Cryptanalysis of Forkciphers

Augustin Bariant, Nicolas David and Gaëtan Leurent

Inria de Paris

October 24, 2020

Tweakable block ciphers

- New parameter : the **tweak**.

$$\widetilde{E} : \{0,1\}^k \times \{0,1\}^t \times \{0,1\}^n \to \{0,1\}^n.$$

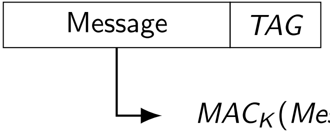
- The encryptions under different tweaks should be independant.
- Allows to encrypt blocks from the same plaintext with small collision probability.
- Independant family of block ciphers.

KIASU-BC

- Based on AES-128.
- 64-bit tweak XORed to the first two rows of the state, each round.
- New attacker model: the attacker can choose the tweak.
- No security loss compared to the AES **according to the designers**.
- However, most attacks on AES-128 reach one more round in KIASU-BC.

Authenticated encryption

- A TAG is added : the **MAC** (Message Authentification Code).



$$MAC_K(Message) = TAG?$$

- The TAG is checked upon reception of the message.
- **Impossible to generate the MAC without the key.**

Forkciphers

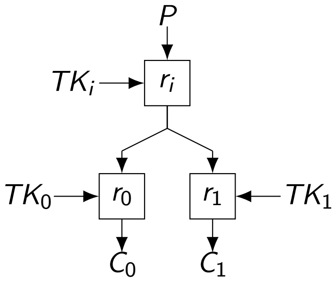
- Family of **authenticated** block ciphers.
- Efficient for **very short messages.**
- Based on existing block ciphers.
- A forkcipher outputs two ciphertexts C_0 et C_1:

$$\widetilde{F} : \{0,1\}^k \times \{0,1\}^t \times \{0,1\}^n \rightarrow \{0,1\}^n \times \{0,1\}^n.$$

- The second ciphertext can be interpreted as a **MAC**.
- The receiver checks if both ciphertexts correspond to the same plaintext.

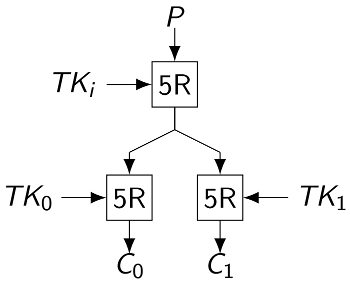
Description of the framework

- The plaintext goes through r_i rounds of blockcipher.
- The state is duplicated.
- The forked state goes through respectively r_0 et r_1, with different roundkeys.
- Both ciphertexts C_0 et C_1 are returned.

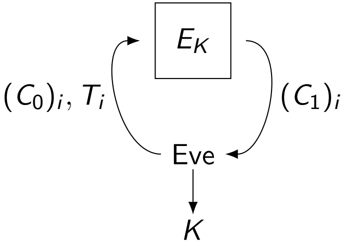


ForkAES

- ForkAES is a forkcipher based on KIASU-BC, with $r_0 = r_1 = r_i = 5$.



A new attacker model



- The attacker can chose a ciphertext C_0 , a tweak and the oracle returns the corresponding C_1.
- **The path from C_0 to C_1 consists of 5 decryption rounds followed by 5 encryption rounds**.

Best attacks on AES and KIASU-BC

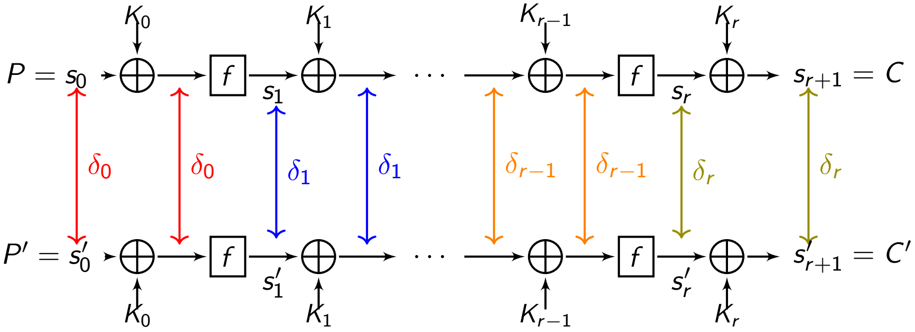
Algorithm	Attack Type	Rounds	Data	Time	Memory
AES-128	Impossible Diff.	7	$2^{106.2}$	$2^{110.2}$	$2^{90.2}$
AES-128	Meet in the Middle	7	2^{97}	2^{99}	2^{98}
KIASU-BC	Impossible Diff.	8	2^{118}	$2^{120.2}$	2^{102}
KIASU-BC	Boomerang	8	2^{103}	2^{103}	2^{60}
KIASU-BC	Meet in the Middle	8	2^{116}	2^{116}	2^{86}

Best attacks on ForkAES

Version	Attack type	Data	Time	Memory	Probability
ForkAES-$*$-4-4	Impossible Diff.	$2^{39.5}$	2^{47}	2^{35}	1
ForkAES-$*$-4-4	Reflection Diff.	2^{35}	2^{35}	2^{33}	1
ForkAES-$*$-5-5	**Truncated Diff.**	2^{73}	2^{73}	2^{58}	2^{-32}
ForkAES-$*$-5-5	**Truncated Diff.**	$2^{97.6}$	$2^{117.6}$	2^{85}	$2^{-5.4}$
ForkAES-$*$-5-5	**Truncated Diff.**	$2^{104.6}$	$2^{123.6}$	2^{96}	0.38

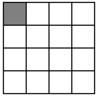
Differential paths

- Idea : Track the difference between a pair of messages.
- Probability of the entire differential path : Product of the probabilities to go from difference δ_i to δ_{i+1} through the round function **f**.



Truncated differential paths

- Set of differential paths.
- We keep track of bytes with no difference.
- Allows to significantly increase the probability of the path.
- We represent differences on a 4x4 matrix :



Active difference on the first byte.

Notations

- Bytes are elements of a 256-element field. Operations operate in this field.
- State bytes are numbered from 0 to 15, according to the following matrix :

0	4	8	12
1	5	9	13
2	6	10	14
3	7	11	15

Properties of S-boxes

- $\mathcal{P}(\delta_i, \delta_o)$ is the probability that the output difference of the S-box is δ_o if the input difference is δ_i.

Properties of S-boxes

- $\mathcal{P}(\delta_i, \delta_o)$ is the probability that the output difference of the S-box is δ_o if the input difference is δ_i.
- Properties of $\mathcal{P}$ for AES S-box:
  - For any non-zero δ_i, there exists a unique δ_o so that $\mathcal{P}(\delta_i, \delta_o) = 2^{-6}$.
  - For any non-zero δ_i, there exist precisely $2^7 - 1$ values δ_o so that $\mathcal{P}(\delta_i, \delta_o) \neq 0$.
  - For any non-zero δ_i and δ_o, $\mathcal{P}(\delta_i, \delta_o) \in \{0, 2^{-7}, 2^{-6}\}$.

Properties of S-boxes

- $\mathcal{P}(\delta_i, \delta_o)$ is the probability that the output difference of the S-box is δ_o if the input difference is δ_i.
- Properties of $\mathcal{P}$ for AES S-box:
  - For any non-zero δ_i, there exists a unique δ_o so that $\mathcal{P}(\delta_i, \delta_o) = 2^{-6}$.
  - For any non-zero δ_i, there exist precisely $2^7 - 1$ values δ_o so that $\mathcal{P}(\delta_i, \delta_o) \neq 0$.
  - For any non-zero δ_i and δ_o, $\mathcal{P}(\delta_i, \delta_o) \in \{0, 2^{-7}, 2^{-6}\}$.
- If δ_i and δ_o are randomly picked, there is in average one solution x to the following equation:

$$SB(x) \oplus SB(x \oplus \delta_i) = \delta_o$$

Properties of S-boxes

- $\mathcal{P}(\delta_i, \delta_o)$ is the probability that the output difference of the S-box is δ_o if the input difference is δ_i.
- Properties of $\mathcal{P}$ for AES S-box:
  - For any non-zero δ_i, there exists a unique δ_o so that $\mathcal{P}(\delta_i, \delta_o) = 2^{-6}$.
  - For any non-zero δ_i, there exist precisely $2^7 - 1$ values δ_o so that $\mathcal{P}(\delta_i, \delta_o) \neq 0$.
  - For any non-zero δ_i and δ_o, $\mathcal{P}(\delta_i, \delta_o) \in \{0, 2^{-7}, 2^{-6}\}$.
- If δ_i and δ_o are randomly picked, there is in average one solution x to the following equation:

$$SB(x) \oplus SB(x \oplus \delta_i) = \delta_o$$

- $\Theta[0]$ is chosen so that $\mathcal{P}(\Theta[0], \Theta[0]/2) = 2^{-6}$.

- $k_5 + k_{11}$ has a zero diagonal (the key corresponding to the junction of both branches).
- Probability 2^{-32}.
- Differential attack with complexity $< 2^{96}$

Ideas of the attack

- Differential path with probability p.

Ideas of the attack

- Differential path with probability p.
- We define four functions P_0, P_1, P'_0 and $P'_1 : \{0,1\}^{96} \rightarrow \{0,1\}^{128}$ so that, for any pair of 96-bit vectors (u, v) :

Ideas of the attack

- Differential path with probability p.
- We define four functions P_0, P_1, P'_0 and $P'_1 : \{0,1\}^{96} \rightarrow \{0,1\}^{128}$ so that, for any pair of 96-bit vectors (u, v) :
  - If $(P_0(u), P_1(v))$ satisfies the differential path, $(P'_0(u), P'_1(v))$ also does with probability $p' \gg p$.

Ideas of the attack

- Differential path with probability p.
- We define four functions P_0, P_1, P'_0 and $P'_1 : \{0,1\}^{96} \rightarrow \{0,1\}^{128}$ so that, for any pair of 96-bit vectors (u, v) :
  - If $(P_0(u), P_1(v))$ satisfies the differential path, $(P'_0(u), P'_1(v))$ also does with probability $p' \gg p$.
  - $(P_0(u), P_1(v))$ and $(P'_0(u), P'_1(v))$ satisfy the differential with probability $p \times p'$.

Ideas of the attack

- We generate a large set of 96-bit vectors u_i.

Ideas of the attack

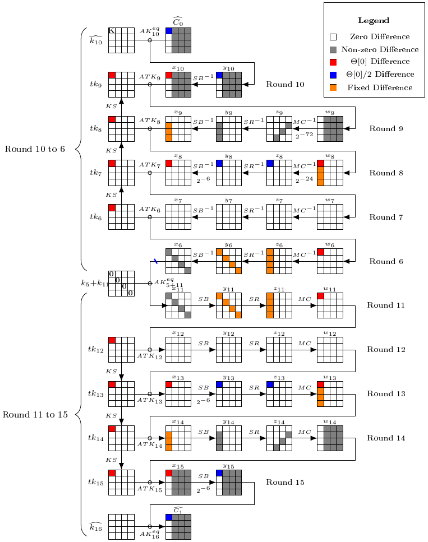
- We generate a large set of 96-bit vectors u_i.
- For each vector u_i of the set, we compute $P_0(u_i)$, $P'_0(u_i)$, $P_1(u_i)$, $P'_1(u_i)$.

Ideas of the attack

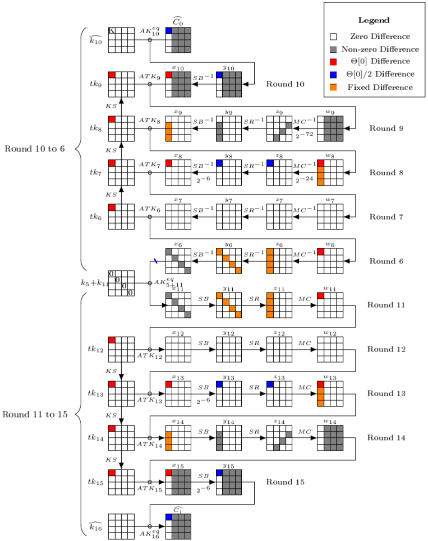
- We generate a large set of 96-bit vectors u_i.
- For each vector u_i of the set, we compute $P_0(u_i)$, $P'_0(u_i)$, $P_1(u_i)$, $P'_1(u_i)$.
- We look for pairs of 96-bit vectors (u_i, v_j) so that $(P_0(u_i), P_1(v_j))$ and $(P'_0(u_i), P'_1(v_j))$ have the differential output difference.

Ideas of the attack

- We generate a large set of 96-bit vectors u_i.
- For each vector u_i of the set, we compute $P_0(u_i)$, $P'_0(u_i)$, $P_1(u_i)$, $P'_1(u_i)$.
- We look for pairs of 96-bit vectors (u_i, v_j) so that $(P_0(u_i), P_1(v_j))$ and $(P'_0(u_i), P'_1(v_j))$ have the differential output difference.
- For each pair of ciphertexts passing the differential path, we deduce key bits. We keep couples (u_i, v_j) if the deduced key bits of both created pairs are compatible.



- (a_0, a_1) is a pair of bytes with difference $\Theta[0]/2$ and $SB^{-1}(a_0) \oplus SB^{-1}(a_1) = \Theta[0]$.
- We denote $b_i = SB^{-1}(a_i)$ and $c_i = SB(b_i + \tau[0])$.
- We guess the first byte of $\widehat{k_{10}}$ and denote it K.



$$P(u,v) = (P_0(u), P_1(v)) = \left(\left(\begin{pmatrix} a_0 + K & u \\ 0 & \\ 0 & \\ 0 & \end{pmatrix}, 0 \right), \left(\begin{pmatrix} a_1 + K & v \\ 0 & \\ 0 & \\ 0 & \end{pmatrix}, \Theta \right) \right)$$

$$P'(u,v) = (P'_0(u), P'_1(v)) = \left(\left(\begin{pmatrix} c_0 + K & u \\ 0 & \\ 0 & \\ 0 & \end{pmatrix}, \tau \right), \left(\begin{pmatrix} c_1 + K & v \\ 0 & \\ 0 & \\ 0 & \end{pmatrix}, \tau + \Theta \right) \right)$$

- If P satisfies the differential path, P' is also inactive during round 7 with probability 1.
- $p = 2^{-114}$, $p' = 2^{-12}$ so $p_{tot} = 2^{-126}$
- The output difference allows to filter every pair.

Efficiency of the filter

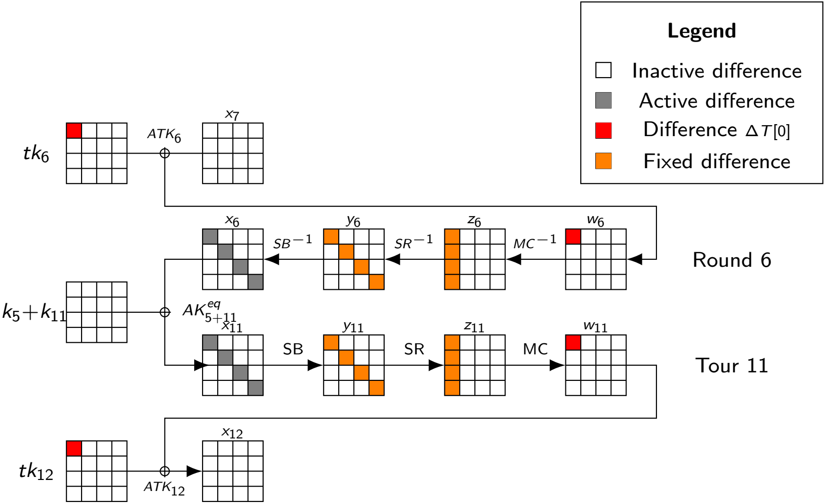
1. Let us generate a set of 2^{63} 96-bit vectors (2^{126} pairs).
2. In average, one couple $(P(u,v), P'(u,v))$ satisfies the differential path.
3. We observe and store collisions between the first column of the ciphertexts of $(P_0(u), P_1(u))$ and of $(P'_0(v), P'_1(v))$.
4. Each collision represents a pair having the right output difference. This happens with probability 2^{-64} for random pairs.
5. In total, we filter out 70 bits, and 2^{56} pairs remain.

- The pairs of a couple satisfying the path must have a common key candidate per column.
- A random couple has a common key candidate for each column with probability $2^7 \times 2^7 / 2^{32} = 2^{-18}$
- In total, we had 2^{56} pairs we filter 2^{54}, so there remains 2^2, for a total of 102 guessed key bytes.
- We end with a exhaustive search on remaining key bytes.

Attack complexity

- The complexity of the attack in (Data, Time, Memory) is:

$$(D, T, M) = (2^{73}, 2^{73}, 2^{58}).$$


Legend
Inactive difference
Active difference
Difference $\Delta T[0]$
Fixed difference
tk_6
ATK_6
x_7
x_6
SB^{-1}
y_6
SR^{-1}
z_6
MC^{-1}
w_6
Round 6
$k_5 + k_{11}$
AK^{eq}_{5+11}
x_{11}
SB
y_{11}
SR
z_{11}
MC
w_{11}
Tour 11
tk_{12}
ATK_{12}
x_{12}

- Difference in states y_6 and y_{11} are exactly the same.

$$
\begin{aligned}
SB(x_6[i]) + SB(x'_6[i])) =& (y_6 + y'_6)[i] \\
SB(k[i] + x_6[i]) + SB(k[i] + x'_6[i]) =& (y_6 + y'_6)[i].
\end{aligned}
$$

- $(x_6[i], x'_6[i])$ and $(k[i] + x_6[i], k[i] + x'_6[i])$ are two pairs with the same difference, and their output difference through the S-box SB is the same.
- For some values of $k[i]$ and of $(y_6 + y'_6)[i]$, these equations have no solution.

- There is a $1/16$ probability that the key is compatible with the tweak difference ($1/2$ per diagonal byte).
- In this case, probability to satisfy round 6 and 11 is 2^{-28} instead of 2^{-32}.
- This result has not been found by authors of "Cryptanalysis of ForkAES", who constructed a similar characteristic with a unique tweak difference.

- New hypothesis : $k_5 + k_{11}$ has a zero diagonal byte (probability 2^{-6}).
- The tweak and the key are compatible with probability 2^{-3}.
- Middle rounds are satisfied with probability $2^{-(24-3)}2^{-21}$.
- The probability that both pairs pass the differential characteristic is 2^{-168}.
- We need 2^{84} vectors of 96 bits.
- The same filter is applied.
- The same key recovering technique is applied.

Complexity and probability of success of the second attack

- There exists three difference of $\Theta[0]$ so that $\mathcal{P}(\Theta[0], \Theta[0]/2) = 2^{-6}$.
- We can perform this attack by rotating the columns.
- We have a probability $1/12$ of having a tweak compatible with the key.
- Probability of success : $3/2^{-7}$.
- Complexity in (Data, Time, Memory) :

$$(D, T, M) = (2^{97.6}, 2^{117.6}, 2^{85}).$$

Attacking even more keys

- No hypothesis on $k_5 + k_{11}$.
- We add an intermediate filter.
- Probability of success: 0.38.
- Complexity in (Data, Time, Memory) :

$$(D, T, M) = (2^{104.6}, 2^{123.6}, 2^{96}).$$

Conclusion

- KIASU-BC is less secure than AES-128.
- ForkAES is far less secure than KIASU-BC.
- Forkciphers need to be carefully analysed, as they give an extra angle of attack to the attacker.
- ForkSkinny, ...

Thank you for your attention