# The Not So Happily-Ever After End of AES' Security Fairytale

Orr Dunkelman

Faculty of Mathematics and Computer Science
Weizmann Institute of Science

Crypto Day 2010 — June $9^{th}$, 2010

מכון ויצמן למדע
WEIZMANN INSTITUTE OF SCIENCE

# Outline

# Block Ciphers

- One of the most basic cryptographic algorithms.
- A symmetric key algorithm (both sides hold secret information).
- Is a transformation of blocks of bits (of size $n$) into new blocks of bits (usually of the same size). Formally: $E : \{0,1\}^n \times \{0,1\}^k \mapsto \{0,1\}^n$ or $E_k : \{0,1\}^n \mapsto \{0,1\}^n$.
- To deal with more (or less) data, some mode of operation is used (ECB, CBC, counter mode, etc.).

# The History of Block Ciphers

## At the beginning

the NSA prevented research in block ciphers, and the block ciphers where chaos, and no public knowledge on how to design a good block cipher was available. And NBS said to IBM, "let there be a block cipher".

- In the mid-70's, the civil need for a secure block cipher led the US authorities to ask IBM to design a civil block cipher.
- The IBM team, headed by Horst Feistel, proposed a block cipher named Lucifer, which had a 64-bit block and 256-bit key.

# The History of Block Ciphers (cont.)

## And the NSA has seen

that the Lucifer was not good. And the NSA has told IBM how to make a better cipher. And the NSA saw the cipher, and said "it's good".

- After Lucifer was rejected (due to security reasons), IBM proposed a new cipher.
- The cipher, later selected as the **Data Encryption Standard (DES)** had a block size of 64 bits, and key size of 56 bits.
- Up to the complementation property of DES * it was considered secure, despite the short key size, and the unknown design criteria.

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

# The History of Block Ciphers (cont.)

## And the land has rested for 14 years

During these years, the best attack could have broken DES reduced to 7 out of its 16 rounds.

- DES was considered secure enough for practical purposes.
- To deal with the short key size, it was suggested to use double and triple encryptions, e.g.,

$$3DES_{K_1,K_2,K_3}(P) = DES_{K_3}(DES^{-1}_{K_2}(DES_{K_1}(P)))$$

# The History of Block Ciphers (cont.)

## And Eli and Adi said

"Let there be differential cryptanalysis", and showed an attack on the full DES faster than exhaustive search.

- Differential cryptanalysis [BS90] was the first evidence that the security of DES was not perfect (offering an attack of $2^{47}$ data and time on the full DES).
- Later, linear cryptanalysis [M93] further reduced the confidence in DES' security (offering an attack of $2^{43}$ data and time on the full DES).

# The History of Block Ciphers — The Late 90's

- Somewhere along the 90's, the way cryptography was used has changed as well.
- Cryptography entered each and every household, which resulted in a more hidden change — encryption was done in software rather than in hardware.
- DES, as history shows, was designed as an hardware-friendly cipher. At the same time, following its bit operations, it was not so software friendly.
- Along with the security issues identified in the early 90's, a need to replace DES was forming.

# The DES Challenges

- At the beginning, NIST refused to replace DES, claiming that 56-bit key cipher is sufficiently secure.
- As a response, a series of DES challenges were issued by RSA labs.
- In each challenge, RSA published a plaintext and its corresponding ciphertext, and offered 10,000$ for the first person to identify the key.
- The first challenge was solved in 75 days (involving 14,000–80,000 computers).
- The second challenge was solved in 39 days.
- The third was solved in 56 hours, using a special machine that the EFF has built (the DES cracker) for 210,000$.

# The AES Competition

- Following the requests for a more software-friendly encryption standard, NIST decided in 1997 to start a competition for a replacement to DES.
- The process was discussed thoroughly with the cryptographic community, and it was decided to hold an open competition.
- The cryptographic community was invited to submit proposals, and the evaluation process was meant to be open, i.e., everybody would get to analyze and comment about the other candidates.
- The block size was set to 128 bits, and three key sizes where required, 128, 192, and 256 bits.

**The target: Be faster and more secure than 3DES.**

# The Candidates

- 21 submissions were sent to NIST, 15 of which satisfied the requirements from the submissions:

| Candidate | Candidate | Candidate | Candidate | Candidate |
|---|---|---|---|---|
| CAST-256 | CRYPTON | DEAL | DFC | E2 |
| FROG | HPC | LOKI97 | MAGENTA | MARS |
| RC6 | Rijndael | SAFER++ | Serpent | TWOFISH |

- The first phase took a year, and at its end, 5 candidates were picked as finalists as they had merits over the other candidates.

# The Finalists

- MARS — designed by the IBM team (headed by Don Coppersmith).
- RC6 — designed by RSA people (headed by Ron Rivest)
- Rijndael — designed by K.U. Leuven post-docs (Joan Daeman and Vincent Rijmen).
- Serpent — designed by an international academic team (Ross Andresson, Eli Biham, and Lars R. Knudsen).
- Twofish — designed by Counterpane (headed by Bruce Schneier).

# The Finalists — Comparison

| Candidate | Type | # of Rounds | Best Attack(s) (as of 2000) |
|---|---|---|---|
| MARS | Generalized Feistel | $8 + 16 + 8$ | 11C [KKS00] or $8 + 5 + 8$ [KS00] |
| RC6 | Generalized Feistel | 20 | 14/14/15 [G+00,KM00] |
| Rijndael | SPN | 10/12/14 | 7/8/8 [F+00] |
| Serpent | SPN | 32 | 6/8/9 [F+00b] |
| Twofish | Feistel | 16 | 6 [F+99] |

# The Finalists — Performance

| Candidate | 32-bit Enc. cycles | 32-bit Dec. cycles | 8-bit Enc. cpb | ASIC fastest |
|---|---|---|---|---|
| MARS | 1600 | 1580 | 572 RAM/5468 ROM/ 2810 | 2.95 MGate/225 Mbps |
| RC6 | 1436 | 1406 | 156 RAM/1060 ROM/ 2130 | 1.64 MGate/203 Mbps |
| Rijndael | 1276 | 1276 | 66 RAM/980 ROM/ 1560 | 0.61 MGate/1950 Mbps |
| Serpent | 1800 | 2102 | 164 RAM/3937 ROM/ 4440 | 0.53 MGate/931 Mbps |
| Twofish | 1254 | 1162 | 90 RAM/2808 ROM/ 1940 | 0.43 MGate/394 Mbps |

32-bit machine in use: C code in Linux/GCC-2.7.2.2/Pentium 133 MHz MMX.
8-bit machine: Z80 CPU.
ASIC: Results due to [IKM00] in 0.35$\mu m$.

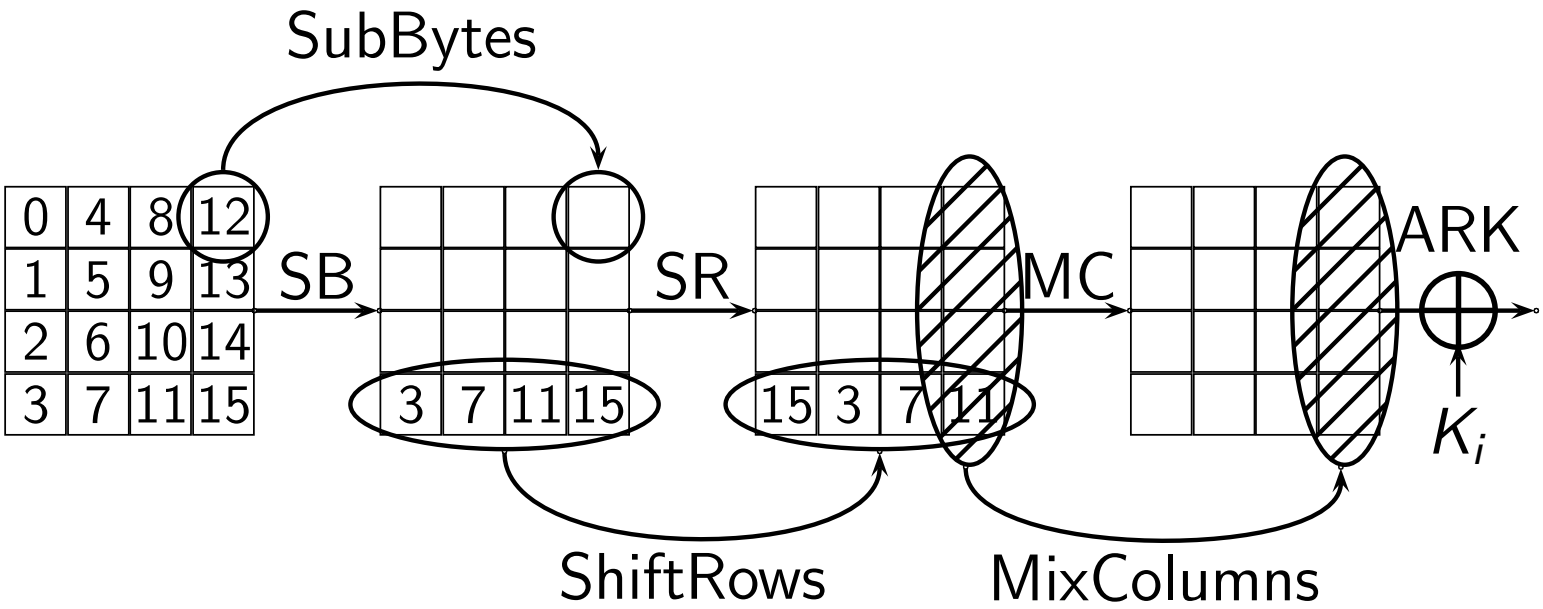
# The Selection — Rijndael as the AES

- On September 2001, Rijndael was announced as the **Advanced Encryption Standard** (AES).
- Rijndael was deemed to offer sufficient security, and affordable performance, i.e., being the fastest on many platforms and hardware friendly.
- Since then, AES implementations were improved:
  - Software implementations that run at $\approx 10$ cycles/byte.
  - New AES instruction in Westmere Intel CPUs allows encryption at 3.8 cycles/byte (and even 0.7 cycles/byte in counter mode).
  - Hardware implementations range from 3.1 Kgates (121 Mbps at 152 MHz using 0.13 $\mu m$) to 44 Gbps (with 250 Kgates).
  - FGPA performance also extremely good (up to 24 Gbps).

# The Advanced Encryption Standard

- The cipher has an SP (substitution-permutation) network structure.
- Block size — 128 bits, Key size — 128, 192, or 256 bits.
- Number of rounds depends on the key length (10/12/14, respectively).

# The Advanced Encryption Standard



# The MixColumns Operation

- MixColumns treats each column of four bytes as four elements over $GF(2^8)$. Then, the column is multiplied by the Matrix:

$$\begin{bmatrix} s'_0 \\ s'_1 \\ s'_2 \\ s'_3 \end{bmatrix} = \begin{bmatrix} 2 & 3 & 1 & 1 \\ 1 & 2 & 3 & 1 \\ 1 & 1 & 2 & 3 \\ 3 & 1 & 1 & 2 \end{bmatrix} \begin{bmatrix} s_0 \\ s_1 \\ s_2 \\ s_3 \end{bmatrix}$$

- The field $GF(2^8)$ is constructed over the (irreducible) polynomial 11B, i.e., $x^8 + x^4 + x^3 + x + 1$.

# The SubBytes Operation

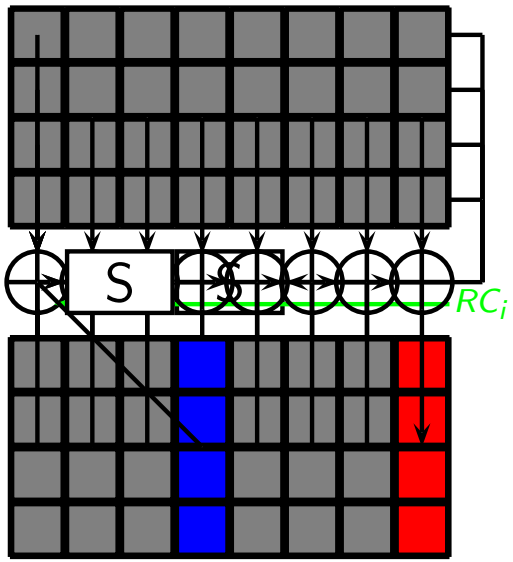
- Given input $x$, compute $y = x^{-1}$ (over the same field, with $0 \stackrel{\triangle}{=} 0^{-1}$).
- Then compute the output as:

$$\begin{bmatrix} z_0 \\ z_1 \\ z_2 \\ z_3 \\ z_4 \\ z_5 \\ z_6 \\ z_7 \end{bmatrix} = \begin{bmatrix} 1 & 0 & 0 & 0 & 1 & 1 & 1 & 1 \\ 1 & 1 & 0 & 0 & 0 & 1 & 1 & 1 \\ 1 & 1 & 1 & 0 & 0 & 0 & 1 & 1 \\ 1 & 1 & 1 & 1 & 0 & 0 & 0 & 1 \\ 1 & 1 & 1 & 1 & 1 & 0 & 0 & 0 \\ 0 & 1 & 1 & 1 & 1 & 1 & 0 & 0 \\ 0 & 0 & 1 & 1 & 1 & 1 & 1 & 0 \\ 0 & 0 & 0 & 1 & 1 & 1 & 1 & 1 \end{bmatrix} \begin{bmatrix} y_0 \\ y_1 \\ y_2 \\ y_3 \\ y_4 \\ y_5 \\ y_6 \\ y_7 \end{bmatrix} \oplus \begin{bmatrix} 1 \\ 1 \\ 0 \\ 0 \\ 0 \\ 1 \\ 1 \\ 0 \end{bmatrix}$$

# AES' Key Schedule Algorithm

The key schedule for AES with $32 \cdot Nk$-bit key:

- Initialize
  $W[0, \ldots, Nk-1] = K[0, \ldots, Nk-1]$.
- For $i = Nk, \ldots, 4 \cdot (7 + Nk) - 1$ do
  - If $i \equiv 0 \bmod Nk$ then
    $W[i] = W[i - Nk] \oplus$
    $SB(W[i-1] \lll 8) \oplus RCON[i/Nk]$,
  - Else if $Nk \equiv 8$ and $i \equiv 4 \bmod 8$ then
    $W[i] = W[i-8] \oplus SB(W[i-1])$,
  - Otherwise
    $W[i] = W[i-1] \oplus W[i - Nk]$,
- The first subkey is $W[0,1,2,3]$, the second is $W[4,5,6,7]$, etc.



# Security Properties

- The S-boxes are based on inversion over $GF(2^8)$.
- The MixColumns operation is an MDS matrix, which along with the ShiftRows operation ensures a full diffusion after two rounds.
- The "wide trail strategy" assures that the number of active S-boxes in any differential characteristic or linear approximation is at least five for two rounds, nine for three rounds, and 25 for four rounds.
- This ensures that any 4-round differential characteristic has probability of no more than $2^{-150}$.

# Security Properties (cont.)

- The security against differential and linear attacks is derived from the fact that there are no good differentials (linear hulls) of high probability.
- In a series of papers, the maximal expected differential and maximal expected linear probabilities for two and four rounds were computed.
- The results are that 4-round AES has no differentials or linear hulls with high enough probability for attacks (bounds have the order of magnitude of $2^{-110}$).
- Hence, any differential/linear attack on more than 6-round AES require about $2^{110}$ data.

# The Fairytale — Last Cipher to be Designed

- AES offers a very strong security.
- AES offers very good performance.
- Very easy to implement (even if you do not understand the entire *Galois Field* thingie).
- Hence, AES seemed the last cipher needed.
- OK, up to some extremely "unusual" scenarios (extremely constrained environments, extremely fast implementations, etc.).
- Hence, AES was deployed quickly, and added to security protocols in record times.
- Even in the SHA-3 competition — 8 submissions use AES components directly, and 6 more use AES-like components.

# Current State of Affairs in Cryptanalysis

- Most cryptanalytic papers discuss certificational attacks:
  - Data complexity — just slightly less than the entire code book.
  - Time complexity — just slightly less than exhaustive search.
  - Memory — store more information than there are particles in the universe.
- These certificational attacks are of great importance:
  1. Why to use a primitive which is less secure than optimal?
  2. By publishing the first step of analysis, others may be able to improve the attacks.
  3. Attacks only get better!

# What a Break is?

- There is an ongoing debate what a broken scheme is. Even from the theoretical point of view.
- The extreme approach: max(Time, Data, Memory) is less than Exhaustive search' time.
- Another approach: (Time, Data, Memory) is better then for generic attacks (e.g., time-memory-data tradeoff attacks).
- Time $\times$ Memory is less than required in exhaustive search.
- Money for finding a key in a given time is less than for a generic attack.

# Certificational Attacks on AES

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

# Security Implications of These Attacks

- Do not use AES-192/AES-256 as-is in Davies-Meyer compression functions.
- Do not assume AES-192/AES-256 to be related-subkey PRFs in security proofs.
- Do not assume that AES' security is perfect.
- But actually, as long as you use AES-192/AES-256 for encryption, and as long as your system does not allow related-subkeys, and as long as your encryption does not use crazy modes of operation, you are just fine.$^{\star\dagger}$

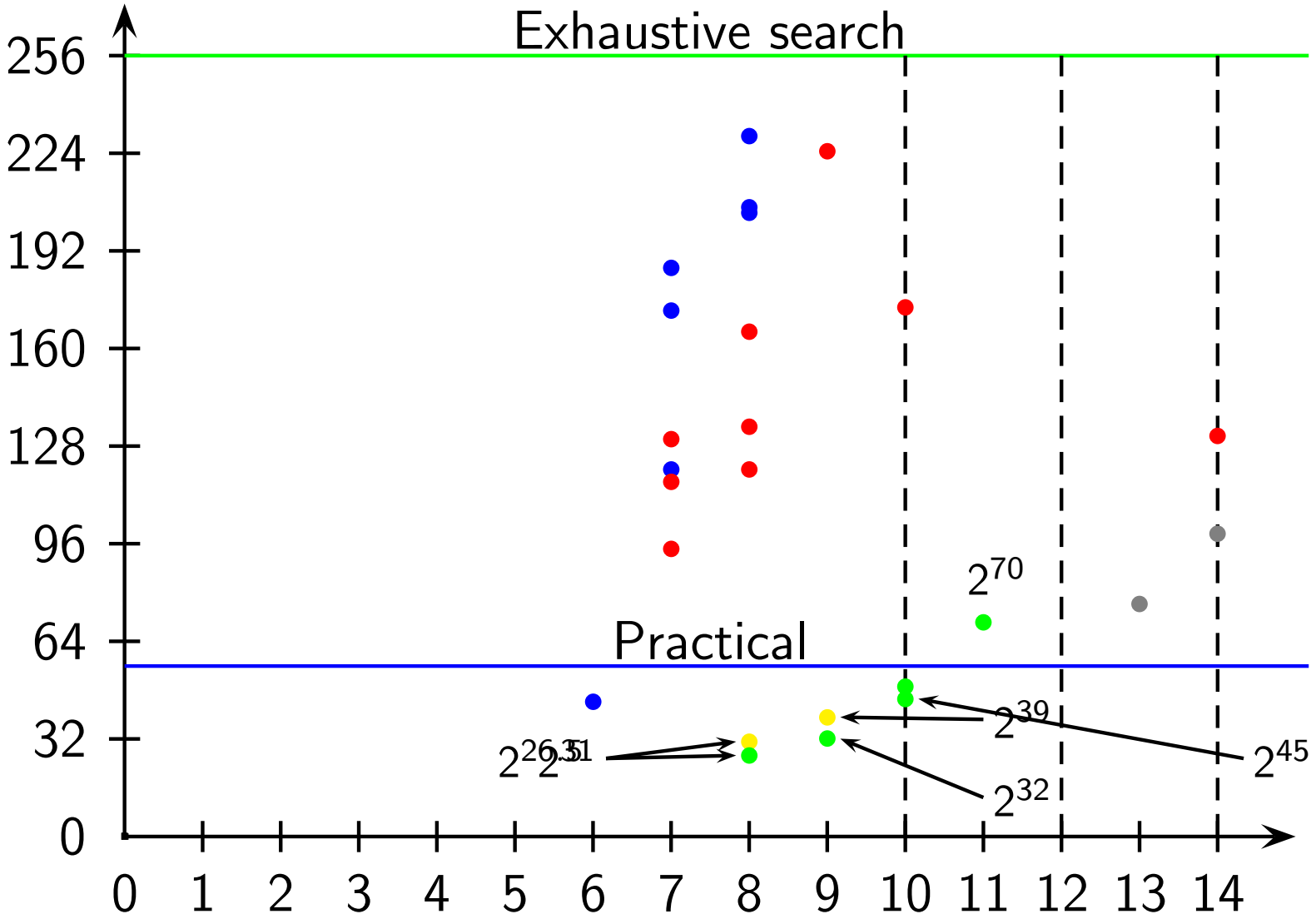
In other words, **No practical security implications for the end-user**

$^{\dagger}$Please consult a cryptographer before using AES in your system.

# What is a Practical Attack?

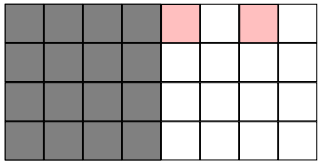
- We upper-bound the complexities of the attack.
- $2^{55}$ DES encryptions are feasible ...
- $2^{61}$ SHA-1 evaluations did not complete ...
- So, let's take $2^{64}$ cycles
  - which are about $2^{56}$ AES encryptions.
- This is also a restriction on the data complexity.

# Time Complexity of Attacks on AES-256

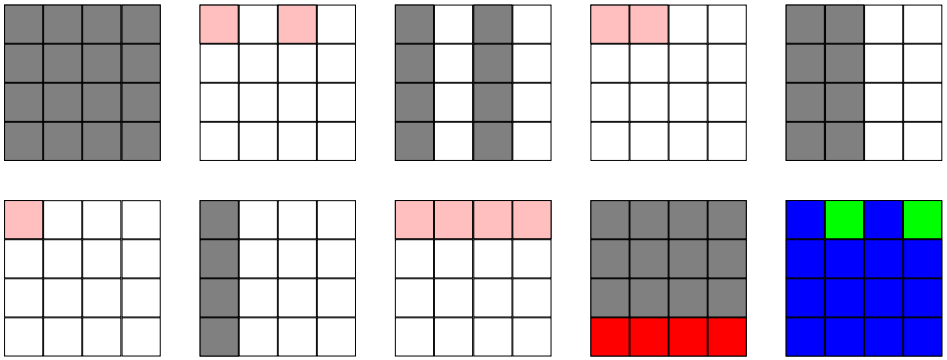


# An Interesting Property of the Key Schedule Algorithm of AES-256

Our results are based on the fact that key difference

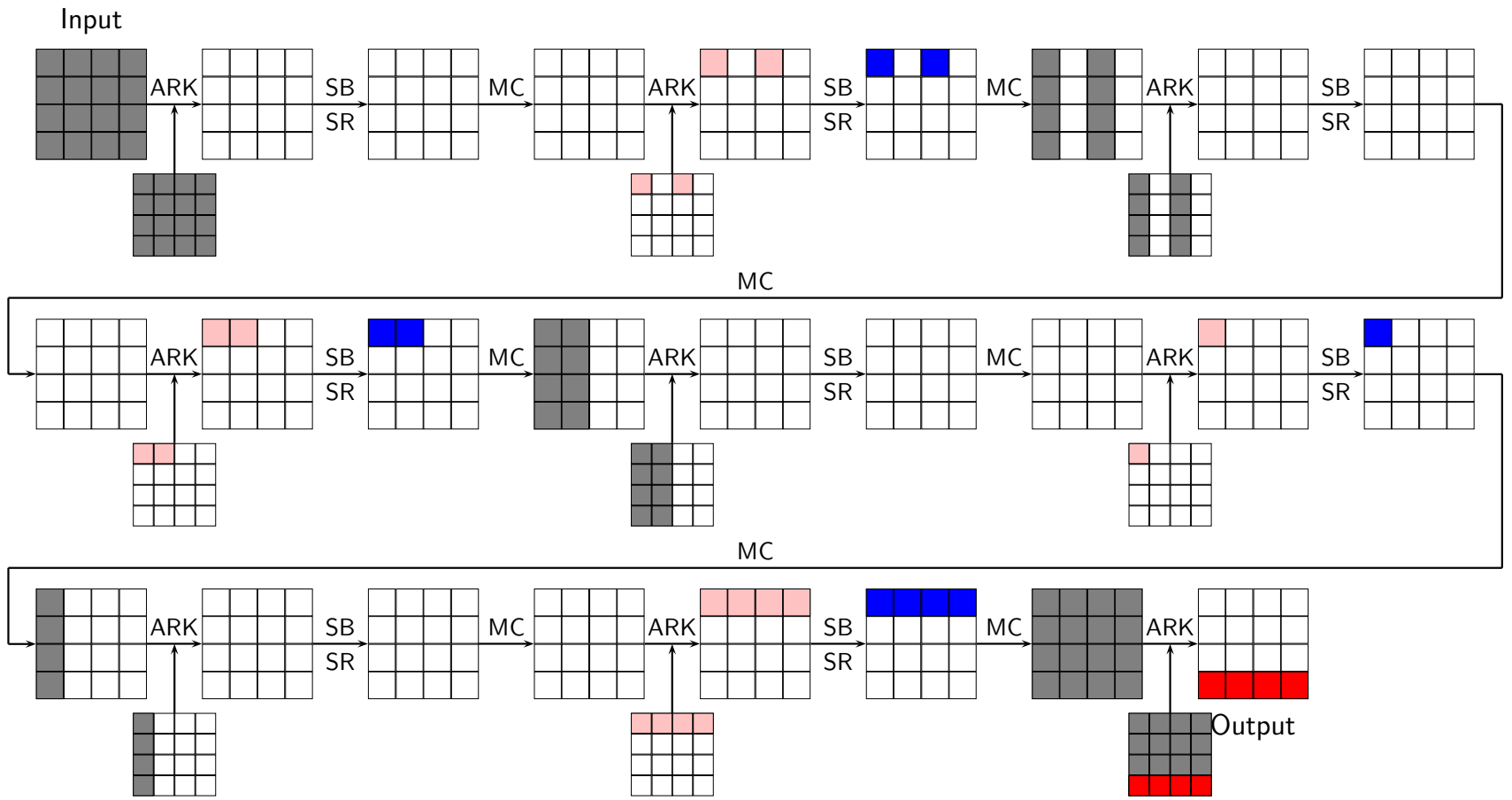


leads to the 10 subkey differences



**With probability 1!**

# An 8-Round Related-Key Differential




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

# Verification of the Differential

- We have experimentally verified the correctness of the 7-round related-key differential derived from the 8-round one (it has probability $2^{-30}$).
- We performed 100 experiments, each with a random key and $2^{32}$ random plaintext pairs.

| Pairs | 0 | 1 | 2 | 3 | 4 | 5 | 6 |
|---|---|---|---|---|---|---|---|
| Theory | 1.8 | 7.3 | 14.7 | 19.5 | 19.5 | 15.6 | 10.4 |
| Experiment | 0 | 10 | 18 | 10 | 28 | 18 | 6 |

| Pairs | 7 | 8 | 9 | 10 | 11 | 12 |
|---|---|---|---|---|---|---|
| Theory | 6.0 | 3.0 | 1.3 | 0.5 | 0.2 | 0.06 |
| Experiment | 8 | 1 | 0 | 0 | 0 | 1 |

# A 10-Round Related-Subkey Differential

- In the related-subkey model, it is possible to pick two keys which satisfy the difference in a slightly different manner.
- The related-subkey allows for shifting the differential by one round.
- This allows an extension of the differential in the backwards direction (despite having a highly active state).
- Which in turn, allows for attacks of practical complexity of up to 10 rounds, and semi-practical of up to 11 rounds.

# Other Attack Scenarios

- The attacks work when the plaintexts are generated not randomly as well.
- For example, when counter mode is used. The encryption system is initialized to two initial states and generates data sequentially. This simplifies the attack model.
- The attacks are applicable when the plaintexts are ASCII characters (as some key differences are suitable).
- Or even when they are ASCII characters representing only numeric values.
- The minimal hamming weight of the key difference is 24.

# Summary of the Attacks

| Rounds | Scenario | Time | Data | Memory | Result |
|---|---|---|---|---|---|
| 8 | Key Diff. – CP | $2^{31}$ | $2^{31}$ | 2 | Distinguisher |
| 8 | Subkey Diff. – CC | $2^{26.5}$ | $2^{26.5}$ | $2^{26.5}$ | 35 subkey bits |
| 9 | Key Diff. – CP | $2^{39}$ | $2^{38}$ | $2^{32}$ | Full key |
| 9 | Subkey Diff. – CC | $2^{32}$ | $2^{32}$ | $2^{32}$ | 56 key bits |
| 10 | Subkey Diff. – CP | $2^{49}$ | $2^{48}$ | $2^{33}$ | Distinguisher |
| 10 | Subkey Diff. – CC | $2^{45}$ | $2^{44}$ | $2^{33}$ | 35 subkey bits |

# Security Implications

- Extending AES-128 key to 256 bits actually reduces security!
- The security margins of AES-256 are significantly smaller than expected.
- Recently presented in [BK10] a 13-round attack in $2^{76}$ data and time (semi-practical).
- NIST is highly unlikely to modify AES' specifications.
- NIST is highly unlikely to start an AES-2 competition.

# Conclusions

- Did we break the full AES with practical complexity?
- Should users be worried?

# Questions?

**Thank you for your attention!**