YubiHSM 2 with Key Storage Provider for Windows Server

Yubico

Jul 08, 2022

CONTENTS

1 Introduction **1**
- 1.1 Configure YubiHSM 2 Key Storage Provider (KSP) for Microsoft Windows Server 1
- 1.2 About the YubiHSM Software 1

2 Prerequisites and Preparations **3**

3 Key Splitting and Key Custodians **5**

4 Install the YubiHSM 2 Tools and Software **7**
- 4.1 About the YubiHSM Software 8
- 4.2 Default YubiHSM 2 Default Device Configuration 8

5 Configure the Primary YubiHSM 2 Device **11**
- 5.1 Overview of Procedure 12
- 5.2 Procedure: Configuring the Primary YubiHSM 2 13

6 Verify the YubiHSM 2 Setup **17**

7 Configure the YubiHSM 2 Software **19**
- 7.1 Configure the KSP Settings in the Windows Registry 19
- 7.2 Configure the YubiHSM 2 Connector Service 21

8 Back Up and Restore Key Material **23**
- 8.1 Back Up the YubiHSM 2 23
- 8.2 Restore Keys on the Secondary YubiHSM 2 Device 24
- 8.3 Verify the Duplicated YubiHSM 2 26

9 Getting Help **29**

10 Terminology **31**
- 10.1 Software 33

11 Copyright **35**

CHAPTER

ONE

INTRODUCTION

1.1 Configure YubiHSM 2 Key Storage Provider (KSP) for Microsoft Windows Server

This guide is intended to help systems administrators deploy YubiHSM 2 for use in a Windows server environment. The expected outcome is that the YubiHSM 2 is installed and configured with authentication keys, audit keys, and wrap keys. This guide also explains how to make backups and restore keys on a YubiHSM 2.

These guidelines for deployment cover basic topics, so the instructions should be modified as required for your specific environment. It is assumed that you are familiar with the concepts and processes for working with Microsoft Windows Server. It is also assumed that the installation is performed on a single Microsoft Windows Server, but the concept can be extended to more servers.

Important: Before deploying to production, we recommend that you use this guide for installing and testing the setup of the YubiHSM 2 with the Microsoft Windows Server installation in a test or lab environment.

1.2 About the YubiHSM Software

The following YubiHSM 2 software is used in this guide. These items are included as part of the archive file you download from the YubiHSM 2 libraries and tools page.

Software	Purpose
YubiHSM Connector	Enables communication between the YubiHSM 2 and applications that use it. We recommend that the YubiHSM Connector run on the host operating system if the calling application is deployed to a VM. The Connector must always be running.
YubiHSM Shell	The administrative command line tool used to interact with and configure the YubiHSM 2 device. If the YubiHSM Shell is installed on a VM, it will connect to the Connector over a networked connection.
YubiHSM Setup	Helps with setting up a device for specific use cases. Currently supports setting up for use with Microsoft Windows KSP.
YubiHSM Key Storage Provider (KSP)	Acts like a driver for the YubiHSM 2 device on Windows and enables it to work with applications that leverage Microsoft's Cryptographic API Next Generation (CNG). Examples of calling applications are Microsoft Certificate Services or Microsoft SQL Server Always Encrypted.

CHAPTER

TWO

PREREQUISITES AND PREPARATIONS

The audience of this document is an experienced systems administrator with a good understanding of Microsoft Windows Server management. In addition, it is helpful to be familiar with the terminology, software, and tools specific to YubiHSM 2. As a primer for these, refer to *Terminology*.

In order to follow the steps provided in this guide, the following prerequisites must be met:

- Access to Microsoft Windows Server 2012 SP2 or higher, installed in a secure computer network. The system administrator must have elevated system privileges.
- The YubiHSM 2 SDK downloaded from the Yubico YubiHSM 2 Release page and available on the system to be used. Installation instructions are given in the following.
- Two (2) YubiHSM 2 devices, one for deployment and one for backup in hardware.
- Key custodians, if your organization policies require them for the YubiHSM 2 deployment. For more information about key custodians and the associated `M of N` key shares, see *Key Splitting and Key Custodians* below.

Important: Although it is possible to configure the YubiHSM 2 on a networked machine, to safeguard its integrity, it is recommended that its configuration be performed on a fresh system in an air-gapped environment, i.e., the steps in this guide should be performed on a stand-alone computer with both Windows Server 2012 SP2 or higher and the YubiHSM 2 software installed. And we recommend that you do not store keys - even under wrap - on network-accessible or otherwise compromisable storage media.

CHAPTER

THREE

KEY SPLITTING AND KEY CUSTODIANS

The preferred method for backing up the YubiHSM 2 keys calls for key splitting and restoring or regenerating, often referred to as setting up an `M of n` scheme (Shamir's Secret Sharing (SSS). This process ensures no individual can export key material from the YubiHSM 2, and provides a way to control the import of key material that has been exported under wrap from one device into other devices. For example, you would export and import objects for backup purposes, as described in *Back Up and Restore Key Material*.

The key that is split among a predetermined number (`n`) of `key custodians` (also known as key shareholders) is known as the wrap key. Each custodian receives their own unique share. In order to use the key, a minimum number of shares (`m`) must be present so that the key can be regenerated (sometimes called "rejoined"). This minimum number of custodians is called the `privacy threshold`. If this threshold is not attained, the wrap key cannot be regenerated. This minimum number, `n`, should be larger than one.

The exact number of key shares and the privacy threshold are determined by the requirements of your organization. If your organization has policies in place that define how this procedure should be performed, be sure you know these policies before proceeding. You should also have a predetermined practice in place specifying both:

- How the key shares must be recorded (written on paper, photographed, locally printed, or some other means) and
- How they must be stored between uses (for example, offsite archive, safety deposit box, sealed envelope).

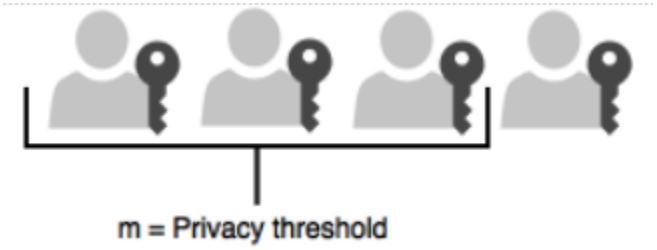

Figure - Privacy threshold

The YubiHSM Setup Tool enables you to perform the key splitting and assigning of shares to key custodians. To carry out the setup process, you need to know who the wrap key custodians will be. During setup, all key custodians must be physically present to record their shares. Exact instructions for key splitting and assigning of shares are given in *Configure the Primary YubiHSM 2 Device*.

CHAPTER

FOUR

INSTALL THE YUBIHSM 2 TOOLS AND SOFTWARE

To complete the procedures in this guide, install the YubiHSM 2 tools and software that will be needed for this.

Tip: A generic prompt, `*$*`, is used in command line examples in this document. Depending on your command line application, your prompt may be different.

Step 1
Unzip the downloaded archives of the SDK containing the YubiHSM libraries and tools and move the contents to an appropriate location.

Step 2
On your Windows system, run both installers:

- `yubihsm-cngprovider-windows-amd64.msi` (YubiHSM Key Storage Provider)
- `yubihsm-connector-windows-amd64.msi` (YubiHSM Connector for Windows)

Step 3
Set the ADCS service dependency for the YubiHSM Connector service via an elevated/admin Windows Command Prompt. This prevents an error which occurs if the ADCS services starts before the YubiHSM connector is running.

a. List the current dependencies with `sc qc "certsvc"`

```
> sc qc "certsvc"
[SC] QueryServiceConfig SUCCESS

SERVICE_NAME: certsvc
   TYPE               : 110  WIN32_OWN_PROCESS (interactive)
   START_TYPE         : 2   AUTO_START
   ERROR_CONTROL      : 1   NORMAL
   BINARY_PATH_NAME   : C:\Windows\system32\certsrv.exe
   LOAD_ORDER_GROUP   :
   TAG                : 0
   DISPLAY_NAME       : Active Directory Certificate Services
   DEPENDENCIES       :
   SERVICE_START_NAME : localSystem
```

b. Add the YubiHSM connector dependency to ADCS with the command: `sc config "certsvc" depend="yhconsrv"`

```
> sc config "certsvc" depend="yhconsrv"
[SC] ChangeServiceConfig SUCCESS
```

c. Once the command is entered, the dependency can be verified with `sc qc "certsvc"`

```
[SC] QueryServiceConfig SUCCESS

SERVICE_NAME: certsvc
   TYPE               : 110  WIN32_OWN_PROCESS (interactive)
   START_TYPE         : 2   AUTO_START
   ERROR_CONTROL      : 1   NORMAL
   BINARY_PATH_NAME   : C:\Windows\system32\certsrv.exe
   LOAD_ORDER_GROUP   :
   TAG                : 0
   DISPLAY_NAME       : Active Directory Certificate Services
   DEPENDENCIES       : yhconsrv
   SERVICE_START_NAME : localSystem
```

d. To remove dependencies for ACDS, use the same command for adding depedencies with a blank depend field: `sc config "certsvc" depend=""`

4.1 About the YubiHSM Software

The following YubiHSM pieces of software are used in this guide. They are included as part of the archive file you downloaded from the Yubico Knowledge Base.

4.2 Default YubiHSM 2 Default Device Configuration

The YubiHSM 2 device comes with a single factory-installed authentication key whose default password is `password`. As part of the configuration in this guide, this default authentication key will be destroyed. If the YubiHSM 2 is reset to its default configuration, any non factory-installed objects stored on it are also destroyed. Reset instructions can be found in Factory Reset.

To ensure that the YubiHSM 2 has not been tampered with, verify that it still has the default configuration by following the steps below:

Step 1

Do one of the following:

- If the application that calls the YubiHSM Connector is **running on a local host**, start the Connector with the command `yubihsm-connector` without additional parameters. In Windows Server 2012 SP2 or higher, `yubihsm-connector.exe` is located in `C:\Program Files\YubiHSM Connector\`.
- If the application is **running on a VM or a different server**, start the YubiHSM Connector on the host operating system in networking mode. For example, if the host machine's IP address is `192.168.100.252`, launch the Connector on the host OS with the command `yubihsm-connector -l 192.168.100.252:12345`.

Tip: For testing or debugging the YubiHSM Connector, the flag `-d` can be set.

Step 2

To gain shell access to the YubiHSM 2, launch the YubiHSM Shell program by opening a Command Prompt and running the command `yubihsm-shell`. If a networked Connector is used, set the parameter `--connect <connector URL>`, for example:

```
$ yubihsm-shell --connector http://192.168.100.252:12345
```

Tip: For testing or debugging the YubiHSM Shell, the flag `-d` can be set.

Step 3

To connect to the YubiHSM 2, at the `yubihsm` command line, type `connect`. A message saying that you have a successful connection is displayed.

Step 4

To open a session with the YubiHSM 2, type `session open 1` (where 1 is the ID of the default authentication key pre-installed on the device).

Step 5

Type in the default password: `password`. A message confirming that the session has been set up successfully is displayed.

Step 6

You now have an administrative connection to the YubiHSM 2 and you can list the objects available by typing `list objects 0` and pressing **Enter**. Your results should be similar to the following:

```
id: 0x0001, type: authentication-key, sequence: 0
```

Step 7

To exit, type `quit`.

CHAPTER

FIVE

CONFIGURE THE PRIMARY YUBIHSM 2 DEVICE

The YubiHSM Setup program is used to perform the initial configuration of the primary YubiHSM 2 device so that the necessary key material is generated on it:

- One wrap key to be split as described in *Key Splitting and Key Custodians*.
- One application authentication key for authenticating to the YubiHSM 2 through the KSP. This allows the KSP to perform operations in the YubiHSM 2.

Note: This initial configuration replaces the default authentication key with a new one, which will only be operable in the same domain as the asymmetric key. The domain concept that is used to compartmentalize the YubiHSM 2 determines this behavior.

Tip: For test purposes you can set the `yubihsm-setup -d` flag to keep the default authentication-key with the administrative privileges; this will allow you to delete keys on the YubiHSM 2 for test purposes only. For production purposes, however, the `yubihsm-setup` command must be executed without the `-d` flag to ensure that the factory preset authentication key is properly deleted from the YubiHSM 2 device.

- One audit key for accessing the internal audit log of the device and resetting the audit log. The audit log retains information about the last 62 operations. It is also used to purge the log if needed. Depending on your local requirements, you may not need to create an audit key. If you are unsure of your requirements, we suggest you create an audit key.

The authentication key and the audit key are exported under wrap to a file in the current working directory on the machine where the YubiHSM Setup program is installed.

Tip: The YubiHSM Setup tool has a help argument that you can call to learn more about its usage.

5.1 Overview of Procedure

The configuration steps to be performed upon inserting the primary factory preset YubiHSM 2 device into the air-gapped system are set out briefly below. These steps are described in detail in the subsequent "Procedure: Configuring the Primary YubiHSM 2."

Step 1
Set up communication between the YubiHSM 2 tools and the device.

Step 2
Run the YubiHSM Setup with the argument `ksp`, specifying the Connector URL if necessary.

Step 3
Start the YubiHSM Setup process and authenticate to the YubiHSM 2 device.

Step 4
Add RSA decryption capabilities if required. If you plan to use your YubiHSM 2 exclusively with an application that only needs signing capabilities, RSA decryption is not required. Active Directory Certificate Services (ADCS), for example, does not require RSA decryption. However, if you are planning on using the same YubiHSM 2 device for something that does require the capability to decrypt RSA, then you do need RSA decryption. The Microsoft SQL Server Always Encrypted, for example, needs RSA decryption capabilities.

Step 5
Enter the name of the domain in which you need the application authentication key and audit key to be available.

Step 6
Generate the **wrap key** and its ID.

Step 7
Split the wrap key into shares and specify the privacy threshold.

Step 8
Have the wrap key custodians record their shares.

Step 9
Create the **application authentication key**.

Step 10
Create the password for the application authentication key.

Step 11
Create the **audit key** (optional).

Original default authentication key is deleted and setup process finishes.

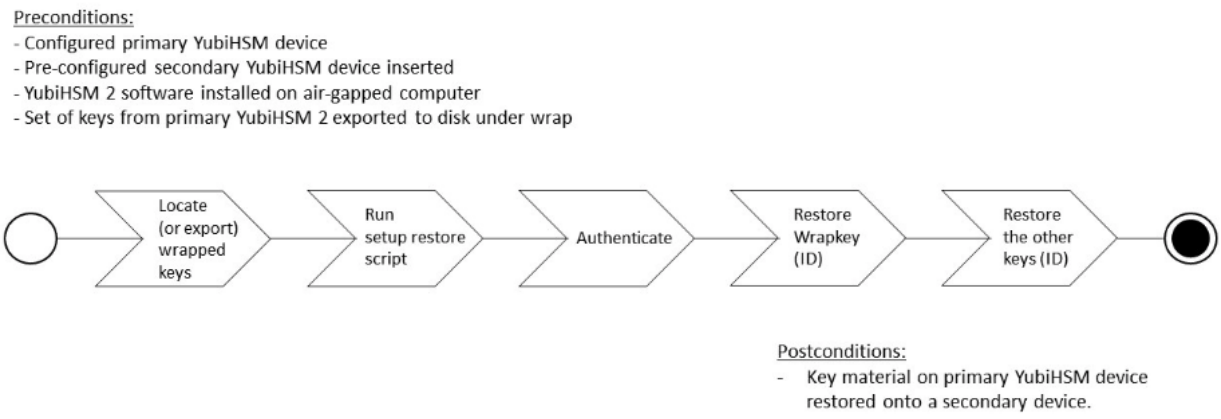

Figure - Flowchart illustrating the YubiHSM 2 setup for Windows

5.2 Procedure: Configuring the Primary YubiHSM 2

Step 1

Enable communication with the YubiHSM 2 device by running the YubiHSM Connector on the system where the device is inserted. If the YubiHSM Connector is running on a host machine to which the YubiHSM 2 is physically connected, the Connector should be started in networked mode. For example, if the host IP address is `192.168.100.252`, the Connector should be started on the host machine with the following command:

```
yubihsm-connector -l 192.168.100.252:12345
```

In this scenario, you can verify that the Connector is running properly by typing the following URL into your web browser:

http://192.168.100.252:12345/connector/status

The output in the web browser should be similar to:

```
status=OK
serial=*
version=1.0.0
pid=*
address=192.168.100.252
port=12345
```

Step 2

In the Command Prompt, navigate to a directory for which you have write access and run the YubiHSM Setup with the argument `ksp`.

```
yubihsm-setup ksp
```

If the application calling it is installed on a machine other than the YubiHSM Connector, use the `connector` flag to specify the Connector URL, for example:

```
yubihsm-setup --connector http://192.168.100.252:12345 ksp
```

Step 3

To start the YubiHSM Setup process, type the default authentication key password `password`. A

message confirms that the default authentication key was used and that you are authenticated to the device:

```
Using authentication key 0x0001
```

Object IDs are displayed in the YubiHSM Setup Tool using hexadecimal numbers. In this case the default authentication key is ID 1 (or `0x0001` in hexadecimal format).

Step 4

You are prompted to add RSA decryption capabilities. Enter `y` or `n`.

Tip: If you are unsure what selection to make, select no (`n`).

Step 5

You are prompted for the domain(s) you need the keys to be available in. Unless you have a requirement to assign more than one domain, enter a number between 1 and 16. In this guide, we assume that domain 1 was entered. The confirmation will look like one of the following:

```
got domains [
One
]
```

or

```
Using domains:
One
Enter wrap keyID (0 to choose automatically):
```

Step 6

You are prompted to generate a wrap key and enter its ID. Do one of the following:

- To manually assign a wrap key ID, type a number for the ID. As object ID 1 is already in use by the default application authentication key, we recommend assigning ID 2 to the wrap key.
- To allow the system to generate a wrap key ID automatically, type `0`.

In both cases, a confirmation message like the following is displayed:

```
Stored wrap key with ID 0x0002 on the device
```

Step 7

You are prompted to specify the number of shares into which the wrap key should be split in order to be distributed to an equal number of key custodians. You are also prompted to specify the privacy threshold, which is the number of shares that must be present for the wrap key to be regenerated. For this example, we assume that the wrap key is split into three shares, of which at least two shares must be present in order to regenerate the key.

Note: For an overview of key custodian activities, see *Key Splitting and Key Custodians*.

Tip: For test purposes, such as in a lab scenario where wrap key sharing is not crucial, it is not necessary to specify that the wrap key should be split between key custodians. Instead, you can use a single key. To do this, when configuring the device using YubiHSM Setup, indicate the number of shares to be 1 and the privacy threshold to be 1 as well.

When prompted, do the following:

a. Enter the number of shares. In this example, enter `3`.

b. Enter the privacy threshold. In this example, enter `2`.

The wrap key thereby generated is saved to the HSM 2 device.

Step 8

When the relevant prompt is displayed, each of the three wrap key custodians should take their turn in front of the screen to record their share.

Important: Each custodian must record the whole string presented, including the prefix (in the following example, `2-1-`) which indicates the number of shares required to regenerate the key (the privacy threshold) and the number identifying where in the sequence the share was created.

The following is an example of a share presented on the screen:

```
2-1-WWmTQj5PHGJQ4H9Y2ouURm8m75QkDOeYzFzOX1VyMpAOeF3YKYZyA...
Have you recorded the key share? (y/n)
```

A notice is displayed, warning that the shares are not stored anywhere.

a. To start having the custodians record the key shares, press **Enter**.

b. The first custodian records his or her share and confirms that the share was recorded by pressing `y`. The screen buffer is cleared before the next share is presented.

c. The next custodian records the key share for the second share, confirms it, and so on.

Step 9

You are prompted to create an **application authentication key**. Since object IDs `1` and `2` are already in use by the default authentication key and the wrap key respectively, the example in this guide assumes that you enter ID `3` for the application authentication key. To allow the system to generate a wrap key ID automatically, type `0`.

Step 10

Create and enter a password of at least eight (8) characters for the application authentication key. Store it so that it cannot be compromised. You will need this password later to configure the YubiHSM KSP DLL, as described in *Configure the YubiHSM 2 Software*. A confirmation message like the following appears:

```
Stored application authentication key with ID 0x0003 on the device
Saved wrapped application authentication key to 0x0003.yhw.
```

The wrapped application authentication key (in this example, `0x0003.yhw`) is exported to the current working directory. Although the keys are encrypted using the wrap key, we recommend that you do not store keys - even under wrap - on network-accessible or any storage media that could be compromised. However, if you will be making a backup (and you should), leave the `*.yhw-file` with the wrapped authentication key where it was saved for now, deleting it **AFTER** you have made the backup.

Step 11

Decide whether to create an **audit authentication key**. To log into the YubiHSM 2 with the audit authentication key, both the key ID and the password will be needed.

a. When prompted to create an audit key, type `y`.

b. When prompted, assign a key ID to the audit key. Make a note of the ID you enter (for example, key ID 4).

c. When prompted, enter the audit key password. Store this password so that it cannot be compromised.

The audit key is exported under wrap to the current working directory. Using our example of key ID 4, the file will be named `0x0004.yhw`.

Step 12

The setup tool (in default mode) finishes by letting you know that the default authentication key has been deleted.

```
Previous authentication key 0x0001 deleted
All done
```

The YubiHSM Setup application exits. The YubiHSM 2 device is now equipped with the symmetric keys for wrap, audit, and application authentication.

CHAPTER

SIX

VERIFY THE YUBIHSM 2 SETUP

Verify the results of the YubiHSM Setup program using the YubiHSM Shell program. Log in using the application authentication key.

Step 1

To verify the YubiHSM 2 setup, in your Command Prompt, run the following command:

```
$ yubihsm-shell
```

If the YubiHSM Connector is running on a host machine to which the YubiHSM 2 is physically connected, start the YubiHSM Shell program in networked mode. For example, if the host server's IP address is `192.168.100.252`, start the YubiHSM Shell program at the VM with the following command:

```
$ yubihsm-shell --connector http://192.168.100.252:12345
```

Step 2

To connect to the YubiHSM 2, at the `yubihsm` prompt, type `connect`. A message verifying that you have a successful connection is displayed.

Step 3

To open a session with the YubiHSM 2, type `session open 3`.

Step 4

Type in the password for the application authentication key. You will receive a confirmation message that `session 0` has been set up successfully.

Step 5

You now have an administrative connection to the YubiHSM 2. You can list the objects available by typing list objects 0. Your results should be similar to the following:

```
Found 3 object(s)
id: 0x0002, type: wrap-key, sequence: 0
id: 0x0003, type: authentication-key, sequence: 0
id: 0x0004, type: authentication-key, sequence: 0
```

As you can see by looking at their IDs, these objects correspond to the wrap key, the application authentication key and the audit key that were just created.

Step 6

To obtain more information about any of the objects and its capabilities — for example, the application authentication key (object ID 3) — run the `objectinfo` command with the appropriate ID format, for example:

```
yubihsm> get objectinfo 0 3 authentication-key
```

The response you receive should look similar to the following:

```
id: 0x0003, type: authentication-key, algorithm:
aes128-yubico-authentication, label: "Application auth key",
length: 40, domains: 1, sequence: 0, origin: imported,
capabilities: exportable-under-wrap:generate-asymmetric-key:
sign-attestation-certificate:sign-pkcs:sign-pss:sign-ecdsa,
delegated_capabilities:exportable-under-wrap:
generate-asymmetric-key:sign-attestation-certificate:sign-pkcs:
sign-pss:sign-ecdsa
```

This indicates that YubiHSM 2 has now been configured to:

- Generate asymmetric objects
- Compute signatures using RSA-PKCS1v1.5
- Compute signatures using RSA-PSS
- Export other objects under wrap
- Import wrapped objects
- Mark an object as exportable under wrap

In addition, this object (the application authentication key, object ID 3) also has delegated capabilities that can be bestowed on other objects that it creates. For more information on delegated capabilities, see Capability.

Step 7

To exit, type `quit`.

CHAPTER

SEVEN

CONFIGURE THE YUBIHSM 2 SOFTWARE

Before using the YubiHSM 2 on Windows, there are two YubiHSM 2 software components to be configured:

- The YubiHSM 2 KSP.
- The YubiHSM 2 Connector service.

The configuration steps are described in the sections below.

Important: Make a backup of your Windows Registry before you make any changes.

7.1 Configure the KSP Settings in the Windows Registry

To enable Microsoft Cryptographic API Next Generation (CNG) to access the YubiHSM 2 KSP, the following registry entries must be changed from their default values. The YubiHSM 64-bit KSP subkey and the YubiHSM 32-bit KSP subkey were created during the YubiHSM SDK installation:

```
HKEY_LOCAL_MACHINE\SOFTWARE\Yubico\YubiHSM
```

The edits to be made produce a result like the one illustrated below:

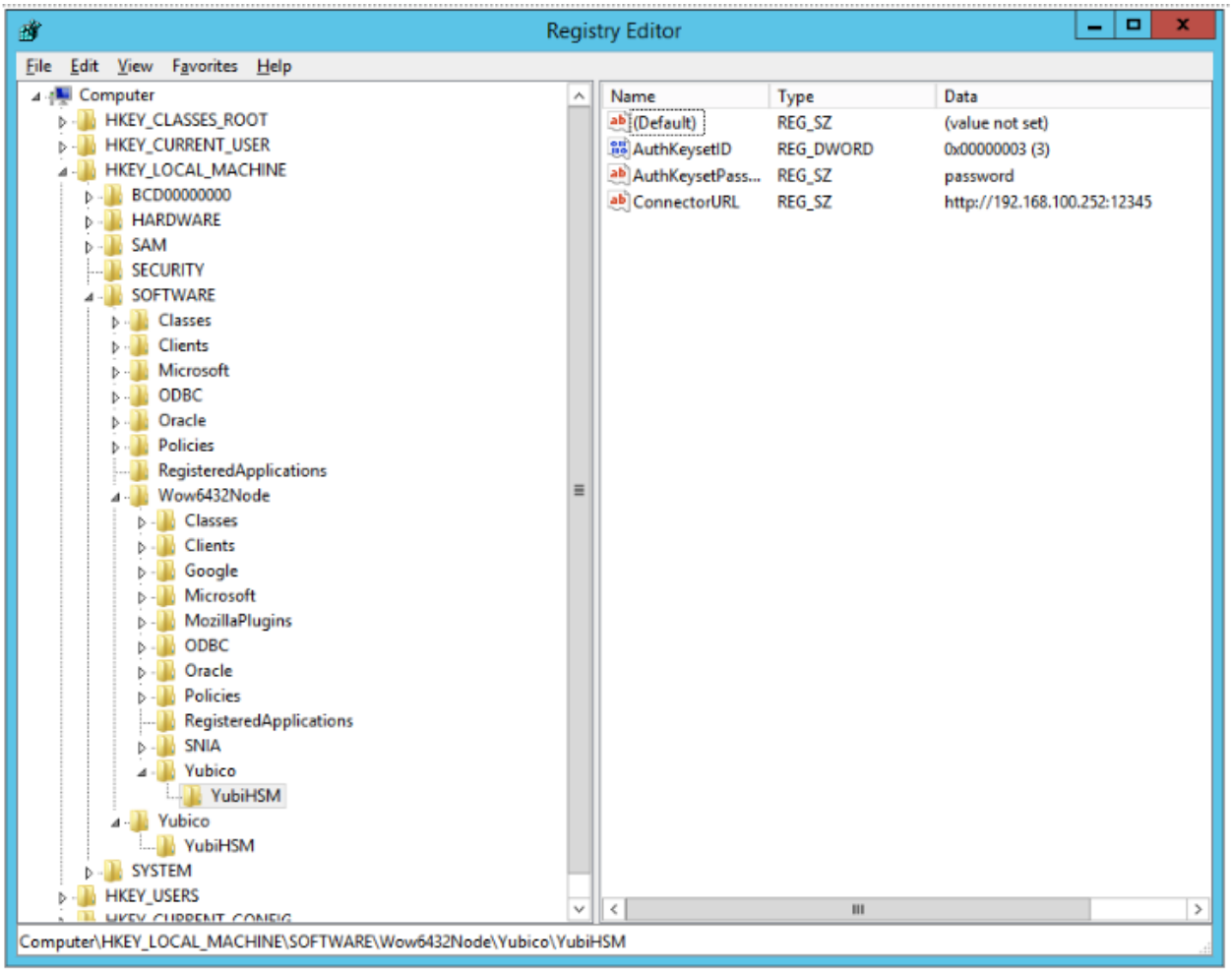

Figure - Registry settings for the YubiHSM 2 KSP

Step 1

Click **Start > Run**, type `regedit` in the Run dialog box, and click **OK**.

Step 2

Select the registry subkey for the **YubiHSM 64-bit KSP**.

```
HKEY_LOCAL_MACHINE\SOFTWARE\Yubico\YubiHSM.
```

Step 3

Change the URI to the IP address and port on which the YubiHSM 2 Connector is listening by editing the following registry entry appropriately, for example:

```
“ConnectorURL”=http://127.0.0.1:12345
```

If the Connector is listening on IP address and port `192.168.100.252:12345`, for example, the `ConnectorURL` value should be changed to:

```
“ConnectorURL”=http://192.168.100.252:12345
```

Step 4

Enter the ID of the application authentication key (object ID 3 was used as an example in this guide;

if you used another object ID be sure to enter that). For our example, because the hexadecimal value of `0x00000003` resolves to 3 in the Windows Registry, change the entry to:

```
"AuthKeysetID"=3
```

Step 5

The application authentication key password is stored in the registry for the KSP to use when authenticating to the device. Enter the new password that you created:

```
"AuthKeysetPassword"={password}
```

Step 6

Select the registry subkey for the `YubiHSM 32-bit KSP`.

```
HKEY_LOCAL_MACHINE\SOFTWARE\Wow6432Node\Yubico\YubiHSM
```

Then repeat steps 3-5 above.

Step 7

To save your changes, exit the Windows Registry.

7.2 Configure the YubiHSM 2 Connector Service

The YubiHSM Connector service reads the configuration file `yubihsm-connector-config.yaml`. Depending on your local setup, for instance if you are running multiple instances of the software on the same host, you may need to edit this configuration file to ensure it is consistent with the Windows Registry, i.e., that the parameters and their values are the same in the configuration file and in the Windows Registry.

On Windows, the `yubihsmconnector.config.yaml` file is located at `C:\programdata\yubiHSM\yubihsmconnector.yaml` - you will need administrator rights to modify the file.

CHAPTER

EIGHT

BACK UP AND RESTORE KEY MATERIAL

We strongly recommend making a backup copy of all production objects residing on your production devices, particularly once the asymmetric key has been generated on the YubiHSM 2. If there is a hardware failure of the production device, having a backup ensures that you can resume operations quickly. The backup process will result in two identical YubiHSM 2 devices with the same number of objects, keys, labels, etc.

Note: Making specific recommendations for governance of your critical key material is out of scope for this guide. Ensure you design these security procedures to meet the requirements of your organization - and then document the procedures. In many cases, they are subject to audits.

8.1 Back Up the YubiHSM 2

The procedure described in this section is appropriate for testing and for smaller installations. For larger and/or more complex installations such as:

- Those whose setup did **not** involve the YubiHSM Setup program
- When moving the YubiHSM 2 device from one server to another

Review the information at Backup and Restore to determine whether the procedures set out there are more appropriate for your situation.

This guide gives instructions for duplicating the following on the secondary device:

- Wrap key (previously created with ID `2`),
- Application authentication key (ID `3`),
- Audit key (ID `4`) (if created previously)

The objects were exported under wrap. The factory-installed authentication key (ID `1`) on the secondary HSM 2 device will be destroyed, just as it was on the primary HSM 2 device. If you use actual wrap key custodians (instead of just doing a proof of concept), you will need the custodians to provide their respective wrap key shares. In the example we used in this guide, 2 out of the 3 custodians/shares must be available.

The backup of the primary YubiHSM 2 should thus contain a duplicate of each of the objects stored on the primary device. Availability of these objects is ensured by the same application authentication key as the one used for the primary YubiHSM 2.

To guarantee integrity, perform these operations in an air-gapped environment.

8.1.1 Overview of Procedure

The backup procedure consists of the steps listed below the following diagram. The steps are explained in detail in the next section, *Restore Keys on the Secondary YubiHSM 2 Device*.

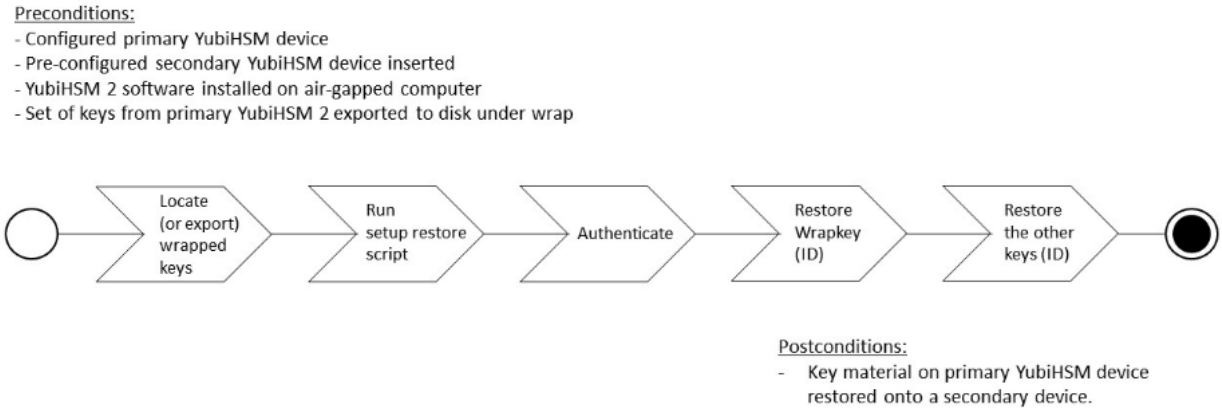

Figure - Flowchart illustrating backup and recovery of YubiHSM 2 keys

1. Locate the wrapped key material that was previously exported in "Procedure: Configuring Primary YubiHSM 2" in *Configure the Primary YubiHSM 2 Device*.
2. Set up communication between the YubiHSM 2 tools and the secondary (backup) YubiHSM 2 device.
3. Start the configuration process and authenticate to the secondary YubiHSM 2 device.
4. Restore the key material onto a secondary YubiHSM 2 device.

Tip: For **test purposes** you can set the `yubihsm-setup -d` flag to keep the default authentication-key with the administrative privileges; this will allow you to delete keys on the YubiHSM 2 for test purposes only. For **production purposes** however, the `yubihsm-setup` command must be executed without the `-d` flag to ensure that the factory preset authentication key is properly deleted on the YubiHSM 2.

8.2 Restore Keys on the Secondary YubiHSM 2 Device

Step 1

Verify that all the keys that were previously exported from the primary YubiHSM 2 under wrap are located in a directory to which you have read access.

If the necessary keys are not yet all available on disk, you can export the keys under wrap by running the following command:

```
yubihsm-setup dump
```

The YubiHSM Setup tool looks for files with the `.yhw` file extension in the current working directory and attempts to read and import them into the YubiHSM 2 device. The wrap key itself will be imported when the wrap key shares are provided to the tool.

For example, the following wrapped key files may be present:

- `0x0003-AuthenticationKey.yhw` (Application authentication key under wrap)
- `x0004-AuthenticationKey.yhw` (Audit authentication key under wrap)

- `x427a-Opaque.yhw` (Certificate under wrap - not referenced by this guide in the configuration of the primary HSM 2)
- `x427a-AsymmetricKey.yhw` (Private asymmetric key under wrap - not referenced by this guide in the configuration of the primary HSM 2)

Tip: If the initial authentication key (by default available as ID 0x0001) has been deleted, the new authentication application key will be identified with the flag `yubihsm-setup --authkey`. For example:

```
$ yubihsm-setup --authkey 0x0003 dump
```

Step 2

To begin the process of restoring the data onto the secondary YubiHSM 2, if the primary YubiHSM 2 device is still inserted into your computer, remove it and insert the secondary device.

Step 3

In the directory containing the `*.yhw` files, run `yubihsm-setup` with the `restore` argument:

```
$ yubihsm-setup restore
```

Step 4

To start the YubiHSM Setup process, type the default authentication key password, which is `password`. A confirmation message like the following is displayed, announcing that the default authentication key was used and that you are authenticated to the YubiHSM 2 device:

```
Using authentication key 0x0001
```

You will now start the restore procedure, by providing the minimum number of wrap key shares required by the privacy threshold defined when setting up the primary HSM 2.

Step 5

When prompted, type the number of shares required by the privacy threshold. In the example used in this guide, we specified that two shares are required to regenerate the key. These must be present in order to proceed.

Step 6

When prompted for share number 1, the wrap key custodian holding the first share inputs this information. A message confirms that the share was received:

```
Received share 2-1WWmTQj5PHGJQ4H9Y2ouURm8m75QkDOeYzFzOX1VyMpAO...
```

Step 7

Continue to have each wrap key custodian enter the share information for each of the wrap key shares required to regenerate the wrap key. Once a sufficient number of wrap key shares have been entered by the wrap key custodians, a final message is displayed, indicating that the wrap key from the primary HSM 2 is now on the secondary HSM 2 as well:

```
Stored wrap key with ID 0x0002 on the device
```

Note: The ID of the wrap key on the secondary device is the same as the ID of the wrap key on the primary device.

Step 8

Once the wrap key has been stored on the secondary HSM 2, the YubiHSM Setup program reads the files containing the application authentication key and, if applicable, the audit key, which were saved to file under wrap during the configuration of the primary device. The output below shows that in this case, the Certificate Authority (CA) root key was also generated and exported along with a private asymmetric key, both under wrap.

```
reading ./0x0004.yhw
Successfully imported object Authkey, with ID 0x0004
reading ./0x0003.yhw
Successfully imported object Authkey, with ID 0x0003
reading ./0x427a-AsymmetricKey.yhw
Successfully imported object Asymmetric, with ID 0x427a
reading ./0x427a-Opaque.yhw
Successfully imported object Opaque, with ID 0x427a
```

Step 9

If there are files containing wrapped objects with the `*.yhw` file extension in this directory that were exported with a wrap key **other than** the one reconstituted by the shares here, the setup tool attempts to read those too, but will fail gracefully. The setup tool restores only the files it can decrypt.

Step 10

The restore process finishes and the setup tool informs you that the default factory-installed authentication key has been deleted.

```
Previous authentication key 0x0001 deleted
All done
```

The YubiHSM Setup application exits.

8.3 Verify the Duplicated YubiHSM 2

You should now have a secondary HSM 2 configured with the three key objects you created on the primary device earlier. Confirm that these key objects are identical to those on the primary device that was configured earlier:

Step 1

In the Command Prompt, run the YubiHSM Shell program:

```
$ yubihsm-shell
```

Step 2

To connect to the YubiHSM 2, at the `yubihsm` prompt, type `connect` and press **Enter**. A message confirming that you have a successful connection is displayed.

Step 3

To open a session with the YubiHSM 2, type `session open 3` (where 3 is the ID for your application authentication key) and press **Enter**.

Step 4

Type in the password for the application authentication key. A message confirming that the session has been set up successfully is displayed.

Step 5

To list the objects, type `list objects 0` (or instead of `0` the session number that was given to you

in step 4). Verify that the secondary device now contains all of the key material that you intended to back up or restore.

Depending on the order in which the keys under wrap were imported, the keys on the secondary device may not be listed in the same sequence as they are on the primary device when the `list` command is used. This has no practical implication and it is just the object IDs that need to be identical on the two devices.

If you have verified that the secondary device now contains all of the key material that you intended to restore, you should now remove the keys under wrap on file in the current working directory. The computer's hard drive can be erased, too.

CHAPTER

NINE

GETTING HELP

Should you require assistance when using this guide to deploy YubiHSM 2 on Windows, start by referencing the product documentation and currently known issues:

- Yubico Developers website
- Yubico Support
- YubiHSM 2 Product Overview
- Known Issues and Limitations

If you need additional help, contact Yubico directly by filling in a ticket on the Yubico Support site.

CHAPTER

TEN

TERMINOLOGY

The following terminology as it relates to YubiHSM 2 is used throughout this guide.

10.1 Software

Term	Description
Default authentication key	Factory-installed AES key used when initializing the device. Possesses all capabilities.
Application authentication key	AES key used to authenticate to the device. Performs operations according to its defined capabilities.
Audit key	AES authentication key with rights to access audit log.
Wrap key	AES key used to protect key material when exporting to file from device and when importing from file to device. Key material exported under wrap will be encrypted and can only be decrypted using the wrap key.
Capability	A description of what operations are allowed on or with an object such as a key.
Cryptographic API Next Generation (CNG)	A CNG is Microsoft's cryptographic architecture, which allows developers to implement applications with features for encryption, electronic signatures, certificate management, etc.
Delegated capability	An operation that an object is allowed to perform by virtue of receiving those permissions from the authentication key or wrap key that was used to create it.
Domain	A logical "container" for objects that can be used to control access to objects on the device.
Object ID	Object IDs are unique identifiers for any kind of object stored on YubiHSM2. An ID can range between 1 and 65535; however, the device can hold a maximum of 256 unique objects.
m of n	

CHAPTER

ELEVEN

COPYRIGHT

Trademarks

Yubico and YubiKey are registered trademarks of Yubico AB. All other trademarks are the property of their respective owners.

Disclaimer

The contents of this document are subject to revision without notice due to continued progress in methodology, design, and manufacturing. Yubico shall have no liability for any error or damages of any kind resulting from the use of this document.

The Yubico Software referenced in this document is licensed to you under the terms and conditions accompanying the software or as otherwise agreed between you or the company that you are representing.

Contact Information

Yubico Inc.
530 Lytton Street
Suite 301
Palo Alto, CA 94301
USA

Click the links to:

- Submit a support request
- Send a Contact Me request
- See additional contact options for getting touch with us

Document Updated

2022-07-08 19:38:04 UTC