The Best Biclique Cryptanalysis of the Lightweight Cipher FUTURE

Gabriel C. de Carvalho
Luis A. B. Kowada

gabrielcc@id.uff.br
luis@ic.uff.br

Sumário

Introdução

Introdução

- Uma **cifra leve** é uma cifra projetada para ser usada em ambientes muito restritos, como sistemas embarcados, dispositivos de rádio e redes de sensores.

Introdução

- Uma **cifra leve** é uma cifra projetada para ser usada em ambientes muito restritos, como sistemas embarcados, dispositivos de rádio e redes de sensores.
- **FUTURE** é um dos desenvolvimentos mais recentes nesse campo (2022).

Introdução

- Uma **cifra leve** é uma cifra projetada para ser usada em ambientes muito restritos, como sistemas embarcados, dispositivos de rádio e redes de sensores.
- **FUTURE** é um dos desenvolvimentos mais recentes nesse campo (2022).
- A cifra tem sido alvo de vários ataques nestes dois anos.

Introdução

- Um **ataque** (ou **criptoanálise**) é um algoritmo que tenta descobrir a chave secreta K a partir de pares de texto claro/cifrado.

$$P \xrightarrow[future]{K} C$$

Introdução

- Um **ataque** (ou **criptoanálise**) é um algoritmo que tenta descobrir a chave secreta K a partir de pares de texto claro/cifrado.

$$P \xrightarrow[future]{K} C$$

- A **criptoanálise biclique** é famosa por sua aplicação à cifra Rijndael (AES), sendo o primeiro ataque à sua versão completa.

Introdução

- Um **ataque** (ou **criptoanálise**) é um algoritmo que tenta descobrir a chave secreta K a partir de pares de texto claro/cifrado.

$$P \xrightarrow[future]{K} C$$

- A **criptoanálise biclique** é famosa por sua aplicação à cifra Rijndael (AES), sendo o primeiro ataque à sua versão completa.
- Foi aplicada a muitas outras cifras na última década (ARIA, PRESENT, LED, Serpent, IDEA, PICCOLO, ...).

Ataques na literatura

Ataque	Ano	Tempo	Dados	Memória	Referência
Meet-in-the-Middle	2023	2^{126}	2^{64}	2^{36}	[2]
Biclique Balanceada	2024	$2^{125.8875}$	2^{48}	2^{32}	[1]
Múltiplas Bicliques	2024	$2^{125.5365}$	2^{48}	2^{32}	[1]
Integral	2024	$2^{123.7}$	2^{63}	≈ 0	[3]
Integral	2024	2^{112}	2^{64}	≈ 0	[3]

Nossas Contribuições

- Todos os ataques atuais à FUTURE têm o mesmo problema: a **complexidade de dados** é muito grande para ser relevante.

Nossas Contribuições

- Todos os ataques atuais à FUTURE têm o mesmo problema: a **complexidade de dados** é muito grande para ser relevante.
- Este artigo apresenta o ataque biclique com **menor complexidade de tempo**, e outro que requer o mínimo de dados.

Nossas Contribuições

- Todos os ataques atuais à FUTURE têm o mesmo problema: a **complexidade de dados** é muito grande para ser relevante.
- Este artigo apresenta o ataque biclique com **menor complexidade de tempo**, e outro que requer o mínimo de dados.
- Ambos foram obtidos através da busca **semi-automatizada** por bicliques baseadas no conceito de **Conjuntos Geradores de Chave**.

Ataques na literatura

Ataque	Ano	Tempo	Dados	Memória	Referência
Meet-in-the-Middle	2023	2^{126}	2^{64}	2^{36}	[2]
Biclique Balanceada	2024	$2^{125.8875}$	2^{48}	2^{32}	[1]
Múltiplas Bicliques	2024	$2^{125.5365}$	2^{48}	2^{32}	[1]
Integral	2024	$2^{123.7}$	2^{63}	≈ 0	[3]
Integral	2024	2^{112}	2^{64}	≈ 0	[3]
CGC-Biclique	**2024**	$\mathbf{2^{125.18}}$	$\mathbf{2^{20}}$	$\approx \mathbf{0}$	**Nosso**
CGC-Estrela	**2024**	$\mathbf{2^{126.38}}$	$\mathbf{1}$	$\approx \mathbf{0}$	**Nosso**

Criptoanálise Biclique

Visão Geral

- A criptoanálise biclique possui uma **fase de preparação** e três passos que são executados em loop.

  Fase de preparação:

Visão Geral

- A criptoanálise biclique possui uma **fase de preparação** e três passos que são executados em loop.

  Fase de preparação:

- Um adversário particiona o espaço de chaves em grupos com 2^{2d} chaves para algum d.

Visão Geral

- A criptoanálise biclique possui uma **fase de preparação** e três passos que são executados em loop.

Fase de preparação:

- Um adversário particiona o espaço de chaves em grupos com 2^{2d} chaves para algum d.
- Cada grupo de chaves é associado a uma matriz $2^d \times 2^d$ K, onde cada elemento $K[i, j]$ representa uma chave no grupo (Há 2^{k-2d} grupos).

Visão Geral

- A criptoanálise biclique possui uma **fase de preparação** e três passos que são executados em loop.

Fase de preparação:

- Um adversário particiona o espaço de chaves em grupos com 2^{2d} chaves para algum d.
- Cada grupo de chaves é associado a uma matriz $2^d \times 2^d$ K, onde cada elemento $K[i, j]$ representa uma chave no grupo (Há 2^{k-2d} grupos).
- $Cifra = f \circ g \circ h$ sendo atacada é uma composição de três subcifras f, g e h.

Visão Geral

Para cada grupo de chaves:

Visão Geral

Para cada grupo de chaves:

- A **biclique é construída** sobre a subcifra f, tal que

$$\forall i, j : S_j \xrightarrow[f]{K[i,j]} C_i,$$

onde $0 \leq i, j < 2^d$, S_j são estados internos da cifra e C_i são textos cifrados.

Visão Geral

Para cada grupo de chaves:

- A **biclique é construída** sobre a subcifra f, tal que

$$\forall i, j : S_j \xrightarrow[f]{K[i,j]} C_i,$$

onde $0 \leq i, j < 2^d$, S_j são estados internos da cifra e C_i são textos cifrados.

- Este é um ataque de texto cifrado escolhido, é possível **obter os textos claros** correspondentes

$$\forall i : C_i \xrightarrow[(f \circ g \circ h)^{-1}]{oraculo\ de\ decifragem} P_i.$$

Visão Geral

Para cada grupo de chaves:

- A **biclique é construída** sobre a subcifra f, tal que

$$\forall i, j : S_j \xrightarrow[f]{K[i,j]} C_i,$$

onde $0 \leq i, j < 2^d$, S_j são estados internos da cifra e C_i são textos cifrados.

- Este é um ataque de texto cifrado escolhido, é possível **obter os textos claros** correspondentes

$$\forall i : C_i \xrightarrow[(f \circ g \circ h)^{-1}]{oraculo\ de\ decifragem} P_i.$$

- Para cada chave $K[i, j]$ no grupo, testa-se (**meet-in-the-middle**)

$$\exists i, j : P_i \xrightarrow[g \circ h]{K[i,j]} S_j.$$

Combinação com Pré-computações

Etapa de pré-computação

Combinação com Pré-computações

Etapa de pré-computação

- O adversário calcula e armazena 2.2^d computações da cifra até uma **variável intermediária** v.

$$\forall i : P_i \xrightarrow[h]{K[i,0]} v^1_{i,0} \text{ and } \forall j : v^2_{0,j} \xleftarrow[g^{-1}]{K[0,j]} S_j.$$

Combinação com Pré-computações

Etapa de pré-computação

- O adversário calcula e armazena 2.2^d computações da cifra até uma **variável intermediária** v.

$$\forall i : P_i \xrightarrow[h]{K[i,0]} v^1_{i,0} \text{ and } \forall j : v^2_{0,j} \xleftarrow[g^{-1}]{K[0,j]} S_j.$$

- Todos os estados internos e subchaves de g e h até v devem ser armazenados.

Combinação com Pré-computações

Etapa de pré-computação

- O adversário calcula e armazena 2.2^d computações da cifra até uma **variável intermediária** v.

$$\forall i : P_i \xrightarrow[h]{K[i,0]} v^1_{i,0} \text{ and } \forall j : v^2_{0,j} \xleftarrow[g^{-1}]{K[0,j]} S_j.$$

- Todos os estados internos e subchaves de g e h até v devem ser armazenados.

Etapa de recomputação

Combinação com Pré-computações

Etapa de pré-computação

- O adversário calcula e armazena 2.2^d computações da cifra até uma **variável intermediária** v.

$$\forall i : P_i \xrightarrow[h]{K[i,0]} v^1_{i,0} \text{ and } \forall j : v^2_{0,j} \xleftarrow[g^{-1}]{K[0,j]} S_j.$$

- Todos os estados internos e subchaves de g e h até v devem ser armazenados.

Etapa de recomputação

- As partes que diferem dos valores armazenados devem ser recomputadas.

Complexidades

- Este ataque é uma otimização de força bruta. Três tipos de complexidades são de interesse: **memória**, **dados** e **tempo**.

Complexidades

- Este ataque é uma otimização de força bruta. Três tipos de complexidades são de interesse: **memória**, **dados** e **tempo**.
- A **complexidade de memória** é dominada pela etapa de **pré-computação**.

Complexidades

- Este ataque é uma otimização de força bruta. Três tipos de complexidades são de interesse: **memória**, **dados** e **tempo**.
- A **complexidade de memória** é dominada pela etapa de **pré-computação**.
- A **complexidade de dados** depende apenas de quantos bits de C são afetados pelos diferenciais $-\Delta$.

Complexidades

- Este ataque é uma otimização de força bruta. Três tipos de complexidades são de interesse: **memória**, **dados** e **tempo**.
- A **complexidade de memória** é dominada pela etapa de **pré-computação**.
- A **complexidade de dados** depende apenas de quantos bits de C são afetados pelos diferenciais $-\Delta$.
- A **complexidade de tempo** é

$$C_{time} = 2^{k-2d}(C_{biclique} + C_{precomp} + C_{recomp} + C_{falpos}).$$

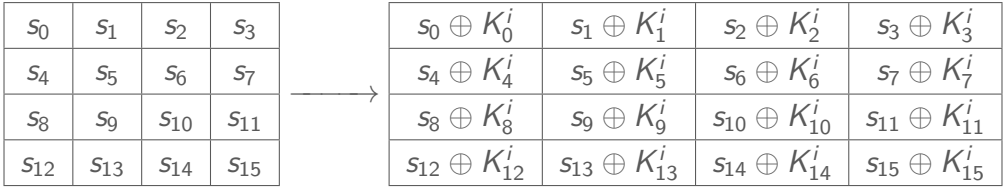
A cifra FUTURE

Estado

- É uma cifra *AES-like*.
- Possui 10 rodadas.
- Blocos de 64 bits.
- Chave de 128 bits.

s_0	s_1	s_2	s_3
s_4	s_5	s_6	s_7
s_8	s_9	s_{10}	s_{11}
s_{12}	s_{13}	s_{14}	s_{15}

AddKey (AK)


s_0 s_1 s_2 s_3
s_4 s_5 s_6 s_7
s_8 s_9 s_{10} s_{11}
s_{12} s_{13} s_{14} s_{15}
$s_0 \oplus K_0^i$ $s_1 \oplus K_1^i$ $s_2 \oplus K_2^i$ $s_3 \oplus K_3^i$
$s_4 \oplus K_4^i$ $s_5 \oplus K_5^i$ $s_6 \oplus K_6^i$ $s_7 \oplus K_7^i$
$s_8 \oplus K_8^i$ $s_9 \oplus K_9^i$ $s_{10} \oplus K_{10}^i$ $s_{11} \oplus K_{11}^i$
$s_{12} \oplus K_{12}^i$ $s_{13} \oplus K_{13}^i$ $s_{14} \oplus K_{14}^i$ $s_{15} \oplus K_{15}^i$

SubCell

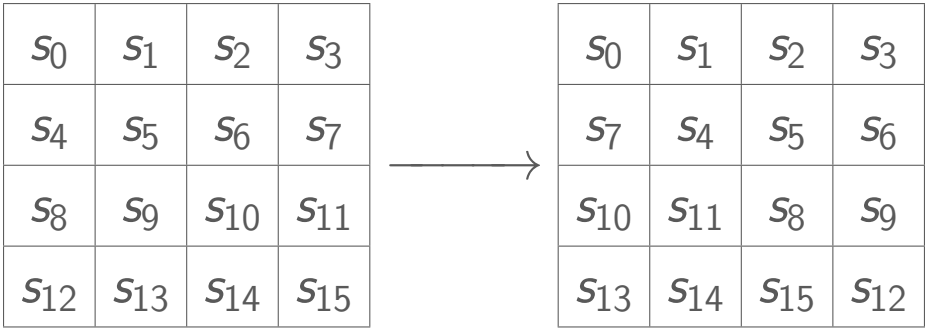
$S =$

0	1	2	3	4	5	6	7	8	9	A	B	C	D	E	F
1	3	0	2	7	E	4	D	9	A	C	6	F	5	8	B

s_0	s_1	s_2	s_3
s_4	s_5	s_6	s_7
s_8	s_9	s_{10}	s_{11}
s_{12}	s_{13}	s_{14}	s_{15}

$S(s_0)$	$S(s_1)$	$S(s_2)$	$S(s_3)$
$S(s_4)$	$S(s_5)$	$S(s_6)$	$S(s_7)$
$S(s_8)$	$S(s_9)$	$S(s_{10})$	$S(s_{11})$
$S(s_{12})$	$S(s_{13})$	$S(s_{14})$	$S(s_{15})$

ShiftRows


s_0 s_1 s_2 s_3
s_4 s_5 s_6 s_7
s_8 s_9 s_{10} s_{11}
s_{12} s_{13} s_{14} s_{15}
s_0 s_1 s_2 s_3
s_7 s_4 s_5 s_6
s_{10} s_{11} s_8 s_9
s_{13} s_{14} s_{15} s_{12}

MixColumns

- Multiplicação em $GF(2^4)$.
- Polinômio $x^4 + x + 1$.

$$\begin{pmatrix} 8 & 9 & 1 & 8 \\ 2 & 2 & 9 & 9 \\ 2 & 3 & 8 & 9 \\ 9 & 9 & 8 & 1 \end{pmatrix}$$

Sequenciamento de Chaves (*Key-Schedule*)

- O sequenciamento de chaves para esta cifra é extremamente simples.

Sequenciamento de Chaves (*Key-Schedule*)

- O sequenciamento de chaves para esta cifra é extremamente simples.
- A chave de 128 bits é particionada em duas, os 64 bits mais à esquerda se tornam X e os outros 64 se tornam Y.

Sequenciamento de Chaves (*Key-Schedule*)

- O sequenciamento de chaves para esta cifra é extremamente simples.
- A chave de 128 bits é particionada em duas, os 64 bits mais à esquerda se tornam X e os outros 64 se tornam Y.
- Então, a $i-$ésima subchave é igual a $X <<< (5 \cdot (\frac{i}{2}))$, se i for par, e $Y <<< (5 \cdot (\frac{i}{2}))$ se i for ímpar.

Sequenciamento de Chaves (*Key-Schedule*)

- O sequenciamento de chaves para esta cifra é extremamente simples.
- A chave de 128 bits é particionada em duas, os 64 bits mais à esquerda se tornam X e os outros 64 se tornam Y.
- Então, a $i-$ésima subchave é igual a $X <<< (5 \cdot (\frac{i}{2}))$, se i for par, e $Y <<< (5 \cdot (\frac{i}{2}))$ se i for ímpar.
- Não utiliza S-boxes em seu agendamento de chaves.

Composição da cifra

- Possui 10 rodadas.
- Na última aplica-se uma subchave a mais em vez de MixColumns.

$$R_i = ShiftRows \circ MixColumns \circ SubCells \circ AK_i$$

$$FUTURE = AK_{10} \circ ShiftRows \circ SubCells \circ AK_9 \circ R_8 \circ R_7 \circ ... \circ R_1 \circ R_0$$

- Os estados são indexados de tal maneira que o $i-$ésimo estado é o resultado da aplicação da $i-$ésima operação ($P = \#0$ e $C = \#40$).

Biclique Balanceada

Fase de Preparação

- A biclique escolhida é $4-$dimensional.

Fase de Preparação

- A biclique escolhida é $4-$dimensional.
- A chave é particionada em $2^{128-2\cdot 4} = 2^{120}$ grupos.

Fase de Preparação

- A biclique escolhida é $4-$dimensional.
- A chave é particionada em $2^{128-2\cdot 4} = 2^{120}$ grupos.
- FUTURE é definida como $FUTURE = f \circ g \circ h$, onde:

Fase de Preparação

- A biclique escolhida é $4-$dimensional.
- A chave é particionada em $2^{128-2\cdot 4} = 2^{120}$ grupos.
- FUTURE é definida como $\mathit{FUTURE} = f \circ g \circ h$, onde:
  - h cifra o texto simples para até o estado $\#17$,

Fase de Preparação

- A biclique escolhida é $4-$dimensional.
- A chave é particionada em $2^{128-2\cdot4} = 2^{120}$ grupos.
- FUTURE é definida como $\mathit{FUTURE} = f \circ g \circ h$, onde:
  - h cifra o texto simples para até o estado $\#17$,
  - g cifra o estado $\#17$ para o estado $\#25$ e

Fase de Preparação

- A biclique escolhida é $4-$dimensional.
- A chave é particionada em $2^{128-2\cdot4} = 2^{120}$ grupos.
- FUTURE é definida como $FUTURE = f \circ g \circ h$, onde:
  - h cifra o texto simples para até o estado $\#17$,
  - g cifra o estado $\#17$ para o estado $\#25$ e
  - f cifra o estado $\#25$ para o texto cifrado.

Fase de Preparação

- Quaisquer duas subchaves são um conjunto gerador para a chave se o índice de uma for par e o da outra for ímpar.

Fase de Preparação

- Quaisquer duas subchaves são um conjunto gerador para a chave se o índice de uma for par e o da outra for ímpar.
- As partições são tais que:

Fase de Preparação

- Quaisquer duas subchaves são um conjunto gerador para a chave se o índice de uma for par e o da outra for ímpar.
- As partições são tais que:
  - O nibble 3 de \$0 é o único nibble ativo de Δ^K e

Fase de Preparação

- Quaisquer duas subchaves são um conjunto gerador para a chave se o índice de uma for par e o da outra for ímpar.
- As partições são tais que:
  - O nibble 3 de \$0 é o único nibble ativo de Δ^K e
  - O nibble 8 de \$1 é o único nibble ativo de ∇^K.

Fase de Preparação

- Quaisquer duas subchaves são um conjunto gerador para a chave se o índice de uma for par e o da outra for ímpar.
- As partições são tais que:
  - O nibble 3 de \$0 é o único nibble ativo de Δ^K e
  - O nibble 8 de \$1 é o único nibble ativo de ∇^K.

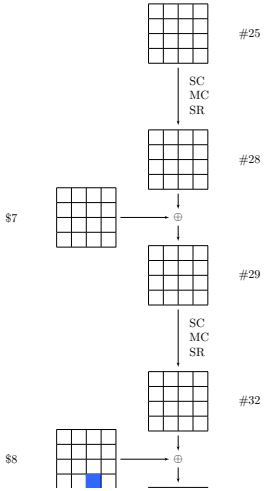
$\$0 =$

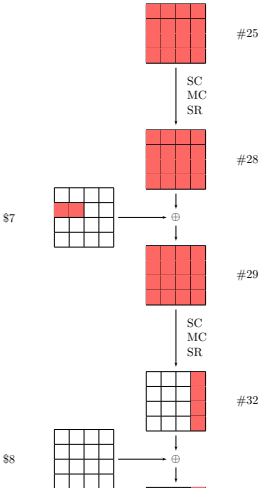
			i

$\$1 =$

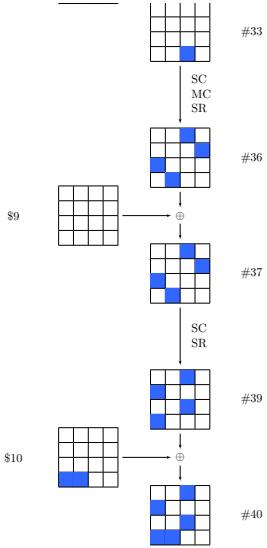
j			

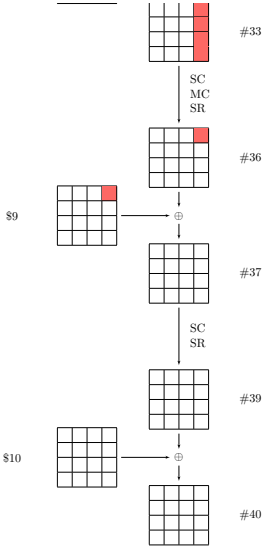
Biclique





Biclique





Combinação com pré-computações

- Aqui é checado se a chave secreta pertence a este grupo.

Combinação com pré-computações

- Aqui é checado se a chave secreta pertence a este grupo.
- A variável intermediária v é o nibble 3 do estado #17.

Combinação com pré-computações

- Aqui é checado se a chave secreta pertence a este grupo.
- A variável intermediária v é o nibble 3 do estado $\#17$.

$$P_i \xrightarrow[h]{K[i,0]} v^1_{i,0} \text{ e } v^2_{0,j} \xleftarrow[g]{K[0,j]} S_j$$

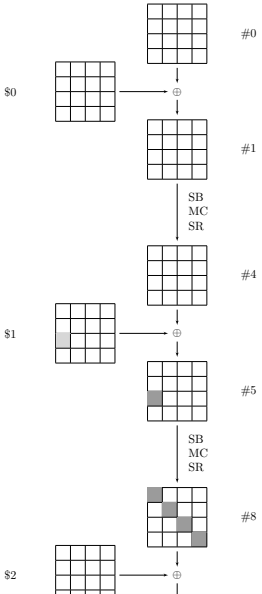
Combinação com pré-computações

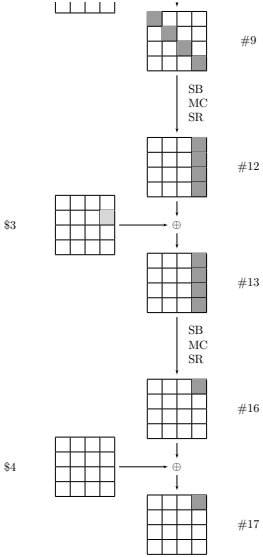
- Aqui é checado se a chave secreta pertence a este grupo.
- A variável intermediária v é o nibble 3 do estado $\#17$.

$$P_i \xrightarrow[h]{K[i,0]} v^1_{i,0} \text{ e } v^2_{0,j} \xleftarrow[g]{K[0,j]} S_j$$

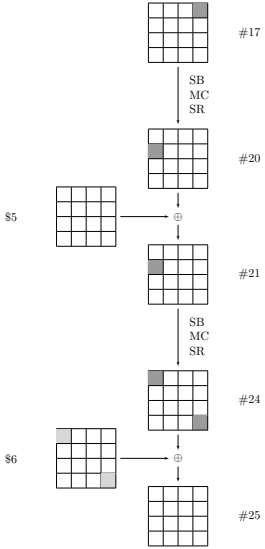
- Todos os nibbles que são afetados por ambas as diferenciais devem ser recomputados.

Recomputação de ida





Recomputação de volta



Complexidades

- A **complexidade de dados** é determinada pelo número de nibbles ativos no texto cifrado.

Complexidades

- A **complexidade de dados** é determinada pelo número de nibbles ativos no texto cifrado.
- Apenas 5 nibbles são afetados e, portanto, apenas $\mathbf{2^{20}}$ pares de textos simples/cifrados são necessários.

Complexidades

- A **complexidade de dados** é determinada pelo número de nibbles ativos no texto cifrado.
- Apenas 5 nibbles são afetados e, portanto, apenas $\mathbf{2^{20}}$ pares de textos simples/cifrados são necessários.
- Em termos de **memória**, o ataque é limitado por 2^4 computações de $g \circ h$.

Complexidades

- A **complexidade de dados** é determinada pelo número de nibbles ativos no texto cifrado.
- Apenas 5 nibbles são afetados e, portanto, apenas $\mathbf{2^{20}}$ pares de textos simples/cifrados são necessários.
- Em termos de **memória**, o ataque é limitado por 2^4 computações de $g \circ h$.
- O cálculo completo de $g \circ h$ consiste em 25 estados e 6 subchaves, com 16 nibbles cada.

Complexidades

- A **complexidade de dados** é determinada pelo número de nibbles ativos no texto cifrado.
- Apenas 5 nibbles são afetados e, portanto, apenas $\mathbf{2^{20}}$ pares de textos simples/cifrados são necessários.
- Em termos de **memória**, o ataque é limitado por 2^4 computações de $g \circ h$.
- O cálculo completo de $g \circ h$ consiste em 25 estados e 6 subchaves, com 16 nibbles cada.
- Portanto, a complexidade de memória é $2^4 \cdot (25 + 16) \cdot 16 = 10.496$ nibbles, o que equivale a 5.248 bytes.

Complexidades

- A **complexidade de tempo** é determinada por

$$C_{total} = 2^{k-2d}(C_{biclique} + C_{precomp} + C_{recomp} + C_{falpos}).$$

Complexidades

- A **complexidade de tempo** é determinada por

$$C_{total} = 2^{k-2d}(C_{biclique} + C_{precomp} + C_{recomp} + C_{falpos}).$$

- O cálculo dos custos é dado pela porcentagem de S-boxes necessárias para realizar o ataque, em comparação com o número total de S-boxes na cifra.

Complexidades

- A **complexidade de tempo** é determinada por

$$C_{total} = 2^{k-2d}(C_{biclique} + C_{precomp} + C_{recomp} + C_{falpos}).$$

- O cálculo dos custos é dado pela porcentagem de S-boxes necessárias para realizar o ataque, em comparação com o número total de S-boxes na cifra.
- Por exemplo,

$$C_{recomp} = (2^8 - 2^4) \cdot (11/160) = 2^{4,0444}$$

Complexidades

- A **complexidade de tempo** é determinada por

$$C_{total} = 2^{k-2d}(C_{biclique} + C_{precomp} + C_{recomp} + C_{falpos}).$$

- O cálculo dos custos é dado pela porcentagem de S-boxes necessárias para realizar o ataque, em comparação com o número total de S-boxes na cifra.
- Por exemplo,

$$C_{recomp} = (2^8 - 2^4) \cdot (11/160) = 2^{4,0444}$$

- Ao final tem-se:

$$C_{total} = 2^{120}(2^{2,0820} + 2^{3,2863} + 2^{4,0444} + 2^{2,5110}) = 2^{125,18}$$

Conclusões

Conclusões

- Apresentamos aqui o ataque de biclique mais rápido e com, de longe, a menor complexidade de dados na cifra FUTURE de rodadas completas.
- É uma biclique balanceada de dimensão 4, que requer apenas 2^{20} pares para ser executado e tem complexidade de tempo de $2^{125,18}$.
- Todos os diagramas completos podem ser encontrados em https://github.com/Clique33/BicliqueFinder.
- Trabalhos futuros sobre esta cifra envolvem a busca por bicliques desbalanceadas.
- O mesmo processo também pode ser aplicado a outras cifras leves, como a família GIFT.

Fim de Apresentação

Referências

- Himadry Sekhar Roy, Prakash Dey, Sandip Kumar Mondal, and Avishek Adhikari.
  Cryptanalysis of full round future with multiple biclique structures.
  Peer-to-Peer Networking and Applications, 17(1):397–409, 2024.
- André Schrottenloher and Marc Stevens.
  Simplified modeling of mitm attacks for block ciphers: New (quantum) attacks.
  IACR Transactions on Symmetric Cryptology, 2023:146–183, 2023.
- Zeyu Xu, Jiamin Cui, Kai Hu, and Meiqin Wang.
  Integral attack on the full future block cipher.
  Tsinghua Science and Technology, 2024.