When Firmware Modifications ATTACK!
EMBEDDED
Embedded

Defense
EMBEDDED

DEFENSE

EXPLOITATION
In ye 'olden days...
Big Bad Internet
Big Bad Internet

Hax0r

Firewall

Server
Big Bad Internet

Hacker

Firewall

Server
Big Bad Internet

HaxOR

FIREWALL

SERVER

Printer
Big Bad Internet

Hax0R

FIREWALL

SERVER

Printer

2/25/13  Cui Costello Stolfo - NDSS 2013
Big Bad Internet

Hax0R

FIRE WALL

Phone

SERVER

Printer
In ZKNOW...
Let’s Talk
HP Koan: How does printer update firmware?...
HP Koan: How does printer update firmware?... Print!
HP Koan: How does printer update firmware?... Print!

Remote firmware update using the LPR command


Complete the following steps to update the firmware by using the LPR command.

1. Type lpr -P -S -o I -OR- lpr -S -Pbinps, where can be either the TCP/IP address or the hostname of the product, and where is the filename of the .RFU file from a command window.

   NOTE: The parameter (-o I) consists of a lowercase "O", not a zero, and a lowercase "L", not a numeral 1. This parameter sets the transport protocol to binary mode.

2. Press Enter on the keyboard. The messages described in the section "Printer messages during the firmware update" appear on the control panel.

   NOTE: The product automatically restarts the firmware to activate the update. At the end of the update process, the Ready message appears on the control panel.

3. Type exit at the command prompt to close the command window.

You see where this is going...
## HP RFU (Remote Firmware Update) File

| 000000 | 40 50 4A 4C 20 43 4F 4D 4D 45 4E 54 20 4D 4F 44 45 4C 3D 48 |
| 000014 | 50 20 4C 61 73 65 72 4A 65 74 20 50 32 30 35 35 64 6E 0A 40 |
| 000028 | 50 4A 4C 20 43 4F 4D 4D 45 4E 54 20 56 45 52 53 49 4F 4E 3D |
| 00003C | 38 33 35 30 34 0A 4B 4C 20 43 4F 4D 4D 45 4E 54 20 44 |
| 000050 | 41 54 45 43 4F 44 45 3D 32 30 31 30 33 33 30 38 0A 4D 50 4A |
| 000064 | 4C 20 55 50 47 52 41 44 45 20 53 49 5A 45 3D 37 39 32 39 39 |
| 000078 | 30 36 00 00 00 00 00 00 00 00 00 2D 00 00 00 55 41 54 00 00 01 |
| 00008C | 20 00 67 67 09 00 00 02 00 00 00 00 00 00 67 FD 00 00 20 |
| 0000B4 | 08 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 |
| 0000DC | 6F 00 00 04 4D 3C 00 68 1D E9 00 00 21 86 00 00 50 91 00 68 3F |
| 0000F0 | 0C 00 68 80 53 00 00 04 00 00 00 00 00 00 00 00 00 00 00 00 |
| 000104 | 83 00 00 4D BF 00 68 C1 A1 00 00 20 23 00 00 4B 2A 00 68 E1 |
| 000118 | 5A 00 69 22 24 00 00 04 00 00 00 00 00 00 00 00 00 00 00 00 |
| 00012C | 5A 00 69 22 00 00 04 00 00 00 00 00 00 00 00 00 00 00 00 00 |
| 000140 | 42 00 00 00 50 24 00 69 64 88 00 00 24 00 00 00 00 00 00 00 00 |
| 000154 | 95 00 00 24 35 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 |
| 000168 | 0E 00 69 D0 4E 00 00 00 00 28 24 00 00 7A 0E 00 69 F8 72 00 00 22 |
| 00017C | 0D 00 00 50 D6 00 6A 1B 3F 00 00 21 3E 00 00 52 00 00 6A 3C |
| 000190 | 0D 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 |

- **@PJL COMMENT MODEL=** P LaserJet P2055dn
- **@PJL COMMENT VERSION=** 835049
- **@PJL COMMENT D ATECODE=** 20100308
- **@PJL COMMENT L UPGRADE SIZE=** 972999
- **@PJL COMMENT L ENTR Y=** 0648-12345
- **@PJL COMMENT E R LANGUAGE=** ACP:RPC
- **M<h>** ! | P h?
- **o** ( **M h** | **P**
- **PS** | **L h**
- **M h** | **K** | **M**
- **Z;i$$N$$iCF:**
- **B** | **P$;id$$T$$i**
- **$5$$Ti$$i$**
- **N** ($$$z$$i$$r$$
- **P j?$$i$$R$$j<
- **K$$j$$p$$Q**
Stating the Obvious

- LPR / Raw printing has **NO AUTHENTICATION MECHANISM**
Stating the Obvious

- LPR / Raw printing has no authentication mechanism
- PJL can be embedded in PostScript (and lots else)
Stating the Obvious

- LPR / Raw printing has no authentication mechanism
- PJL can be embedded in postscript (and lots else)

- Malicious RFU + Doc Format Attack Vector
  
  Self-propagating printer malware
  Embedded advanced persistent asset
  Embedded spear-phishing, etc
The Plan

- **Reverse RFU Format**
- **Construct Printer Rootkit**
- **Repack Malicious RFU**
- **Embed in Document**
Reverse RFU Format

What didn’t work…

• Stare at binary blob
• Common FS headers
• Googling
Reverse RFU Format

What **did** work…

- Reversing the bootloader
Reverse RFU Format

What did work…

- Reversing the bootloader
- Monkey soldering
- Arduino
- Duct-tape
Reverse RFU Format

Main SoC Boots from SPI flash chip

Main SoC = Mystery ARM
No datasheet

Spansion Flash
Have datasheet!

2055DN Formatter Board
<table>
<thead>
<tr>
<th>Operation</th>
<th>Command</th>
<th>One Byte Command Code</th>
<th>Description</th>
<th>Address Bytes</th>
<th>Mode Bit Cycle</th>
<th>Dummy Bytes</th>
<th>Data Bytes</th>
</tr>
</thead>
<tbody>
<tr>
<td>Read</td>
<td>READ</td>
<td>(03h) 0000 0011</td>
<td>Read Data bytes</td>
<td>3</td>
<td>0</td>
<td>0</td>
<td>1 to $\infty$</td>
</tr>
<tr>
<td></td>
<td>FAST_READ</td>
<td>(0Bh) 0000 1011</td>
<td>Read Data bytes at Fast Speed</td>
<td>3</td>
<td>0</td>
<td>1</td>
<td>1 to $\infty$</td>
</tr>
<tr>
<td></td>
<td>DOR</td>
<td>(3Bh) 0011 0011</td>
<td>Dual Output Read</td>
<td>3</td>
<td>0</td>
<td>1</td>
<td>1 to $\infty$</td>
</tr>
<tr>
<td></td>
<td>QOR</td>
<td>(6Bh) 0110 0111</td>
<td>Quad Output Read</td>
<td>3</td>
<td>0</td>
<td>2</td>
<td>1 to $\infty$</td>
</tr>
<tr>
<td></td>
<td>DIOR</td>
<td>(BBh) 1011 0111</td>
<td>Dual I/O High Performance Read</td>
<td>3</td>
<td>1</td>
<td>0</td>
<td>1 to $\infty$</td>
</tr>
<tr>
<td></td>
<td>QIOR</td>
<td>(EBh) 1110 0111</td>
<td>Quad I/O High Performance Read</td>
<td>3</td>
<td>1</td>
<td>2</td>
<td>1 to $\infty$</td>
</tr>
<tr>
<td></td>
<td>RDID</td>
<td>(9Fh) 1001 1111</td>
<td>Read Identification</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>1 to 81</td>
</tr>
<tr>
<td></td>
<td>READ_ID</td>
<td>(90h) 1001 0000</td>
<td>Read Manufacturer and Device Identification</td>
<td>3</td>
<td>0</td>
<td>1</td>
<td>1 to $\infty$</td>
</tr>
<tr>
<td>Write Control</td>
<td>WREN</td>
<td>(06h) 0000 0110</td>
<td>Write Enable</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td></td>
<td>WRDI</td>
<td>(04h) 0000 0100</td>
<td>Write Disable</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>Erase</td>
<td>P4E</td>
<td>(20h) 0010 0000</td>
<td>4 KB Parameter Sector Erase</td>
<td>3</td>
<td>0</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td></td>
<td>P8E</td>
<td>(40h) 0100 0000</td>
<td>8 KB (two 4 KB) Parameter Sector Erase</td>
<td>3</td>
<td>0</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td></td>
<td>SE</td>
<td>(D8h) 1101 1000</td>
<td>64 KB Sector Erase</td>
<td>3</td>
<td>0</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td></td>
<td>BE</td>
<td>(60h) 0110 0000 or (C7h) 1100 0111</td>
<td>Bulk Erase</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>Program</td>
<td>PP</td>
<td>(02h) 0000 0010</td>
<td>Page Programming</td>
<td>3</td>
<td>0</td>
<td>0</td>
<td>1 to 256</td>
</tr>
<tr>
<td></td>
<td>QPP</td>
<td>(32h) 0011 0010</td>
<td>Quad Page Programming</td>
<td>3</td>
<td>0</td>
<td>0</td>
<td>1 to 256</td>
</tr>
<tr>
<td>Status &amp;</td>
<td>RDSR</td>
<td>(05h) 0000 0101</td>
<td>Read Status Register</td>
<td>3</td>
<td>0</td>
<td>0</td>
<td>1 to $\infty$</td>
</tr>
<tr>
<td>Configuration</td>
<td>WRR</td>
<td>(01h) 0000 0001</td>
<td>Write (Status &amp; Configuration) Registers</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>1 to $\infty$</td>
</tr>
<tr>
<td>Register</td>
<td>RCR</td>
<td>(35h) 0011 0101</td>
<td>Read Configuration Register (CFG)</td>
<td>0</td>
<td>0</td>
<td>1</td>
<td>1 to $\infty$</td>
</tr>
<tr>
<td></td>
<td>CLSR</td>
<td>(30h) 0011 0000</td>
<td>Reset the Erase and Program Fail Flag (SR5 and SR6) and restore normal operation</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>1</td>
</tr>
<tr>
<td>Power Saving</td>
<td>DP</td>
<td>(B9h) 1011 1001</td>
<td>Deep Power-Down</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td></td>
<td>RES</td>
<td>(ABh) 1010 1011</td>
<td>Release from Deep Power-Down Mode</td>
<td>0</td>
<td>0</td>
<td>3</td>
<td>0</td>
</tr>
<tr>
<td></td>
<td>OTPP</td>
<td>(42h) 0100 0010</td>
<td>Programs one byte of data in OTP memory space</td>
<td>3</td>
<td>0</td>
<td>1</td>
<td>1</td>
</tr>
<tr>
<td></td>
<td>OTPR</td>
<td>(4Bh) 0100 1011</td>
<td>Read data in the OTP memory space</td>
<td>3</td>
<td>0</td>
<td>0</td>
<td>1 to $\infty$</td>
</tr>
</tbody>
</table>
Reverse RFU Format
Reverse RFU Format
Reverse RFU Format

Awesomesauce!
Security Analysis

- No memory-space isolation / separation
- No “kernel”-level security
- Everything runs as supervisor mode on CPU

- Any vulnerability anywhere leads to full compromise
**POC Printer Rootkit**

- 3KB of ARM assembly
- Print-job interceptor
- Reverse-IP Proxy
- Engine-controller hijacker
- Live Demo @ 28c3
Embed in Doc

(reflexive attack)

\[
\begin{array}{cccc}
550 & 4.242549 & 6.061096 & 4.8480 \\
04 & 7.905825 & 4.242549 & 6.061096 \\
6 & 3.030548 & 6.061096 & 0.000000 \\
\end{array}
\]

\[
\text{[34,302.39801, 92.950996, mf]}
\]

\[
(+)s_{\text{rep}}p_{\text{end}}F_{\%\text{Trailer}}F_{\%\text{EOJ}}F_{\%\text{-12345}}F_{\%\text{-12345x@PjL ENT}}
\]

\[
\text{ENTER LANGUAGE=ACL}\%
\]

\[
\text{y UAT g \ g g g g g g g g g g g M g M P g w t}
\]

\[
(\text{M h h h h h h h h h h h h h h h h h h h h h h h h h h h h h h h h h h h h h h h h h h h h h h h h h h h h h h h h h h h h h h h h h h h h h h h h h h h h h h h h h h h h h h h h h h h h h h h h h h h h h h h h h h h h h h h h h h h h h h h h h h h h h h h h h h h h h h h h h h h h h h h h h h h h h h h h h h h h h h h h h h h h h h h h h h h h h h h h h h h h h h h h h h h h h h h h h h h h h h h h h h h h h h h h h h h h h h h h h h h h h h h h h h h h h h h h h h h h h h h h h h h h h h h h h h h h h h h h h h h h h h h h h h h h h h h h h h h h h h h h h h h h h h h h h h h h h h h h h h h h h h h h h h h h h h h h h h h h h h h h h h h h h h h h h h h h h h h h h h h h h h h h h h h h h h h h h h h h h h h h h h h h h h h h h h h h h h h h h h h h h h h h h h h h h h h h h h h h h h h h h h h h h h h h h h h h h h h h h h h h h h h h h h h h h h h h h h h h h h h h h h h h h h h h h h h h h h h h h h h h h h h h h h h h h h h h h h h h h h h h h h h h h h h h h h h h h h h h h h h h h h h h h h h h h h h h h h h h h h h h h h h h h h h h h h h h h h h h h h h h h h h h h h h h h h h h h h h h h h h h h h h h h h h h h h h h h h h h h h h h h h h h h h h h h h h h h h h h h h h h h h h h h h h h h h h h h h h h h h h h h h h h h h h h h h h h h h h h h h h h h h h h h h h h h h h h h h h h h h h h h h h h h h h h h h h h h h h h h h h h h h h h h h h h h h h h h h h h h h h h h h h h h h h h h h h h h h h h h h h h h h h h h h h h h h h h h h h h h h h h h h h h h h h h h h h h h h h h h h h h h h h h h h h h h h h h h h h h h h h h h h h h h h h h h h h h h h h h h h h h h h h h h h h h h h h h h h h h h h h h h h h h h h h h h h h h h h h h h h h h h h h h h h h h h h h h h h h h h h h h h h h h h h h h h h h h h h h h h h h h h h h h h h h h h h h h h h h h h h h h h h h h h h h h h h h h h h h h h h h h h h h h h h h h h h h h h h h h h h h h h h h h h h h h h h h h h h h h h h h h h h h h h h h h h h h h h h h h h h h h h h h h h h h h h h h h h h h h h h h h h h h h h h h h h h h h h h h h h h h h h h h h h h h h h h h h h h h h h h h h h h h h h h h h h h h h h h h h h h h h h h h h h h h h h h h h h h h h h h h h h h h h h h h h h h h h h h h h h h h h h h h h h h h h h h h h h h h h h h h h h h h h h h h h h h h h h h h h h h h h h h h h h h h h h h h h h h h h h h h h h h h h h h h
Let's Quantify
Disclosure: November 21st

Firmware Release: December 23rd

56

Printer Firmwares Have Been Updated

2005 - 2011

2/25/13

Cui Costello Stolfo - NDSS 2013
CVE: CVE-2011-4161  SSRT: 100692 rev.6
How many vulnerable printers are there in the world?

MONTHS AFTER PATCH RELEASE

How many vulnerable printers are there in the world?
<table>
<thead>
<tr>
<th>P otentially vulnerable printers</th>
<th>90,847</th>
</tr>
</thead>
<tbody>
<tr>
<td>Printers with identifiable</td>
<td>74,770</td>
</tr>
<tr>
<td>firmware datecode</td>
<td></td>
</tr>
<tr>
<td>Number of patched printers</td>
<td>808</td>
</tr>
<tr>
<td>Overall patch rate</td>
<td>1.08%</td>
</tr>
</tbody>
</table>

**TABLE I**

**Observed population of printers vulnerable to the HP-RFU attack on IPv4.**

MONTHS AFTER PATCH RELEASE

HOW MANY VULNERABLE PRINTERS ARE THERE IN THE WORLD?
How many vulnerable printers are there in the world?

Potentially vulnerable printers
Printers with identifiable firmware datecode
Number of patched printers
Overall patch rate

TABLE I
OBSERVED POPULATION OF PRINTERS VULNERABLE TO THE HP-RFU ATTACK ON IPv4.

MONTHS AFTER PATCH RELEASE

HOW MANY VULNERABLE PRINTERS ARE THERE IN THE WORLD?
Potentially vulnerable printers
Printers with identifiable firmware datecode
Number of patched printers
Overall patch rate

|                              | 76,288 | 5659 | 7.42% |

**TABLE I**

**Observed population of printers vulnerable to the HP-RFU attack on IPv4.**

**14 months after patch release**

**How many vulnerable printers are there in the world?**
Interesting Findings

- EDU has the most vulnerable printers

<table>
<thead>
<tr>
<th></th>
<th>Count</th>
<th>Avg Age (years)</th>
<th>Oldest Firmware</th>
</tr>
</thead>
<tbody>
<tr>
<td>Education</td>
<td>48,626</td>
<td>4.13</td>
<td>1993-08-20</td>
</tr>
<tr>
<td>ISP</td>
<td>4,650</td>
<td>3.70</td>
<td>1994-10-12</td>
</tr>
<tr>
<td>Enterprise</td>
<td>2,842</td>
<td>4.02</td>
<td>1992-12-16</td>
</tr>
<tr>
<td>Military</td>
<td>201</td>
<td>4.63</td>
<td>1999-10-30</td>
</tr>
<tr>
<td>Government</td>
<td>126</td>
<td>4.33</td>
<td>1996-12-20</td>
</tr>
</tbody>
</table>

TABLE III
Organizational distribution of vulnerable printers.
Interesting Findings

- **EDU has the most vulnerable printers**
- **Average printer is ~4.5 years old**

<table>
<thead>
<tr>
<th>Region</th>
<th>Count</th>
<th>Avg Age (years)</th>
<th>Oldest Firmware</th>
</tr>
</thead>
<tbody>
<tr>
<td>N. America</td>
<td>47,840</td>
<td>4.46</td>
<td>1992-12-16</td>
</tr>
<tr>
<td>Europe</td>
<td>14,196</td>
<td>4.16</td>
<td>1993-08-20</td>
</tr>
<tr>
<td>Asia</td>
<td>10,353</td>
<td>3.77</td>
<td>1998-09-02</td>
</tr>
<tr>
<td>Oceania</td>
<td>1,081</td>
<td>4.79</td>
<td>1998-09-02</td>
</tr>
<tr>
<td>S. America</td>
<td>673</td>
<td>3.43</td>
<td>1999-01-27</td>
</tr>
<tr>
<td>Africa</td>
<td>60</td>
<td>4.56</td>
<td>2001-04-26</td>
</tr>
</tbody>
</table>
Interesting Findings

- EDU has the most vulnerable printers
- Average printer is ~4.5 years old
- Found **201 vulnerable printers in DOD (All removed)**
Interesting Findings

• EDU has the most vulnerable printers
• Average printer is ~4.5 years old

• Found 201 vulnerable printers in DOD (All removed)
• Found 6 vulnerable printers in HP (3 still there)
Patch out
Problem Solved?
VULNERABLE THIRD-PARTY LIBRARIES

zlib: CA-2002-07, CERT- {68062, 238678} Discovered in 2002, zlib ver. 1.1.3 and earlier have a double free bug that allows arbitrary code execution [20]. In 2005 the vendor was notified of a buffer overflow in zlib ver. 1.2.1 and 1.2.2 [21]. The vendor was notified of a DOS condition in zlib ver. 1.2.0.x and 1.2.x in 2004 [22].

OpenSSL: CVE- {2009-3245, 2006-3738, 2006-4339} There are over 100 known vulnerabilities in various versions of OpenSSL. We scanned for the above three critical vulnerabilities in our firmware update dataset because they involve features that are likely to be reachable via network attack. The first two vulnerabilities can lead to arbitrary code execution. The last vulnerability can bypass x.509 certificate verification.
<table>
<thead>
<tr>
<th>Description</th>
<th>Value</th>
</tr>
</thead>
<tbody>
<tr>
<td>Printer models analyzed</td>
<td>63</td>
</tr>
<tr>
<td>RFU images analyzed</td>
<td>373</td>
</tr>
<tr>
<td>All RFUs w/ at least 1 vulnerability</td>
<td>300</td>
</tr>
<tr>
<td>Latest RFUs w/ at least 1 vulnerability</td>
<td>41 (65.1%)</td>
</tr>
<tr>
<td>Most common zlib version</td>
<td>1.1.4</td>
</tr>
<tr>
<td>Most common OpenSSL version</td>
<td>0.9.7b</td>
</tr>
</tbody>
</table>

**TABLE VI**

**THIRD-PARTY LIBRARY VULNERABILITY ANALYSIS OBSERVATIONS.**
THIRD-PARTY LIBRARY VULNERABILITIES FOUND IN PRINTER Firmware UPDATES

Printer Models

RFUs w/ Vuln. OpenSSL
RFUs w/ Vuln. zlib
Take Aways
HACK A

Printer

Phone
Hack A

OWN A

Printer
Phone

Anti-Virus
IDS
Patch-Tuesday

Server
Signed code ≠ Secure code
M E E T
SYMBIOTE

RAID 2011
Defending Legacy Embedded Systems With Software Symbiotes

ACSAC 2011
From Prey To Hunter: Transforming Legacy Embedded Devices Into Exploitation Sensor Grids

USENIX/WOOT 2011
Killing the Myth of Cisco IOS Diversity

NDSS 2013
When Firmware Modifications Attack: A Case Study of Embedded Exploitation
Symbiote Structure

Drop in a Defensive Symbiote Payload
Much Thanks!

ANUP KOTALWAR
JATIN KATARIA
YUAN KANG
Much Thanks!

ANUP KOTALWAR
JATIN KATARIA
YUAN KANG
ang@cs.columbia.edu