SIGS SE February 2017

Keep your fingers off my keys

today & tomorrow

Marcel Dasen
VP Engineering

Securosys SA

Keys?

- **Encryption keys**
  - asymmetric e.g. RSA, ECC public/private **key** pairs for wrapping
  - symmetric e.g. AES, RC4, 3DES **keys**
- **Signature keys**
  - asymmetric e.g. DSA, ECDSA public/private **key** pairs for signing (of object hashes)
  - hash algorithms e.g. SHA-2, SHA-3
- **Certificates**
  - digitally signed **public keys**

The digital “certificate”

Public

Subject identifier: securosys.ch Issuer identifier: swisssign.com
Subject public key, e.g. RSA 4096
validity period
“checksum” = hash of above
Issuer signature, e.g. RSA 4096

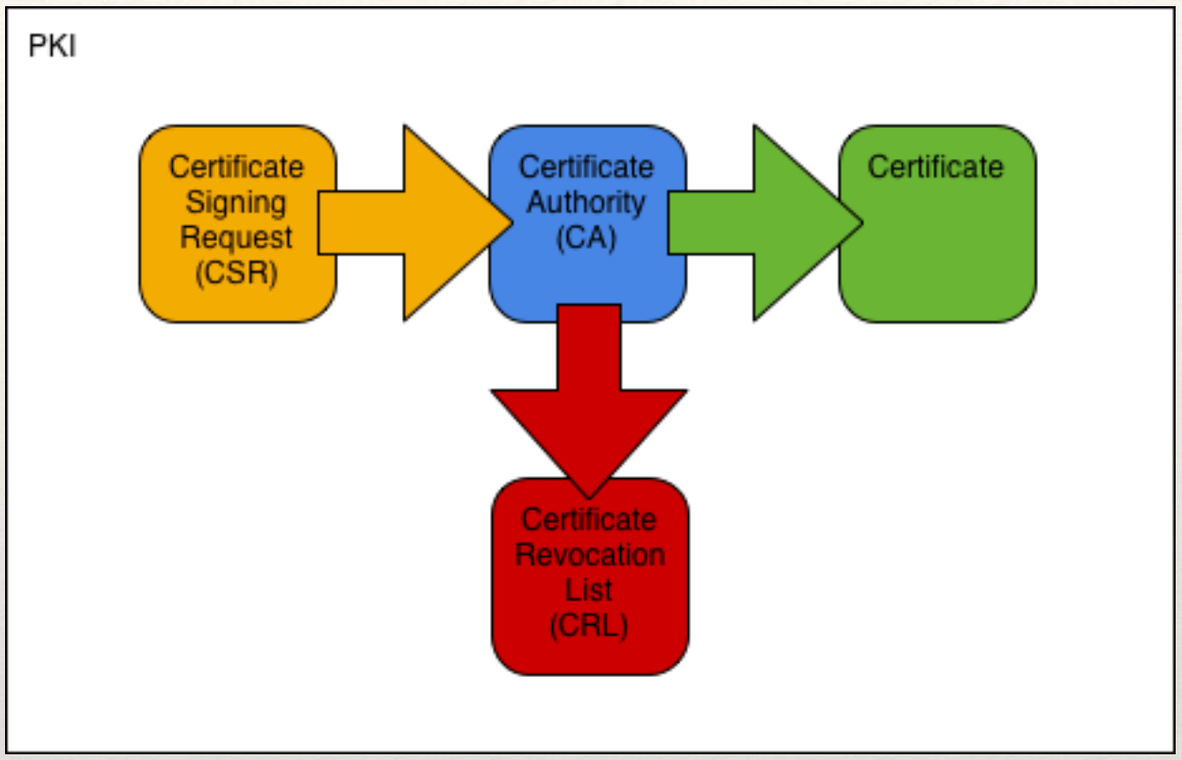
Secured

Issuer signature validation key, e.g. RSA 4096
Subject private key
Issuer signature key

securosys

Key and Certificate Generation

- A CSR requires to
  - generate a key pair
  - safely store the private key for later usage
- CA requires to (after RA assures validity of CSR)
  - sign CSR with the secret key
  - safely store the secret key
- CA issues the Certificate
- Keys can be “revoked” thus revocation list CRL are required

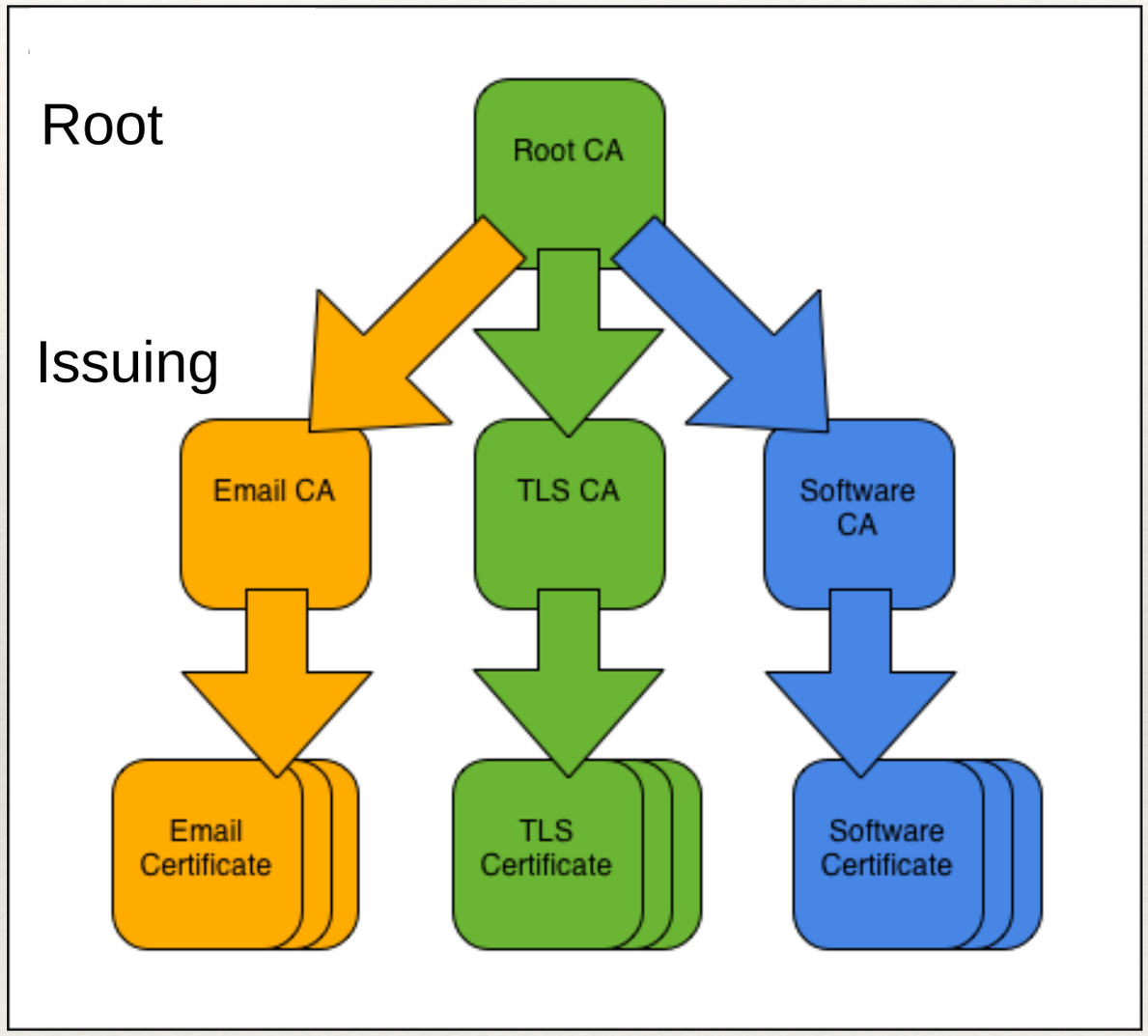


Keys have to be managed!

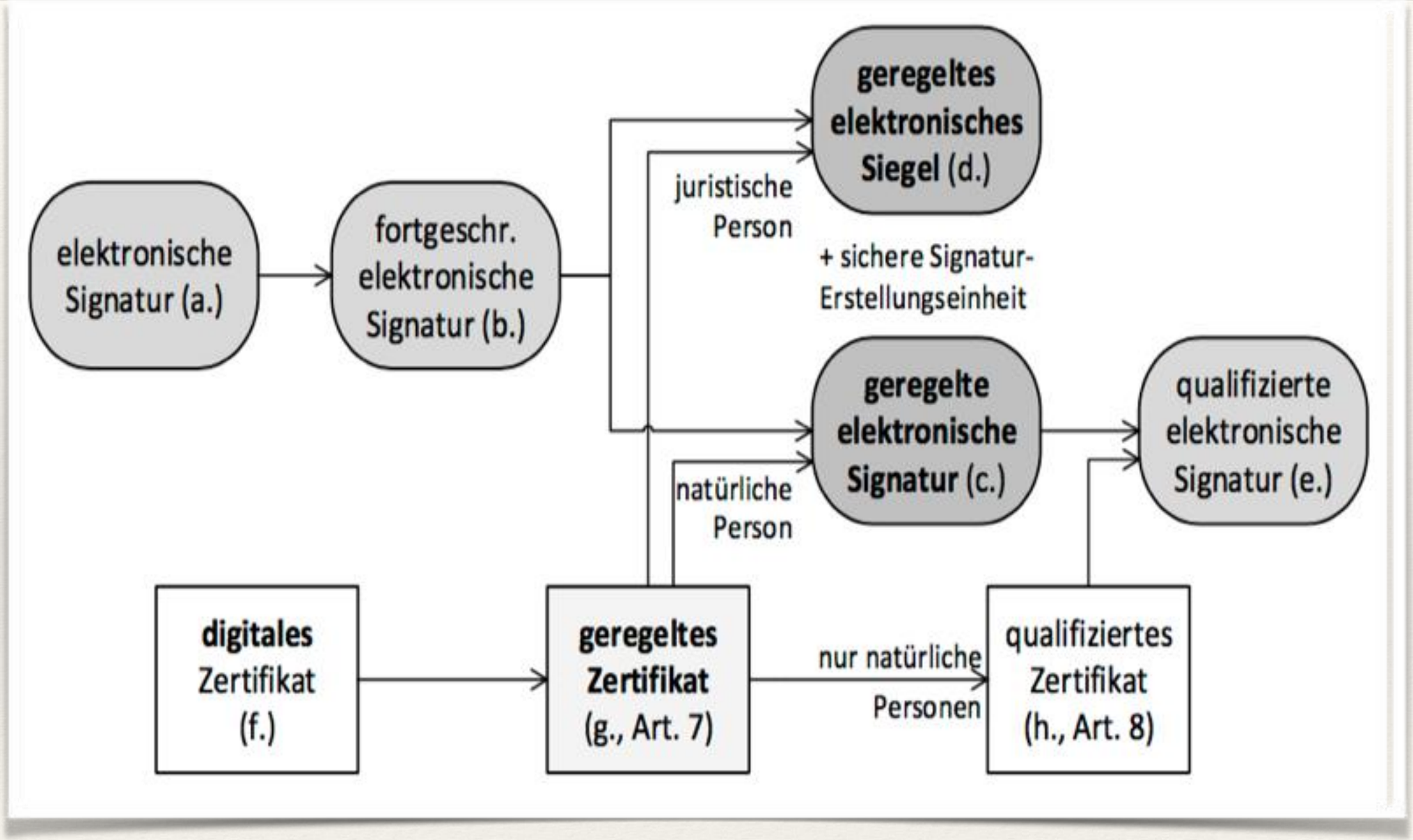
securosys

Key Management

- Typically trust is organized hierarchically
- Even in simple PKI case multiple key pairs are involved
- Different applications have different policies. This is best organized using multiple issuing certificates
- Keys can be “revoked” thus each CA needs to generate (and sign) revocation lists



“qualifizierte” elektronische Signatur



- Codified in ZertES (SR 943.02), equivalent codifications in all EU countries [3]
- issuing device (CC EAL4+, FIPS140-2 L3, or equivalent) and audited procedures (CP/CPS)

securosys

Where are cryptographic keys used?

- Payment infrastructure
  - payment card
  - Clearing, Settlement (e.g. SIC)
- “geregelte” digital signature
  - MWST
  - e-GOV/HEALTH/PASS,…
- Web server authentication
  - (e.g. https), “SSL” certificates
  - EV-certificates
- Infrastructure Services
  - IPSEC/SSL VPN
  - DNSSEC
  - SMTP(mail)
- Personal digital signatures
  - S/MIME (email)
  - ssh, login
- Data Storage
  - TDE (Oracle, SQL Server)
  - drive encryption
  - Cloud storage and service (CASB)

securosys

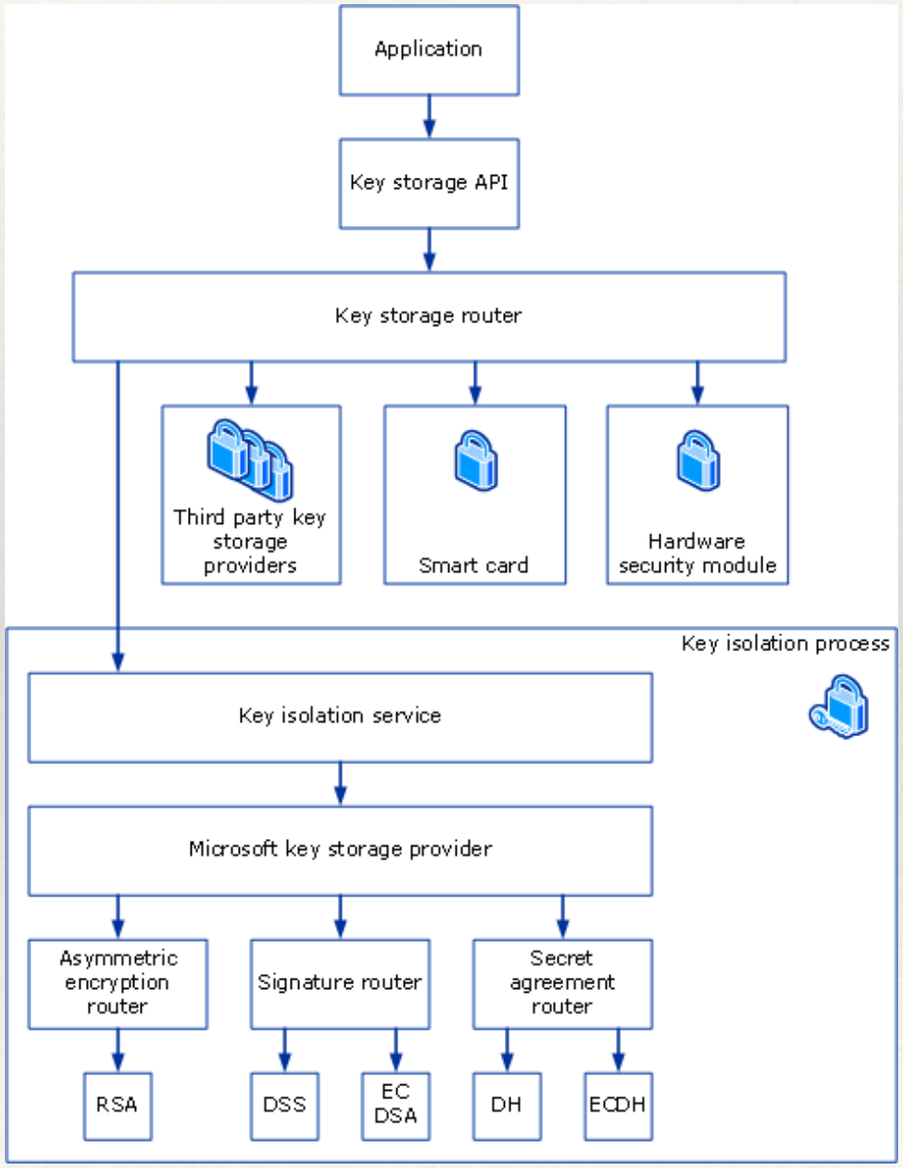
Private & Secret Key Storage

Unix (& Linux)

- Keys are typically stored in files, e.g. in users home directory
- Keys are protected by user / group file permissions
- privileged users (e.g. “root”) can access everything

```
dasen$ ls -al .ssh
drwx------  1 dasen  staff   264 Jan 27  2015 .
drwxr-xr-x+ 1 dasen  staff  1792 May 12 16:21 ..
-rw-------  1 dasen  staff  1766 Jan 27  2015 id_rsa
-rw-r--r--  1 dasen  staff   400 Jan 27  2015 id_rsa.pub
```

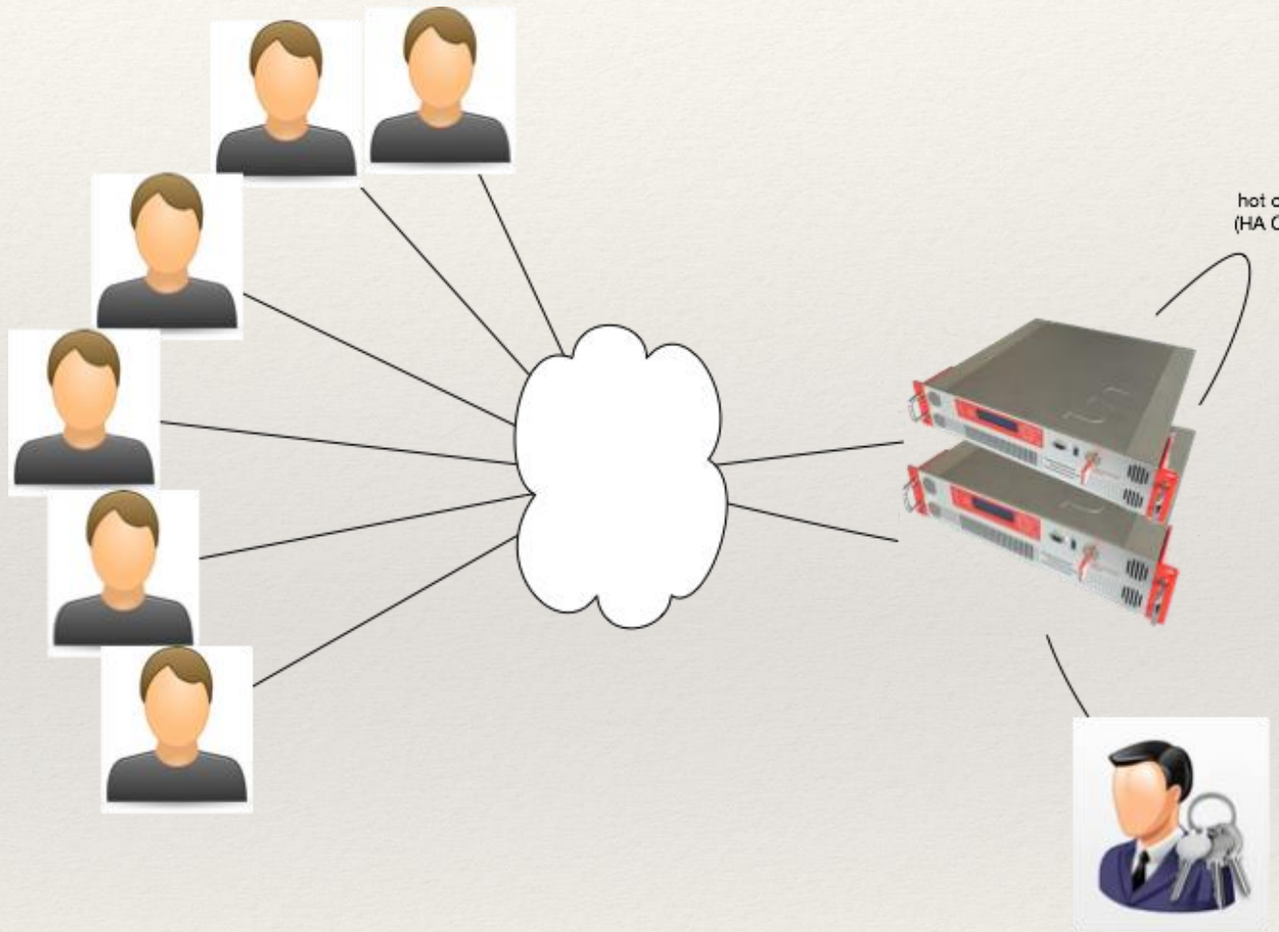
Microsoft Windows



securosys

The managed Key Store

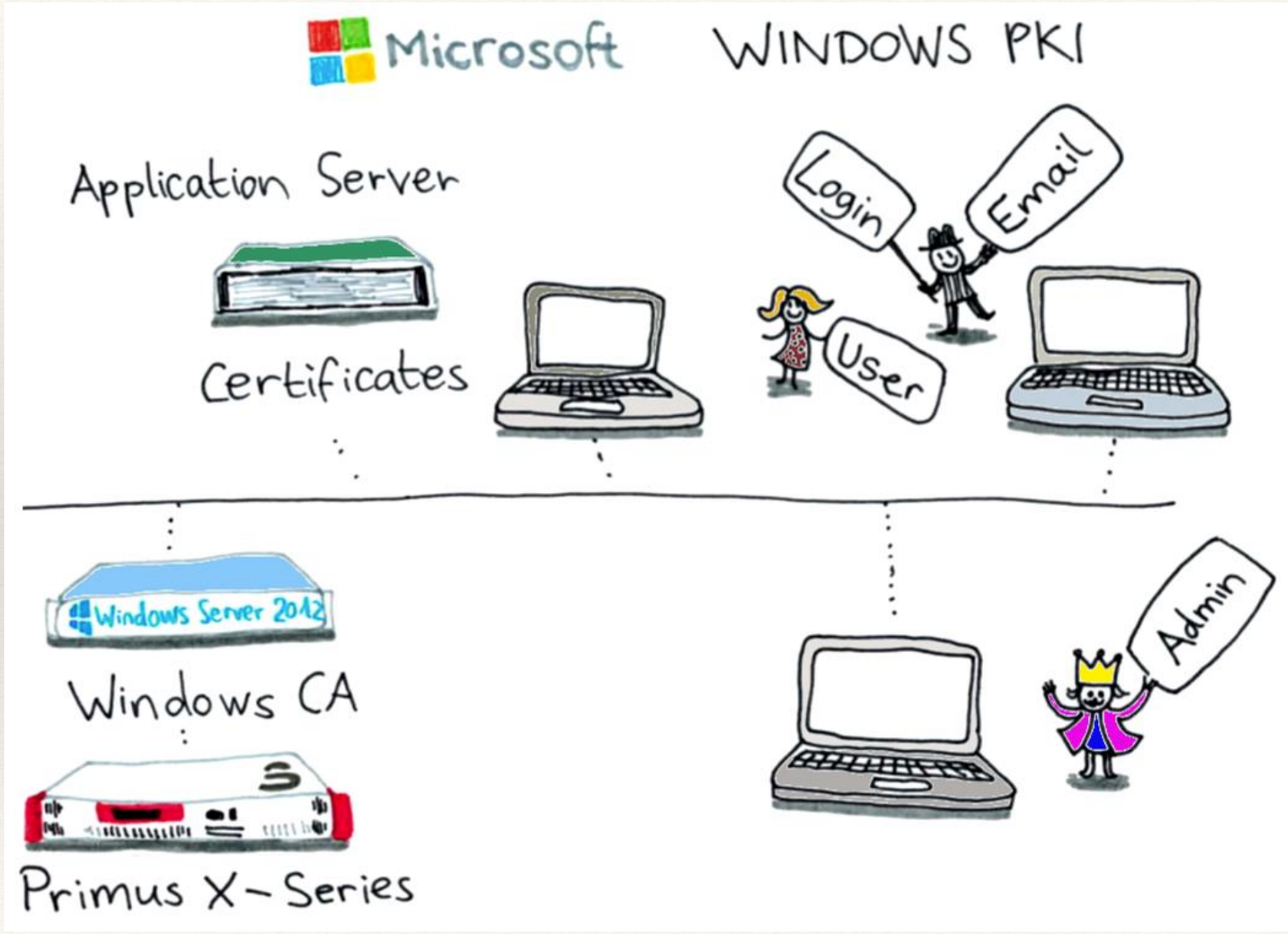
- Dedicated Module storing encryption keys
- Device protects keys against loss and thus serves as the **Trust Anchor** for the infrastructure.
- One central place to store private and secret keys



Private and secret keys never leave the Security Module

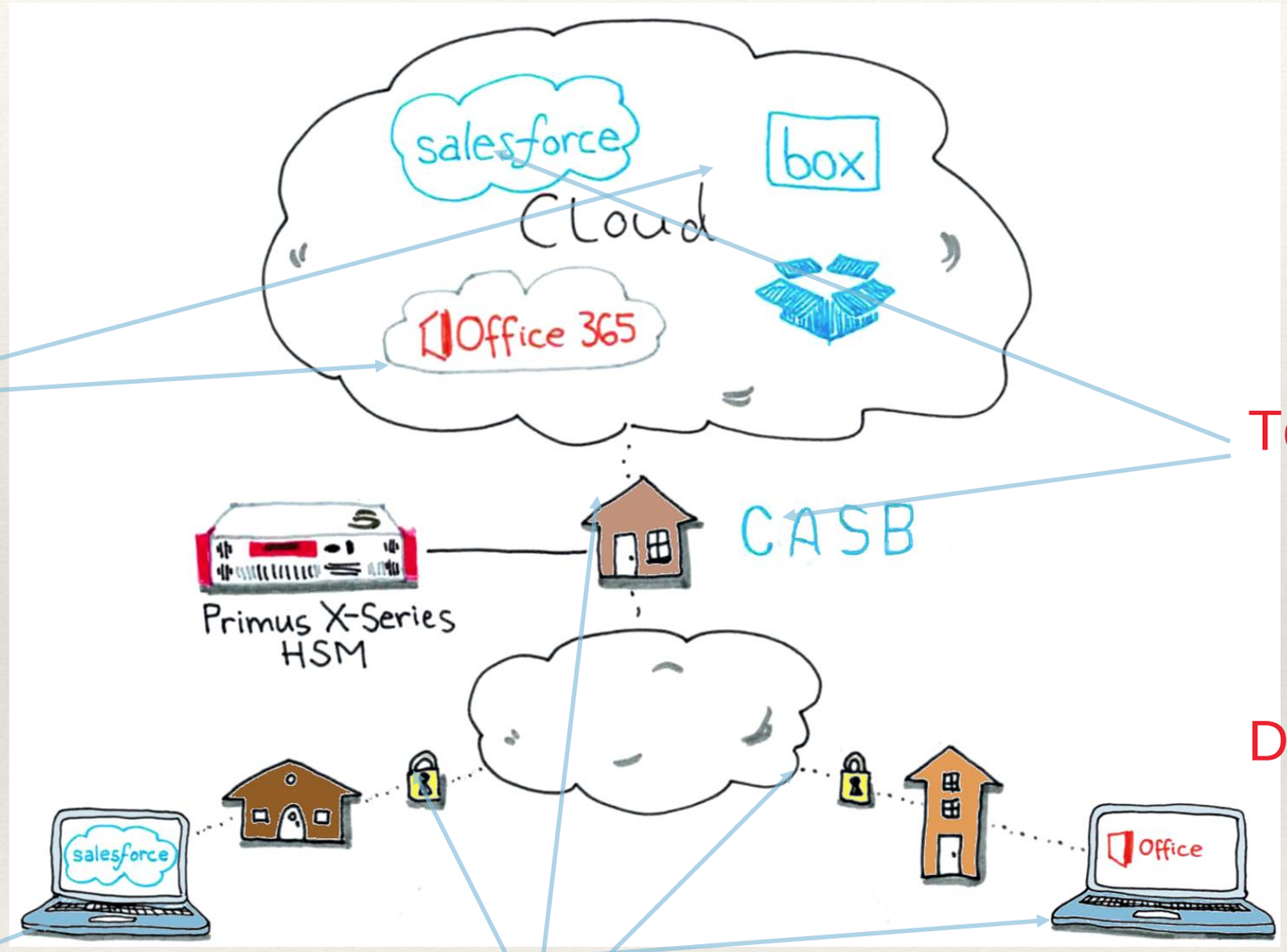
securosys

On premise IT with PKI



securosys

Cloud Security



Storage Encryption

BYOK

Tokenisation

DB Encryption

User authentication

VPN link encryption

securosys

The Post Quantum World

- Asymmetric public key algorithms
  - **RSA** (integer factorization) broken by Shor's algorithm
  - **DSA, DH, ECDSA, ECDH** (discrete logarithm) broken by derived Shor algorithm
- Symmetric key algorithms
  - **AES, 3DES, …**
  - strength reduced from n-bits to n/2-bits by Grovers algorithm
  - **SHA(-256)** strength reduced n-bits to n/2-bits

… but…

Algo	qubits	runtime	expected # qubits to break
RSA	2n	c * n^3	~6000qb @ RSA3072
ECC	f’(n)(f(n)	c * n^3	~3000qb @ ECC512
AES	c * n	O(2^(n/2))	~6500qb @ AES 256

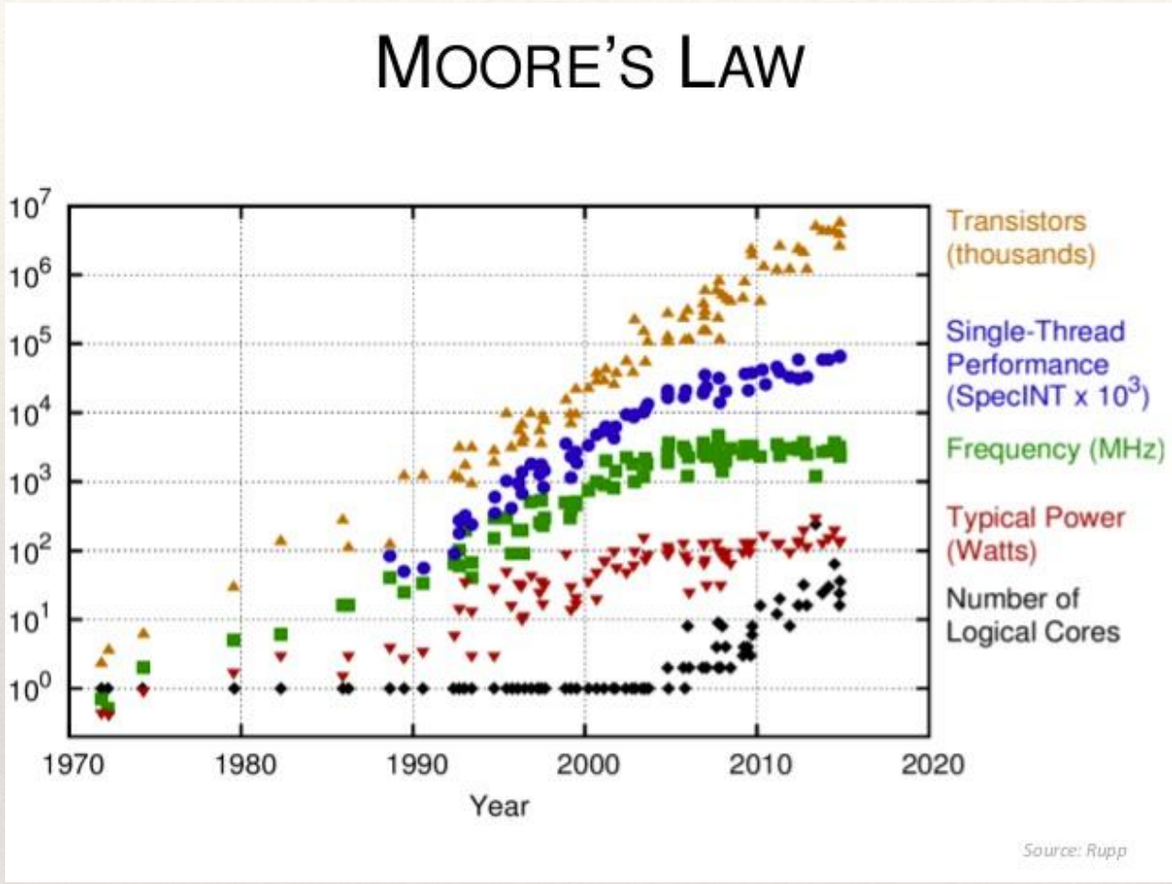
State of the Art: Today O(10) qubits

Proos and Zalka, 2008
Grasses et. al, 2015

securosys

…when should we worry?

- Assuming MOORE's "law", we will have to worry in 20-50y from now.
- But does it apply at all?



Quantum Computers are NOT on a predictable technological scaling path, but assume new DISRUPTIVE technologies.

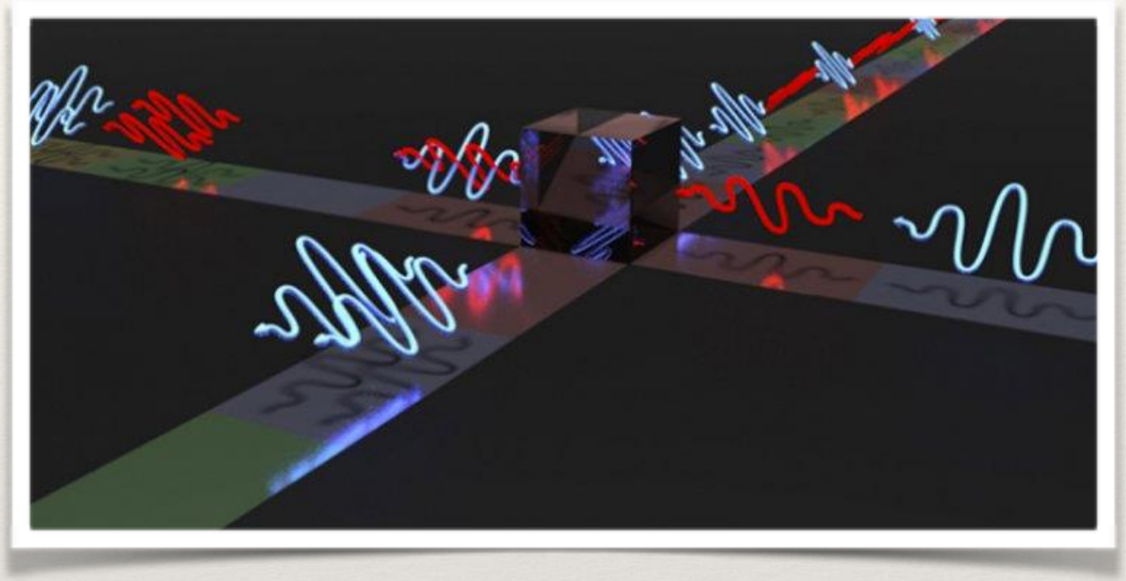
securosys

If I worry anyhow ?

- Use Protocols mostly based symmetric (AES) cryptography
  - AES
  - Kerberos
    - Safely lock away the key (e.g. HSM)
- Use long key sizes for asymmetric algorithms
  - gives you more time to transition to new algorithms
- Be prepared for post quantum algorithms
  - Future signatures and keys may consist of much longer bit strings,
- Use upgradable crypto hardware and software

Should I wait for PQC ?

- Various Research efforts on going
  - EU SAFEcrypto
    - http://www.safecrypto.eu
  - NIST PQCrypto
    - http://csrc.nist.gov/groups/ST/post-quantum-crypto
- Candidates for PQC
  - Lattice-based (NTRU, BLISS, LWE)
  - Code-based (McEliece)
  - …



These crypto systems are years away from adoption

securosys

Address Current Threats!

- Harden your infrastructure
- Use PKI with strong cryptography
- Manage your keys
- Work with trusted suppliers

securosys

Summary

- Digital keys are a crucial part of the digital world.
- Keys must be safeguarded. You want to know where and when they are accessed
- Keys must be managed
- PKI is the state of the art
- Keys are trusted if generated with a trustworthy key generator.
- PQC is not ready for adoption

securosys

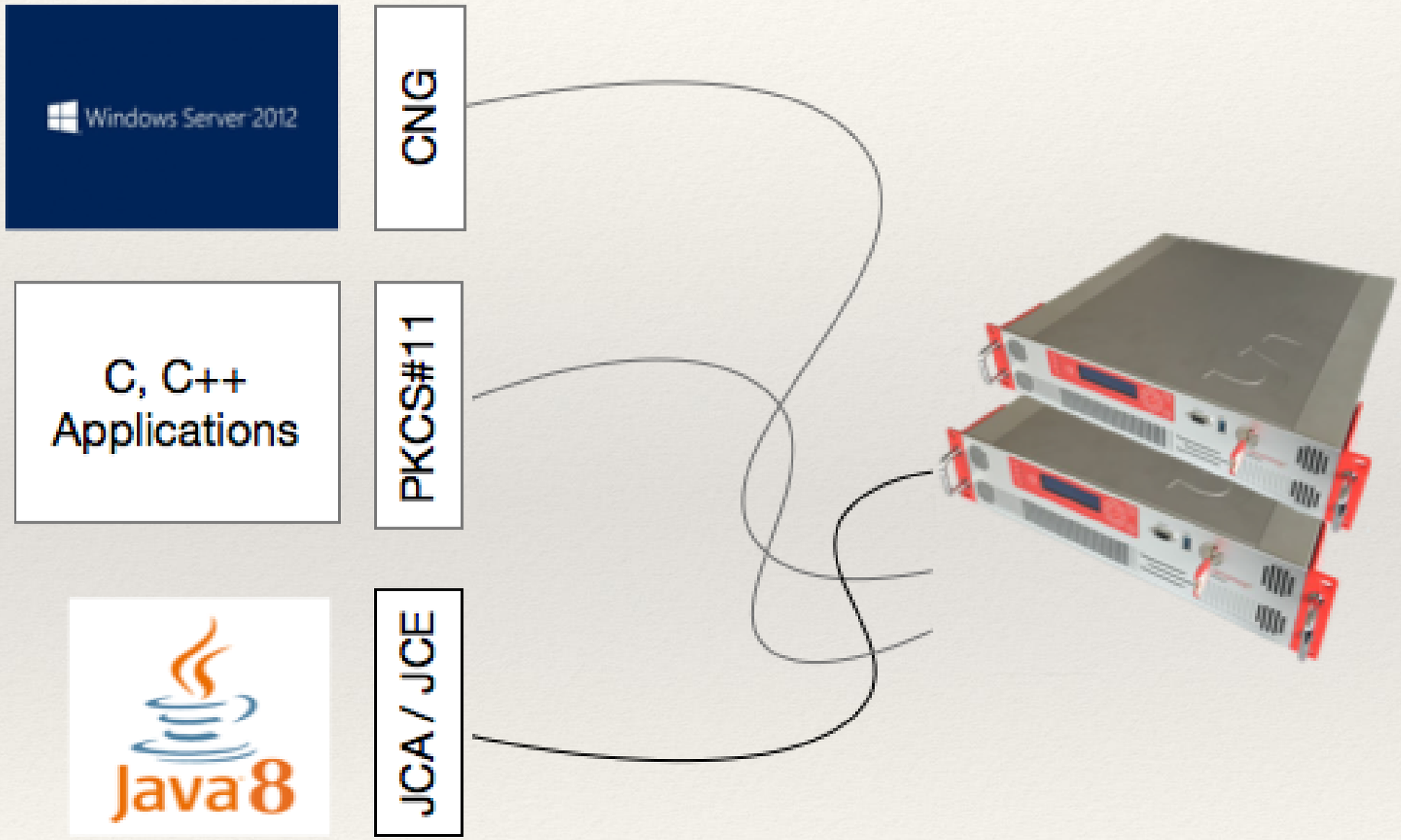
securosys

SWISS SECURITY TECHNOLOGIES
FOR COMMUNICATIONS SYSTEMS

Do I care? What are the risks?

- **Loss of** “qualifiziertes Zertifikat”, “geregeltes Zertifikat” **signature certificate private key**
  - Legally binding signatures can be issued
- **Loss of communication certificate private keys**
  - interception possibility (eaves dropping, man in the middle)
  - potentially conflict with data protection laws
- **Loss of storage keys**
  - loss of intellectual property,
  - business continuity cost (e.g. after loss of credit card holder data)
  - potentially conflict with data protection laws
- **Loss of “SSL” certificate private key** (e.g. EV certificate)
  - Web site, email impersonation => attack vector
  - Risk for reputation with customers / partners

Integration of a trust anchor



Generating and managing keys

- Use a secure key generation device (HSM)
  - Hardware random number generator
    - Validated encryption algorithms
- Keep private keys in protected storage
  - Hardware security module
- Use a Software CA or Windows built-in CA