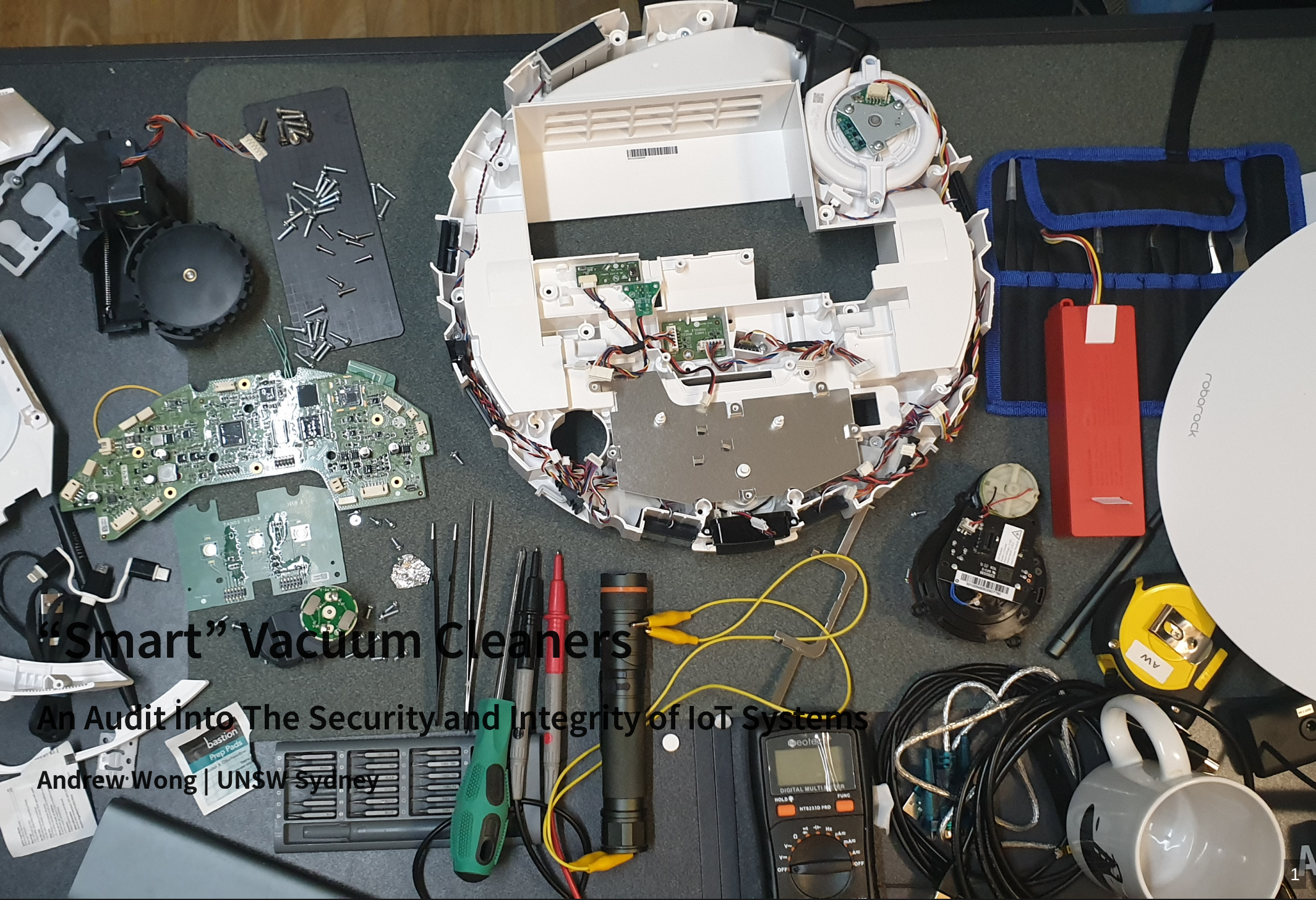


“Smart” Vacuum Cleaners

An Audit into The Security and Integrity of IoT Systems

Andrew Wong | UNSW Sydney

Introduction

Internet of Things (IoT) and Smart Home devices are everywhere.

Q: Can we completely trust a device's {security, privacy}?
A: no

We should always verify and test things where possible!

Introduction

Internet of Things (IoT) and Smart Home devices are everywhere.

Q: Can we completely trust a device's {security, privacy}?
A: no

- Developers are humans.
  - Humans make mistakes.
    - Developers make ~~mistakes~~ bugs
- Or maybe secret company agendas?

We should always verify and test things where possible!

Introduction

Internet of Things (IoT) and Smart Home devices are everywhere.

Q: Can we completely trust a device's {security, privacy}?
A: no

- Developers are humans.
  - Humans make mistakes.
    - Developers make ~~mistakes~~ bugs
- Or maybe secret company agendas?

We should always verify and test things where possible!

About Me

Andrew Wong
Computer Engineering @ UNSW Sydney

e: andrew.j.wong@student.unsw.edu.au

Interests

Making things, breaking things… mainly the latter

Background Information

4.1

- Robotic home cleaning appliances
- Founded in July 2014, Beijing
- Partnered with Xiaomi in September 2014
  - Investments + Partnership

- September 2016 - Mi Home Robotic Vaccuum Cleaner
  - Very first product!
- ⋮ Roborock S5, E2, E3
- June 2019 - Roborock S6
- ⋮ Roborock S5 Max, S4, S6 Pure, S6 MaxV, E4, S4 Max
- January 2021 - S7

Background Information

Roborock S6 Vacuum Cleaner

- Chips, specs

- ADB Port?

Xiaomi Smartphone Application

Ubuntu

Background Information

Widespread product release

- Buy ‘generic’, resell
- Buy into the framework
- i.e. Tuya (smart home), Xiaomi
- Use their ecosystem / framework
  - And their associated risks

MENTION SOMETHING ABOUT ALEXA, GOOGLE ASSISTANT, SIRI IN THE INTRODUCTION?

https://github.com/OpenMiHome/mihome-binary-protocol/blob/master/doc/PROTOCOL.md

- Liberation from coupled service
- Privacy?

RRoR (DustCloud) MiCloudFaker Valetudo

Xiaomi Integration

MilO HomeAssistant OpenHab

Rationale

Security is important!

Check things for (y)ourself!

Statement

How have manufacturers of IoT / smart home devices addressed the increasing concerns of digital privacy and product security?

Proposal

Digital Privacy

Investigate the nature of network data (i.e. content, frequency, destination) from the Roborock S6, and how the data is used.

Product Security

Investigate potential security vulnerabilities of the Roborock S6, and assess the effectiveness of current security fortifications.

Literature Review

2017 - Dennis Giese

- Found ways to root the Mi Home Robotic Vacuum Cleaner and the Roborock S6
- https://dontvacuum.me
- Continued analysis on later Xiaomi (and derivative) vacuum cleaners

Dustcloud

TODO: OTHER PAPERS

Plan

- Research
- Get the Roborock S6 vacuum cleaner
- Acquisition and capture of network activity
- Find a way in (it runs Linux!)
- Image the system for offline analysis
- Reverse engineering and binary analysis of firmware and software
  - Look through binaries for security vulnerabilities and fortifications

Contingency

If we can’t get into the device?

- Option 1 - Protocol analysis (network traffic)
  - i.e. Inspect the data and its nature
    - Content, Frequency, Destination
- Option 2 - Investigate the Xiaomi Home smartphone application (used to communicate with the device)
  - i.e. Decompile the Android APK file and look for security vulnerabilities and fortifications

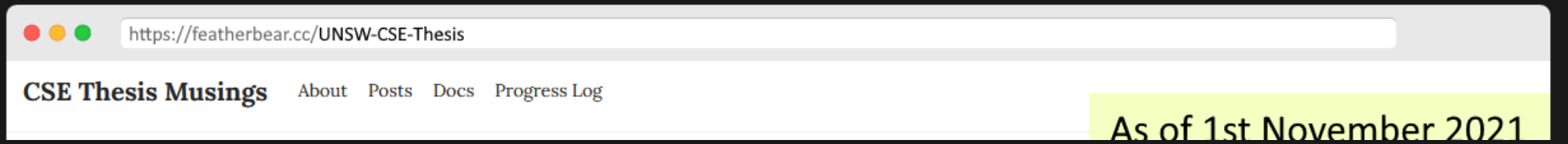
Future Plans

- See what the sensors see
- Circuit board decomposition
- Analyse the custom ADB binary serving the USB port

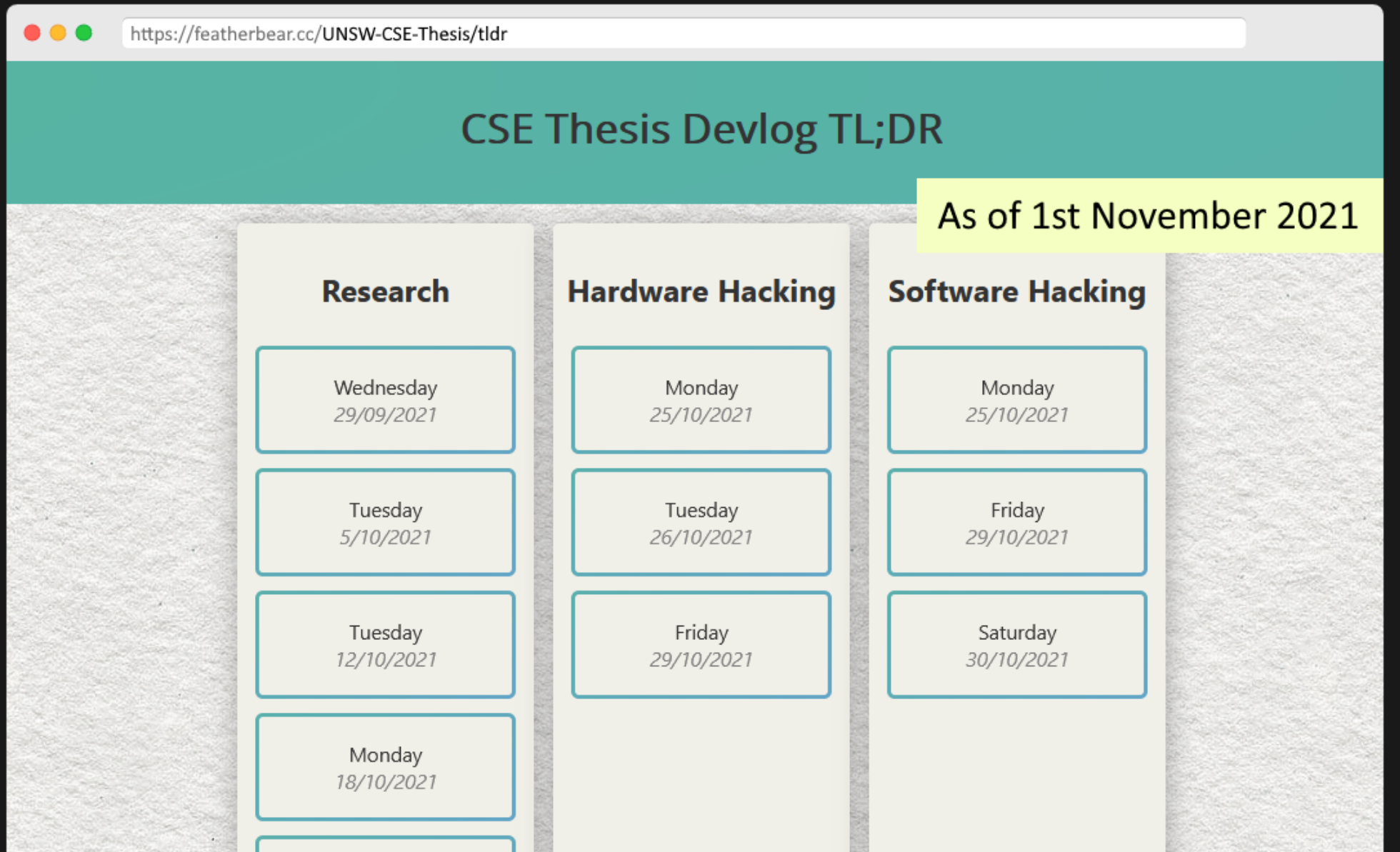
Project Timeline

Rolling Research

featherbear.cc/UNSW-CSE-Thesis


https://featherbear.cc/UNSW-CSE-Thesis
CSE Thesis Musings
About
Posts
Docs
Progress Log
As of 1st November 2021

featherbear.cc/UNSW-CSE-Thesis/tldr



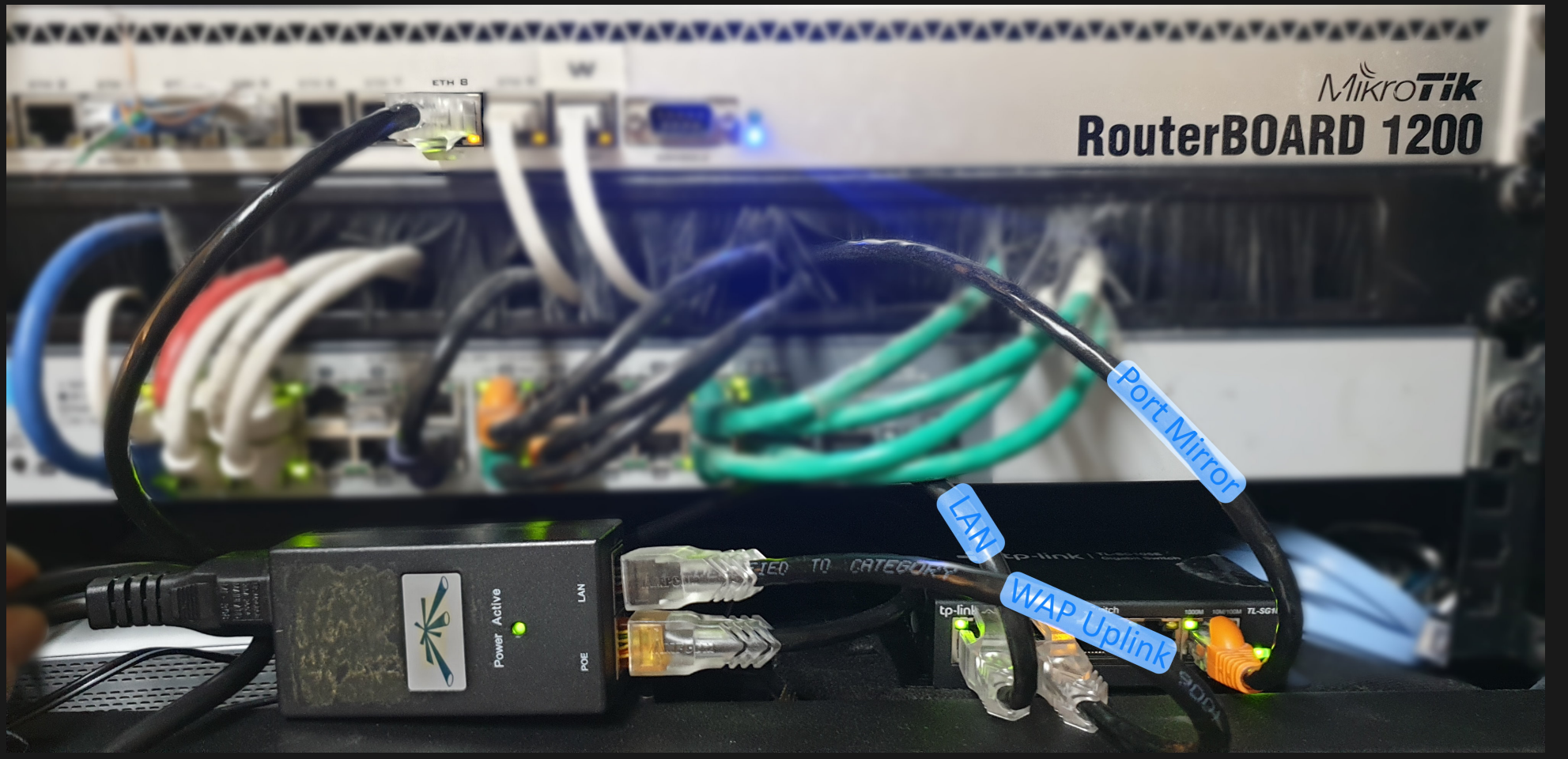
Monday
1/11/2021

Research, Upskill, Tooling

- How to capture network activity without compromising my home network?
- Interfacing with JTAG / UART / Serial
- Linux forensics
- Learn the ARM ISA
  - Processor Modes, Protection Rings?

Preliminary Results

Network Setup

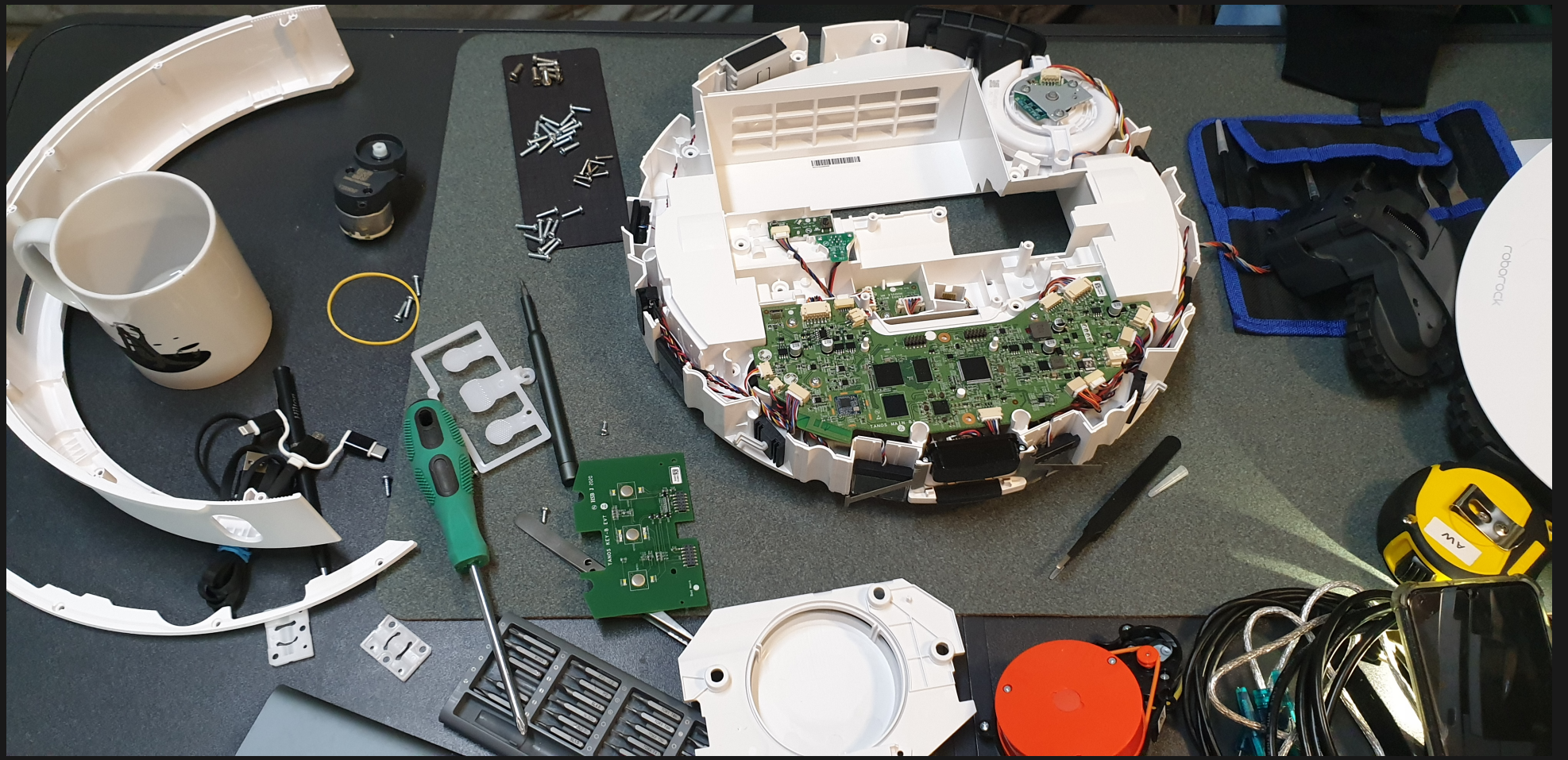


[Initial] Packet Capture

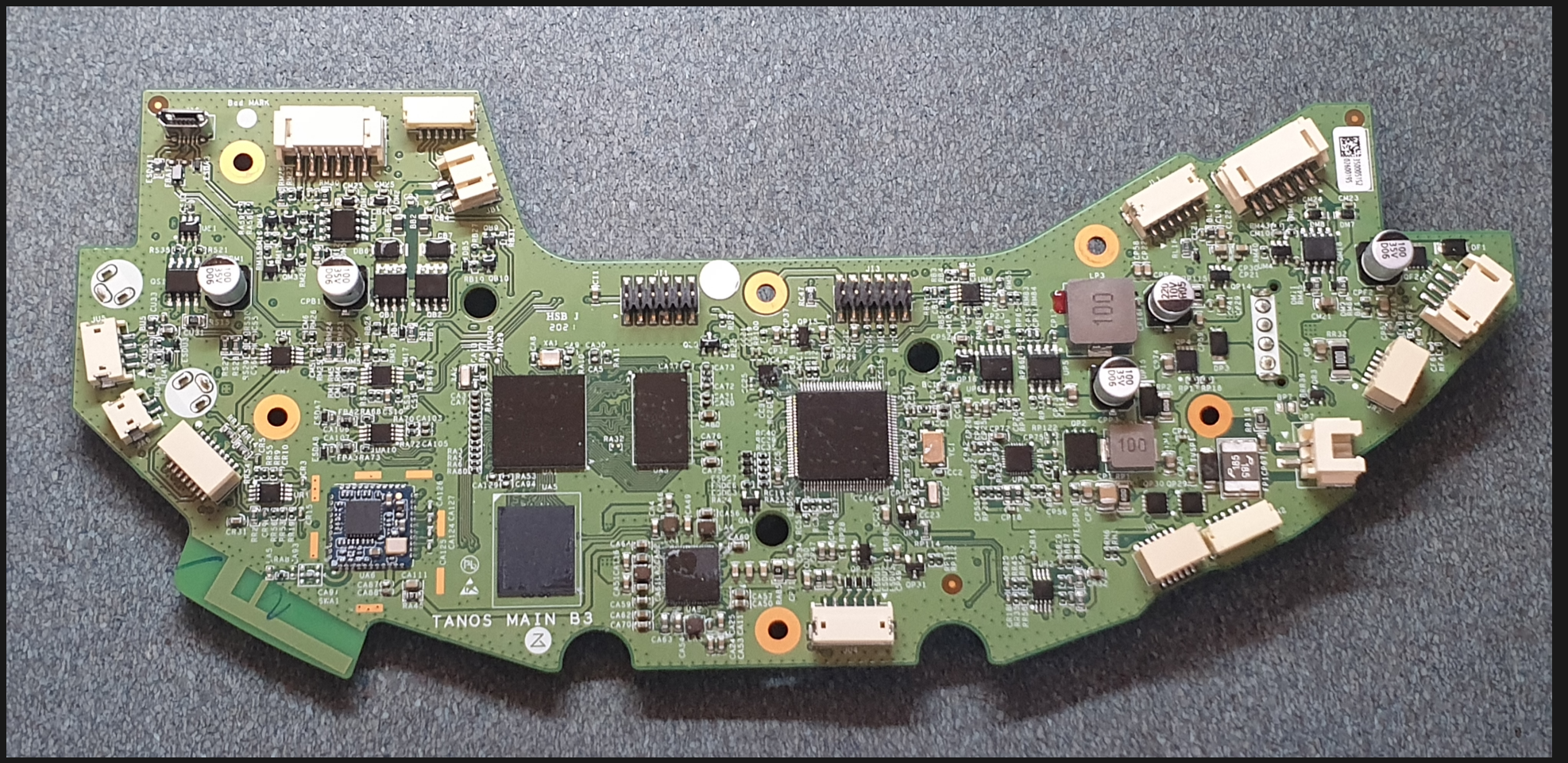
No.	Time	Source	Destination	Protocol	Length	Info
1291…	4315.804111	BeijingX_1d:24:c4	Broadcast	ARP	107	Who has 10.10.10.1? Tell 10.10.10.8
1291…	4315.804161	Routerbo_cf:36:21	BeijingX_1d:24:c4	ARP	89	10.10.10.1 is at 00:0c:42:cf:36:21
1291…	4315.805362	10.10.10.8	110.43.0.85	TCP	121	41134 → 80 [SYN] Seq=0 Win=14600 Len=0 MSS=1460 SACK_PERM=1 TSval=4294939886 TSecr=0 WS=64
1291…	4316.203673	110.43.0.85	10.10.10.8	TCP	121	80 → 41134 [SYN, ACK] Seq=0 Ack=1 Win=8192 Len=0 MSS=1340 SACK_PERM=1 WS=32
1291…	4316.205277	10.10.10.8	110.43.0.85	TCP	107	41134 → 80 [ACK] Seq=1 Ack=1 Win=14656 Len=0
1291…	4316.205876	10.10.10.8	110.43.0.85	HTTP	299	GET /gslb?tver=2&id=322119905&dm=sg.ott.io.mi.com&timestamp=1635171460&sign=Lc9j28ajJwk7nMufGq2APVGiElKwzagUsZ%2FyuIRj79Q%3D HTTP/1.1
1292…	4316.505288	0.0.0.0	255.255.255.255	DHCP	389	DHCP Discover - Transaction ID 0x731c1bc8
1292…	4316.515522	0.0.0.0	255.255.255.255	DHCP	389	DHCP Discover - Transaction ID 0x7448626d
1292…	4316.605227	110.43.0.85	10.10.10.8	TCP	101	80 → 41134 [ACK] Seq=1 Ack=199 Win=30336 Len=0
1292…	4316.605288	110.43.0.85	10.10.10.8	HTTP/JSON	300	HTTP/1.1 400 Bad Request , JavaScript Object Notation (application/json)
1292…	4316.606968	10.10.10.8	110.43.0.85	TCP	107	41134 → 80 [ACK] Seq=199 Ack=200 Win=15680 Len=0
1292…	4316.606968	110.43.0.85	10.10.10.8	TCP	101	80 → 41134 [FIN, ACK] Seq=200 Ack=199 Win=30336 Len=0
1292…	4316.607530	10.10.10.8	110.43.0.85	TCP	107	41134 → 80 [FIN, ACK] Seq=199 Ack=200 Win=15680 Len=0
1292…	4316.608113	10.10.10.8	110.43.0.83	TCP	121	55090 → 80 [SYN] Seq=0 Win=14600 Len=0 MSS=1460 SACK_PERM=1 TSval=4294939967 TSecr=0 WS=64
1292…	4316.608683	10.10.10.8	110.43.0.85	TCP	107	41134 → 80 [ACK] Seq=200 Ack=201 Win=15680 Len=0
1292…	4316.625602	110.43.0.85	10.10.10.8	TCP	101	[TCP Out-Of-Order] 80 → 41134 [FIN, ACK] Seq=200 Ack=199 Win=30336 Len=0
1292…	4316.628140	10.10.10.8	110.43.0.85	TCP	113	[TCP Dup ACK 129208#1] 41134 → 80 [ACK] Seq=200 Ack=201 Win=15680 Len=0 SLE=200 SRE=201
1292…	4316.802243	10.10.10.7	10.10.10.1	DNS	122	Standard query 0xed04 A eas.outlook.com
1292…	4316.815908	110.43.0.85	10.10.10.8	TCP	300	[TCP Out-Of-Order] 80 → 41134 [FIN, PSH, ACK] Seq=1 Ack=199 Win=30336 Len=199
1292…	4316.817453	10.10.10.8	110.43.0.85	TCP	113	[TCP Dup ACK 129208#2] 41134 → 80 [ACK] Seq=200 Ack=201 Win=15680 Len=0 SLE=1 SRE=201
1292…	4316.971851	10.10.10.7	10.10.10.1	DNS	125	Standard query 0x02dc A account.xiaomi.com
1292…	4317.006784	110.43.0.85	10.10.10.8	TCP	101	80 → 41134 [ACK] Seq=201 Ack=200 Win=30336 Len=0
1292…	4317.027181	110.43.0.85	10.10.10.8	TCP	101	80 → 41134 [RST] Seq=201 Win=0 Len=0
1292…	4317.047587	110.43.0.83	10.10.10.8	TCP	121	80 → 55090 [SYN, ACK] Seq=0 Ack=1 Win=8192 Len=0 MSS=1340 SACK_PERM=1 WS=32
1292…	4317.049209	10.10.10.8	110.43.0.83	TCP	107	55090 → 80 [ACK] Seq=1 Ack=1 Win=14656 Len=0
1292…	4317.049348	10.10.10.8	110.43.0.83	HTTP	300	GET /gslb?tver=2&id=322119905&dm=sg.ot.io.mi.com&timestamp=1635171461&sign=y4ipkGw7yjTyoKEoXTTQF0D2IsRB5T2O%2BDrkec5%2FhHg%3D HTTP/1.1
1292…	4317.216362	110.43.0.85	10.10.10.8	TCP	101	80 → 41134 [RST] Seq=201 Win=0 Len=0
1292…	4317.483104	110.43.0.83	10.10.10.8	TCP	101	80 → 55090 [ACK] Seq=1 Ack=200 Win=30464 Len=0
1292…	4317.483104	110.43.0.83	10.10.10.8	HTTP/JSON	300	HTTP/1.1 400 Bad Request , JavaScript Object Notation (application/json)
1292…	4317.483135	110.43.0.83	10.10.10.8	TCP	101	80 → 55090 [FIN, ACK] Seq=200 Ack=200 Win=30464 Len=0
1292…	4317.485013	10.10.10.8	110.43.0.83	TCP	107	55090 → 80 [ACK] Seq=200 Ack=200 Win=15680 Len=0
1292…	4317.485075	10.10.10.8	110.43.0.83	TCP	107	55090 → 80 [FIN, ACK] Seq=200 Ack=201 Win=15680 Len=0
1292…	4317.486424	10.10.10.8	10.10.10.1	DNS	122	Standard query 0x8180 A sg.ot.io.mi.com
1292…	4317.489390	110.43.0.83	10.10.10.8	TCP	101	[TCP Out-Of-Order] 80 → 55090 [FIN, ACK] Seq=200 Ack=200 Win=30464 Len=0
1292…	4317.490689	10.10.10.8	110.43.0.83	TCP	113	[TCP Dup ACK 129225#1] 55090 → 80 [ACK] Seq=201 Ack=201 Win=15680 Len=0 SLE=200 SRE=201
1292…	4317.507735	0.0.0.0	255.255.255.255	DHCP	389	DHCP Discover - Transaction ID 0x57bd3221
1292…	4317.517990	0.0.0.0	255.255.255.255	DHCP	389	DHCP Discover - Transaction ID 0x3260a72
1292…	4317.702426	110.43.0.83	10.10.10.8	TCP	300	[TCP Out-Of-Order] 80 → 55090 [FIN, PSH, ACK] Seq=1 Ack=200 Win=30464 Len=199
1292…	4317.703797	10.10.10.8	110.43.0.83	TCP	113	[TCP Dup ACK 129225#2] 55090 → 80 [ACK] Seq=201 Ack=201 Win=15680 Len=0 SLE=1 SRE=201
1292…	4317.901091	10.10.10.7	10.10.10.1	DNS	121	Standard query 0x83d2 A www.google.com
1292…	4317.911588	110.43.0.83	10.10.10.8	TCP	101	80 → 55090 [ACK] Seq=201 Ack=201 Win=30464 Len=0

- No LAN-LAN packets???
- incomplete test - misconfigured packet capture setup

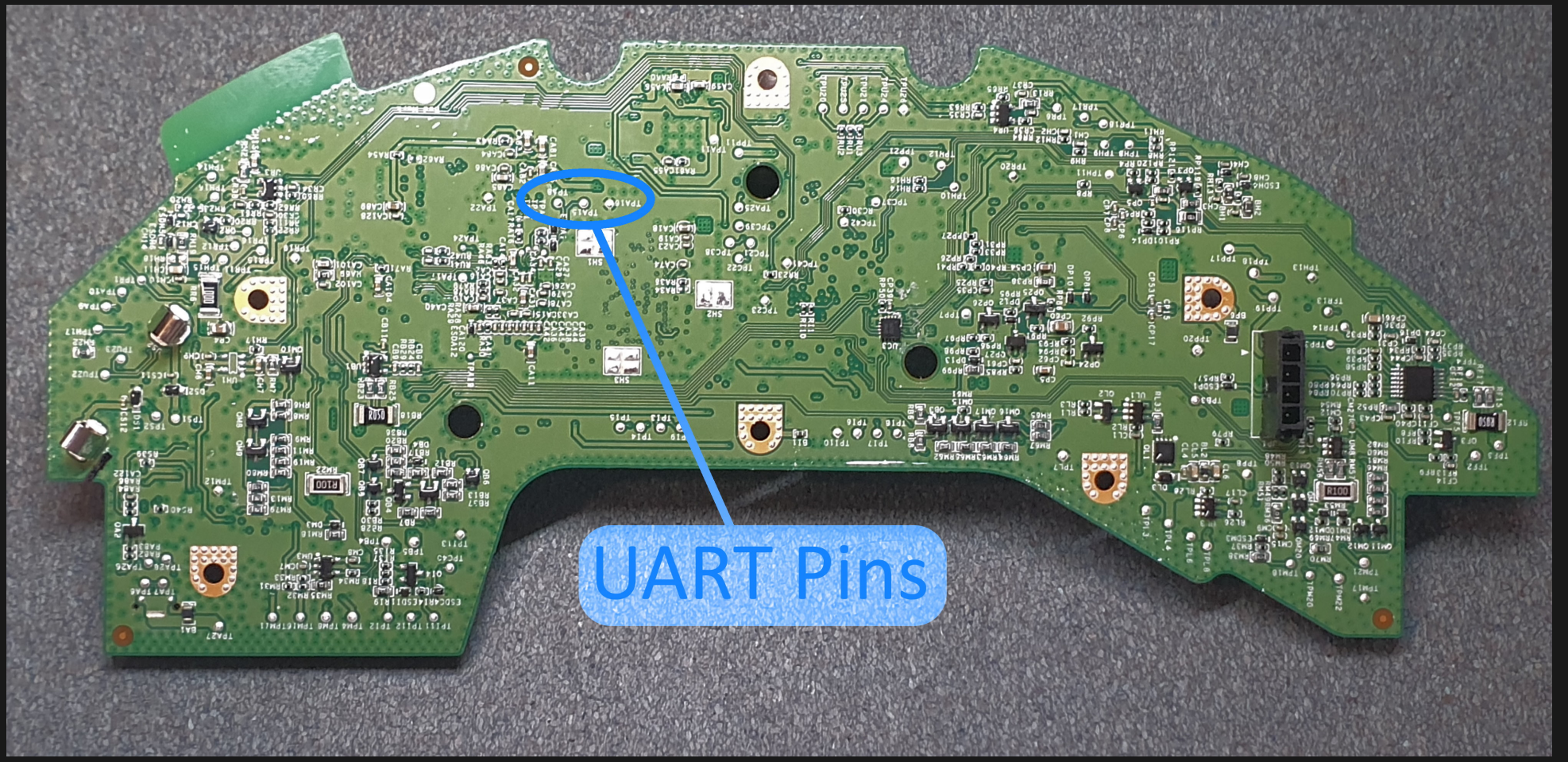
Teardown



Initial Breakdown and Pinout (where needed)

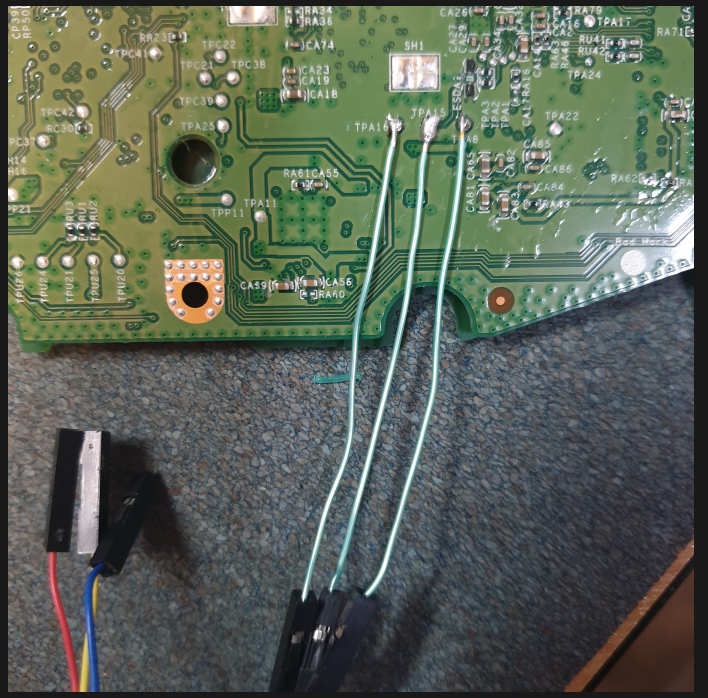


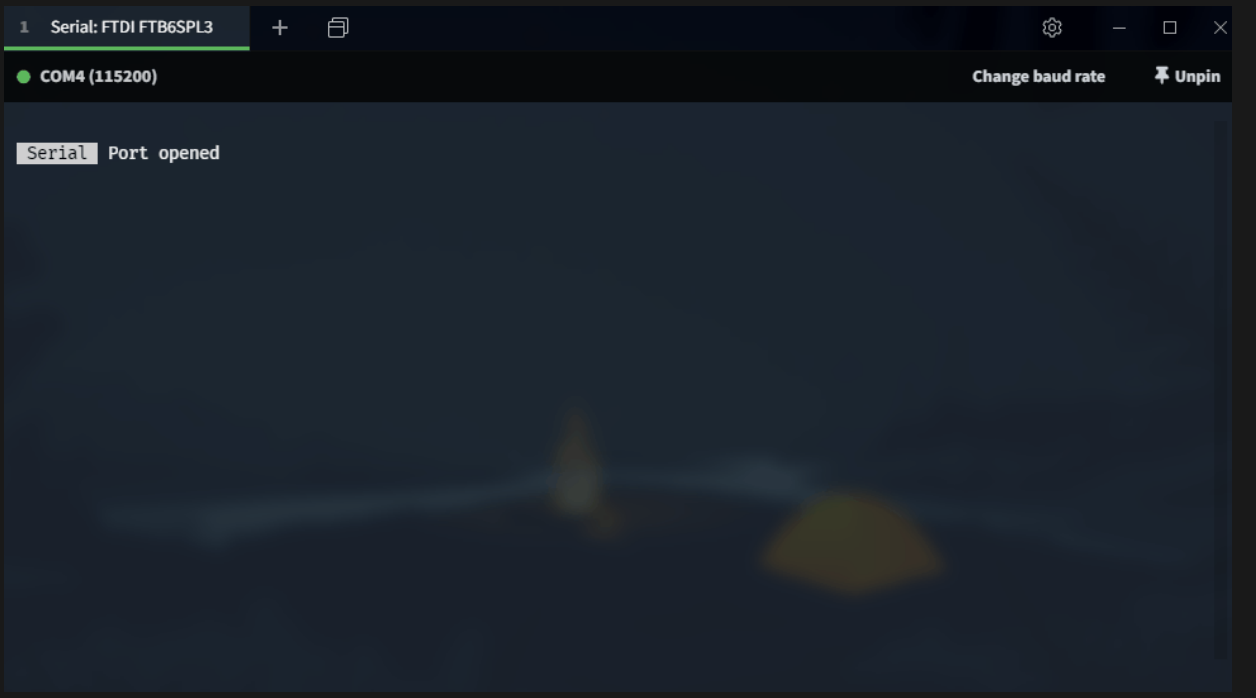
Identification of the UART pins



Serial Access

- Serial (baud=115200) gives us a shell!





Need a root password though…

U-Boot Bootloader

https://github.com/allwinner-zh/bootloader/blob/master/u-boot-2011.09/board/sunxi/board_common.c#L843-L847

```
842         }
843         else if(c == 's')               //shell mode
844         {
845                 gd->force_shell = 1;
846                 gd->debug_mode = 1;
847         }
848         return 0;
849 }
```

- Able to enter the bootloader shell if s is pressed during init

```
flag_bootA:0×1
upgrade stage:0×63
No upgradeover system found, check if has normal system, pmu: 0×0
board_common.c:check_android_misc:will be boot A system
to be run cmd=run setargs_mmc boot_normal
boot A system
WORK_MODE_BOOT
[      1.996]Hit any key to stop autoboot:  0
sunxi#
```

Root!

```
sunxi#ext4load
ext4load - load binary file from a Ext4 filesystem

Usage:
ext4load <interface> <dev[:part]> [addr] [filename] [bytes]
          - load binary file 'filename' from 'dev' on 'interface'
                to address 'addr' from ext4 filesystem
sunxi#ext4load mmc 2:6 0 vinda
Loading file "vinda" from mmc device 2:6
16 bytes read
sunxi#md 0 4
00000000: 5b415243 51454346 54505042 525f5655    CRA[FCEQBPPTUV_R
```

```
rockrobo login: root
Password:
Welcome to Ubuntu 14.04.3 LTS (GNU/Linux 3.4.39 armv7l)

 * Documentation:  https://help.ubuntu.com/

The programs included with the Ubuntu system are free software;
the exact distribution terms for each program are described in the
individual files in /usr/share/doc/*/copyright.

Ubuntu comes with ABSOLUTELY NO WARRANTY, to the extent permitted by
applicable law.

root@rockrobo:~#
```

Moving Ahead

- Dump the firmware and begin RE / forensics
- Redo (and further investigate) live system analysis

Any Questions?