键盘输入安全研究

李鹏伟[1,2] 傅建明[1,2,3] 沙乐天[1,2] 丁 爽[1,2]
(武汉大学计算机学院 武汉 430072)[1]
(武汉大学空天信息安全与可信计算教育部重点实验室 武汉 430072)[2]
(武汉大学软件工程国家重点实验室 武汉 430072)[3]

摘 要 键盘记录器是网络用户面临的主要安全威胁之一。以保障敏感信息的安全输入为出发点,分析了按键信息传输流程,系统总结了键盘输入信息所面临的来自物理层、内核层、应用层的截获、查询、旁路等类型的安全威胁以及现有研究和应用的相应防御措施;介绍了软键盘所面临的偷窥、消息截获、截屏等威胁及其防御措施,提出鼠标记录攻击、控件分析攻击等新的威胁以及相应的对策;然后对现有研究和应用的敏感信息输入进行了安全性测试;最后介绍了现有研究中基于行为的键盘记录器检测方法。

关键词 键盘记录器,敏感信息输入,键盘,软键盘
中图法分类号 TP309 **文献标识码** A

Research of Keyboard Input Security

LI Peng-wei[1,2] FU Jian-ming[1,2,3] SHA Le-tian[1,2] DING Shuang[1,2]
(School of Computer Science, Wuhan University, Wuhan 430072, China)[1]
(State Key Laboratory of Aerospace Information Security and Trusted Computing, Ministry of Education, Wuhan University, Wuhan 430072, China)[2]
(State Key Lab of Software Engineering, Wuhan University, Wuhan 430072, China)[3]

Abstract Keylogger is one of the most serious threats to the Internet users. In order to protect sensitive information input, the study summarized the threats along the process of information inputting using a keyboard and corresponding preventive measures. These threats work at different levels (physical/ring 0/ring 3) and use different attack approach (Query/Hook/Bypass). We summarized the threats faced by soft keyboard such as peep, screenshots, or information intercepted. We also proposed new attacks based on mouse behavior record, element analysis and measures to defense these attacks. We then tested the performance of security measures which are employed by existing research and applications. At last, the existing research in behavior-based keyloggers detection was summarized.

Keywords Keylogger, Sensitive information input, Keyboard, Virtual keyboard

1 引言

用户使用网络游戏、网上通讯和网上购物等互联网服务之前,必须通过键盘输入用户身份的凭证,如用户名和口令,以实现用户身份的认证、后台服务的访问授权。这些凭证信息往往成为攻击者的攻击目标。旁路攻击可以间接获取身份凭证,如利用键盘输入录像[6]、按键声音[46]、按键振动[42]、按键手姿等获得键盘输入内容,这需要攻击者离键盘距离比较近,且获取的凭证不准确。键盘记录器(Keylogger)也称按键嗅探器(Key Sniffers)、间谍木马程序(Trojan-Spy),是具有击键记录功能的恶意代码,一般通过消息截获或者对特定位置进行轮询等信息流攻击来实现。键盘记录器具有远程信息获取能力且捕获的精度较高。目前,键盘记录器发展迅速,已成为最常用的信息窃取技术[4,26,48],形成了完整的产业链[1,2],是互联网用户面临的一个主要安全威胁[40]。

键盘记录器的工作流程可分为安装键盘记录器、记录按键信息、上传按键信息3个阶段,每一阶段有其对应的防御和检测措施。首先,键盘记录器可能需要加载驱动或注入线程,现有的杀毒软件、主动防御软件可以检测到该行为,并阻止其安装。其次,干扰或屏蔽键盘的记录可以降低键盘记录器的准确性,从而提高身份凭证输入的安全性[50]。最后,可以利用键盘记录器间断性上传按键信息的行为来对此类程序进行检测[13]。现有研究提出了利用键盘记录器各行为特征(记录按键信息的行为[51]、存储按键信息的行为[10]、上传按键信息的行为[13])以及各个行为之间关联性[11]识别和检测键盘记录器的多种方法。

本文第2节介绍通过敲击物理键盘输入信息的流程,以及针对该流程存在的信息流攻击和旁路攻击;第3节讲述与

到稿日期:2012-12-17 返修日期:2013-04-15 本文受国家自然科学基金项目(61202387),国家科技重大专项项目(2010ZX03006-001-01),高等学校博士学科点专项科研基金(20120141110002)资助。
李鹏伟(1987—),男,博士生,主要研究方向为恶意代码,E-mail:xu_pang@163.com(通信作者);傅建明(1969—),男,教授,博士生导师,主要研究方向为网络安全、系统安全;沙乐天(1985—),男,博士生,主要研究方向为漏洞挖掘、网络安全。

键盘记录器共生的安全机制，如加密、扰动、路径变化等；第 4 节介绍软键盘原理并对其安全性进行分析；第 5 节实验测试常用键盘和软键盘输入的安全性，其结果显示键盘记录器的安全威胁依然严重；第 6 节综述现有研究通过程序行为检测键盘记录器的各种方法。最后总结全文。

2 键盘输入的流程和威胁

物理键盘主要有 PS/2 键盘、USB 键盘、无线键盘 3 类。其中，USB 键盘由于 USB 设备本身存在信息传递、缓存清理方面的安全缺陷，因此信息容易泄露；无线键盘在输出键盘信息时，其无线信号容易被其他设备捕获[35]。2007 年，韩国政府强烈推荐在进行电子交易时使用 PS/2 键盘[17]。目前，笔记本电脑及大部分台式机使用的键盘为 PS/2 接口，因此对 PS/2 键盘的输入流程进行分析。

2.1 物理键盘的信息输入流程

Windows 操作系统中，PS/2 键盘输入信息的传递流程如下：

1）用户敲击按键，一个按键被按下时，键盘发送相应电信号到计算机主板上的键盘控制器（i8042）；

2）键盘控制器告知 CPU 有键按下，同时将按键信息以键盘扫描码的形式写到键盘 I/O 接口（其中 Ox60 端口保存按键扫描码，Ox64 端口记录键盘控制器的状态），并产生中断请求 IRQ1；

3）操作系统根据 IOAPIC 重定位表查到 IRQ1 所对应的中断号（正常情况为 0x93），再根据中断向量表（IDT）查得这一中断对应的中断处理函数的地址，调用中断处理程序（ISR）进行处理；ISR 读出 0x60 端口的键盘扫描码，将之转换成系统扫描码，封装包含按键信息的 IO 请求包（IRP），将 IRP 发往键盘端口驱动（PS/2 键盘的端口驱动为 i8042 prt. sys）；

4）键盘端口驱动将按键信息发往键盘类驱动（Kdbclass. sys，所有类型键盘通用）；

5）键盘类驱动将按键信息封装到系统消息中发往 csrss. exe，按键信息首先被保存到系统消息队列中；

6）csrss. exe 将按键信息分发到各个应用程序的线程消息队列；

7）焦点窗口所属的程序从线程消息队列中读取被转为 ASCII 码（如果需要，还需要经过输入法编辑器 IME 的处理）的按键信息，并调用 use32. dll 更新用户界面。

表 1 键盘输入的直接威胁

攻击类型|层次	攻击方式	位置
截获|物理	硬件键盘记录器：通常位于键盘和机箱的连接处，通过记录电信号得到按键信息	1
查询|Ring 0	端口扫描：直接访问键盘的端口（0x60）获得按键扫描码[28]	2
截获|Ring 0	修改 IOAPIC 重定位表：将 IRQ1 重定位到自定义的键盘中断项（不再是原来的 IDT 0x93），从而获取按键信息	2,3 之间
截获|Ring 0	修改 IDT 中断向量表：通过改变 IDT 0x93 对应的中断处理函数的地址，使其让其指向自定义的中断处理函数（不再是原来的 ISR）	2,3 之间
截获|Ring 0	Hook 分发函数：修改处理按键的驱动（可以是端口驱动，也可以是类驱动）对象的分发函数的指针，使之指向自定义的驱动程序	3 或 4
截获|Ring 0	键盘驱动过滤：枚举驱动对象下的所有设备，并创建一个过滤设备附加上去，从而获取包含按键信息的 IRP	4
截获|Ring3	全局键盘钩子：利用 WH_KEYBOARD_LL Hook 函数在键盘消息被投递到线程输入队列前截获消息。如果某个进程涉及键盘操作（调用 keybd_event 函数），线程环境就会被切换到 WH_KEYBOARD_LL Hook 的线程环境，记录按键信息后再切换到原进程[30]	5
查询|Ring3	GetRawInput：键盘记录器注册想要获取原始输入的设备，接收 WM_INPUT 消息[31]	5
截获|Ring3	日志钩子：WH_JOURNALRECORD Hook（日志钩子）用来监视和记录输入事件，使用 WH_JOURNALPLAYBACK Hook 来回放[27]	5
查询|Ring 3	读取按键状态：使用 GetKeyState、GetKeyboardState、GetasyncKeyState 等函数轮询按键状态（按下/弹起）。其中 GetasyncKeyState 通过访问系统消息队列获取按键状态，GetKeyState、GetKeyboardState 则是通过访问线程消息队列实现	5 或 6
截获|Ring3	键盘钩子：使用 WH_KEYBOARD Hook 来监视线程消息队列中的键盘消息，当焦点窗口所属线程从线程消息队列中获取键盘消息时就会调用键盘钩子函数。需要进行线程注入[32]	6
截获|Ring3	消息钩子：利用 WH_GETMESSAGE 可截获输入法编辑器（IME，Input Method Editor）发往应用程序的消息，可以记录汉字	7

2.2 信息流攻击

如图 1 所示，键盘信息的传输过程中面对多种威胁。其中键盘输入的信息流威胁主要是指键盘的输入信息在传送过程中被控制和分流，包括输入信息的截获（Hook）和对输入信息的查询（Query）。二者的区别在于后者仅能够获取信息，前者还可以控制按键信息，比如屏蔽、修改等。详细的攻击参见表 1，其中攻击位置对应图 1 中左侧数字。

其中，硬件键盘记录器是在按键信息以电信号的形式传到计算机主板的过程中窃取按键信息，通常位于机箱和键盘连接处[8,9]。攻击者至少需要一次接触目标计算机的机会。硬件记录器工作于物理层，可以记录到所有按键信息且难以通过软件手段对其进行检测，在近年来得到了很大发展，已经开始商业化生产并大量销售。此类键盘记录器通常可以记录数百万个按键值，部分记录器可以通过无线电波方式发送所记录的按键信息。文献[52]设计实现了一种利用隐蔽时间信道传出按键信息的键盘记录器 JitterBug。JitterBug 适用于按键信息被立刻上传的场景，通过轻微延迟按键信息来编码输出所记录到的按键信息，从而解决了 Keylogger 的信息传出的问题，可以形成一种“供应链攻击”，可通过生产大量此类键盘来窃取信息。

键盘记录程序如果可以加载驱动程序，则有多种方法可以在按键信息到达系统消息队列之前获得按键信息[33]。内核层（Ring 0）键盘记录器实现难度较大，但记录能力较强。上文中作用于键盘类驱动（kdbclass. sys）之前的记录器只能记录 PS/2 键盘的击键信息。应用层（Ring 3）键盘记录器则使用特定 Windows API 在应用层获取按键信息，实现简单，可以记录多种键盘的按键信息，得到了更广泛的应用。

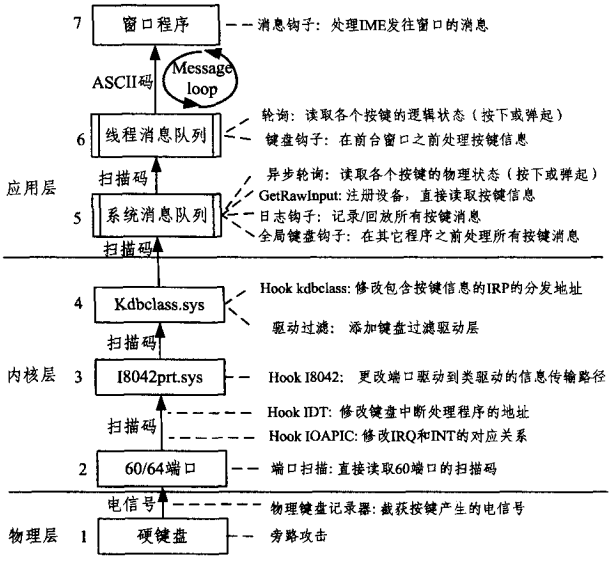

图 1　键盘信息传送流程及安全威胁

2.3　旁路攻击

在用户敲击键盘的过程中。攻击者有可能通过肉眼或者摄像头窥视目标用户的击键行为。文献[6]设计实现了通过录像得出键盘输入内容的自动化方法 ClearShot。ClearShot 通过分析手的动作与按键，区分出最可能的击键，排除不可能的击键，然后利用语言分析、内容分析推断输入内容，重组句子并更正错误。

也有设备可以根据击键声音传播的角度和距离来推断出按键。文献[49]设计了利用键盘外部的录音设备所记录的音频信息推断出键盘输入内容的方法，该方法通过机器学习和拼写检查提高了推断的准确性，其中文章的正确恢复率为 96%，90%的 5 字母密码可以通过少于 20 次的尝试得出，80%的 10 字母密码可以通过少于 75 次的尝试得出。

文献[42]利用 iPhone 的感应器分析键盘振动来判断按键，其具体方法可分为学习、攻击两个阶段。学习阶段提取出各个字符所对应的原始加速度传感数据特征，包括单个字符的特征和字符对的特征（例如 canoe 包括 C、A、N、O、E 5 个字符和 ca、an、no、oe 4 个字符对）。攻击阶段获得原始传感器数据，通过 LR（左右，针对字符）、NF（远近，针对字符对）两个神经网络模型判断字符的左右和字符对的远近，查词库获得单词（例如 LLN. LRF. RRF. RLF 对应 canoe），正确率高达 80%。

随着摄像、录音、传感器技术的发展，旁路攻击的可能性日渐增加。此类攻击准确性有限，但实现相应防御十分困难，一般只能通过用户的警惕性或者使用软键盘来提高输入信息的安全性。

3　键盘输入安全防护

键盘记录器功能简单，形式多样，且未必具有寄生、自我复制、网络连接等恶意代码常见特征，因此难以准备检测。另外很多正常软件也存在键盘记录模块，区分键盘记录行为本身是否合法也是一个难题[36]。所以有必要将敏感信息和普通信息的键盘输入相区分，为敏感信息的输入设置专门的保护方案，即在计算机上存在键盘记录器的条件下，通过一定方法使得键盘记录器记录无法正确记录键盘输入的敏感信息。常见方法是在键盘信息处理的较早阶段对按键信息进行加密、添加干扰字符或者改变按键信息的传输流程。在“上游”进行此类处理后，“下游”程序将难以获取正确的按键信息。

3.1　物理层防御

可通过芯片加密的方式抵御硬件键盘记录器的威胁。例如文献[9]提出了一种对物理键盘进行加密的方法：在键盘内的芯片进行加密，在主板芯片处进行解密。这一做法可以使最常见的作业于键盘与机箱连接处的硬件键盘记录器记录不到正确的按键信息。实际应用中，也可以将解密工作方法放在更上层进行（如浏览器中），从而避免各种软件键盘记录器的威胁。此类方法涉及硬件，部署实施较为困难。此外，对于旁路攻击，即基于录像、声音、振动等的记录方式，在芯片处进行加密是无效的。

3.2　内核层防御

随着 rootkit 键盘记录器的增多，现在的防护措施更趋向于在内核层进行。常见的有以下方法。

1）硬件虚拟化

近年来硬件虚拟化得到了快速发展。据此，文献[44]设计了基于硬件辅助虚拟化技术的反键盘记录器模型。文献[44]利用 Intel-VT 技术实现了 VMM，以监控键盘消息的处理过程。WMM 可以直接控制 60 端口，因此可以对现有所有软件键盘记录器形成有效防御。其缺陷主要在于该方法仅适用于保护虚拟机内的键盘输入，且需要硬件支持（CPU 必须支持硬件辅助虚拟化），应用范围有限。

2）在 0x60 端口处插入干扰信息

文献[14]提出了在 0x60 端口处插入干扰信息的具体方法。由于 0x60 端口位处键盘信息传输链的首段，在此处加入干扰信息之后，各种键盘记录程序记录的按键信息就包含了很多用户并未实际输入的干扰字符。

3）Hook IOAPIC

IOAPIC 重定位表标识每个 IRQ 被重定位到哪个中断处理函数，可以对此表加以修改，将 IRQ1 重定位到定制的安全 IDT 项对键盘信息进行处理[29]，绕过之后的键盘记录程序。

4）Hook IDT

同样，可以自定义一个 ISR 去接管键盘中断[16]，即修改 IDT 中断处理程序的地址，使之指向自定义的函数，此函数可直接将数据从 0x60 端口读出，进行安全的处理。

5）驱动层加密

添加自定义的驱动过滤层，对数据进行加密后再传给原来的键盘驱动程序[37]。

3.3　应用层防御

在应用层进行防御的好处是适用于所有键盘（PS/2 键盘、USB 键盘、无线键盘，甚至虚拟键盘），且在应用层实现加密、加入干扰信息、改变信息传输流程等都较为方便。

以加入干扰信息为例，通过逆向分析可知 QQ2012 的密码框采用如下安全措施：1）利用调试钩子（WH_DEBUG Hook）禁止调用键盘钩子；2）利用全局键盘钩子（WH_KEYBOARD_LL Hook）来加入干扰信息并进行键码转换；3）通过不断安装、卸载全局键盘钩子来保证自己的钩子处于钩子链的最前端，以在其它键盘钩子之前获取按键信息。文献[15, 50]设计了生成、添加和移除干扰信息的具体方案。文献[15]利用“焦点切换技术”来保证所添加的干扰信息不影响用户的正常输入：在用户输入字符的间隔将焦点切换到登录框以外

的程序后产生干扰按键信息，然后再将焦点切换回来。采用此类措施后，应用层键盘记录程序将受到虚假按键的干扰。文献[50]利用一个自定义的 keynoise. dll 实现按键干扰信息的添加和移除，其中添加干扰信息的位置处于系统消息队列之前(图 1 中 4,5 之间)，通过 keybd event()实现，移除干扰信息的位置为调用 use32. dll 前(图 1 中 7 处)，从而避开了几乎所有应用层键盘记录程序。文献[50]同时对产生具有较强迷惑作用干扰信息的具体方法进行了研究。

整体来说，如上文所述，所有“截获”型键盘记录方法同时可用来进行加密、插入干扰字符、改变按键信息传输流程等各种处理。其中插入干扰字符的方法除了可以用“拦截”的方法实现外，还可以通过写端口、加入消息等方式实现，因此应用较为广泛。

就攻防强度而言，越底层其强度越大。各防御方法对于更底层的攻击无效，例如对于驱动层加密，端口扫描、Hook IOAPIC、Hook IDT 等方法仍可记录按键。文献[14]在 60 端口处加入干扰信息，Hook IDT、Hook IOAPIC、驱动过滤等之后的攻击就会受到干扰信息的影响；hook IOAPIC 在 i8042prt. sys 之前改变信息传输流程，i8042prt. sys 处或之后的攻击就无法截获按键。同一层次的攻防则胜负未知，也可能会导致系统错误(如 IDT 项的反复修改)。应用层的防御方法的主要缺陷是对于内核层和物理层的攻击无效。就适用范围而言，越底层的方法，其应用范围越窄。类驱动层及其以上方法适用于所有物理键盘；端口驱动层方法则仅适用于特定键盘；hook IOAPIC、端口扫描等方法还需考虑 CPU 和主板的型号。因此现有研究和应用中，各层次的攻防方法都得到了一定应用。

3.4 新的敏感信息输入方式

为了避免键盘记录的威胁，部分研究和应用采用了新的认证方式。一些应用采用“提示输入一部分秘密信息”的方法：例如用户密码为 8 位的字符串，登录时服务端随机生成 4 个小于 8 的随机数，例如“7、1、5、6”，要求用户输入密码的第 7、1、5、6 位进行登录。这样，键盘记录器无法通过一次记录获知敏感信息。但在多次记录之后，仍然有很大几率得到密码。文献[9]提出一种新的密码验证技术，其同样是利用键盘输入进行认证，这一认证方案中的密码不再是所输入的特定字符，而是输入字符所构成的图形以及这个图形中各按键的点击次数。若采用此类方式进行认证，键盘记录程序难以通过一次记录推断出密码，但是用户对密码的记忆与输入更为不便，其安全性也未能得到充分证明，难以得到广泛应用。

4 软键盘原理及安全性分析

软键盘(Soft Keyboard)又称虚拟键盘(Virtual Keyboard)、屏幕键盘(On Screen Keyboard)，是一种通过软件模拟键盘输入的技术。使用者通过鼠标点击屏幕的按钮输入信息。使用软键盘输入信息时，用户所输入的是鼠标信息，物理层和内核层的攻击即使截获鼠标信息也往往难以猜测其语义。与物理键盘输入相比，软键盘增强了安全性，得到了广泛应用。

4.1 软键盘的按键信息输入流程

软键盘可以是通用的，也可以是专用的。软键盘(如操作系统、输入法、杀毒软件等所附带的软键盘)在生成虚拟按键信息后，会将这些按键信息发送到系统消息队列，再由操作系统将按键信息分发给当前拥有输入焦点的窗口程序，以在任意编辑框中进行输入。专用软键盘(各种网上银行的软键盘)则会将按键信息直接发给特定的窗口程序，其只能在特定的密码框内进行输入。

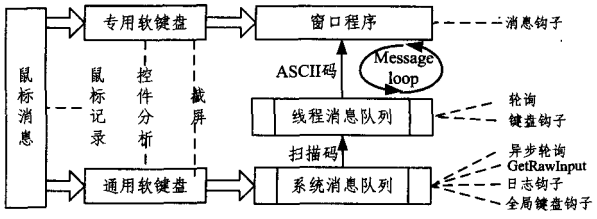

图 2 通用/专用软键盘输入流程及威胁

4.2 软键盘面临的威胁及相应防御措施

尽管绕开了物理和内核层键盘记录器的威胁，攻击者仍然有可能通过偷窥、消息截获、截屏、鼠标记录、控件分析等方法窃取到软键盘的输入内容。

4.2.1 偷窥(Shoulder Surfing)

现在的大多数密码框之所以设计为输入字符后显示为 * 号，就是为了防止偷窥。由于需要在屏幕上显示，使用软键盘被偷窥的可能性比使用物理键盘更大。这里的偷窥者，可以是人，也可以是摄像/录像设备。文献[19]设计了利用暗中录制的低质量录像来恢复出用户输入字符的方案，其具体方法是根据键盘布局和用户点击的大致位置来进行按键推断。由于文献[19]中算法所要求的图片质量不高，攻击者甚至可以通过屏幕在光滑物体(如眼镜)上的镜像来得知按键信息。这样即使在使用者的对面，攻击者也能窃取到输入信息。由文献[19]中算法可以看出，看到用户的点击位置要比看清用户点击的具体字符容易，键盘布局的动态变换可以在很大程度上提高输入的安全性。另外可采用光学方法来防止偷窥[4]。

4.2.2 消息截获

对于将按键信息发送到系统消息队列的通用软键盘，攻击者可以通过应用层键盘记录程序截获按键信息。对于由浏览器来解析执行的软键盘，攻击者有可能使用 IE 的 COM 接口编程来截获软键盘的输入[23]。对于此类攻击，通过加密或者加入干扰信息[15]等方式来进行保护可取得良好效果。

4.2.3 截屏(Screenshot)

截屏是一种抓取屏幕显示信息的技术。间谍程序可以在用户每次点击软键盘的时候，截取鼠标周围的一小块图像，利用截得的一系列图像可观察、得到用户的输入信息。主要防护措施有：

1)在软键盘运行期间，禁止截屏。

禁止的方法可以是阻断所有截屏功能的 API、进行显卡驱动过滤等。例如，卡巴斯基 2012 版的虚拟键盘采用挂钩 shadow ssdt 表的方式实现了防截屏功能，当虚拟键盘开启时截取的画面是全黑色的。

2)利用余晖效应。

数字图片使用动画的方式显示，每次显示的是数字的一部分，当动态显示时，人眼可以分辨出具体数字，但是截图就只能截取一部分[20]。

3)按键隐藏

在点击按键时，字符变为“ * ”。文献[21] 设计了一种可以免遭截屏攻击的安全软键盘：当鼠标移到某一按键上时，这

个键所处的整行按键显示为“*”;当鼠标点击某一按键时,全部按钮显示为“*”,持续 0.2 秒并在此期间变换布局。文献[22]采取了类似的措施:用户在记住自己准备点击的按键的位置后,点击“隐藏按键”按钮,隐藏按键值后再点击目标键。此类方法提高了软键盘的安全性,但降低了输入效率,带来了不便。另外,实验发现,大多数情况下用户不会在按键隐藏的短暂时间内进行鼠标移动,攻击者可以通过录像或者选择适当的截屏时机来获取输入。

4.2.4 鼠标记录

与键盘类似,鼠标信息的输入也要经过物理、内核、应用 3 个层次,经历发出电信号、产生中断、驱动程序处理、发往系统消息队列和线程消息队列等过程。防止这一漫长过程中的鼠标信息泄露是困难的。现有的通过根据点击位置推断软键盘按键的方法一般都需要得知软键盘窗口出现的时间和位置等信息,应当对这些信息进行保护,以提高软键盘输入信息的安全性。

另外,存在只根据鼠标记录推断出按键攻击的可能:假设攻击者知道软键盘的布局情况、按键大小、间距等信息,且记录了计算机用户在使用软键盘进行输入过程中的鼠标左键单击的坐标,则可基本推断出用户所点击的按键。因为所有的点击都必然位于软键盘内,可以据此推断软键盘的位置。得到软键盘位置后即可根据软键盘布局、点击坐标计算出用户所点击的按键。其中,每个可能的软键盘位置对应一系列可能的字符输入(键盘的位置存在多种可能,不能完全确定,但可能性的数目有限,一般来说,密码位数越多,其分布的范围就越大,可能的结果数也就越少)。对于此类威胁,只能从软键盘方面着手,通过在密码输入过程中改变软键盘窗口的位置、大小或按键的位置、大小来抵御这一威胁。其中改变按键的位置,即对软键盘内的按键排布进行随机化最为简单方便,现有多种应用中采取了这一措施,例如腾讯 QQ 软键盘、中国农业银行网上银行软键盘、中国建设银行网上银行软键盘等在启用时,都进行了一次软键盘布局随机化。文献[21]中软键盘则在每次点击键盘后都进行软键盘布局随机化。

4.2.5 控件分析

窗口又称视窗、GUI(Graphical User Interface,图形用户界面),主要用于获取用户输入、展示程序输出,是用户与程序的交互平台。窗口控件,又称元素,即主窗口的子窗口,是指附着于主窗口上的各个部件。常见控件种类有菜单栏、状态栏、标题栏、文件列表、下拉框等[52]。

不论虚拟键盘是作为一个独立的窗口出现(如 windows 自带 OSK)还是在浏览器窗口(如用于网上银行密码输入的软键盘)内出现,现有大多数应用中,虚拟键盘上的按键都是作为控件出现的。因此,可以通过获取控件的使用信息来推断出被点击的按键。现有研究和应用中的软键盘只有极少数考虑到了这方面的防御。

具体来说,可使用获取窗口属性的 Windows API 以及 MSAA(Microsoft Active Accessibility)两种方式获取信息。

(1)利用窗口相关的 Windows API 获取按键信息。以 OSK 为例,主窗口“屏幕键盘”上的每个按键为一个子窗口,按键内容即子窗口的标题可通过 GetWindowText()函数获得;每个按键拥有一个窗口 ID,可通过 GetWindowLong()函数获得。按键 A,D 的窗口信息如图 3 所示。

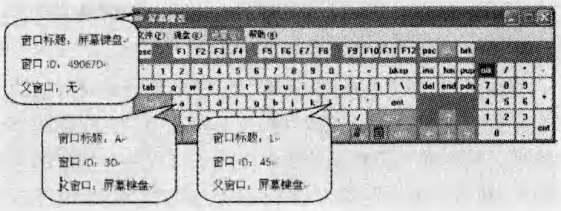

图 3 利用窗口相关的 Windows API 获取按键信息

(2)利用 MSAA 方式获取按键信息。微软的 MSAA 提供一组 API 和 COM 接口,实现了以程序方式访问 UI(User Interface)元素信息或操作 UI 元素的功能。

通过 MSAA 所提供的 AccessibleObjectFromWindow、AcessibleObjectFromPoint、WindowFromAcessible 等函数可以获取目标元素的 IAccessible 接口,进而获取元素的名称、状态、ID 等信息。

对于控件分析,现有研究和应用中对这一威胁还不够重视。各类软键盘可以通过以下途径避开此类威胁:浏览器上的软键盘可将图片作为按钮以防止记录程序直接记取字符,并隐藏按键的 ID 值或对其做随机化处理;设计实现通用软键盘时,可将软键盘面板定义为一个特定的数据结构,点击面板的不同位置(而非不同按钮)输出不同的虚拟键码;特定应用程序的软键盘应自行绘制而不是直接调用 Windows 的通用方式,例如 QQ 软键盘就是通过自定义的 TXGuiFoundation 实现,外部程序难以获得其具体信息。

5 键盘输入的安全性测试

5.1 物理键盘输入的安全性测试

利用各种键盘记录器尝试记录国内常用的网络服务的账号密码,其部分结果如表 2 所列 。

表 2 利用键盘记录器窃取网络服务的密码

	163 邮箱[1)]、花旗银行(中国)[2)]、中国建设银行[3)]、MSN、新浪 UC	QQ2012	支付宝[4)]	中国农业银行[5)]
键盘钩子	正确记录	为空	为空	为空
低级键盘钩子	正确记录	乱码	为空	为空
键盘驱动过滤	正确记录	正确记录	正确记录	为空
键盘端口扫描	正确记录	正确记录	正确记录	正确记录
硬件键盘记录器	正确记录	正确记录	正确记录	正确记录

其中,163 邮箱、中国建设银行等没有针对键盘记录采取

1) http://email.163.com/

2) https://www.citibank.com.cn/CNGCB/JSO/signon/DisplayUsernameSignon.do

3) https://ibsbjstar.ccb.com.cn/app/V5/CN/STY1/login.jsp

4) https://auth.alipay.com/login/index.htm 安全控件版本:3.2.1.0

5) https://easyabc.95599.cn/SelfBank/netBank/zh_CN/entrance/logonSelf.aspx 安全控件版本为 2.3.3.101

任何防护措施，可以采用任意一种键盘记录程序窃取其账号密码。QQ2012利用调试钩子和低级键盘钩子加入了干扰信息并进行了键码转换，增加了应用层键盘记录的难度；支付宝同样实现了应用层的键盘消息的保护，但可以通过内核层的任意一种键盘记录程序正确记取QQ和支付宝的密码。中国农业银行网上银行等无法通过作用于中断处理之后的键盘记录程序进行记录。硬件键盘记录器可以记录到键盘的所有按键信息。以上结果表明国内各常见应用对键盘记录的防范措施有限，难以保证用户敏感信息的安全性。

5.2 软键盘输入的安全性测试

下面以国内常见软件和应用的软键盘为例进行实验，尝试通过消息截获[6]、截屏、鼠标记录、控件分析去获取各虚拟键盘的点击信息。实验结果如表3所列，Y表示可以获取，N表示不可获取，P表示一定程度上可以获取。

表3 软键盘攻击测试实验结果

	OSK	卡巴斯基	QQ	农业银行	兴业银行	建设银行	深圳发展银行
消息截获	Y	Y	N	N	N	N	N
截屏	Y	N	Y	N	Y	Y	Y
鼠标记录	Y	Y	P	N	P	P	N
窗口相关API	Y	N	N	P	P	N	P
MSAA	Y	N	N	N	Y	Y	N

Windows自带软键盘，没用采用任何安全措施，采用以上任何方法均可正确记录。卡巴斯基软键盘采用挂钩shadow ssdt表的方式实现了防截屏功能。QQ软键盘进行了布局随机化，但其随机化方式为行内随机移位，因此其布局总数有限，只有 $13^2\times11\times10=1.86\times10^4$ 种，在穷举布局的基础上可以通过鼠标记录推断出按键。QQ软键盘由于并非采用通过方式实现，因此无法通过控件分析方式记录按键。

中国建设银行网上银行是最早使用软键盘的网上银行之一。建行软键盘在每次出现时会对数字按键进行布局随机化，增加了根据鼠标记录推断按键的难度，但如果密码中所含数字按键的比例较低，仍有很大可能推断出被点击的按键序列。与建设银行相反，兴业银行的布局随机化只随机化数字和字符按键，数字键位置不变，这样，如果密码中数字比例较高，正确推断密码的难度将会降低。中国建设银行、兴业银行、南京银行、花旗银行(中国)等均可通过截图方式记录按键，采用窗口相关API无法获得其控件信息，但采用MSAA方式可正确记录按键值。

中国农业银行网上银行软键盘具有很强的安全性，每次打开时，其键盘布局会进行完全随机化，无法根据鼠标点击位置推断按键；使用软键盘过程中无法进行截屏；采用窗口相关API和MSAA方法均无法获取控件文本，且获取到的大部分控件ID信息为随机值。但是，其各个按键窗口句柄和窗口ID存在规律。例如，3次打开软键盘；每次都依次点击软键盘上“1”、“2”、“3”、“9”、“0”、“[”、“]”、“—”、“=”“A”、“B”、“C”、“Y”、“Z”等按键，利用GetWindowLong函数记录所点击按键的ID值，结果如表4所列。可以看出虽然每次打开软键盘，各个按键所对应的ID值不是固定的，但各个按键的ID值遵循固有的规律。以第三次记录数据为例，中国农业银行软键盘上的68个按键{“·”,“1”,…,“Z”,“?”,“—”,“+”}分别对应{1DA,IDB,…,21A,21B,21C,21D}等68个窗口ID值。因此，类似于上一节中的情况，如果只有一个ID无法得知其对应的按键，则在拥有多个点击数据时可以推断出按键可能。例如，假设记录到被点按键序列对应的ID序列为{1EC,1EE,22F}，其长度为22F−1EC+1=68，可推断出输入序列为“·2+”；假设记录到的ID序列为{1E8,1E9,228}，其长度为228−1E8+1=65，则输入序列存在4种不同的可能“·1?”“12|”“23—”“234+”；最多的可能性是68种，6−8位密码平均约有15种可能——对于攻击者而言，这样的数量是完全可以接受的。

表4 中国农业各个按键所对应的窗口ID

	1	2	3	9	0	[	]	−	=	A	B	C	Y	Z
第一次	1	1	1	1	1	1	1	1	1	1	1	1	2	2
	E	E	E	F	F	F	F	F	F	F	F	F	1	1
	8	9	A	0	1	2	3	A	B	C	D	E	4	5
第二次	1	1	1	1	1	1	1	1	1	1	1	1	1	1
	9	9	9	A	A	A	A	A	B	B	B	B	C	C
	D	E	F	5	6	7	8	F	0	1	2	3	9	A
第三次	1	1	1	1	1	1	1	1	1	1	1	1	2	2
	D	D	D	E	E	E	E	E	E	E	E	F	0	0
	B	C	D	3	4	5	6	D	E	F	0	1	7	8

邮政储蓄和深圳发展银行的软键盘进行了分行布局随机化，因此难以通过鼠标记录推断按键，但是可以截屏，和中国农业银行软键盘一样，可以通过GetWindowLong函数获取的控件ID来推断按键。

6 基于行为的键盘记录器检测

常见的恶意代码检测方式分为基于签名/攻击行为模式的检测方法和利用正常软件的行为轮廓检测恶意代码两种。相应地，可以利用键盘记录器签名和的攻击行为进行检测，也可以利用键盘记录器有别于正常软件的行为特征。其中，基于签名/特征值的方法无法检测未知的或者经过变形的键盘记录器[54]，因此现在研究常通过行为分析来实现对键盘记录器的检测。

6.1 基于攻击行为的键盘记录器检测

可利用键盘记录器在记录按键信息、保存信息、上传信息等阶段的行为特征来检测非法的键盘记录程序。

1)对HOOK行为的检测：大多数键盘记录程序都采用hook内核或者应用层的函数来实现，可以根据这一特征进行检测。例如，可以采用输入表挂钩(IAT hook)的方法对调用SetWindowsHookEx的行为进行监控；文献[51]通过静态程序分析对函数指针进行精确的完成性验证，防止挂钩内核消息队列的攻击；文献[12]建立了一个针对二进制、关注上下文的检测模型，用来检测钩子类威胁。

2)对驱动过滤行为的检测：驱动过滤是在内核层进行键盘记录的最简单方法。对于此类攻击，可以通过监视键盘驱动栈来检测非法的键盘驱动[55]。

3)根据内存行为检测：文献[54]建立了细粒度的写内存模型，通过分析用户击键行为和程序写内存行为的关联性发现键盘记录程序。

6) 消息截获和截屏的方法非常多，此处仅通过应用层的各种消息钩子以及视频输入系统相关API进行尝试。

4)根据输入输出行为检测：文献[10]通过伪造特定的键盘输入，观察程序的输入(键盘输入字符)、输出(IO行为)来检测键盘记录器，实现了黑盒模式的检测，不需要了解键盘记录器的具体细节。

5)根据很多键盘记录程序都会周期性上传、发送所记录到的按键信息的特点，通过网络流量分析来检测键盘记录程序[13]。

6)根据实现键盘记录的整体流程检测：键盘记录器工作流程可分为键盘记录、文件访问、网络连接3个阶段，文献[11]通过这3者时间上的关联来进行检测。文献[40]提出了一种隐式流敏感的检测方法，即通过检测收集击键信息和上传击键行为的行为特征来检测键盘记录程序。

以上方法均有良好的创新性且对键盘记录器的检测具有一定效果，但同时也都存在缺陷：对于1)、2)，键盘记录程序未必是hook或者驱动过滤型程序；对于3)－6)，键盘记录器有可能设法绕过这些检测措施，例如可以通过改变发送按键信息间隔时间来绕过流量分析检测，通过改变输出按键信息的时机和方式绕过输出行为检测、整体流程检测。

6.2 基于正常行为轮廓的键盘记录器检测

键盘记录器的动作都是在暗中进行的，可以通过检测非用户触发的文件创建[56]或者网络连接行为[57,58]来发现这一威胁。文献[56-58]所提方法的目的并非针对键盘记录器，但鉴于隐蔽性是键盘记录器最主要的特点之一，从用户意愿与动作角度出发对其进行检测是可行的。

结束语 本文关注点为敏感信息通过键盘输入经操作系统传送到应用程序这一过程的安全问题，从攻防两个角度进行考虑，将现有攻击方法分为物理/内核/应用3个层次、截获/查询/旁路3种方式；对现有研究和应用中的防御方法进行了分析。对物理键盘而言，物理层的威胁几乎无法检测/防御，键盘记录软件也日渐多样化、底层化，现有防御措施难以兼顾安全性与通用性。近年来，软键盘在国内网上银行等应用中得到了广泛的应用，但是目前针对软键盘安全性的研究较少，现有应用中的“安全软件盘”也都存在一定的安全问题。本文对软键盘的安全性进行了较为全面、深入的分析并提出了新的攻防方法，例如只根据鼠标记录推断出按键、利用控件分析记录软键盘信息及抵御这一攻击的具体方法。此外，本文设计实现了对现有攻击防御的层次效果分析实验、对软键盘鼠标记录的攻击实验、控件分析攻击/防御实验，并对现有研究和应用中的键盘、软键盘保护措施进行了测试。实验结果证明，现在的安全措施还不足以应对键盘记录器的威胁，亟待进行更为深入的研究。

本研究只针对用户通过物理/虚拟键盘输入信息以及操作系统处理键盘信息的过程，没有全面分析键盘记录器的安装、键盘信息在软件处理和网络传输阶段的安全防护等问题，对一些具体细节、不太常用的方法没有详细介绍。

参 考 文 献

[1] Chahrvin S, Line S. Keyloggers－your security nightmare? [J]. Computer Fraud & Security, 2007(7): 10-11

[2] Holz T. Markus Engelberth Felix Freiling, Learning More About the Underground Economy: A Case-Study of Keyloggers and Dropzones[C]//Computer Security-ESORICS. 2009

[3] Chang H. The study on end-to-end security for ubiquitous commerce[J]. Supercomput, 2011, 55: 228-245

[4] Yamamoto H, Hayasaki Y, Nishida N. Securing information display by use of visual cryptography[J]. Optics Letters, September 1, 2003, 28(17): 1564-1566

[5] Hung C-W, Hsu F-H, Chen S-J, et al. QTE-based Solution to Keylogger Attacks[C]//SECURWARE 2012: The Sixth International Conference on Emerging Security Information, Systems and Technologies. 2012

[6] Balzarotti D, Cova M, Vigna G. ClearShot: Eavesdropping on Keyboard Input from Video[C]//IEEE Symposium on Security and Privacy. 2008

[7] 付永庆，郑莉，邵学辉. 一种监听键盘录入信息的新方法[J]. 哈尔滨工程大学学报，2008，29(2)

[8] Sagiroglu S. Keyloggers [J]. IEEE Technology and Society Magazine, 2009, 28(3): 10-17

[9] Daniel G. keyboard encryption[J]. IEEE, 2002, 21(3): 40-42

[10] Ortolani S, Giurida C, Crispo B. Bait Your Hook: A Novel Detection Technique for Keyloggers[C]//RAID 2010. LNCS 6307, 2010: 198-217

[11] Fu Jun, Liang Yi-wen, Tan Cheng-yu, et al. Detecting Software Keyloggers with Dendritic Cell Algorithm[C]//2010 International Conference on Communications and Mobile Computing. 2010

[12] Yin Heng, Poosankam P, Hanna S, et al. HookScout: Proactive Binary-Centric Hook Detection[J]. DIMVA 2010. LNCS 6201, 2010: 1-20

[13] Sreenivas R S, Anitha D R. Detecting keyloggers based on traffic analysis with periodic behavior[J]. Network Security, 2011, 2011(7): 14-19

[14] Yim K. A new noise mingling approach to protect the authentication password[C]//2010 International Conference on Complex, Intelligent and Software Intensive Systems. 2010

[15] Herley C, Florencio D. Microsoft Research, Redmond. How To Login From an Internet Cafe Without Worrying About Keyloggers[C]//Symposium on Usable Privacy and Security (SOUPS)'06

[16] Li Chun-xiao, Raghunathan A, Jha N K. A Secure User Interface for Web Applications Running Under an Untrusted Operating System[C]//2010 10th IEEE International Conference on Computer and Information Technology(CIT 2010)

[17] Mannan M, van Oorschot P C. Using a Personal Device to Strengthen Password Authentication from an Untrusted Computer[C]//Proceedings of the 11th International Conference on Financial Cryptography and Data Security. 2007: 88-103

[18] Mihajlov M, Jerman-Blaz˘ic B. On designing usable and secure recognition-based graphical authentication mechanisms[J]. Interacting with Computers, 2011, 23: 582-593

[19] Raguram R, White A M, Goswami D. Fabian Monrose and Jan-Michael Frahm iSpy: Automatic Reconstruction of Typed Input fromCompromising Reflections[C]//CCS'11. Chicago, Illinois, USA, October 2011

[20] Lim J. Defeat Spyware With Anti-Screen Capture Technology Using Visual Persistence[C]// Symposium On Usable Privacy and Security(SOUPS). 2007

[21] Gong Shang-fu, Lin Jun, Sun Yi-zhen. Design and Implementation of Anti-Screenshot Virtual Keyboard Applied in Online Banking[C]// E-Business and E-Government (ICEE). 2010: 1320-1322

[22] Agarwal M, Mehra M, Pawar R, et al. Secure Authentication using Dynamic Virtual Keyboard Layout[C]// International Conference and Workshop on Emerging Trends in Technology(ICWET 2011). TCET, Mumbai, India, 2011

[23] Mohanty D. Defeating Virtual Keyboard Protection[C]// Coffee and Security(C&S)2006

[24] Dorrendorf L, Gutterman Z. Cryptanalysis of the Random Number Generator of the Windows Operating System[J]. ACM Transactions on Information and System Security, Publication date, 2009, 13(1)

[25] Krishnamurthy, Hema. Method and system for a secure virtual keyboard[P]. United States Patent 8176324

[26] Heron S, Director M, Box N. The rise and rise of the keyloggers [J]. Network Security, 2007(6): 4-6

[27] 魏东,车文刚,段继磊,等. 利用日志钩子函数实现 windows 系统信息的捕获[J]. 昆明理工大学学报:理工版,2002(4)

[28] 高志新,应力. 访问键盘芯片获取 PS/2 键盘按键信息[J]. 信息安全与通信保密,2011(6)

[29] 王海晨,施勇,薛质. 基于 Windows 平台的安全密码框研究与实现[J]. 信息安全与通信保密,2011(4)

[30] Vishnani K, Pais A R, Mohandas R. An In-Depth Analysis of the Epitome of Online Stealth: Keyloggers; and Their Countermeasures[C]// ACC 2011, Part III. CCIS 192, 2011: 10-19

[31] MSDN. About Row Input[OL]. http://msdn. microsoft. com/en-us/library/windows/desktop/ms645543%28v=vs. 85%29. aspx

[32] Nasaka K, Takami T, Yamamoto T. A keystroke logger detection using keyboard-input-related API monitoring[C]// 2011 International Conference on Network-Based Information Systems

[33] 谭文,杨潇,邵坚磊. 寒江独钓——Windows 内核安全编程[M]. 北京:电子工业出版社,2009:57-97

[34] Lee K, Kim W, Bae K, et al. A Solution to Protecting USB Keyboard Data[C]// 2010 International Conference on Broadband, Wireless Computing, Communication and Applications

[35] 王成东,刘泰康,姜云. 无线键盘辐射信号的频谱测试与分析[J]. 计算机安全,2012(2)

[36] Knight E, Rhythm L. Skeleton keys: the purpose and applications of keyloggers Network Security[OL]. http://hdl. handle. net/2327/196879, October 2010

[37] QFX Software-Anti-Keylogging Software [OL]. http://www. qfxsoftware. com/

[38] Stefan D, Shu Xiao-kui, Yao Dan-feng. Robustness of keystroke-dynamics based biometrics against synthetic forgeries [J]. Computer& Security, 2012, 31

[39] http://zh. wikipedia. org/wiki/%E7%B4%B0%E8%83%9E%E8%87%AA%E5%8B%95%E6%A9%9F

[40] 李佳静,梁知音,韦韬,等. 一种隐式流敏感的木马间谍程序检测方法[J]. 软件学报,2010,21(6):1426-1437

[41] Yin H, Song D, Egele M, et al. Panorama: Capturing system-wide information flow for malware detection and analysis[C]// Proc. of the 14th ACM Conf. on Computer and Communications Security. New York: ACM Press, 2007: 116-127

[42] Marquardt P, Verma A, Carter H, et al. (sp)iPhone: Decoding Vibrations From Nearby Keyboards Using Mobile Phone Accelerometers[C]// CCS'11. Chicago, Illinois, USA, October 2011: 17-21

[43] LO'gorman L. Comparing passwords, tokens, and biometrics for user authentication [C]// Proceedings of the IEEE. 2003

[44] 马建坤,黄皓. 基于硬件辅助虚拟化技术的反键盘记录器模型[J]. 计算机科学,2011,38(11)

[45] 高艳,管晓宏,孙国基,等. 基于实时击键序列的主机入侵检测[J]. 计算机学报,2004,27(3)

[46] http://www. securityweek. com/study-reveals-75-percent-individuals-use-same-password-social-networking-and-email

[47] Lee K, Yim K, Security K. A Technological Review[C]// Fifth International Conference on Innovative Mobile and Internet Services in Ubiquitous Computing. 2011

[48] Lab K, Nikolay G. Keylogger: How they work and how to detect them[OL]. http://www. viruslist. com/en/analysis? pubid=204791931, 2007

[49] Zhuang Li, Zhou Feng, Tygar J D. Keyboard Acoustic Emanations Revisited[C]// CCS'05. Alexandria, Virginia, USA, November 2005

[50] Ortolani S, Crispo B. NoisyKey: Tolerating Keyloggers via Keystrokes Hiding[C]// USENIX2012

[51] Wei Jin-peng, Pu C. Towards a general defense against kernel Queue hooking attacks[J]. Computers & Security, 2011, 31(2): 176-191

[52] Shah G, Molina A, Blaze M. Keyboards and Covert Channels[C]// Security '06. 15th USENIX Security Symposium

[53] 李鹏伟,丁爽,傅建明. 面向异常检测的窗口识别[C]// 第九届中国信息与通信安全学术会议,CCICS. 2012

[54] Ortolani S, Giurida C, Crispo B. KLIMAX: Proefling Memory Write Patterns to Detect Keystroke-Harvesting Malware[C]// RAID 2011

[55] Majid F. Detecting keylogger virus by monitoring keyboard driver stack[J]. Al-Mansour Journal Issue, 2011(16)

[56] Xu Kui, Yao Dan-feng, Ma Qiang, et al. User-Behavior Based Detection of Infection Onset[R]. Technical Report TR-10-09. Computer Science, Virginia Tech

[57] Cui Wei-dong, Katz R H, Tan Wai-tian. Design and Implementation of an Extrusion-based Break-In Detector for Personal Computers[C]// The 21st Annual Computer Security Applications Conference(ACSAC). December 2005

[58] Lu Long, Yegneswaran V, Porras P, et al. BLADE: An Attack-Agnostic Approach for Preventing Drive-By Malware Infections [C]// CCS 2010