OpenDNSSEC.com

John Dickinson

OpenDNSSEC.com

- A collaboration between
  - Kirei AB
  - John Dickinson
  - .se
  - NLNetlabs
  - Nominet
  - IANA

OpenDNSSEC.com

- History and Aims
- Security Modules
  - Hardware Security Modules
  - Software Security Modules
- KASP
- The OpenDNSSEC signer
  - Design
  - Discussion

OpenDNSSEC.com

- History
  - DNSSEC works
  - Lot of DNSSEC tools exist
    - BIND, LDNS, ...
  - .se signed using smartcards
  - Nominet developed proof of concept signer that could utilise a HSM
  - Rick Lamb added HSM support to BIND

OpenDNSSEC.com

- We learned
  - DNSSEC signing is hard to do correctly
    - Keys are hard to manage
      - Lots of files everywhere
      - Filename gives you no information
        - Kjadickinson.co.uk.+005+58327
    - Keys are vulnerable
      - No encryption of private keys
    - Some form of keystore is necessary
      - Why reinvent the wheel?
      - Use a HSM!

OpenDNSSEC.com

- DNSSEC signing is hard to do correctly
  - No place to keep config or metadata
  - Some kind of database is needed
    - KASP
  - Difficult to keep zone signed
    - re-sign and reload needed
    - Might need signing acceleration

OpenDNSSEC.com

- Aims
  - To provide documentation and examples of using HSMs for DNSSEC
  - To design and develop the perfect signer!

To Make DNSSEC easy!

OpenDNSSEC.com

Hardware Security Modules.

OpenDNSSEC.com

- What is a HSM?
  - Stores keys (master keys) in hardware
  - Performs operations with those keys
- Why use one?
  - Security (FIPS)
    - Private key never allowed outside the HSM
    - You know where your keys are
  - Performance
    - 1 – 14,000 signatures per second.
- Are they expensive?
  - $50 - $50,000

OpenDNSSEC.com

- HSM
  - Cryptoflex egate
    - usb token, 1 key, 1 signature/second
  - Sun SCA6000
    - Master key on device
    - Keys in encrypted file on disk
    - PCIE Card, FIPS140-2 level 3
  - AEP Keeper
    - Network device
    - Keys held on device
    - FIPS140-2 level 4 (Crypto is destroyed if tampered)
  - Also from nCipher, SafeNet, IBM

OpenDNSSEC.com

- Software Security Modules
  - opencryptoki
    - http://sourceforge.net/projects/opencryptoki
  - softpkcs11
    - http://people.su.se/~lha/soft-pkcs11/
  - solaris (opensolaris)
    - http://hg.opensolaris.org/sc/src/netvirt/usr/src/lib/pkcs11/

OpenDNSSEC.com

- Using a SM
  - PKCS#11
    - Standard (http://www.rsa.com/rsalabs/node.asp?id=2133)
    - Supported by all HSMs
    - Supplied as library with HSM
  - OpenSSL
    - via engine that uses PKCS#11
    - or some native engine
    - cannot manage keys
  - Others
    - NSS

OpenDNSSEC.com

- Using PKCS#11
  - Connect
    - C_Initialize()
    - C_OpenSession()
    - C_Login()
  - Search for objects (keys)
    - C_FindObjectsInit()
    - C_FindObjects()
    - C_FindObjectsFinal()
  - Sign Data
    - C_SignInit()
    - C_Sign()

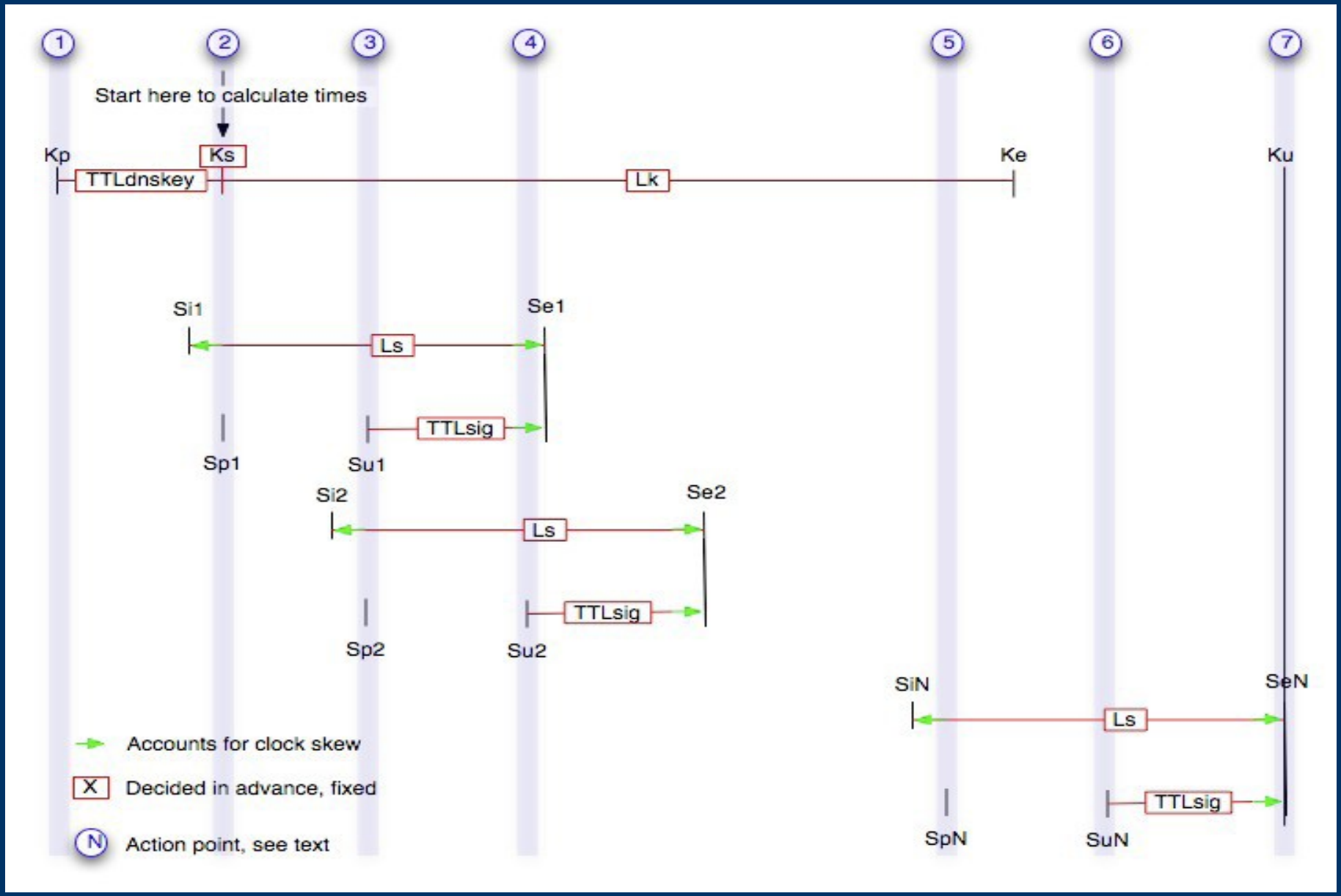
OpenDNSSEC.com

- Tools
  - opensc project
    - http://www.opensc-project.org/
    - smart card libraries and tools
    - pkcs#11 engine for OpenSSL
    - pkcs11-tool is useful for testing

OpenDNSSEC.com

Key and Signing Policy

OpenDNSSEC.com



OpenDNSSEC.com

- Example of current tools
  - Key generation

```
dnssec-keygen -a rsasha1 -b 512 -n zone example.com
Kexample.com.+005+63933
dnssec-keygen -a rsasha1 -b 2048 -f KSK -n zone example.com
Kexample.com.+005+57514
```

  - Signing

```
cat Kexample.com.+005+57514.key >> example.com
cat Kexample.com.+005+63933.key >> example.com
dnssec-signzone -o example.com -t -k Kexample.com.+005+57514 \
    example.com Kexample.com.+005+63933
```

OpenDNSSEC.com

- Key and Signing Policy
  - Signing Policy
    - Re-sign interval
    - Refresh interval
    - Jitter
    - DNSKEY and RRSIG TTL

OpenDNSSEC.com

- Key and Signing Policy
  - NSEC Parameters
    - Type NSEC/NSEC3
    - NSEC TTL
    - NSEC3 settings

OpenDNSSEC.com

- Key and Signing Policy
  - Key Parameters ZSK and KSK
    - Length
    - Algorithm
    - Lifetime
    - Location
    - Overlap

OpenDNSSEC.com

- Key Metadata
  - ID
  - Zone
  - Algorithm
  - Length
  - Usage
  - Flags
  - Key Tag (Revoked as well)
  - Location
  - Timestamps

OpenDNSSEC.com

- KASP used by
  - Signer
  - Key management
  - Monitoring and alerting
  - Registry system
  - Nameservers
- Accessed
  - Database
  - via API

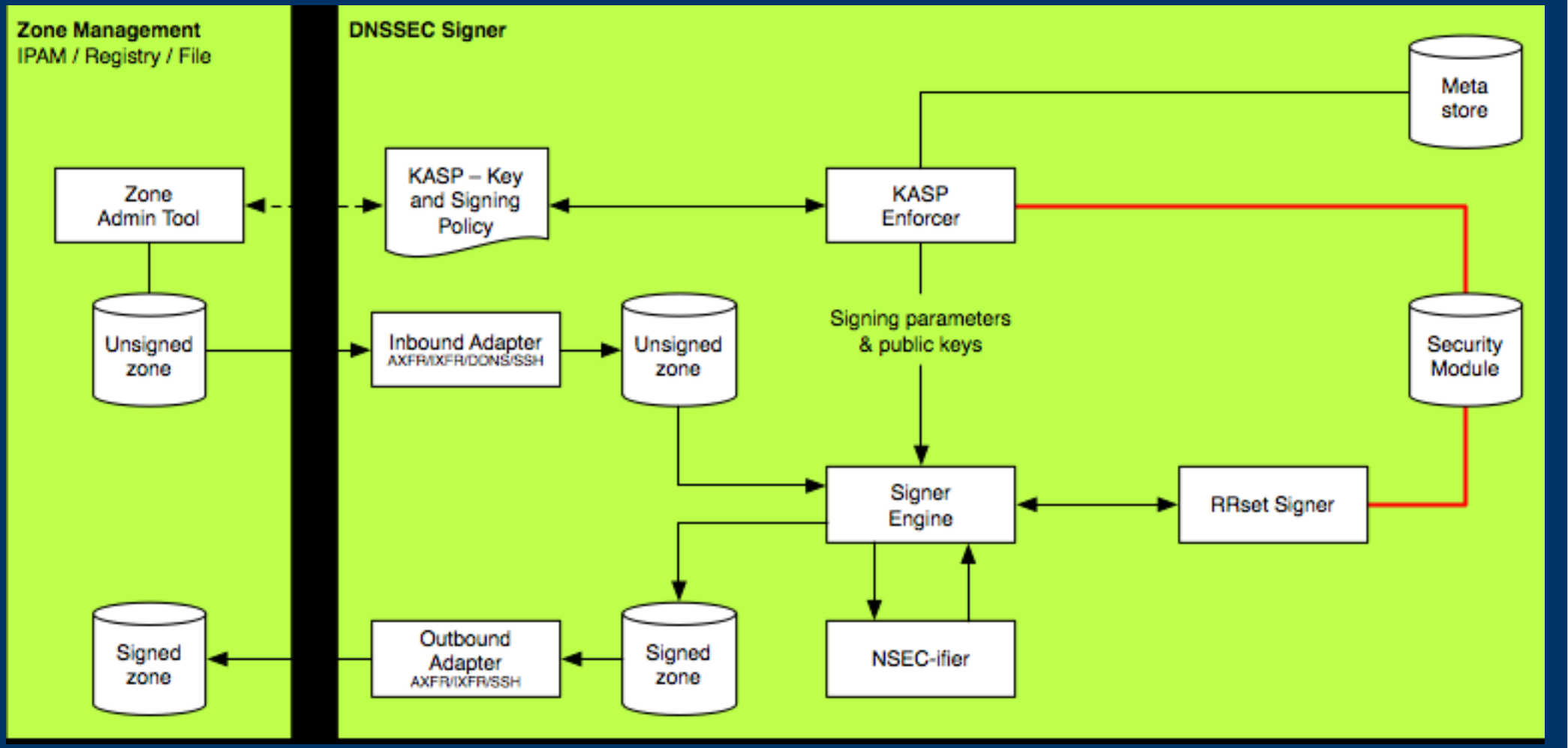
OpenDNSSEC.com

OpenDNSSEC Design

OpenDNSSEC.com

- Requirements
  - RFC 4033,4034,4035,5011,5155
  - Performance 1 to >40,000 sig/sec
  - Secure
  - Automatic
    - No human intervenetion
    - Alerting via SNMP and logs

OpenDNSSEC.com



OpenDNSSEC.com

- KASP Enforcer
  - Enforces the policy described by KASP.
  - Runs as daemon
  - Ensures enough keys exist
  - Removes old keys
  - Ensures the signer is run as needed

OpenDNSSEC.com

- Signer
  - Logically separated
    - signer
    - nsecifier
    - rrset signer
  - NSECifier
    - Adds NSEC(3) RR's
  - RRSet signer
    - Signs an RRSet
    - Talks to the keystore

OpenDNSSEC.com

- Adaptors
  - Inbound and Outbound zone data
- Variety of mechanisms
  - XFR
  - svn
  - files
  - ssh
  - database
  - DDNS

OpenDNSSEC.com

- Key URL
  - Allows the specification of multiple key locations
    - Files
    - HSM
    - SSM
    - other

```
file:///var/keys/key1
pkcs11:///usr/lib/libpkcs11.so?id=45&label=abc
```

OpenDNSSEC.com

- Questions

jad@jadickinson.co.uk