

最新电压毛刺(Power Glitch)攻击与防御方法研究

段晓毅 李 莉 武玉华 靳济芳

(北京电子科技学院电子信息工程系 北京 100070)

摘 要 电压毛刺(Power Glitch)攻击是通过快速改变输入到芯片的电压,使得芯片里的某些晶体管受到影响,引起一个或多个触发器进入错误状态,从而导致处理器会跳过或实施错误的操作,使芯片内隐藏的信息随着产生的错误而泄露出来。对电压毛刺攻击与防御技术的最新进展情况进行了综述。在攻击方面,针对攻击目的的不同,详细介绍了 RSA-CRT 签名运算、RSA 非 CRT 签名运算、对非易失存储器的攻击技术。防御技术分别介绍了电压毛刺检测电路和掩码,并分析了各种防御方案的优缺点。

关键词 电压毛刺攻击,智能卡,硬件安全,RSA 攻击,存储器攻击

中图法分类号 TP393.08 **文献标识码** A

Advanced Evolution of Power Glitch Attack and Resistance Techniques

DUAN Xiao-yi LI Li WU Yu-hua JIN Ji-fang

(Department of Electronics and Information Engineering, Beijing Electronic Science and Technology Institute, Beijing 100070, China)

Abstract Power glitch attacks are fast changes in the power supplied to the device and designed to affect its normal operation. By this way, the cpu can be made to execute a number of completely different wrong instructions. This paper summarized the latest development of power glitch attack and defense techniques. Aimed at various implementations of attack on RSA signature With Chinese Remainder Theorem, attack on RSA signature without CRT and attack on EEPROM were described. Then the available countermeasures of power glitch were shown, which consist of MASK methods, detecting voltage glitch attacks. Moreover, the relative merits of these countermeasures were analyzed.

Keywords Power glitch attack, Smart card, Device security, Attack on RSA, Attack on eeprom

1 引言

自从 Kocher 在文献[1]中提出侧信道攻击的概念后,出现了许多针对嵌入式系统(如智能卡)的攻击。其中由 Boneh 在文献[2]中提出的注入缺陷攻击最为有效。注入缺陷的目的是使目标产生一个非正常的操作,从而使隐藏的信息随着产生的缺陷而泄露出来。注入缺陷的方法有许多种,如改变供电电压、改变外部时钟、改变环境温度、使用激光、x-rays 或离子束等[6]。

毛刺(Glitch)攻击是注入缺陷攻击中的一种,其原理是通过快速改变输入到微控制器的信号,以影响它的正常运行。通常 Glitch 是叠加在电源上或时钟信号上,但 Glitch 也可以是外加的短暂电场或电磁脉冲。每个晶体管和与它相连的线路构成有时延特性的 RC 电路。处理器的最大可用时钟频率取决于该电路的最大延迟。同样地,每个触发器在接收输入电压和由此引起的输出电压之间有个特征时间窗口。这个窗口由给定的电压和温度来确定。如果用时钟 Glitch(比正常的时钟脉冲要短得多)或电源 Glitch(电源电压的快速波动)将会影响芯片里的某些晶体管,会导致一个或多个触发器进入错误状态,处理器会跳过或实施错误的操作[4]。

本文对毛刺攻击中的电压毛刺(Power Glitch)攻击的最

新攻击与防御方法进行综述。

2 电压毛刺攻击技术

电源供应电压的波动会导致晶体管阈值电平的漂移。结果就是一些触发器在采样时它们输入的时间不同,会导致控制器故障,某些电子模块会暂时失效,因此会跳过或实施错误的操作[5],或是从存储器中读出非正确的值。

2.1 控制器故障攻击

如当 CPU 在正常运行时,如果把芯片的电压从 VCC 变到 0 并维持几个纳秒,则处理器会跳过一些指令的执行并且在 Glitch 攻击后的几毫秒内恢复正常的执行[6],如图 1 所示。

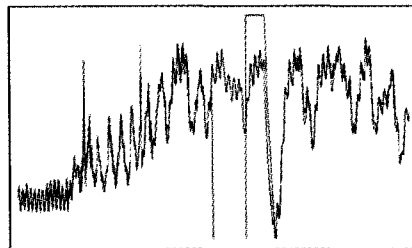


图 1 Glitch 攻击功耗图

图 1 中蓝线表示芯片正常工作时的电压图,红线表示在受到 Glitch 攻击后的图形,从图中可以看出,在受到攻击后,

本文受北京电子科技学院青年基金(YQNJ0907)资助。

段晓毅(1979—),男,博士,讲师,主要研究方向为信息系统安全,E-mail:duanxiaoyi@besti.edu.cn。

芯片停止工作一段时间,在跳过一些操作后继续执行。

2.1.1 攻击 RSA-CRT 签名运算

在文献[7]中介绍了通过 Glitch 攻击 RSA-CRT 签名算法的方法,其原理如下:

$N=p \times q$ 是 RSA 密钥的模, p 和 q 分别是两个大素数。 E 是 RSA 公钥的幂, d 是 RSA 私钥的幂, 并且满足 $e \times d = 1 \pmod{(p-1)(q-1)}$ 。计算 $d_p = d \pmod{(p-1)}$ 为 CRT 计算的幂。 $I_q = q \pmod{p}$ 的逆。则对信息 m 的签名可以通过如下计算得到:

1. $S_p = m^{d_p} \pmod{p}$, $S_q = m^{d_q} \pmod{q}$
2. $S = \text{CRT}(S_p, S_q) = S_q + q \times ((S_p - S_q) \times I_q \pmod{p})$

仅当其中一个模幂运算出错时(仅计算 S_p 或 S_q 出错,而不是两个都出错),可以通过错误的签名值 $\tilde{S}$ 得到 N 的因式分解。例如,在计算 S_p 时产生了错误,得到 $\tilde{S}_p$,则在计算 CRT 时,将会得到错误的签名信息 $\tilde{S} = \text{CRT}(\tilde{S}_p, S_q)$ 。把正确的签名信息 S 和错误的签名信息 $\tilde{S}$ 代入到 $\text{GCD}(S - \tilde{S}, N)$ 中,则可以得到素数 q 。

攻击者通过在计算 S_p 或 S_q 时产生 Glitch,使得计算错误,即可以分别得到 p 和 q 。

2.1.2 攻击 RSA 的非 CRT 签名运算

Schmidt 在文献[8]中提出了一种新的通过 Glitch 对 RSA 算法的攻击方案。其原理是当想得到私钥 d 的某一位值时,在 RSA 运算过程中加入电压毛刺,使处理器产生错误跳过这一位对应的平方运算,最后通过结果推算出私钥 d 的这一位的值。具体操作如下:

对于 RSA 运算中的模幂计算,通常情况下是通过模乘和模平方运算来实现的。模平方和模乘算法可以从幂的最高位开始运行,也可以从最低位开始运行,从最高位开始的方法如下:

```
L_to_R_exp(m,e,N){
  R=1;
  For(i=t; i>=0; i--){
    R=R * R mod N
    If(bit on position i of e=1)
      R=R * m mod N}
```

当攻击者进行攻击时,输入一个明文 m 进行签名操作,在签名过程中首先跳过最后一个平方运算,得到一个错误的签名结果 Sig_0 。可以通过下面的公式计算出正确的签名结果 Sig ,如下

$$\text{Sig} = \begin{cases} (\text{Sig}_0)^2, & e_0 = 0 \\ (\text{Sig}_0)^2 m^{-1} \pmod{n}, & e_0 = 1 \end{cases} \quad (1)$$

可以通过此公式判断对于 e_0 的攻击是否成功。当成功攻击 e_0 得到 Sig_0 后,可以通过计算独立地得到剩下的错误签名值。每一步都包含计算目标比特 e_k 的两个可能值, $k \in \{1, \dots, t-1\}$,

$$\text{Sig}_k = \begin{cases} \text{Sig}_{k-1}, & e_k = 0 \\ \text{Sig}_{k-1} m^{2^{k-1}} \pmod{n}, & e_k = 1 \end{cases} \quad (2)$$

随后,对相关的位的平方运算进行攻击,并得到错误的签名值 Sig_k 与预计算的值进行比较。如果与预计算的两个结果都不匹配,则表示攻击没有成功,再一次进行攻击并获得错误的签名值。否则能成功得到要攻击的 e_k 的值,并对下一位 e_{k+1} 进行攻击。与其它方案比,这个方案能够检查攻击是否

执行成功。

2.2 对数据传输或非易失存储器进行攻击

电压毛刺攻击的另一种目的是通过加入 Glitch,使得从存储器中读出的值非正确^[9]。

在嵌入式系统中,常用的非易失可读存储器一般是 EEPROM 或者是 FLASH。EEPROM 采用了一种称为 FLOTOX (floating-gate tunneling oxide) 晶体管浮栅器件作为可支持电擦除过程的可编程器件,如图 2 所示。当把一个约 10V 的电压加到这一很薄的绝缘层时,电子通过隧穿机理穿入或穿出浮栅。

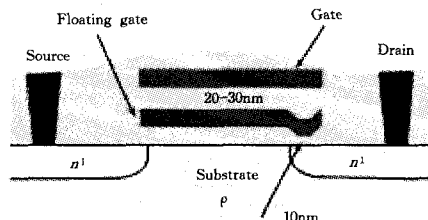


图2 FLOTOX 三极管

这就带来了阈值控制的问题, V_t 可能低于 0V,从而有效地产生一个即使栅上施加 0V 电压也不会关闭的器件。使用选择器件的原因是阈值电压很难精确地控制,因为这取决于生产过程中的一些变化和浮栅上初始存储的电荷。如果由于初始储存的电荷使得不可能可靠地达到希望的内部电压,那么存储器单元将不能正确工作。为了避免这个问题,与 FLOTOX 晶体管串联一个选择管,并连接到字线和位线。选择管使用正常的字线电平,而 FLOTOX 晶体管有一个位于两个可能阈值之间的合适的栅电压,如图 3 所示。

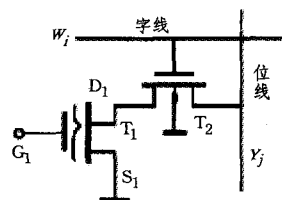


图3 EEPROM 存储单元

当对 EEPROM 进行读时,如果在读期间,EEPROM 的电压增加,使得电压超过阈值,则不论 EEPROM 里面的真实值是多少,读出的值都将会是 0^[4,8]。

使用 Glitch 攻击可以通过如下方式取得存储在芯片中的密钥,当密钥正从 EEPROM 传输到 RAM 时,攻击者产生一个电压脉冲,从而使读出的密钥为固定值 00。为了攻击得到 n 个字节的密钥,攻击者可以在芯片运行时产生 $n-1$ 个电压脉冲,从而使从 EEPROM 中读出的密钥为 00 00...00 00 xx 00 00...00 00,使用这个错误的密钥对已知明文进行加密得到结果 C ,由于整个密钥中未知部分是有一个字节的 xx ,因此通过 256 次的暴力破解,可以获得 xx 的值,通过在不同的位置产生 Glitch,攻击者可以完全获得整个 n 个字节的密钥^[6]。

文献[4]给出了通过 Glitch 对 MC68C05B6 的攻击案例,当 MC68C05B6 在执行 AND &0100 指令时,如果电源电压减少 50%~70%,取出的值是 0FFH 而不是实际的值,这会对对应熔丝的加密状态。

- [5] 黄显明. 智能卡攻击技术分析及安全防范策略综述[J]. 金卡工程, 2008, 12(4): 49-52
- [6] Bar-El H, Choukri H, Naccache D, et al. The Sorcerer's Apprentice Guide to Fault Attacks[J]. Proceedings of the IEEE, 2006, 92(2): 370-382
- [7] Boneh D, DeMillo R A, Lipton R J. On the importance of eliminating errors in cryptographic computations [J]. Journal of Cryptology, 2001, 14(2): 101-119
- [8] Schmidt J-M, Herbst C. A Practical Fault Attack on Square and Multiply[C]// Fault Diagnosis and Tolerance in Cryptography (FDT08). Washington DC, USA, August 2008, 10: 53-58
- [9] Kommerling O, Kuhn M. Design principles for tamper resistant smartcard processors[C]//Proc. USENIX Workshop on Smart-card Technology, 1999: 9-20
- [10] Kim E S, Kim J H. Voltage glitch detection circuits and methods thereof[P]. US Patent Office, US007483328B2, Pub. Date: Jan. 2009
- [11] Kim C Y, Jun S J, Kim E S. Voltage-glitch detection device and method for securing integrated circuit device from voltage glitch attack[P]. US Patent Office, US 20030226082A1, Pub. Date: Dec. 2003
- [12] Yanci G, Pickles A, Arslan S, et al. Detecting Voltage Glitch Attacks on Secure Devices[C]//ECSIS Symposium on Bio-inspired Learning and Intelligent Systems for Security (BLISS '08). Aug. 2008: 75-80
- [13] 马哲. 一种防电源毛刺攻击的检测电路[P]. 中国国家知识产权局, 200910088706. 7, 2011-01-12
- [14] Chari S, Jutla C, Rao J R, et al. A cautionary note regarding evaluation of AES candidates on smartcards [C]//Proc. of the 2nd AES Candidate Conference. Rome, Italy, 1999: 133-147
- [15] Ciet M, Joye M. Practical fault countermeasures for Chinese Remaindering based RSA[C]// Fault Diagnosis and Tolerance in Cryptography (FDT05). Edinburgh, Scotland, UK, September 2005: 124-131
- [16] Giraud C. Fault resistant RSA implementation[C]// Fault Diagnosis and Tolerance in Cryptography (FDT05). Edinburgh, Scotland, UK, September 2005: 142-151
- [17] Kim C H, Quisquater J-J. Fault Attacks for CRT Based RSA: New Attacks, New Results, and New Countermeasures [J]. Information Security Theory and Practices. Smart Cards, Mobile and Ubiquitous omputins Ssystems lecture Notes in Computer Science, 2007, 4462: 215-228

(上接第 420 页)

```

FilterChain chain) throws IOException, ServletException {
    HttpServletRequest request=(HttpServletRequest)req;
    HttpServletResponse response=(HttpServletResponse)res;
    HttpSession session=request.getSession();
    String login=(String)session.getAttribute("user-
        name");
    if(login==null)
    {response.sendRedirect("/ni/index.htm");}
    chain.doFilter(req,res);}

    public void init(FilterConfig arg0) throws ServletException {}

```

因为没有登录的用户 SESSION 是空的,所以未登录的用户是无法登录要过滤的页面。只建立过滤器类并不能实现页面的过滤,还要在 Myeclipse 的. xml 文件中加入以下代码,也就是把要过滤的网页放到过滤器中,代码如下:

```

<filter><filter-name>user</filter-name>
    <filter-class>db.GLQ</filter-class></filter>
.....//部分代码省略;
<filter-mapping><filter-name>user</filter-name>
    <url-pattern>/br.jsp</url-pattern></filter-mapping>

```

4.2 客户端设计

客户端用户登录过程由两个部分组成:第一部分是可视化界面部分;第二部分是用户登录过程部分。登录过程利用 AJAX 异步通信机制,允许客户端只发送用户名和密码到服务器,这样,客户端无需提交整个页面,从而减小了传输的数据量^[6]。

登录过程需要执行两个操作:第一个操作是收集信息;第二个操作是发送请求到服务器。

在操作收集过程中,用户输入用户名、密码字段,程序将其放在一个要提交到服务器的字符串中。然后将这些值通过 AJAXUpdater 对象提交到服务器。

AJAXUpdater 对象是实现 AJAX 异步通信的方法,它可以接受参数,包括目标 URL、请求成功时调用的函数、请求失败时调用的函数、使用 HTTP 方法以及包含提交参数的字符串^[7]。

AJAXUpdadter 对象将会等待服务器返回一个 XML 文

档,当数据从服务器正确返回时,将调用 MakeScrip() 函数, MakeScrip() 函数将利用从服务器端获取的数据创建登录后的窗口界面^[8]。

通过对服务器端和客户端的设计和实现,用户可以登录 BBS 论坛,在输入框中依次输入姓名和密码,点击“登录”按钮,此时输入框下“正在登录”信息提示用户客户端发送客户信息(而不是提交整个页面)到服务器端。如果服务器在数据库中找到用户信息,那么用户登录成功。此后,服务器在数据库中寻找用户的配置信息和用户登录的网站地址,返回给用户的回调函数。回调函数根据返回的信息创建 3 个子窗口装载相关页面,作为登录后的窗口界面。

结束语 根据本文所提供的方法构建基于 AJAX 的富客户端,大大提高了响应速度,降低了传统 Web 模式下的等待,实现了接近桌面型应用程序的用户体验。尽管目前 AJAX 还处于发展阶段,尚存不足之处,需要进一步的解决,但是随着技术的不断更新和完善,使用 AJAX 构造富客户端的模式必将会给传统的 Web 应用程序带来一场新的变革,成为 Web 的新动力 AJAX 技术将有光明的前景。

参 考 文 献

- [1] Gross C. AJAX 最佳模式与实践[M]. 北京:电子工业出版社, 2007
- [2] Eichorn J. 深入理解 Ajax: 基于 JavaScript 的 RIA 开发[M]. 北京:人民邮电出版社, 2007
- [3] 杨仁和. AJAX 设计模式[M]. 北京:电子工业出版社, 2010
- [4] 刘宜兰. smart Client 就在前方[J]. 程序员, 2008(3)
- [5] 杨国瑞, 张思博. 基于 Ajax 的 Web 应用架构设计[J]. 现代电子技术, 2006, 15(230): 95-98
- [6] Asleson R, Schutta N T. Foundations of Ajax[M]. 北京:人民邮电出版社, 2006: 24-36
- [7] McCarthy P. 面向 Java 开发人员的 Ajax: 构建动态的 Java 应用程序, 2005
- [8] 崔瑶, 钱小江. 基于 Ajax 的动态搜索功能设计和应用[J]. 网络安全与技术应用, 2007, 12(5): 61-64