Attacks in the multi-user setting: Discrete logarithm, Even-Mansour and PRINCE

Pierre-Alain Fouque[1], Antoine Joux[2], Chrysanthi Mavromati[3]

[1] Université Rennes 1 and Institut Universitaire de France
[2] CryptoExperts and Chaire de Cryptologie de la Fondation de l'UPMC
[3] Sogeti/ESEC R&D Lab and UVSQ Laboratoire PRiSM

12 June 2014

The multi-user setting

Cryptographers prove the security of their schemes in a single-user model.

In real world: There are many users, each with a different key, sending each other encrypted data.

Multi-user setting

Main ideas

- Graph of key relations
- New variant of memory-less collision attacks

Generic discrete logarithm

- Single-user discrete log: time $\sqrt{N}$ (generic group)
- Multi-user discrete log (L logs):
  - studied by Kuhn and Struik
  - use of the parallel version of the Pollard rho technique with distinguished points
  - time $\sqrt{NL}$, $L \leq N^{1/4}$

Distinguished points for discrete logarithms

- Define a random function $f : \mathcal{G} \to \mathcal{G}$

$$f(z) = \begin{cases} z^2 & \text{if } z \in \mathcal{G}_1, \\ gz & \text{if } z \in \mathcal{G}_2, \end{cases}$$

  where $\mathcal{G}_1 \cup \mathcal{G}_2 = \mathcal{G}$.
- Define a distinguished subset S_0
- Build chains from random startpoints: $z_{i+1} = f(z_i)$
- Stop chain when $z_\ell = d \in S_0$

$$g^{x_1} = y_1 \xrightarrow{f} y_2 \xrightarrow{f} y_3 \xrightarrow{f} y_4 \xrightarrow{f} \qquad log_g d = A x_1 + B$$

$$d = d'$$

$$g^{x_1'} = y_1' \xrightarrow{f} y_2' \xrightarrow{f} y_3' \xrightarrow{f} y_4' \xrightarrow{f} \qquad log_g d' = A' x_1' + B'$$

New method

$$g^{x^{(0)}} = y_0^{(0)} \xrightarrow{f} \cdots \xrightarrow{f} \cdots$$

$$g^{x^{(1)}} = y_0^{(1)} \xrightarrow{f} \cdots \xrightarrow{f} \cdots$$

$$\xrightarrow{f} \cdots \xrightarrow{f}$$

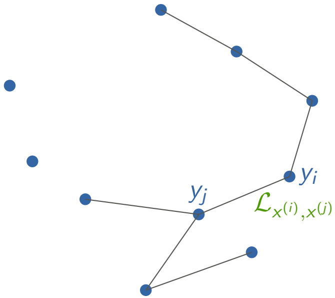
$$\vdots$$

$$\xrightarrow{f} \cdots \xrightarrow{f}$$

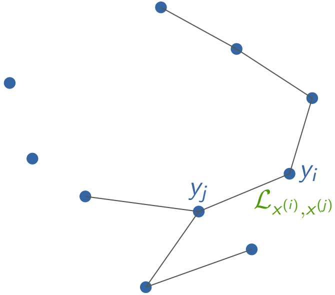
linear relation between $x^{(i)}$ and $x^{(j)}$

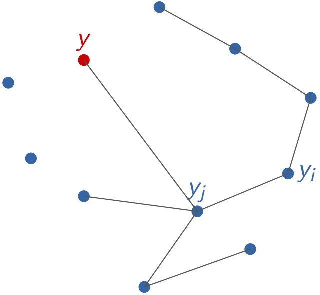
$$g^{x^{(L)}} = y_0^{(L)} \xrightarrow{f} \cdots \xrightarrow{f} \cdots$$

New method - Construct the graph



New method - Construct the graph



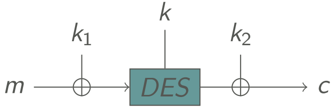


$\rightarrow$ learn all keys in connected component

Description of Even-Mansour

Introduced by Even and Mansour at [Asiacrypt '91].

- motivated by the DESX construction [Rivest, 1984]

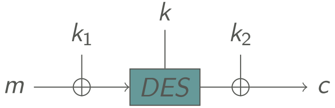


DES key k, whitening keys k_1, k_2

Description of Even-Mansour

Introduced by Even and Mansour at [Asiacrypt '91].

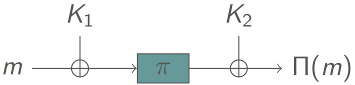
- motivated by the DESX construction [Rivest, 1984]



DES key k, whitening keys k_1, k_2

- minimal construction of a blockcipher

$$\Pi_{K_1,K_2}(m) = \pi(m \oplus K_1) \oplus K_2$$



- keyed permutation family Π_{K_1,K_2}
- π is a public permutation on n-bit values ($N = 2^n$)
- two whitening keys K_1, K_2 of n-bits

Known results in the single-user model

Main result: Any attack with D queries to Π and T off-line computation (queries to the public permutation π) has an upper bound of $O(DT/2^n)$ on probability of success.

Single-Key EM: Proved secure with the same bound [Dunkelman *et al.*]

Slidex attack - Single key case

[Dunkelman *et al.*, 2012]

Assume that two plaintexts $(P, P^{'})$ satisfy $P \oplus P^{'} = K$ (slid pair).

Slidex attack - Single key case

[Dunkelman *et al.*, 2012]

Assume that two plaintexts $(P, P^{'})$ satisfy $P \oplus P^{'} = K$ (slid pair).

Apply the Davies-Meyer construction to Π and π:

$$F(P) = \Pi(P) \oplus P \text{ and } f(P) = \pi(P) \oplus P$$

$$\begin{aligned} F(P^{'}) &= \Pi(P^{'}) \oplus P^{'} = \Pi(P \oplus K) \oplus P \oplus K \\ &= \pi(P \oplus \cancel{K} \oplus \cancel{K}) \oplus \cancel{K} \oplus P \oplus \cancel{K} \\ &= \pi(P) \oplus P = f(P) \end{aligned}$$

$$\Rightarrow F(P^{'}) = f(P)$$

Slidex attack - Single key case

[Dunkelman *et al.*, 2012]

Assume that two plaintexts $(P, P^{'})$ satisfy $P \oplus P^{'} = K$ (slid pair).

Apply the Davies-Meyer construction to Π and π:

$$F(P) = \Pi(P) \oplus P \text{ and } f(P) = \pi(P) \oplus P$$

$$\begin{aligned} F(P^{'}) &= \Pi(P^{'}) \oplus P^{'} = \Pi(P \oplus K) \oplus P \oplus K \\ &= \pi(P \oplus \cancel{K} \oplus \cancel{K}) \oplus \cancel{K} \oplus P \oplus \cancel{K} \\ &= \pi(P) \oplus P = f(P) \end{aligned}$$

$$\Rightarrow F(P^{'}) = f(P)$$

Find a collision,

$$\boxed{\pi(\mathbf{P}) \oplus \mathbf{P} = \mathbf{\Pi}(\mathbf{P}^{'}) \oplus \mathbf{P}^{'}}$$

Then, $P \oplus P^{'}$ is a good candidate for K.

Slidex attack - Extending to the two key case

Fix $\delta \in \{0,1\}^n$
Assume that two plaintexts (P, P') satisfy:

$$P \oplus P' = K_1 \text{ or } P \oplus P' = K_1 \oplus \delta.$$

$$F(P) = \Pi(P) \oplus \Pi(P \oplus \delta) \quad \text{and} \quad f(P) = \pi(P) \oplus \pi(P \oplus \delta)$$

$$\Rightarrow F(P') = f(P) \quad \text{and} \quad F(P' \oplus \delta) = f(P)$$

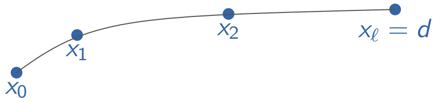
Find a collision,

$$\boldsymbol{\Pi}(\mathbf{P}') \oplus \boldsymbol{\Pi}(\mathbf{P}' \oplus \delta) = \pi(\mathbf{P}) \oplus \pi(\mathbf{P} \oplus \delta)$$

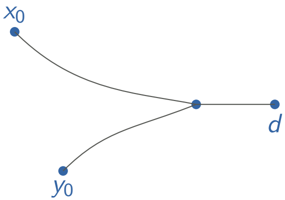
Then, $P \oplus P'$ and $P \oplus P' \oplus \delta$ are good candidates for K_1.

The distinguished points method

- Define a function f on a set S of size N.
- Define a distinguished subset S_0 of S
- Build chains from random startpoints: $x_{i+1} = f(x_i)$
- Stop chain when $x_\ell = d \in S_0$
- Store (x_0, d, ℓ)

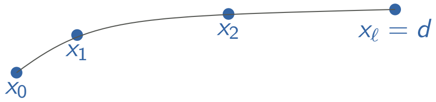


How do we construct a collision?



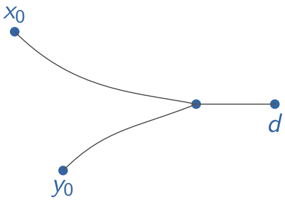
The distinguished points method

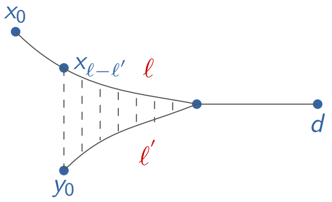
- Define a function f on a set S of size N.
- Define a distinguished subset S_0 of S
- Build chains from random startpoints: $x_{i+1} = f(x_i)$
- Stop chain when $x_\ell = d \in S_0$
- Store (x_0, d, ℓ)



How do we construct a collision?

How do we recover a chain?





Application on Even-Mansour - First trial

Goal: Find a collision between a set of chains using the public permutation π and a chain obtained from the keyed permutation Π

Define $F(P) = \Pi(P) \oplus \Pi(P \oplus \delta)$ and $f(P) = \pi(P) \oplus \pi(P \oplus \delta)$

$\rightarrow$ These chains can cross but not merge

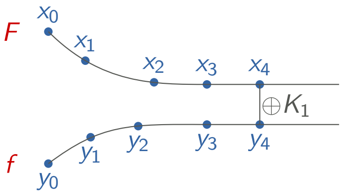
Application on Even-Mansour - New idea

Define new functions:

$$F(P) = P \oplus \Pi(P) \oplus \Pi(P \oplus \delta) \text{ and}$$
$$f(P) = P \oplus \pi(P) \oplus \pi(P \oplus \delta)$$

- Assume that two plaintexts $(P, P^{'})$ satisfy: $P^{'} = P \oplus K1$ or $P^{'} = P \oplus K_1 \oplus \delta$
- Then $F(P^{'}) = f(P) \oplus K_1(\text{resp.} \ \oplus \delta)$

$\rightarrow$ These chains can become parallel



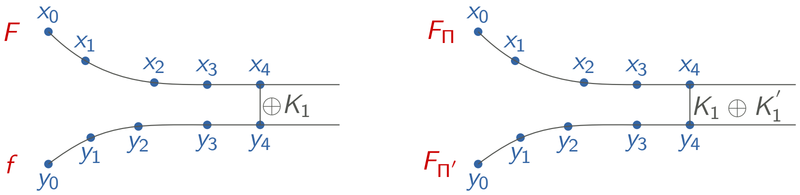
Application on Even-Mansour - New idea

Define new functions:

$$F(P) = P \oplus \Pi(P) \oplus \Pi(P \oplus \delta) \text{ and}$$
$$f(P) = P \oplus \pi(P) \oplus \pi(P \oplus \delta)$$

- Assume that two plaintexts $(P, P^{'})$ satisfy:
  $P^{'} = P \oplus K1$ or $P^{'} = P \oplus K_1 \oplus \delta$
- Then $F(P^{'}) = f(P) \oplus K_1(\text{resp.} \ \oplus \delta)$

$\rightarrow$ These chains can become parallel



Detection of parallel chains with distinguished points

- For f chains: define a distinguished point P as a point with a value of $\pi(P) \oplus \pi(P \oplus \delta) \in S_0$

- For F chains: define a distinguished point $P^{'}$ as a point with a value of $\Pi(P^{'}) \oplus \Pi(P^{'} \oplus \delta) \in S_0$

- If $P^{'} = P \oplus K1$ and P is a distinguish point in the f chain, then:

$$\begin{aligned} \Pi(P^{'}) \oplus \Pi(P^{'} \oplus \delta) &= \pi(P^{'} \oplus K_1) \oplus \cancel{K_2} \oplus \pi(P^{'} \oplus K_1 \oplus \delta) \oplus \cancel{K_2} \\ &= \pi(P) \oplus \pi(P \oplus \delta) \end{aligned}$$

  and then $P^{'}$ is a distinguished point in the F chain

- $\rightarrow P \oplus P^{'} = K_1$

New attack on Even-Mansour

- Build chains from $f(P) = P \oplus \pi(P) \oplus \pi(P \oplus \delta)$
  - Stop if $\pi(P) \oplus \pi(P \oplus \delta)$ arrives at a distinguished point
- Build chains from $F(P) = P \oplus \Pi(P) \oplus \Pi(P \oplus \delta)$
  - Stop if $\Pi(P) \oplus \Pi(P \oplus \delta)$ arrives at a distinguished point
- These chains cannot merge but can become parallel
  - Assume $P^{'} = P \oplus K_1$ or $P^{'} = P \oplus K_1 \oplus \delta$
  - $\rightarrow F(P^{'}) = f(P) \oplus K_1$ ($\oplus\delta$ respectively)
- We only need to store endpoints (don't have to recompute chains)

Attack Even-Mansour in the multi-user setting

- Build chains from f of length $2^{n/3}$
- Build chains from F of length $2^{n/3}$ for each user
- Construct a graph:
  - Nodes are labelled by the users and the unkeyed user
  - If $F^{(i)} = F^{(j)}$ (for users $(i), (j)$), then add a vertex between the two nodes
  - $\rightarrow K_1^{(i)} \oplus K_1^{(j)} (\oplus \delta)$
  - If we find a single collision between a user and the unkeyed user, then we learn all keys (in the connected component)

Analysis of the attack:

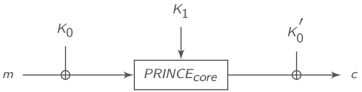
For $2^{n/3}$ users, $2^{n/3}$ queries/user, $2^{n/3}$ unkeyed queries
$\rightarrow$ recover almost all $2^{n/3}$ keys

Description of PRINCE

PRINCE [Borghoff *et al.*, Asiacrypt 2012]

- 64-bit lightweight block cipher
- 128-bit key k split into equal parts: $k = k_0 \| k_1$
- extension to 192 bit: $k = (k_0 \| k_1) \rightarrow (k_0 \| k_0^{'} \| k_1)$
- $k_0^{'}$ derived from k_0 by using the linear function L':
  $L'(k_0) = (k_0 \ggg 1) \oplus (k_0 \gg 63)$
- α-reflection property

$$\forall (k_0 \| k_0^{'} \| k_1),\ D_{(k_0 \| k_0^{'} \| k_1)}(\cdot) = E_{(k_0^{'} \| k_0 \| k_1 \oplus \alpha)}(\cdot)$$



$$E_k(m) = k_0^{'} \oplus Pcore_{k_1}(m \oplus k_0)$$

Attacks on PRINCE in the single and multi-user setting

Attack in the multi-user setting

Total cost 2^{65} operations for deducing k_0 and k_1 of 2 users in a set of 2^{32}.

Attack in the single-user setting

$$T_{off} = 2^{96}, T_{on} = 2^{32}, M = 2^{32}$$

$$\mathbf{DT_{off}} = 2^{128}$$

Conclusion

- Propose two new algorithmic ideas to improve collision based attacks
- Application of the first idea to solve the discrete logarithm problem in the multi-user setting
- Application of both ideas to the Even-Mansour scheme
- Propose two new attacks for PRINCE
  - The attacks have been applied to DESX with some differences

Conclusion

- Propose two new algorithmic ideas to improve collision based attacks
- Application of the first idea to solve the discrete logarithm problem in the multi-user setting
- Application of both ideas to the Even-Mansour scheme
- Propose two new attacks for PRINCE
  - The attacks have been applied to DESX with some differences

Thank you for your attention!