Defending Against Malicious USB Firmware with GoodUSB

Dave Tian, Adam Bates, Kevin Butler

University of Florida

ACSAC'15, L.A., CA, USA
Dec 10th, 2015
When USB goes bad...
When USB goes bad...
When USB goes bad...
When USB goes bad…

BadUSB - On Accessories that Turn Evil by Karsten Nohl + Jakob Lell

USB Rubber Ducky
The most lethal duck ever to grace an unsuspecting USB port

SALE PRICE
$42.99

Stop Scan Enter
BadUSB Attack

• Malicious device controls all responses during USB Enumeration.
• Requests additional interfaces/drivers to be loaded on its behalf.
• Device can even lie about its own identity.

These actions are invisible to the user
USB Interfaces represent a set of unrestricted host permissions.

- **OS knows nothing about the device**
  - but loads drivers to make the device happy anyway!

- **User knows something about the device**...
  - E.g., from the appearance of the device
  - but no one is asking them about it!
GoodUSB

• **Encodes user’s expectations to authorize USB activity**
  • *Let the user determine what the device should do*

• **Tracks device’s claimed identity**
  • *Let the OS recognize if the device appears to have been plugged in before*

• **Profiles suspicious devices in virtualized honeypot**
  • *Let the user see what the device does*
Design Challenges

1. How can we mediate USB Enumerations?

2. How can we verify the identity of USB devices?

3. If a device is suspicious, what can we do about it?
USB Mediator

- GoodUSB sits between the user and the device
  - Enforcing policies
  - Redirecting devices to honeypot
• **GoodUSB is implemented in the Linux kernel**
  
• Identifying the device firmware

• Enforcing policies at the driver level
Linux Kernel Enhancement

• Device Class Identifier
  • SHA1 (USB descriptors)

• Kernel Hub Thread Instrumentation
  • Suspend the driver binding

• Netlink Socket
  • Communicate with the user-space
<table>
<thead>
<tr>
<th>Device</th>
<th>Storage</th>
<th>Audio</th>
<th>HID</th>
<th>Storage</th>
</tr>
</thead>
<tbody>
<tr>
<td>Flash Drive</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Headset</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Smartphone</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Charger</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>
Identifying USB Devices

Product: Logitech USB Headset
Manufacturer: Logitech
Configuration Num: 1
Interface Total Num: 4
Please choose the desired device functionality:

- USB Storage (thumb drive, portable disk, SD reader)
- USB Keyboard
- USB Mouse
- USB Joystick
- USB Wireless
- USB Cellphone (iPhone, Nexus, Galaxy)
- USB Tablet (iPad, Nexus, Tab)
- USB Microphone
- USB Sound (sound card, speaker, headph
- USB Hub (USB port extension)
- USB Video (WebCam)
- **USB Headset**
- USB Charger (E-cig, portable battery, toy)
- USB Communication (USB-USB networking, ATM/Ethernet)
- USB Printer
- USB Scanner
- USB UNKNOWN

I already registered this device! | Reigister device

GoodUSB: Select a Security Picture

- Dinosaur 1
- Dinosaur 2
- Dinosaur 3
- Dinosaur 4

Suspend Registration | Complete Registration
Profiling USB Devices

- **QEMU KVM**
  - *USB device pass-thru vs. USB host controller pass-thru*

- **USB Monitor**
  - *A udev rule to start USB device profiling*

- **USB Profiler**
  - *Generate a comprehensive USB device report for inspection*
  - *usbmon, lsusb, usbhid-dump, usb-devices, tcpdump*
usbpro HID analyzer started:
=====================================================================
_F2__x_t_term.ENTER
_p_w_d.ENTER
_i_d.ENTER
_c_a_t_SPACE/etc/passwd.ENTER
______________________________
usbpro HID analyzer done
Evaluation

• USB Headset
• USB Rubber Ducky
• Teensy 3.1
• Smartphones
### Overhead in Microseconds

<table>
<thead>
<tr>
<th>Category</th>
<th>Overhead</th>
</tr>
</thead>
<tbody>
<tr>
<td>1st Enumeration</td>
<td>7 (5.2%)</td>
</tr>
<tr>
<td>Subsequent Enumerations</td>
<td>7 (5.0%)</td>
</tr>
<tr>
<td>Honeypot Redirection</td>
<td>262.1 (N/A)</td>
</tr>
</tbody>
</table>
Discussion

• Drivers
  • Enforcing *Least Privilege*
  • Vendor-specific
• Compatibility with specialized USB devices
• Usability
Conclusion

- BadUSB Attacks: unconstrained privileges
- GoodUSB: enforcing the permission model by
  - encoding users’ expectation into driver loading
  - tracking devices’ claimed identities
  - profiling suspicious devices in the honeypot

Take-away:
Do NOT plug in an unknown USB thumb drive without GoodUSB enabled!
Get GoodUSB:
https://github.com/daveti/GoodUSB

All bugs are introduced by:
daveti@ufl.edu

THANKS!
Acknowledgements

• This work is supported in part by the US National Science Foundation under grant numbers CNS-1540217 and CNS-1540218, as well as by the Florida Cyber Consortium.

• We also thank reviewers of ACSAC and colleagues at FICS for their invaluable comments.
GoodUSB: architecture

GoodUSB Daemon (gud)
- Policy Engine
- Graphical Interface
- Device Database

USB Honeypot (HoneyUSB)
- USB Profiler
- USB Monitor
- QEMU KVM

Kernel Hub Thread

Kernel Virtual Machine
- Host Ctrl Passthrough

Device Class Identifier
- Host Ctrl 0
- Host Ctrl 1
- Host Ctrl 2

USB Interface Drivers
- Limited HID

User Space

Kernel Space

VirtIO

Netlink

USB Interface

Limited HID
Don’t judge a book by its cover
Penetration Tools

USB Rubber Ducky
The most lethal duck ever to grace an unsuspecting USB port

- **Write**
  - payloads with a simple scripting language or online payload generator including:
    - WiFi AP with disabled firewall
    - Reverse Shell binary injection
    - Powershell wget & execute
    - Retrieve SAM and SYSTEM
    - Create Wireless Association

- **Encode**
  - the Ducky Script using the cross-platform open-source duck encoder, or download a pre-encoded binary from the online payload generator.
  - Carry multiple payloads, each on its own micro SD card.

- **Load**
  - the micro SD card into the ducky then place inside the generic USB drive enclosure for covert deployment.

- **Deploy**
  - the ducky on any target Windows, Mac and Linux machine and watch as your payload executes in mere seconds.
HID Attacks with Ducky