Even-Mansour and the multi-users setting

Antoine Joux

ESC'2013 – Mondorf-les-bains

January 17^{th},2013

A reminder on Even-Mansour

- Let π be a public permutation on n-bit values, $N = 2^n$
- Define :

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

- Security against attacks that uses D queries to Π_{K_1,K_2} and T to π:
  - Success probability is upper bounded by $O(DT/N)$
  - Many known attacks [D91], [BW2000], [DKS2012]

Simplest known attack

- Appears in eprint version of [DKS2012]
- Find a collision:

$$\begin{aligned} \pi(P) \oplus \pi(P') &= \Pi(P) \oplus \Pi(P') \quad \text{or} \\ \pi(P) \oplus \Pi(P) &= \pi(P') \oplus \Pi(P'). \end{aligned}$$

- Then $P \oplus P'$ is a good candidate for K_1
- And $\pi(P) \oplus \Pi(P')$ is a good candidate for K_2
- Can be done in a memoryless way: Using cycle finding on:

$$\pi(P) \oplus \Pi(P).$$

Simplest known attack – Single key case

- Also from [DKS2012]
- Find a collision:

$$\pi(P) \oplus P \quad = \quad \Pi(P') \oplus P'.$$

- Then $P \oplus P'$ is a good candidate for K
- Non adaptive (using memory T) and any D/T ratio

Adapting to the two key case

- Fix δ and find a collision:

$$\pi(P) \oplus \pi(P \oplus \delta) \quad = \quad \Pi(P') \oplus \Pi(P' \oplus \delta).$$

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

Can we do better ?

- Reduce memory for all D/T ratio ?
- Reduce memory and remain adaptive ?

- Natural option: Use distinguished point technique ?

Distinguished point technique

- Basic application: invert a random function F
- Precomputation: build chains $F^*(X_i)$ ending on a distinguished point. Store startpoint, endpoint, length.
- Starting from $Y = F(X)$ compute chain.
- When find distinguished point, lookup chain, go back to start and recompute.

Trial 1

- Build chains from $\pi(P) \oplus \pi(P \oplus \delta)$
- Build chains from $\Pi(P) \oplus \Pi(P \oplus \delta)$
- Problem: They can cross but not merge !

New option

- Build chains from $f(P) = P \oplus \pi(P) \oplus \pi(P \oplus \delta)$
- Stop on distinguished point if $\pi(P) \oplus \pi(P \oplus \delta)$ satisfies some DP property.
- Build chains from $F(P) = P \oplus \Pi(P) \oplus \Pi(P \oplus \delta)$
- These chains don't merge either
- They can become **parallel**:
  - Assume $P' = P \oplus K_1$ or $P' = P \oplus K_1 \oplus \delta$
  - Then $F(P') = f(P) \oplus K_1$ (resp. $\oplus\delta$)
- Thanks to this:
  - Don't need to recompute chains, only store endpoints
- Use your favorite number of chains/length of chains/keyed request compromise

Multi-user setting

- Basic attack 1
  - Build $2^{n/3}$ chains of length $2^{n/3}$ ($T = N^{2/3}$)
  - Then $D = 2^{n/3}$ for each user
  - Breaks constant fraction of keys
- Basic attack 2
  - Build one chain of length $2^{n/3}$
  - Then $D = 2^{n/3}$ for each user
  - Breaks about one key

Multi-user setting – New attack

- Start as basic attack 2
  - Build a few chains of length $2^{n/3}$ for π
  - Then a few chains of length $2^{n/3}$ for each user
- Build a graph:
  - Nodes: Users nodes + Unkeyed node
  - Vertices: Between two nodes have a collision
  - A vertex means xor of keys known (up to δ)
- Properties of the graph
  - Expects $O(2^{n/3})$ vertices
  - Threshold effect: graph gains a large connected component
- Learns large fraction of keys simultaneously

Conclusion

Questions ?