

---

# HSM Proxy Product Overview

Version 0.4.0

Frank Cornelis

Copyright © 2013 FedICT

## Abstract

This document gives an overview of the HSM Proxy product.

1. Introduction .....	1
2. Architecture .....	1
3. Secure implementation .....	2
A. HSM Proxy Overview License .....	3
B. HSM Proxy Project License .....	3

## 1. Introduction

Proper key management has always been a challenge in the context of service oriented architectures. Especially given recent trends where services are being deployed on cloud infrastructure, it is important to have a maintainable key management system as part of your service offering.

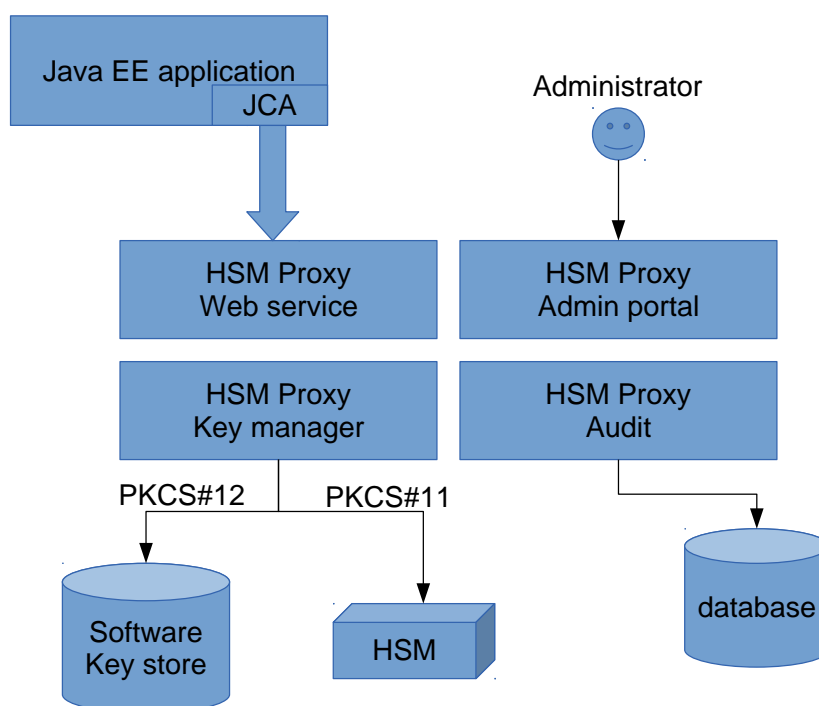
Another important aspect when deploying HSM infrastructure is ease of integration within your Java EE application. Where most HSM deployments come with feature-rich PKCS#11 libraries, such native libraries are often perceived as hostile by the Java developer who is used to automated dependency management systems like Maven.

Finally cloud infrastructure is also challenging in the area of auditing where you want to have an audit record for every key usage performed by the system.

Even if you don't want to immediately invest in rather expensive hardware security modules, an HSM Proxy solution is still meaningful as this allows you to manage your critical crypto keys by a different (UNIX) process than the process hosting your application's application server. This reduces the risk that a hacked application immediately gives access to your private key material.

## 2. Architecture

The basic architecture of the HSM Proxy product is depicted in [Figure 1, "HSM Proxy Architecture"](#)



**Figure 1. HSM Proxy Architecture**

Central in the HSM Proxy architecture is the key management. The HSM Proxy can load keys from both software key stores (PKCS#12 format) as well as hardware security modules (via the PKCS#11 interface). The HSM Proxy features a key alias mapping mechanism. This allows your application to choose its own key aliases, independent of the actual key aliases within the different internal key stores.

Towards your applications the HSM Proxy exposes a secured web service. To ease integration into your Java (EE) applications, the HSM Proxy project also delivers a JCA security provider.

Via an administrator portal, the HSM Proxy administrators can easily manage the different key stores and register applications.

The HSM Proxy also features an audit system that creates an audit record for each created signature.

### 3. Secure implementation

The HSM Proxy is implemented as a Java EE 6 application targeting JBoss AS 7.2 and JBoss EAP 6.1 application servers. By minimizing the dependency on JBoss specific features, the HSM Proxy can easily be ported to other application servers.

Special attention was given to a secure implementation of the product. The HSM Proxy features a Role Based Access Control (RBAC) mechanism at all application layers (Model, View, and

Control). The web service WS-Security security mechanism and RBAC security mechanism received in-depth testing using the Arquillian testing framework.

## A. HSM Proxy Overview License



This document has been released under the [Creative Commons 3.0](http://creativecommons.org/licenses/by-nc-nd/3.0/) [http://creativecommons.org/licenses/by-nc-nd/3.0/] license.

## B. HSM Proxy Project License

The HSM Proxy Project source code has been released under the GNU LGPL version 3.0.

This is free software; you can redistribute it and/or modify it under the terms of the GNU Lesser General Public License version 3.0 as published by the Free Software Foundation.

This software is distributed in the hope that it will be useful, but WITHOUT ANY WARRANTY; without even the implied warranty of MERCHANTABILITY or FITNESS FOR A PARTICULAR PURPOSE. See the GNU Lesser General Public License for more details.

You should have received a copy of the GNU Lesser General Public License along with this software; if not, see <http://www.gnu.org/licenses/>.

