Related-Key Attacks

Orr Dunkelman

Department of Computer Science, University of Haifa
Faculty of Mathematics and Computer Science
Weizmann Institute of Science

June 2nd, 2011

Outline

The Related-Key Model

- Introduced by Biham and independently by Knudsen in 1993 [B93,K93].
- A block cipher is a keyed permutation, i.e., $E : \{0,1\}^n \times \{0,1\}^k \to \{0,1\}^n$ (or $E_k : \{0,1\}^n \to \{0,1\}^n$).
- Regular cryptanalytic attacks attack E by controlling the input/output of $E_k(\cdot)$.
- In related-key attacks the adversary can ask to control k (chosen key attacks).
- This make look like a very strong notion, but the model allows for the adversary to control only the relation between keys.

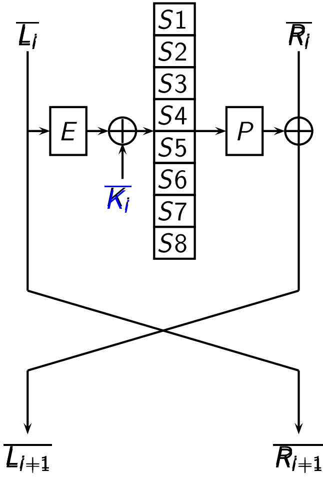
The Related-Key Model (cont.)

- In standard attacks, the adversary can query an oracle for E_k.
- In related-key attacks, the adversary can query the oracles $E_{k_1}, E_{k_2}, \ldots$
- The adversary is either aware of the relation between the keys or **can choose** the relation.
- This model which may look strong is actually not so far fetched:
  - Real life protocols allow for that.
  - When the block cipher is used as a compression function — the adversary may control actually control the key.
  - In some cases, there are properties so "strong", that it is sufficient to have access to encryption under one key.

DES's Complementation Property

- If the key is bitwise complemented, so are all the subkeys.
  $K \to K_1, K_2, \ldots, K_{16}$ **and**
  $\overline{K} \to \overline{K_1}, \overline{K_2}, \ldots, \overline{K_{16}}$
- If the input to the round function is also bitwise complemented, the complementation is canceled.
- In other words, the input to the S-boxes is the same. **And the output of the S-boxes (and the round).**
- **DES's complementation property**:

$$DES_K(P) = \overline{DES_{\overline{K}}(\overline{P})}$$



Using the Complementation Property

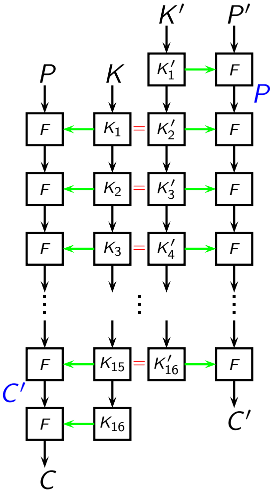
- Using the complementation property it is possible to speed up exhaustive key search of DES by a factor of 2.
- The adversary asks for the encryption of P and $\overline{P}$.
- Let $C_1 = E_K(P)$ and $C_2 = E_K(\overline{P})$, where K is the unknown key.
- For each possible key k whose most significant bit is 0:
  1. Check whether $DES_k(P) = C_1$ (if yes, k is the key).
  2. Check whether $\overline{DES_k(P)} = C_2$ (if yes, $\overline{k}$ is the key).

Note that $\overline{DES_k(P)} = C_2 \Rightarrow \overline{(C_2)} = DES_k(P)$.

As $C_2 = DES_K(\overline{P})$, then $\overline{DES_K(\overline{P})} = DES_k(P)$, i.e., $K = \overline{k}$.

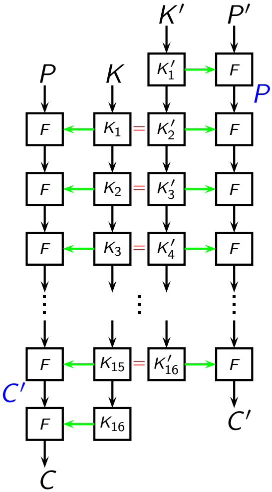
A Related-Key Attack on a Slightly Modified DES

- Assume that all the rotations in the key schedule are all by 2 bits to the left.
- Consider two keys K and K', such that the subkeys produced by the key schedule algorithm satisfy $K_i = K'_{i+1}$ (i.e., $K_1 = K'_2, K_2 = K'_3, \ldots$).
- Then the first 15 rounds of encryption under K are just like the last 15 rounds of encryption under K'.



A Related-Key Attack on a Slightly Modified DES

- Let $P = F_{K'_1}(P')$.
- Due to the equality between the functions, P and P' share 15 rounds of the encryption.
- Thus, $C = F_{K_{16}}(C')$.
- Given (P, C) and (P', C'), deducing K'_1 and K_{16} (given DES's round function) is easy.



A Related-Key Attack on a Slightly Modified DES

- Ask for the encryption of 2^{16} plaintexts $P'_i = (A, x'_i)$ under K'. Let $C'_i = E_{K'}(P'_i)$.
- Ask for the encryption of 2^{16} plaintexts $P_i = (y'_j, A)$ under K. Let $C_j = E_K(P_j)$.

1. By birthday arguments there is a pair of values P'_i which is encrypted under one round to P_j. From this point forward, they are "evolving" together, and thus, $C_j = F_{K_{16}}(C'_i)$.
2. From Feistel properties, that means that the left half of C'_i is equal to the right half of C_j.

A Related-Key Attack on a Slightly Modified DES

- Search for a pair of ciphertexts C_i' and C_j such that the left half of C_i' is equal to the right half of C_j.
- Deduce that $P_j = F_{K_1'}(P_i')$ and that $C_j = F_{K_{16}}(C_i')$, and retrieve the key.
- This pair is called *a related-key plaintext pair*.
- Using this pair it is easy to deduce K_1' and K_{16} (which are also share bits between themselves).

Data complexity: 2^{16} CPs under two related-keys (the relation was chosen by the adversary).
Time complexity: 2^{17} encryptions (the analysis phase is very efficient).

A Second Attack on a Slightly Modified DES

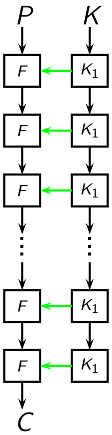
- For this modification of DES, it is possible to offer an attack which has access to only one key.
- The attack is an extension of the complementation property:

  Each key K has 5 other keys which induce a related-encryption process.

- Hence, using 2^{34} chosen plaintexts encrypted under **one**, we can analyze 6 keys(!) using a trial encryption.

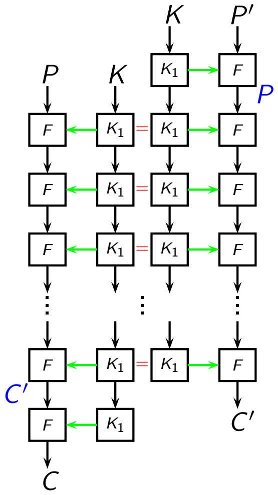
The Slide Attack

- Presented by Biryukov and Wagner in 1999.
- Can be applied to ciphers with the same keyed permutation.
- Independent of the number of rounds of the cipher.
- To some extent, this attack is a related-key plaintext attack when the key is its own related-key.



An Example — Slide Attack on 2K-DES

- Consider a variant of DES with $2r$ rounds, where the subkeys are $(K_1, K_2, K_1, K_2, \ldots, K_1, K_2)$.
- This variant has 96-bit key, and if r is large enough, no conventional attacks apply.



A Related-Key Attack on a 2K-DES (cont.)

- Take 2^{32} known plaintexts, P_i (and their corresponding ciphertexts C_i).
- Let $f_{K_1,K_2}(\cdot)$ be two rounds of DES with the subkeys K_1 and K_2.
- Then, the data set is expected to contain two plaintexts P_i and P_j such that $f_{K_1,K_2}(P_i) = P_j$ and $f_{K_1,K_2}(C_i) = C_j$ (denoted as a *slid pair*).

How do you Find the Slid Pair?

- Generally speaking, the best way to find the slid pairs is to try all of them.
- So in this attack, the adversary considers each pair (P_i, P_j) (there are 2^{64} pairs, as the pair is ordered).
- For each pair, the adversary has two equations to solve:

$$f_{K_1,K_2}(P_i) = P_j; \qquad f_{K_1,K_2}(C_i) = C_j$$

- This can be done very easily.
- For each solution (if exists), verify the suggested key.
- Time complexity — 2^{64} times solving the above set.
- A possible improvement: Guess some part of K_1 (or K_2) which gives filtering on the pairs, and then there are less pairs to analyze.

How do you Find the Slid Pair? (cont.)

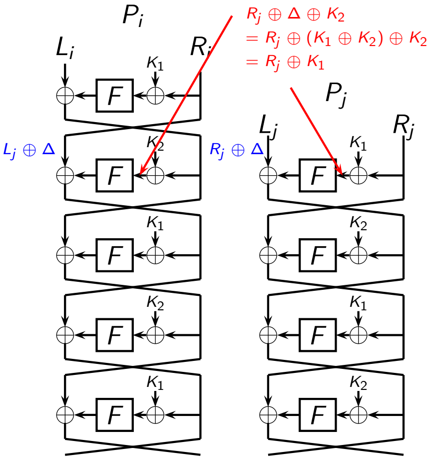
- This leads to a very interesting approach in block ciphers cryptanalysis.
- To break a cipher X (to find the secret key), we need a slid pair.
- To find this slid pair, we take many candidate pairs.
- For each candidate pair, we analyze which key it suggests.
- Then, if the key suggested is correct we found the slid pair. . . . which is what we need for finding the right key.

Summary of the Slide Attack

- Independent of the number of rounds.
- Generation of a slid pair in $O(2^{n/2})$ known plaintexts (or $2^{n/4}$ for Feistel block ciphers).
- Works if $F_K(P_i) = P_j, F_K(C_i) = C_j$ is sufficient for finding K.

Complementation Slide Attack

- Consider 2K-DES.
- Let $\Delta = K_1 \oplus K_2$.
- Consider two plaintexts P_i, P_j such that if $X = f_{K_1}(P_i)$ then $X_i = P_j \oplus (\Delta, \Delta)$.
- This relation remains until $C_j = f_{K_2}(C_i) \oplus (\Delta, \Delta)$.

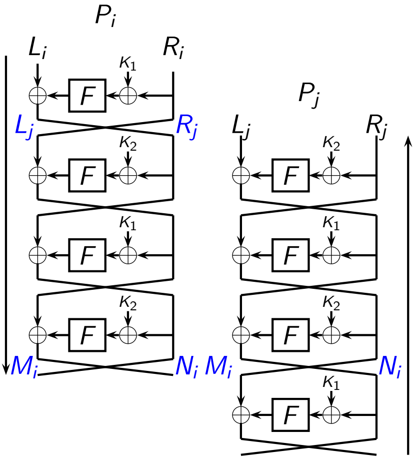


Complementation Slide Attack

- As half of the data is unchanged by $f(\cdot)$, the identification of slid pairs is easier.
- Starting with 2^{32} known plaintexts, and use the filter condition on the differences (right half of P_i XOR the left half of P_j is equal to the right half of C_i XOR the left half of C_j) to discard most of the wrong candidate keys.
- There is a small technicality here that makes the attack fail. If you recall, the difference in the data words is of 32 bits, and of the subkey is in 48-bit words.
- Hence, this attack works, only if Δ is a legitimate output of $E(\cdot)$ of DES (i.e., the actual difference in the plaintext is $E^{-1}(\Delta)$).

Slide Attack with a Twist

- Consider encryption and decryption in a Feistel block cipher.
- They are the same up to the order of subkeys.
- Now, consider 2K-DES, with one round slide in the encryption direction and the decryption direction...
- Given 2^{32} known plaintexts, it is possible to find a twisted slid pair and repeat the analysis.



Slide Attack with a Twist (cont.)

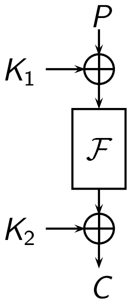
- This time, it is possible to analyze only one subkey (K_1), as the relations are

$$f_{K_1}(N_i) = C_j \oplus M_i; \qquad f_{K_1}(R_i) = R_j \oplus L_i.$$

- This allows applying a chosen plaintext and ciphertext attacks with 2^{16} of each.
- The adversary asks for the encryption of (A, x) and the decryption of (A, y).
- Note that this variant *actually works*.
- And do note that you can combine the two techniques.

The Even-Mansour Block Cipher

- Suggested by Even and Mansour in 1991, as a generalization of the DESX approach.
- Apparently, even if you know the internal key of DESX, the system is still secure.
- Main idea: Change the keyed permutation in the middle to an n-bit pseudo-random permutation $\mathcal{F}$.
- Block size: n bits, Key size: 2n bits.



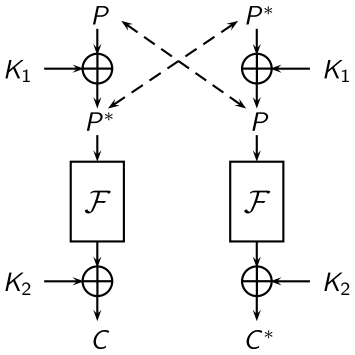
$$EM^{\mathcal{F}}_{K_1,K_2}(P) = \mathcal{F}(P \oplus K_1) \oplus K_2$$

Security of the Even-Mansour Scheme

- A simple attack that requires 2 plaintext/ciphertext pairs and 2^n time (so security is n-bits at most).
- There is a **proof** that any attack that uses D plaintext/ciphertext pairs, and T queries to $\mathcal{F}$, has success rate of $O(DT/2^n)$.
- There is a differential attack that offers this tradeoff [D92].
- There is also a slide with a twist attack that uses $2^{n/2}$ data and time.

Slide with a Twist Attack on Even-Mansour

- Consider two plaintexts P and P^* such that $P^* = P \oplus K_1$.
- The inputs to $\mathcal{F}$ are swapped, which means that so does the outputs.
- Hence, $C \oplus C^* = \mathcal{F}(P) \oplus \mathcal{F}(P^*)$.
- So the attack starts with $2^{n/2}$ plaintexts P_i, each is encrypted to the corresponding C_i, and a collision in the values of $C_i \oplus \mathcal{F}(P_i)$ is expected to suggest a slid pair.



Slide with a Twist Attack on Even-Mansour

- The attack requires $D = 2^{n/2}$ known plaintexts.
- To generate the table, $T = 2^{n/2}$ additional queries to $\mathcal{F}$ are made.
- The success rate is the probability of having a slid pair, which is quite high.
- We note that having even slightly less than $O(2^{n/2})$ plaintexts results in the failure of the attack.
- So this attack satisfies the bound, but at the same time, offers no tradeoff.

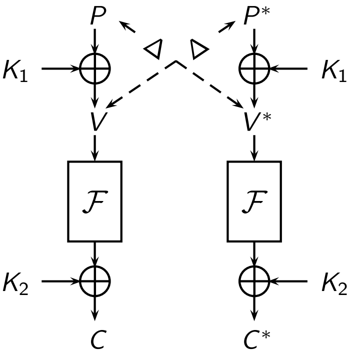
Motivation

- The slide attack requires one slid pair to work.
- To find such a pair, we need at least $2^{n/2}$ known plaintexts.
- If we are given less data, can we somehow compensate for the lack of slid pairs with some computation?

SlideX Attack on Even-Mansour

- Consider two plaintexts P and P^* such that $P^* = P \oplus K_1 \oplus \Delta$.
- Then:

$$EM^{\mathcal{F}}_{K_1,K_2}(P) = \mathcal{F}(P \oplus K_1) \oplus K_2$$
$$= \mathcal{F}(P^* \oplus \Delta) \oplus K_2$$
$$EM^{\mathcal{F}}_{K_1,K_2}(P^*) = \mathcal{F}(P^* \oplus K_1) \oplus K_2$$
$$= \mathcal{F}(P \oplus \Delta) \oplus K_2$$



- Hence,

$$EM^{\mathcal{F}}_{K_1,K_2}(P) \oplus \mathcal{F}(P \oplus \Delta) = EM^{\mathcal{F}}_{K_1,K_2}(P^*) \oplus \mathcal{F}(P^* \oplus \Delta)$$

SlideX Attack on Even-Mansour (cont.)

- We define a SlideX pair, as a pair which actually satisfies the required relation $P = P^* \oplus K_1 \oplus \Delta$.
- To check for the SlideX pair, we take the D plaintext/ciphertext pairs (P_i, C_i), and for each Δ guess, we construct a table of all values $C_i \oplus \mathcal{F}(P_i \oplus \Delta)$.
- The trick here, is that we check $O(D^2)$ pairs by each such guess of Δ.
- Hence, we repeat the construction of the table $O(2^n/D^2)$ times, each time with D calls to $\mathcal{F}$, or $T = O(2^n/D)$ times in total.

And we're done!

SlideX vs. Slide (with a Twist)

- The attack can work with any given amount of data.
- As a SlideX pair is actually a SlideX tuple (with respect to some Δ), we can increase the number of Δ's to compensate for the reduced data.
- Additionally, we just need to store $O(D)$ values, so if $D \ll 2^{n/2}$, we can use a significantly smaller amount of memory.

Related-Key Differential Attacks

- Consider the complementation property of DES:

$$DES_K(P) = \overline{DES_{\overline{K}}(\overline{P})}$$

- This equality can be rewritten as:

$$DES_K(P) \oplus DES_{\overline{K}}(\overline{P}) = FFFF\ FFFF\ \ FFFF\ FFFF_x$$

- Does this looks familiar?
- This motivated Kelsey, Schneier and Wagner to introduce related-key differentials.

Related-Key Differentials (cont.)

- The probability of regular differential is:

$$Pr_{P,K}[E_K(P) \oplus E_K(P \oplus \Delta P) = \Delta C]$$

- The probability of related-key differential is:

$$Pr_{P,K}[E_K(P) \oplus E_{K \oplus \Delta K}(P \oplus \Delta P) = \Delta C]$$

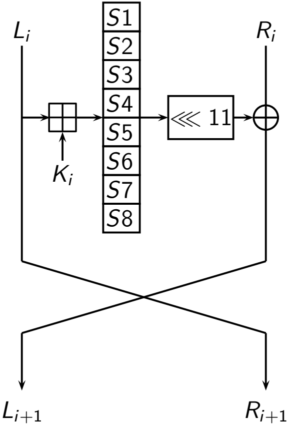
- The key difference leads to subkey differences, that may be used to cancel the differences in the input to the round function.
- The reminder of the differential attack using a related-key attack is quite the same (up to the use of two keys).
- Usually, the key relation is by a difference, but other relations may be used as well.*

*Note that the relation $K' = K \wedge Const$ and $K' = K \vee Const$, for any constant $Const$, allow for a trivial key recovery attack.

The Block Cipher GOST

- The Soviet/Russian block cipher standard (GOST 28147-89).
- 64-bit block, 256-bit key, 32 rounds.
- S-boxes: 4×4. Implementation specific.
- Key schedule very simple, take $K = (K_1, K_2, \ldots, K_8)$:

Round	1	2	3	4	5	6	7	8
Subkey	K_1	K_2	K_3	K_4	K_5	K_6	K_7	K_8
Round	9	10	11	12	13	14	15	16
Subkey	K_1	K_2	K_3	K_4	K_5	K_6	K_7	K_8
Round	17	18	19	20	21	22	23	24
Subkey	K_1	K_2	K_3	K_4	K_5	K_6	K_7	K_8
Round	25	26	27	28	29	30	31	32
Subkey	K_8	K_7	K_6	K_5	K_4	K_3	K_2	K_1



Related-Key Differentials in GOST

- Flipping the MSBs of all key words, flips the MSB of all the subkeys.
- Flipping the two MSBs of the plaintext words, leads to the same input entering the S-boxes in all rounds.
- Thus, under a key difference $(80000000_x, 80000000_x, \ldots, 80000000_x)$ the plaintext difference $(80000000_x, 80000000_x)$ leads to ciphertext difference $(80000000_x, 80000000_x)$ with probability 1.
- Can speed up exhaustive search by a factor of 2 (like in DES).
- Or for a very simple distinguishing attack (with 2 chosen plaintexts).

Recovering the Key in GOST in a Related-Key Attack

- For a differential key recovery attack we need a differential with nontrivial probability.
- Pick $\Delta K = (40000000_x, 40000000_x, \ldots, 40000000_x)$.
- An input difference $\Delta = (40000000_x, 40000000_x)$ remains unchanged after one round with probability $1/2$.
- Thus, it is easy to build a 30-round related-key differential with probability 2^{-30} for GOST.
- Then, GOST can be attacked using standard differential techniques.

The Differences from Regular Differentials

- Despite the above there are few subtle differences between regular differentials and related-key differentials.
- The amount of possible pairs, for example. In a one-key scenario, for a given input difference there are 2^{n-1} possible distinct pairs (n being the block size). In two-key scenario — 2^n.
- Consider an input difference to an s-bit round function. Once the key is fixed, for any given input difference, there are at most 2^{s-1} output differences. In the related-key model there are 2^s (if there is a key difference, of course).

Certificational Attacks on AES

- Recently, in a series of papers, several certificational attacks on the full AES-192 and AES-256 were proposed:
  1. In [BKN09] the first attack on the full AES-256 is reported:
     - 2^{131} data and time in the related-key model (2^{35} related keys).
     - Several attacks on AES-256 in Davies-Meyer (a transformation into a compression function).
  2. In [BK09] attacks on AES-192 and AES-256:
     - A 2^{99} data/time attack on AES-256 in the related-subkey model (using 4 related keys).
     - A 2^{176} data/time attack on AES-192 in the related-subkey model.

The Related-Subkey Model

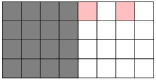
- This new model was recently introduced in [BK09].
- In related-key attacks, a simple relation R is used for the keys K_1, K_2.
- In related-subkey attacks, R is a simple relation between two subkeys, RK_1, RK_2.
- The two subkeys are then handled by the key schedule algorithm to obtain the actual keys.
- This slightly less intuitive approach (and less practical one) can be "covered" by the theoretical treatment by just expanding the set of "good relations".

The Related-Subkey Model (cont.)

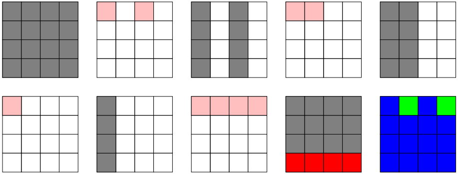
- Despite the fact that this model may seem too strong, it is not.
- There are cases where the required relations can be satisfied:
  - Hash functions built on top of AES-256,
  - Protocols which allow such related-subkey tampering,
  - and when the key schedule algorithm is not too strong, an adversary may use more keys in the related-key model.
- In any case, in the theoretical settings, a block cipher should not show this type of weakness (ideal cipher model).

An Interesting Property of the Key Schedule Algorithm of AES-256

The key difference

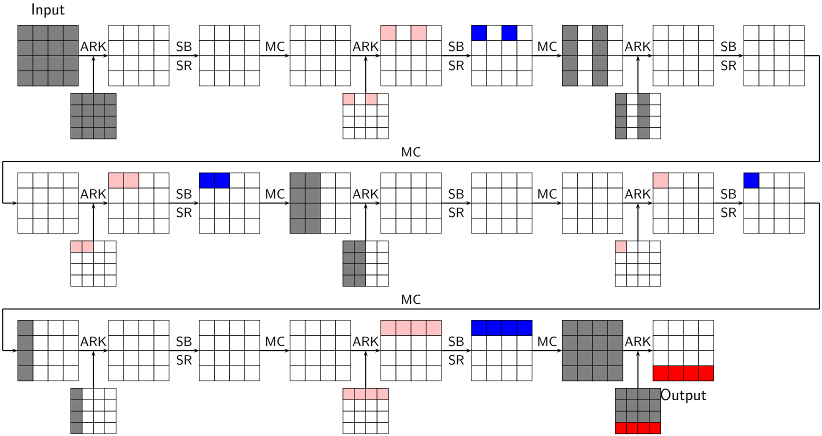


leads to the 10 subkey differences



With probability 1!

An 8-Round Related-Key Differential of AES-256



The probability is 2^{-56}. It can be transformed into a truncated one predicting 24 bits of difference with probability 2^{-36}.

A 10-Round Related-Subkey Differential

- In the related-subkey model, it is possible to pick two keys which satisfy the difference in a slightly different manner.
- The related-subkey allows for shifting the differential by one round.
- This allows an extension of the differential in the backwards direction (despite having a highly active state).
- Which in turn, allows for attacks of practical complexity of up to 10 rounds.

Questions?

Thank you for your Attention!