Overview

Contents

Product overview

The Product overview topics describe the IBM Security Key Lifecycle Manager product (formerly called IBM Tivoli Key Lifecycle Manager) and its business and technology context.

They include information about:

- Product features and functions
- Technologies and architecture on which the product is based
- The user model and roles underlying the product features
- The graphical interfaces and tools that support various user roles

What's new in this release

IBM Security Key Lifecycle Manager provides a centralized and automated key management solution to protect keys that are used for encrypting data at rest. With the new set of features and enhancements, IBM Security Key Lifecycle Manager version 3.0.1 offers improved key management capabilities for the key management infrastructure to protect data.

IBM Security Key Lifecycle Manager master key management
You can now move an IBM Security Key Lifecycle Manager master key from the Java keystore to HSM (Hardware Security Module) and vice versa. The existing Master Key REST Service is enhanced to support this feature.

Multi-layer key wrapping
To enhance data security, IBM Security Key Lifecycle Manager now supports master key for device groups. The device group master key encrypts the managed objects, such as keys and certificates, of a device group, and the system-level master key encrypts this device group master key. Thus, providing a two-layered key wrapping. For more information, see Enable or Disable Master Key for Device Group REST Service.

Rapid key rotation
With the new support for device group master key, you can now refresh the master key that is used for encrypting managed objects, such as keys and certificates, of a device group. You can also refresh the master keys for all the device groups. For more information, see Refresh Master Key for Device Group REST Service.

Graphical user interface (GUI) enhancement
You can now use the IBM Security Key Lifecycle Manager graphical user interface to export and import symmetric and private keys. For more information, see Exporting and importing keys.

New operating system support
Ubuntu 16 on x86-64 support is now available for IBM Security Key Lifecycle Manager.

For more information about all the supported operating systems, see Operating system requirements.

License usage metrics

IBM Security Key Lifecycle Manager writes the license usage information to software identification tag files. Versions of the IBM License Metric Tool that support these tag files, can then generate license consumption reports.

IBM® License Metric Tool helps you maintain your license compliance. To generate IBM Security Key Lifecycle Manager license usage metrics, you must install and configure IBM License Metric Tool. IBM License Metric Tool is not bundled with IBM Security Key Lifecycle Manager.

IBM License Metric Tool discovers the software that is installed in your infrastructure, helps you to analyze the consumption data, and generates audit reports. For more information about using IBM License Metric Tool, see http://www-01.ibm.com/support/knowledgecenter/SS8JFY/lmt_welcome.html.

Each instance of a running IBM Security Key Lifecycle Manager runtime environment generates an IBM Software License Metric Tag file. The metrics that are monitored are `Authorized User`, `Install`, `Processor Value Unit (PVU)`, `Client Device`, and `Resource Value Unit (RVU)`.

`Authorized User`
: An Authorized User is a unique person who is given access to the program. The program may be installed on any number of computers or servers and each Authorized User can have simultaneous access to any number of instances of the program at one time. Licensee must obtain separate, dedicated entitlements for each Authorized User given access to the program in any manner directly or indirectly (for example, through a multiplexing program, device, or application server) through any means. An entitlement for an Authorized User is unique to that Authorized User and may not be shared, nor may it be reassigned other than for the permanent transfer of the Authorized User entitlement to another person.

`Install`
: Install is a unit of measure by which the program can be licensed. An install is an installed copy of the program on a physical or virtual disk made available to be executed on a computer. Licensee must obtain an entitlement for each installation of the program.

`Processor Value Unit (PVU)`
: The number of PVU entitlements required is based on the processor technology and the number of processors made available to the program.

`Client Device`
: A Client Device is a single-user computing device or special-purpose sensor or telemetry device that requests the execution of or receives for execution a set of commands, procedures, or applications from or provides data to another computer system that is typically referred to as a server or is otherwise managed by the server. Multiple Client Devices may share access to a common server.

  Licensee must obtain entitlements for every Client Device that runs, provides data to, uses services provided by, or otherwise accesses the program and for every other computer or server on which the program is installed.

`Resource Value Unit (RVU)`
: RVU Proofs of Entitlement (PoEs) are based on the number of units of a specific resource that are used or managed by the program. Licensee must

obtain sufficient entitlements for the number of RVUs required for licensee's environment for the specific resources. RVU entitlements are specific to the program and the type of resource and may not be exchanged, interchanged, or aggregated with RVU entitlements of another program or resource.

Location of software identification tag files

The software identification tag files have the extension `.swidtag`. The files are read periodically by the IBM License Metric Tool after it is configured to scan for these files. The tag file for IBM Security Key Lifecycle Manager is located at: `<SKLM_HOME>\swidtag\<product_id>.swidtag`

Supported languages

IBM Security Key Lifecycle Manager supports various languages. The graphical (web) user interface labels, messages, and values can be displayed in both English language and in languages other than English. However, IBM Security Key Lifecycle Manager supports only the systems that are localized to a single locale.

IBM Security Key Lifecycle Manager supports the following languages:

- English
- French
- German
- Italian
- Japanese
- Korean
- Simplified Chinese
- Spanish
- Traditional Chinese

Features overview

Use IBM Security Key Lifecycle Manager to manage the lifecycle of the keys and certificates of an enterprise. You can manage symmetric keys, secret keys, asymmetric key pairs, and certificates.

IBM Security Key Lifecycle Manager has the following key features:

- Role-based access control that provides permissions to do tasks such as create, modify, and delete for specific device groups. Most permissions are associated with specific device groups.
- Extension of support to devices by using industry-standard Key Management Interoperability Protocol (KMIP) for encryption of stored data and the corresponding cryptographic key management.
- Support for encryption-enabled 3592 tape drives, LTO tape drives, DS5000 storage servers, DS8000 Turbo drives, and other devices.
- A graphical user interface, command-line interface, and REST interface to manage keys, certificates, and devices.
- Encrypted keys to one or more devices to which IBM Security Key Lifecycle Manager server is connected.
- Storage of key materials for the self-signed certificates that you generate, private key, and the key metadata in a database.

- Cross-platform backup and restore to protect IBM Security Key Lifecycle Manager data, such as the configuration files and current database information.
- Cross-platform backup utility to run backup operation on IBM Tivoli Key Lifecycle Manager 1.0, 2.0, 2.0.1, IBM Security Key Lifecycle Manager 2.5, 2.6, 2.7, and IBM Encryption Key Manager, 2.1. You can restore these backup files on current version of IBM Security Key Lifecycle Manager across operating systems.
- Migration of IBM Security Key Lifecycle Manager 2.5, 2.6, and IBM Encryption Key Manager 2.1 during installation.
- Audit records based on selected events that occur as a result of successful operations, unsuccessful operations, or both. Installing or starting IBM Security Key Lifecycle Manager writes the build level to the audit log.
- Support for configuring Hardware Security Module (HSM) to store the master key, which protects the key materials that are stored in the database.
- A set of operations to automatically replicate current active files and data across operating systems. This replication enables cloning of IBM Security Key Lifecycle Manager environments on multiple servers in a manner that is independent of operating systems and directory structure of the server.
- Support for configuring LDAP (Lightweight Directory Access Protocol) server for user authentication. You can configure IBM Security Key Lifecycle Manager users in any of the LDAP repositories, such as IBM Security Directory Server or Microsoft Active Directory.
- Server Configuration Wizard to configure IBM Security Key Lifecycle Manager for SSL/TLS handshake. The SSL handshake enables the server and client devices to establish the connection for secure communication.
- HSM-based encryption for creating secure backups and replication when IBM Security Key Lifecycle Manager is configured with HSM to store the master key.
- Device group export and import operations to move device group data across multiple instances of IBM Security Key Lifecycle Manager.
- IBM Security Key Lifecycle Manager writes the license usage information to software identification tag files. IBM License Metric Tool helps you maintain your license compliance.
- Multi-Master configuration to achieve continuous availability of synchronized data across multiple instances of IBM Security Key Lifecycle Manager.

Key serving

IBM Security Key Lifecycle Manager enables definition and serving of keys. IBM Security Key Lifecycle Manager also enables definition of keys or groups of keys that can be associated with a device. Different devices require different key types. After you configure devices, IBM Security Key Lifecycle Manager deploys keys to the devices that request them.

Key group

An IBM Security Key Lifecycle Manager key group contains keys. A key can be a member of only one key group.

Deleting a key group *also deletes all the keys* in the key group.

Key metadata

Metadata for an IBM Security Key Lifecycle Manager key includes information such as a key alias, algorithm, and activation date.

Metadata can also include a key description, expiration date, retirement date, destroy date, compromise date, key usage, backup time, and state, such as active. IBM Security Key Lifecycle Manager stores the metadata for a key in the IBM Security Key Lifecycle Manager database.

Key and certificate states

Cryptographic objects, in their lifetime, transition through several states that are a function of how long the keys or certificates are in existence and whether data is protected. Other factors also affect the state of a cryptographic object, such as whether the key or certificate is compromised.

IBM Security Key Lifecycle Manager maintains these cryptographic object states.

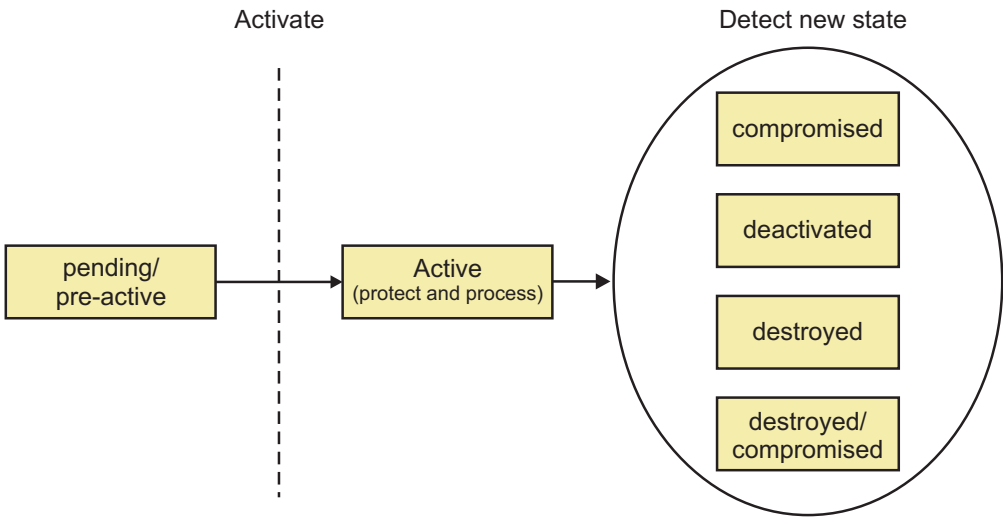

Figure 1. Cryptographic object states

The state of a key or certificate defines the allowed usage:

pending
A certificate request entry is pending the return of a certificate that is approved and certified by a certificate authority.

pre-active
Object exists but is not yet usable for any cryptographic purpose, such as migrated certificates with a future use time stamp.

active
Object is in operational use for protecting and processing data that might use **Process Start Date** and **Protect Stop Date** attributes. For example, protecting includes encryption and signature issue. Processing includes decryption and signature verification.

compromised
The security of the object is suspect for some reason. A compromised object never returns to an `uncompromised` state, and cannot be used to protect data. Use the object only to process cryptographically protected information in a client that is trusted to handle compromised cryptographic objects.

IBM Security Key Lifecycle Manager retains the state of the object immediately before it was compromised. To process data that was previously protected, the compromised object might continue to be used.

deactivated
Object is not to be used to apply cryptographic protection such as encryption or signing. However, if extraordinary circumstances occur, the object can be used with special permission to process cryptographically protected information. For example, processing includes decryption or verification.

destroyed
Object is no longer usable for any purpose. This status causes the object to be removed from the product.

destroyed-compromised
Object is no longer usable for any purpose. This status causes the object to be removed from the product.

An object that is no longer active might change states from:

- Deactivated to destroyed.
- Deactivated to compromised.
- Compromised to destroyed-compromised.
- Destroyed to destroyed-compromised.

IBM Security Key Lifecycle Manager keystore

IBM Security Key Lifecycle Manager can store symmetric keys, public keys, private keys, their associated certificate chains, and trusted certificates.

When IBM Security Key Lifecycle Manager generates a new key, the key and the metadata for the key is stored in a key table in the IBM Security Key Lifecycle Manager database. The key material is protected by using a master key. When you create a certificate request, IBM Security Key Lifecycle Manager creates a key entry that is in a pending state.

Encryption-enabled 3592 tape drives and LTO tape drives

IBM Security Key Lifecycle Manager supports encryption-enabled 3592 tape drives and LTO tape drives. Drives without encryption enablement are not supported. Encryption is run at full line speed in the tape drive after compression.

For information about the devices that IBM Security Key Lifecycle Manager supports, see the **Details** -> **Technical details** page at http://www-03.ibm.com/software/products/en/key-lifecycle-manager.

Enterprise Storage: DS8000® Storage Controller (2107, 242x)

IBM Security Key Lifecycle Manager supports the DS8000 Storage Controller (IBM System Storage DS8000 Turbo drive).

This support requires the appropriate microcode bundle version on the DS8000 Storage Controller, Licensed Internal Code level 64.20.xxx.0, or higher.

IBM System Storage®: DS5000 Storage Controller (1818-51A, 1818-53A, and 1814-20A)

IBM Security Key Lifecycle Manager supports the DS5000 storage server (IBM System Storage DS5000).

This support is for DS5000 series storage systems (DS5100, DS5300, and DS5020) with Self-Encrypting Fibre Channel Drives (FDE/SED drives). The optional

Full-Disk Encryption Premium Feature must also be purchased and enabled in the storage subsystem. The systems include the following storage controllers:

- 1818-51A, 1818-53A, FC 7358 DS5000 Disk Encryption Activation
- 1814-20A, FC 7410 DS5020 Disk Encryption Activation

See *IBM DS Storage Manager 10.70 Installation and Host Support Guide* for more information in setting the DS5000 storage subsystem to support IBM Security Key Lifecycle Manager.

Backup and restore

IBM Security Key Lifecycle Manager provides cross-platform backup and restore functions to protect IBM Security Key Lifecycle Manager critical information. You can create cross-platform compatible backups and restore the same across operating systems. For example, backups on a Linux system can be restored on a Windows system, and vice versa.

Use IBM Security Key Lifecycle Manager to protect data with these functions:

Backup

A backup is a secondary copy of active production information that is used when a recovery copy is needed to get a user back to work. When a disaster occurs, a backup can get the business up and running again. Since backups are focused on constantly changing business information, they are short-term and often overwritten. You might maintain copies of backup files on a secure computer at a geographically separate location.

Depending on your site requirements, you can maintain a replica computer that provides another IBM Security Key Lifecycle Manager server, including a backup of critical data. The replica computer enables quick recovery at times when the primary IBM Security Key Lifecycle Manager server is not available.

Restore

A restore returns the IBM Security Key Lifecycle Manager server to a known state, by using backed-up production data, such as the IBM Security Key Lifecycle Manager keystore and other critical information. You must restore the entire backup data. You cannot do a partial restore of data.

Audit

IBM Security Key Lifecycle Manager provides audit records on distributed systems in Common Base Event (CBE) format. The audit records are stored in a flat file in the audit log. You can also configure IBM Security Key Lifecycle Manager to generate audit records in syslog format and send them to a syslog server.

Automated clone replication

IBM Security Key Lifecycle Manager automated clone replication uses a program to clone a master IBM Security Key Lifecycle Manager server with up to 20 copies.

You can configure the program to replicate keys and also other configuration information, such as when new keys that are rolled over. This program automates the replication of everything that is needed. Automated clone replication ensures continuous key and certificate availability to the encrypting devices.

IBM Security Key Lifecycle Manager provides a set of operations to replicate current active files and data across systems. This replication enables cloning of IBM Security Key Lifecycle Manager environments to multiple servers in a manner that is independent of operating systems and directory structure of the server. For example, you can replicate data from a master server on a Windows system to a clone server on a Linux system. When the automated replication program is run, the following IBM Security Key Lifecycle Manager data is replicated:

- Data in the IBM Security Key Lifecycle Manager database tables.
- All keys materials in the IBM Security Key Lifecycle Manager database.
- IBM Security Key Lifecycle Manager configuration files (except the replication configuration file).

Note: This data is taken as part of an IBM Security Key Lifecycle Manager backup. During a replication, the replication configuration file is not backed-up and passed to the clone.

IBM Security Key Lifecycle Manager replication configuration parameters are defined in the `ReplicationSKLMConfig.properties` configuration file. You can use the graphical user interface, command-line interface, or REST interface to change properties of the replication configuration file. You must configure the replication configuration file on all systems that are part of the replication process. Each instance of IBM Security Key Lifecycle Manager is defined as either the *master*, the system that is to be cloned, or a *clone*, the system that the data is being replicated on.

Master key in Hardware Security Module

You can configure IBM Security Key Lifecycle Manager with Hardware Security Module (HSM) to store the master key, which protects key materials that are stored in the database.

HSM adds extra protection to the storage and use of the master key. IBM Security Key Lifecycle Manager supports HSM-based encryption for creating secure backups and replication when HSM is configured to store the master key.

For HSM configuration information, see Hardware Security Module usage in IBM Security Key Lifecycle Manager.

LDAP integration with IBM Security Key Lifecycle Manager server

LDAP (Lightweight Directory Access Protocol) supports the management of user IDs and passwords at an enterprise level instead of management of this data on individual systems. You can integrate IBM Security Key Lifecycle Manager with LDAP user repositories.

You can configure IBM Security Key Lifecycle Manager users in any of the LDAP repositories, such as IBM Security Directory Server or Microsoft Active Directory to access IBM Security Key Lifecycle Manager server and call server APIs and CLIs. You must add and configure LDAP user repository to the federated repository of WebSphere® Application Server. For more information in LDAP configuration, see LDAP configuration

Server Configuration Wizard

You can use the Server Configuration Wizard to configure server and the client device for SSL/TLS handshake. The SSL/TLS handshake enables IBM Security Key Lifecycle Manager server and client devices to establish the connection for secure communication.

Immediately after you install IBM Security Key Lifecycle Manager, the only available option is to configure IBM Security Key Lifecycle Manager for SSL/TLS handshake by using the Server Configuration Wizard. To open, click the **Review the configuration parameters and/or create an SSL server certificate** link. The wizard offers a guided approach to set up SSL handshake process. For more information about SSL/TLS handshake, see Scenario: Setup for SSL handshake between IBM Security Key Lifecycle Manager server and client device.

Device group data export and import

IBM Security Key Lifecycle Manager exports and imports device group data while maintaining data integrity.

You can export device group data from an IBM Security Key Lifecycle Manager instance. You can then import a previously exported device group data into a different IBM Security Key Lifecycle Manager instance that has the same version as of the source IBM Security Key Lifecycle Manager instance.

The device group export and import tasks are useful to easily move device group data across IBM Security Key Lifecycle Manager instances on same or different operating systems according to your business needs.

IBM Security Key Lifecycle Manager Multi-Master setup

You can set up IBM Security Key Lifecycle Manager instances with Multi-Master configuration to achieve continuous availability of data across multiple IBM Security Key Lifecycle Manager deployment environments.

A Multi-Master cluster contains multiple IBM Security Key Lifecycle Manager master servers. All master servers in the cluster point to a single data source that is configured for DB2 high availability disaster recovery (HADR). Multi-Master configurations have the following advantages:

- Provides data redundancy that is achieved by using the DB2 HADR feature.
- Provides real-time availability of data that is created by one IBM Security Key Lifecycle Manager master server to other master servers in the cluster.
- Masters can be located in several physical sites, that is, distributed across the network.

Technical overview

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

IBM Security Key Lifecycle Manager 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.

IBM Security Key Lifecycle Manager provides the following major features:

- Managing symmetric keys, asymmetric key pairs, and X.509 V3 certificates.
- Managing the creation and lifecycle of keys, which contain metadata on their intended usage.
- For disaster recovery, providing protected backup of critical data. For example, on distributed systems, backup includes cryptographic key data (actual keys and certificates that are managed), metadata about the keys, and configuration files.
- For continuous key and certificate availability to the encrypting devices, providing automated clone replication program to replicate keys and also other configuration information, such as when new keys that are rolled over.
- File-based audit logs that vary, depending on the operating system. On distributed systems, audit logs contain data in a flat file that is based on the Common Base Event (CBE) security event specification. You can also configure IBM Security Key Lifecycle Manager to generate audit records in syslog format and send them to a syslog server.

Keys overview

An encryption key is typically a random string of bits 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 constructed this way, the harder it is to break the encryption code.

Cryptographic algorithm and key length

IBM Security Key Lifecycle Manager uses two types of algorithms, such as symmetric algorithms and asymmetric algorithms for data encryption.

Symmetric, or secret key encryption, uses a single key for both 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 allowed by AES. By default, IBM Security Key Lifecycle Manager generates 256-bit AES keys.

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

IBM Security Key Lifecycle Manager uses both 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.

Supported key sizes and import and export restrictions

IBM Security Key Lifecycle Manager can serve either 2048 or 1024-bit keys to devices. Older keys that were generated as 1024-bit keys can continue to be used.

Import PKCS#12 file	Export PKCS#12 file	Key Generation Size in Bits
Yes	Yes	2048

Security standard compliance for IBM Security Key Lifecycle Manager

You configure IBM Security Key Lifecycle Manager to work with various security standards to meet the specified security requirements for encryption.

The standards include Federal Information Processing Standards (FIPS) publication 140-2, NIST Special Publication (SP) 800-131A, and NSA Suite B.

Federal Information Processing Standard compliance:

The federal government requires all its cryptographic providers to be FIPS 140 certified. This standard is also adopted in a growing private sector community. The certification of cryptographic capabilities by a third party in accordance with government standards are increased value in this security-conscious world.

If you export private keys to a PKCS#12 file, ensure that the file with the key is wrapped by using a FIPS-approved method before the file leaves the computer.

IBM Security Key Lifecycle Manager itself does not provide cryptographic capabilities and therefore does not require or obtain, FIPS 140-2 certification. However, IBM Security Key Lifecycle Manager takes advantage of the cryptographic capabilities of the IBM JVM in the IBM Java Cryptographic Extension component. The capabilities allow the selection and use of the IBMJCEFIPS cryptographic provider, which has a FIPS 140-2 level 1 certification.

For more information about the IBMJCEFIPS provider and its selection and use, see the IBM Security information for Java documentation (https://www.ibm.com/support/knowledgecenter/SSYKE2_8.0.0/com.ibm.java.security.component.80.doc/security-component/fips.html).

For the procedure on how to configure FIPS, see Configuring compliance for FIPS in IBM Security Key Lifecycle Manager.

See the documentation from specific hardware and software cryptographic providers for information about whether their products are FIPS 140-2 certified.

Note: Setting the **fips** configuration property to on causes IBM Security Key Lifecycle Manager to use the IBMJCEFIPS provider for all cryptographic functions.

NSA Suite B compliance in IBM Security Key Lifecycle Manager:

You can configure IBM Security Key Lifecycle Manager to comply with standards that are specified by the US National Security Agency (NSA) to define security requirements for encryption.

NSA Suite B requires TLS 1.2 protocol and cipher suites that are configured with a minimum level of security of 128 bits by using ECDSA-256 and ECDSA-384 for client or server authentication. To support the Suite B profile, the following Java system property is provided:

```
com.ibm.jsse2.suiteB=128|192|false
```

When you set the **com.ibm.jsse2.suiteB** system property, IBMJSSE2 ensures adherence to the specified security level. IBMJSSE2 validates that the protocol, keys, and certificates comply with the requested profile. For more information, see

https://www-01.ibm.com/support/knowledgecenter/SSYKE2_8.0.0/com.ibm.java.security.component.80.doc/security-component/jsse2Docs/suiteb.html.

To enable Suite B compliance in IBM Security Key Lifecycle Manager, you must configure the `SKLMConfig.properties` properties file with the following option:

```
suiteB=128|192
```

When you configure **`suiteB`** with the value `128` or `192`, the following properties are added to the properties file or the values are updated if the properties are existed in the file.

```
TransportListener.ssl.protocols=SSL_TLSv2
requireSHA2Signatures=true
autoScaleSignatureHash=true
useThisECKeySize=256(if suiteB is 128)|384(if suiteB is 192)
```

For the procedure on how to configure IBM Security Key Lifecycle Manager for Suite B compliance, see Configuring IBM Security Key Lifecycle Manager for Suite B compliance.

NIST SP 800-131A compliance in IBM Security Key Lifecycle Manager:

You can configure IBM Security Key Lifecycle Manager to communicate over secure sockets in compliance with the National Institute of Standards and Technology (NIST) Special Publication (SP) 800-131A standard in strict mode.

NIST SP 800-131 is a US government computer security standard that is used to accredit cryptographic modules. For more information about the SP 800 series of computer security publications, see the NIST website.

For NIST SP 800-131A compliance configuration information, see Configuring compliance for NIST SP 800-131A in IBM Security Key Lifecycle Manager.

Key management by using the Key Management Interoperability Protocol

The IBM Security Key Lifecycle Manager server supports Key Management Interoperability Protocol (KMIP) communication with clients for key management operations on cryptographic material. The material includes symmetric and asymmetric keys, certificates, and templates that are used to create and control their use.

The Key Management Interoperability Protocol is part of an Organization for the Advancement of Structured Information Standards (OASIS) standardization project for encryption of stored data and cryptographic key management.

You can use the IBM Security Key Lifecycle Manager graphical user interface to manage and control cryptographic materials (objects) that are supported by the server. For detailed information about KMIP profiles that are supported by IBM Security Key Lifecycle Manager, KMIP objects, and other KMIP-related details, see the Key Management Interoperability Protocol (KMIP) section.

Key serving management

The IBM Security Key Lifecycle Manager solution assists IBM encryption-enabled devices in generating, protecting, storing, and maintaining encryption keys. You can use keys to encrypt and decrypt information that is written to and read from devices.

IBM Security Key Lifecycle Manager acts as a background process that is waiting for key generation or key retrieval requests sent to it through a TCP/IP communication path between itself and the tape library, tape controller, tape subsystem, device driver, or tape drive. When a drive writes encrypted data, it first requests an encryption key from IBM Security Key Lifecycle Manager.

AES keys and the 3592 tape drive:

When a 3592 tape drive writes encrypted data, it first requests an encryption key from IBM Security Key Lifecycle Manager.

On receipt of the request, IBM Security Key Lifecycle Manager generates an Advanced Encryption Standard (AES) key. The key is served to the tape drive in two protected forms:

- Encrypted or wrapped, by using Rivest-Shamir-Adleman (RSA) key pairs. 3592 tape drives write this copy of the key to the cartridge memory and extra places on the tape media in the cartridge for redundancy.
- Separately wrapped for secure transfer to the tape drive where it is unwrapped upon arrival. The key inside is used to encrypt the data that is written to the tape.

When an encrypted tape cartridge is read by a 3592 tape drive, the protected AES key on the tape is sent to IBM Security Key Lifecycle Manager where the wrapped AES key is unwrapped. The AES key is then wrapped with a different key for secure transfer back to the tape drive. The key is unwrapped and used to decrypt the data that is stored on the tape. IBM Security Key Lifecycle Manager also allows protected AES keys to be rewrapped, or rekeyed, by using different RSA keys from the original ones that are used when the tape was written. Rekeying is useful when an unexpected need arises to export volumes to business partners whose public keys were not included. It eliminates rewriting the entire tape and enables the data key of a tape cartridge to be re-encrypted with the public key of a business partner.

Asymmetric keys and the 3592 tape drive:

In addition to 256-bit AES symmetric data keys, IBM Security Key Lifecycle Manager also uses public/private (asymmetric) key cryptography to protect the symmetric data encryption keys. These keys are generated and retrieved as they pass between IBM Security Key Lifecycle Manager and 3592 tape drives.

Public/private key cryptography is also used to verify the identity of the tape drives to which IBM Security Key Lifecycle Manager serves keys.

When a 3592 tape drive requests a key, IBM Security Key Lifecycle Manager generates a random symmetric data encryption key. Use public/private key cryptography to wrap the data encryption key by using a key encryption key, which is the public key of an asymmetric key pair.

The wrapped data key, along with key label information about what private key is required to unwrap the symmetric key, forms a digital envelope, called an externally encrypted data key structure. The structure is stored in the tape header area of any tape cartridge that holds data encrypted by using this method. The key that you use to decrypt the data is stored with the data on the tape itself, protected by asymmetric, public/private key wrapping. The public key that you use to wrap the data key is obtained from one of the following two sources:

- A public key (part of an internally generated public/private key pair) stored in the keystore.
- A certificate (from a business partner, for example) stored in the keystore.

The certificates and keys that are stored in the keystore are the point of control that permits a tape drive or library to decrypt the data on the tape. Without the information in the keystore, the tape cannot be read. It is important to prevent unauthorized users from obtaining the private keys from the keystore. You must always keep the keystore available to you to read the tapes.

The data encryption key is stored *only* on the tape, in a wrapped, protected form. When an encrypted tape is to be read by a 3592 tape drive, the tape drive sends the externally encrypted data key to IBM Security Key Lifecycle Manager. IBM Security Key Lifecycle Manager determines from the alias or key label which private key encryption key from its keystore to use to unwrap the externally encrypted data key and recover the data encryption key.

After the data encryption key is recovered, it is then wrapped with a different key, which the tape drive can decrypt. The key is then sent back to the tape drive, enabling the tape drive to decrypt the data.

IBM Security Key Lifecycle Manager uses aliases, also known as key labels, to identify the public/private keys that are used to wrap the externally encrypted data key when you encrypt with 3592 tape drives. You can define specific aliases for each tape device by using the IBM Security Key Lifecycle Manager graphical user interface or command-line interface.

IBM Security Key Lifecycle Manager allows the definition of at least two aliases (certificates or key labels) for each encrypting tape drive. The aliases allow access to the encrypted data at another location within your organization or outside it. The private key for one of these aliases must be known. If you do not want to specify two different key labels or aliases, you can define both aliases with the same value.

AES keys and the LTO tape drive:

When an LTO tape drive writes encrypted data, it first requests an encryption key from IBM Security Key Lifecycle Manager.

Upon receipt of the request, IBM Security Key Lifecycle Manager obtains an existing AES key from a keystore. The key is then wrapped for secure transfer to the tape drive. The key is then unwrapped and used to encrypt the data that is written to the tape.

When an encrypted tape is read by an LTO tape drive, IBM Security Key Lifecycle Manager obtains the required key from the keystore. The key is based on the information in the Key ID on the tape, and serves it to the tape drive wrapped for secure transfer.

Symmetric keys and the LTO tape drive:

IBM Security Key Lifecycle Manager uses only symmetric data keys for encryption tasks on the LTO tape drive.

When an LTO tape drive requests a key, IBM Security Key Lifecycle Manager uses the alias that is specified for the tape drive. If no alias was specified for the tape drive, IBM Security Key Lifecycle Manager uses an alias from a key group, key alias list, or range of key aliases.

The keys from the key group are used in a round robin fashion to help balance the use of keys more evenly.

The selected alias is associated with a symmetric data key that was preinstalled in the keystore. IBM Security Key Lifecycle Manager sends the data key to the LTO tape drive to encrypt the data. The selected alias is also converted to an entity called data key identifier, which is written to tape with the encrypted data. IBM Security Key Lifecycle Manager can use the data key identifier to identify the correct data key that is required to decrypt the data when the LTO tape is read.

AES keys and the DS8000 Turbo drive:

When the DS8000 Turbo drive starts, the device requests an unlock key from IBM Security Key Lifecycle Manager.

If the DS8000 Turbo drive requests a new key for its unlock key, IBM Security Key Lifecycle Manager generates an Advanced Encryption Standard (AES) key. The key is then served to the drive in the following two protected forms:

- Encrypted (wrapped) by using Rivest-Shamir-Adleman (RSA) key pairs. The DS8000 Turbo drive stores this copy of the key on the array in an unencrypted partition.
- Separately wrapped for secure transfer to the drive where it is unwrapped upon arrival and the key inside is used to unlock the array.

If the DS8000 Turbo drive requests an existing unlock key, the protected AES key on the array is sent to IBM Security Key Lifecycle Manager where the wrapped AES key is unwrapped. The AES key is then wrapped with a different key for secure transfer back to the DS8000 Turbo drive. The key is unwrapped and used to unlock the array.

Asymmetric keys and the DS8000 Turbo drive:

IBM Security Key Lifecycle Manager also uses public/private (asymmetric) key cryptography to protect 256-bit AES symmetric data encryption keys as they pass between IBM Security Key Lifecycle Manager and the DS8000 Turbo drive.

Public/private key cryptography is also used to verify the identity of the tape drives to which IBM Security Key Lifecycle Manager serves keys. When a DS8000 Turbo drive requests a new key, IBM Security Key Lifecycle Manager generates a random symmetric data encryption key. Use public/private key cryptography to wrap the data encryption key by using a key encryption key, which is the public key of an asymmetric key pair.

The wrapped data key, along with key label information about that private key that is required to unwrap the symmetric key, forms a digital envelope, called an externally encrypted data key structure. The structure is stored in the tape header area of any tape cartridge that holds data encrypted using this method. The key that you use to decrypt the data is stored with the data on the tape itself, protected by asymmetric, public/private key wrapping. The public key that is used to wrap the data key is obtained from one of the following two sources:

- A certificate (from a business partner, for example) stored in the keystore.
- A public key (part of an internally generated public/private key pair) stored in the keystore.

The certificates and keys that are stored in the keystore are the point of control that allows a DS8000 Turbo drive to be unlocked. Without the information in the keystore, the DS8000 Turbo drive cannot be unlocked.

You must prevent unauthorized users from obtaining the private keys from the keystore, and to always keep the keystore available to you to unlock the arrays. The data encryption key is stored only on the DS8000 Turbo drive in a wrapped, protected form.

To unlock a DS8000 Turbo drive, the DS8000 Turbo drive sends the externally encrypted data key to IBM Security Key Lifecycle Manager. IBM Security Key Lifecycle Manager determines from the alias or key label which private key encryption key from its keystore to use to unwrap the externally encrypted data key and recover the data encryption key. After the data encryption key is recovered, it is then wrapped with a different key, which the tape drive can decrypt. The key is sent back to the tape drive to enable the tape drive for data decryption.

IBM Security Key Lifecycle Manager uses aliases, also known as key labels, to identify the public/private keys that you use to wrap the unlocking key. You can define specific aliases for each device. IBM Security Key Lifecycle Manager allows the definition of up to two aliases (certificates or key labels) for each DS8000 Turbo drive to prevent deadlock conditions. IBM Security Key Lifecycle Manager must be on the same system as the DS8000 Turbo drive. The DS8000 Turbo drive must unlock before the IBM Security Key Lifecycle Manager can come up. The private key for one of these aliases must be known. If you do not want to specify two different key labels or aliases, you can define both aliases with the same value.

AES keys and the DS5000 storage server:

When a DS5000 storage server starts, the device requests a key from IBM Security Key Lifecycle Manager to unlock disk drives.

In response, IBM Security Key Lifecycle Manager obtains an existing AES key from the keystore. IBM Security Key Lifecycle Manager wraps the AES key for secure transfer to the DS5000 storage server, which unwraps and uses the key to unlock disk drives.

Symmetric keys and the DS5000 storage server:

IBM Security Key Lifecycle Manager uses only symmetric data keys as the unlock key for a DS5000 storage server.

When a DS5000 storage server requests a key, IBM Security Key Lifecycle Manager uses the alias that the request specifies to get the key. If the DS5000 storage server request does not specify an alias, IBM Security Key Lifecycle Manager obtains an alias from the list of keys that are associated with the requesting DS5000 storage server. Keys from the list are served in round robin fashion to balance the use of keys evenly.

The selected alias is associated with a symmetric data key that was preinstalled in the keystore. IBM Security Key Lifecycle Manager sends the symmetric data key to

the device to unlock the disk drives of this array. The selected alias is also converted to an entity that is termed a data key identifier, which the DS5000 storage server stores. IBM Security Key Lifecycle Manager can use the data key identifier to identify the correct data key when needed.

Overview of device group export and import

When multiple IBM Security Key Lifecycle Manager instances are maintained across operating systems, you might need to move device group data from one instance to another according to your business requirements. You can use the device group export and import operations to export and import data across IBM Security Key Lifecycle Manager instances with the same version as of the source IBM Security Key Lifecycle Manager instance, on the same or different operating systems, while maintaining data integrity. The exported device group data is encrypted and protected through a password.

Device groups for all the default device types are created during installation of IBM Security Key Lifecycle Manager. When you add a device type by using the graphical user interface, command-line interface, or REST interface, the corresponding device group is created in the database. Name of the device group is same as the device type that you created.

Device group export

You can export device group by using the IBM Security Key Lifecycle Manager graphical user interface or REST interface. The export device group operation creates a compressed archive with the extension `.exp` in a location that you specify. Except for the `manifest` and `summary.json` files, all the following files of the archive are encrypted by the password that was specified during device group export operation.

- Manifest file, which lists all the device group data files in the archive
- `summary.json`, which contains summary information for the device group
- Files specific to devices
- Files specific to keys
- Files specific to certificates

Device group import

You can import device group data to an IBM Security Key Lifecycle Manager instance from an encrypted archive that was exported from another IBM Security Key Lifecycle Manager instance. During device group import operation, you must specify the password that was used for device group export operation to import and decrypt data. Use the IBM Security Key Lifecycle Manager graphical user interface or REST interface to import device group.

Note: You must restart the server after you run the device group import operation.

Device group import conflicts

At times, the device group data that is imported might conflict with an existing data in the database. For example, a key in the imported device group might be a duplicate key of a device group in the current instance of IBM Security Key Lifecycle Manager where the data is being imported. When conflicts occur, they must be resolved before the import process can continue.

The device group import operation includes the following tasks:

- Saving export file in the target IBM Security Key Lifecycle Manager server where the device group is being imported. You must have the same encryption password that was used for creating the export file to extract and decrypt data
- Evaluating duplicates between the data that is imported and the data in the target server
- Resolving the conflicts
- Importing device group data to the target server

You can view the list of conflicting items, if any, during device group import operation. Then, you can export the conflict information to a file in comma-separated values (CSV) format for further analysis.

For information about how to export the device group, see Export and import of device groups.

Main components

IBM Security Key Lifecycle Manager key management solution includes the IBM Security Key Lifecycle Manager server, WebSphere Application Server, and DB2®.

When you install IBM Security Key Lifecycle Manager, its components WebSphere Application Server and Db2 are also installed.

Runtime environment

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.

Database server

IBM Security Key Lifecycle Manager stores key materials in a Db2 relational database. Use IBM Security Key Lifecycle Manager to manage the Db2.

Deployment on Windows and systems such as Linux or AIX

On Windows systems and other systems such as Linux or AIX, the IBM Security Key Lifecycle Manager installation program deploys the IBM Security Key Lifecycle Manager server and required middleware components on the same computer. You must ensure that the computer has the required memory, speed, and available disk space to meet the workload.

IBM Security Key Lifecycle Manager can run on a member server in a domain controller environment, but is not supported on a primary or backup domain controller.

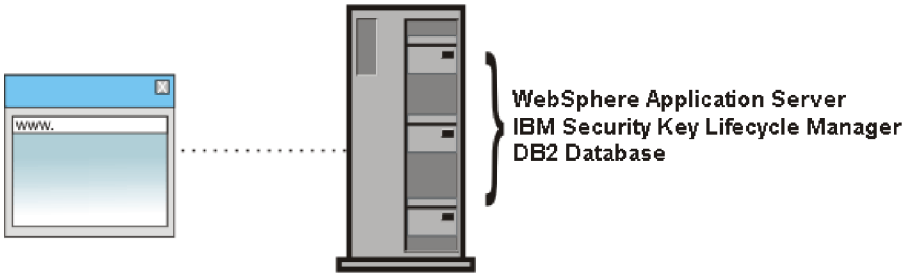

Figure 2. Main components on Windows systems and systems such as Linux or AIX

Deployment of a primary and replica server

To ensure availability, deploy both a primary IBM Security Key Lifecycle Manager server and, on a separate system, a replica of the primary IBM Security Key Lifecycle Manager server.

On Windows, Linux, or AIX systems, both computers must have the required memory, speed, and available disk space to meet the workload. The operating system and middleware components must be the same on both computers. The installation paths must also be the same.

Figure 3. Primary and replica IBM Security Key Lifecycle Manager server

Replica system requirements

A replica system must have an identical operating system, database, and IBM Security Key Lifecycle Manager application, including critical data from a current IBM Security Key Lifecycle Manager server backup file. The installation paths must also be the same.

Ensure that the same version and fix levels exist on both systems for these requirements:

- Operating system and fixes or patches.
- Db2 and required free disk space. The database must exist on the same system on which the IBM Security Key Lifecycle Manager server runs.
- IBM Security Key Lifecycle Manager server.

  You must manually copy the current IBM Security Key Lifecycle Manager server backup file to the replica system. IBM Security Key Lifecycle Manager does not automatically synchronize data between two IBM Security Key Lifecycle Manager servers.

Backup and restore overview

Back up and restore tasks provide protection for critical data, and require consideration of your site practices to ensure server availability and runtime capabilities.

IBM Security Key Lifecycle Manager creates backup files in a manner that is independent of operating systems and directory structure of the server. The backup files contain critical data for the current state of the IBM Security Key Lifecycle Manager server. Your site practices must consider how to ensure that key serving is available.

You can use the cross-platform backup utility to run backup operation on earlier versions of IBM Security Key Lifecycle Manager and IBM Tivoli Key Lifecycle

Manager to back up critical data. You can restore these backup files on current version of IBM Security Key Lifecycle Manager to an operating system that is different from the one it was backed up from.

Note: In IBM Security Key Lifecycle Manager, Version 3.0, the Solaris operating system is not supported. If you are using IBM Security Key Lifecycle Manager on Solaris systems, use the cross-platform backup utility to back up the data. You can then run the restore operation to restore data on a IBM Security Key Lifecycle Manager, Version 3.0 system that is deployed on any of the supported operating systems, such as Windows, Linux, or AIX.

The IBM Security Key Lifecycle Manager backup and restore operations support the use of AES 256-bit key length for data encryption/decryption to conform to the PCI DSS (Payment Card Industry Data Security Standard) standards for increased data security.

Encryption methods to back up IBM Security Key Lifecycle Manager data

IBM Security Key Lifecycle Manager supports the following encryption methods for backups:

Password-based encryption
: During the backup process, a password is specified to encrypt the backup key, and you must specify the same encryption password to decrypt and restore the backup files.

HSM-based encryption
: You can configure IBM Security Key Lifecycle Manager to use Hardware Security Module (HSM) for storing the master encryption key. During the backup process, the backup key is encrypted by the master key, which is stored in HSM. During the restore process, the master key in HSM decrypts the backup key. Then, the backup key is used to restore backup contents.

High performance backup and restore

High performance backup and restore provide backup and restoration of large amounts of encryption keys. You can configure IBM Security Key Lifecycle Manager for high performance backup and restore operations by setting the following parameter in the `SKLMConfig.properties` configuration file.

```
enableHighScaleBackup=true
```

Note:

- You cannot create a cross-platform compatible backup file if IBM Security Key Lifecycle Manager is configured for high performance backup and restore activities. You can use the backup file to restore data in an identical operating environment. The operating system, middleware components, and directory structures must be identical on both systems.
- The `db2restore.log` file is created during restore process only when IBM Security Key Lifecycle Manager is configured for high performance backup and restore operations.

For information about how to back up large amount of data, see Backing up large amount of data.

Password-based encryption for backups

When you use the password-based encryption method for backups, you must specify an encryption password during the backup process. The same password must be specified to restore backups.

When you run the IBM Security Key Lifecycle Manager backup operation, a backup archive is created. The backup key in the archive encrypts backup contents. During the restore process, backup contents are restored by specifying the password that was used when you created the backups.

The backup archive contains the following files:

- Manifest file, which lists all the IBM Security Key Lifecycle Manager data files in the archive.
- Backup keystore where the backup key is stored
- Truststore and keystore with the master key
- IBM Security Key Lifecycle Manager configuration files
- IBM Security Key Lifecycle Manager data dumps

For information about how to back up IBM Security Key Lifecycle Manager data by using password-based encryption, see Backing up data with password-based encryption.

HSM-based encryption for backups

You can configure IBM Security Key Lifecycle Manager to use Hardware Security Module (HSM) for storing the master encryption key, which protects the key materials that are stored in the database.

When you run the IBM Security Key Lifecycle Manager backup operation, a backup archive is created. The backup key in the archive encrypts backup contents. The master key in HSM encrypts the backup key. During the restore process, master key, which is stored in HSM, decrypts the backup key. Then, the backup key is used to restore backup contents.

If you use HSM to store the master key, the backup archive contains the following files:

- Manifest file, which lists all the IBM Security Key Lifecycle Manager data files in the archive.
- IBM Security Key Lifecycle Manager configuration files
- IBM Security Key Lifecycle Manager data dumps

For information about how to back up IBM Security Key Lifecycle Manager data by using HSM-based encryption, see Backing up data with HSM-based encryption.

HSM-based encryption is the default method for the backups when HSM is configured to store the master key. You can also use the password-based encryption for the backups when HSM is configured by setting the following property in the `SKLMConfig.properties` file.

```
enablePBEInHSM=true
```

Note:

- If HSM is not configured, you can only use password-based encryption for the backups.

- If the value for **enablePBEInHSM** is not set or set to any other value than `true`, the value is assumed as `false`.
- You can restore the backup file that is created by using either password-based or HSM-based encryption irrespective of the value set for **enablePBEInHSM**.

For information about how to back up IBM Security Key Lifecycle Manager data by using password-based encryption when HSM is configured, see Backing up data with password-based encryption when HSM is configured.

Categories of data in a backup file

A backup file of IBM Security Key Lifecycle Manager contains critical data. For example, depending on your configuration, it can include the key materials, configuration file, and other information.

The following categories of data require backup protection:

IBM Security Key Lifecycle Manager configuration files
: Properties that define selected IBM Security Key Lifecycle Manager activities such as audit settings and other values that you customize for your system configuration.

IBM Security Key Lifecycle Manager database
: Data about IBM Security Key Lifecycle Manager objects such as devices, key groups, certificates, key materials, and drives.

Backup file security

Ensure that you do not accidentally corrupt a backup file or misplace its encryption password.

To provide security for the backup files, use the following guidelines:

- Retain a copy of backup files in a location that is not on the IBM Security Key Lifecycle Manager computer, and not in the IBM Security Key Lifecycle Manager directory path. The separate location ensures that other processes cannot remove audit logs and backup files if IBM Security Key Lifecycle Manager is removed.
- Do not edit the files in a backup archive. The files become unreadable.
- Ensure that you retain the password that was used to encrypt a backup file. The same password is required to decrypt and restore the backup file.

Restore

A restore returns the IBM Security Key Lifecycle Manager server to a known state, by using backed-up production data, such as the IBM Security Key Lifecycle Manager key materials and other critical information.

IBM Security Key Lifecycle Manager supports restore operation across operating systems. You can restore IBM Security Key Lifecycle Manager backup files on an operating system that is different from the one it was backed up from. For example, you can restore a backup that was taken on a Linux system on to a Windows system.

Retrieve a copy of backup files from a location that you specified earlier, which is not in the IBM Security Key Lifecycle Manager directory path. When password-based encryption is used, you must know the password that was used to encrypt a backup file. Use the password to decrypt and restore the file on the primary IBM Security Key Lifecycle Manager server.

Note: When you use HSM-based encryption for the backups, you need not specify the password to restore data.

Before you start a restore task, isolate the system for maintenance. Take a backup of the existing system. You can later use this backup to bring the system back to original state if any issues occur during the restore process. IBM Security Key Lifecycle Manager server automatically restarts after the restore process is complete. Verify the environment before you bring the IBM Security Key Lifecycle Manager server back into production.

Replication overview

IBM Security Key Lifecycle Manager provides a set of operations to replicate current active files and data across systems. This replication enables cloning of IBM Security Key Lifecycle Manager environments to multiple servers in a manner that is independent of operating systems and directory structure of the server. For example, you can replicate data from a master server on a Windows system to a clone server on a Linux system.

You can define each instance of IBM Security Key Lifecycle Manager as either the master or a clone server. There can be only one master server with a maximum of 20 clones.

Master server
Master server is the primary system that is being replicated. Replication process is triggered only when the new keys, and devices are added or modified on the master server.

Clone server
Clone server is the secondary system to which the data is replicated.

Encryption methods to back up data for replication activities

IBM Security Key Lifecycle Manager supports the following encryption methods for backups:

Password-based encryption
When you configure the master server for automated replication, a password is specified to encrypt the backup key. When data is replicated on the clone server, the same encryption password is used to decrypt and restore the backup files.

HSM-based encryption
You can configure IBM Security Key Lifecycle Manager to use Hardware Security Module (HSM) for storing the master encryption key on master and clone servers. When you run the replication program, the backup key on the master server is encrypted by the master key, which is stored in HSM. When data is replicated on the clone server, the master key in HSM decrypts the backup key. Backup key is used to restore the backup contents.

For replication configuration information, see Replication configuration.

Backup encryption methods for replication activities

IBM Security Key Lifecycle Manager supports password-based encryption and HSM-based encryption for backups and replication activities.

Password-based encryption

When you run the IBM Security Key Lifecycle Manager automated replication program on the master server, you must specify a password to encrypt the backup key. This backup key is used to encrypt backup contents. The encrypted backup data, the backup key, and the password are replicated on the clone server that you configured for replication. The clone server uses the replicated password to decrypt and restore the backup files.

For information about how to back up and replicate IBM Security Key Lifecycle Manager data by using password-based encryption, see Configuring a master server with password-based encryption for backups.

HSM-based encryption

When you run the automated replication on the master server, data is backed up and encrypted by a backup key. If Hardware Security Module (HSM) is configured with IBM Security Key Lifecycle Manager, master key in HSM encrypts the backup key. When data is replicated on the clone server with HSM configured, the master key, which is stored in HSM, decrypts the backup key. Then, the backup key is used to restore backup contents.

HSM-based encryption is the default method for the backups and replication when HSM is configured to store the master key. You can also use the password-based encryption when HSM is configured by setting the following property in the `SKLMConfig.properties` file.

`enablePBEInHSM=true`

Note:

- If HSM is not configured, you can only use password-based encryption for the backups and replication.
- If the value for **`enablePBEInHSM`** is not set or set to any other value than `true`, the value is assumed as `false`.
- You can replicate and restore a backup file that is created by using either password-based or HSM-based encryption irrespective of the value set for **`enablePBEInHSM`**.

For information about how to back up and replicate IBM Security Key Lifecycle Manager data by using HSM-based encryption, see Configuring a master server with HSM-based encryption for backups.

For information about how to back up and replicate IBM Security Key Lifecycle Manager data by using password-based encryption when HSM is configured, see Configuring a master server with password-based encryption when HSM is configured.

Key management in a Multi-Master environment

In IBM Security Key Lifecycle Manager, high-availability can be achieved by using Multi-Master cluster configuration. All IBM Security Key Lifecycle Manager master servers in the multi-master cluster point to a single data source to ensure real-time availability of latest data to all the masters.

To provide continuous data availability to all the IBM Security Key Lifecycle Manager masters in a Multi-Master cluster, DB2 high availability disaster recovery (HADR) configuration is used. DB2 HADR is a database replication feature that

provides a high-availability solution. HADR protects against data loss by replicating data changes from a source database, called primary, to a target database, called the standby. DB2 HADR supports up to three standby databases in your Multi-Master setup.

Key features of IBM Security Key Lifecycle Manager Multi-Master configuration

- Keys that are created on an IBM Security Key Lifecycle Manager master are accessible to other IBM Security Key Lifecycle Manager masters in the cluster.
- IPP devices and KMIP clients that are registered on an IBM Security Key Lifecycle Manager master can access keys on another master in the cluster.
- Graphical user interface and REST interface to configure IBM Security Key Lifecycle Manager master servers for Multi-Master setup.

For more information about how to configure IBM Security Key Lifecycle Manager master servers for a Multi-Master setup, see Multi-Master configuration.

For more information about Multi-Master REST services, see Multi-Master configuration REST services.

User roles

IBM Security Key Lifecycle Manager provides a super user (`klmSecurityOfficer` and `klmGUICLIAccessGroup`) role and the means to specify more limited administrative roles to meet the needs of your organization. By default, the `SKLMAdmin` user ID has the `klmSecurityOfficer` role.

For backup and restore tasks, IBM Security Key Lifecycle Manager also installs the `klmBackupRestoreGroup` to which no user IDs initially belong. Installing IBM Security Key Lifecycle Manager creates predefined administrator, operator, and auditor groups to manage LTO tape drives.

The `WASAdmin` user ID has the authority to create and assign these roles, and to change the password of any IBM Security Key Lifecycle Manager administrator. To set administration limits for IBM Security Key Lifecycle Manager, use the `WASAdmin` user ID on the WebSphere Integrated Solutions Console to create roles, users, and groups. Assign roles and users to a group. For example, you might create a group and assign both users and a role that limits user activities to administer only LTO tape drives. You must assign a role to a new user before that user attempts to log in to IBM Security Key Lifecycle Manager.

Before you begin, complete the following tasks:

- Determine the limits on device administration that your organization requires.

  For example, you might determine that a specific device group has its own administration.
- Estimate how many administrative users might be needed over an interval of time. For ease of use, consider specifying a group and a role to specify their tasks.

  For example, you might specify a group that has a limited range of permissions to manage only 3592 tape drives.

Relations between users, groups, roles, and protected objects

To do useful work on protected objects, an IBM Security Key Lifecycle Manager user must have one or more roles. The role must enable an action such as create an object, such as a device, in the LTO device family.

A user can be a member of a group. A group might have one or more roles. A role specifies authorization for an operation on protected objects. For example, protected objects include devices, device groups, cryptographic objects (certificates, keys, key pairs, and key groups), and rollover settings for certificates and key groups.

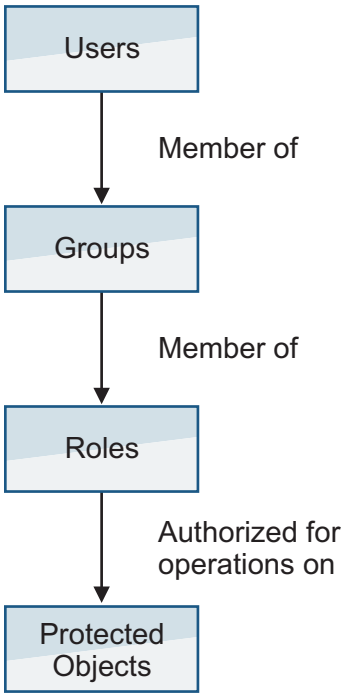

Figure 4. Relations between users, groups, roles, and protected objects

You can use WebSphere Integrated Solutions Console to create child groups with different permissions within a parent group. However, IBM Security Key Lifecycle Manager recognizes the permissions of only the parent group, not the permissions of its child groups.

Available permissions

Installing IBM Security Key Lifecycle Manager creates the SKLMAdmin user ID, which has the `klmSecurityOfficer` role as the default super user. The installation process also deploys predefined permissions to the WebSphere Application Server list of administrative roles.

A *permission* from IBM Security Key Lifecycle Manager enables an action or the use of a device group. A *role* in IBM Security Key Lifecycle Manager is one or more permissions. However, in the WebSphere Application Server graphical user interface, the term *role* includes both IBM Security Key Lifecycle Manager permissions and roles.

IBM Security Key Lifecycle Manager installation creates the following default groups.

klmSecurityOfficerGroup

Installation assigns the `klmSecurityOfficer` role to this group. The `klmSecurityOfficer` role replaces the previous `klmApplicationRole` role in the group that was named `klmGroup`. `klmSecurityOfficerGroup` replaces `klmGroup`.

The `klmSecurityOfficer` role has:

- Root access to the entire set of permissions and device groups that are described in Table 1 and Table 2 on page 28.
- Permission to any role or device group that might be created.
- The `suppressmonitor` role.

  The WebSphere Application Server provides the `suppressmonitor` role to hide tasks in the left pane of the WebSphere Integrated Solutions Console that an IBM Security Key Lifecycle Manager administrator does not use. Hidden items are associated with the application server, including WebSphere Application Server administrative tasks in the `Security`, `Troubleshooting`, and `Users and Groups` folders.

klmBackupRestoreGroup

Back up and restore IBM Security Key Lifecycle Manager.

LTOAdmin

Administer devices in the LTO device family with actions that include create, view, modify, delete, get (export), back up, and configure.

LTOOperator

Operate devices in the LTO device family with actions that include create, view, modify, and back up.

LTOAuditor

Audit devices in the LTO device family with actions that include view and audit.

klmGUICLIAccessGroup

Provides IBM Security Key Lifecycle Manager graphical user interface and command-line interface access to the users. Every product user must be a part of this group.

Note: Along with this access to the group, the users must be provided other accesses to be a functional product user.

A user who has any one of the permissions in Table 1 can view:

- IBM Security Key Lifecycle Manager global configuration parameters that are defined in the SKLMConfig.properties file.
- The key server status and last backup date.

Table 1. Permissions for actions

Permission	Enables these actions	Unrelated to device groups	Associated with device groups
`klmCreate`	Create but not view, modify, or delete objects.		✓
`klmDelete`	Delete objects, but not view, modify, or create objects.		✓
`klmGet`	Export a key or certificate for a client device.		✓

Table 1. Permissions for actions (continued)

Permission	Enables these actions	Unrelated to device groups	Associated with device groups
`klmModify`	Modify objects, but not view, create, or delete objects.		✓
`klmView`	View objects, but not create, delete, or modify objects. For example, you must have this permission to see the tasks you want to do on the graphical user interface.		✓
`klmAdminDeviceGroup`	Administer. Create a device group, set default parameters, view, delete an empty device group. This permission does not provide access to devices, keys, or certificates.	✓	
`klmAudit`	View audit data by using the **`tklmServedDataList`** command.	✓	
`klmBackup`	Create and delete a backup of IBM Security Key Lifecycle Manager data.	✓	
`klmConfigure`	Read and change IBM Security Key Lifecycle Manager configuration properties, or act on SSL certificate. Add, view, update, or delete the keystore.	✓	
`klmRestore`	Restore a previous backup copy of IBM Security Key Lifecycle Manager data.	✓	

The `klmSecurityOfficer` role also has root access to permissions for all device groups.

Table 2. Device groups

Permission	Allows actions on these objects
`LTO`	LTO device family
`TS3592`	3592 device family
`DS5000`	DS5000 device family
`DS8000`	DS8000 device family
`BRCD_ENCRYPTOR`	BRCD_ENCRYPTOR device group
`ONESECURE`	ONESECURE device group
`ETERNUS_DX`	ETERNUS_DX device group
`XIV`	XIV device group
`IBM_SYSTEM_X_SED`	IBM_SYSTEM_X_SED device group
`GPFS (IBM Spectrum Scale)`	GPFS device family
`GENERIC`	Objects in the GENERIC device family.
userdevicegroup	A user-defined instance such as `myLTO` that you manually create, based on a predefined device family such as LTO.

Multiple permissions

To work on devices, a user must have permissions for one or more actions and one or more device groups.

Errors occur if a user has:

Action permissions, but no device group permission
For example, the user has the set of action permissions that include view, create, modify, delete. However, the user has no device group permission to receive an action.

Device group permissions, but no action permission
For example, the user has device group permissions that include LTO and 3592. However, the user has no action permission to take against a device group.

A new role for a new device group, but no action permissions
For example, the user has a new role myLTO that was created for a new device group named myLTO. However, the user has no other action permissions.

Permissions might be:

- Directly assigned.

  For example, your role as a user might have view and modify permissions for a specific device group.
- Obtained by group membership.

  Permissions are specific to a device group. You might be a member of two user groups. For example, membership in one user group might grant view and modify permissions for use with an LTO device group. A second user group might grant view, create, and modify permissions for use with a 3592 device group. You can view and modify a device in either device group. However, you can complete a create action only for devices in the 3592 device group.

Data such as keys and certificates are associated with a device group. Such data is visible only in graphic user interface pages for the device group to which the data is associated. A user with permissions to several device groups can change the association of data from one device group to another for which the user holds appropriate permissions.

Some properties or attributes in the IBM Security Key Lifecycle Manager database are associated with device groups. For example, the **symmetricKeySet** attribute in the IBM Security Key Lifecycle Manager database is associated with the predefined LTO device group. To change the attribute, your role must have a permission to the modify action and a permission to the LTO device group.

Predefined groups to manage LTO tape drives

Installing IBM Security Key Lifecycle Manager creates predefined administrative groups to manage LTO tape drives. You can use these groups as a model to define similar administrative groups for other device groups.

LTOAdmin group

You can use membership in the LTOAdmin group to administer devices in the LTO device family with actions that include create, view, modify, delete, get (export), back up, and configure.

This group includes the following permissions:

Table 3. Permissions for actions

Permission	Enables these actions
LTO	LTO device family
klmCreate	Create but not view, modify, or delete objects.
klmDelete	Delete objects, but not view, modify, or create objects.
klmGet	Export a key or certificate for a client device.
klmModify	Modify objects, but not view, create, or delete objects.
klmView	View objects, but not create, delete, or modify objects.
klmAudit	View audit data by using the **tklmServedDataList** command.
klmBackup	Create and delete a backup of IBM Security Key Lifecycle Manager data.
klmConfigure	Read and change IBM Security Key Lifecycle Manager configuration properties, or act on SSL certificate.
suppressmonitor	Hide tasks in the left pane of WebSphere Integrated Solutions Console that an IBM Security Key Lifecycle Manager administrator does not need to use.

LTOOperator group

You can use membership in the LTOOperator group to operate devices in the LTO device family with actions that include create, view, modify, and back up.

This group includes the following permissions:

Table 4. Permissions for actions

Permission	Enables these actions
LTO	LTO device family.
klmCreate	Create but not view, modify, or delete objects.
klmModify	Modify objects, but not view, create, or delete objects.
klmView	View objects, but not create, delete, or modify objects.
klmBackup	Create and delete a backup of IBM Security Key Lifecycle Manager data.
suppressmonitor	Hide tasks in the left pane of the WebSphere Integrated Solutions Console that an IBM Security Key Lifecycle Manager administrator does not need to use.

LTOAuditor group

You can use membership in the LTOAuditor group to audit devices in the LTO device family with actions that include view and audit.

This group includes the following permissions:

Table 5. Permissions for actions

Permission	Enables these actions
`LTO`	LTO device family.
`klmView`	View objects, but not create, delete, or modify objects.
`klmAudit`	View audit data by using the **`tklmServedDataList`** command.
`suppressmonitor`	Hide tasks in the left pane of the WebSphere Integrated Solutions Console that an IBM Security Key Lifecycle Manager administrator does not need to use.

WebSphere Application Server roles

WebSphere Application Server provides roles that you might need to use. For example, you might need to view or change the WebSphere Application Server configuration. You might assign users and groups to administrative user roles and administrative group roles.

The roles include monitor, configurator, operator, administrator, security manager, and other roles.

For more information, search for *administrative roles* in the WebSphere Application Server documentation (http://www.ibm.com/support/knowledgecenter/SSAW57_9.0.0/as_ditamaps/was900_welcome_ndmp.html).

Release information

The Release information topics describe information that is specific to this release of IBM Security Key Lifecycle Manager.

System requirements

Your environment must meet the minimum system requirements to install IBM Security Key Lifecycle Manager.

For information about hardware and software requirements, see the "Installing and configuring" section on IBM Knowledge Center for IBM Security Key Lifecycle Manager. The hardware and software requirements that are published are accurate at the time of publication.

Alternatively, see the detailed system requirements document at http://pic.dhe.ibm.com/infocenter/prodguid/v1r0/clarity/softwareReqsForProduct.html.

1. Enter `IBM Security Key Lifecycle Manager`.
2. Select the product version. For example, `3.0`.
3. Select the operating system.
4. Click **Submit**.

Installation images and fix packs

Obtain IBM Security Key Lifecycle Manager installation files from the IBM® Passport Advantage® website and fix packs from Fix Central. You can also obtain the files by another means, such as a DVD as provided by your IBM sales representative.

The Passport Advantage website provides packages, referred to as eAssemblies, for various IBM products at http://www-01.ibm.com/software/passportadvantage/pao_customer.html.

You can use Fix Central to find the fixes that are provided by IBM Support for various products, including IBM Security Key Lifecycle Manager at https://www-945.ibm.com/support/fixcentral. With Fix Central, you can search, select, order, and download fixes for your system with a choice of delivery options. A IBM Security Key Lifecycle Manager product fix might be available to resolve your problem.

Notices

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

IBM 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:

IBM Director of Licensing
IBM Corporation
North Castle Drive, MD-NC119
Armonk, NY 10504-1785
US

For license inquiries regarding double-byte character set (DBCS) information, contact the IBM Intellectual Property Department in your country or send inquiries, in writing, to:

Intellectual Property Licensing
Legal and Intellectual Property Law
IBM Japan, Ltd.
19-21, Nihonbashi-Hakozakicho, Chuo-ku
Tokyo 103-8510, Japan

The following paragraph does not apply to the United Kingdom or any other country where such provisions are inconsistent with local law :

INTERNATIONAL BUSINESS MACHINES CORPORATION 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. IBM may make improvements and/or changes in the product(s) and/or the program(s) described in this publication at any time without notice.

Any references in this information to non-IBM 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 IBM product and use of those Web sites is at your own risk.

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

Licensees of this program who wish to have information about it for the purpose of enabling: (i) the exchange of information between independently created programs and other programs (including this one) and (ii) the mutual use of the information which has been exchanged, should contact:

IBM Director of Licensing
IBM Corporation
North Castle Drive, MD-NC119
Armonk, NY 10504-1785
US

Such information may be available, subject to appropriate terms and conditions, including in some cases payment of a fee.

The licensed program described in this document and all licensed material available for it are provided by IBM under terms of the IBM Customer Agreement, IBM International Program License Agreement or any equivalent agreement between us.

Any performance data contained herein was determined in a controlled environment. Therefore, the results 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 measurement may have been estimated through extrapolation. Actual results may vary. Users of this document should verify the applicable data for their specific environment.

Information concerning non-IBM products was obtained from the suppliers of those products, their published announcements or other publicly available sources. IBM has not tested those products and cannot confirm the accuracy of performance, compatibility or any other claims related to non-IBM products. Questions on the capabilities of non-IBM products should be addressed to the suppliers of those products.

All statements regarding IBM's future direction or intent are subject to change or withdrawal without notice, and represent goals and objectives only.

All IBM prices shown are IBM's suggested retail prices, are current and are subject to change without notice. Dealer prices may vary.

This information is for planning purposes only. The information herein is subject to change before the products described become available.

This information contains examples of data and reports used in daily business operations. To illustrate them as completely as possible, the examples include the names of individuals, companies, brands, and products. All of these names are fictitious and any similarity to the names and addresses used by an actual business enterprise is entirely coincidental.

COPYRIGHT LICENSE:

This information contains sample application programs in source language, which illustrate programming techniques on various operating platforms. You may copy, modify, and distribute these sample programs in any form without payment to IBM, for the purposes of developing, using, marketing or distributing application programs conforming to the application programming interface for the operating platform for which the sample programs are written. These examples have not been thoroughly tested under all conditions. IBM, therefore, cannot guarantee or imply reliability, serviceability, or function of these programs. The sample programs are provided "AS IS", without warranty of any kind. IBM shall not be liable for any damages arising out of your use of the sample programs.

Each copy or any portion of these sample programs or any derivative work, must include a copyright notice as follows:

© (your company name) (year). Portions of this code are derived from IBM Corp. Sample Programs. © Copyright IBM Corp. _enter the year or years_.

If you are viewing this information in softcopy form, the photographs and color illustrations might not be displayed.

Terms and conditions for product documentation

Permissions for the use of these publications are granted subject to the following terms and conditions.

Applicability
These terms and conditions are in addition to any terms of use for the IBM website.

Personal use
You may reproduce these publications for your personal, noncommercial use provided that all proprietary notices are preserved. You may not distribute, display or make derivative work of these publications, or any portion thereof, without the express consent of IBM.

Commercial use
You may reproduce, distribute and display these publications solely within your enterprise provided that all proprietary notices are preserved. You may not make derivative works of these publications, or reproduce, distribute or display these publications or any portion thereof outside your enterprise, without the express consent of IBM.

Rights Except as expressly granted in this permission, no other permissions, licenses or rights are granted, either express or implied, to the publications or any information, data, software or other intellectual property contained therein.

IBM reserves the right to withdraw the permissions granted herein whenever, in its discretion, the use of the publications is detrimental to its interest or, as determined by IBM, the above instructions are not being properly followed.

You may not download, export or re-export this information except in full compliance with all applicable laws and regulations, including all United States export laws and regulations

IBM MAKES NO GUARANTEE ABOUT THE CONTENT OF THESE PUBLICATIONS. THE PUBLICATIONS ARE PROVIDED "AS-IS" AND WITHOUT WARRANTY OF ANY KIND, EITHER EXPRESSED OR

IMPLIED, INCLUDING BUT NOT LIMITED TO IMPLIED WARRANTIES OF MERCHANTABILITY, NON-INFRINGEMENT, AND FITNESS FOR A PARTICULAR PURPOSE.

Trademarks

IBM, the IBM logo, and ibm.com are trademarks or registered trademarks of International Business Machines Corp., registered in many jurisdictions worldwide. Other product and service names might be trademarks of IBM or other companies. A current list of IBM trademarks is available on the web at http://www.ibm.com/legal/copytrade.shtml.

Adobe, Acrobat, PostScript and all Adobe-based trademarks are either registered trademarks or trademarks of Adobe Systems Incorporated in the United States, other countries, or both.

IT Infrastructure Library is a registered trademark of the Central Computer and Telecommunications Agency which is now part of the Office of Government Commerce.

Intel, Intel logo, Intel Inside, Intel Inside logo, Intel Centrino, Intel Centrino logo, Celeron, Intel Xeon, Intel SpeedStep, Itanium, and Pentium 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.

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

ITIL is a registered trademark, and a registered community trademark of the Office of Government Commerce, and is registered in the U.S. Patent and Trademark Office.

UNIX is a registered trademark of The Open Group in the United States and other countries.

Java and all Java-based trademarks and logos are trademarks or registered trademarks of Oracle and/or its affiliates.

Cell Broadband Engine is a trademark of Sony Computer Entertainment, Inc. in the United States, other countries, or both and is used under license therefrom.

Linear Tape-Open, LTO, the LTO Logo, Ultrium, and the Ultrium logo are trademarks of HP, IBM Corp. and Quantum in the U.S. and other countries.

Index

Numerics

A

B

C

D

E

F

G

H

I

K

L

M

N

O

P

R

S

T

U

W

X