September 28, 2016 @ASK2016

Nonlinear Invariant Attack

NTT Secure Platform Laboratories and Kobe University
Yosuke Todo

Overview.

- Joint work with Gregor Leander and Yu Sasaki.
- New type of cryptanalyses.
  - This attack works on the weak-key setting.
- Surprising practical extensions.
  - Ciphertext-only message recovery attack!!
- Good applications.
  - Scream, iScream, and Midori64.

Summary of results.

Distinguishing attack under known-plaintext setting.

Target	# of weak keys	Data complexity.	Distinguishing probability.
SCREAM	2^{96}	k	$1-2^{1-k}$
iSCREAM	2^{96}		
Midori64	2^{64}		

The distinguishing attack incidentally recovers 1 bit of secret key.

Message-recovery attack under ciphertext-only setting.

Target	# of weak keys	Maximum # of recovered bits.	Data complexity.	Time complexity.
SCREAM	2^{96}	32 bits	33 ciphertexts	$32^3 = 2^{15}$
iSCREAM	2^{96}	32 bits	33 ciphertexts	$32^3 = 2^{15}$
Midori64-CTR	2^{64}	32h bits	33h ciphertexts	$32^3 h = 2^{15} h$

h is the number of blocks in the mode of operations.

Outline

1. **Nonlinear invariant attack.**
   - **Map of related attacks.**
     - **Linear and nonlinear cryptanalyses.**
     - **Invariant subspace attack.**
   - **Distinguishing attack.**
2. Surprising extension toward practical attack.
   - What's happened if vulnerable ciphers are used in well-known mode of operations?
3. How to find nonlinear invariant.
   - Appropriate nonlinear invariants.
   - How to find nonlinear invariant for KSP round functions.
4. Practical attack on full SCREAM.

Two streams join in new attacks.

Linear attack
[Matsui 1993]

Nonlinear attack
[Harpes et al. 1995]

Invariant subspace attack
[Gregor et al. 2011]

Nonlinear invariant attack [Todo, Gregor, Yu 2016]

NTT

Stream from linear attacks.

Linear attack
[Matsui 1993]

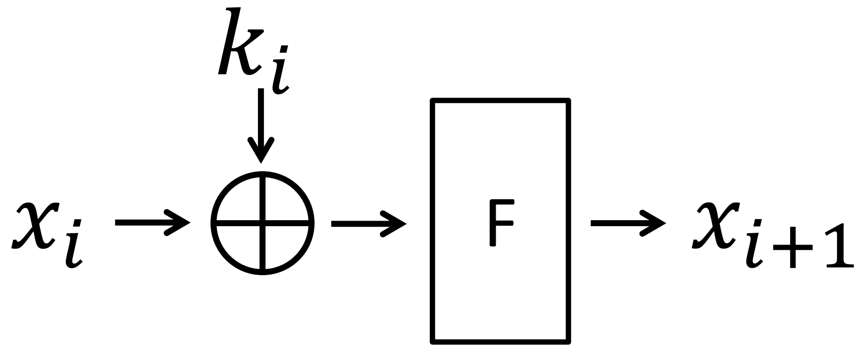
Nonlinear attack
[Harpes et al. 1995]

Invariant subspace attack
[Gregor et al. 2011]

Nonlinear invariant attack [Todo, Gregor, Yu 2016]

Linear attack [Matsui 93].

Key-alternating structure.



$$f_i(x_i) \longrightarrow f_{i+1}(x_{i+1})$$

$f_i(x_i) \oplus f_{i+1}(x_{i+1}) \approx \text{const}$

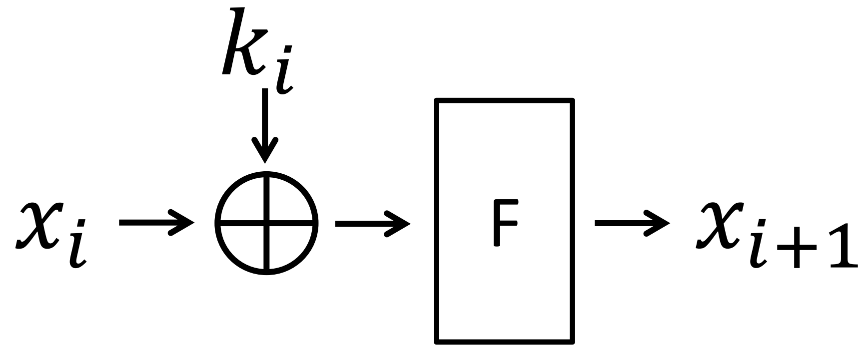
holds with high probability.

next-round

f_i and f_{i+1} are linearly Boolean functions.

Nonlinear attack [Harpes et al.95].

Key-alternating structure.



$$g_i(x_i) \longrightarrow g_{i+1}(x_{i+1})$$

$g_i(x_i) \oplus g_{i+1}(x_{i+1}) \approx \text{const}$

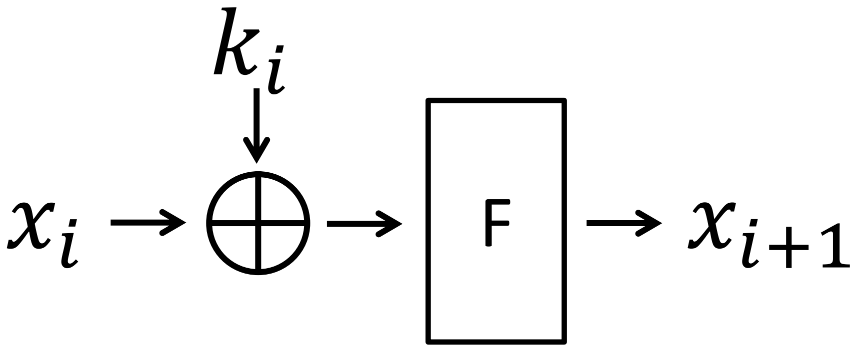
holds with high probability.

next-round

g_i and g_{i+1} are nonlinearly Boolean functions.

Insurmountable problem.

Key-alternating structure.

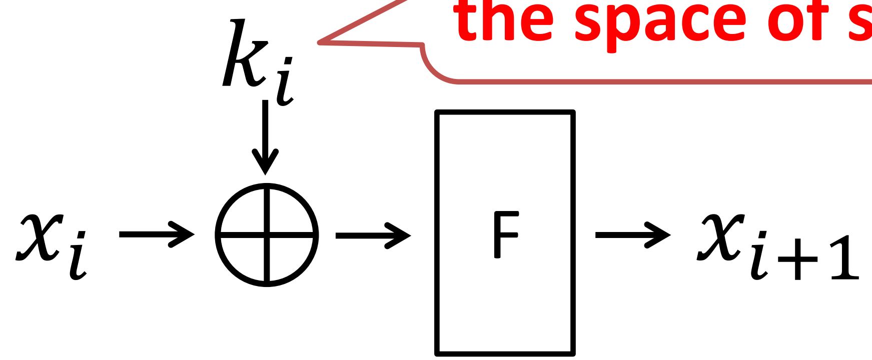


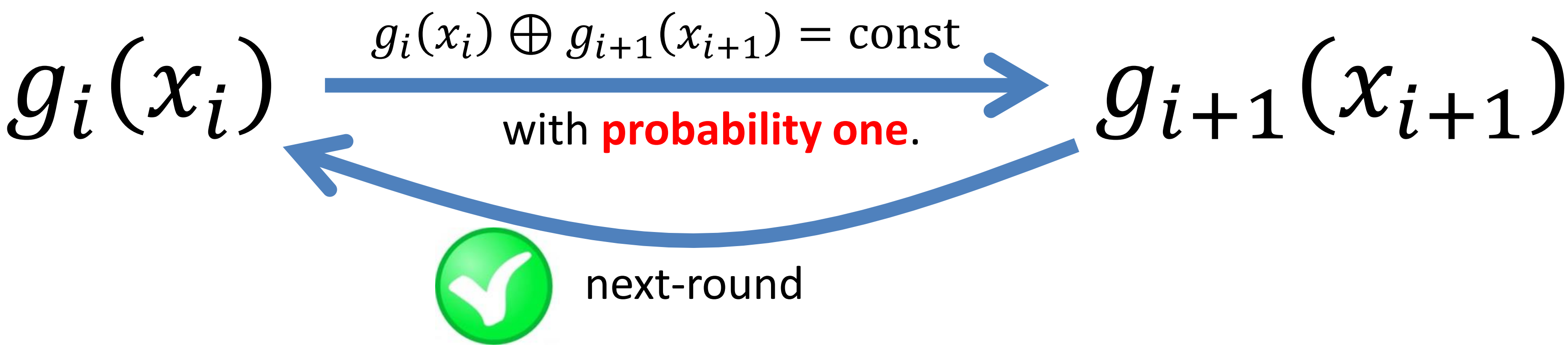
- The actual propagation of nonlinear mask depends on the **specific value** of the state.
- Therefore, we cannot join nonlinear masks for two rounds.

Nonlinear invariant attack.

Key-alternating structure.

Alternatively, we limit the space of secret keys.





g_i and g_{i+1} are nonlinearly Boolean functions.

Stream from invariant subspace attacks.

Linear attack
[Matsui 1993]

Nonlinear attack
[Harpes et al. 1995]

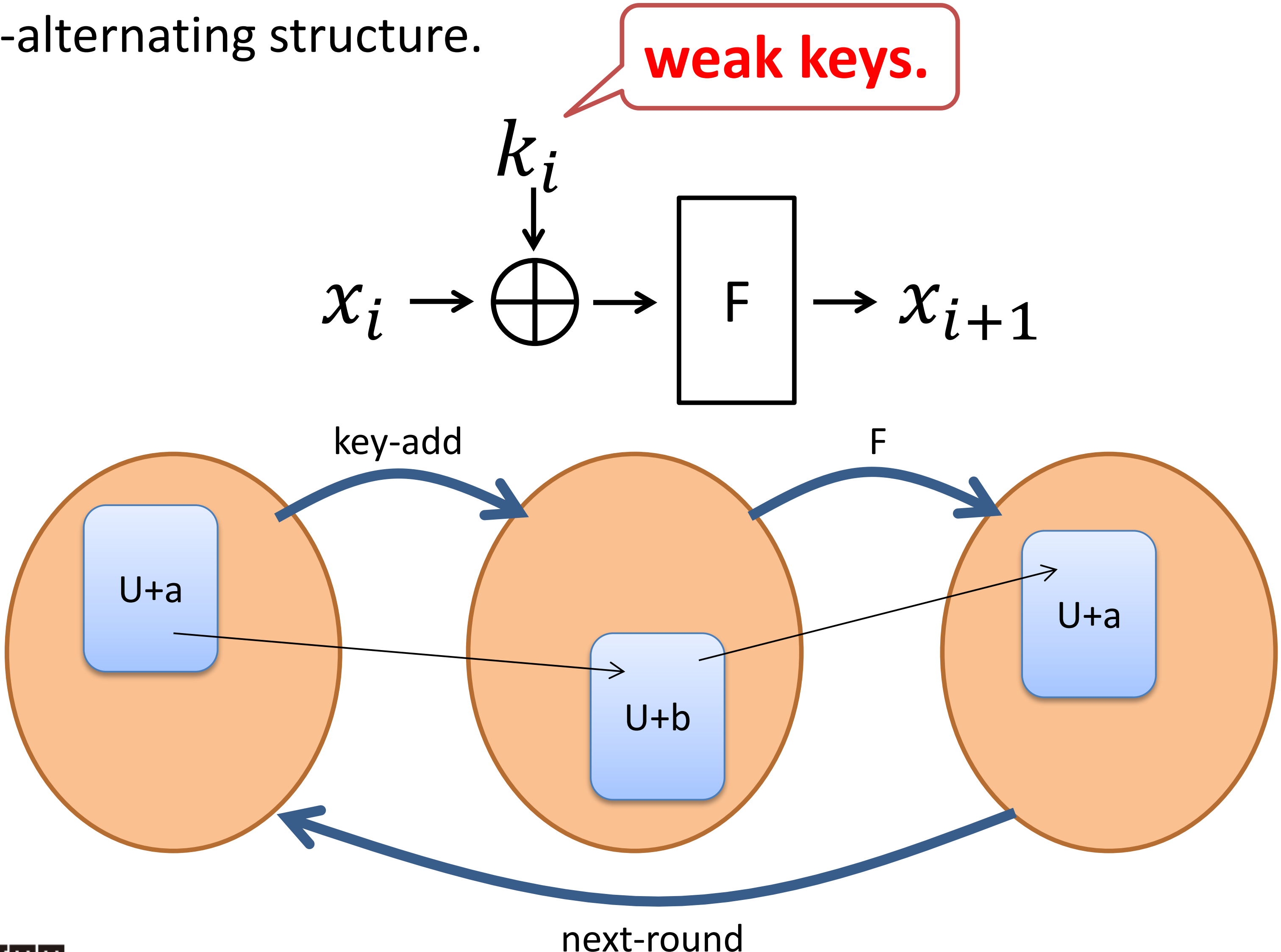
Invariant subspace attack
[Gregor et al. 2011]

Nonlinear invariant attack [Todo, Gregor, Yu 2016]

NTT

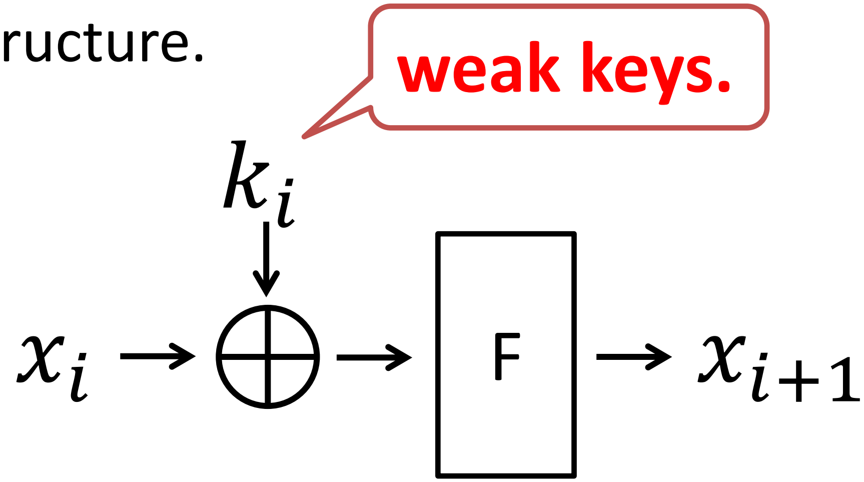
Invariant subspace attacks

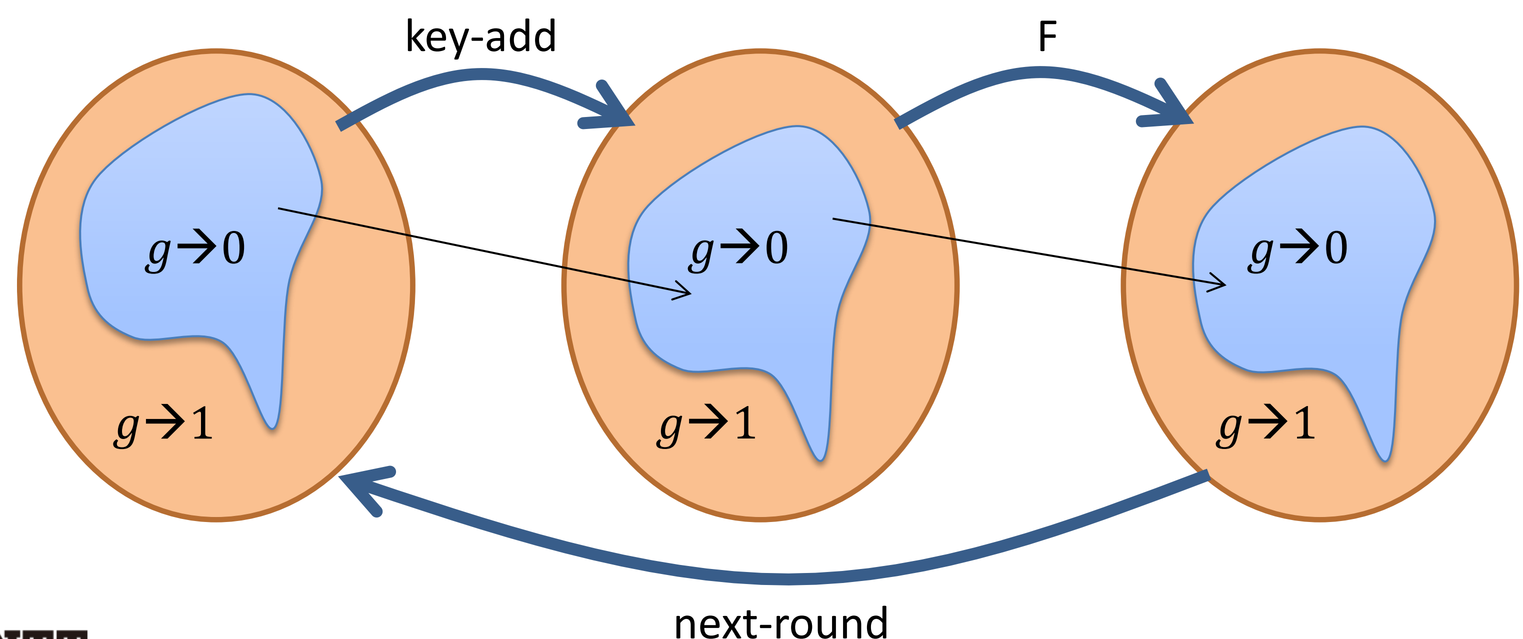
Key-alternating structure.



Nonlinear invarinat attack.

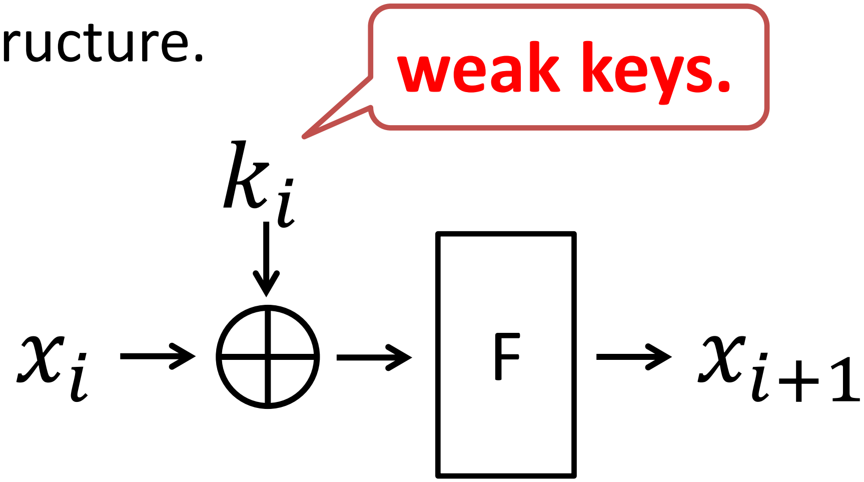
Key-alternating structure.

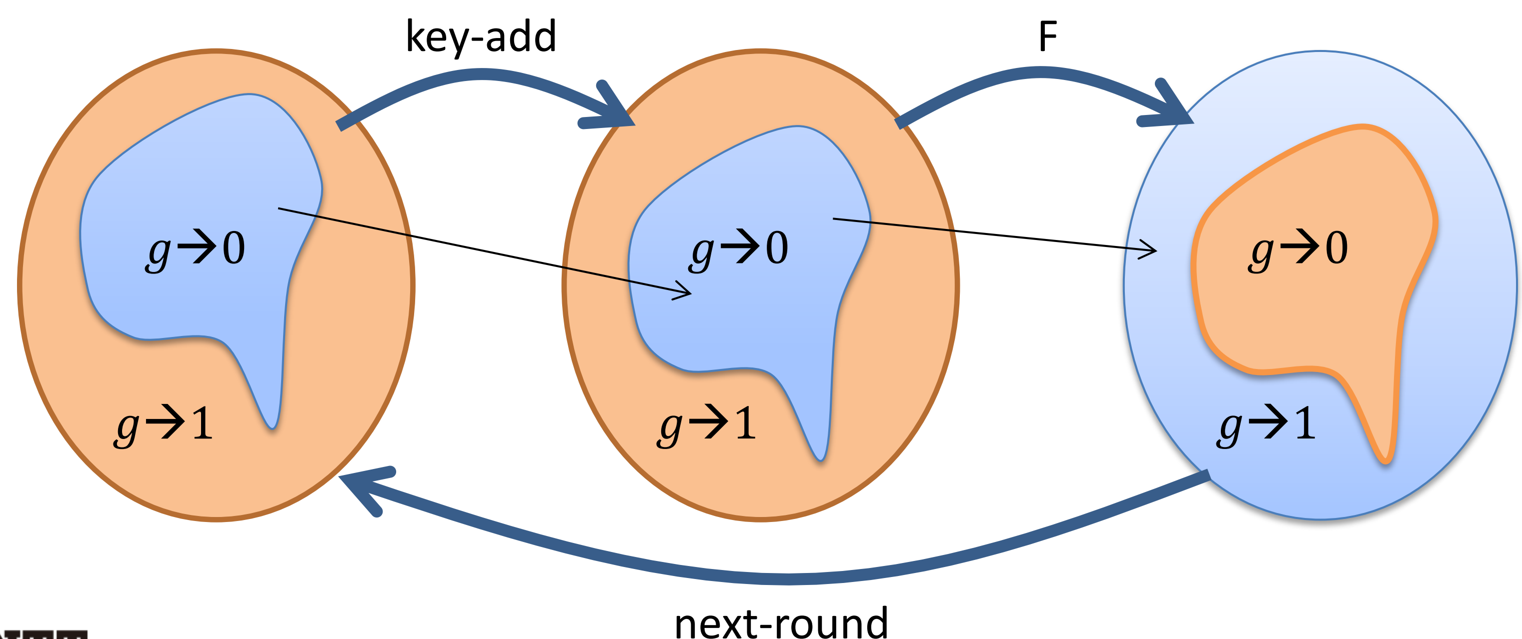




Nonlinear invarinat attack.

Key-alternating structure.





Distinguishing attack.

- If the block cipher has the nonlinear invariant, we can easily distinguish from ideal ciphers.

1. Collect k known plaintexts (p_i, c_i).
2. Compute $g_p(p_i) \oplus g_c(c_i)$ for k pair.
   Then k XORs are always the same.
   The probability that ideal ciphers have this property is 2^{-k+1}.

- At most one bit of information leaks from $g_p(p_i) \oplus g_c(c_i)$.

Outline

1. Nonlinear invariant attack.
   - Map of related attacks.
     - Linear and nonlinear cryptanalyses.
     - Invariant subspace attack.
   - Distinguishing attack.
2. **Surprising extension toward practical attack.**
   - **What's happened if vulnerable ciphers are used in well-known mode of operations?**
3. How to find nonlinear invariant.
   - Appropriate nonlinear invariants.
   - How to find nonlinear invariant for KSP round functions.
4. Practical attack on full SCREAM.

Practical attacks.

strong

Assumption.

Chosen-plaintext attacks (CPA)

- is natural assumption for cryptographers.
- is debatable in practical case.

Known-plaintext attacks (KPA)

- is very weak assumption for cryptographers.
- sometimes holds in practical case.

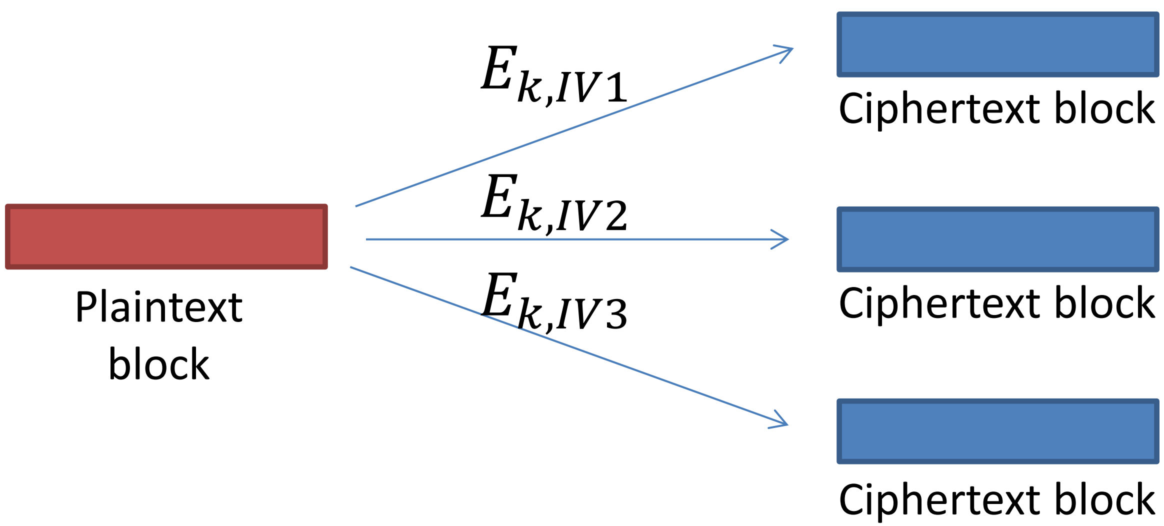
Ciphertext-only attacks (COA)

- is unlikely to happen for cryptographers.
- is information-theoretically impossible w/o assumptions.
- causes non-negligible risks in practical use if possible.

weak

Our attack assumptions.

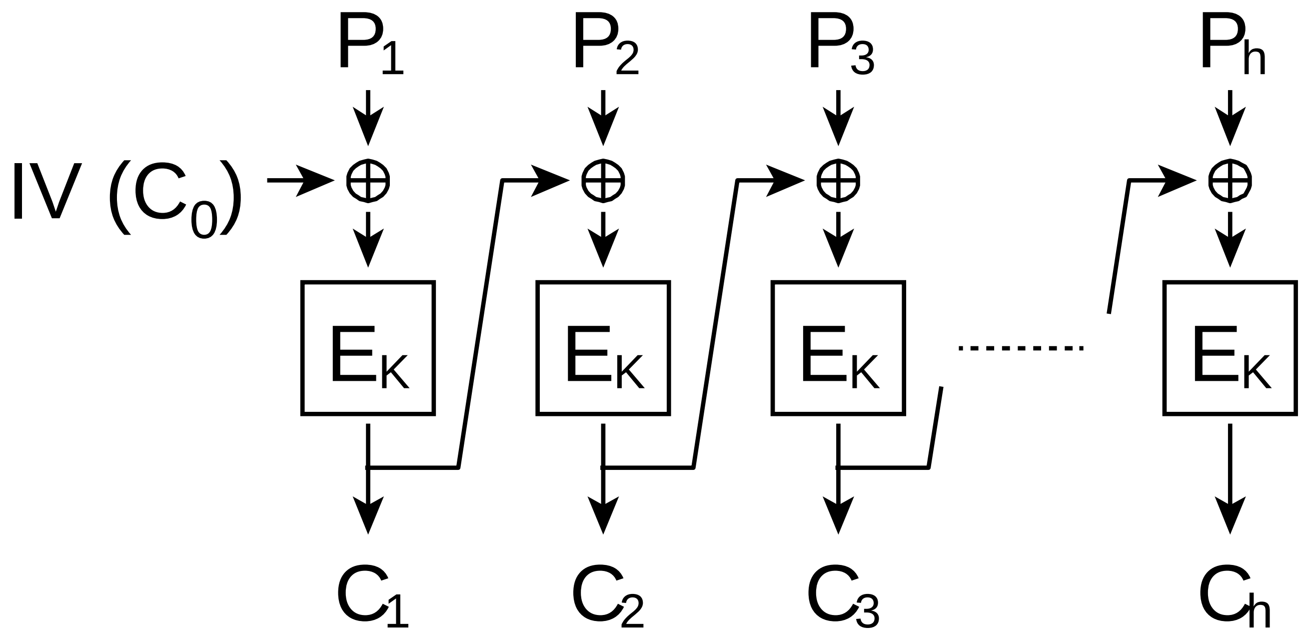
- Attackers can collect multiple ciphertext blocks whose original message is the same but the IV is different.
- Then, we can recover the part of message.



Is this assumption practical?

- It's very difficult questions because it depends on applications.
- We believe it's more practical than KPA.
- Example of vulnerable application.
  - Application sometimes sends the ciphertext of a password for the authentication. And, attackers know the behavior of the application.

CBC mode.



If $\mathrm{E_k}$ has nonlinear invariants,

$$g_p(C_{i-1} \oplus P_i) \oplus g_c(C_i) = \mathrm{const}$$

Message-recovery attacks.

- Attackers know IV and ciphertexts, and $g_p(C_{i-1} \oplus P_i) \oplus g_c(C_i)$ is always constant.
- We collect multiple (C_{i-1}, C_i) whose corresponding P_i is the same.
- By guessing P_i, we can recover it only from ciphertexts.
  - Bits of P_i that involve the nonlinear term of the function g can be recovered.
  - Practically, the time complexity to recover t bits of P_i is at most t^3.

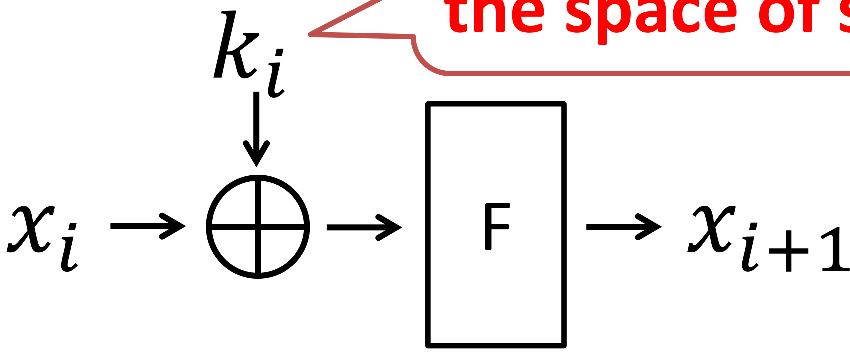
Outline

1. Nonlinear invariant attack.
   - Map of related attacks.
     - Linear and nonlinear cryptanalyses.
     - Invariant subspace attack.
   - Distinguishing attack.
2. Surprising extension toward practical attack.
   - What's happened if vulnerable ciphers are used in well-known mode of operations?
3. **How to find nonlinear invariant.**
   - **Appropriate nonlinear invariants.**
   - **How to find nonlinear invariant for KSP round functions.**
4. Practical attack on full SCREAM.

Nonlinear invariant attack.

Key-alternating structure.

Alternatively, we limit the space of secret keys.



$g_i(x_i) \oplus g_{i+1}(x_{i+1}) = \mathrm{const}$

$$g_i(x_i) \rightarrow g_{i+1}(x_{i+1})$$

with **probability one**.

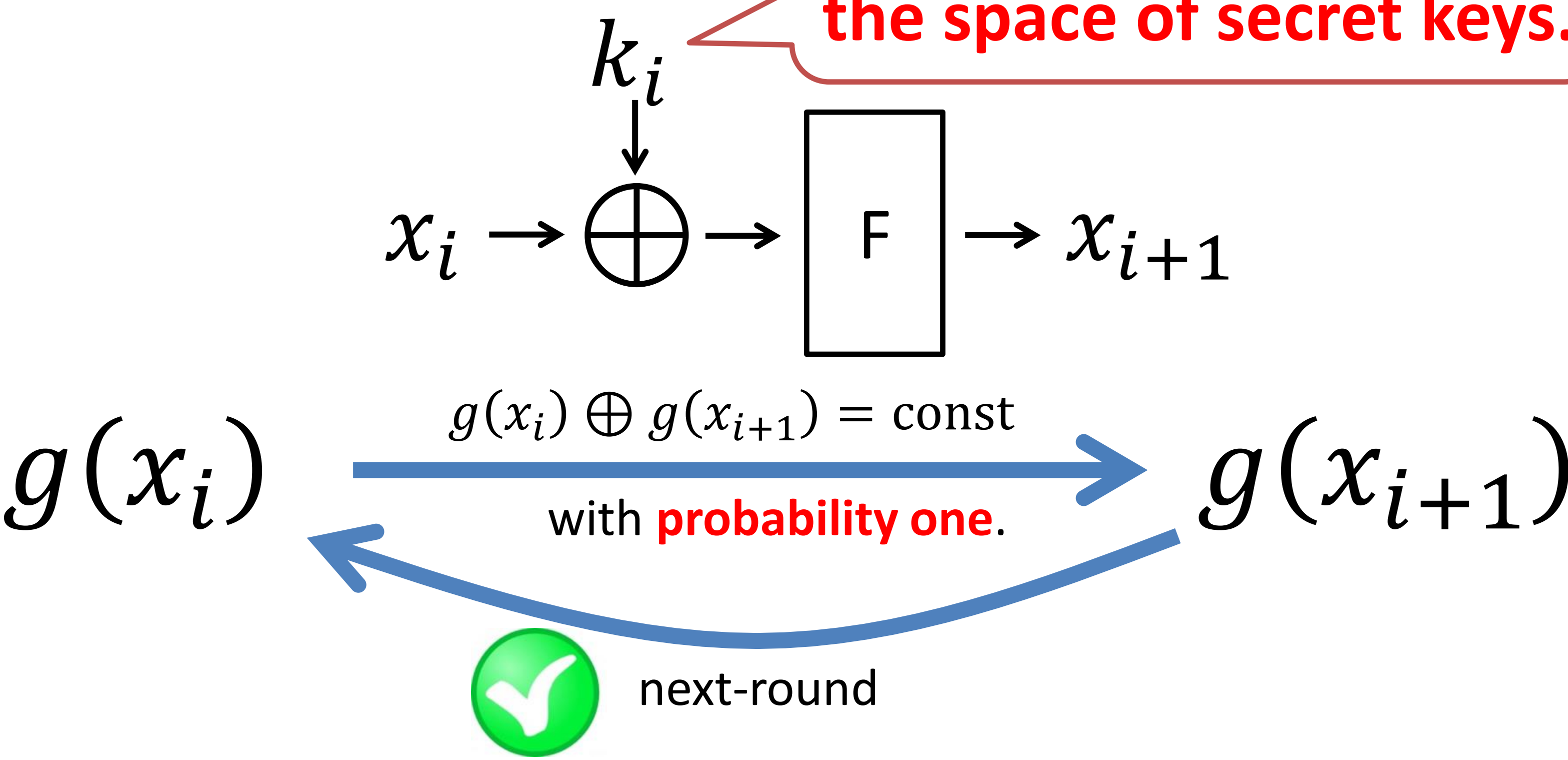
? next-round

We have to search for nonlinear invariants that hold in arbitrary number of rounds.

Nonlinear invariant attack.

Key-alternating structure.

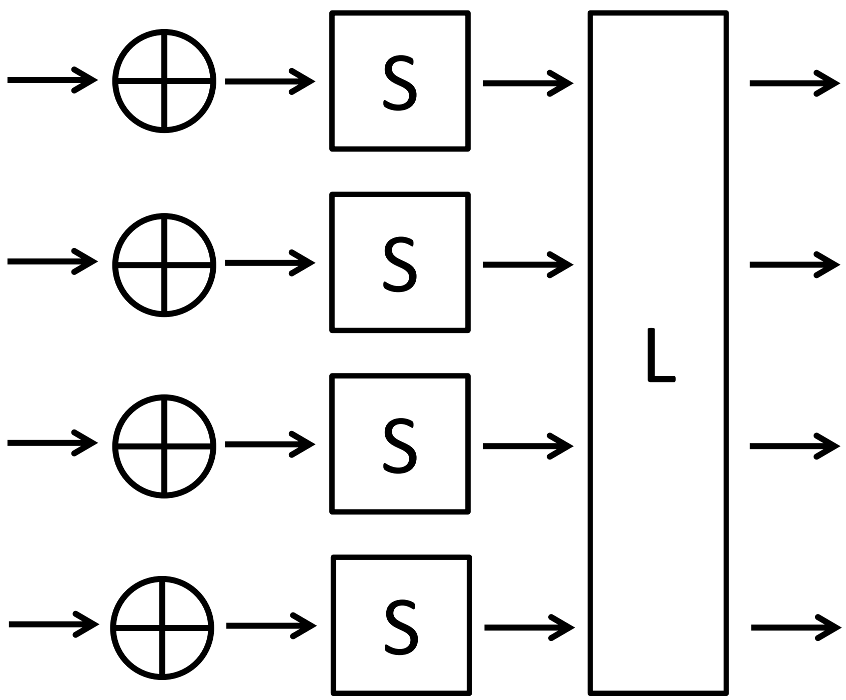
Alternatively, we limit the space of secret keys.



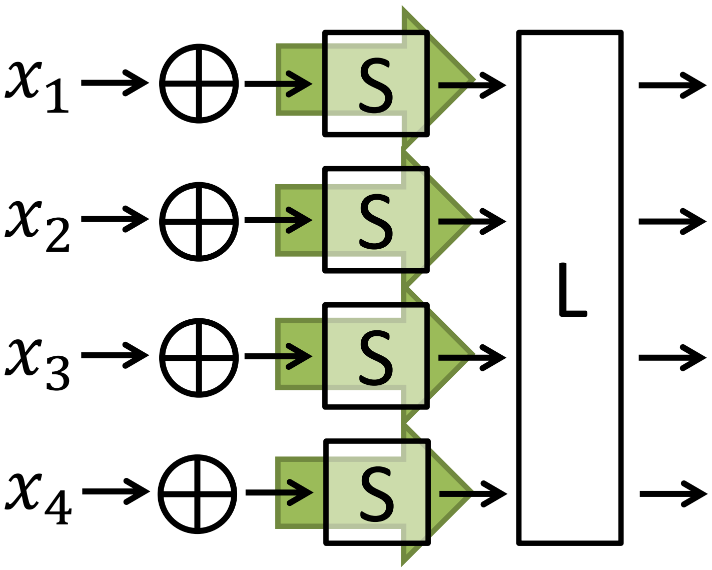
The property tribially holds if $g_i = g_{i+1}$.

Searching for nonlinear invariants.

- Assume that KSP-type round function.



Nonlinear invariants for S-box.



$$g_i(x_i) \oplus g_i\big(S(x_i)\big) = \text{cons}$$

- Because the bit size of S-boxes is generally small, it's not difficult to find nonlinear invariant for S-boxes.

Example.

- Nonlinear invariant for the S-box in Scream.

$$g(x) = x_1 x_2 \oplus x_0 \oplus x_2 \oplus x_5$$

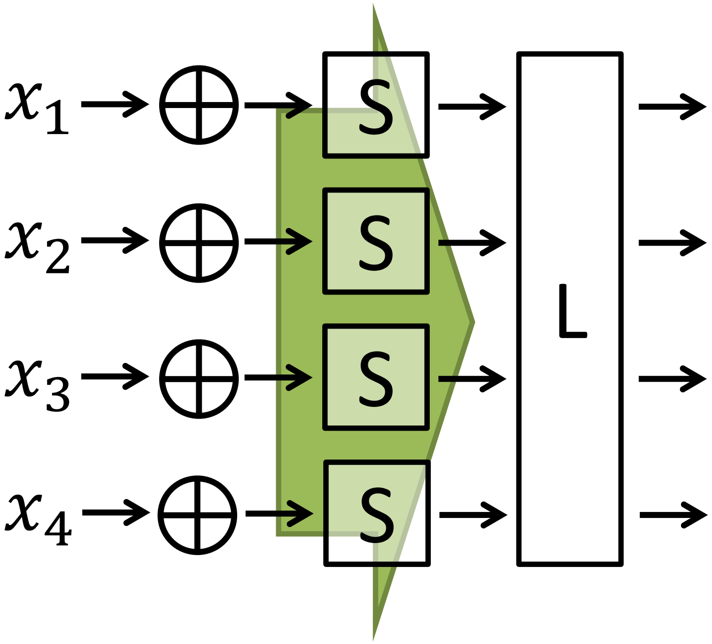
Then, for all $x \in \mathbb{F}_2^8$, $g(x) = g\big(S(x)\big) \oplus 1$.

- Nonlinear invariant for the S-box in Midori64.

$$g(x) = x_2 x_3 \oplus x_0 \oplus x_1 \oplus x_2$$

Then, for all $x \in \mathbb{F}_2^4$, $g(x) = g\big(S(x)\big)$.

Nonlinear invariant for S-box layer.

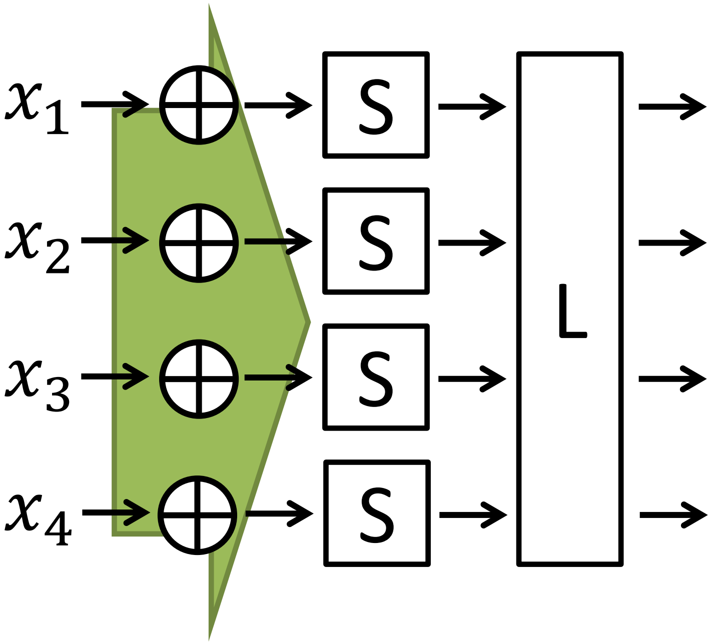


$$g_i(x_i) \oplus g_i\big(S(x_i)\big) = \text{cons}$$

$$g(x) = \bigoplus_{i \in \Lambda} g_i(x_i)$$

- If the function g_i is nonlinear invariant for the ith S-box, the function $\bigoplus_{i \in \Lambda} g_i(x_i)$ becomes nonlinear invariant for the S-box layer for any set Λ.

Nonlinear invariants for key XORing.



$$g(x) \oplus g(x \oplus k) = \text{cons}$$

$$g(x) = \bigoplus_{i \in \Lambda} g_i(x_i)$$

- If "1s" in k are involved in only linear term of the function g, $g(x \oplus k) = g(x) \oplus g(k)$.
- $g(x) \oplus g(x \oplus k) = g(k) = \text{cons}$.

Example.

- Nonlinear invariant for the S-box in Scream.

$$g(x) = x_1 x_2 \oplus x_0 \oplus x_2 \oplus x_5$$

If $k_1 = k_2 = 0$,

$$g(x \oplus k) = g(x) \oplus g(k)$$

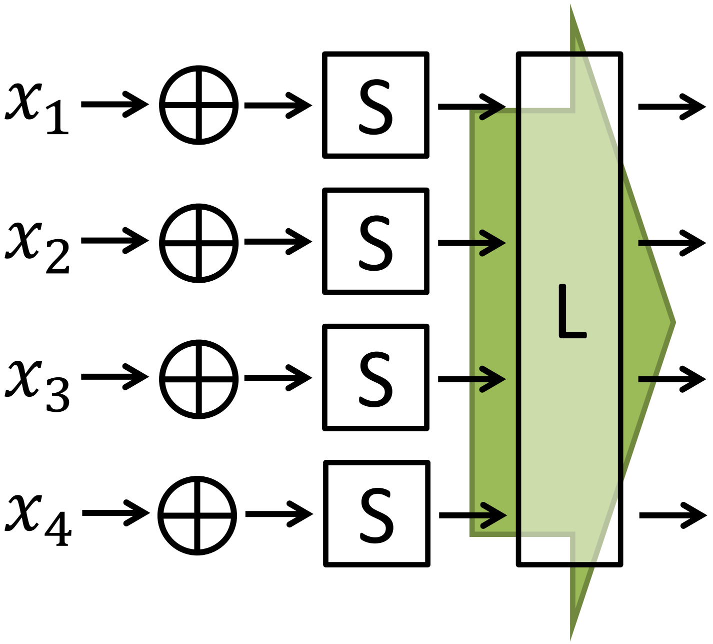
- Nonlinear invariant for the S-box in Midori64.

$$g(x) = x_2 x_3 \oplus x_0 \oplus x_1 \oplus x_2$$

If $k_2 = k_3 = 0$,

$$g(x \oplus k) = g(x) \oplus g(k)$$

Nonlinear invariant for linear layer.



$$g(x) \oplus g\big(L(x)\big) = \text{cons}$$

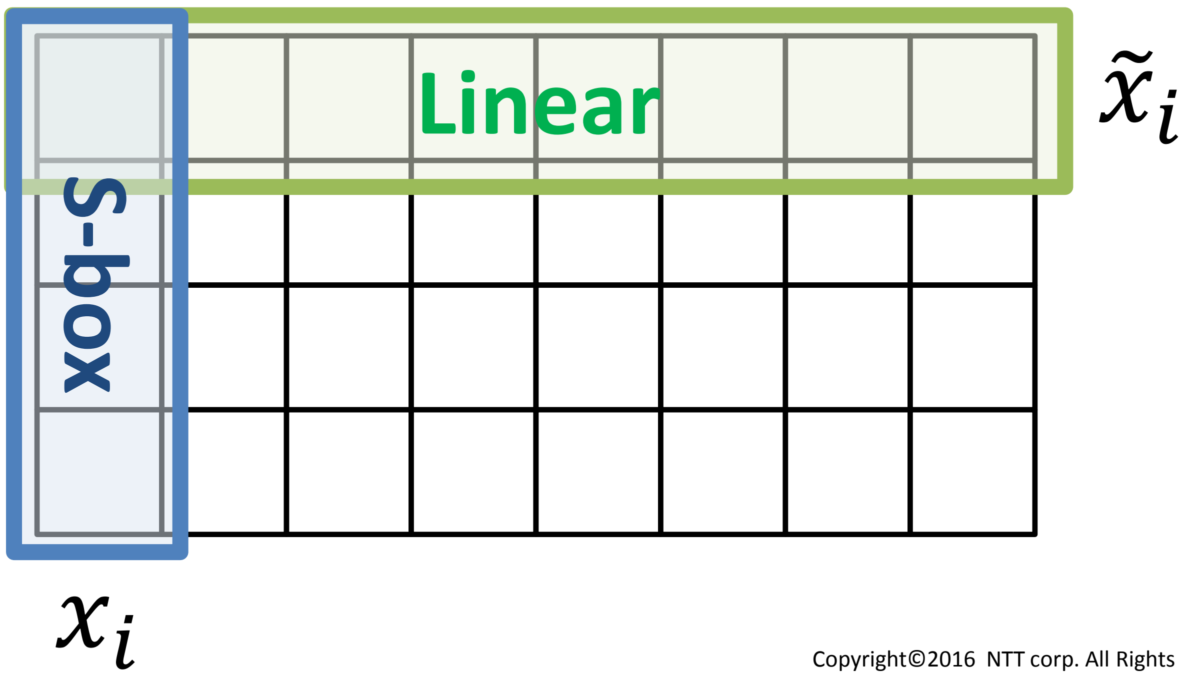
$$g(x) = \bigoplus_{i=1}^{n} g_i(x_i)$$

- If the linear function is **binary orthogonal** and there is a **quadratic invariant** for the S-box, $\oplus\, g_{i=1}^{n}(x_i)$ is nonlinear invariant for the linear layer.

Why binary orthogonal is weak?

- Let $\tilde{x}_i$ be the bit-string by concatenating ith input of all S-boxes. Then, the quadratic invariant is represented as

$$\bigoplus_{i=1}^{n} g_i(x_i) = \bigoplus_{i=1}^{m} \bigoplus_{j=1}^{m} \gamma_{i,j} \langle \tilde{x}_i, \tilde{x}_j \rangle$$



Why binary orthogonal is weak?

- Let $\tilde{x}_i$ be the bit-string by concatenating ith input of all S-boxes. Then, the quadratic invariant is represented as

$$g(x) = \bigoplus_{i=1}^{n} g_i(x_i)$$
$$= \bigoplus_{i=1}^{m} \bigoplus_{j=1}^{m} \gamma_{i,j} \langle \tilde{x}_i, \tilde{x}_j \rangle$$

- Let M be the binary orthogonal matrix, and

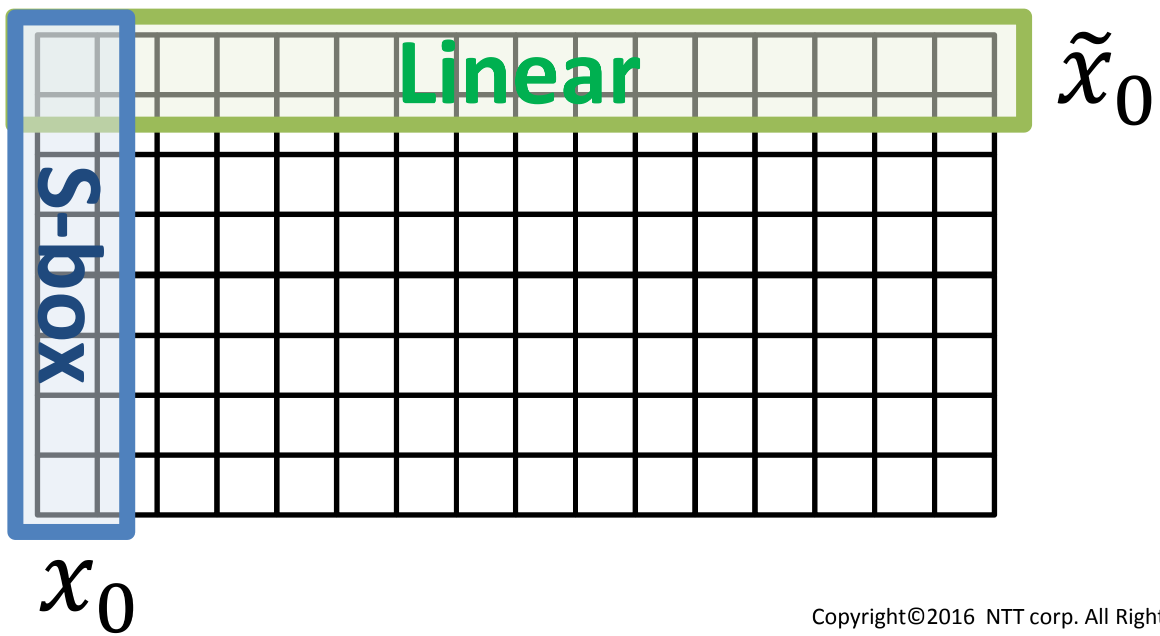
$$g(L(x)) = \bigoplus_{i=1}^{m} \bigoplus_{j=1}^{m} \gamma_{i,j} \langle M\tilde{x}_i, M\tilde{x}_j \rangle$$
$$= \bigoplus_{i=1}^{m} \bigoplus_{j=1}^{m} \gamma_{i,j} \langle \tilde{x}_i, \tilde{x}_j \rangle$$
$$= \bigoplus_{i=1}^{n} g_i(x_i)$$

Outline

1. Nonlinear invariant attack.
   - Map of related attacks.
     - Linear and nonlinear cryptanalyses.
     - Invariant subspace attack.
   - Distinguishing attack.
2. Surprising extension toward practical attack.
   - What's happened if vulnerable ciphers are used in well-known mode of operations?
3. How to find nonlinear invariant.
   - Appropriate nonlinear invariants.
   - How to find nonlinear invariant for KSP round functions.
4. **Practical attack on full SCREAM.**

SCREAM.

- AE proposed for CAESAR.
- LS-design with an orthogonal matrix.
- The secret key is directly used as round keys.
- The round constant is XORed with only $\tilde{x}_0$.



Nonlinear invariant of SCREAM.

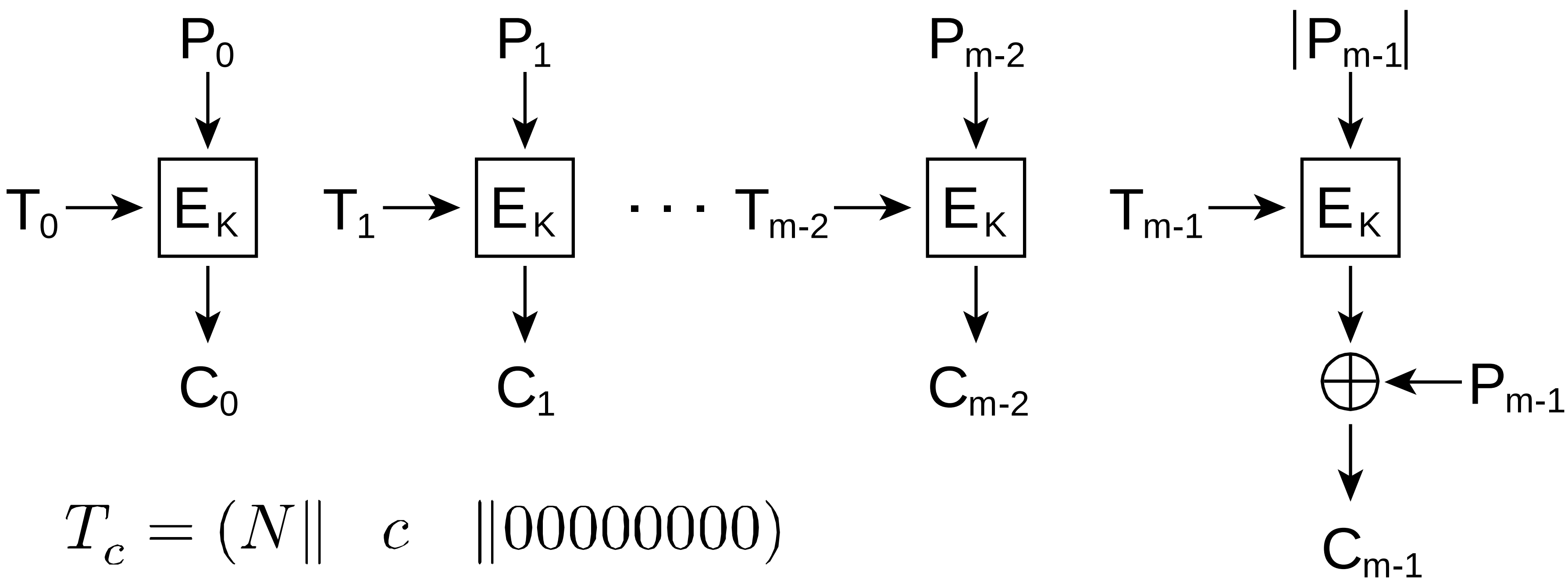
- Nonlinear invariant for Scream.

  $$g(x) = \langle \tilde{x}_1, \tilde{x}_2 \rangle \oplus |\tilde{x}_0| \oplus |\tilde{x}_2| \oplus |\tilde{x}_5|$$

- Since $\tilde{x}_0$ is linearly affected by the function g, the distributive law holds for addConst.
  - $g(x \oplus rc) = g(x) \oplus g(rc)$.
- If $\tilde{k}_1$ and $\tilde{k}_2$ of the secret key are zero (weak keys), the distributive law holds for addRK.
  - $g(x \oplus k) = g(x) \oplus g(k)$.

Application to SCREAM AE.

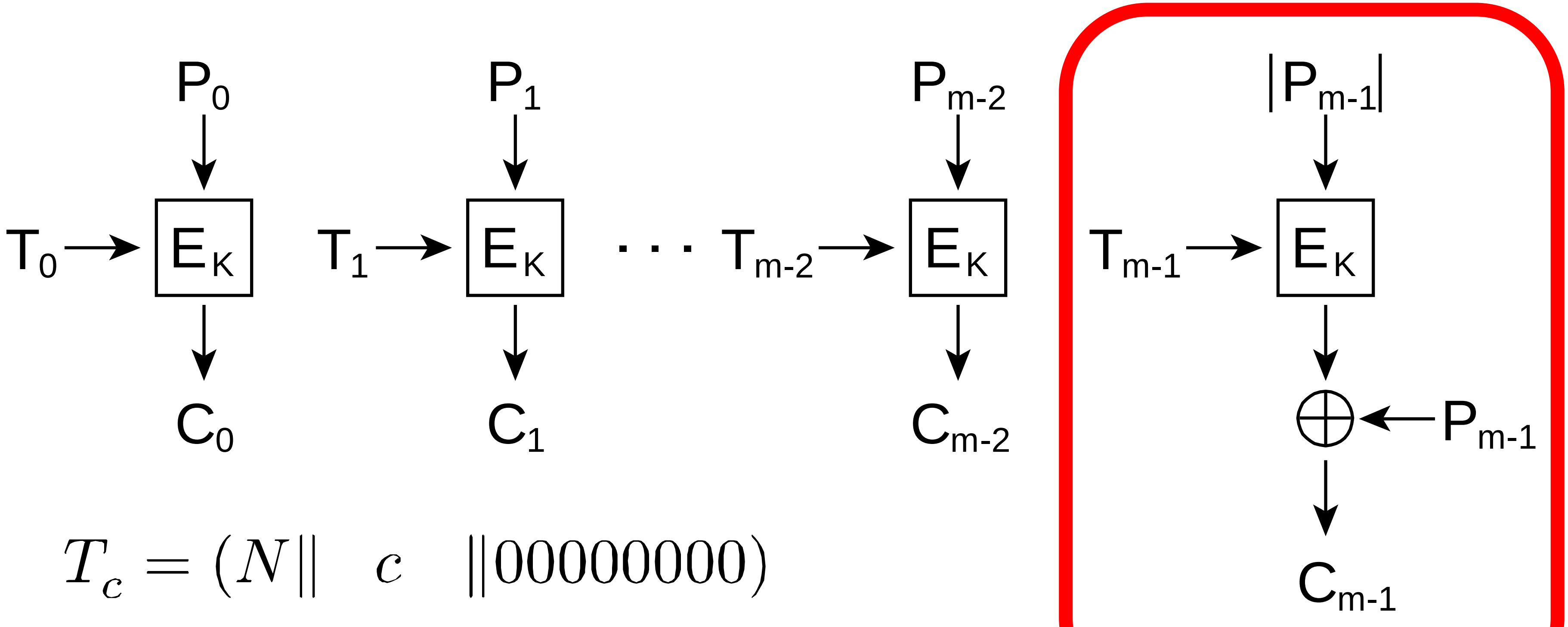
- SCREAM authenticated encryption.



$$T_c = (N \| \quad c \quad \| 00000000)$$

Application to SCREAM AE.

- SCREAM authenticated encryption.



$$T_c = (N \| \quad c \quad \| 00000000)$$

$$g(|P_{m-1}|) \oplus g(P_{m-1} \oplus C_{m-1}) = \text{const}$$

known guess known

Summary of results.

Distinguishing attack under known-plaintext setting.

Target	# of weak keys	Data complexity.	Distinguishing probability.
SCREAM	2^{96}	k	$1 - 2^{1-k}$
iSCREAM	2^{96}		
Midori64	2^{64}		

Message-recover attack under ciphertext-only setting.

Target	# of weak keys	Maximum # of recovered bits.	Data complexity.	Time complexity.
SCREAM	2^{96}	32 bits	33 ciphertexts	$32^3 = 2^{15}$
iSCREAM	2^{96}	32 bits	33 ciphertexts	$32^3 = 2^{15}$
Midori64-CTR	2^{64}	32h bits	33h ciphertexts	$32^3 h = 2^{15} h$

h is the number of blocks in the mode of operations.

Conclusion.

- Proposal of nonlinear invariant attack.
- Method to find nonlinear invariants.
- Nonlinear invariant attack on Scream, iScream, and Midori64.
  - We can recover the 32bits of message in the last block on SCREAM (iSCREAM) AEs.
  - We can recover the 32bits of message in every block on CBC, CTR, CFB, OFB modes.