



|  |
| --- |
| X-Road: Configuration Proxy Architecture  **Technical Specification** |
| Version: 1.2  21.10.2015  12 pages  Doc. ID: ARC-CP |

|  |  |  |  |
| --- | --- | --- | --- |
| Date | Version | Description | Author |
| 05.06.2015 | 0.1 | Initial version | Ilja Kromonov |
| 03.07.2015 | 0.2 | Added deployment diagram | Ilja Kromonov |
| 08.07.2015 | 0.3 | Added configuration distribution workflow description; updated deployment diagram | Ilja Kromonov |
| 09.09.2015 | 0.4 | Updated diagrams, editorial changes, technology matrix | Ilja Kromonov |
| 16.09.2015 | 1.0 | Editorial changes made | Imbi Nõgisto |
| 24.09.2015 | 1.1 | Reference added | Kristo Heero |
| 21.10.2015 | 1.2 | SSCD and passwordstore related information added | Ilja Kromonov |

Table of Contents

[1 Introduction 5](#_Toc433272231)

[1.1 Overview 5](#_Toc433272232)

[1.2 Terms and Abbreviations 5](#_Toc433272233)

[1.3 References 5](#_Toc433272234)

[2 Component View 6](#_Toc433272235)

[2.1 Web Server 6](#_Toc433272236)

[2.2 Configuration Processor 6](#_Toc433272237)

[2.3 Signer 6](#_Toc433272238)

[2.4 Configuration Client 7](#_Toc433272239)

[2.5 Password Store 7](#_Toc433272240)

[2.6 SSCD 7](#_Toc433272241)

[3 Interfaces 8](#_Toc433272242)

[3.1 Downloading Configuration 8](#_Toc433272243)

[4 Configuration Distribution Workflow 9](#_Toc433272244)

[5 Technology Matrix 11](#_Toc433272245)

[6 Deployment View 12](#_Toc433272246)

License

This work is licensed under the Creative Commons Attribution-ShareAlike 3.0 Unported License. To view a copy of this license, visit http://creativecommons.org/licenses/by-sa/3.0/.

# Introduction

This document describes the architecture of the X-Road configuration proxy. For more information about X-Road and the role of the configuration proxy see [ARC-G ].

This document presents an overview of the components of the configuration proxy and the interfaces between these components. It is aimed at technical readers who want to acquire an overview of inner workings of the configuration proxy.

## Overview

The configuration proxy acts as an intermediary between X-Road servers in the matters of global configuration exchange.

The goal of the configuration proxy is to download an X-Road global configuration from a provided configuration source and further distribute it in a secure way. Optionally, the downloaded global configuration may be modified to suit the requirements of the configuration proxy owner.

The configuration proxy can be configured to mediate several global configurations (from multiple configuration sources).

## Terms and Abbreviations

* Configuration proxy instance – a process within the configuration proxy that deals with distributing the global configuration files of a specific X-Road instance.

## References

ARC-G Cybernetica AS. X-Road Architecture.

PR-GCONF Cybernetica AS. X-Road: Protocol for Downloading Configuration.

PKCS11 Cryptographic Token Interface Standard. RSA Laboratories, PKCS#11.

UG-CP Cybernetica AS. X-Road v6 Configuration Proxy Manual.

# Component View

Figure 1 shows the main components and interfaces of the X-Road configuration proxy. The components and the interfaces are described in detail in the following sections.

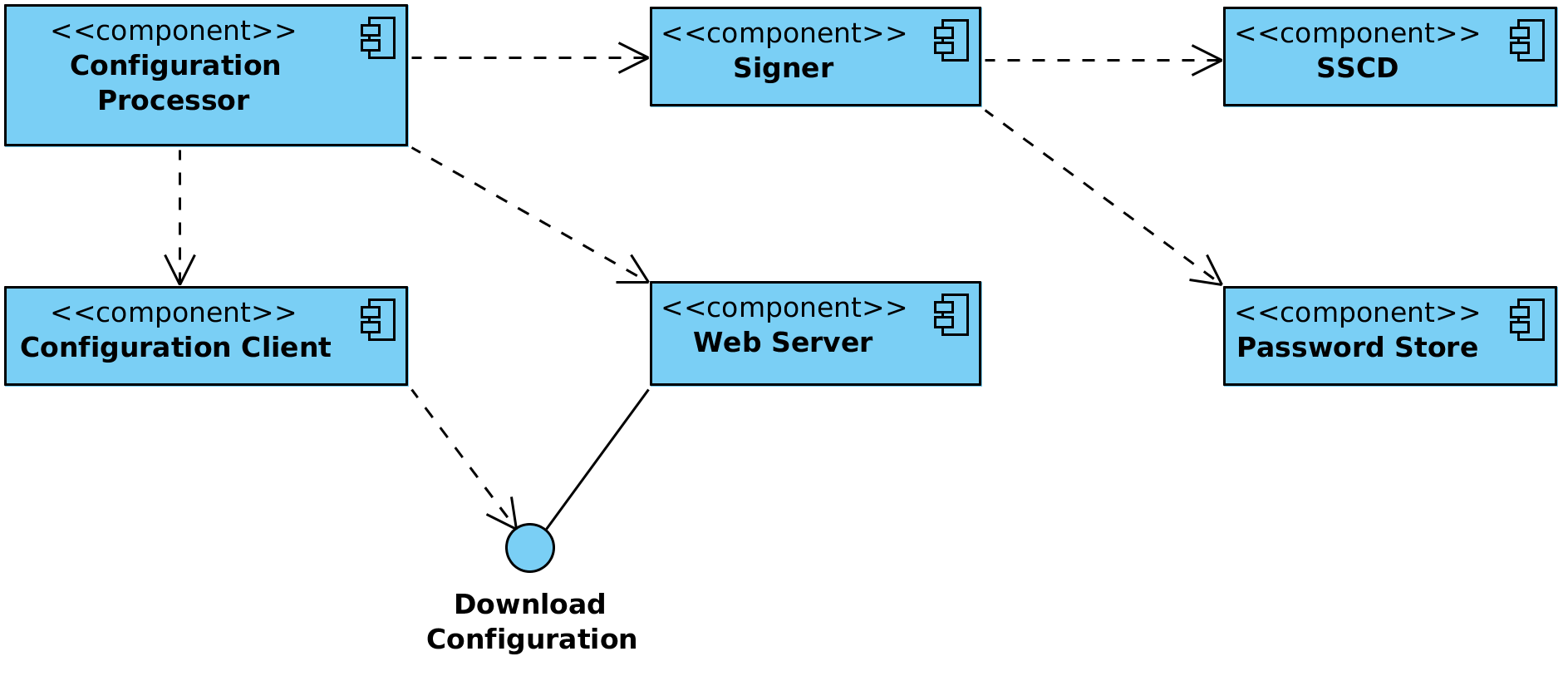


Figure 1. Configuration proxy component diagram

## Web Server

The global configuration files the configuration proxy generates need to be made available to configuration clients. The HTTP-based protocol used for downloading configuration is described in [PR-GCONF ]. Technically, the configuration consists of a set of files that are shared out using standard web server (nginx[[1]](#footnote-1) web server is used). The configuration processor component prepares and signs the configuration files and then copies them to the web server's output directory where they are distributed via standard means. See Section 3.1 for details on configuration distribution.

## Configuration Processor

A configuration processor is responsible for initiating the download of global configuration files for the configured X-Road instance and preparing them for distribution to configuration clients.

A cron job configured on the server periodically (every minute) starts the configuration processor, which sequentially performs it's functions for all configured configuration proxy instances.

## Signer

The signer component is responsible for managing the keys and certificates used for signing the global configuration. Signer is called from the configuration processor to create the signature for the configuration.

## Configuration Client

The configuration client is responsible for downloading remote global configuration files. The source location of the global configuration is taken from the anchor file that the configuration processor provides when invoking the configuration client.

## Password Store

Stores security token passwords in a shared memory segment of the operating system that can be accessed by the signer. Allows security token logins to persist, until the configuration proxy is restarted, without compromising the passwords.

## SSCD

The SSCD (Secure Signature Creation Device) is an optional hardware component that provides secure cryptographic signature creation capability to the signer.

The SSCD needs to be a PKCS #11 (see [PKCS11]) compliant hardware device that can be optionally used by the configuration proxy for signing the generated global configuration files it generates. The use of the interface requires that a PKCS #11 compliant device driver is installed and configured in the configuration proxy system.

# Interfaces

## Downloading Configuration

Configuration clients download the generated global configuration files from the configuration proxy (protocol used for downloading configuration is described in [PR-GCONF ]). The configuration proxy uses this interface to download global configuration files from a remote configuration source.

The configuration download interface is a synchronous interface that is provided by the configuration proxy. It is used by configuration clients such as security servers and other configuration proxies.

The interface is described in more detail in [ARC-G].

Should the configuration proxy fail to download the global configuration files from a remote configuration source, no updated configuration directory will be generated for that source's configuration proxy instance. The old configuration directory will remain valid until it's validity period expires.

# Configuration Distribution Workflow

X-Road configuration proxy periodically downloads the global configuration from configured sources. Each global configuration consists of a number of XML files (see [PR-GCONF ] for detailed information about configuration structure). The configuration proxy then redistributes the downloaded configuration files to other configuration clients.

Configuration files are distributed in accordance with the protocol for downloading configuration (see Section 3.1 ). MIME messages are generated to represent configuration directories of the global configurations being distributed.

The configuration proxy can distribute files from as many configuration sources as necessary. For each configuration source a configuration proxy instance is configured.

The following process is performed for each configuration proxy instance.

1. A cron job sends a request to start the configuration processor for the given configuration proxy instance.
2. Configuration processor component calls the configuration client to download the global configuration files from the configured source.
3. Configuration processor component generates configuration directory referencing the downloaded files.
4. Configuration processor component sends a signing request (containing hash of the directory) to the signer component. Signer signs the hash and responds with the signature.
5. Configuration processor component updates the configuration directory to contain the received signature.
6. Configuration processor component moves the configuration files to the distribution directory of the web server.
7. Security servers make HTTP requests and download the configuration.

This process is illustrated in the sequence diagram in figure 2.

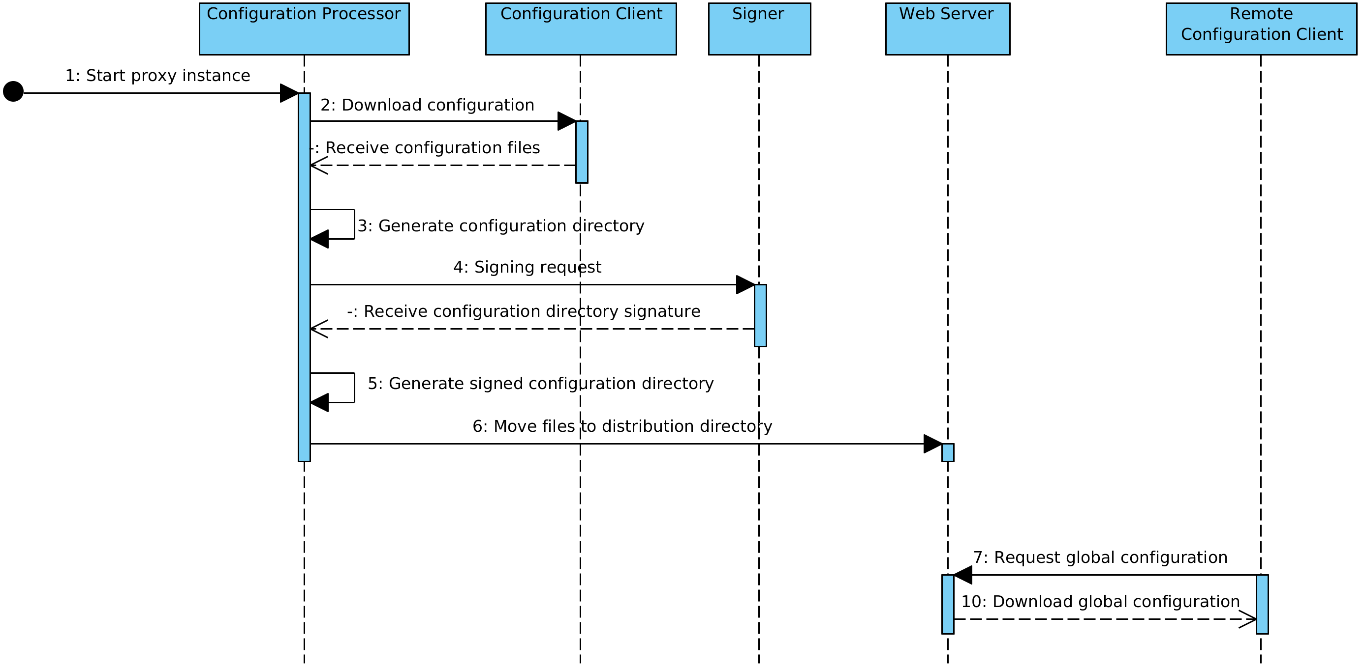


Figure 2. Configuration creation sequence diagram

# Technology Matrix

Table 1 presents the list of technologies used in the configuration proxy and the mapping between technologies and configuration proxy components.

Table 1. Technology matrix of the configuration proxy

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Technology** | **Web Server** | **Configuration Processor** | **Signer** | **Configuration Client** |
| Java 8 |  | X | X | X |
| Logback |  | X | X | X |
| Akka 2.X |  | X | X |  |
| nginx | X |  |  |  |
| upstart | X | X | X | X |
| PKCS #11[[2]](#footnote-2) |  |  | X |  |

# Deployment View

The configuration proxy is deployed on a separate server on the configuration provider's side and, optionally, on the configuration client's side as well.

Figure 3 shows a possible deployment scenario for the configuration proxy. In the presented scenario the instance #1 central server is distributing it's internal and external configuration through a configuration proxy to instance #1 members and an instance #2 configuration proxy, respectively. The instance #2 central server distributes it's internal configuration directly to instance #2 members and it's external configuration directly to instance #1 members.

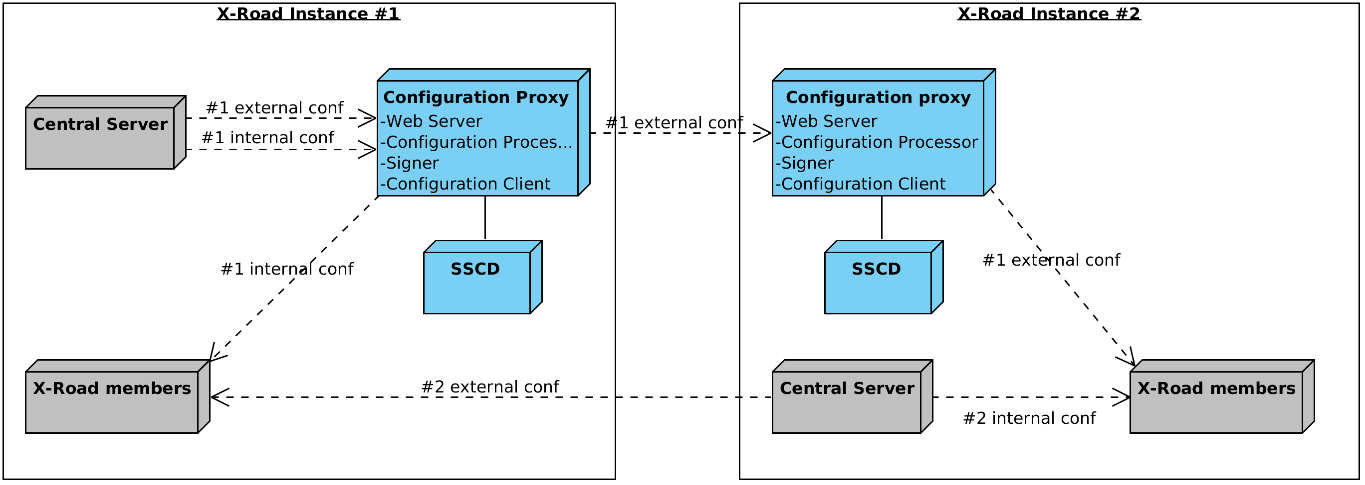


Figure 3. Configuration proxy deployment example

Optionally, an SSCD can be connected with the configuration proxy if global configuration signatures are to be provided by a cryptographic hardware device.

For detailed information on installing and configuring the configuration proxy refer to the configuration proxy manual [UG-CP].

1. See <http://nginx.org/> for details. [↑](#footnote-ref-1)
2. The use of hardware cryptographic devices requires that a PKCS #11 driver is installed and configured in the system. [↑](#footnote-ref-2)