Quantum Security Analysis of AES

Xavier Bonnetain, María Naya-Plasencia, André Schrottenloher

Inria, France

Outline

Introduction

Context

- We are studying the security of **block ciphers** in the presence of **quantum adversaries**

The adversary's power

Quantum adversaries are capable of **local quantum computations**, of **classical encryption / decryption queries**, and possibly of **quantum queries**.

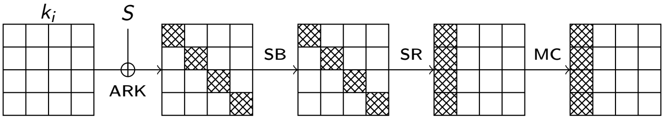
- Some constructions have been broken using **quantum queries** (*e.g.* the Even-Mansour cipher).
- But they usually have a strong algebraic structure.

The AES

It is an SPN with 128-bit blocks of 4×4 bytes. An AES round:

1. XORs the round key k_i (**ARK**)
2. applies the AES S-Box to each byte (**SB**)
3. shifts the j-th row by j bytes left (**SR**)
4. multiplies each column by the AES MDS matrix (**MC**)

The **AES key-schedule** expands the master key k into $r+1$ round keys $k_0, \dots k_r$.
There are three variants: AES-128 ($\mathbf{r = 10}$), AES-192 ($\mathbf{r = 12}$), AES-256 ($\mathbf{r = 14}$).



Example: exhaustive key search on AES-256

Classical key-recovery

Make 3 queries to the encryption black-box, try all keys until the encryptions match (2^{256} equivalent AES encryptions).

- reduced-round attacks going below this complexity determine the **security margin** of AES.

Quantum key-recovery

Make 3 queries to the encryption black-box, use Grover's algorithm to find the key that matches ($\simeq 2^{128}$ equivalent AES encryptions, **as a quantum circuit**).

- what is the **quantum** security margin of AES?

Contributions of this paper

- We study **quantum key-recovery attacks** on reduced-round AES: key-recoveries below Grover's exhaustive search
- Our best attacks require **standard encryption queries** only
- Some of these ideas also gave new time-space tradeoffs for **classical** attacks

	Classical		Quantum	
Version	Rounds attacked	Method	Rounds attacked	Method
AES-128	7	ID or DS-MITM	**6**	**Square**
AES-192	8	DS-MITM	**7**	**Square**
AES-256	9	DS-MITM	**8**	**DS-MITM**

A Framework for Search Problems

Our starting point

How much does Grover search cost?

- We count the number of **quantum gates** (*i.e.* time) in the **quantum circuit model**
- We use the counts of Grassl *et al.* (PQCRYPTO 16)
- In quantum circuits, the most costly component is the AES S-Box: we can **count everything in number of S-Boxes**

8-round AES-256

With 3 classical known-plaintext queries, the key can be recovered in $2^{138.04}$ quantum AES S-Boxes.

Grassl et al., *"Applying Grover's Algorithm to AES: Quantum Resource Estimates"*, PQCRYPTO 2016

Classical search vs. quantum search

Let X be a search space, P a predicate, $X_P \subseteq X = \{x \in X, P(x)\}$. We define:
Filter $x \in X$ **such that** $P(x)$, a "filter" that samples X_P using samples from X.

Classical search as a filter

- sample elements $x \in X$
- evaluate $P(x)$

until $P(x) = \textbf{true}$

We sample from X_P in time:

$$\frac{|\mathbf{X}|}{|\mathbf{X_P}|}\left(c_{\text{Sample}}(X) + c_{\text{Eval}}(P)\right)$$

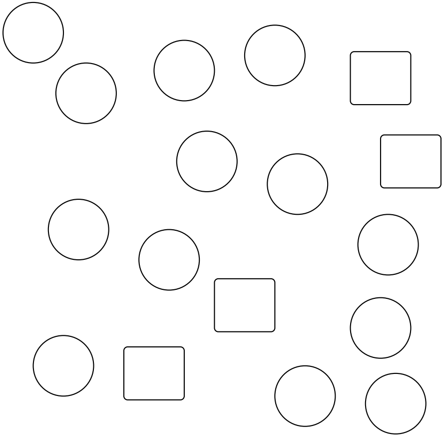
Quantum search as a filter

- start from the uniform superposition over X
- use Grover's algorithm to obtain the uniform superposition over X_P

$$\sqrt{\frac{|\mathbf{X}|}{|\mathbf{X_P}|}}\left(q_{\text{Sample}}(X) + q_{\text{Eval}}(P)\right)$$

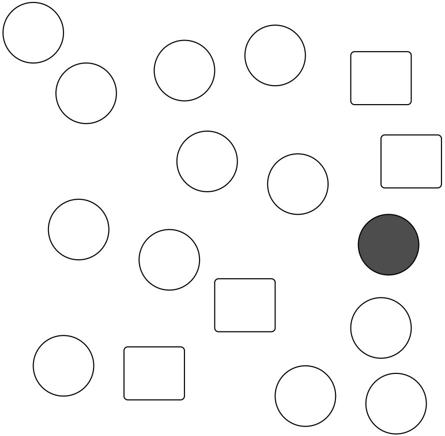
Classical search vs. quantum search (ctd.)

In the classical realm, we test elements x at random until we have found (a random) $x \in X_P$.



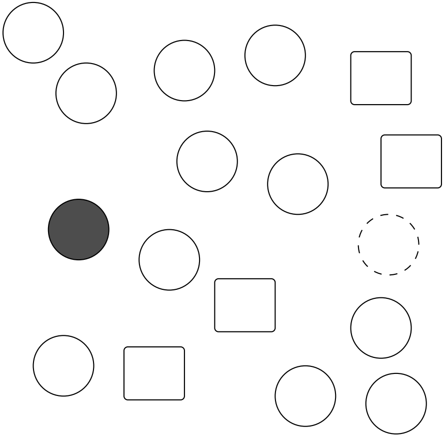
Classical search vs. quantum search (ctd.)

In the classical realm, we test elements x at random until we have found (a random) $x \in X_P$.



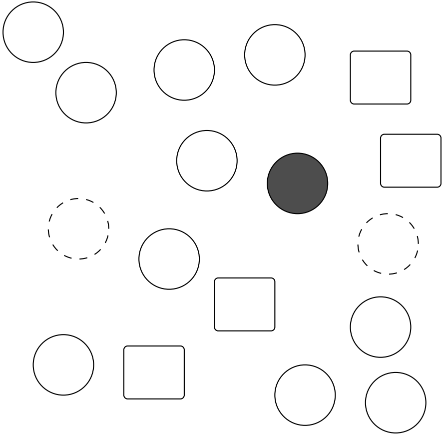
Classical search vs. quantum search (ctd.)

In the classical realm, we test elements x at random until we have found (a random) $x \in X_P$.



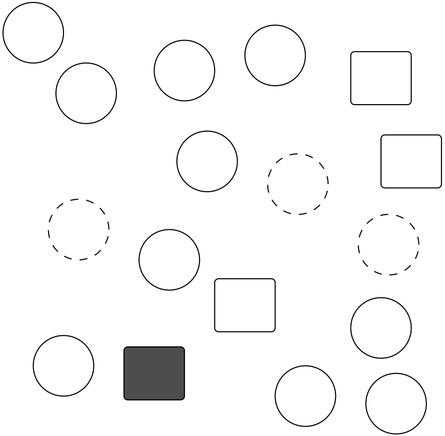
Classical search vs. quantum search (ctd.)

In the classical realm, we test elements x at random until we have found (a random) $x \in X_P$.



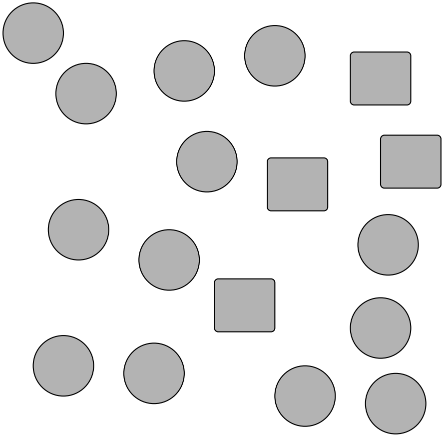
Classical search vs. quantum search (ctd.)

In the classical realm, we test elements x at random until we have found (a random) $x \in X_P$.



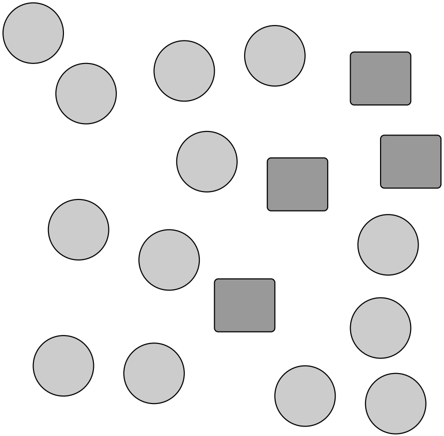
Classical search vs. quantum search

In the quantum realm, we move globally from X to X_P.



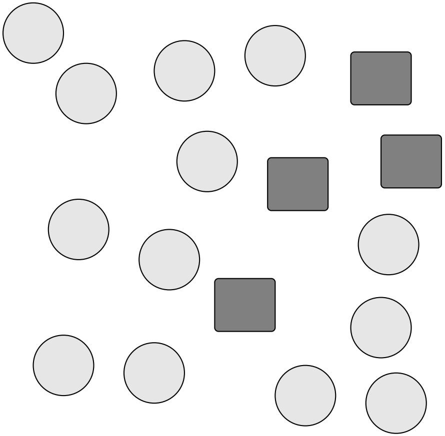
Classical search vs. quantum search

In the quantum realm, we move globally from X to X_P.



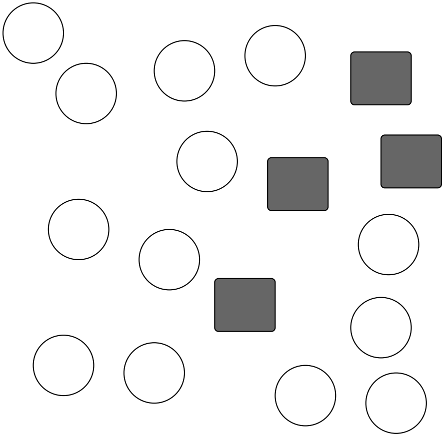
Classical search vs. quantum search

In the quantum realm, we move globally from X to X_P.



Classical search vs. quantum search

In the quantum realm, we move globally from X to X_P.



Nested searches

An example: evaluating a conjunction predicate.

$$c_{\mathsf{Sample}}(X_{P_1 \wedge P_2}) = \frac{|\mathbf{X}|}{|\mathbf{X_{P_1 \wedge P_2}}|}\left(c_{\mathsf{Sample}}(X) + c_{\mathsf{Eval}}(P_1) + c_{\mathsf{Eval}}(P_2)\right)$$

Less naively (lazy evaluation):

$$c_{\mathsf{Sample}}(X_{P_1 \wedge P_2}) = \frac{|\mathbf{X}|}{|\mathbf{X_{P_1 \wedge P_2}}|}\left(c_{\mathsf{S}}(X) + c_{\mathsf{Eval}}(P_1)\right) + \underbrace{\frac{|\mathbf{X_{P_1}}|}{|\mathbf{X_{P_1 \wedge P_2}}|} c_{\mathsf{Eval}}(P_2)}_{\text{Test only when } P_1 \text{ is } \textbf{true}}$$

$$c_{\mathsf{Sample}}(X_{P_1 \wedge P_2}) = \frac{|\mathbf{X_{P_1}}|}{|\mathbf{X_{P_1 \wedge P_2}}|}\left(\underbrace{\frac{|\mathbf{X}|}{|\mathbf{X_{P_1}}|}\left(c_{\mathsf{Sample}}(X) + c_{\mathsf{Eval}}(P_1)\right)}_{\text{Sample } X_{P_1}} + c_{\mathsf{Eval}}(P_2)\right)$$

$\Longrightarrow$ nested **filters**

Generic principle

Quantumly, the same **lazy evaluation** is simply a Grover search, in which the "sample" is another Grover search.

$$c_{\text{Sample}}(X_{P_1 \wedge P_2}) = \frac{|\mathbf{X_{P_1}}|}{|\mathbf{X_{P_1 \wedge P_2}}|}\Bigg(\underbrace{\frac{|\mathbf{X}|}{|\mathbf{X_{P_1}}|}\left(c_{\text{Sample}}(X) + c_{\text{Eval}}(P_1)\right)}_{\text{Sample } X_{P_1}} + c_{\text{Eval}}(P_2)\Bigg)$$

$$q_{\text{Sample}}(X_{P_1 \wedge P_2}) = \sqrt{\frac{|\mathbf{X_{P_1}}|}{|\mathbf{X_{P_1 \wedge P_2}}|}}\left(\sqrt{\frac{|\mathbf{X}|}{|\mathbf{X_{P_1}}|}}\left(q_{\text{Sample}}(X) + q_{\text{Eval}}(P_1)\right) + q_{\text{Eval}}(P_2)\right)$$

To any classical combination of **Filters**, corresponds a quantum procedure whose time complexity is obtained by square-rooting the number of iterations.

A quantum attack recipe

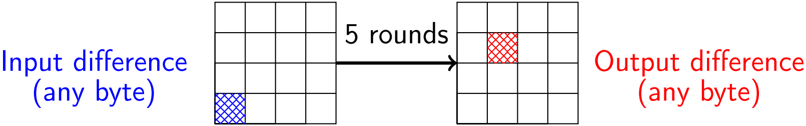
- Write a classical attack as a sequence of nested **Filters**
- Replace each **Filter** by a quantum search
- Replace the number of iterations by their square-roots
- If the search terms are dominant, this may be a quantum attack as well!

Technical postprocessing: handle non-classical factors and probabilities of success.

Quantum DS-MITM attack on 8-round AES-256

A rebound distinguisher

If a → differential is ensured, encryption of some differences in produces a specific result in .



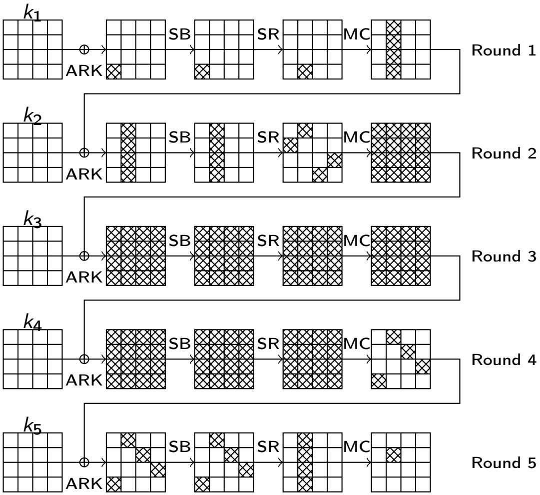
Main Property

Consider a pair giving → . If we make the difference in take some arbitrary values (δ-sequence) and collect the sequence of output differences in , there are only 2^{192} (24 byte-conditions) possibilities.

Demirci and Selçuk, *"A Meet-in-the-Middle Attack on 8-Round AES"*, FSE 2008

Derbez, Fouque and Jean, *"Improved Key Recovery Attacks on Reduced-Round AES in the Single-Key Setting"*, EUROCRYPT 2013

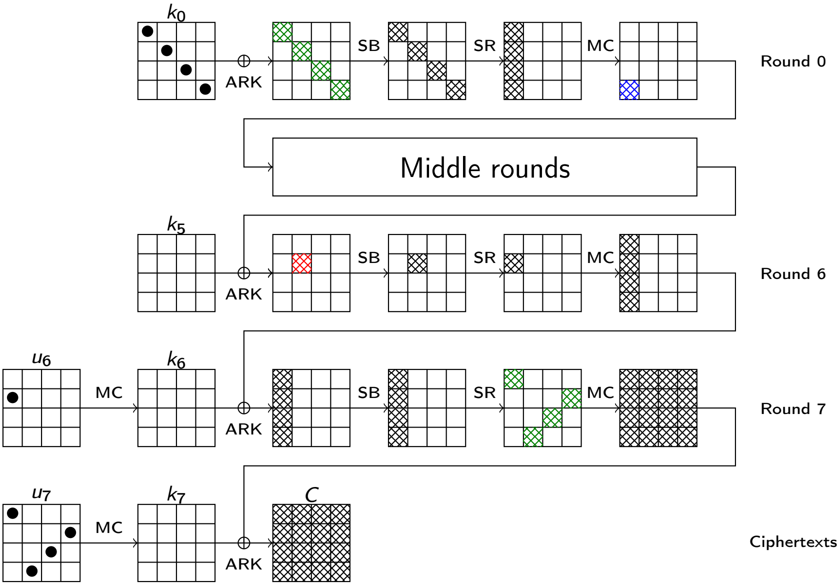
A rebound distinguisher (ctd.)



Rebound distinguisher: guess 24 internal state bytes and solve AES S-Box differential equations:

Given Δ_x, Δ_y, find the pairs x, y, x', y' such that $S(x) = y$, $S(x') = y', x \oplus x' = \Delta_x, y \oplus y' = \Delta_y$.

The classical attack tabulates the middle rounds. . . we don't.


k_0
ARK
SB
SR
MC
Round 0
Middle rounds
k_5
ARK
SB
SR
MC
Round 6
u_6
MC
k_6
ARK
SB
SR
MC
Round 7
u_7
MC
k_7
ARK
C
Ciphertexts

Attack layout

1. Query the AES black-box and find enough (2^{48}) input-output pairs satisfying the conditions
2. For each value of the • key bytes (10 of them), we have approx. one pair that satisfies $\rightarrow$

Testing a guess of the • key bytes

- Find a pair which gives $\rightarrow$
- Make new queries to vary the difference in
- Compute the corresponding δ-sequence in
- Find if the sequence in belongs to the $2^{24\times8}$ possibilities: **another search inside the search**

A classical attack

The number of "degrees of freedom" to search through:

$$\underbrace{10}_{\text{Key bytes}} + \underbrace{24}_{\text{Middle state bytes}} = 34 > \underbrace{32}_{\text{Exhaustive search}}$$

We reduce it with 4 relations between the key bytes • and the middle states:

$$\underbrace{10}_{\text{Key bytes}} + \underbrace{24}_{\text{Middle state bytes}} - \underbrace{4}_{\text{Relations}} = 30 < \underbrace{32}_{\text{Exhaustive search}}$$

- A middle-rounds encryption of a δ-sequence is approx. 5 times an AES
- We have $2^{30\times 8} = 2^{240}$ δ-sequences to evaluate
- Only $2^{250.3}$ S-Boxes against $2^{263.8}$ for exhaustive search

Some details to work out

Solving the differential S-Box equation: required for sieving in the middle. We give a circuit to do this with around 2 S-Box computations (of Grassl *et al.*).

Quantum queries: seem necessary at first sight; can be removed: 2^{88} classical queries.

Quantum-accessible memory: seems necessary at first sight; can be removed: 2^{89} classical memory.

An update

Jaques *et al.* have improved the S-Box circuit gate count by a factor 26. This changes the relative cost of solving the S-Box differential equation.

- Fortunately, this is not the dominating term, so our complexity in S-Boxes still holds.

Jaques et al., *"Implementing Grover oracles for quantum key search on AES and LowMC"*, EUROCRYPT 2020

New classical trade-offs

The classical DS-MITM attack tabulates the rebound distinguisher and sieves the subkey bytes.

- We propose to swap these steps: tabulate the subkey bytes and sieve the degrees of freedom in the distinguisher
- This yields new trade-offs (9 rounds of AES-256 in data 2^{113}, time 2^{210} and memory 2^{194})

Conclusion

Conclusion

- First security analysis of AES in a quantum setting
- We wrote our attacks (Square, DS-MITM) in a unified search framework
- We showed how to quantumly exploit the S-Box structure
- We reached an 8-round attack on AES-256
- We found new trade-offs for classical DS-MITM attacks

Thank you!