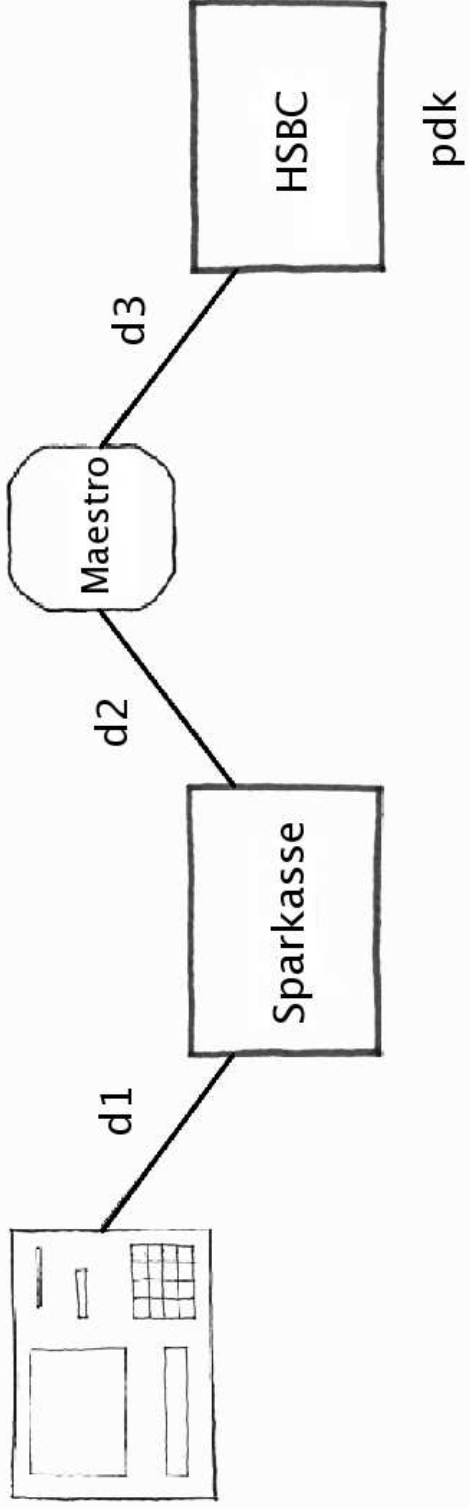


A Formal Theory of Key Conjuring

Véronique Cortier, Stéphanie Delaune, Graham Steel



 School of
informatics







km
data, pin, imp, ...

data, pin, imp, ...

{ d1 } $km \oplus data$

{ pdk1 } $km \oplus pin$

CCA API

Encrypt Data:

Host \rightarrow HSM : $\{ \text{xkey} \}_{\text{km} \oplus \text{data}}, \text{xmsg}$

HSM \rightarrow Host : $\{ \text{xmsg} \}_{\text{xkey}}$

Export Key:

Host \rightarrow HSM : $\{ \text{KEK} \}_{\text{km} \oplus \text{exp}}, \text{data}, \{ \text{D1} \}_{\text{km} \oplus \text{data}}$

HSM \rightarrow Host : $\{ \text{D1} \}_{\text{KEK} \oplus \text{data}}$

Formal Modelling

Encrypt data v0.1

$$\text{xmsg}, \{\text{xkey}\}_{\text{km} \oplus \text{data}} \rightarrow \{\text{xmsg}\}_{\text{xkey}}$$

Encrypt Data v0.2

$$x, y \rightarrow \{X\}_{\text{dec}(y, \text{km} \oplus \text{data})}$$

Encrypt Data v0.21

$$\text{chkOdd}(\text{dec}(y, \text{km} \oplus \text{data})), x, y \rightarrow \{X\}_{\text{dec}(y, \text{km} \oplus \text{data})}$$

Key Conjuring

$$\text{chkOdd}(\text{dec}(y, km \oplus \text{data})), x, y \rightarrow \{X\}_{\text{dec}(y, km \oplus \text{data})}$$

Attacker tries random values instead of y

Obtains an encrypted data key.

Encrypt Data:

$$x \xrightarrow{\text{new } n} \{X\}_{\text{dec}(n, km \oplus \text{data})}, n, \text{chkOdd}(\text{dec}(n, km \oplus \text{data}))$$

More Parity Checks

Key Import:

Host \rightarrow HSM : $\{ \text{KEY1} \}_{\text{KEK} \oplus \text{TYPE}}, \text{TYPE}, \{ \text{KEK} \}_{\text{km} \oplus \text{imp}}$

HSM \rightarrow Host : $\{ \text{KEY1} \}_{\text{km} \oplus \text{TYPE}}$

$y, \text{xtype}, z \rightarrow \{ \text{dec}(y, \text{dec}(z, \text{km} \oplus \text{imp}) \oplus \text{xtype}) \}_{\text{km} \oplus \text{xtype}}$

$\text{chkE}(\text{xtype})$

$\text{chkO}(\text{dec}(z, \text{km} \oplus \text{imp}))$

$\text{chkO}(\text{dec}(y, \text{dec}(z, \text{km} \oplus \text{imp}) \oplus \text{xtype}))$

More Key Conjuring

$y, \text{xtype}, z \rightarrow \{\text{dec}(y, \text{dec}(z, \text{km} \oplus \text{imp}) \oplus \text{xtype})\}_{\text{km} \oplus \text{xtype}}$

$\text{chkE}(\text{xtype})$

$\text{chkO}(\text{dec}(z, \text{km} \oplus \text{imp}))$

$\text{chkO}(\text{dec}(y, \text{dec}(z, \text{km} \oplus \text{imp}) \oplus \text{xtype}))$

Key Import:

$y, \text{xtype} \xrightarrow{\text{new } n} \{\text{dec}(y, \text{dec}(n, \text{km} \oplus \text{imp}) \oplus \text{xtype})\}_{\text{km} \oplus \text{xtype}, n}$

$\text{chkE}(\text{xtype})$

$\text{chkO}(\text{dec}(y, \text{dec}(n, \text{km} \oplus \text{imp}) \oplus \text{xtype}))$

More Key Conjuring

$y, \text{xtype}, z \rightarrow \{\text{dec}(y, \text{dec}(z, \text{km} \oplus \text{imp}) \oplus \text{xtype})\}_{\text{km} \oplus \text{xtype}}$

$\text{chkE}(\text{xtype})$

$\text{chkO}(\text{dec}(z, \text{km} \oplus \text{imp}))$

$\text{chkO}(\text{dec}(y, \text{dec}(z, \text{km} \oplus \text{imp}) \oplus \text{xtype}))$

xtype, z	$\xrightarrow{\text{new } n}$	$\{\text{dec}(n, \text{dec}(z, \text{km} \oplus \text{imp}) \oplus \text{xtype})\}_{\text{km} \oplus \text{xtype}},$
$\text{chkE}(\text{xtype}),$		n
$\text{chkO}(\text{dec}(z, \text{km} \oplus \text{imp}))$		$\text{chkO}(\text{dec}(n, \text{dec}(z, \text{km} \oplus \text{imp}) \oplus \text{xtype}))$

More Key Conjuring

$y, \text{xtype}, z \rightarrow \{\text{dec}(y, \text{dec}(z, km \oplus \text{imp}) \oplus \text{xtype})\}_{km \oplus \text{xtype}}$

$\text{chkE}(\text{xtype})$

$\text{chkO}(\text{dec}(z, km \oplus \text{imp}))$

$\text{chkO}(\text{dec}(y, \text{dec}(z, km \oplus \text{imp}) \oplus \text{xtype}))$

$y, z \xrightarrow{\text{new } n} \{\text{dec}(y, \text{dec}(z, km \oplus \text{imp}) \oplus n)\}_{km \oplus n}, n$

$\text{chkO}(\text{dec}(z, km \oplus \text{imp}))$

$\text{chkO}(\text{dec}(y, \text{dec}(z, km \oplus \text{imp}) \oplus n))$

$\text{chkE}(n)$

Class of Rules

An API rule is a rule of the form $\text{chk}_1(u_1), \dots, \text{chk}_k(u_k), x_1, \dots, x_n \rightarrow t$, where

- x_1, \dots, x_n are variables,
- t is a term such that $\text{vars}(t) \subseteq \{x_1, \dots, x_n\}$,
- u_1, \dots, u_k are terms of Base type not headed with \oplus ,
- $\text{chk}_i \in \{\text{chkOdd}, \text{chkEven}\}$, $1 \leq i \leq k$.

Assume that the rule only involves *pure* terms.

Formal Transformation

Let $R_l \rightarrow R_r = \text{chk}_1(u_1), \dots, \text{chk}_k(u_k), x_1, \dots, x_n \rightarrow t$

for $u_j = \text{dec}(z, t')$, let $\sigma = \{z \mapsto n\}$

$$(R_l \setminus \{z, \text{chk}_j(u_j)\} \xrightarrow{\text{new } n} R_r \cup \{z, \text{chk}_j(u_j)\})\sigma$$

and for composite t' ... see the paper.

Formal Results

- Soundness of some reductions
 - e.g. offline key conjuring not necessary
- Decidability of a certain class
 - Finite number of key conjuring steps

Decidability proof:

Pre-compute all possible reductions by enc/dec eqns

Show terms stay in normal form after instantiation

Conclusions

- Have proposed a defn of 'key conjuring', and a formal transform
- Models e.g. Bond import/export attack (see paper)
- Future work:
 - Extend class
 - Conjuring to arbitrary depth
 - Implementation