MALICIOUS USB DEVICES:
IS THAT AN ATTACK VECTOR IN YOUR POCKET OR ARE YOU JUST HAPPY TO SEE ME?

Adrian Crenshaw
Special Thanks

- Tenacity Solutions
  http://www.tenacitiesolutions.net/

- PJRC
  http://www.pjrc.com/

- ISSA Kentuckiana Chapter
  http://www.issa-kentuckiana.org/
About Adrian

- I run Irongeek.com
- I have an interest in InfoSec education
- I don’t know everything - I’m just a geek with time on my hands
- (ir)Regular on the ISDPodcast
  http://www.isd-podcast.com/

http://Irongeek.com
Maltronics: Malicious Hardware

- Hardware instead of software
- (though software is of course involved in payloads and firmware)
- A few possible non USB examples:
  - PS/2 Keyloggers
  - Backdoored routers
  - Trigger self destructing hardware
  - Weakened crypto chips
A little background on USB

- Hardware has identifying strings to help the OS figure out what drivers to load
- USB devices have
  - Vendor IDs
  - Product IDs
- Great list here: http://www.linux-usb.org/usb.ids
- And Sometimes:
  - Serial Numbers
Malicious USB devices

So, why am I concentrating on USB?
- More real examples, as oppose to scare mongering
- Ease of installation
- Lack of admin rights needed in many cases
- Easy to hide or social engineer someone else into installing them

http://Irongeek.com
Proposed categories for malicious USB devices

- USB Mass Storage containing malware
- U3 Thumb drives with "evil" autorun payloads
- Hardware key loggers
- Programmable HID USB Keyboard Dongle Devices
The USB mass-storage device class is a standard way to present digital storage to a host operating system.

Many devices utilize the USB mass-storage device class standard: flash drives, card readers, digital cameras, MP3/media players, digital picture frames and a plethora of other items.

Not necessarily intentional, could be a quality control issue.
Trigger vectors

- Autorun
- Intentionally ran by user (drivers or installs)
- Exploit a software bug, for example:
  - WMF (Windows Metafile) vulnerability
  - LNK (Link) file vulnerability
Real world examples

- Mariposa botnet client on the Vodafone
- Malware shipped on Apple Video iPods
- Digital Photo Frames and Other Gadgets Infected
Detection and Mitigation

- User awareness (best of luck with that!)
- Disable Autorun
- Keep patches up to date
- Run an anti-malware/anti-virus system
- Lockdown what hardware can be installed
U3 Thumb drives with "evil" autorun payloads

- U3 was meant to make running applications off of a flash drive easier
- U3 drives make part of the device look like a CD device so OS will try to use Autorun
- Attackers can use this functionality to automate attacks like hash dumping, or installing Remote Access Trojans
- Core difference between this and “USB Mass Storage containing malware” is intent

http://lrongeek.com
Trigger vectors

- Autorun
- If Autorun is disabled, attacker could just run it directly
Real world examples

- Hak5 Switchblade
  http://www.hak5.org/w/index.php/USB_Switchblade

- Butturini’s Incident Response Payload
  http://www.hak5.org/w/index.php/U3_Incident_Response_Switchblade

- Steve Stasiukonis of Secure Network Technologies Inc pen-test story
  http://www.darkreading.com/security/perimeter/showArticle.jhtml?articleID=208803634
Detection and Mitigation

- User awareness (best of luck with that!)
- Disable Autorun
- Keep patches up to date
- Run an anti-malware/anti-virus system
- Lockdown what hardware can be installed
Why will I focus on the last two categories?

- Anti-malware is a fairly well covered topic, and these storage based classes of USB devices are just two more vectors.
- Anti-malware packages from major vendors are already focusing on these issues.
- The autorun issues have been covered a lot.
- The next two categories don’t have to rely on malware or autorun.
Hardware key loggers

- Hardware key loggers are fairly simple devices conceptually.

- Essentially they are installed between the keyboard and the computer, and then log all of the keystrokes that they intercept to their onboard flash memory.

- A snooper can then come along later to pick up the key logger and extract the captured data (passwords, documents, activity, etc.).
None really, though to dump the logs certain keystrokes may have to be sent
- KeyCarbon: phxlog
- KeeLog: Hold k+b+s

After the recovery trigger, some type the contents of the log and some become drives where you can copy off the log
Real world examples

- People don’t seem to want to tell the world they got owned
- The practicality of USB hardware key loggers from the attacker’s perspective varies for many reasons:
  1. The cost can be high, from about $60 to well over $300. How many attackers would wish to leave behind such an expensive piece of equipment that they may not be able to collect later?
  2. Physical access to the system is required for a short time to install, and possibly to retrieve. So far the wireless key logger the author has tested has been less than reliable.
  3. USB is a tougher spec to implement than PS/2. Some keyboard/PC combinations simply fail to function, or function with odd side effects.

http://lrongeek.com
Detection and Mitigation

- Physical security
- Lockdown what hardware can be installed may work in some cases but not many
- Physical inspection
- Notice odd problems that could mean there is a USB keylogger present
  - Odd USB vendor/product IDs?
  - Inline devices not working from a keyboards built-in hub?
  - Reports of slow USB speed with inline devices?
More details that you ever wanted to know about USB hardware keyloggers

Programmable HID USB Keyboard Dongle Devices

- Simple microcontroller based device that acts as a USB HID (Human Interface Device)
- Can be used to script any actions a keyboard and mouse can do
- Way more information can be found here: http://www.irongeek.com/i.php?page=security/programmable-hid-usb-keystroke-dongle
Why would an attacker (or you) want a programmable keystroke device?

- Likely types faster than you can, without errors
- Works even if U3 autorun is turned off
- Draws less attention than sitting down in front of the terminal would. The person turns their head for a minute, the pen-tester plugs in their programmable USB key stroke dongle, and Bobs your uncle, instant pwnage.
- Can also be set to go off on a timer when you know a target will be logged in
- Can be Hidden in other items
- Just use your imagination!

http://Irongeek.com
Payloads

1. Add a user to the box or the domain.
2. Run a program that sets up a back door.
3. Copy files to a thumb drive, or upload them to a site that the attacker controls.
4. Go to a website that the victim has a session cookie for, and do some sort of transaction (sort of like Cross Site Request Forgery, but hardware based).
Trigger vector

- Timer based
- Lock Keys
- Light, motion or some other environmental conditions
Samples and Demos

http://Irongeek.com
Real world examples

- Not much yet, just a lot of talk at conferences:
  - Dave Kennedy “SET 0.6 Release with Special PHUKD Key” talk at B-Sides Las Vegas 2010.
  - Dave Kennedy and Josh Kelly “Powershell...omfg” at Defcon 18 and Blackhat Las Vegas 2010 [link]
  - Richard Rushing “USB - HID, The Hacking Interface Design” at BlackHat USA 2010
  - Monta Elkins s “Hacking with Hardware: Introducing the Universal RF USB Keyboard Emulation Device – URFUKED”.
  - USB Rubber Ducky [link]
  - Adrian Crenshaw talk “Programmable HID USB Keystroke Dongle: Using the Teensy as a Pen Testing Device” [link]

[link]
Detection and Mitigation

- User awareness, just don’t plug everything in!
- Physical security
- Lockdown what hardware can be installed
- Physical inspection
- Anomalies
USB LOCKDOWN

There is some hope
USB Lockdown: Windows

On Windows 7/Vista look at the following GPO options:
Computer Configuration->Administrative Templates->System->Device Instillation->Device Instillation Restrictions
USB Lockdown: Windows

[HKEY_CURRENT_USER\Software\Microsoft\Windows\CurrentVersion\Group Policy Objects\{EA879B20-EDB8-4FBB-972D-DDD85F5D90AA}Machine\Software\Policies\Microsoft\Windows\DeviceInstall\Restrictions]
"DenyRemovableDevices"=dword:00000001

[HKEY_CURRENT_USER\Software\Microsoft\Windows\CurrentVersion\Group Policy Objects\{EA879B20-EDB8-4FBB-972D-DDD85F5D90AA}Machine\Software\Policies\Microsoft\Windows\DeviceInstall\Restrictions\DeniedPolicy]
"SimpleText"="Disabled because Adrian Said So!!!

If device was inserted when policy is in place, you may have to go into device manager to enable the device even after the policy is unset.
USB Lockdown: Windows
Adrian's recommendation for preventing PHUKD devices from functioning

1. Enable both "Allow administrators to override Device Installation Restriction policies" and "Prevent installation of removable devices".

2. Set "Display a custom message title when device installation is prevented by a policy setting" and "Display a custom message when installation is prevented by a policy setting" to something meaningful so the user knows why the hardware did not install properly.

3. Whenever you install a new device on purpose, manually go into device manager, and install the drivers using the "Update Driver Software..." option.

More information at:

Figured I could do it with UDEV rules, scratched my head for awhile

Michael Miller from the PaulDotCom podcast mailing list helped point me to some info that helped:
http://www.mjmwired.net/kernel/Documentation/usb/authorization.txt

Some commands to find info about your devices so you can write a script:

- Simple list of USB devices
  lsusb

- See a list of input devices:
  cat /proc/bus/input/devices

- Get info about a device:
  udevadm info -a -p /sys/bus/usb/devices/usb1/1-3

- Test your scripts:
  udevadm test /sys/bus/usb/devices/usb1/1-3

- See info about devices as they a inserted and removed:
  udevadm monitor or tail /var/log/messages
Crappy UDEV Example

- /etc/udev/rules.d/01-usbblockdown.rules
- Still needs work, use at your own risk

#Script by Adrian Crenshaw
#With info from Michael Miller, Inaky Perez-Gonzalez and VMWare

#By default, disable it.
#ACTION=="add", SUBSYSTEMS=="usb", RUN="/bin/sh -c 'echo 0 >/sys$DEVPATH/authorized'"
ACTION=="add", SUBSYSTEMS=="usb", RUN="/bin/sh -c 'for host in /sys/bus/usb/devices/usb*; do echo 0 > $host/authorized_default; done'"

#Enable hub devices. There may be a better way than this.
ACTION=="add", ATTR{bDeviceClass}=="09", RUN="/bin/sh -c 'echo 1 >/sys$DEVPATH/authorized'"

#Other things to enable
ACTION=="add", ATTR{idVendor}=="046d", ATTR{idProduct}=="0809", RUN="/bin/sh -c 'echo 1 >/sys$DEVPATH/authorized'"
ACTION=="add", ATTR{serial}=="078606B90DD3", RUN="/bin/sh -c 'echo 1 >/sys$DEVPATH/authorized'"
ACTION=="add", ATTR{product}=="802.11 n WLAN", RUN="/bin/sh -c 'echo 1 >/sys$DEVPATH/authorized'"
ACTION=="add", ATTR{idVendor}=="413c", ATTR{idProduct}=="2106", RUN="/bin/sh -c 'echo 1 >/sys$DEVPATH/authorized'"

- Read up on UDEV rules and sysfs:
  http://www.reactivated.net/writing_udev_rules.html
  http://en.wikipedia.org/wiki/Sysfs
  http://lrongeek.com
FINDING THE NEEDED INFORMATION

USB Big Brother Time
Device Manager

- Built into Windows, but a pain to use for finding information about USB devices
USBDeview

- Lists Vendor/Product IDs and tons of information about devices!
- USBDeview can remove devices, list them remotely and tons more. See help file for all the options.
USBDeview: Remote collection

- Create a file that looks something like:
  \192.168.1.13
  \192.168.1.14
  \skynet
  \cthulhu

- Run command:
  USBDeview /remotefile boxes.txt /scomma myoutput.csv

- Import the CSV file into and Database or Spreadsheet application to search and correlate.

- More details at:
Things to do with the data

- Track “Typhoid Mary” thumb drives by their serial number. (Conficker)
- Look for known suspicious Vendor IDs.
- Computers with two keyboards?
Events

- Louisville Infosec

- DerbyCon 2011, Louisville Ky

- Skydogcon/Hack3rcon/Phreaknic/Notacon/Outerz0ne
  [http://phreaknic.info](http://phreaknic.info)
  [http://www.outerz0ne.org/](http://www.outerz0ne.org/)

http://lrongeek.com
QUESTIONS?

42