National update: Simple Signing Device

Presentation of SSD on TF-EMC2 on 22.9.2010 Copenhagen

CSC – Tieteen tietotekniikan keskus Oy
CSC – IT Center for Science Ltd.

TOC

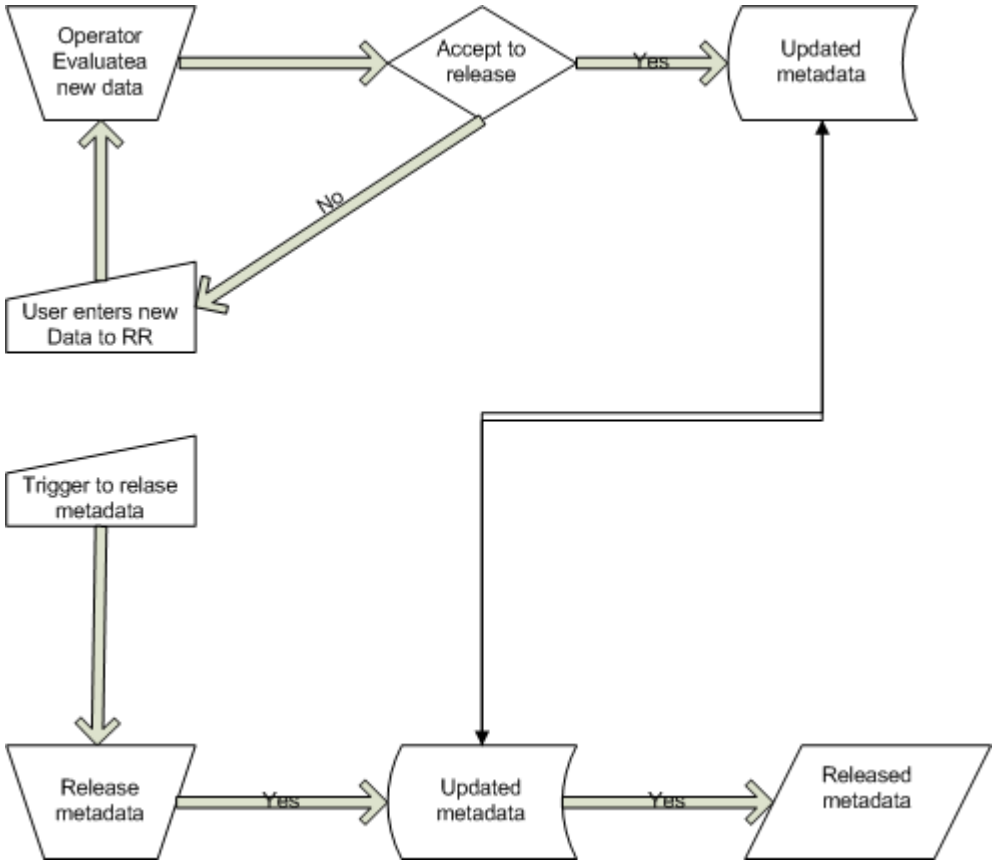
- Background
- Reasons
- Solution

Us – Haka Group

- Manne Miettinen
- Mikael Linden
- Arto Tuomi
- Timo Mustonen
- Janne Lauros
- Mika Suvanto has left the building

Haka metadata process

- Changes to metadata are usually new idp/sp, certificates, contact information
- Metadata change request usually 3-5 times a week
- Metadata release is triggered usually by ValidUntil, agreed idp/dp deployment, Certificate change
- Usually 1-.2 times a week
- Not something you would call dynamic



Haka metadata process

- View we use ’daily’ for making updates to next metadata to be released

Oulun yliopisto	oulu.fi	Haka	1.3, 2.0				19.7.2010 15:15	enabled
Pirkanmaan ammattikorkeakoulu	piramk.fi	Haka	1.3				5.3.2010 12:37	**disabled**
Pohjois-Karjalan ammattikorkeakoulu	pkamk.fi	Haka	1.3				18.12.2009 10:38	enabled
Vaasan ammattikorkeakoulu	puv.fi	Haka	1.3, 2.0				22.6.2010 15:20	enabled
Rovaniemen ammattikorkeakoulu	ramk.fi	Haka	1.3, 2.0				9.11.2009 08:37	enabled
Saimaan ammattikorkeakoulu	saimia.fi	Haka	1.3, 2.0				23.11.2009 13:10	enabled
Satakunnan ammattikorkeakoulu	samk.fi	Haka	1.3, 2.0				14.5.2010 10:43	enabled
Savonia-ammattikorkeakoulu (vanha)	savonia-amk.fi	Haka	1.3				21.1.2009 14:46	**disabled**
Savonia-ammattikorkeakoulu	savonia.fi	Haka	1.3, 2.0				31.5.2010 08:43	enabled
Sibelius-Akatemia	siba.fi	Haka	1.3, 2.0				4.5.2010 11:58	enabled

- View we use for ’weekly’ metadata releases

Following resources/homeOrgs may have changed since last Haka metadata update:

- Finnish Online University of Applied Sciences Portal

Published metadata file last updated: 6.9.2010 09:37:45

Following resources/homeOrgs may have changed since last Haka/Kalmar metadata update:

- Finnish Online University of Applied Sciences Portal

Published metadata file last updated: 6.9.2010 09:37:45

View & Diff

File	XML diff
haka-metadata.xml	(diff)
haka-kalmar-metadata.xml	(diff)
arp.sites.xml	(diff)
attribute-filter.xml	(diff)
WAYF config	(diff)
md5sum.idp	
md5sum.sp	

Configuration Files

Write Haka metadata
Write Kalmar metadata

Continue to publishing

... at pelto.funet.fi

Haka metadata process

- What I as operator feel is good about it
  - The release process is very static
    - No moving parts. It is very hard to break the system. Basically, what the client sees, is a file sitting on address http://haka.funet.fi/fed/haka-metadata.xml and that is what it actually is too, no automation.

- What I as operator feel is bad about it
  - The release process is very static
    - ValidUntil.. Having low ValidUntil value just does not suite the model. Of course fixing this does not cause major overhaul to system but does require some modifications.

How did we handle the key then??

Currently we are storing the private key in a java keystorage protected by alias and password combination hardcoded to php code…

.. Which is okay if we trust our staff never to make any mistakes but..

In an Audit.. How do we

- Explain the origin of the key. Has it been generated by nn on a machine named zz. How safe was the process? Is it even unique?
- Who are the people that have had access to the key in plain text during it’s lifecycle?
- Where the key is actually located? Has it been sent in emails, copied to usb sticks?
- We cannot answer convincingly to any of those questions

SSD – Goals

- How to ensure that anyone (e.g. federation operator staff, laptop/server support staff, intruders) does not accidentally or intentionally compromise the private key?
- How to monitor the use of private key (i.e. who has signed what) without introducing tedious ceremonies?
- How to spot if the private key is compromised? Again, without tedious ceremonies?
- And how to do this with minimum effort (there aren't that many of us)

SSD – Solution

- Facilities
- Processes
- Roles
- And the HSModule and supporting Sw

SSD – Roles

- Basic User
  - Is the metadata operator, has access to private key operation.

- Master User
  - Master User can grant Basic User rights ro SSD.

- Facilities Officer
  - Is the only one allowed to enter HSM room and handle the SSD.

- Key Officers 1& 2
  - Are responsible for storing the divided back up of private key. The only role that can be mixed with other roles.

SSD – Facilities

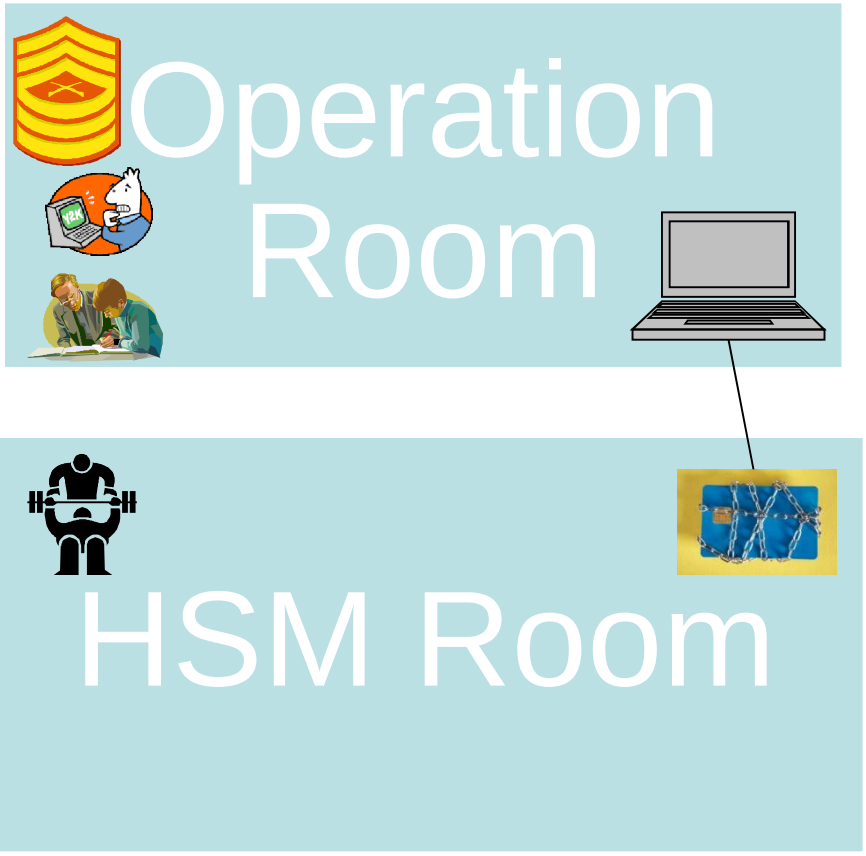
- Two rooms with restricted access
- Users having Master/Basic User role do not have clearance to HSM Room
- Only user having Facilities officer role is allowed to HSM Room

Operation Room

HSM Room

SSD – Facilities

- In operation room we have secure laptop. The laptop has networking disabled
- In HSM room we SSD cards in secure cabinet
- Laptop is connected to SSD cards with usb readers. Wiring goes through wall



SSD – What does it consist of?

- Smart Card
- SSD Application on the smart card
- Applications running on the laptop
  - SSD Maintenance
  - SSD Metadatatool

SSD – Smart Card

Gemalto TOP GX4

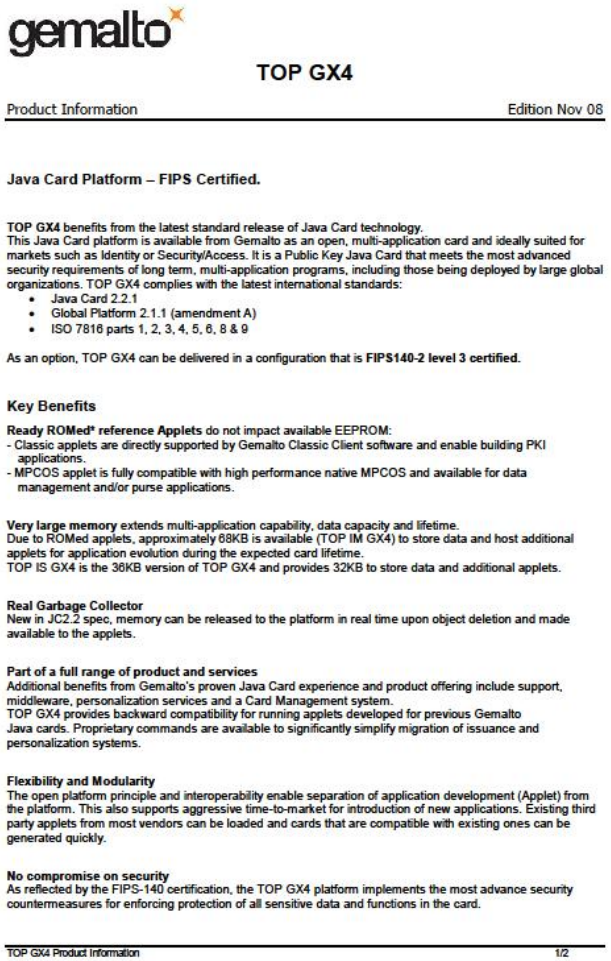
FIPS140-2 Level 3 Certified

Java Card 2.2.1

Workable tools

Slow

Cheap (~15€/pcs)



gemalto

TOP GX4

Product Information — Edition Nov 08

Java Card Platform – FIPS Certified.

TOP GX4 benefits from the latest standard release of Java Card technology.
This Java Card platform is available from Gemalto as an open, multi-application card and ideally suited for markets such as Identity or Security/Access. It is a Public Key Java Card that meets the most advanced security requirements of long term, multi-application programs, including those being deployed by large global organizations. TOP GX4 complies with the latest international standards:

- Java Card 2.2.1
- Global Platform 2.1.1 (amendment A)
- ISO 7816 parts 1, 2, 3, 4, 5, 6, 8 & 9

As an option, TOP GX4 can be delivered in a configuration that is **FIPS140-2 level 3 certified.**

Key Benefits

Ready ROMed® reference Applets do not impact available EEPROM:
- Classic applets are directly supported by Gemalto Classic Client software and enable building PKI applications.
- MPCOS applet is fully compatible with high performance native MPCOS and available for data management and/or purse applications.

Very large memory extends multi-application capability, data capacity and lifetime.
Due to ROMed applets, approximately 68KB is available (TOP IM GX4) to store data and host additional applets for application evolution during the expected card lifetime.
TOP IS GX4 is the 36KB version of TOP GX4 and provides 32KB to store data and additional applets.

Real Garbage Collector
New in JC2.2 spec, memory can be released to the platform in real time upon object deletion and made available to the applets.

Part of a full range of product and services
Additional benefits from Gemalto's proven Java Card experience and product offering include support, middleware, personalization services and a Card Management system.
TOP GX4 provides backward compatibility for running applets developed for previous Gemalto Java cards. Proprietary commands are available to significantly simplify migration of issuance and personalization systems.

Flexibility and Modularity
The open platform principle and interoperability enable separation of application development (Applet) from the platform. This also supports aggressive time-to-market for introduction of new applications. Existing third party applets from most vendors can be loaded and cards that are compatible with existing ones can be generated quickly.

No compromise on security
As reflected by the FIPS-140 certification, the TOP GX4 platform implements the most advance security countermeasures for enforcing protection of all sensitive data and functions in the card.

TOP GX4 Product Information — 1/2

SSD – The Option II

IBM 4764

http://www-03.ibm.com/security/cryptocards//pcixcc/overproduct.shtml

FIPS140-2 Level 4 Certified

Fast

Expensive – approx. 8600$

FIPS-2 Physical Security Requirements

	General Requirements for all Embodiments	Single-Chip Cryptographic Modules Multiple	Multiple-Chip Embedded Cryptographic Modules	Multiple-Chip Standalone Cryptographic Modules
Security Level 1	Production-grade components (with standard passivation).	No additional requirements.	If applicable, production-grade enclosure or removable cover.	Production-grade enclosure.
Security Level 2	Evidence of tampering (e.g., cover, enclosure, or seal).	Opaque tamper-evident coating on chip or enclosure.	Opaque tamper-evident encapsulating material or enclosure with tamper-evident seals or pick-resistant locks for doors or removable covers.	Opaque enclosure with tamper-evident seals or pick-resistant locks for doors or removable covers.
Security Level 3	Automatic zeroization when accessing the maintenance access interface. Tamper response and zeroization circuitry. Protected vents.	Hard opaque tamper-evident coating on chip or strong removal-resistant and penetration resistant enclosure.	Hard opaque potting material encapsulation of multiple chip circuitry embodiment or applicable Multiple-Chip Standalone Security Level 3 requirements.	Hard opaque potting material encapsulation of multiple chip circuitry embodiment or strong enclosure with removal/penetration attempts causing serious damage.
Security Level 4	EFP or EFT for temperature and voltage.	Hard opaque removal-resistant coating on chip.	Tamper detection envelope with tamper response and zeroization circuitry.	Tamper detection/ response envelope with tamper response and zeroization circuitry.

SSD – Application

- Lifecycle consists of
  - Initializiation phase
  - Active phase
  - Inactive phase

SSD – Application

- Initialization phase functionality
  - Create Master Pin
  - Create Key Pair
  - Export Key Pair
  - Import Key Pair
  - Finalize Initialization

SSD – Application

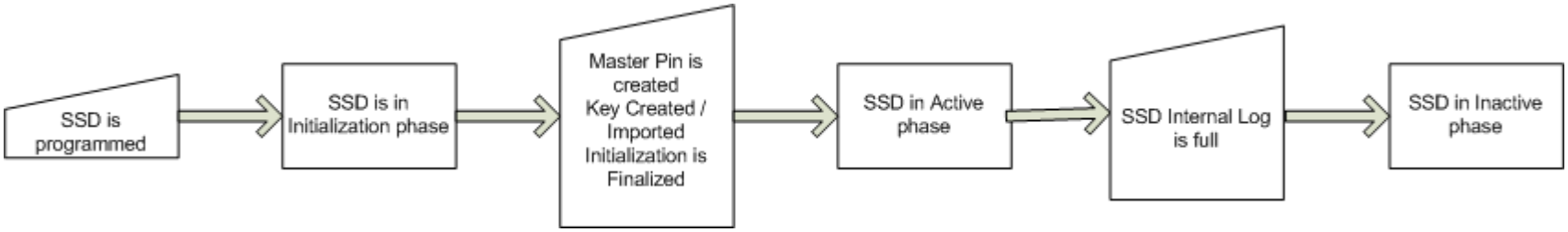
- Active phase functionality
  - Add User
  - Remove User
  - Login User
  - Export Public Key
  - Prv Key Operation

SSD – Application

- Inactive (and Active) phase functionality
  - Get Serial Number
  - Get Status
  - Get Log
  - Get Sw Version

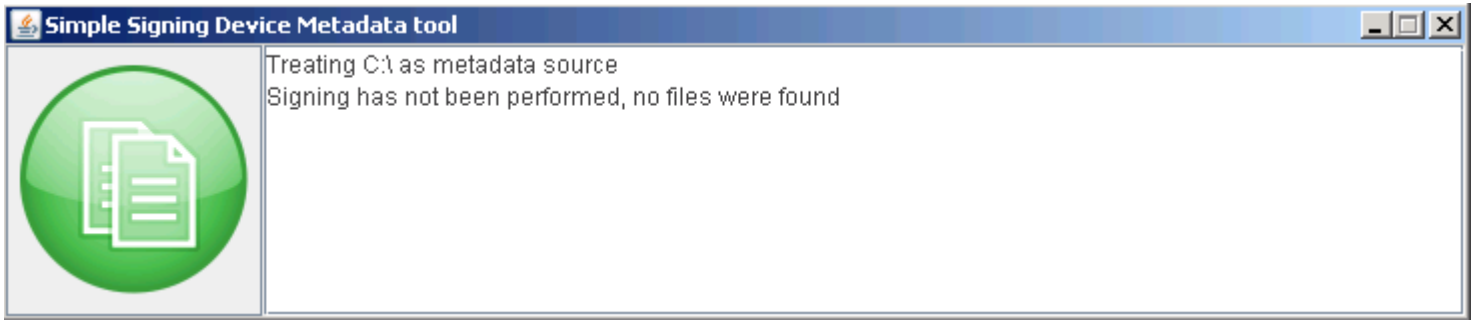
SSD – Application

- Simplified Lifecycle



SSD – Metadatatool

- Runs on secure laptop
- User can sign xml files
  - Requires User Id and Pin code



SSD – Maintenance tool

- Runs on a secure laptop
  - Initialize SSD
  - Manage users
  - View SSD internal log
  - Create CSR

SSD – Maintenance tool

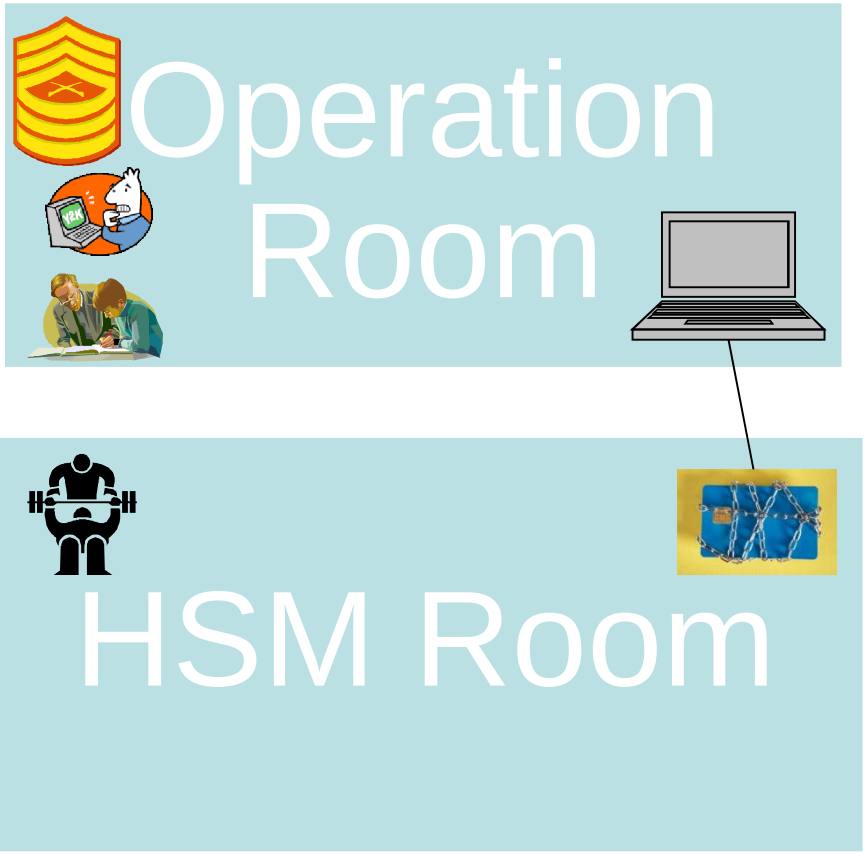
- Internal Log

SSD Log

Entry Id	Action	Description
0	KEY_PAIR_CREATED	N/A
1	USER_ADDED	User 0 added
2	USER_ADDED	User 1 added
3	SHA1PKCS1DEC_OPERATION	User 1 decrypted 4b5d6d682154868d9408d0d7ed3d4fd5b905131c
4	SHA1PKCS1DEC_OPERATION	User 1 decrypted fe48c5dea89e3d5a1d2a727c76c7d0e7270b6d3e
5	SHA1PKCS1DEC_OPERATION	User 1 decrypted fe48c5dea89e3d5a1d2a727c76c7d0e7270b6d3e
6	SHA1PKCS1DEC_OPERATION	User 0 decrypted fcee60fcad8e442e745979d2f78be8004d27bb95
7	SHA1PKCS1DEC_OPERATION	User 0 decrypted fcee60fcad8e442e745979d2f78be8004d27bb95
8	SHA1PKCS1DEC_OPERATION	User 0 decrypted fcee60fcad8e442e745979d2f78be8004d27bb95
9	SHA1PKCS1DEC_OPERATION	User 0 decrypted fcee60fcad8e442e745979d2f78be8004d27bb95
10	SHA1PKCS1DEC_OPERATION	User 0 decrypted fcee60fcad8e442e745979d2f78be8004d27bb95
11	SHA1PKCS1DEC_OPERATION	User 0 decrypted 4b5d6d682154868d9408d0d7ed3d4fd5b905131c
12	SHA1PKCS1DEC_OPERATION	User 0 decrypted fcee60fcad8e442e745979d2f78be8004d27bb95
13	SHA1PKCS1DEC_OPERATION	User 0 decrypted fcee60fcad8e442e745979d2f78be8004d27bb95
14	SHA1PKCS1DEC_OPERATION	User 0 decrypted fcee60fcad8e442e745979d2f78be8004d27bb95
15	SHA1PKCS1DEC_OPERATION	User 0 decrypted fcee60fcad8e442e745979d2f78be8004d27bb95
16	SHA1PKCS1DEC_OPERATION	User 0 decrypted fcee60fcad8e442e745979d2f78be8004d27bb95
17	SHA1PKCS1DEC_OPERATION	User 0 decrypted fcee60fcad8e442e745979d2f78be8004d27bb95
18	SHA1PKCS1DEC_OPERATION	User 0 decrypted fcee60fcad8e442e745979d2f78be8004d27bb95

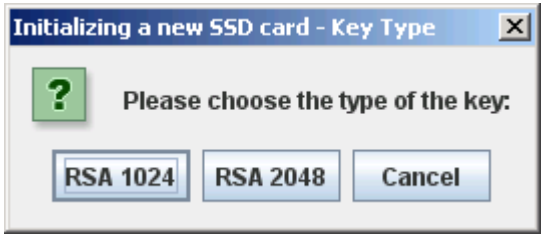
SSD – Initializing SSD

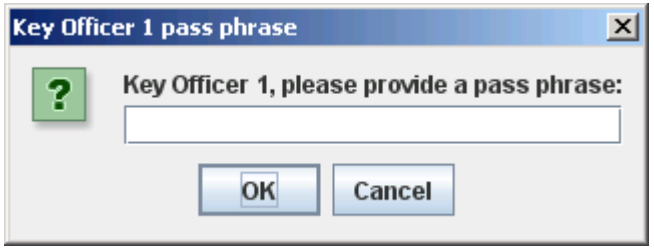
- Facilities officer moves the SSD to cabinet inHSM Room ..
- SSD Maintenance tool is launched in operation room to start process



SSD – Initializing SSD

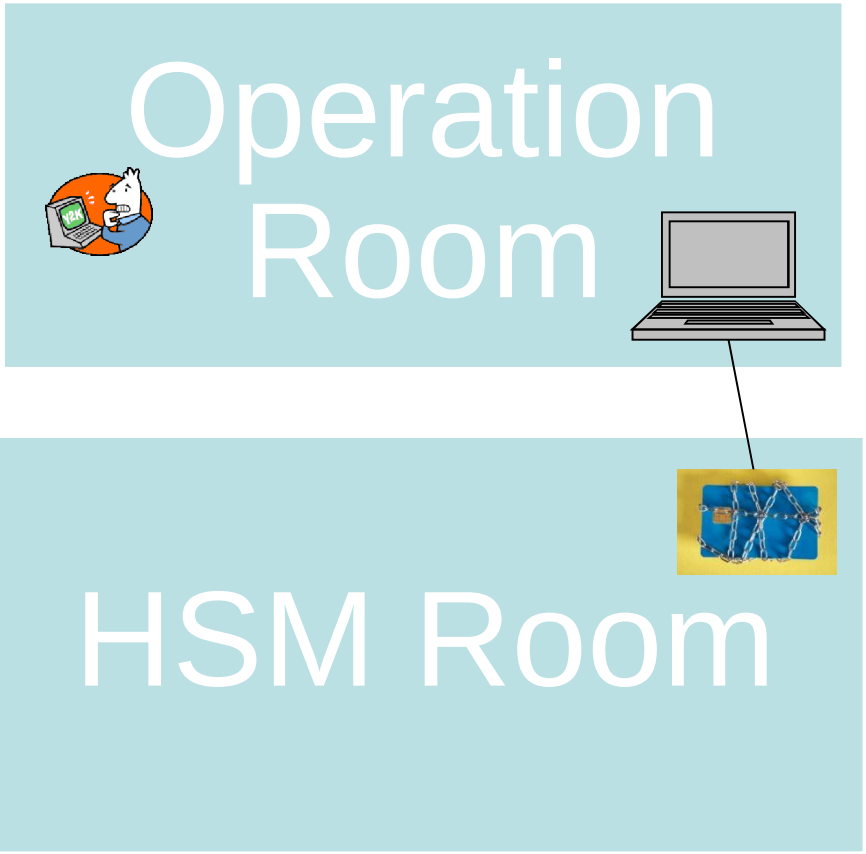
- Next->next type of initialization
- Backup of the key can be created only during initialization process
- Backup is not mandatory
- Key officers get halves of the key encrypted by passphrases..





SSD – Signing with SSD

- Signing requires precense of only the Basic User.
- Every prv key operation is recorded to internal log of the SSD.



SSD – So we have a system such that

- No person has to be in contact with plain text private key
- It is very unlikely that hamfisted operator could compromise the key by mistake
- It is also unlikely that one disgruntled employee is able to take a copy of the key
- We are able to sign without tedious ceremonies

SSD – what else

- Offline is offline..
- Limited number of log entries..
- This is result of first round and I expect we have to modify concept a bit in the future (which actually is not bad but fun)
- We have now fairly safe way of creating and using private key in controlled fashion, but is metadata signing the right application for it ?-)