Marko Vukolic <mvu@zurich.ibm.com>

29 September 2010

KMIP Server-to-server: recapitulation

Outline

- Server-to-server (s2s) use cases
  - *presented on an earlier F2F meeting in 2009*
- Summary of steps taken in KMIPv1 and next steps
  - *presented in March '10 during one of the TC calls*

Server to server (s2s) use cases

- Backup, Data Loss Prevention
- Load balancing/Delegation
- Propagating key material closer to endpoints, e.g.,
  - Example 1 (A retail store)
    - A retail store operation with each store relying on encrypted storage
    - network connectivity with the central key management server (CKMS) not reliable
    - small subset of the keys needed to be served locally, but the management is at CKMS
    - Keys at local key-management servers could be read-only, with pre-allocated usage or lease time
    - The local server needs to communicate with the CKMS
  - Example 2 (e-commerce websites)
    - Multiple e-commerce websites centrally managed (CKMS)
    - Some keys need to be pushed down from CKMS (readable locally), i.e., with CKMS exporting the keys
- Propagating key material updates towards the central key manager
  - A large multinational bank needs the information about cryptographic material from Location B in central Location A (but not vice versa)

Server to server (s2s) use cases (cnt'd)

- Business-partner data exchange
  - Propagation of keys between KMIP servers to facilitate business partner data exchange
- Partitioning and M&A
  - A KMIP server needs to be partitioned into more servers
  - A company acquires another and cryptographic objects from different KMIP servers need to be merged
- KMIP server acting as the gateway/proxy
  - A less capable KMIP server may need to proxy client's request to the more capable KMIP server (e.g., to interact with a PKI)
- Replication (fault-tolerance)
- Exchange of different server policies and their enforcement

Summary

- Useful operations are optional (Notify, Put)
  - **KMIPv2:** make Notify and Put mandatory for a s2s compliant KMIP server
- **KMIPv2:** More attributes are needed
  - e.g., Master, Slaves, Backup flag, Working server (denoting the origin of the backed-up objects)
- Other issues (**KMIPv2**)
  - UUID, Name collisions across different servers
  - Locate does not return an indication to the client whether there are more objects matching the query, nor the means to "resume" such a Locate (**KMIPv2**)
- Bulk export/import can be only partially emulated (using batched operations)
  - **KMIPv1.0 fix:** support for "Get All Attributes"
- The behavior of Put when Replaced Unique Identifier ruuid is specified, but the object with ruuid does not exist on the remote end needs to be specified (**fixed in KMIPv1.0**)
- Notify does not support notification about deleted attributes (**fixed in KMIPv1.0**)
- Other issues (**fixed in KMIPv1.0**)
  - Cannot Locate all
  - Locate supports wildcards only for Name and Object group

Next steps summary 

- Server representation/registration (cf. client registration)
- Define additional attributes
  - Master/Slave, Backup flag, Working server*, …
  - Interact with AC (e.g., Slave permissions)

*attr names are provisional here

- Say something about UUID, Name collisions across servers
- Provide means to continue/resume a Locate
- *Examine the possible additional use-cases with a wider TC participation*