第34卷 第21期 计 算 机 工 程 2008年11月
Vol.34 No.21 Computer Engineering November 2008
·博士论文· 文章编号：1000—3428(2008)21—0020—03 文献标识码：A 中图分类号：TN918

基于Cache行为的旁路攻击

张 鹏，邓高明，赵 强，陈开颜
（军械工程学院计算机工程系，石家庄 050003）

摘 要：分析新型高速缓冲存储器(Cache)旁路攻击技术，给出一种Cache旁路攻击方法。针对S盒操作使用查找表处理的数据加密标准(DES)算法实现，通过获取DES加密过程中前2轮加密运算对应的Cache命中信息，结合数学分析方法，可以有效地缩小DES密钥搜索空间。对Cache存储器行为和数学分析攻击进行仿真实现的结果显示，通过2^6个选择明文，大约耗费2^{30}次离线DES加密时间成功地恢复了DES密钥。给出了防御Cache攻击的基本对策。

关键词：旁路攻击；高速缓冲存储器；数据加密标准；S盒

Side Channel Attack Based on Cache Behaviors

ZHANG Peng, DENG Gao-ming, ZHAO Qiang, CHEN Kai-yan
(Dept. of Computer Engineering, Ordnance Engineering College, Shijiazhuang 050003)

【Abstract】 This paper presents a new type of side channel attack based on Cache behaviors. For the implementation of Data Encryption Standard(DES) which takes the operations of looking up tables, with the assistant of mathematical analysis, the search area of the secret key can be reduced effectively by discovering the Cache hit information during the first 2 rounds of DES. The result from the experiment of Cache behaviors-based attack simulation shows that the DES key can be recovered with 2^6 chosen-plaintexts in 2^{30} times off-line DES encryption. Several countermeasures for attack of this type are introduced.

【Key words】 side channel attack; Cache; Data Encryption Standard(DES); S-box

利用高速缓冲存储器(Cache)的行为进行密码破解，是近年来出现的一种新型旁路攻击(side channel attack)方法。这一思想最早是由Kelsey等人[1]提出的，随后Tsunoo等人[2]、Bonneau等人[3]、Page[4]和Bernstein[5]进行了实现。这类攻击使得针对通用计算机密码系统开展攻击成为可能。本文在对基于Cache行为信息的旁路攻击技术进行分析的基础上，对Page的思想进行了部分改进，并仿真实现。

1 Cache基础

为了缓解CPU与主存储器之间的存取速度差，在现代计算机系统中一般都采用了Cache技术。Cache是位于CPU和DRAM主存之间的小规模高速缓冲存储器，它保存着CPU最常用的数据或指令。以数据Cache为例，CPU在取数据时，先在Cache中进行查找，当Cache中保存着CPU要读写的数据时，CPU直接访问Cache，这称为“命中”(Cache hit)。在Cache中不含有CPU所需的数据时需访问主存，这称为“不命中”(Cache miss)。Cache在CPU的读取期间按照一定策略(如LRU，最近最少使用法)淘汰和更新数据。当一次访问数据不超过Cache容量时，发生Cache不命中的主要原因是首次访问该数据或者Cache中的数据被更新。

1.1 地址映射

Cache最重要的设计要素之一是映射策略。通常采用3种映射技术：全相联映射，直接映射和组相联映射。

Cache的数据块大小称为行，主存的数据块大小称为块。行与块是等长的。设i表示主存块号，j表示Cache行号，主存容量为2^m块，Cache容量为2^c行，每个字块中含2^b个字。其中，$i=0,1,2,\cdots,m-1$；$j=0,1,2,\cdots,c-1$。最常见的组相联方式是将Cache的行分成2^{c-r}组，每组2^r行。主存的字块存放到哪个组是固定的，至于存到该组哪一行是灵活的，即有如下函数关系：

$$j=(i \bmod 2^{c-r})\times 2^r + k \quad 0 \leqslant k \leqslant 2^r+1$$

主存地址中的低b位对应块内地址，中间c-r位对应Cache的组地址，高m-c+r位对应Cache行的标志。当CPU给定一个主存地址时，首先用中间的c-r位找到Cache的相应组，然后将高m-c+r位与该组2^r行中的所有标记同时进行比较。哪行的标记与之相符，哪行命中。此后再以主存地址的低b位检索此行的具体字，完成所需要求的存取操作。如果此组没有一行的标记与之相符，即发生Cache不命中。

1.2 装入策略

CPU与Cache之间的数据交换是以字为单位，而Cache与主存之间的数据交换是以字块为单位。一个块由若干个字组成。当发生Cache不命中时，用主存读周期把此字从主存读出送到CPU，与此同时，把含有这个字的整个字块从主存中读出送到Cache中。因此，如果下次CPU访问的字(不一定与上次完全相同)处于这个装入字块中时，将产生一次命中。在本文的实验中，设定的装入字块尺寸是4 Byte(b=2)，这也是最常见的装入字块尺寸。

1.3 Cache行为

即使是同样一条访问存储器的指令，由于涉及到的目标数据当前是否处于Cache中，同Cache的历史状态有关，因

基金项目：国家自然科学基金资助项目(60571037)
作者简介：张 鹏(1976－)，男，博士研究生，主研方向：军事信息安全；邓高明，硕士研究生；赵 强，教授、博士生导师；陈开颜，副教授、博士研究生
收稿日期：2007-12-16 **E-mail**：zhangp210@163.com

此此时是否发生命中是不确定的。这将导致数据访问操作的不确定。

所谓 Cache 行为，就是指在数据访问时，发生 Cache 命中或不命中时系统的操作行为。这种行为可以通过各种旁路物理方式被泄露。如表现在运行时间上，Cache 命中的运行时间将显著小于不命中的时间；表现在功率消耗上，Cache 命中时的功耗将显著小于不命中时的功耗；表现在电磁辐射上，Cache 命中与不命中时的辐射信号将显著不同。这些旁路信息也正是进行 Cache 攻击时利用的信息源。

2 针对 DES 的 Cache 攻击

许多对称加密算法(如 DES, AES)在具体的实现中都采用了查表(lookup table)操作，典型的 DES 加密算法中就有查找 S 盒的操作，其中的 S 盒就是 8 个 4×16 的数组，并且查找 S 盒时引用的下标就含有密钥的信息。这样，通过分析加密过程中 Cache 的“命中”和“不命中”的情况，就能得到查找数组时下标之间的关系，进而推断出密钥的信息。Page 针对 DES 算法进行攻击，用 2^{10} 个选择明文，大约进行 2^{32} 次 DES 运算可以获取密钥。本实验针对通用的 DES 算法[6]，只需要花费大约 2^{30} 次 DES 运算时间就可以获取密钥。

2.1 DES 加密算法

数据加密标准(DES)，采用 64 位的分组长度和 56 位的密钥长度，它将 64 位的输入经过一系列变换得到 64 位的输出。

DES 加密算法的伪码实现如下：

```
void des( M, K[] ) {
  D = IP( M );
  L' = D[ 1 ... 32 ];
  R' = D[ 33 ... 64 ];
  for( round = 1; round <= 16; round++ )    {
    L = R';
    R = E( R' );
    R = R ^ K[ round ];
    R = SB1[ R[ 1 ... 6 ] ] ~ SB2[ R[ 7 ... 12 ] ] ~
    SB3[ R[ 13 ... 18 ] ] ~ SB4[ R[ 19 ... 24 ] ]~
    SB5[ R[ 25 ... 30 ] ] ~ SB6[ R[ 31 ... 36 ] ]~
    SB7[ R[ 37 ... 42 ] ] ~ SB8[ R[ 43 ... 48 ] ];
    R = P( R );
    R = L' ^ R;
    L' = L;
    R' = R;      }
  M = IPr( R' ~ L' );   }
```

其中，符号[1 ... 6]表示取第 1 bit~第 6 bit；~表示比特串的直接相联，如 10~110=10110；*IP*, *E*, *P*, *IPr* 分别表示初始置换、扩展置换、P 置换和最终逆置换；*M* 表示明文。

2.2 查找 S 盒时密钥与明文的相关位分析

针对 DES 的 Cache 攻击需要假设已知 DES 的算法和加密过程中 Cache 的行为。在本攻击实例中只需要知道前 2 轮加密的过程。

从图 1 中可以清楚地知道加密过程中的数据流动。其中 *S*1 和 *S*2 分别代表第 1 轮和第 2 轮的查找 S 盒的置换选择过程。可以看到，查找数组 *S*1 和 *S*2 的 2 个下标 I_1 和 I_2 分别为

$$I_1 = K_1 \oplus E(R_1) \tag{1}$$

$$I_2 = K_2 \oplus E(R_2) = K_2 \oplus E(L_1 \oplus P(S(K_1 \oplus E(R_1)))) \tag{2}$$

考察 8 个 S 盒中第 1 个 S 盒 SB1 的下标(图 2)，得：

$I_1[1\cdots6]= I_2[1\cdots6]$，此时 2 轮查找 S 盒的同一个元素，会产生一个 Cache 命中；

$I_1[1\cdots6]\neq I_2[1\cdots6]$，此时 2 轮查找 S 盒的不是同一个元素，会产生一个 Cache 不命中。

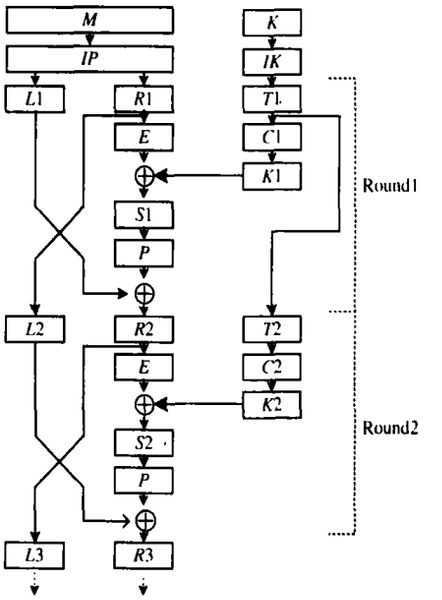

图 1 DES 前 2 轮加密过程

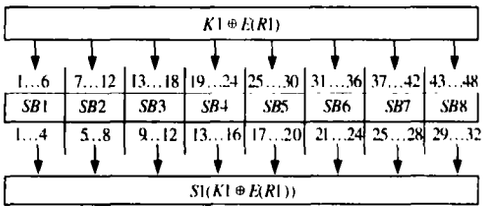

图 2 进入 8 个 S 盒的相关密钥位

由于装入策略的影响，再加上 DES 查找 S 盒时是以 6 位下标中的 1,6 位作为行索引，中间 4 位作为列索引，因而第 4,5 位的值对是否产生 Cache 命中没有影响，因此可得到：

$I_1[1,2,3,6]=I_2[1,2,3,6]$，此时 2 轮查找 S 盒中的同一个字块，会产生一个 Cache 命中；

$I_1[1,2,3,6]\neq I_2[1,2,3,6]$，此时 2 轮查找 S 盒中的非同一个字块，会产生一个 Cache 不命中。

根据 DES 算法的结构可以推导出：

$$I_1[1,2,3,6] = K_1[1,2,3,6] \oplus E(R_1)[1,2,3,6] = K[10,51,34,17] \oplus M[7,57,49,25] \tag{3}$$

$$I_2[1,2,3,6] = K_2[1,2,3,6] \oplus E(R_2)[1,2,3,6] = K_2[1,2,3,6] \oplus (L_1 \oplus P(S(K_1 \oplus E(R_1))))[32,1,2,5] \triangleq K_2[1,2,3,6] \oplus L_1[32,1,2,5] \oplus X = K[2,43,26,9] \oplus M[8,58,50,26] \oplus X \tag{4}$$

其中，

$$X = SB7(K[61,21,38,63,15,20] \oplus M[5,63,55,47,39,31])[1] \sim SB4(K[59,01,36,27,18,41] \oplus M[35,27,19,11,3,61])[4] \sim SB2(K[33,57,02,09,19,42] \oplus M[33,25,17,9,1,59])[3] \sim SB8(K[45,14,13,62,55,31] \oplus M[39,31,23,15,7,57])[1] \tag{5}$$

令式(3)中 $M[7,57,49,25]=[0,0,0,0]$，当 $I_1[1,2,3,6]=I_2[1,2,3,6]$，则有：

$$K[10,51,34,17]=K[2,43,26,9] \oplus M[8,58,50,26] \oplus X \tag{6}$$

式(6)涉及到 56 位完整密钥中的 30 位。通过随机选择满足 $M[7,57,49,25]=[0,0,0,0]$的明文 M(如实验选择 32 个)，可以获得 32 个式(6)，然后对 30 位密钥进行穷举搜索，显然正确的密钥必须满足这 32 个等式。这样将排除大量的不正确密钥。同样的方法分析第 2 个 S 盒可以推断出 56 位密钥中的

29 位，其中有 16 位和第 1 个 S 盒是重复的。所以用 2 个 S 盒可以推断出 56 位密钥中的 43 位。剩下的 13 位密钥可以通过分析更多的 S 盒或者是强力攻击得到。

2.3 对 Page 方法的一点改进

Page 对于 2 个 S 盒具有重复密钥位这一特点进行了利用。他分别针对 2 个 S 盒进行密钥穷举，然后再比较重复密钥位，保留重复密钥位相同的猜测密钥结果，需要 2 次规模相同的密钥穷举操作。本实验对于第 1 个 S 盒得到的每个猜测密钥结果，固定与第 2 个 S 盒重叠的 16 个密钥位，然后针对与第 2 个 S 盒相关的其余密钥位进行穷举操作，这样第 2 次穷举操作的时间将减小到可以忽略不计。由于实验的耗时绝大部分来自于穷举操作，因此这一改进使得攻击方法的时间比 Page 大约减少一半。

2.4 攻击的仿真实现

(1)选择明文。通过物理观测的方法获得 2 组满足式(6)的明文，每组 32 个。用软件仿真了一个 16 KB 的 4 路组关联的 Cache，并且对 Cache 行为(功率)进行监控。在真实攻击的时候，Cache 的功率轨迹可以通过高精度数字示波器采样获取。选择 2 组明文的要求分别是 DES 加密第 2 轮查找第 1 个、第 2 个 S 盒时采样电平为低(表示相关点功率小，亦即产生 Cache 命中)，不符合该条件的随机明文只需简单地丢弃。总共将获取选择明文 $2\times32=2^6$ 个。

(2)获取部分密钥。搜索满足式(6)且与第 1 个 S 盒相关的 30 位部分密钥(至多 32×2^{30} 次判别运算，大约等于 2^{30} 次 DES 运算时间，一般得到 4 个结果)，然后针对每个结果，固定与第 2 个 S 盒重叠的 16 个密钥位，并针对第 2 个 S 盒进行搜索相关的剩余 13 位密钥(运行时间可以忽略)，将得到可能的 43 位密钥(一般只有唯一结果)。这一步可以脱离实际的密码设备，在更高速的计算机上离线进行。

(3)获取剩余密钥。用穷举的方法(至多 2^{13} 次 DES 运算)得到剩余的 13 位密钥。

整个仿真攻击过程在 Celeron 2.53 GHz, 512 MB 内存的计算机上大约需要 13 min。

3 结束语

在访问数组时 Cache 表现出来的与数组下标相关的“命中”和“不命中”的行为特征给密码分析技术提供了一条新的途径。由于普遍采用了查找 S 盒的数组访问操作来实现对称加密算法，使得本文中的 Cache 分析技术具有一定的通用性。这也验证了密码界认同的“S 盒存在脆弱性”的观念。

这种分析技术是以访问数组时 Cache 的行为特性为前提的，因而采用没有查表操作的 DES 实现方式是防御这种攻击的有效途径，Bernstein 提出过一种没有查表操作的 AES 加密算法的实现方式[5]，另一种防御这种攻击的途径就是在加密运算之前对 Cache 进行“预热”，即将 S 盒的内容预先调入 Cache 之中，这将增加攻击的难度。这 2 种方法同样都以牺牲加密运算的速度为代价。

参考文献

[1] Kelsey J, Schneier B, Wagner D, et al. Side Channel Cryptanalysis of Product Ciphers[C]//Proc. of the 5th European Symposium on Research in Computer Security. Louvain-la-Neuve, Belgium, Berlin: Springer-Verlag, 1998: 97-110.

[2] Yukiyasu T, Teruo S, Tomoyasu S, et al. Cryptanalysis of DES Implemented on Computers with Cache[C]//Proc. of the 5th International Workshop on Cryptographic Hardware and Embedded Systems. [S. l.]: Springer-Verlag, 2003: 62-76.

[3] Bonneau J, Mironov I. Cache-collision Timing Attacks Against AES[C]//Proc. of Cryptographic Hardware and Embedded Systems. [S. l.]: Springer-Verlag, 2006: 201-215.

[4] Page D. Theoretical Use of Cache Memory as a Cryptanalytic Side-channel[R]. Department of Computer Science, University of Bristol, Tech. Rep.: CSTR-02-003, 2002.

[5] Bernstein D J. Cache-timing Attacks on AES[EB/OL]. (2005-04-14). http://cr.yp.to/antiforgery/cachetiming-20050414.pdf.

[6] 卢开澄. 计算机密码学——计算机网络中的数据保密与安全[M]. 3 版. 北京: 清华大学出版社, 2003.

(上接第 19 页)

也随之改变，$r_T^1 \oplus r_S^2 \oplus r_S^3$ 未知，即使出现危及某个标签安全，它先前的通信并不能被跟踪。

与文献[6]协议相比，该协议减少了一种 CRC 操作，但增加了 PRNG 的计算；随着标签数目的增加而后台数据库计算复杂性相应地增加；由于 PRNG 设定了种子后，再从随机序列依次提取随机数较简单，因此减少了标签的计算复杂性。

5 结束语

RFID 技术的全面应用的关键是降低标签的成本，而对低成本标签的安全性提高是一个有意义的研究。本文仅使用 PRNG 操作设计的认证协议遵循了 Gen-2 RFID 规范，实现了双向认证、标签的匿名性和前向安全性。今后工作包括：减少后台数据库的计算复杂性；在标签与后台数据库失去同步后，讨论如何恢复同步等问题。

参考文献

[1] Weis S A, Sarma S E, Rivest R L. Security and Privacy Aspects of Low-cost Radio Frequency Identification Systems[C]//Proc. of the 1st Security in Pervasive Computing. [S. l.]: Springer-Verlag, 2003: 201-212.

[2] EPCglobal Inc.. Class 1 Generation 2 UHF RFID Protocol for Communications at 860 MHz ~ 960 MHz[EB/OL]. (2007-08-22). http://www.epcglobalinc.org/standards/uhfc1g2/uhfc1g2_1_0_9-standard-20050126.pdf.

[3] Peris L P. M2AP: A Minimalist Mutual-authentication Protocol for Low-cost RFID Tags[C]//Proceedings of the 3rd International Conference on Ubiquitous Intelligence and Computing. Wuhan, China: [s. n.], 2006: 912-923.

[4] Juels A. Strengthening EPC Tags Against Cloning[C]//Proceedings of the 4th ACM Workshop on Wireless Security. Cologne, Germany: [s. n.], 2005: 67-76.

[5] Karthikeyan S, Nesterenko M. RFID Security Without Extensive Cryptography[C]//Proc. of the 3rd ACM Workshop on Security of Ad Hoc and Sensor Networks. Alexandria, VA, USA: [s. n.], 2005: 63-67.

[6] Duc D N, Park J. Enhancing Security of EPCglobal GEN-2 RFID Tag Against Traceability and Cloning[C]//Proc. of the 2006 Symposium on Cryptography and Information Security. Beijing, China: [s. n.], 2006.