OpenDNSSEC

2.1.7

Integration Guide

CryptoServer HSM

SecurityServer 4.45.5.1

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	2026-07-20
Status	PUBLISHED
Document No.	IG-2026-0076

All rights reserved

No part of this documentation may be reproduced in any form (printing, photocopy or according to any other process) without the written approval of Utimaco IS GmbH or be processed, reproduced or distributed using electronic systems.

Utimaco IS GmbH reserves the right to modify or amend the documentation at any time without prior notice. Utimaco IS GmbH assumes no liability for typographical errors and damages incurred due to them.

All trademarks and registered trademarks are the property of their respective owners.

Table of Contents

1 Introduction

This guide is part of the information and support provided by Utimaco. Additional documents produced to support your Utimaco SecurityServer product can be found in the document directory of the Utimaco SecurityServer product bundle. All Utimaco SecurityServer product documentation are available on Utimaco's website at *https://utimaco.com/*.

1.1 About This Guide

This guide provides an integration explaining how to integrate an Utimaco CryptoServer Hardware Security Module (HSM) with OpenDNSSEC. Utimaco HSM is used to secure the private keys for Key Signing Key (KSK) and Zone Signing Key (ZSK) and offload signing operations on the HSM.

1.2 Target Audience for This Guide

This guide is intended for administrators of OpenDNSSEC and of Utimaco HSMs.

1.3 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 *Tested Versions*

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.

1.4 Abbreviations

The following abbreviations are used in this guide:

Abbreviation	Meaning
CD	Compact Disc
CSADM	CryptoServer Command-line Administration Tool
DNS	Domain Name System
DNSSEC	Domain Name System Security Extension
DSA	Digital Signature Algorithm
GOST	Gosudarstvennyi Standard or Government Standard
GUI	Graphical User Interface

Abbreviation	Meaning
HSM	Hardware Security Module
IP	Internet Protocol
KASP	Key and Signing Policies
KSK	Key-Signing Keys
LAN	Local Area Network
MBK	Master Backup Key
PCIe	PCI Express Interface
PIN	Personal Identification Number
PKCS#11	Public-Key Cryptography Standard #11
RSA	Rivest-Shamir-Adleman
SO	Security Officer
URL	Uniform Resource Locator
ZSK	Zone-Signing Keys

Table 2: List of abbreviations

2 Overview

2.1 OpenDNSSEC

OpenDNSSEC is a system to manage zones. It takes in unsigned zones and policies and produces and maintains DNSSEC signed zones. OpenDNSSEC is responsible for signing, resigning, key generation, and fetching/distributing zones to and from nameservers. The policies are defined in the KASP (Key and Signing Policies). These describe the configuration for DNSSEC such as key lengths and lifetimes.

2.2 Utimaco CryptoServer HSM

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 a universal, independent security component for heterogeneous computer systems.

3 Integration Requirements and Prerequisites

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

This guide assumes that the user has already installed and configured required Software.

3.1 Tested Versions

The integrations that have been successfully tested with the Utimaco HSM with OpenDNSSEC Server.

Operating Systems	OpenDNSSEC	Utimaco Security Server Version	Utimaco HSM
Debian 11	2.1.7	SecurityServer 4.45.5.1 p11tool2 from product package Utimaco SecurityServer	CryptoServer CSe-Series/Se -Series

Table 3: List of tested versions

3.2 Software Requirements

Software	Software Requirements
SecurityServer	4.45.5.1
HSM Interface	SecurityServer PKCS#11 Provider

Table 4: List of software requirements

3.3 Hardware Requirements

Hardware	Hardware Requirements
Utimaco LAN HSM	CryptoServer CSe-Series/Se-Series LAN with firmware SecurityServer 4.45.5.1
Utimaco PCI-e HSM	CryptoServer CSe-Series/Se-Series PCI-e with firmware SecurityServer 4.45.5.1

Table 5: List of hardware requirements

Set up an account on the Utimaco support portal and request download access at the following URL: https://support.hsm.utimaco.com/.

3.4 Prerequisites

Before you begin, please ensure that:

- The CryptoServer is set up and configured. Refer to the CryptoServer documentation to set up the HSM.
- The CryptoServer Default Admin has been replaced with a new admin user.
- The MBK has been created and stored onto each HSM. Refer to the CryptoServer documentation to set up the MBK.
- The operating system is listed in *Tested Versions*.
- The SecurityServer is listed in *Tested Version.*
- There is a user with admin privileges, as it is required for installing few packages on to the OpenDNSSEC server.
- You allow port 53 through firewall.
- You familiarize yourself with the DNS and OpenDNSSEC documents and setup process.

4 Installing and Configuring Utimaco SecurityServer Software

4.1 Download and Install Utimaco Software

If you have not already done so, please create and request a Utimaco Support Portal Account. This will allow you to download the software components needed for this installation.

1. Copy the downloaded software at the appropriate location on the OpenDNSSEC Server.
2. Create a utimaco folder under `/opt` directory and further create 2 directories: `/opt/utimaco/bin` and `/opt/utimaco/lib`.

›_ Console

```
# mkdir -p  /opt/utimaco/bin

# mkdir /opt/utimaco/lib
```

3. Copy the pkcs11 library file `libcs_pkcs11_R3.so` from Utimaco CryptoServer software to the `/opt/utimaco/lib` directory and make the file executable.

›_ Console

```
# cp ~/path_to_application_folder/lib/libcs_pkcs11_R3.so /opt/utimaco/lib
```

4. Copy the csadm and p11tool2 files from Utimaco CryptoServer software to `/opt/utimaco/bin` directory and make both the files executable.

›_ Console

```
# cd ~/path_to_application_folder

# cp csadm p11tool2 /opt/utimaco/bin

# chmod +x /opt/utimaco/bin/csadm /opt/utimaco/bin/p11tool2
```

4.2 CryptoServer PKCS#11 Configuration

1. Create the directory `/etc/utimaco`. Locate the Utimaco PKCS#11 configuration file in your SecurityServer directory, `Linux/x86-64/Crypto_APIS/PKCS11_R3/sample`. Copy the Utimaco PKCS#11 configuration `file cs_pkcs11_R3.cfg` into `/etc/utimaco` directory.

›_ Console

```
# mkdir /etc/utimaco

# cd <install directory>/Software/Linux/x86-

64/Crypto_APIs/PKCS11_R3/sample # cp cs_pkcs11_R3.cfg /etc/utimaco

# cd /etc/utimaco
```

2. Edit the `cs_pkcs11_R3.cfg` file and make the appropriate changes to the file.

cs_pkcs11_R3.cfg

```
[Global]

# For unix:

Logpath = /tmp

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

Keepalive = true

# Set the Device to connect with

[CryptoServer] # Device specifier  Device = <HSM_IP>
```

For more information regarding the commands and command parameters please check the Utimaco CryptoServer documentation. The device may be a CryptoServer (PCIe or LAN) device. The device line will follow one of these patterns, based on the HSM form-factor:

Device = 288@<HSM IP address> Hardware (LAN) HSM

OR

Device = /dev/cs2.0 Hardware (PCIe) HSM

To make your testing easier, it would be good to enable the PKCS#11 log file. That can be enabled by editing the **Logging Loglevel**. Set the **LogPath** and **Logging Loglevel** to 1. For testing you may want to increase it to 4.

The added **LogPath** points to a writable directory, not to a file.

If you encounter problems, check the log file named `cs_pkcs11_R3.log` in the **LogPath** defined directory. When you are done testing, you should change **Logging** to 1 or 2. This will limit the logging to only critical and important messages.

4.3 Create SO User and Initialize a Slot

You should initialize a slot with a custom label using p11tool2.

First, using p11tool2 create the SO or Security Officer and then using p11tool2 command initialize the slot that you want to use, and the slot user as shown below.

›_ Console

```
# ./p11tool2 slot=<slot_no> Label=<token_label> Login=ADMIN,ADMIN.key
InitToken=<SO_PIN>

# ./p11tool2 slot=<slot_no> LoginSO=<SO_PIN> InitPin=<CryptoUser_PIN>
```

```
root@opendnssec:~# p11tool2  slot=9 Label=OpenDNSSEC Login=ADMIN,ADMIN.key InitToken=123456
root@opendnssec:~#
root@opendnssec:~# p11tool2 slot=9 LoginSO=123456 InitPin=123456
root@opendnssec:~#
```

Figure 1 : Slot initialization output

5 Integrating OpenDNSSEC with Utimaco HSM

5.1 Installing OpenDNSSEC

1. (Optional) It is recommended to update the system with the latest security patch.

›_ Console

```
# apt update -y && apt  upgrade -y
```

2. Install the OpenDNSSEC package with the command below.

›_ Console

```
# apt install opendnssec -y
```

```
root@opendnssec:~# apt install opendnssec -y
Reading package lists ... Done
Building dependency tree ... Done
Reading state information ... Done
The following additional packages will be installed:
  libdbd-sqlite3-perl libdbi-perl libhsm-bin libldns3 libsofthsm2 opendnssec-common opendnssec-enforcer
  opendnssec-enforcer-sqlite3 opendnssec-signer softhsm2 softhsm2-common sqlite3
Suggested packages:
  libclone-perl libmldbm-perl libnet-daemon-perl libsql-statement-perl opendnssec-doc sqlite3-doc
The following NEW packages will be installed:
  libdbd-sqlite3-perl libdbi-perl libhsm-bin libldns3 libsofthsm2 opendnssec opendnssec-common opendnssec-enforcer
  opendnssec-enforcer-sqlite3 opendnssec-signer softhsm2 softhsm2-common sqlite3
0 upgraded, 13 newly installed, 0 to remove and 0 not upgraded.
Need to get 3296 kB of archives.
After this operation, 9944 kB of additional disk space will be used.
Get:1 http://debian-archive.trafficmanager.net/debian bullseye/main amd64 libdbi-perl amd64 1.643-3+b1 [780 kB]
Get:2 http://debian-archive.trafficmanager.net/debian bullseye/main amd64 libdbd-sqlite3-perl amd64 1.66-1+b1 [179 kB]
Setting up libdbd-sqlite3-perl:amd64 (1.66-1+b1) ...
Setting up libhsm-bin (1:2.1.7-2) ...
Setting up opendnssec-enforcer-sqlite3 (1:2.1.7-2) ...
Setting up opendnssec-enforcer (1:2.1.7-2) ...
Setting up opendnssec (1:2.1.7-2) ...
Processing triggers for man-db (2.9.4-2) ...
Processing triggers for libc-bin (2.31-13+deb11u5) ...
root@opendnssec:~#
root@opendnssec:~#
```

Figure 2 : Installing OpenDNSSEC

5.2 Configuring OpenDNSSEC to Use Utimaco HSM

5.2.1 Setting Up Utimaco CryptoServer Library in OpenDNSSEC Configuration File

1. Open the OpenDNSSEC configuration file `/etc/opendnssec/conf.xml` and edit the details in **<RepositoryList> …… </Repository>** as below.

›_ Console

```
<?xml version="1.0" encoding="UTF-8"?>

<Configuration>

        <RepositoryList>

                <Repository name="utimaco">


<Module>/opt/utimaco/lib/libcs_pkcs11_R3.so</Module>

                        <TokenLabel>odnssec</TokenLabel>
                        <PIN>1234</PIN>

                        <SkipPublicKey/>

          </Repository>
```

<Pin> …. </PIN> attribute is optional and can be commented as per requirement. IF the PIN is not set in the configuration file, user will need to enter the CryptoUser PIN every time.

<Repository name="utimaco"> an important attribute is the name of the local repository which is used in `kasp.xml` to identify storage of the keys. User can give repository name as per their requirement. Refer to the *OpenDNSSEC Documentation* for more details of the `conf.xml` file.

2. Run the command below to verify the connection with HSM.

›_ Console

```
# ods-hsmutil info
```

```
root@opendnssec:~# ods-hsmutil info
Repository: utimaco
        Module:         /opt/utimaco/lib/libcs_pkcs11_R3.so
        Slot:           9
        Token Label:    OpenDNSSEC
        Manufacturer:   Utimaco IS GmbH
        Model:          CryptoServer
        Serial:         CS711108_0009
root@opendnssec:~#
root@opendnssec:~#
```

Figure 3 : Console output of ods-hsmutil info

5.2.2 Testing Key Generation onto the Utimaco HSM using OpenDNSSEC

1. Run the below command to generate the keys using the `ods-hsmutil` utility onto Utimaco HSM.

›_ Console

```
# ods-hsmutil generate utimaco rsa 2048
```

```
root@opendnssec:~# # ods-hsmutil generate utimaco rsa 2048
root@opendnssec:~# ods-hsmutil generate utimaco rsa 2048
Generating 2048 bit RSA key in repository: utimaco
Key generation successful: dd68f48d0f2f67c5cd6b3d8f6e01b93a
root@opendnssec:~#
root@opendnssec:~#
```

Figure 4 : Generate RSA key output

2. Using `p11tool2` utility, verify the keys are created on the HSM using the below command.

›_ Console

```
# p11tool2 slot=9 loginuser=123456 listobjects
```

```
root@opendnssec:~# p11tool2 slot=9 loginuser=123456 listobjects


CKO_PUBLIC_KEY:

+ 1.1
  CKA_KEY_TYPE                  = CKK_RSA
  CKA_LABEL                     = dd68f48d0f2f67c5cd6b3d8f6e01b93a
  CKA_ID                        =
                0xDD68F48D 0F2F67C5 CD6B3D8F 6E01B93A | h   /g  k= n  :|



CKO_PRIVATE_KEY:

+ 2.1
  CKA_KEY_TYPE                  = CKK_RSA
  CKA_SENSITIVE                 = CK_TRUE
  CKA_EXTRACTABLE               = CK_FALSE
  CKA_LABEL                     = dd68f48d0f2f67c5cd6b3d8f6e01b93a
  CKA_ID                        =
                0xDD68F48D 0F2F67C5 CD6B3D8F 6E01B93A | h   /g  k= n  :|
```

Figure 5 : ListObjects output

3. List the contents of OpenDNSSEC Repository and verify the keys created is listed below with the command below.

›_ Console

```
# ods-hsmutil list utimaco
```

```
root@opendnssec:~# ods-hsmutil list utimaco

Listing keys in repository: utimaco
1 key found.

Repository           ID                                Type
----------           --                                ----
utimaco              dd68f48d0f2f67c5cd6b3d8f6e01b93a  RSA/2048
root@opendnssec:~#
root@opendnssec:~#
```

Figure 6 : List DNSSEC keys output

4. Test the HSM by generating a DNSKEY Resource Record using the previously created RSA key.

›_ Console

```
# ods-hsmutil dnskey  dd68f48d0f2f67c5cd6b3d8f6e01b93a  utimaco 256 5
```

```
root@opendnssec:~# ods-hsmutil dnskey  dd68f48d0f2f67c5cd6b3d8f6e01b93a  utimaco 256 5
utimaco.        3600    IN      DNSKEY  256 3 5 AwEAAbDy93M81mb9fS2A7tANgcbcRM8CRQXmUo9HH+1iL3cAn0n64ZHq4bTApQLnz6OE1CQPoIlEy0
ld4NR5l54vvIweckIp5ePKLx1svzlLlZ4/123490V2o0Szzd0blRd/siK9bPSYN4kDdt/cUp/RGGshS8coVRMoN/mf9cfKvtHhg7KH1+1lrJFIjAYYtNPxmllNMTpV
ceomwR6jNLDwV+VALmo/PXZI9B8uQFKHunI9gNPUO1GrOazjVzPUojokFy2A2XWeAMgfTudFSWONNGsMSRjKohSTMPyIPbQEYzTfMNVpNTXybDRVQhx5bhSlpE6Dgz
xo6luKwX1KKWQ5pRk= ;{id = 41485 (zsk), size = 2048b}
root@opendnssec:~#
root@opendnssec:~#
```

Figure 7 : Generate resource record DNSKEY output

5. (Optional) If the user wants to delete the key from the local repository, use the command below.

›_ Console

```
# ods-hsmutil remove dd68f48d0f2f67c5cd6b3d8f6e01b93a
```

```
root@opendnssec:~# ods-hsmutil remove dd68f48d0f2f67c5cd6b3d8f6e01b93a
Key remove successful.
root@opendnssec:~#
root@opendnssec:~#
```

Figure 8 : Delete DNSSEC key output

Deleting the key from local repository using the above command will also delete the key from HSM. To verify run p11tool2 command as described in the step 2 above.

6. (Optional) Now, as the communication between the Utimaco HSM and OpenDNSSEC is established and verified, you can run a few built-in tests as shown below.

›_ Console

```
# ods-hsmutil test utimaco
```

```
root@opendnssec:~# ods-hsmutil test utimaco
Testing repository: utimaco

Generating 512-bit RSA key... OK
Extracting key identifier... OK, 6a0a899b7551ae5775beca2bccf73d1b
Signing (RSA/SHA1) with key... OK
Signing (RSA/SHA256) with key... OK
Deleting key... OK

Generating 768-bit RSA key... OK
Extracting key identifier... OK, 9a367620d5f39c225f5ecb4ec345b82b
Signing (RSA/SHA1) with key... OK
Signing (RSA/SHA256) with key... OK
Deleting key... OK

Generating 1024-bit RSA key... OK
Extracting key identifier... OK, 780c20ba875bfa4c0edbeb175dc4b67f
Signing (RSA/SHA1) with key... OK
Signing (RSA/SHA256) with key... OK
Signing (RSA/SHA512) with key... OK
Deleting key... OK

Generating 1536-bit RSA key... OK
Extracting key identifier... OK, 899c1c8729de54079e8918bcdb8e34c3
Signing (RSA/SHA1) with key... OK
Signing (RSA/SHA256) with key... OK
Signing (RSA/SHA512) with key... OK
Deleting key... OK
```

Figure 9 : DNSSEC built-in test output

```
Extracting key identifier ... OK, f107fd675aed8fb3efbc672af5a21c0a
Signing (RSA/SHA1) with key ... OK
Signing (RSA/SHA256) with key ... OK
Signing (RSA/SHA512) with key ... OK
Deleting key ... OK

Generating 512-bit DSA key ... Failed
generate key pair: CKR_TEMPLATE_INCONSISTENT

Generating 768-bit DSA key ... Failed
generate key pair: CKR_TEMPLATE_INCONSISTENT

Generating 1024-bit DSA key ... Failed
generate key pair: CKR_TEMPLATE_INCONSISTENT

Generating 512-bit GOST key ... Failed
generate key pair: CKR_MECHANISM_INVALID

Generating ECDSA Curve P-256 key ... OK
Extracting key identifier ... OK, 5c6169185fbf3afd9936cf38fac97d86
Signing (ECDSA/SHA256) with key ... Generating ECDSA Curve P-384 key ... OK
Extracting key identifier ... OK, d0a59b247323d21119eac8a93cb51065
Signing (ECDSA/SHA384) with key ... Generating ED25519 key ... Failed
generate key pair: CKR_MECHANISM_INVALID

Generating ED448 key ... Failed
generate key pair: CKR_MECHANISM_INVALID

Generating 1024 bytes of random data ... OK
Generating 32-bit random data ... 2403403717
Generating 64-bit random data ... 5581746184730968092
root@opendnssec:~#
root@opendnssec:~#
```

Figure 10 : DNSSEC built-in test output

DSA, GOST, ED25519 and ED448tests are expected to be failed, as they are not supported by the HSM.

5.2.3 Configuring OpenDNSSEC Policy to Use Utimaco Repository

1. Open the `/etc/opendnssec/kasp.xml` file and change the Repository to Utimaco in all the `<Repository></Repository>` tags.

/etc/opendnssec/kasp.xml

```
<?xml version="1.0" encoding="UTF-8"?>


<KASP>


        <Policy name="default">

                <Description>A default policy that will amaze you and your
friends</Description>

                <Signatures>

                        <Resign>PT2H</Resign>
                        <Refresh>P3D</Refresh>

                        <Validity>

                                <Default>P14D</Default>
<Denial>P14D</Denial>

                        </Validity>

                        <Jitter>PT12H</Jitter>

                        <InceptionOffset>PT3600S</InceptionOffset>

                        <MaxZoneTTL>P1D</MaxZoneTTL>

                </Signatures>


                <Denial>

                        <NSEC3>

                                <!-- <TTL>PT0S</TTL> -->

                                <!-- <OptOut/> -->

                                <Resalt>P100D</Resalt>

                                <Hash>

                                        <Algorithm>1</Algorithm>
```

/etc/opendnssec/kasp.xml

```
                                <Iterations>5</Iterations>
                                <Salt length="8"/>
                        </Hash>
                </NSEC3>                </Denial>


        <Keys>
                <!-- Parameters for both KSK and ZSK -->
                <TTL>PT3600S</TTL>
                <RetireSafety>PT3600S</RetireSafety>
                <PublishSafety>PT3600S</PublishSafety>
                <!-- <ShareKeys/> -->
                <Purge>P14D</Purge>

                <!-- Parameters for KSK only -->
                <KSK>
                        <Algorithm length="2048">8</Algorithm>
                        <Lifetime>P1Y</Lifetime>
                        <Repository>utimaco</Repository>
                </KSK>

                <!-- Parameters for ZSK only -->
                <ZSK>
                        <Algorithm length="1024">8</Algorithm>
                        <Lifetime>P90D</Lifetime>
                        <Repository>utimaco</Repository>
```

/etc/opendnssec/kasp.xml

```
                        <!-- <ManualRollover/> -->
                </ZSK>
                                </Keys>


        <Zone>
                <PropagationDelay>PT43200S</PropagationDelay>
                <SOA>
                        <TTL>PT3600S</TTL>
                        <Minimum>PT3600S</Minimum>
                        <Serial>unixtime</Serial>
                </SOA>                  </Zone>


        <Parent>
                <PropagationDelay>PT9999S</PropagationDelay>
                <DS>
                        <TTL>PT3600S</TTL>
                </DS>
                <SOA>
                        <TTL>PT172800S</TTL>
                        <Minimum>PT10800S</Minimum>
                </SOA>                  </Parent>


    </Policy>


    <Policy name="lab">
```

/etc/opendnssec/kasp.xml

```
                <Description>Quick turnaround policy for lab work</Description>

                <Signatures>

                        <Resign>PT10M</Resign>
<Refresh>PT30M</Refresh>

                        <Validity>

                                <Default>PT1H</Default>
<Denial>PT1H</Denial>

                        </Validity>

                        <Jitter>PT1M</Jitter>

                        <InceptionOffset>PT3600S</InceptionOffset>

                        <MaxZoneTTL>PT300S</MaxZoneTTL>

                </Signatures>


                <Denial>

                        <NSEC/>                 </Denial>


                <Keys>

                        <!-- Parameters for both KSK and ZSK -->

                        <TTL>PT300S</TTL>
                                                <RetireSafety>PT360S</
RetireSafety>

                        <PublishSafety>PT360S</PublishSafety>

                        <!-- <ShareKeys/> -->

                        <Purge>P14D</Purge>


                        <!-- Parameters for KSK only -->

                        <KSK>
```

/etc/opendnssec/kasp.xml

```
                        <Algorithm length="2048">8</Algorithm>
                        <Lifetime>P1Y</Lifetime>
                        <Repository>utimaco</Repository>
                </KSK>

                <!-- Parameters for ZSK only -->
                <ZSK>
                        <Algorithm length="1024">8</Algorithm>
                        <Lifetime>PT4H</Lifetime>
                        <Repository>utimaco</Repository>
                        <!-- <ManualRollover/> -->
                </ZSK>                  </Keys>

        <Zone>
                <PropagationDelay>PT300S</PropagationDelay>
                <SOA>
                        <TTL>PT300S</TTL>
                        <Minimum>PT300S</Minimum>
                        <Serial>unixtime</Serial>
                </SOA>                  </Zone>

        <Parent>
                <PropagationDelay>PT9999S</PropagationDelay>
                <DS>
                        <TTL>PT3600S</TTL>
```

/etc/opendnssec/kasp.xml

```
                        </DS>
                        <SOA>
                                <TTL>PT172800S</TTL>
                                <Minimum>PT10800S</Minimum>
                        </SOA>                  </Parent>


        </Policy>
</KASP>
```

2. Edit the `zonelist.xml` file and set up ZoneList and Zone name as per the requirement.

> Refer to the *OpenDNSSEC Documentation* for more details of the `kasp.xml` file.
>
> Further visit the official OpenDNSSEC documentation at *https://wiki.opendnssec.org/* for any queries related to configuration.

5.3 Import the Updated KASP

1. Initialize the OpenDNSSEC Database with the command below.

›_ Console

```
# ods-enforcer-db-setup
```

```
root@opendnssec:~# ods-enforcer-db-setup
*WARNING* This will erase all data in the database; are you sure? [y/N] y
Database setup successfully.
root@opendnssec:~#
root@opendnssec:~#
```

Figure 11 : Set up OpenDNSSEC output

2. Start the OpenDNSSEC daemon.

›_ Console

```
# ods-control start
```

Figure 12 : Start OpenDNSSEC daemon

Refer to the OpenDNSSEC Documentation for detailed Information regarding the Enforcer and Signer at *https://wiki.opendnssec.org/*.

3. To sign zones, you need to give OpenDNSSEC information on the policy of how to sign zones. Import the policy described in `/etc/opendnssec/kasp.xml` with the below command.

›_ Console

```
# ods-enforcer policy import
```

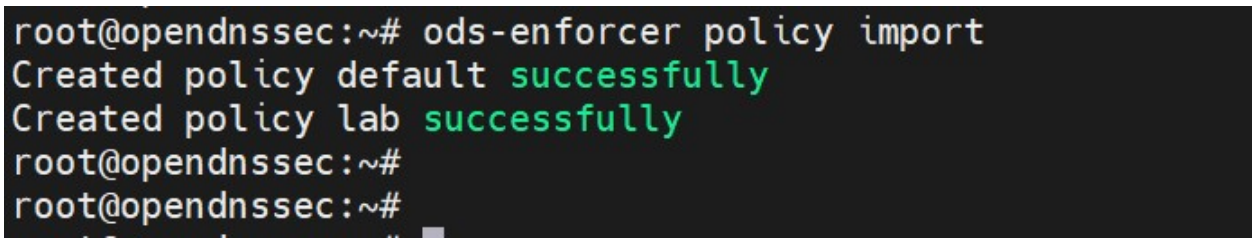

Figure 13 : Import Open DNSSEC policy output

5.4 Adding and Signing a Zone

OpenDNSSEC supports multiple ways to import and export zones. The most basic and straight forward way is the file backend. It will read an unsigned zone file, sign it and output a signed zone file.

1. You will now add a zone and sign it. OpenDNSSEC will search for unsigned zone files in `/var/opendnssec/unsigned/` folder. Create the sample zone file `test.com` with the below details.

/var/opendnssec/test.com

```
$ORIGIN test.com.

$TTL 86400

  test.com.    3600    IN      SOA     ns.test.com. username.test.com. 1

86400 7200 2419200 300 test.com.            IN      NS      ns ns
IN      A       172.23.0.10
```

2. Add the zone `test.com` with the command below. This zone file will be fetched by OpenDNSSEC, keys will be generated, and zone will be signed according to the policy setup into the `kasp.xml` file.

›_ Console

```
# ods-enforcer zone add -z test.com
```

```
root@opendnssec:~# ods-enforcer zone add -z test.com
input is set to /var/lib/opendnssec/unsigned/test.com.
output is set to /var/lib/opendnssec/signed/test.com.
Zone test.com added successfully
root@opendnssec:~#
root@opendnssec:~#
```

Figure 14 : Signing zone output

3. To display the Zone List, run the below command.

›_ Console

```
# ods-enforcer zone list
```

```
root@opendnssec:~#
root@opendnssec:~# ods-enforcer zone list
Database set to: /var/lib/opendnssec/kasp.db
Zones:
Zone:                           Policy:       Next change:              Signer Configuration:
test.com                        default       Wed Nov  9 02:03:22 2022  /var/lib/opendnssec/signconf/test.com.xml
root@opendnssec:~#
```

Figure 15 : List zone output

4. Use the command below to list the Zone keys.

›_ Console

```
# ods-enforcer key list
```

```
root@opendnssec:~# ods-enforcer key list -z test.com
Keys:
Zone:                           Keytype: State:    Date of next transition:
test.com                        KSK      publish   2023-11-17 15:09:19
test.com                        ZSK      ready     2023-11-17 15:09:19
root@opendnssec:~#
```

Figure 16 : List zone keys output

5.5 Key Rollover

1. OpenDNSSEC also provides the option for Manual rollover of the keys. To display the Rollover date of the keys for a zone, use the command below.

›_ Console

```
# ods-enforcer rollover list -z test.com
```

```
root@opendnssec:~# ods-enforcer rollover list -z test.com
Keys:
Zone:                            Keytype: Rollover expected:
test.com                         KSK      2024-11-16 01:09:19
test.com                         ZSK      2024-02-15 01:09:19
root@opendnssec:~#
```

Figure 17 : Rollover key list output

Automatic Rollover of KSK (Key-Signing Keys) and ZSK (Zone-Signing Keys) is supported with Utimaco HSM, and it happens as per the policy configured in `kasp.xml` . For more detailed information about the Key States and Key Rollover visit the official OpenDNSSEC documentation at *https://wiki.opendnssec.org/*.

2. Use the command to roll the KSK manually. This will roll the KSK key in a timely manner following the policy used for the zone test.com.

›_ Console

```
# ods-enforcer key rollover --zone test.com --keytype KSK
```

```
root@opendnssec:~# ods-enforcer key rollover --zone test.com --keytype KSK
rolling KSK for zone test.com
root@opendnssec:~#
root@opendnssec:~#
```

Figure 18 : Manually rollover KSK output

Similarly, if a user wants to manually rollover ZSK, replace `--keytype` with ZSK.

›_ Console

```
# ods-enforcer key rollover --zone test.com --keytype ZSK
```

```
root@opendnssec:~# ods-enforcer key rollover --zone test.com --keytype ZSK
rolling ZSK for zone test.com
root@opendnssec:~#
root@opendnssec:~#
```

Figure 19 : Manually rollover ZSK output

3. Verify that the key has been rolled over successfully.

›_ Console

```
# ods-enforcer rollover list
```

```
root@opendnssec:~# ods-enforcer rollover list -z test.com
Keys:
Zone:                           Keytype: Rollover expected:
test.com                        KSK      2023-11-17 03:53:17
test.com                        ZSK      2023-11-17 03:53:17
root@opendnssec:~#
```

Figure 20 : Rollover key list output

This completes the integration of OpenDNSSEC with Utimaco HSM.

6 Troubleshooting

Error	Diagnosis
LoginUser= failed: 05.12.2021 23:45:45 src/p11adm_R3.c[429] p11_login: C_Login [type=1] returned Error 0x00000102 (CKR_USER_PIN_NOT_INITIALIZED)	PKCS#11 Slot is not initialized.
The CryptoServer PKCS#11 Library R3 is not initialized. Error CKR_CRYPTOKI_NOT_INITIALIZED occurred	PKCS#11 Slot is not initialized.
hsm_session_init(): PKCS#11 module load failed: /opt/utimaco/lib/libcs_pkcs11_R3.so	1. Check if the library is present on the mentioned Path 2. Verify if the library is not corrupted
Unable to find policy default needed for adding the zone!	Verify if the policy is defined and imported.
WARNING This will erase all data in the database; are you sure? [y/N] y Error: unable to drop existing schema!	Ensure the Database schema not changed after the configuration.

Table 6: List of errors and their diagnoses

7 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: *http://hsm.utimaco.com*.

8 References

Reference	Title/Company	Document No.
[CSADMIN]	CryptoServer – csadm Manual/Utimaco IS GmbH	2009-0003
[CSTrSh]	CryptoServer Troubleshooting/Utimaco IS GmbH	M011-0008-en
[CSADMIN2]	CryptoServer_csadm_Manual_Systemadministrators.pdf	2009-0003
[CSP11Tool2]	CryptoServer_p11tool2_Manual.pdf	2012-0004
[CSPKCSM]	CryptoServer - PKCS#11 P11CAT Manual	M013-0001-en
[CSLAN5]	CryptoServerLAN_Manual_Systemadministrators.pdf	2018-0004

Table 7: References

9 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.