Marko Vukolic mvu@zurich.ibm.com, Robert Haas rha@zurich.ibm.com

23 November 2010

KMIP Server-to-server: use-cases and status

Server to server (s2s): Focal use cases

1. Propagating key material closer to endpoints, e.g.,
   - Example 1 (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
2. 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)
3. Business-partner data exchange
4. Propagation of keys between KMIP servers to facilitate business partner data exchange
5. Partitioning
   - A KMIP server needs to be partitioned into more servers
6. 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)

Server to server (s2s): Deferred and excluded use cases

- Deferred use cases:
  - Replication (fault-tolerance)
  - Exchange of different server policies and their enforcement
  - M&A
    - A company acquires another and cryptographic objects from different KMIP servers need to be merged

- Excluded use cases (to be handled via mechansims outside KMIP):
  - Backup, Data Loss Prevention
  - Load balancing/Delegation

KMIP Implications of s2s: 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
- 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)
  - support for "Get All Attributes" (**fixed in KMIPv1.0**)
- 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 

- Write up detailed scenarios around focal use-cases
- Address server representation/registration (cf. client registration)
  - "Entity" to represent servers as well, incl. contact info (IP address) to facilitate communication
- Define additional attributes
  - Master/Slave
  - Interact with AC (e.g., Slave permissions). Can a Slave perform read-only, or pre-allocated usage, or …
- Say something about UUID, Name collisions across servers
- Provide means to continue/resume a Locate