COSADE 2013
Paris, France
May 7-8, 2013

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

Fan (Terry) ZHANG[1], Xinjie ZHAO[2,3], Shize GUO[3],
Tao WANG[2], Zhijie (Jerry) SHI[1]

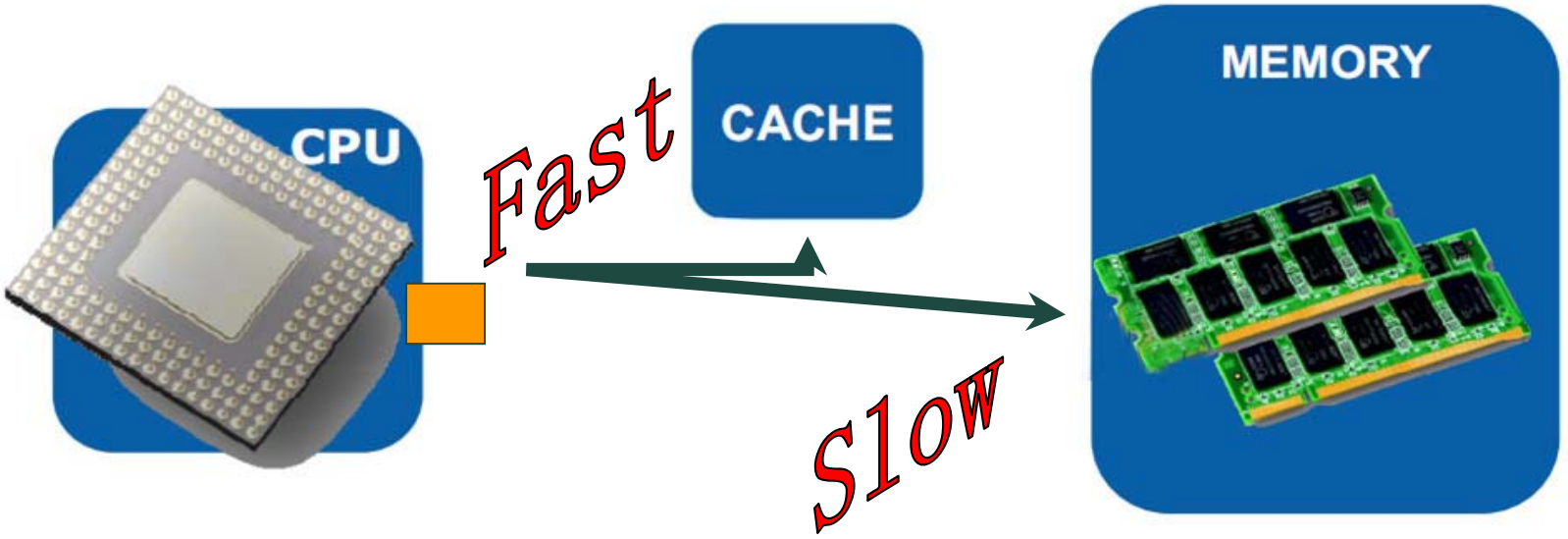
1 University of Connecticut, USA
2 Ordnance Engineering College, China
3 The Institute of North Electronic Equipment, China

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

1 Introduction

Cache attack is one type of side-channel attack by exploiting leakages of cache accesses from microprocessors.

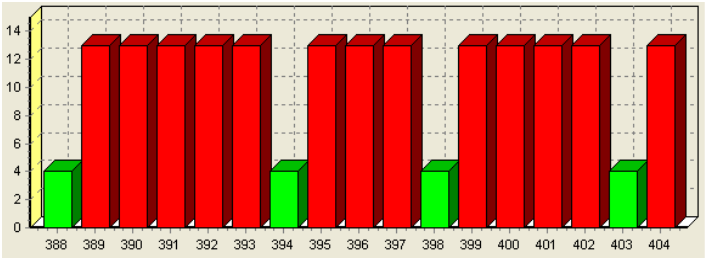


1 Introduction

Three typical cache attacks

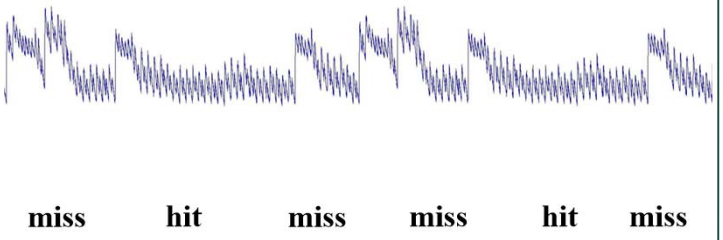
Access-driven

Exploiting accessed cache addresses leaked



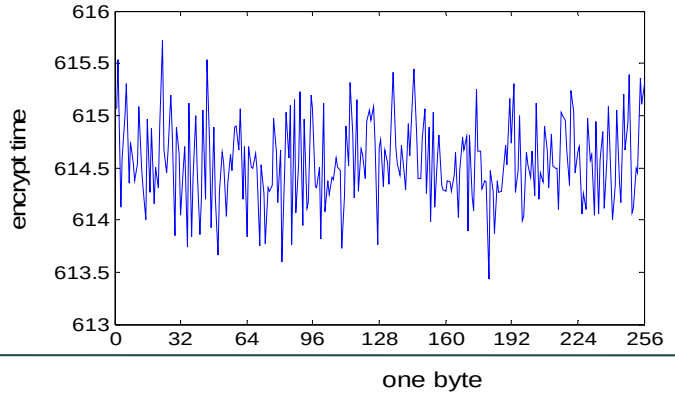
Trace-driven

Exploiting cache hit/miss events leaked



Time-driven

Exploiting encryption time leaked



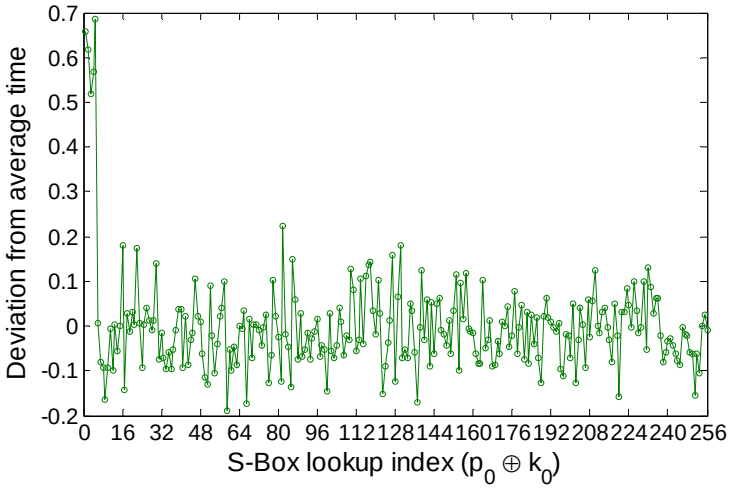
Simple, generic, also the focus of our work!

2 Revisits of Profiled Cache-Timing Template attack

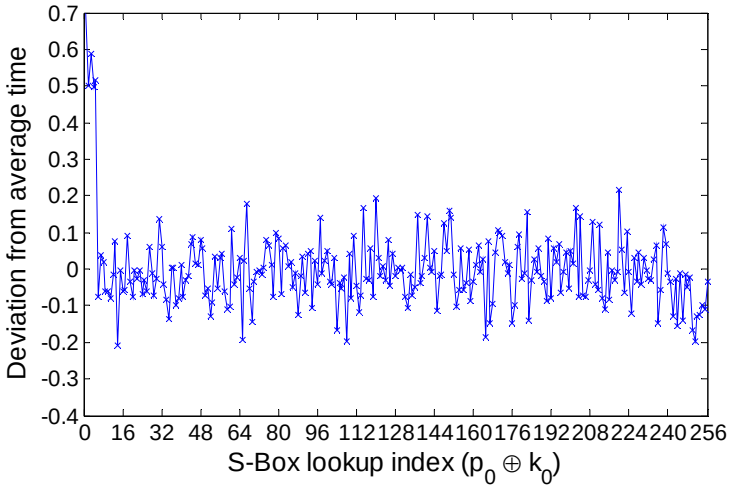
First Proposal

D. J. Bernstein. Cache-timing attacks on AES.
Available online at http://cr.yp.to/papers.html#cachetiming, 2005.

Template server S_{tp}



Target server S_{tg}

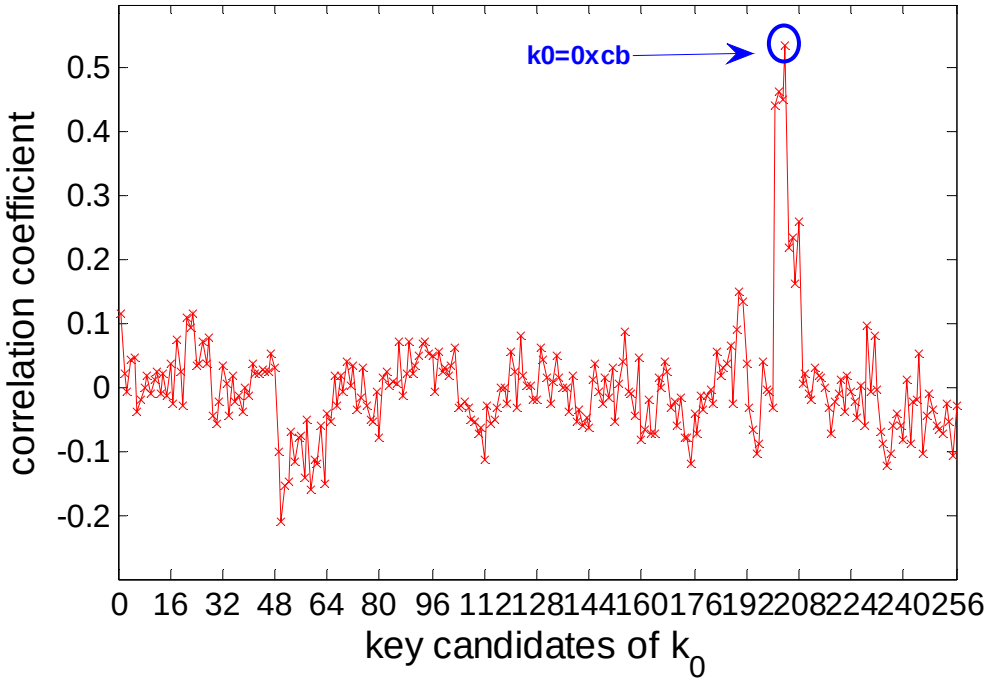


Cache-timing template: average encryption time for different table index.

Attack precondition: The cache-timing templates generated from the two servers are identical.

2 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

2 Revisits of Profiled Cache-Timing Template attack

Advantages:

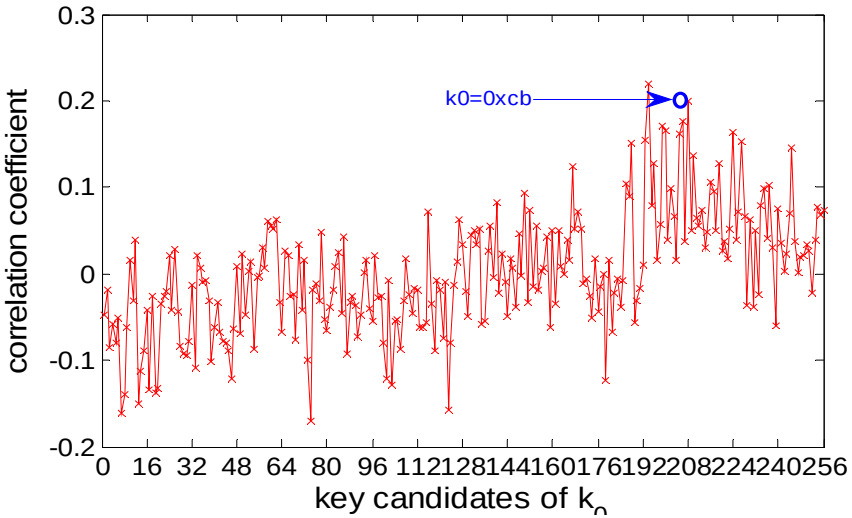
1) simple: only the total encryption time is required

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

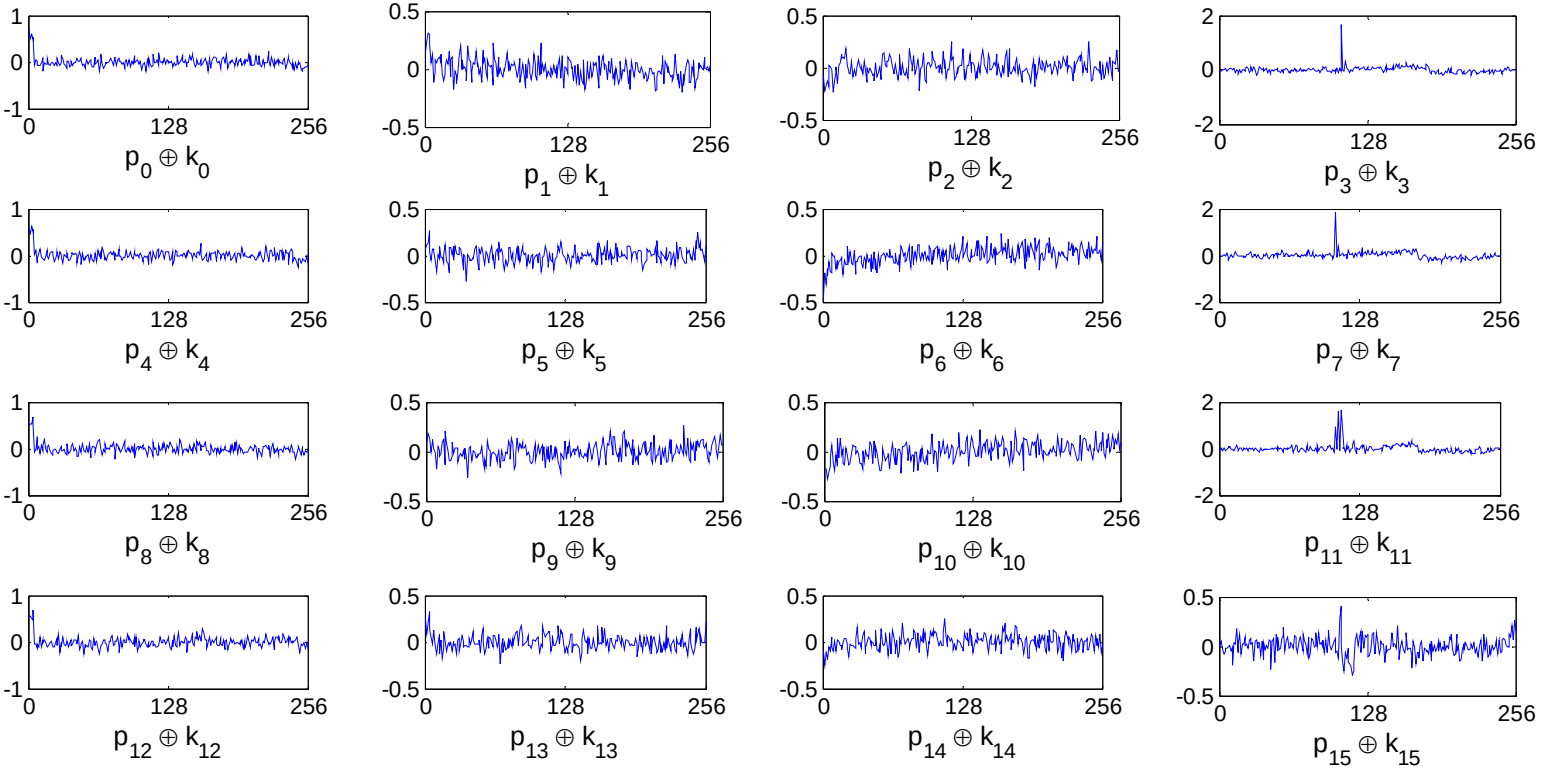


Results of attacking two different servers with the same processor

Can we launch cache attacks without extra template server?

3 Non-Profiled Cache-Timing Template Attack

Main Idea



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

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

2. The templates for lookup different tables are different!

Can we use the internal template for key recovery?

3 Non-Profiled Cache-Timing Template Attack

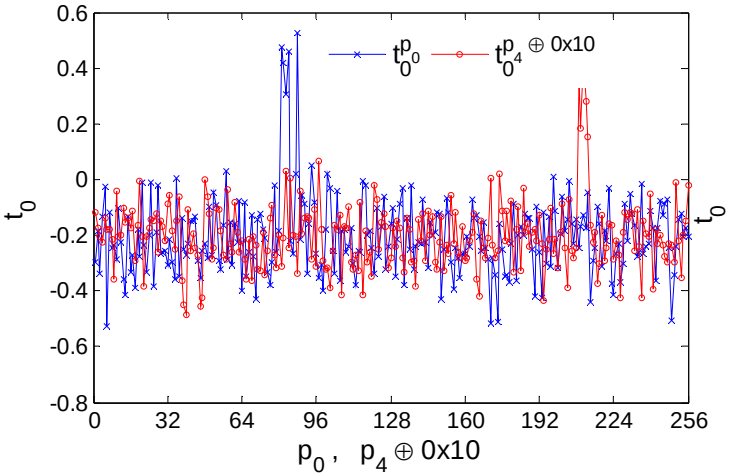
Yes, we can! Below is an example to recover $k_0 \oplus k_4$

1 Build T by p_0

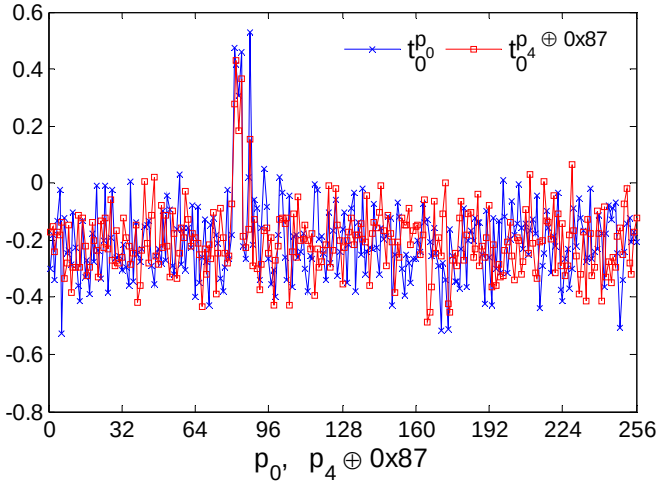
2 Build T' by $p_4 \oplus (k_0 \oplus k_4)'$

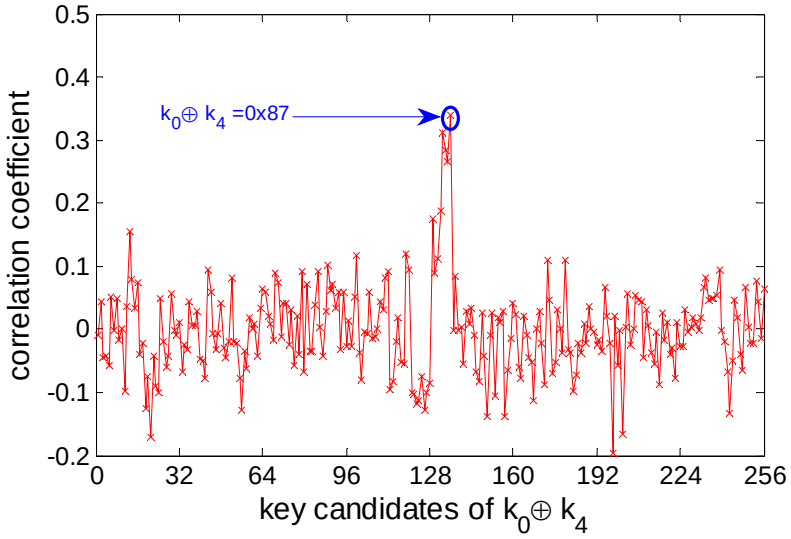
3 Calculate the correlation coefficients, the one with the largest value is related with $k_0 \oplus k_4$

Wrong key guess 0x10



Correct key guess 0x87

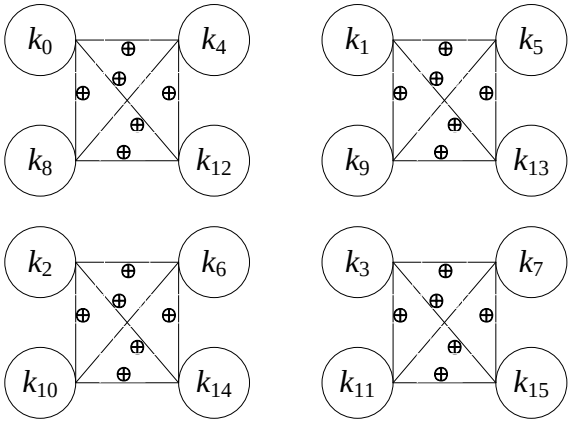




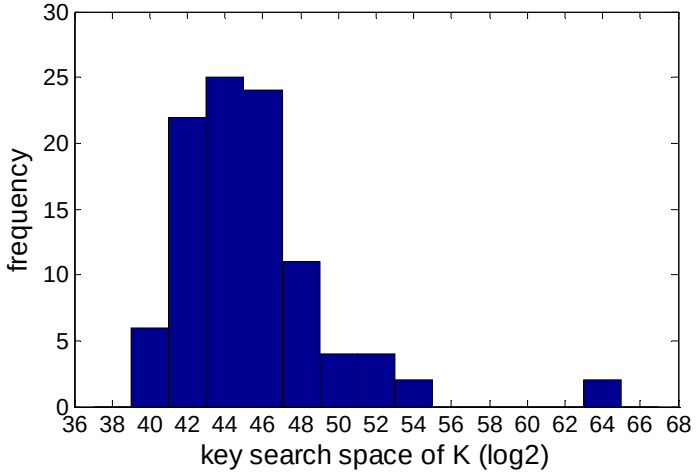
4 With Application to AES

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

Case 1: Attacking the first round of AES in OpenSSL v1.0.0 with four 1KB tables.

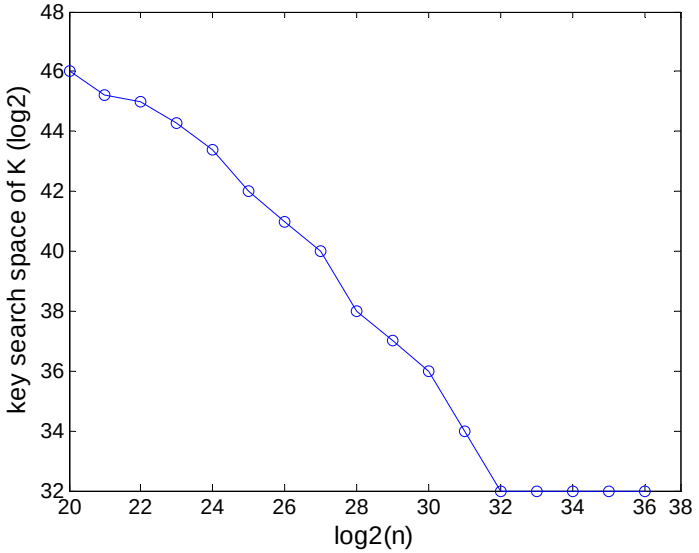


2^{20} samples, 100 attacks



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

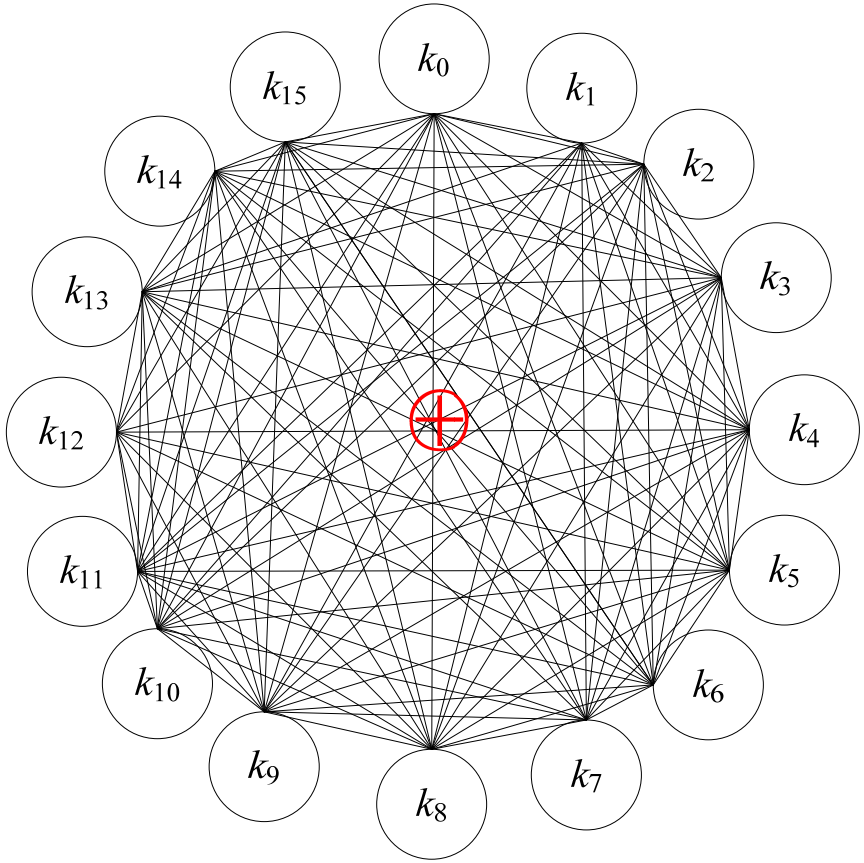
2^n samples



4 With Application to AES

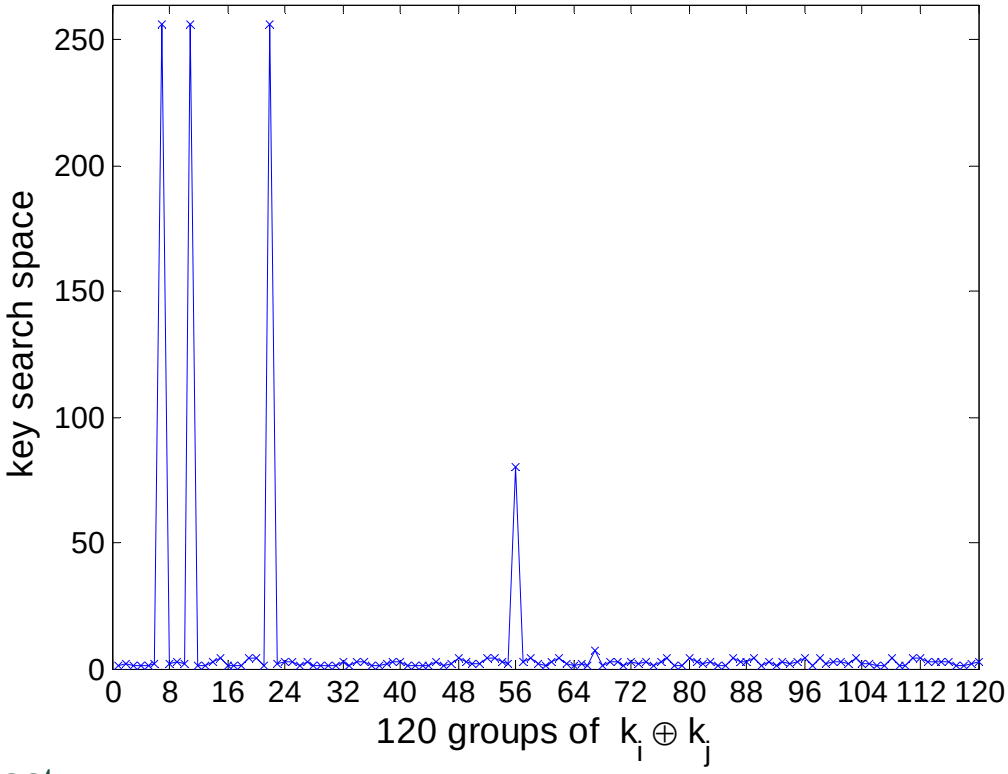
Intel(R) Core(TM) i3-2330 CPU, 2.19 GHZ, 4GB memory, Window7, 64-bit OS.

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.

2^{20} samples, 100 attacks



AES with 2KB table is more vulnerable!

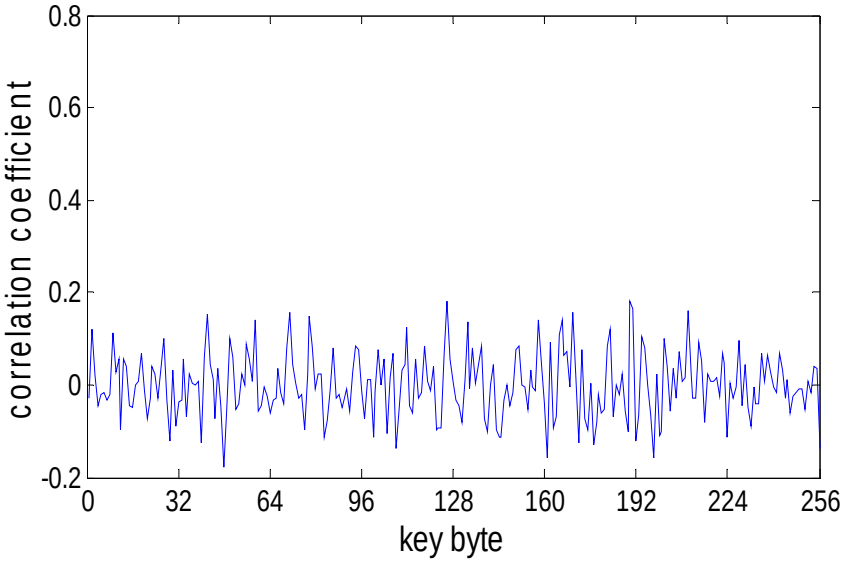
4 With Application to AES

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

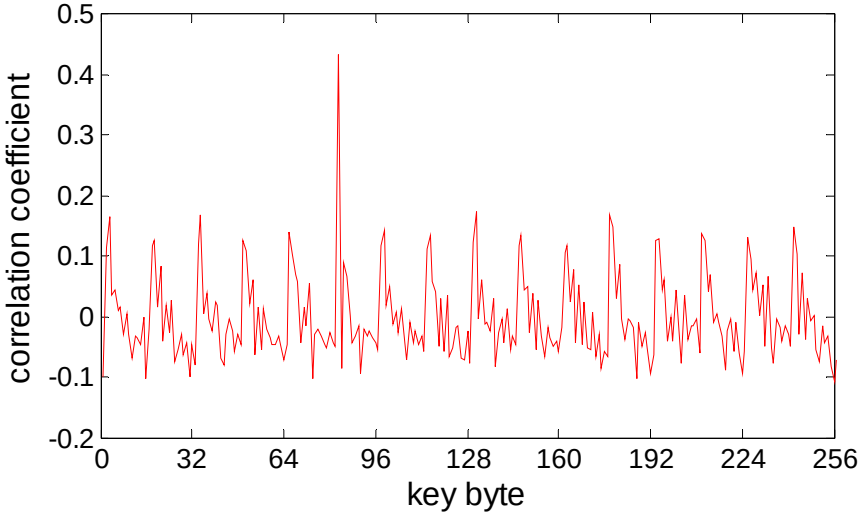
Case 3: Remote attack on the first round of AES in OpenSSL v1.0.0 with one 2KB table.

Preliminary attack results

2^{25} samples



Profiled attack



Nonprofiled attack

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

5 Conclusion

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