An NCC Group Publication

NCC Threat Brief

USB keyboards by post – use of embedded keystroke injectors to bypass autorun restrictions on modern desktop operating systems

Prepared by:
NCC Group's Technical Directors Forum

NCC Group's Technical Directors Forum represents the global senior technical leadership team from across NCC Group.

The Forum provides strategic direction for NCC Group's research, engineering and capability development and is the focal point for encouraging innovation across all of the Group's service offerings.

In addition to these internally focused activities, the Forum also provides analysis and comment on the technical issues facing our customers in today's fast moving cyber landscape.

Contents

1 Introduction

This threat brief discusses the existence of embedded USB keyboard like devices that are being used in an increasingly widespread manner. The purpose of these embedded USB keyboard like devices is to bypass the security enhancements in modern operating systems or configuration settings [11] that stop the automatic execution of code from USB devices.

Instead these embedded devices present themselves as keyboards and inject keystrokes as a user would in order to perform the marketer's desired operations.

2 USB keyboards by post

To support Windows 8 marketing activities Microsoft sent, including NCC Group, packages such such as these:

Contained inside of this cardboard advert is a flip out USB connection and the following instructions:

When plugged in to a Microsoft Windows or Apple OS X machine the following events occur:

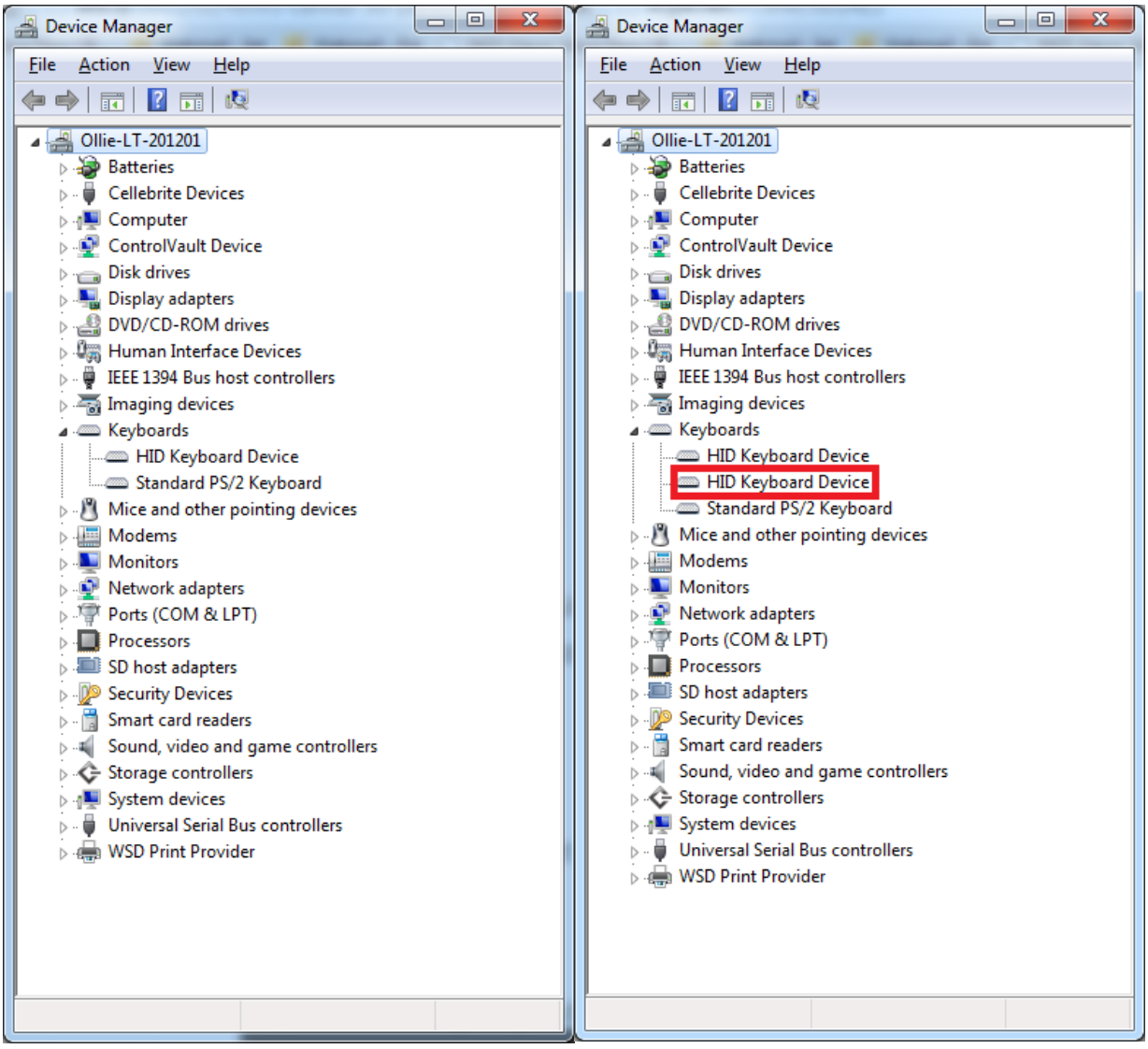
- **Windows:** Start Menu is loaded and the run dialogue opened
- **Apple Mac OS X:** Browser is brought to the foreground
- **Both:** URL is supplied to the browser

The host machine then visits the URL provided in the default browser. The obvious potential for misuse by a malicious actor masquerading as a legitimate marketing exercise or an unscrupulous marketer looking to extract sensitive information is high.

3 Technology

3.1 Proof of keyboard

NCC initially hypothesized that the technology in use was a USB device presenting itself to the operating system as a Human Interface Device (HID) and injecting keystrokes due to observing typing like behavior. This hypothesis was proven by looking at the Windows Device Manager before and after insertion.



Before insertion

After insertion

As can be seen in the image above a second keyboard is clearly visible on the system after the USB device is inserted.

3.2 Manufacturer

Extracting the USB device from the cardboard packaging revealed the manufacturer as Kyp [1] and the solution based upon their Webkey product [2].

The actual underlying technology is owned by a separate company called Vizkey [8]. Vizkey advertises that their technology will:

"*work on virtually all computers with a USB port including PCs, Macs, Netbooks, Laptops and PC/ Android tablets.*"

This platform agnostic way of instantiating a browser and visiting a URL of choice creates a credible vector of compromise on most operating systems.

3.3 COTS technology widely available

While this particular product is interesting in its application and while protected by patents there are numerous COTS products with the same capability. These products include USB Rubber Ducky [3], Glitch [4] (when combined with HIDIOUS or HID Injection Over USB Suite [5]) and Teensy [6].

4 Scale of usage

While this particular example is related to Windows 8, Microsoft is not the first to leverage such technology. Hyundai were noted in 2010 by security research Seth Fogy [6] as using the same solution. There are also examples of numerous other well-known companies discussed in case studies by Kyp [9] and Vizkey [10] as utilizing the same technology.

5 Detection and mitigation

Using USB devices view from NirSoft [12] we can see that the use of the device is logged in the device history. However we can also see that the device doesn't have a serial number but does have several other enumerable properties.

Serial Number	Created Date	Last Plug/Un...	VendorID	ProductID	Firmware Revis...	USB Class	USB SubCl...	USB Protoc...
	29/10/2012 23:56:03	30/10/2012 09:19:06	05ac	020b	3.01	03	01	01

Interestingly the vendor ID used by the device is `0x05ac` which according to the USB ID repository is Apple Inc [13] while the product ID is a Pro Keyboard. This misrepresentation would complicate both detection and blocking in a larger organisation. The reason that Apple is used for the vendor (VID) and product (PID) is to bypass Apple's keyboard wizard on Mac OSX which would require the user to supply random keys in order to allow the keyboard to fucntion.

Where Apple keyboards of this particular model are not used then endpoint protection software such as Lumension Device Control [14] can be used to mitigate the risk posed. Other approaches to mitigate the threat include using inbuilt features in modern versions of Windows [16] or on Linux via UDEV rules.

6 Conclusion

The legitimate use of key stroke injectors by marketers only goes further to confuse end users about the risks posed by USB devices and organizations ability to manage risk. Further, the misrepresentation of the device with regards to vendor and product ID only complicates both detection and mitigation within organizations.

Organizations should look to educate users as to the risks of inserting USB devices received through the post while encouraging them to inform security departments upon receipt.

7 References & further reading

The following references were used when producing this threat brief.

1. Kyp
   http://www.kyp.com/
2. Kyp Webkey
   http://www.kyp.com/Our-Solutions/Case-Studies/Webkey.aspx?lang=en-GB
3. USB Rubber Ducky
   http://hakshop.myshopify.com/products/usb-rubber-ducky
4. Glitch
   http://www.kickstarter.com/projects/1186217328/the-glitch
5. HIDIOUS: HID Injection Over Usb Suite
   http://www.hackfromacave.com/projects/hidious.html
6. Teensy
   http://www.offensive-security.com/metasploit-unleashed/Teensy_USB_HID_Attack
7. Hyundai
   http://www.ciscopress.com/articles/article.asp?p=1636214
8. Vizkey
   http://www.visiblecomputing.com/vizkey.html
9. Kyp Case Studies
   http://www.kyp.com/Our-work.aspx?tag=Webkey
10. Vizkey Case Studies
    http://www.visiblecomputing.com/casestudies_arvato.html
11. How to disable the Autorun functionality in Window
    http://support.microsoft.com/kb/967715
12. NirSoft USB Devices View
    http://www.nirsoft.net/utils/usb_devices_view.html
13. The USB ID Repository
    http://usb-ids.gowdy.us/read/UD/05ac
14. Lumension Device Control
    http://www.lumension.com/device-control-software/usb-security-protection/features-and-benefits.aspx
15. Microsoft - Step-By-Step Guide to Controlling Device Installation Using Group Policy
    http://msdn.microsoft.com/en-us/library/bb530324.aspx
16. Irongeek Guide to Locking Down Windows 7 Against Attacks Using GPO
    http://www.irongeek.com/i.php?page=security/locking-down-windows-vista-and-windows-7-against-malicious-usb-devices

8 Thanks

The NCC Technical Directors Forums wishes to thank Andrew Davies of NCC Group for his contributions to this paper.