PKCS#11

Graham Steel

LSV, ENS-Cachan

and

School of Informatics, University of Edinburgh

PKCS#11

- RSA standard interface for cryptographic devices
- Also known as ‘Cryptoki’
- Current version 2.20
- Clulow’s analysis: version 2.01

Goals: Standardisation, resource sharing, vendor neutrality. . .

WrapKey(.,.,.)

UnWrapKey(.,.,.)

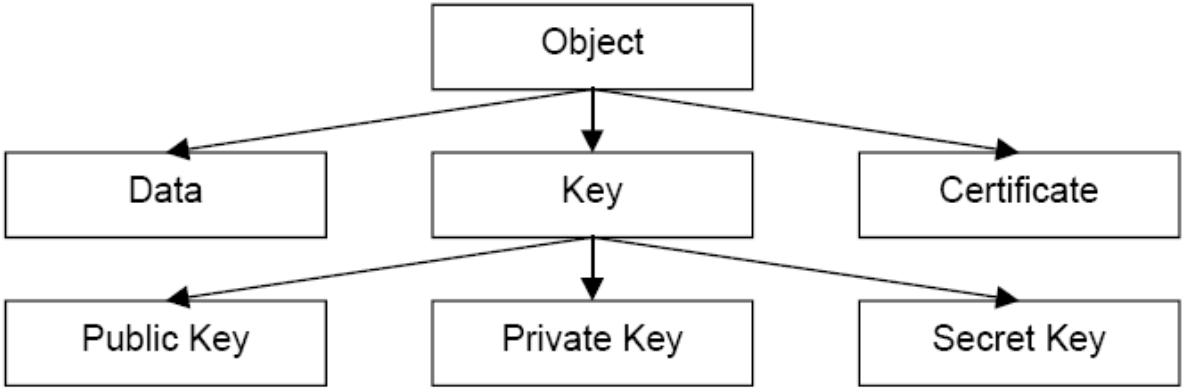
KeyGenerate(.,.,.)

EncryptData(.,.,.)

DecryptData(.,.,.)

PKCS#11

Model



Attributes: encrypt, decrypt, sign, check, wrap, unwrap, sensitive, extractable,. . .

Access and Sessions

- Two profiles: SO and user
- Identified by 4 digit PIN
- SO can set PINs
- Can be replaced by custom access control (e.g. smartcards)

Key Management - 1

Generate Key:

$$\text{Host} \rightarrow \text{Device} : \text{Genkey.}$$

$$\text{Device} \rightarrow \text{Host} : \mathsf{handle}(\mathsf{ks})$$

Generate Key Pair:

$$\text{Host} \rightarrow \text{Device} : \text{GenkeyPair.}$$

$$\text{Device} \rightarrow \text{Host} : \mathsf{handle}(\mathsf{k}_{\mathsf{priv}}), \mathsf{k}_{\mathsf{pub}}$$

Key Management - 2

Wrap:

$$\text{Host} \rightarrow \text{Device} : \text{wrap}, \text{handle}(k1), \text{handle}(k_s).$$

$$\text{Device} \rightarrow \text{Host} : \{k_s\}_{k1}$$

IF: $\text{att}(k1, \text{wrap}) \wedge \text{att}(ks, \text{extractable})$

Unwrap:

$$\text{Host} \rightarrow \text{Device} : \text{unwrap}, \text{handle}(k1), \{k_s\}_{k1}.$$

$$\text{Device} \rightarrow \text{Host} : \text{handle}(ks)$$

IF: $\text{att}(k1, \text{unwrap})$

Key Usage

$$\text{Host} \rightarrow \text{Device} : \text{encrypt, Data, handle(k1)}$$

$$\text{Device} \rightarrow \text{Host} : \{\text{Data}\}_{\text{k1}}$$

IF: $\text{att}(\text{k1}, \text{encrypt})$

Decrypt:

$$\text{Host} \rightarrow \text{Device} : \text{decrypt, } \{\text{Data}\}_{\text{k1}}\text{, handle(k1)}$$

$$\text{Device} \rightarrow \text{Host} : \text{Data}$$

IF: $\text{att}(\text{k1}, \text{decrypt})$

Key Separation Attack

Wrap:

$$\mathsf{I} \rightarrow \mathsf{Device} : \mathsf{wrap}, \mathsf{handle}(\mathsf{k1}), \mathsf{handle}(\mathsf{k_s}).$$

$$\mathsf{Device} \rightarrow \mathsf{I} : \{\mathsf{k_s}\}_{\mathsf{k1}}$$

Decrypt:

$$\mathsf{I} \rightarrow \mathsf{Device} : \mathsf{decrypt}, \{\mathsf{k_s}\}_{\mathsf{k1}}, \mathsf{handle}(\mathsf{k1})$$

$$\mathsf{Device} \rightarrow \mathsf{I} : \mathsf{k_s}$$

Fix(?) 1

No key may have both wrap and decrypt set

Wrap:

$$\text{Host} \rightarrow \text{Device} : \text{wrap}, \text{handle}(\text{k1}), \text{handle}(\text{k}_\text{s}).$$

$$\text{Device} \rightarrow \text{Host} : \{\text{k}_\text{s}\}_{\text{k1}}$$

Unset attribute wrap.

Set attribute decrypt.

Decrypt:

$$\text{I} \rightarrow \text{Device} : \text{decrypt}, \{\text{k}_\text{s}\}_{\text{k1}}, \text{handle}(\text{k1})$$

$$\text{Device} \rightarrow \text{I} : k_s$$

Fix(?) 2

Decrypt and Wrap attributes cannot be unset once set

Unwrap:

$$\mathsf{I} \rightarrow \mathsf{Device} : \mathsf{unwrap}, \mathsf{handle}(\mathsf{k1}), \{k_s\}_{k1}.$$

$$\mathsf{Device} \rightarrow \mathsf{I} : \mathsf{handle}(\mathsf{ks})$$

Unwrap again:

$$\mathsf{I} \rightarrow \mathsf{Device} : \mathsf{unwrap}, \mathsf{handle}(\mathsf{k1}), \{k_s\}_{k1}.$$

$$\mathsf{Device} \rightarrow \mathsf{I} : \mathsf{handle}(\mathsf{ks}')$$

Trojan Public Key

PK version of ‘Wrap key’ accepts a key, not a handle

Wrap:

$$\mathsf{I} \rightarrow \mathsf{Device} : \mathsf{wrap}, \mathsf{K_I}, \mathsf{handle}(\mathsf{k_s}).$$

$$\mathsf{Device} \rightarrow \mathsf{I} : \{k_s\}_{K_I}$$

Trojan Symmetric Key

I knows $\mathsf{k1}_{\mathsf{pub}}$

Unwrap:

$$\mathsf{I} \rightarrow \mathsf{Device} : \mathsf{unwrap}, \mathsf{handle}(\mathsf{k1}_{\mathsf{priv}}), \{k_I\}_{k1_{pub}}.$$

$$\mathsf{Device} \rightarrow \mathsf{I} : \mathsf{handle}(\mathsf{K_I})$$

Weaker Key Export

Using ECB mode export double-length key $k =< k1, k2 >$ under single length key k_s

Wrap:

$$\mathsf{I} \rightarrow \mathsf{Device} : \mathsf{wrap}, \mathsf{handle}(\mathsf{k_s}), \mathsf{handle}(\mathsf{k}).$$

$$\mathsf{Device} \rightarrow \mathsf{I} : \{\mathsf{k}\}_{\mathsf{k_s}} = \{\mathsf{k1}\}_{\mathsf{k_s}}, \{\mathsf{k2}\}_{\mathsf{k_s}}$$

Export k_s under itself and crack offline

Wrap:

$$\mathsf{I} \rightarrow \mathsf{Device} : \mathsf{wrap}, \mathsf{handle}(\mathsf{k_s}), \mathsf{handle}(\mathsf{k_s}).$$

$$\mathsf{Device} \rightarrow \mathsf{I} : \{\mathsf{k_s}\}_{\mathsf{k_s}}$$

Modelling

- Have model in AVISPA IF format
  - Handle as hash function with nonce
  - Global set of attributes
- Can detect some attacks and validate some fixes
- BUT several problems remain

Questions

- Just bin PKCS#11?

  or provide a verified configuration?

  or analysed fixes proposed by industry?

- Any hope of a cryptographically sound verification?

  Weak encryption algorithms

  Key cycles

Summary

- PKCS#11 without restrictions is broken
- Work in progress on fixing (at DY + algebra level)
- Future aim: cryptographically sound proof