FORTANIX TECHNICAL DOCUMENTATION

**Fortanix**<sup>®</sup>

# Integration Guide

# FORTANIX SIGNING PROVIDER FOR TMKMS

VERSION 1.0





# TABLE OF CONTENTS

1.0	INTRODUCTION	.2
2.0	DEFINITIONS	.2
3.0	COMPILING TMKMS WITH FORTANIX DSM	.3
3.1	Compiling from Source Code (using the git command)	.3
3.2	Installing with cargo install Command	.3
4.0	SET UP FORTANIX DSM	.4
4.1	Create a Fortanix DSM Group	.4
4.2	Create an App in Fortanix DSM	.4
4.3	Generate a Key	.5
4.4	Configure tmkms	.7
4.5	Running the tmkms Application	.8
5.0	DOCUMENT INFORMATION	.9
5.1	Document Location	.9
5.2	Document Updates	.9

## 1.0 INTRODUCTION

This document describes how **Fortanix Data Security Manager (DSM)** integrates with **Tendermint Key Management System (TMKMS)** to provide remote signing capabilities that enable the signing of Ignite (previously, Tendermint) blockchain proposals and votes. With this integration, you can now create, manage, and use validator keys with Fortanix DSM SaaS. For any proposal or vote, TMKMS validates that double-signing is not being attempted and then signs the proposal or vote by invoking the sign operation offered by DSM SaaS which manages the validator keys.

## 2.0 **DEFINITIONS**

#### • Fortanix Data Security Manager

Fortanix DSM is the cloud solution secured with Intel® SGX. With Fortanix DSM, you can securely generate, store, and use cryptographic keys and certificates, as well as secrets, such as passwords, API keys, tokens, or any blob of data.

#### • Accounts

A Fortanix DSM account is the top-level container for security objects managed by the Fortanix DSM. An account is generally associated with an organization, rather than an individual. Security objects, groups, and applications belong to exactly one account. Different accounts are fully isolated from each other. *See <u>support</u> for more information.* 

#### • Fortanix Data Security Manager Security Objects

A security object is any datum stored in Fortanix DSM (for example a key, a certificate, a password, or other security objects). Each security object is assigned to exactly one group. Users and applications assigned to the group have permission to see the security object and to perform operations on it. *See <u>support</u> for more information.* 

## 3.0 COMPILING TMKMS WITH FORTANIX DSM

Refer to the main <u>README.md</u> for compiling tmkms from the source code. You will need the prerequisites mentioned under the *Section: Supported Platforms* in the readme.

The following are the two ways to install tmkms with Fortanix DSM. In these methods, you must pass the --features=fortanixdsm parameter to cargo:

#### 3.1 COMPILING FROM SOURCE CODE (USING THE git COMMAND)

tmkms can be compiled directly from the Git repository source code using the following command:

```
$ git clone https://github.com/iqlusioninc/tmkms.git && cd tmkms
[...]
$ cargo build --release --features=fortanixdsm
```

If successful, this will produce a tmkms executable located at ./target/release/tmkms.

3.2 INSTALLING WITH cargo install COMMAND

With Rust (1.40+) installed, you can install tmkms using the following command:

cargo install tmkms --features=fortanixdsm

Or you can install a specific version (recommended), using the following command:

cargo install tmkms --features=fortanixdsm --version=0.4.0

This command installs tmkms directly from packages hosted on Rust's [crates.io] service. The package authenticity is verified using the [crates.io index] which is a Git repository and by SHA-256 digests of released artifacts.

However, if newer dependencies are available, it may use newer versions besides the ones which are "locked" in the source code repository. We cannot verify whether those dependencies do not contain malicious code. If you would like to ensure the dependencies in use are identical to the main repository, please build from the source code instead.

# **Fortanix**<sup>®</sup>

## 4.0 SET UP FORTANIX DSM

This section explains how to create secure keys in Fortanix DSM.

### 4.1 CREATE A FORTANIX DSM GROUP

1. To generate/import a Consensus key, first, create a Fortanix DSM group.

Fortanix Data Security Manager	ccm.demo / Groups / View group	🗘 🖽 🛕 🛞 User Fortanix 🗸
Dashboard     Groups     Apps	a TMKMS group (No description)	
Security Objects Keys, Secrets, Certs	Activity chart	
<ul> <li>Plugins</li> <li>Tasks</li> </ul>	INFO USERS (I) APPS (II) SECURITY OBJECTS PLUGINS (II) EXTERNAL ROLES (II)	
Audit Log Settings		+ NEW USER
	Quorum approval policy adds an extra level of protection to sensitive key operations.	
	Set with the set of the	
	Change group     Destroy Key, Delete key     Deathoate Key	

FIGURE 1: CREATE GROUP

## 4.2 CREATE AN APP IN FORTANIX DSM

Create an app in Fortanix DSM of type **REST API** and copy the app's **API KEY**. This API Key is

added to the tmkms.toml configuration file later. *Refer to Section 4.4*.

#### SIGNING PROVIDER FOR TMKMS

Fortanix Data Security Manager	<u> </u>			Q
<ul> <li>Dashboard</li> <li>Groups</li> <li>Apps</li> <li>Security Objects</li> </ul>	ccm-demo / Apps / View app TMKMS default / REST API UUID: 072145a5-f57b-43f0-9e1e-13d6 Activity chart	Copy the API key TMKMS RESTAPI API KEY	X USERNAME/PASSWORD	×
<ul> <li>keyt servet Leve</li> <li>Users</li> <li>Plugins</li> <li>Tasks</li> <li>Audit Log</li> <li>Settings</li> </ul>	INFO SECURITY OBJECTS USAC Enabled ③ <i>TMKMS dsm Ø</i> API Key SDK provides tools, compilers, head	API Key: MDcyMTQ1YTUtZjU3Yi00M MxY2Y40jFGMkp5ZmNpU HNKNHZ1R1NSU2M4d0kw b1p6ZGMxU3ZhblgweUdU TRR	M2YwLTIIMWUtMTNkNji0Yj IGh5dUZxQUp4QTFreUh6T vQVd5TVNVZDhkdURJVVBk JQy1NYjBadFZzYVpGS0pFS	<b>GS C</b> Lest updated: 6:15 pm 2022, 1:56 pm
	help system that developers can us COPY API KEY Change authentication method > Groups 孟 TMKMS group Default group P	CANCEL C	Dermissions:	

FIGURE 2: CREATE AN APP AND COPY THE API KEY

## 4.3 GENERATE A KEY

tmkms supports keys for accounts and consensus signing. The type of key must be **EC CurveEd25519** for the Consensus key and **Secp256k1** for the Account key. This guide explains the signing procedure using the Consensus key.

Generate a key called **consensus-key** in the same group created in *Section 4.1* so that the API key of the app created in *Section 4.2* can be used to access this key. The key type must be **EC** of Curve **Ed25519**. This key is used by the validator for consensus signing.



#### SIGNING PROVIDER FOR TMKMS

Fortanix Data Security Manager	📮 🗸 🔘 User Fortanix 🗸
	ccm-demo / Groups / View group
② Dashboard	
	(No description)
🖻 Apps	UUID. 4cb0f6cb-243a-4c0f-b7de-8a5b3b9e5baa 🗋
Security Objects Keys, Secrets, Certs	Activity chart
③ Users	
🐡 Plugins	INFO USERS (6) APPS (7) SECURITY OBJECTS PLUGINS (7) EXTERNAL ROLES (7)
🕐 Tasks	Q Search
🖉 Audit Log	ADD SECURITY OBJECT 📋 DELETE SELECTED 🔗 DESTROY SELECTED 🕆 DISABLE LOGGING 🗋 ENABLE LOGGING
් Settings	
	NAME ▲     KCV     KEYOPS     CREATED BY     CREATED     TYPE     SIZE/CURVE     EXPIRES     @
	Consensus-key . (1) User Fortanix (You) a few seconds ago EC O Ed25519 Never
	Items per page: 10 ▼ (《) < PREVIOUS

FIGURE 3: CONSENSUS KEY

2. Copy the Key ID or the key name of this key to add it to the tmkms.toml configuration file later. *Refer to Section 4.4*.

Fortanix Data Security Manager	
	ccm-demo / Security Objects / View security object
② Dashboard	pre-active active deactivated destroyed
盂 Groups	
F Apps	
Security Objects Keys, Secrets, Certs	
(2) Users	UUID: 6dd2/60a-158e-48bc-8c5e-b2850b1d58de
- Plugins	Activity chart COPY UUID
Vg	COPY GOOGLE EKMS URI
🕑 Tasks	INFO ATTRIBUTES/TAGS KE
🔄 Audit Log	
් Settings	Enabled ⑦

#### FIGURE 4: COPY KEY ID

Alternatively, you can also import an existing Tendermint key.

To import an existing Tendermint key, use the following script to convert a Tendermint key to Fortanix DSM accepted key format.

```
#!/bin/bash
# Usage: tendermint-ed25519.sh <input-tendermint> <output-private-
p8der> <output-public-p8der>
gokey=$(jq -r .priv_key.value $1 | base64 -d| xxd -p -c 64)
echo 302e 0201 0030 0506 032b 6570 0422 0420 "${gokey:0:64}" | xxd -p
-r > $2
echo 302a 3005 0603 2b65 7003 2100 "${gokey:64}" | xxd -p -r > $3
```

#### 4.4 CONFIGURE tmkms

To perform the setup, tmkms needs a configuration file that contains the authentication details needed to authenticate to Fortanix DSM with an API key.

The file tmkms.toml contains this configuration. You can specify the path to the config with either -c /path/to/tmkms.toml or tmkms will look in the current working directory for the same file. For example:

```
[[providers.fortanixdsm]]
api_endpoint = "https://<fortanix_dsm_url>"
api_key =
"Nzk5MDQ3ZGUtN2Q2NS000TRjLTgzMDMtNjQwMTlhYzdmOGUzOlF1SU93ZXJsOFU4VUdEWEd
QMmx1dFJOVj1vMTRsd3lhNnVDNVNhVkpZOVhzYVgyc0pOVGRQVGJ0RjZJdmVLMy00X05iTEh
xMkowamF3UGVPaXJEWEd3"
signing_keys = [
        { chain_ids = ["$CHAIN_ID"], type = "account", key_id = "72e9ed9e-
9eb4-46bd-a135-e78ed9bfd611" },
        { chain_ids = ["$CHAIN_ID"], type = "consensus", key_name = "My Key"
},
]
```

#### Where,

• api\_key is the Fortanix DSM app API Key. Refer to *Section 4.2* for steps to get the app API Key.

#### SIGNING PROVIDER FOR TMKMS

key\_id and key\_name is the Fortanix DSM key UUID and key name respectively. You can
either pass the key UUID or key name for signing. Refer to *Section 4.3* for steps to create a
Consensus key or import an existing key.

## 4.5 RUNNING THE TMKMS APPLICATION

The tmkms.toml configuration file now has the required details:

- The Fortanix DSM App API Key to authenticate to Fortanix DSM.
- The Fortanix DSM Key ID/Key name is used by the Validator for consensus signing.
- 1. Start tmkms using the following command:

```
$ tmkms start
```

This will read the configuration from the tmkms.toml file in the current working directory.

or

To explicitly specify the path to the configuration, use the -c flag:

```
$ tmkms start -c /path/to/tmkms.toml
```

Run the tmkms application. Go to the detailed view of the consensus-key and in the Activity
 Logs section notice that this key is used by the Validator for consensus signing.

Fortanix Data Security Manager	pre-active deactivated destroyed		
	consensus-key	+ NEW OBJECT	
② Dashboard			
品 Groups	UUID: a5f48dc6-a512-45a8-8e90-737076e71d1f DCOPY ID 🗸 🕜		
F Apps	Activity chart	×	
Security Objects Keys, Secrets, Certs			
Ø Users     Ø	INFO ATTRIBUTES/TAGS KEY ROTATION KEY ACCESS JUSTIFICATION		
Plugins	Enabled ⑦		
Ο Tasks	(No description) 🖉		
🖉 Audit Log		Activity Logs 🕝 Last updated: 6:52 pm 🛽 DOWNLOAD LOCS	
ල් Settings	Type EC ⊙ Curve Ed25519	Last used: May 3rd 2022, 10:51:34 pm	
	Comparent contraction of the second s	Keep detailed log for the object	
	Public key published	SDKMS will keep a full audit log for this object. You can disable logging to increase performance.	
	Group	✓ App "TMKMS" used key "consensus-key" to perform "Signing" operation Ma	
	品 TMKMS group	y 3, 2022 10:51:34 pm	

FIGURE 5: EXECUTION LOG FOR KEY

# 5.0 DOCUMENT INFORMATION

#### 5.1 DOCUMENT LOCATION

The latest published version of this document is located at the URL: <u>https://support.fortanix.com/hc/en-us/articles/6999736664468-Using-Fortanix-Signing-Provider-for-Tendermint-KMS</u>

## 5.2 DOCUMENT UPDATES

This document will typically be updated on a periodic review and update cycle. For any urgent document updates, please send an email to: <a href="mailto:support@fortanix.com">support@fortanix.com</a>

© 2016 – 2023 Fortanix, Inc. All Rights Reserved.

Fortanix® and the Fortanix logo are registered trademarks or trade names of Fortanix, Inc.

All other trademarks are the property of their respective owners.

**NOTICE**: This document was produced by Fortanix, Inc. (Fortanix) and contains information which is proprietary and confidential to Fortanix. The document contains information that may be protected by patents, copyrights, and/or other IP laws. If you are not the intended recipient of this material, please destroy this document and inform <u>info@fortanix.com</u> immediately.