Inserting Backdoors in Tweakable Block Ciphers

Thomas Peyrin
(joint work with Haoyang Wang)

NTU - Singapore

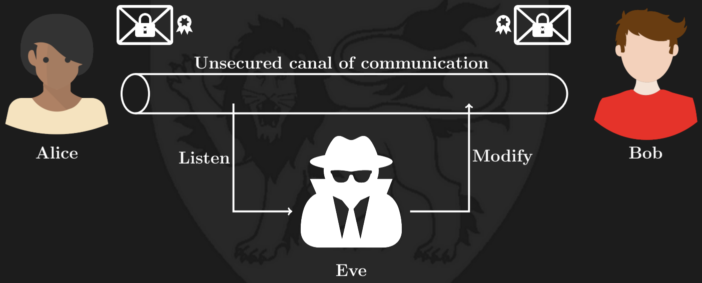
Technology Innovation Institute
UAE - May 27, 2021

Outline

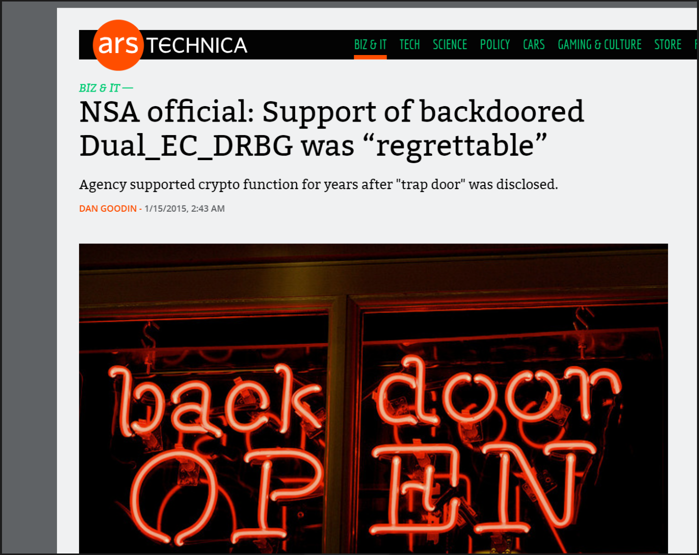
1. Backdoors in Symmetric-Key Cryptography
2. Previous Backdoors Attempts
   - ▷ In the wild
   - ▷ Academic work
3. The MALICIOUS Framework
   - ▷ Preliminaries
   - ▷ The MALICIOUS Framework
   - ▷ The MALICIOUS Security
4. LowMC-M : A backdoored TBC Variant of LowMC
5. Future Directions

Encryption algorithms

Encryption algorithms are used to protect messages sent through an unsecured channel against malicious adversaries.



Can we trust encryption algorithms?



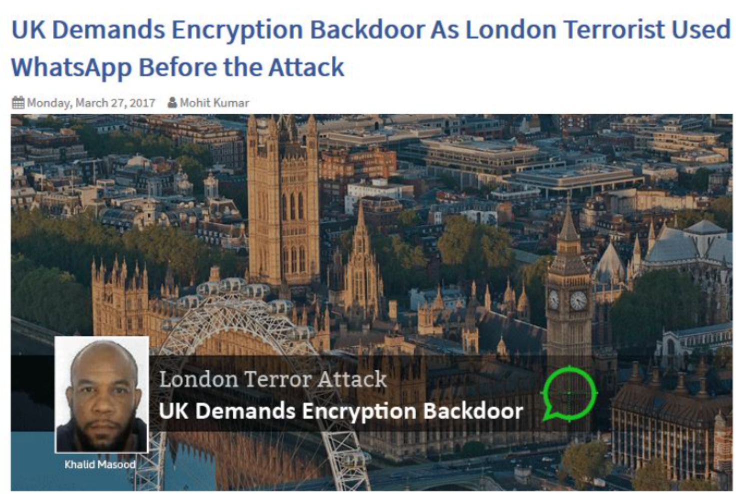
BIZ & IT —

NSA official: Support of backdoored Dual_EC_DRBG was “regrettable”

Agency supported crypto function for years after "trap door" was disclosed.

DAN GOODIN - 1/15/2015, 2:43 AM

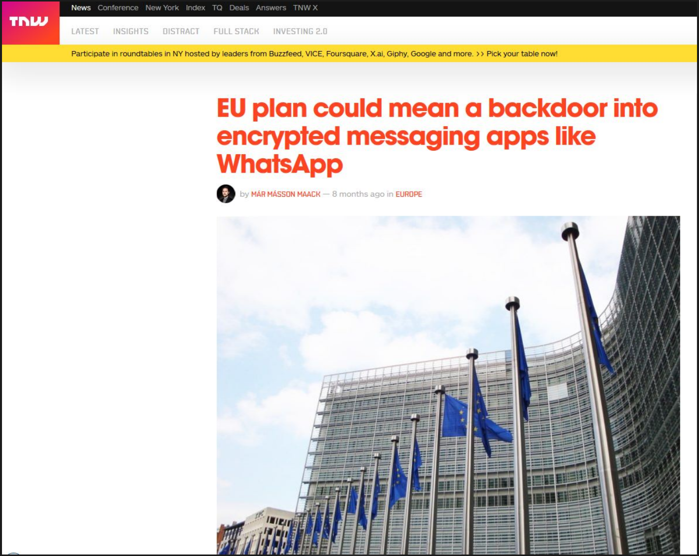
Can we trust encryption algorithms?



UK Demands Encryption Backdoor As London Terrorist Used WhatsApp Before the Attack

Monday, March 27, 2017 Mohit Kumar

Can we trust encryption algorithms?

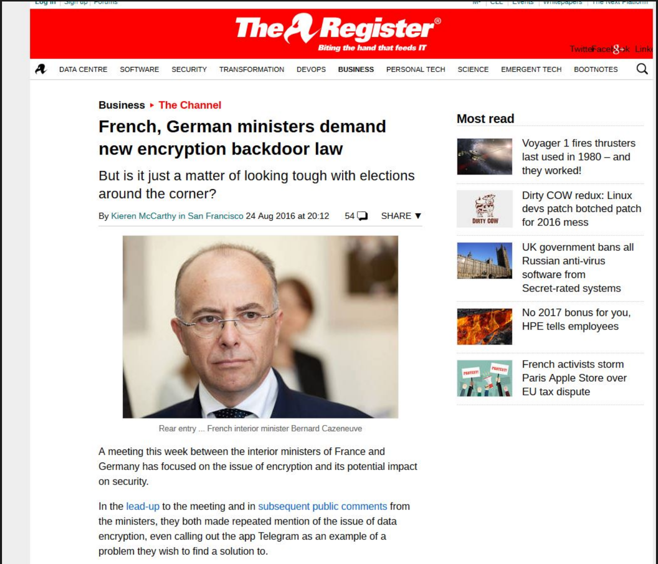

TNW
News Conference New York Index TQ Deals Answers TNW X
LATEST INSIGHTS DISTRACT FULL STACK INVESTING 2.0

Participate in roundtables in NY hosted by leaders from Buzzfeed, VICE, Foursquare, X.ai, Giphy, Google and more. >> Pick your table now!

EU plan could mean a backdoor into encrypted messaging apps like WhatsApp

by MÁR MÁSSON MAACK — 8 months ago in EUROPE

Can we trust encryption algorithms?


Log in | Sign up | Forums
Events | Whitepapers | The Next Platform
The Register®
Biting the hand that feeds IT
Twitter Facebook Google+ LinkedIn
DATA CENTRE
SOFTWARE
SECURITY
TRANSFORMATION
DEVOPS
BUSINESS
PERSONAL TECH
SCIENCE
EMERGENT TECH
BOOTNOTES
DIRTY COW
PROTEST!

Business ▸ The Channel

French, German ministers demand new encryption backdoor law

But is it just a matter of looking tough with elections around the corner?

By Kieren McCarthy in San Francisco 24 Aug 2016 at 20:12 54 SHARE ▼

Rear entry ... French interior minister Bernard Cazeneuve

A meeting this week between the interior ministers of France and Germany has focused on the issue of encryption and its potential impact on security.

In the lead-up to the meeting and in subsequent public comments from the ministers, they both made repeated mention of the issue of data encryption, even calling out the app Telegram as an example of a problem they wish to find a solution to.

Most read

Voyager 1 fires thrusters last used in 1980 – and they worked!

Dirty COW redux: Linux devs patch botched patch for 2016 mess

UK government bans all Russian anti-virus software from Secret-rated systems

No 2017 bonus for you, HPE tells employees

French activists storm Paris Apple Store over EU tax dispute

Backdoors and Backdoors

Two types of backdoors :

- ▷ Most of time, backdoors in a security system refer to weaknesses intentionally created at implementation level, such as key distribution/key management protocols, hardcoded secret authorised login accounts (sometimes for developers), malware hidden in a program, etc.
- ▷ The other type is the **cryptographic backdoor**, which is embedded during the **design phase of a cryptographic algorithm**.

Backdoors and Trapdoors

Trapdoor $\neq$ Backdoor

In this talk, we will design a cipher that has a trapdoor that can also be used as backdoor.

Backdoors expectations

What do we want from a backdoor/trapdoor?

- (easy) the backdoor/trapdoor should allow the attacker to easily obtain some information about the secret key or some plaintext
- (medium) the backdoor should be hard to detect given only blackbox access to the cipher
- (medium) the backdoor/trapdoor should not introduce weaknesses in the cipher when the backdoor/trapdoor is not used
- (hard) the backdoor should be hard to detect/retrieve given the cipher specifications
- (impossible?) the backdoor/trapdoor should be hard to detect/retrieve given the cipher specifications and given some communication transcripts where the backdoor/trapdoor has been used

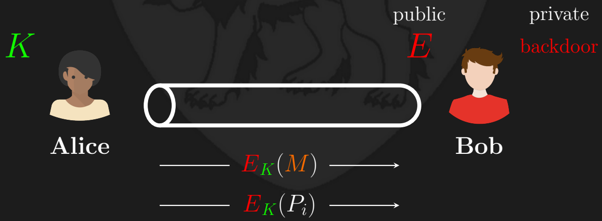
Why studying backdoors

Why studying backdoors ?

- ▷ for **fun** !
- ▷ better understand what is doable in terms of backdooring a cipher
- ▷ better detection/protection against backdoors insertions
- ▷ backdoored ciphers have a lot of useful applications for governmental agencies
- ▷ use backdoors at your advantage : public-key encryption from a backdoored cipher ? (Blaze *et al.* - CRYPTO'95)

Public-key encryption from backdoored cipher

- ▷ Bob creates a block cipher E with a backdoor and makes it widely available (it is a system-wide public parameter). This represents his public key, while the backdoor represents the private key.
- ▷ If Alice wants to send a confidential message to Bob, she generates a random session key K, encrypts her message M and a fixed set of plaintexts P_i with Bob's cipher and sends the ciphertexts to Bob.
- ▷ Bob uses the backdoor and the known plaintexts to recover key K and can eventually recover the message M.



public

private

K

E

backdoor

Alice

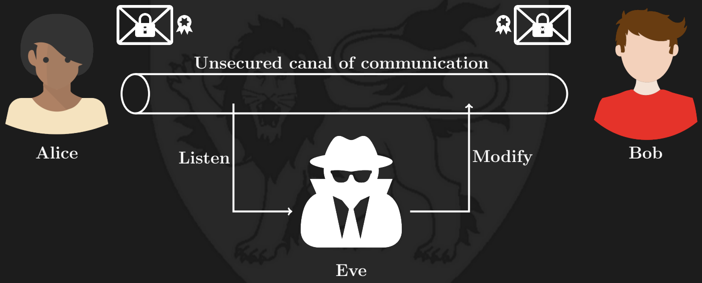
Bob

$E_K(M)$

$E_K(P_i)$

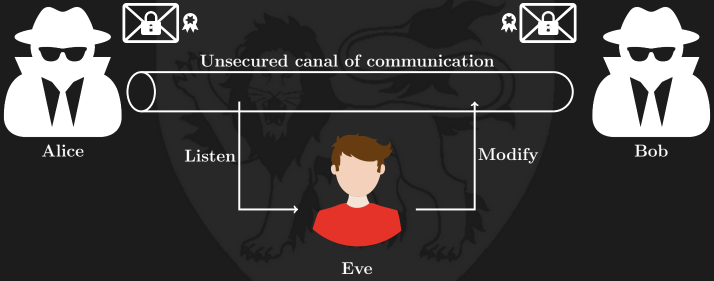
Backdoor model

In the backdoor world, **we are Eve** and the **adversaries are the normal users**, trying to exploit flaws in our backdoor.



Backdoor model

In the backdoor world, **we are Eve** and the **adversaries are the normal users**, trying to exploit flaws in our backdoor.



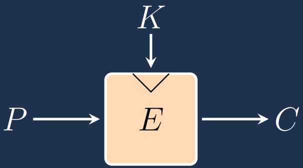
How are (many) ciphers designed

How are (many) ciphers designed?

What is a block cipher?

Block cipher E :

- E_K maps a n-bit plaintext P (unencrypted text) into a n-bit ciphertext C (encrypted text) with k-bit secret key K. Typically, $n = k = 128$ (AES-128 : current worldwide standard).
- Must be invertible !



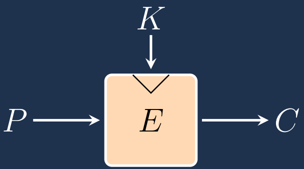
One should NOT be able to :

- recover the secret k-bit key K faster than brute-force (2^k)
- extract any information about the plaintext or the ciphertext

What is a block cipher?

Block cipher E :

- E_K maps a n-bit plaintext P (unencrypted text) into a n-bit ciphertext C (encrypted text) with k-bit secret key K. Typically, $n = k = 128$ (AES-128 : current worldwide standard).
- Must be invertible!



Many applications of block ciphers :

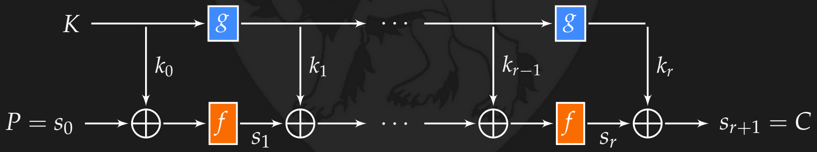
- **Confidentiality.** When used in an operating mode, it allows to securely transmit data over an insecure channel
- **Building block for other cryptography primitives.** Such as hash functions, stream-ciphers, MACs, etc.

General construction of a block cipher : iterated block ciphers

An iterated block cipher is composed of two parts :

- a key schedule that generates $r+1$ subkeys $K \to (k_0, \ldots, k_r)$
- an internal permutation f repeated r times
  (also named round function)

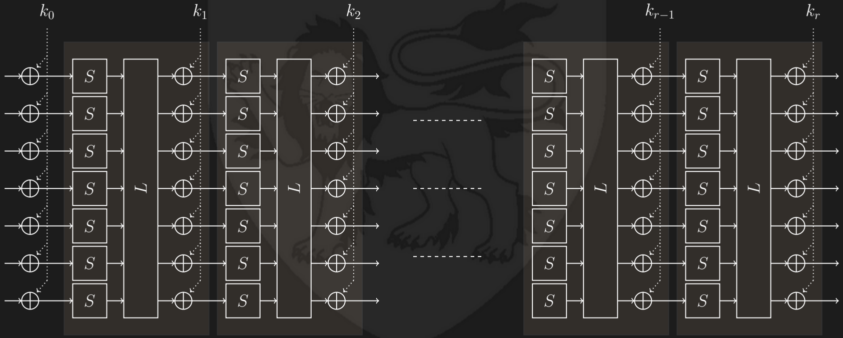
An iterative design allows compact implementations (put the round function in a `for` loop) and simplicity of analysis.



General construction of a block cipher : iterated block ciphers

An iterated block cipher is composed of two parts :

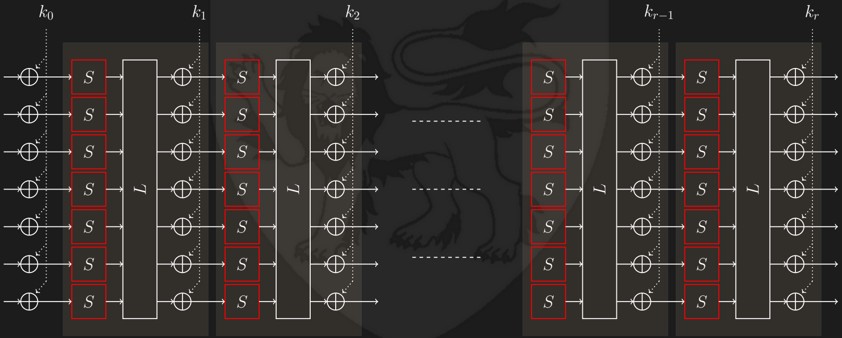
- a key schedule that generates $r+1$ subkeys $K \rightarrow (k_0, \ldots, k_r)$
- an internal permutation f repeated r times (also named round function)



General construction of a block cipher : iterated block ciphers

An iterated block cipher is composed of two parts :

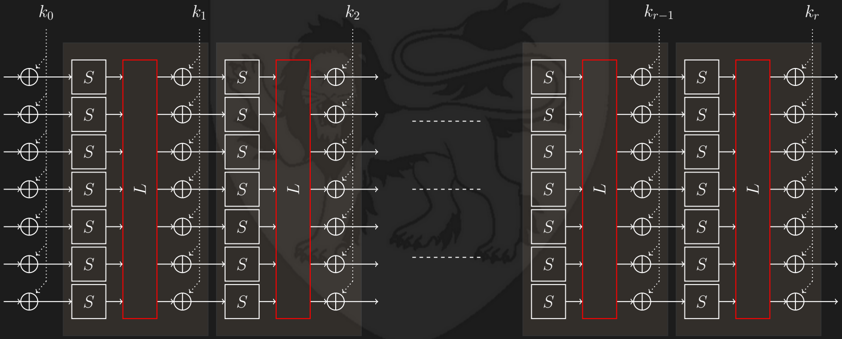
- a key schedule that generates $r+1$ subkeys $K \rightarrow (k_0, \ldots, k_r)$
- an internal permutation f repeated r times (also named round function)



General construction of a block cipher : iterated block ciphers

An iterated block cipher is composed of two parts :

- ▷ a key schedule that generates $r+1$ subkeys $K \rightarrow (k_0, \ldots, k_r)$
- ▷ an internal permutation f repeated r times (also named round function)



Outline

1 Backdoors in Symmetric-Key Cryptography

2 Previous Backdoors Attempts

- ▷ In the wild
- ▷ Academic work

3 The MALICIOUS Framework

- ▷ Preliminaries
- ▷ The MALICIOUS Framework
- ▷ The MALICIOUS Security

4 LowMC−M : A backdoored TBC Variant of LowMC

5 Future Directions

Outline

1. Backdoors in Symmetric-Key Cryptography
2. Previous Backdoors Attempts
   - ▷ In the wild
   - ▷ Academic work
3. The MALICIOUS Framework
   - ▷ Preliminaries
   - ▷ The MALICIOUS Framework
   - ▷ The MALICIOUS Security
4. LowMC−M : A backdoored TBC Variant of LowMC
5. Future Directions

NSA's Clipper Chip

Clipper Chip :

- **designed by the NSA** and promoted by the US government
- introduced in 1993 and discontinued in 1996 (it had flaws - Blaze CCS'94)
- the chip encrypts voice communications for telcos
- 64-bit block cipher is `Skipjack` (declassified in 1998), with 80-bit key
- **key escrow feature** : two clipper chips will output an encrypted version of the session key used for communication, with a hardcoded device key that can be retrieved by the US government

NATIONAL SECURITY AGENCY
UNITED STATES OF AMERICA

Dual_EC_DBRG

Dual_EC_DBRG :

Dual Elliptic Curve Deterministic Random Bit Generator.
Used to generate random keys. ISO and ANSI standards.
Nobody knows where the parameters came from...

Timeline :

- **2003 :** Dual_EC_DRBG enters in ANSI X9.82
- **2004 :** the RSA Security made Dual_EC_DRBG the default PRNG in their BSAFE crypto lib.
- **2005-2007 :** works from the academic community suggested the existence of a **backdoor** in Dual_EC_DRBG
- **Sept. 2013 :** Snowden leaks confirm the backdoor
- **Dec. 2013 :** Reuters reported that before NIST standardized Dual_EC_DRBG, **NSA** paid RSA Security $10 million in a secret deal to use Dual_EC_DRBG as the default

Russian Kuznyechik block cipher and Streebog hash function

Kuznyechik block cipher and Streebog hash function :

Russian GOST R standards (2012 and 2015).
Also IETF (both), ISO/IEC standard (Streebog)
Nobody knows how the Sbox was generated...

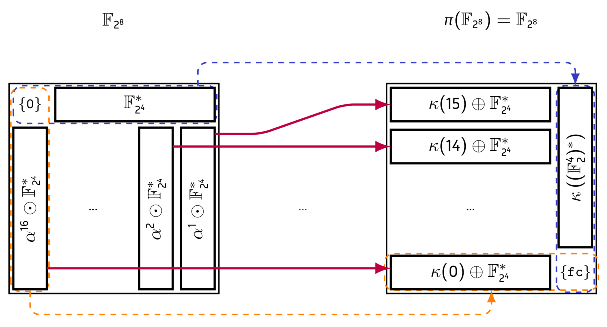
Timeline :

- Perrin and colleagues identified since 2015 weaknesses in the Sbox of Kuznyechik and Streebog, basically a strong algebraic structure
- The designers claimed that the Sbox was generated randomly
- This structure is EXTREMLY unlikely to be present in a randomly generated Sbox
- Designers recently claimed that they “lost” the program that randomly generated the Sbox ...

Russian Kuznyechik block cipher and Streebog hash function

Kuznyechik block cipher and Streebog hash function :

Russian GOST R standards (2012 and 2015).
Also IETF (both), ISO/IEC standard (Streebog)
Nobody knows how the Sbox was generated...


$\mathbb{F}_{2^8}$
$\pi(\mathbb{F}_{2^8}) = \mathbb{F}_{2^8}$
$\{0\}$
$\mathbb{F}^*_{2^4}$
$\alpha^{16} \odot \mathbb{F}^*_{2^4}$
...
$\alpha^{2} \odot \mathbb{F}^*_{2^4}$
$\alpha^{1} \odot \mathbb{F}^*_{2^4}$
...
$\kappa(15) \oplus \mathbb{F}^*_{2^4}$
$\kappa(14) \oplus \mathbb{F}^*_{2^4}$
...
$\kappa(0) \oplus \mathbb{F}^*_{2^4}$
$\kappa((\mathbb{F}_2^4)^*)$
$\{\mathtt{fc}\}$

(Image from who.paris.inria.fr/Leo.Perrin/pi.html)

NSA's SIMON and Speck ?

SIMON and Speck lightweight block ciphers :

Block ciphers designed by **NSA**
Very lightweight (RFID tags, IoT devices, etc.)
No design analysis/justification provided...

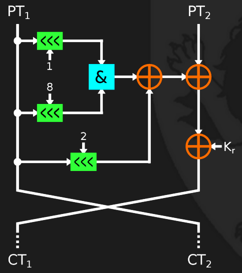
Timeline :

- **2013 :** NSA releases SIMON and Speck, two families of lightweight encryption algorithms (block ciphers)
- **Since 2014 :** the NSA tries to standardise SIMON and Speck at ISO
- **2016 and 2017 :** small block sizes variant (32-bit blocks !) are eventually removed from the standardization push
- **2017 :** ISO rejects SIMON and Speck under the pressure of experts from the academic community and from ISO

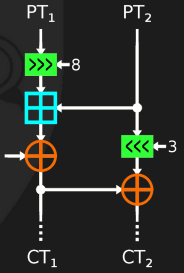
NSA's SIMON and Speck ?

SIMON and Speck lightweight block ciphers :

Block ciphers designed by **NSA**
Very lightweight (RFID tags, IoT devices, etc.)
No design analysis/justification provided...



SIMON round function



Speck round function

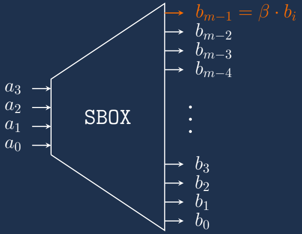
Outline

1. Backdoors in Symmetric-Key Cryptography
2. Previous Backdoors Attempts
   - ▷ In the wild
   - ▷ Academic work
3. The MALICIOUS Framework
   - ▷ Preliminaries
   - ▷ The MALICIOUS Framework
   - ▷ The MALICIOUS Security
4. `LowMC-M` : A backdoored TBC Variant of `LowMC`
5. Future Directions

First tries : Rijmen-Preneel 1997

Rijmen and Preneel - FSE 1997

Idea : hide a high-probability linear relation in an expending $n \to m$ S-box.
Using the backdoor, you can get the secret key using simple linear cryptanalysis.

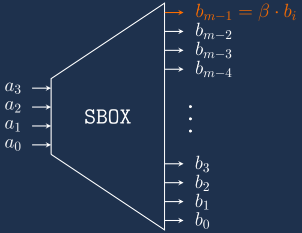


- ▷ they show that **only one such weakness** is introduced
- ▷ they show that this weakness is hard to find compared to a randomly chosen Sbox (because the Sbox output is large, like $m = 64$ bits or more)
- ▷ they presented backdoored versions of `CAST` and `LOKI` block ciphers

First tries : Rijmen-Preneel 1997

Rijmen and Preneel - FSE 1997

Idea : hide a high-probability linear relation in an expending $n \rightarrow m$ S-box.
Using the backdoor, you can get the secret key using simple linear cryptanalysis.



Broken by Wu *et al.* (ASIACRYPT'98) as well as the general strategy proposed.

They show how to find the backdoor in the Sbox and explain that it is **impossible to adjust the parameters of the Sbox to make the detection difficult**.

First tries : partition-based backdoor ciphers

Partition-based backdoor ciphers (Paterson - FSE'99)

Idea : the round function preserves a **partition of the message space** no matter the round keys used, and hence the same applies to the full cipher. You can create a backdoor out of it using a carefully crafted key schedule.

They proposed a backdoored version of `DES` block cipher from this idea.

However, **the backdoor is detectable** and the cipher not resistant against simple differential cryptanalysis.

Work generalised by Bannier and Filiol to SPN ciphers, with the proposal of `BEA-1` backdoored block cipher.
However, no concrete backdoor security is provided.

Kleptography

Kleptography (Young and Yung - CRYPTO'96)

Kleptography is the study of stealing information securely and **subliminally**. They shows how to build a covert key exchange into the Diffie–Hellman key exchange protocol (very close to the Dual_EC_DBRG backdoor), RSA key generation, etc.

Also several backdoors in secret block ciphers using **subliminal channel** (FSE'98, SAC'04, ACISP'03), but assuming that the cipher specifications are unknown to the adversary.

Backdooring SHA-1

Hash function backdoor (Aumasson *et al.* - SAC'14) :

Idea : take an existing costly cryptanalysis against a hash function (SHA-1) and **change constants at your advantage** to reduce the attack's cost and **find an actual collision**.

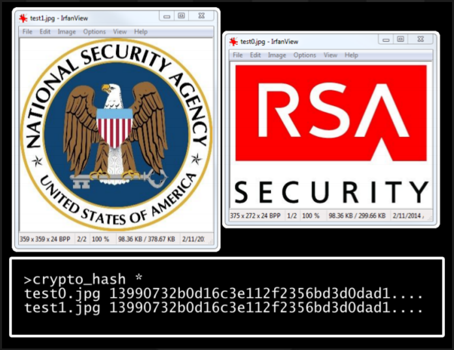
Backdoor : publish the new design - only the designer will know that collision, while it will remain intractable for other users to find one.

Applications : many file formats are vulnerable against this type of collision - one can create colliding images, documents, executables, ...

Backdooring `SHA-1`

Hash function backdoor (Aumasson *et al.* - SAC'14) :

Idea : take an existing costly cryptanalysis against a hash function (`SHA-1`) and **change constants at your advantage** to reduce the attack's cost and **find an actual collision**.


test1.jpg - IrfanView
File Edit Image Options View Help
NATIONAL SECURITY AGENCY
UNITED STATES OF AMERICA
359 x 359 x 24 BPP 2/2 100 % 98.36 KB / 378.67 KB 2/11/20
test0.jpg - IrfanView
File Edit Image Options View Help
RSA
SECURITY
375 x 272 x 24 BPP 1/2 100 % 98.36 KB / 299.66 KB 2/11/2014 ,

```
>crypto_hash *
test0.jpg 13990732b0d16c3e112f2356bd3d0dad1....
test1.jpg 13990732b0d16c3e112f2356bd3d0dad1....
```

Modified `SHA-1` collision (image from Aumasson *et al.* - SAC'14)

Backdoor security

Security notions for backdoors

- ▷ **Undetectability** : given the specifications of the cipher and the general form of the backdoor, it should be impossible for an attacker to tell if the cipher is backdoored or not.
  Is there a backdoor ?
- ▷ **Undiscoverability** : given the specifications of backdoored cipher, it should be impossible for an attacker to retrieve the backdoor.
  What is the backdoor ?
- ▷ **Untraceability** : given an attack based on the backdoor, it should not reveal any information about the backdoor itself.
  Can I retrieve the backdoor when being used ?

Academic research

- Relatively limited number of works focus on the research of cryptographic backdoors.
- Almost all designs were either broken or don’t provide strong security arguments.

In this work (published at CRYPTO 2020) :

- We propose the **MALICIOUS framework** to embed backdoors into tweakable block ciphers.
- We show that our backdoor is efficient.
- We provide a concrete security analysis for our backdoor.
- We provide a cipher example `LowMC-M`, and analyse its backdoor security and classical cipher security.
- Our design is undetectable, undiscoverable, but traceable.

Outline

1. Backdoors in Symmetric-Key Cryptography
2. Previous Backdoors Attempts
   - ▷ In the wild
   - ▷ Academic work
3. The MALICIOUS Framework
   - ▷ Preliminaries
   - ▷ The MALICIOUS Framework
   - ▷ The MALICIOUS Security
4. LowMC–M : A backdoored TBC Variant of LowMC
5. Future Directions

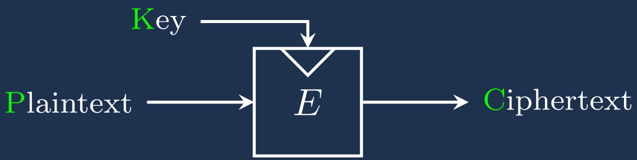
How to create a backdoored tweakable block cipher ?

Outline

1. Backdoors in Symmetric-Key Cryptography
2. Previous Backdoors Attempts
   - ▷ In the wild
   - ▷ Academic work
3. The MALICIOUS Framework
   - ▷ Preliminaries
   - ▷ The MALICIOUS Framework
   - ▷ The MALICIOUS Security
4. LowMC-M : A backdoored TBC Variant of LowMC
5. Future Directions

Block ciphers

A **block cipher** is a family of permutations operating on a fixed-length block of message, the family being parametrized by the secret key input.

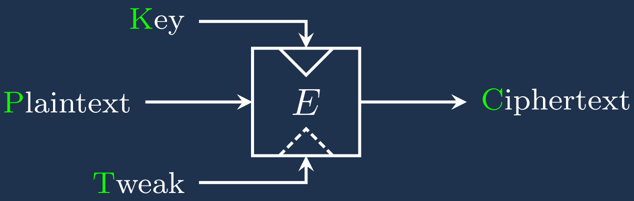


Example : Advanced Encryption Standard (`AES`) with 128-bit block and key

Tweakable block ciphers

A **tweakable block cipher** accepts an additional input, so-called **tweak**, in order to parametrize the family of permutations even if the key is fixed.

- ▷ No need to keep the tweak secret.
- ▷ An attacker could even have full control over the tweak, i.e., choosing whatever value he wants.



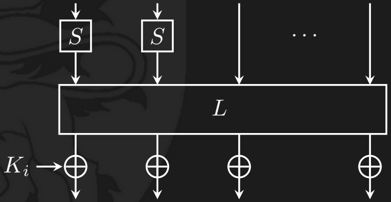
Examples : `Deoxys-TBC`, `SKINNY`, ...

Block ciphers with partial non-linear layers

Substitution-Permutation Network (SPN)

SPN is a method of designing iterated block ciphers, an SPN round consists of a linear layer and a non-linear layer.

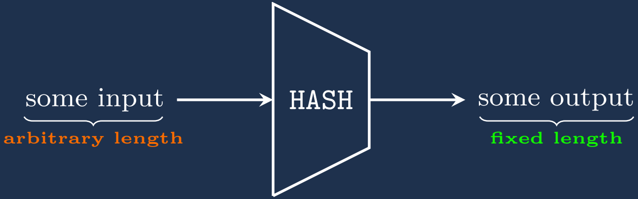
Partial non-linear layer : the non-linear layer (S-boxes) is only applied to a subpart of the internal state.



Hash functions

A hash function is a function which maps an arbitrary length input to a fixed length output.

- **Security properties :** collision resistance, preimage resistance and second preimage resistance, up to the output size n (resp. $2^{n/2}$, 2^n, 2^n ideally)

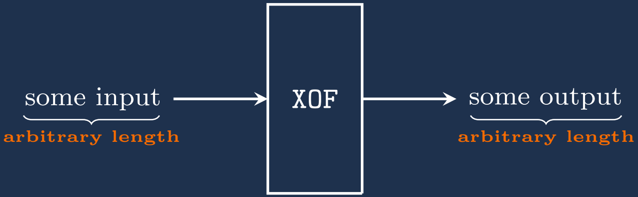


Examples : SHA-2, SHA-3.

Extendable-output functions - XOF

An extendable-output function (XOF) is a generalization of a hash function which maps an arbitrary length input to an arbitrary length output.

- ▷ **Security properties :** collision resistance, preimage resistance and second preimage resistance up to a certain security level that does not fully depend on the output size.



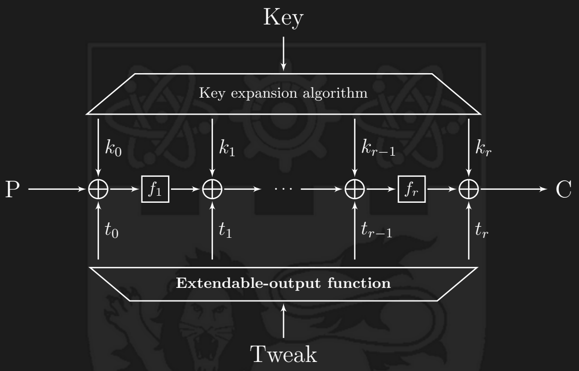
$$\underbrace{\text{some input}}_{\text{arbitrary length}} \longrightarrow \boxed{\texttt{XOF}} \longrightarrow \underbrace{\text{some output}}_{\text{arbitrary length}}$$

Examples : SHAKE128, SHAKE256.

Outline

1. Backdoors in Symmetric-Key Cryptography
2. Previous Backdoors Attempts
   - ▷ In the wild
   - ▷ Academic work
3. **The MALICIOUS Framework**
   - ▷ Preliminaries
   - ▷ The MALICIOUS Framework
   - ▷ The MALICIOUS Security
4. `LowMC-M` : A backdoored TBC Variant of `LowMC`
5. Future Directions

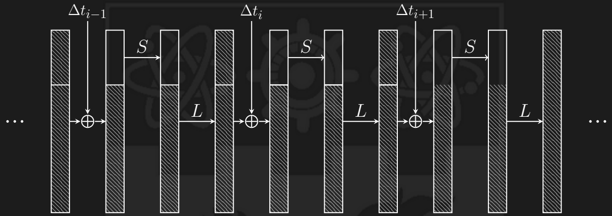
The MALICIOUS framework



The MALICIOUS construction : a framework to build a key-alternating tweakable block cipher such that :

- The non-linear layer of each round function f_i is partial.
- The sub-tweaks are obtained from the original tweak T through an XOF, independently of the key input : $\texttt{XOF}(T) = t_0 || t_1 \cdots || t_r$.

The backdoor in MALICIOUS



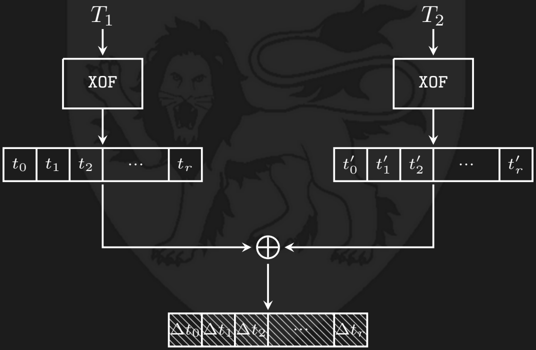
Related-tweak diff. characteristic with probability 1

- The difference on the non-linear part of the round function is always cancelled by the sub-tweak addition.
- The differential characteristic is built from **a secret tweak pair**, we call it the malicious tweak pair.
- Attacking with the backdoor is under the chosen-tweak scenario.

How to build the backdoor?

Step 1 : computing the sub-tweak differences

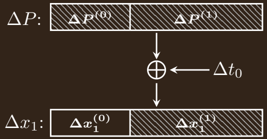
- Choose a malicious pair of tweaks (T_1, T_2) and keep it secret.
- Compute the corresponding sub-tweak differences $\Delta t_0, \cdots, \Delta t_r$ through the XOF.



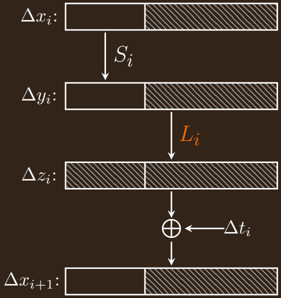
How to build the backdoor?

Step 2 : building the linear layers L_i

- Select a plaintext difference ΔP, satisfying $\Delta P^{(0)} = \Delta t_0^{(0)}$ such that $\Delta x_1^{(0)} = 0$.



- Generate the differential characteristic round by round, by selecting an appropriate linear layer L_i each round, satisfying $L_i(\Delta y_i)^{(0)} = \Delta t_i^{(0)}$, so that $\Delta x_{i+1}^{(0)} = 0$.

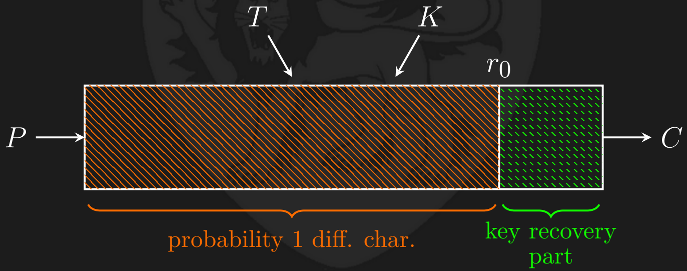


Note : it is possible to embed multiple diff. characteristics.

How to use the backdoor?

Using the backdoor

- use sufficiently many rounds so that the cipher will resist state-of-the-art cryptanalysis (say r_0 rounds)
- as explained, build the cipher while embedding a probability 1 diff. characteristic over the r_0 rounds
- add a few more rounds on top of r_0, to allow for the attacker to apply the **key recovery part**.



Outline

1. Backdoors in Symmetric-Key Cryptography
2. Previous Backdoors Attempts
   - ▷ In the wild
   - ▷ Academic work
3. The MALICIOUS Framework
   - ▷ Preliminaries
   - ▷ The MALICIOUS Framework
   - ▷ The MALICIOUS Security
4. LowMC-M : A backdoored TBC Variant of LowMC
5. Future Directions

The MALICIOUS Backdoor Security

We want to analyse :

- ▷ the complexity for the attacker to find a tweak pair that activates the backdoor
- ▷ whether other weaknesses have been introduced when adding the backdoor

The MALICIOUS Backdoor Security

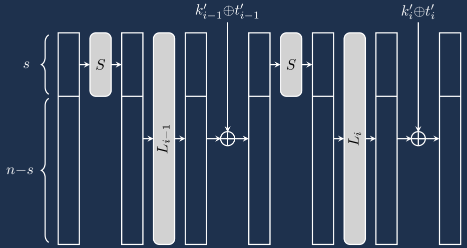
We want to analyse :

- ▷ the complexity for the attacker to find a tweak pair that activates the backdoor
- ▷ whether other weaknesses have been introduced when adding the backdoor

Equivalent Representation of MALICIOUS

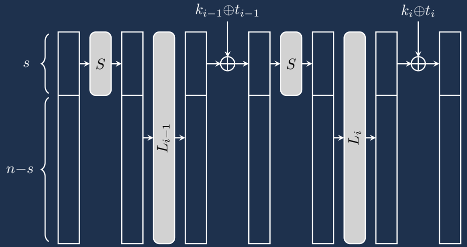
$n * r$

sub-tweak bits



$n + s * (r - 1)$

sub-tweak bits



Target-difference resistance

Definition : Target-difference resistance

A hash function H is **target-difference resistant** if it is hard to find two inputs x and y such that $H(x) \oplus H(y) = \Delta$, where Δ is a non-zero constant.

The generic attack complexity for target-difference resistance is the same as the classical collision resistance (where $\Delta = 0$), that is the birthday bound $O(2^{n/2})$.

The backdoor is protected by the XOF

Finding the malicious tweak pair (T_1, T_2) is difficult even if the differential characteristic is public known. The complexity is the target-difference resistance of the XOF used in the framework.

$$\texttt{XOF}(T_1) \oplus \texttt{XOF}(T_2) = \Delta t_0 || \Delta t_1 || \cdots || \Delta t_r$$

We have $n + s * (r - 1)$ bits of subtweak material, so an attacker has to pay $2^{(n+s*(r-1))/2}$ complexity.

Finding a malicious tweak pair costs $2^{(n+s*(r-1))/2}$ complexity.

If you want at least k-bit security, you need at least $r = (2k - n)/s + 1$ rounds (with $k = n = 128$, we have $r \geq 128/s + 1$).

The MALICIOUS Backdoor Security

We want to analyse :

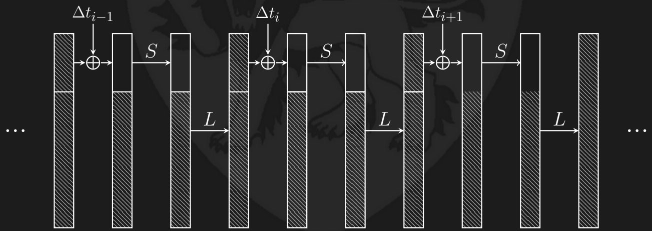
- the complexity for the attacker to find a tweak pair that activates the backdoor
- whether other weaknesses have been introduced when adding the backdoor

Other probability 1 differential paths - Beyne and Li (ePrint)

Other probability 1 differential paths

There are **other subtweak differences** that could lead to other probability 1 differential paths on r rounds (that is no active Sbox). There is actually one such path for each of the possible 2^n plaintext difference and each possible 2^{n-s} internal state difference value truncated to $(n-s)$ bits.

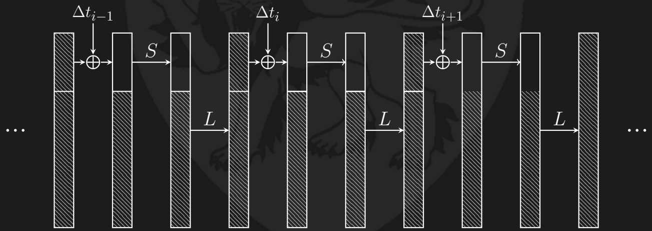
$$\mathrm{XOF}(T'_1) \oplus \mathrm{XOF}(T'_2) = \Delta t'_0 || \Delta t'_1 || \cdots || \Delta t'_r$$



Other probability 1 differential paths - Beyne and Li (ePrint)

There are 2^{2n-s} of them, so on a $n + s * (r-1)$ bits hash, by (optimistically) targeting all of them at the same time the complexity is $2^{(s*r-n)/2}$.

If you want at least k-bit security, you need at least $r = (2k+n)/s$ rounds (with $k = n = 128$, we have $r \geq 384/s$).

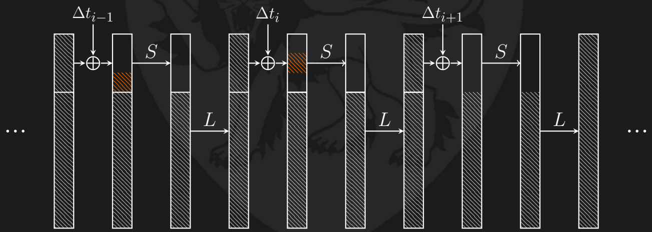


Other related-tweak differential paths

Other related-tweak differential paths

We could consider differential paths on r rounds that have **a bit more than 0 active Sbox**, which leads to other interesting subtweak differences for the attacker

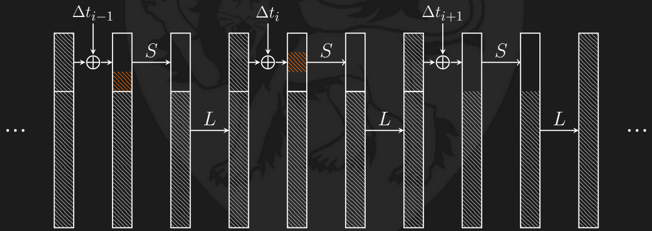
$$\mathtt{XOF}(T'_1) \oplus \mathtt{XOF}(T'_2) = \Delta t'_0 || \Delta t'_1 || \cdots || \Delta t'_r$$



Other related-tweak differential paths

Indeed, 1, 2, 3, ... active Sboxes will still produce a differential path that is exploitable for attacks (yet for a higher complexity than the original one). For x active Sboxes, you have an extra factor of $\binom{r*s/3}{x} * 2^{5x}$ more paths (with 3-bit Sbox).

If you want at least k-bit security, you need at least $r = (3k + n)/s$ rounds (with $k = n = 128$, we have $r \geq 512/s$).



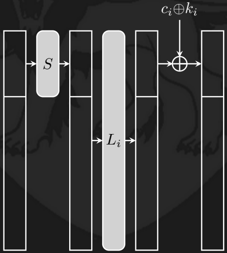
Outline

1. Backdoors in Symmetric-Key Cryptography
2. Previous Backdoors Attempts
   - ▷ In the wild
   - ▷ Academic work
3. The MALICIOUS Framework
   - ▷ Preliminaries
   - ▷ The MALICIOUS Framework
   - ▷ The MALICIOUS Security
4. LowMC-M : A backdoored TBC Variant of LowMC
5. Future Directions

LowMC- Albrecht *et al.* - EUROCRYPT 2015

LowMC block cipher (Albrecht *et al.* EUROCRYPT'15)

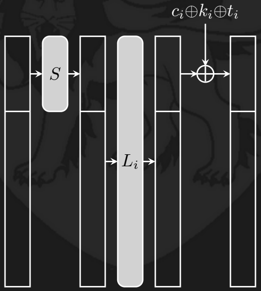
- **goal :** to minimize multiplicative size and depth
- **applications** : practical instantiations of MPC and FHE, where linear operations are free (compared to non-linear)
- **approach :** a partial non-linear SPN, with 3-bit Sbox, and linear layers being random $n \times n$ invertible binary matrices
- several parameter sets available, depending on the security aimed, the non-linear layer size, etc.



LowMC-M : a backdoored TBC variant of LowMC

The LowMC-M tweakable block cipher :

- a tweakable block cipher, not just block cipher
- round function and key schedule same as LowMC (we add the tweak layer)
- the linear layer L_i matrices are not chosen randomly, but have to be built to embed the backdoor
- the tweak schedule uses SHAKE128 or SHAKE256 as XOF



Backdoor security of LowMC-M (undetectability)

Undetectability (✓) :

The attacker is unable to detect whether an instance of LowMC-M is embedded with a backdoor or not.

Argument :

You can't distinguish between LowMC-M and a similar cipher where the malicious subtweak difference has been chosen randomly and not via the XOF (that would thus contain no exploitable backdoor)

Backdoor security of `LowMC-M` (undiscoverability)

Undiscoverability (✓) :

It is computationally difficult for the attacker to recover the backdoors.

Argument :

As mentioned in the backdoor security analysis of the MALICIOUS framework, as long as r is large enough, the target-difference resistance of the `XOF` makes sure that the attacker can't recover the malicious tweak pair or an equivalent one.

Backdoor security of `LowMC-M` (untraceability)

Untraceability (✗) :

If the backdoor is used in an attack, it will reveal the information of the backdoor (since it is chosen-tweak chosen-plaintext attack).

Argument :

An attacker simply checks all the pairs of tweaks he sees with a birthday-like search. One of them will lead to zero difference in all Sboxes in the first r_0 rounds : this is the malicious tweak pair. Once he has the backdoor, he can use it as well.

Classical security of LowMC-M

What about the classical security of LowMC-M?

Attacks without using the tweak

The security of LowMC-M can be reduced to the security of LowMC which remains strong currently

- Without considering the tweak, LowMC-M is an equivalent representation of LowMC.
- Even if a LowMC-M instance is backdoored, we show that its customized linear layer matrices can be considered as independently and randomly chosen from the view of the attacker.

Cryptanalysis of LowMC−M and proposal of LowMC−M v2 :

For LowMC−M v1, we originally used the same number of rounds as LowMC, which was a mistake since there is extra tweak material that the attacker can use.

Cryptanalysis results of LowMC−M appeared recently :

- Beyne and Li (ePrint 2020) : probability 1 differential paths on $(2k+n)/s$ rounds, differential-linear (n/s more rounds covered) and key recovery (k/s more rounds)
- Liu *et al.* (ePrint 2020) : algebraic technique to improve difference enumeration technique (also improves attacks on LowMC)

Our solution (LowMC−M v2) :

Simply increase the number of rounds of LowMC−M accordingly. Our general MALICIOUS framework still holds.

LowMC-M v2 parameter sets

Various LowMC-M v2 instances :

block size n	non-linear s	key size k	data d	rounds r	#diff. a	XOF
128	3	128	64	294	43	SHAKE128
	6	128	64	147	21	SHAKE128
	9	128	64	99	14	SHAKE128
	30	128	64	32	5	SHAKE128
	90	128	64	17	2	SHAKE128
256	3	256	64	555	85	SHAKE256
	9	256	64	186	28	SHAKE256
	60	256	64	30	5	SHAKE256
	120	256	64	19	3	SHAKE256

Outline

1. Backdoors in Symmetric-Key Cryptography
2. Previous Backdoors Attempts
   - In the wild
   - Academic work
3. The MALICIOUS Framework
   - Preliminaries
   - The MALICIOUS Framework
   - The MALICIOUS Security
4. `LowMC-M` : A backdoored TBC Variant of `LowMC`
5. **Future Directions**

Open Problems

Open Problems :

- Can we use the framework to build other backdoored cryptographic primitives ? Such as hash functions, MACs, etc. ?
- Is it possible to base the backdoor on other, more sophisticated cryptanalysis techniques ? Boomerang ? Impossible differential attacks ? etc.
- **1 million $ question :** can we create a backdoored block cipher that is **untraceable** ?

On untraceability

Obtaining untraceability ?

Maybe untraceability can be obtained by **hiding the backdoor queries in multiple unrelated queries** ?

Ex : when using `LowMC-M`, instead of just querying the malicious tweak pair, query a lot (2^x) of other useless random tweak queries. The attacker would have to check all 2^{2x} pairs of tweak to find the malicious one.

Problem : birthday attack can be used to find the malicious pair, the attacker just checks that difference is 0 in the Sboxes

Solution ?

- ▷ build a backdoor that uses quadruplets (or more) instead of pairs of tweaks
- ▷ build a backdoor whose existence can't be checked linearly

Thank you !