Breaking Symmetric Cryptosystems using Quantum Period Finding

Marc Kaplan[1,2] Gaëtan Leurent[3]
Anthony Leverrier[3] María Naya-Plasencia[3]

[1]LTCI, Télécom ParisTech

[2]School of Informatics, University of Edinburgh

[3]Inria Paris

Crypto 2016

Motivation

What would be the impact of quantum computers on symmetric cryptography?

- Some physicists think they can build quantum computers
- NSA thinks we need quantum-resistant crypto (or do they?)

Motivation

What would be the impact of quantum computers on symmetric cryptography?

- Some physicists think they can build quantum computers
- NSA thinks we need quantum-resistant crypto (or do they?)

Expected impact of quantum computers

- Some problems can be solved much faster with quantum computers
  - Up to exponential gains
  - But we don't expect to solve all NP problems

Impact on public-key cryptography

- RSA, DH, ECC broken by Shor's algorithm
  - Breaks factoring and discrete log in polynomial time
  - Large effort to develop quantum-resistant algorithms

Impact on symmetric cryptography

- Exhaustive search of a n-bit key in time $2^{n/2}$ with Grover's algorithm
  - Common recommendation: double the key length (AES-256)
  - Is there more?

Expected impact of quantum computers

- Some problems can be solved much faster with quantum computers
  - Up to exponential gains
  - But we don't expect to solve all NP problems

Impact on public-key cryptography

- RSA, DH, ECC broken by Shor's algorithm
  - Breaks factoring and discrete log in polynomial time
  - Large effort to develop quantum-resistant algorithms

Impact on symmetric cryptography

- Exhaustive search of a n-bit key in time $2^{n/2}$ with Grover's algorithm
  - Common recommendation: double the key length (AES-256)
  - Is there more?

Expected impact of quantum computers

- Some problems can be solved much faster with quantum computers
  - Up to exponential gains
  - But we don't expect to solve all NP problems

Impact on public-key cryptography

- RSA, DH, ECC broken by Shor's algorithm
  - Breaks factoring and discrete log in polynomial time
  - Large effort to develop quantum-resistant algorithms

Impact on symmetric cryptography

- Exhaustive search of a n-bit key in time $2^{n/2}$ with Grover's algorithm
  - Common recommendation: double the key length (AES-256)
  - Is there more?

Previous work: breaking Even-Mansour encryption

Kuwakado & Morii [ISITA '12]

The Even-Mansour cipher can be broken with quantum queries

Even-Mansour cipher [Even & Mansour, Crypto '97]

- Simple block cipher construction, from a public permutation P
  - $E_k(x) = P(x \oplus k_1) \oplus k_2$

- Security proof
  - Attacker is given oracle access to P and E
  - "If P is a random permutation, attacks against E_k with time T and data D are possible only if $DT > 2^n$"

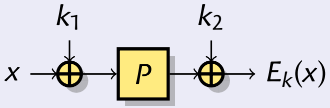
Previous work: breaking Even-Mansour encryption

Kuwakado & Morii [ISITA '12]

The Even-Mansour cipher can be broken with quantum queries

Even-Mansour cipher [Even & Mansour, Crypto '97]

- Simple block cipher construction, from a public permutation P
  - $E_k(x) = P(x \oplus k_1) \oplus k_2$



- Security proof
  - Attacker is given oracle access to P and E
  - "If P is a random permutation, attacks against E_k with time T and data D are possible only if $DT > 2^n$"

Classical attack against Even-Mansour

Slide with a twist attack [Biryukov & Wagner, Eurocrypt '00]
Using $2^{n/2}$ known plaintext $y_i = E_k(x_i)$

1. Assume that a pair of plaintext satisfy $x' = x \oplus k_1$
   - $E_k(x) = P(\underbrace{x \oplus k_1}_{x'}) \oplus k_2$, $\quad E_k(x') = P(\underbrace{x' \oplus k_1}_{x}) \oplus k_2$
   - $E_k(x) \oplus E_k(x') = P(x) \oplus P(x') = k_2$
   - $E_k(x) \oplus P(x) = E_k(x') \oplus P(x')$
2. Attacker computes $y_i \oplus P(x_i) = E_k(x_i) \oplus P(x_i)$, looks for collisions
3. When $y_i \oplus P(x_i) = y_j \oplus P(x_j)$, try $k_1 = x_i \oplus x_j$

Quantum attack against Even-Mansour

Kuwakado & Morii, [ISITA '12]

The Even-Mansour cipher can be broken with quantum queries

- Build the same function as in the classical attack:

$$\begin{aligned} f : \{0,1\}^n &\to \{0,1\}^n \\ x &\mapsto E_{k_1,k_2}(x) \oplus P(x) = P(x \oplus k_1) \oplus P(x) \oplus k_2. \end{aligned}$$

$$f(x) = f(x \oplus k_1)$$

- There is a quantum algorithm to recover k_1 with $O(n)$ queries
  - Simon's algorithm (period-finding)
  - Superposition queries to f: $\sum_x \psi_x |x\rangle|0\rangle \mapsto \sum_x \psi_x |x\rangle|f(x)\rangle$

Quantum attack against Even-Mansour

Kuwakado & Morii, [ISITA '12]

The Even-Mansour cipher can be broken with quantum queries

- Build the same function as in the classical attack:

$$\begin{aligned} f : \{0,1\}^n &\to \{0,1\}^n \\ x &\mapsto E_{k_1,k_2}(x) \oplus P(x) = P(x \oplus k_1) \oplus P(x) \oplus k_2. \end{aligned}$$

$$f(x) = f(x \oplus k_1)$$

- There is a quantum algorithm to recover k_1 with $O(n)$ queries
  - Simon's algorithm (period-finding)
  - Superposition queries to f: $\sum_x \psi_x |x\rangle|0\rangle \mapsto \sum_x \psi_x |x\rangle|f(x)\rangle$

Quantum attack against Even-Mansour

Kuwakado & Morii, [ISITA '12]

The Even-Mansour cipher can be broken with quantum queries

- Build the same function as in the classical attack:

$$\begin{aligned} f : \{0,1\}^n &\to \{0,1\}^n \\ x &\mapsto E_{k_1,k_2}(x) \oplus P(x) = P(x \oplus k_1) \oplus P(x) \oplus k_2. \end{aligned}$$

$$f(x) = f(x \oplus k_1)$$

1. Build a quantum circuit for f, from a circuit for E_k
2. Apply Simon's algorithm to recover k_1

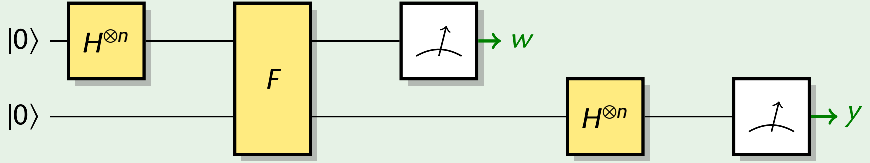
Simon's Algorithm

Definition (Simon's problem)

Given $f : \{0,1\}^n \to \{0,1\}^n$ such that there exists $\delta \in \{0,1\}^n$ with $f(x) = f(x') \Leftrightarrow x \oplus x' \in \{0^n, \delta\}$, find δ.

- Classical algorithms require $O(2^{n/2})$ queries (finding collisions)
- Simon's algorithm require $O(n)$ quantum queries

One step of Simon's algorithm returns $y \perp \delta$



Simon's Algorithm

Definition (Simon's problem)

Given $f : \{0,1\}^n \to \{0,1\}^n$ such that there exists $\delta \in \{0,1\}^n$ with $f(x) = f(x') \Leftrightarrow x \oplus x' \in \{0^n, \delta\}$, find δ.

- Classical algorithms require $O(2^{n/2})$ queries (finding collisions)
- Simon's algorithm require $O(n)$ quantum queries

Weaker promise

$f(x) = f(x') \Leftarrow x \oplus x' \in \{0^n, \delta\}$ *i.e.* $\forall x,\ f(x) = f(x \oplus \delta)$

- There are extra collisions $f(x) = f(x')$ with arbitrary $x \oplus x'$
- If there is no structure in these collisions, we can still recover δ
- Complexity increase by a factor $O(1/(1-\varepsilon))$, with $\varepsilon = \max_{t \neq \{0,\delta\}} \Pr_x[f(x) = f(x \oplus t)]$

Sim

Definition (Sim

Given $f : \{0,1\}^n$

$f(x) = f(x') \Leftrightarrow x$

- Classical alg
- Simon's algo

Weaker promise

$f(x) = f(x') \Leftarrow x \oplus x'$

- There are extra
- If there is no stru
- Complexity incre $\mathrm{Pr}_x[f(x) = f(x \oplus t)]$

In this talk, Simon's algorithm is a magic black box

https://en.wikipedia.org/wiki/File:Computer-kitten.jpg

About the model

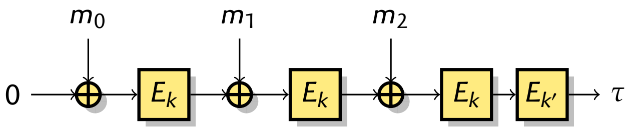
Superposition queries

- Access quantum circuit implementing the primitive with a secret key
- Stronger assumption than building a circuit from public values (*e.g.* Shor's algorithm to break RSA, ECC)

- Simple and clean generalisation of classical oracle
- Very powerful model (for the adversary)
  - But there exist secure schemes
  - Aim for security in the strongest possible model
- Not a threat against classical crypto devices
  - But... Are we sure a classical device has no quantum effects?
  - Also interesting for black-box crypto

Outline

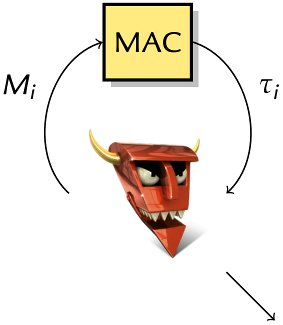
CBC-MAC



- One of the first MAC [NIST, ANSI, ISO, '85?]
- Based on CBC encryption mode
- Security proof [Bellare, Kilian & Rogaway '94]
  - "If E is a secure block cipher, there are no forgery attacks against CBC-MAC with less than $2^{n/2}$ blocs"

Classical security notions: CPA security

- Key-recovery: given access to a MAC oracle, extract the key
- Forgery: given access to a MAC oracle, forge a valid pair



M, τ s.t. $M \notin \{M_i\}$ and $\tau = \mathrm{MAC}(M)$

Quantum Security Notion

qCPA: quantum Chosen Plain Attack [Boneh & Zhandry, EC'13]

- Access to a quantum MAC oracle (superposition queries)
- Output $k+1$ valid message/tags after k queries

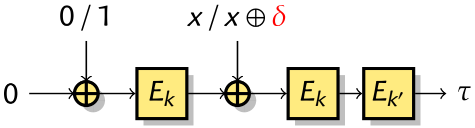
MAC

$\sum_x \psi_x |x\rangle|0\rangle$

$\sum_x \psi_x |x\rangle|\mathrm{MAC}(x)\rangle$

$\{M_i, \tau_i\}$ s.t. $\tau_i = \mathrm{MAC}(M_i)$

Quantum attack against CBC-MAC



- Consider the following function:

$$f : \{0,1\} \times \{0,1\}^n \to \{0,1\}^n$$
$$b, x \mapsto \mathrm{MAC}(b \,\|\, x) = E_{k'}\Big(E_k\big(x \oplus E_k(b)\big)\Big)$$
$$f(0,x) = E_{K'}(E_k(x \oplus E_k(1)))$$
$$f(1,x) = E_{K'}(E_k(x \oplus E_k(0)))$$

- $f(b,x) = f(b \oplus 1, x \oplus \delta)$, with $\delta = E_k(0) \oplus E_k(1)$
  - Simon's algorithm recovers $1 \,\|\, \delta$
  - Produce forgeries: $\mathrm{MAC}(0 \,\|\, m) = \mathrm{MAC}(1 \,\|\, m \oplus \delta)$

Attack structure

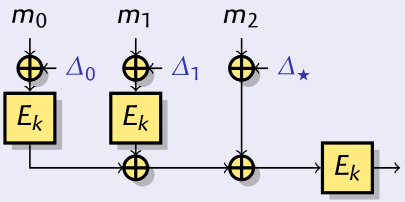
1. Define a function f with $f(x \oplus \delta) = f(x)$ for some interesting δ
2. Build quantum circuit for f, use Simon's algorithm to recover δ
   - $t = O(n)$ quantum queries
3. Use δ to produce forgeries
   - One classical query gives two messages/MAC pairs
   - Repeat until more valid messages than queries ($t + 1$ times)

Applications of Simon's algorithm

- Breaks most common MAC and AEAD modes
- Corresponds to classical attacks with $2^{n/2}$ queries
  - Query f with $2^{n/2}$ values, look for collisions

PMAC: Parallelisable MAC with secret offsets

PMAC



- CBC-MAC structure for 2-block M
- Same attack

$$f : \{0,1\} \times \{0,1\}^n \to \{0,1\}^n$$

$$b, x \mapsto \mathrm{MAC}(b \,\|\, x)$$

$$f(b,x) = E_k(E_k(m_0 \oplus \Delta_0) \oplus m_1 \oplus \Delta_\star)$$

$$f(b,x) = f(b \oplus 1, x \oplus \delta)$$

$$\delta = E_k(\Delta_0) \oplus E_K(\Delta_0 \oplus 1)$$

PMAC variant

- No message goes directly into the state
- Alternative attack

$$f : \{0,1\}^n \to \{0,1\}^n$$

$$x \mapsto \mathrm{MAC}(x \,\|\, x)$$

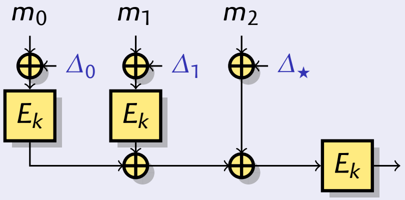
$$f(x) = E_k(E_k(x \oplus \Delta_0) \oplus E_k(x \oplus \Delta_1))$$

$$f(x) = f(x \oplus \delta)$$

$$\delta = \Delta_0 \oplus \Delta_1$$

PMAC: Parallelisable MAC with secret offsets

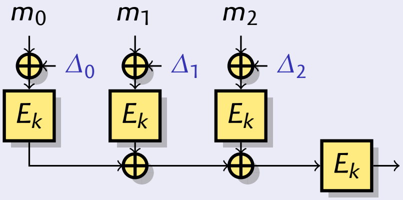
PMAC



- CBC-MAC structure for 2-block M
- Same attack

$$
\begin{aligned}
f : \{0,1\} \times \{0,1\}^n &\to \{0,1\}^n \\
b, x &\mapsto \mathrm{MAC}(b \,\|\, x) \\
f(b,x) &= E_k(E_k(m_0 \oplus \Delta_0) \oplus m_1 \oplus \Delta_\star) \\
f(b,x) &= f(b \oplus 1, x \oplus \delta) \\
\delta &= E_k(\Delta_0) \oplus E_K(\Delta_0 \oplus 1)
\end{aligned}
$$

PMAC variant

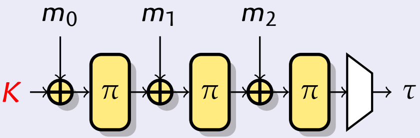


- No message goes directly into the state
- Alternative attack

$$
\begin{aligned}
f : \{0,1\}^n &\to \{0,1\}^n \\
x &\mapsto \mathrm{MAC}(x \,\|\, x) \\
f(x) &= E_k(E_k(x \oplus \Delta_0) \oplus E_k(x \oplus \Delta_1)) \\
f(x) &= f(x \oplus \delta) \\
\delta &= \Delta_0 \oplus \Delta_1
\end{aligned}
$$

Sponge-based modes

Full-width sponge



- Same structure as CBC-MAC
- Same attack

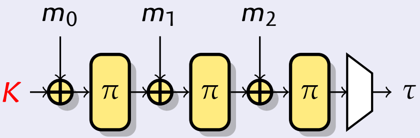
$$
\begin{aligned}
f : \{0,1\} \times \{0,1\}^n &\to \{0,1\}^n \\
b, x \quad &\mapsto \mathrm{MAC}(b \,\|\, x) \\
f(b,x) &= \pi(\pi(K \oplus b) \oplus x) \\
f(b,x) &= f(b \oplus 1, x \oplus \delta) \\
\delta &= \pi(K) \oplus \pi(K \oplus 1)
\end{aligned}
$$

Normal sponge

- Can't cancel the full state difference
- No attack found

Sponge-based modes

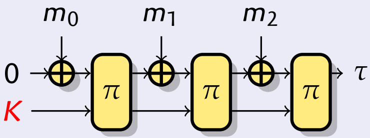
Full-width sponge



- Same structure as CBC-MAC
- Same attack

$$\begin{aligned} f : \{0,1\} \times \{0,1\}^n &\to \{0,1\}^n \\ b, x &\mapsto \mathrm{MAC}(b \,\|\, x) \\ f(b,x) &= \pi(\pi(K \oplus b) \oplus x) \\ f(b,x) &= f(b \oplus 1, x \oplus \delta) \\ \delta &= \pi(K) \oplus \pi(K \oplus 1) \end{aligned}$$

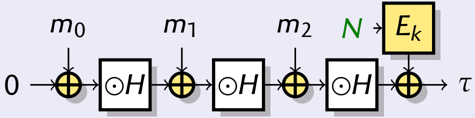
Normal sponge



- Can't cancel the full state difference
- No attack found

Nonce-based modes

Nonce at the end (GMAC)



- Same structure as CBC-MAC
- Same attack

$$
\begin{aligned}
f_N : \{0,1\} \times \{0,1\}^n &\to \{0,1\}^n \\
b, x &\mapsto \mathrm{MAC}(b \,\|\, x) \\
f_N(b,x) &= b \cdot H^2 \oplus x \cdot H \oplus E_k(N) \\
f_N(b,x) &= f(b \oplus 1, x \oplus \delta) \\
\delta &= H
\end{aligned}
$$

Nonce at the beginning (CCM)

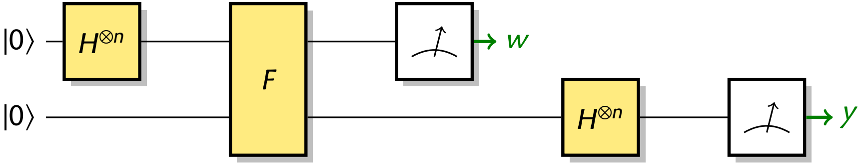
- State difference depend on N
- No fixed period δ
- No attack found

Dealing with the nonce

- We can't really apply Simon's algorithm to f_N
  - We don't choose N
  - Each oracle call will use a different N
- Luckily, one step of Simon's algorithm makes a single call to f_N
  - The family f_N satisfies Simon's promise with the same δ
  - One step gives y with $y \perp \delta$
  - Classical repetition, classical linear algebra

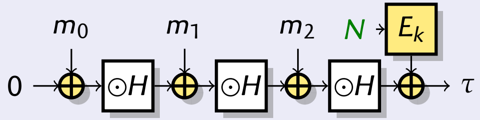
Dealing with the nonce

- We can't really apply Simon's algorithm to f_N
  - We don't choose N
  - Each oracle call will use a different N
- Luckily, one step of Simon's algorithm makes a single call to f_N
  - The family f_N satisfies Simon's promise with the same δ
  - One step gives y with $y \perp \delta$
  - Classical repetition, classical linear algebra



Nonce-based modes

Nonce at the end (GMAC)



- Same structure as CBC-MAC
- Same attack

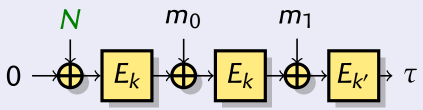
$$
\begin{aligned}
f_N : \{0,1\} \times \{0,1\}^n &\to \{0,1\}^n \\
b, x &\mapsto \mathrm{MAC}(b \,\|\, x)
\end{aligned}
$$

$$f_N(b,x) = b \cdot H^2 \oplus x \cdot H \oplus E_k(N)$$

$$f_N(b,x) = f(b \oplus 1, x \oplus \delta)$$

$$\delta = H$$

Nonce at the beginning (CCM)



- State difference depend on N
- No fixed period δ
- No attack found

Quantum attack against AEAD

- When M is empty, AEAD becomes MAC
- A lot of AEAD modes process A before N
  - Structure similar to GMAC
  - $\tau = \text{MAC}(A, N, C) = \phi(A) * \psi(M, N)$
  - Efficiency argument: pre-computation
  - Notable counter-example: CCM
- Previous attack on MACs can be applied
  - $f_{\text{AEAD}}(x) = f_{\text{MAC}}(x) * g(N)$
  - $f_{\text{MAC}}(x) = f_{\text{MAC}}(x \oplus \delta) \Rightarrow f_{\text{AEAD}}(x) = f_{\text{AEAD}}(x \oplus \delta)$
  - PMAC $\rightarrow$ OCB
  - GMAC $\rightarrow$ GCM
- Also attacks not based on this property in the paper
  - Alternative attack against OCB

Quantum attack against AEAD

- When M is empty, AEAD becomes MAC
- A lot of AEAD modes process A before N
  - Structure similar to GMAC
  - $\tau = \mathrm{MAC}(A, N, C) = \phi(A) * \psi(M, N)$
  - Efficiency argument: pre-computation
  - Notable counter-example: CCM
- Previous attack on MACs can be applied
  - $f_{\mathrm{AEAD}}(x) = f_{\mathrm{MAC}}(x) * g(N)$
  - $f_{\mathrm{MAC}}(x) = f_{\mathrm{MAC}}(x \oplus \delta) \Rightarrow f_{\mathrm{AEAD}}(x) = f_{\mathrm{AEAD}}(x \oplus \delta)$
  - PMAC $\rightarrow$ OCB
  - GMAC $\rightarrow$ GCM
- Also attacks not based on this property in the paper
  - Alternative attack against OCB

Quantum security of modes of operations

Applications of Simon's algorithm

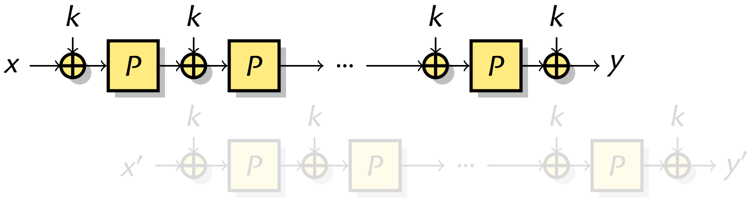
Common MAC and AEAD modes broken with superposition queries:

- CBC-MAC, PMAC, GMAC, GCM, OCB
- 8 CAESAR candidates: **AEZ**, **CLOC**, **COLM**, Minalpher, **OCB**, OMD, **OTR**, POET

Secure modes

- Common encryption modes are mostly quantum-secure
  [Unruh, Targhi, Tabia & Anand, PQC'16]
- Efficient MACs & AEAD secure against quantum attacks?
  - Boneh & Zhandry: quantum safe Carter-Wegman MAC, where the randomness depend on the message
- Do we have the right security definition?

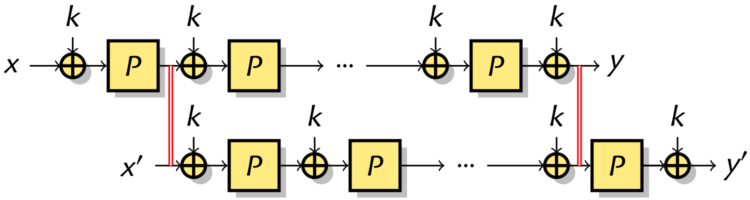
Classical slide attacks



- Cryptanalysis of block ciphers [Biryukov & Wagner, FSE '99]
- Applicable if all rounds are identical $E_k(P(x \oplus k)) = P(E_k(x)) \oplus k$

1. Assume a pair $x' = P(x \oplus k)$, then $y' = P(y) \oplus k$
   - $x \oplus P^{-1}(x') = P(y) \oplus y' = k$
   - $x \oplus P(y) = P^{-1}(x') \oplus y'$
2. Attacker looks for collision between
   - $x_i \oplus P(y_i)$
   - $P^{-1}(x_j) \oplus y_j$
3. When $x_i \oplus P(y_i) = P^{-1}(x_j) \oplus y_j$, try $k = x_i \oplus P^{-1}(x_j)$

Classical slide attacks



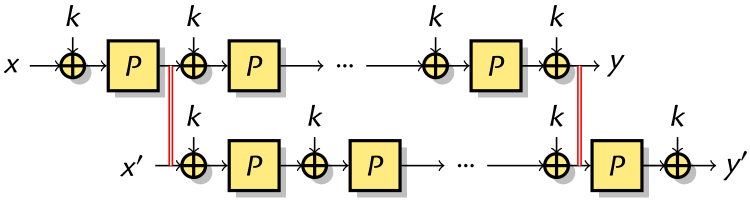
- Cryptanalysis of block ciphers
- Applicable if all rounds are identical

[Biryukov & Wagner, FSE '99]

$E_k(P(x \oplus k)) = P(E_k(x)) \oplus k$

1. Assume a pair $x' = P(x \oplus k)$, then $y' = P(y) \oplus k$
   - $x \oplus P^{-1}(x') = P(y) \oplus y' = k$
   - $x \oplus P(y) = P^{-1}(x') \oplus y'$
2. Attacker looks for collision between
   - $x_i \oplus P(y_i)$
   - $P^{-1}(x_j) \oplus y_j$
3. When $x_i \oplus P(y_i) = P^{-1}(x_j) \oplus y_j$, try $k = x_i \oplus P^{-1}(x_j)$

Quantum slide attacks



- $E_k(P(x \oplus k)) = P(E_k(x)) \oplus k$
- Build the same function as in the classical attack:

$$f : \{0,1\} \times \{0,1\}^n \to \{0,1\}^n$$
$$b, x \quad \mapsto \begin{cases} x \oplus P(E_k(x)) & \text{if } b = 0, \\ x \oplus E_k(P(x)) & \text{if } b = 1. \end{cases}$$

- $f(0,x) = P(E_k(x)) \oplus x = E_k(P(x \oplus k)) \oplus k \oplus x = f(1, x \oplus k)$
  - Simon's algorithm recovers $1 \parallel k$

Conclusion

About Simon's algorithm

- Simon's algorithm breaks real problems!
- Simon's algorithm can be extended
  1. Find a t s.t. $f(x \oplus t) = f(x)$ with high probability
  2. Recover δ with a weaker promise:
     - $f(x) = f(x \oplus \delta)$
     - $\Pr_x[f(x) = f(x \oplus t)]$ small for $t \neq 0, \delta$
  3. Recover δ from a nonce-based family of functions with $f_N(x) = f_N(x') \Leftrightarrow x \oplus x' \in \{0^n, \delta\}$

Conclusion

About Simon's algorithm

- Simon's algorithm breaks real problems!
- Simon's algorithm can be extended

Applications to crypto

- Common MAC and AE modes broken with superposition queries
- Some cryptanalysis techniques can also be improved
- Impact:
  - There are better quantum attacks than Grover for symmetric crypto
  - Even if the NSA has a quantum computer, they can NOT break current symmetric cryptosystems with this attack.

Conclusion

About Simon's algorithm

- Simon's algorithm breaks real problems!
- Simon's algorithm can be extended

Applications to crypto

- Common MAC and AE modes broken with superposition queries
- Some cryptanalysis techniques can also be improved
- Impact:
  - There are better quantum attacks than Grover for symmetric crypto
  - Even if the NSA has a quantum computer, they can NOT break current symmetric cryptosystems with this attack.