securosys

Securosys Transaction Security Broker

Unparalleled transaction security

Introduction

Securosys *Hardware Security Modules (HSM)* are not only optimized for the physical protection of private key material like most legacy HSMs; Securosys HSMs provide control of the keys' usage with specific and sophisticated authorizations, which is essential for the security of modern financial applications. *Smart Key Attributes (SKA)* allow fine-grained policies to be defined for different actions, with keys based on groups, quorums and time restrictions, and any combination of these.

Securosys *Transaction Security Broker (TSB)* makes the implementation of SKAs much easier thanks to its REST API and internal state management. It runs as a standalone engine, connects to an external database instance and integrates the SKA-enabled Securosys HSM – and is thus uncritical for security, since all security relevant operations are carried out in the HSM.

Details

The TSB integrated with the SKA-enabled Securosys HSM provides the most granular control over key actions and operations. It allows the finance organization to set highly customizable policies for authorizing operations and transactions, blocking or unblocking the keys, and changing the policies themselves. The use-cases range from *n* to *m* quorums, time-locks that allow systems to trigger alarms and block key operations, to time-outs that ensure that suspended transaction requests cannot be misused in the future, and any combinations of these. Approval can happen on a mobile, desktop or physical cryptographic device and can also be protected by the HSM's keys.

Approval process

The approval process is typically as follows:

1) An approval for the usage of a key for a transaction payload is requested
2) The HSM checks key attributes and returns the approval policies along with the payload and timestamp signature to the TSB
3) The business application fetches the approval request from the TSB and broadcasts it to the approval clients
4) The TSB waits for the approvals until the policy is met, then sends the required approval data together with the payload to the HSM
5) The HSM checks the authorization data against the key attributes (SKA), the specific payload, and optionally the signed timestamp
6) If the criteria are met, the HSM signs the payload and returns the signature

Benefits

HSM Security

- Keys are never exposed outside of the HSM
- Tamper protection during transport, storage and operation
- Two true random number generators for hardware with high entropy
- Highest availability
- Designed, developed and manufactured in Switzerland

TSB Simple Setup

- Via REST API
- Available in a Docker container

Application Performance

- Hardware accelerated digital signing, up to 4000 RSA signatures at 2048 bits per second
- Handle larger key sizes without severe performance loss

Policies

Policies are a set of one or multiple rules. The following authorization rules can be defined:

- Quorum - *n* out of *m* authorization is required
- Delay - the minimum time between the receipt of the authorization request by the HSM and the actual signing of the payload
- Timeout - the maximum time between triggering the request and its authorization

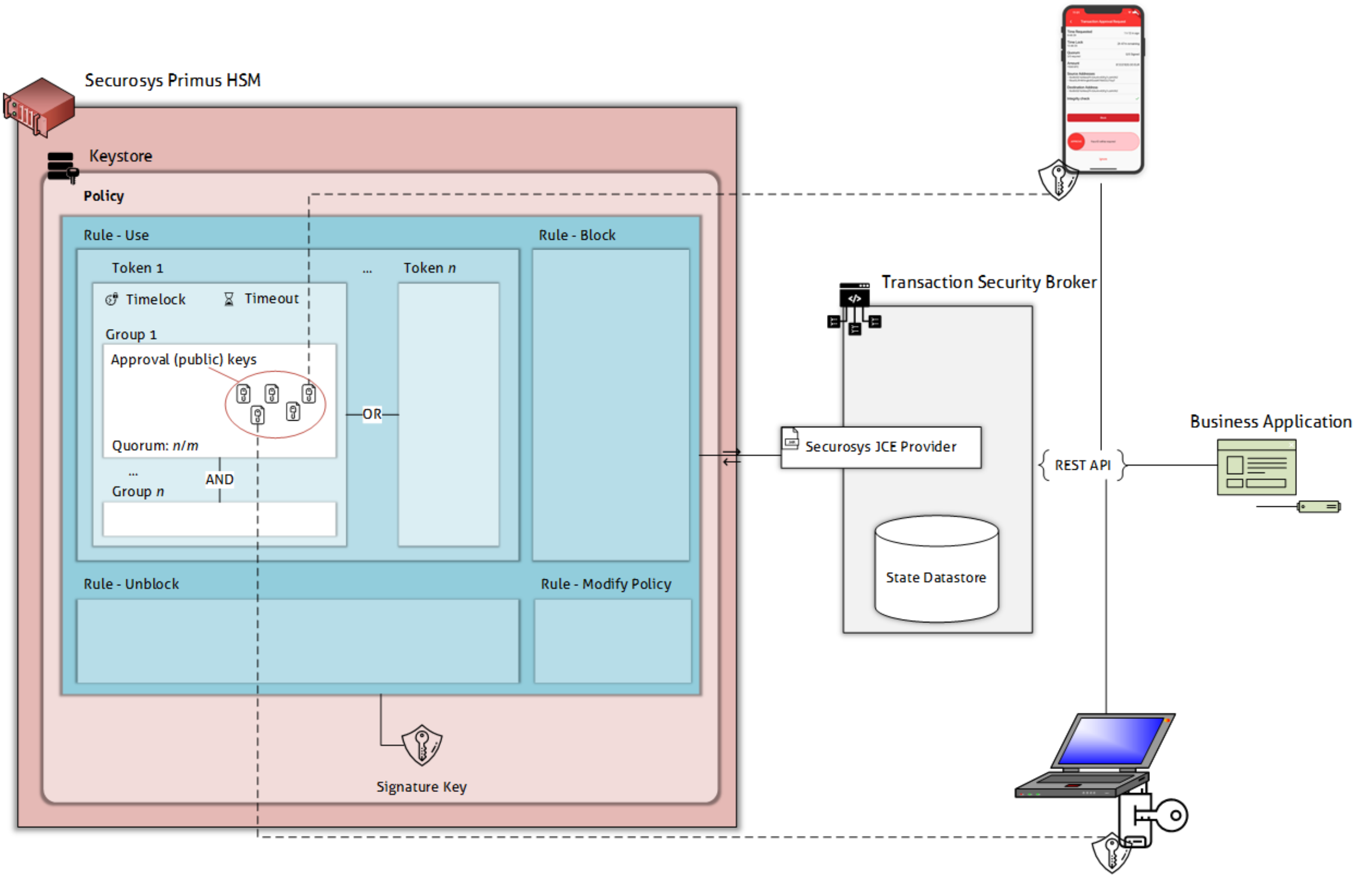
There can be multiple rules for each key allowing various levels of authorization for each key.

In addition, different rules may apply to various operations with the key:

- Usage (e.g. signature, encryption)
- Key blocking
- Key unblocking
- Change of the attributes (policies)

All of the rules can be combined and assigned down to the individual key.

Architecture



The *Transaction Security Broker* is not a standalone product but requires Primus HSM and the relevant license.