

# By-design Backdooring of Encryption System Can We Trust Foreign Encryption Algorithms?

Arnaud Bannier & Eric Filiol (speaker)

filiol@esiea.fr

ESIEA

Operational Cryptology and Virology Lab  $(C + V)^O$



# Agenda

- 1 Introduction: what is the issue?
- 2 History of known (and less known) backdoored algorithms
- 3 Description of BEA-1
  - Theoretical Background
  - BEA-1 Presentation and Details
- 4 BEA-1 Cryptanalysis
- 5 Conclusion and Future Work



# Summary of the talk

- 1 Introduction: what is the issue?
- 2 History of known (and less known) backdoored algorithms
- 3 Description of BEA-1
- 4 BEA-1 Cryptanalysis
- 5 Conclusion and Future Work



# Key Question

Just imagine that if unconditionally secure systems (computer, information security) would be possible (theoretically AND practically), would it be desirable to export them?

- The answer is NO due to
  - National Security Issues (Intelligence, Defense, Police, Justice...)
  - Strategic dominance, information assurance...
  - Economic warfare & dominance (since 1989)



# From Export Control to Domestic Control

[Get Latest Articles to Your Inbox](#) [Subscribe Now!](#)

[Home](#) [Hacking](#) [Tech](#) [Deals](#) [Cyber Attacks](#) [Malware](#) [Spying](#)

 **The Hacker News**™  
Security in a serious way

 +1,698,300

 421,550

 2,070,880

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

Monday, March 27, 2017 Mohit Kumar



**London Terror Attack**  
**UK Demands Encryption Backdoor**

The government has once again started asking for backdoor in encrypted services, arguing that it can not give enough security to its citizens because the terrorists are using encrypted apps to communicate and plot an attack.

Following last week's terrorist attack in London, the UK government is accusing technology firms to give terrorists "a place to hide," saying Intelligence agencies must have access to encrypted messaging applications such as WhatsApp to prevent such attacks.

### POPULAR STORIES



Cryptocurrency Mining Scripts Now Run Even After You Close Your Browser



macOS High Sierra Bug Lets Anyone Gain Root Access Without a Password



Google Detects Android Spyware That Spies On WhatsApp, Skype Calls



Learn Ethical Hacking Online: A to Z Training Courses



Hackers Exploit Recently Disclosed Microsoft Office Bug to Backdoor PCs



HP Silently Installs Telemetry



# From Export Control to Domestic Control

Log in | Sign up | Forums

The Register®  
Biting the hand that feeds IT

TwitterFacebookLinkedIn

A


[DATA CENTRE](#) [SOFTWARE](#) [SECURITY](#) [TRANSFORMATION](#) [DEVOPS](#) [BUSINESS](#) [PERSONAL TECH](#) [SCIENCE](#) [EMERGENT TECH](#) [BOOTNOTES](#) [Q](#)

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


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

(ESIEA - (C + V)<sup>O</sup> lab)

Black Hat Europe 2017

5 / 44

# From Export Control to Domestic Control




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



by MÁR HÁSSON MAACK — 8 months ago in EUROPE



# From Export Control to Domestic Control



Search Techdirt

Search

TECHDIRT | WIRELESS NEWS | CASE STUDIES | NET NEUTRALITY | FREE SPEECH | **TECHDIRT DEALS!**

Preferences Register Sign In

Main Submit a Story RSS

 **PODCAST** Techdirt - Tom Wheeler Reacts To Trump's FCC



<< Top EU Privacy Campaigner Says He Wants Lots... Ajit Pai Doesn't Want You Talking About... >>



## German Government Official Wants Backdoors In Every Device Connected To The Internet

from the *bring-back-the-old-Germany-we-know-and-hate!* dept

The US Department of Justice is reviving its anti-encryption arguments despite not being given any signals from the administration or Congress that undermining encryption is something either entity desires. The same thing is happening in Germany, with Interior Secretary Thomas de Maizière continuing an anti-encryption crusade very few German government officials seem interested in joining.

The key difference in de Maizière's push is that he isn't limiting potential backdoors to cell phones. He appears to believe anything connected to the internet should be backdoored... possibly even the cars German citizens drive. (h/t Riana Pfefferkorn)

*The RedaktionsNetzwerk Deutschland (RND) reported that Thomas de Maizière had written up a draft proposal for the Interior minister conference, taking place next week in Leipzig, which he has called "the legal duty for third parties to allow for secret surveillance."*

According to the RND, the proposal would "dramatically extend" the state's powers to spy on its citizens.

And it's not just backdoors being suggested. De Maizière wants all electronics to be law enforcement-complicit. All things - especially those connected to the Internet - should be constructed with government access in mind.

*For example, the modern locking systems on cars are so intelligent that they even warn a driver if their car is shaken a little bit. De Maizière wants the new law to ensure that these alerts would not be sent out to a car owner if the police determined it to be justified by their investigation.*

De Maizière wants the government to be able to intercept and block notifications sent from cars to the people that own them. But it's far more than smarter cars being compromised on behalf of the government. If de Maizière gets his way, it will be every connected device everywhere.

De Maizière also wants the security services to have the ability to spy on any device connected to the internet. Tech companies would have to give the state "back door" access to private tablets and computers, and even to smart TVs and

Follow Techdirt

Insider Shop - Show Your Support!

READ POSTS EARLY. JOIN THE INSIDER CHAT & MORE

 **INSIDER SHOP** 

Advertisement

Report this ad | Hide Techdirt ads

Essential Reading

Hot Topics

**5.8** Ajit Pai Doesn't Want You Talking About Court Ruling That Undermines His Bogus Claim That The FTC Will Protect Consumers

**5.3** Security Researcher Held In Jail For 8 Months Because He Wrote An Angry Blog Post, Released For Now

**5.3** Ajit Pai Attacked Hollywood & Silicon Valley Because Even Republicans Are Against His Net Neutrality Plan

New To Techdirt?

Explore some core concepts:

The Future Of Music Business Models (And Those Who Are Already There)

An Economic Explanation For Why DRM Cannot Open Up New Business Model Opportunities

Infinity Is Your Friend In Economics

read all >

Techdirt Deals

Report this ad | Hide Techdirt ads

Techdirt Insider Chat

something real bad before we try to regulate them





# 70 Years of Control

- Since the end of WWII, cryptology is under control. This control has never weakened
- UKUSA (5 eyes)/9 eyes/14 eyes SIGINT Seniors Europe...
- International Traffic in Arms regulations (ITAR, part 121) and subsequent regulations (Wassenaar...)
  - If cryptology is allowed/free of use, then it is under control
  - 1997 is a key year (withdrawn from ITAR) and early 2000s in Europe: the rise of connected world. The control will be far easier (computer, OS, network...)
- Cryptology is the most critical part in security: who is controlling cryptology, is controlling everything



# The Wassenaar Agreement

- Almost all G-20 countries have a national regulation regarding cryptology (use/import/export) or at least have signed an international regulation
- <http://rechten.uvt.nl/koops/cryptolaw/>
- <http://www.wassenaar.org/> - 42 members
- Cryptology is listed in part 5b  $\Rightarrow$  exporting encryption algorithms with key size greater than 56 bits (symmetric cryptology) is subject to export control!
- As a consequence, the world diffusion of encryption algorithms whose key size  $\geq 128$  bits is a clear violation of the Wassenaar agreement ... unless some sort of control has been organized/enforced.



# What does “Operational cryptanalysis” Means?

- Intelligence/operational point of view: **really** breaking an encryption system means
  - Accessing the plaintext in a time shorter than the life of the information (regarding its operational value)
  - Practically speaking: a matter of hours (supercomputing time is horribly expensive)
  - With a reduced amount of encrypted data (a few Kb to a few Mb)
  - Must be played a large number of times (a clever enemy changes the key very often, encrypted traffic explodes)
- Academic attacks have just... an academic interest!



# Control Techniques

- The control techniques depend on the target context/environment

| Type                         | Data       | NSA Programs  | Techniques   | Examples   |
|------------------------------|------------|---|--|--|
| Connected                    | Plaintext  | PRISM, Xkeyscore...   | Data collection, wiretapping, eavesdropping, agreements with industry/providers....  | Google, Facebook, Apple, Microsoft (including Skype)...                        |
|                              | Ciphertext | Bullrun/Edgehill...   | Malware, 0-day exploitation, random generator control, security standards control, controlling CAs, bugging software, applied cryptanalysis...                 | Heartbleed, RSA, Google/ANSSI, Mail.ru, Alibaba...                             |
| Connected by private network | Ciphertext | Cottonmouth, Godsurge, TOR attack, Quantum, Foxacid, Firework, Bulldozer... | Malware, 0-day exploitation, random generator control, controlling CAs, security standards control, bugging software, hardware bugs, mathematical trapdoors... | TOR network, Gasprom, Petrobras, French MFA, Aeroflot, Total. Airbus, SWIFT... |
| Non-connected (offline)      | Ciphertext | TAO, still unknown projects???  | Tempest techniques, mathematical backdoors, hardware bugging, Humint   | Hans Buehler Case (1995). Gov, MIL, Sensitive companies                        |

# Trapdoor vs Backdoors

- Trapdoors are an intended and necessary feature in asymmetric cryptology
- Backdoors are an undesirable feature
- Implementation backdoors
  - Key escrowing, key management and key distribution protocols weaknesses (refer to recent CIA leak)
  - So called OS/software (recurrent) vulnerabilities (invoking developers incompetence is much clever)
  - Hackers are likely to find and use them as well



# Mathematical Backdoors

- Key Principle

- Put a secret flaw at the design level while the algorithm remains public
- Finding the backdoor must be an intractable problem while exploiting it must be “easy”

- Two kind of backdoors

- “Natural weakness” known by the tester/certifier (e.g NSA case with differential cryptanalysis)
- Intended weakness put by the encryption algorithm designer

- Extremely few open and public research in this area

- Known existence of NSA and GCHQ research programs

- Sovereignty issue: can we trust foreign encryption algorithms?



# Aim of our Research

- Try to answer to the key question
  - *“How easy and feasible is it to design and to insert backdoors (at the mathematical level) in encryption algorithms?”*



# Aim of our Research

- Try to answer to the key question
  - *“How easy and feasible is it to design and to insert backdoors (at the mathematical level) in encryption algorithms?”*
- Explore the different possible approaches
  - The present work is a first step
  - We consider a particular case of backdoors here (linear partition of the data spaces)





# Aim of our Research

- Try to answer to the key question
  - *“How easy and feasible is it to design and to insert backdoors (at the mathematical level) in encryption algorithms?”*
- Explore the different possible approaches
  - The present work is a first step
  - We consider a particular case of backdoors here (linear partition of the data spaces)
- For more details on technical aspects, please refer to our free book
  - Available on <https://www.intechopen.com/books/partition-based-trapdoor-ciphers>



# Summary of the talk

- 1 Introduction: what is the issue?
- 2 History of known (and less known) backdoored algorithms
- 3 Description of BEA-1
- 4 BEA-1 Cryptanalysis
- 5 Conclusion and Future Work

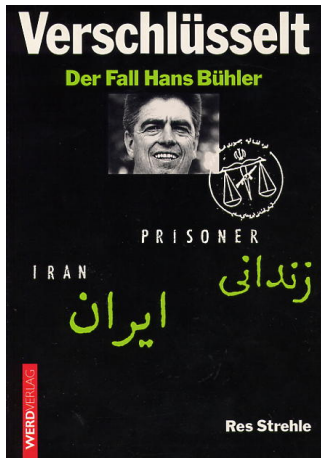


# Cryptography Industry After WWII til the 90s

- In Switzerland, Crypto AG/Gretag hold more than 90 % of the world market (since 1945)
- Almost all countries/organizations (120 in 1995) were buying cryptomachines for gvt, mil, diplomatic, economic needs except a very few (USA, France, UK...).
- **1995** The Hans Buehler case changed the cryptologic face of the (cryptographic) world.



# The Hans Buelher Case



# The Hans Buelher Case

- Crypto AG's top marketing representative arrested in Teheran in 1992.
- Leaks in the Press (Berlin Club bombing, Chapur Bakhtiar assassination in Paris) by Gov. officials gave hints to Iranian govt that cryptography was probably backdoored.
- 9 months in Iranian jails
- Reveals the scandal: NSA, BND and others have infiltrated Crypto AG, Gretag and other companies to put trapdoors in export versions of crypto machines systematically
- The UKUSA/ANZUS countries were able to read openly most of the world encrypted traffic during nearly 50 years
- Exploited the fact that encryption algorithms were not public!



## SIGHTINGS

### Greatest Intel Coup Of The Century? The NSA's Crypto AG

By Wayne Madsen  
Covert Action Quarterly 63, 30 Jan 1999,  
From Stig Agermose (stig.agermose@get2net.dk)  
2-7-99

Crypto AG: The NSA's Trojan Whore?

FOR AT LEAST HALF A CENTURY, THE US HAS BEEN  
INTERCEPTING AND DECRYPTING THE TOP SECRET  
DOCUMENTS OF MOST OF THE WORLD'S GOVERNMENTS

"...allows the US to play high-stakes diplomatic poker with a mirror  
behind everyone else's back."

It may be the greatest intelligence scam of the century: For decades,  
the US has routinely intercepted and deciphered top secret encrypted  
messages of 120 countries. These nations had bought the world's  
most sophisticated and supposedly secure commercial encryption  
technology from Crypto AG, a Swiss company that staked its  
reputation and the security concerns of its clients on its neutrality.



# The Crypto AG Case

Once the cipher machines were rigged to include the secret decryption key, the BND and NSA codebreakers could use the transmitted key to read any message sent by Crypto AG's 120 country customers. One previous Crypto AG employee contends that all developmental Crypto AG equipment had to be sent for approval to the NSA and to the German Central Cipher Bureau (Zentralstelle für Chiffrierung [ZfCH]), now the Federal Information Security Agency (Bundesamt für Sicherheit in der Informationstechnik [BSI] which is also Department 62 of the BND) in Bad Godesberg, near Bonn.

In other cases, Crypto AG was apparently forced to market encryption equipment manufactured in the US, sent to Crypto, and passed off as Swiss equipment. In the 1970s, as Crypto was moving from electro-mechanical to computerized crypto units, a former Crypto AG engineer in Switzerland inspected one of the first prototype computerized machines sent from the US. He remarked that since the code could be easily broken, he found the machine useless. But when he told his superiors that he could improve the encryption process if he was given access to the mathematical functions, two US cryptographic "experts" refused to disclose the information.

According to a confidential Crypto AG memorandum, one of the NSA "experts" may have been Nora L. Mackabee, an NSA cryptographer who is now retired on a horse farm in Maryland along with her husband Lester, another retired NSA employee. Between August 19 and 20, 1975, three Crypto AG engineers huddled with Mackabee (identified as representing "IA" - most likely "intelligence agency") along with three Motorola engineers and one other American, Herb Frank. One Motorola engineer recalled that Frank was probably from another US intelligence agency based in northern Virginia but described him as a non-technical person who seemed to be making the administrative arrangements for Mackabee.

Crypto engineer Juerg Spoerndli, who was responsible for designing the firm's encryption equipment, had heard from older engineers



# The Crypto AG Case

Crypto engineer Juerg Spoerndli, who was responsible for designing the firm's encryption equipment, had heard from older engineers about the visits in earlier years by mysterious Americans. He concluded that NSA was ordering the design changes through German intermediaries. He confirmed the manipulation and admitted that in the late 1970s, he was "ordered to change algorithms under mysterious circumstances"<sup>25</sup> to weaken his cipher units.





# The Crypto AG Case

During the sensitive Anglo-Irish negotiations of 1985, the NSA's British counterpart, the GCHQ, was able to decipher the coded diplomatic traffic being sent between the Irish embassy in London and the Irish Foreign Ministry in Dublin. It was reported in the Irish press that Dublin had purchased a cryptographic system from Crypto AG worth more than a million Irish pounds. It was also reported that the NSA routinely monitored and deciphered the Irish diplomatic messages. Later, during the Falklands War, British GCHQ operators were able to decrypt classified Argentine message traffic because the Argentines were using rigged Crypto AG cipher machines. Former British Foreign Office minister Ted Rowlands publicly stated that GCHQ had penetrated Argentine diplomatic codes.



# The Crypto AG Case

The NSA program also likely extends to companies in NATO and pro-US countries which have close relationships with GCHQ, NSA, and the BND. Even neutral countries' firms are not off-limits to NSA manipulations. A former Crypto AG employee confirmed that high-level US officials approached neutral European countries and argued that their cooperation was essential to the Cold War struggle against the Soviets. The NSA allegedly received support from cryptographic companies Crypto AG and Gretag AG in Switzerland, Transvertex in Sweden, Nokia in Finland, and even newly-privatized firms in post-Communist Hungary.

In 1970, according to a secret German BND intelligence paper, supplied to the author, the Germans planned to "fuse" the operations of three cryptographic firms-Crypto AG, Grattner AG (another Swiss cipher firm), and Ericsson of Sweden. Securocrats often turn to the boogeyman of "rogue" nations in order to justify the expense and ethical necessity of eavesdropping on all forms of international communication, but in reality many intercepts involve messages by neutral or allied nations.



# Backdoor Example...Among Many Others

- Example drawn from a serie of a cryptomachines sold in early 90s and rigged by the NSA
- Base key  $K$  (changed every day, week. . . ) and a message key  $K_m$
- A Boolean function defined over  $\mathbb{F}_2^n$  which is not correlation-immune is used as a critical primitive
- How to trap the Boolean function:
  - Use a message key  $K_m = (k_m^0, k_m^1, \dots, k_m^i, \dots, k_m^{2^{n-1}-1})$  of size  $2^{n-1}$
  - Xor it by half to the Boolean function truth table

$$\forall x_i \in [0, 2^{n-1} - 1], \quad f(x_i) \leftarrow f(x_i) \oplus k_m^i$$

$$\text{and } f(x_i + 2^{n-1}) \leftarrow f(x_i + 2^{n-1}) \oplus k_m^i$$

- The Boolean function remains highly correlated to a few of its input
- Many other tricky variants possible

# The Bullrun Program

- Goal: bypass operationally any cryptology protection
  - Tampering with national standards (NIST is specifically mentioned) to promote weak, or otherwise vulnerable cryptography (e.g Dual\_EC\_DRBG, see further)
  - Influencing standards committees to weaken protocols (or influencing to bar strong algorithms [Gost])
  - Working with hardware and software vendors to weaken encryption and random number generators
  - Identifying and cracking vulnerable keys
  - Establishing a Human Intelligence division to infiltrate the global telecommunications industry
  - ...
- Annual budget: 250 millions \$ per year.



# The Bullrun Program

theguardian

[News](#) | [Sport](#) | [Comment](#) | [Culture](#) | [Business](#) | [Money](#) | [Life & style](#) | [Travel](#) | [Environment](#) | [T](#)

[News](#) > [World news](#) > [The NSA files](#)

Series: Glenn Greenwald on security and liberty

[Previous](#) | [Next](#) | [Index](#)

## Revealed: how US and UK spy agencies defeat internet privacy and security

- NSA and GCHQ unlock encryption used to protect emails, banking and medical records
- \$250m-a-year US program works covertly with tech companies to insert weaknesses into products
- Security experts say programs 'undermine the fabric of the internet'

• [Q&A: submit your questions for our privacy experts](#)

James Ball, Julian Borger and Glenn Greenwald

Guardian Weekly, Friday 6 September 2013

 [Jump to comments \(4142\)](#)



[Article history](#)



[The NSA Files: Decoded](#)

- Dual Elliptic Curve Deterministic Random Bit Generator (Dual\_EC\_DRBG). Used to generate random keys. ISO and ANSI standards
- Used in many environments (Blackberry, SSL/TLS)
- Fixed choice of constants P and Q makes most of the backdoor (see <http://blog.cryptographyengineering.com/2013/09/the-many-flaws-of-dualecdrbg.html>)
- Shumow-Ferguson Crypto 2007
- Nobody knows where Dual\_EC\_DRBG parameters came from
- In SSL/TLS, NSA can recover the pre-master secret (RSA handshake) easily



## A.1 Constants for the Dual\_EC\_DRBG

The **Dual\_EC\_DRBG** requires the specifications of an elliptic curve and two points on the elliptic curve. One of the following NIST **approved** curves with associated points **shall** be used in applications requiring certification under [FIPS 140]. More details about these curves may be found in [FIPS 186]. If alternative points are desired, they **shall** be generated as specified in Appendix A.2.

Each of following curves is given by the equation:

$$y^2 = x^3 - 3x + b \pmod{p}$$

Notation:

$p$  - Order of the field  $F_p$ , given in decimal

$n$  - Order of the Elliptic Curve Group, in decimal .

$a$  - (-3) in the above equation

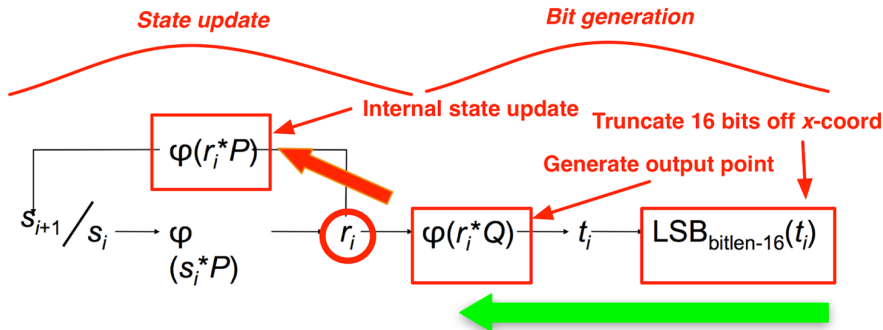
$b$  - Coefficient above

The  $x$  and  $y$  coordinates of the base point, i.e., generator  $G$ , are the same as for the point  $P$ .

### A.1.1 Curve P-256

```
p = 11579208921035624876269744694940757353008614\
3415290314195533631308867097853951
n = 11579208921035624876269744694940757352999695\
522413576034242259061068512044369
b = 5ac635d8 aa3a93e7 b3ebbd55 769886bc 651d06b0 cc53b0f6 3bce3c3e
27d2604b
Px = 6b17d1f2 e12c4247 f8bce6e5 63a440f2 77037d81 2deb33a0
f4a13945 d898c296
Py = 4fe342e2 fe1a7f9b 8ee7eb4a 7c0f9e16 2bce3357 6b315ece
cbb64068 37bf51f5
Qx = c97445f4 5cdef9f0 d3e05e1e 585fc297 235b82b5 be8ff3ef
ca67c598 52018192
Qy = b28ef557 ba31dfcb dd21ac46 e2a91e3c 304f44cb 87058ada
2cb81515 1e610046
```







# Dual\_EC\_DRBG RSA B-Safe: Timeline

- 2004 - RSA makes Dual\_EC\_DRBG the default CSPRNG in BSAFE
- 2005 - ISO/IEC 18031:2005 and NIST SP 800-90A include Dual\_EC\_DRBG.
- 2006 2007 Works suggesting the existence of a NSA backdoor (K. Gjoosten, Berry Schoenmakers and Andrey Sidorenko, Shumow/Fergusson)
- June 2006 - NIST SP 800-90A (final) is published, includes Dual\_EC\_DRBG (defects pointed out by Kristian Gjoosten and al. not fixed).
- June/Sep. 2013 Snowden leak about Bullrun and Dual\_EC\_DRBG
- 19 Sep. 2013 - RSA Security advises its customers to stop using Dual\_EC\_DRBG
- Dec. 2013 - Reuters reports this is a result of a secret \$10 million deal with NSA
- April 2014 - NIST removes Dual\_EC\_DRBG as a cryptographic algorithm, recommending *"that current users of Dual\_EC\_DRBG transition to one of the three remaining approved algorithms as quickly as possible"*

- NIST standard meant that you could only get the FIPS 140-2 validation (Cryptographic Module Validation Program) only if you used the original compromised  $P$  and  $Q$  values
- FIPS 140-2 statistical test suite (now NIST STS) are THE *de facto* world standard for cryptography statistical evaluation/validation
- Passing successfully the tests does not mean your generator is secure
- Can we still trust FIPS 140-2 tests?
- Issue of statistical test simulability (Filiol, 2006): *“if your statistical tests are known, they can be simulated to bypass them”*
- Cryptography statistical validation should use a secret national process/set of tests



# NSA's Simon & Speck

- June 2013: public release by the NSA of Speck and Simon, two NSA's families of encryption algorithms (block ciphers)
- Since 2014, efforts by the NSA to standardise the Simon and Speck ciphers at ISO
- Sept. 2017, ISO rejects Simon and Speck standardisation under the pressure of experts from the academic community and from ISO



Discover Thomson Reuters \*\*\*



Distrustful U.S. allies force spy agency to back down in encryption fight



Special Report: "Treachorous shenanigans" - The Inside story of Mugabe's downfall



Argentina faces 'hope and ho submarine search

#CYBER RISK SEPTEMBER 21, 2017 / 7:03 AM / 2 MONTHS AGO

## Distrustful U.S. allies force spy agency to back down in encryption fight

Joseph Menn

8 MIN READ



SAN FRANCISCO (Reuters) - An international group of cryptography experts has forced the U.S. National Security Agency to back down over two data encryption techniques it wanted set as global industry standards, reflecting deep mistrust among close U.S. allies.



INDUSTRY / SECURITY

## NSA Tried to Push Global Encryption Standards "Because It Knew How to Break Them"



By Rafa Shaikh

Sep 22, 2017

14  
SHARES

SHARE

TWEET

SUBMIT



*No one trusts NSA any longer, not even its allies! Suspecting backdoors, US allies push NSA to step back from encryption fight*

The cybersecurity world continues to mistrust the [National Security Agency of the United States](#). According to a latest Reuters report, an international group of cryptography experts from the country's closest allies has forced the NSA to back down over two encryption techniques that the agency wanted to turn into global industry standards.

### US allies accuse NSA of manipulating global cryptography standards

The agency is back in the bad books again after it was discovered by the US allies that the intelligence agency was trying to manipulate international encryption standards. Citing interviews and emails, Reuters reported:



# Summary of the talk

- 1 Introduction: what is the issue?
- 2 History of known (and less known) backdoored algorithms
- 3 Description of BEA-1
  - Theoretical Background
  - BEA-1 Presentation and Details
- 4 BEA-1 Cryptanalysis
- 5 Conclusion and Future Work



- Start from an algorithm with backdoor  $E_{backdoor}$ 
  - In BEA-1, the backdoor is essentially made of “secret” S-boxes



# Design Transformation

- Start from an algorithm with backdoor  $E_{backdoor}$ 
  - In BEA-1, the backdoor is essentially made of “secret” S-boxes
- Use a one-way transformation  $S$ 
  - Computing  $E = S(E_{backdoor})$  is computationally easy (here  $E$  is BEA-1)
  - Computing  $E_{backdoor}$  from  $E$  is computationally intractable unless you know some secret information  $S'$  such that  $S' \circ S = \text{Identity}$ .
  - $E$  exhibits all desirable cryptographic properties





# Design Transformation

- Start from an algorithm with backdoor  $E_{backdoor}$ 
  - In BEA-1, the backdoor is essentially made of “secret” S-boxes
- Use a one-way transformation  $S$ 
  - Computing  $E = S(E_{backdoor})$  is computationally easy (here  $E$  is BEA-1)
  - Computing  $E_{backdoor}$  from  $E$  is computationally intractable unless you know some secret information  $S'$  such that  $S' \circ S = \text{Identity}$ .
  - $E$  exhibits all desirable cryptographic properties
- BEA-1 secret S-Boxes  $\leftrightarrow$  BEA-1 public S-Boxes.



# Partition-based Trapdoors

- Based on our theoretical work (Bannier, Bodin & Filiol, 2016; Bannier & Filiol, 2017)
  - Generalization of Paterson's work (1999)



# Partition-based Trapdoors

- Based on our theoretical work (Bannier, Bodin & Filiol, 2016; Bannier & Filiol, 2017)
  - Generalization of Paterson's work (1999)
- BEA-1 is inspired from the *Advanced Encryption Standard* (AES)
  - BEA-1 is a Substitution-Permutation Network (SPN)
  - BEA-1 stands for *Backdoored Encryption Algorithm* version 1



## Definition (Linear Partition)

A partition of  $\mathbb{F}_2^n$  made up of all the cosets of a linear subspace is said to be *linear*.

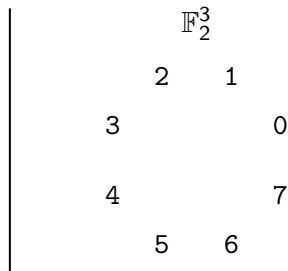


# Linear Partitions

## Definition (Linear Partition)

A partition of  $\mathbb{F}_2^n$  made up of all the cosets of a linear subspace is said to be *linear*.

Example of a linear partition over  $\mathbb{F}_2^3$ :



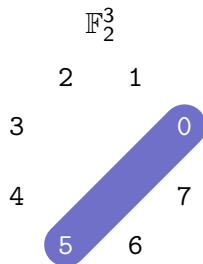
# Linear Partitions

## Definition (Linear Partition)

A partition of  $\mathbb{F}_2^n$  made up of all the cosets of a linear subspace is said to be *linear*.

Example of a linear partition over  $\mathbb{F}_2^3$ :

- $V = \{000, 101\} = \{0, 5\},$



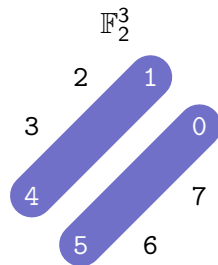
# Linear Partitions

## Definition (Linear Partition)

A partition of  $\mathbb{F}_2^n$  made up of all the cosets of a linear subspace is said to be *linear*.

Example of a linear partition over  $\mathbb{F}_2^3$ :

- $V = \{000, 101\} = \{0, 5\},$
- $001 + V = \{001, 100\} = \{1, 4\},$



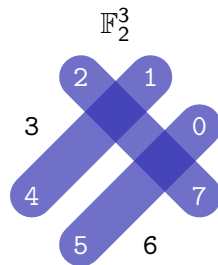
# Linear Partitions

## Definition (Linear Partition)

A partition of  $\mathbb{F}_2^n$  made up of all the cosets of a linear subspace is said to be *linear*.

Example of a linear partition over  $\mathbb{F}_2^3$ :

- $V = \{000, 101\} = \{0, 5\},$
- $001 + V = \{001, 100\} = \{1, 4\},$
- $010 + V = \{010, 111\} = \{2, 7\},$





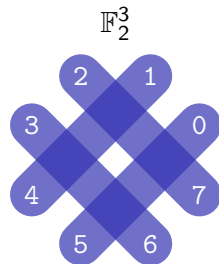
# Linear Partitions

## Definition (Linear Partition)

A partition of  $\mathbb{F}_2^n$  made up of all the cosets of a linear subspace is said to be *linear*.

Example of a linear partition over  $\mathbb{F}_2^3$ :

- $V = \{000, 101\} = \{0, 5\},$
- $001 + V = \{001, 100\} = \{1, 4\},$
- $010 + V = \{010, 111\} = \{2, 7\},$
- $011 + V = \{011, 110\} = \{3, 6\},$



# Linear Partitions

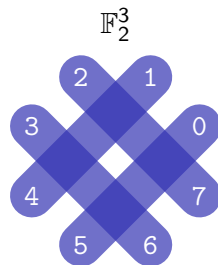
## Definition (Linear Partition)

A partition of  $\mathbb{F}_2^n$  made up of all the cosets of a linear subspace is said to be *linear*.

Example of a linear partition over  $\mathbb{F}_2^3$ :

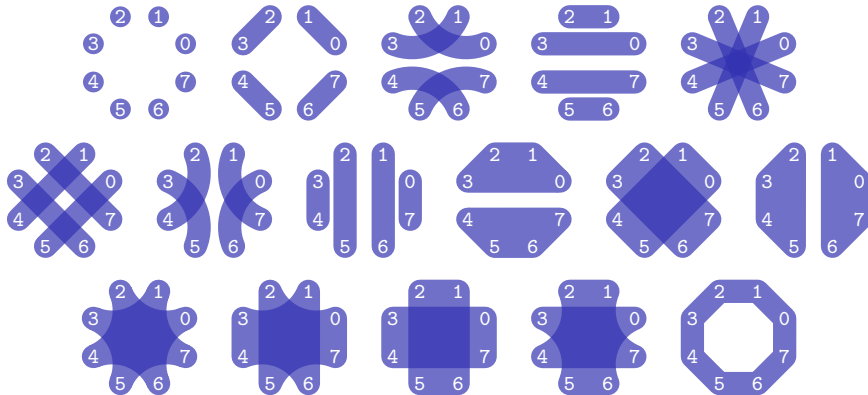
- $V = \{000, 101\} = \{0, 5\},$
- $001 + V = \{001, 100\} = \{1, 4\},$
- $010 + V = \{010, 111\} = \{2, 7\},$
- $011 + V = \{011, 110\} = \{3, 6\},$

$$\mathcal{L}(V) = \{\{0, 5\}, \{1, 4\}, \{2, 7\}, \{3, 6\}\}.$$



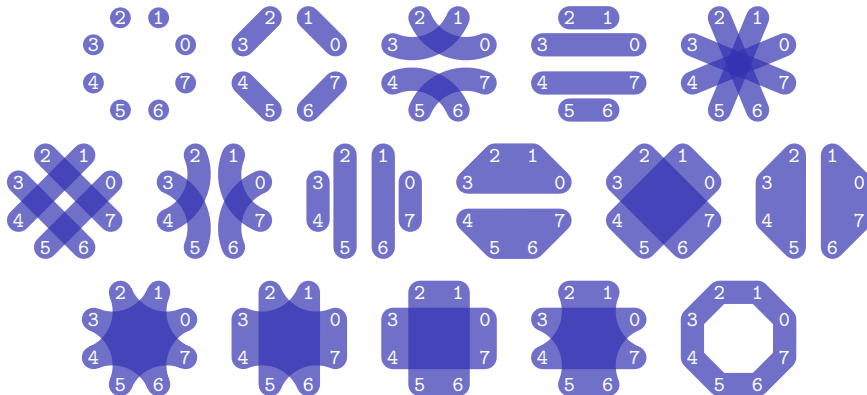
# Linear Partitions

The 16 linear partitions over  $\mathbb{F}_2^3$ :



# Linear Partitions

The 16 linear partitions over  $\mathbb{F}_2^3$ :



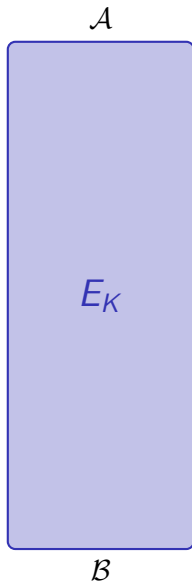
There are 229 755 605 linear partitions over  $\mathbb{F}_2^{10}$ .



# Partition-Based Backdoor SPN

## Assumption

The SPN maps  $\mathcal{A}$  to  $\mathcal{B}$ , no matter what the round keys are.



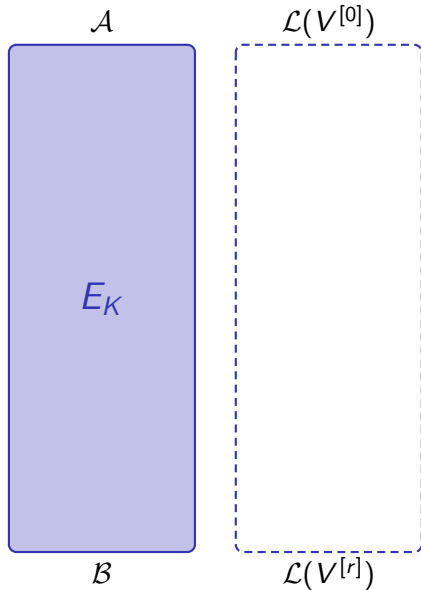
# Partition-Based Backdoor SPN

## Assumption

The SPN maps  $\mathcal{A}$  to  $\mathcal{B}$ , no matter what the round keys are.

Theoretical results :

- $\mathcal{A}$  and  $\mathcal{B}$  are linear,



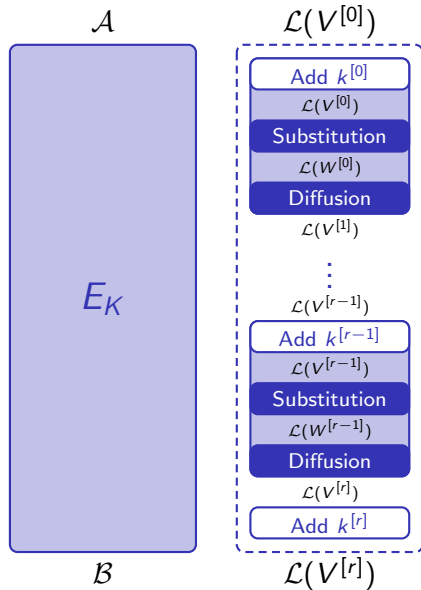
# Partition-Based Backdoor SPN

## Assumption

The SPN maps  $\mathcal{A}$  to  $\mathcal{B}$ , no matter what the round keys are.

Theoretical results :

- $\mathcal{A}$  and  $\mathcal{B}$  are linear,
- $\mathcal{A}$  is transformed through each step of the SPN in a deterministic way,



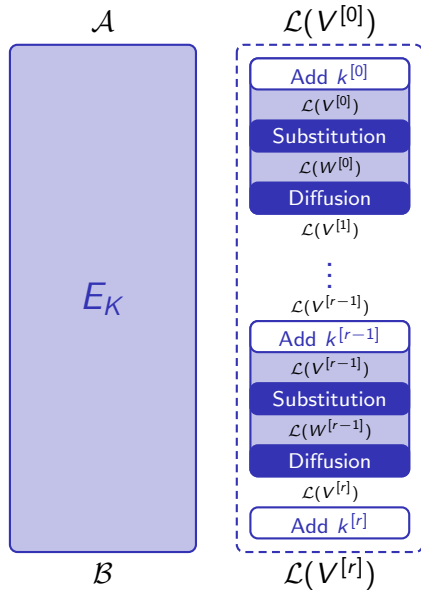
# Partition-Based Backdoor SPN

## Assumption

The SPN maps  $\mathcal{A}$  to  $\mathcal{B}$ , no matter what the round keys are.

Theoretical results :

- $\mathcal{A}$  and  $\mathcal{B}$  are linear,
- $\mathcal{A}$  is transformed through each step of the SPN in a deterministic way,
- At least one S-box maps a linear partition to another one.





# BEA-1 Key Features

- Parameters

- BEA-1 operates on 80-bit data blocks
- 120-bit master key and twelve 80-bit round keys
- 11 rounds (the last round involves two round keys)



# BEA-1 Key Features

- Parameters

- BEA-1 operates on 80-bit data blocks
- 120-bit master key and twelve 80-bit round keys
- 11 rounds (the last round involves two round keys)

- Primitives & base functions

- Key schedule & key addition (bitwise XOR)
- Substitution layer (involves four S-Boxes over  $\mathbb{F}_2^{10}$ )
- Diffusion layer (ShiftRows and MixColumns operations)
- Linear map  $M : (\mathbb{F}_2^{10})^4 \rightarrow (\mathbb{F}_2^{10})^4$



# BEA-1 Key Features

- Parameters

- BEA-1 operates on 80-bit data blocks
- 120-bit master key and twelve 80-bit round keys
- 11 rounds (the last round involves two round keys)

- Primitives & base functions

- Key schedule & key addition (bitwise XOR)
- Substitution layer (involves four S-Boxes over  $\mathbb{F}_2^{10}$ )
- Diffusion layer (ShiftRows and MixColumns operations)
- Linear map  $M : (\mathbb{F}_2^{10})^4 \rightarrow (\mathbb{F}_2^{10})^4$

- S-Boxes, linear map  $M$  and pseudo-codes for the different functions are given in our free book



# BEA-1 Key Features

- Parameters

- BEA-1 operates on 80-bit data blocks
- 120-bit master key and twelve 80-bit round keys
- 11 rounds (the last round involves two round keys)

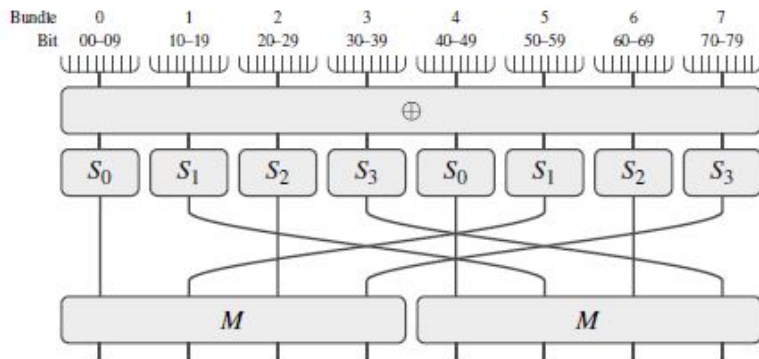
- Primitives & base functions

- Key schedule & key addition (bitwise XOR)
- Substitution layer (involves four S-Boxes over  $\mathbb{F}_2^{10}$ )
- Diffusion layer (ShiftRows and MixColumns operations)
- Linear map  $M : (\mathbb{F}_2^{10})^4 \rightarrow (\mathbb{F}_2^{10})^4$

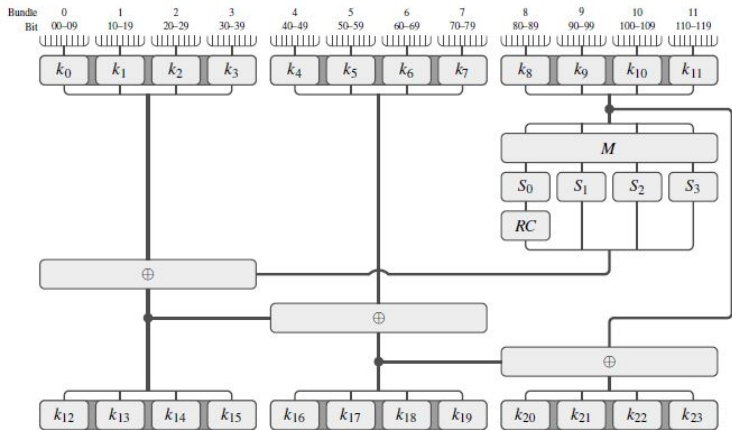
- S-Boxes, linear map  $M$  and pseudo-codes for the different functions are given in our free book
- BEA-1 is statistically compliant with FIPS 140 (US NIST standard) and resists to linear/differential attacks.



# BEA-1 Round Function



# BEA-1 Key Schedule

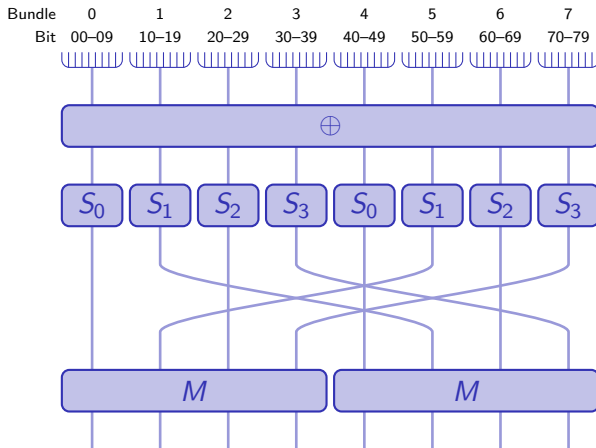


# Summary of the talk

- 1 Introduction: what is the issue?
- 2 History of known (and less known) backdoored algorithms
- 3 Description of BEA-1
- 4 BEA-1 Cryptanalysis
- 5 Conclusion and Future Work

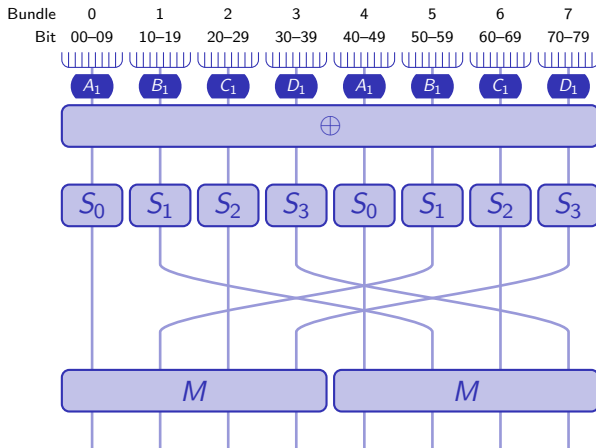


# Linear Partitions and the Round Function

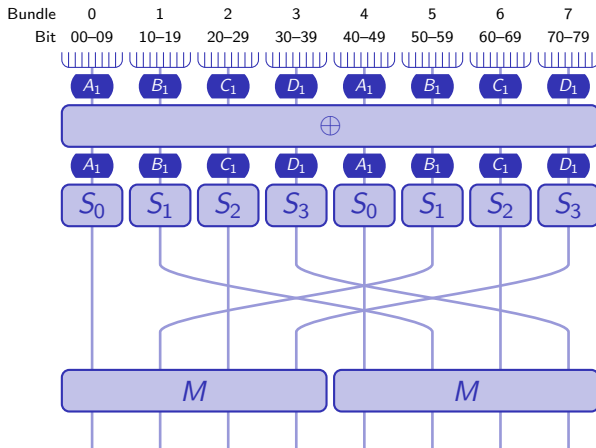




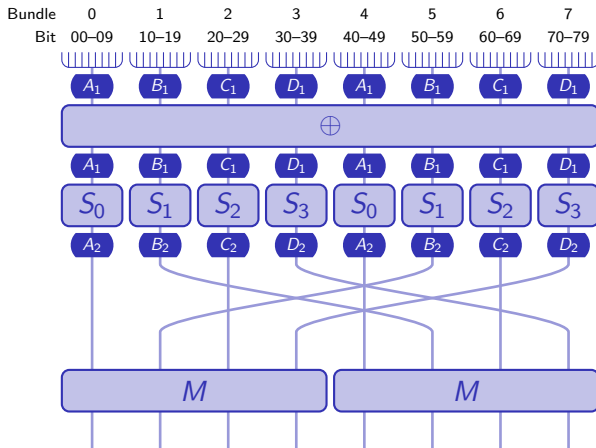
# Linear Partitions and the Round Function



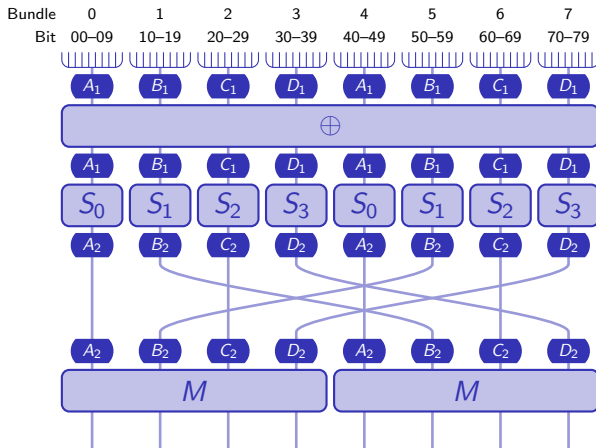
# Linear Partitions and the Round Function



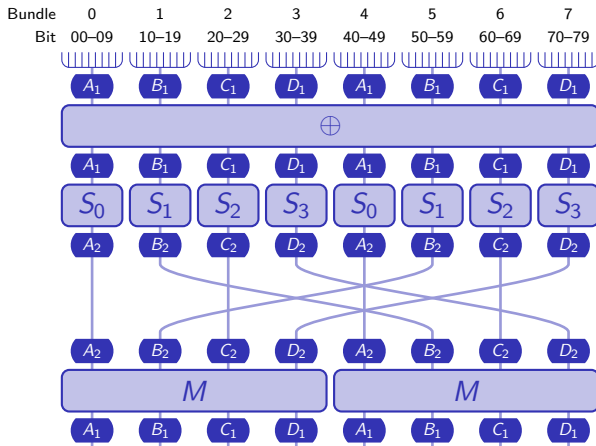
# Linear Partitions and the Round Function



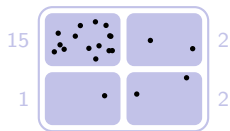
# Linear Partitions and the Round Function



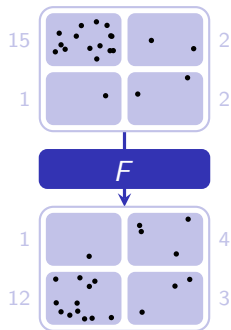
# Linear Partitions and the Round Function



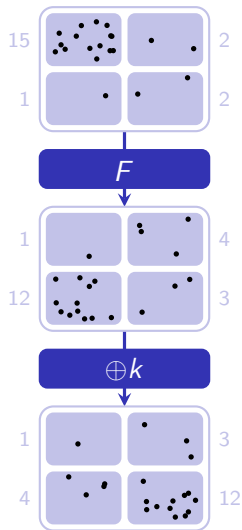
# Principle of the Cryptanalysis



# Principle of the Cryptanalysis



# Principle of the Cryptanalysis

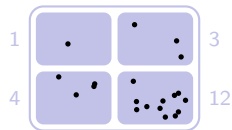
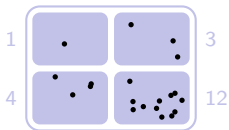
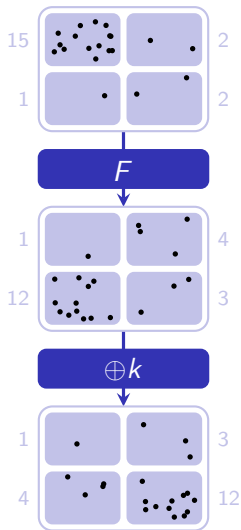




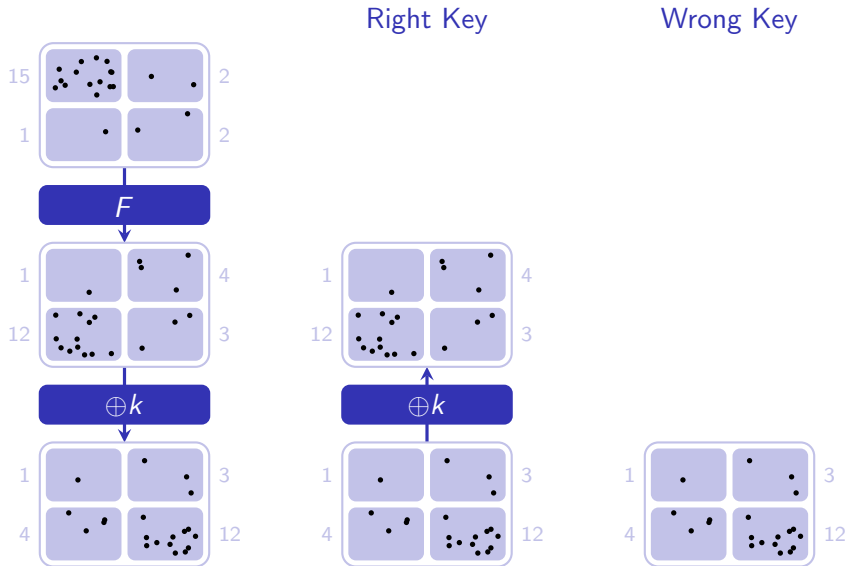
# Principle of the Cryptanalysis

Right Key

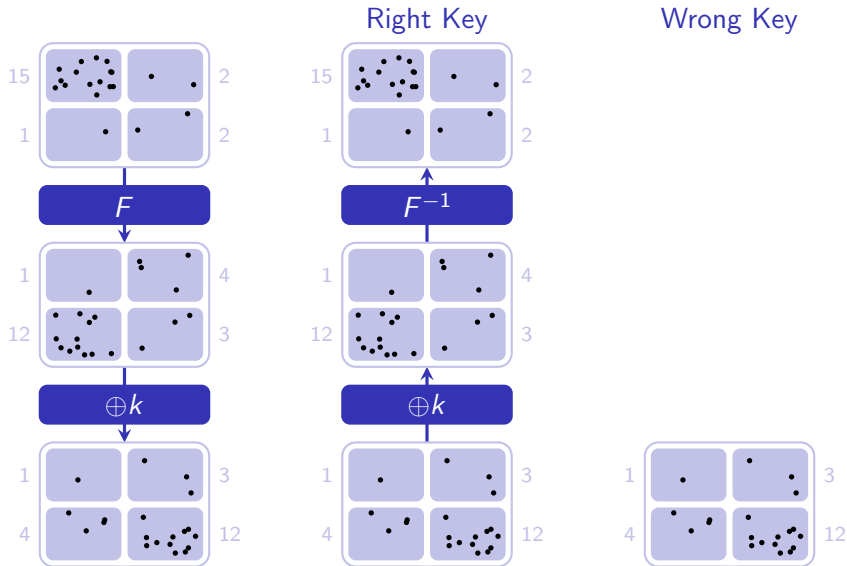
Wrong Key



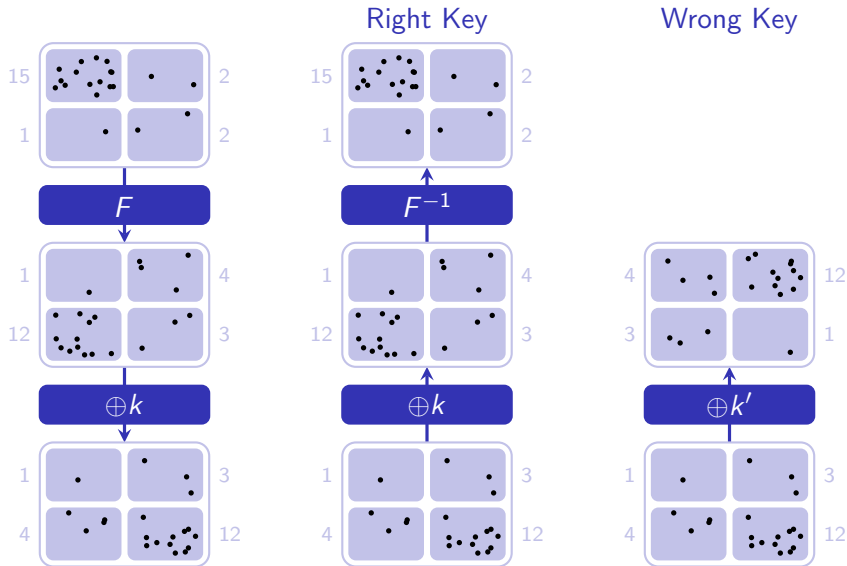
# Principle of the Cryptanalysis



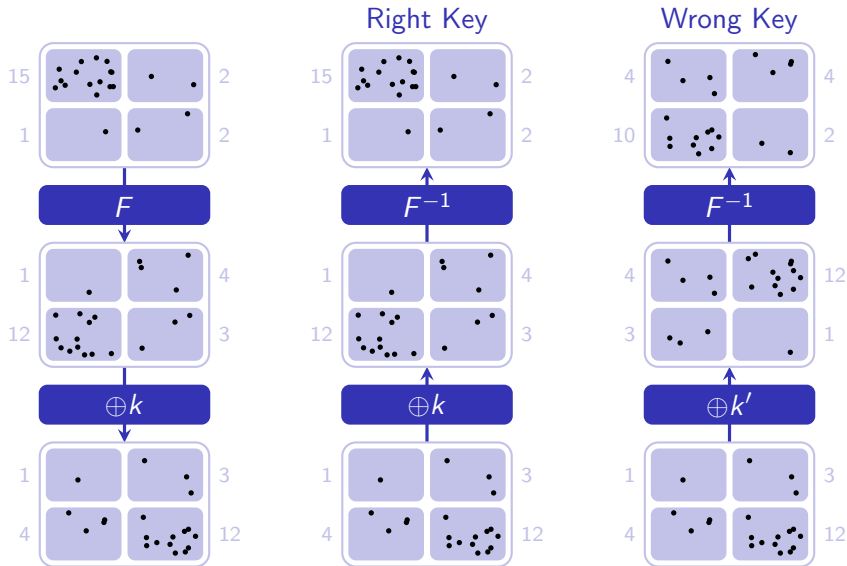
# Principle of the Cryptanalysis



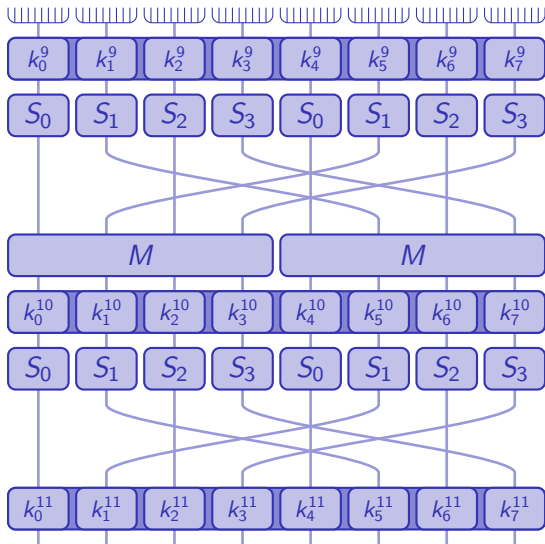
# Principle of the Cryptanalysis



# Principle of the Cryptanalysis



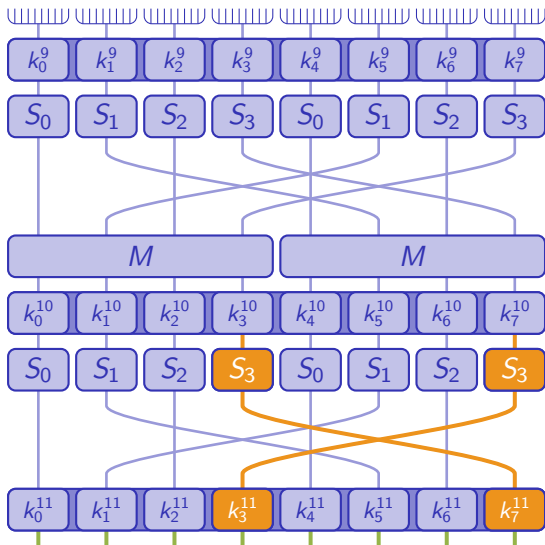
# Overview of the Cryptanalysis



Find the output coset of  
 $(A_2 \times B_2 \times C_2 \times D_2)^2$ .



# Overview of the Cryptanalysis



Brute force:

$(k_0^{11}, k_1^{11}, k_2^{11}, k_3^{11}, k_4^{11}, k_5^{11}, k_6^{11}, k_7^{11})$

Test the  $2^{15}$  saved keys:

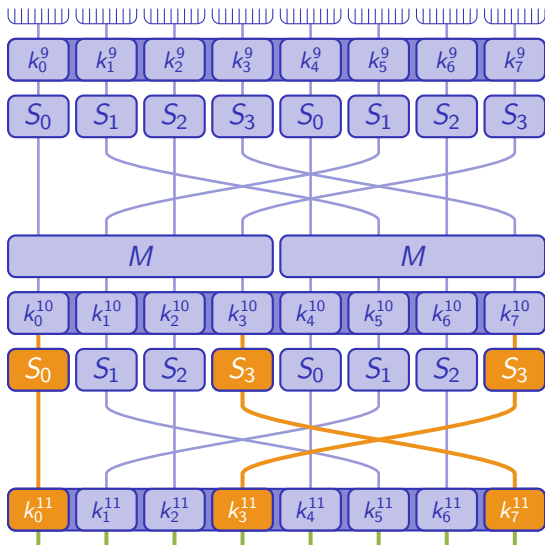
$(k_0^{11}, k_1^{11}, k_2^{11}, k_3^{11}, k_4^{11}, k_5^{11}, k_6^{11}, k_7^{11})$

Save the  $2^{15}$  best keys:

$(k_0^{11}, k_1^{11}, k_2^{11}, k_3^{11}, k_4^{11}, k_5^{11}, k_6^{11}, k_7^{11})$



# Overview of the Cryptanalysis



Brute force:

$(k_0^{11}, k_1^{11}, k_2^{11}, k_3^{11}, k_4^{11}, k_5^{11}, k_6^{11}, k_7^{11})$

Test the  $2^{15}$  saved keys:

$(k_0^{11}, k_1^{11}, k_2^{11}, k_3^{11}, k_4^{11}, k_5^{11}, k_6^{11}, k_7^{11})$

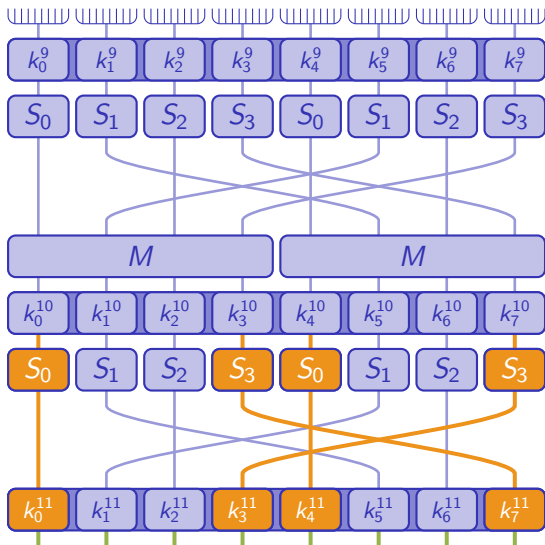
Save the  $2^{15}$  best keys:

$(k_0^{11}, k_1^{11}, k_2^{11}, k_3^{11}, k_4^{11}, k_5^{11}, k_6^{11}, k_7^{11})$





# Overview of the Cryptanalysis



Brute force:

$(k_0^{11}, k_1^{11}, k_2^{11}, k_3^{11}, k_4^{11}, k_5^{11}, k_6^{11}, k_7^{11})$

Test the  $2^{15}$  saved keys:

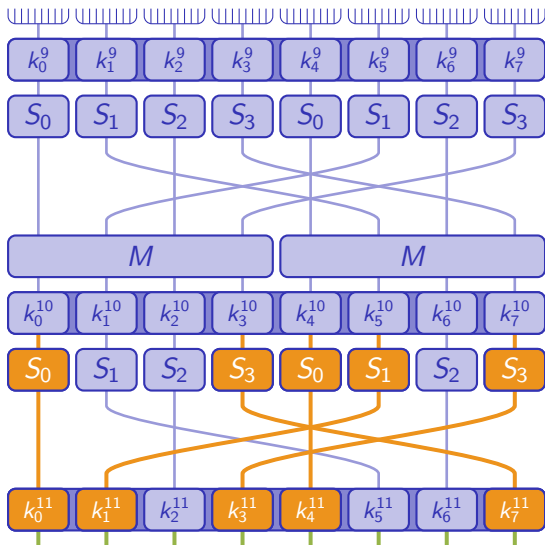
$(k_0^{11}, k_1^{11}, k_2^{11}, k_3^{11}, k_4^{11}, k_5^{11}, k_6^{11}, k_7^{11})$

Save the  $2^{15}$  best keys:

$(k_0^{11}, k_1^{11}, k_2^{11}, k_3^{11}, k_4^{11}, k_5^{11}, k_6^{11}, k_7^{11})$



# Overview of the Cryptanalysis



Brute force:

$(k_0^{11}, k_1^{11}, k_2^{11}, k_3^{11}, k_4^{11}, k_5^{11}, k_6^{11}, k_7^{11})$

Test the  $2^{15}$  saved keys:

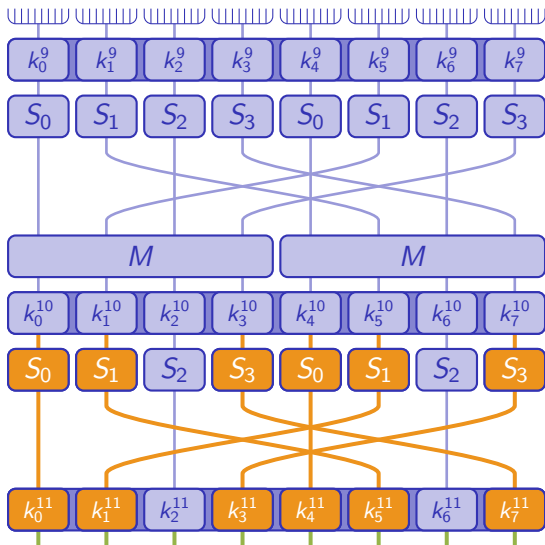
$(k_0^{11}, k_1^{11}, k_2^{11}, k_3^{11}, k_4^{11}, k_5^{11}, k_6^{11}, k_7^{11})$

Save the  $2^{15}$  best keys:

$(k_0^{11}, k_1^{11}, k_2^{11}, k_3^{11}, k_4^{11}, k_5^{11}, k_6^{11}, k_7^{11})$



# Overview of the Cryptanalysis



Brute force:

$(k_0^{11}, k_1^{11}, k_2^{11}, k_3^{11}, k_4^{11}, k_5^{11}, k_6^{11}, k_7^{11})$

Test the  $2^{15}$  saved keys:

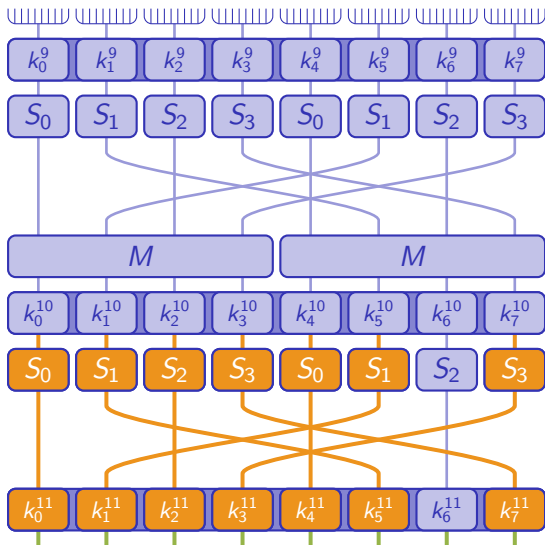
$(k_0^{11}, k_1^{11}, k_2^{11}, k_3^{11}, k_4^{11}, k_5^{11}, k_6^{11}, k_7^{11})$

Save the  $2^{15}$  best keys:

$(k_0^{11}, k_1^{11}, k_2^{11}, k_3^{11}, k_4^{11}, k_5^{11}, k_6^{11}, k_7^{11})$



# Overview of the Cryptanalysis



Brute force:

$(k_0^{11}, k_1^{11}, k_2^{11}, k_3^{11}, k_4^{11}, k_5^{11}, k_6^{11}, k_7^{11})$

Test the  $2^{15}$  saved keys:

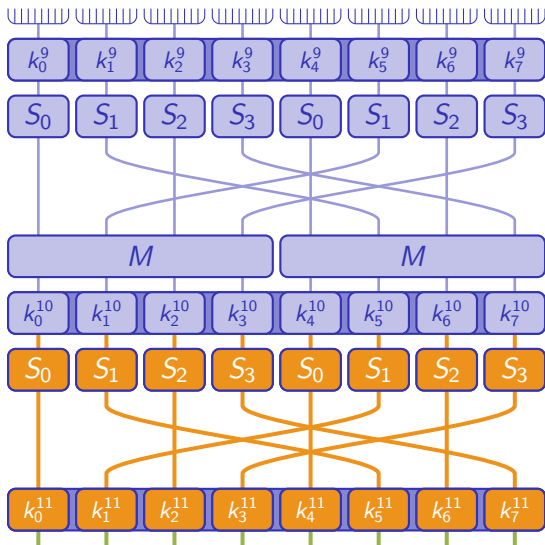
$(k_0^{11}, k_1^{11}, k_2^{11}, k_3^{11}, k_4^{11}, k_5^{11}, k_6^{11}, k_7^{11})$

Save the  $2^{15}$  best keys:

$(k_0^{11}, k_1^{11}, k_2^{11}, k_3^{11}, k_4^{11}, k_5^{11}, k_6^{11}, k_7^{11})$



# Overview of the Cryptanalysis



Brute force:

$(k_0^{11}, k_1^{11}, k_2^{11}, k_3^{11}, k_4^{11}, k_5^{11}, k_6^{11}, k_7^{11})$

Test the  $2^{15}$  saved keys:

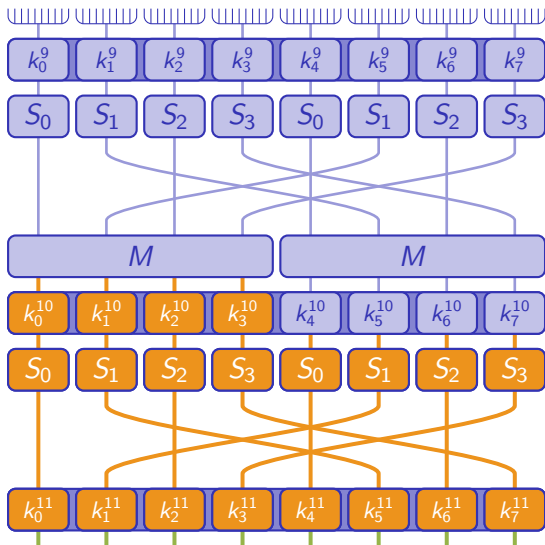
$(k_0^{11}, k_1^{11}, k_2^{11}, k_3^{11}, k_4^{11}, k_5^{11}, k_6^{11}, k_7^{11})$

Save the  $2^{15}$  best keys:

$(k_0^{11}, k_1^{11}, k_2^{11}, k_3^{11}, k_4^{11}, k_5^{11}, k_6^{11}, k_7^{11})$



# Overview of the Cryptanalysis



According to the key schedule:

$$k_0^{10} = k_0^{11} \oplus k_4^{11}$$

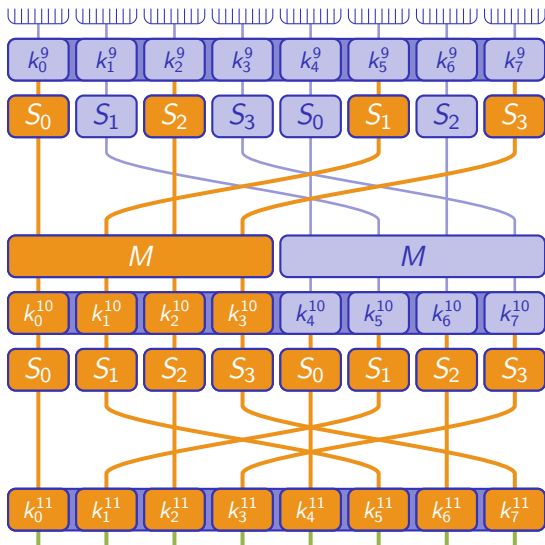
$$k_1^{10} = k_1^{11} \oplus k_5^{11}$$

$$k_2^{10} = k_2^{11} \oplus k_6^{11}$$

$$k_3^{10} = k_3^{11} \oplus k_7^{11}$$



# Overview of the Cryptanalysis

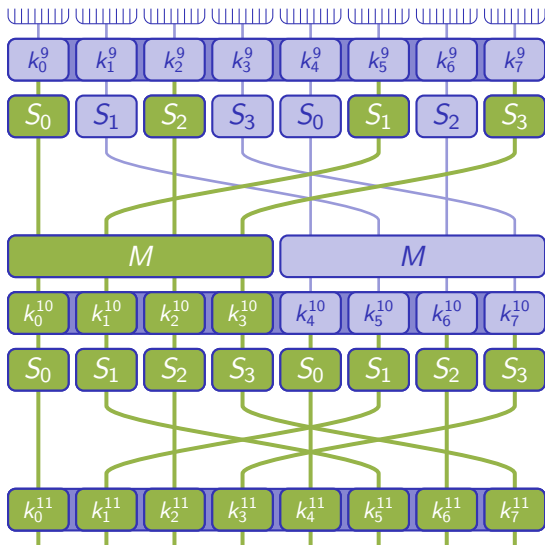


Test the  $2^{15}$  saved keys:

$(k_0^{11}, k_1^{11}, k_2^{11}, k_3^{11}, k_4^{11}, k_5^{11}, k_6^{11}, k_7^{11})$



# Overview of the Cryptanalysis



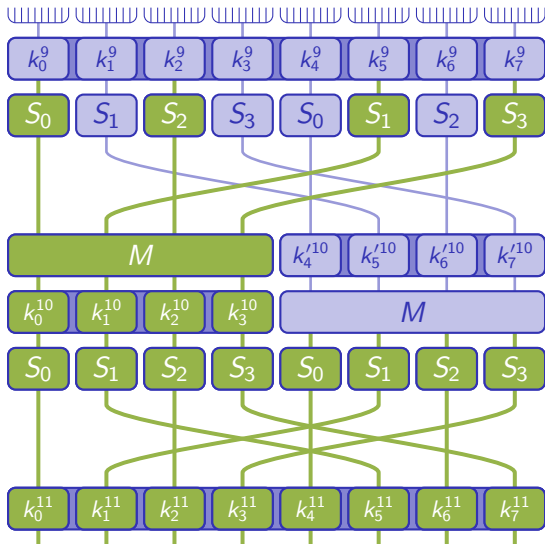
Save the best key:

$(k_0^{11}, k_1^{11}, k_2^{11}, k_3^{11}, k_4^{11}, k_5^{11}, k_6^{11}, k_7^{11})$





# Overview of the Cryptanalysis

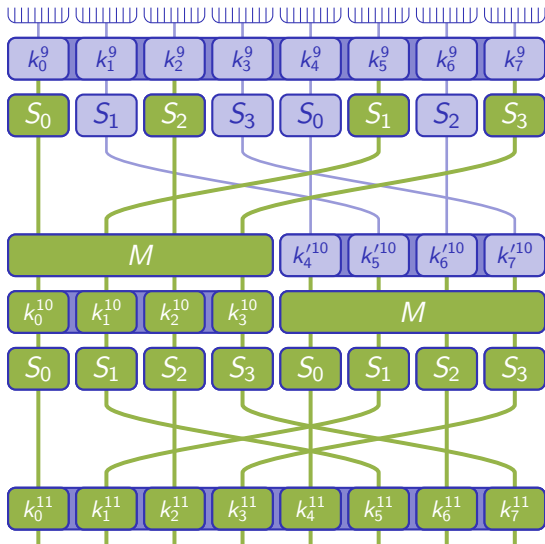


Observe that:

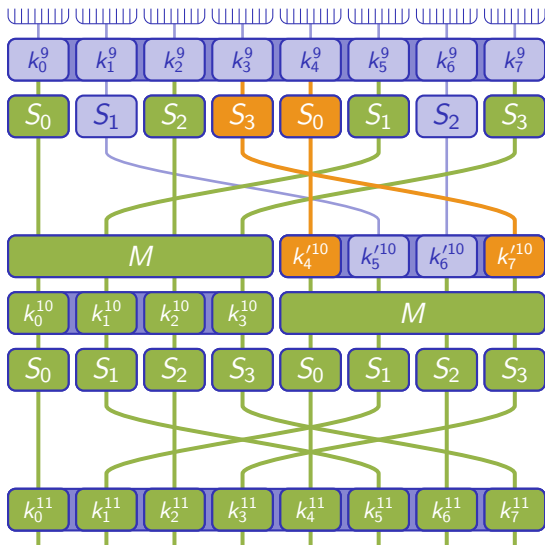
$$(k_4^{10}, k_5^{10}, k_6^{10}, k_7^{10}) \\ = M(k_4'^{10}, k_5'^{10}, k_6'^{10}, k_7'^{10})$$



# Overview of the Cryptanalysis



# Overview of the Cryptanalysis



Brute force:

$(k_4'^{10}, k_5'^{10}, k_6'^{10}, k_7'^{10})$

Test the  $2^{15}$  saved keys:

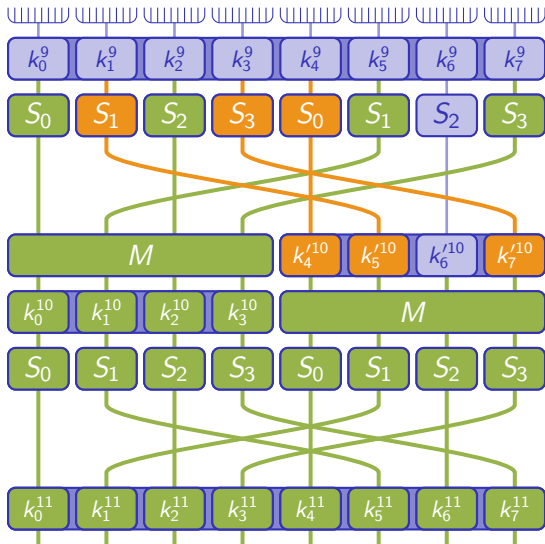
$(k_4'^{10}, k_5'^{10}, k_6'^{10}, k_7'^{10})$

Save the  $2^{15}$  best keys:

$(k_4'^{10}, k_5'^{10}, k_6'^{10}, k_7'^{10})$



# Overview of the Cryptanalysis



Brute force:

$(k_4'^{10}, k_5'^{10}, k_6'^{10}, k_7'^{10})$

Test the  $2^{15}$  saved keys:

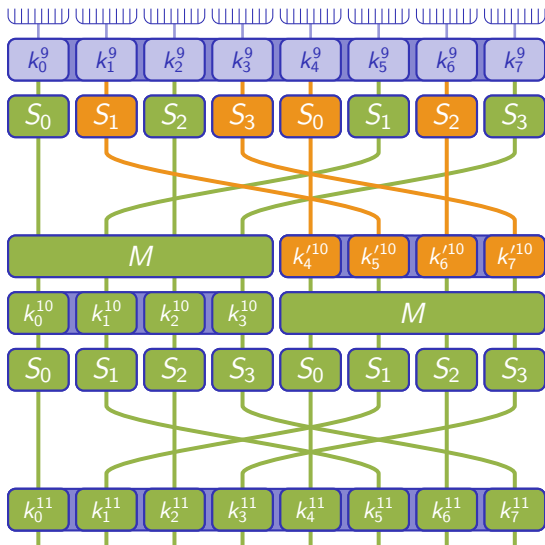
$(k_4'^{10}, k_5'^{10}, k_6'^{10}, k_7'^{10})$

Save the  $2^{15}$  best keys:

$(k_4'^{10}, k_5'^{10}, k_6'^{10}, k_7'^{10})$



# Overview of the Cryptanalysis



Brute force:

$(k_4'^{10}, k_5'^{10}, k_6'^{10}, k_7'^{10})$

Test the  $2^{15}$  saved keys:

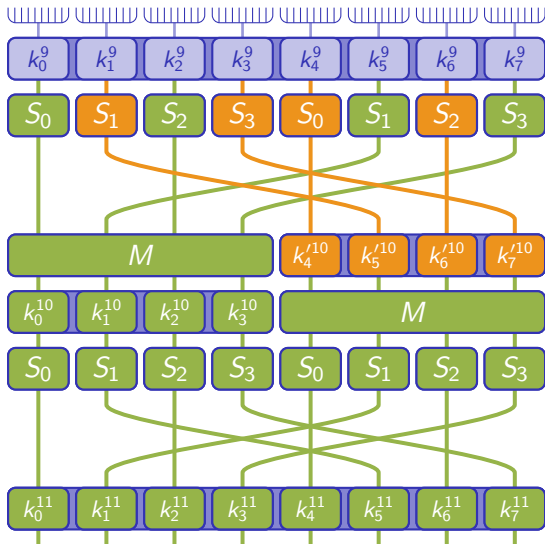
$(k_4'^{10}, k_5'^{10}, k_6'^{10}, k_7'^{10})$

Save the  $2^{15}$  best keys:

$(k_4'^{10}, k_5'^{10}, k_6'^{10}, k_7'^{10})$



# Overview of the Cryptanalysis



For each saved key,  
deduce the cipher key and test it



# Cryptanalysis Summary

- Probabilities for the modified cipher
  - $S_0, S_1, S_2$ : 944/1024,  $S_3$ : 925/1024



- Probabilities for the modified cipher

- $S_0, S_1, S_2$ :  $944/1024$ ,  $S_3$ :  $925/1024$
- Round function:  $(944/1024)^6 \times (925/1024)^2 \approx 2^{-1}$





- Probabilities for the modified cipher

- $S_0, S_1, S_2$ :  $944/1024$ ,  $S_3$ :  $925/1024$
- Round function:  $(944/1024)^6 \times (925/1024)^2 \approx 2^{-1}$
- Full cipher:  $(2^{-1})^{11} = 2^{-11}$



- Probabilities for the modified cipher

- $S_0, S_1, S_2$ :  $944/1024$ ,  $S_3$ :  $925/1024$
- Round function:  $(944/1024)^6 \times (925/1024)^2 \approx 2^{-1}$
- Full cipher:  $(2^{-1})^{11} = 2^{-11}$
- If 30 000 plaintexts lie in the same coset,  $30\,000 \times 2^{-11} \approx 15$  ciphertexts lie in the same coset on average



- Probabilities for the modified cipher

- $S_0, S_1, S_2$ : 944/1024,  $S_3$ : 925/1024
- Round function:  $(944/1024)^6 \times (925/1024)^2 \approx 2^{-1}$
- Full cipher:  $(2^{-1})^{11} = 2^{-11}$
- If 30 000 plaintexts lie in the same coset,  $30\,000 \times 2^{-11} \approx 15$  ciphertexts lie in the same coset on average

- Complexity of the cryptanalysis

- Data: 30 000 plaintext/ciphertext pairs ( $2 \times 300$  Kb)



- Probabilities for the modified cipher

- $S_0, S_1, S_2$ : 944/1024,  $S_3$ : 925/1024
- Round function:  $(944/1024)^6 \times (925/1024)^2 \approx 2^{-1}$
- Full cipher:  $(2^{-1})^{11} = 2^{-11}$
- If 30 000 plaintexts lie in the same coset,  $30\,000 \times 2^{-11} \approx 15$  ciphertexts lie in the same coset on average

- Complexity of the cryptanalysis

- Data: 30 000 plaintext/ciphertext pairs ( $2 \times 300$  Kb)
- Time:  $\approx 10$ s on a laptop (Core i7, 4 cores, 2.50GHz)



- Probabilities for the modified cipher

- $S_0, S_1, S_2$ : 944/1024,  $S_3$ : 925/1024
- Round function:  $(944/1024)^6 \times (925/1024)^2 \approx 2^{-1}$
- Full cipher:  $(2^{-1})^{11} = 2^{-11}$
- If 30 000 plaintexts lie in the same coset,  $30\,000 \times 2^{-11} \approx 15$  ciphertexts lie in the same coset on average

- Complexity of the cryptanalysis

- Data: 30 000 plaintext/ciphertext pairs ( $2 \times 300$  Kb)
- Time:  $\approx 10$ s on a laptop (Core i7, 4 cores, 2.50GHz)
- Probability of success  $> 95\%$



## Cryptanalysis demo



# Summary of the talk

- 1 Introduction: what is the issue?
- 2 History of known (and less known) backdoored algorithms
- 3 Description of BEA-1
- 4 BEA-1 Cryptanalysis
- 5 Conclusion and Future Work



# Conclusion

- Proposition of an AES-like backdoored algorithm (80-bit block, 120-bit key, 11 rounds)
  - The backdoor is at the design level
  - Resistant to most known cryptanalyses
  - But absolutely unsuitable for actual security
  - Illustrates the issue of using foreign encryption algorithms which might be backdoored





# Conclusion

- Proposition of an AES-like backdoored algorithm (80-bit block, 120-bit key, 11 rounds)
  - The backdoor is at the design level
  - Resistant to most known cryptanalyses
  - But absolutely unsuitable for actual security
  - Illustrates the issue of using foreign encryption algorithms which might be backdoored
- Future work
  - First step in a larger research work
  - Use of more sophisticated combinatorial structures
  - Considering key space partitioning
  - Other backdoored algorithms to be published.



Thank you for your attention  
Questions & Answers

