New Slide Attacks on Almost Self-Similar Ciphers

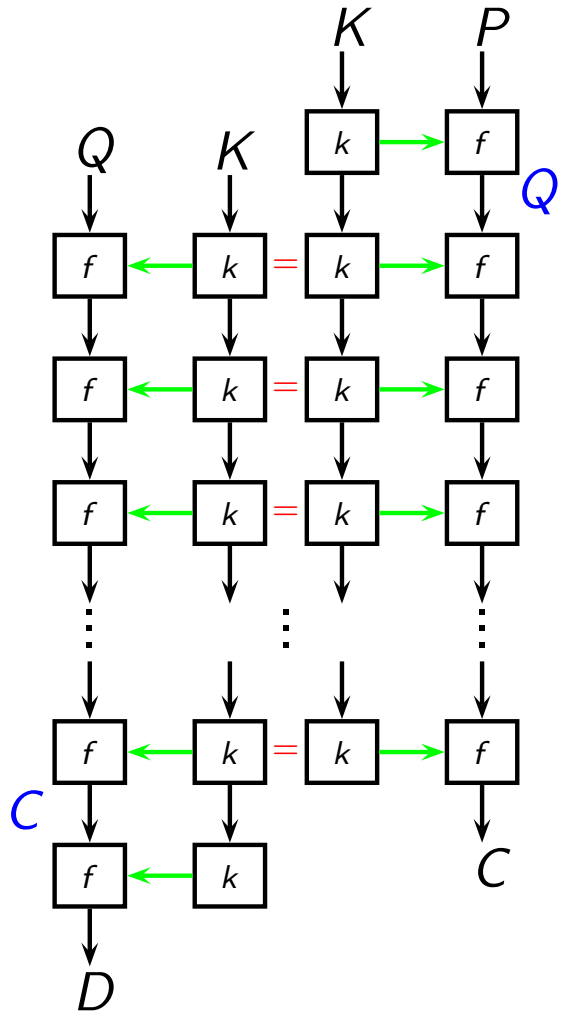
Orr Dunkelman, Nathan Keller,
Noam Lasry, Adi Shamir

Eurocrypt 2020

Outline

Slide Attacks [BW99]

- Adaptation of Related-Key Attacks [B93,K92] to the case where the key is self-related.
- Can be applied to ciphers with the same keyed permutation.
- Independent in the number of rounds of the cipher.



Slide Attacks [BW99] (cont.)

- Slid pair satisfies

$$\begin{cases} Q = f_k(P), \\ D = f_k(C), \end{cases} \tag{1}$$

- Slide attacks are composed of two main steps:
  - Find such a slid pair,
  - Use slid pair to extract key.
- Actually, in many attacks the way to verify that a given pair is a slid pair, is to verify that it suggested the correct key.

Generating Slid Pairs

- At random (pick $2^{n/2}$ known plaintexts for n-bit block),
- For Feistels of different types, one can find pairs:
  - 1K-DES — in $2^{n/4}$ chosen plaintexts [BW99],
  - 2K-DES — in $2^{n/4}$ chosen plaintexts or $2^{n/4}$ chosen ciphertexts [BW00],
  - 4K-DES — in $2^{n/4}$ chosen plaintexts and ciphertexts [BW00].

Generating Slid Pairs — Chains [F01]

- Given a slid pair (P, Q), their ciphertexts (C, D) are also a slid pair!
- Actually, if (P, Q) are slid pairs, so does $(E_k^\ell(P), E_k^\ell(Q))$ for any ℓ.
- This is useful when the attack of $f(\cdot)$ requires more than a single slid pair.

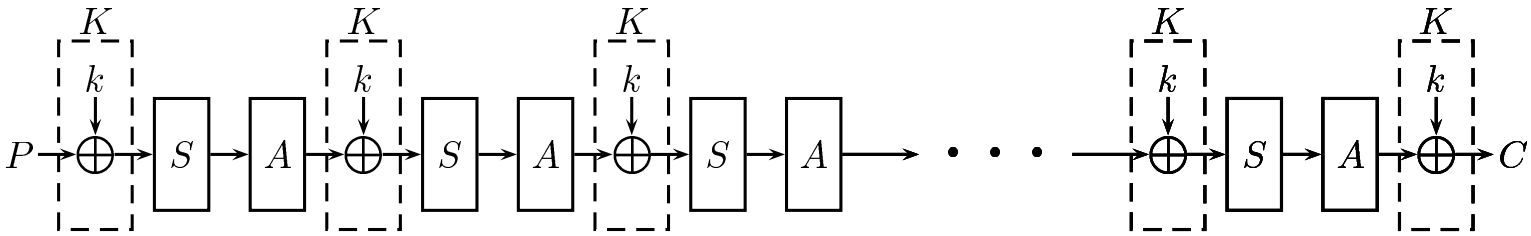
Other Extensions and Generalizations

- Slide detection using cycles [BDK07]
- Reflection attacks [K08]
- Slidex [DKS12]
- Quantum slide attacks [B+18]

Several Applications of Slide Attacks

- 1K-DES, 2K-DES, 4K-DES ([BW99,BW00])
- 3K-DES ([B+18])
- 1K-AES ([B+18])
- Misty1 ([DK15])
- KeeLoq ([I+08,C+08,...])
- FF3 ([DV17,HMT19])

A Generic SPN (1K-AES)



- If (P, Q) are a slid pair, then

$$Q = A(S(K(P))) = A(S(P \oplus k))$$

which can be re-written as

$$P \oplus k = S^{-1}(A^{-1}(Q))$$

- As S and A are unkeyed, we can easily compute $Q' = S^{-1}(A^{-1}(Q))$.

A Slide Attack on 1K-AES [B+18]

- Take $2^{n/2}$ known plaintexts.
- A slid pair (P, Q) (and corresponding ciphertext (C, D)) satisfies:

$$\begin{cases} Q = A(S(P \oplus k)) \\ D = A(S(C)) \oplus k \end{cases} \tag{2}$$

- Or in other words:

$$P \oplus Q' = k = D \oplus A(S(C))$$

- Which allows immediate identification (as $P \oplus A(S(C)) = Q' \oplus D$).

The Basic Assumption of Slide Attacks

- All the round functions are the same,

The Basic Assumption of Slide Attacks

- All the round functions are the same,
- It is possible to generate chains (because of the previous assumption).

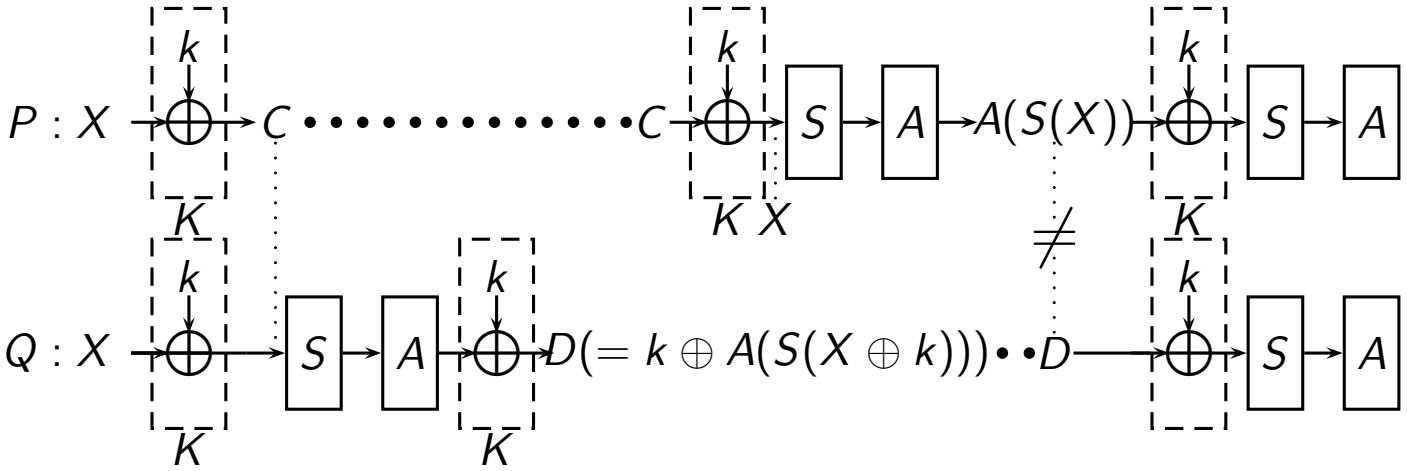
The Basic Assumption of Slide Attacks

- All the round functions are the same,
- It is possible to generate chains (because of the previous assumption).

Problem: in SPNs

the last round is different!

Last Round Function $\Rightarrow$ No Slid Chains



Not All SPNs are the Same

- Many SPNs have a different last round,

Not All SPNs are the Same

- Many SPNs have a different last round,
- For example, AES has no MixColumns in the last round.
- This complicates things even more — the relation between the ciphertexts of the slid pair is more complicated!

Not All SPNs are the Same

- Many SPNs have a different last round,
- For example, AES has no MixColumns in the last round.
- This complicates things even more — the relation between the ciphertexts of the slid pair is more complicated!
- Consider 1K-AES with the last round without MixColumns. Then

$$\begin{cases} Q = ARK(MC(SR(SB(P)))) \\ \end{cases}$$

Not All SPNs are the Same

- Many SPNs have a different last round,
- For example, AES has no MixColumns in the last round.
- This complicates things even more — the relation between the ciphertexts of the slid pair is more complicated!
- Consider 1K-AES with the last round without MixColumns. Then

$$\begin{cases} Q = ARK(MC(SR(SB(P)))) \\ \Rightarrow \\ D = ARK(SR(SB(ARK(MC(ARK(C)))))) \end{cases} \quad (3)$$

Slid Sets

- A slid set is composed of two λ-structures $\{\mathcal{P}\}$ and $\{\mathcal{Q}\}$ such that
$$f_k(\{\mathcal{P}\}) = \{\mathcal{Q}\}$$
- In other words, we obtain 2^s (s-bit S-boxes) slid pairs from each such set.
- This increases the signal that can be used for detecting slid sets!

Slid Sets for Attacking 2K-AES

- Take λ-set of plaintexts $\{\mathcal{P}\}_i$ (e.g., saturate the input of S-box 0).
- Ask for their encryption to obtain $\{\mathcal{C}\}_i$.
- Construct the sets $\{\mathcal{Q}\}_j$ (such that $S^{-1}(A^{-1}(\{\mathcal{Q}\}_j))$ is a λ-set).
- Ask for their encryption to obtain $\{\mathcal{D}\}_j$.
- Try to match the slid set $(\{\mathcal{C}\}_i, \{\mathcal{D}\}_j)$.

Matching the Slid Sets

- Apply $A(S(\{\mathcal{C}\}_i))$ to obtain $\{\tilde{\mathcal{C}}\}_i$.
- "Swap" the order of K and A in $\{\mathcal{D}\}_j$.
- For a slid set

$$A^{-1}\{\mathcal{D}\}_j = S(\{\tilde{\mathcal{C}}\}_i \oplus k)) \oplus A(k).$$

- This actually "breaks" the last two rounds into several independent S-boxes.

Matching the Slid Sets

- Apply $A(S(\{\mathcal{C}\}_i))$ to obtain $\{\tilde{\mathcal{C}}\}_i$.
- "Swap" the order of K and A in $\{\mathcal{D}\}_j$.
- For a slid set

$$A^{-1}\{\mathcal{D}\}_j = S(\{\tilde{\mathcal{C}}\}_i \oplus k)) \oplus A(k).$$

- This actually "breaks" the last two rounds into several independent S-boxes.
- We just need to link the sets without guessing the key k.
- Luckily, we can count multiplicities of different values in each S-box [DKS10].

Hypercube of Slid Pairs

- Consider a slid pair (P, Q).
- Change the input of P to some S-box (e.g., 0).
- The change in the value after one round is inside an affine space of size 2^s.
- So, from a slid pair (P, Q), we can "generate" a second pair $(P_i \oplus a, Q_j \oplus A(a'))$.†

Hypercube of Slid Pairs

- Consider a slid pair (P, Q).
- Change the input of P to some S-box (e.g., 0).
- The change in the value after one round is inside an affine space of size 2^s.
- So, from a slid pair (P, Q), we can "generate" a second pair $(P_i \oplus a, Q_j \oplus A(a'))$.[†]

Hypercube of Slid Pairs

- Consider a slid pair (P, Q).
- Change the input of P to some S-box (e.g., 0).
- The change in the value after one round is inside an affine space of size 2^s.
- So, from a slid pair (P, Q), we can "generate" a second pair $(P_i \oplus a, Q_j \oplus A(a'))$.†

But wait!

There is more!

Hypercube of Slid Pairs (cont.)

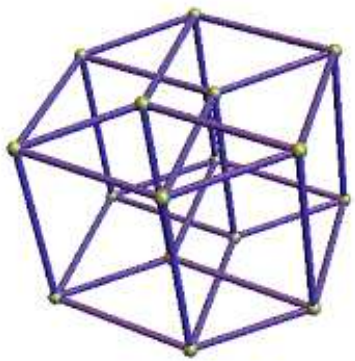
- Assume that (P, Q) is a slid a pair.
- Also assume that $(P \oplus a, Q \oplus A(a'))$ is a slid pair,

Hypercube of Slid Pairs (cont.)

- Assume that (P, Q) is a slid a pair.
- Also assume that $(P \oplus a, Q \oplus A(a'))$ is a slid pair, and that $(P \oplus b, Q \oplus A(b'))$ is a slid pair, where a and b each "activates" a different S-box.

Hypercube of Slid Pairs (cont.)

- Assume that (P, Q) is a slid a pair.
- Also assume that $(P \oplus a, Q \oplus A(a'))$ is a slid pair, and that $(P \oplus b, Q \oplus A(b'))$ is a slid pair, where a and b each "activates" a different S-box.
- Then also $(P \oplus a \oplus b, Q \oplus A(a') \oplus A(b'))$ is a slid pair.
- Of course, if there are more S-boxes, one can take the base slid pair, "generate" some related slid-pairs, and then combine all of them to form an *hypercube* of slid pairs.



Attacking 1K-AES with Secret S-boxes

- We can use the hypercube of slid pairs to attack 1K-AES when the S-box is unknown.
- For AES' parameters ($n = 128$, $s = 8$):
  - The attack is based on finding hypercubes of slid pairs of dimension 5.
  - Each such hypercube has a probability of $(2^{-8})^5 = 2^{-40}$ to indeed offer 32 slid pairs.
  - We identify whether a hypercube is correct by observing consistency in the ciphertexts.
  - Once a hypercube is detected, it also offers input/output pairs to the $ARK \circ SB$ layer.
  - 45 such hypercubes are needed to fully recover the S-box.

Suggestive Plaintext Structures

- One problem many slide attacks face is the cycle: a slid pair is found, when the key it suggests is correct.
- In many cases that means we need to try all possible pairs to find the slid pair.
- Many variants (including the above two) bypass the problem by finding a per-plaintext property (rather than per-pair one).
- Suggestive plaintext structures approach the problem differently.

Suggestive Plaintext Structures (cont.)

- The main idea is that we associate with each plaintext P that we test, a (partial) key candidate.
- Thus, when iterating over the plaintexts, we obtain (partial) key suggestions, which can be used to determine the slid counterpart.
- This implies a simple attack on 1K-AES with success rate of 1 given $2 \cdot 2^{n/2}$ chosen plaintexts:
  - Pick $2^{n/2}$ plaintexts P_i such that their lower half is 0.
  - Pick $2^{n/2}$ plaintexts Q_j such that the upper half of $S^{-1}(A^{-1}(Q_j))$ is 0.
- We are assured that there is a slid pair (P_i, Q_j).

Suggestive Plaintext Structures (cont.)

- The main idea is that we associate with each plaintext P that we test, a (partial) key candidate.
- Thus, when iterating over the plaintexts, we obtain (partial) key suggestions, which can be used to determine the slid counterpart.
- This implies a simple attack on 1K-AES with success rate of 1 given $2 \cdot 2^{n/2}$ chosen plaintexts:
  - Pick $2^{n/2}$ plaintexts P_i such that their lower half is 0.
  - Pick $2^{n/2}$ plaintexts Q_j such that the upper half of $S^{-1}(A^{-1}(Q_j))$ is 0.
- We are assured that there is a slid pair (P_i, Q_j).
- Moreover, the upper half of the key is equivalent to the upper half of P_i!

Suggestive Plaintext Structures (cont.)

- The main idea is that we associate with each plaintext P that we test, a (partial) key candidate.
- Thus, when iterating over the plaintexts, we obtain (partial) key suggestions, which can be used to determine the slid counterpart.
- This implies a simple attack on 1K-AES with success rate of 1 given $2 \cdot 2^{n/2}$ chosen plaintexts:
  - Pick $2^{n/2}$ plaintexts P_i such that their lower half is 0.
  - Pick $2^{n/2}$ plaintexts Q_j such that the upper half of $S^{-1}(A^{-1}(Q_j))$ is 0.
- We are assured that there is a slid pair (P_i, Q_j).
- Moreover, the upper half of the key is equivalent to the upper half of P_i!
- The 1K-AES attack is similar to the one of [B+18] with the addition of "splice and cut".

Attacking 1K-AES with Incomplete Diffusion in the Last Round

- Consider 1K-AES where the last round lacks MixColumns.
- Pick two Q_j's structures (one fixed to 0, and one fixed to 1).
- Each P_i suggests:
  - Upper half of the key,
  - A friend $R_i = P_i \oplus (0, 0, 0, 1)$ which also has a slid pair (in the second ciphertext structure). Denote its corresponding ciphertext by F_i.
- If P_i is the correct plaintext, we can partially decrypt C_i and F_i to obtain the difference of the upper half of the ciphertexts from the two Q_j structures.

Substitution Slide Attack

- Can be used to attack 1K-AES with a completely different last round diffusion, i.e., A'.
- Moreover, the resulting attack requires $2^{n/2}$ known plaintexts!
- So we need to identify the slid pair, without trying **all** pairs.

Substitution Slide Attack (Cont.)

- Consider the equations for a slid pair (P_i, P_j):

$$P_j = A(S(K(P_i))) \Rightarrow P_j = A(S(P_i \oplus k)) \Rightarrow$$
$$k = P_i \oplus S^{-1}(A^{-1}(P_j))$$
$$C_j = K(A'(S(K(A(A'^{-1}(K(C_i))))))) \Rightarrow$$
$$S^{-1}(A'-1(C_j \oplus K)) = K(A(A'^{-1}(K(C_i))))$$

- This allows through a series of substitutions to obtain

$$S^{-1}(A'^{-1}(K(C_j))) \oplus A(A'^{-1}(S^{-1}(A^{-1}(P_j))) \oplus S^{-1}(A^{-1}(P_j)) =$$
$$A(A'^{-1}((P_i)) \oplus P_i \oplus A(A'^{-1}(C_i)).$$

Substitution Slide Attack (Cont.)

- The attack is thus composed of the following steps:
  - Evaluate $A(A'^{-1}((P_i)) \oplus P_i \oplus A(A'^{-1}(C_i))$ for all plaintexts.
  - Guess $n/4$ bits of the key:
    - For all P_i's:
    - Among the $2^{n/4}$ candidate P_j's, check the substituted condition.
    - Of course, this is done efficiently using hash tables...
  - Once a suggestion is made, test the proposed $k = P_i \oplus S^{-1}(A^{-1}(P_j))$

Summary

- Introduced 4 new slide techniques:
  - Slid Sets
  - Hypercube of slid pairs
  - Suggestive plaintext structures
  - Substitution slide
- While these techniques are useful for SPNs, they can be widely used for other schemes.

Results

Cipher	Technique	Complexity (general)		AES-like	
		Data/Memory	Time	Data/Memory	Time
Known S-Boxes					
1-KSAf	Slide [B+18]	$2^{n/2}$ (KP)	$2^{n/2}$	2^{64} (KP)	2^{64}
1-KSAt	Suggestive str.	$3 \cdot 2^{n/2}$ (CP)	$4 \cdot 2^{n/2}$	$2^{65.6}$ (CP)	2^{66}
1-KSAt	Sub. slide	$2^{n/2}$ (KP)	$2^{3n/4}$	2^{64} (KP)	2^{96}
2-KSAf	Slid sets	$2^{(n+s)/2+1}$ (CP)	$2^{(n+s)/2+1}$	2^{69} (CP)	2^{69}
2-KSAf	Slide + Key Guessing	$(n/s)2^{n/2}$ (CP)	$2^{n/2+s}$	2^{68} (CP)	2^{72}
2-KSAf	Slide + Pt/Ct Coll.	See full version for details		$2^{82‡}$ (CP)	2^{82}
2-KSAtpi †	Slid sets	$2^{(n+m)/2+1}$ (CP)	$\max\{2^{(n+m)/2+1}, 2^{2m}\}$	2^{78} (CP)	2^{78}
3-KSAfi †	Slid sets	$2^{(n+m)/2+1}$ (CP)	$\max\{2^{(n+m)/2+1}, 2^{2m}\}$	2^{81} (CP)	2^{81}
Secret S-Boxes					
1-KSAf	Slid sets	$1.17\sqrt{s}2^{(n+s)/2}$ (CP)	$1.17\sqrt{s}2^{(n+s)/2}$	$2^{70.3}$ (CP)	$2^{70.3}$
1-KSAf	Hypercube	$\sqrt{s}2^{n/2+s(s+3)/4+1}$ (CP)	$\sqrt{s}2^{n/2+s(s+3)/4+1}$	2^{88} (CP)	2^{88}

KP – Known Plaintext; CP – Chosen Plaintext; For AES-like $n = 128, s = 8$

† – this version has incomplete diffusion layer, m denotes the "word" size of the linear operation.

‡ – memory complexity of this attack is 2^{47}.

Thank you for your Attention!

Full version:

`https://eprint.iacr.org/2019/509`