Protect your private keys with inexpensive crypto devices

Marlon Dutra
Production Engineer, Security
Facebook Inc.

Me

- Net/sec/sys admin since 1995
- Moved from Brazil to the US in 2012
- Production Engineer @Facebook since 2012
- Tech lead in Infrastructure Security
- Team owns several crypto services, SSH infra, etc
- 4th time presenting to MUG
  - 2008 - LTSP case study in Brazil
  - 2014 - Facebook traffic infra
  - 2016 - SSH with CA-signed certificates
  - 2018 - this talk

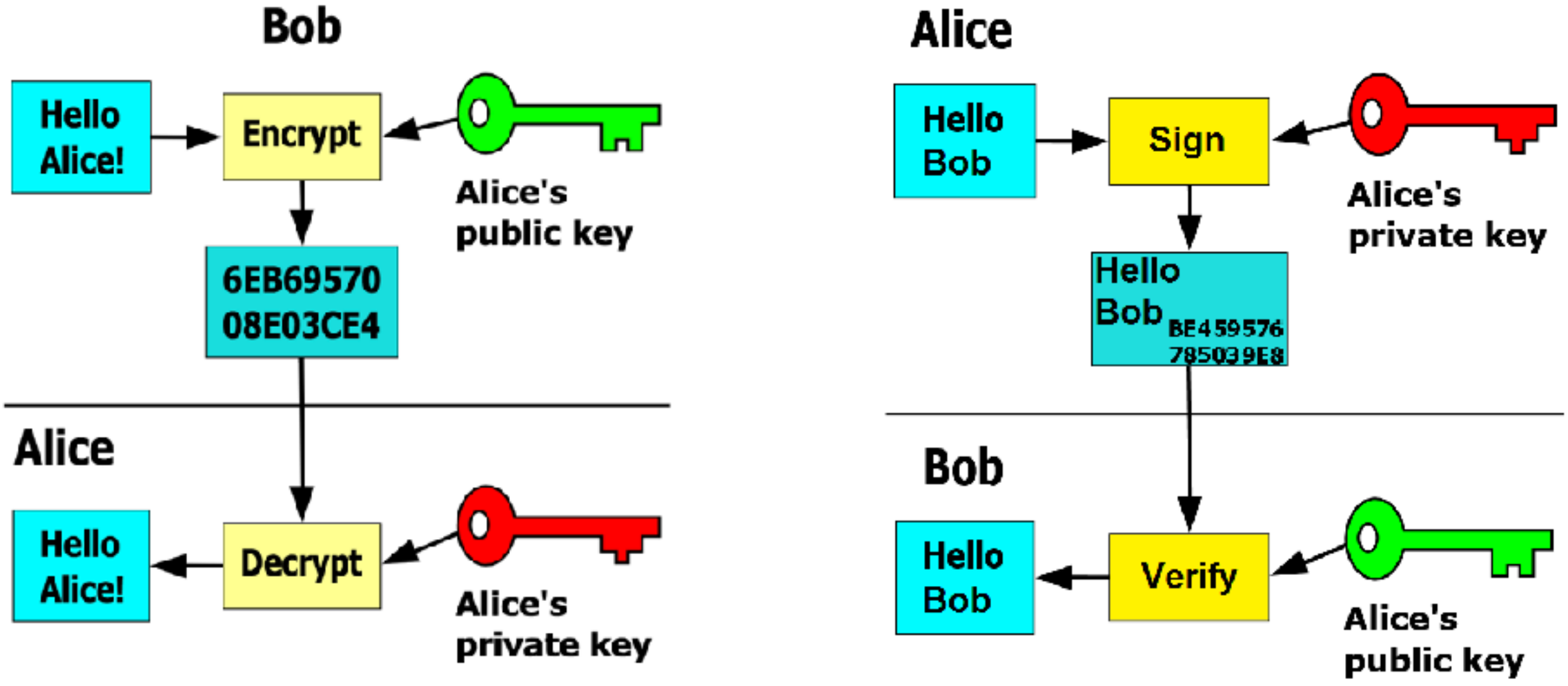
Crypto devices

- AKA
  - HSM = hardware security module
  - PIV = personal identification verification
  - Smart card
  - Secure enclave
  - Token, or Crypto token
  - etc...

Crypto devices

- Basic functionalities
  - Generate keys **without ever exposing private keys**
  - Import keys
  - Expose primitive crypto operations
    - encrypt, decrypt, sign, verify, random...
- Support multiple ciphers
  - Asymmetric at least (RSA, DSA)
- Authentication for operations
  - PIN, smart card, quorum (secret sharing), etc

Asymmetric crypto



https://en.wikipedia.org/wiki/Public-key_cryptography

Shapes

Interfacing

- Software
  - PKCS#11
  - Microsoft CNG
  - Proprietary libraries
- Physical
  - USB
  - PCIe
  - NFC
  - IP (Network)

PKCS#11

The PKCS #11 standard defines a platform-independent API to cryptographic tokens, such as hardware security modules (HSM) and smart cards...

-- Wikipedia

https://en.wikipedia.org/wiki/PKCS_11

Tools

- OpenSC (PKCS#11 tool set, libraries, etc)
  - https://github.com/OpenSC/OpenSC/wiki
- OpenSSL
- OpenSSH
- Yubico PIV Tool
  - https://developers.yubico.com/PIV/Tools/Yubico_PIV_Tool.html

Initializing a Yubikey

```
$ yubico-piv-tool -a set-chuid
$ yubico-piv-tool -a change-pin -P 123456
$ yubico-piv-tool -a change-puk -P 12345678
$ MGMT=$(head -c 24 /dev/urandom | hexdump -v -e '/1 "%02X"')
$ echo $MGMT
$ yubico-piv-tool -a set-mgm-key -n "$MGMT"
```

Generating key

```
$ yubico-piv-tool -a generate -s 9c -k -o pubkey.pem
$ cat pubkey.pem
$ yubico-piv-tool -a verify-pin -a selfsign-certificate -s 9c \
  -S '/CN=Marlon Dutra/' --valid-days=10957 \
  -i pubkey.pem -o cert.pem
$ openssl x509 -text -noout -in cert.pem
$ yubico-piv-tool -a import-certificate -s 9c -k -i cert.pem
```

Testing encryption

```
$ echo "super secret message" > msg
$ openssl rsautl -encrypt -inkey pubkey.pem -pubin \
  -in msg -out msg.crypt

$ pkcs11-tool --decrypt -i msg.crypt -m RSA-PKCS
super secret message
```

Testing signing

```
$ pkcs11-tool -s -m SHA256-RSA-PKCS -i msg -o signature

$ openssl dgst -sha256 -verify pubkey.pem -signature signature msg
Verified OK
```

OpenSSH auth

```
$ yubico-piv-tool -a read-certificate -s 9c -K SSH
ssh-rsa AAAAB3N...
```

- Copy&paste to .ssh/authorized_keys on a server

```
$ ssh -o PKCS11Provider=/usr/local/lib/opensc-pkcs11.so \
  root@your_server_hostname
Enter PIN for 'PIV Card Holder pin (PIV_II)':
```

OpenSSH CA

- User generates key pair and sends pubkey for CA to sign

```
$ ssh-keygen -f user
```

- CA generates a user certificate

```
$ ssh-keygen \
  -s cert.pem \
  -D /usr/local/lib/opensc-pkcs11.so \
  -I user \
  -n user \
  -V +52w \
  -z 12345 \
  user.pub
```

OpenSSL

```
$ openssl ...
```

OpenSSL library (libssl)

key engines (OpenSC libp11)

PKCS#11 (OpenSC opensc-pkcs11.so)

OpenSSL config

```
openssl_conf = openssl_init


[openssl_init]
engines = engine_section


[engine_section]
pkcs11 = pkcs11_section


[pkcs11_section]
engine_id = pkcs11
dynamic_path = /engines/pkcs11.dylib
MODULE_PATH = /usr/local/lib/opensc-pkcs11.so
init = 0
```

OpenSSL CLI

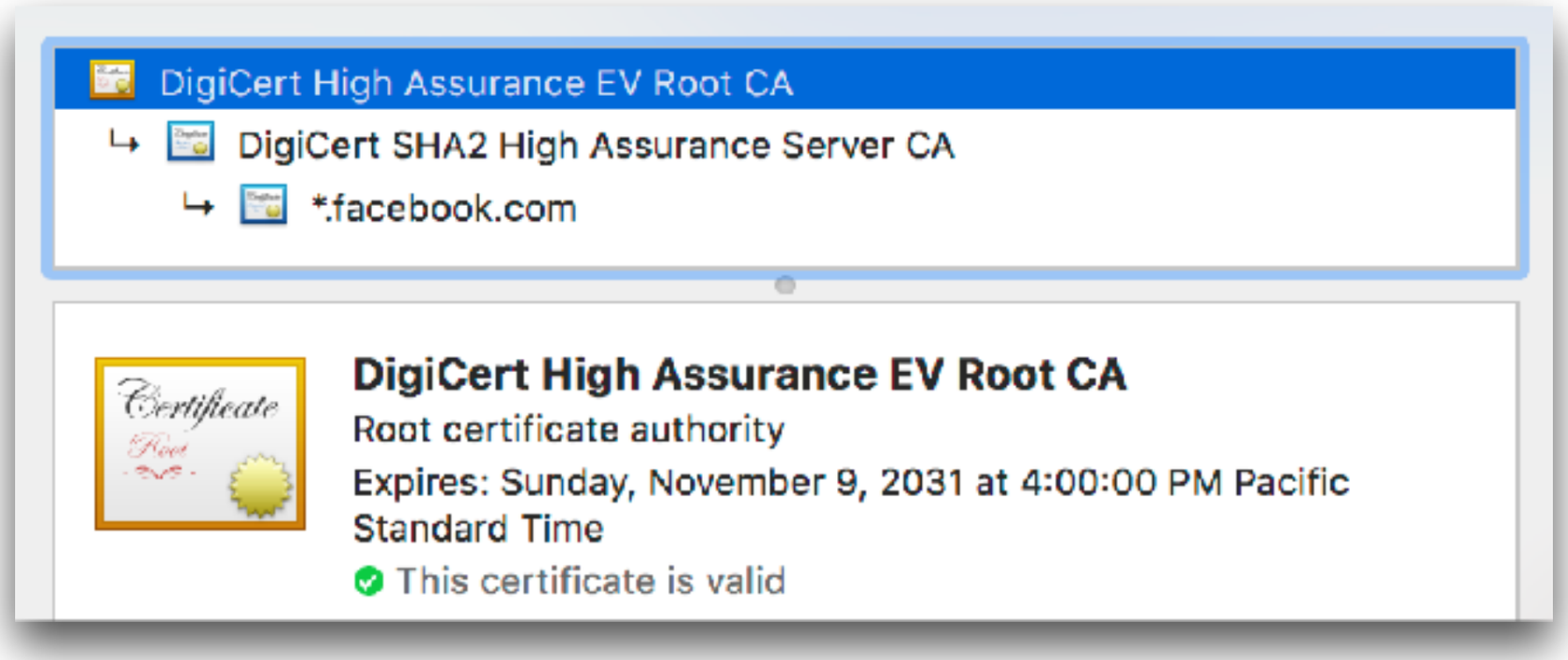
```
$ openssl dgst \
  -engine pkcs11 -keyform engine -sign slot_0 \
  -sha256 -out signature msg

$ openssl dgst -sha256 -verify pubkey.pem \
  -signature signature msg
Verified OK
```

Security

- Goals
  - Protect against key theft
  - Secure logging
  - Tamper resistance
- Non-goals
  - Key misuse

PKI Security

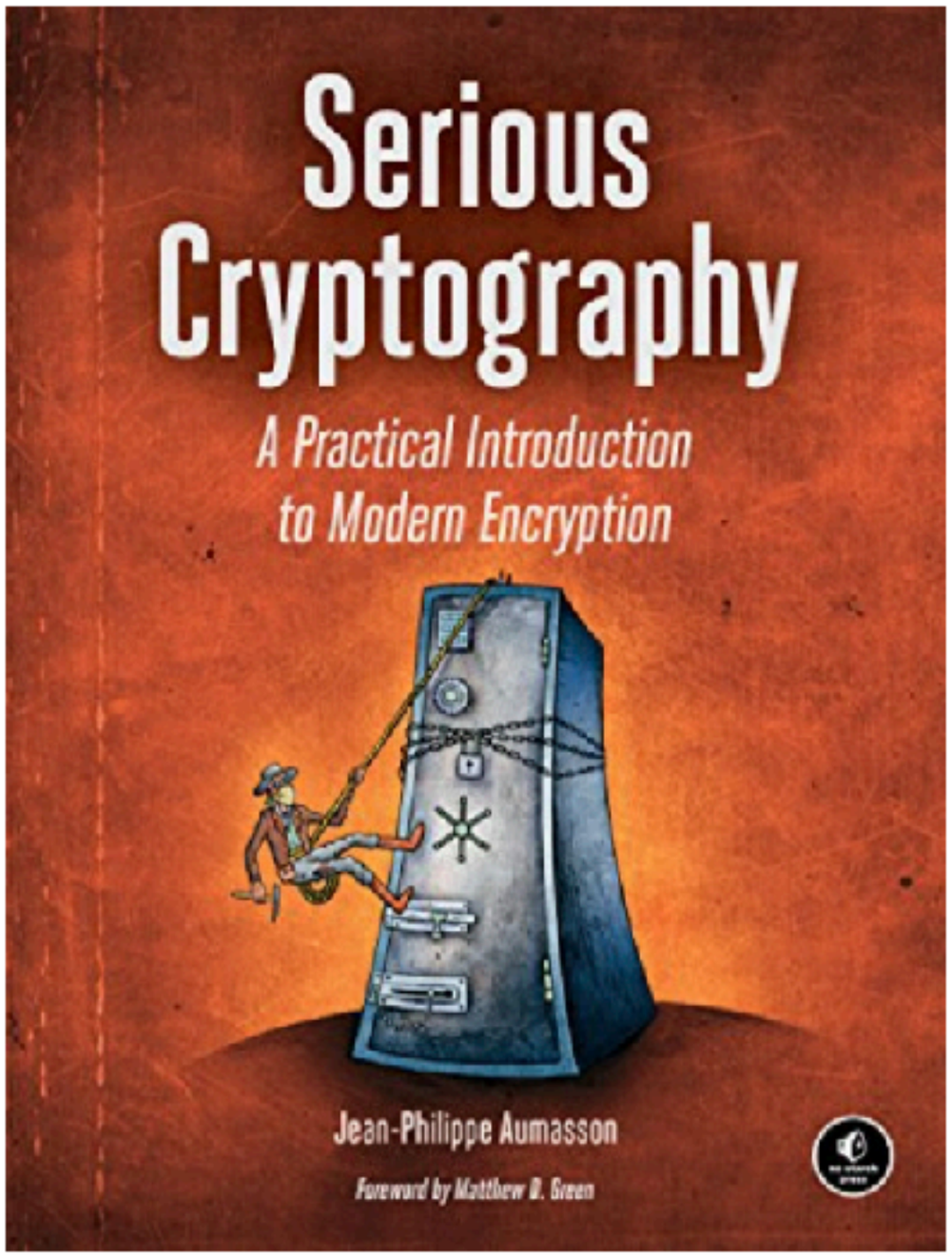


- Root CA is kept offline, valid for 30+ years
- Intermediate CAs are online, shorter validity
- Root can revoke intermediates
- Root cannot be revoked
- Root compromise is a major event (read about DigiNotar)

All command examples

https://github.com/mfdutra/scripts/blob/master/hsm_talk/examples.sh

Read more



http://a.co/5mOJek9

SSH blog post

http://bit.ly/ssh_fb

Q&A