Hewlett Packard Enterprise

Business white paper

HPE Enterprise Secure Key Manager

Key Protection Best Practices

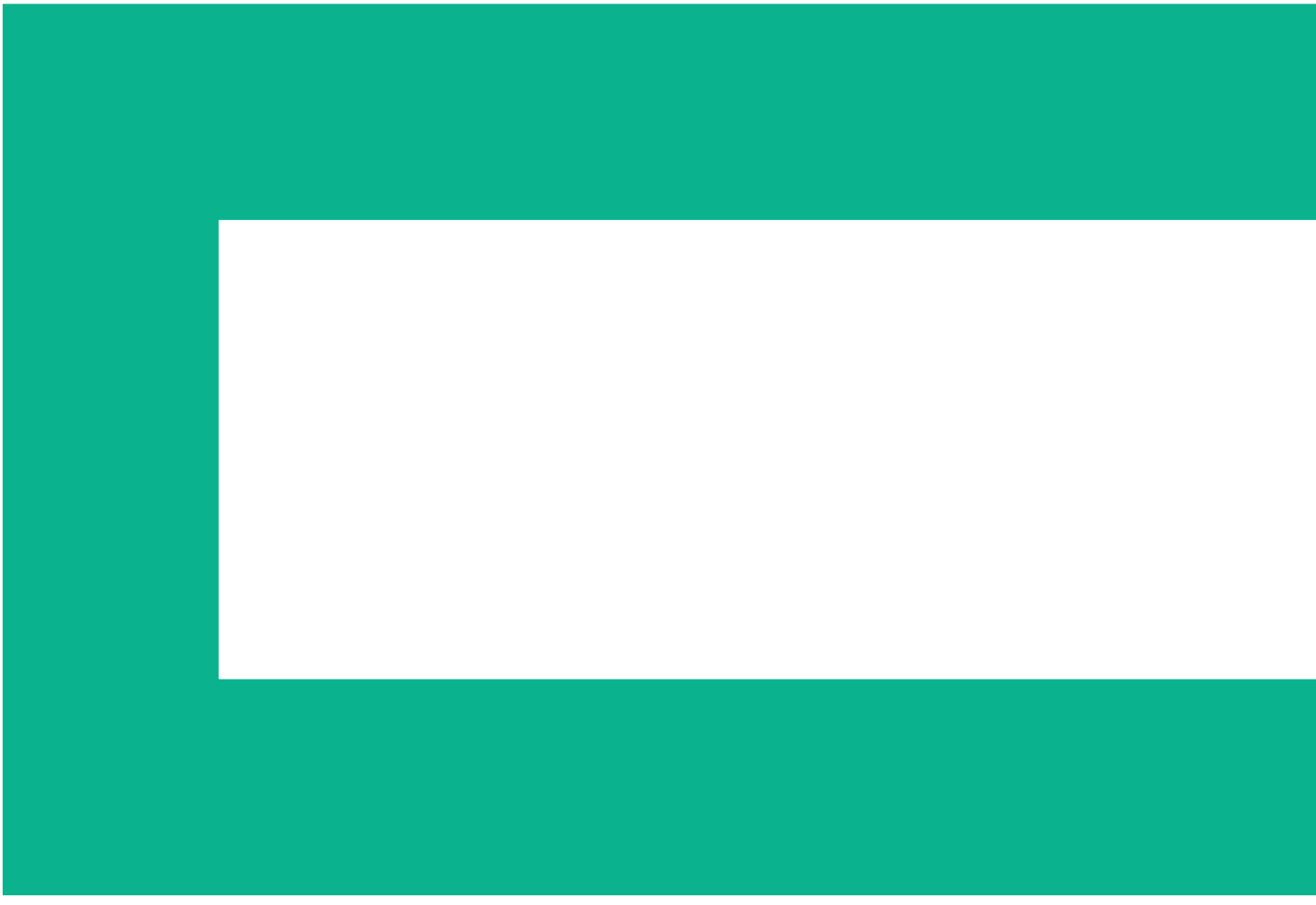

Table of contents

Introduction

The HPE Enterprise Secure Key Manager (ESKM) provides a secure, centralized key management solution for data encryption environments, and merges seamlessly into successfully integrated HPE enterprise solutions. Its user guide provides the necessary information for configuring the ESKM, but users might have more indepth questions about the capabilities and requirements for using the ESKM in specific key management instances. The HPE ESKM Best Practices Guide will answer many of those questions and provide guidance as to how to keep your ESKM key management environment in optimal condition.

About the HPE Enterprise Secure Key Manager

Figure 1: HPE Enterprise Secure Key Manager

The HPE Enterprise Secure Key Manager is a unified solution for encryption key management and security policy enforcement across the enterprise. HPE ESKM automates generation and retrieval of encryption keys for multiple client applications and devices based on security policies set by your security officers. Key management transactions occur quickly and transparently to business application users. HPE ESKM supports customer policies for client access and key sharing groups, and it flexibly accommodates new policies, new client types, and new use cases for secure key management services. HPE ESKM provides reliable lifetime access to keys using mirrored storage, high availability clustering, automatic multi-site key replication, and full backup/restore. For continuous availability, encryption clients can access ESKM cluster members across multiple geographic locations using transparent path and node failover capabilities.

The ESKM is a hardened server appliance that has been validated as a complete hardware and software security solution under the rigorous FIPS 140-2 Level 2 security requirements for cryptographic modules.

About the Best Practices Guide

This paper will focus on seven main topics relating to the operation and protection of the Enterprise Secure Key Manager:

- Recommended configuration options
- Password management
- Rekeying
- Users, groups, and administrator accounts
- Log maintenance
- Backup strategies
- Clustering

User questions such as, “How often should I change administrator account passwords?”, “How long do I need to maintain the ESKM logs?”, “Is there any way to automate the backups of my ESKM?”, and “What happens if I forget the admin password?” will be covered in these topics. This paper is intended to provide suggestions and examples as to how to integrate the ESKM with existing security and data protection policies. The thing to keep in mind is that, as an ESKM user, you have invested in a data encryption solution in order to secure data. It only makes sense to also protect your ESKM environment as best as possible.

Recommended Configuration Options

The purpose of this section is to provide recommendations on how to securely configure the ESKM. Prior to completing the configuration of the ESKM, it is essential to review your company-wide security policy and ensure that the chosen configuration settings comply with the ESKM’s security policy.

This section will focus on operating the ESKM in FIPS mode, enabling TLS/SSL, setting the recommended authentication settings, and configuring a dedicated Network Interface Connector (NIC) to support external clients. In particular, the following recommended steps will be covered:

- Configuring NIC2 to support external clients
- Enable FIPS mode
- Create CA and server certificates
- Enable TLS/SSL
- Enable security-relevant flags
- Enable mutual authentication

The initial setup of the ESKM is described in the ESKM Installation Guide and requires the ESKM to be configured via the serial interface. After completing these configuration steps, the administrator is required to complete the configuration of the ESKM via the web-based management interface or the Command Line Interface (CLI). The configuration steps discussed in this section should be completed during this time.

Configure NIC2 to support external clients

In certain situations it may be necessary for external clients to connect to the ESKM. An example of an external client may be the HPE Helion product, as a potential internet-based client. To maintain strong security HPE recommends that you configure and dedicate one port on NIC2 to support external clients. Specific ports on NIC1 will be used for internal client connections as well as ESKM management and clustering. Configuration changes for NIC1 and NIC2 is accomplished via the CLI or the web-based management interface.

The first step in the process is to define the IP address for NIC2 and the port number external clients will use to connect to the ESKM, the default value is port 9000 Provide this information to the network management team so they can correctly configure the firewall to ensure that only connections to that IP address/port number are allowed through the firewall.

NOTE: The KMS Server Settings allow you to specify the port number that will be used by both internal clients which connect to NIC1 and also by external clients which connect to NIC2.

The next step in the process is to edit and save the KMS Server settings to specify **All** for the IP address, allow key and policy configuration operations, allow key export, and if necessary change the port number from its default value.

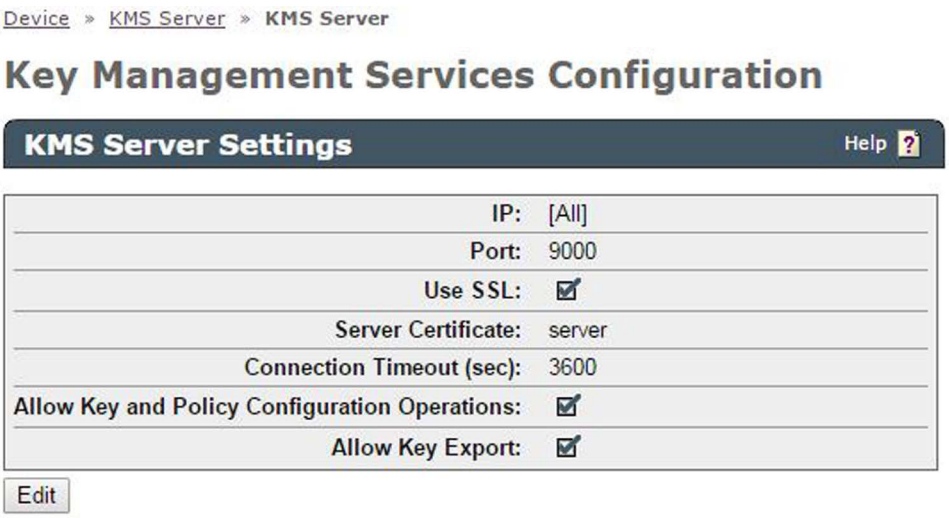

Figure 2: Configuring the KMS Server Settings

See the KMS Server Setting section in the HPE Enterprise Secure Key Manager users guide for additional guidance.

The next step is to add the IP address of NIC2 to the ESKM network interface list.

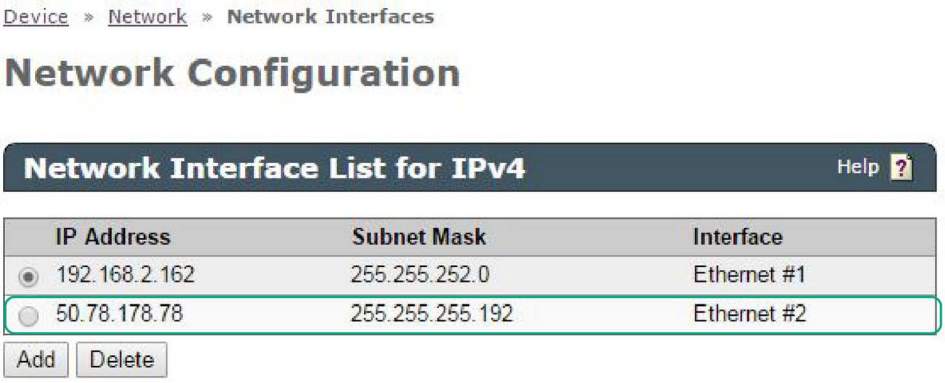

Figure 3: Adding NIC2

See the Network interface section in the HPE Enterprise Secure Key Manager users guide for additional guidance.

The final step is to use the “gateway” and “outgoing gateway” CLI commands to configure a gateway for NIC2.

NOTE: Only one NIC can be configured as the outgoing gateway.

After making these configuration changes you can then connect an Ethernet cable between the ESKM NIC2 and the switch/router.

FIPS Mode

The ESKM has been validated to FIPS 140-2 level 2 and supports a FIPS mode of operation that can be enabled by the administrator. HPE recommends for the ESKM to operate in FIPS mode. This can be configured via the CLI with the “fips compliant” command in config mode or via the web-based management interface by clicking on the “Set FIPS Compliant” button under Security > High Security, as shown below.

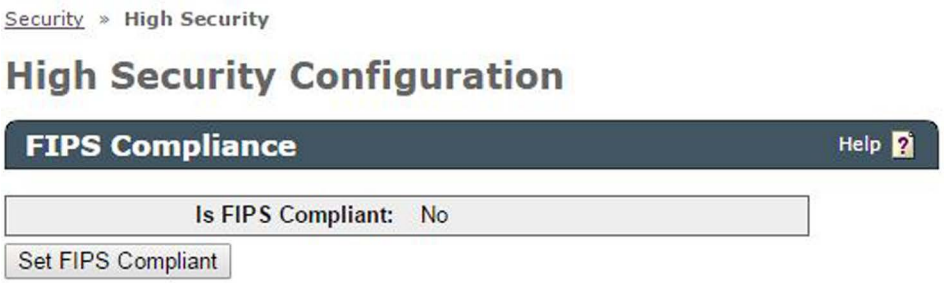

Figure 4: Enabling FIPS mode

When operating in a FIPS mode of operation, the ESKM cannot use non-FIPS-approved algorithms for cryptographic operations. Non-approved TLS/SSL cipher suites will be disabled and only FIPS-approved keys can be created. The ESKM will also disable global keys, FTP, LDAP, and SSL 3.0.

To verify that the ESKM is operating in FIPS mode, enter “show fips status” in the CLI or verify the following settings in the web-based management interface under High Security:

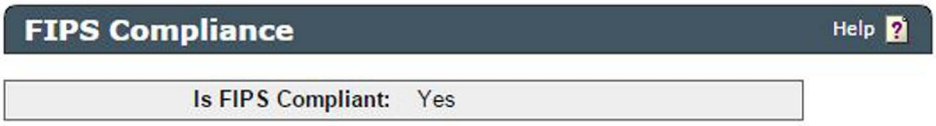

Figure 5: Operating in FIPS mode

The ESKM allows a FIPS Status Report to be created. This can be enabled under Security > FIPS Status Server. The report can be viewed by entering the following address and default port number into a web browser: https://<ESKM IP address>:9081/status.html. The report provides information on the state of the ESKM and the FIPS mode status. It also provides a list of power-up and conditional self-tests that were performed, including the date and time of execution and whether they succeeded or failed.

TLS (Transport Layer Security)/SSL (Secure Sockets Layer)

When operating in a FIPS mode of operation, the communication between the local user and ESKM must be protected using TLS 1.0. This is particularly required when transmitting keys during a key import or export to avoid the transmission of plaintext keys. The ESKM does not implicitly enable TLS/SSL when FIPS mode is enabled and requires TLS/SSL to be enabled separately. When enabling TLS/SSL, the ESKM requires at a minimum for the server to authenticate to the client. Thus, prior to enabling TLS/SSL, the server certificate needs to be created. See the ESKM Installation Guide and Users Guide for additional guidance on how to create sever and Local CA certificates. HPE recommends selecting a key size of 2048 bits when creating these certificates.

After successfully creating the Local CA and server certificate, TLS/SSL can be enabled. Under KMS Server Settings, as shown below, enable "Use SSL", select the newly created Sever Certificate, and enable "Allow Key and Policy Configuration Operations" and "Allow Key Export". These two flags need to be enabled to allow the client (local user) to perform secure key management services. The ESKM will by default use TLS 1.0 and only cipher suites that use FIPS-approved algorithms can be used when operating in FIPS mode.

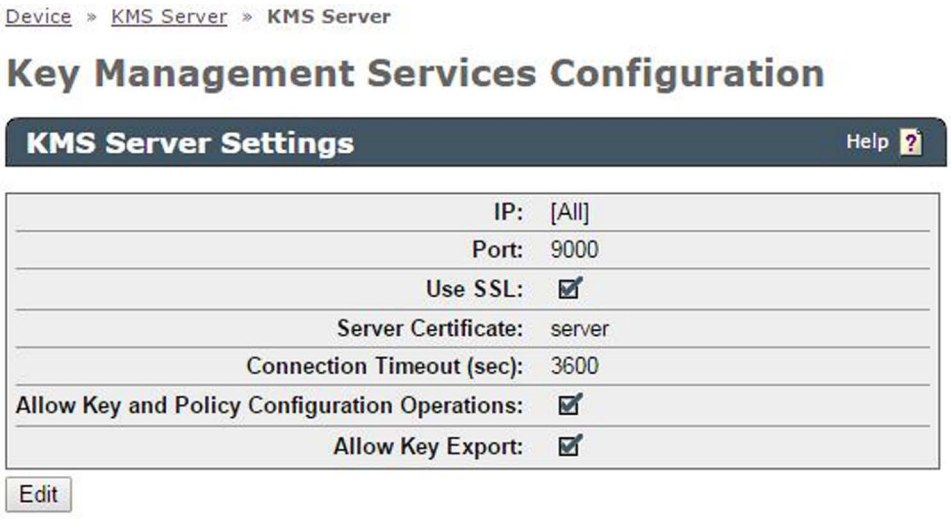

Figure 6: Enabling SSL/TLS

Mutual Authentication

Enabling Client Certificate Authentication

To prevent attacks such as man-in-the-middle attacks, HPE recommends enabling mutual authentication. This can be configured via the CLI or the web-based management interface by enabling client certificate authentication. Prior to enabling this feature, it is important to review the company-wide security policy to identify any specific requirements for creating client certificates. The security policy may mandate how to create and store client key pairs and specify how client certificate requests should be signed.

The ESKM provides Certificate Authority services and the Local CA can be used to sign client certificate requests. The ESKM may also be used to create the client key pair. This may not be ideal for a couple of reasons. Firstly, the client key pair should be stored securely by the client and exposure of the private key should be kept to a minimum. Secondly, the ESKM will not create the client certificate request from the corresponding key pair. Customers are therefore recommended to securely create their own RSA key pair and corresponding certificate request. HPE recommends unique client certificates with a key size of at least 2048 bits.

NOTE: When using the ESKM SDK 1.1.5 or above to communicate with the ESKM and the library is operating in FIPS mode, the encrypted private key must be in PKCS#8 format. Please see the HPE Enterprise Secure Key Manager Client Developer's Guide C Application Programming Interface for additional guidance.

The certificate request can be signed by the ESKM's Local CA or a third-party CA. When using a third-party CA to sign client certificate requests, the third-party CA certificate needs to be imported into the ESKM as a Known CA. The CA (Local CA or third-party CA) used to sign the client certificate requests needs to be added to a Trusted CA List profile. The ESKM comes with the default Trusted CA List profile called Default. Additional profile names can be created.

After successfully adding the CA to the Trusted CA List, client certificate authentication can be enabled. The authentication settings can be found under KMS Server Authentication Settings under the Device tab, as shown below.

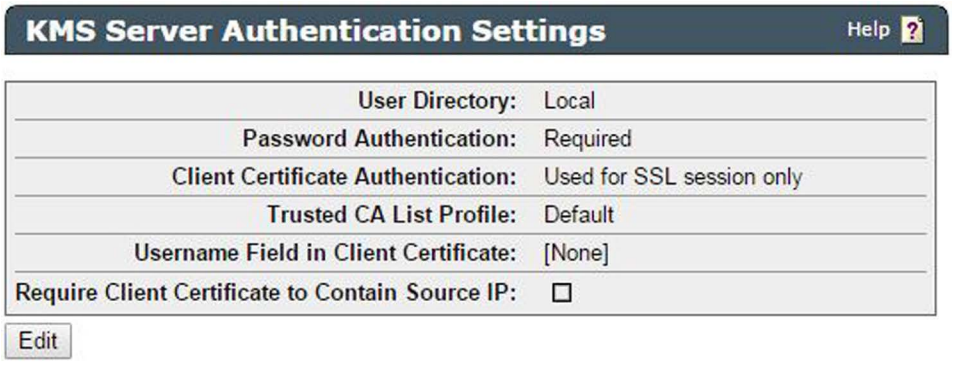

Figure 7: Enabling Client Certificate Authentication

User Directory and Password Authentication should always be set to Local and Required, respectively. There are three options for Client Certificate Authentication: Not Used, Used for SSL Session only, Used for SSL session and username (most secure). At the very least, Used for SSL Session should be selected. The third option Used for SSL session and username is the most secure option and requires the username to be specified inside the certificate. Several fields can be used for this purpose, among these the User ID, Common Name, Surname, Email address, and Organizational Unit. When enabling this feature, the ESKM will verify that the username provided along with the password matches the username specified in the certificate. This approach may not be suitable for every environment and requires a thorough review of the architecture and company-wide security policy. In addition to specifying the username inside the certificate, the source IP can be included for an additional level of security. Please review the HPE Enterprise Secure Key Manager users guide for additional guidance on these options.

Enabling Web Admin Client Certificate Authentication

In addition to supporting mutual authentication via the KMS interface (i.e., port 9000), the ESKM allows mutual authentication to be enabled when connecting to the web-based management interface. This feature is recommended if strong security is a requirement and administrators are able to provide their client certificate when accessing the ESKM Graphical User Interface (GUI). This may not be suitable for every environment, especially if administrators are hoping to use any web browser or machine (connected to the network) to access the ESKM GUI. Prior to enabling Web Admin Client Certificate Authentication, the following steps must be performed:

- Create admin key pair
- Create admin certificate request and specify the administrator login name (e.g., admin) as the Common Name
- Sign the certificate request with a Known CA or Local CA
- If the certificate request is signed with a Known CA, import the Known CA certificate into the ESKM
- Add the Known CA or Local CA to the Trusted CA List
- Convert the admin certificate from PEM to PKCS12 format
- Import the certificate to a Web browser

The Web Admin Client Certificate Authentication feature should only be enabled after completing the aforementioned steps. See the figure below for the two configuration changes required to enable Web Admin Client Certificate Authentication. Note that this feature will immediately be enabled. The administrator will be logged off and will need a valid certificate to return. If required, the feature can be disabled via the CLI. See the HPE Enterprise Secure Key Manager users guide for additional guidance on the above steps.

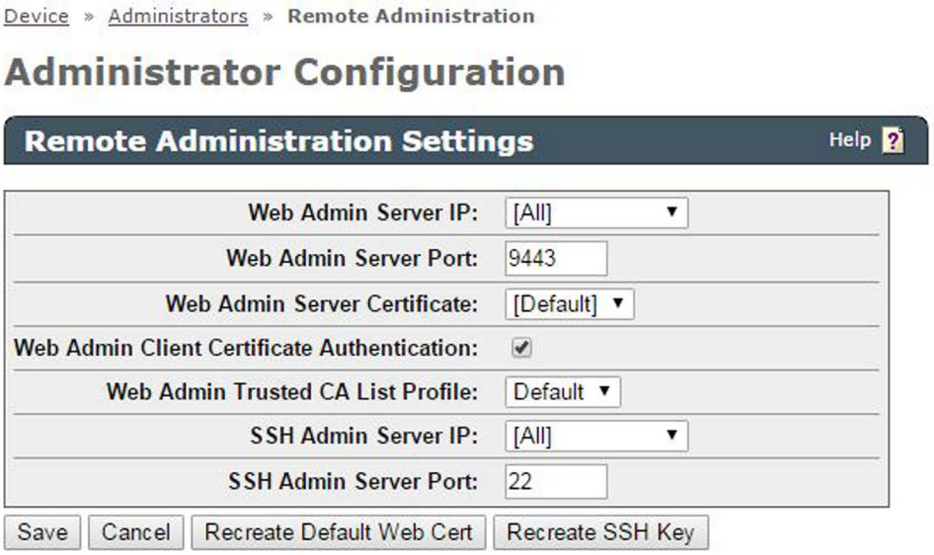

Figure 8: Enabling Web Admin Client Certificate Authentication

Password Management

The various passwords necessary to perform tasks on the ESKM will, on occasion, need to be changed. Sometimes this is just to maintain good security practices; other times it will be a necessity due to a personnel change. In any event, there are differing needs for different ESKM passwords, and administration policies should be set in place to reflect these needs.

The passwords that will be used on the ESKM are as follows:

- Administrator passwords for the administrator accounts on the ESKM
- User account passwords that clients use to connect to the ESKM
- Cluster passwords that allow the ESKM nodes to securely replicate information
- Backup passwords that will be needed to execute backups and restores of ESKM data

Administrator passwords, even if the account permissions do not allow complete access to the ESKM, should be changed regularly. These accounts are too important to be allowed to become a security risk by rarely or never changing the passwords.

User accounts are used by key creators/users to access the ESKM key management services and may allow access to sensitive keys. Similar to administrator passwords, best practices mandate changing these passwords at some reasonable interval. When enabling client certificate authentication, user accounts are protected by an additional layer of security and do not solely rely on the password string. Nevertheless, it is important to frequently replace these passwords.

Backup passwords are used to protect backup files and are provided to the ESKM while creating the backup. Backup files are encrypted and may contain keys, configuration settings, or both keys and configuration settings. It is important to keep track of the password used to create a backup file, as it is required to successfully restore your backup file. Backup files can only be restored with the corresponding backup password. HPE will not be able to recover your keys from backup files without the corresponding backup password. As a result, it is essential to adequately manage your backup passwords.

Cluster passwords can be changed less frequently, because they cannot be changed without deleting the cluster completely. This can be done, but is not a trivial process, and care must be taken to ensure that no information is overwritten during the re-creation of the cluster following the password change.

Password protection is, as always, a high priority. Password knowledge should be kept to a minimum, and should only be shared when necessary. A select few employees should be designated as security officers with access to the ESKM nodes, and no one person should have complete administrator access exclusively. In the event that a security officer leaves their position or is terminated, administrator, user account, and backup passwords should be changed immediately to protect the integrity of the ESKM configuration. Knowledge of the cluster password would only allow an ESKM to join a cluster; this is less of a security concern. If the administrator passwords have been protected, unauthorized personnel would not be able to access an ESKM to join it to a cluster.

The intervals between password changes depend on your security requirements. Some users need to be highly restrictive in how long they maintain passwords, while others might find changing them too frequently to be troublesome. However, HPE Security recommends your organization have a stated policy concerning the changing of passwords that is communicated to all security officers and other necessary corporate personnel. It is also advisable to stagger the password changes within that policy, so that, for example, at least one full-access administrator account is unchanged during the process. An exception to this is when a security officer changes jobs or leaves the company, especially one that has significant administrator access to the ESKM.

In the event of a security officer personnel change, HPE Security strongly recommends that the passwords for administrator accounts, user accounts, and backups should all be changed to protect ESKM integrity. This procedure should be handled quickly but deliberately, so that access to the ESKM configuration is secured but not in a haphazard manner. It is best to have a documented procedure in place to handle such a situation. One possible procedure is the following:

1. Delete the former security officer's administrator account or change the corresponding password immediately. When deleting the account, consider creating a new administrator account with the same permissions. Have the replacement security officer use the new account.

   NOTE: Only an administrator with High Access Administrator right is able to change an administrator password or delete an administrator account.

2. Have each remaining security officer change their administrator account password, preferably with at least one other security officer present to witness the password change.
3. Change the user account passwords on both the ESKM and the enrolled client.

   IMPORTANT: Step 3 could briefly interrupt the ability for the client to retrieve keys during the change and verification; this should be done outside the backup window at the earliest convenience.

4. If an automated script is being used to run the backup jobs, change the password information in the script.

If the administrator account for the former security officer is less critical, this policy could be lessened, depending on the circumstances. For example, if the account only had the permissions to run backup and restore jobs, that account could not change other administrator account passwords and therefore would not pose a threat to those accounts. In that scenario, it might not be considered necessary to change all the other administrator accounts or other passwords. It might be considered sufficient to delete that account and create a new one with a different name and password for the replacement officer. In addition, the backup passwords should also be changed in this scenario.

In any event, it is critical to have a documented policy in place to ensure that the change in security personnel has no effect on the integrity of the ESKM system and the data protected by the encryption keys.

IMPORTANT: It is absolutely crucial that multiple administrator account passwords are remembered and available at all times. If all the administrator passwords are lost or forgotten, the ESKM system must be reinitialized in order to clear out the administrator accounts. This will result in a deletion of the keys maintained by the ESKM. Keys can be restored from an ESKM backup. If separate backups for keys and configuration information are not used, be sure to avoid restoring the administrator's settings. Take appropriate steps (while maintaining the proper security measures listed previously) to ensure that the administrator passwords are always available.

Rekeying

Similar to the requirement of frequently changing the password, it is recommended to have a procedure in place for rekeying keys. The rekeying requirement may be based on time, key usage or other factors and are often mandated by auditors or for regulatory/compliance reasons.

The versioned key feature allows clients to easily rekey client keys by performing a rekey without having to recreate the core attributes of a key (including the name, algorithm, and key size). The ESKM allows clients to create 4000 versions of a key and provides the capability to set the state of a versioned key. Clients may use the custom attribute to set the expiration date of a key and use this to determine when it is time to rekey. In addition to rekeying client keys, HPE recommends changing the keys used to protect the communication from and to the ESKM. These include the TLS keys and certificates, as discussed above, used to protect the KMS interface (i.e., port 9000). The management interfaces are protected using SSL/TLS (GUI) and SSH (CLI). The keys are first created during the initial installation. To recreate these keys, go to **Device > Administrators > Remote Administration**, as shown below, and click on **Recreate Web Cert** to recreate the server keys and cert and Recreate SSH Key to recreate the keys used to protect the remote CLI communication.

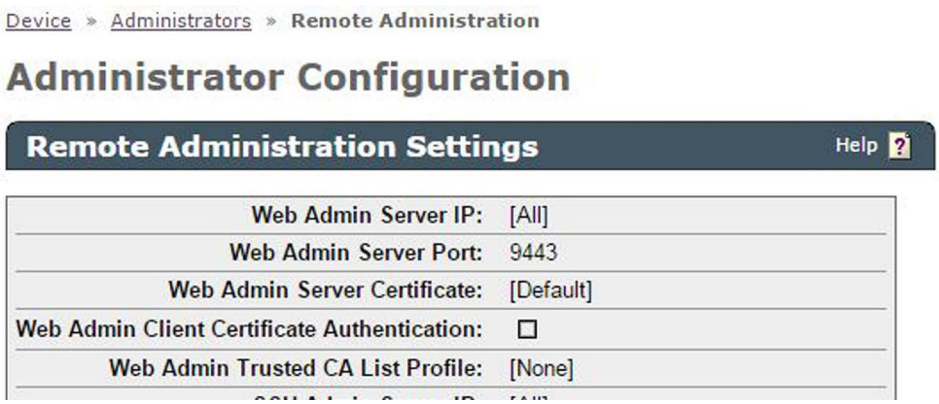

Figure 9: Creating new Web Cert and SSH Keys

Users, Groups, and Administrator Accounts

User/Group Procedures

User accounts on the Enterprise Secure Key Manager are created to allow clients to log in to create and retrieve keys from the ESKM. These are different from administrator accounts, which handle the administrative and maintenance duties with the ESKM. Key Sharing Groups are used to allow access to keys by multiple user accounts. For example, if a client creates a key and another user requires access to that same key, the user seeking access to the key can be added to the group of users that is allowed to access the key. User accounts and groups can be handled in one of two ways on the Enterprise Secure Key Manager. Accounts and groups can be managed locally on the ESKM and shared among clustered nodes. This is the preferred method, as this maintains the Federal Information Processing Standards (FIPS) compliance for the nodes.

If centrally managing all user and group accounts is a priority, the ESKM can be set up to use a Lightweight Directory Access Protocol (LDAP) to query an Active Directory (AD) or other LDAP server and the ESKM accounts can be stored there. This functionality must be enabled in ESKM—by default, only local users and groups are recognized. This is also an either/or scenario—either local accounts are used or LDAP accounts, but not both. Again, if LDAP users and groups are used, the ESKM configuration will no longer be FIPS-compliant. This needs to be taken into consideration when setting up the ESKM. HPE recommends operating the ESKM in FIPS mode and therefore does not recommend the usage of LDAP.

When creating users on the ESKM, HPE Security recommends creating users without User Admin Permission and Change Password Permission. It is not necessary for users to have such privileged permissions in order to utilize the ESKM key management functionality.

Administrator Accounts

It is best to have more than one administrator account with full-access privileges, but these should be limited to two or three at most. This way, if a full-access administrator account password is forgotten, the account can be deleted and re-created by the alternate full-access account. Or, if there is a personnel change involving full-access administrators, that account can quickly be deleted to eliminate that possible security risk.

Some users might want to set up alternate full-access administrator accounts and delete the default “admin” account, but this should be done with great caution. Ensure that the alternate accounts do indeed have full ESKM access before removing the admin account. Other accounts can be set up for tasks that require less access, such as backups or key creation, without needing to provide full system access. Administrators should take into consideration how many employees need to have access to the ESKM configuration and limit that access as much as possible, while still allowing enough access to maintain the configuration smoothly.

Multiple Credentials

The ESKM can be configured to require multiple administrator accounts to confirm configuration changes by enabling the **Multiple Credentials** feature. Once this is enabled, all changes to the ESKM, whether it be adding user accounts, creating certificates, or even changing the administrator account password, must be confirmed by the chosen number of administrators. It is an all-or-nothing feature, so it might be considered too troublesome, in some instances. However, Multiple Credentials does provide an extremely potent level of security for those who want to make sure that nothing happens in the ESKM environment without consensus approval.

An additional aspect of Multiple Credentials is the ability to grant credentials on a temporary basis. For example, say a security officer will be temporarily unavailable due to an offsite meeting. By enabling the **Allow Time-Limited Credentials** feature and selecting the maximum number of minutes that the credentials can be granted, that officer can grant permissions to another officer to allow selected operations to be approved. Once that time period expires, the second officer will no longer be able to approve the selected operations. This can provide continuous protection without having to grant additional permissions to administrator accounts on a permanent basis.

Log Maintenance

The ESKM logs are helpful in providing troubleshooting information and validation services information and for monitoring overall system performance. The four logs, by default, are set to be rotated once a day (**Activity** and **Client Event**) or once a week (**System** and **Audit**). When a log is rotated, by default, the current log is closed and archived internally and a new log is opened. When the limit on archived logs is reached, the oldest log is removed upon archival of the next log.

After a certain period of time, there might be little value for many users in maintaining the ESKM logs. However, there could be others who, for regulatory compliance reasons, must maintain the ESKM logs for years, if not indefinitely. Since the ESKM is not designed to be a long-term log storage repository, the ESKM provides the ability to archive older log files to an external log server, if maintaining the log files is a necessity. The HPE Security ArcSight SIEM (Security Information and Event Management) product can be used in tandem with the ESKM to store the excess log files. In addition, the SIEM product provides tools that can provide useful analysis of the logs, enabling users to study long-term trends and take appropriate action, if necessary, to improve the overall performance of the ESKM configuration.

The syslog feature that allows event notifications to be sent to an external server is not a recommended method of transferring log information from the ESKM. Syslog data is sent unencrypted and unsigned, so it is completely unprotected. In addition, this does not supplant the regular log rotation—if left by default, the oldest logs will be deleted when the expiration date occurs. It is much better to use the log transfer feature within the ESKM that can be set up to transmit logs externally once they are closed. Using the log signing feature, combined with the SCP protocol for transmitting the files externally, will protect the logs once they leave the ESKM. Digitally signing the logs ensures that each log can be validated, which could be important for compliance reasons.

NOTE: When logs are exported or archived, the log signing certificate must also be exported or archived with them. To view the logs, both the log and the certificate are required. For more information, see the log configuration procedures section of the HPE Enterprise Secure Key Manager User Guide.

Backup Strategies

While clustering ESKM nodes is an effective way of exchanging keys and configuration data to allow for single node failures, it is by no means the complete solution for protecting the ESKM environment. Performing regular backups of the ESKM nodes is essential to ensure that your encryption key solution is protected in a disaster-recovery scenario. In the event that a node becomes out-of-sync with the rest of the cluster due to a connectivity failure, using the backup utility is critical to being able to distribute the un-replicated keys to the other cluster nodes.

Because of the out-of-sync possibility listed above, it is necessary to back up each ESKM node, even in a clustered environment. Since this could affect several nodes, some of which might be in offsite locations, it is best to develop a way to automate those backups to make administering the ESKMs easier.

The ESKM provides three ways of backing up the keys and configuration:

1. Internal backup stored on the ESKM.
2. Backup by downloading the data via browser (this encrypts and saves the data to the local computer via the browser interface).
3. Backup to an external server using SCP (secure file transfer) to copy the backup file.

There are advantages and disadvantages to each method. Backing up internally to the ESKM is the quickest way of running a backup, but provides no disaster-recovery protection. Backup via browser provides disaster-recovery protection since the data is stored outside the ESKM and is OS independent (because the browser handles the transfer). Backing up to an external server via SCP provides both disaster-recovery protection and the ability to be automated, but SCP is an older secure protocol and, if the desire is to send the data to a Windows server, requires additional software as SCP is not a recognized protocol on Windows. SCP still works to secure the backup data, however, and so this method is the preferred solution for backing up the ESKM.

To perform an SCP backup to an external Unix/Linux server, a user account needs to be set up on the external server so that SCP has a way to log in. Set up the account and test it by attempting to log in to the server. Once that is complete, run a test backup from the ESKM GUI to the Unix/Linux server. Enter the login and password for the server user account and select the user account's home directory as the backup repository. Putting it elsewhere on the server can cause transfer failures due to insufficient permissions, so the home directory is almost always the best choice

NOTE: Do not use the server's root account—it is more secure not to log into any Unix or Linux server remotely as root, and in many cases the system is set up not to allow it.

If a Windows server is the desired destination for the backup repository, some additional software is required to make SCP available. An open-source program called Cygwin (downloadable from **cygwin.com**) creates a Linux-like environment on a Windows 2000 or 2003 server. Traditional Linux commands can be run from a command-line, such as the `scp` command we need for our backups.

To set up Cygwin on Windows:

1. Download Cygwin from **cygwin.com**.
2. Run **setup.exe**, which downloads files from the Internet. If this is not available, the files can be downloaded to a repository and then setup can be pointed to that directory.

3. Install the default packages first. This installs a base package, to which additional packages can be added without installing the full installation set.
4. After the base package is installed, the following packages need to be added to be able to receive SCP-transmitted backup files. Some of these packages are also necessary for setting up an automated backup script, which will be detailed later: cron (found in the Admin section) cygrunsrv (found in the Admin section) expect (found in the Interpreters section) OpenSSH (found in the Net section) vim, or your favorite text editor (found in the System section)
5. After installing these packages, the Cygwin environment only has to be provided a user account with which the ESKM will connect and deposit the backup files. First, a user account and group name should be set up in Windows using the Local Users and Groups section of Computer Management.
6. The Cygwin shell can be started now by double-clicking the Cygwin icon (if you allowed the defaults during setup that put the icon on the Desktop). If not, follow the **Start/Programs** path to the **Cygwin Bash Shell** and run it that way.
7. Perform the following steps to set up Cygwin to recognize the user account and group:
   a. Move to Cygwin etc directory (default is **c:\cygwin\etc**) **mkgroup -l -g {local group} > group**—appends new group into Cygwin group file **mkpasswd -l -u {local user account} > passwd**—appends new user account into Cygwin passwd file
   b. Create a subdirectory in the **C:\cygwin\home** directory to serve as the new user account's home directory. The home directory automatically created by the `mkpasswd` command should default to **/home/{user account name}**, which translates in Windows as **C:\cygwin\home\{user account name}**.
8. Now, SSHD must be set up as a service to allow the Windows server to receive SCP connections. From the Cygwin bash prompt, run `ssh-host-config` and provide the following answers to the script's questions:
   a. Answer **yes** to enabling privilege separation
   b. Answer **yes** to "create local user sshd"
   c. Answer **yes** to "install sshd as a service"
   d. Answer **ntsec binmode tty** to set the environment variables for 'CYGWIN='
   e. Start the SSHD service by entering `net start sshd` from the Cygwin bash prompt.

   This gives the Windows server the ability to communicate with the ESKM using SCP (SSHD also supports SCP). If your server is running Windows firewall or another firewall program, be sure to enable port 22 for SCP traffic.
9. Run a test backup from the ESKM GUI, using the Windows user account and listing the repository set up in the user account's home directory.

NOTE: The instructions may require some tweaking depending on the server, operating system, and Cygwin version being used. It may be simpler to avoid installing Cygwin and install a Linux machine instead.

Scheduled Backups

You can now use the Schedule Backup feature to define the backup schedule.

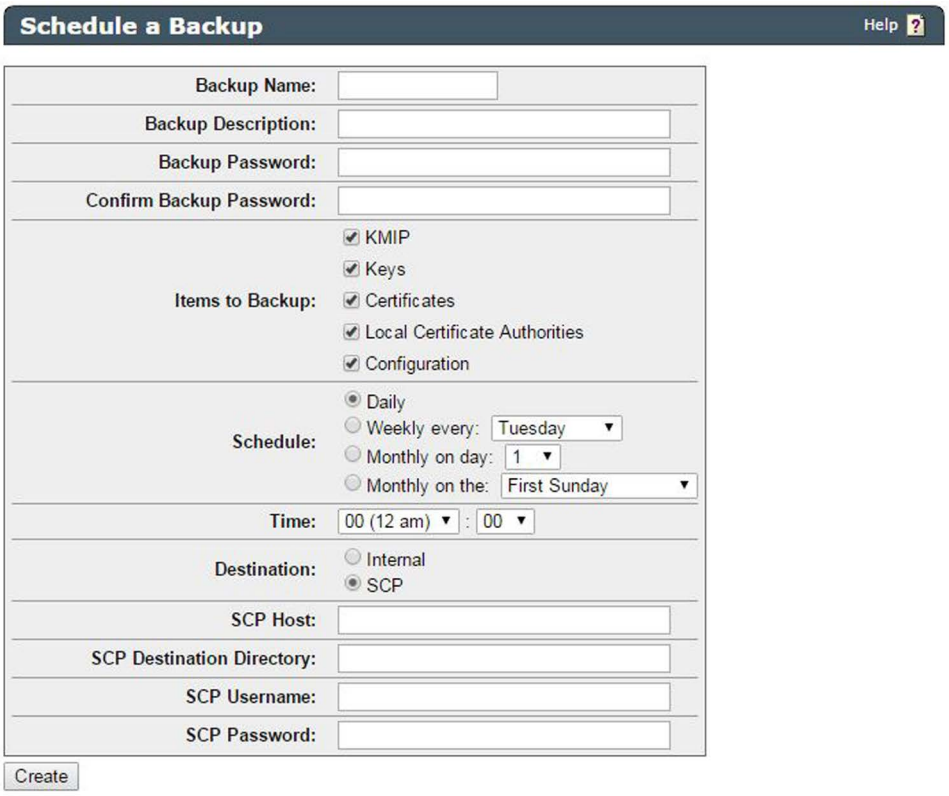

Figure 10: Schedule a Backup

By scheduling frequent backups, an ESKM can be adequately protected with multiple restore points. As ESKM backup jobs take very little time to run, even with hundreds of keys, the jobs can be run multiple times a day to make sure that added keys and configuration changes are protected and available for restore, if necessary. Keep in mind that each ESKM node, even in a clustered configuration, needs to be backed up, so multiple versions of this script will be needed to accommodate a multi-ESKM node environment.

See the Schedule Backup section in the **HPE Enterprise Secure Key Manager Users Guide** for additional guidance.

IMPORTANT: If the encryption keys maintained by the ESKM are not protected via backup and the keys are lost due to a disaster or some other reason, the data encrypted with those keys is irretrievable—by the user, by HPE, or by any other third party. It is vitally important to ensure that the ESKM keys and configuration are properly protected.

Protecting the ESKM backup server

With the exported SCP backups now residing on the external server, the next issue is this: How is the external server itself protected? It could be included in the same backup policy as any other server within the backup application; however, consideration must be given to what might happen if the external server were to be lost along with the ESKM nodes in a disaster. Two options are possible:

- One option is to locate the external server in another site and back up the ESKM nodes remotely to the other site. The external server itself could be backed up across the WAN link, but it might be better just to provide a backup device at the remote site. This would provide protection against a single-site failure, as the ESKM data could be restored across the WAN link or at the remote site, if the ESKM environment must be rebuilt there.
- Another option is to create a script that periodically transfers the ESKM backup files from the external server to another server in a remote site. This would provide the same offsite protection as the first option, without necessarily requiring the setup of a backup device at the remote site. The downside is that, without a backup application providing retry capabilities, logging information, and so on, a file transfer failure to the remote site might more easily go unnoticed.

Clustering

HPE recommends that customers take advantage of the clustering functionality within the ESKM to aid with administration and to provide failover capability. It is possible to set up the cluster configuration so that nodes exist in different locations, to provide some assistance in a disaster-recovery situation. Since the only requirement is that the nodes be able to reach each other through TCP/IP, this can be done fairly easily.

The exchange of configuration and key changes in an ESKM cluster happens automatically and immediately—if a key is added or a configuration change is made, the other nodes immediately receive the changes. If cluster nodes are placed in different locations, especially longer-distance locations, the usual limitations with long-distance replication still apply; however, even if an interruption in connectivity between cluster nodes occurs, change replication will occur as long as the connectivity is re-established

NOTE: Twenty-four hour replication retry is present in the HPE Enterprise Secure Key Manager.

In the event that a connectivity issue prevents replication of ESKM changes for longer than 24 hours, there is a procedure that can return everything to normal across all the cluster nodes.

Note that while there is a feature button on the ESKM in the cluster configuration section of the ESKM GUI called **Synchronize With** that pulls the key and configuration information from another cluster node, it is highly recommended that this feature not be used to resync a cluster. This procedure will overwrite the keys and most of the configuration information with what it finds on the other node. This could potentially result in a more out-of-sync situation than before, resulting in a loss of keys. Instead, follow these steps to resync cluster nodes:

1. Stop all client activity to prevent any further key additions, or wait until all client activities have completed.
2. Run full backups on the out-of-sync node and one of the other cluster nodes. Save the backups to an external server that all nodes can reach.
3. Restore ONLY the keys from the other cluster node to the out-of-sync node, if the keys are the difference. Key restoration is an **append** procedure, rather than an **overwrite** procedure.
4. If any keys were added to the out-of-sync node, restore ONLY the keys from the out-of-sync node to the other cluster nodes. All the keys should now be back in sync.
5. If changes were made in configuration, such as adding a new user or deselecting a check box, these changes will have to be duplicated manually on the out-of-sync node.

The procedure for adding a new node to an existing cluster is detailed in the HPE Enterprise Secure Key Manager users guide. However, there might be a situation in which an ESKM that has already been managing keys for a specific client to join an existing cluster setup. To do that, follow these steps:

1. Stop all client activity to prevent any further key additions, or wait until all client activities have completed.
2. Run full backups on the new node and the primary cluster node. Save the backups to an external server.
3. Download the cluster key to an external server.
4. Join the new node to the cluster. This will overwrite all key and configuration information on the new node.
5. Restore ONLY the keys from the new node backup to all the cluster nodes.
6. Manually add any user/group information for the new node library on one of the cluster nodes. User/group info will automatically be replicated on the other cluster nodes. Do not forget to add the key generation policy.
7. Restore the keys from the new node backup to the new node to restore its own keys.
8. Create a new server certificate on the new node. The name of the server certificate must match the name specified on the primary node. Make sure that the certificate request is signed by the same local CA as on the primary node.
9. Update the client environment to include the IP address of this new cluster node.

Conclusion

The HPE Enterprise Secure Key Manager environment, when configured using the suggestions in this white paper, will provide an effective and secure repository for your encryption keys environment. As with any computer environment, monitoring diligence and proper maintenance are critical to its success. Monitoring logs, running frequent system backups, and intelligent user management of the ESKM environment are all necessary elements and should be fundamental parts of any overall backup strategy.

For more information

For more information on the HPE ESKM visit **hpe.com/software/ESKM**.

Hewlett Packard Enterprise

4AA2-1403ENN, April 2016, Rev. 4