key pair is generated, the public key often is used to encrypt, and the private key often is used to decrypt.

SKLM uses symmetric and asymmetric keys. Symmetric encryption enables high-speed encryption of user or host data. Asymmetric encryption, which is necessarily slower, protects the symmetric key.

1.2.3 SKLM creates, stores, and manages keys

The SKLM creates key material by using a random number generator. It stores the keys in a secure DB2 database. Requests for keys are serviced over a TCP/IP connection.

For the System x SEDs, SKLM creates a key container that is used by external devices. The SED device stores its MEK that is encrypted under the KEK that is provided by SKLM. At System x server start, the devices contact SKLM to obtain the KEK.

1.2.4 SSL/TLS session security

The connection between the System x server and SKLM is secured through SSL/TLS protocols.

To retrieve a KEK from SKLM, the device must authenticate the server. This authentication is performed by using SSL protocols. Before a key exchange operation is started, the proper security mechanisms must be in place.

A digital certificate is generated at the SKLM key manager. This certificate is exported by using the SKLM command-line interface. The exported certificate is then imported into each device that uses keys from SKLM. Also, each device generates and exports a digital certificate to be imported as a client certificate by the SKLM key manager.

1.3 Deployment scenarios

Regardless of how the encryption keys are managed (whether they are configured on the local RAID adapter or provided from an external key management server), the manner in which the data is encrypted is identical. The component that does change is how the keys are managed by the user and the level of protection and interaction that is involved in the deployment. That is, the MEK is always used in the same manner; it is the management of the KEK that changes that is based on requirements of the solution.

To help explain the various manners in which SED drives can be deployed, we created four sample scenarios that range from unsecured configurations to centrally managed key management servers. Each scenario describes the drawbacks and benefits of the implementation and highlights environments in which it often is used.

The following scenarios are not industry-standard scenarios; instead, they are constructs that were developed by the authors to describe the various ways that SED drives can be implemented in an environment:

- Scenario 1: No key required

- Scenario 2 encrypted: Unattended mode
- Scenario 3 encrypted: Attended mode
- Scenario 4 encrypted: External key management

1.3.1 Scenario 1: No key required

As shown in Figure 1-4, SEDs are used in System x servers with no other configuration beyond the standard array and virtual drive management that is used in the deployment of a typical server.

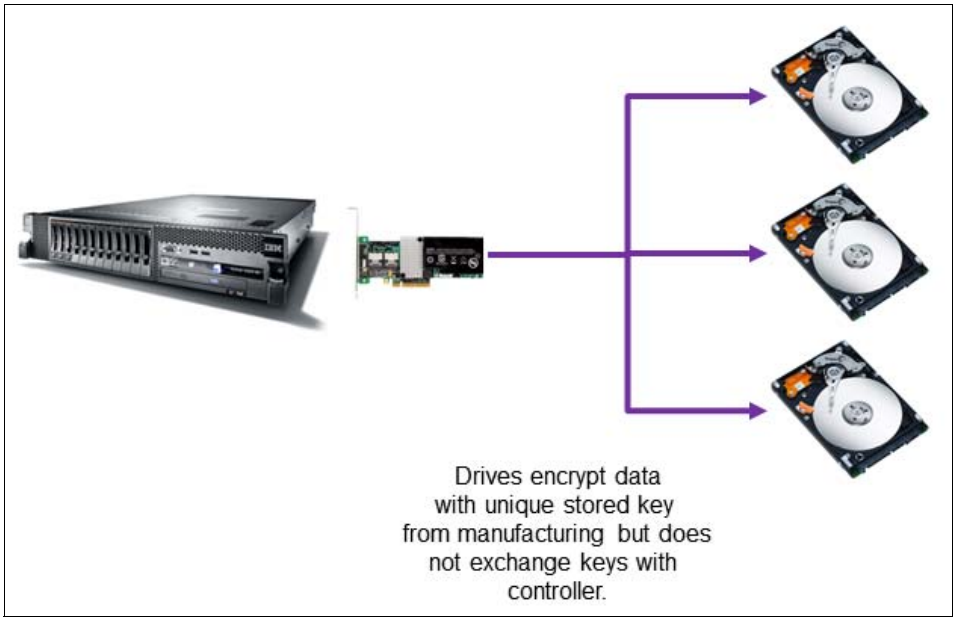

Figure 1-4 No key required

This scenario is based on the deployment of SEDs in the place of regular storage devices with no other configuration steps that are performed beyond the standard creation of arrays and virtual disks. Although this scenario does not use the security features of the drives, it does allow for the use of the secure instant erase function of the SED technology. In this case, the data is not protected against theft; however, the devices can be securely erased instantly for data disposal.

The major drawback of this implementation is that a failed drive cannot be erased because it cannot be accessed to randomize the MEK and requires alternative data disposal, such as physical destruction of the device.

This implementation often is used when an organization is unsure of the technology or not ready for the deployment of a solution that requires key management. By deploying the SEDs in this manner, an organization can introduce the drives to their environment without any changes to their deployment or management methods. When ready, the organization can enable the extra functionality with no effect on the data that is stored on the devices.

This implementation includes the following advantages:

- Understood technology that was in practical use for years
- Operating system intervention is not required (not apparent)
- No specialized service requirements
- Secure erase function
- No licensing requirements

This implementation includes the following disadvantages:

- Data is not protected against physical theft of drives
- Failed drives cannot be erased

1.3.2 Scenario 2 encrypted: Unattended mode

Scenario 2 builds on the configuration of Scenario 1 and takes it a step further by using the local KEK management of the RAID adapter to encrypt the MEK that is present on the installed SEDs. This scenario is referred to as *unattended mode* because no user intervention is required during the regular boot cycle of the server.

Figure 1-5 on page 10 shows the implementation of Scenario 2.

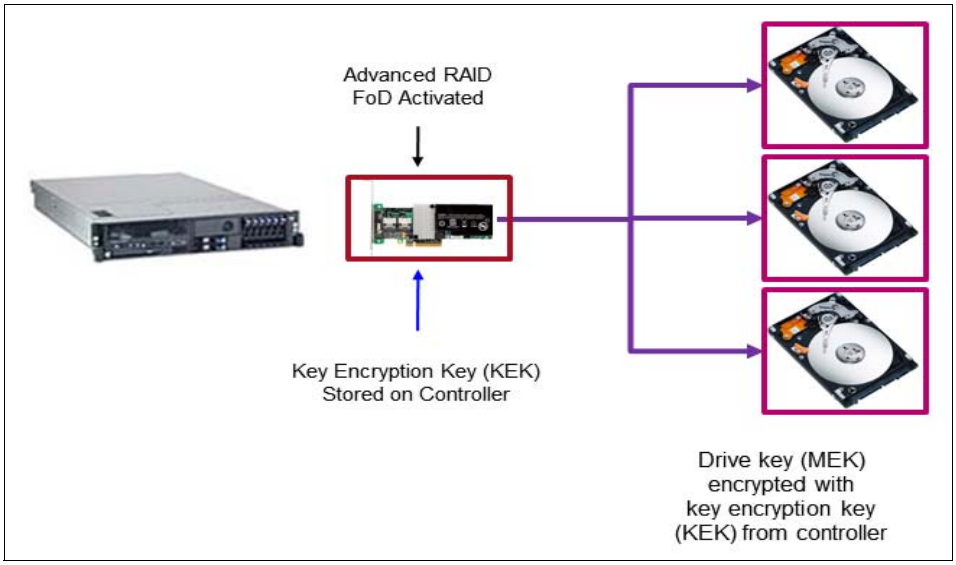

Figure 1-5 Unattended mode

Scenario 2 shows a deployment in which an organization wants to use the encryption capability of secured virtual disks while minimizing the effect on the environment regarding key management. In this implementation, after the arrays and virtual disks are created, drive security is configured on the RAID controller, which generates a KEK and the virtual disks are secured by using this KEK to encrypt the MEK that is present on the drives within the target array. This configuration effectively binds the disks to the controller, which renders the data sanitized if it is removed because the drive cannot access the required KEK to decrypt the drive MEK.

This scenario is an improvement over Scenario 1 because the drives cannot be instantly erased only; any drive that might fail automatically has the data sanitized because another controller cannot be used to recover the data. When the appropriate KEK is unavailable to the drive, the MEK cannot be read. This condition protects against a failed drive having the controller board replaced because the KEK is not stored anywhere on the disk.

This implementation includes the following advantages:

- No operating system intervention is required (not apparent)
- Drive data is protected against theft (data is encrypted)
- No boot time intervention is required
- Instant secure data disposal

- Encryption can be activated at any point from local or remote GUI or command line without data loss

This implementation includes the following disadvantages:

- Data is not protected against theft of the complete server because the controller provides the keys to the drives at boot time without intervention
- More service steps are required for controller replacement to reset keys
- More setup is needed to establish the initial keys
- After it is encrypted, a volume cannot be decrypted without destroying data

Effect on service and support

The downside of this scenario is the effect on server maintenance. Because the keys for the volumes are stored on the RAID controller in this scenario, any effect on that RAID adapter requires more steps to recover the data. After a new controller is installed, any secured volumes show as a foreign encrypted array.

For the new controller to access the data on this volume, the encryption key that is used to secure the volume on the original RAID controller must be restored to the new RAID controller. Therefore, it is imperative that whenever disk encryption is configured, any security keys are backed up and tracked to minimize the effect of service actions.

Figure 1-6 shows the different impact situations to a service call on the server. These situations address a drive failure and controller failure.

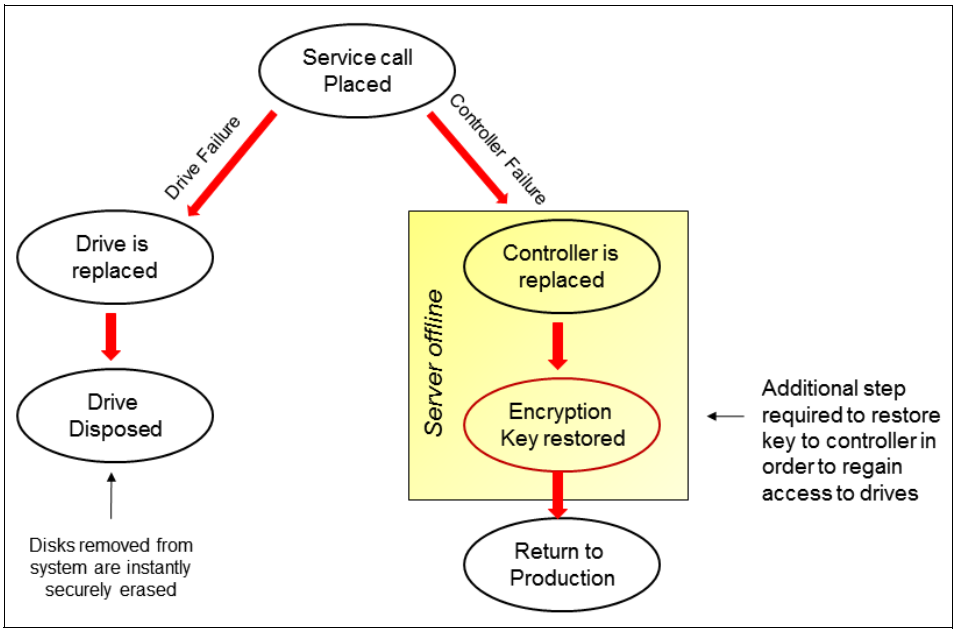

Figure 1-6 Service and support impact

1.3.3 Scenario 3 encrypted: Attended mode

Scenario 3 takes the configuration in Scenario 2 and adds to it a boot time pass phrase that must be provided to the controller at boot time to prevent the KEK from being passed to the drives without proper authorization. For this reason, this scenario (as shown in Figure 1-7) is referred to as the *attended mode* because intervention is required when the server is booted.

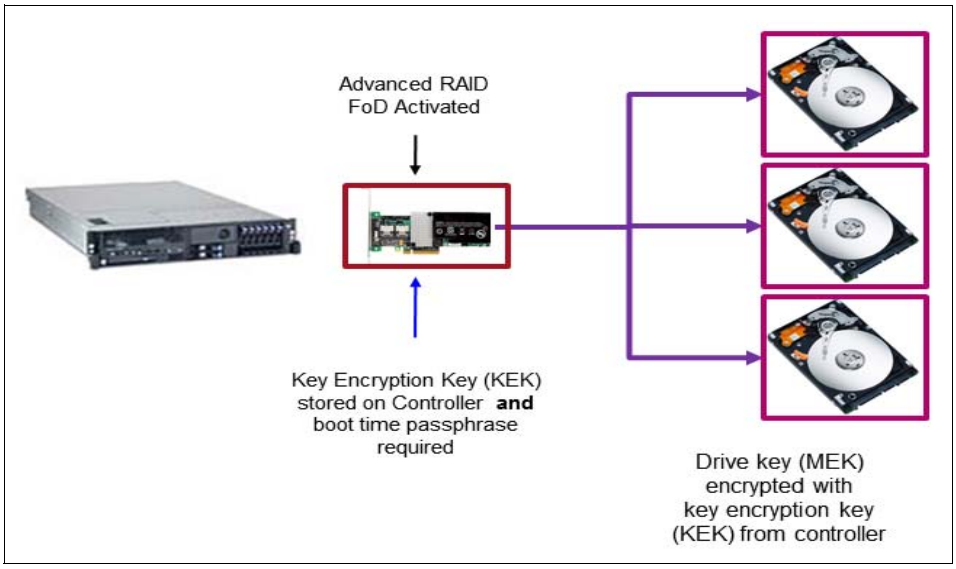

Figure 1-7 Scenario 3

Scenario 3 is identical to the configuration of Scenario 2 with the simple exception that the RAID controller is configured with a boot-time password. Although this change is simple, it affects the implementation by making two important changes.

The first change is that user level integration is required at any time that the server is restarted. During post, the server pauses at the RAID controller firmware initialization window and prompts the operator for a valid password. If one cannot be provided, any volumes that were secured are not accessible. If a valid password is provided (whether via local keyboard or a remote KVM), the RAID controller then passes the KEK to the disks to unlock the MEK and the boot process continues normally.

The second change that this configuration introduces is the management of the password. This change is significant because the password must be manually entered by a user at boot time. The main question this issue raises is who has access to the password and what is the plan of action if that user is not available in an after-hours situation.

The main benefit of Scenario 3 over Scenario 2 is the protection of the data if there is a theft or the server is decommissioned. In Scenario 2, if someone can obtain the entire server, the encryption is rendered irrelevant because the keys are automatically passed to the disks at boot time. By introducing the password in Scenario 3, all data is rendered unreadable unless the password can be provided.

This implementation includes the following advantages:

- No operating system intervention is required (not apparent)
- Entire server is protected against data theft (data is encrypted) and requires boot-time intervention
- Instant secure data disposal

This implementation includes the following disadvantages:

- More service steps are required for controller replacement to reset keys
- More setup is needed to establish the initial keys
- After it is encrypted, a volume cannot be decrypted without destroying data
- Password must be entered when the server is restarted

Effect on service and support

The downside of this scenario is the effect on server maintenance. In this scenario, any effect on that RAID adapter requires more steps to recover the data because the keys for the volumes are stored on the RAID controller. After a new controller is installed, any secured volumes show as a foreign encrypted array.

For the new controller to access the data on this volume, the encryption key that was used to secure the volumes on the original RAID controller must be restored to the new RAID controller. Therefore, it is imperative that whenever disk encryption is configured, any security keys are backed up and tracked to minimize the effect on service actions.

Figure 1-8 shows the different impact situations to a service call on the server. These situations address a drive failure and controller failure. The service impact for Scenario 3 is identical to Scenario 2.

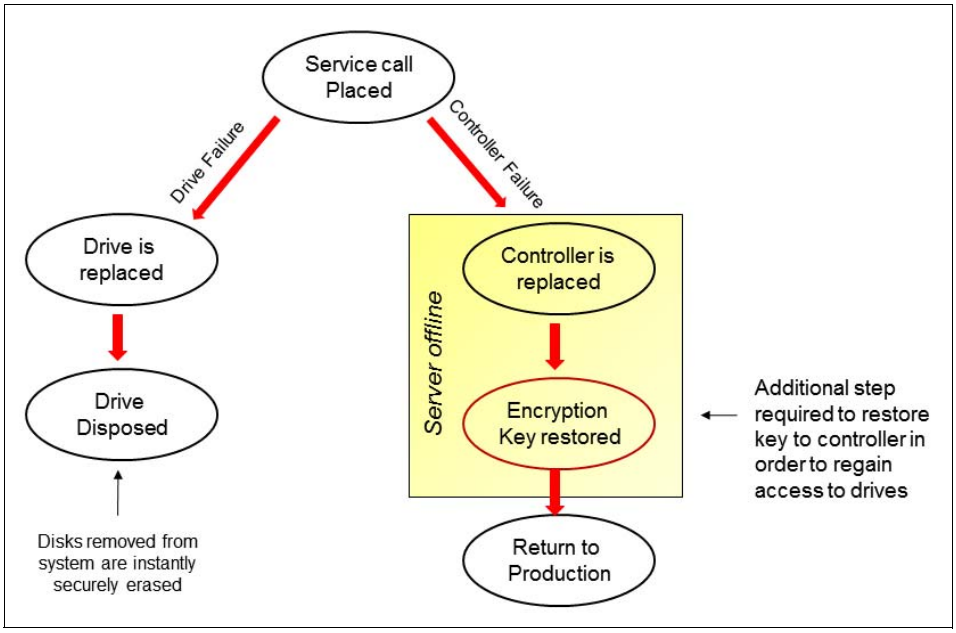

Figure 1-8 Service and support impact

1.3.4 Scenario 4 encrypted: External key management

Scenario 4 is the most beneficial scenario for deployments of all sizes because it adds centralized KEK management to the environment. This management provides the full benefit of SEDs to the organization while avoiding the need for boot time intervention or the manual input of passwords to secure a server.

Figure 1-9 shows the components that comprise this solution.

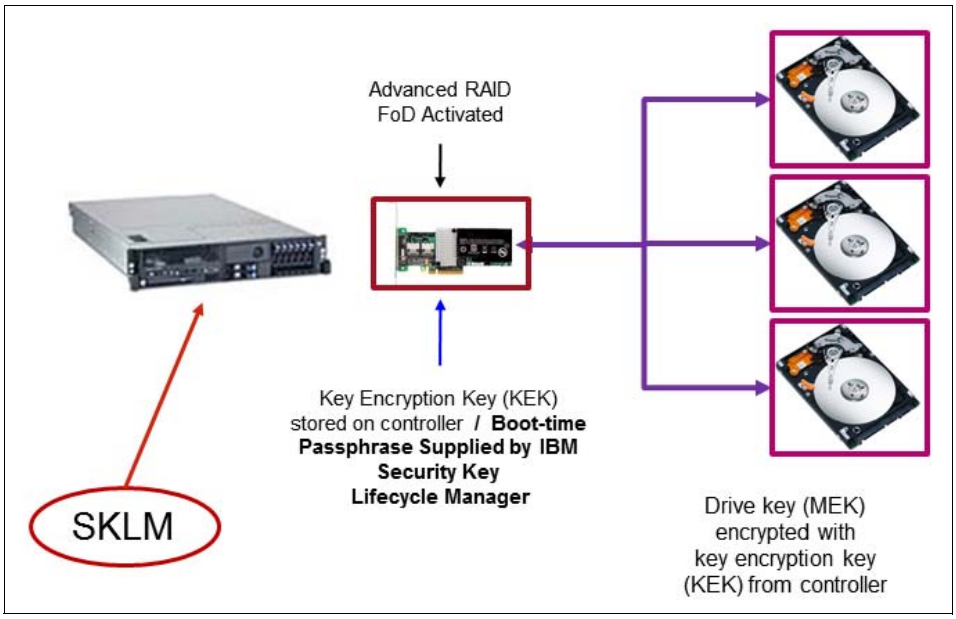

Figure 1-9 Scenario 4

In this scenario, an external key management server provides KEK keys to the server at boot time. This ability requires firmware support because the server must establish a trusted network connection to a key management server (such as SKLM) during boot and receive a KEK key that is used to decrypt the MEK before the server can complete the boot process.

As in scenarios 2 and 3, the volumes on the RAID controller are secured encrypting the MEK keys; however, the KEK keys are not stored locally on the controller. This configuration allows for the central management of the KEK keys for larger deployments of servers (including remote branch offices) and removes the necessity of boot time intervention if there is a server restart. Also, this process automatically sanitizes all data on the server by removing it from the corporate network where the key management server is or by revoking the keys from the management server when a server is retired or repurposed.

Remote key management requires the following components to be in place to support this boot time process:

- Integrated Management Module (IMM) v2 firmware support

  The server must support external key management and must have an IMM firmware level at or greater than when the support was introduced for a product (for more information, see Chapter 2, “Supported systems and sample configuration” on page 17).
- A supported external key management server must be configured and accessible on the IMM network.
- IMM v2 must be configured with a self-signed certificate or one that was signed by a certificate authority.
- IMM v2 must have a key management server certificate installed.
- Key management server must have the target system certificate installed.
- RAID adapter must be configured to use an external key management source.

Although this scenario does initially introduce some complexity to the network, the use of an external key management server allows for a simplification of key management over previous scenarios and provides for better scalability for larger or distributed environments. Many organizations already have key management servers in their data centers to handle the

needs of securing data-at-rest for formats (such as tape) where the data often is encrypted for off-site storage.

This scenario represents the implementation that was described within the scope of this book.

This implementation includes the following advantages:

- No operating system intervention is required (not apparent)
- Entire server is protected against data theft (data is encrypted)
- No boot time intervention is required (keys are managed by SKLM)
- Instant secure data disposal

This implementation includes the following disadvantages:

- More service steps are required for controller replacement to reset keys
- More is needed setup to establish the initial keys
- After it is encrypted, it cannot be disabled without destroying data
- Central SKLM infrastructure must be created and maintained

Effect on service and support

The downside of this scenario is the effect on server maintenance. If the RAID controller is replaced in this scenario, the new RAID controller must be configured to import the existing drive configuration and it must be set correctly to use an external key management server.

If the system board is replaced, the following tasks must be completed:

1. Restore IMM configuration parameters to enable communication with the external key manager:
   a. Reapply needed FoD options.
   b. Restore external key manager addresses.
   c. Restore server and key manager certificates.
   d. Restore original Server UUID.
2. At the SKLM server, accept the System x server (if a new certificate is used).

When external keys are managed by using SKLM, the server UUID is used to associate any specific System x server with the existing KEK that is needed to decrypt the MEK. Therefore, when the system board is replaced, the Server UUID must be restored before the server can obtain the existing KEK from the key manager and gain access to the SEDs at boot time. Any change in the key that is allocated to the repaired server renders all of data inaccessible by design.

Figure 1-10 on page 16 shows the recovery procedure for failed components in this scenario, including drive failure, controller failure, and planar failure.

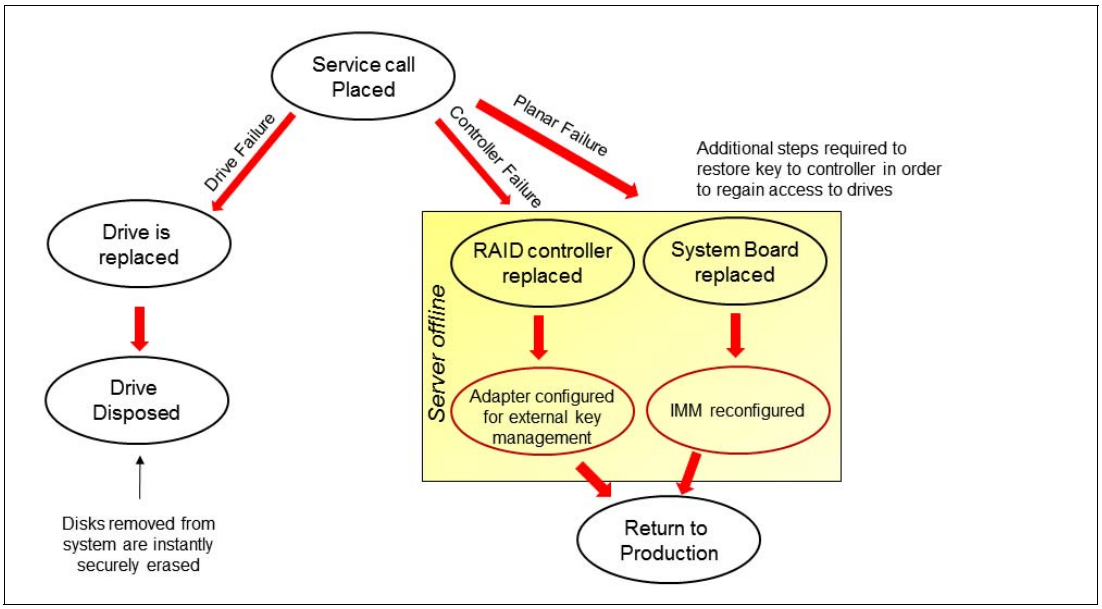

Figure 1-10 Service and support impact

1.4 Conclusion

As described in this chapter, there are several methods in which SEDs can be deployed. The best solution for a specific implementation depends on many factors, including the level of security that is required, the number of systems that are deployed, and the availability of managed key servers.

If key management servers can be on the corporate network, the deployment of Scenario 4 can provide the highest degree of central management and flexibility for the IT environment. It is the installation of Scenario 4 that we describe further in Part 2 of this book.

2

Supported systems and sample configuration

In this chapter, we describe the supported configurations and options for self-encrypting drives in System x servers. We also describe our example configuration that was used as a proof of concept to create the installation instructions for this book.

This chapter includes the following topics:

- Supported systems and options
- Example configuration
- Conclusion

© Copyright Lenovo 2015. All rights reserved.

2.1 Supported systems and options

The servers, RAID adapters, and drives that are described in this section are supported at the time this writing. For more information about the current list of supported configurations, see this website:

http://www.lenovo.com/us/en/serverproven/

2.1.1 Supported servers

The supported System x server systems for external key management (as of this writing) are listed in Table 2-1.

Table 2-1 Supported servers

Server	Machine Type
System x3100 M5	5457
System x3250 M5	5458
System x3300 M4	7382
System x3500 M4	7383
System x3500 M4 (E5-xxxxV2)	7383, E5-xxxxV2
System x3530 M4	7160
System x3530 M4 (E5-xxxxV2)	7160, E5-xxxxV2
System x3630 M4	7158
System x3630 M4 (E5-xxxxV2)	7158, E5-xxxxV2
System x3550 M4	7914
System x3550 M4 (E5-xxxxV2)	7914, E5-xxxxV2
System x3550 M5	5463
System x3650 M4	7915
System x3650 M4 (E5-xxxxV2)	7915, E5-xxxxV2
System x3650 M4 HD	5460
System x3650 M5	5462
System x3750 M4	8722/8733
System x3750 M4	8752/8718
System x3850 X6/x3950 X6	3837
NeXtScale™ nx360 M5	5465

2.1.2 Supported RAID adapters

At the time of this writing, M5110(e) and M5210(e) RAID adapters are supported for the use of external key management with servers that are supported for ServerProven®, as described in 2.1.1, “Supported servers ” on page 18. The installation of any RAID 5, RAID 6,

or supported cache modules automatically enables support for securing SED-based virtual drives; external key management requires the purchase of a Features on Demand (FoD) license.

Table 2-2 lists the supported RAID adapters and the corresponding upgrades.

Table 2-2 Supported RAID controllers

Part number	Description
Supported RAID adapters M5110	
81Y4481	ServeRAID M5110 SAS/SATA Controller for System x
Onboard	ServeRAID M5110e SAS/SATA Controller for System x
One of the following upgrades is required to support SEDs with the M5110 RAID controller	
81Y4544	ServeRAID M5100 Series Zero Cache/RAID 5 Upgrade for System x
81Y4484	ServeRAID M5100 Series 512 MB Cache/RAID 5 Upgrade for System x
81Y4487	ServeRAID M5100 Series 512 MB Flash/RAID 5 Upgrade for System x
81Y4559	ServeRAID M5100 Series 1 GB Flash/RAID 5 Upgrade for System x
47C8670	ServeRAID M5100 Series 2 GB Flash/RAID 5 Upgrade for System x
Supported RAID adapters M5210	
46C9110	ServeRAID M5210 SAS/SATA Controller for System x
Onboard	ServeRAID M5210e SAS/SATA Controller for System x
One of the following upgrades is required to support SEDs with the M5210 RAID controller	
47C8708	ServeRAID M5200 Series Zero Cache/RAID 5 Upgrade
47C8656	ServeRAID M5200 Series 1 GB Cache/RAID 5 Upgrade
47C8660	ServeRAID M5200 Series 1 GB Flash/RAID 5 Upgrade
47C8664	ServeRAID M5200 Series 2 GB Flash/RAID 5 Upgrade
47C8668	ServeRAID M5200 Series 4 GB Flash/RAID 5 Upgrade
Supported RAID adapters M1215	
46C9114	ServeRAID M1215 SAS/SATA Controller
The following upgrade is required to support SEDs with the M1215 RAID controller	
46C9114	ServeRAID M1215 SAS/SATA Controller

For more information about the supported controllers and options, see the following website:

http://www.lenovo.com/us/en/serverproven/

2.1.3 Supported SEDs

The SEDs that are supported at the time of publication are listed in Table 2-3 on page 20. This rapidly growing list of devices should be considered a subset of supported options only. For more information about supported SEDs for a specific server model, see this website:

http://www.lenovo.com/us/en/serverproven/

Table 2-3 Supported SEDs

Option part number	Description
90Y8944	146GB 15K 6Gbps SAS 2.5" SFF G2HS SED
00AJ116	146GB 15K 6Gbps SAS 2.5" G3HS SED
00NA281	300GB 15K 12Gbps SAS 2.5" G3HS 512e SED
00NA286	600GB 15K 12Gbps SAS 2.5" G3HS 512e SED
90Y8913	300GB 10K 6Gbps SAS 2.5" SFF G2HS SED
00AJ106	300GB 10K 6Gbps SAS 2.5" G3HS SED
90Y8908	600GB 10K 6Gbps SAS 2.5" SFF G2HS SED
00AJ101	600GB 10K 6Gbps SAS 2.5" G3HS SED
00NA291	600GB 10K 12Gbps SAS 2.5" G3HS 512e SED
81Y9662	900GB 10K 6Gbps SAS 2.5" SFF G2HS SED
00AJ076	900GB 10K 6Gbps SAS 2.5" G3HS SED
00NA296	900GB 10K 12Gbps SAS 2.5" G3HS 512e SED
00AD085	1.2TB 10K 6Gbps SAS 2.5" G2HS SED
00AJ151	1.2TB 10K 6Gbps SAS 2.5" G3HS SED
00NA301	1.2TB 10K 12Gbps SAS 2.5" G3HS 512e SED
00NA476	1.8TB 10K 6Gbps SAS 2.5" G2HS 512e SED
00NA306	1.8TB 10K 12Gbps SAS 2.5" G3HS 512e SED
00W1533	2TB 7.2K 6Gbps NL SAS 3.5" G2HS SED
00ML218	2TB 7.2K 6Gbps NL SAS 3.5" G2HS 512e SED
00FN238	2TB 7.2K 12Gbps NL SAS 3.5" G2HS 512e SED
00W1543	4TB 7.2K 6Gbps NL SAS 3.5" G2HS SED
00ML223	4TB 7.2K 6Gbps NL SAS 3.5" G2HS 512e SED
00FN248	4TB 7.2K 12Gbps NL SAS 3.5" G2HS 512e SED
00ML228	6TB 7.2K 6Gbps NL SAS 3.5" G2HS 512e SED
00FN258	6TB 7.2K 12Gbps NL SAS 3.5" G2HS 512e SED

Not all drives are supported in all servers. For more information about the supported drives, see this website:

http://www.lenovo.com/us/en/serverproven/

2.1.4 Supported SKLM environments

Support for System x servers was included beginning with IBM Security Key Lifecycle Manager (SKLM) 2.5.0.2. This support requires the base installation of SKLM 2.5 with a minimum of service pack 2 installed, which brings the final version to 2.5.0.2.

Operating system support

The currently supported operating systems (OSs) for SKLM version 2.5 are listed in Table 2-4. In this book, we focus on x86 environments; therefore, those environments are listed first in Table 2-4. For more information, see this website:

`http://www.ibm.com/support/knowledgecenter/api/content/SSWPVP_2.5.0/com.ibm.sklm.doc_2.5/cpt/cpt_ic_release_oview_sw.html`

Table 2-4 SKLM operating system requirements

Operating system	Use DB2 Workgroup Server Edition Version 10.1
Windows Server 2008 R2 (64-bit in 32-bit mode for all Intel and AMD processors), which includes the following editions: ► Standard Edition ► Enterprise Edition	X
Windows Server 2012 (64-bit in 32-bit mode for all Intel and AMD processors) for Standard Edition	X
RedHat Enterprise Linux Version 5.0 Update 6.0 and Version 6.0 Update 3 on x86 64-bit in 32-bit mode	X
SuSE Linux Enterprise Server Version 10 on x86 64–bit mode and Version 11 on x86 64–bit mode	X
Sun Server Solaris 10 (SPARC 64–bit in 32-bit mode). Consider the following points: ► If raw devices are used, apply patch 125100-07 ► SKLM runs in a 32–bit JVM	X
AIX version 6.1 and version 7.1 in 32-bit mode. POWER7 processor-based servers are supported. A 64-bit AIX kernel is required. Use AIX 6.1 Technology Level 2. The minimum C++ runtime level requires the xlC.rte 9.0.0.8 and xlC.aix61.rte 9.0.0.8 (or later) files. These files are included in the June 2008 IBM C++ Runtime Environment Components for AIX package.	X
RedHat Enterprise Linux Version 5.0 Update 6.0, and Version 6.0 Update 3 (System z) on x86 64–bit mode	X
SuSE Linux Enterprise Server Version 11 (System z) on x86 64–bit mode	X

Windows 2008 R2: The web interface of SKLM can be accessed remotely from another system's browser or locally with a browser that is installed on your SKLM server. The default browser that is installed with Windows 2008 R2 is Internet Explorer 8, which must be updated to a newer version to support the SKLM interface. For more information about support, see "Browser requirements" on page 22.

Hardware requirements

The current hardware requirements for SKLM version 2.5 are listed in Table 2-5 on page 22. For more information about updated hardware requirements, see this website:

`http://www.ibm.com/support/knowledgecenter/api/content/SSWPVP_2.5.0/com.ibm.sklm.doc_2.5/cpt/cpt_ic_release_oview_hw.html`

Table 2-5 SKLM hardware requirements

System components	Minimum values[a]	Suggested values[b]
System memory (RAM)	4 GB	4 GB
Processor speed	► Linux and Windows systems: 3.0 GHz dual processors ► AIX and Sun Solaris systems: 1.5 GHz (4-way)	► Linux and Windows systems: 3.0 GHz dual processors ► AIX and Sun Solaris systems: 1.5 GHz (4-way)
Disk space free for SKLM and prerequisite products, such as DB2	5 GB	5 GB
Disk space free in `/tmp` or `C:\temp`	2 GB	2 GB
Disk space free in `/home` directory for DB2	5 GB	6 GB
Disk space free in `/var` directory for DB2	512 MB on Linux and UNIX operating systems	512 MB on Linux and UNIX operating systems

a. Minimum values: These values enable a basic use of SKLM.
b. Recommended values: You must use larger values that are appropriate for your production environment. The most critical requirements are to provide adequate system memory and free disk and swap space. Processor speed is less important.

In addition to the hardware requirements that are listed in Table 2-5, consider the following points:

- All file systems must be writable.
- On Windows OS, the following free space is required in addition to that of your DB2 product:
  - 40 MB in the system drive
  - 60 MB in the `/temp` folder that is specified by the `temp` environment variable
- On Linux and UNIX OSs, you must install your DB2 product in an empty directory. If the directory that you specify as the installation path contains subdirectories or files, your DB2 installation can fail. On Linux and UNIX OSs, 4 GB of free space is required in the `$HOME` directory.
- Installing into mapped network drives or mounted partitions is not supported.
- If installation locations of more than one system component fall on the same Windows drive or UNIX partition, the cumulative space to contain all those components must be available in that drive or partition.

Browser requirements

Supported browsers for SKLM are listed by OS in Table 2-6. For more information about browser support, see this website:

`http://www.ibm.com/support/knowledgecenter/SSWPVP_2.5.0/com.ibm.sklm.doc_2.5/cpt/cpt_ic_release_oview_browserreqs.html?lang=en`

Table 2-6 SKLM browser support

Browser	Fix pack	AIX	Sun Server Solaris SPARC	Windows 2008 R2	Windows 2012	RedHat Enterprise Linux	SUSE Linux Enterprise Server
Microsoft Internet Explorer 9	None	N/A	N/A	X	X	N/A	N/A
Microsoft Internet Explorer 10	None	N/A	N/A	X	X	N/A	N/A
Mozilla Firefox ESR 17	None	X	N/A	X	X	X	X

Firefox ESR

Mozilla Firefox is supported in the Extended Support Release (ESR) version. This installation differs from the usual Firefox. ESR versions are supported and updated for approximately one year, which can help large organizations and software products to keep a version standard for longer than the normal browser release cycle. The update check for the ESR browser provides only security and patches to its ESR version. The browser does not prompt you for a major browser update until a new ESR version is available.

For more information about FIrefox and for the latest Firefox ESR downloads, see this website:

`https://ftp.mozilla.org/pub/mozilla.org/firefox/releases/`

You might find that some software is not yet supported on the latest ESR version. If you are running into issues with the SKLM interface or are looking for a supported ESR version of Firefox, see the following website where all previous Firefox versions are hosted:

`https://ftp.mozilla.org/pub/mozilla.org/firefox/releases/`

With Firefox ESR installed, your About Mozilla Firefox window displays the information that is similar the information that is shown in Figure 2-1.

Figure 2-1 ESR About window

2.2 Example configuration

In this section, we describe the equipment and test configuration that we assembled for writing this book. Although it is representative of a possible deployment, it should not be considered an official reference architecture. The details of the configuration are shown as a point of reference when we are reviewing the sample command lines that were used and details that were entered in data fields as shown in the figures.

2.2.1 Configuration overview

For the purposes of creating this publication, we assembled a test configuration that consisted of several target servers (x3850 X6, x3650 M4 HD, and x3650 M4), a dedicated management network for IMM traffic, a pair of domain name servers, and two virtualized SKLM servers. The SKLM servers on the 192.168.90.x subnet were routed to the 192.168.254.x management network so that the SKLM servers and IMMs can communicate.

Although the SKLM servers can be deployed as physical servers, our recommendation is to create virtual servers where possible. This configuration allows an environment to not have several redundant SKLM servers and use the clustering capabilities of virtualized clusters for high availability and portability.

Figure 2-2 shows a high-level configuration of the environment that was used for testing.

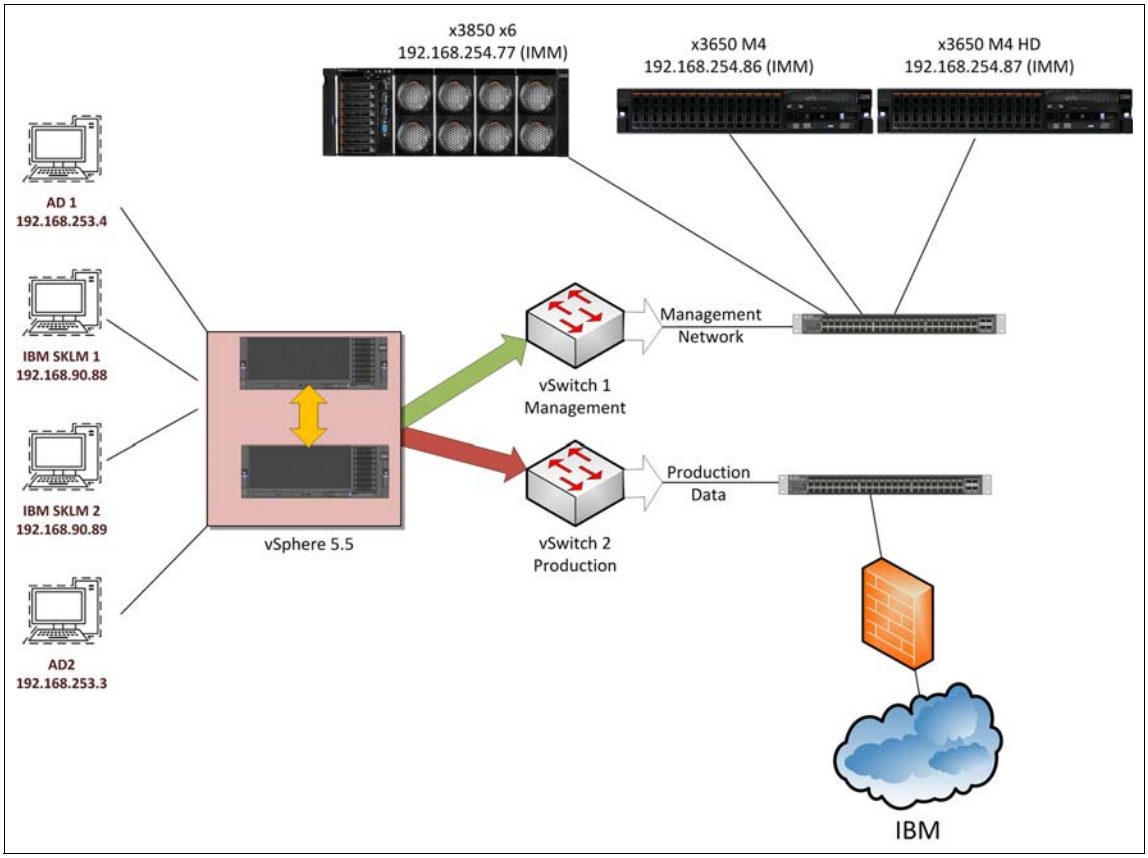

Figure 2-2 Example configuration

As shown in Figure 2-2 on page 24, a simple configuration was assembled as a proof of concept for the creation of the installation procedures for this publication. For simplicity, we

used an existing VMware ESXi cluster that hosted the DNS servers for the lab and created two Windows 2012 virtual servers on which we installed the SKLM servers for the test environment.

The ESXi cluster had access to two networks: one was an internal network to our lab and the other was a connection to the lab network that allowed for remote jump box capability for team members that were not local. This dual network topology is not a requirement for a typical SKLM installation.

The target test systems for this exercise were an x3850 X6, x3650 M4, and x3650 M4 HD, which are all supported in the initial SKLM support announcement. Each of these servers' IMMv2 adapter was connected to the lab network. This connection is critical because all configuration of the IMM and all communication and exchange of security keys is handled over this connection. The removal of the network connection from an IMM results in key encryption keys (KEKs) not being available to the System x server RAID controller at boot time. As a result, the server is unable to unlock the drives and all data is unavailable.

Figure 2-3 shows a minimum configuration that is required to test the basic functions.

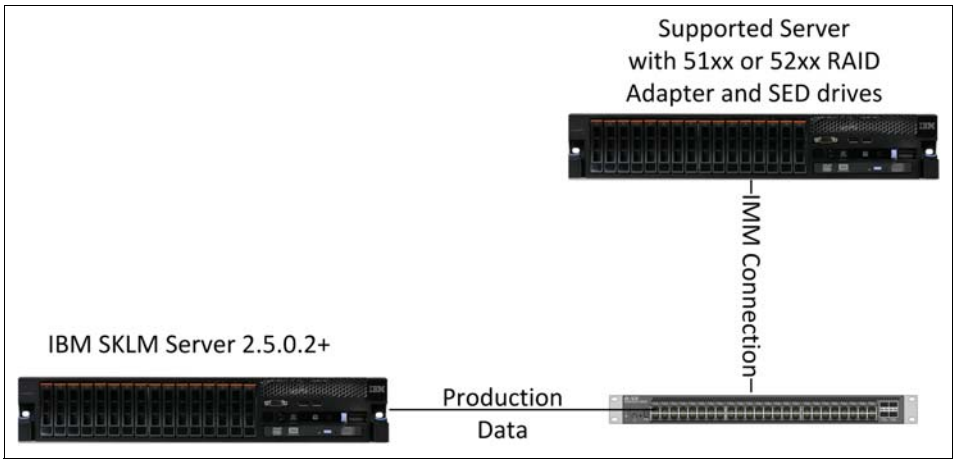

Figure 2-3 Basic configuration

This configuration shows the absolute minimum environment that is required as a proof of concept and does not represent the optimal configuration for a production-based deployment of this solution.

A production deployment must include a minimum of two SKLM servers for redundancy because any loss of communication to an SKLM server results in any server that uses SEDs to not access data if they are restarted.

2.2.2 Configuration

In this section, we provide some specific information about the hardware and software environment that was used in our proof of concept. You also want to check the links that are provided in 2.1, "Supported systems and options" on page 18 for the latest SKLM requirements, and use the latest server, RAID controller, and drive firmware where possible.

Hardware environment

Table 2-7 lists the hardware environment that was used for purposes of this publication, including each server, its RAID controller, Integrated Management Module (IMM) firmware, Unified Extensible Firmware Interface (UEFI), drive type, and firmware levels. The UEFI code

is important because it contains the boot code and the drivers for the RAID controller and SKLM environment.

Table 2-7 Proof of concept hardware details

Server Model	RAID Controller and firmware	IMM Level	UEFI level	SEDs and firmware
System x3650 M4	M5110-e Firmware package version 23.22.0-24, April 24, 2014 ► RAID 5 Upgrade ► Cache offload	1AOO58T, June 8, 2014	VVE142AUS, June 4, 2014	Two 900 GB, 10,000 RPM 6Gbps 2.5-inch SAS SEDs, firmware E56B Other non-encrypting drives also installed.
System x3650 M4 HD	M5210-e, Firmware package version 24.2.1-0027, April 8, 2014 With advanced software options for RAID 5 Upgrade	1AOO58T, June 8, 2014	VVE142BUS, July 2, 2014	12 900 GB, 10,000 RPM 6Gbps 2.5” SAS SEDs, firmware E56B
System x3850 X6	Firmware package version 24.2.1—027, April 8, 2014	1AOO58S, June 2014	A8E112B, August 2014	Four 900 GB, 10,000 RPM 6Gbps 2.5” SAS SEDs, firmware E56B

Figure 2-4 shows an example of the advanced upgrades from our x3650 M4 M5110-e controller, including FoD upgrades. These upgrades can be displayed by interrupting the boot of a System x server at the splash window by pressing F1, then clicking **System Settings** → **Storage** → **Select your desired controller** → **Controller Management** → **Advanced** → **Manage MegaRAID Advanced Software Options**.

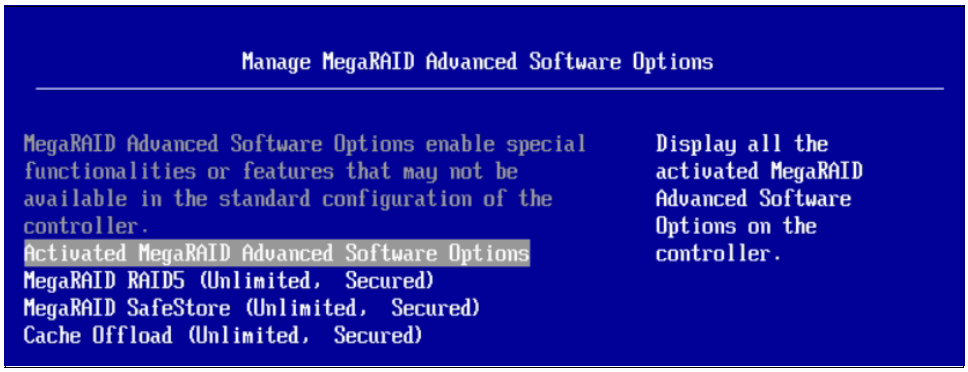

Figure 2-4 MegaRAID advanced software options

Hypervisor and virtual machine environment

For simplicity, flexibility, and high availability, we set up our proof of concept SKLM for this book on a VMware environment. We based our SKLM VMware virtual machine (VM) resources on the normal physical hardware specifications.

The use of a virtualized environment is a good option because you can easily add resources, such as memory and processors later if you encounter performance issues. However, SKLM is not meant to be accessed regularly by many users. Often it is administered by a small team and touched only for necessary tasks, such as adding, removing, or replacing hardware, upgrading the software, or verifying the setup. Other tasks (such as replication) should be automated.

Migrating VMs between hosts with vMotion and setting up high availability with cluster functionality also is crucial to keeping your SKLM servers running whenever a system that is using SEDs needs a key exchange to access its drives while booting. If VMware Distributed Resource Scheduler (DRS) or any other load balancing capacity on a virtualization cluster is used, no two SKLM servers should ever be on the same physical host. This configuration can increase the risk of hardware failure, which results in the loss SKLM access.

Table 2-8 lists the configuration details of the VMs we used for SKLM.

Table 2-8 Proof of concept VM details

Virtual machine	ESXi build	Virtual CPU	Virtual Memory	Disk size
SKLM Master, Windows 2012	5.5, build 1331820	2 total (1 processor with 2 cores each, or 2 processors with 1 core each)	4 GB	100 GB
SKLM Clone #1, Windows 2012	5.5, build 1331820	2 total (1 processor with 2 cores each, or 2 processors with 1 core each)	4 GB	100 GB

Operating system and software environment

Table 2-9 lists the operating systems and SKLM software that were installed and used during the writing of this publication. The Windows 2008 R2 system was used mostly for testing and validating our SKLM work on that OS; most of the tasks and figures that were performed to create this publication were done on the Windows 2012 server. For a simplified setup, we disabled the Enhanced Security Configuration (ESC) in Microsoft Internet Explorer, and disabled the Windows firewalls. Port information for SKLM and its components is supplied in Chapter 3, “IBM Security Key Lifecycle Manager setup” on page 31 to assist you with creating the correct firewall rules to allow the software to function in a production environment, for which we recommend keeping the firewalls enabled.

Table 2-9 Operating System and software details

SKLM server	SKLM version	Operating System	Browser used	Internet Explorer ESC	Windows Firewalls
Master	2.5.0.2 (SKLM 2.5.0.0 with fix pack 2 installed)	Windows 2012	Firefox ESR 17.0.11 Internet Explorer 11 (build 11.0.9600.17239)	Off	Off
Clone	2.5.0.2 (SKLM 2.5.0.0 with fix pack 2 installed)	Windows 2012	Firefox ESR 17.0.11 Internet Explorer 11 (build 11.0.9600.17239)	Off	Off

We connected to the SKLM web interface with Firefox 31 and Firefox ESR 24 without any issues, but left those browsers out of the table because they are not explicitly supported browser versions.

IMM connection considerations

To connect to each System x3650 M4 and System x3850 X6 system, we mostly used Firefox ESR 24 and Internet Explorer 11 with Java version 7 update 60 to use the IMM remote control feature for the configuring drives and RAID controllers.

2.3 Conclusion

In this chapter, we described the current list of supported servers, RAID adapters, drives, and software that are instrumental in creating a working solution.

We also described the environment that was used to create the materials for this publication in addition to serving as a template for a proof of concept system should you want to reproduce our configuration in your own environment.

Part 1

Hands-on configuration

In this part, we describe the steps to implement the lab setup that was introduced in Chapter 2, “Supported systems and sample configuration” on page 17.

We describe the configuration of IBM Security Key Lifecycle Manager, the Integrated Management Console on System x servers, and the Unified Extensible Firmware Interface.

© Copyright Lenovo 2015. All rights reserved.

3

IBM Security Key Lifecycle Manager setup

In this chapter, we describe a basic installation of IBM Security Key Lifecycle Manager (SKLM) on Windows Server 2012. For more information about supported operating systems, see the SKLM product documentation, which can be found at the IBM Knowledge Center at this website:

http://www.ibm.com/support/knowledgecenter/SSWPVP/welcome

This chapter includes the following topics:

- Acquiring installation files
- SKLM installation
- Validate SKLM installation
- Apply SKLM licensing
- Generating an SKLM server certificate
- Production environment considerations

© Copyright Lenovo 2015. All rights reserved.

3.1 Acquiring installation files

In this section, we describe the files that are necessary for a successful SKLM installation. For more information about purchasing and acquiring SKLM, see Appendix C, “Licenses and software” on page 177.

3.1.1 Operating system packages

Some other operating system (OS) packages or features might be needed for components of the SKLM installation. Installing more features and packages might require an Internet connection or the OS installation media.

Our Windows 2012 and Windows 2008 R2 proof of concept environments required the installation of the .NET feature and the basic OS installation.

3.1.2 SKLM installation package

SKLM is delivered in a package that is approximately 4 GB. The SKLM package includes the following IBM software components that are needed for a complete SKLM environment:

- IBM SKLM
- IBM DB2
- IBM WebSphere Application Server

Important: SKLM modifies WebSphere Application Server during the installation process. Therefore, you must not install SKLM into a WebSphere Application Server instance that another product uses. Also, do not install SKLM into a WebSphere Application Server instance that is provided by another product or you might encounter issues.

For more information about validating your files before you proceed with the installation, see 3.2.3, “Validating SKLM Windows installation files” on page 46.

3.1.3 Acquiring SKLM updates

SKLM version 2.5.0.0 requires fix pack 2 or newer to integrate Lenovo System x server support into the user interface. As described in this chapter, the installation of fix pack 2 brings your version to 2.5.0.2.

To acquire fixes and updates for SKLM, you use the IBM Support Portal. We recommend that you acquire the fix pack before starting the installation process to have a fully updated instance of SKLM at the completion of this chapter. The SKLM fix pack download that was available at the time of our installation was approximately 250 MB. You should install the fix pack after completing the SKLM base installation. Then, the fix pack file must be available to the system on which you are installing SKLM.

Locating and downloading updates by using the IBM Support Portal

Complete the following steps to acquire the latest SKLM fix pack:

1. Browse to the BM Support Portal, which is available at this website:

   `http://www.ibm.com/support`

2. If you have an IBM id, log on as shown in Figure 3-1. If you do not have an IBM id, click **Create IBM id**. Creating an account is free, and signing in is required to retrieve the fixes.

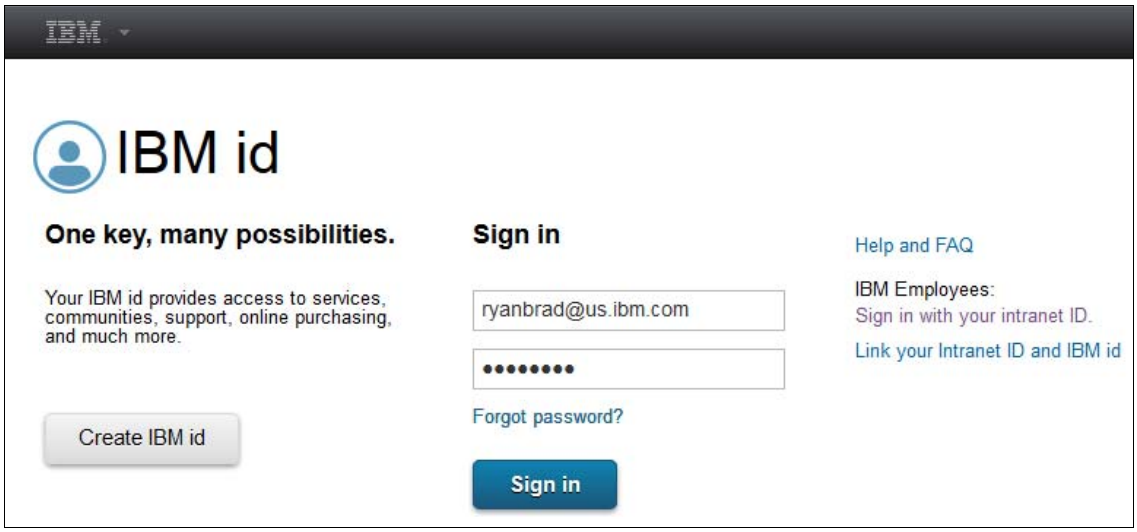

Figure 3-1 IBM Support Portal login or id creation

3. As shown in Figure 3-2, enter `SKLM` or begin entering `Security Key Lifecycle Manager` in the Product lookup field to locate the product. Select **Security Key Lifecycle Manager** to begin the update acquisition process.

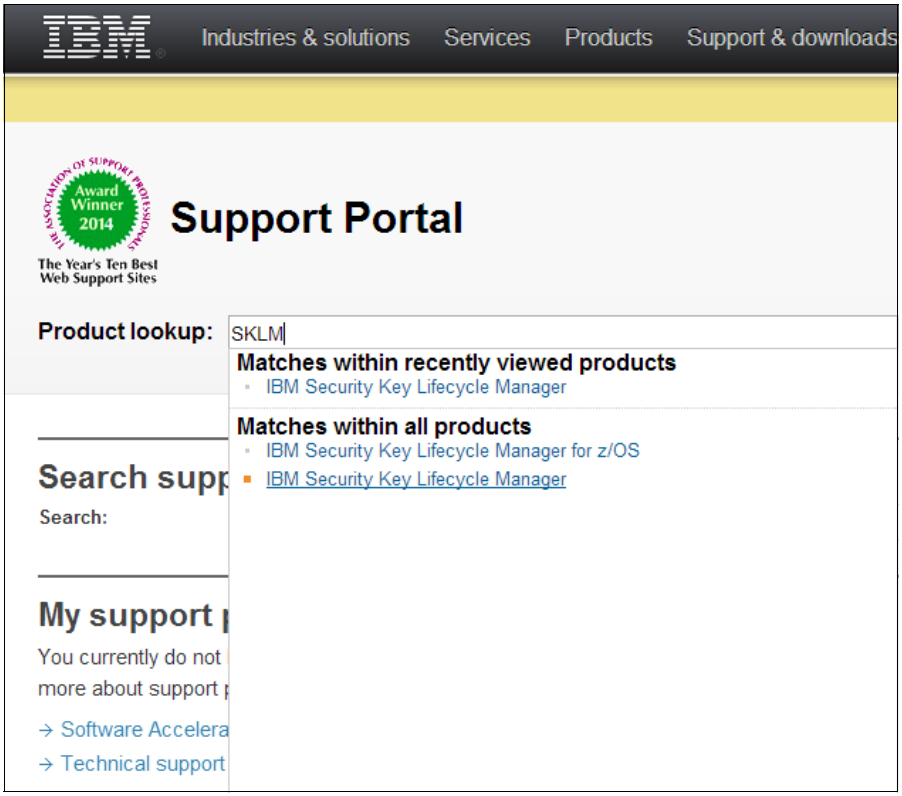

Figure 3-2 SKLM product lookup

4. When prompted to narrow your search, we recommend that instead you leave the options cleared and click **Go**, as shown in Figure 3-3. It is better to list all fixes and choose your wanted version because you might inadvertently limit a wanted option.

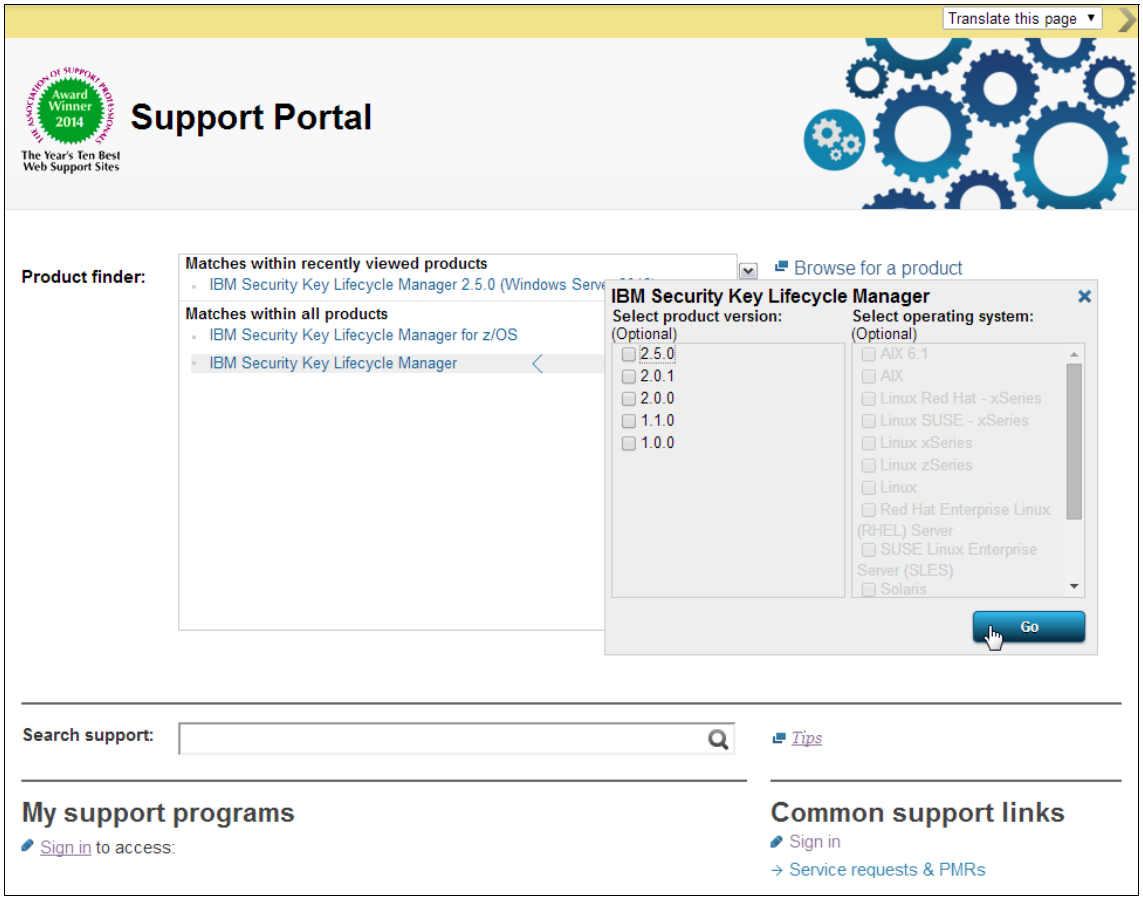

Figure 3-3 Product lookup

5. The support page reloads and shows your selection of SKLM. In our case, the following choice is listed first:

   `IBM Fix Central – 2.5.0-ISS-SKLM-FP0002.README.html`

   This choice contains the latest fix pack that we want. However, we recommend getting a complete list. To get this list, select **Downloads (fixes & PTFs)**, as shown in Figure 3-4.

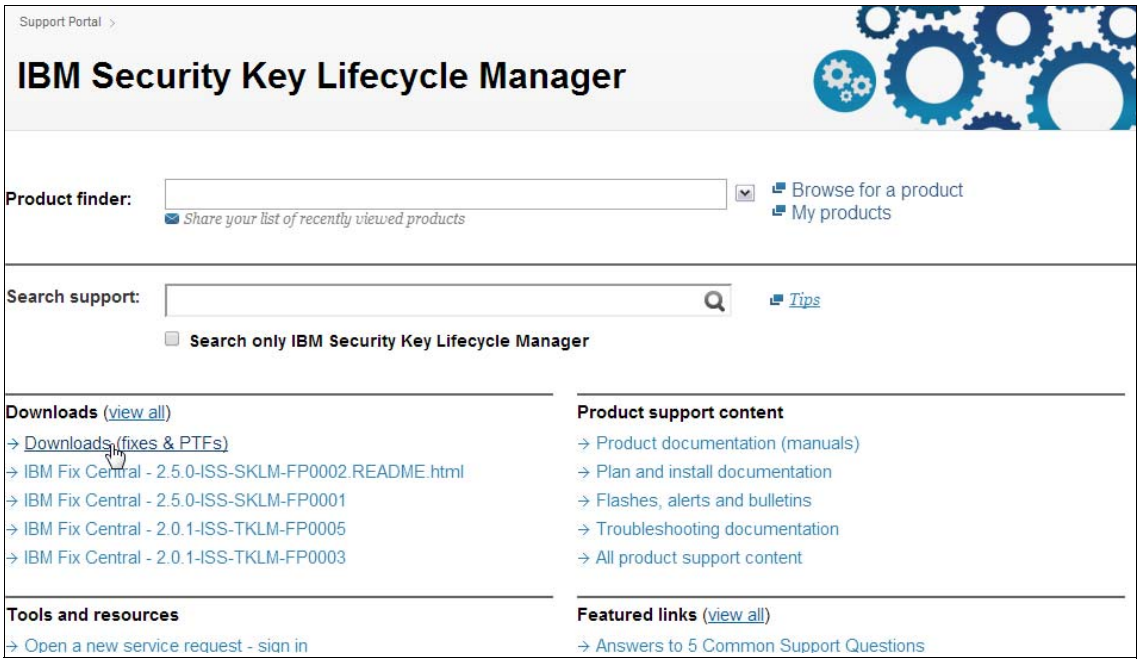

Figure 3-4 Support Portal downloads link

6. We do not recommend limiting your options by version level. When prompted to refine your list, select **All** for the version to get a complete picture, and then select a specific OS, if wanted. Currently, SKLM fix packs are bundled and your download includes all supported operating systems. Select **Continue** after you make your selections in the Refine my fix list window, as shown in Figure 3-5.

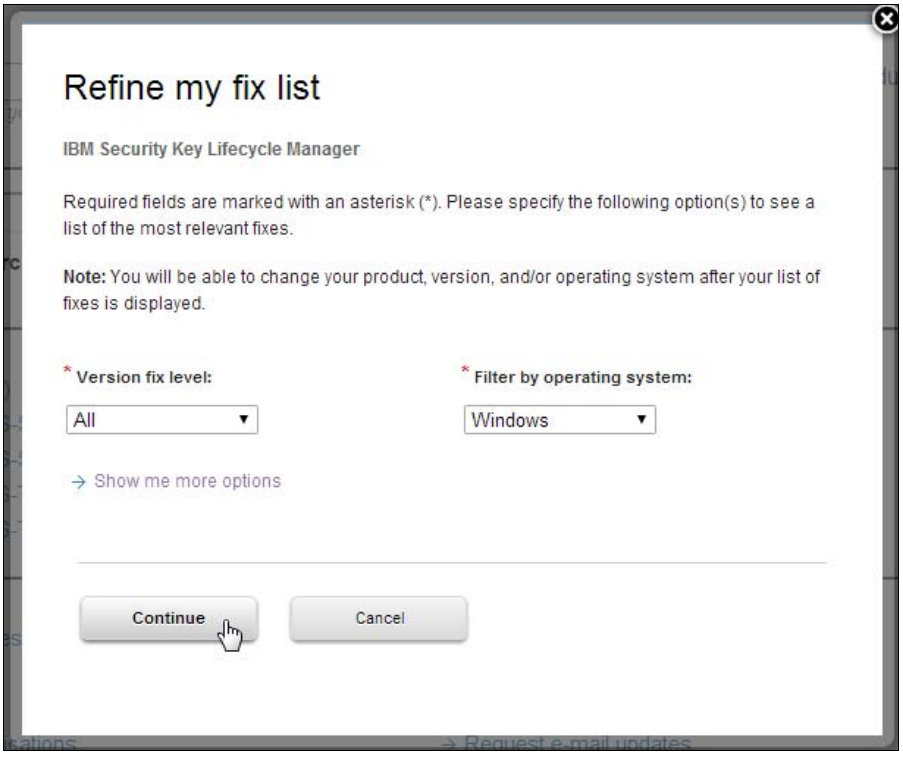

Figure 3-5 Download refinement options

7. A list of fixes is displayed. If no results are shown in this window, it is likely that your search was too narrow. For instance, if you are looking for the upgrade to version 2.5.0.1, you must select version 2.5.0.0 as the installed version on which you are searching. Select your wanted fix pack and click **Continue**, as show in Figure 3-6.

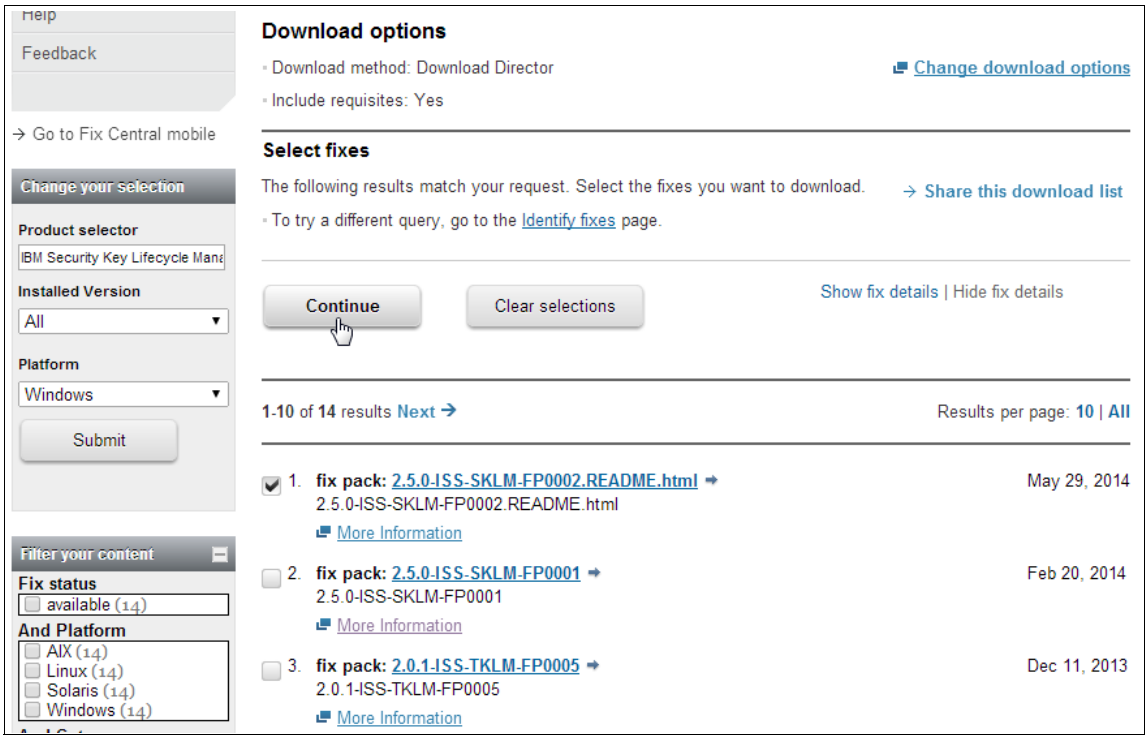

Figure 3-6 Selecting the fix pack

8. If you signed in with your IBM id, you are taken directly to the download page. In this example, we use the Download Director Java applet as our selected method for download. Select **Download now** (as shown in Figure 3-7) to begin the download process.

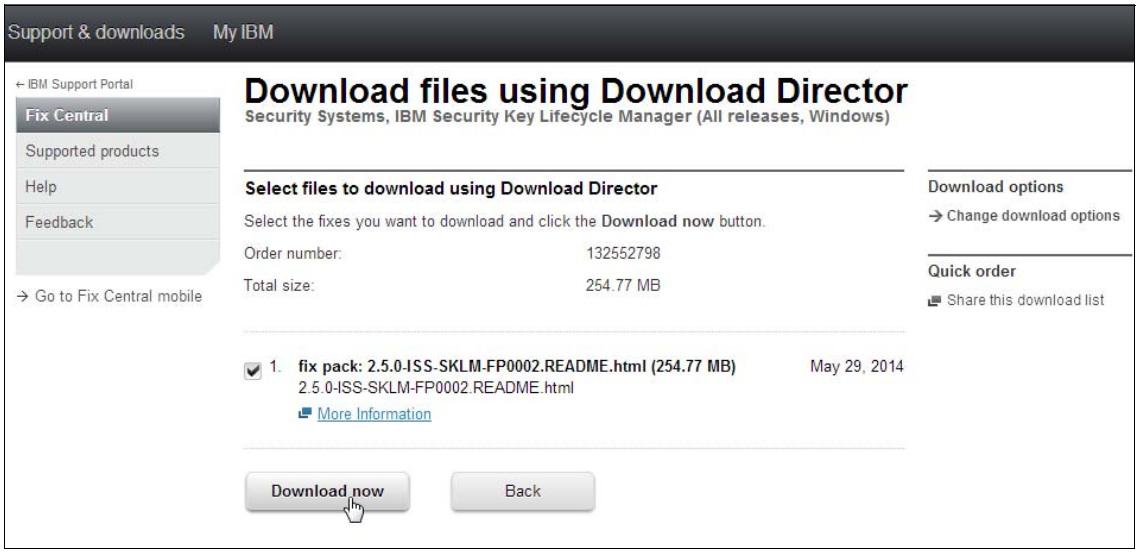

Figure 3-7 Downloading the fix pack

The details of your download show that fix packs for all support OS types are being downloaded, as shown in Figure 3-8.

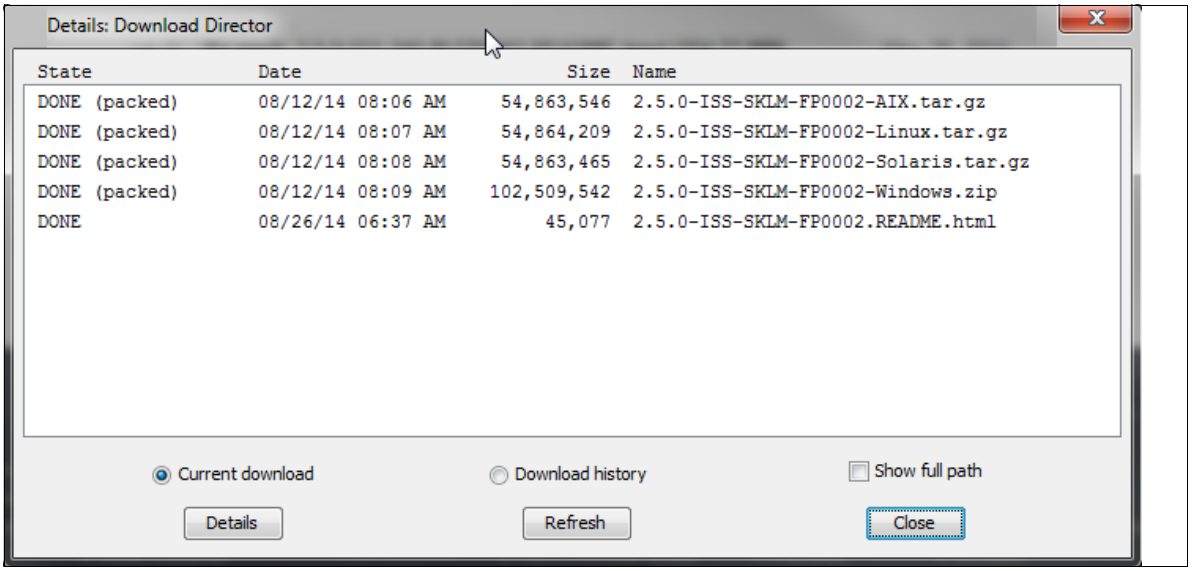

Figure 3-8 Fix pack files

9. When your download completes, proceed to the SKLM installation as described in 3.2, “SKLM installation” on page 39. Later, you copy and install the appropriate fix pack to your SKLM server.

Locating and downloading updates using the Support Portal browse

Instead of the use of the product search as described in “Locating and downloading updates by using the IBM Support Portal” on page 32, you might prefer to browse for your product fixes in the IBM Support Portal or IBM Fix Central.

If you prefer to browse for your fixes in the IBM Support Portal, an example of the selections you must make is shown in Figure 3-9.

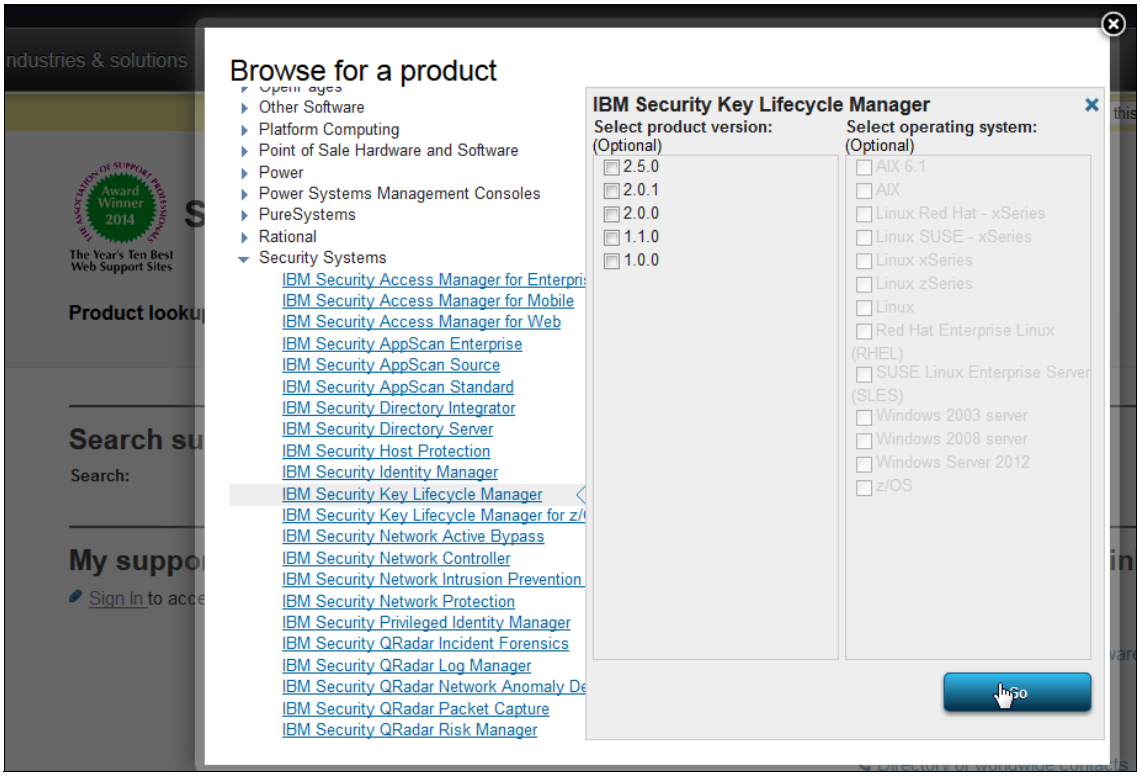

Figure 3-9 Browse support for SKLM fixes

Alternatively, if you prefer IBM Fix Central, the selections are shown in Figure 3-10.

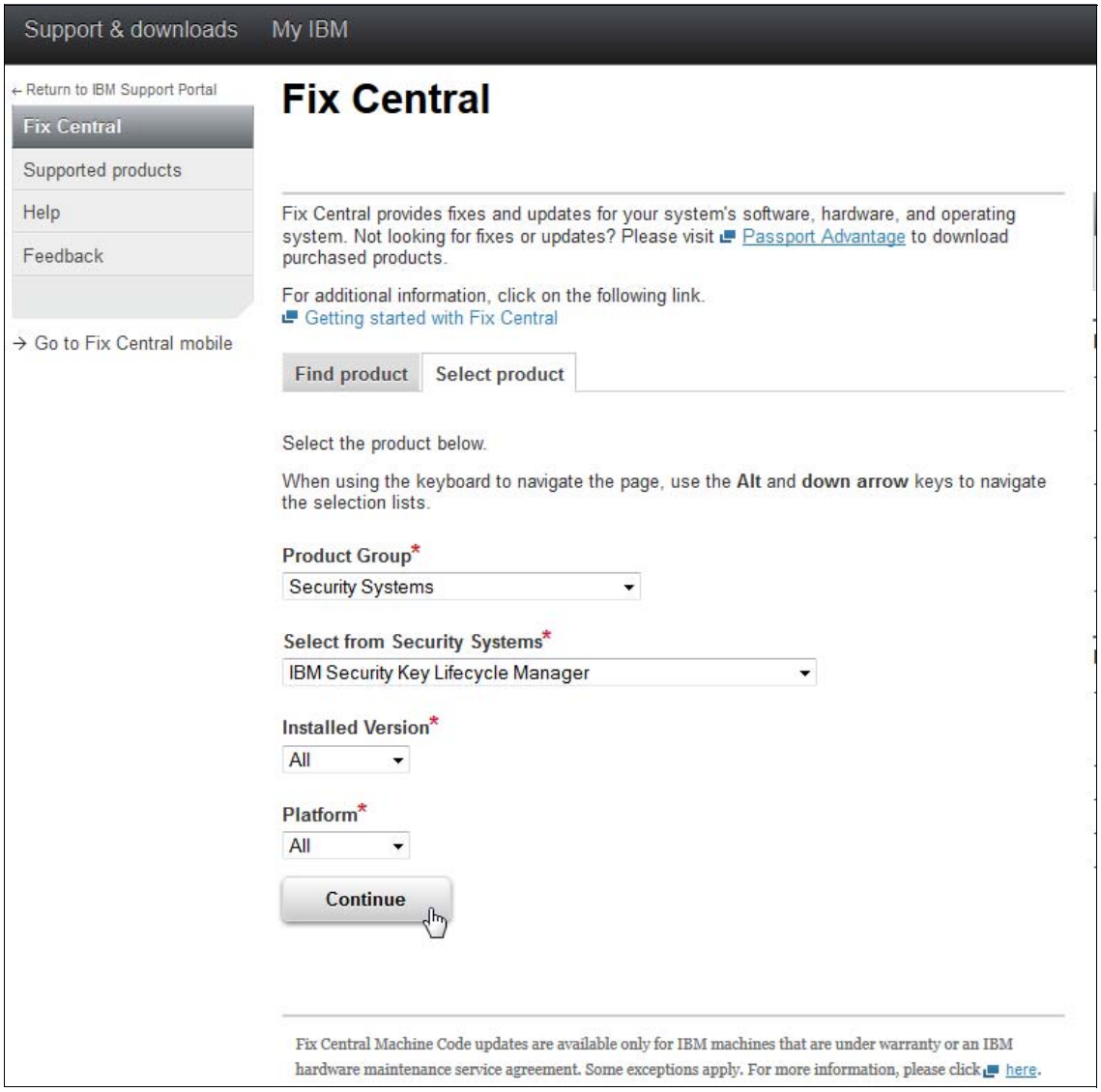

Figure 3-10 SKLM selections in Fix Central

3.2 SKLM installation

In this section, we describe the following steps for a basic setup of SKLM:

- Operating system firewall and setting considerations
- Installing prerequisites
- Validating SKLM Windows installation files
- Running the installation and dynamic updates
- Updating SKLM with the latest fix pack

3.2.1 Operating system firewall and setting considerations

Before starting the SKLM installation, some default OS and firewall settings must be changed. In this section, we describe some of those settings. For more information, see this website:

http://www.ibm.com/support/knowledgecenter/SSWPVP_2.5.0/com.ibm.sklm.doc_2.5/cpt/cpt_insguide_tklm_postinstall_processesrunning.html?lang=en

Windows considerations

Default Windows firewall settings do not allow remote connections to all SKLM component interfaces. To expedite your installation and validation, disable the Windows firewall temporarily. You should enable all Windows firewalls and create firewall rules for SKLM if you are setting up a production system that connects to the Internet. Table 3-1 lists the ports that must be granted access for an SKLM environment on Windows.

Table 3-1 Default ports that are required for Windows

Component	Required ports
SKLM HTTPS access to UI and REST services	9080
WebSphere Application Server integrated console HTTPS access	9083
DB2	50010
SSL port listening for KMIP messages at installation time	5696
SSL port for device messages	441

Linux considerations

For Linux installations, Security-Enhanced Linux (SELinux) must be disabled to allow the installer to make system changes.

By default, SKLM and its components use the ports that are listed in Table 3-2 when it is running on Linux or AIX.

Table 3-2 Default ports that are required for Linux and AIX

Component	Required ports
SKLM	9080 - 9099
DB2	50010

3.2.2 Installing prerequisites

In this section, we describe the prerequisites that we completed before beginning the SKLM installation on Windows Server 2012. You might need an Internet connection or the installation media for your OS to complete this section. For more information about installations on any other OS, see the SKLM installation guide that is available at this website:

http://www.ibm.com/support/knowledgecenter/#!/SSWPVP_2.5.0/com.ibm.sklm.doc_2.5/top/landing-install.html

Operating system installation

These prerequisite and SKLM installation instructions are intended to be run after a system or virtual machine or virtual server is loaded with a supported OS. In our case, they pertain to an installation on Windows Server 2012, which is a 64-bit OS.

Linux prerequisites

On Linux operating systems, SKLM requires the `compat-libstdc++` package, which contains `libstdc++.so.6`. It also requires the `libaio` package, which contains the asynchronous library that is required for DB2 database servers.

To determine whether you have the `libstdc` package available, run the following command:

```
rpm -qa  | grep -i "libstdc"
```

If the package is not installed, locate the RPM file on your original installation media and install it by using the following commands:

```
find installation_media -name compat-libstdc++*
rpm -ivh full_path_to_compat-libstdc++_rpm_file
```

To determine whether you have the `libaio` package, run the following command:

```
rpm -qa  | grep -i "libaio"
```

If the package is not installed, locate the RPM file on your original installation media and install it by using the following commands:

```
find installation_media -name libaio*
rpm -ivh full_path_to_libaio_rpm_file
```

On Red Hat Enterprise Linux 64-bit systems, DB2 installation requires that two separate `libaio` packages must be installed before **`db2setup`** is run. These packages are both named `libaio`. However, there are two different RPM files to install, one of which is an i386 RPM file; the other is an x86_64 RPM file.

Windows prerequisites

On Windows operating systems, SKLM uses the .NET Framework. In this section, we describe the installation of this prerequisite feature.

.NET Framework installation

Complete the following steps to install the .NET Framework to avoid warnings during the SKLM installation process and issues with SKLM and its components during use:

1. Open the Windows Server Manager Dashboard and select **Add roles and features**, as shown in Figure 3-11.

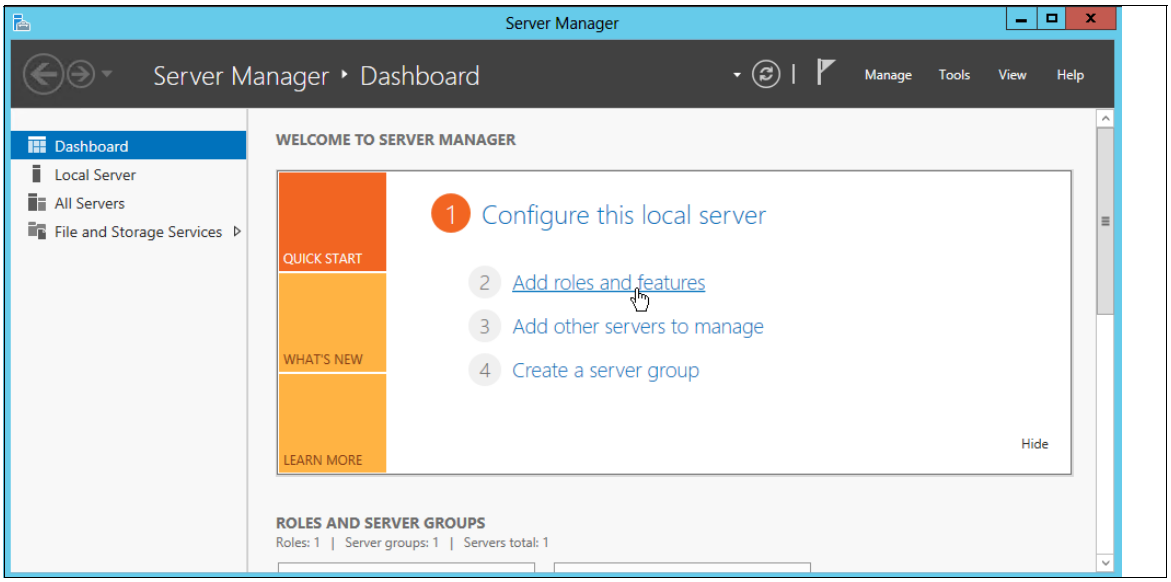

Figure 3-11 Windows Server Manager Dashboard

2. Click **Next** in the Before You Begin window.
3. Leave Role-based or feature-based installation selected by default, and then click **Next** in the Installation Type window, as shown in Figure 3-12.

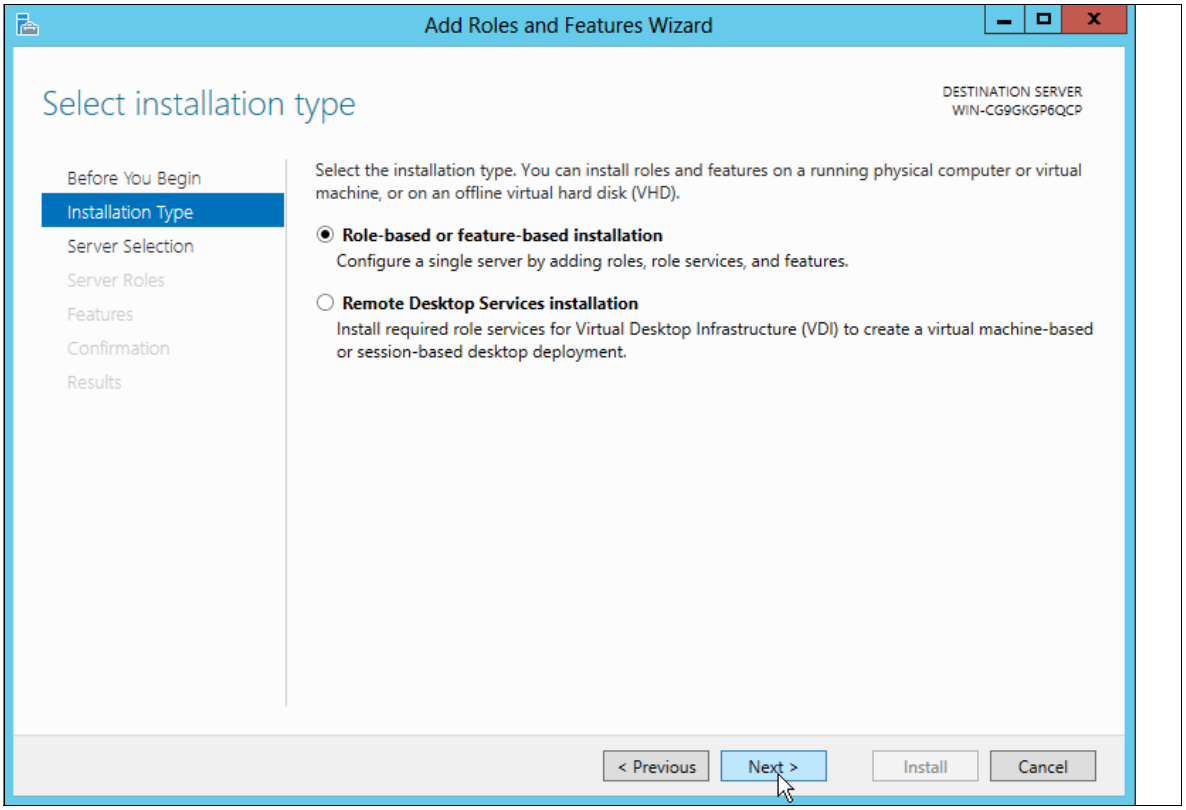

Figure 3-12 Installation type window

4. Select the Windows instance on which you are going to set up SKLM. Click **Next** in the Server Selection window, as shown in Figure 3-13.

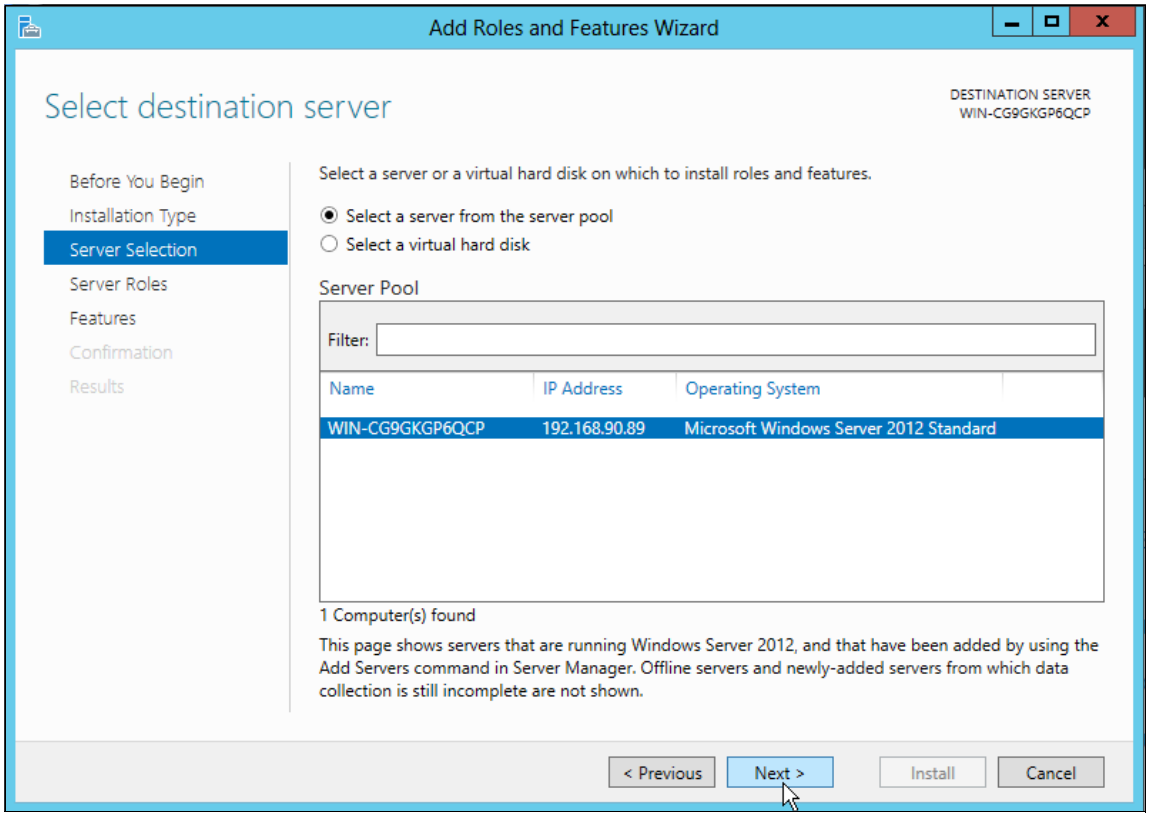

Figure 3-13 Server selection window

5. Click **Next** in the Server Roles window without making any selections.
6. Select **.Net Framework 3.5 Features** in the Features window and then click **Next**, as show in Figure 3-14.

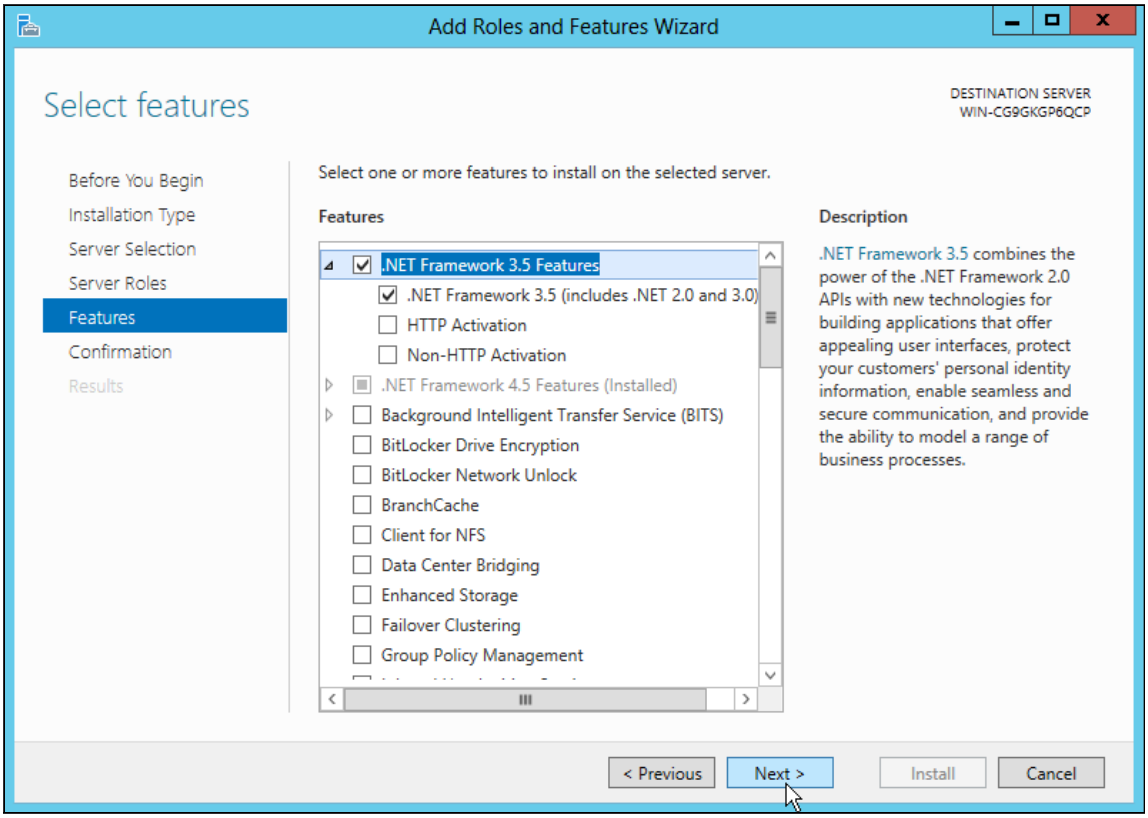

Figure 3-14 Feature selection window

7. In the Confirmation window, you might need to specify an alternative source path to point to your installation media if you do not have an Internet connection. Because we have an Internet connection in our example, we confirm the .NET Framework selection and click **Install,** as shown in Figure 3-15.

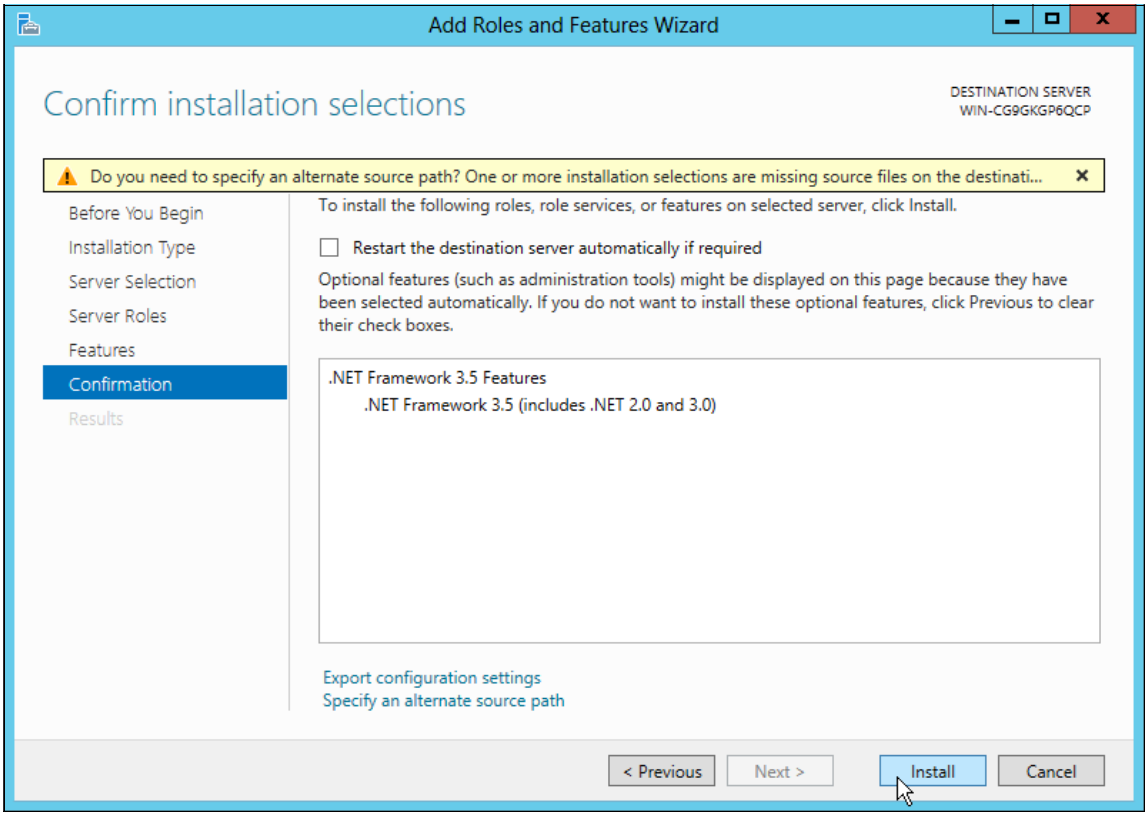

Figure 3-15 Confirmation window

8. Your results can vary based on several factors, including Internet connection speed. In our example, the .NET installation required less than 10 minutes. Upon successful completion, click **Close** in the Results window, as shown in Figure 3-16.

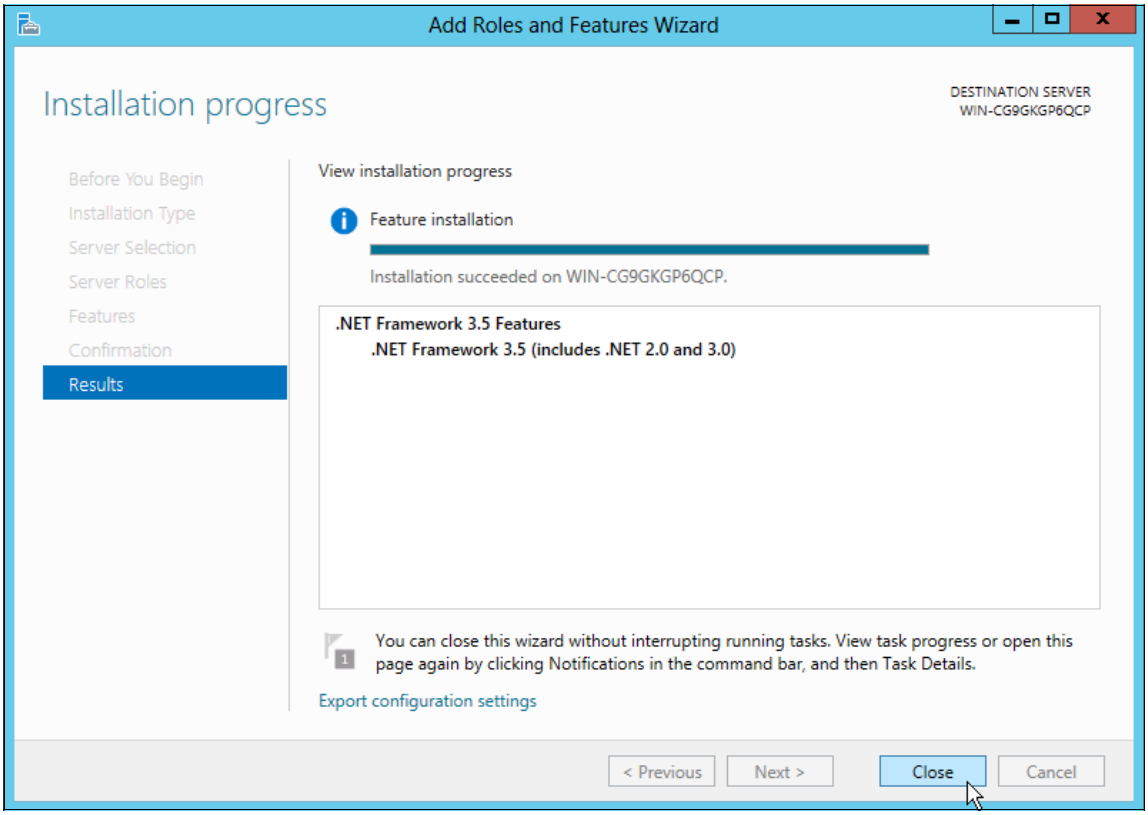

Figure 3-16 Installation results window

Your system is now ready to proceed with the SKLM installation file validation.

3.2.3 Validating SKLM Windows installation files

In this section, we describe how to ensure that you have the correct installation files for SKLM. Although your files can vary slightly by version or the package in which they were delivered, this section gives you an idea of the files that contained within the installation package.

Complete the following steps:

1. Copy the SKLM installation package to the file system on which you installed SKLM. Our installation package for Windows version 2.5.0.0 was approximately 4 GB (compressed and extracted).
2. Extract your SKLM installation files and validate that the size and file structure is correct. Our `.zip` package file name is `SKLM_2.5_WIN64_ML`; you might have an eAssembly for SKLM 2.5 that is named `CIRX2ML`. If the package is a `.tar` file instead of a `.zip`, you want to use a third-party tool (such as 7-zip) that can extract `.tar` files in Windows.
3. After the files are extracted, browse to the SKLM directory and open the `disk1` subdirectory. Figure 3-17 shows an example of the installation package file structure within the `disk1` directory.

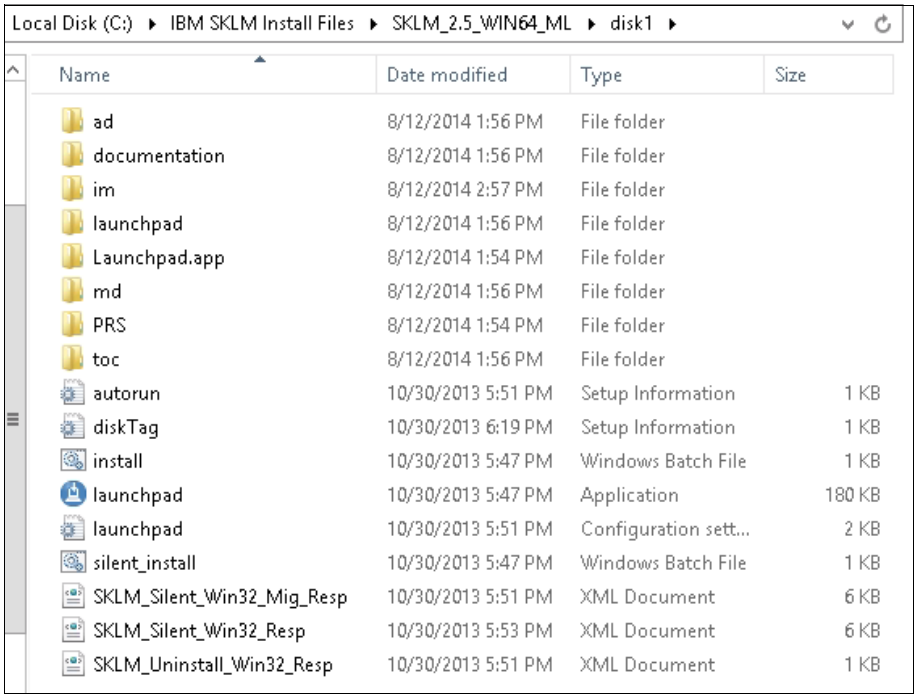

Figure 3-17 SKLM installation package file structure

4. After you validate that your SKLM installation files are correct, you can proceed to the installation process.

3.2.4 Running the installation and dynamic updates

In this section, we describe the steps that are used to install SKLM on Windows Server 2012 and perform component updates with the installation. The update process requires an active Internet connection. If you do not have an active Internet connection, ignore the options for updates during the installation process.

Complete the following steps:

1. Locate the `Launchpad` executable file in the `disk1` subdirectory of your SKLM installation package. Right-click the file and select **Run as administrator**, as show in Figure 3-18.

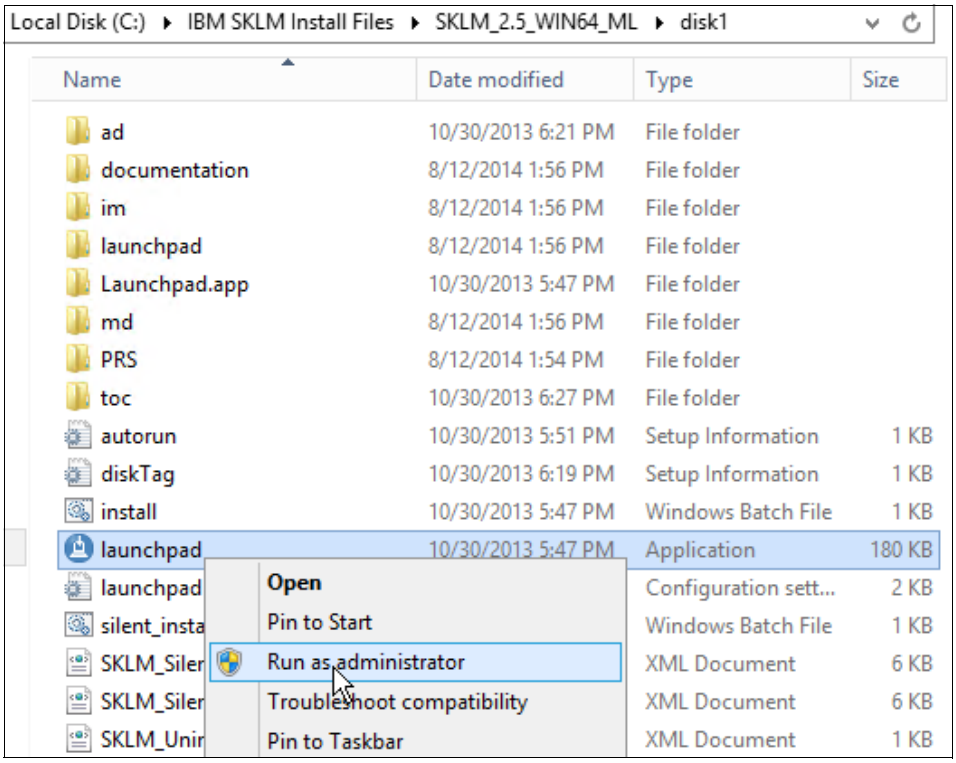

Figure 3-18 Start SKLM installation wizard

2. If you want to change the language from English, select your preferred language in the launchpad window and click **OK**.

3. Under Product Overview, select **Install IBM Security Key Lifecycle Manager** to begin the installation process, as shown in Figure 3-19.

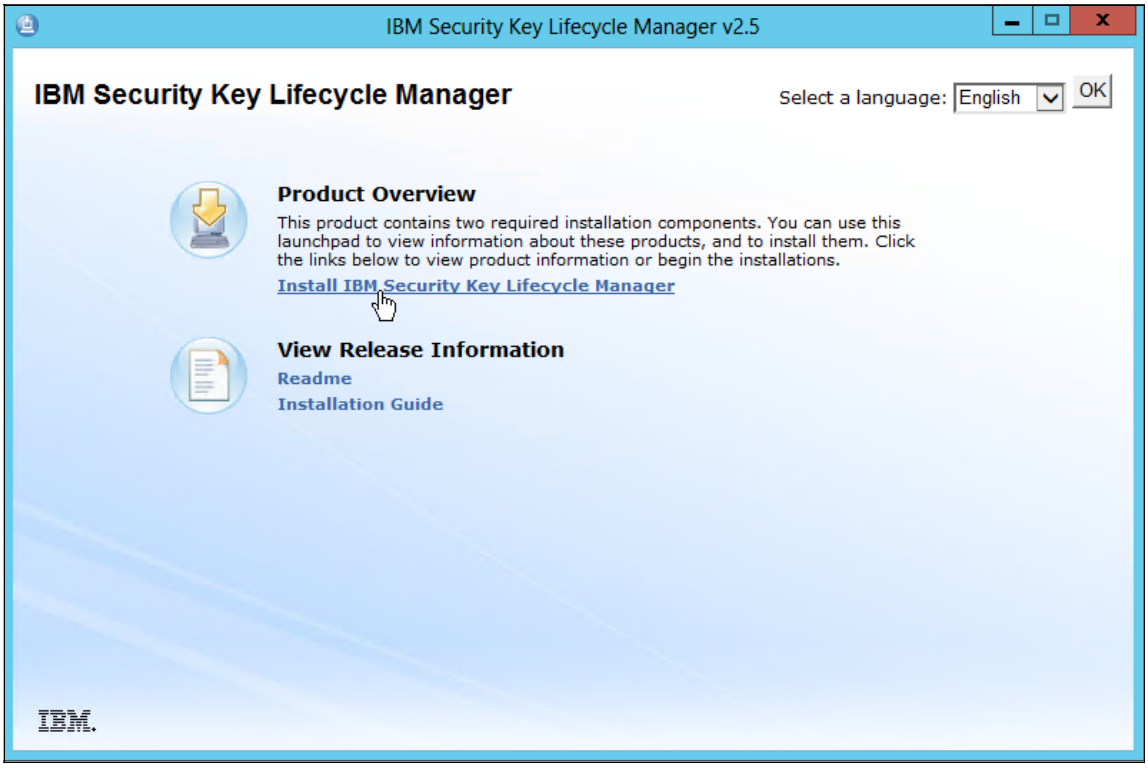

Figure 3-19 SKLM installer language selection and installation start

4. Figure 3-20 shows an expanded view of the Install Packages window. If your system is connected to the Internet and you want to get the latest updates for the components of your SKLM installation, click **Check for Other Versions, Fixes, and Extensions**.

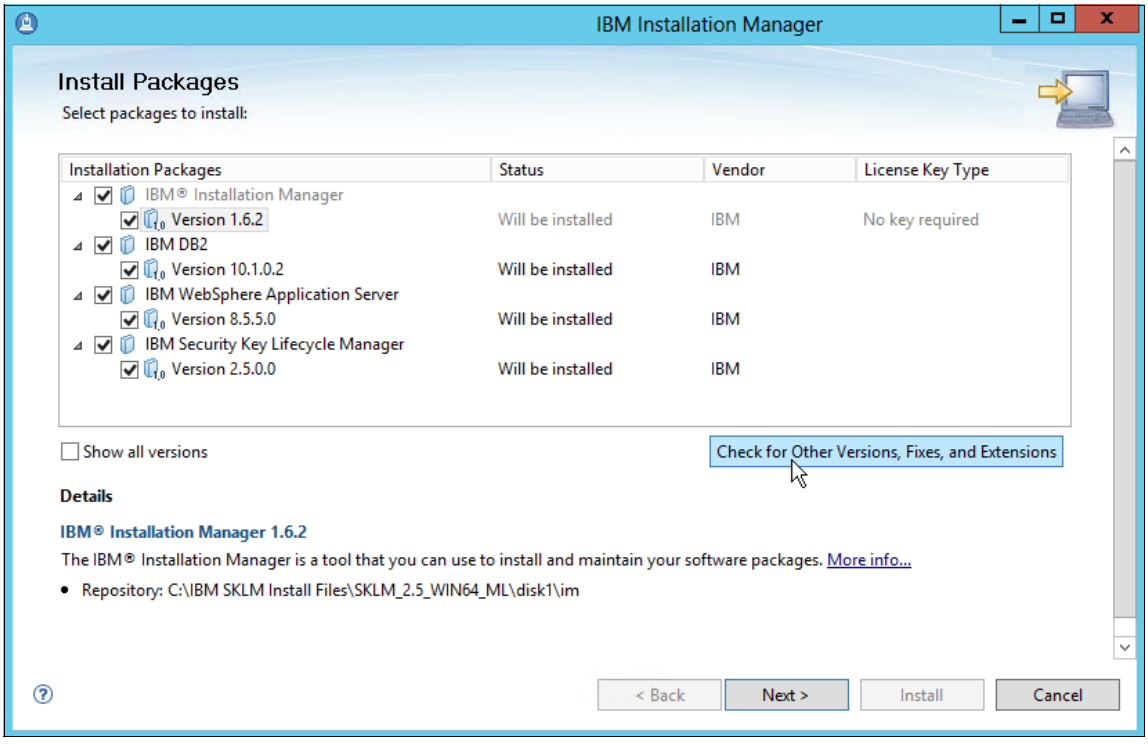

Figure 3-20 Optional update check for SKLM components

5. After the option to check for other versions is selected, a progress window displays the Operation in progress. Then, you are prompted to enter your IBM ID login information. Enter your information and lick **OK**, as shown in Figure 3-21.

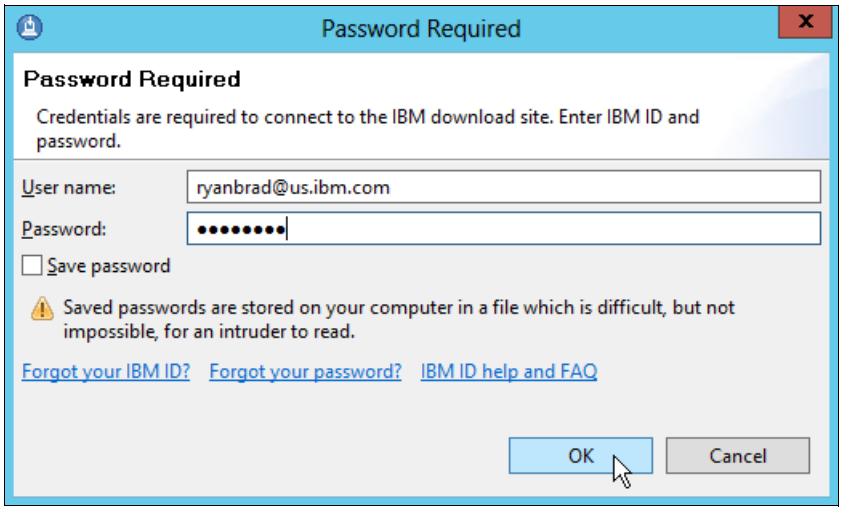

Figure 3-21 IBM ID login for downloads

6. Upon successful login, a Search Result window displays a notification that other versions were found, as shown in Figure 3-22.

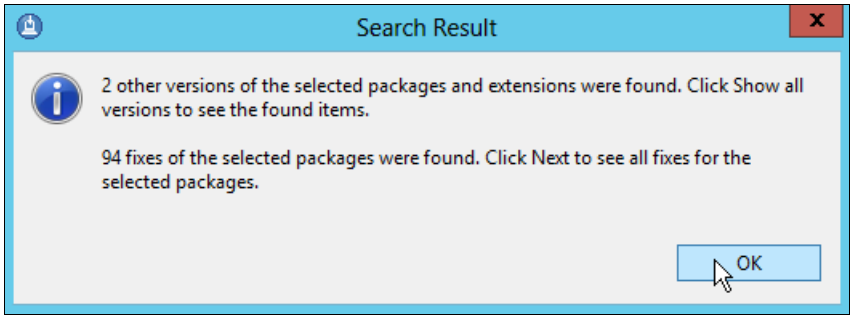

Figure 3-22 Found fixes notification

7. Select the **Show all versions** option (as shown in Figure 3-23) to list any new updates that were found (only the latest versions are selected). Leave those selections at default; you want the latest code to be installed here or fixes can conflict.

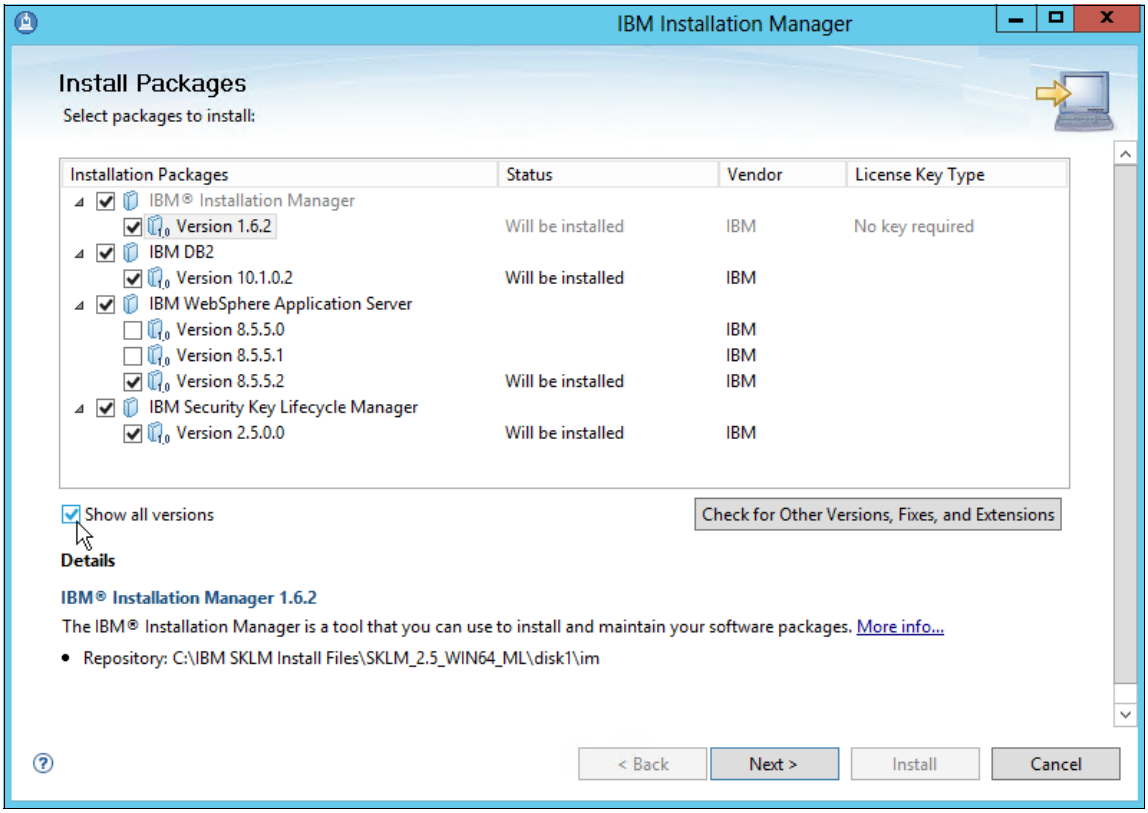

Figure 3-23 Install package selection

8. Click **Next** when you are ready to proceed. A new progress window opens as the installer collects information and prepares for installation.
9. The next window shows fixes and updates that might be applicable. All of these fixes and updates might not be applicable to your installation and some might result in errors if they are selected. Figure 3-24 shows the fixes that we selected for IBM WebSphere Application Server; in our case, the latest version 8.5.5.2.

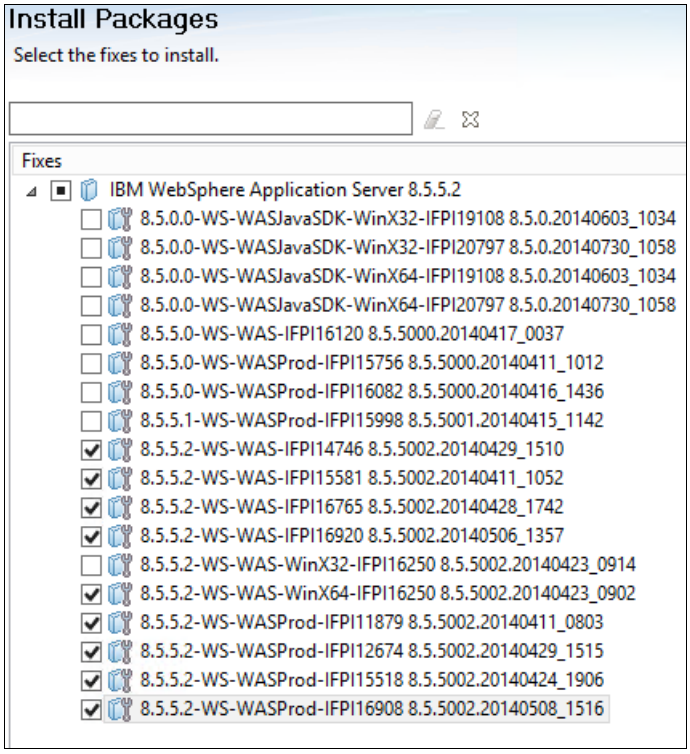

Figure 3-24 Install package selection for 64-bit Windows

We selected the fixes that are applicable to our OS architecture only. Selecting any packages that are denoted with WinX32 on a 64-bit OS, such as Windows Server 2012, results in errors that halt the installation.

10. During our proof of concept, the installation did not find any available updates for DB2. SKLM-related fix packs must be installed after the base software is installed. After the applicable fixes for your installation are selected, click **Next**.
11. Read and accept the license agreement. Click **Next**.
12. In the next window, you can change the installation path for the resources that are shared between the IBM components of SKLM (Installation Manager, SKLM, WebSphere Application Server, and DB2) and the installation path for IBM Installation Manager. We use IBM Installation Manager later to manage and install updates (we kept the default paths). Later in this process, we ran the Installation Manager and command-line instructions with administrator privileges in Windows as prompted in this window. Click **Next** when you are satisfied with the installation paths. Our setup is shown in Figure 3-25.

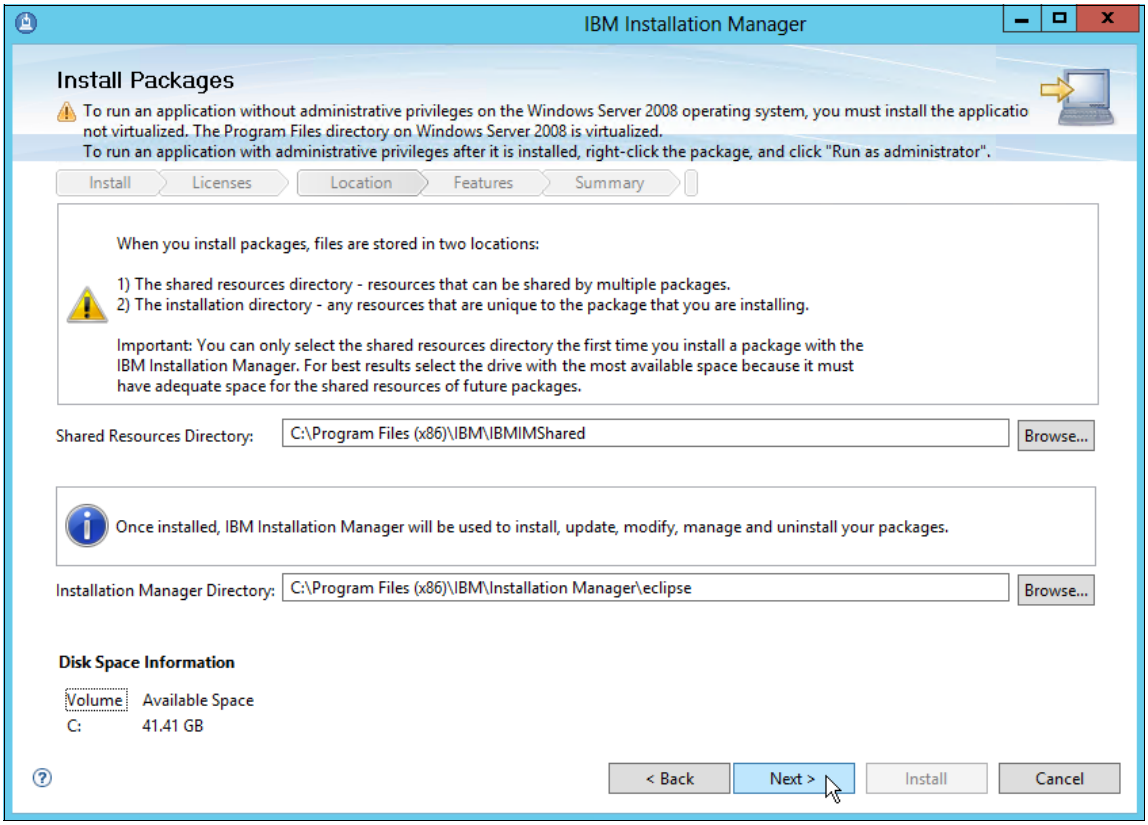

Figure 3-25 First installation directory window

13. The window that is shown in Figure 3-26 provides the option to change where IBM Installation Manager and the other SKLM components are installed. As shown in Figure 3-26, you can highlight the root level of the Package Group Name tree and modify the installation directory. We accepted all of the defaults in this step.

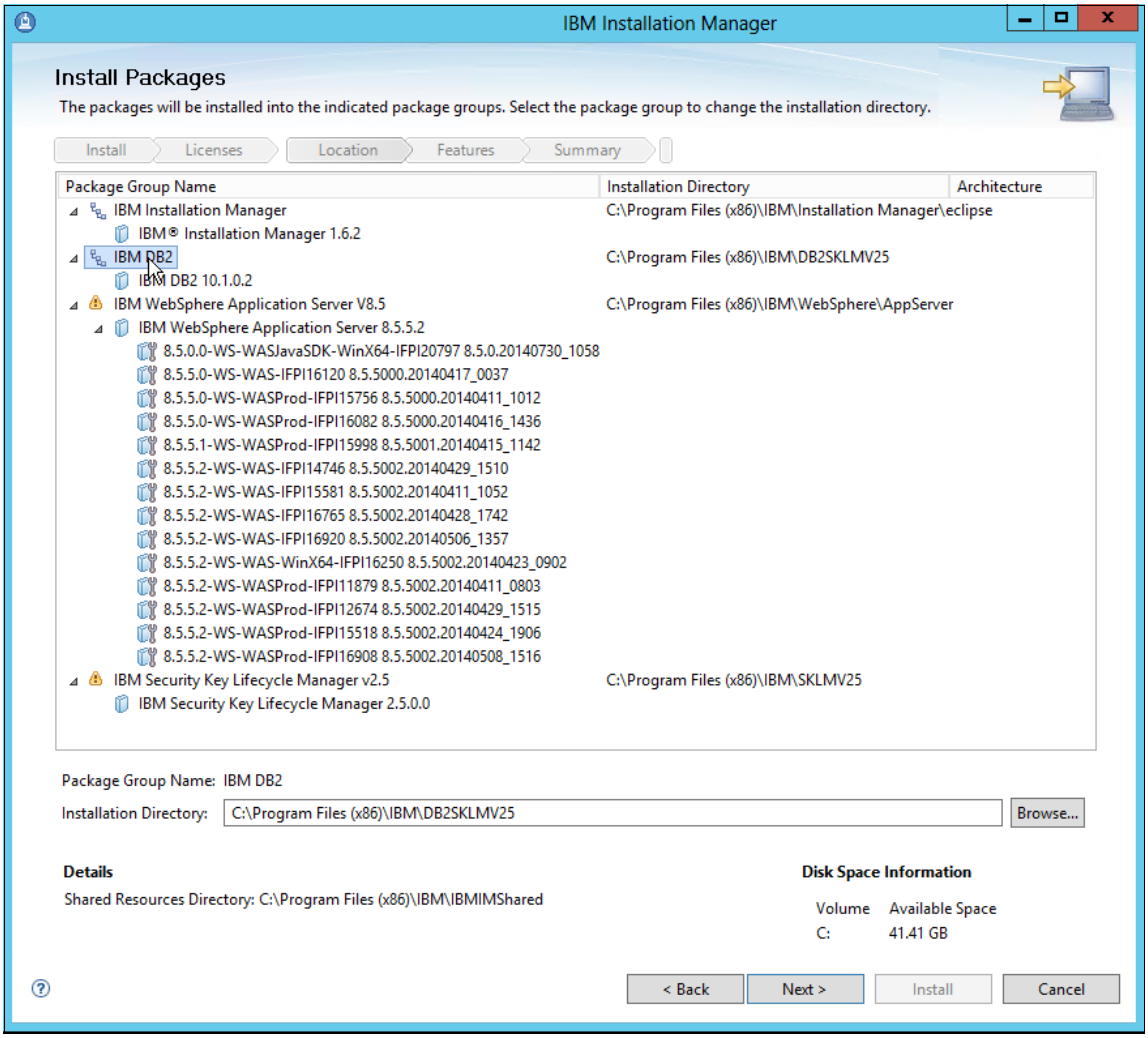

Figure 3-26 Second installation directory window in which DB2 directory is highlighted

14. When you are satisfied with the installation directories, click **Next**.
15. As shown in Figure 3-27, you can select any language translations that you want to install. All text in all components might not support translation. When complete, click **Next**.

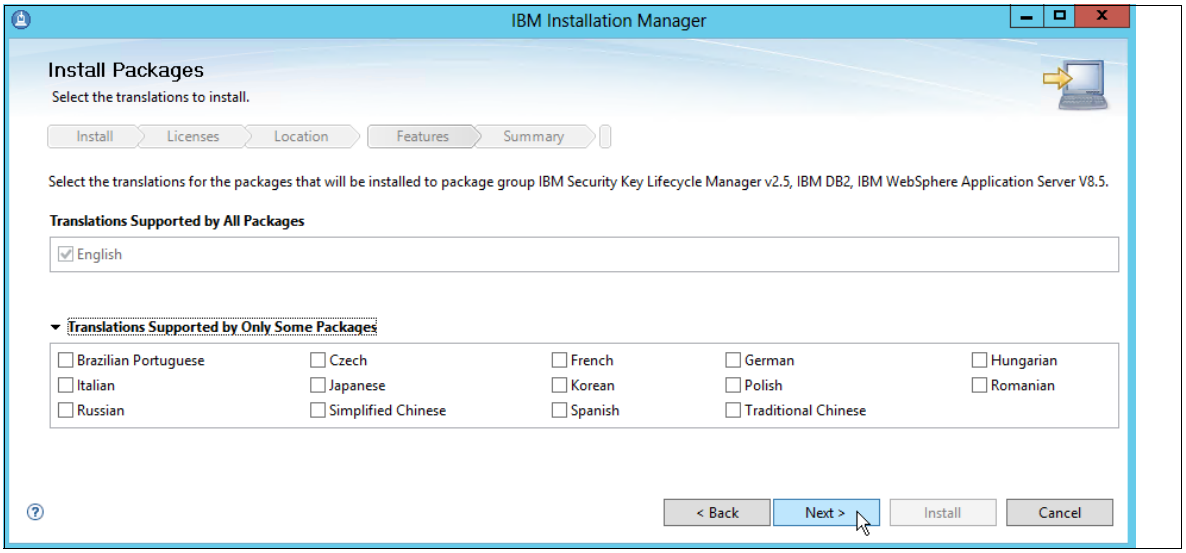

Figure 3-27 Language translation options

16. When the next step of the installation process loads, a progress bar briefly displays and indicates that some feature validation is being completed. Here you can expand the display and see all of the packages that were selected for installation. If you installed the .NET Framework, no components show dependencies, as shown by the SKLM example in Figure 3-28.

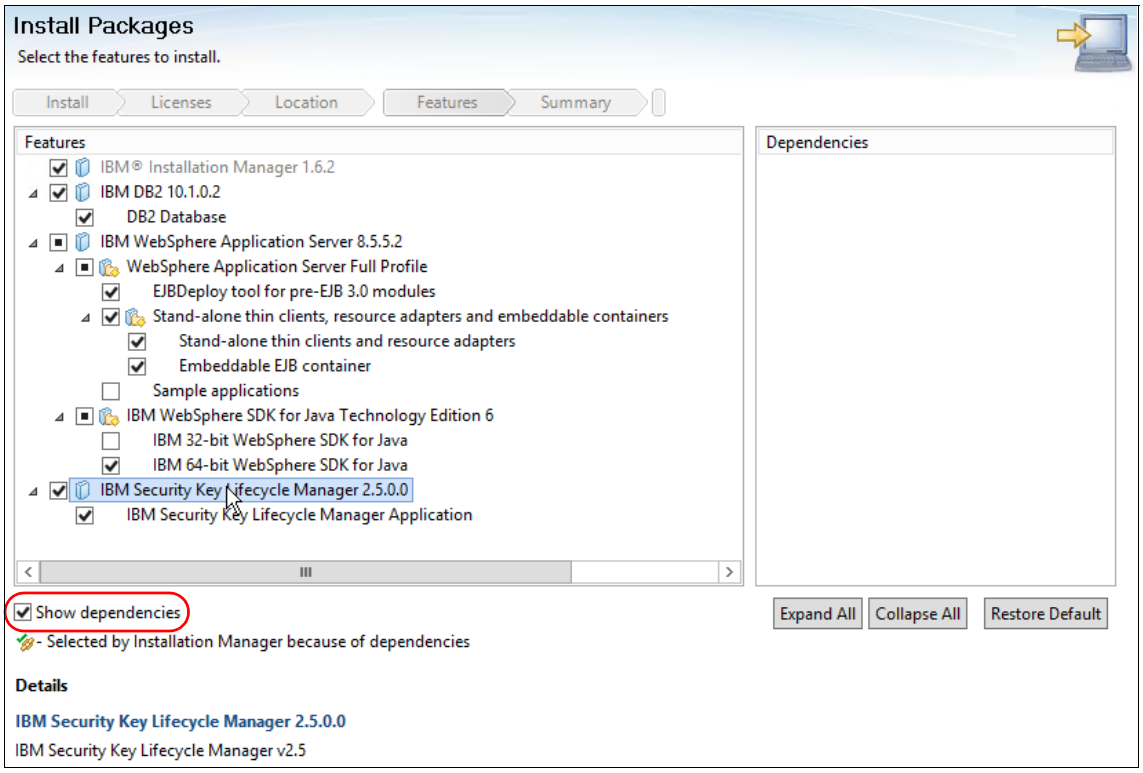

Figure 3-28 Dependency check

17. When you are done reviewing the features, click **Next**.

18. In the next window, we accept the default configuration selection for our new instance of DB2 for our SKLM installation. Here, you must provide login credentials for the DB2 administrator account.

    The DB2 account is created as a user account for the OS. Therefore, the account must meet any password requirements for your OS accounts and password limitations for DB2.

    In our example, complex passwords were enforced for Windows 2012. However, we did find during our fix pack installation that using an exclamation point (!) caused an error when we attempted to validate DB2 credentials. For this reason, we recommend limiting DB2 passwords to alphanumeric characters only if possible. DB2 also has user ID restrictions. For more information, see this website:

    `http://publib.boulder.ibm.com/infocenter/cmgmt/v8r3m0/index.jsp?topic=%2Fcom.ibm.sysadmin.hlp%2Fmua10010.htm`

    We elected to accept the default `sklmdb2` suggestion, and default suggestions for home directory, database name, and port. After your DB2 configuration selections are completed, click **Next**, as shown in Figure 3-29.

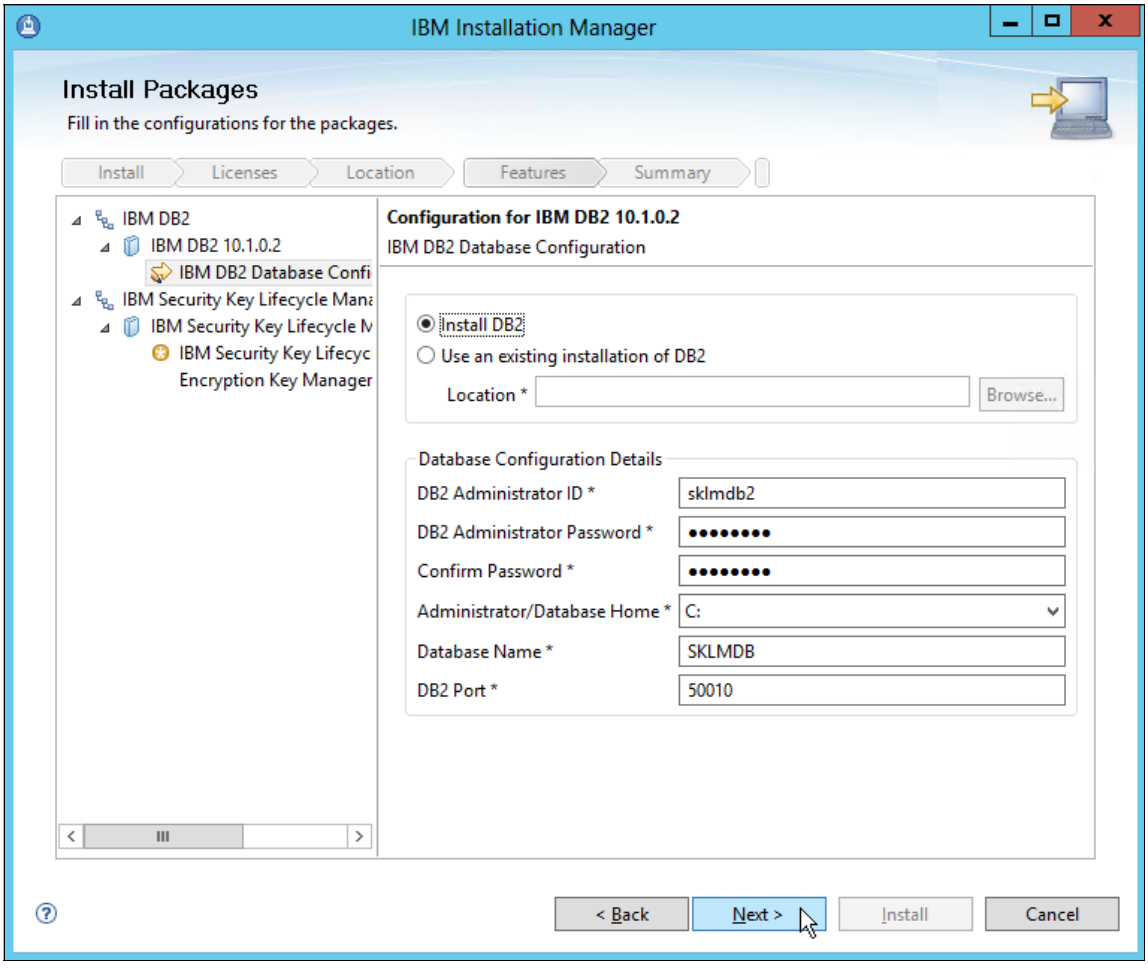

Figure 3-29 DB2 properties

19. As shown in Figure 3-30, you provide login credentials for the administration accounts of the remaining SKLM components. First, a password is required for WebSphere

Application Server. Enter your wanted password, then use the scroll bar to move the window to the right.

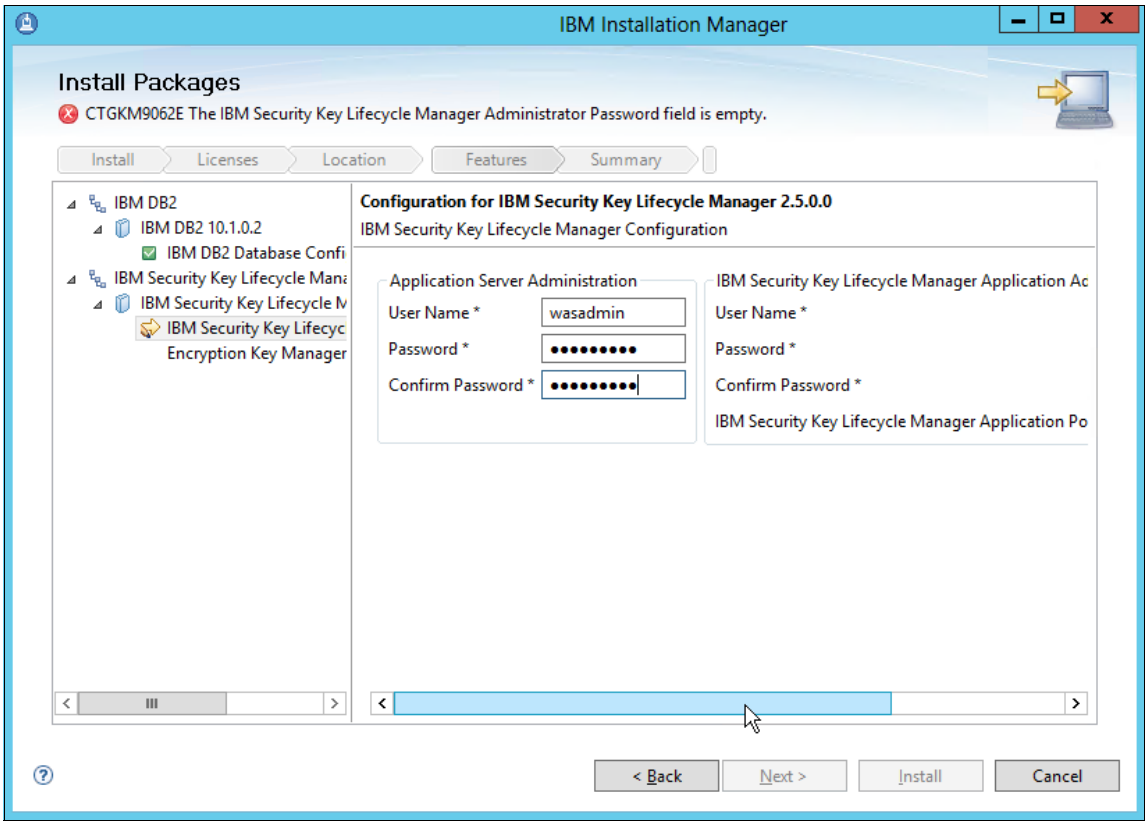

Figure 3-30 WebSphere Application Server credentials

20. Scrolling to the right reviews the entries for SKLM login credentials. Enter a password and confirm the wanted port. We accepted the default, port **9080**. After a password is entered, click **Next**, as shown in Figure 3-31.

> **Important:** Make sure that you record your login credentials for each software component that is listed.

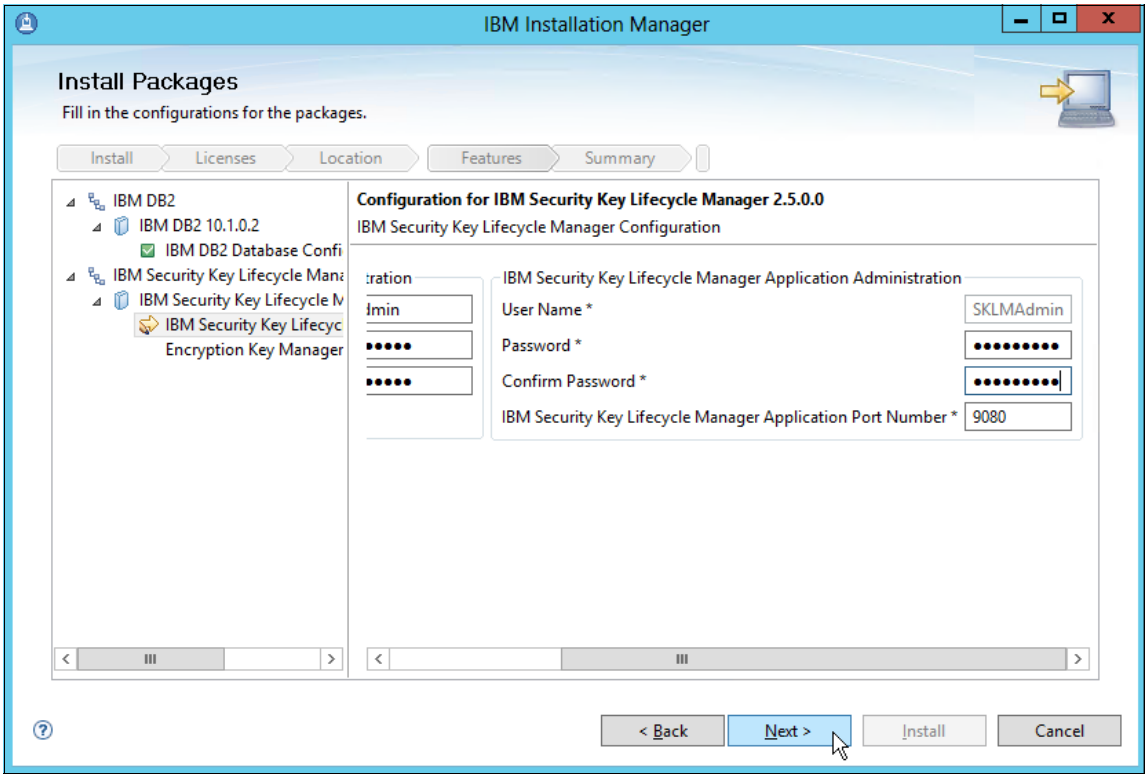

Figure 3-31 SKLM credentials

21. Click **Next** without making any selections in the next window, which provides you the option for Migrating Encryption Key Manager, as shown in Figure 3-32. Encryption Key Manager is a product for managing encrypted drives and tape storage systems and is not covered in this publication.

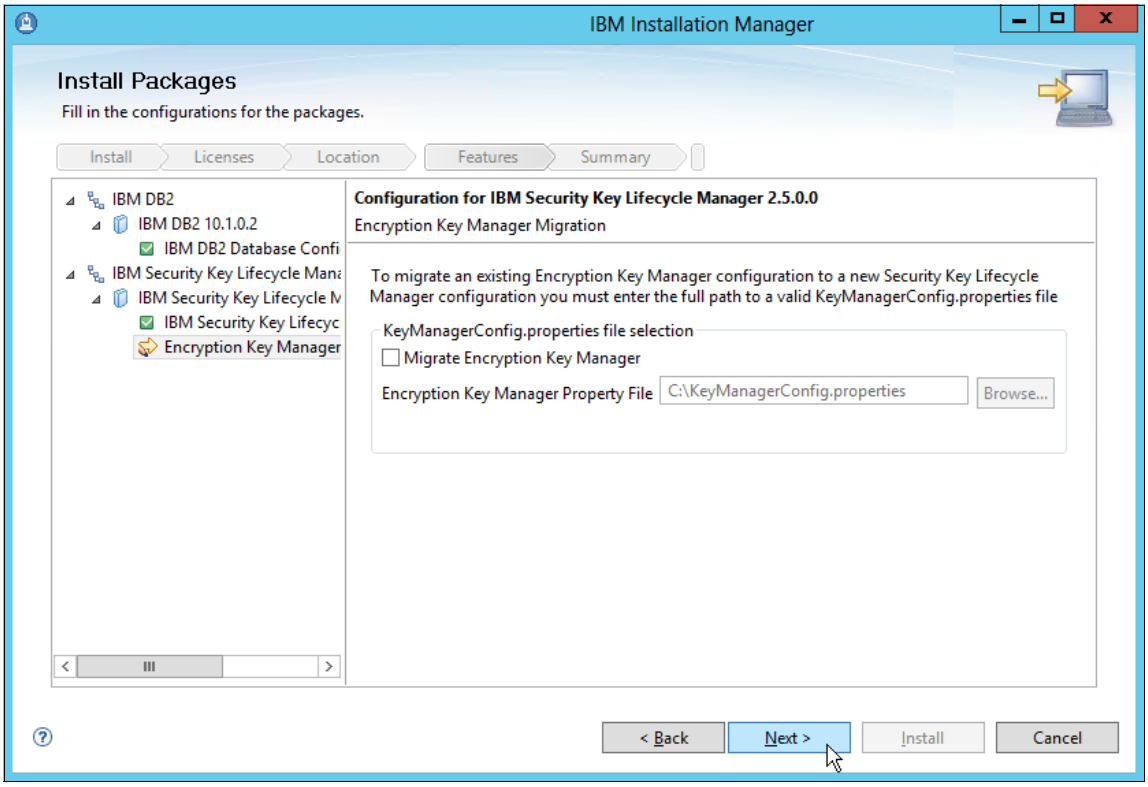

Figure 3-32 Migration option

22. After you review your selection in the summary window, click **Install** (as shown in Figure 3-33) to start the download of the update and the installation process.

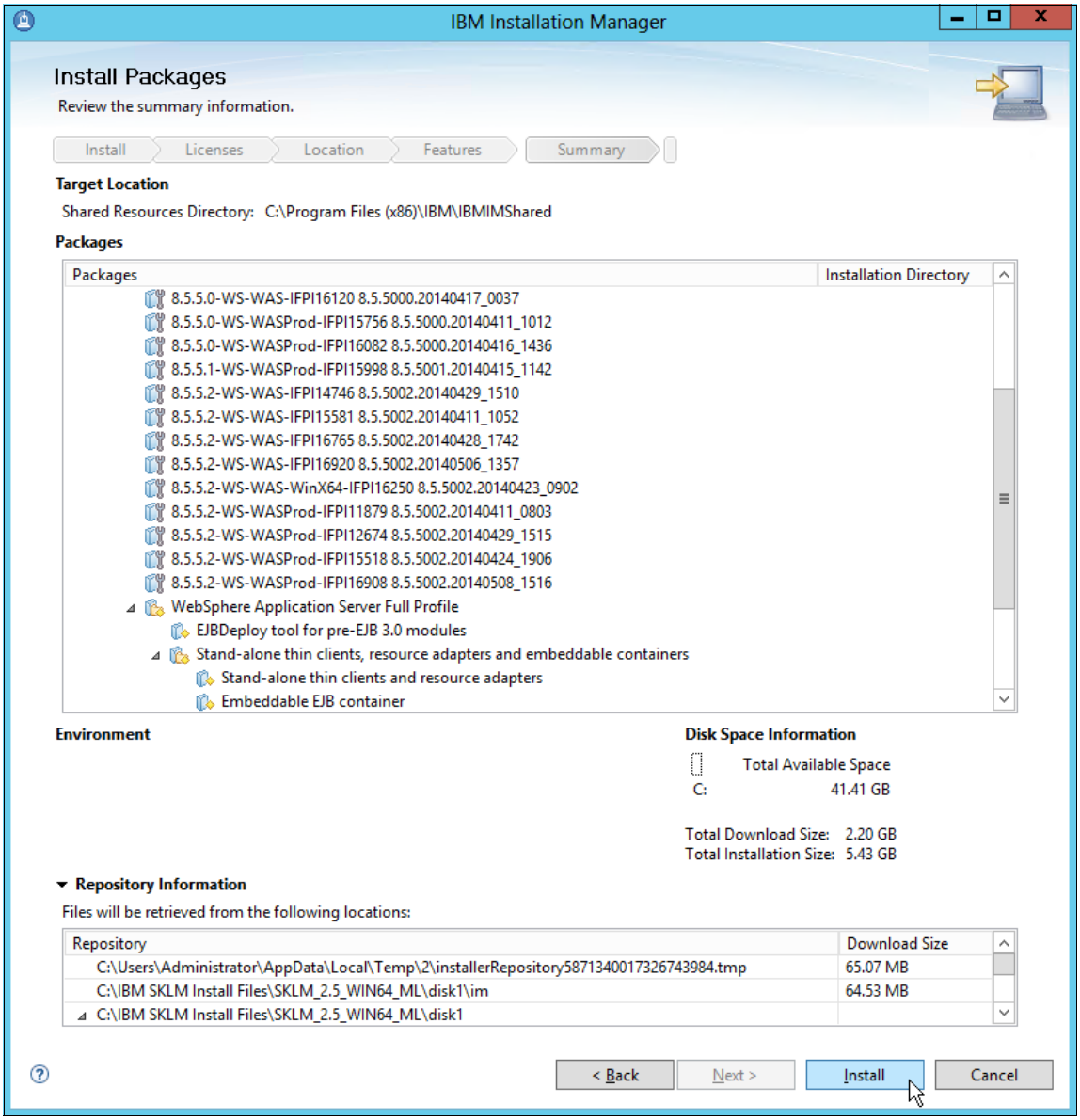

Figure 3-33 Begin installation

23. Throughout the installation process, the information at the bottom of the window indicates the installation progress, download speeds, and general information about the task that is performed, as shown in Figure 3-34. Our download totaled 2.2 GB, and the installation took approximately 17 minutes to complete. Your installation time can vary based on your update selections, system performance, and network connection.

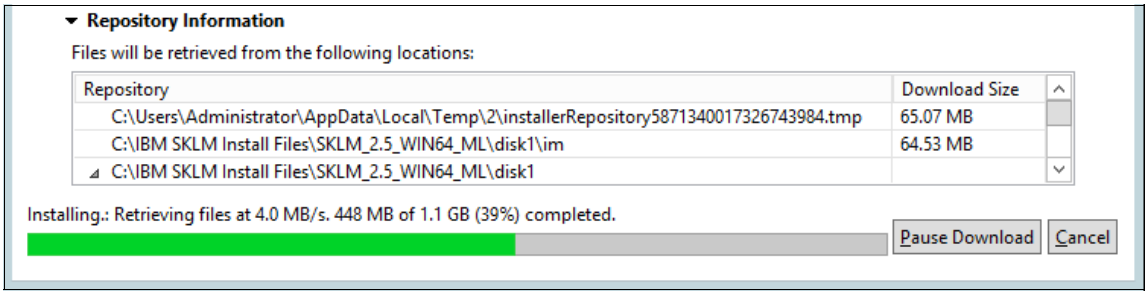

Figure 3-34 Installation progress

24. Upon completion, a success window opens, as shown in Figure 3-35. Select **None** as the option to create a profile. We install the latest fix pack (as described in 3.2.5, “Updating SKLM with the latest fix pack” on page 62) before we configure properties of the SKLM server.

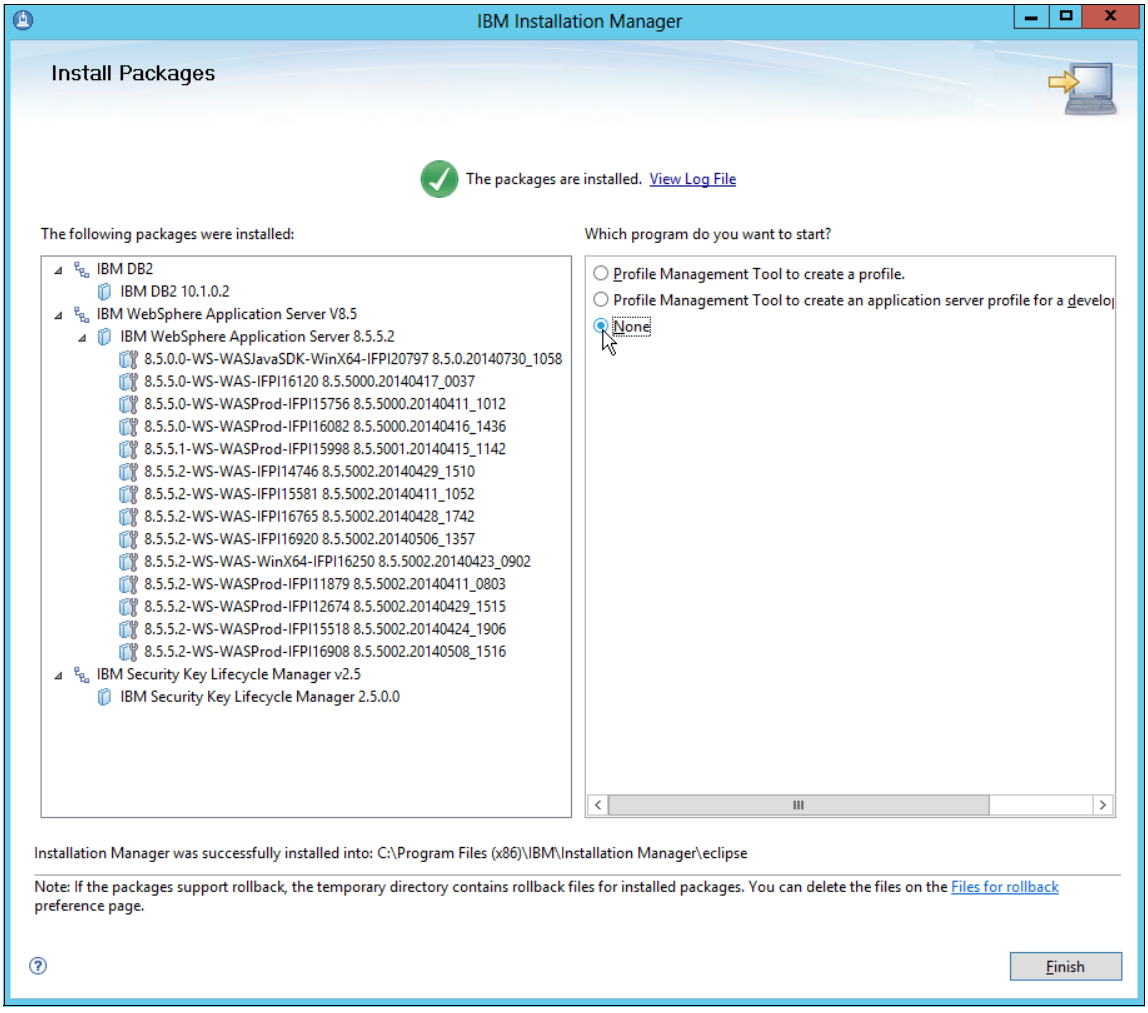

Figure 3-35 Installation complete

25. Select **Finish** to close the installation wizard. You can minimize the IBM Installation Manager window or leave it in the background because we use it in the next section to install the fix pack.

3.2.5 Updating SKLM with the latest fix pack

In this section, we describe the steps to install a fix pack to the SKLM component of your installation. In this example, a base installation of SKLM version 2.5.0.0 is updated to version 2.5.0.2 with the installation of fix pack 2. Version 2.5.0.2 is the first version with System x server options included in the user interface. Before you begin the process, ensure that you have the files that are required for the fix pack installation as described in 3.1.3, “Acquiring SKLM updates” on page 32.

For more information about fix pack information and installation, see the readme file that accompanies the fix pack download. In our example, that file was `2.5.0-ISS-SKLM-FP0002.README.html`.

Complete the following steps:

1. The fix pack files must be copied locally to the virtual or physical server on which SKLM was installed. As shown in Figure 3-36, we create a directory with a descriptive name, `sklm_fixpack_repo_win`, to copy our current (and possibly) future fix packs into.

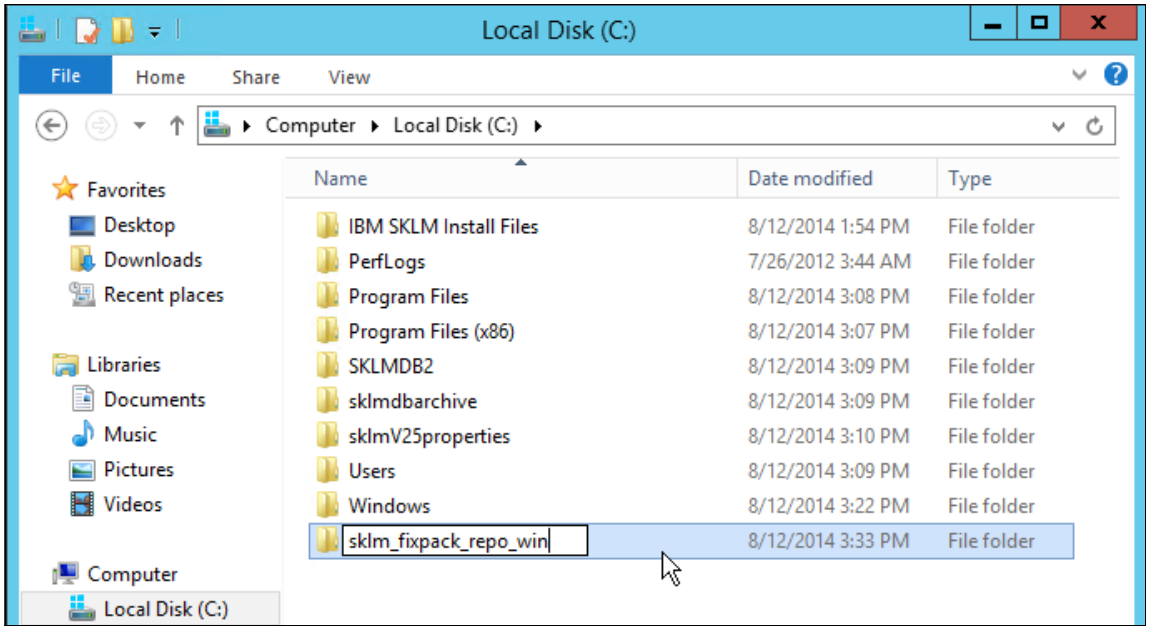

Figure 3-36 Create fix pack directory

2. Copy the fix pack `.zip` file (in our example, `2.5.0-ISS-SKL-FP0002-Windows.zip`) to the new SKLM server directory. For instance, this process can be done for a remote Windows system by sharing a local drive or the local clipboard to your SKLM server with a Windows Remote Desktop Connection or by way of a network file share.
3. Extract the contents of the `.zip` file into the fix pack directory that you created in Step 1.
4. Validate the size and contents of the fix pack. In our example, the fix pack directory was approximately 100 MB.
5. You must start the IBM Installation Manager. The default location on a Windows system is `C:\Program Files (x86)\IBM\Installation Manager\eclipse\IBMIM`. Locate this application on your SKLM system, right-click it, then select **Run as administrator**, as shown in Figure 3-37.

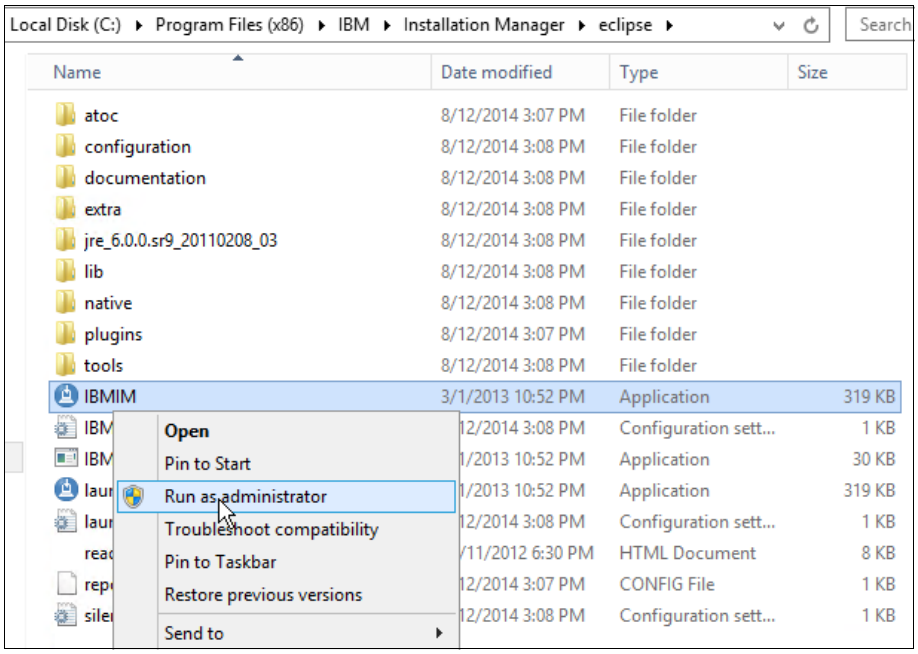

Figure 3-37 Selecting Run as administrator option

6. You must import the fix pack location as a repository in IBM Installation Manager. Select **File** → **Preferences**, as shown in Figure 3-38.

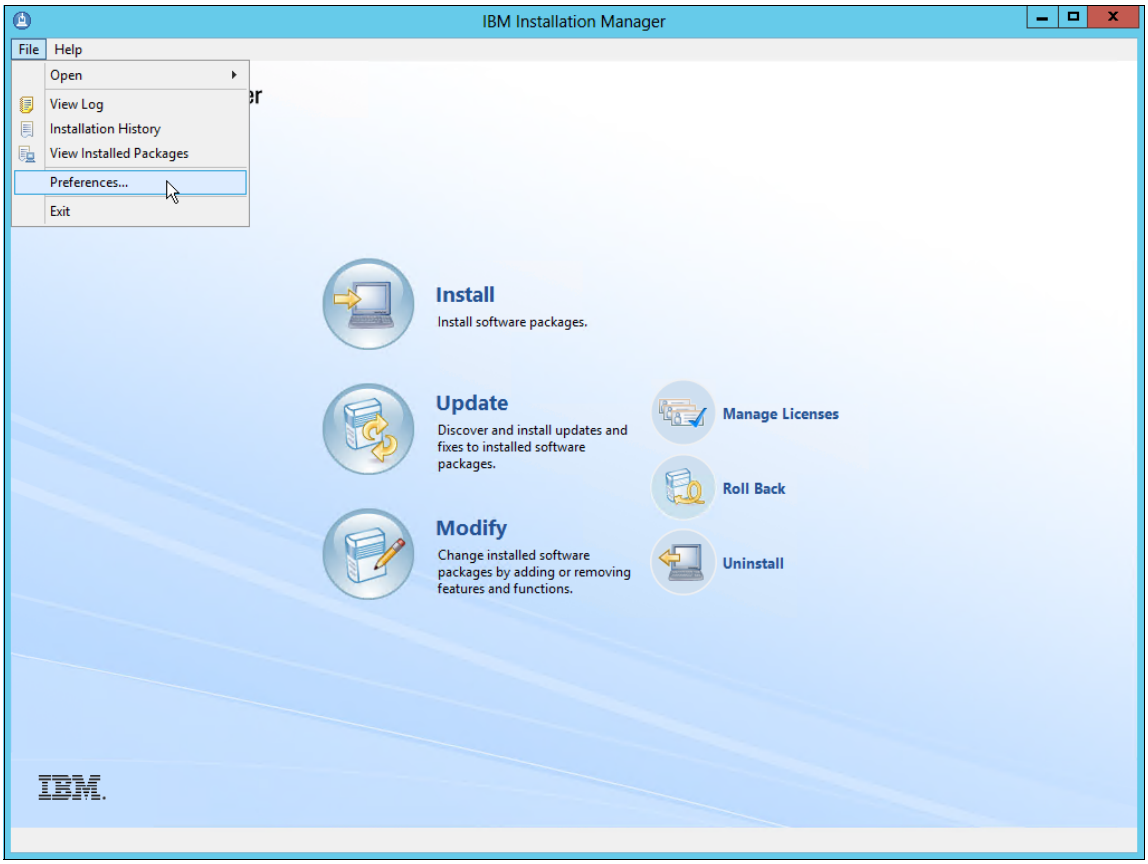

Figure 3-38 Open preferences

7. In the Preferences window, select **Repositories** from the left side pane, as shown in Figure 3-39.

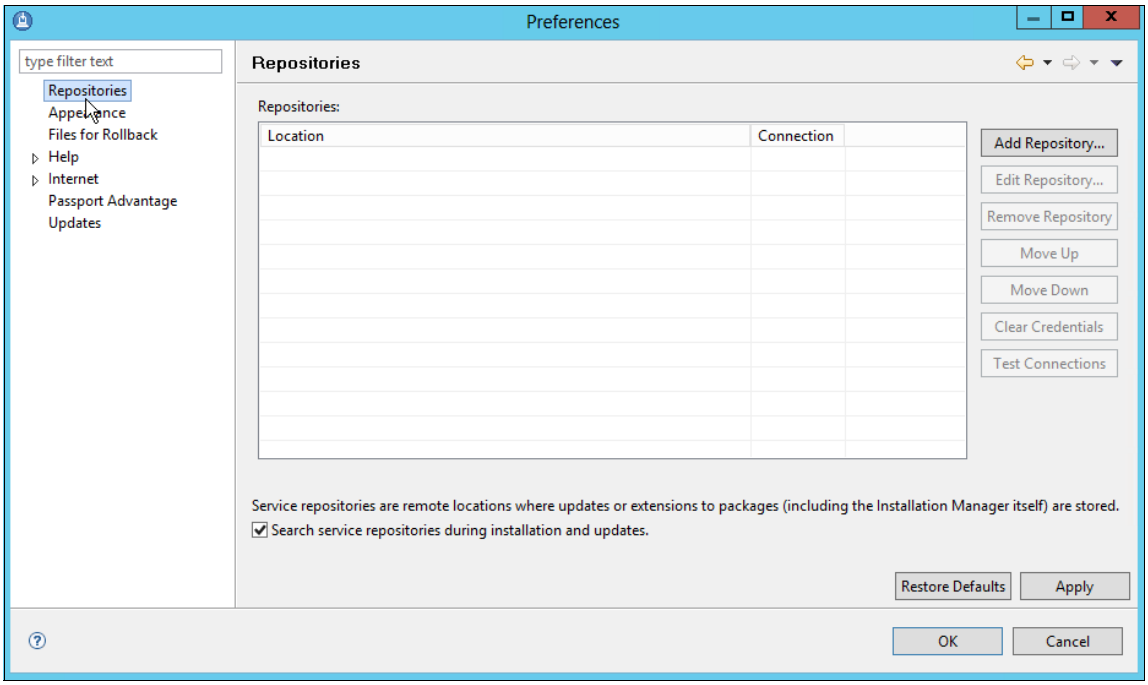

Figure 3-39 Select repositories

8. Select **Add Repository…** on the right side. Then, click **Browse** in the Add Repository window that opens.
9. Browse to the `repository.config` file within your fix pack directory, select it, and click **Open**, as shown in Figure 3-40. On our system, this file was in the `C:\sklm_fixpack_repo_win\2.5.0-ISS-SKLM-FP0002-Windows\` directory.

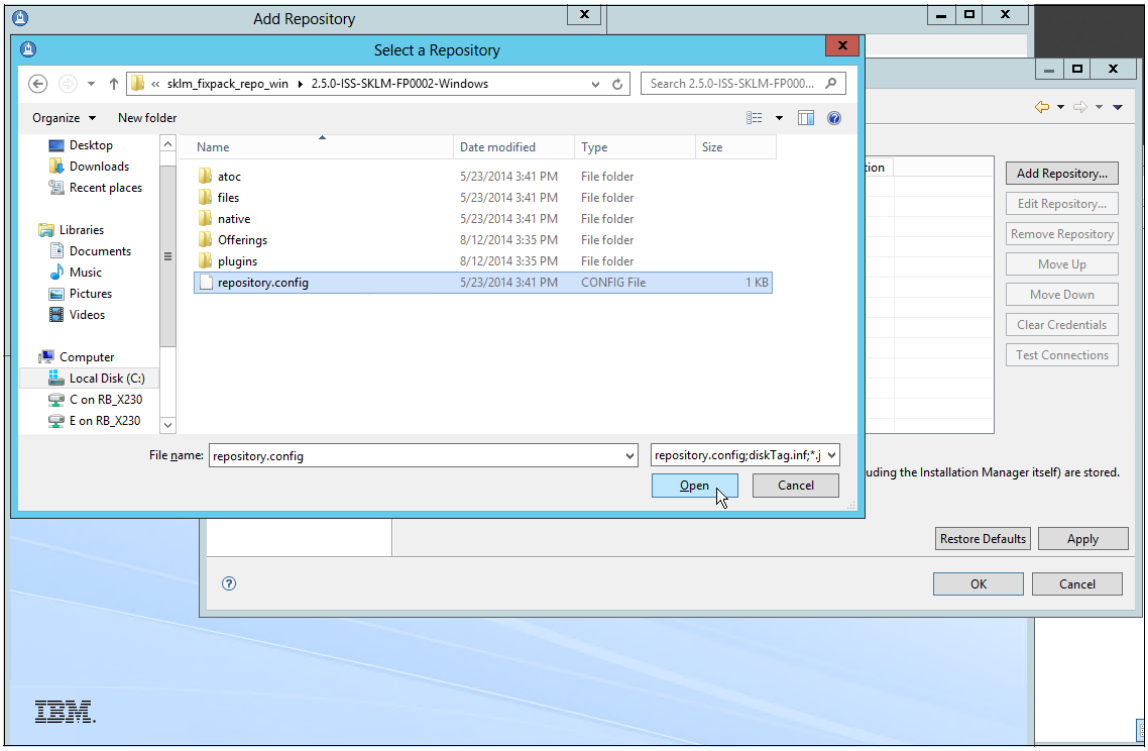

Figure 3-40 Open repository.config file

10. In the Add Repository window, select **OK** to import the directory as a fix pack repository, as shown in Figure 3-41.

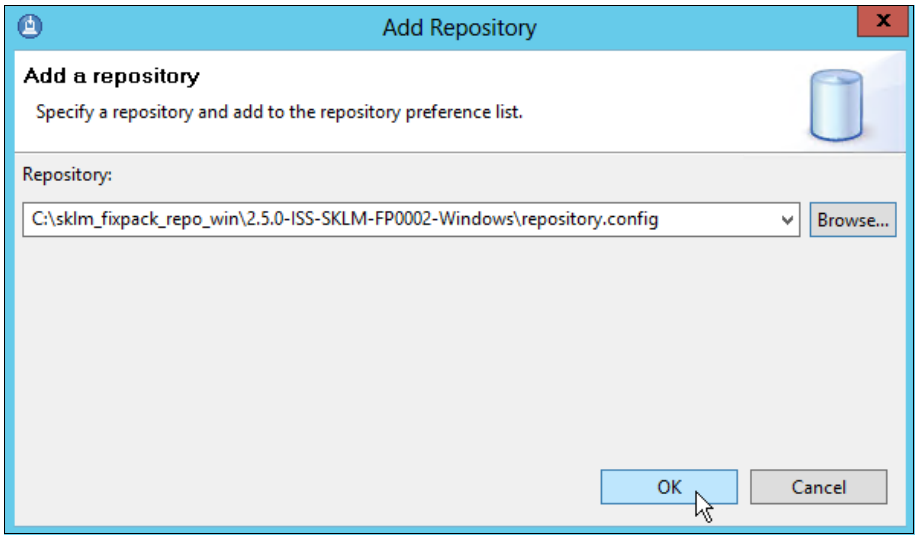

Figure 3-41 Import repository

11. After the repository is imported, ensure that the Search service repositories during installation and update option is not selected. (SKLM does not support the use of Internet-based repositories.) As shown in Figure 3-42, click **Apply**, then click **OK** finalize the import and changes. Then, close the Preferences window.

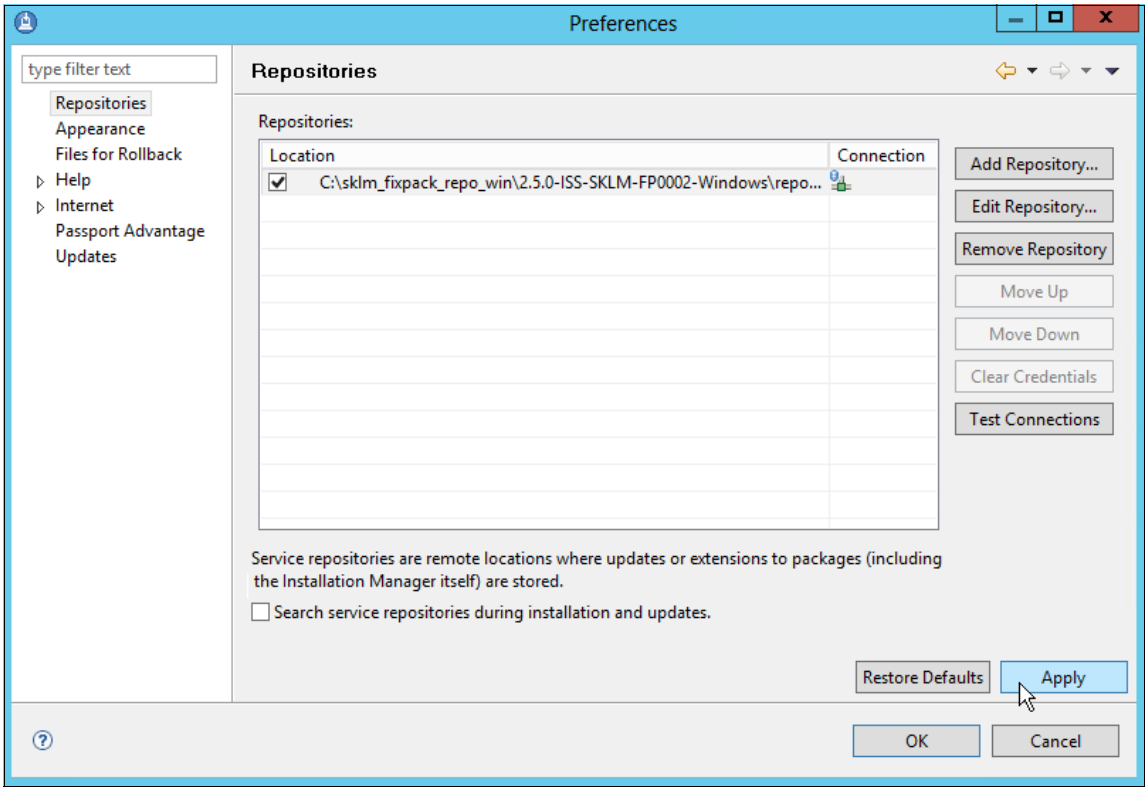

Figure 3-42 Apply new repository

12. In Installation Manager welcome window, select **Update**, as shown in Figure 3-43.

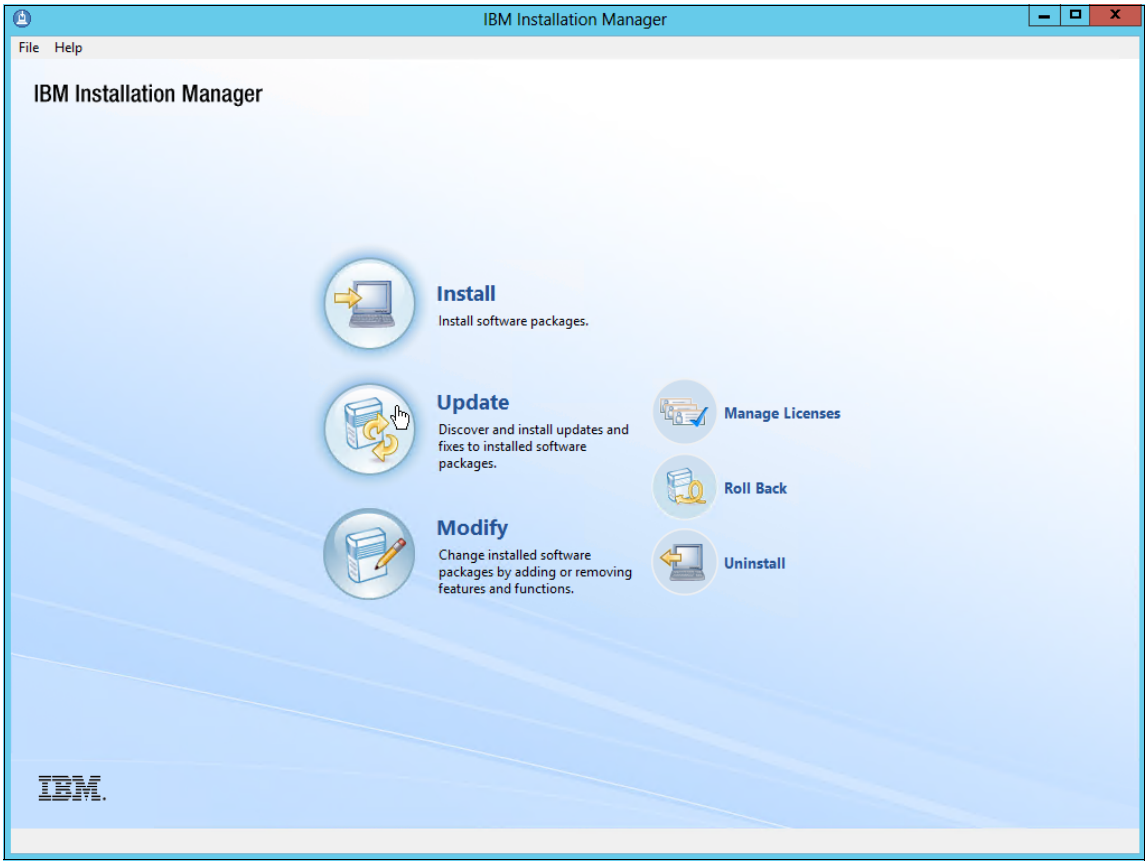

Figure 3-43 Selecting Update option

13. In the Update Packages window (as shown in Figure 3-44), select **IBM Security Key Lifecycle Manager** as the only package group to update because we did not import any update repositories for the other components now. Select **Next**.

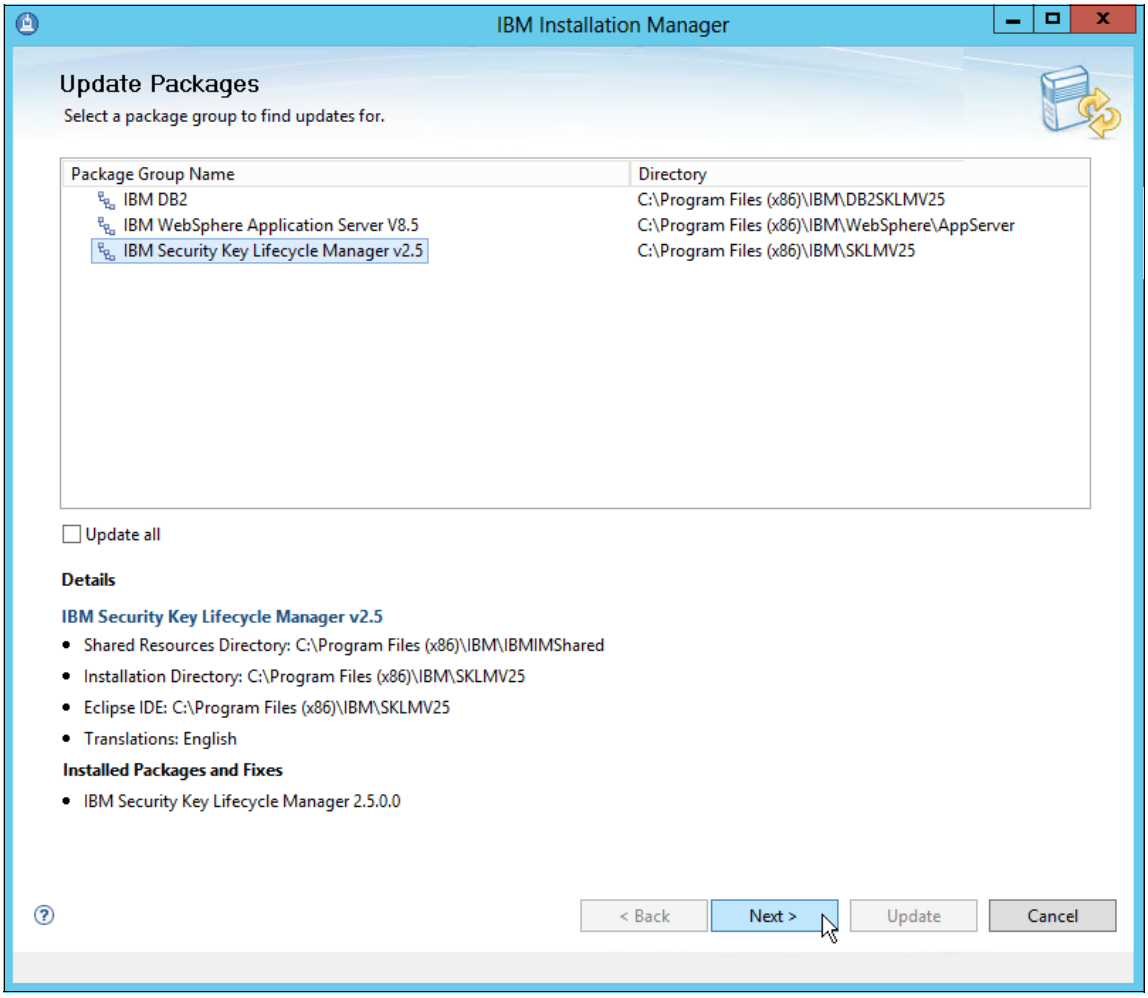

Figure 3-44 Update Packages selection

14. In the next window, we find only one update package, even when **Show recommended only** is not selected. If we imported multiple fix packs or had fix packs from a previous update, we want to select **Show recommended only** or click **Select Recommend**. Click **Next**, as shown in Figure 3-45.

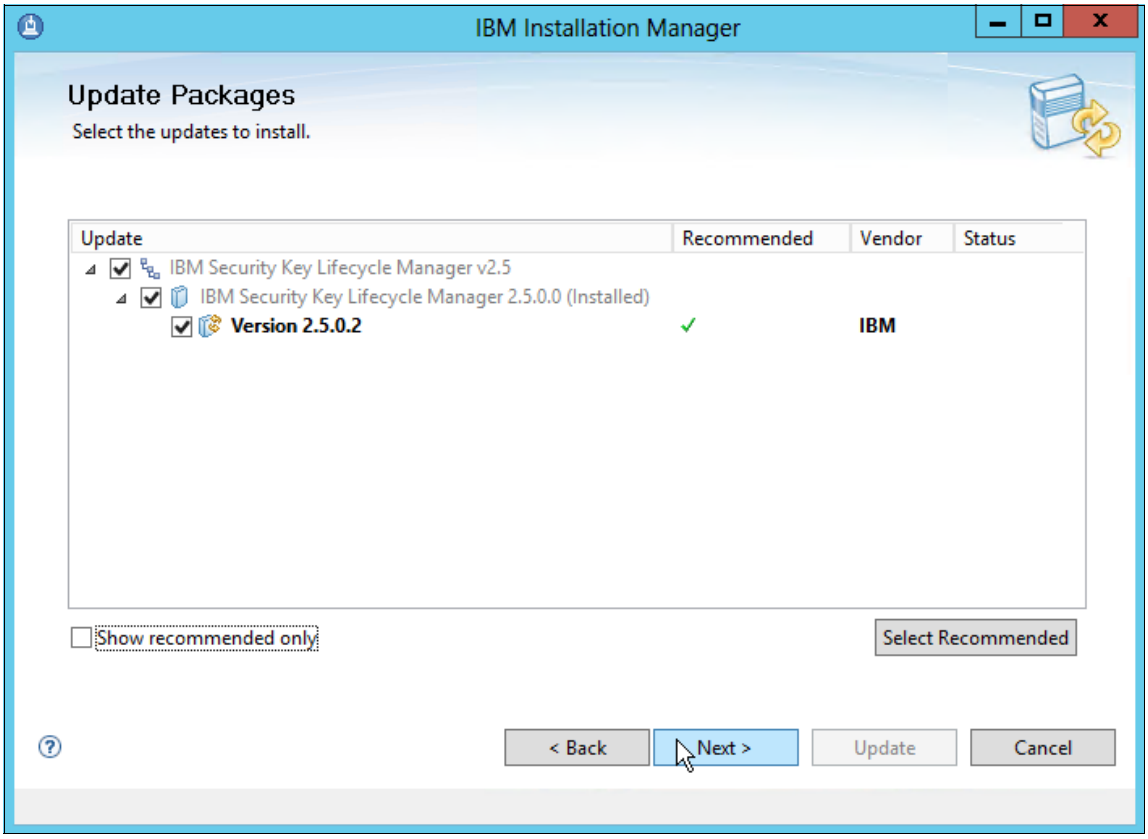

Figure 3-45 Package recommendations

15. Read and accept the terms in the license agreement; then, click **Next**.
16. In the Summary window, confirm the details of your fix pack installation and select **Next**, as shown in Figure 3-46.

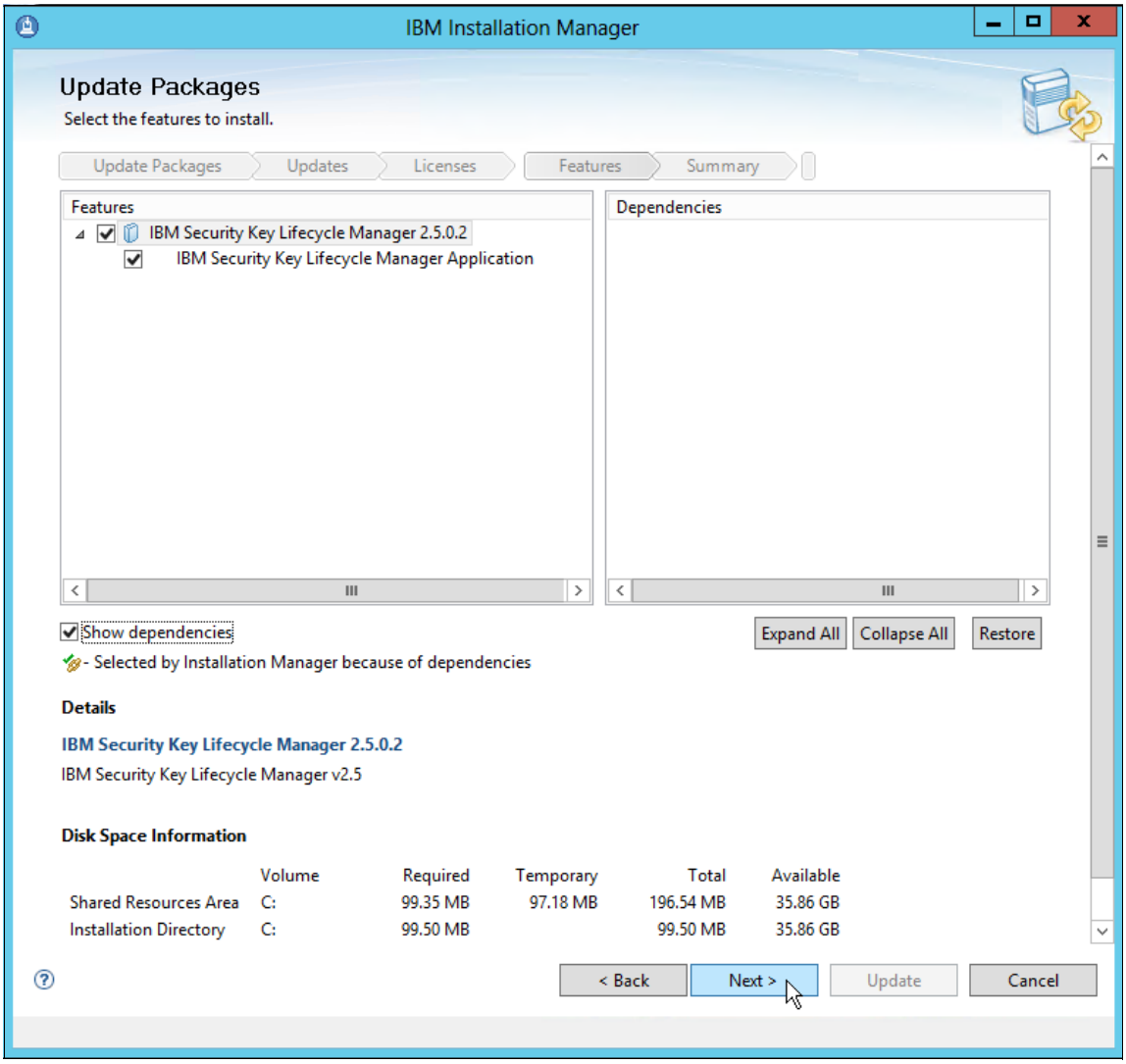

Figure 3-46 Confirm summary

17. Enter the login credentials for the administration account of each SKLM software component. Select **Validate Credentials**, as shown in Figure 3-47, and wait several seconds to 1 minute for IBM Installation Manager to test the connectivity to each component.

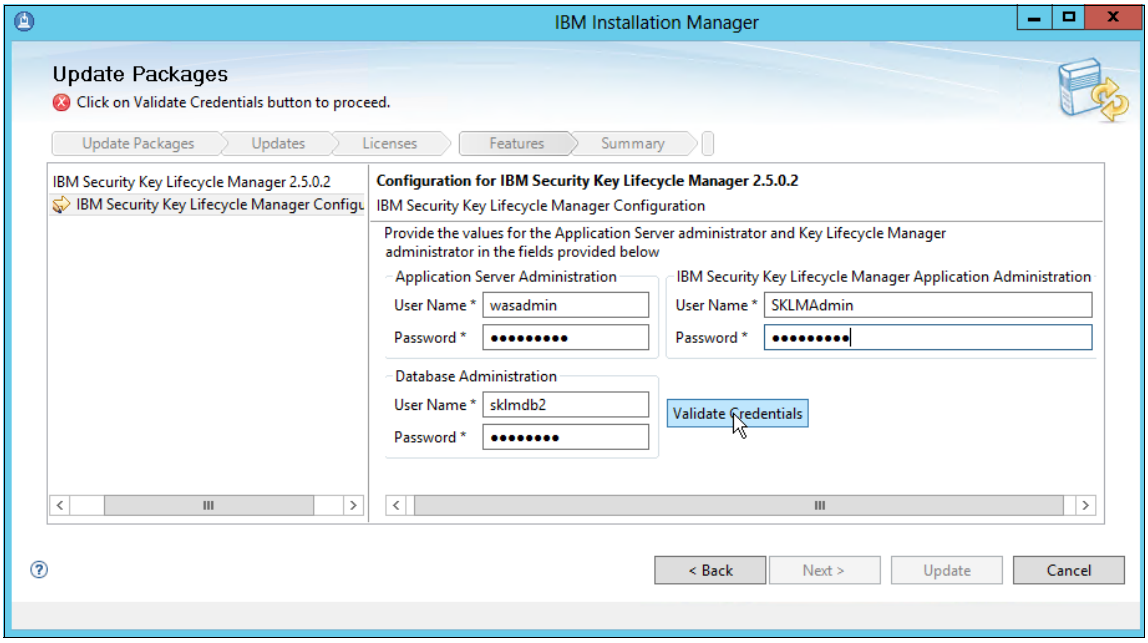

Figure 3-47 Validate Credentials

While the validation is attempted, an hourglass appears as your mouse pointer. It can appear as though the application is frozen, but do not do anything until the process completes.

If the validation succeeds, no error messages are returned and you can select **Next** to continue the update.

If validation fails, an error message, such as `CTGKM9070E The credentials could not be validated at the moment`, is displayed. Often, this message appears because some login credentials were entered incorrectly. If this error recurs, it might be a result of a user name or password that does not meet your OS or software component requirements. It can also be that it was not passed correctly from IBM Installation Manager to your software component.

18. Review the information in the summary window and select **Update** when you are ready to install the fix pack, as shown in Figure 3-48.

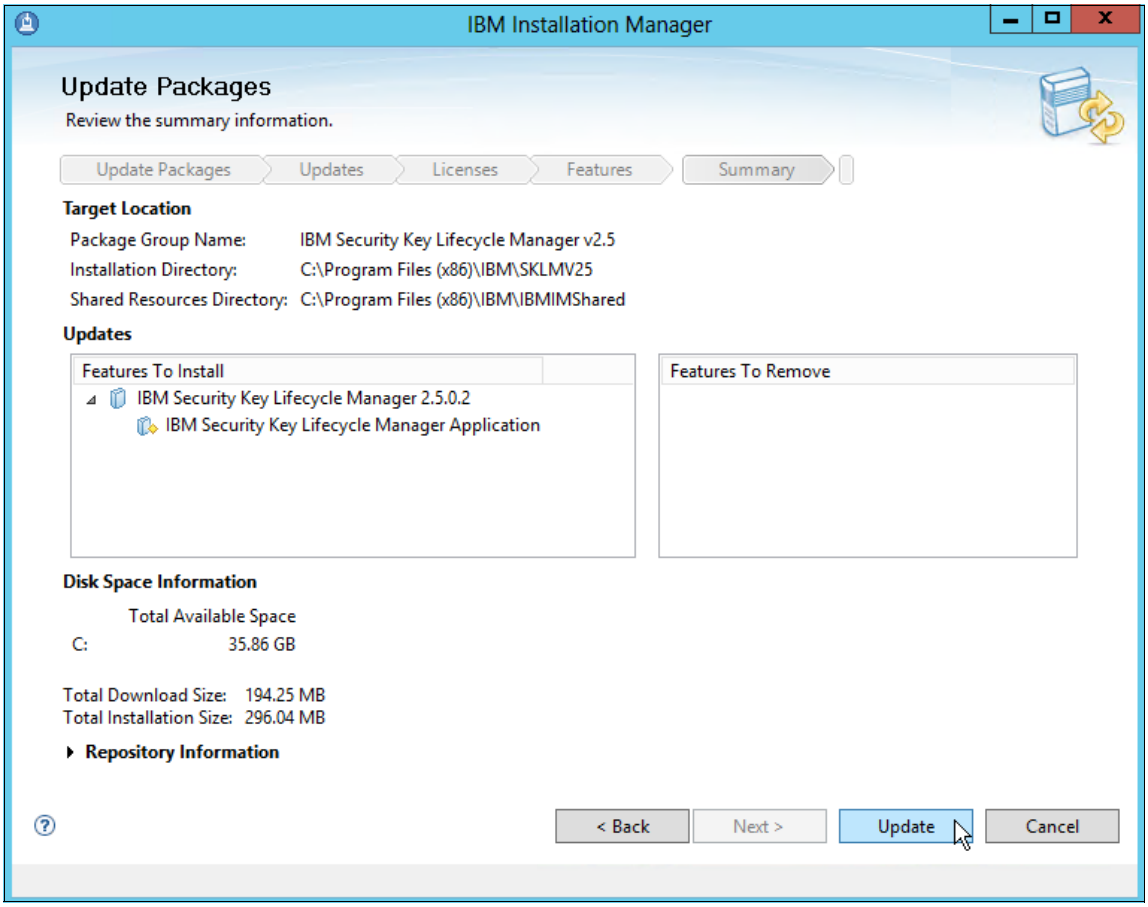

Figure 3-48 Begin updates

19. When the process is complete, you are presented with a message that indicates that the packages are updated. Select **Finish**, as shown in Figure 3-49.

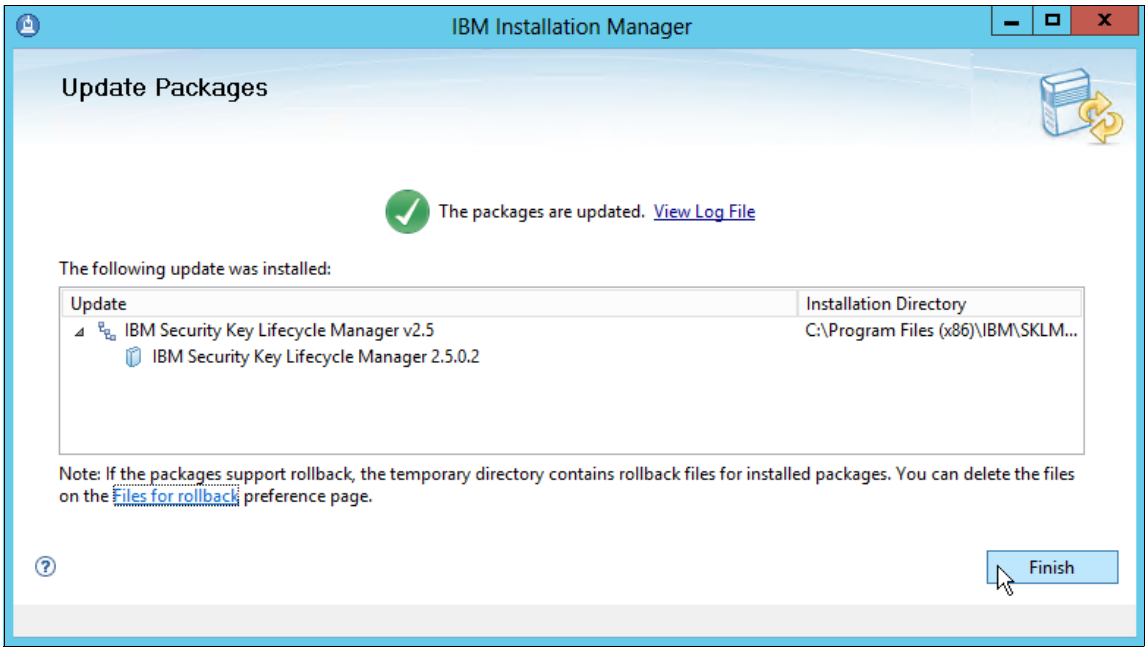

Figure 3-49 Successful update

Your SKLM installation and updates are now complete. We recommend that you restart your SKLM server to ensure that all updates are incorporated, and all services begin correctly upon boot before you begin the SKLM configuration. Use the following command for an immediate reboot in Windows:

```
shutdown /r /t 0
```

Upon restart, your installation and update tasks for a basic SKLM installation are complete. Proceed through the rest of this chapter to validate and configure your environment.

3.3 Validate SKLM installation

In this section, we describe some basic validation tasks of the SKLM installation. We also provide some outlines abut how to access the different components SKLM.

3.3.1 Checking for errors

The SKLM instance can be verified at a basic level by requesting the version and build information from the command line as described next.

For more information about what services should be running and the ports that should be active, see this website:

```
http://www.ibm.com/support/knowledgecenter/SSWPVP_2.5.0/com.ibm.sklm.doc_2.5/cpt/cpt_insguide_tklm_postinstall_processesrunning.html?lang=en
```

Complete the following steps:

1. Open a command prompt and browse to the WebSphere Application Server `bin` directory. On our Windows Server 2012 with default directory locations, this directory is at `C:\Program Files (x86)\IBM\WebSphere\AppServer\profiles\KLMProfile\bin`
2. Enter the wsadmin shell by using the following command:

   `.\wsadmin.bat –lang jython –username <sklm administrator> –password <administrator password>`

   where <*sklm administrator*> is an SKLM admin account (often the default SKLMAdmin) and <*administrator password*> is the password for that account.

   The expected command output is shown in Example 3-1, in which a successful connection to WebSphere Application Server is indicated.

Example 3-1 Successful connection

```
PS C:\Program
Files(x86)\IBM\WebSphere\AppServer\profiles\KLMProfile\bin>.\wsadmin.bat
-username SKLMadmin -password PasswOrd! -lang jython

WASX7209I: Connected to process "server1" on node SKLMNode using SOAP
connector;  The type of process is: UnManagedProcess

WASX7031I: For help, enter: "print Help.help()"
```

3. At the wsadmin prompt run the following command:

   `print AdminTask.tklmVersionInfo()`

   The output should include the following status:

```
IBM Security Key Lifecycle Manager Version = 2.5.0.2
IBM Security Key Lifecycle Manager Build Level = 201405231453
```

4. If all commands run and the version information is what is expected, your SKLM installation was successful.

3.3.2 Accessing components

In this section, we confirm that you can access the user interface for each component of the SKLM install, including SKLM, WebSphere Application Server, and DB2.

Access the SKLM web interface

Complete the following steps to access the SKLM web interface:

1. To connect to the SKLM, browse to:

   `https://<SKLM server address>:9080/ibm/SKLM/login.jsp`

   where <*SKLM server address*> is the IP address or host name of the SKLM server.

2. Add any browser connection exceptions and accept any warnings that are presented. The SKLM login window opens, as shown in Figure 3-50 on page 74.

Figure 3-50 SKLM login window

3. Log in to the web interface of SKLM and click **Help** → **About**, as shown in Figure 3-51. In our example, we kept the default SKLM administrator account `SKLMAdmin`. The user name is not case-sensitive. You should record your component passwords during the installation for future reference.

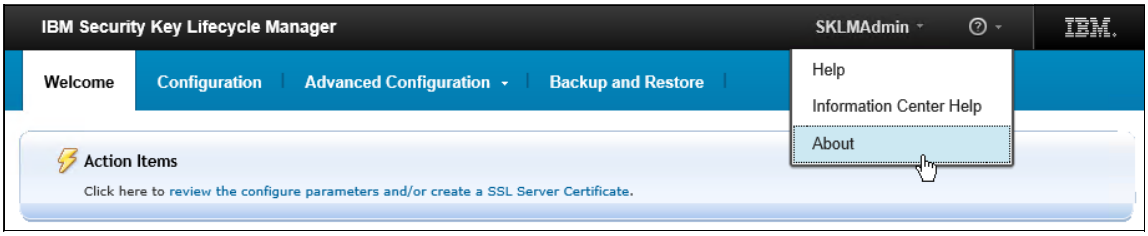

Figure 3-51 Help menu: About

The SKLM version window should reflect the SKLM version and fix pack that you installed, as shown in Figure 3-52.

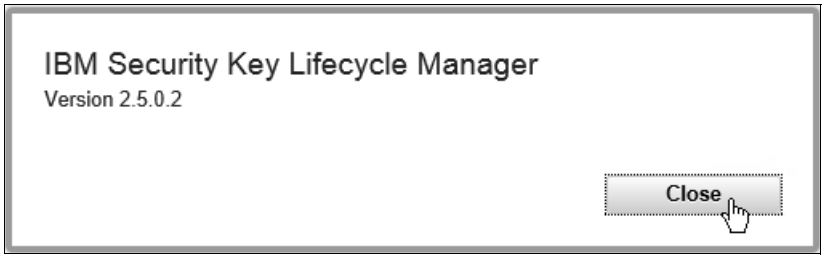

Figure 3-52 SKLM version window

Accessing the WebSphere Application Server web interface

Complete the following steps to access the WebSphere Application Server web interface:

1. To connect to WebSphere Application Manager, browse to:

   `https://<WAS server address>: 9083/ibm/console/logon.jsp`

   where *<WAS server address>* is the IP address or host name of the WebSphere Application Server.

2. Add any browser connection exceptions and accept any warnings that are presented. The login window opens, as shown in Figure 3-53.

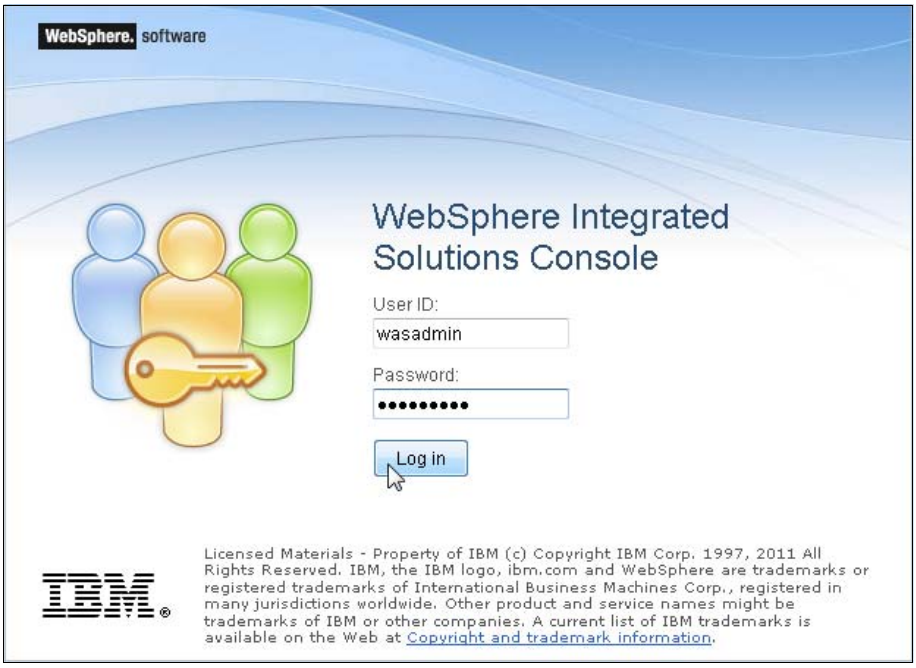

Figure 3-53 WebSphere Application Server login window

3. Log in to the WebSphere Application Server interface. In our example, we used the default administrator account `wasadmin`. You should see the WebSphere Application Server version that you selected or downloaded during the installation process, as shown in Figure 3-54.

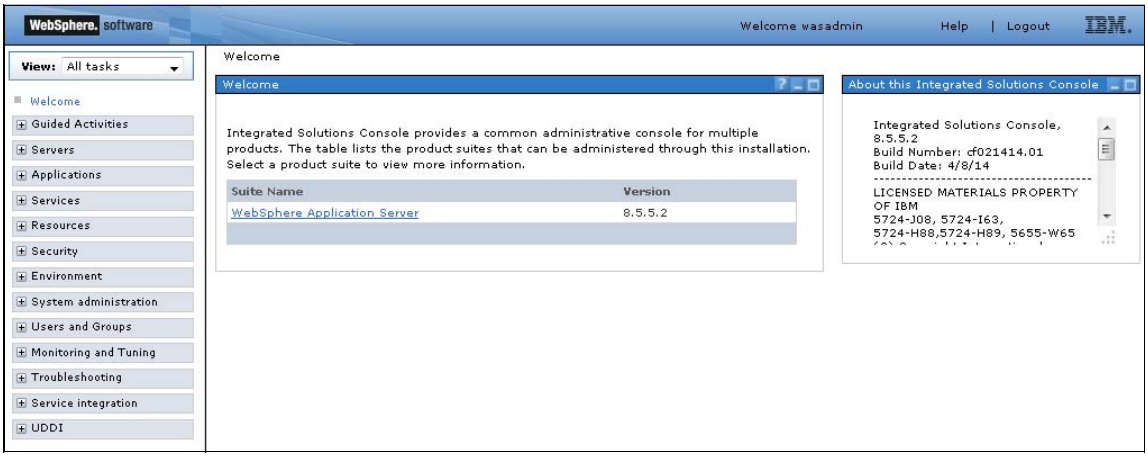

Figure 3-54 WebSphere Application Server Welcome window

The WebSphere Application Server interface often is used to set up user and group permissions while you configure a production environment. However, this process is outside of the scope of this publication.

Browser sessions: You must avoid shared browser sessions that use WebSphere Application Server and SKLM to prevent unpredictable results on the server. When you use multiple browser windows on the same client, the session might be shared.

For example, the session is always shared when you use a Firefox browser. Depending on your registry settings or how you opened your browser window, the session might also be shared in Internet Explorer.

Accessing the DB2 interface

Do not access the SKLM DB2 instance directly. Instead, use the SKLM and WebSphere Application Server interfaces and rely on them to interact with the DB2 database.

If you see the DB2 welcome window that is shown in Figure 3-55 after your installation completes, close the window. The default database and DB2 settings for SKLM are configured.

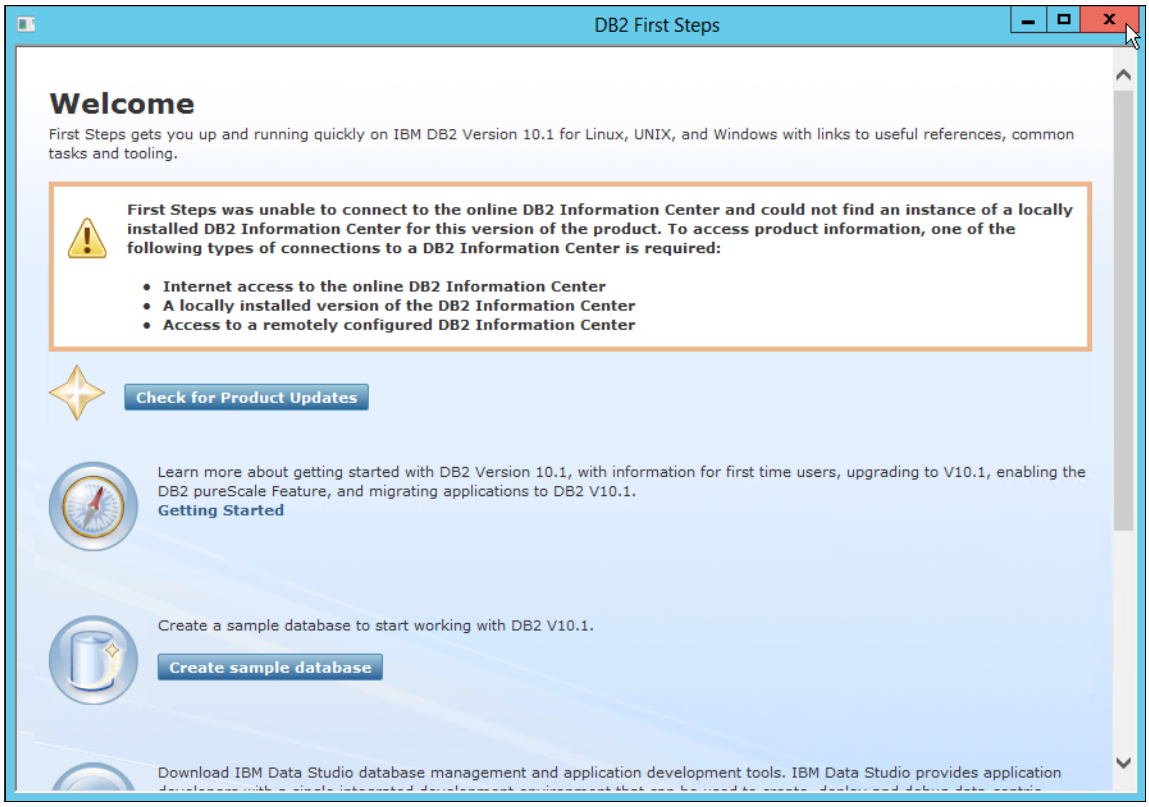

Figure 3-55 DB2 first steps window

3.4 Apply SKLM licensing

SKLM is licensed based on the number of instances installed. The installation of one instance (our primary SKLM server) is now complete. The SKLM license is built in, and no other steps must be taken to apply our license to SKLM. Per the SKLM license agreement, one master instance and one clone installation are included. To install more clones for added redundancy, more SKLM licenses must be purchased.

3.5 Generating an SKLM server certificate

The first step in preparing the SKLM configuration is to generate a certificate for the SKLM server. In our environment, we access SKLM by using a jumpbox behind our firewall; therefore, the SKLM web interface is not publicly broadcasted. For this reason, we describe how to generate a self-signed server certificate. This process also provides a simpler and quicker example setup. You might want to get a signed certificate from a certificate authority (CA) if your management network is accessed from the Internet by your administrators.

Complete the following steps:

1. Log in to the SKLM web interface and browse to the Configuration tab. Then, select **SSL/KMIP**, as shown in Figure 3-56.

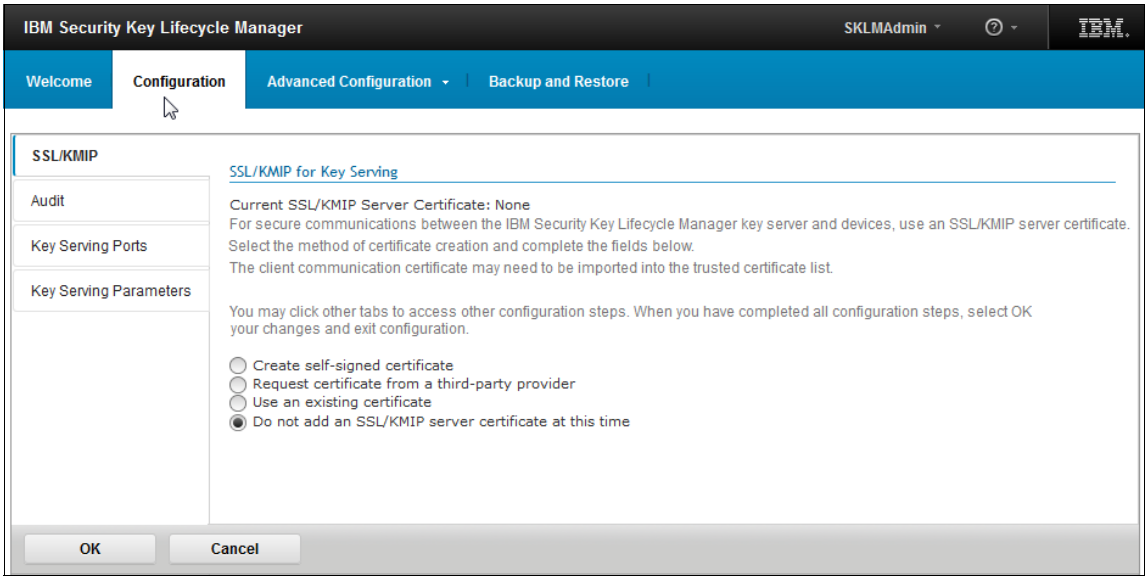

Figure 3-56 SKLM configuration window

2. In the next window, select **Create self-signed certificate**. Enter the requested certificate information. As shown in Figure 3-57, we entered the following information:
   - Certificate label: A descriptive label that appears in the SKLM interface.
   - Certificate description: Plain text can be used here, but the IP or host name of the SKLM server can be used.
   - Validity period: In our example, we kept the default. Because this section covers a basic installation, we do not describe certificate expiration.
   - Algorithm: In our example, we use the default certificate signature algorithm, RSA.

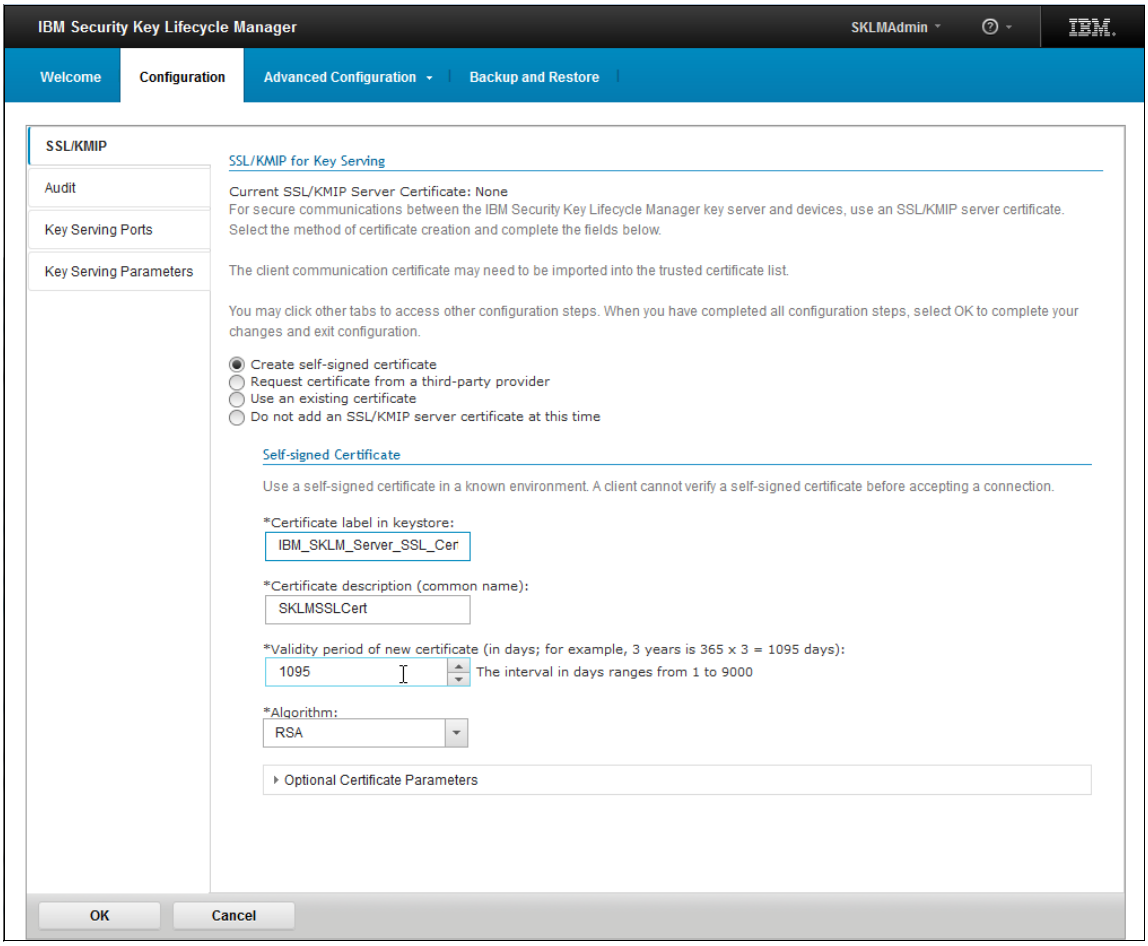

Figure 3-57 Self-signed certificate parameters

3. You can enter your organization and location information into the certificate, as shown under the Optional Certificate Parameters section in Figure 3-58. Click **OK** when your certificate information is complete.

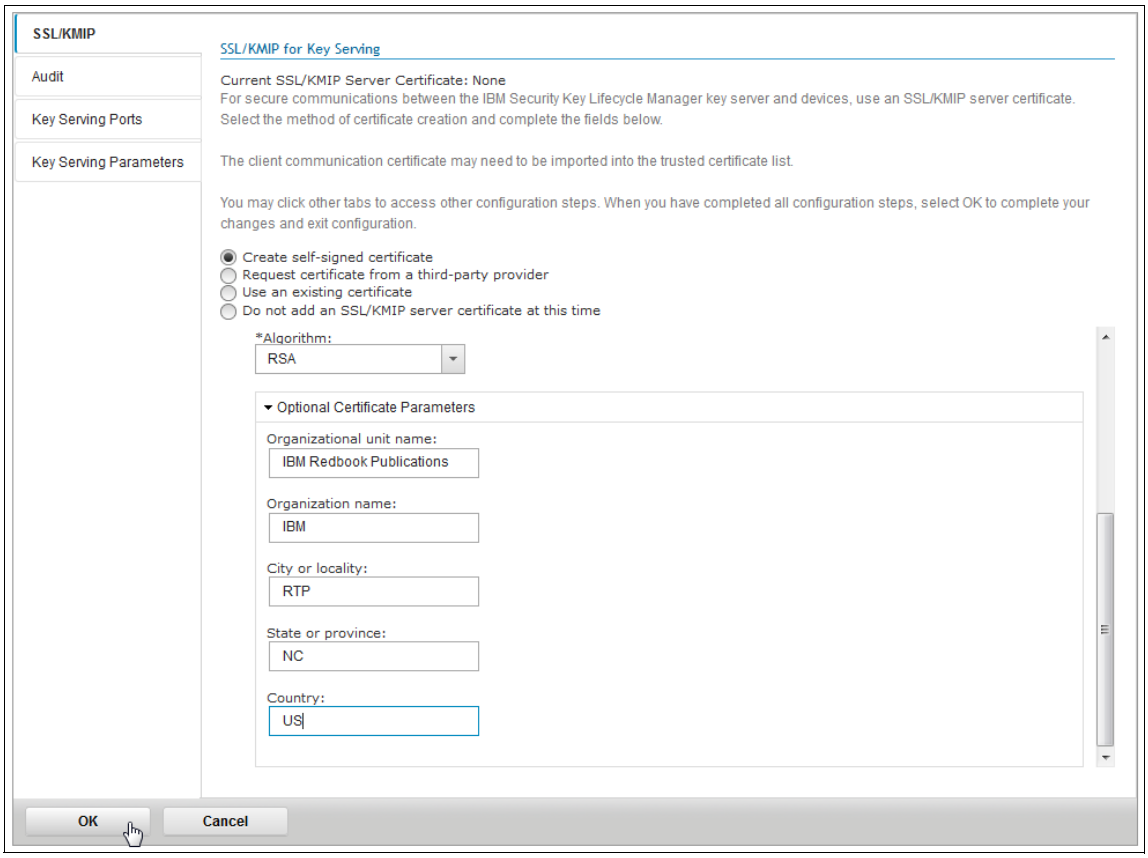

Figure 3-58 Certificate organization and location

4. After the certificate is generated, you see an overview of the configuration settings. The text that is shown in Figure 3-59 indicates important messages. Because we do not have any System x servers with SEDs that are registered in SKLM yet, we do not perform a backup now. See the SKLM backup process that is described in 6.2, “SKLM backup and restore” on page 133 after we import keys for our first server. At that point, data backup is critical.

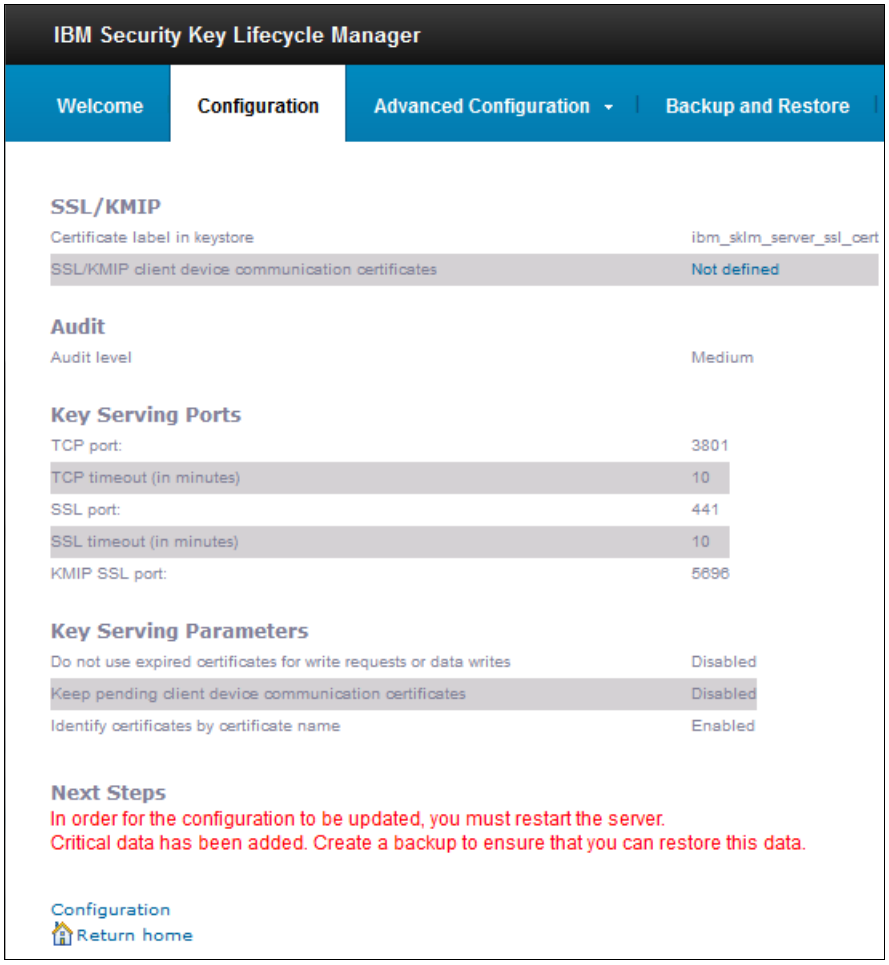

Figure 3-59 Certificate created

5. Reboot your server to ensure that the certificate is created. After the reboot, log back in to SKLM and click **Advanced Configuration** → **Server Certifications**, as shown in Figure 3-60.

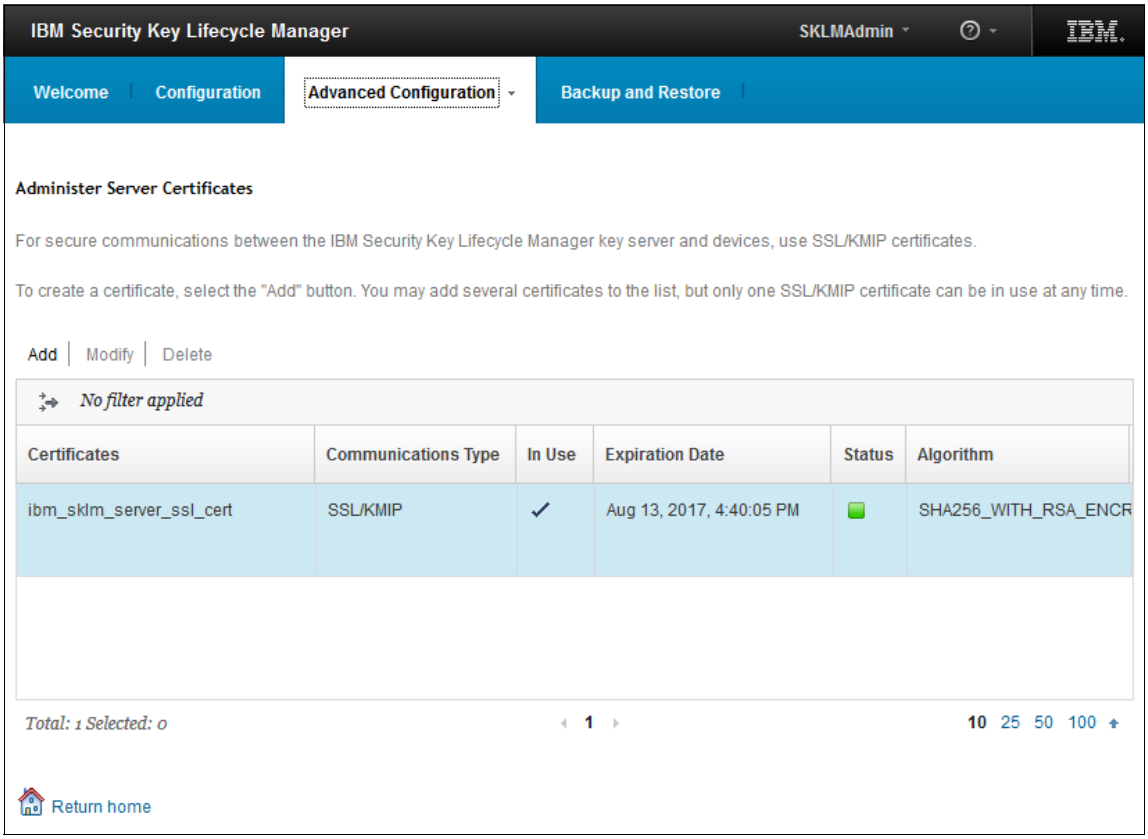

Figure 3-60 New server certificate in use

As an alternative to the use of the Configuration tab, you can generate a server certificate by clicking **Advanced Configuration** → **Server Certificates**, and then selecting **Add**. This approach uses a wizard that is similar to the steps that are described in this section. However, the wizard does not give you the option to import a signed certificate.

Exporting the SKLM server certificate

To use the new SKLM server certificate for an exchange with System x servers, the certificate must be exported by using the command-line interface (CLI). Complete the following steps:

1. Browse to the WebSphere Application Server `bin` directory. In our Windows Server 2012 set up with default directory locations, this directory is at:

   `C:\Program Files (x86)\IBM\WebSphere\AppServer\profiles\KLMProfile\bin`

2. Enter the wsadmin shell by using the following command:

   `.\wsadmin.bat –lang jython –username <sklm administrator> –password <administrator password>`

   where <*sklm administrator*> is an SKLM admin account (often the default SKLMAdmin) and <*administrator password*> is the password for that account.

   The command returns the feedback that is shown in Example 3-2.

Example 3-2 Starting the wsadmin shell

```
PS C:\Program
Files(x86)\IBM\WebSphere\AppServer\profiles\KLMProfile\bin>.\wsadmin.bat
-username SKLMadmin -password Passw0rd! -lang jython
```

```
WASX7209I: Connected to process "server1" on node SKLMNode using SOAP
connector;  The type of process is: UnManagedProcess

WASX7031I: For help, enter: "print Help.help()"
```

3. To list all certificates and obtain the UUID of the server certificate, run the following command:

   ```
   print AdminTask.tklmCertList()
   ```

   The output that contains your server certificate is shown in Example 3-3.

Example 3-3 Certificate list

```
wsadmin>print AdminTask.tklmCertList()
CTGKM0001I Command succeeded.
uuid = CERTIFICATE-cb226137-577e-4f38-9fb4-6d31c803666c
alias = ibm_sklm_server_ssl_cert
key store name = defaultKeyStore
key state = ACTIVE
issuer name = CN=SKLMSSLCert, OU=IBM Redbook Publications, O=IBM, L=RTP, ST=NC,
C=US
subject name = CN=SKLMSSLCert, OU=IBM Redbook Publications, O=IBM, L=RTP,
ST=NC, C=US
creation date = 8/14/14 4:40:05 PM Eastern Daylight Time
expiration date = 8/13/17 4:40:05 PM Eastern Daylight Time
serial number = 197512119346104
```

4. After the UUID of your SKLM server is found, run the following command to export the server certificate:

   ```
   print AdminTask.tklmCertExport('-uuid <server_UUID> -format DER -fileName
   <SKLM_Server_Certificate.der>')
   ```

   where <*server_UUID*> is the SKLM server UUID, and <*SKLM_Server_Certificate.der*> is the wanted fully qualified file name for your exported certificate `.der` file. We created a new directory (`C:\certs`) to contain our certificates.

   The command output is shown in Example 3-4.

Example 3-4 Exporting the certificate

```
wsadmin>print AdminTask.tklmCertExport ('-uuid
CERTIFICATE-cb226137-577e-4f38-9fb4-6d31c803666c -format base64 -fileName
C:\certs\win2k12_sklm.der')
CTGKM0001I Command succeeded.
C:\certs\win2k12_sklm.der
```

5. You might want to record the location of your exported server certificate. This location must be imported into the Integrated Management Module (IMM) of a server to configure your SED key management with SKLM.

3.6 Production environment considerations

In this chapter, we reviewed only a basic setup of one SKLM server. Your SKLM setup is not yet ready for production. Your SKLM server is a critical component of your drive encryption environment, and installing a single, basic instance of SKLM into production exposes you to a high risk of losing access to all of your encrypted data.

Important: If your SKLM server is lost and cannot be recovered, you lose access to all encrypted data on the SEDs it is managing.

We recommended the following practices for a production SKLM environment:

- Create at least one other SKLM server to act as a secondary key management server for the initial primary SKLM instance. Up to five secondary servers are supported by SKLM; however, only up to three can be used by System x servers with the SKLM Feature on-Demand key. For consistency with SKLM product documentation, we refer to the primary SKLM server instance as the *master*, and all of the replicas or secondary instances as *clones* in this publication.
- Set up automatic replication to keep master and clones in sync when changes are made.
- At minimum, keep master and clone servers on different physical hardware. The SKLM servers might be virtual, but those virtual servers should always be on different physical hardware to minimize the possibility of an SKLM outage when hardware is offline.
- Where possible, also configure master and clone SKLM servers on different logical subnets for redundancy and security. The SKLM server must have network access to the IMMs of the System x servers it is managing, but is not required to be on the same layer 2 network.
- Where possible, locate the SKLM master and clone servers in different data centers. SKLM servers should be replicated to disaster recovery sites to ensure the best chance of recovering access to encrypted data if there is a site-wide catastrophe.
- Perform regular backups of your master SKLM server. Record the passwords for each of those backups in a safe place.
- Do not leave backup files locally on the SKLM server; instead, copy them to another storage device or devices. You want to copy your SKLM backups to an offsite location or another data center, especially when site replication of an SKLM server is not possible.
- Do not encrypt your backup files or store them on encrypted devices.

3.7 Conclusion

After the installation, update, initial configuration, and export of certificates that is described in this chapter are complete, you can proceed with your System x server setup. Chapter 4, “Integrated Management Module configuration” on page 85, Chapter 5, “UEFI configuration” on page 109, and Chapter 6, “Managing your System x server SED deployment” on page 123 help you configure your System x servers and SEDs for management by the SKLM server you just set up.

4

Integrated Management Module configuration

In this chapter, we describe the configuration of the Integrated Management Module (IMM), including importing and exporting certificates, and the IBM Security Key Lifecycle Manager (SKLM) target servers. Three options are described that can be used to configure the IMM. The first option uses the graphical web-based interface, the second uses the IMM command line, and the third option works with the Advanced Settings Utility™ (ASU).

For a small deployment, the web interface is the simplest and most intuitive method. If large numbers of servers are deployed, our recommendation is to use the ASU method because it allows for scripting and automation for most of the common settings.

This chapter includes the following topics:

- Introduction to IMM certificates
- Configuring the IMM by using the web-based interface
- Configuring the IMM by using the IMM Command Line Interface
- Configuring the IMM by using the Advanced System Utility

© Copyright Lenovo 2015. All rights reserved.

4.1 Introduction to IMM certificates

To allow for the key management server to trust the source of a key request, a certificate mechanism is used to build a trusted relationship between the key management server and the IMM in the server that is configured. First, you export a certificate on the key management server (in our example, SKLM) and client (IMM) side. In the second step, you import these certificates on the alternative device.

For more information about creating and exporting the SKLM server certificate, see Chapter 3, “IBM Security Key Lifecycle Manager setup” on page 31, in which we describe how to create and export a self-signed certificate or a certificate that is signed by a signing authority.

If a certificate was configured on the IMM to use HTTPS or encrypted communication with the adapter, it is not necessary to generate a separate certificate for the SKLM communication. The existing key can be used for operations. A new certificate must be generated only if there is no certificate for the adapter.

4.2 Configuring the IMM by using the web-based interface

In this section, we describe the configuration of the IMM for remote key management by using the graphical web-based interface.

4.2.1 Accessing the IMM web interface

Complete the following steps to configure the IMM of the server by using the web-based interface:

1. Log in to the web console by using a supported web browser to connect to the IP address of the IMM, which must be configured. If the IMM is not configured, it is at a DHCP-provided address (if a DHCP server was available when power was applied to the server) or the default IP address of `192.168.70.125`.

   Note: We recommend that you set a static or reserved IP address for the IMM before you continue this process because you must use the IP address of the IMM in most of the following procedures. For more information about how to configure the IMM, see the IBM Integrated Management Module II User’s Guide, which is available at this website:

   `http://www.ibm.com/support/entry/portal/docdisplay?lndocid=MIGR-5086346`

2. To access the IMM login window, browse to the IP address or DNS name of the IMM to be configured, as shown in Figure 4-1.

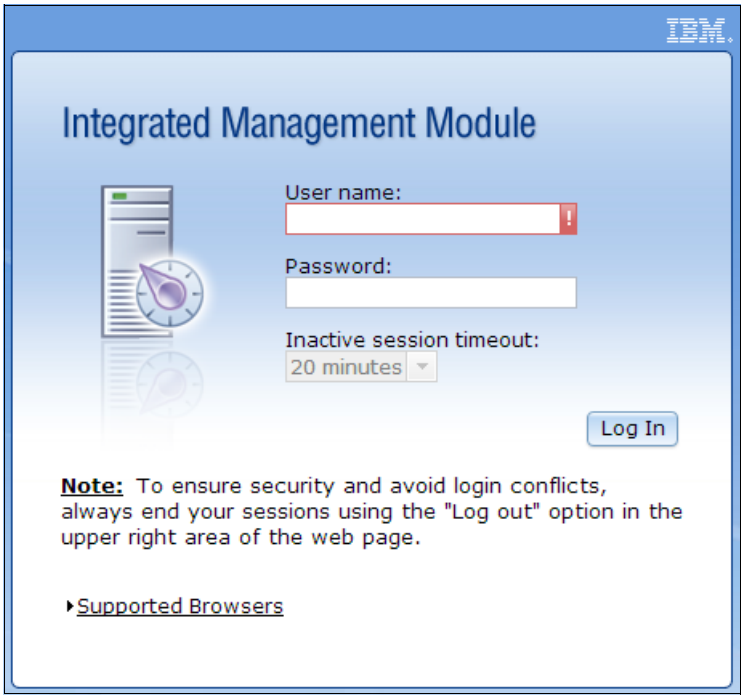

Figure 4-1 Log in to the IMM interface

After you provide the appropriate credentials, the IMM home window opens.

Your certificates must know what time it is: Before you continue with the rest of this process, it is critical that you set up the date and time for the IMM, whether manually or through a Network Time Protocol (NTP) server. This time must be correct or the certificates that are generated do not work or can provide problematic connections.

If the CMOS settings on a server are cleared or the system board is replaced on a server, this time must be verified to be accurate or the IMM cannot communicate with the SKLM server. This issue can result in the server becoming unable to access the encrypted drives until the issue is resolved.

The date and time on the server Unified Extensible Firmware Interface (UEFI), which is reported to the operating system, does not use the same clock as the IMM. The clock on the IMM is a separate device that is used by the IMM only. Figure 4-2 shows the fields that must be verified.

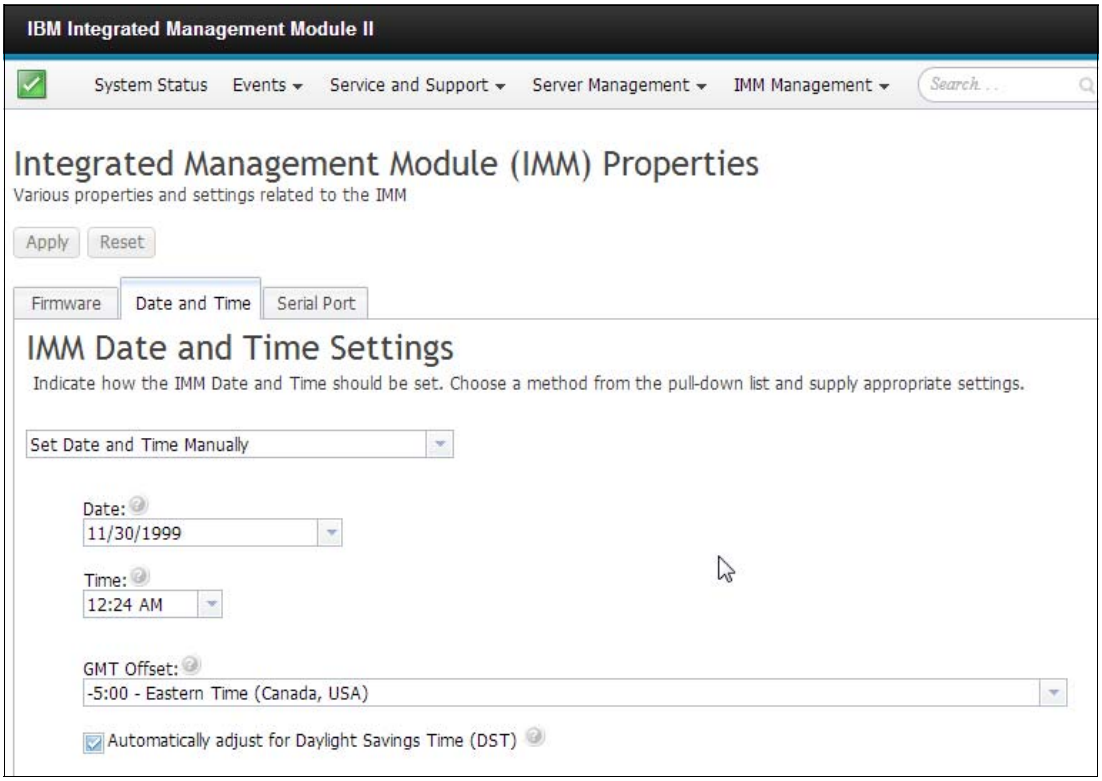

Figure 4-2 Ensure correct time and date

3. Verify that the appropriate license or Features on Demand (FoD) key are installed on the server by browsing to the IMM Management pull-down menu and select **Security**, as shown in Figure 4-3.

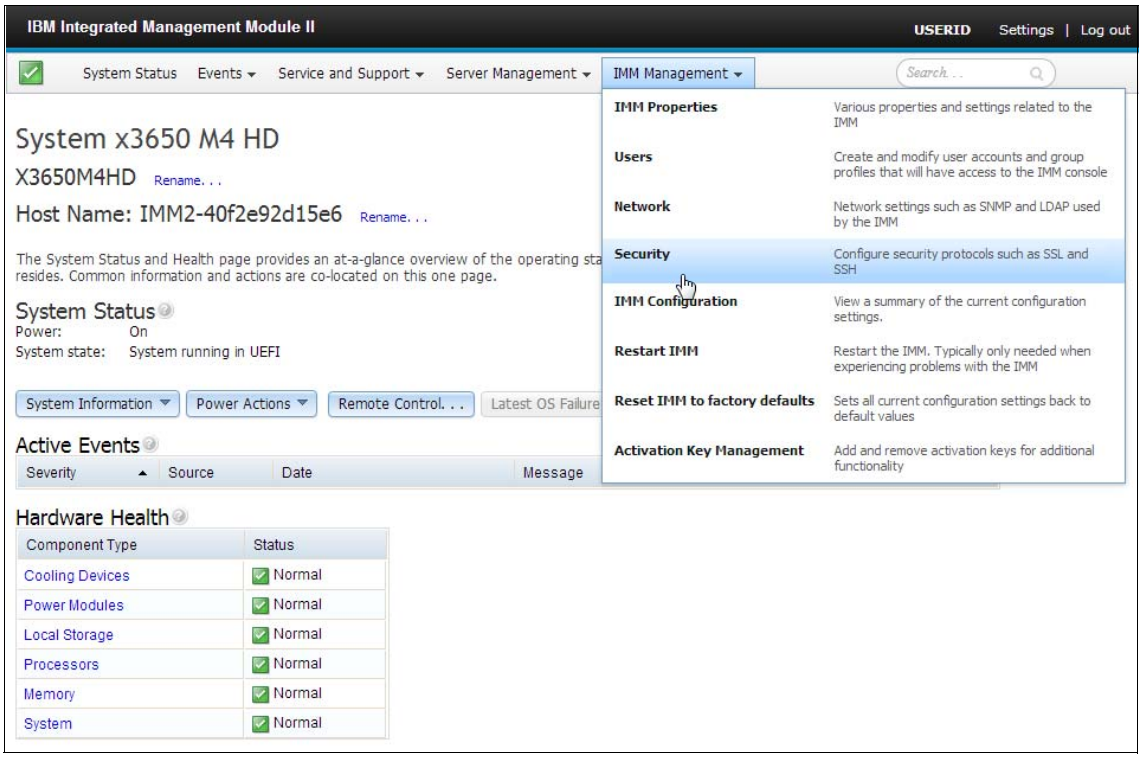

Figure 4-3 Access the Security configuration menu

4. If the Drive Access tab is missing from the IMM Security window (as shown in Figure 4-4 on page 90), you must install the FoD activation key for external key management on the server. This issue can occur if the FoD activation key was not installed on the IMM or was not restored if the system board was replaced.

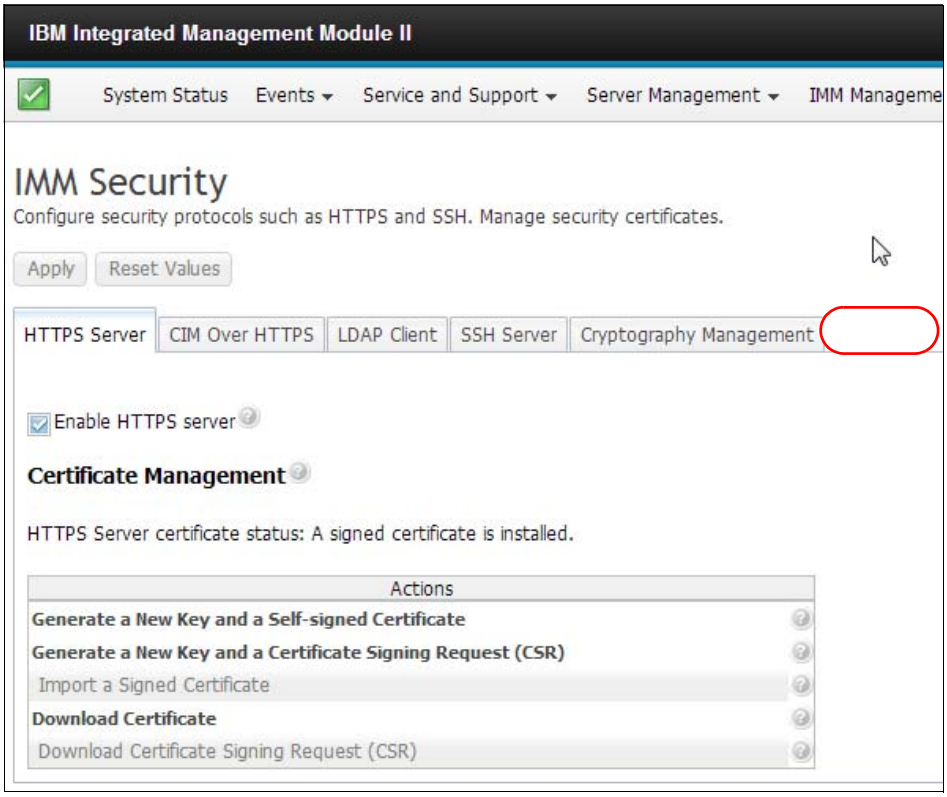

Figure 4-4 Verify FoD key Installation

If the Drive Access tab is present, you can skip 4.2.2, “Installing the FoD activation key” on page 90 and proceed to 4.2.3, “Creating a self-signed certificate” on page 92.

4.2.2 Installing the FoD activation key

If the server does not have the appropriate FoD key installed to allow for the configuration of an external key management server, you must provide this key to activate the functionality. If the server was licensed but had the IMM replaced because of service, the FoD activation key must be recovered from a backup or you might need to contact support to replace the key.

If you do not have experience with FoD key management, several resources are available, such as the Lenovo Press paper *Using Features on Demand*, REDP-4895, available from:

`http://lenovopress.com/redp4895`

The following methods also are available to assist with the management of FoD keys:

- IBM Integrated Management Module II (IMM2)

  By using this server-based management interface, users can install and remove FoD activation keys. The interface can be accessed by using a web browser, command line, or ASU.

- IBM Systems Director

  The Systems Director is a centrally managed FoD functionality with which users can download, install, and manage activation keys:

  `http://ibm.com/systems/software/director/`

- IBM ToolsCenter™

The IBM ToolsCenter is a collection of server management tools to help manage your System x and BladeServer environment. The IBM ToolsCenter provides a download portal for server management tools, such as DSA and ASU:

`http://ibm.com/support/entry/portal/docdisplay?lndocid=TOOL-CENTER`

- IBM Dynamic System Analysis™ (DSA)

  The DSA is an operating system or pre-boot tool with which users can install and manage FoD activation keys:

  `http://ibm.com/support/entry/portal/docdisplay?lndocid=SERV-DSA`

- Advanced Settings Utility

  The ASU is a command line-based utility with which users can install and manage FoD activation keys:

  `http://ibm.com/support/entry/portal/docdisplay?lndocid=TOOL-ASU`

After the FoD is installed, select the **Drive Access** tab to continue, as shown in Figure 4-5.

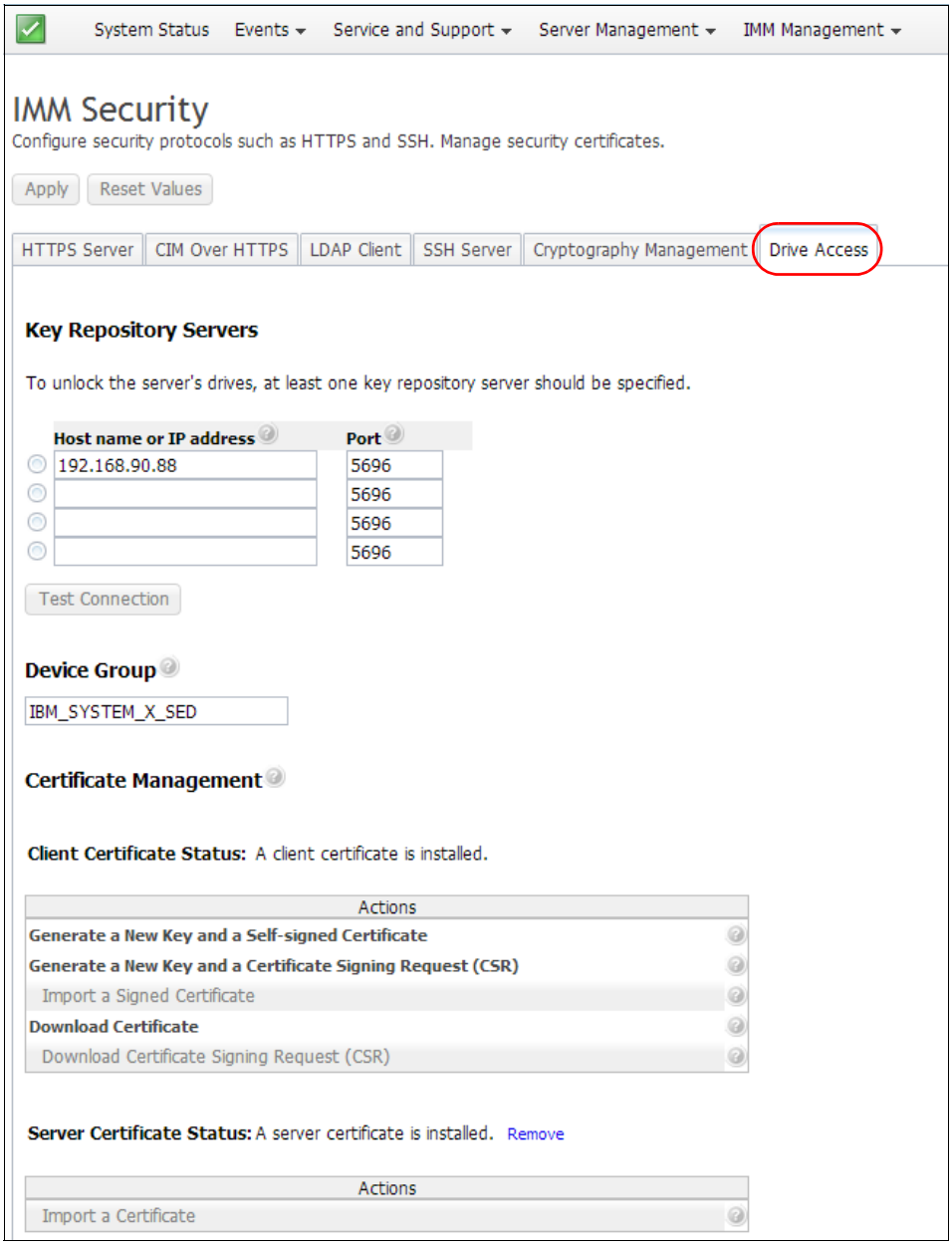

Figure 4-5 Accessing Drive Access tab

4.2.3 Creating a self-signed certificate

From the perspective of the SKLM key manager, endpoint devices (such as a System x server) that request keys are considered clients.

The target System x server with self-encrypting drives (SEDs) might have a client certificate that is configured on the IMM. At the time of writing, servers that come directly from the manufacturing facility do not have a client certificate present. Adding a preloaded certificate at the point of manufacture might change in the future.

If Download Certificate is disabled on the Drive Access tab, a certificate must be generated to continue.

Complete the following steps to create a self-signed certificate:

1. Select **Generate a New Key and a Self-signed Certificate**, as shown in Figure 4-6 on page 93.

> **Nomenclature:** The keys that are referenced in the IMM web interface are public/private key pairs as used with certificates.

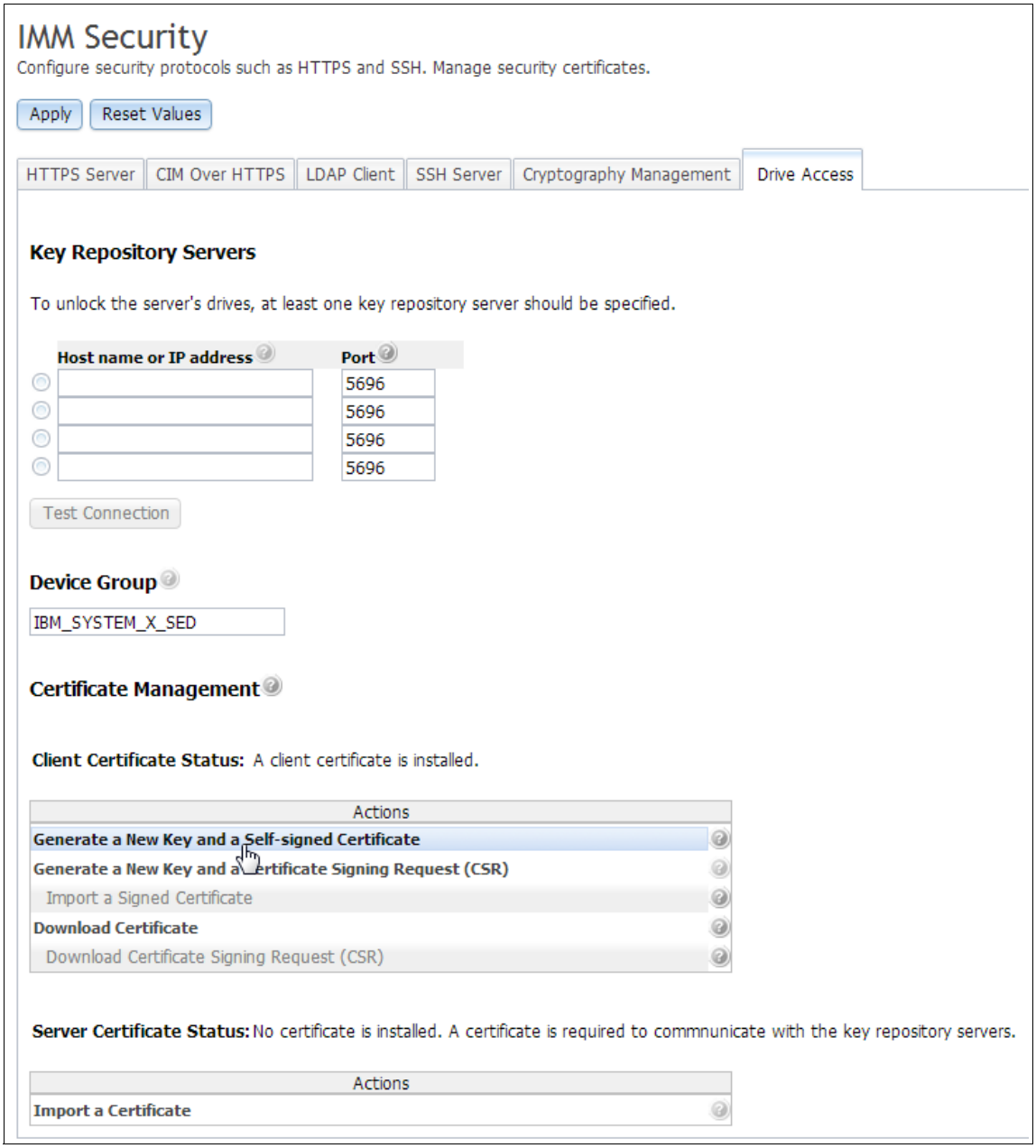

Figure 4-6 Generate self-signed certificate

2. In the certificate generation panel, ensure that you complete the appropriate fields. Of special importance is the IMM Host Name field. The IMM Host Name must match the URL

that is used to access the IMM (fully qualified name or IP address). In our example configuration, we specified the IP address of the server.

Figure 4-7 shows how the fields were completed for our test system in our example.

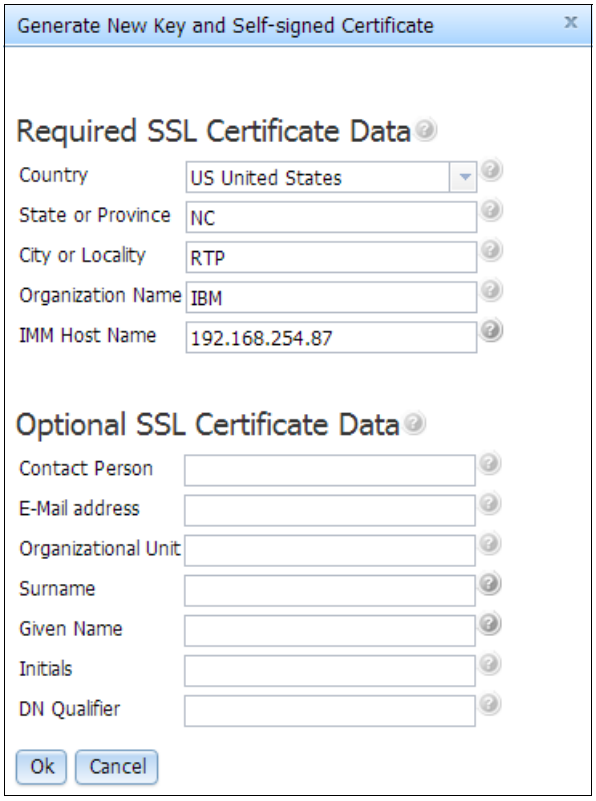

Figure 4-7 Self-signed certificate panel

3. After the certificate is generated, select **Download Certificate** to create a local copy of the certificate file. This file must be uploaded to the SKLM server as described in Chapter 3, “IBM Security Key Lifecycle Manager setup” on page 31.

   Figure 4-8 shows the status field, which indicates that a certificate was successfully created.

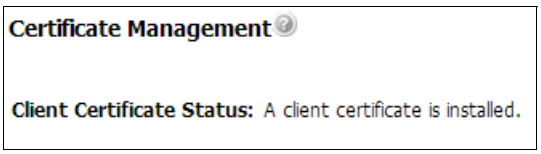

Figure 4-8 Client certificate created

The downloaded certificate must be copied to a local file store on the SKLM server because SKLM does not support importing keys from non-local storage medium. For example, you cannot reference a network share from the SKLM server to import the certificate. In our test configuration, we created a network share on the SKLM server where we copied the certificates to as they were created.

You also must ensure that you provide relevant names to the certificate files when they are stored. In our configuration, we used the machine type and serial number of the server that created the certificate as the file name.

Figure 4-9 shows the appropriate area of the configuration page on the IMM to select the certificate download option.

Figure 4-9 Downloading IMM Certificate

4. Select **Download Certificate**.

4.2.4 Generating a Certificate Signing Request

If your environment requires the use of a certificate signing authority, use the Certificate Signing Request (CSR) request option instead of the self-signed certificate to create a CSR file that can be saved to the local system and authenticated with the signing authority.

As shown in Figure 4-10, the dialog box is identical to the self-signed certificate that we described in 4.2.3, “Creating a self-signed certificate” on page 92 with the addition of the requirements for credentials to generate the CSR.

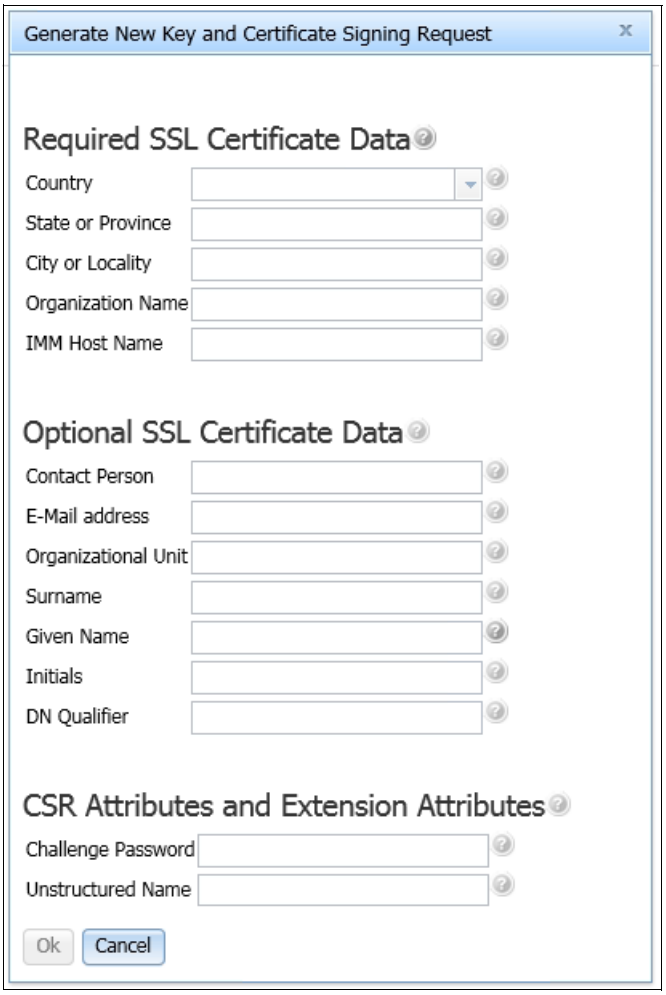

Figure 4-10 CSR request form

4.2.5 Downloading the Certificate Signing Request

After the CSR request form is completed, the option to download the CSR request becomes available. Select **Download Certificate Signing Request** to start downloading the file, as shown in Figure 4-11. You then sign the certificate with the certification authority.

Figure 4-11 Download CSR request file

4.2.6 Importing a signed certificate

This option is disabled by default and becomes available only after a CSR request is generated. The signed certificate that you upload must correspond with the CSR that was generated as described in 4.2.4, “Generating a Certificate Signing Request” on page 95.

After the CSR is signed with the certificate authority, the next step is to take the resulting certificate file and upload it to the IMM. To upload the certificate, select **Import a Signed Certificate** and follow the prompts to import the result of a CSR, as shown in Figure 4-12. The file that is uploaded to the IMM must be the same file that you uploaded to the SKLM server to ensure that they are identical.

Figure 4-12 Import signed certificate

4.2.7 Importing SKLM server certificate

After a client-side certificate is created or uploaded to the IMM, the next step is to import the certificate that was generated on the SKLM server. For more information about the steps to create this certificate, see Chapter 3, “IBM Security Key Lifecycle Manager setup” on page 31.

To import the certificate, select **Import Certificate** in the Server Certificate section of the IMM interface configuration page, as shown in Figure 4-13.

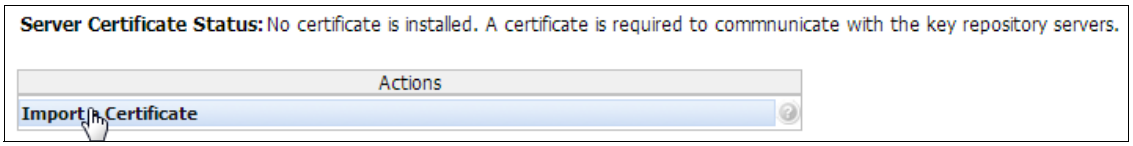

Figure 4-13 Import Server Certificate

In the next window, choose **Select Certificate File...** and browse to the appropriate SKLM certificate file. Then, select **OK** to import the certificate, as shown in Figure 4-14.

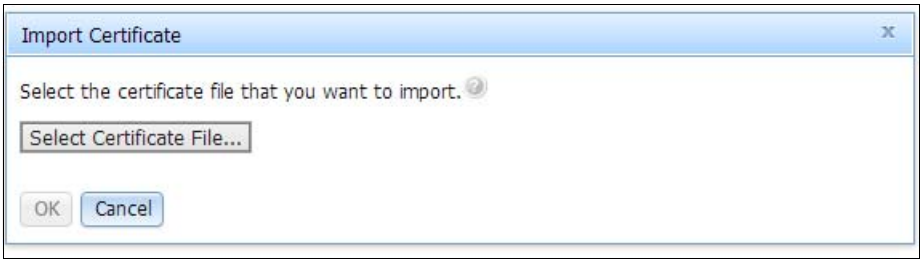

Figure 4-14 Select Certificate File

After the upload process is complete, the Server Certificate Status updates to reflect that the certificate is now installed, as shown in Figure 4-15.

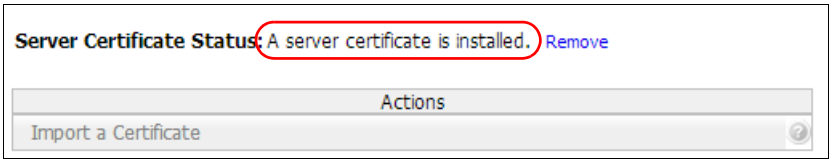

Figure 4-15 Server Certificate Installed

4.2.8 Configure the device group

Unless a custom group is created in SKLM (which is outside the scope of this publication), you must use the default device group IBM_SYSTEM_X_SED, which is entered in the Device Group field, as shown in Figure 4-16 on page 98.

If you created a custom group within the SKLM server to manage groups of System x servers, you must update the group name in this field in place of the default value, as shown in Figure 4-16 on page 98.

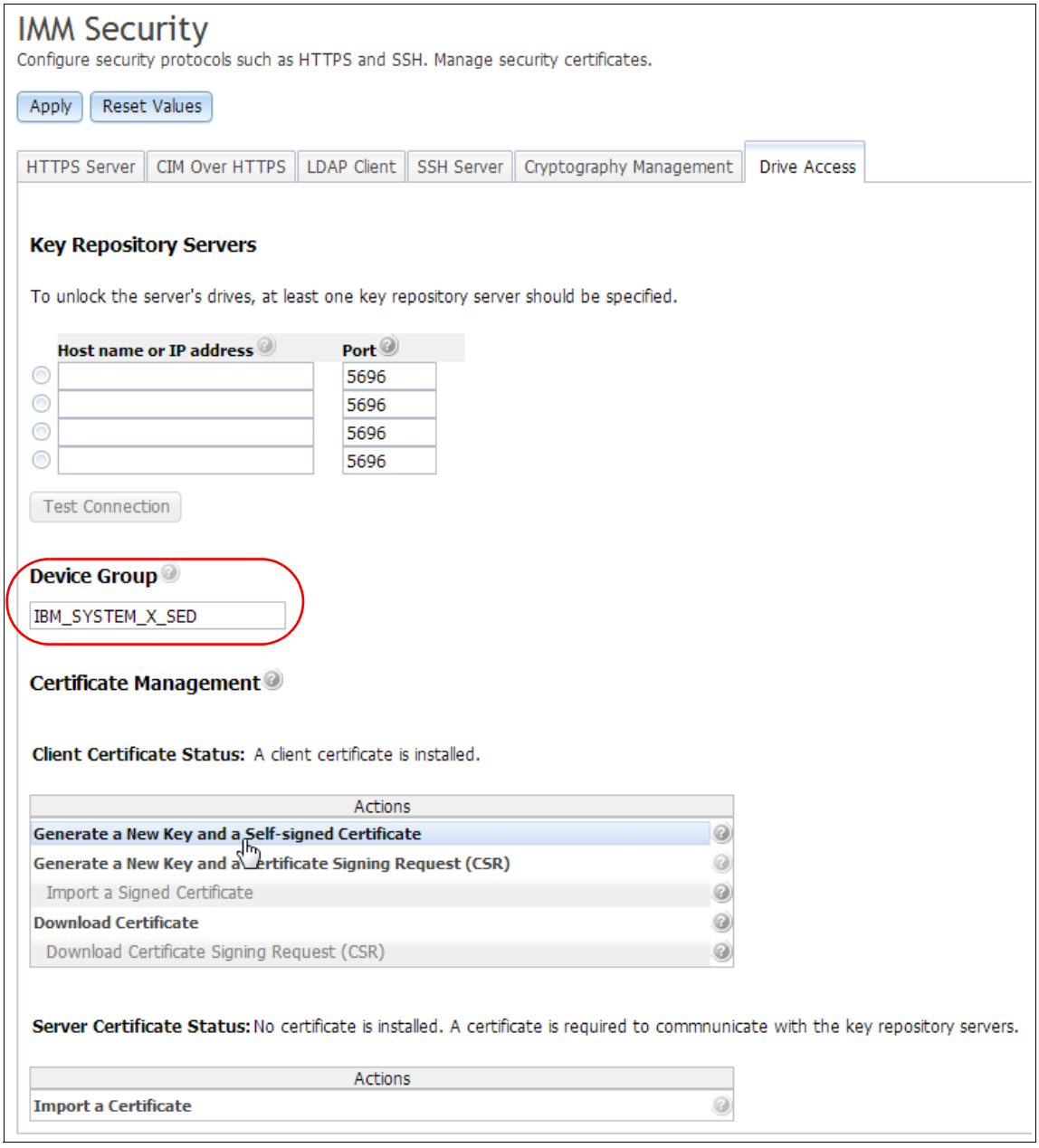

Figure 4-16 Default Device Group

4.2.9 Configuring key repository (SKLM) servers

After all other sections are complete, you must configure the key repository servers that the IMM connects to at boot time to request the required KEK key to unlock the SEDs.

In our sample configuration, we use a single SKLM server for simplicity. For a production environment, it is recommended that two SKLM servers at the minimum are used in a redundant configuration. If the IMM cannot connect to a key management server during boot, the server cannot access any encrypted drives by design. This configuration is used to prevent access to data on a server that was removed from the corporate network that is hosting the key management server.

If two or more key management servers are present in the environment (up to a maximum of four), the servers must be configured in the appropriate fields, as shown in Figure 4-17.

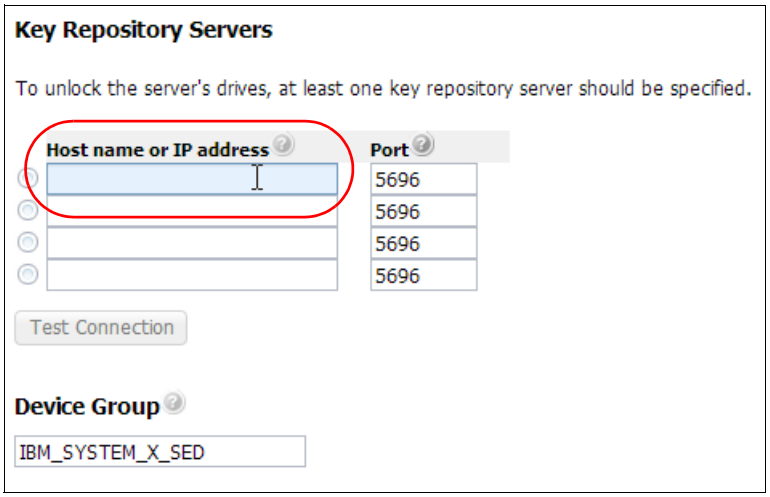

Figure 4-17 Key Repository Server configuration

4.2.10 Test the connection to SKLM

The last step in the process is to test the connection from the IMM to the key management server.

You must test each target server individually to ensure that all servers have the appropriate certificates installed and can be contacted through the network.

Before you test the connections, select **Apply** at the top of the window to update all of the settings to the IMM. After the apply process is complete, select the radio button to the left of the server connection that you want to test. Then, select **Test Connection**, as shown in Figure 4-18.

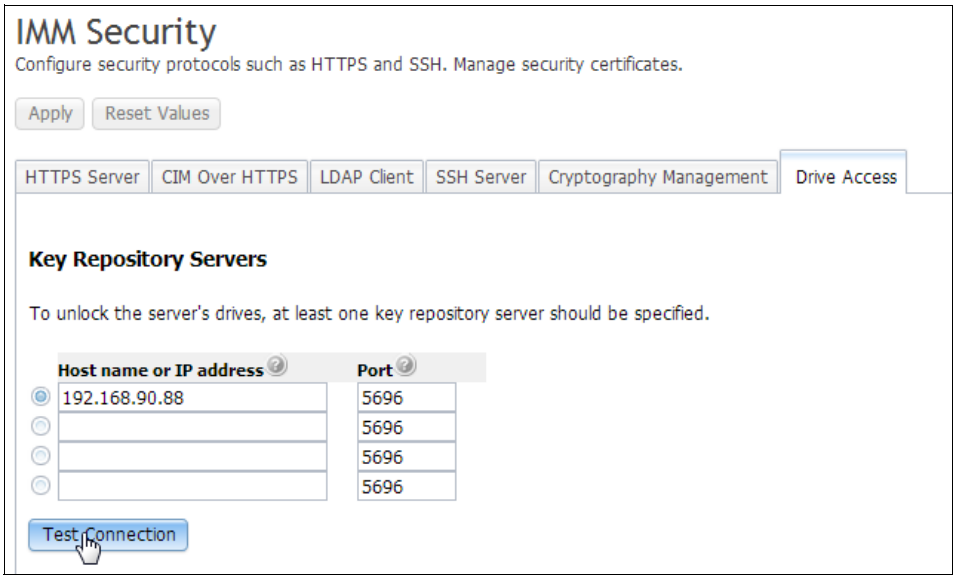

Figure 4-18 Test connections

If you do not receive a response, confirm that you selected **Apply** and the appropriate radio button before repeating the test. If you correctly configured the IMM, you receive a success message as shown in Figure 4-19.

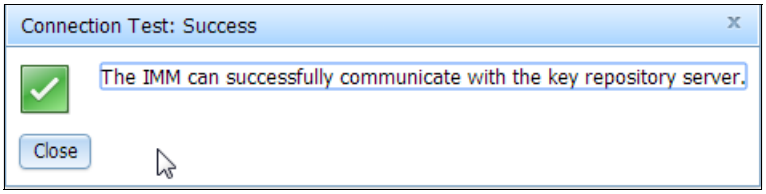

Figure 4-19 Successful connection test

4.2.11 Troubleshooting

During our test, we encountered the error that is shown in Figure 4-20 on one of our test systems. This error is the result of attempting to upload a certificate file from the key management server to the IMM when the IMM has an invalid time and date configured. Correct the time and date as described in 4.2.1, “Accessing the IMM web interface” on page 86. Then, try to upload the certificate again.

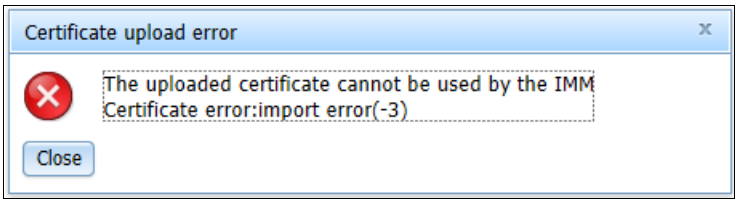

Figure 4-20 Certificate upload error

If you receive an error when you are attempting to configure the key repository server where the settings fail to apply, the workaround is to configure the repository server entries as described in 4.3.1, “Initial setup” on page 101.

After all previous steps in this chapter are completed successfully, reboot the Server to enter the UEFI configuration.

4.3 Configuring the IMM by using the IMM Command Line Interface

In this section, we repeat the same configuration steps as described in 4.2, “Configuring the IMM by using the web-based interface” on page 86. However, now we use the command-line capabilities of the IMM v2 adapter.

4.3.1 Initial setup

If the IMM was not manually configured, it is at a DHCP-provided address if a DHCP server was available when power was applied to the server or the default IP address of `192.168.70.125`.

We recommend that you set a static or reserved IP address for the IMM now before you continue to the next steps because you must use the IP address of the IMM in most of the following procedures. For more information about configuring the IMM, see this website:

`http://www.ibm.com/support/entry/portal/docdisplay?lndocid=MIGR-5086346`

Your certificates must know what time it is: Before you continue with the rest of this procedure, it is critical that you set up the date and time for the IMM, whether manually or through an NTP server. This time must be correct or the certificates that are generated do not work or provide problematic connections.

If the CMOS settings on a server are cleared or the system board is replaced on a server, this time must be verified to be accurate or the IMM cannot communicate with the SKLM server. This issue can result in the server becoming unable to access the encrypted drives until the issue is resolved.

The date and time on the server UEFI (which is reported to the operating system) does not use the same clock as the IMM. The clock on the IMM is a separate device that is used by the IMM only.

4.3.2 Installing FoD activation key

If the server does not have the appropriate FoD key installed to allow for the configuration of an external key management server, you must provide this key to activate the functionality. If the server was licensed but had the IMM replaced because of service, the FoD activation key must be recovered from a backup or you might need to contact support to have the key replaced.

If you do not have experience with FoD key management, see the resources that are listed in 4.2.2, “Installing the FoD activation key” on page 90.

To activate an FoD key by using the IMM command line, use the following **`keycfg`** command to display, add, or delete activation keys:

```
keycfg
   -add
      -ip tftp ip address
```

```
   -pn port number (of tftp/sftp server - default 69/22)
   -u username (for sftp server)
   -pw password (for sftp server)
   -f filename
-del  n (where n is a valid ID number from listing)
-deltype x (where x is a Type value)
```

4.3.3 Creating a self-signed certificate

From the perspective of the SKLM key manager, endpoint devices (such as a System x server) that request keys are considered clients.

A System x server with SEDs might have a client certificate that is configured on the IMM. At the time of this writing, servers that come directly from the manufacturing facility do not have a client certificate present. Adding a preloaded certificate at the point of manufacture might change in the future.

If the certificate is present, you can skip the section about creating certificates and proceed to 4.3.6, “Importing SKLM server certificate” on page 104.

You can check whether there is an IMM client certificate in place by using the following command:

```
sslcfg –client
```

If the result of the command indicates the status as `enabled`, a certificate is installed. You can re-create the certificate, even if one is present.

To create a self-signed certificate by using the IMM command line, run the following **`sslcfg`** command:

```
sslcfg [-options]
```

The following options are available:

- `-server: SSL Server status (enabled, disabled)`

  SSL can be enabled only if a certificate is in place.

- `-client: SSL Client status (enabled, disabled)`

  SSL can be enabled if a server or client certificate is in place.

- `-cim: CIM over HTTPS status (enabled, disabled)`

  SSL can be enabled if a server or client certificate is in place.

- `-cert: Generate a self-signed certificate (server, client, cim, storekey)`
- `-csr: Generate a CSR (server, client, cim, storekey)`
- `-csrform: The format of the CSR will be exported in (der, pem)`
- `-i: IP address for TFTP/SFTP server when uploading a certificate,`

Use the following command to download a certificate or CSR:

```
-pn: port number (of tftp/sftp server - default 69/22)
-u: username (for sftp server)
-pw: password (for sftp server)
-l: filename (when downloading or uploading a certificate or CSR)
```

If not specified during download, the default name for that file is used and displayed, as shown in the following example:

```
-dnld: Downloads the specified file
```

This option takes no arguments, but must be used with the following command:

```
-cert/-csr (server/client/cim/storekey), as well as -i (and optionally -l)
-upld: Imports the specified certificate
```

This option takes no arguments, but must be used with the following command:

```
-cert (server/client/cim/storekey) and -i and -l
-tcx: Trusted certificate x for the ssl client (x = 1, 2, 3 or 4) (import,
download, remove)
```

The following options are required for generating a self-signed certificate or CSR:

```
-c:  Country (2 letter code)
-sp:  Quote-delimited State or Province (max 60 chars)
-cl:  Quote-delimited City or Locality (max 50 chars)
-on:  Quote-delimited Organization Name (max 60 chars)
-hn:  IMM hostname (max 60 chars)
```

The following options are available for generating a self-signed certificate or CSR:

```
-cp:  Quote-delimited Contact Person (max 60 chars)
-ea:  Email Address (max 60 chars)
-ou:  Quote-delimited Organizational Unit (max 60 chars)
-s:  Quote-delimited Surname (max 60 chars)
-gn:  Quote-delimited Given Name (max 60 chars)
-in:  Quote-delimited Initials (max 20 chars)
-dq:  Quote-delimited DN Qualifier (max 60 chars)
```

The following options are available for generating a CSR:

```
-cpwd: Challenge Password (min 6 chars, max 30 chars)
-un:  Quote-delimited Unstructured Name (max 60 chars)
```

In our example configuration, the following command was used:

```
sslcfg –cert –c:US –sp:NC –cl:RTP –on:IBM –hn:192.168.254.87
```

After the self-signed certificate is created, download it by using the following command:

```
sslcfg –dnld –ip <IP address of tftp server> -l <filename to save file> -cert
```

4.3.4 Generating a CSR

If your environment requires the use of a certificate signing authority, use the CSR request option instead of the self-signed certificate to create a CSR file that can be saved to the local system and authenticated with the signing authority.

To generate a CSR request file, we used the following **sslcfg** command:

```
sslcfg –csr –c:US –sp:NC –cl:RTP –on:IBM –hn:192.168.254.87 –csrform:der
```

After the CSR request file is created, use the following command to download it (substitute the appropriate values):

```
sslcfg –dnld –ip <IP address of tftp server> -l <filename to save file> -csr
```

4.3.5 Importing a signed certificate

This option can be used to upload a signed certificate to the IMM after the CSR that was created in 4.3.4, “Generating a CSR” on page 103 is signed by a certificate authority. The signed certificate that you upload must correspond with the CSR that was generated.

After the CSR is signed with the certificate authority, it must be uploaded to the IMM by using the following command:

```
sslcfg –upld –ip <IP address of tftp server> -l <filename to upload> -cert
```

4.3.6 Importing SKLM server certificate

After a client-side certificate is created or uploaded to the IMM, the next step is to import the certificate that was generated on the SKLM server. For more information about the steps that are used to create this certificate, see Chapter 3, “IBM Security Key Lifecycle Manager setup” on page 31.

Use the **storekeycfg** command to upload the certificate that is generated by the key management server. The command features the following syntax:

```
storekeycfg
   -add
      -ip tftp/sftp ip address
      -pn port number of tftp/sftp server (default 69/22)
      -u username (for sftp server)
      -pw password (for sftp server)
      -f filename
      -del
      -dgrp <device group> (device group name)
      -sxip <host name/ip_addr> (server x host name/ip addr
         (x can be 1, 2, 3 or 4))
      -sxpn <port_number> (server x port number
         (x can be 1, 2, 3 or 4))
      -testx (test server x connection (x can be 1, 2, 3 or 4))
```

Use the following example command to upload a server key to the IMM:

```
storekeycfg –add –ip <tftp or sftp server address> -u <username if sftp> -pw
<username if sftp> -f <filename of certificate to upload>
```

In our example, we used the following command:

```
storekeycfg –add –ip 1.2.3.4 –u username –pw password –f certificate.der
```

4.3.7 Configuring the device group

Unless a custom group is created in SKLM (which is outside the scope of this publication), you should use the default device group IBM_SYSTEM_X_SED, which is also populated by using the storekeycfg command.

If you created a custom group within the SKLM server to manage groups of System x servers, you must update the group name by using the following command:

```
storekeycfg –dgrp NEW_DEVICE_GROUP
```

4.3.8 Configuring key repository (SKLM) servers

After all other sections are complete, you must configure the key repository servers that the IMM connects to at boot time to request the required KEK key that is needed to unlock the SEDs.

In our sample configuration, we use a single SKLM server for simplicity. For a production environment, it is recommended that two SKLM servers at the minimum are used in a redundant configuration. If the IMM cannot connect to a key management server during boot, the server cannot access any encrypted drives by design. This configuration prevents the access of data on a server that was removed from the corporate network that is hosting the key management server.

If two or more key management servers are present in the environment (up to a maximum of four), they be configured by repeating the following steps for each of the key management target servers.

This example command sets the first key management server to `192.168.90.88` as required by our sample configuration, as shown in the following example:

```
storekeycfg –s1ip 192.168.90.88
```

If there are other key management servers in the environment, repeat this command for each of the other servers, substituting the `1` with the server entry you want to change.

For example, to add a second key management server, you use the following command:

```
storekeycfg –s2ip 192.168.90.89
```

4.3.9 Testing the connection to SKLM

After you complete all of the preceding steps, you must test the connection from the IMM to each of the configured key management servers.

To test the first connection, run the following command:

```
storekeycfg –test1
```

To test any other configured key management servers, repeat the command, substituting the `1` for the server entry you want to test. For example, to test the connection to a second configured key management server, run the following command:

```
storekeycfg –test2
```

This command results in the following response:

```
Operation completed successfully.
```

4.4 Configuring the IMM by using the Advanced System Utility

In this section, we describe the same procedures that were used in 4.2, “Configuring the IMM by using the web-based interface” on page 86 and 4.3, “Configuring the IMM by using the IMM Command Line Interface” on page 101. However, in this section, we use ASU commands that can be scripted and automated for configuring the IMMs of remote servers.

Also, when ASU commands are used to configure a remote host, add the following options to the command lines:

```
-host <IP address of IMM> -user <username (default: USERID)> -password <password
(default: PASSSWORD)>
```

4.4.1 Creating a self-signed certificate

Before proceeding, verify whether the IMM of the target server has a client certificate.

Use the **asu show** command to view the status of a particular certificate.

At the command line, enter the following command:

```
asu show IMM.SSL_HTTPS_SERVER_CERT
```

The command results in following output:

```
IMM.SSL_HTTPS_SERVER_CERT=Private Key and CA-signed cert installed, Private Key
stored, CSR available for download.
```

If the result is that a certificate is installed, the certificate does not need to be re-created and can be downloaded as described in 4.4.2, “Generating a CSR” on page 106.

If the result is that a certificate is not installed, a certificate must be created. You can use the **asu** command to generate a self-signed certificate, which is certificate that is signed.

At the command line, enter the following command:

```
asu generate IMM.SSL_HTTPS_SERVER_CERT asu.xml
```

The command results in following output:

```
Certificate was generated successfully!
```

4.4.2 Generating a CSR

You can use the following command to generate a CSR request file that can be downloaded from the IMM and signed by using a certificate signing authority (if required):

```
asu generate IMM.SSL_HTTPS_SERVER_CSR asu.xml
```

The command results in the following output:

```
Certificate was generated successfully!
```

After the CSR certificate request is created successfully, the next step is to download it to make it available for signing against a signing authority. This process is done by using the following command:

```
asu export IMM.SSL_HTTPS_SERVER_CSR asu_csr.der
```

The command results in the following output:

```
Certificate was exported successfully!
```

The `asu_csr.der` file is saved in the current working directory from which the **asu** command was run.

You can export a certificate or a certificate sign request. If a certificate sign request is signed by an independent certificate authority, it is referred to as a CA-signed certificate.

4.4.3 Importing a signed certificate

After you export a certificate (as described in 4.4.2, “Generating a CSR” on page 106), you must sign it by using an independent certificate authority. You can import the CA-signed certificate only (which is different than a self-signed certificate) into the IMM by using the ASU tool.

For example, to upload the results of signing a CSR, enter the following command:

```
asu import IMM.SSL_SKR_CLIENT_CERT asu_cert.der
```

The command results in the following output:

```
Certificate was imported successfully!
```

If a signed certificate is installed on the IMM as shown in 4.4.1, “Creating a self-signed certificate” on page 106, the existing certificate must be deleted before you can upload a new certificate.

The certificate `asu_cert.der` is a CA-signed certificate after `asu_csr.der` is signed by using your own certificate authority.

4.4.4 Importing SKLM server certificate

To import the SKLM server certificate, use the same process that is described in 4.4.3, “Importing a signed certificate” on page 107. The only difference in this instance is to specify that you are importing a certificate for the key management server. Therefore, you must substitute the `IMM.SSL_CLIENT_TRUSTED_CERT_SKR` in the command line that is used to import CSR signed certificate, as shown in the following example:

```
asu import IMM.SSL_CLIENT_TRUSTED_CERT_SKR ISKLM_Server_Cert.der
```

4.4.5 Configuring key repository servers

Up to four key repository servers can be configured within the IMM. Use the following command to see what is configured on the target IMM:

```
asu show –host <IP Address> -user <username> -password <password>
```

In the resulting output, you find the following fields:

```
IMM.SKR_Server1_HostName_IPAddress=192.168.90.87
IMM.SKR_Server1_Port=5696
IMM.SKR_Server2_HostName_IPAddress=
IMM.SKR_Server2_Port=5696
IMM.SKR_Server3_HostName_IPAddress=
IMM.SKR_Server3_Port=5696
IMM.SKR_Server4_HostName_IPAddress=
IMM.SKR_Server4_Port=5696
```

In this example, a single target key management server was configured. Use the following command to modify these settings:

```
asu set ASU IMM.SKR_Server1_HostName_IPAddress=<ip address> –host <IP Address>
-user <username> -password <password>
```

4.4.6 Configuring the device group

Use the following command to view the current device user group:

```
asu show IMM.SKR_DEVICE_GROUP –host <IP Address> -user <username> -password
<password>
```

The command results in the following output:

```
IMM.SKR_DEVICE_GROUP=IBM_SYSTEM_X_SED
```

The default device user group is `IBM_SYSTEM_X_SED` and should be left at the default unless you configured a different group on the key management server.

If required, you can change the default group by using the following command:

```
asu set IMM.SKR_DEVICE_GROUP= <new group name> –host <IP Address> -user
<username> -password <password>
```

4.5 Conclusion

In this chapter, we described three different methods to configure the IMM on the target server: the web interface, IMM command line, and the ASU tool.

We recommend that for any large deployment, you consider the use of the ASU method because you can use it to script the configurations. This ability allows for the simplification of deployments of large distributed configurations and the automation of many of the settings, which often are consistent across servers, such as the addresses of the key management servers.

5

UEFI configuration

In this chapter, we describe the configuration for the Unified Extensible Firmware Interface (UEFI) of the target server. This description includes configuring a basic RAID set as an example and enabling the encryption functions of the RAID adapter.

This chapter includes the following topics:

- Enabling storage controller encryption
- Configuring virtual disks

© Copyright Lenovo 2015. All rights reserved.

5.1 Enabling storage controller encryption

By default, encryption is disabled on IBM RAID adapters. The following modes of encryption can be enabled on the adapter:

- Local encryption key management in which the RAID adapter manages and maintains the key encryption key (KEK) that is used to encrypt the local media encryption key (MEK), which is stored on the drive. For more information, see Chapter 1, “Technology primer” on page 1.
- Configure the adapter to request a KEK from an external key management server, such as IBM Security Key Lifecycle Manager (SKLM) at boot time.

In this section, we describe the following tasks:

- Setting the adapter for an external key management server
- Accepting pending request on the SKLM server

5.1.1 Setting the adapter for an external key management server

To configure the adapter for an external key management server (EKMS) complete the following steps:

1. At the initial power-on of the server, select **F1** when you are prompted to do so at the UEFI configuration window, as shown in Figure 5-1 on page 110.

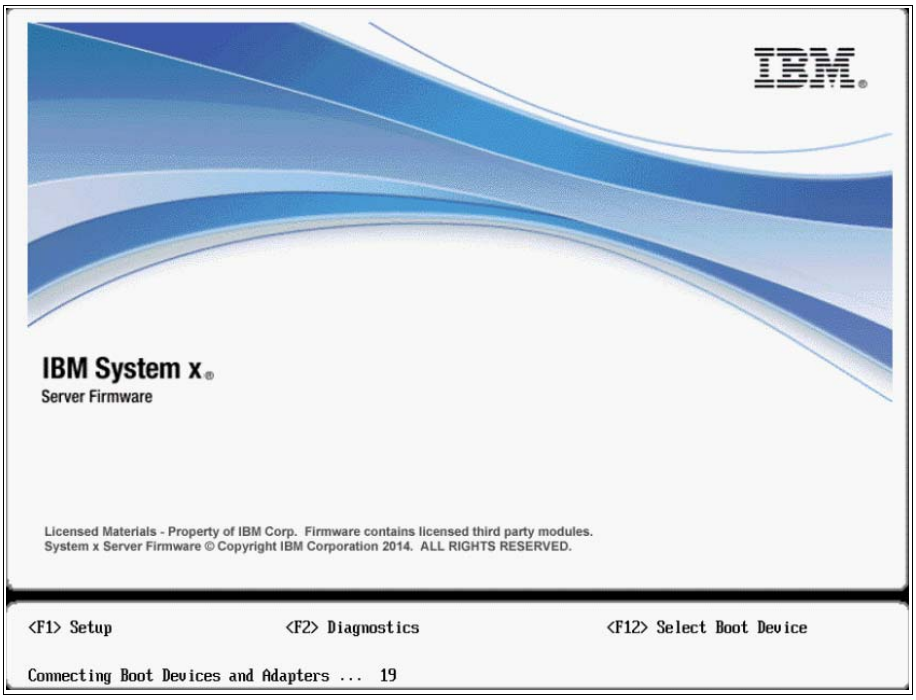

Figure 5-1 Initial UEFI welcome window

2. The System Configuration and Boot Management window opens. Select **System Settings**, as shown in Figure 5-2.

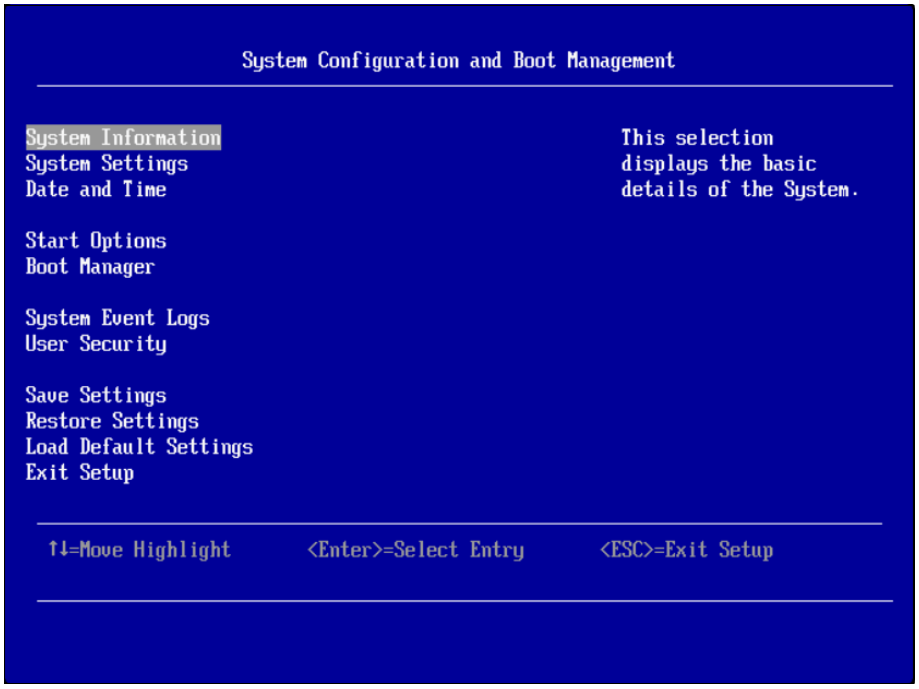

Figure 5-2 System Configuration and Boot Management

3. In the System Settings window, select **Storage,** as shown in Figure 5-3.

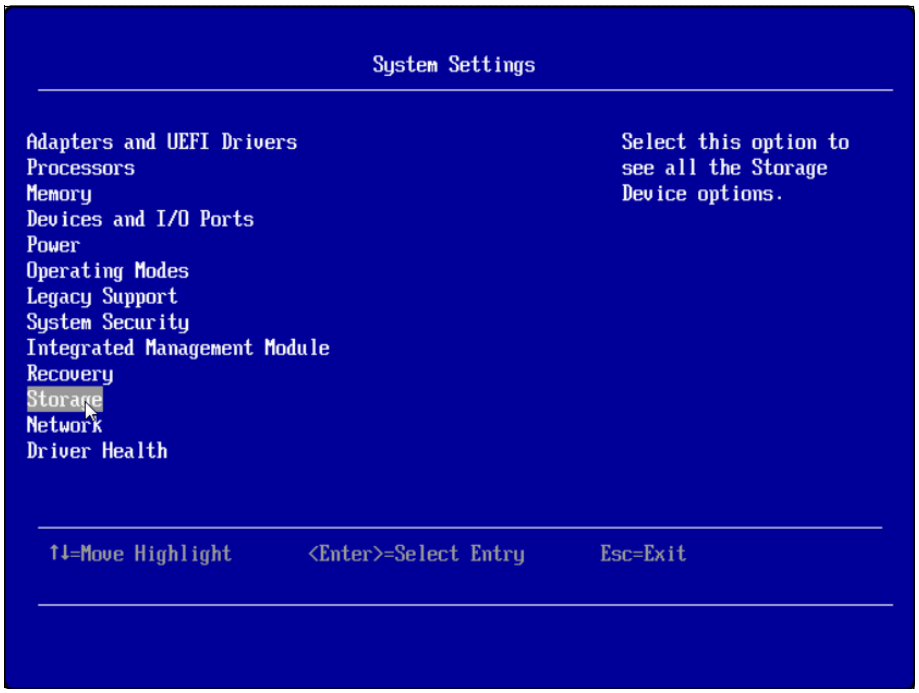

Figure 5-3 Selecting the Storage option

4. In the next window, you must select the appropriate RAID adapter that manages the self-encrypting drives (SEDs). If several adapters are installed in the server and they are all managing SEDs, you must repeat these steps for each adapter that is installed in the system.

In our test configuration, we installed a single M5210 RAID adapter in an x3650 M4 HD server. The Storage window opens, as shown in Figure 5-4. Select the highlighted adapter.

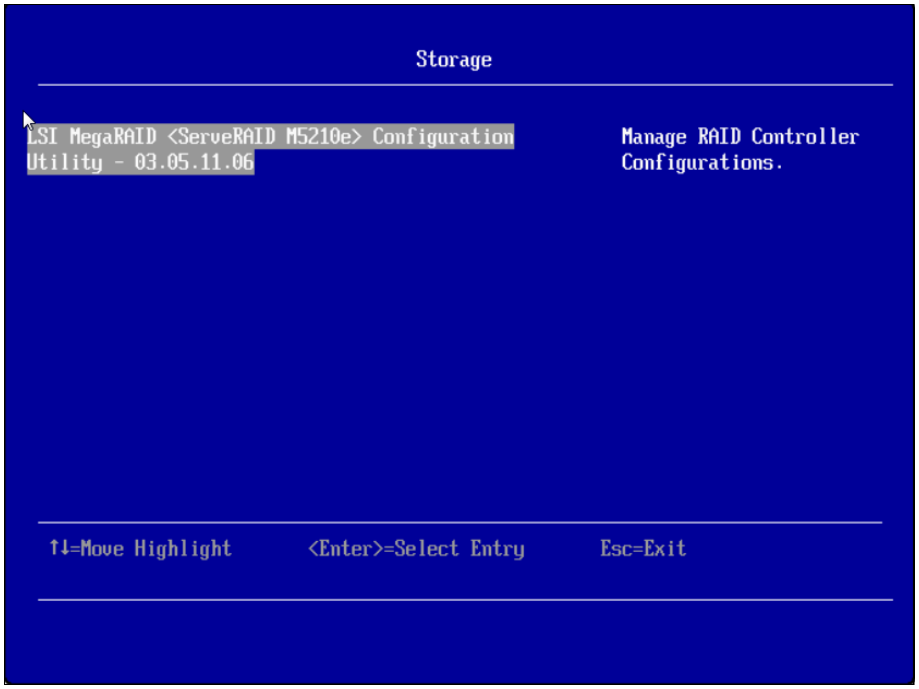

Figure 5-4 RAID adapter selection

5. Select the **Controller Management** option, as shown in Figure 5-5.

Figure 5-5 Selecting Controller Management

6. In the Controller Management window, you configure the selected adapter for an EKMS source. Scroll to the bottom of the list of options and select **Advanced...**, as shown in Figure 5-6.

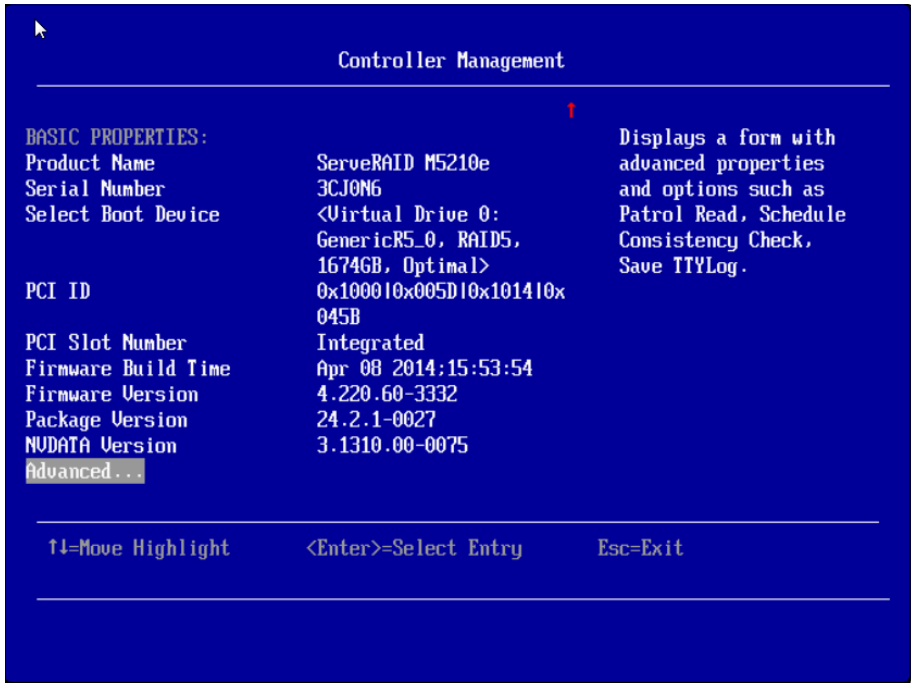

Figure 5-6 Advanced selection options

7. The Advanced Selection menu opens. For the purposes of this configuration, select **Enable Drive Security**, as shown in Figure 5-7.

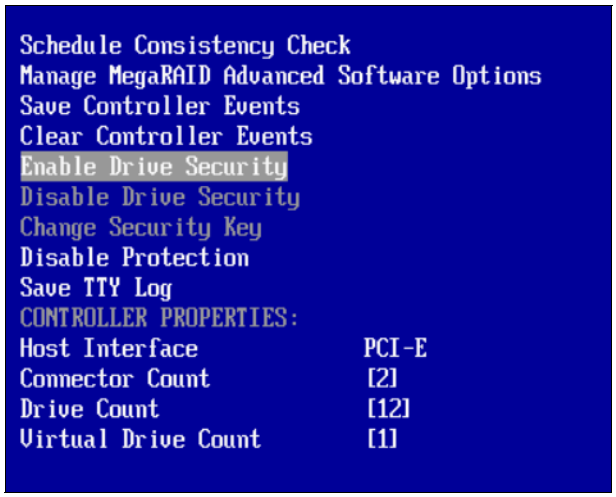

Figure 5-7 Enable Drive Security selection

a. If you find that the **Enable Drive Security** option is disabled in the Change Security Key menu, the controller was set up for encryption. Any required changes should be done by selecting the **Change Security Key** option, as shown in Figure 5-8 on page 114.

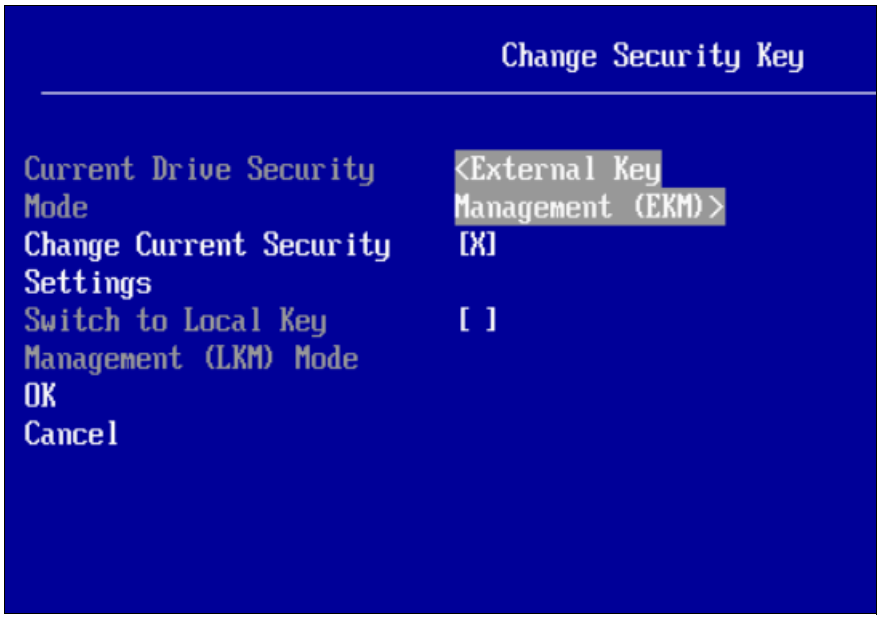

Figure 5-8 Change Security Key menu

Select this option only if drive security is enabled and you must change the settings.

> **Attention:** Do not disable drive encryption unless you are sure that there is no data to be retained on the attached drives. Disabling drive encryption results in a secure wipe of all attached SEDs and you cannot recover the data.

b. Select an EKMS as the key source for the controller by clicking **External Key Management (EKM)** in the menu as shown in Figure 5-8 on page 114. After it is selected, you return to the normal configuration flow to the Enable Drive Security menu.

8. In the Enable Drive Security menu, select **External Key Management** by pressing the Space bar while the cursor is in the appropriate selection box, as shown in Figure 5-9 on page 115. Select **OK** and follow the prompts to exit back to the main setup window where you are prompted to reboot the server to continue. Confirm the reboot.

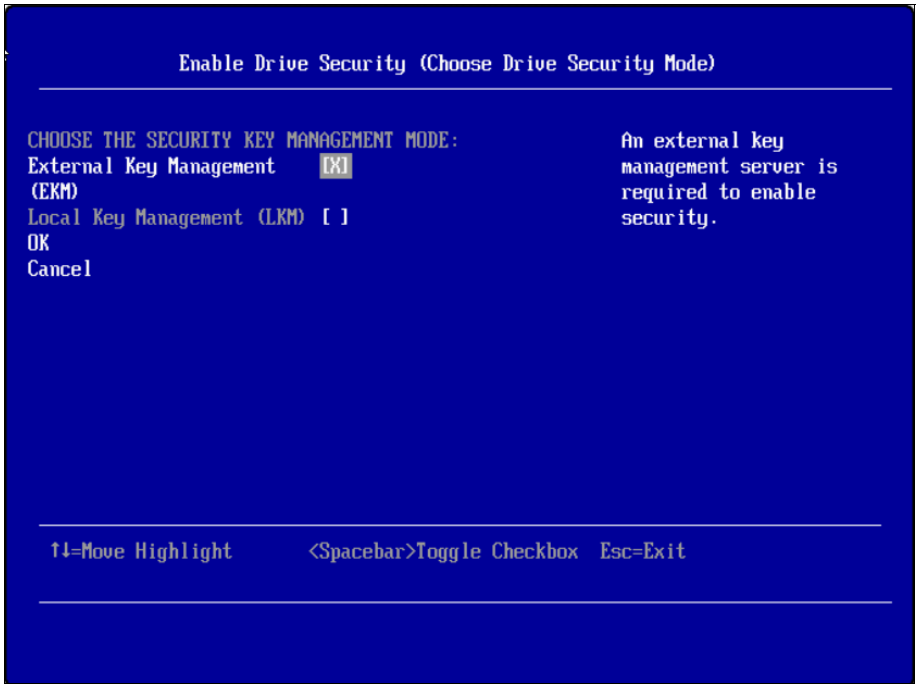

Figure 5-9 Enable EKMS

9. If you configured the Hold new device requests pending my approval option as described in Chapter 3, “IBM Security Key Lifecycle Manager setup” on page 31, you see the boot message that is shown in Figure 5-10 after the server reboots.

Figure 5-10 First Boot error message

This message appears because the server is encountering a trusted connection with the SKLM server, but the initial key is not accepted. For more information about fixing this issue, see 5.1.2, “Accepting pending request on the SKLM server” on page 116.

5.1.2 Accepting pending request on the SKLM server

Browse to your SKLM server home window. Click the **Pending Device Requests** hyperlink at the upper left of the page to browse to the Pending Accept page, as shown in Figure 5-11.

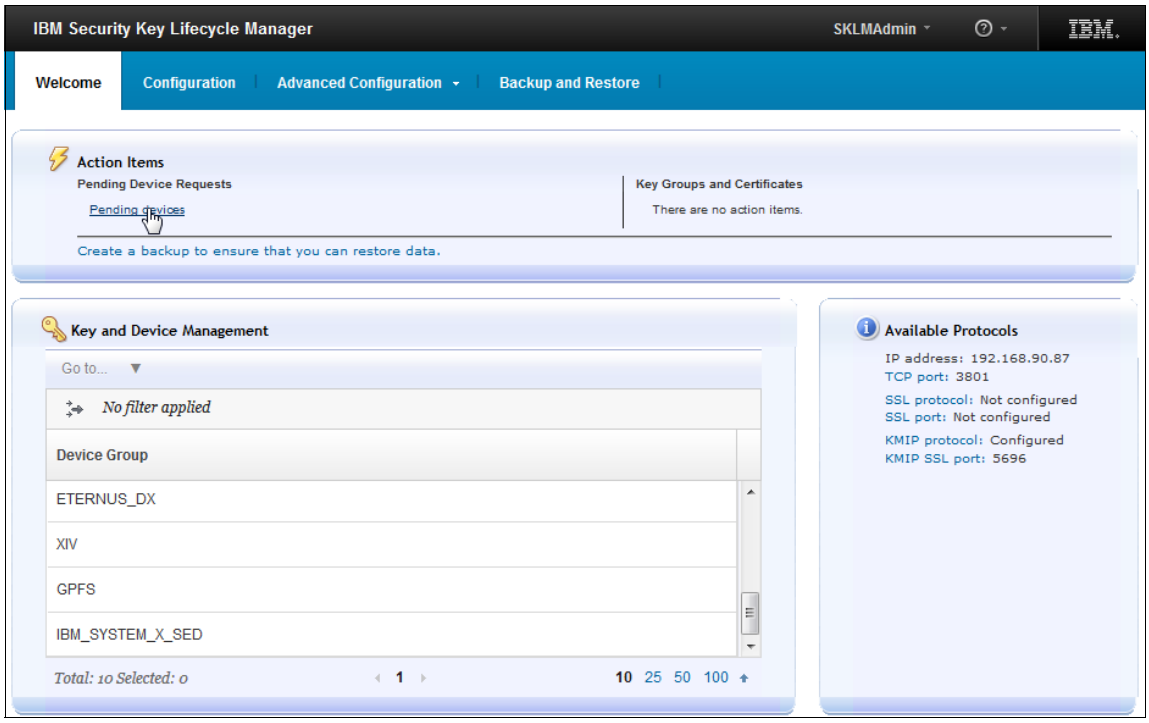

Figure 5-11 Pending Device Requests

The Pending Accept window lists a device entry for the system you configured. This process allows the SKLM server to accept the key request from the target system. Select the corresponding device in the list with a left mouse click, then click **Accept** at the top of the page, as shown in Figure 5-12.

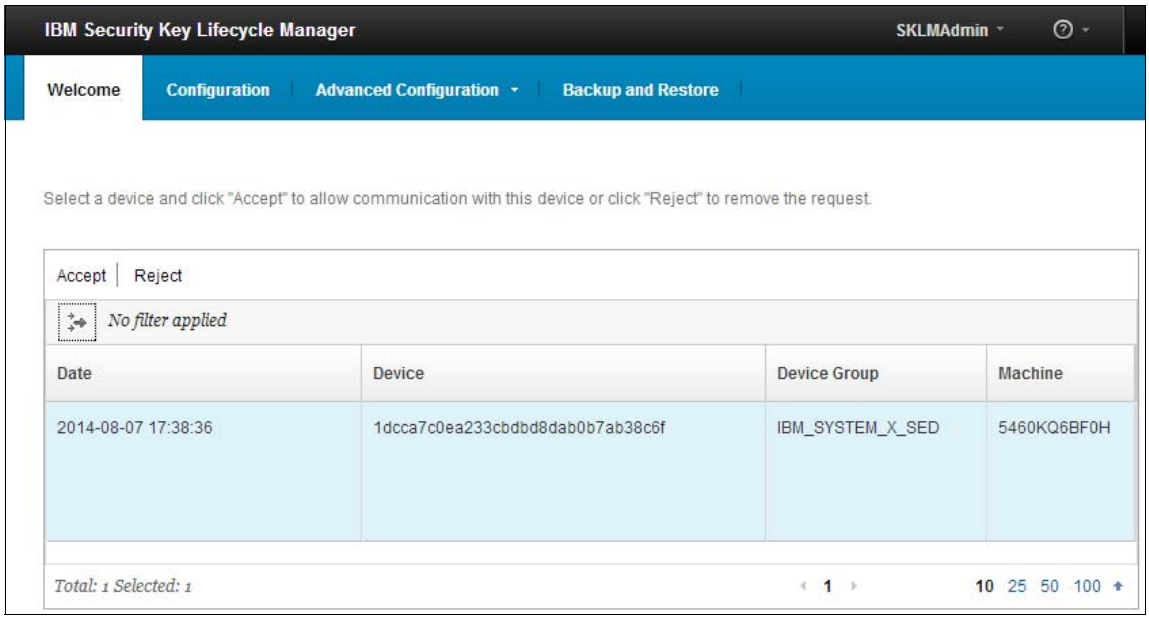

Figure 5-12 Accept pending device request

Next, reboot the target machine one more time for the system to obtain a key from the SKLM server. Accepting or responding to the First Boot error message is not required.

On this next reboot, the server continues to boot without error. The target server is now ready for further configuration. We now prepare the RAID configuration.

5.2 Configuring virtual disks

In this section, we describe the steps to configure the virtual disks on the RAID adapter and secure the resulting virtual disks by using the UEFI configuration interface. We also describe the following tasks:

- Setting up a basic RAID volume
- Activating encryption on virtual drives

5.2.1 Setting up a basic RAID volume

In this section, we describe how to create a simple RAID volume by using the UEFI text-based RAID configuration tool, which is required because the operating system is not yet installed; therefore, there is no access to the graphical configuration utility.

Complete the following steps:

1. Boot the server to the main UEFI window (as described in 5.1, “Enabling storage controller encryption” on page 110) by using the **F1** key at the start splash window.
2. From the main menu, select **Storage** to open the RAID adapter configuration panel and select **Configuration Management**, as shown in Figure 5-13.

Figure 5-13 Main RAID configuration menu

3. In the Configuration Management menu, select **Create Virtual Drive**, as shown in Figure 5-14.

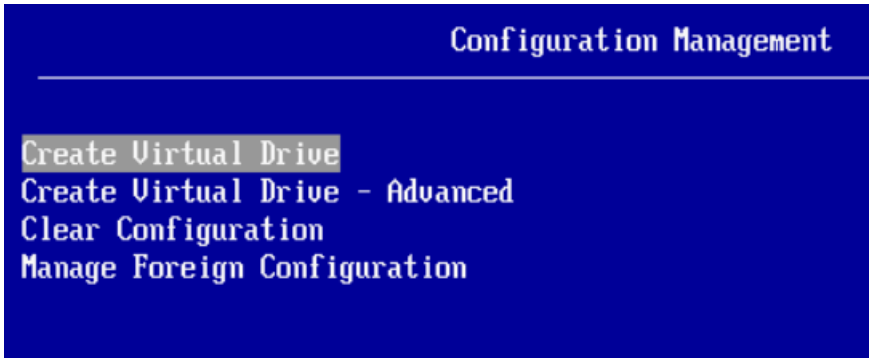

Figure 5-14 Create Virtual Drive option

If there are specific configuration requirements in your environment, you can choose to select the Advanced option. For this example, all that is required is a base two drive RAID 1 volume for the operating system installation. The steps to create the volume are described in this section as a walkthrough.

In the following menus, you can select the options that match the requirements of your deployment as they do not affect the ability to perform the encryption steps that you must follow.

If you do have a combination of SEDs and non-SEDs installed in the system, you should select the advanced option to ensure that the appropriate drives are configured for the volume.

4. In the Create Virtual Drive menu, select the RAID type, as shown in Figure 5-15.

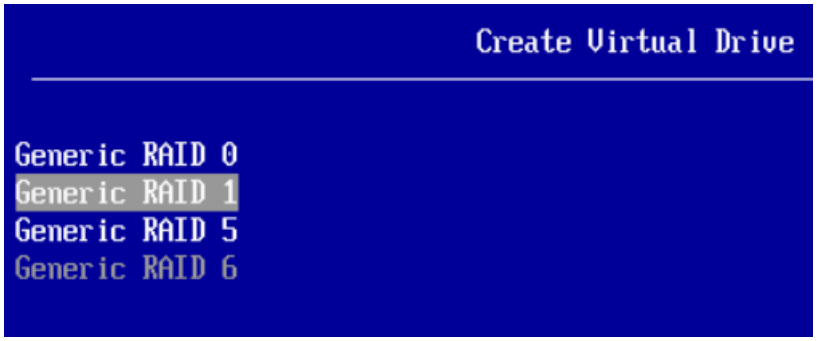

Figure 5-15 RAID selection

5. For the sample configuration, select RAID 1. Check all of the settings in the Generic R1 menu. Scroll down to the bottom of the list to select the **Save Configuration** option, as shown in Figure 5-16.

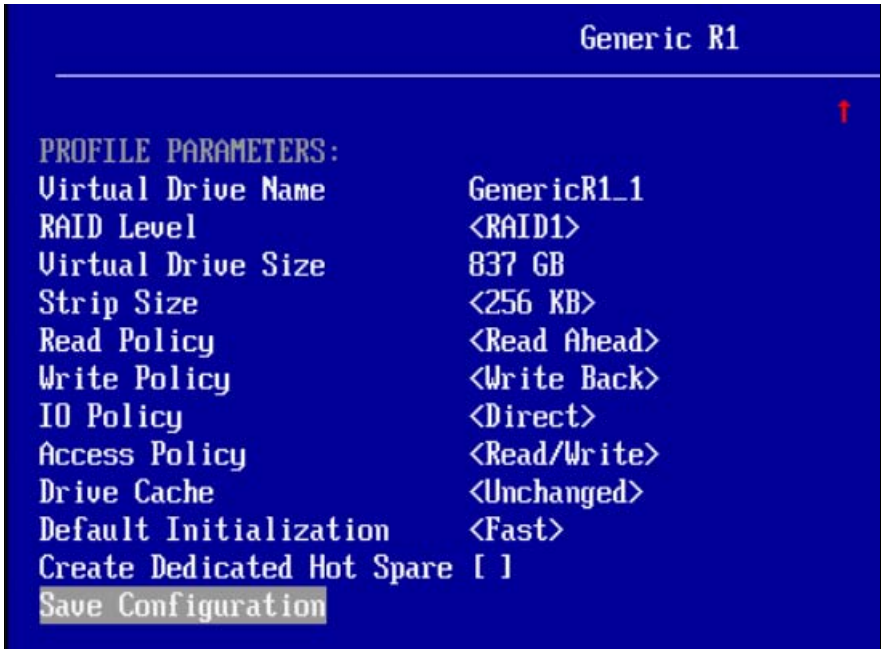

Figure 5-16 Save Configuration

6. The Data Loss warning panel opens. Because you lose any data on the selected drives for the array, ensure that it is an acceptable action and press the spacebar while you highlight **Confirm**. Then, select **Yes** to create the virtual drive.

   A success message indicates that the operation was completed successfully, as shown in Figure 5-17.

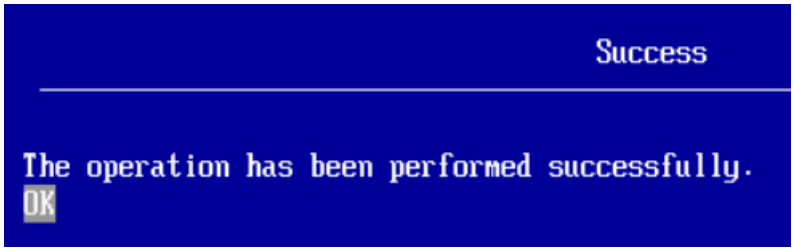

Figure 5-17 Successful completion

You can repeat these steps to create more virtual drives as required, if you have sufficient SEDs installed for the other requirements. For this example configuration, only a single RAID 1 virtual drive is configured.

Next, we activate the encryption for the new virtual drive.

5.2.2 Activating encryption on virtual drives

In this section, we describe how to activate the encryption on the virtual drive that was created in 5.2.1, “Setting up a basic RAID volume” on page 117.

Complete the following steps:

1. Browse to the main menu for the UEFI storage devices, as described in 5.2.1, “Setting up a basic RAID volume” on page 117. Select **Virtual Drive Management**, as shown in Figure 5-18.

Figure 5-18 Virtual Drive Management

2. In the Virtual Drive Management menu, you see a list of the available virtual drives. In the example configuration, only a single RAID 1 volume was created; therefore, only a single entry is displayed. If you created different or other volumes in the previous process, those volumes are presented here.

   Select the virtual drive on which you want to activate the encryption. In the sample configuration, the selection window resembled the one that is shown in Figure 5-19 on page 120.

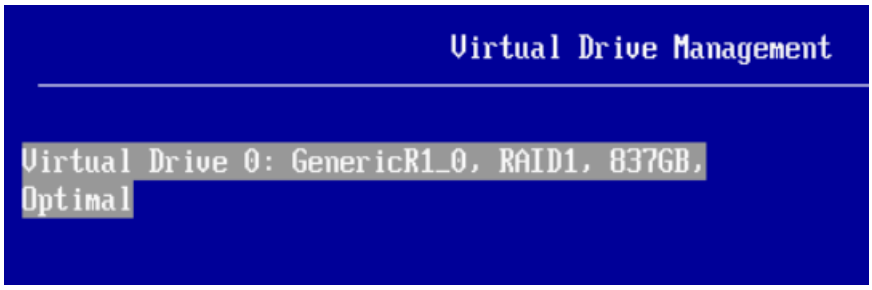

Figure 5-19 Virtual Drive Selection

3. In the Virtual Drive configuration panel, select the **Select Operation** entry at the top of the list, as shown in Figure 5-20.

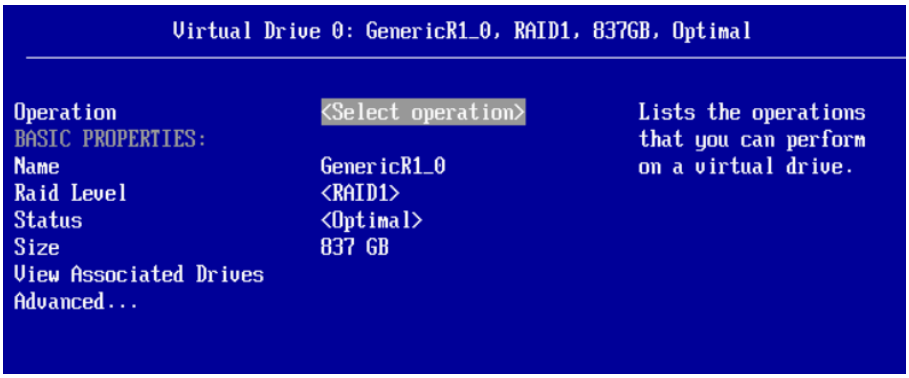

Figure 5-20 Select Operation

4. Select **Secure Virtual Drive** to start the encryption of the selected virtual drive, as shown in Figure 5-21 on page 121.

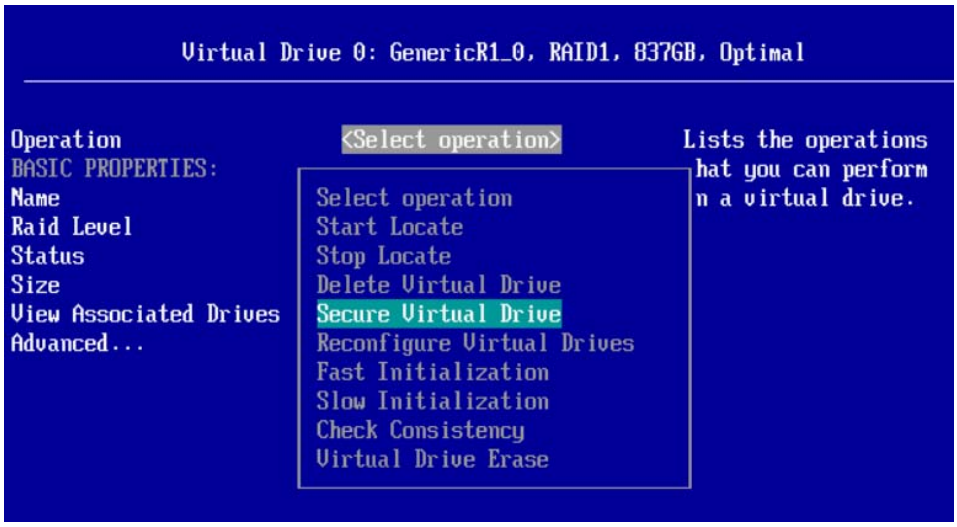

Figure 5-21 Secure Virtual Drive

5. The virtual drive is encrypted by using the key that is provided at boot time via the SKLM server. If you return to the Virtual Drive properties, you can see the Secured entry listed as `<Yes>`, as shown in Figure 5-22.

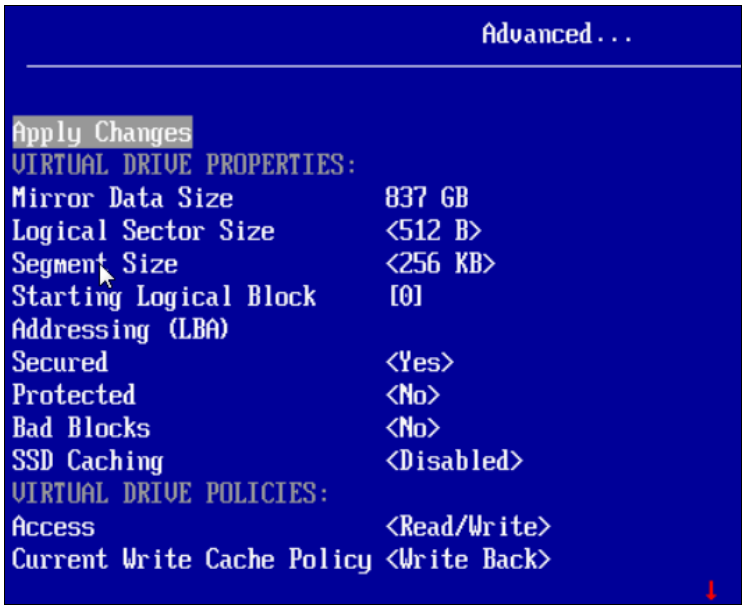

Figure 5-22 Virtual Drive Secured

6. After the virtual drive is verified as being secured, you can exit the UEFI and reboot the server. If you created more virtual drives as part of this exercise, you must select each of the drives and repeat the steps to enable Secure Virtual Drive.

5.3 Conclusion

The drives are now secured and the system is ready for installing the operating system. Because all of the encryption and key management is performed in the system firmware, no other steps are required when you install any operating system.

6

Managing your System x server SED deployment

In this chapter, we describe managing the encryption keys and certificates that are needed for your IBM Security Key Lifecycle Manager (SKLM) and self-encrypting drive (SED) environment.

This chapter builds upon the exchanges that were made between the server (or servers) and SKLM. It also describes some other administrative tasks, such as backup and restore, which were not yet performed but are critical to preserving our encryption key management setup.

This chapter includes the following topics:

- Certificate exchange and device acceptance review
- SKLM backup and restore

© Copyright Lenovo 2015. All rights reserved.

6.1 Certificate exchange and device acceptance review

In previous chapters of this book, we described a certificate exchange between the SKLM key manager and a System x server with SEDs. We also described registering a new device with SKLM. In this section, we review and elaborate upon those steps.

6.1.1 Certificate exchange

Chapter 4, “Integrated Management Module configuration” on page 85 included the instructions to create and download a System x server certificate by using the Integrated Management Module (IMM) of a System x server with the SKLM Feature on-Demand key activated. Before the System x server connection to SKLM can be tested, complete the following steps to import its certificate into SKLM.:

1. Copy the certificate file that was downloaded from the IMM locally to the SKLM server so that it can be imported. Our System x3650 M4 certificate is shown in Figure 6-1.

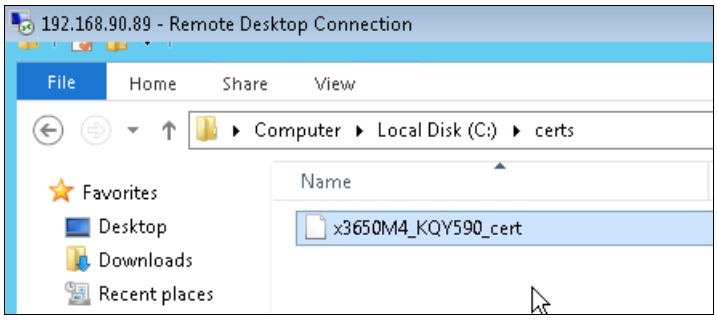

Figure 6-1 Certificate file

2. Log in to the SKLM web interface by using the following URL:

   `https://<SKLM server address>:9080/ibm/SKLM/login.jsp`

   where <*SKLM server address*> is the IP address or host name of the SKLM server.

3. Click **Advanced Configuration** → **Server Certificates**. In the SKLM installation steps that are described in 3.5, “Generating an SKLM server certificate” on page 78, you generated an SKLM server certificate, which is shown in Figure 6-2. This certificate also was imported into the IMM of your System x server per the instructions that are described in 4.2.7, “Importing SKLM server certificate” on page 97.

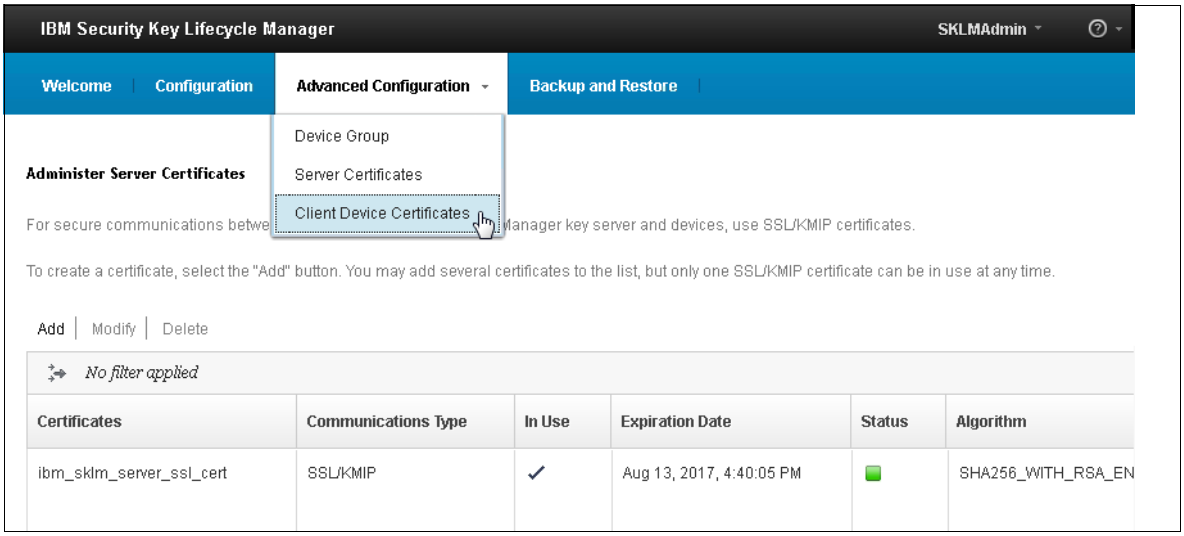

Figure 6-2 Server certificate present

4. Click **Advanced Configuration** → **Client Device Certificates**. Then, click **Import**, as shown in Figure 6-3 on page 125.

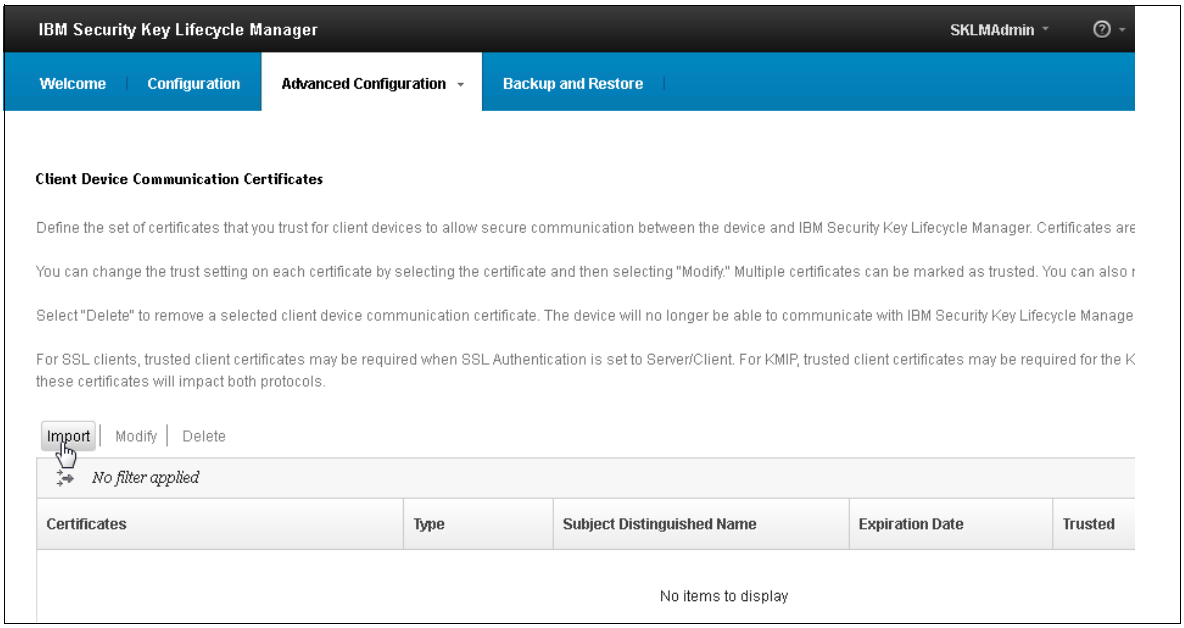

Figure 6-3 Import client device certificate

5. Enter a display name in the Import SSL/KMIP Certificate pop-up dialog so that you can identify this System x server in SKLM. Then, select **Browse** and locate, and select the certificate to import, as shown in Figure 6-4 on page 126.

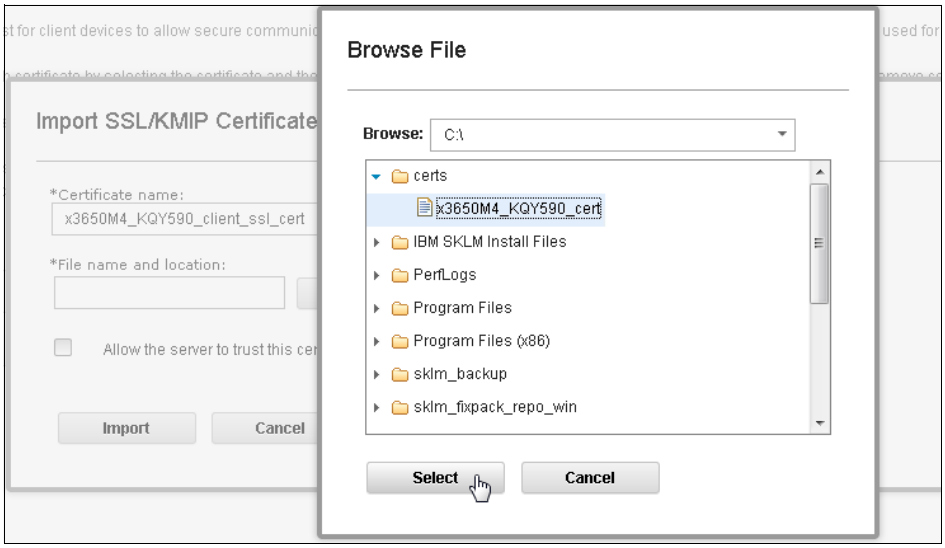

Figure 6-4 Select certificate

6. Verify your entries and select **Allow the server to trust this certificate and communicate with the associated client device**, as shown in Figure 6-5. Then, select **Import.**

Figure 6-5 Trust and import certificate

7. A warning to back up your SKLM data is shown. Select **Close**, as shown in Figure 6-6. It is critical that you back up your SKLM data whenever new devices are added. To perform this backup, follow the backup steps that are described in 6.2, “SKLM backup and restore” on page 133.

Figure 6-6 Backup reminder

Your certificate is now imported and trusted in SKLM, as shown in Figure 6-7.

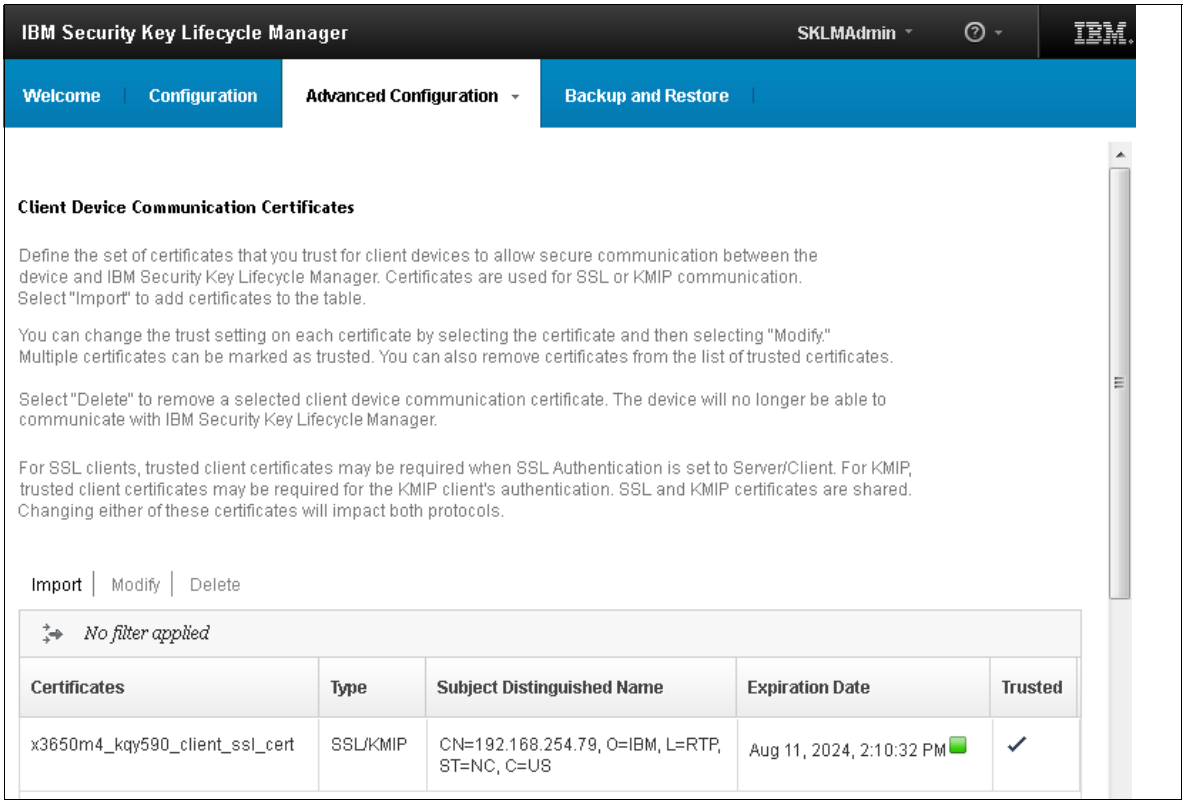

Figure 6-7 Certificate imported

8. Test the connection with the IMM on your System x server, as shown in Figure 6-8. For more information, see 4.2.10, “Test the connection to SKLM” on page 99.

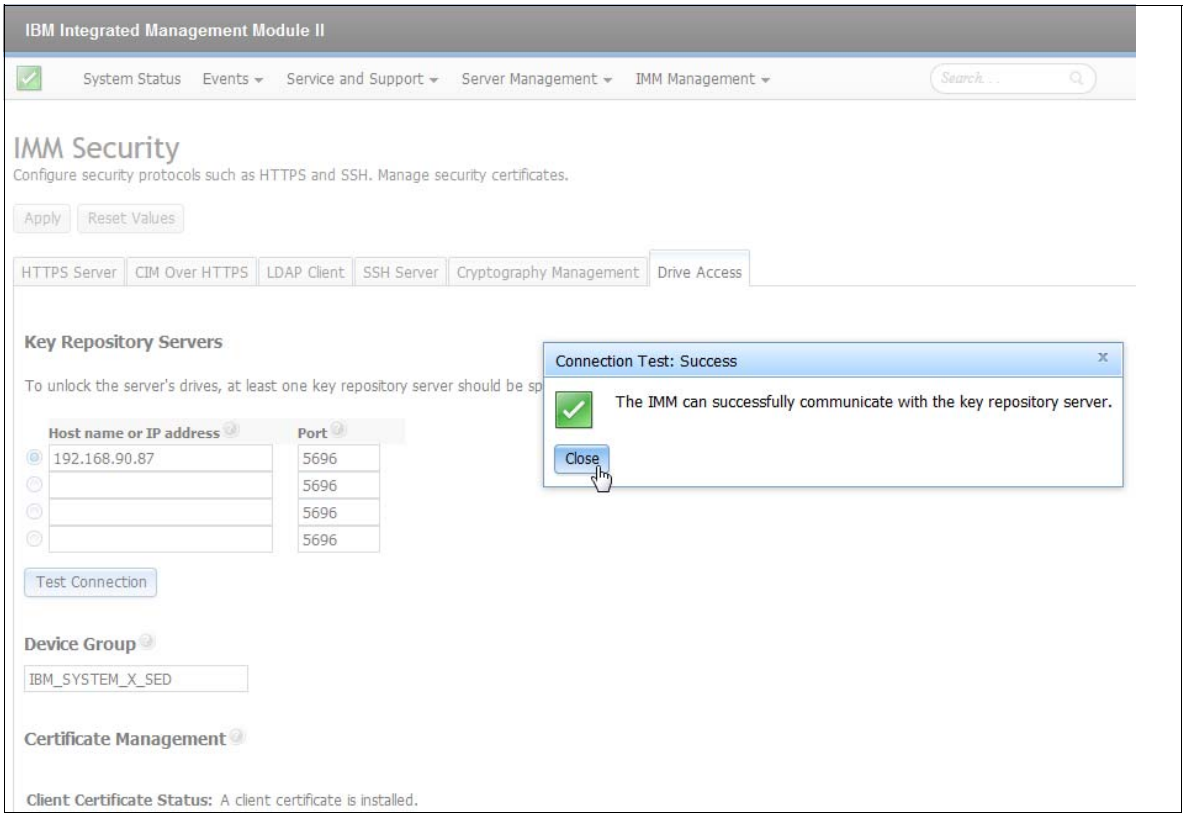

Figure 6-8 Successful IMM connection

6.1.2 Certificate acceptance options

In Chapter 5, “UEFI configuration” on page 109, we described the process that is used for configuring a System x server to use external key management for drive security. We also described the process that is used for accepting the device and its key request in SKLM. By using the tasks that are described in this section, we verify that those tasks are complete and review the details around the process.

Holding new devices for approval

The first step is to change (or at least know) the settings for new devices that attempt to connect to SKLM. We recommend that when new devices attempt a connection to SKLM, those devices are held for approval. Doing so allows you to acknowledge and control the connections without the extra effort to configure them manually. This hold also provides a reminder to back up the SKLM data whenever a new device is added, which is critical to keeping a valid backup that supports all deployed System x servers.

Complete the following steps to set up this configuration:

1. Connect to the SKLM web interface and browse to the Welcome tab, as shown in Figure 6-9 on page 129. In the Welcome window under Key and Device Management, right-click **IBM_SYSTEM_X_SED Device Group**. This group is the default key group for all System x servers.

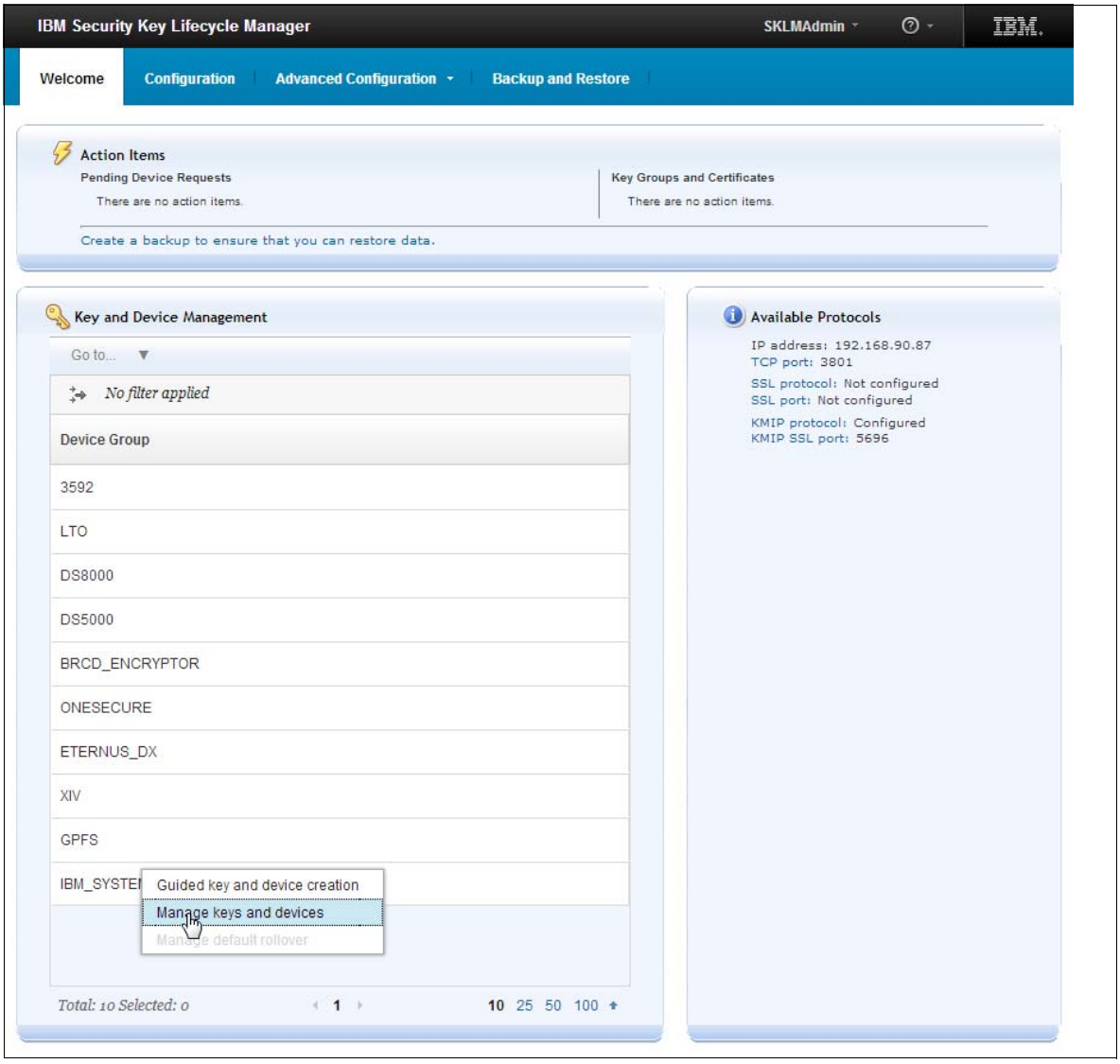

Figure 6-9 Manage System x keys and devices

2. From the pop-up menu, select **Hold new device requests for communication**, as shown in Figure 6-10. This setting is saved automatically. All future connection requests are held in a pending state.

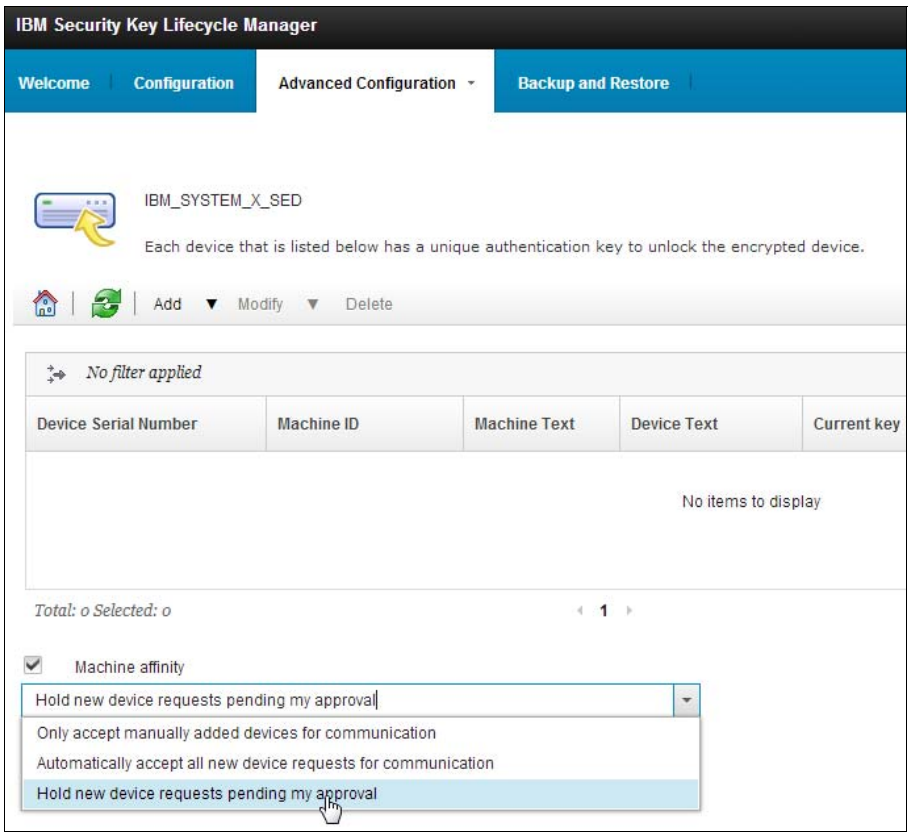

Figure 6-10 Hold new device requests for approval

Accepting new devices

In this section, we describe the process that is used to accept a new device after it contacts the SKLM server, and allow it to retrieve a key encryption key for the controller to access SEDs on the next server boot. This task was introduced in Chapter 5, “UEFI configuration” on page 109.

Complete the following steps:

1. Browse to the Welcome tab. On the Action Items dashboard, click **Pending devices**, as shown in Figure 6-11. This process adds the server and its RAID controller as a new device so that encryption key exchanges can be made.

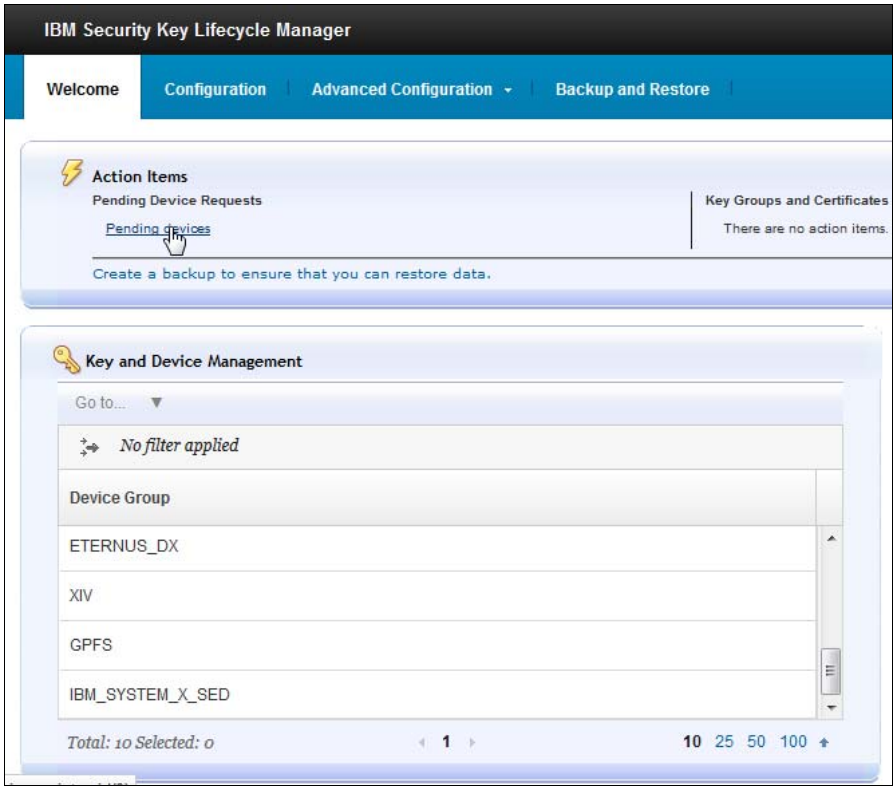

Figure 6-11 Device waiting acceptance

2. Select the new device based on the time stamp, device group, and machine information and select **Accept**, as shown in Figure 6-12.

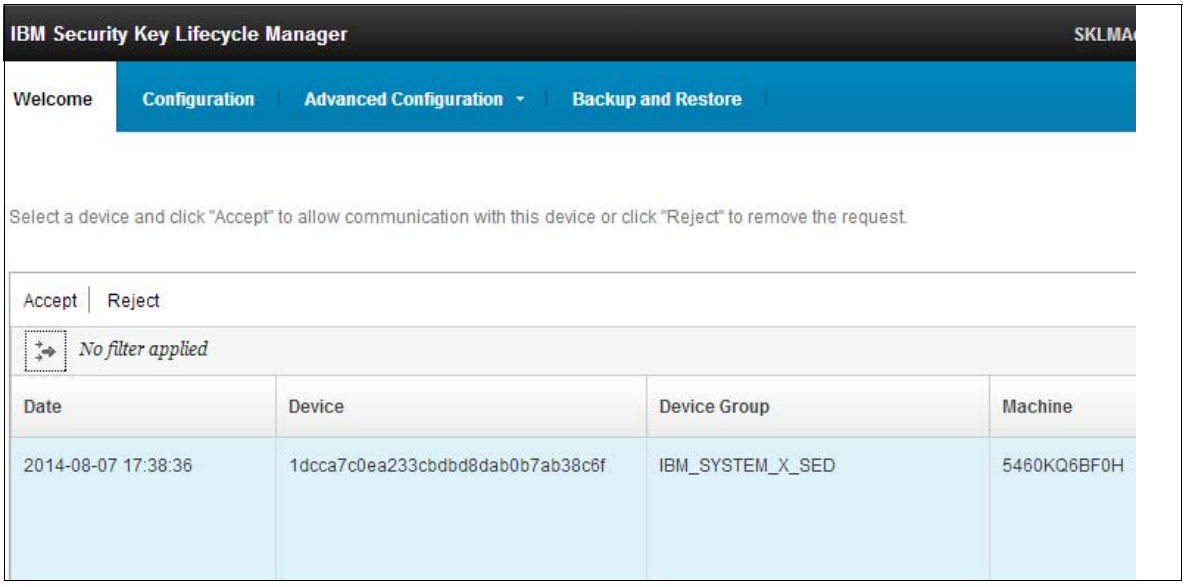

Figure 6-12 Accept device window

3. In the Accept Device Request dialog, we recommend that you select **Modify and Accept** (as shown in Figure 6-13) to provide a description of your device. The message warns you to perform a backup after this procedure.

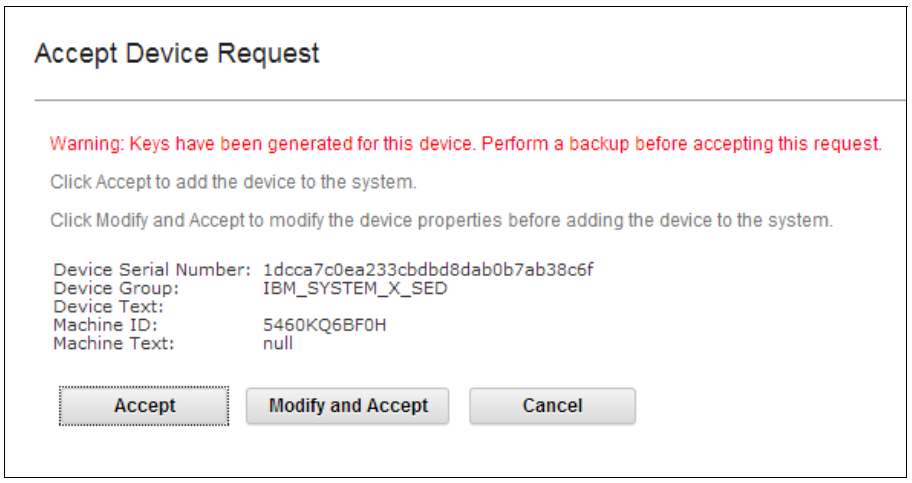

Figure 6-13 Modify and accept device

4. Enter some information about your server in the Device text field and a more comprehensive Device Description. Then, select **Add Device**, as shown in Figure 6-14.

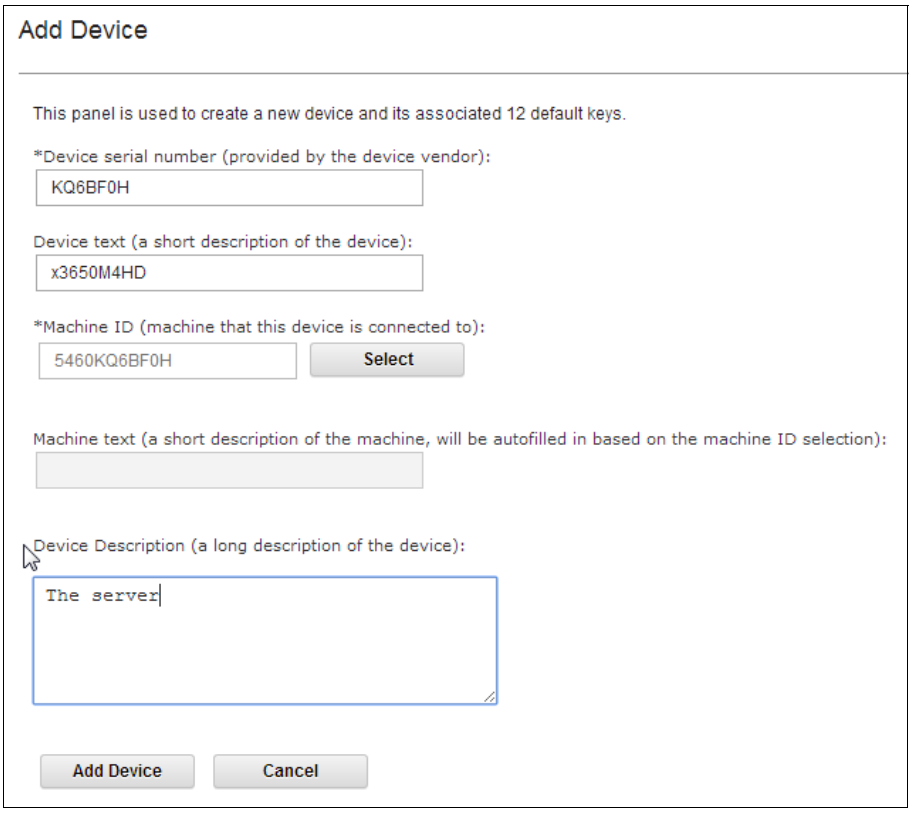

Figure 6-14 Describe and accept

5. You now added the new device in SKLM where it is ready to exchange encryption keys. The Current Key field is not initially populated, as shown in Figure 6-15. The server must be rebooted. During the next boot phase, it contacts the SKLM server for a key encryption key and populates the field. The text in the Current Key field is not the actual encryption key; instead, it is only a display name for it. The display name rotates or changes with each reboot of the server.

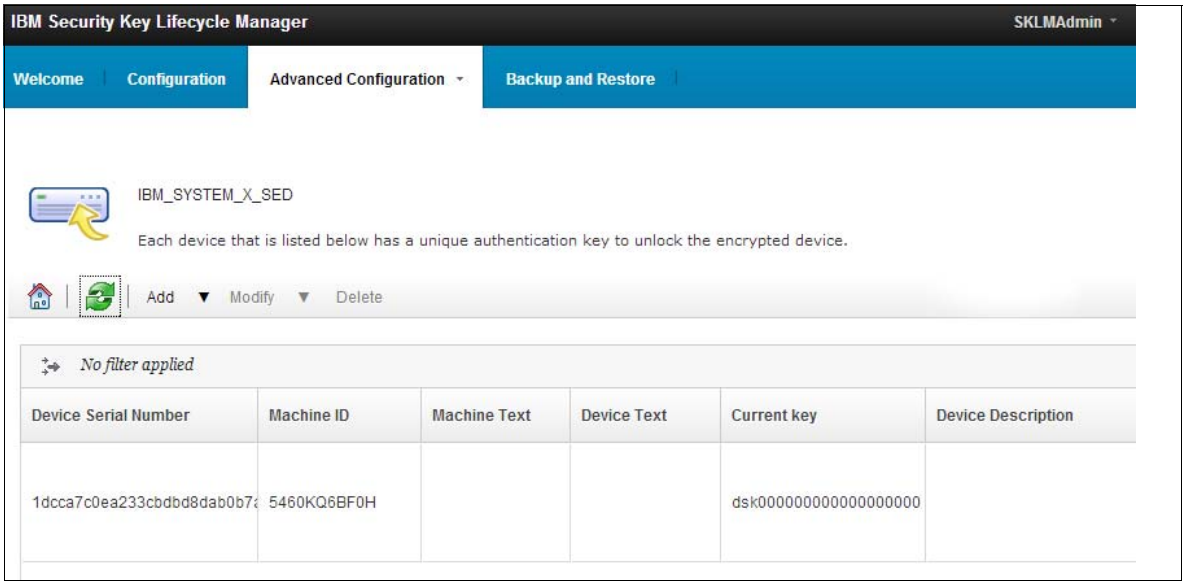

Figure 6-15 Device added and key generated

6.2 SKLM backup and restore

In this section, we describe the backup and restore tasks for SKLM key manager server data by using the SKLM web interface. It is important to back up the SKLM server immediately after any changes or additions, especially if redundant SKLM servers are not configured.

6.2.1 SKLM data backup

The following steps demonstrate how to create a backup of your SKLM server. Backup files are created locally on the SKLM server, although backups can be started with a web session from a remote system.

We created a local directory, `C:\sklm_backups`, for storing our proof of concept backup files. Backups should not remain solely on the local SKLM server. The password for the backup should be recorded, and the backup data should be copied to a separate system.

If possible, copy the backup to a separate physical data center to eliminate the risk that all backups are destroyed if the SKLM server failed and the data center was lost.

Complete the following steps:

1. Log in to the SKLM web interface at the following URL:

   `https://<SKLM server address>:9080/ibm/SKLM/login.jsp`

   where *<SKLM server address>* is the IP address or host name of the SKLM server.

2. Browse to the Backup and Restore tab. Then, select **Create Backup**, as shown in Figure 6-16 on page 134.

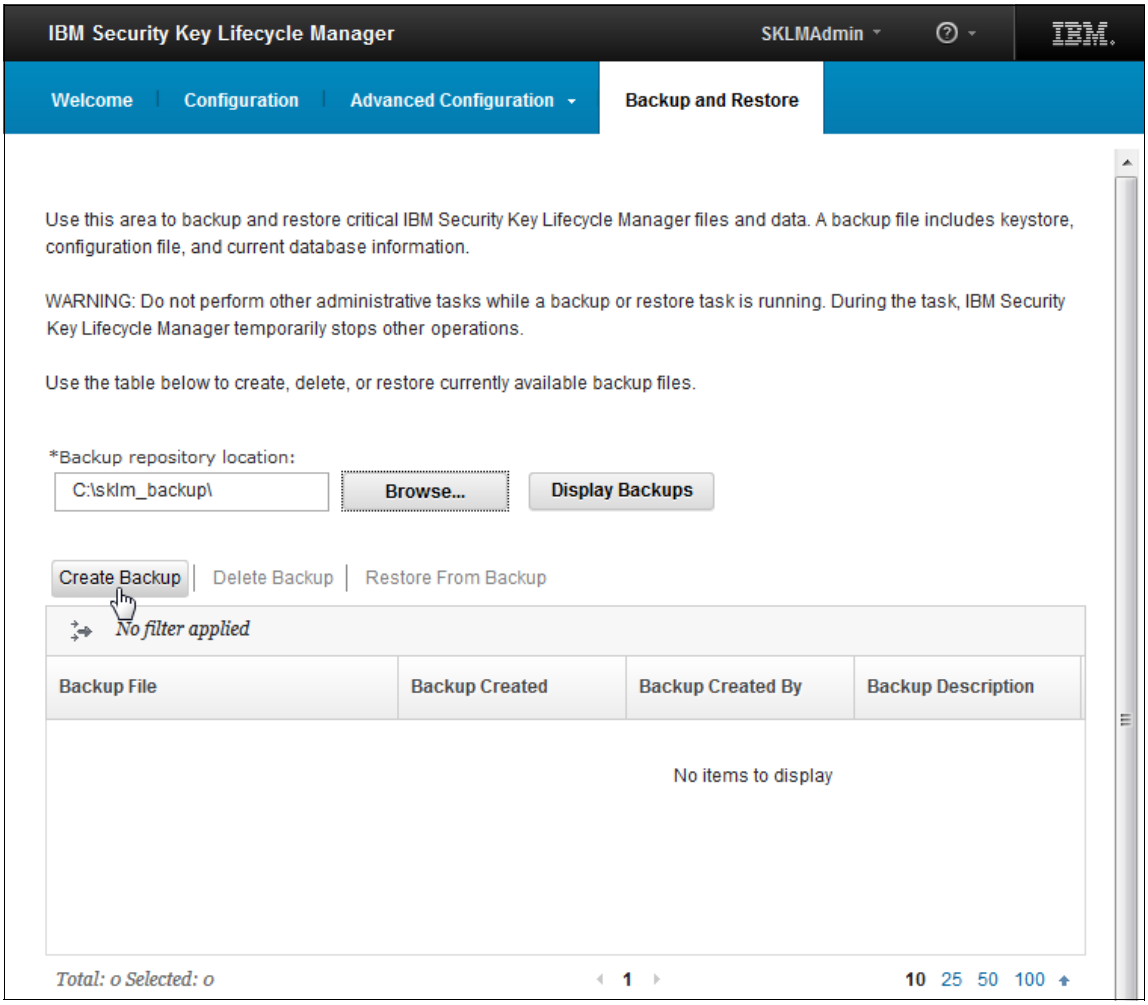

Figure 6-16 Create backup

3. Enter a backup location, create and confirm a password for the backup, and provide a description for your backup. When the backup is created successfully, be sure to record the password for that backup file. The password is required to restore the data and cannot be recovered later. Select **Create Backup** to start the backup process, as shown in Figure 6-17.

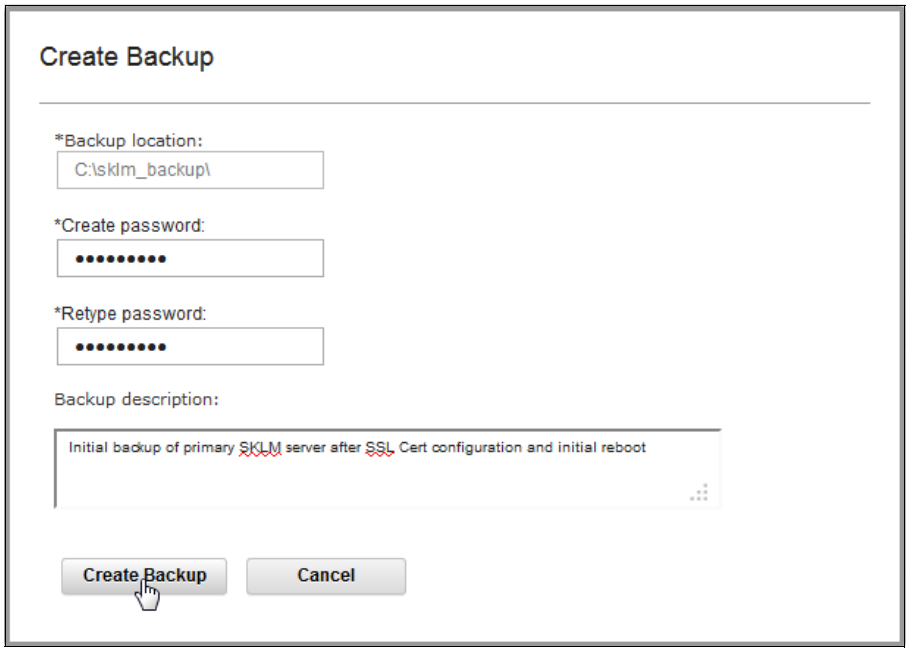

Figure 6-17 Backup location and password

4. Confirm your backup settings by selecting **OK** in the next window.

   The following pop-up (as shown in Figure 6-18) shows that a backup was successfully created.

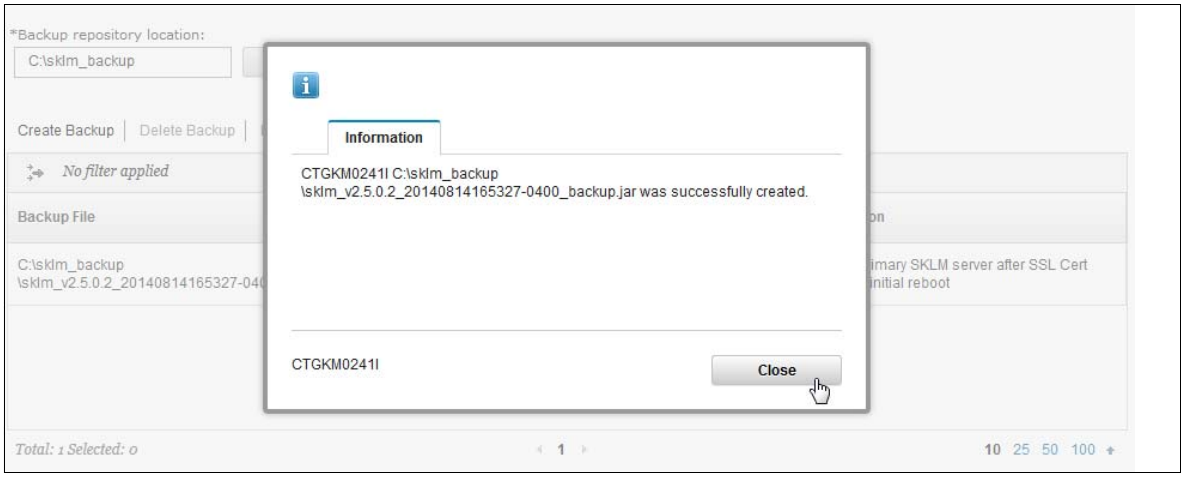

Figure 6-18 Backup created

5. Select **Return home** (as shown in Figure 6-19) for a quick way to reference your backups.

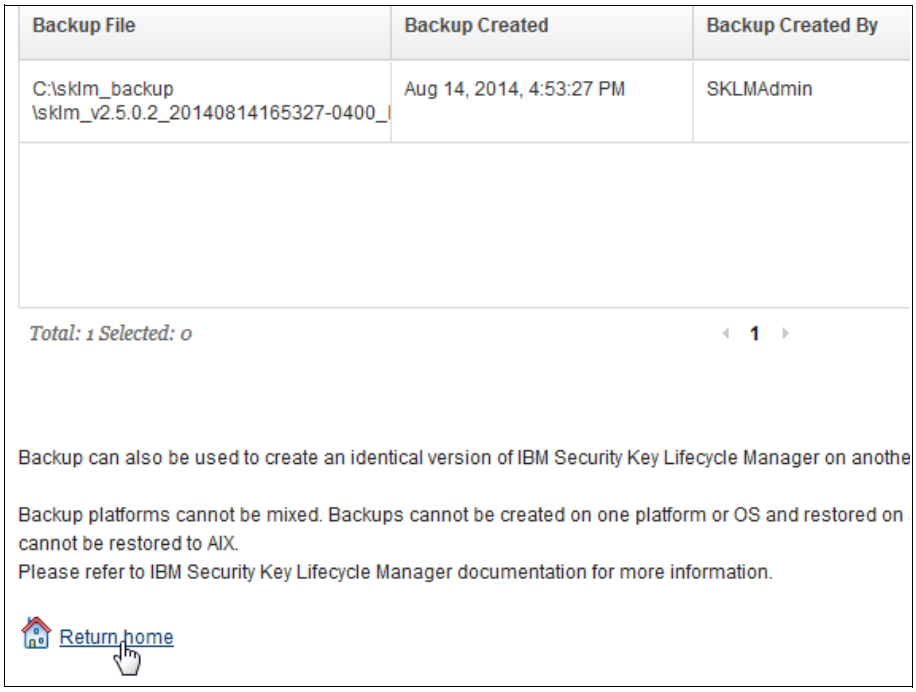

Figure 6-19 Return home option

6. In the Action Items area on the home page (as shown in Figure 6-20), you can see when the last backup was performed and a link to go to the backup and restore page.

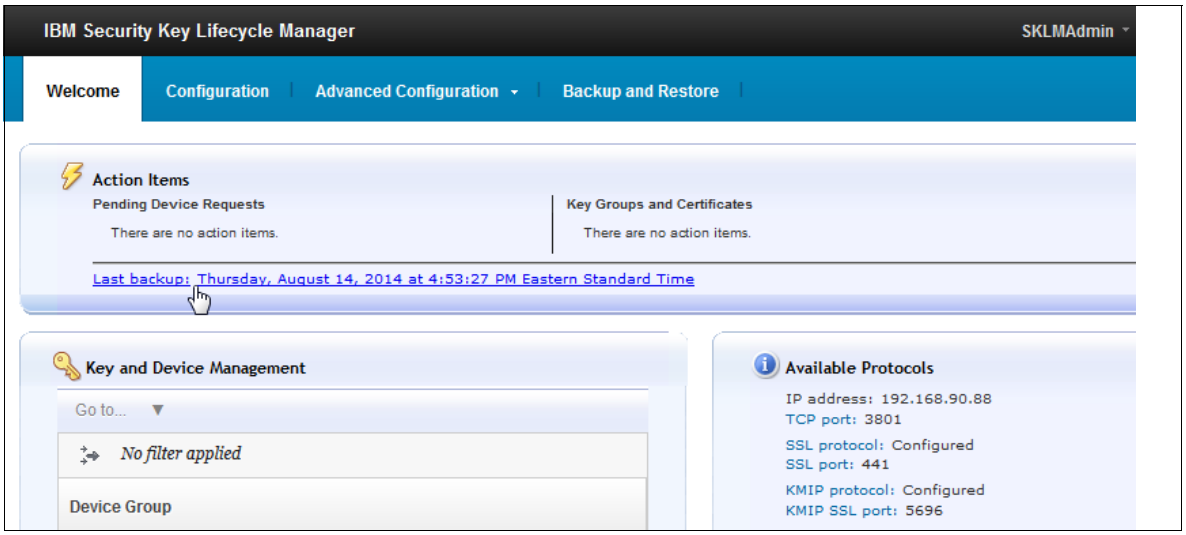

Figure 6-20 Action Items dashboard

7. Ensure that your backup file was created as expected by checking the location to which you saved it. In our example, the backup file is named `sklm_v2.5.0.2_20140814165327-0400_backup.jar`, and it is approximately 15 MB. However, this is a small setup. Each managed SKLM server can grow the database by up to a few MB, so account for much larger backups (depending on your environment).

Note: You should record the password for your backup file and copy it to an auxiliary storage location for safety.

6.2.2 Restoring SKLM data to existing installation

There can be a time when you might need to roll back SKLM to an earlier back up or try to recover data to a new SKLM installation if an SKLM server fails.

Complete the following steps to perform the rollback by using the SKLM web interface:

1. Log in to the SKLM web interface and browse to the Backup and Restore tab. Select **Browse**. In this example, we restore backup data to a new installation of SKLM, which is intended for a secondary SKLM instance. The backup must be copied to the local SKLM file system to restore it, so we copied it into a directory named `C:\sklm_backups`.

   However, the backup process can be run from a remote system with access to the SKLM web interface. In the Browse Directory dialog, select from the drop-down menu the local drive that contains your backup. Then, select the directory that contains your backups and click **Select**, as shown in Figure 6-21.

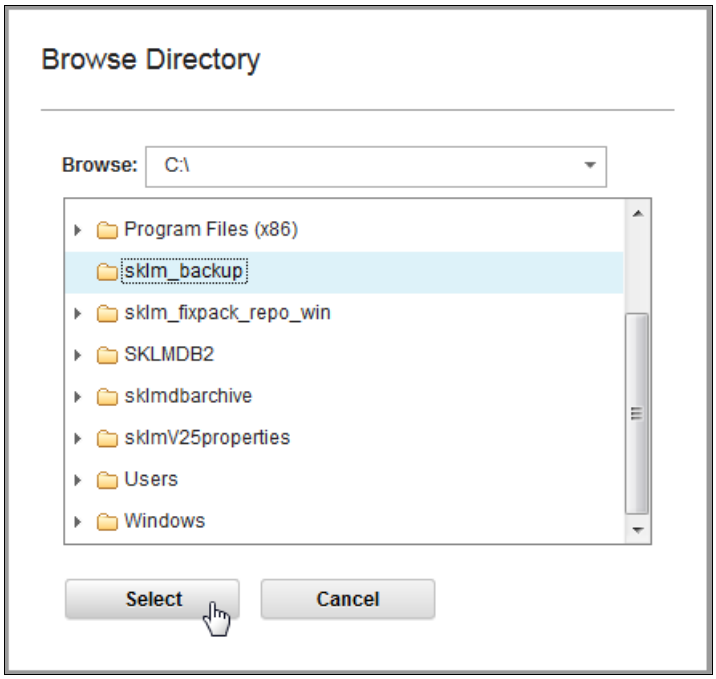

Figure 6-21 Browse directory

2. In the next dialog, select **Display Backups** (as shown in Figure 6-22) to import the backups in that directory to the web interface.

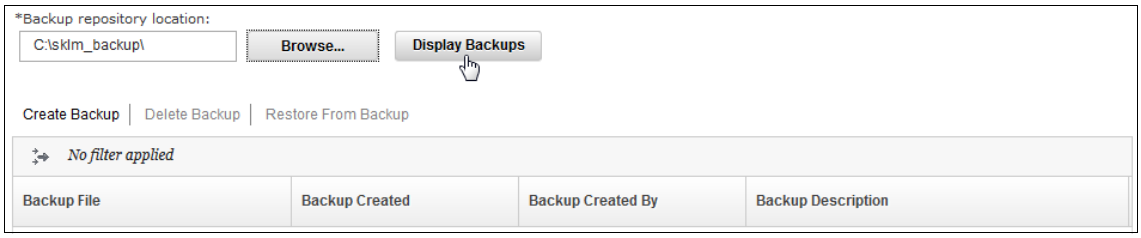

Figure 6-22 Display backups

3. Select the backup that you want to restore and click **Restore From Backup**, as shown in Figure 6-23.

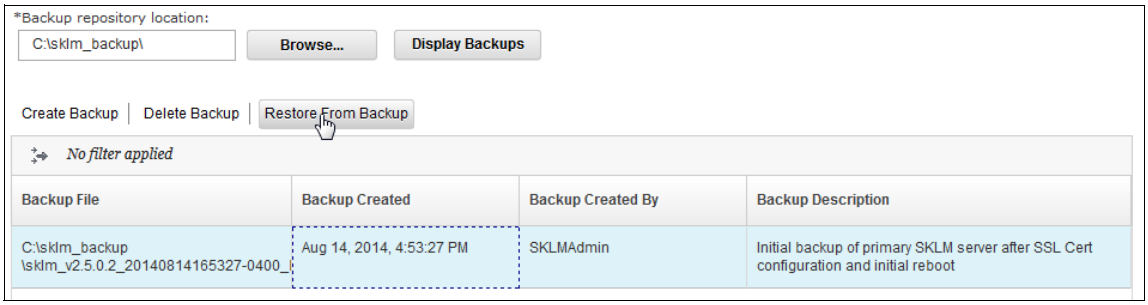

Figure 6-23 Restore selected backup

4. Confirm your backup file and enter the associated password from when you created the backup. Then, select **Restore Backup** to start the final confirmation, as shown in Figure 6-24.

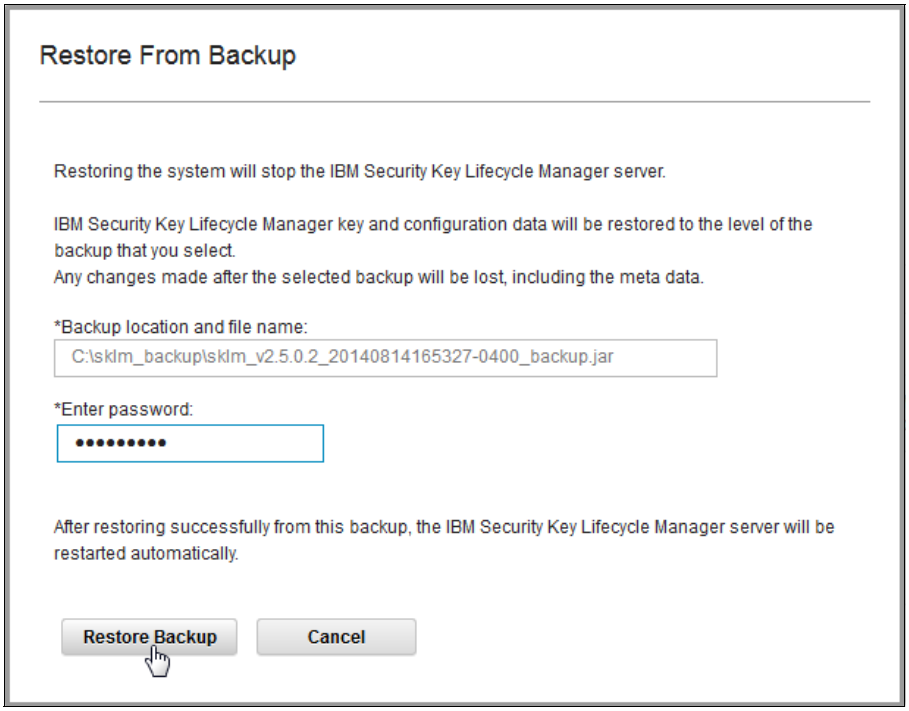

Figure 6-24 Restore backup with password

5. In next dialog, the SKLM server restarts automatically after a restore because of the default SKLM properties. Your SKLM cannot be accessed during the restart; therefore, a restore should not be run if any servers are going to be rebooting and trying to contact SKLM for encryption keys now.

   When you are ready for the restore process and reboot, select **OK,** as shown in Figure 6-25 on page 139.

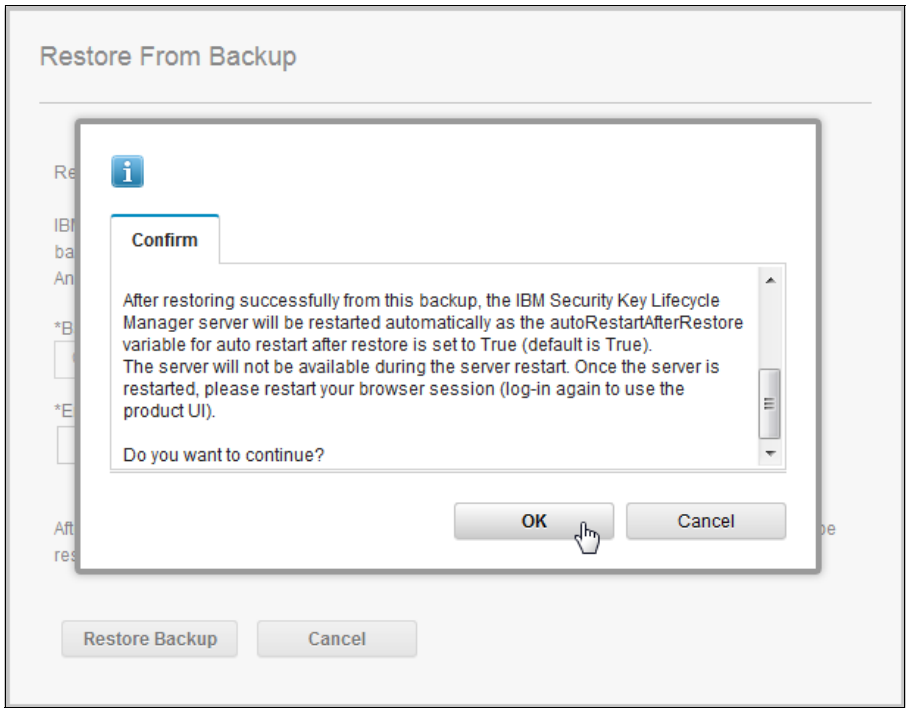

Figure 6-25 Confirm restore

6. During the restore process, a pop-up with a progress indicator appears. A progress bar also appears temporarily on the Backup and Restore tab. The small back-up in our example took less than 2 minutes to restore.

After the restore process is successful, the SKLM web services and related processes restart, but not your physical or virtual server. We recommend that you wait 3 minutes for the restart to be safe. Then, log out of the SKLM interface if your session does not time out. Log back in and verify that your data was restored.

6.3 Conclusion

In this chapter, we described how to add new System x server encryption devices and allow them access to SKLM to retrieve key encryption keys. We also highlighted the importance of creating SKLM backups and outlined how to perform backup and restore processes.

The information in this chapter represents only a portion of the configuration and education that is needed to implement a production SKLM environment. Several other concepts, such as SKLM server replication, user and group access control, and key expiration, should be reviewed and implemented for a robust SKLM environment. For more information about configuration and administration of SKLM 2.5, see the product documentation in the IBM Knowledge Center, which is available at this website:

`http://www.ibm.com/support/knowledgecenter/SSWPVP_2.5.0/com.ibm.sklm.doc_2.5/welcome.htm?lang=en`

Part 2

Appendixes

© Copyright Lenovo 2015. All rights reserved.

A

Local key management alternatives

In this appendix, we describe the required steps to create virtual drives that use encryption keys local to the RAID adapter to which the self-encrypting drives (SEDs) are attached. The intention of this simplified guide is to act as a primer for organizations that want to deploy in a local key management mode with the intention of switching to external or centralized management later.

Important: Localized key management does not require the purchase of any Feature on Demand keys to function. The controller does require having at least a cache or flash module that is installed to activate SED drive support.

Two methods are described here. The first method uses the UEFI-based management interface to set up RAID security on a new system before the operating system is installed. The second method uses the graphical management tool within the Operating System.

This appendix includes the following topics:

- Using the UEFI-based management utilities for new installations
- Using the graphical MegaRAID Storage Manager

© Copyright Lenovo 2015. All rights reserved.

Using the UEFI-based management utilities for new installations

This section describes the use of the text-based management tools that are integrated into the UEFI of the System x server.

Keeping your data safe: Activating this option does not destroy any data that is on any configured virtual drives. However, after a virtual drive is set to protected mode, disabling this option results in the loss of access to the data and must be restored from a backup source.

Accessing the UEFI storage management tool

Complete the following steps to access the UEFI-based storage management tools:

1. Power on or reboot the server by using any preferred method. When you see the window that is shown in Figure A-1, press F1 to boot the server to the UEFI setup menu.

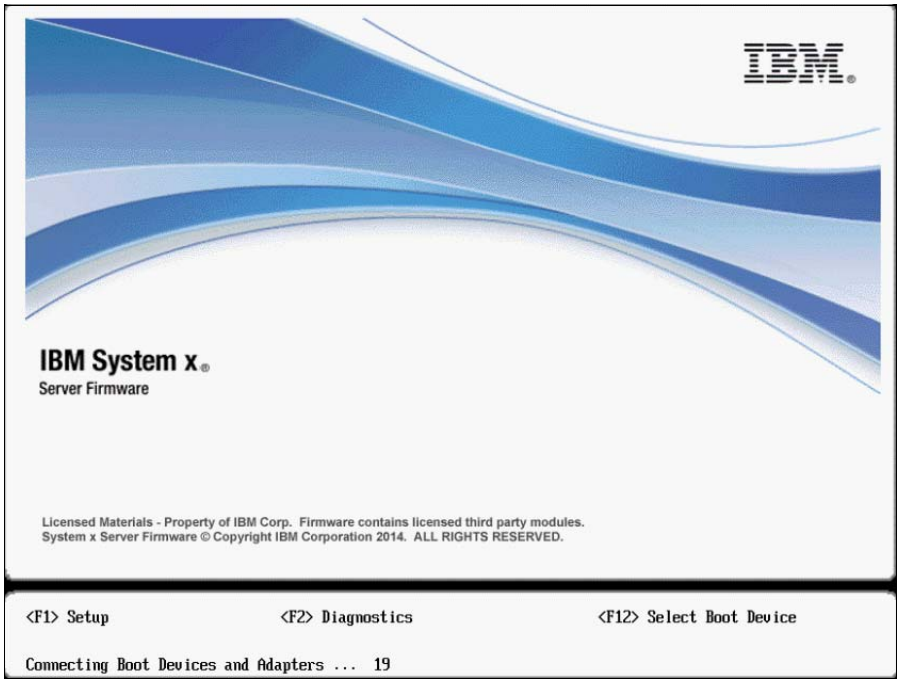

Figure A-1 UEFI welcome window

2. At the main UEFI configuration window, select **System Settings**, as shown in Figure A-2.

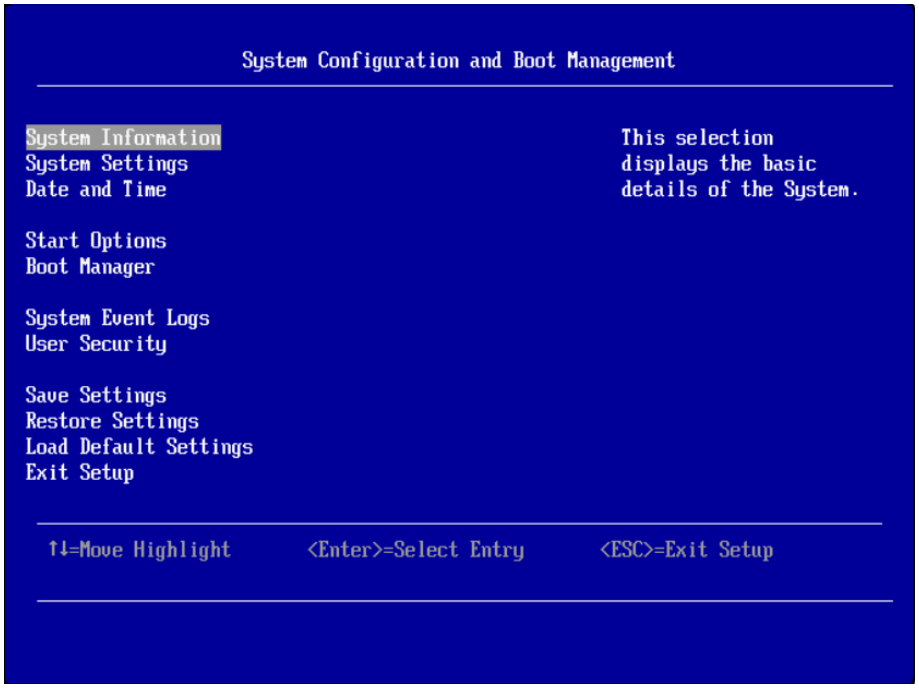

Figure A-2 Main selection window

3. In the System Settings window, select **Storage** to open the storage configuration panel, as shown in Figure A-3.

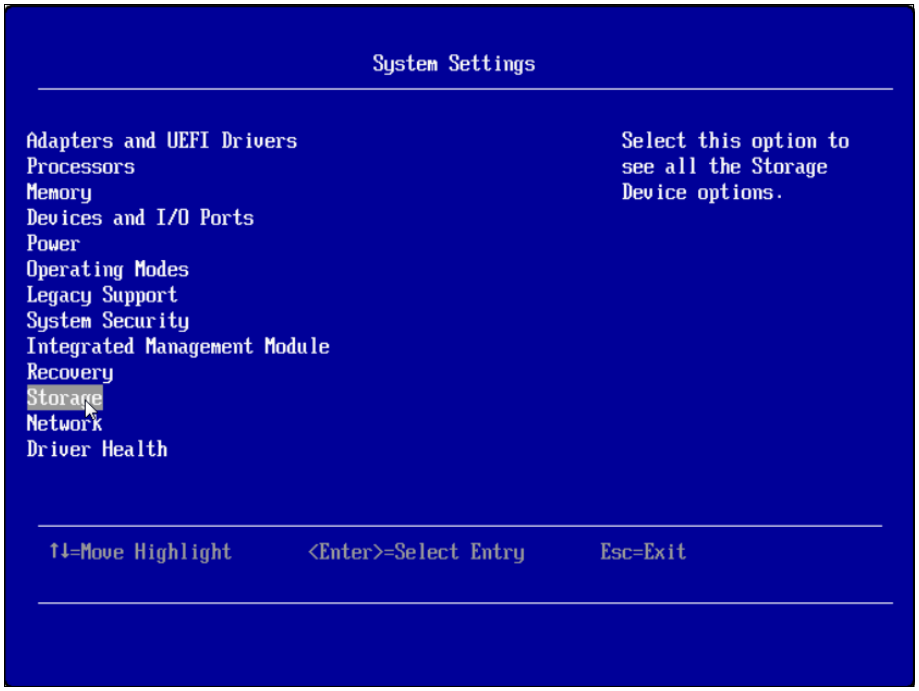

Figure A-3 Selecting Storage option

4. Select the RAID controller that you want to configure for drive security. If there are multiple adapters that are installed in the server, you must configure each of the controllers that are managing SEDs. In this window, select the RAID controller that is to be configured, as shown in Figure A-4.

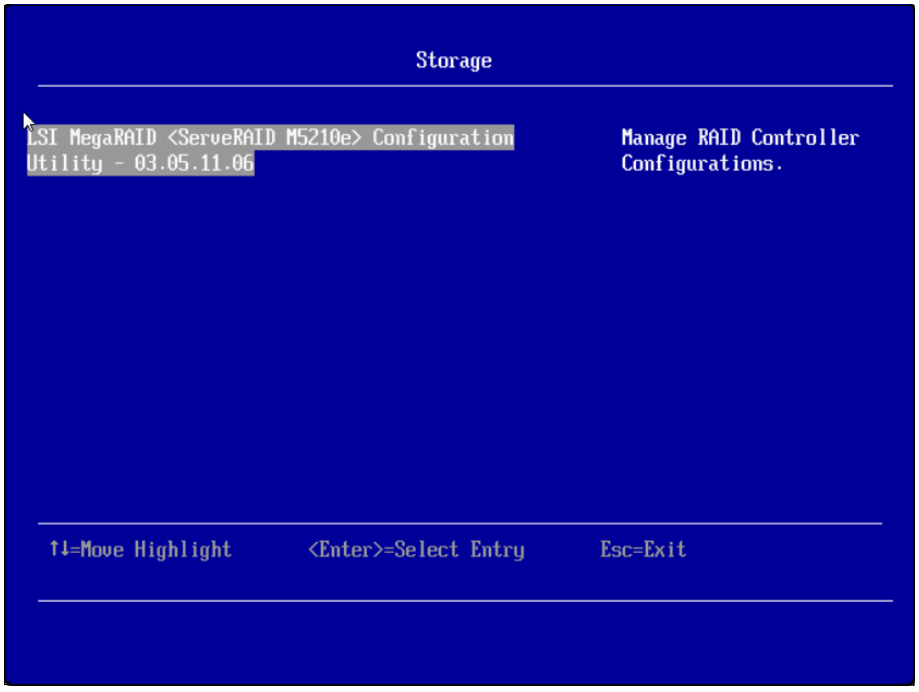

Figure A-4 RAID adapter selection

5. In the RAID Controller Management panel, select **Advanced**, as shown in Figure A-5.

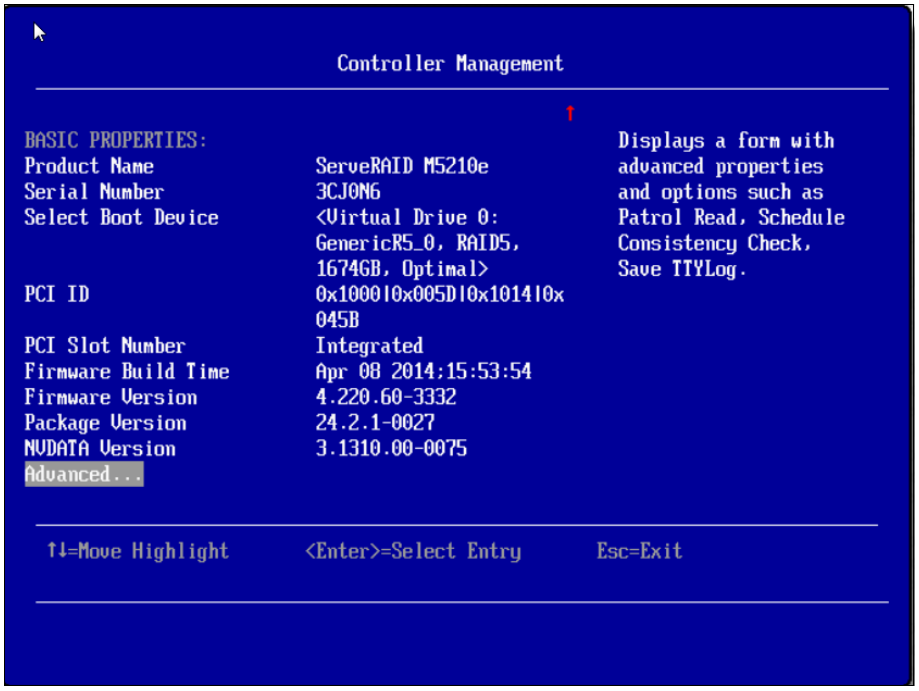

Figure A-5 Advanced management

You now must complete the steps in the following sections that match the required implementation scenario.

Enabling controller-based security (Scenario 2)

The following steps guide you through the configuration process for the RAID adapter as described in 1.3.2, “Scenario 2 encrypted: Unattended mode” on page 10.

Complete the following steps:

1. In the Advanced Management interface on the controller, select **Enable Drive Security** to browse to the security settings panel, as shown in Figure A-6.

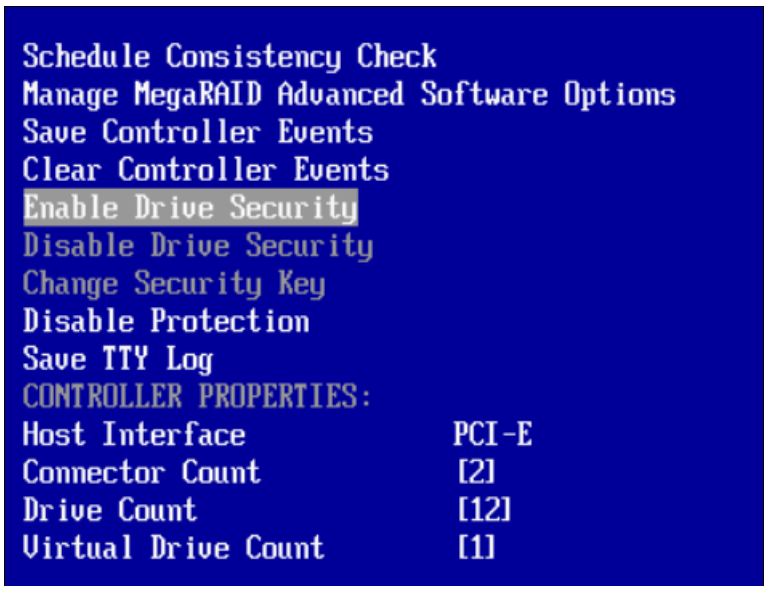

Figure A-6 Enable drive security

2. In the Choose Drive Security Mode dialog, ensure that Local Key Management is selected. Then, select **OK**, as shown in Figure A-7.

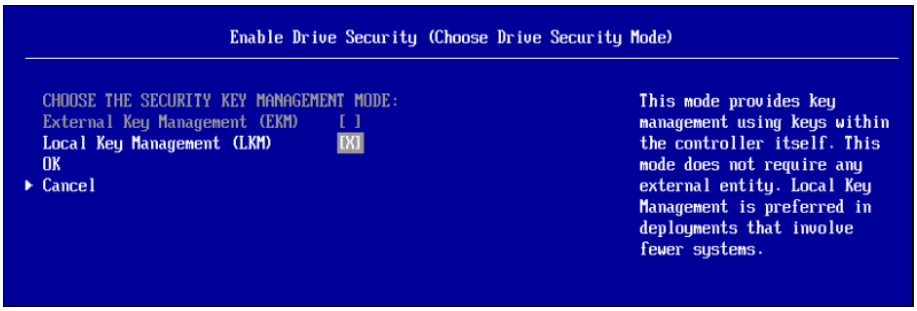

Figure A-7 Choose Drive Security Mode

3. In the Enable Drive Security configuration dialog, there are several options that can be configured, as shown in Figure A-8.

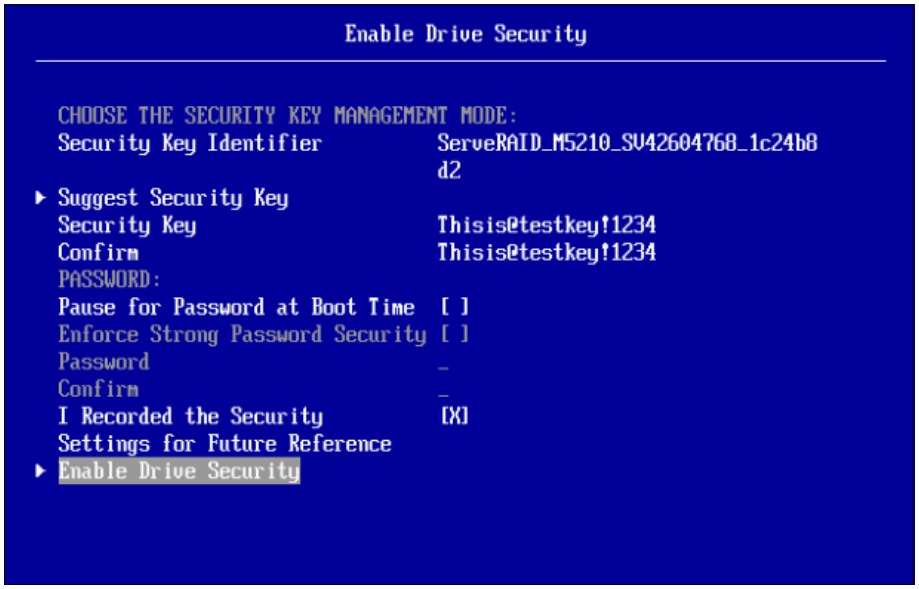

Figure A-8 Enable Drive Security

The following fields are available:

- Security Key identifier: This information is a simple text description for the key configuration on the adapter.
- Suggest Security Key: This action creates a random security key for the adapter. Use this option for the best security if a specific key is not required.
- Security Key: This field includes a random string that is generated by the Suggest Security Key action or can be created by the user if a specific value is preferred.
- Confirm: This field must be the same as the Security Key field for verification.
- Pause for Password at Boot Time: This option must be cleared for use in Scenario 2—Unattended Boot.
- Enforce Strong Password Security: This option enforces strong password rules on the boot time password field
- Password: This field contains the boot-time password if the Pause for password at boot time is selected. This field should remain blank for Scenario 2 deployments.

To configure the adapter for Scenario 2, ensure that a valid security key is entered and identically entered in the Confirm field. Next, ensure that the Pause for Password at Boot Time option is not selected.

4. Ensure that the created security key is documented. Then, select **I recorded the Security Settings for Future Reference**, and then select **Enable Drive Security**.

   A warning message opens that confirms that drive security is to be enabled, as shown in Figure A-9.

Figure A-9 Warning

5. Select **Confirm** and then select **Yes**.
6. When you return to the Advanced Controller Management dialog, select **Apply Changes** at the bottom of the list.

Enabling boot-time password (Scenario 3)

To set a RAID controller to conform to Scenario 3, as described in 1.3.3, “Scenario 3 encrypted: Attended mode” on page 11, complete the steps that are described in “Enabling controller-based security (Scenario 2)” on page 147. Then, complete the following steps:

1. Select **Change Security Key**, as shown in Figure A-10.

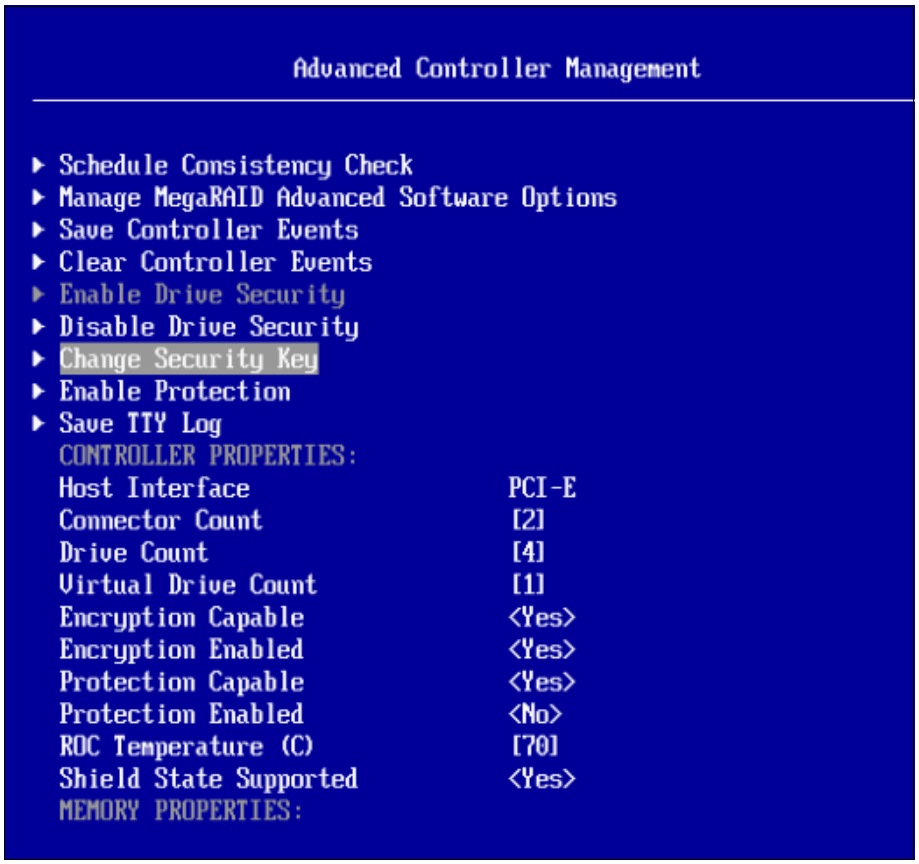

Figure A-10 Change Security Key

2. Ensure that **Change Current Security Settings** is selected. Then, select **OK**, as shown in Figure A-11.

Figure A-11 Change Current Security Settings

3. Select **Pause for Password at Boot Time** in the Change Security Key dialog, then select **Password** and enter the boot time password that must be supplied when the server boots**.** Next, select the **I Recorded the Security Settings for Future Reference** option to confirm that the documentation for the system was updated. Finally, select **Change Security Key** at the bottom of the list to commit the changes, as shown in Figure A-12.

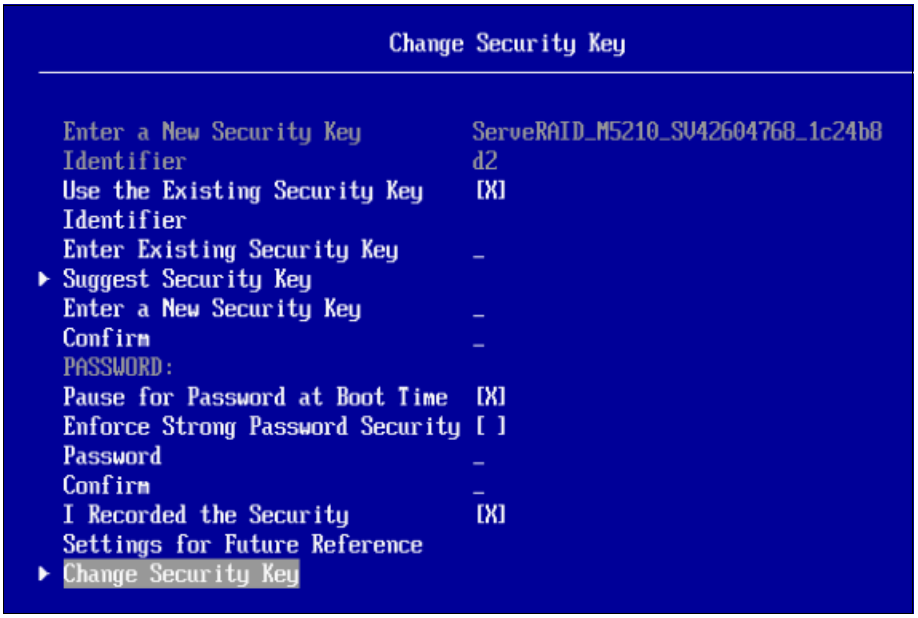

Figure A-12 Change Security Key

Modifying the security key

The security key can be changed on a controller at any time without data loss to secured virtual drives.

Complete the following steps to modify the security key of a configuration:

1. Browse to the Advanced Controller Management window and select **Change Security Key**, as shown in Figure A-13.

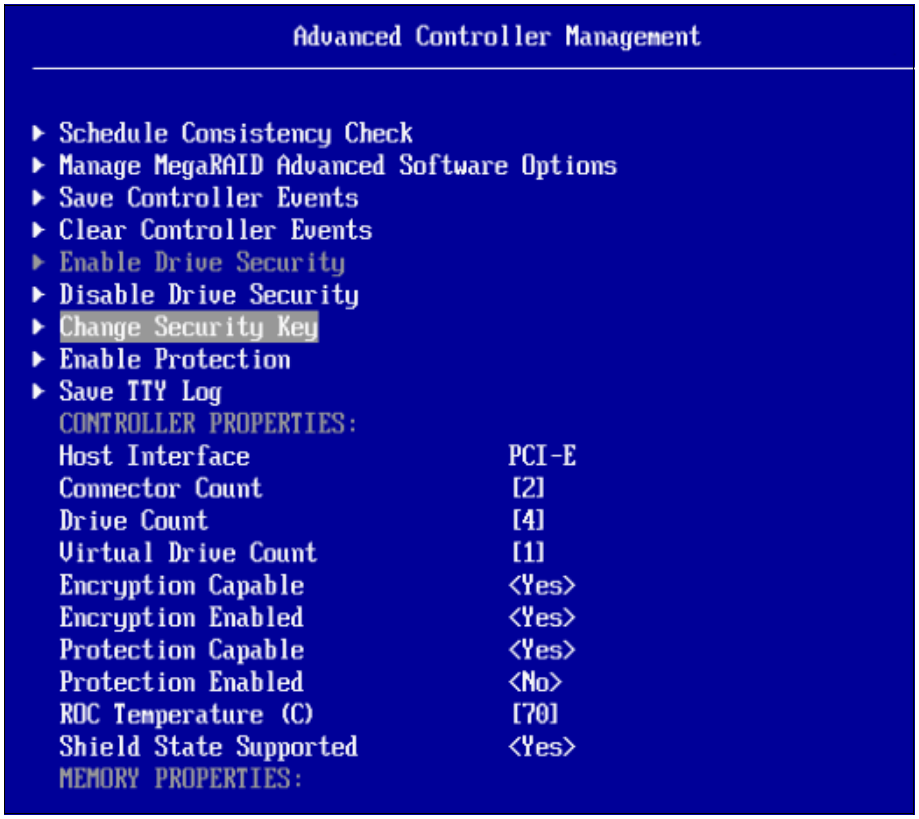

Figure A-13 Selecting Change Security Key option

2. Select **Change Current Security Settings**. Then, select **OK**, as shown in Figure A-14.

Figure A-14 Change Current Security Settings

3. Create a key by selecting **Suggest Security Key** to generate a new random key or by manually entering a key in the Enter a New Security Key field, as shown in Figure A-15.

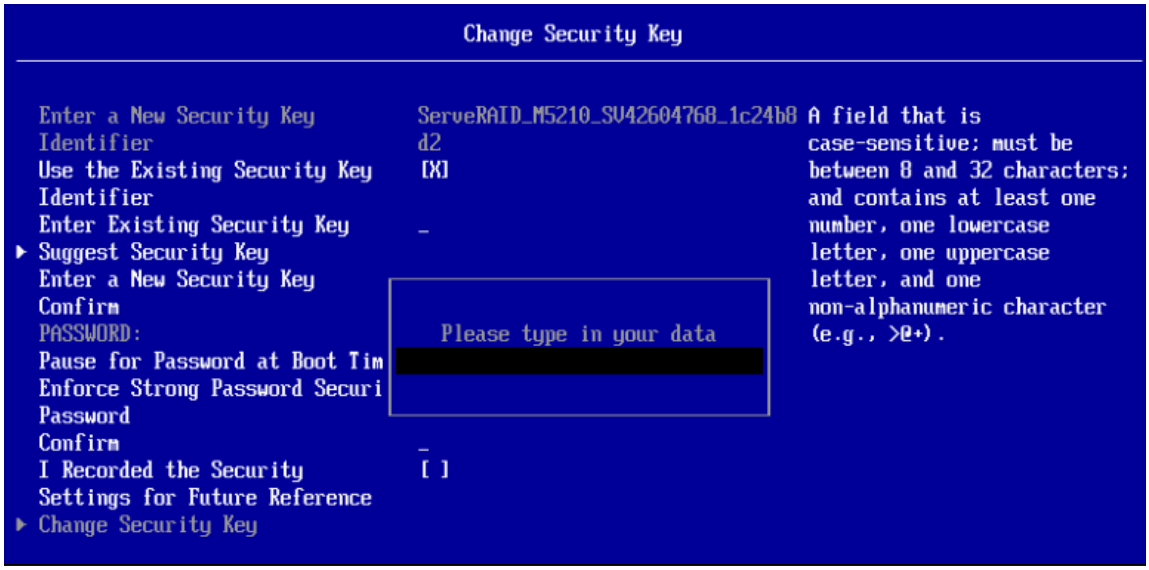

Figure A-15 Enter New Security Key

4. Ensure the security key documentation for the server was updated, then select **I Recorded the Security Settings for Future Reference** and **Change Security Key** to confirm the changes.

Creating and securing a virtual drive

Before you attempt to secure a virtual drive, ensure that the steps to configure the controller Drive Security settings were completed.

Complete the following steps to create a virtual drive and secure it with the controller security key:

1. Browse to the RAID controller Main Menu and select **Configuration Management**, as shown in Figure A-16.

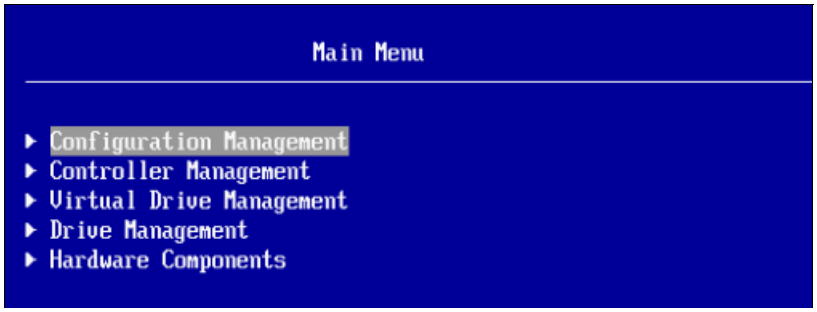

Figure A-16 Main Menu

2. In the Configuration Management dialog, select **Create Virtual Drive**, as shown in Figure A-17.

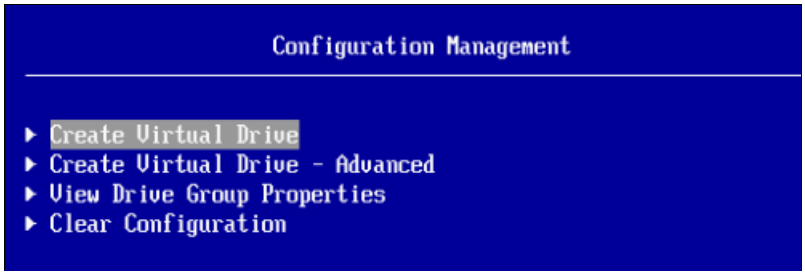

Figure A-17 Create Virtual Drive option

3. Select the **RAID type (RAID 0,1,5)** and press Enter.
4. In the Drive Selection Criteria option dialog, ensure that SEDs are selected (if there are mixed drive types in the system), as shown in Figure A-18.

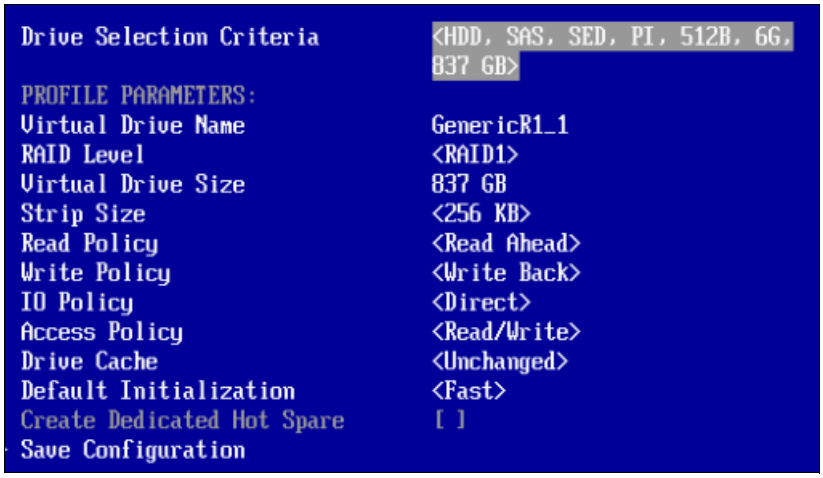

Figure A-18 Drive Selection Criteria

5. From the Main Menu for the RAID controller, select **Virtual Drive Management**, as shown in Figure A-19.

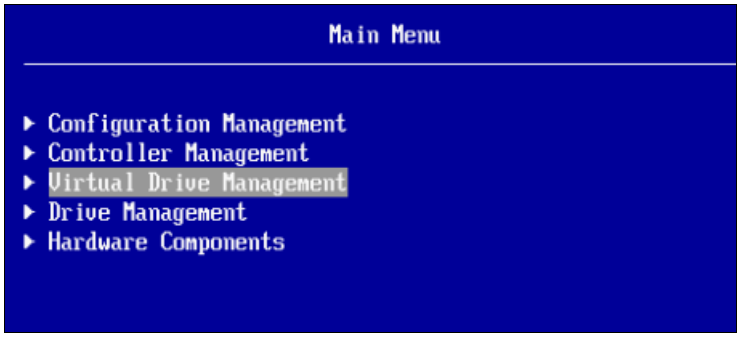

Figure A-19 Virtual Drive Management option

6. If multiple virtual drives are configured on the controller, select the drive that you want to secure and press Enter.
7. From the Virtual Drive configuration menu, select <**Select Operation**>, then choose **Secure Virtual Drive**, as shown in Figure A-20.

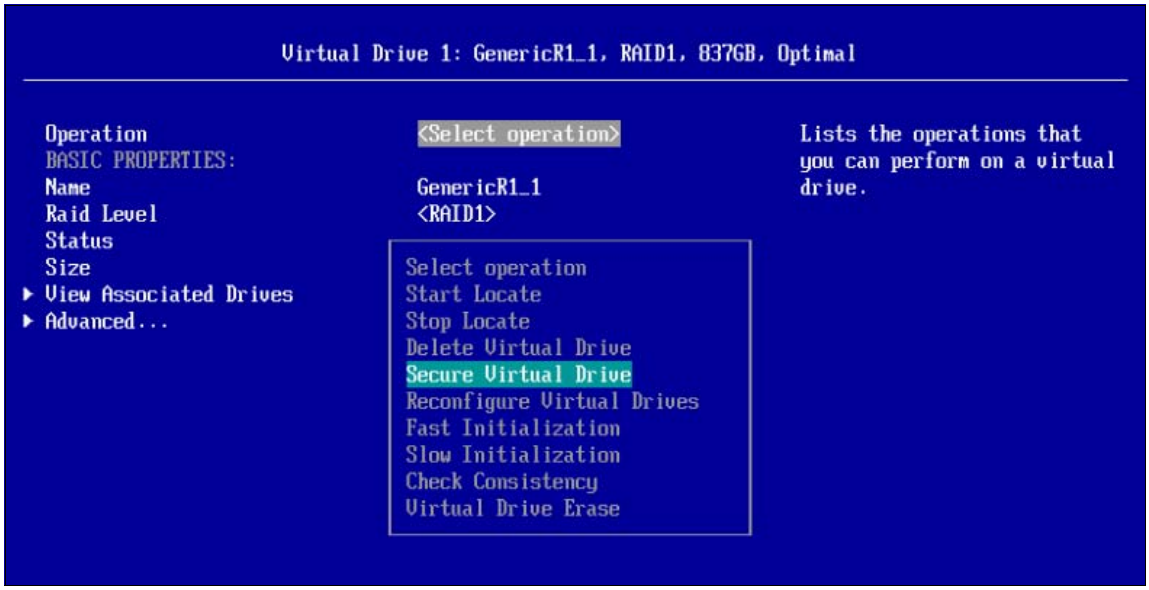

Figure A-20 Secure Virtual Drive option

8. In the Configure Virtual Drive properties dialog, select **Go**, as shown in Figure A-21.

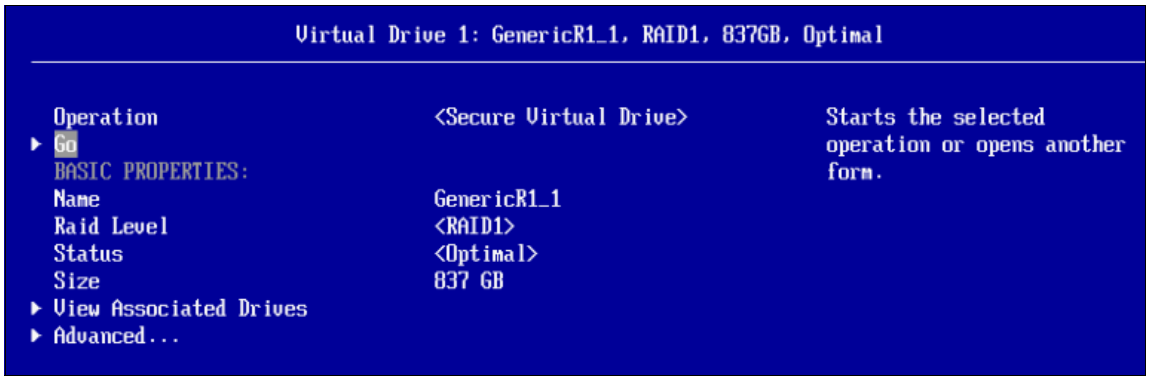

Figure A-21 Apply Secure Virtual Drive option

9. Read and understand the warning that is presented, which indicates that a virtual drive cannot be unsecured without the data on the array being lost. Select **Confirm** to continue. Then, select **Yes**, as shown in Figure A-22.

Figure A-22 Secure Warning

Enabling security on a virtual drive

To secure a virtual drive, complete step 5 on page 153 - step 9.

Configuring a Security Key on a replacement RAID adapter

If a controller is replaced because of a failure or problem determination procedure, it is critical that the security key (which was documented when Drive Security was activated) is entered on the new adapter to enable access to secured virtual drives.

To set the previous security key on the new adapter, complete the steps that are described in “Enabling controller-based security (Scenario 2)” on page 147. When you are performing these steps, ensure that when the Drive Security Key is entered, the key that is used on the previous adapter is entered instead of generating a key.

After these steps are completed, use standard procedures to import the virtual drive group configurations, which are listed as Secure Foreign Volumes.

Using the graphical MegaRAID Storage Manager

In this section, we describe the use of the graphical MegaRAID Storage Manager (MSM). It is assumed that you are familiar with the installation and basic use of the MSM tool to connect to an installed RAID controller. The scenario references for this section are described in 1.3.2, “Scenario 2 encrypted: Unattended mode” on page 10, and 1.3.3, “Scenario 3 encrypted: Attended mode” on page 11.

Enabling drive security on an installed RAID controller (Scenario 2)

Activating this option does not destroy any data that is on any of the configured virtual drives. After a virtual drive is set to protected mode, disabling this option results in the loss of access to the data and it must be restored from a backup source.

Begin the setup process by starting the MSM utility and entering credentials to access the target system or local system, as required. In a system with the RAID controller drive security set to disabled, the key icon that is next to the RAID adapter in the Physical tab is gray, as shown in Figure A-23.

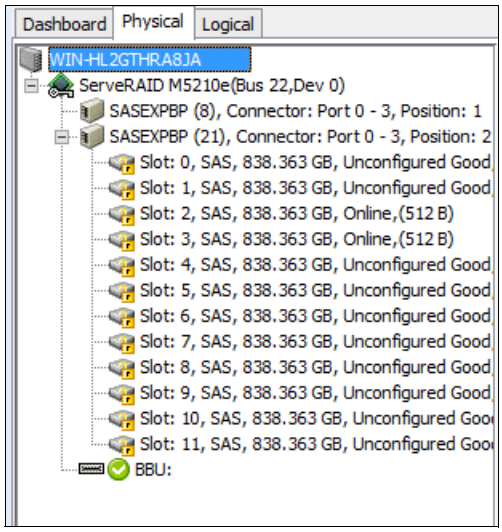

Figure A-23 Controller Security Disabled

1. Right-click **RAID controller** in the MSM utility to display the configuration options panel, as shown in Figure A-24.

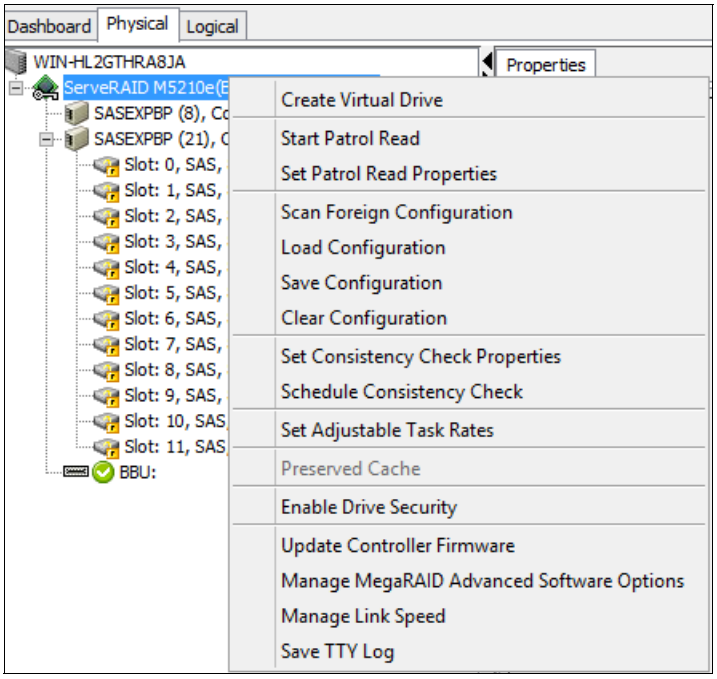

Figure A-24 RAID Adapter Options panel

2. From the RAID adapter options, select **Enable Drive Security**, which displays the configuration panel that is shown in Figure A-25. By using this configuration panel, you create the security key for the controller.

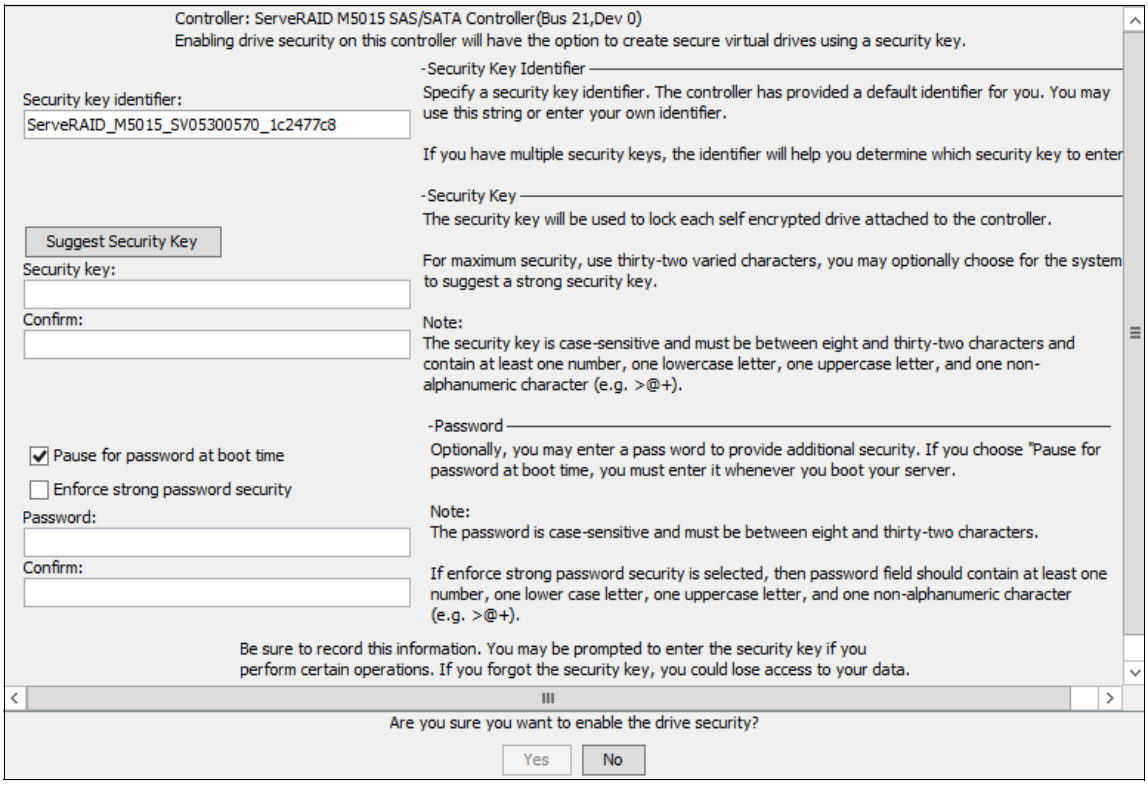

Figure A-25 Security Key Details panel

Within the Security Key Details configuration panel, the following options can be configured:

- Security key identifier: This text is a simple text description for the key configuration that is on the adapter.
- Suggest Security Key: By clicking this button, a random security key is created for the adapter. Use this option for the strongest security if a specific key is not required.
- Security key: This field includes a random string that is generated by the Suggest Security Key button or it can be filled by the user if a specific value is preferred.
- Confirm: This field must be the same as the Security key field for verification.
- Pause for password at boot time: This option must be cleared for Scenario 2—Unattended Boot.
- Enforce strong password security: This option enforces strong password rules on the boot time password field.
- Password: This field contains the boot-time password if the Pause for password at boot time option is selected. For Scenario 2 deployments, this field should remain blank.

3. In the Enable Drive Security configuration panel, create a controller key by selecting **Suggest Security Key** or entering a Custom Security Key that meets the strong password rules that were documented in the configuration panel.
4. Confirm the security key to be used in the Confirm dialog box.
5. Document the key that is to be used in some manner because this key is required to recover from a failed controller replacement. Failure to provide this key renders any data on the secured virtual drives inaccessible.

   Ensure that the Pause for password at boot time option is not selected.

6. Scroll down in the Enable Drive Security dialog to show the **I recorded the security settings for future reference** option, as shown in Figure A-26.

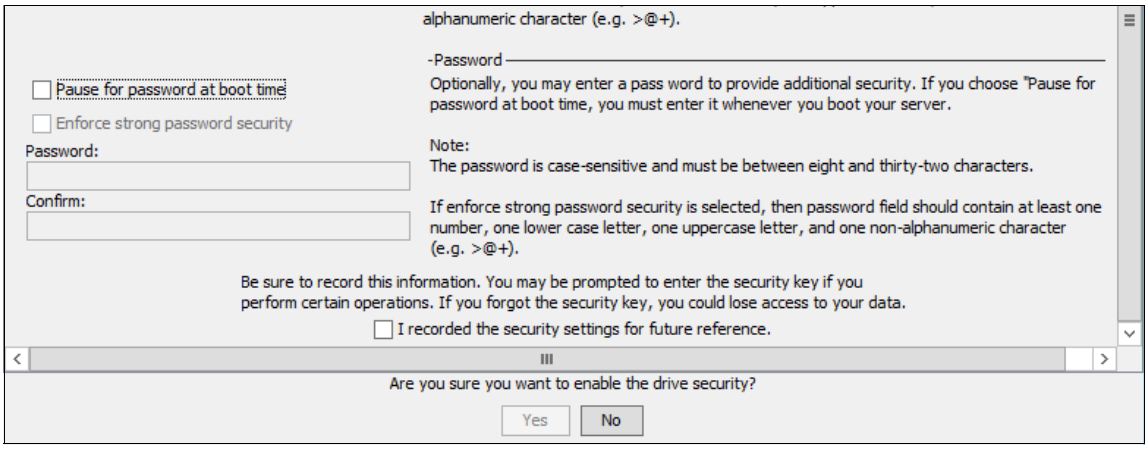

Figure A-26 Documentation verification

7. Ensure that any documentation regarding the security key in use is updated and stored for recovery purposes.
8. Select **Yes** to finalize the procedure.

After these steps are completed, you return to the main configuration window in MSM and a gold-colored key is shown next to the controller to indicate that security is enabled on that controller. Figure A-27 shows a controller with security enabled.

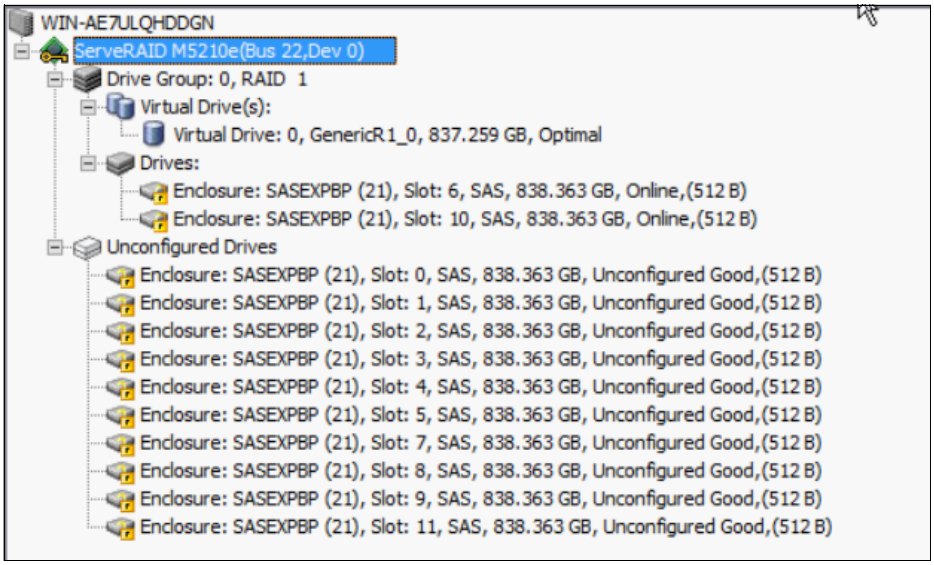

Figure A-27 Controller security enabled

Enabling boot-time password (Scenario 3)

Complete the following steps if the installation requires the configuration of a boot time password:

1. Complete the steps that are described in “Enabling drive security on an installed RAID controller (Scenario 2)” on page 155 to enable drive security on an installed RAID adapter.

2. Right-click the RAID controller in the MSM utility and select **Change Security Settings**, as shown in Figure A-28.

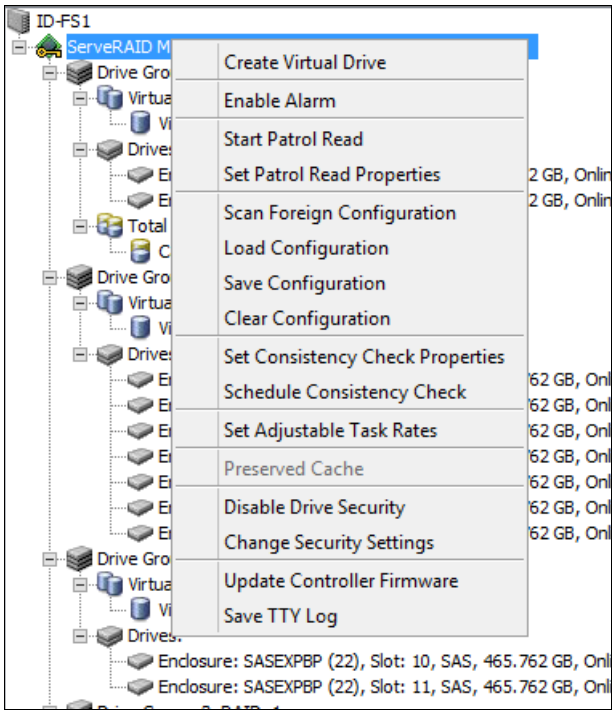

Figure A-28 RAID Controller options

3. In the Change Security Key Details dialog, select **Pause for password at boot time** option, as shown in Figure A-29.

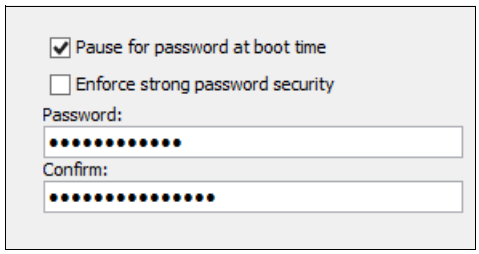

Figure A-29 Attended Mode setup

4. In the Password field, enter the password that is required during the boot process to allow the system to boot and access the encrypted drives. This password is not used to encrypt the drives; instead, it is a password that enables the controller to use the encryption key that was configured earlier to access the encrypted drives.

   This password is required when the server is rebooted while the Pause for password at boot time option is selected.

   Selecting this option does not cause data loss to data that is stored on the drives.

5. Select **OK** to complete the configuration process.

Modifying a controller security key

When Drive Security is enabled on a RAID controller, the security key can be modified at any time without any loss of data that us stored on the virtual drives.

Complete the following steps to modify the security key of a configuration:

1. Right-click the **RAID controller** in the MSM utility and select **Change Security Settings**, as shown in Figure A-30 on page 160.

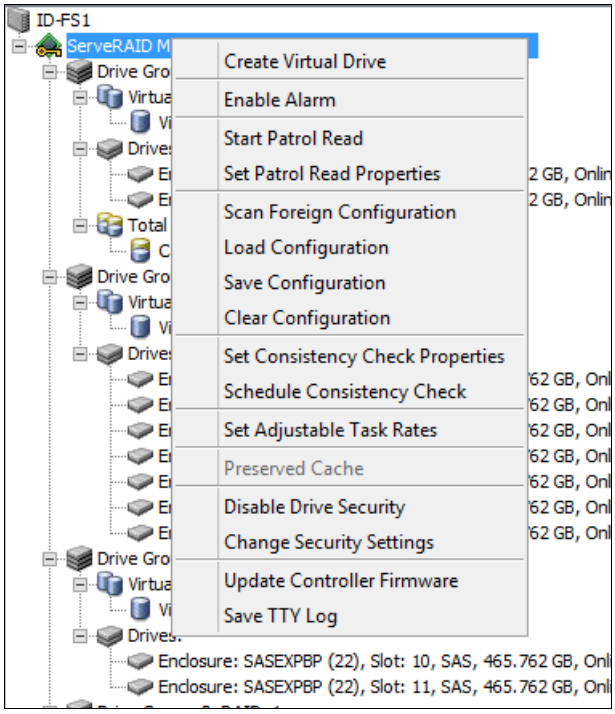

Figure A-30 RAID adapter options

2. In the Change Drive Security configuration panel, select **Enter a new security key identifier**, as shown in Figure A-31.

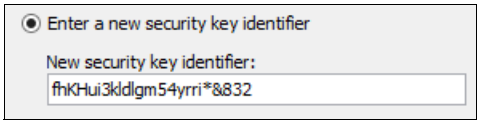

Figure A-31 Enter new security key

3. Scroll down in the Enable Drive Security window to show the **I recorded the security settings for future reference** option and select it, as shown in Figure A-32.

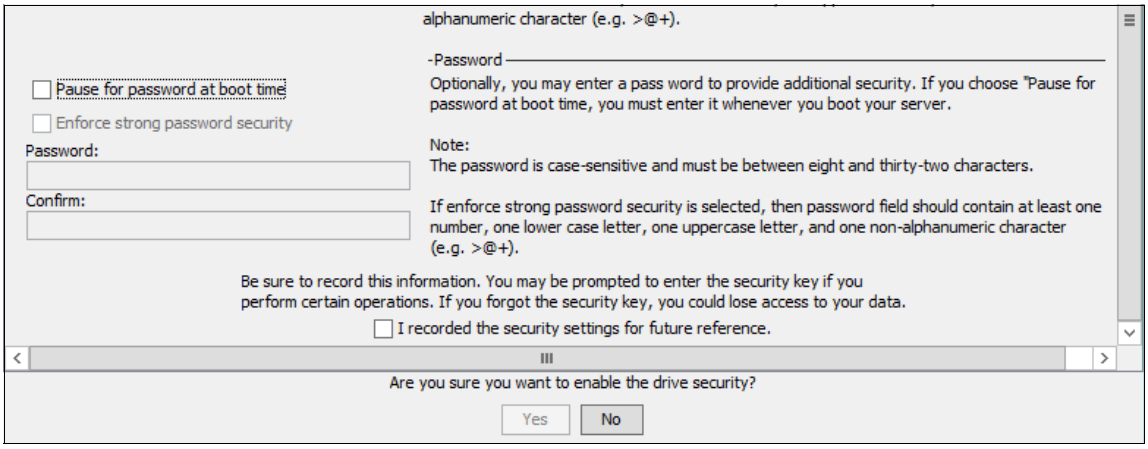

Figure A-32 Documentation verification

4. Ensure that any documentation regarding the key that is used on the server is updated to reflect the change in the security key.
5. Select **Yes** to close the dialog box and apply the changes.

Creating a secured virtual drive

Complete the following steps to create a virtual drive that is automatically secured at the time it is created:

1. Enable drive security by using the UEFI or MSM method.
2. From the Dashboard tab of the MSM utility, select **Create virtual drive**. The virtual drive configuration wizard starts, as shown in Figure A-33.

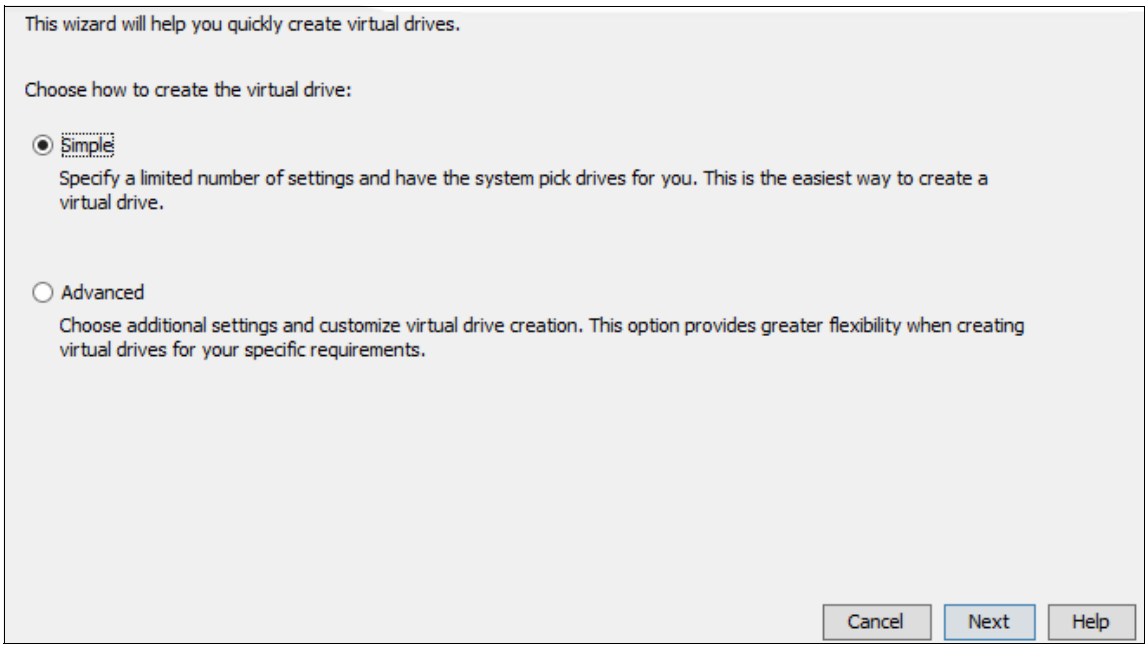

Figure A-33 Creating virtual drives

3. For the purposes of this example select **Simple**. Click **Next**.
4. Select **Use unconfigured drives**, as shown in Figure A-34. Click **Next**.

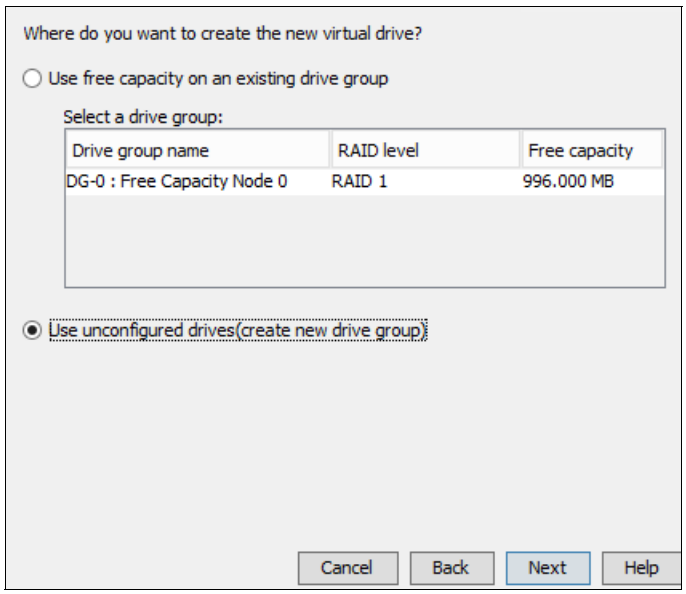

Figure A-34 Allocate capacity

5. Choose the appropriate RAID level, select the **Use drive security** option, and then click **Next**. In this example, a simple RAID 1 virtual drive with 4300 GB SEDs was created.

   The completed dialog box is shown in Figure A-35.

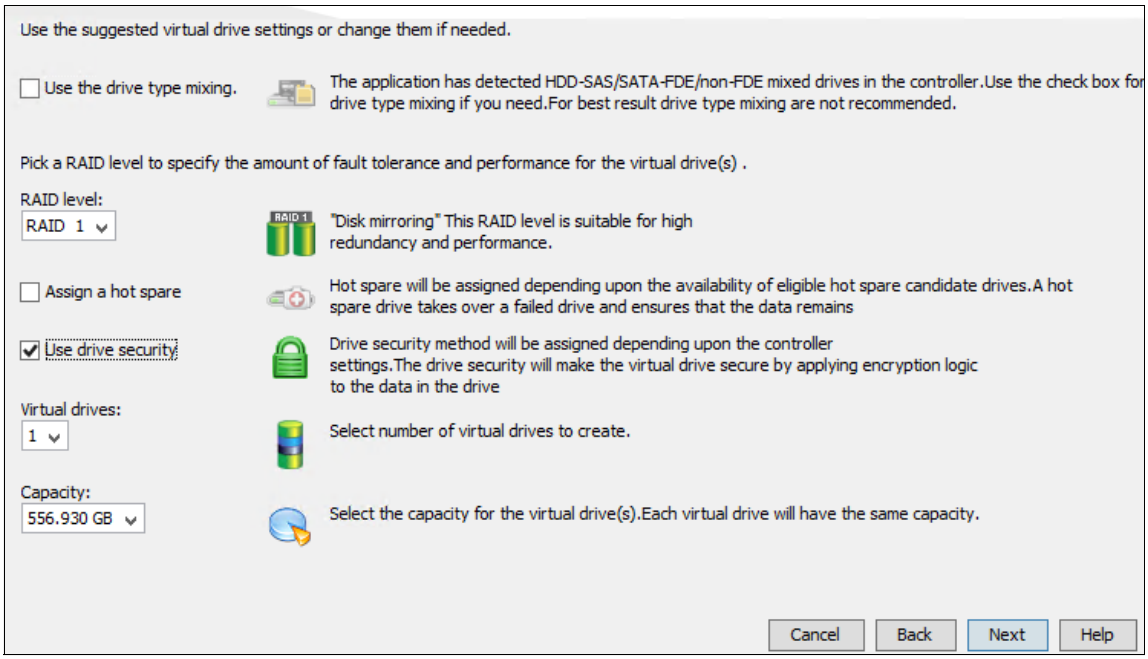

Figure A-35 Virtual drive settings

6. After the Create Virtual Drive Summary dialog is shown, select **Finish**.

   A virtual drive is created that can be viewed in the Logical tab of the MSM utility. The drives that are selected as elements of the array are shown under the virtual drive with gold-colored padlocks in the closed position next to them. These icons indicate that the drives are in secured mode, as shown in Figure A-36.

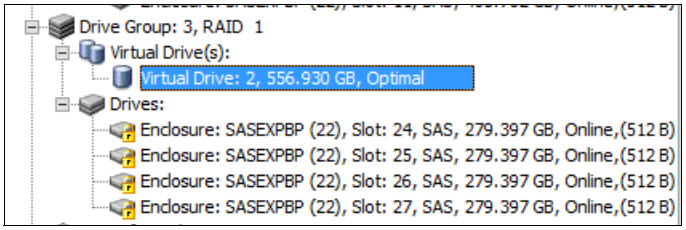

Figure A-36 Secured virtual drive

When you select the secured virtual drive in the properties window of the MSM, you can see an indication that the drives are secured, as shown in Figure A-37.

Figure A-37 Secured drive properties

Securing a virtual drive

The following procedure is used to activate virtual drive encryption on an array. This virtual drive often is created by using any standard method of virtual drive management as described in the MegaRAID user guide.

Securing a virtual drive: Securing a virtual drive must be done at the Drive Group level. Securing a virtual drive in a Drive Group with multiple virtual drives configured is not supported.

A mix of SED and non-SED Drive Groups on a single controller is supported. Also, it is supported to have some SED Drive Groups that are secured while others are not secured on the same RAID controller.

Complete the following steps to secure a virtual drive:

1. Verify that the drives in the target virtual drive can support drive encryption. This verification can be done by validating the part numbers of the drives, or in the MSM utility. SED drives appear with a gold-colored padlock next to the disk, as shown in Figure A-38.

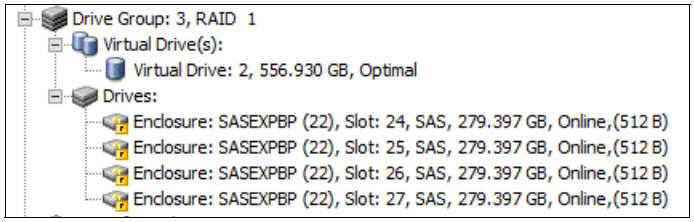

Figure A-38 Unsecured SEDs

Also, encryption capability can be validated by selecting the drives in the virtual drive and verifying their capability in the Drive Properties section of the MSM utility. An example of an SED that is unsecured is shown in Figure A-39.

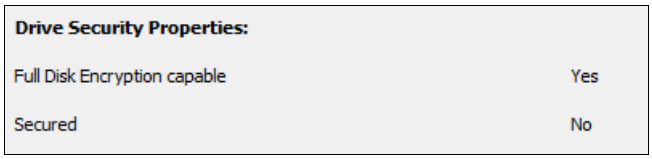

Figure A-39 SED drive properties

2. Right-click the Drive Group that includes the virtual drive that is to be secured to see the available options. In this example, an unsecured four-drive RAID 1 virtual drive was created. Figure A-40 shows the options that are available to the drive group.

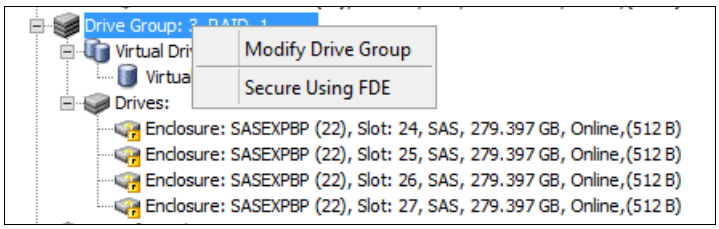

Figure A-40 Secure by using FDE

3. Select **Secure using FDE**. (FDE refers to Full Disk Encryption, which is the method that is used to enable the drive security on the Drive Group.)
4. Select the **Confirm** option. Make note the warning (as shown in Figure A-41). Click **OK**.

Figure A-41 Confirm Secure Drive Group panel

5. Verify that the Drive Group was secured. This confirmation can be done by visually inspecting the status of the padlock icons next to the drives in the MSM utility, as shown in Figure A-42.

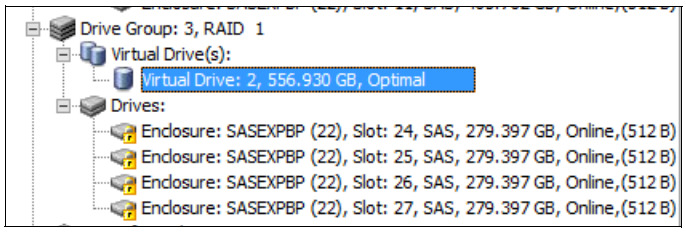

Figure A-42 Secured virtual drive

Also, the virtual drive's properties are updated to reflect that the drive was secured, as shown in Figure A-43.

IO Policy	Direct IO
Write Policy:	
Current Write Policy	Write Back
Default Write Policy	Write Back with BBU
Access Policy:	
Current Access Policy	Read Write
Default Access Policy	Read Write
Drive Security Properties:	
Secured	Yes

Figure A-43 Secured drive properties

Disabling security on a controller

Disabling drive security on a controller with secured virtual drives results in the loss of data.

Unsecuring a virtual drive: There is no method to unsecure a single virtual drive. If you must remove encryption from a single virtual drive and preserve data on other virtual drives, do not disable controller security. The method to remove encryption from a single virtual drive is to delete that Drive Group, which removes the data from that virtual drive and returns the drives to an unsecured state that are ready for configuration into a new virtual drive.

Complete the following steps to disable security on a controller:

1. All secured Drive Groups that are configured on the controller must be deleted by using standard procedures for deleting virtual drives as described in the appropriate MegaRAID Controller Users Guide.
2. Right-click the target controller in the MSM utility to show the controller options, as shown in Figure A-44.

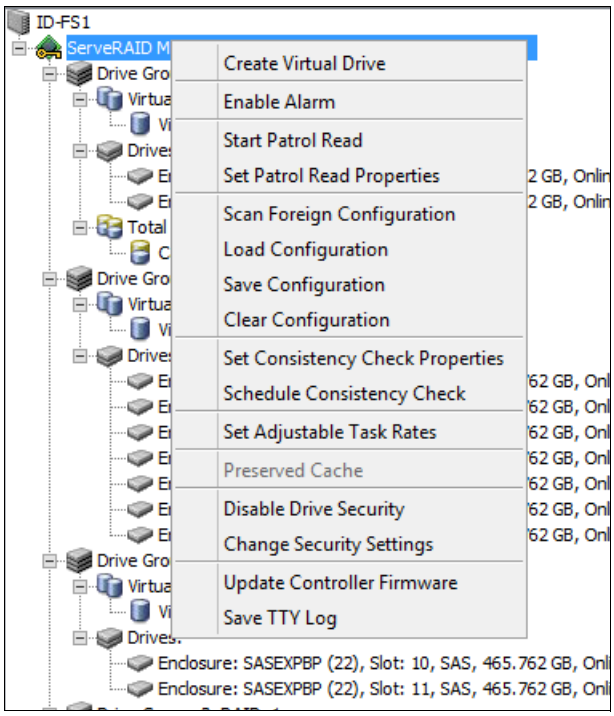

Figure A-44 Disable Drive Security option

3. Select **Disable Drive Security**. The warning that is shown in Figure A-45 opens.

Figure A-45 Confirm Disable Drive Security panel

4. Click **Yes** after you read the warning.

The gold key that is next to the controller in the MSM utility changes from a gold color to a gray color, which indicates that the controller does not have Drive Security enabled.

Replacing a controller with secured virtual drives

If a RAID controller that is configured with secured virtual drives is replaced because it failed, the RAID controller must be configured with the same security key that was used to secure the drives initially. Failure to do so renders the drives inaccessible.

Boot drive implications: If the boot drives for the operating system were attached to the replaced controller and were secured, the MSM is unavailable to configure the security key. The security key must be configured by using the UEFI to gain access to the boot drives.

To set the security key on a controller, complete the steps that are described in “Enabling drive security on an installed RAID controller (Scenario 2)” on page 155. Ensure that the key that is configured in this process is identical to the key that is used when the drives were secured initially.

Conclusion

In this appendix, we described the required steps to configure an installed M51xx or M5200 xx RAID controller in a System x server for use in local security key managed environments, which were described as Scenario 2 and 3 in Chapter 1, “Technology primer” on page 1.

For more information about managing the M series RAID adapters from System x, see the Installation and User Guides for the respective adapters.

B

Troubleshooting

In this appendix, we provide troubleshooting hints and tips. Although the information in this appendix is not a complete list of all possible errors and outcomes, it does provide many common issues and resolutions that we encountered during testing for this publication.

This appendix includes the following topics:

- IBM SKLM installation, update, and login issues
- IMM configuration
- Unified Extensible Firmware Interface issues

© Copyright Lenovo 2015. All rights reserved.

IBM SKLM installation, update, and login issues

In this section, we provide information to aid you with some errors, warnings, and issues you can encounter while setting up your IBM Security Key Lifecycle Manager (SKLM) environment.

Error message: Problems were found with the packages and fixes in package group IBM WebSphere Application Server V8.5

The update process during the SKLM installation displays all available fixes. Some of these fixes might not be applicable to your installation. If you proceed with all packages or some non-applicable packages selected, an error can occur, as shown in Figure B-1 on page 170.

Change your package selections to apply to the WebSphere Application Server version that you want to install and the architecture of your operating system; WinX32 only for 32-bit Windows and WinX64 only for 64-bit operating systems.

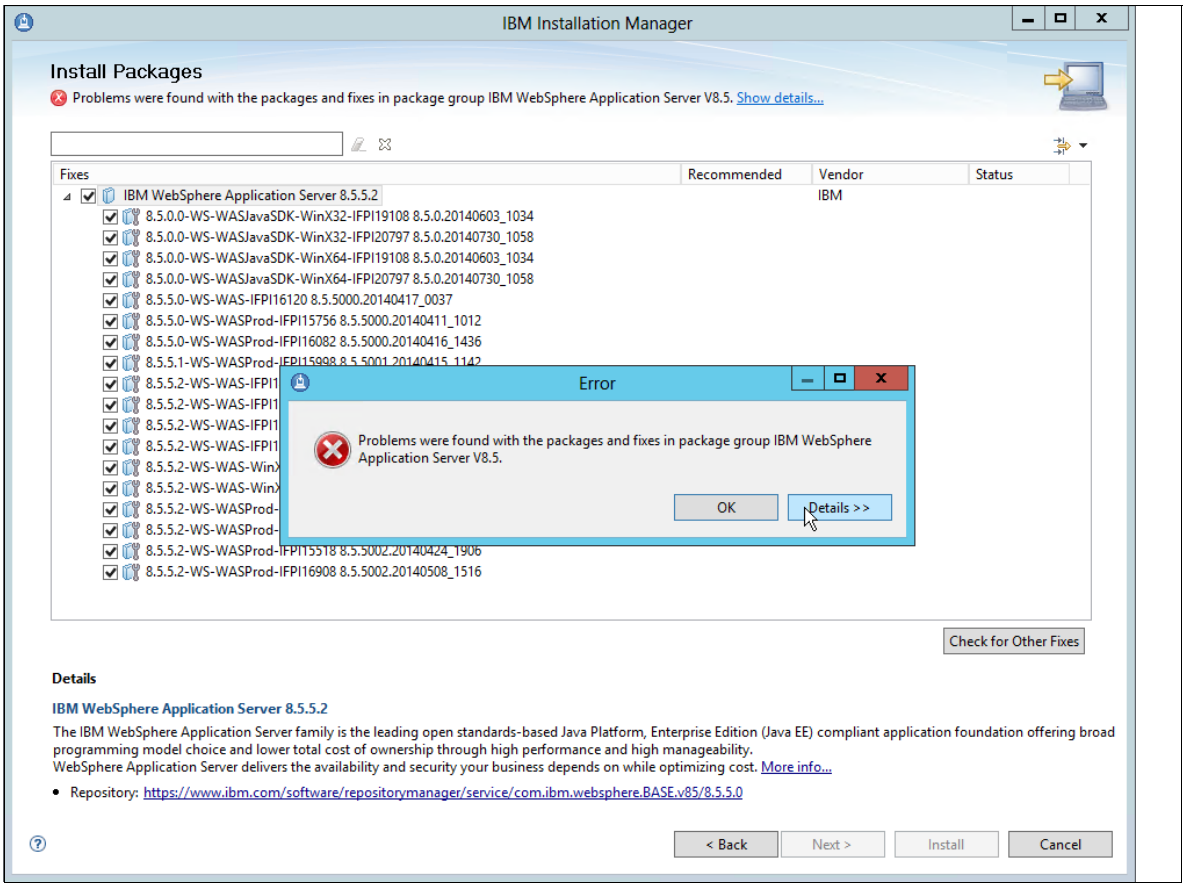

Figure B-1 Package installation error

SKLM web interface fails to load with JSP Processing Error

This error occurs when an incorrect case is used in the SKLM URL. A common problem is the use of the following URL:

```
https://[SKLM IP address]:9080/ibm/sklm/login.jsp
```

Instead, use the following URL format:

`https://[SKLM IP address]:9080/ibm/SKLM/login.jsp`

where SKLM is uppercase instead of lowercase.

This discrepancy is shown in Figure B-2 on page 171.

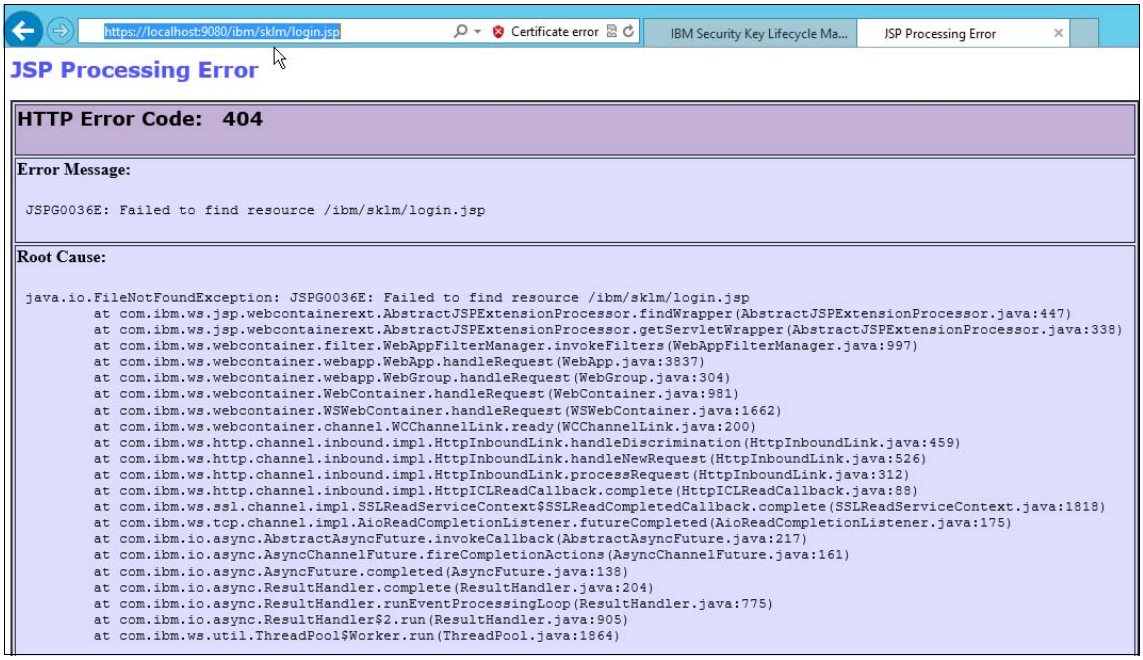

Figure B-2 URL case sensitivity

Cannot install Installation Manager on RHEL 6.0/6.1 (64-bit)

You might encounter an issue in which IBM Installation Manager cannot be installed or started on a 64-bit Linux machine. This error occurs because Installation Manager is a 32-bit application.

See the following website to help you install the necessary 32-bit libraries on your RHEL system:

`https://www.ibm.com/support/docview.wss?uid=swg21459143`

IMM configuration

In this section, we highlight the following issues that can be encountered when you are configuring the Integrated Management Module (IMM) for external SED key management:

- Security certificate not trusted error
- Test Connection non-responsive
- IMM certificate upload error
- Adding key management server error

Security certificate not trusted error

When you connect to the IMM controller on a System x server by using an `https` browser connection without the use of a properly signed certificate, you receive a security certificate not trusted error, similar to the error that is shown in Figure B-3. The exact format of the error can vary based on the browser that is used.

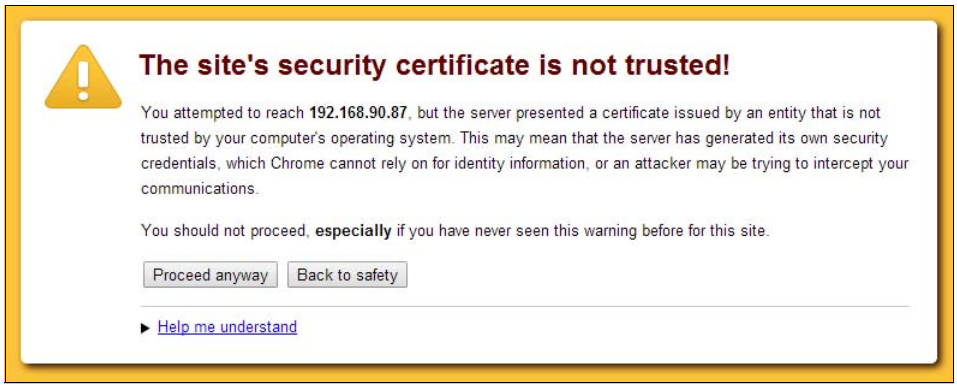

Figure B-3 Security certificate not trusted error

This error is the result of the use of a self-signed certificate for the https communications. To resolve the problem, use a security certificate that is signed by a signing authority or click **Proceed anyway** to continue with the self-signed certificate.

Test Connection non-responsive

When you configure the external key management servers and attempt to test the connection, the resulting web page might appear to be non-responsive. This error can occur because the option for the target server to be tested is not properly selected, as shown in Figure B-4.

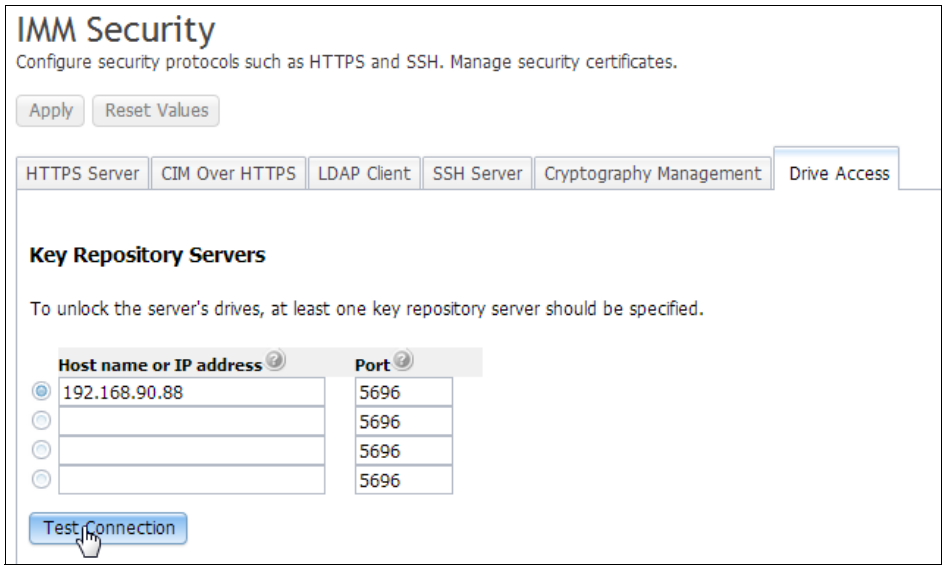

Figure B-4 Test Connection error

To resolve this problem, select the appropriate option and click **Test Connection**.

IMM certificate upload error

You might receive a Certificate upload error (as shown in Figure B-5) during the import process of the key management server certificate to the IMM.

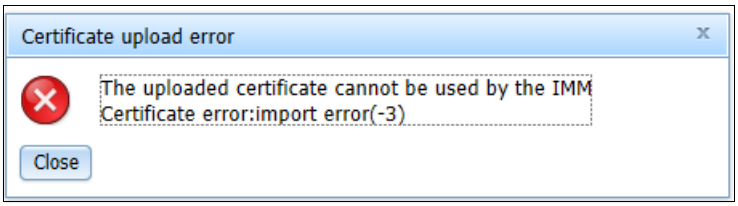

Figure B-5 Certificate upload error

This error often occurs because the time and date are configured incorrectly on the IMM to which you are trying to upload the certificate. This error can occur for the following reasons:

- The server was correctly configured.
- The system board was replaced and the server was not reconfigured correctly.
- The CMOS was reset on the server and the server was not reconfigured correctly.
- The certificates include time stamps and finite lifespan that are associated to the file. If an IMM is at the default date of 2000, the certificate expires according to the IMM.

The corrective action is to ensure that the time and date are set correctly on the IMM before continuing to configure the system. For more information about how to correctly configure your servers, see 4.2, “Configuring the IMM by using the web-based interface” on page 86.

Adding key management server error

During our proof of concept, we encountered a scenario in which the IMM did not accept the entries for the key management server and a server addition error was displayed, as shown in Figure B-6.

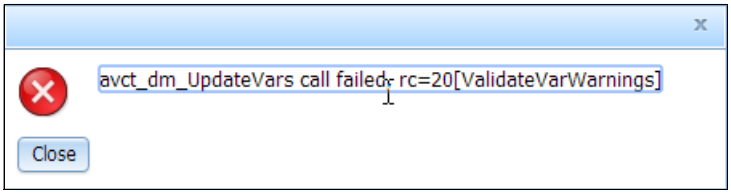

Figure B-6 Server addition error

The solution for this error is to use the IMM command-line utility to add the entries directly. This task is done by logging in to the command-line telnet session to the IMM and using the following **storekeycfg** command:

```
storekeycfg -sxip <host name/ip_addr> - server x host name/ip addr (x can
be 1, 2, 3 or 4)
```

Adding a server with this command to position 1 of 4 is shown in the following example:

```
storekeycfg –s1ip 1.2.3.4
```

To perform the same command (except for position 2 of 4), the following command is used:

```
storekeycfg –s2ip 1.2.3.4
```

Use the **storekeycfg** command with no other parameters to retrieve a list of the configured servers, as shown in Example B-1.

Example: B-1 List of currently configured servers

```
storekey-server Trusted Certificate: Available.
s1ip:   192.168.90.87  s1pn:  5696
s2ip:                  s2pn:  5696
s3ip:                  s3pn:  5696
s4ip:                  s4pn:  5696
Group device: IBM_SYSTEM_X_SED
```

Unified Extensible Firmware Interface issues

This section describes issues that you might encounter when you configure the Unified Extensible Firmware Interface (UEFI) components of the solution.

UEFI boot error

During the early boot process, the server might experience the communication error with the EKMS prompting the user for input that is shown in Figure B-7.

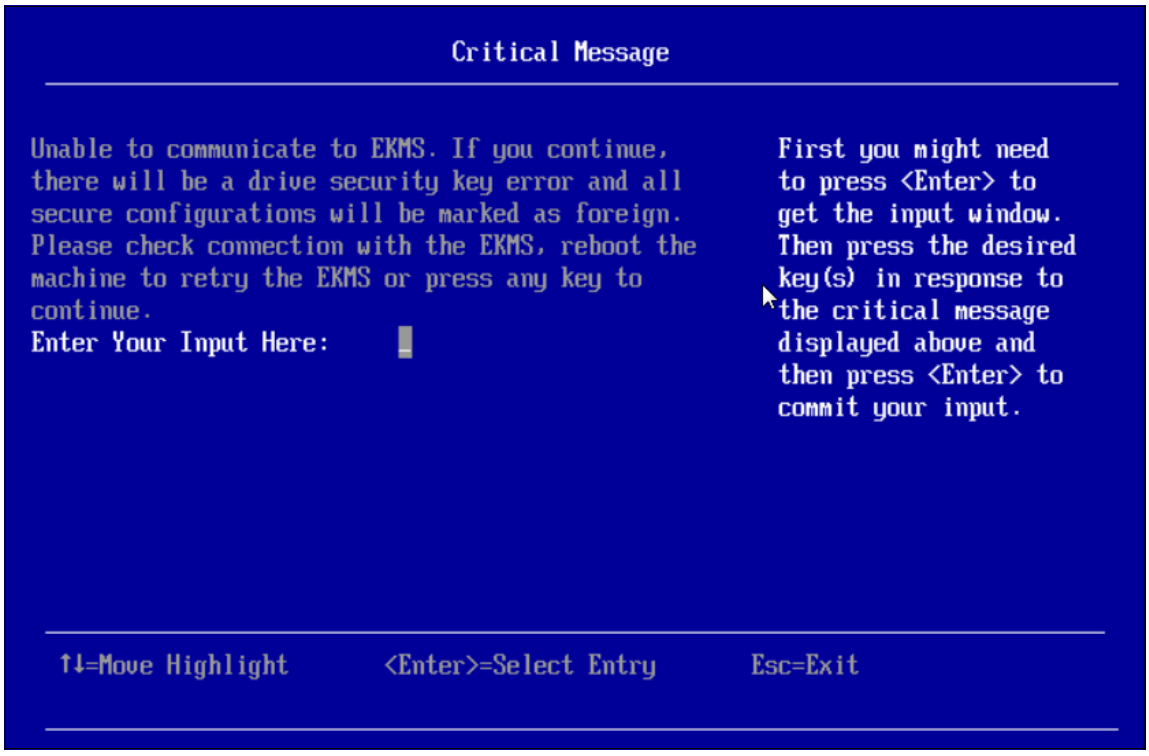

Figure B-7 UEFI boot error

This error message indicates that the server cannot communicate with the key management server. This error can be caused by the following issues:

- The server certificate is still pending acceptance on the key management server.

  Fix: Accept the server connection from the key management server interface.

- The IMM network connection was disconnected.

  Fix: Re-establish the network connection to the IMM adapter.

- The IMM network configuration is not configured properly.

  Fix: Ensure that the IMM network settings (including default gateway and DNS, if necessary) are configured correctly. If DHCP is in use, ensure that the IMM can communicate with the DHCP server.

- A RAID adapter with an existing configuration for external key management was installed in a server that is not set up for remote key management.

  Fix: Ensure that the IMM and UEFI of the server are configured to allow the server to establish communications with a remote key management server, as described in Chapter 5, “UEFI configuration” on page 109.

Conclusion

In this troubleshooting guide, we provided some basic tips for situations that you might encounter when you configure a System x server for remote key management.

This appendix is not intended as a general System x troubleshooting guide.

C

Licenses and software

In this appendix, we describe the required products and features to successfully deploy centralized key management for System x servers with self-encrypting drives (SEDs) that are managed by IBM Security Key Lifecycle Manager (SKLM).

This appendix includes the following topics:

- SKLM for System x SEDs Feature on Demand
- IBM Security Key Lifecycle Manager Basic Edition

© Copyright Lenovo 2015. All rights reserved.

SKLM for System x SEDs Feature on Demand

When you create an environment or expand the capabilities of an existing environment, you must verify that the selected server and RAID adapter are supported for the SKLM for System x w/SEDs FoD option.

The lists that are supported at time of this writing are described in Chapter 2, “Supported systems and sample configuration” on page 17.

The SKLM for System x SEDs - FoD is listed in ServerProven under System Management Upgrades, which is available at this website:

`http://www.ibm.com/systems/info/x86servers/serverproven/compat/us/xseries/upgrades/smmatrix.html`

Supported ServeRAID controllers require an upgrade to support the encryption functions of SEDs. The RAID upgrade options with SED support vary by controller and are described in 2.1.2, “Supported RAID adapters” on page 18. Controllers without the upgrades still accept SED drives as regular devices, which enables them to be used as conventional drives.

Purchase the SKLM for System x SEDs - FoD option

The SKLM for System x SEDs – FoD option can be purchased for use with previously acquired servers or included on the order for a new server.

The part numbers vary by geography and are listed in Table C-1. These part numbers are available with one- or three-year subscriptions and support (there is no difference in functionality between the two parts).

Table C-1 SKLM for System x SEDs – FoD options

Description	US and Canada and AP	EMEA and LA	Feature Code
SKLM for System x w/SEDs - FoD per Install w/1Yr S&S	00D9998	00FP648	A5U1
SKLM for System x w/SEDs - FoD per Install w/3Yr S&S	00D9999	00FP649	AS6C

The option is licensed on a per server basis. You need to purchase only one of the listed part numbers for each server where you want to activate the FoD.

The part numbers that are listed in Table C-1 on page 178 include authorization for the System x server to connect to the SKLM Basic Edition software product (as described in “Activate the Feature on Demand”) and receive key management services. No other parts are required to deploy the solution.

Activate the Feature on Demand

If you purchase a new server and your configurator supports adding the SKLM for System x SEDs – FoD option to the server, the FoD is activated as part of the server build process in manufacturing.

If you purchase the SKLM for System x SEDs – FoD option separate from the server or your chosen configurator does not support adding the option to the server, you receive an FoD authorization code and instructions for obtaining an FoD activation key to be applied at the IMM on your System x server.

For more information about the FoD activation process, see *Using Features on Demand*, REDP-4895, which is available this website:

http://lenovopress.com/redp4895

IBM Security Key Lifecycle Manager Basic Edition

IBM Security Key Lifecycle Manager Basic Edition (previously known as Tivoli Key Lifecycle Manager) is the IBM key management software product that System x servers interact with to obtain the key (KEK) that is required to access the SEDs. SKLM provides key management services to various endpoint devices beyond System x servers with SEDs.

For more information about SKLM Basic Edition software and supported devices, see this website:

http://www.ibm.com/software/products/en/key-lifecycle-manager

Purchase IBM Security Key Lifecycle Manager Basic Edition

SKLM Basic Edition is available in the Passport Advantage ordering system under the part numbers that are listed in Table C-2 on page 179. A single license allows for a primary and backup SKLM server to be deployed. For more information about ordering the product, see this website:

https://www.ibm.com/software/howtobuy/buyingtools/paexpress/Express?PO=E1&part_number=D0887LL&catalogLocale=en_US&Locale=en_US&country=USA&PT=jsp&CC=USA&VP=&TACTICS=&S_TACT=&S_CMP=&brand=none

Table C-2 SKLM Basic Edition part numbers

Description	Part number
SKLM Basic Ed per Installation LIC+SW S&S 12 Mo	D0887LL
SKLM Basic Ed per Installation Annual SW S&S Rnwl	E06JMLL
SKLM Basic Ed per Install SW S&S Reinstate 12 Mo	D0888LL

System x servers with SEDs that use SKLM for key management require a successful connection to the SKLM Basic Edition system to successfully boot and access the locally stored and encrypted data. Therefore, it is recommended that you implement redundant SKLM Basic Edition key managers.

When you set up the SKLM for System x SEDs – FoD option on your servers, you configure addresses for up to four SKLM Basic Edition key managers, one primary SKLM and up to three secondary systems. Although SKLM Basic Edition supports up to five secondary key managers, the SKLM for System x SEDs – FoD option and Integrated Management Module (IMM) configuration allows up to three secondary key managers only.

For more information about the downloadable installation images for IBM Security Key Lifecycle Manager, see the following IBM Passport Advantage website:

http://www.ibm.com/software/lotus/passportadvantage/pao_customer.html

In Passport Advantage, you can download or request the following media packs (or eAssemblies) of your entitled software:

- Installation images for AIX systems:

  http://www.ibm.com/support/knowledgecenter/api/content/SSWPVP_2.5.0/com.ibm.sklm.doc_2.5/cpt/cpt_ic_download_aix.html

- Installation images for Solaris systems:

  http://www.ibm.com/support/knowledgecenter/api/content/SSWPVP_2.5.0/com.ibm.sklm.doc_2.5/cpt/cpt_ic_download_solaris.html

- Installation images for Windows systems:

  http://www.ibm.com/support/knowledgecenter/api/content/SSWPVP_2.5.0/com.ibm.sklm.doc_2.5/cpt/cpt_ic_download_windows.html

In this publication, we performed the installation that was based on the SKLM 2.5 installation images for Windows, which are contained in the eAssembly package file name CIRX2ML.tar.

Related publications

The publications listed in this section are considered particularly suitable for a more detailed discussion of the topics covered in this book.

Lenovo Press publications

The following Lenovo Press publications provide additional information about the topic in this document.

- *Self-Encrypting Drives for System x*, TIPS0761

  http://lenovopress.com/tips0761
- *Using System x Features on Demand*, REDP-4895

  http://lenovopress.com/redp4895

IBM Redbooks publications

The following IBM Redbooks publications provide additional information about the topic in this document.

- *IBM Tivoli Key Lifecycle Manager for z/OS*, REDP-4472

  http://www.redbooks.ibm.com/abstracts/REDP4472.html
- *Using IBM Tivoli Key Lifecycle Manager: Business Benefits and Architecture Overview*, REDP-4529

  http://www.redbooks.ibm.com/abstracts/REDP4529.html
- *IBM DS8880 Data-at-rest Encryption*, REDP-4500

  http://www.redbooks.ibm.com/abstracts/REDP4500.html
- *Implementing the Storwize V7000 and the IBM System Storage SAN32B-E4 Encryption Switch*, SG24-7977

  http://www.redbooks.ibm.com/abstracts/SG247977.html
- *IBM System Storage Data Encryption*, SG24-7797

  http://www.redbooks.ibm.com/abstracts/SG247797.html

Other publications and online resources

These publications and websites are also relevant as further information sources:

- IBM Security Key Lifecycle Manager documentation on the IBM Knowledge Center:

  http://www.ibm.com/support/knowledgecenter/SSWPVP/welcome
- IBM Security Key Lifecycle Manager product page:

  http://www.ibm.com/software/products/en/key-lifecycle-manager/

© Copyright IBM Corp. 2015. All rights reserved.

Lenovo

Centrally Managing Access to Self-Encrypting Drives in Lenovo System x Servers

LENOVO PRESS

(0.5” spine)
0.475”<->0.873”
250 <-> 459 pages

Lenovo

Centrally Managing Access to Self-Encrypting Drives in Lenovo System x Servers

Understand self-encrypting drive technology and centralized key management systems

Centralized key management using IBM Security Key Lifecycle Manager

Manage and troubleshoot your SED-based server

Comprehensive guide for implementing a managed solution for SED drives

Data security is one of the paramount requirements for organizations of all sizes. Although many companies invested heavily in protection from network-based attacks and other threats, few effective safeguards are available to protect against potentially costly exposures of proprietary data that results from a hard disk drive being stolen, misplaced, retired, or redeployed.

Self-encrypting drives (SEDs) can satisfy this need by providing the ultimate in security for data-at-rest and can help reduce IT drive retirement costs in the data center. Self-encrypting drives are also an excellent choice if you must comply with government or industry regulations for data privacy and encryption.

To effectively manage a large deployment of SEDs in Lenovo System x servers, an organization must rely on a centralized key management solution. This Lenovo Press book explains the technology behind SEDs and demonstrates how to deploy a Lenovo key management solution that uses IBM Security Key Lifecycle Manager and properly setup your System x servers.

BUILDING TECHNICAL INFORMATION BASED ON PRACTICAL EXPERIENCE

At Lenovo Press, we bring together experts to produce technical publications around topics of importance to you, providing information and best practices for using Lenovo products and solutions to solve IT challenges.