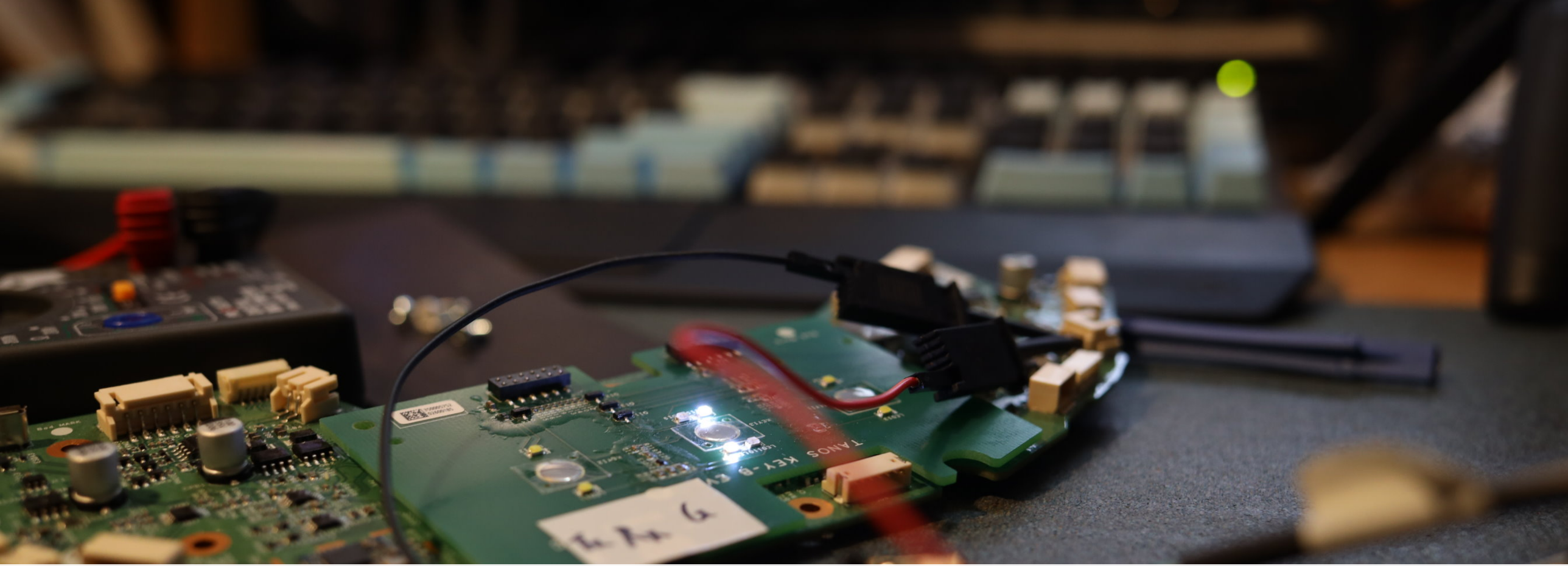


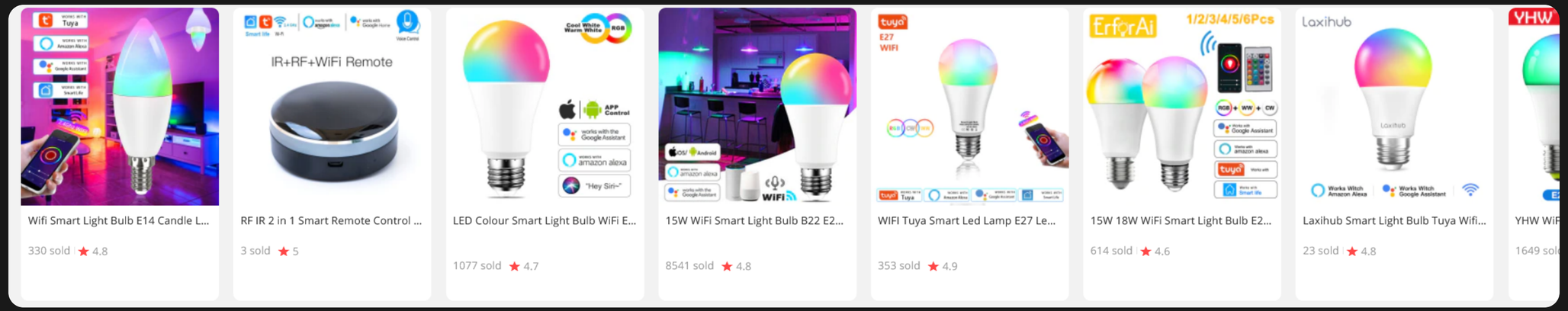
“Smart” Vacuum Cleaners

An Audit Into The Security and Integrity of IoT Systems

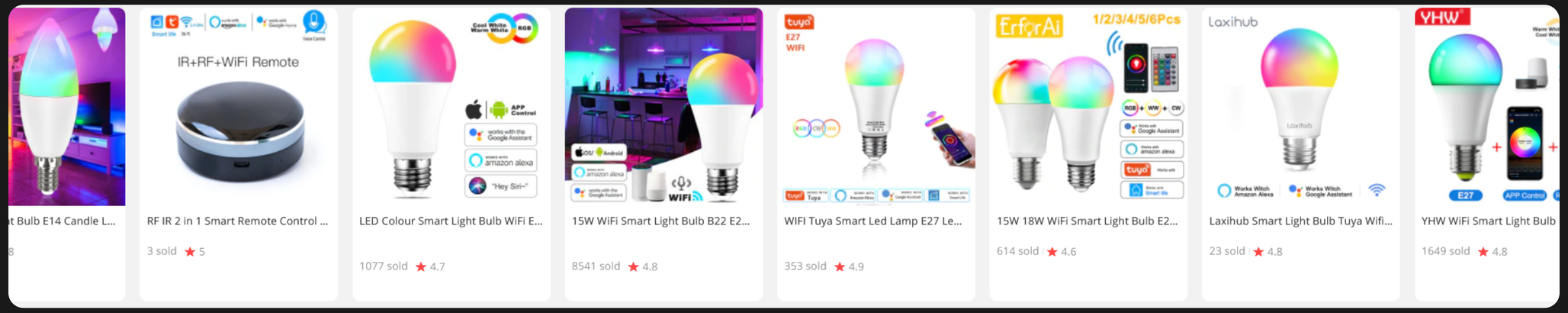
Andrew Wong | UNSW Sydney

Today’s Agenda

- Topic Recap
  - The world of IOT devices
  - Thesis statement
- Where we left off
  - Previous results and findings
- New progress
- Discussion
- Conclusion



…so there are a lot of IOT devices and IOT brands out there…



Are competing products looking suspiciously similar to you?

Most are white-labelled products, the biggest ecosystem vendor being tuya

Pros

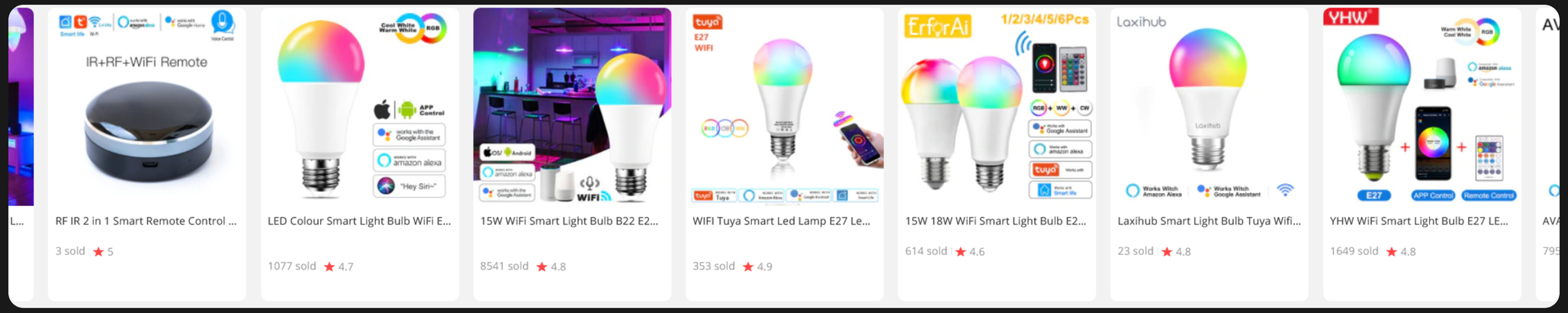
Use someone else's code

Fast profit turnaround

Cons

⚠ Use someone else's code

Potentially security vulnerabilities



IOT ecosystems often have a centralised system to manage their fleet (devices).

Pros

A centralised management is so much simpler/easier/faster/cheaper/‘better’ than a decentralised one.

Cons

⚠ Device functionality dependent on system availability
⚠ Little transparency about what/where/when/why data is transmitted

Statement

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

(Specifically Roborock / Xiaomi / Tuya)

- Digital Privacy
  - Investigate the nature of network data
  - i.e. content, frequency, destination, usage
- Product Security
  - Investigate security vulnerabilities
  - Assess the effectiveness of security fortifications

Statement

Our device in scope: Roborock S6 (2019)

A smart vacuum cleaner, with integrations to both tuya and Mi (depending on model)

It works pretty (very) well, according to reviews.
But is it safe to connect to your home?

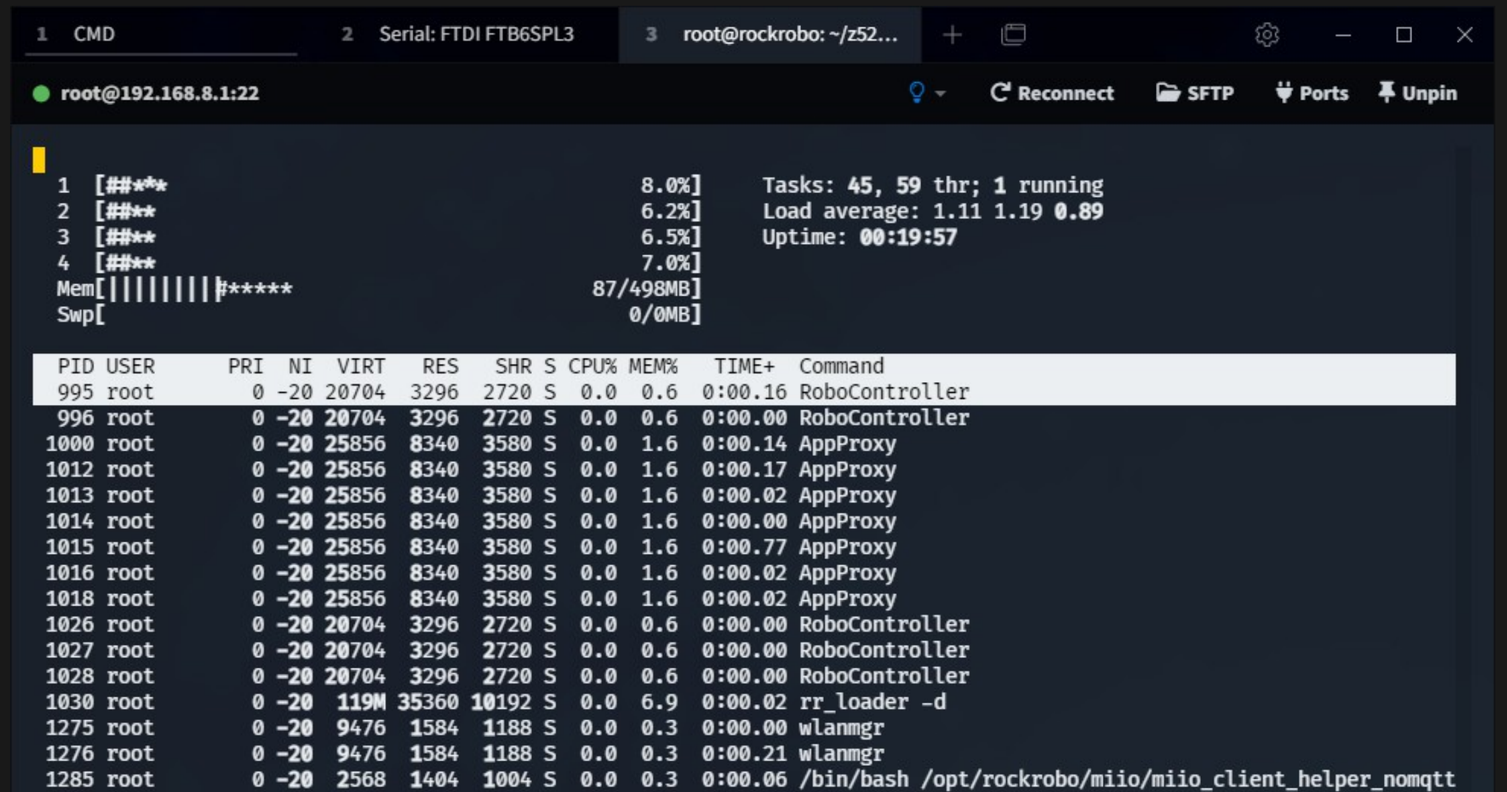
Talk about thesis A talk about thesis b Talk about thesis c

Thesis B | Results

Fingerprinting

Processes

Everything is running as root



Thesis B - Recovery partition manipulation (see proof of concept)

Issues, thoughts & discussions

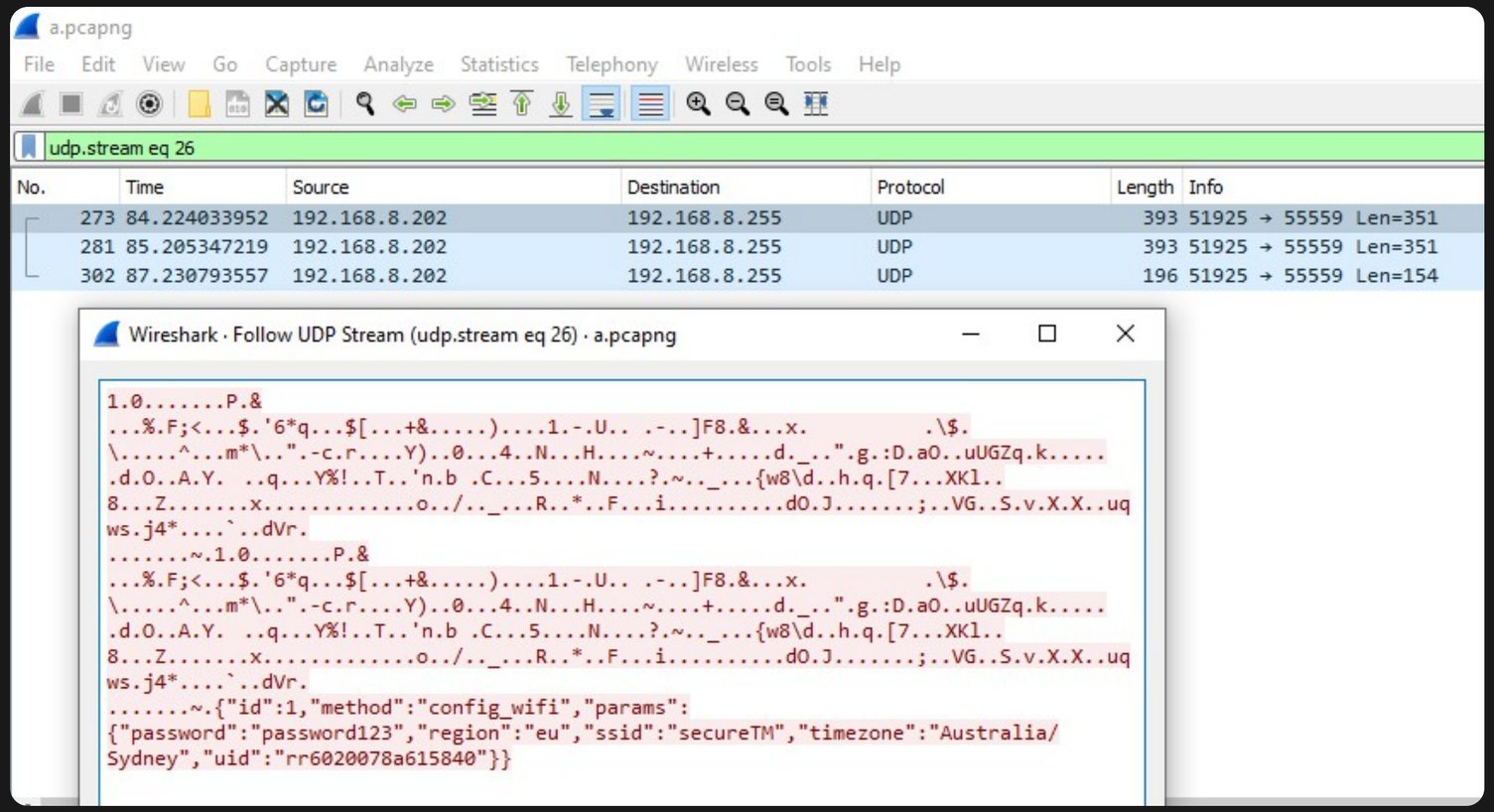
How have manufacturers of IoT / smart home devices addressed the increasing concerns of digital privacy and product security?

Recovery partition is modifiable

- Can be modified to contain malicious software that persists a factory reset
- Mountable - `mount /dev/mmcblk0p7 ...`
- Why? Allows easy updates of the ‘factory image’
- But the partition could somehow be encrypted

Speaking of packets…

WiFi credentials in plain text during setup



Thesis B - Inspection of system services (`netstat`, `iptables`, `ip6tables`)

Fingerprinting

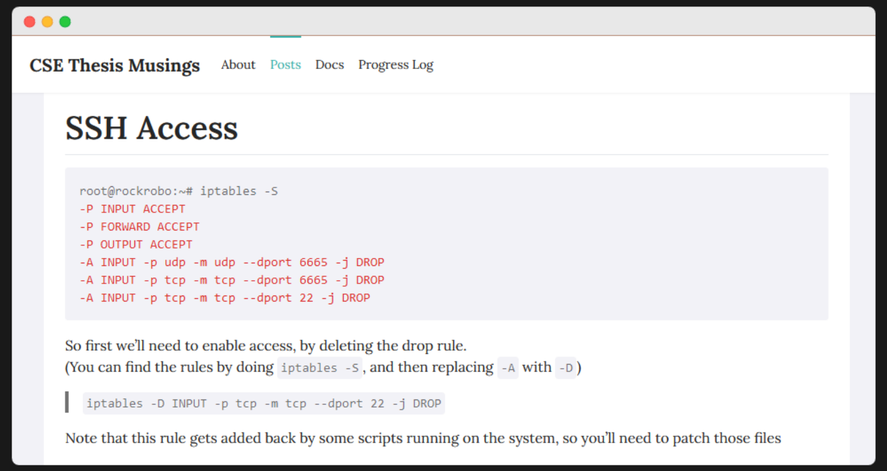
Ports

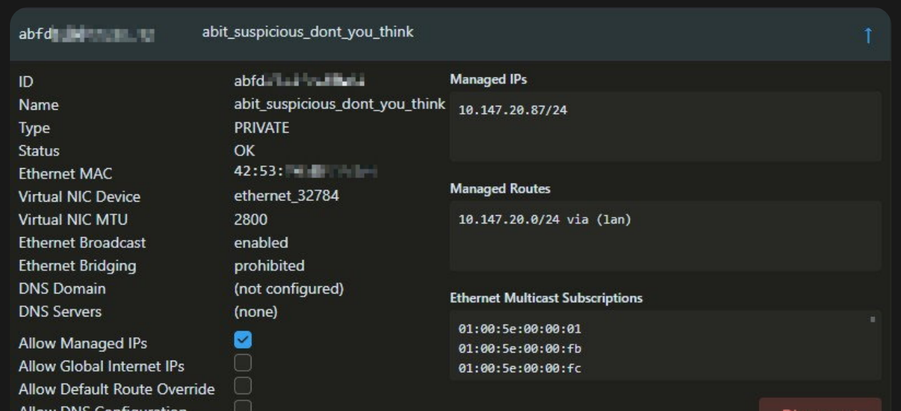
```
root@rockrobo:~# netstat -nltp
Active Internet connections (only servers)
Proto Recv-Q Send-Q Local Address           Foreign Address         State       PID/Progra
tcp        0      0 127.0.0.1:54322         0.0.0.0:*               LISTEN      991/miio_c
tcp        0      0 127.0.0.1:54323         0.0.0.0:*               LISTEN      991/miio_c
tcp        0      0 0.0.0.0:22              0.0.0.0:*               LISTEN      1644/sshd
tcp        0      0 127.0.0.1:55551         0.0.0.0:*               LISTEN      998/rriot_
tcp        0      0 0.0.0.0:6668            0.0.0.0:*               LISTEN      998/rriot_
tcp6       0      0 :::22                   :::*                    LISTEN      1644/sshd
```

`tcp/22` and `tcp/6668` are exposed

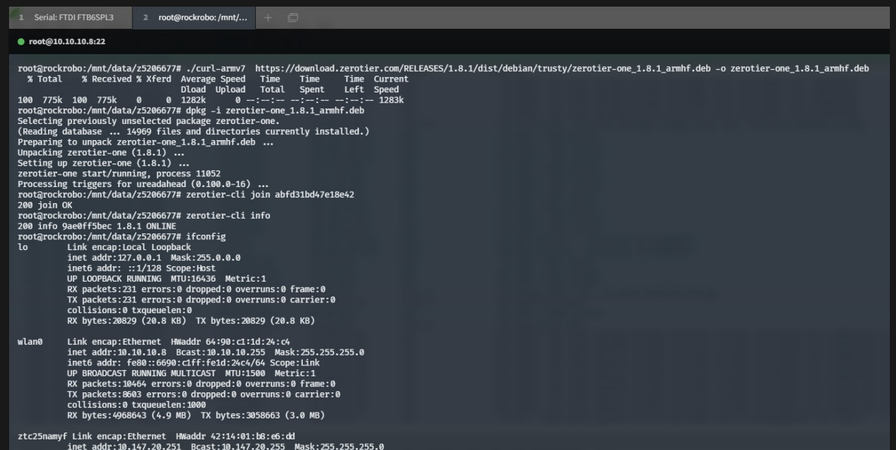
Thesis B - Remote access persistence (see proof of concept)

Going wireless - establishing SSH





- Remove iptables rule to gain access
  - (and so could an attacker)
- Can I add persistent access?
  - Yes, modify `rrwatchdoge.conf`
- Can also add remote access
  - e.g. ZeroTier



Thesis B | Observations

(some) Interesting Files

/var/log/apt/history.log

Installed packages that are not part of the base system

```
Start-Date: 2016-01-25  11:18:05
Commandline: /usr/bin/apt-get install rsync
Install: rsync:armhf (3.1.0-2ubuntu0.2)
End-Date: 2016-01-25  11:18:11

Start-Date: 2016-04-05  12:30:59
Commandline: /usr/bin/apt-get install ccrypt
Install: ccrypt:armhf (1.10-4)
End-Date: 2016-04-05  12:31:01

Start-Date: 2016-04-25  09:58:29
Commandline: /usr/bin/apt-get install tcpdump
Install: tcpdump:armhf (4.5.1-2ubuntu1.2), libpcap0.8:armhf (1.5.3-2, automatic)
End-Date: 2016-04-25  09:58:33
```

- Why does a vacuum cleaner need `rsync` or `tcpdump`?

Thesis B - Investigating `rrlogd`

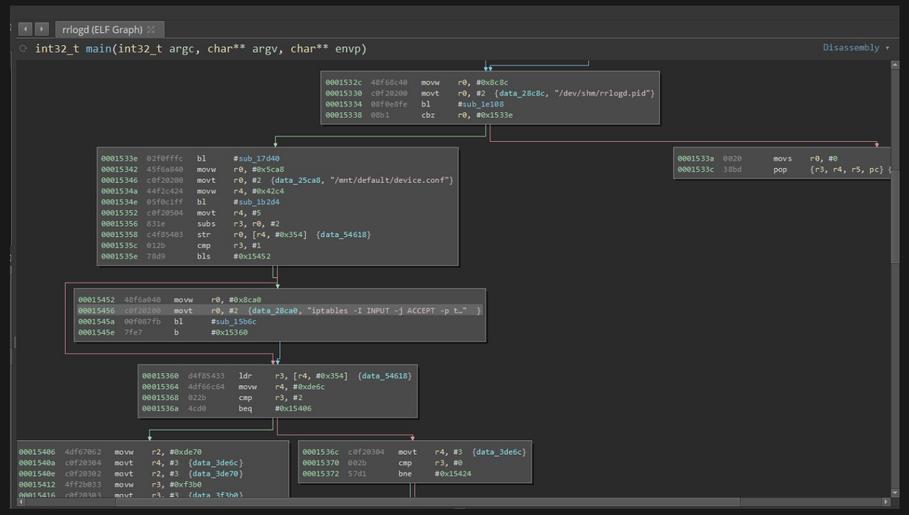
(some) Interesting Files

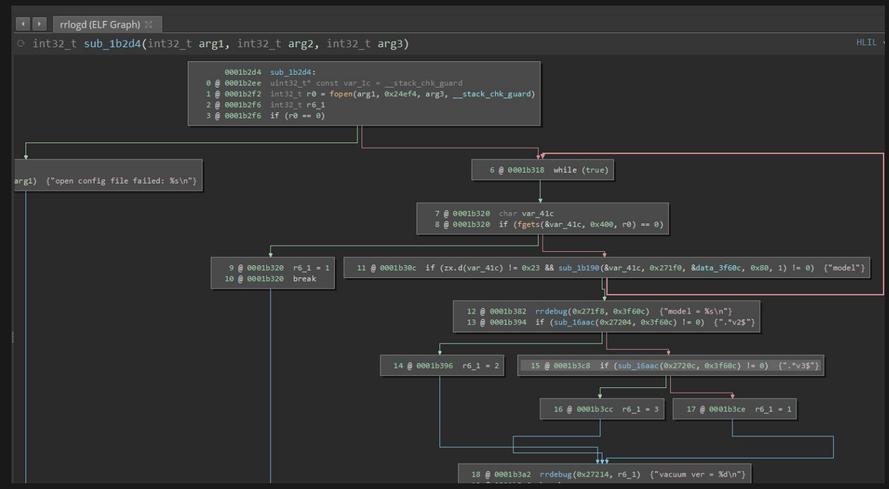
`mmcblk0p8/opt/rockrobo/rrlog/rrlogd`

Logs are encrypted at rest (after being packed)

Originally used to be a symmetric key, now using a public key

😕 Logging program has the functionality to unblock port 22?





(some) Interesting Files

mmcblk0p7/usr/bin/adbd

- Custom ADB binary
- Had a brief look (more)

```
locksec_init_key: can not find the prefix str from adb conf file, use default
locksec_init_key: can not find the suffix str from adb conf file, use default
locksec_init_serial: adb read 465 bytes from /proc/cpuinfo
locksec_init_key: locksec_init_key, rockrobo%()+-[]_8a80ab8936d76c118000:;<=>?@{}rubyde
locksec_apply_key: locksec_apply_key, erI09cyW%()+-[]_8a80ab8936d76c118000:;<=>?@{}CzD2
locksec_apply_passwd: adb source str: erI09cyW%()+-[]_8a80ab8936d76c118000:;<=>?@{}CzD2
locksec_apply_passwd: locksec_apply_passwd, passwd: 0y[ad8@w
```

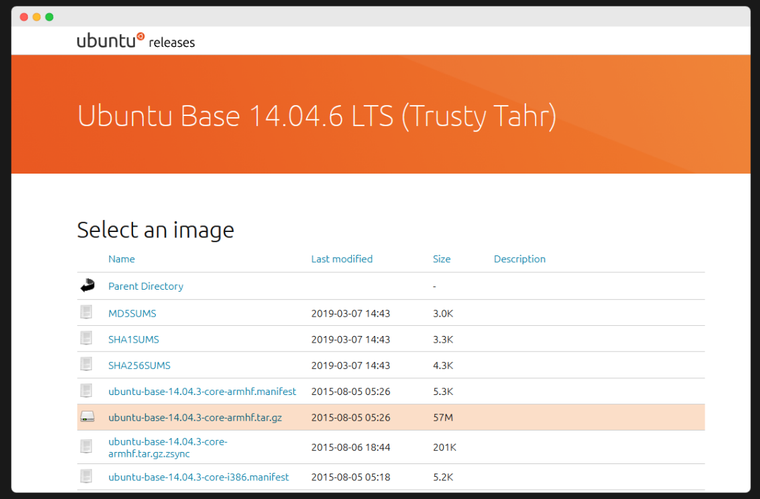
Related files

- mmcblk0p6/vinda
- mmcblk0p6/adb.conf
- mmcblk0p8/var/log/upstart/adbd.log

Thesis B | Future Challenges

Current Challenges

File Inspection Approach 2 - Binary Comparisons



Compare executable files and find differences in binary function

bindiff, binwalk, ssdeep, sdhash

As seen in A Large-Scale Analysis of the Security of Embedded Firmwares - Andrei C, Jonas Z, Aur'elien F, Davide B

Current Challenges

Intercepting encrypted data / TLS traffic

- Ubuntu 14.04 has some issues (?)
  - `PolarProxy` is too new (libc requirements)
  - apt update doesn’t work with socks5:// or http proxies properly???
- Routing?
- Hook into the encryption/decryption process somehow?
  - Use `Frida`?
  - Or look at the data communicated by the smartphone app?
    - `Objection` tool didn’t work with the RoboRock app

Thesis B Completion

- Analysis of firmware binaries to identify vulnerabilities
  - Still in progress
- Search for unsecured secrets, logs, configurations
  - Completed (excluding encrypted `rrlog` files)

Revised Thesis C Plan

- WAN - security, PII
- LAN - traffic? (tcpdump)
- Firmware upgrade
- Update to a newer firmware version and look at changes
- Check what files gets cleared during a format
- Binary assessment
- Verify IPv6 SSH access

The IPv6 World

Incoming Timeline

- 22T2 W1 - IPv6 SSH verification, continue binary assessment
- 22T2 W2 - WAN traffic analysis
  - Look at network behaviour
  - Try view WAN data pre-encryption / post-decryption
- 22T2 W4 - Update to latest version (and hope we don't get locked out)
  - Do another vacuum clean, reimage, compare binaries
- 22T2 W5 - Factory reset device, check for remnant files
- 22T2 W8 - Demo submission
- 22T2 W11 - Report submission

Let's Talk Threats

Talk about threat models

What files?

- compare against stock
- lots of hash matches

threat models

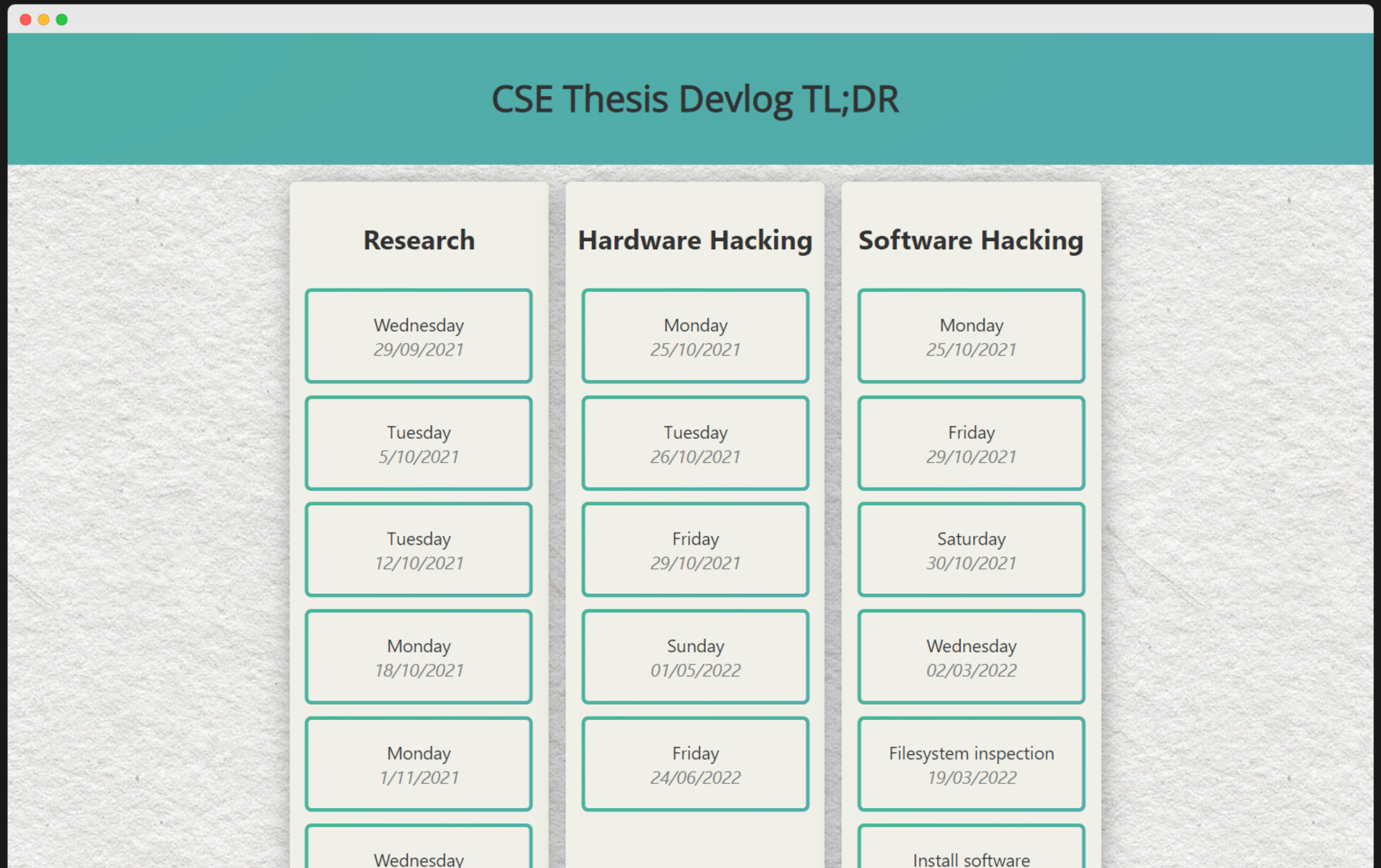
work done in thesis c

hooking into code..

Network analysis

Address the statement

Thesis in a Year



Thank You

Andrew Wong

w: featherbear.cc/UNSW-CSE-Thesis

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