A New Non-Profiled Cache-Timing Template Attack on AES

Fan (Terry) ZHANG¹, Xinjie ZHAO²,³, Shize GUO³, Tao WANG², Zhijie (Jerry) SHI¹

1 University of Connecticut, USA
2 Ordnance Engineering College, China
3 The Institute of North Electronic Equipment, China
Outline

1. Introduction
2. Revisits of Profiled Cache-Timing Template attack
3. Non-Profiled Cache-Timing Template Attack
4. With Application to AES
5. Conclusion
Cache attack is one type of side-channel attack by exploiting leakages of cache accesses from microprocessors.
Three typical cache attacks

**Access-driven**
Exploiting accessed cache addresses leaked

**Trace-driven**
Exploiting cache hit/miss events leaked

**Time-driven**
Exploiting encryption time leaked

Simple, generic, also the focus of our work!
First Proposal


Template server $S_{lp}$

Target server $S_{lg}$

Cache-timing template: average encryption time for different table index.

Attack precondition: The cache-timing templates generated from the two servers are identical.
Revisits of Profiled Cache-Timing Template attack

Attack principle: Firstly, the adversary built the template $T$ from $S_{tp}$, then predict the template $T'$ from $S_{tg}$ by guessing a key byte. As the correct guess, $T$ and $T'$ have the largest correlation coefficients.

Results of attacking on the same machine
Advantages:
1) simple: only the total encryption time is required
2) generic: can be applied to attack different processors

Disadvantage:
1) requires a target server
2) how to find a template server that is “identical” to the target server

Can we launch cache attacks without extra template server?
Non-Profiled Cache-Timing Template Attack

Main Idea

1. The templates for lookup the same table are identical!
2. The templates for lookup different tables are different!

Templates of 16 table lookups in the first round of AES in OpenSSL v1.0.0 (Four 1KB tables)

Can we use the internal template for key recovery?
Yes, we can! Below is an example to recover $k_0 \oplus k_4$

1. Build $T$ by $p_0$

2. Build $T'$ by $p_4 \oplus (k_0 \oplus k_4)'$

3. Calculate the correlation coefficients, the one with the largest value is related with $k_0 \oplus k_4$
Case 1: Attacking the first round of AES in OpenSSL v1.0.0 with four 1KB tables.

The key search space of AES can be at most reduced to $2^{32}$. 

$2^n$ samples
Intel(R) Core(TM) i3-2330 CPU, 2.19 GHZ, 4GB memory, Window7, 64-bit OS.

**Case 2:** Attacking the first round of AES in OpenSSL v1.0.0 with one 2KB table.

The key search space of AES can be at most reduced to $2^8$.

AES with 2KB table is more vulnerable!
Case 3: Remote attack on the first round of AES in OpenSSL v1.0.0 with one 2KB table.

Preliminary attack results

A few key bytes (6 out of 16 bytes) of AES can be recovered, we are still working on it.
Conclusion:

1. We propose an nonprofiled cache-timing template attacks.

2. The proposed attacks do not require the extra template server, which increases the practicability of cache-timing template attacks.

3. Our attacks on AES show that AES in OpenSSL implemented with 2KB table is more vulnerable to four 1KB tables!
Thanks!

Q & A

Email: fan.zhang@engineer.uconn.edu
zhaoxinjieem@163.com