Tuple cryptanalysis of ARX with application to BLAKE and Skein

J.-Ph. Aumasson G. Leurent W. Meier F. Mendel
N. Mouha R. C-W Phan Y. Sasaki P. Susil

Tuple cryptanalysis basics

Tuples vs. ARX

Application to Skein

Application to BLAKE

5 names for a similar attack strategy:

square / saturation / integral / internal collision / multiset

See Biryukov/Shamir, J. Crypt. 23(4), 2010

Exploit propagation of **multiset properties**

A multiset is a set with multiplicities, e.g.

$$\begin{aligned}\{0,0,2,3,3,3,3,6\} &= \{3,6,2,0,3,3,0,3\} \\ &= \{(0,2),(2,1),(3,4),(6,1)\}\end{aligned}$$

Multiset cryptanalysis often uses 256-element byte multisets

Some multiset properties:

- **C** (constant), e.g. $\{7, 7, 7, \ldots, 7, 7\}$
- **P** (permutation), e.g. $\{0, 1, 2, \ldots, 254, 255\}$
- **E** (even multiplicities), e.g. $\{0, 0, 1, 1, \ldots, 127, 127\}$
- **A** (ADD-balanced), e.g. $\{x_0, x_1, \ldots, x_{254}, -\sum_{i=0}^{254} x_i\}$
- **B** (XOR-balanced), e.g. $\{x_0, x_1, \ldots, x_{254}, \bigoplus_{i=0}^{254} x_i\}$
- **F** (sums to 2^{w-1})

C and **E** preserved by arbitrary functions

P preserved by bijective functions

A(**B**) preserved by ADD-linear (XOR-linear) maps

Etc.

Tuples = **ordered** multisets

$$(0, 1, \ldots, 254, 255) \neq (255, 254, \ldots, 1, 0)$$

Ordering makes a big difference in ARX analysis, because of binary operators ($+, \oplus$) rather than unary S-boxes (à la SASAS)

Notations, for tuples $T = (T_0, \ldots, T_{255})$ and $S = (S_0, \ldots, S_{255})$:

- $\mathbf{C}(T) \Rightarrow \mathbf{B}(T)$
- $T + S = (T_0 + S_0, \ldots, T_{255} + S_{255})$
- $\mathbf{C}(T) \wedge \mathbf{P}(S) \Rightarrow \mathbf{P}(T + S)$

Tuple properties independent of the word size
$\Rightarrow$ properties of 8-bit reduced Skein extend to 64-bit version

$\mathbf{C} + \mathbf{P} = \mathbf{P}$, e.g. $(2, 2, \ldots, 2) + (0, 1, \ldots, 255) = (2, 3, \ldots, 255, 0, 1)$
$\mathbf{P} \ggg n = \mathbf{P}$
$\mathbf{B} \ggg n = \mathbf{B}$ (tuple elements XOR to zero)
$\mathbf{A} \ggg n \neq \mathbf{A}$ (due to carries, doesnt ADD to zero)

$\mathbf{P} + \mathbf{P} = \mathbf{A}$: let T, S be $\mathbf{P}$ tuples,

$$\sum_{i=0}^{255}(T_i + S_i) = \sum_{i=0}^{255} i + \sum_{i=0}^{255} i = 128 + 128 \equiv 0$$

Corollary: $\mathbf{P} + \mathbf{P} \neq \mathbf{P}$

Generalizes to 2^w-element tuples of w-bit elements…

Let T be a **P** tuple, and S st $S_i = -T_i$, $i = 0, \ldots, 255$:

$$\mathbf{E}(T \oplus S)$$

$i \oplus (-i)$ occurs twice for all i's, thus no odd multiplicity

---

If $T^0, T^1, \ldots, T^{2n}$ are $2n+1$ tuples, then we have

$$\mathbf{P}\left(\sum_{i=0}^{2n} T^i\right)$$

because $(2n+1)$ is coprime with 2^w (e.g. 256) and thus all $i \times (2n+1)$ are distinct

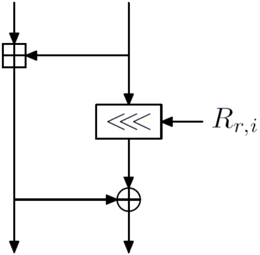
X: unidentified/no property

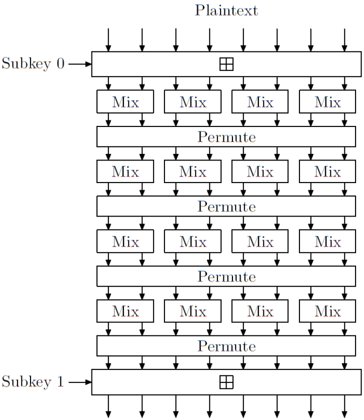
+	**A**	**B**	**C**	**E**	**F**	**P**
A	**A**	X	**A**	X	**F**	**F**
B	X	X	X	X	X	X
C	**A**	X	X	**E**	**F**	**P**
E	X	X	**E**	X	X	X
F	**F**	X	**F**	X	**A**	**A**
P	**F**	X	**P**	X	**A**	**A**

$\oplus$	**A**	**B**	**C**	**E**	**F**	**P**
A	X	X	X	X	X	X
B	X	**B**	**B**	**B**	X	**B**
C	X	**B**	**C**	**E**	X	**P**
E	X	**B**	**B**	**B**	X	**B**
F	X	X	X	X	X	X
P	X	**B**	**P**	**B**	X	**B**

$\ggg$	**A**	**B**	**C**	**E**	**F**	**P**
n	X	**B**	**C**	**E**	X	**P**

Tuples vs. 
Skein


$R_{r,i}$


Plaintext
Subkey 0
Mix
Mix
Mix
Mix
Permute
Mix
Mix
Mix
Mix
Permute
Mix
Mix
Mix
Mix
Permute
Mix
Mix
Mix
Mix
Permute
Subkey 1

$$\mathbf{MIX} : (x, y) \mapsto (x + y, (x + y) \oplus (y \ggg R))$$

Because a **P** tuples satisfies **B** (XOR-balance):

$$\mathbf{MIX}(\mathbf{C}, \mathbf{P}) = (\mathbf{C} + \mathbf{P}, (\mathbf{C} + \mathbf{P}) \oplus (\mathbf{P} \ggg r)) = (\mathbf{P}, \mathbf{P} \oplus \mathbf{P}) = (\mathbf{P}, \mathbf{B})$$

$\mathbf{A} \oplus \mathbf{P} = \mathbf{X}\ldots$

$$\mathbf{MIX}(\mathbf{P}, \mathbf{P}) = (\mathbf{P} + \mathbf{P}, (\mathbf{P} + \mathbf{P}) \oplus (\mathbf{P} \ggg r)) = (\mathbf{A}, \mathbf{X})$$

MIX	A	B	C	E	F	P
A	AX	XX	AX	XX	FX	FX
B	XX	XX	XX	XX	XX	XX
C	AX	XX	CC	EB	FX	PB
E	XX	XX	EE	XX	XX	XX
F	FX	XX	FX	XX	AX	AX
P	FX	XX	PP	XX	AX	AX

MIX^{-1}	A	B	C	E	F	P
A	XX	XX	XX	XX	XX	XX
B	XX	XB	XB	XB	XX	XB
C	XX	XB	CC	EE	XX	PP
E	XX	XB	XE	XX	XX	XB
F	XX	XX	XX	XX	XX	XX
P	XX	XB	AP	XB	XX	XB

Direct extension of **MIX** transformation rules to Threefish rounds

Simple inside-out known-key distinguishers

Theory vs. practice:

0	XX	XX	**PP**	**AP**
1	**CC**	**CC**	**PP**	XX
2	**CC**	**CC**	**AP**	**CC**
3	**CC**	**PC**	**CC**	**CC**
4	**PC**	**CC**	**CC**	**CP**
5	**CP**	**CB**	**PC**	**PC**
6	X**B**	**PB**	**PP**	**P**X
7	XX	**A**X	XX	XX

0	**BA**	XX	**PP**	**AP**
1	**CC**	**CC**	**PP**	XX
2	**CC**	**CC**	**BP**	**CC**
3	**CC**	**PC**	**CC**	**CC**
4	**PC**	**CC**	**CC**	**CP**
5	**CP**	**CB**	**PC**	**PC**
6	**FB**	**PP**	**PP**	**P**X
7	**E**X	**E**X	X**B**	**AB**
8	XX	XX	**F**X	XX

Local analysis overlooks properties due to structural dependencies…

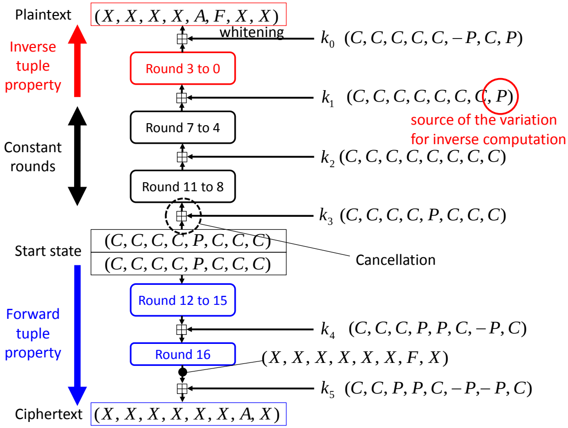
Threefish-1024:

0	XX	**AP**	**PP**	XX	XX	XX	**PP**	XX
1	**CC**	**PP**	XX	**CC**	**CC**	**CC**	**AP**	XX
2	**CC**	**CC**	**CC**	XX	**CC**	**PP**	**CC**	**CC**
3	**CC**	**CC**	**AP**	**CC**	**CC**	**CC**	**CC**	**CC**
4	**CC**	**CC**	**CC**	**PC**	**CC**	**CC**	**CC**	**CC**
5	**CC**	**CC**	**PC**	**CC**	**CP**	**CC**	**CC**	**CC**
6	**CB**	**CC**	**CC**	**PC**	**CC**	**CC**	**CP**	**PC**
7	**FC**	**CB**	**PC**	**CP**	**CP**	**PC**	**PC**	**C**X
8	**FF**	**BP**	**PP**	**P**X	**PF**	**PF**	X**P**	**PF**
9	**A**X	**BB**	XX	**E**X	**A**X	**BB**	**BB**	**B**X
10	XX	XX	XX	XX	XX	**F**X	XX	XX

Extension to chosen-key distinguisher

Exploit subkey difference cancellation, as in previous works

17 rounds attacked in 2^{64}



Tuples vs. BLAKE

ChaCha-inspired **G** core function:

$$
\begin{aligned}
a &\leftarrow a + b + (m_{\sigma_r(2i)} \oplus c_{\sigma_r(2i+1)}) \\
d &\leftarrow (d \oplus a) \ggg 16 \\
c &\leftarrow c + d \\
b &\leftarrow (b \oplus c) \ggg 12 \\
a &\leftarrow a + b + (m_{\sigma_r(2i+1)} \oplus c_{\sigma_r(2i)}) \\
d &\leftarrow (d \oplus a) \ggg 8 \\
c &\leftarrow c + d \\
b &\leftarrow (b \oplus c) \ggg 7
\end{aligned}
$$

A round applies **G** to the 4 columns then to the 4 diagonals of the 4×4 state

G tuples transformations:

CCPC $\mapsto$ **CPPC** $\mapsto$ **PXAP**:

$$\begin{aligned} a &\leftarrow \mathbf{C} + \mathbf{C} + \mathbf{C} = \mathbf{C} \\ d &\leftarrow (\mathbf{C} \oplus \mathbf{C}) \ggg 16 = \mathbf{C} \\ c &\leftarrow \mathbf{P} + \mathbf{C} = \mathbf{P} \\ b &\leftarrow (\mathbf{C} \oplus \mathbf{P}) \ggg 12 = \mathbf{P} \end{aligned}$$

$$\begin{aligned} a &\leftarrow \mathbf{C} + \mathbf{P} + \mathbf{C} = \mathbf{P} \\ d &\leftarrow (\mathbf{C} \oplus \mathbf{P}) \ggg 16 = \mathbf{P} \\ c &\leftarrow \mathbf{P} + \mathbf{P} = \mathbf{A} \\ b &\leftarrow (\mathbf{P} \oplus \mathbf{A}) \ggg 12 = \mathbf{X} \end{aligned}$$

PCCC $\mapsto$ **PPPP** $\mapsto$ **AXXX**
CPCC $\mapsto$ **PPPP** $\mapsto$ **AXXX**
CCCP $\mapsto$ **CPPP** $\mapsto$ **PXXB**

Best choice of starting tuple is **CCPC**?

Best $\mathbf{G}^{-1}$ choice: **PCCC** $\mapsto$ **PCCP** $\mapsto$ **PCPB**

2.5-round inside-out known-key dist'er

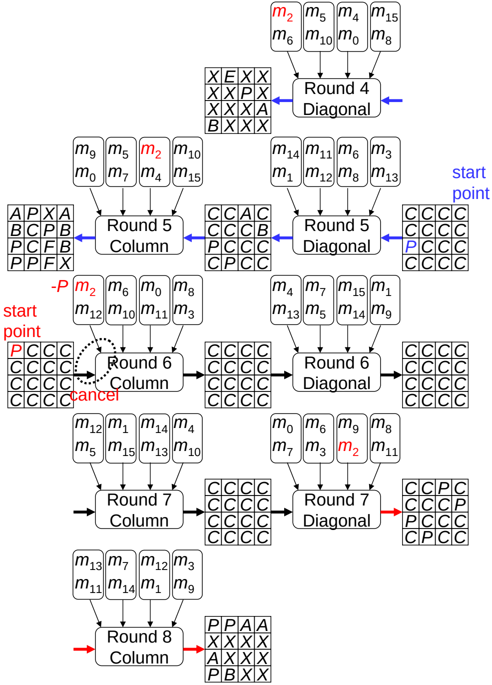
Backwards: 1.5 round

$$\begin{pmatrix} \mathbf{X} & \mathbf{E} & \mathbf{X} & \mathbf{X} \\ \mathbf{X} & \mathbf{X} & \mathbf{P} & \mathbf{X} \\ \mathbf{A} & \mathbf{X} & \mathbf{X} & \mathbf{A} \\ \mathbf{B} & \mathbf{X} & \mathbf{X} & \mathbf{X} \end{pmatrix} \leftarrow \begin{pmatrix} \mathbf{A} & \mathbf{P} & \mathbf{A} & \mathbf{A} \\ \mathbf{B} & \mathbf{P} & \mathbf{C} & \mathbf{B} \\ \mathbf{P} & \mathbf{B} & \mathbf{F} & \mathbf{C} \\ \mathbf{P} & \mathbf{P} & \mathbf{X} & \mathbf{F} \end{pmatrix} \leftarrow \begin{pmatrix} \mathbf{C} & \mathbf{C} & \mathbf{A} & \mathbf{C} \\ \mathbf{C} & \mathbf{C} & \mathbf{C} & \mathbf{B} \\ \mathbf{P} & \mathbf{C} & \mathbf{C} & \mathbf{C} \\ \mathbf{C} & \mathbf{P} & \mathbf{C} & \mathbf{C} \end{pmatrix} \leftarrow \begin{pmatrix} \mathbf{C} & \mathbf{C} & \mathbf{C} & \mathbf{C} \\ \mathbf{C} & \mathbf{C} & \mathbf{C} & \mathbf{C} \\ \mathbf{P} & \mathbf{C} & \mathbf{C} & \mathbf{C} \\ \mathbf{C} & \mathbf{C} & \mathbf{C} & \mathbf{C} \end{pmatrix}$$

Forwards: 1 round

$$\begin{pmatrix} \mathbf{C} & \mathbf{C} & \mathbf{C} & \mathbf{C} \\ \mathbf{C} & \mathbf{C} & \mathbf{C} & \mathbf{C} \\ \mathbf{P} & \mathbf{C} & \mathbf{C} & \mathbf{C} \\ \mathbf{C} & \mathbf{C} & \mathbf{C} & \mathbf{C} \end{pmatrix} \rightarrow \begin{pmatrix} \mathbf{P} & \mathbf{C} & \mathbf{C} & \mathbf{C} \\ \mathbf{X} & \mathbf{C} & \mathbf{C} & \mathbf{C} \\ \mathbf{A} & \mathbf{C} & \mathbf{C} & \mathbf{C} \\ \mathbf{P} & \mathbf{C} & \mathbf{C} & \mathbf{C} \end{pmatrix} \rightarrow \begin{pmatrix} \mathbf{A} & \mathbf{P} & \mathbf{X} & \mathbf{X} \\ \mathbf{X} & \mathbf{X} & \mathbf{X} & \mathbf{X} \\ \mathbf{X} & \mathbf{X} & \mathbf{X} & \mathbf{X} \\ \mathbf{B} & \mathbf{X} & \mathbf{X} & \mathbf{X} \end{pmatrix}$$

Some **X**'s may still have some detectable structure…


Round 4
Diagonal
start
point
Round 5
Column
Round 5
Diagonal
-P
start
point
Round 6
Column
cancel
Round 6
Diagonal
Round 7
Column
Round 7
Diagonal
Round 8
Column

4 rounds?

Recap:

- Tuple attacks extend integral et al. attacks
- Efficiently verifiable on word-reduced versions
- Correctness empir'y and analyt'y verifiable
- Efficient attacks (2^{64} for Skein, 2^{32} for BLAKE)
- Only used as bananas, but potential key-recovery

Todo:

- Bit-level refinements (à la Z'aba et al. [FSE08])
- Verify/extend attacks on Skein and BLAKE
- Detect and trace more properties?

Tuple cryptanalysis of ARX with application to BLAKE and Skein

J.-Ph. Aumasson G. Leurent W. Meier F. Mendel
N. Mouha R. C-W Phan Y. Sasaki P. Susil