PROPAGATION OF SUBSPACES IN PRIMITIVES WITH MONOMIAL SBOXES
APPLICATIONS TO RESCUE AND VARIANTS OF THE AES

Aurélien Boeuf\textsuperscript{1}, Anne Canteaut\textsuperscript{1}, Léo Perrin\textsuperscript{1}

\textsuperscript{1}Inria Paris

Journées C2 2023, Najac
Which Round Functions?

- $s_i \in \mathbb{F}_q$ (finite field of size $q$).

The round function of an SPN (Substitution-Permutation Network) Block Cipher. Design basis for the AES, very popular.
**Rescue [AABDS’20]**

- Defined in $\mathbb{F}_p$ with $p$ prime $\simeq 2^{64}$ (unusually big!).

![Diagram of Rescue algorithm]

2 rounds of RESCUE (repeated $N \simeq 10$ times).

- Defined for any MDS matrix $M$ and round constants $c_i$.  

[Diagram showing the algorithm with input states $S_0$, $S_1$, $S_2$, and output states, with intermediate operations and round constants $c_0$, $c_1$, $c_2$, $c_3$, $c_4$, $c_5$.]
**Definition**

Differential uniformity of a function $F$:

$$\delta(F) = \max_{\sigma \neq 0, \beta} \left| \left\{ F(x + \sigma) - F(x) = \beta \text{ s.t. } x \in (\mathbb{F}_p)^m \right\} \right|$$
**Definition**

Differential uniformity of a function $F$:

$$
\delta(F) = \max_{\sigma \neq 0, \beta} \left| \{ F(x + \sigma) - F(x) = \beta \ \text{s.t.} \ x \in (\mathbb{F}_p)^m \} \right|
$$

→ This quantity must be minimized.
High Differential Uniformities in Rescue

Graph taken from eprint.iacr.org/2020/820.
High Differential Uniformities in Rescue

Graph taken from eprint.iacr.org/2020/820.
The cause? Affine spaces of dimension 1 nicely mapping from one to another.

\[ \begin{pmatrix} z \\ X \end{pmatrix} \xrightarrow{2 \text{ rounds}} \begin{pmatrix} aX + b \\ cX + d \end{pmatrix} \xrightarrow{2 \text{ rounds}} \begin{pmatrix} eX + f \\ gX + h \end{pmatrix} \]
High Differential Uniformities in Rescue

The cause? Affine spaces of dimension 1 nicely mapping from one to another.

\[
\begin{pmatrix} z \\ X \end{pmatrix} \xrightarrow{\text{2 rounds}} \begin{pmatrix} aX + b \\ cX + d \end{pmatrix} \xrightarrow{\text{2 rounds}} \begin{pmatrix} eX + f \\ gX + h \end{pmatrix}
\]

- 1 round or 3 rounds: the function is not affine.
- Because \( p \) is big (\( \geq 2^{64} \)), affine spaces of dim 1 are also big.
Structure of our work

High Differential Uniformities in Rescue

Affine Space Chains
Affine Space Chains

Note $a + \langle v \rangle := \{ a + Xv \text{ such that } X \in \mathbb{F}_p \}$.

$$a_0 + \langle v_0 \rangle \xrightarrow{f} a_1 + \langle v_1 \rangle \xrightarrow{f} \ldots \xrightarrow{f} a_N + \langle v_N \rangle$$
**Main Observation**

![Diagram with labeled inputs and output]

**Rescue round.**

Write elements of \(
\begin{pmatrix}
0 \\
0 \\
a
\end{pmatrix}
\) + \(v\) \(\langle 1 \rangle\) as \(\begin{pmatrix} s_0 \\ s_1 \\ s_2 \end{pmatrix} = \begin{pmatrix} X \\ vX \\ a \end{pmatrix}\).
Main Observation

Rescue round.

\[
\begin{pmatrix}
    s_0 \\
    s_1 \\
    s_2
\end{pmatrix}
= \begin{pmatrix}
    X \\
    \nu X \\
    a
\end{pmatrix}
\rightarrow
\begin{pmatrix}
    X^\alpha \\
    \nu^\alpha X^\alpha \\
    a^\alpha
\end{pmatrix}
= \begin{pmatrix}
    0 \\
    0 \\
    a^\alpha
\end{pmatrix}
+ X^\alpha \begin{pmatrix}
    1 \\
    \nu^\alpha \\
    0
\end{pmatrix}
\]

This is the most important part! It only relies on the fact that the Sbox is a monomial.
**Definition**

An affine space of dimension 1 is **separable** if and only if there exists a representation of it denoted $a + \langle v \rangle$ such that:

\[
\forall 1 \leq i \leq m, \ a_i \cdot v_i = 0.
\]

or, equivalently, $\text{supp}(v) \cap \text{supp}(a) = \emptyset$. 

• $(a_0, b) + \langle 0, b \rangle$ is a separable affine space for all $a$ and $b$.

• $(0, 1) + \langle 1, 1 \rangle$ is not.
**Definition**

An affine space of dimension 1 is **separable** if and only if there exists a representation of it denoted \( \mathbf{a} + \langle \mathbf{v} \rangle \) such that:

\[
\forall 1 \leq i \leq m, \quad a_i \cdot v_i = 0.
\]

or, equivalently, \( \text{supp}(\mathbf{v}) \cap \text{supp}(\mathbf{a}) = \emptyset \).

**Examples**

- \( \begin{pmatrix} a \\ 0 \end{pmatrix} + \langle \begin{pmatrix} 0 \\ b \end{pmatrix} \rangle \) is a separable affine space for all \( a \) and \( b \).
**Separable Affine Spaces**

**Definition**

An affine space of dimension 1 is **separable** if and only if there exists a representation of it denoted \( a + \langle v \rangle \) such that:

\[
\forall 1 \leq i \leq m, \ a_i \cdot v_i = 0.
\]

or, equivalently, \( \text{supp}(v) \cap \text{supp}(a) = \emptyset \).

**Examples**

- \( \begin{pmatrix} a \\ 0 \end{pmatrix} + \langle \begin{pmatrix} 0 \\ b \end{pmatrix} \rangle \) is a separable affine space for all \( a \) and \( b \).
- \( \begin{pmatrix} 0 \\ 1 \end{pmatrix} + \langle \begin{pmatrix} 1 \\ 1 \end{pmatrix} \rangle \) is not.
**Main Observation**

Rescue round.

\[
\begin{pmatrix}
0 \\
0 \\
a^\alpha
\end{pmatrix}
+ X^\alpha
\begin{pmatrix}
1 \\
v^\alpha \\
0
\end{pmatrix}
\rightarrow
M
\begin{pmatrix}
0 \\
0 \\
a^\alpha
\end{pmatrix}
+ X^\alpha M
\begin{pmatrix}
1 \\
v^\alpha \\
0
\end{pmatrix}
\]
**Main Observation**

Rescue round.

\[
M \begin{pmatrix} 0 \\ 0 \\ a^\alpha \end{pmatrix} + X^\alpha M \begin{pmatrix} 1 \\ v^\alpha \\ 0 \end{pmatrix} \rightarrow M \begin{pmatrix} 0 \\ 0 \\ a^\alpha \end{pmatrix} + \begin{pmatrix} c_0 \\ c_1 \\ c_2 \end{pmatrix} + X^\alpha M \begin{pmatrix} 1 \\ v^\alpha \\ 0 \end{pmatrix}
\]
Main Observation

\[
M \begin{pmatrix} 0 \\ 0 \\ a^\alpha \end{pmatrix} + \begin{pmatrix} c_0 \\ c_1 \\ c_2 \end{pmatrix} + \left\langle M \begin{pmatrix} 1 \\ \nu^\alpha \\ 0 \end{pmatrix} \right\rangle
\]
Main Observation

\[
M \begin{pmatrix} 0 \\ 0 \\ a^\alpha \end{pmatrix} + \begin{pmatrix} c_0 \\ c_1 \\ c_2 \end{pmatrix} + \left\langle M \begin{pmatrix} 1 \\ \nu^\alpha \\ 0 \end{pmatrix} \right\rangle
\]

For this space to be separable, we need that there exists \( \lambda \in \mathbb{F}_p \) such that

\[
M \begin{pmatrix} 1 \\ \nu^\alpha \\ 0 \end{pmatrix} \quad \text{and} \quad M \begin{pmatrix} 0 \\ 0 \\ a^\alpha \end{pmatrix} + \begin{pmatrix} c_0 \\ c_1 \\ c_2 \end{pmatrix} + \lambda M \begin{pmatrix} 1 \\ \nu^\alpha \\ 0 \end{pmatrix}
\]

have disjoint supports.
Main Result

**Theorem**

The image of a separable affine space \( \mathbf{a} + \langle \mathbf{v} \rangle \) by a round of a monomial SPN is an affine space. Also, the image is still separable if and only if there exists \( \lambda \) in \( \mathbb{F}_p \) such that:

\[
\forall i \in \text{supp}(M \circ S)(\mathbf{v}), c_i = \lambda (M \circ S)(\mathbf{v})_i - (M \circ S)(\mathbf{a})_i
\]
Theorem

The image of a separable affine space \( \mathbf{a} + \langle \mathbf{v} \rangle \) by a round of a monomial SPN is an affine space. Also, the image is still separable if and only if there exists \( \lambda \) in \( \mathbb{F}_p \) such that:

\[
\forall i \in \text{supp}(M \circ S)(\mathbf{v}),
\]

\[
c_i = \lambda (M \circ S)(\mathbf{v})_i - (M \circ S)(\mathbf{a})_i;
\]
This differential uniformity graph spells “. . -. -. --. . -. . -. - -. -” (ILOVENAJAC) over 71 rounds ($m = 2$, $\mathbb{F}_{2^6}$).
Conclusion

• Bad choice of round constants may lead to affine space chains or high differential uniformities.
• It's possible to define "backdoored" primitives that enforce this kind of behaviour.
• Such weak designs satisfy state-of-the-art security arguments (APN Sbox, MDS matrix, wide-trail strategy...). Usual security arguments are not sufficient in the AO context.
• Look out for similar algebraic patterns in AO primitives; they can improve algebraic attacks.
**Conclusion**

- Bad choice of round constants may lead to affine space chains or high differential uniformities.
**Conclusion**

- Bad choice of round constants may lead to affine space chains or high differential uniformities.
- It’s possible to define “backdoored” primitives that enforce this kind of behaviour.
Conclusion

• Bad choice of round constants may lead to affine space chains or high differential uniformities.

• It’s possible to define “backdoored” primitives that enforce this kind of behaviour.

• Such weak designs satisfy state-of-the art security arguments (APN Sbox, MDS matrix, wide-trail strategy...). Usual security arguments are not sufficient in the AO context.
Conclusion

• Bad choice of round constants may lead to affine space chains or high differential uniformities.

• It’s possible to define “backdoored” primitives that enforce this kind of behaviour.

• Such weak designs satisfy state-of-the art security arguments (APN Sbox, MDS matrix, wide-trail strategy...). Usual security arguments are not sufficient in the AO context.

• Look out for similar algebraic patterns in AO primitives; they can improve algebraic attacks.
Thank you for listening!

Questions?
High Differential Uniformities in Rescue

\[ \delta(F) = \max_{\sigma \neq 0, \beta} |\{ F(x + \sigma) - F(x) = \beta \text{ s.t. } x \in (\mathbb{F}_p)^m \}|. \]

\[ \forall X \in \mathbb{F}_p, F \begin{pmatrix} z \\ X \end{pmatrix} = \begin{pmatrix} eX + f \\ gX + h \end{pmatrix}. \]
\[
\delta(F) = \max_{\sigma \neq 0, \beta} \left| \{ F(x + \sigma) - F(x) = \beta \ \text{s.t.} \ x \in (\mathbb{F}_p)^m \} \right|.
\]

\[
\forall X \in \mathbb{F}_p, F \begin{pmatrix} z \\ X \end{pmatrix} = \begin{pmatrix} eX + f \\ gX + h \end{pmatrix}.
\]

\[
F \begin{pmatrix} z \\ X + 1 \end{pmatrix} - F \begin{pmatrix} z \\ X \end{pmatrix} = \begin{pmatrix} e(X + 1) + f \\ g(X + 1) + h \end{pmatrix} - \begin{pmatrix} eX + f \\ gX + h \end{pmatrix}
\]

\[
= \begin{pmatrix} e \\ g \end{pmatrix} = \beta
\]
High Differential Uniformities in Rescue

\[ \delta(F) = \max_{\sigma \neq 0, \beta} |\{ F(x + \sigma) - F(x) = \beta \ \text{s.t.} \ x \in (\mathbb{F}_p)^m \}|. \]

\[ \forall X \in \mathbb{F}_p, F \begin{pmatrix} z \\ X \end{pmatrix} = \begin{pmatrix} eX + f \\ gX + h \end{pmatrix}. \]

\[ F \begin{pmatrix} z \\ X + 1 \end{pmatrix} - F \begin{pmatrix} z \\ X \end{pmatrix} = \begin{pmatrix} e(X + 1) + f \\ g(X + 1) + h \end{pmatrix} - \begin{pmatrix} eX + f \\ gX + h \end{pmatrix} \]

\[ = \begin{pmatrix} e \\ g \end{pmatrix} = \beta \]

\[ \rightarrow \delta(F) \geq p \]
Arithmetization-Oriented Symmetric Primitives

- Term coined for the first time in a 2020 paper from Aly et al.
- Symmetric primitives with a “simple” arithmetic description.
- Minimize verification cost in Zero-Knowledge schemes and other advanced protocols.
- Generally defined over a large finite field $\mathbb{F}_q$. ($q \geq 2^{64}$ or so.)
- Heavy use of monomials for nonlinear functions as random permutations are hard to analyze.
Arithmetization-Oriented Symmetric Primitives

- Term coined for the first time in a 2020 paper from Aly et al.
- Symmetric primitives with a “simple” arithmetic description.
- Minimize verification cost in Zero-Knowledge schemes and other advanced protocols.
- Generally defined over a large finite field $\mathbb{F}_q$. ($q \geq 2^{64}$ or so.)
- Heavy use of monomials for nonlinear functions as random permutations are hard to analyze.

**Example**

Primitives using the nonlinear component $S : x \mapsto x^3$ (MIMC and variants, RESCUE...).
Rescue’s Design Choices

- Alternate $x^\alpha$ and $x^{\alpha^1}$ for resistance against algebraic attacks.
- $x^\alpha$ has good cryptographic properties (APN for $\alpha = 3$).
- Wide-trail strategy is used, like in the AES, as a security argument.
- For the Sbox, having a monomial followed by an affine transformation of the representation like in the AES may be nice, but... no subfield in $\mathbb{F}_p$.

Main motivation: Are the usual security arguments sufficient?
Rescue’s Design Choices

- Alternate $x^\alpha$ and $x^{\frac{1}{\alpha}}$ for resistance against algebraic attacks.
Rescue’s Design Choices

- Alternate $x^\alpha$ and $x^{\frac{1}{\alpha}}$ for resistance against algebraic attacks.
- $x^\alpha$ has good cryptographic properties (APN for $\alpha = 3$).
Rescue’s Design Choices

- Alternate $x^\alpha$ and $\frac{1}{\alpha}$ for resistance against algebraic attacks.
- $x^\alpha$ has good cryptographic properties (APN for $\alpha = 3$).
- Wide-trail strategy is used, like in the AES, as a security argument.

For the Sbox, having a monomial followed by an affine transformation of the representation like in the AES may be nice, but... no subfield in $\mathbb{F}_p$. Main motivation: Are the usual security arguments sufficient?
Rescue’s Design Choices

- Alternate $x^\alpha$ and $x^{\frac{1}{\alpha}}$ for resistance against algebraic attacks.
- $x^\alpha$ has good cryptographic properties (APN for $\alpha = 3$).
- Wide-trail strategy is used, like in the AES, as a security argument.
- For the Sbox, having a monomial followed by an affine transformation of the representation like in the AES may be nice, but... no subfield in $\mathbb{F}_p$. 
Rescue’s Design Choices

- Alternate $x^\alpha$ and $x^{\frac{1}{\alpha}}$ for resistance against algebraic attacks.
- $x^\alpha$ has good cryptographic properties (APN for $\alpha = 3$).
- Wide-trail strategy is used, like in the AES, as a security argument.
- For the Sbox, having a monomial followed by an affine transformation of the representation like in the AES may be nice, but... no subfield in $\mathbb{F}_p$.

Main motivation: Are the usual security arguments sufficient?
Our Designs

- **Stir**, a weak instance of **Rescue**.

---

1Thomas Peyrin and Haoyang Wang, *The MALICIOUS Framework: Embedding Backdoors into Tweakable Block Ciphers*
Our Designs

- **Stir**, a weak instance of **Rescue**.
- **Snare**, a tweakable cipher with a secret weak tweak. Directly based on the MALICIOUS framework \(^1\).

---

\(^1\)Thomas Peyrin and Haoyang Wang, *The MALICIOUS Framework: Embedding Backdoors into Tweakable Block Ciphers*
Our Designs

- **Stir**, a weak instance of **Rescue**.
- **Snare**, a tweakable cipher with a secret weak tweak. Directly based on the MALICIOUS framework\(^1\).

---

\(^1\)Thomas Peyrin and Haoyang Wang, *The MALICIOUS Framework: Embedding Backdoors into tweakable Block Ciphers*
Our Designs

- **Stir**, a weak instance of **Rescue**.
- **Snare**, a tweakable cipher with a secret weak tweak. Directly based on the MALICIOUS framework\(^1\).
- AES-like ciphers where we can introduce and control differential uniformity spikes.

\(^1\)Thomas Peyrin and Haoyang Wang, *The MALICIOUS Framework: Embedding Backdoors into Tweakable Block Ciphers*
- Based on RESCUE.
- MDS matrix $M$ and round constants $r$ are carefully chosen to impose one affine space chain over the whole permutation.
\[
\begin{pmatrix}
0 \\
0 \\
0
\end{pmatrix} + \left\langle \begin{pmatrix}
v_1 \\
v_2 \\
0
\end{pmatrix} \right\rangle \longrightarrow \begin{pmatrix} 0 \\ 0 \\ a_3 \end{pmatrix} + \left\langle \begin{pmatrix} v'_1 \\ v'_2 \\ 0 \end{pmatrix} \right\rangle \longrightarrow \ldots \longrightarrow \begin{pmatrix} 0 \\ 0 \\ 0 \end{pmatrix} + \left\langle \begin{pmatrix} v''_1 \\ v''_2 \\ 0 \end{pmatrix} \right\rangle
\]

- Yields \( p \approx 2^{64} \) solutions to the “CICO problem”. This breaks security arguments in sponge constructions.
**Snare**

- $H$ is some hash function, like SHAKE256.
- The $t_i$ are the tweak hashes.
Idea: Choose $r_i = -H(T^*)_i$ for some secret tweak $T^*$.

→ When $T = T^*$, $r_i$ and $t_i$ annihilate one another and an invariant vector space appears.
Snare

\[ \langle \begin{pmatrix} 1 \\ \rho \\ 0 \end{pmatrix} \rangle \xrightarrow{1 \text{ round}} \langle \begin{pmatrix} 1 \\ \rho \\ 0 \end{pmatrix} \rangle \rightarrow \ldots \rightarrow \langle \begin{pmatrix} 1 \\ \rho \\ 0 \end{pmatrix} \rangle \]
\begin{equation*}
\begin{pmatrix}
1 \\
\rho \\
0
\end{pmatrix}
\xrightarrow{1 \text{ round}}
P_1(K_0)
\begin{pmatrix}
1 \\
\rho \\
0
\end{pmatrix}
\rightarrow \ldots 
\rightarrow
P_n(K_0)
\begin{pmatrix}
1 \\
\rho \\
0
\end{pmatrix}
\end{equation*}
\[
\begin{pmatrix}
1 \\
\rho \\
0
\end{pmatrix} \xrightarrow{\text{1 round}} P_1(K_0) \begin{pmatrix}
1 \\
\rho \\
0
\end{pmatrix} \longrightarrow \ldots \longrightarrow P_n(K_0) \begin{pmatrix}
1 \\
\rho \\
0
\end{pmatrix}
\]

- Retrieve $K_0$ with multivariate polynomial solving (Gröbner bases), with $m$ times less equations as the general case.

→ Algebraic attack complexity put to the $m$th root!
Affine Space Chain vs Affine Function

- Last design is based on affine space chains.
- Having an affine space chain doesn’t mean that the function itself is affine.
- In the beginning we measured high differential uniformities because the function itself is affine on these subspaces.
- Can we recreate that?
Affine Space Chain vs Affine Function

- Last design is based on affine space chains.
- Having an affine space chain doesn’t mean that the function itself is affine.
- In the beginning we measured high differential uniformities because the function itself is affine on these subspaces.
- Can we recreate that?

\[
a_1 + Xv_1 \rightarrow a_2 + (X^\alpha + \lambda)v_2 \rightarrow a_3 + (X^\alpha + \lambda)^{\frac{1}{\alpha}}v_3
\]
Morse Code with Differential Uniformity

- Same thing as **Snare**, but with elements over $\mathbb{F}_{2^n}$ and the inverse function $x \mapsto x^{-1}$ as an Sbox.
Morse Code with Differential Uniformity

Idea: Same strategy as Snare, but make it so that the mapping from the input to output affine space is itself affine every 2 or 3 rounds!
Morse Code with Differential Uniformity

Idea: Same strategy as Snare, but make it so that the mapping from the input to output affine space is itself affine every 2 or 3 rounds!

- For a 2-round delay, the coefficient $X$ of the affine space basis verifies $X \rightarrow X^{-1} \rightarrow X$ (Case $\lambda = 0$).
- High differential uniformity every 2 or 3 rounds (controlled by our choices of $r_i$).