Nexus

Certificate Manager

7.18.1

Integration Guide

CryptoServer HSM

SecurityServer 4.30.0

utimaco®

Imprint

Copyright 2026	Utimaco IS GmbH Krefelder Straße 220 52070 Aachen Germany
Phone	AMERICAS +1-844-UTIMACO (+1 844-884-6226) EMEA +49 800-627-3081 APAC +81 800-919-1301
Internet	https://support.hsm.utimaco.com/
e-mail	support@utimaco.com
Document Version	1.0.0
Date	2026-07-03
Status	PUBLISHED
Document No.	IG-2026-0049

Table of Contents

1 Introduction

Thank you for purchasing our CryptoServer Security System. We hope you are satisfied with our product. This paper provides an integration guide explaining how to integrate an Utimaco CryptoServer Hardware Security Module (HSM) with the Nexus Certificate Manager.

Please do not hesitate to contact us if you have any complaints or comments.

1.1 About this Guide

This guide describes how to enable HSM integration with Nexus Certificate Manager server, including Nexus Certificate Manager server installation. For more detailed information regarding Nexus Certificate Manager, please refer to the documentation provided by Nexus.

1.1.1 Target Audience for this Guide

This guide is primarily intended for Nexus Certificate Manager administrators and HSM administrators.

1.1.2 Document Conventions

The following conventions are used in this guide:

Convention	Use	Example
Bold	Items of the Graphical User Interface (GUI), e.g., menu options	Select **Details** and click on **Properties** button
`Monospaced`	Code that is given for explanation or as an example, file paths	`certreq.exe -new request.inf IISCertRequest.csr`
Italic	References and important terms	Operating system listed in *Software Requirements*

Table 1: Document conventions

We use special icons to highlight the most important notes and information.

Here you will find important safety information that should be followed.

Here you will find additional notes or supplementary information.

This message marks the result expected after the successful execution of an instruction.

1.1.3 Abbreviations

We use the following abbreviations in this guide:

Abbreviation	Meaning
HSM	Hardware Security Module
PKI	Public Key Infrastructure
CA	Certificate Authority
MBK	Master Backup Key
eID	e-Identity
CM	Certificate Manager
Abbreviation	*Meaning*

Abbreviation	Meaning
CMDB	Certificate Manager database
JDBC	Java Database Connectivity
CIS	Certificate Issuing System
CF	Certificate Factory
CMSNMP	Certificate Manager: Simple Network Management Protocol
CXI	Cryptographic eXtended Interface

Table 2: List of abbreviations

2 Overview

Nexus Certificate Manager is a flexible certificate authority (CA) platform with support for multitenancy and multiple use cases. It creates, issues, and manages the life-cycle of eIDs for people, infrastructure and things. The product manages certificates as well as private and public keys in public key infrastructures (PKI).

If the security of keys and certificates generated needs to be enhanced, the Nexus Certificate Manager needs to be configured to use a Hardware Security Module (HSM). When the HSM module is enabled with Nexus, this strengthens the protection of keys and certificates.

CryptoServer is a hardware security module developed by Utimaco IS GmbH. CryptoServer is a physically protected, specialized computer unit designed to perform sensitive cryptographic tasks and to securely manage, as well as store, cryptographic keys and data. It can be used as an universal, independent security component for heterogeneous computer systems.

3 Prerequisites and Requirements

Ensure that the system environment you will be using meets the following hardware and software requirements.

3.1 Software Requirements

Software	Software Requirements
Operating system	Windows Server: 2012 R2, 2016, 2019 CentOS 7,8 Red Hat Enterprise Linux: 7,8 SUSE Linux Enterprise Desktop 15.1 OpenSUSE Leap 15
Database	Microsoft SQL Server Express and Enterprise editions: 2012, 2017, 2019 Oracle Express and Enterprise editions:18.3, 19 PostgreSQL: 11, 11.5, 12 MySQL: 8.0 SQLite: 3.31
Java	Oracle Java SE JRE 11, OpenJDK 11. (64 bit)
Web application server	Apache Tomcat version 9.0
Nexus Certificate Manager	Nexus Certificate Manager 7.18.1 Nexus Personal Desktop 5.2.1

Table 3: List of software requirements

3.2 Hardware Requirements

Hardware	Hardware Requirements
Utimaco LAN HSM	CryptoServer CSe-Series/Se-Series LAN with firmware SecurityServer 4.30.0
Utimaco PCI-e HSM	CryptoServer CSe-Series/Se-Series PCI-e with firmware SecurityServer 4.30.0

Table 4: List of hardware requirements

4 Nexus Certificate Manager

The following instructions closely follow those given in [NIGCM]. That is also where more details on the installation of the Nexus Certificate Manager can be found, if needed.

1. Make sure that you have access to the license file before starting the installation.
2. Make `install_server.sh` executable. Execute the command:

›_ Console

```
chmod a+x install_server.sh
```

3. Execute `install_server.sh`.
4. Enter the full installation path, e.g. `/opt/cm/server`.

If CM clients are already installed on this computer, the server installation should not be made to the same directory.

5. Confirm the path.
6. Enter the name of the license file.
7. Confirm the file name.
8. Enter the number or name of each server component, for which you have the license, separated by a space character (" ").

The CMDB database should already exist and the JDBC component should be included to create the database connection parameters.

Install all server components at the same time. It is not possible to add new components later. To do this, the existing components should be uninstalled before they can be reinstalled (see Section 3.11.2 in [NIGCM]). If you have a license for the Key Archive and Recovery or Card Production Manager functions, but you are not going to use them, you should set the following parameters in `cm.conf`:

```
CardProductionManager.start=false
KARFactory.start=false
```

9. Select a database engine.
10. Select the machine where the database is installed. The default is `localhost`.
11. Select the JDBC port for the database.
12. Enter the database instance name.
13. Enter the name of the database CM user. The default is `lcmreq`.
14. Enter the password for the `lcmreq` user.
15. Await a message from the installation program, confirming a successful installation. If an error message occurs, correct the indicated error and restart the installation by using the `setup.sh` shell script located in the `<install_root>/install` directory.
16. Optionally, start the CMSNMP monitor: `<install_root>/bin/cmsnmp start`

If the CM SNMP monitor is installed, the `*.agent.connectToSupervisor` configuration parameter should be set to **true** in the configuration files `cis.conf` and `cm.conf`. `*.agent.connectToSupervisor = true`

17. Start CIS: `<install_root>/bin/cis start`
18. Start CF: `<install_root>/bin/cf start`
19. Check that the two processes have started and that the two instances of the Java process have started as well.
20. Check the log files in `<install_root>/logs` to see that the start was successful.
21. Continue with the configuration as described in the following chapters.

4.1 Executing Nexus CM as a System Service

1. The `<install_root>/bin/cmservices` script can be used to install or remove the CF, CIS and CMSNMP services as system services. To install them as system services, run the following command: `<install_root>/bin/cmservices install`

`<install_root>/bin/cmservices` script can also be used to start, stop or check the status of all services (e.g. . `/cmservices status`).

2. Start the services:

```
service cmsnmp
start service cis
start service cf start
```

3. Check the log files in `<install_root>/logs` to see that the services started successfully. If not, see the following step for further troubleshooting.

4. The CM server components will use the first found Java executable file available on the `$PATH` . To use a specific Java version, or if the `$PATH` does not include a directory where Java can be found, you can explicitly specify which Java version should be used by setting a parameter like the following in `<install_root>/bin/*_launch.conf` : `JAVA=/usr/jvm/jdk-default/bin/java`

When installing the CM services on a system that uses a systemd, it is possible that the default `$PATH` , as seen by the systemd service wrappers, does not include the expected Java version. If this is the case, then calling the service cis start, for example, might not start the service. It will instead be reported as active (exited) by service cis status. To correct this, see the previous step of how to explicitly set the Java executable file to use.

4.2 Executing Nexus CM as Service User

It is recommended to execute the CM services as a separate non-root service user. This can be configured by performing the following steps:

1. Make sure that the CM services are stopped.

2. Create a user (e.g., `useradd cmuser`).
3. Create the service user owner of the directory `<install_root>`. This can be done, for example, by executing the following command: `chown -R cmuser:cmuser /opt/cm/server/`
4. Update the `<install_root>/bin/*_launch.conf` files, so they specify the line with the "SERVICEUSER" for example: `SERVICEUSER=cmuser`

5 Configuring PKCS#11

5.1 Introduction and Prerequisites

Before the PKCS#11 interface and library can be used, some manual actions have to be performed. Follow the steps below to configure the PKCS#11 library and initialize a PKCS#11 slot. For further information about the PKCS#11 setup, please refer to [CSPKCSDG].

5.2 Installing PKCS#11 on the Workstation

The installation file for the PKCS#11 is located on the Product CD. The installer creates an environment variable called `CS_PKCS11_R2_CFG`. It contains the path to Utimaco's PKCS#11 configuration file. By default, we have to copy the file to `/etc/utimaco` in Linux. We have to set the `CS_PKCS11_R2_CFG` environment variable to point to this location.

In order to be able to access the HSM via PKCS#11, the configuration file needs to be modified.

1. Set the path to the logfile and set the desired log level.

```
[Global] # For unix:

Logpath = /tmp # For windows:

# Logpath = C:/ProgramData/Utimaco/PKCS11_R2

# Loglevel (0 = NONE; 1 = ERROR; 2 = WARNING; 3 = INFO; 4 = TRACE) Logging = 4
```

2. Set the IP address of the HSM.

```
[CryptoServer]

# Device specifier (here: CryptoServer is CSLAN with IP address 127.0.0.1)

Device = 127.0.0.1
```

3. Optionally, make additional modifications to the configuration file, such as setting up an external store as described in [CSPKCSM]. We suggest to modify the PKCS#11 config file to `KeepAlive` flag active.

```
[Global]

# Prevents expiring session after inactivity of 15 minutes

KeepAlive = true
```

5.3 Generating an MBK

One of the steps of the HSM initialization is generating a new MBK, which can be used for creating backups, for using an external storage and for synchronizing HSM clusters. By default, MBK is an AES256 key, though it is also possible to generate a DES MBK by using the csadm tool for backward compatibility reasons.

It is required to generate an MBK for the HSM to become operational. Without an MBK no cryptographic operations can be run.

To generate an MBK:

1. Open the Crypto Administration Tool.
2. Achieve the permission level of at least 02000000.
3. Click **Manage MBK** to access the **Remote Master Backup Key Management** window and select the **Generate** tab.
4. Type the name of the MBK in the **MBK Name** section.
5. Select the backup mode of the MBK shares as either **XOR** or **m out of n**.

- If **m out of n** was selected, it is necessary to select the number of m (shares) and n (shares) by using the drop-down menus, set by default as 2 and 3.

6. In case that this MBK also needs to be imported at the same time into the HSM, select the **Automatic MBK Import** option.
7. Click **Generate**.
8. Select whether the MBK shares should be saved on smartcards or as keyfiles by selecting either the **Smartcard Token** or the **Keyfile Token** option.

- If you chose to export the MBK shares on smartcards, follow the instructions on the smart card reader to export all of the m parts.

5.4 Importing the MBK

In case the MBK was not imported when it was generated, or if you want to upload it to another HSM, you need to import it from the keyfiles or smartcards that carry its parts. The MBK was divided into multiple parts by using Shamir's Secret Sharing or XOR and can be restored by using the "m out of n" or XOR principle.

1. Make sure that at least m out of n smart cards/keyfiles are available.
2. Open the CryptoServer Administration Tool.
3. Achieve the permission level of at least 02000000.
4. Click **Manage MBK** to access the **Remote Master Backup Key Management** window and select the **Import** tab.
5. Select the type of MBK that will be imported and the value m.
6. Click **Import**.
7. Select whether to save the MBK parts on smartcards or as keyfiles by selecting either the **Smartcard Token** or **Keyfile Token** option.
8. Follow the instructions to import all of the m parts.

5.5 Initializing PKCS#11 on the HSM

In addition to PKCS#11, the PKCS#11 graphical interface tool (P11CAT) and the PKCS#11 command line interface (p11tool2) are installed. This chapter shows how to use the P11CAT in order to initialize the PKCS#11 Slot 0. There are 10 active PKCS#11 slots by default. The number of PKCS#11 slots can be modified in the PKCS#11 configuration file.

1. Make sure that the PKCS#11 configuration file contains the IP address of your HSM and that the HSM is running.
2. Open the P11CAT tool on your workstation. When opening the tool for the first time, the slots should be as shown in the figure below.

Slot List

Slot ID	Token Init.	PIN Init.	Login Status
0000 0000			
0000 0001			
0000 0002			
0000 0003			
0000 0004			
0000 0005			
0000 0006			
0000 0007			
0000 0008			
0000 0009			

Figure 1 : PKCS#11 slots before initialization

3. Select row **0000 0000** under the Slot ID on the top left-hand side in the Slot List.

4. Click on **Login/Logout**.
5. Click on **Login Generic**.
6. Login as a user with the permission mask at least 20000000 (User Manager permission).
7. Click on **Slot Management**.
8. Create a Security Officer (SO) for Slot 0. Click on **Init Token**. Write the Token Label. Set the SO PIN. Confirm the SO PIN. Click on **Init Token**. Observe the changed Token Init. status for Slot 0.
9. Log out the ADMIN user. Click on **Login/Logout**. Click on **Logout All**.
10. Login as the SO. Click on **Login/Logout**. Click on **Login SO**. Enter the SO PIN. Click on **Login**.
11. Click on **Slot Management**.
12. Create the User for the Slot 0. Select **Init PIN**. Enter the **Normal User PIN**. Confirm the **Normal User PIN**. Click on **Init PIN**. Observe the changed PIN Init. status for the Slot 0.
13. Log out the SO. Click on **Login/Logout**. Click on **Logout All**.

The Slot 0 is now initialized. An application or a user can now connect to the PKCS#11 Slot 0 and create or store objects on the slot. Find further information on creating or deleting objects and users in [CSPKCSM] and [LPKCSHD].

5.6 Configuring PKCS#11 Connectivity on Nexus Manager

To integrate the HSM cryptographic engine with the safe key storage, configuration of the Certificate Issuing System (CIS) is necessary. Add the following lines to the `/opt/cm/server/config/cis.conf` configuration file. Then, restart the CIS service to make the changes to become effective.

```
cis.crypto.device.5.name = Utimaco HSM (RSA) cis.crypto.device.5.type = PKCS11
;cis.crypto.device.5.label = insert Label cis.crypto.device.5.slotid = insert Slot
ID

cis.crypto.device.5.library = /etc/utimaco/libcs_pkcs11_R2

cis.crypto.device.5.algorithm = RSA cis.crypto.device.5.algorithmOID =
1.2.840.113549.1.1.1
cis.crypto.device.5.keysize = 1024 2048 4096 8192 16384
cis.crypto.device.5.rsaPubExp = 65537
;cis.crypto.device.5.pinPad = false cis.crypto.device.5.pin = 123456

cis.crypto.device.5.hashInCis = false



cis.crypto.device.6.name = Utimaco HSM (ECDSA)

cis.crypto.device.6.type = PKCS11 ;cis.crypto.device.6.label = insert Label
cis.crypto.device.6.slotid = insert Slot ID

cis.crypto.device.6.library = /etc/utimaco/libcs_pkcs11_R2
cis.crypto.device.6.algorithm = EC cis.crypto.device.6.algorithmOID = EC

cis.crypto.device.6.parameter = secp192r1, secp224r1  secp256r1, secp384r1,
secp521r1
cis.crypto.device.6.rsaPubExp = 65537 ;cis.crypto.device.6.pinPad = false
cis.crypto.device.6.pin
= 123456 cis.crypto.device.6.hashInCis = false
```

6 Creating the CA Keys

1. Set the basic configuration for creating the CA keys. Carefully check that all settings are correct and click OK.

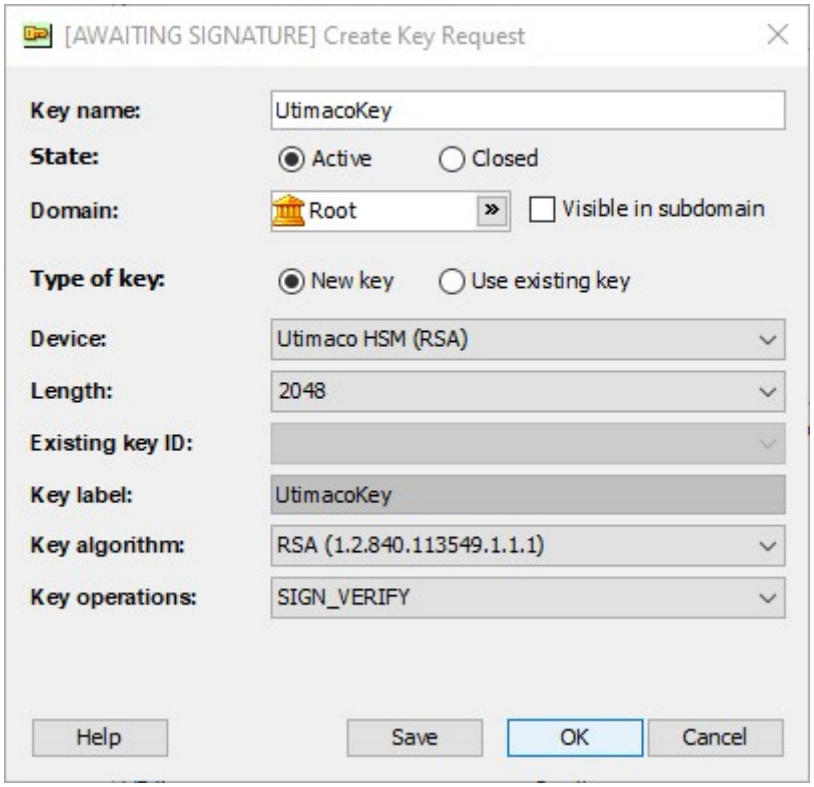

Figure 2 : Creating a CA key

Be careful to select the Utimaco device!

2. Select the Nexus security office user and enter the pin, which was previously set in a configuration file.

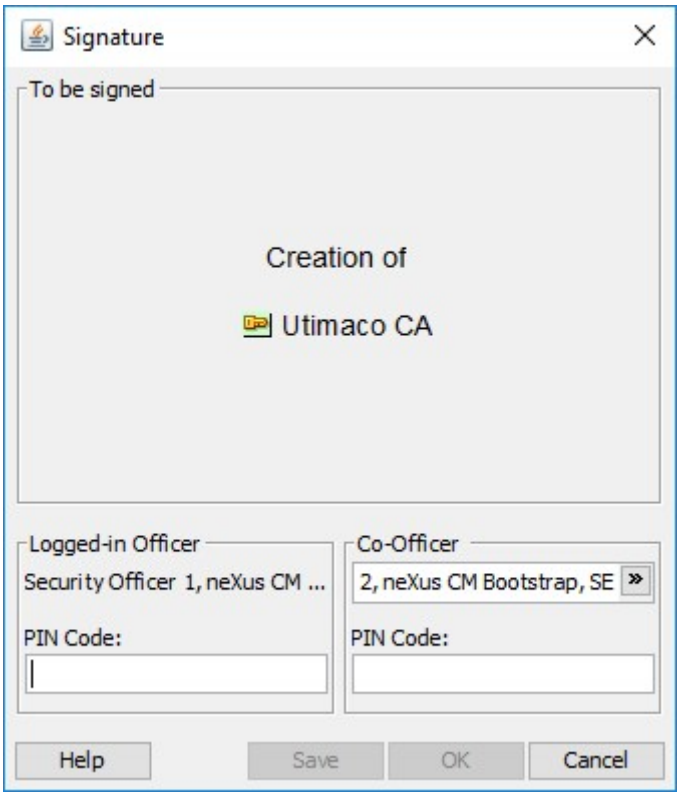

Figure 3 : Key creation

6.1 Creating the CA

When creating a root ca/CA, set the **self-signed** and the basic configuration for creating a root ca/CA. Then, carefully check that all settings are correct and click **OK**.

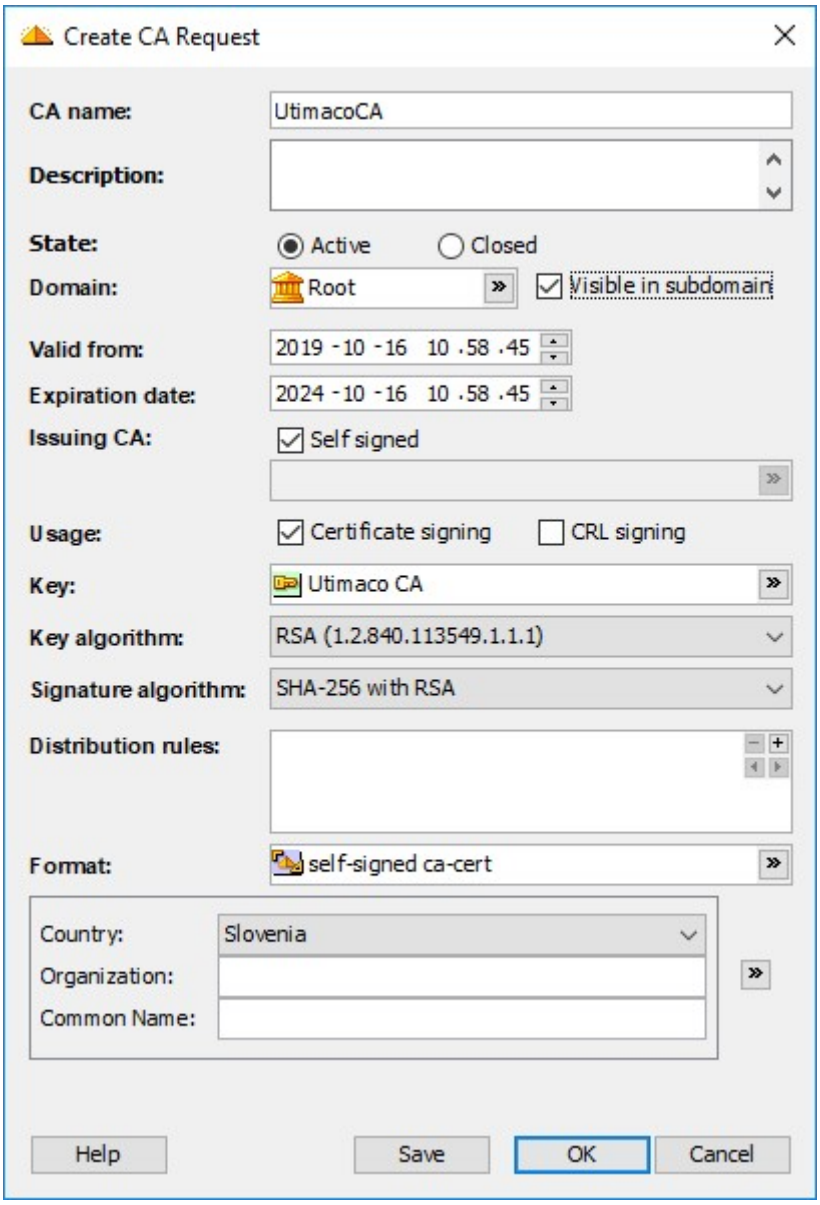

Figure 4 : Root ca/CA create

Be sure to select the issuing CA if you are creating a sub-ca.

7 Backup and Restore

The Utimaco HSM enables different ways of making backups of either the entire database of keys or just groups of keys. All the backups are encrypted by using the Master Backup Key (MBK), generated by the system administrator, when setting up the HSM. More information about the MBK can be found in [CSADMIN].

7.1 Backing up and Restoring Key Database

All of the keys inside the HSM are stored in a CXI database and it is possible to back up the entire database at once. In the same way, the user database can be backed up as well. In FIPS mode this feature is not supported.

1. Open the Crypto Administration Tool.
2. Make sure that the HSM is connected and in the operational mode.
3. Click on **Login/Logoff** to open the **Login/Logoff User** window.
4. Log in the appropriate users to achieve the permission level of at least 22000000.
5. Click on **Backup/Restore** to open the **CryptoServer Database Backup/Restore Wizard** window.

    a. In the **Command** section select either to:

- **Backup databases from source CryptoServer to backup directory**,
- **Restore databases from backup directory to target CryptoServer**,
- **Copy databases from source CryptoServer to target CryptoServer**.

    b. In the **Settings** section:

- Select the appropriate **Source CryptoServer**,
- If available, select the appropriate **Target CryptoServer**,
- In the **Backup directory** section type the appropriate backup directory path (set to `C:\Program Files\Utimaco\CryptoServer\Administration` as default), or click … to browse for the appropriate directory.

6. Select the databases to backup or restore.
7. Click **Execute**.

8. A confirmation window appears.

For a FIPS backup, please use the P11CAT or the P11tool2 tools.

7.2 Backing up and Restoring a Key Database with P11CAT

It is possible to back up separate PKCS#11 slots by using either the P11CAT or the p11tool2. In this guide we use the P11CAT for the backup procedure. Please refer to [CSP11TOOL2] for the backup procedure with the p11tool2.

1. Open the PKCS#11 CryptoServer Administration (P11CAT).
2. Log in to the slot you wish to back up as the Cryptographic User (achieve the permission level of at least 00000002).
3. Click on **Backup/Restore**.
4. Click on **Backup/Restore Keys**.
5. Select one among the 4 options. Click the one that corresponds to your case. The possibilities are to perform either:

- **Backup Internal Keys**.
- **Backup External Keys**.
- **Restore Key Backup to Internal Key Store**.
- **Restore Key Backup to External Key Store**.

6. A pop-up window opens.

    a. Select the directory where the key database will be backed up, and type the name of the key database in the section **File name**.

    b. If you chose to restore a key backup to an internal or an external key store, select the directory where your backup is located.

    c. To confirm your choice, click on **Save**.

7. In the **Status** window a log of the performed action is displayed.

8 FIPS Requirements

All the steps are identical for the HSM in FIPS 140-2 approved mode. The only difference is that the backup of the entire key database is not possible. In this case the P11CAT or the p11tool2 are used for backing up the keys.

Note that although the integration does not require extra steps, the HSM running in FIPS mode will accept ONLY FIPS compliant parameters. Be careful to select FIPS compliant algorithms, key lengths and elliptic curves when generating new keys. For more information about the FIPS compliant algorithms please refer to the CryptoServer User and Administration Guides.

9 Further Information

This document forms a part of the information and support which is provided by the Utimaco IS GmbH. Additional documentation can be found on the product CD in the Documentation directory.

All CryptoServer product documentation is also available at the Utimaco IS GmbH website: *https://utimaco.com*.

10 References

Reference	Title/Company	Document No.
[CSPKCSM]	CryptoServer - PKCS#11 P11CAT Manual	M013-0001-en
[LPKCSHD]	Learning PKCS#11 in Half a Day	2015-0008
[CSPKCSDG]	CryptoServer PKCS#11 R2 Developer Guide	2012-0007
[NIGCM]	Nexus Installation Guide Certificate Manager 7.18.1	2018-12-14

Table 5: References

11 Contact and Support Information

You can reach us from Monday to Friday, 09.00 a.m. to 05.00 p.m., Central European Time (CET).

Utimaco IS GmbH
Krefelder Str. 220
52070 Aachen
Germany

RMA Query

If you need to send the device back to Utimaco IS GmbH, please open a new RMA case (Return Merchandise Authorization). We request that you use the following web address. RMA cases cannot be opened by email or phone.

https://support.hsm.utimaco.com/support/rma/new

Other Support Queries

- Mail (preferred contact method)
  support@utimaco.com
  Attach the diagnostic information to your email.
- Web portal
  https://support.hsm.utimaco.com/support/cases/new/
  The diagnostic information will be requested in our response if necessary.
- By phone
  AMERICAS +1-844-UTIMACO (+1 844-884-6226)
  EMEA +49 800-627-3081
  APAC +81 800-919-1301
  The diagnostic information will be requested in our response if necessary.