Mở đầu

Gần đây mình có làm một project nghiên cứu nho nhỏ với đề tài ngắn gọi lại là: Hack 3 con camera. Thế nào lại báo cáo đúng thời điểm ca sĩ nào đó bị lộ hàng nóng vì camera bị hack (͡° ͜ʖ ͡°). Thấy anh em cùng phòng hỏi quá nên mình viết một bài “demystify” lỗi trên mấy con cam cùi vậy.

Target 1.

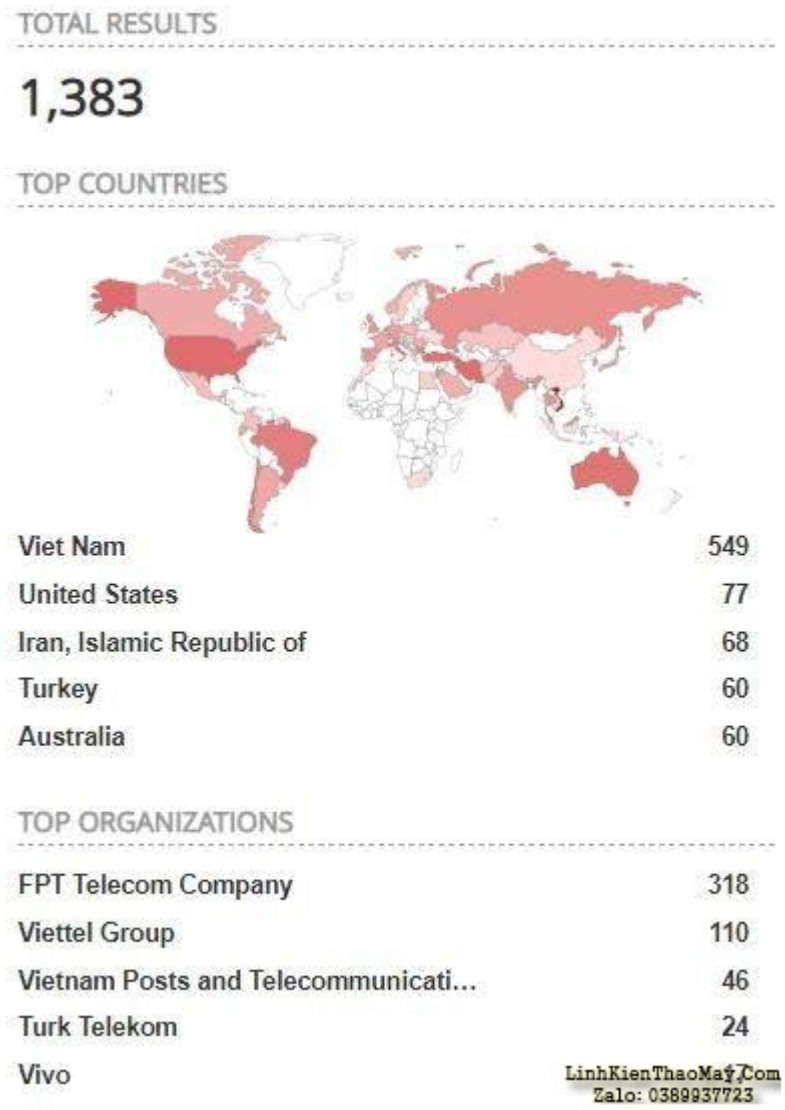
Target đầu tiên là một con camera cây nhà lá vườn được mình “trưng dụng”. Nhớ lại thì ngày trước mua vài con gắn tại cửa hàng của gia đình. Trên camera không có bất cứ thông tin gì từ nhà sản xuất ngoại trừ credential mặc định.

Recon.

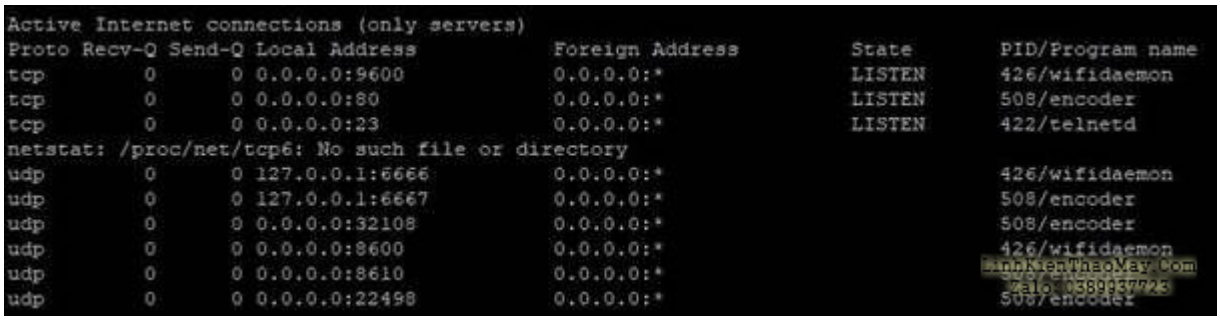
Nmap scan thấy 3 port tcp: 23 – telnet, 80 – http, 9600 (?).

Mình sử dụng 1 wordlist các password hay dùng của mấy con camera Trung Quốc bruteforce thử telnet. Ra credential là root – 123456.

Do UPnP được tự động bật, Shodan scan thấy nhiều thiết bị:



Confirm các port mở:



Tới đây thì mình tìm cách lấy binary trên thiết bị, do busybox trên thiết bị hỗ trợ tftp nên cách dễ nhất là setup server gửi file về.

Phân tích binary encoder.

```
                                                $ file encoder
encoder: ELF 32-bit LSB executable, ARM, EABI5 version 1 (SYSV), dynamically linked, i
nterpreter /lib/ld-, stripped
                                                $ checksec ./encoder
[*] '                                               encoder'
    Arch:     arm-32-little
    RELRO:    No RELRO
    Stack:    No canary found
    NX:       NX disabled
    PIE:      No PIE (0x8000)
    RWX:      Has RWX segments
```


Binary không sử dụng các cơ chế bảo vệ nào, nếu tìm được stack overflow khả năng cao sẽ lên được shell.

Tuy nhiên với target này không cần đến stack overflow, mình tìm thấy kha khá lỗi cho phép sử dụng command injection với data từ web interface.

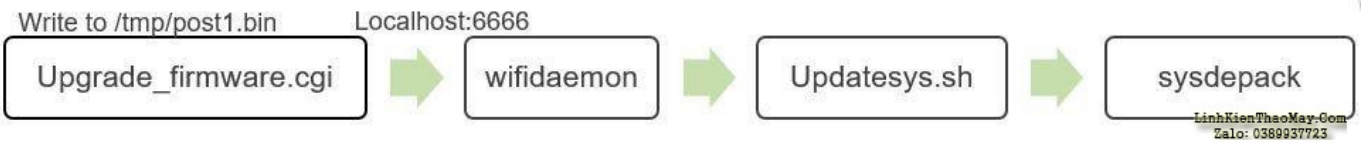
Chức năng update binary.

Thiết bị sử dụng GoAhead webserver với chức năng riêng được implement qua CGI module. Các CGI module được compile trực tiếp vào binary, access bằng các link có đuôi .cgi và xác thực sử dụng username và password admin trong request.

Riêng với chức năng update binary, update html, camera control và decoder control. Xác thực CGI bị bỏ qua.

```
v11 = strstr(v4, "upgrade_htmls.cgi");        // these cgis do not need auth
if ( strstr(v4, "upgrade_firmware.cgi") )
  cgis_val = 2;
else
  cgis_val = v11 != 0;
if ( strstr(v4, "decoder_control.cgi") )
  cgis_val = 3;
if ( strstr(v4, "camera_control.cgi") )
{
  sub_158EC(v5);
  cgis_val = 4;
}
if ( (unsigned __int8)cgis_val | (unsigned __int8)auth_level )
{
ABEL_16:                                        // CGI pointers
  v13 = ((int (__fastcall *)(char *, char *, signed int))dword_DD6E8[11 * v9 + 9])(&s, v4,
```


Firmware file được xử lý bằng nhiều binary:



RE một lúc thì có được cấu trúc update file info header.

```
firm_header_struct struc ; (sizeof=0x8C, mappedto_34)
                                        ; XREF: sub_8CAC/r
dirname         DCB 64 dup(?)
filename        DCB 64 dup(?)
size            DCD ?                   ; XREF: sub_8CAC+E8/r
                                        ; sub_8CAC+1BC/r ...
version         DCD ?                   ; XREF: sub_8CAC+FC/r
factory         DCD ?                   ; XREF: sub_8CAC+110/r
firm_header_struct ends
```


Trong đoạn code parse update header tại sysdepack có lỗi command injection thực thi lệnh qua system.

```
printf("filename:%s\n", s.filename);
printf("size:%d\n", s.size);
printf("version:%d\n", s.version);
printf("factory:%d\n", LOBYTE(s.factory));
v2 = strlen(s.filename);
printf("filename len %d\n", v2);
if ( !s.filename[0] )
{
  puts("file name is null");
  v8 = 0;
  break;
}
memset(&command, 0, 0x80u);
sprintf(&command, "mkdir -p /%s", &s);    // mkdir w dirname + filename if not null terminated
printf("cmd:%s\n", &command);
system(&command);
```


– Chỉ cần set dirname + file name = lệnh là đã có 128 byte buffer cho command. Set size = -1 để thoát update function mà không ghi đè firmware.

```
v6 = check_size(s.size);
if ( v6 )
{
  printf("disk isn't space and update failed filename:%s\n", s.filename);
  v8 = -1;
  break;
}
```


Có thể sử dụng lỗi này để thực hiện unauth RCE. Tuy nhiện thiết bị reboot ngay sau khi chạy script update nên với lỗi này không làm gì được nhiều.

Chức năng mail và ftp.

Mình cũng thấy các lỗi injection tương tự trong chức năng gửi mail và ftp. Tuy nhiên cần cung cấp credential để dùng 2 chức năng cgi này.

Test script được tạo trong chức năng testftp sau đó được thực thi:

```
v4 = fopen("/tmp/ftpupload.sh", "wb");
if ( v4 )
{
  memset(&s, 0, 0x80u);
  strcpy((char *)&s, "/bin/ftp -n<<!\n");
  v5 = strlen((const char *)&s);
  fwrite(&s, 1u, v5, v4);
  memset(&s, 0, 0x80u);
  sprintf((char *)&s, "open %s %d\n", &unk_16C218, word_16C32C);
  v6 = strlen((const char *)&s);
  fwrite(&s, 1u, v6, v4);
  memset(&s, 0, 0x80u);
  sprintf((char *)&s, "user %s %s\n", &dword_16C258, &dword_16C278);
  v7 = strlen((const char *)&s);
  fwrite(&s, 1u, v7, v4);
  memset(&s, 0, 0x80u);
  strcpy((char *)&s, "binary\n");
  v8 = strlen((const char *)&s);
  fwrite(&s, 1u, v8, v4);
```


Chèn lệnh trong chức năng testmail qua địa chỉ mail trong biến v1, lần này câu lệnh được gọi bằng popen:

```
v1 = a1;
v2 = call_popen("killall mailx");
if ( sub_45514(v2) )
  return -1;
memset(&s, 0, 0x80u);
sprintf(&s, "echo \"mail test ok\" | mailx -v -s \"mail
call_popen(&s);
```


Leak info.

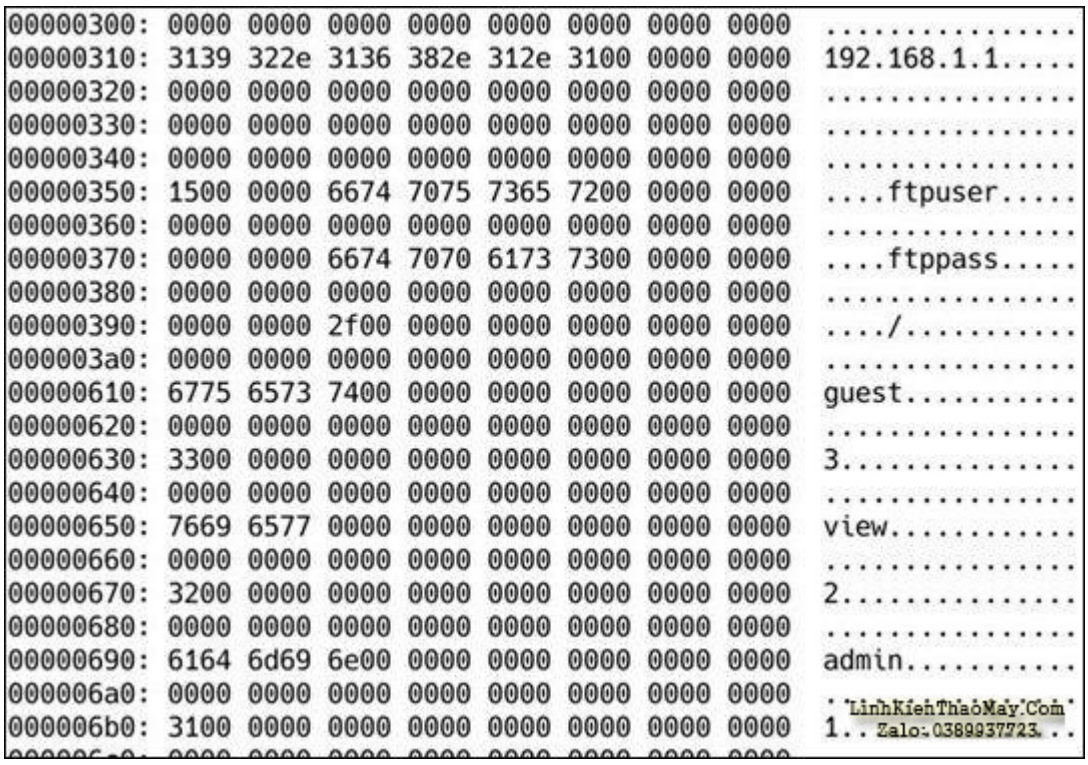
Đối với các request file không có đuôi .cgi. Đoạn code xử lý authentication tóm gon như sau:

```
if(!strstr(request, "loginuse") | !strstr(request, "loginpas")){
    if(!strstr(request, "user") | !strstr(request, "pwd")){
        //DO ACCESS CONTROL
    }
}
return 0;
```

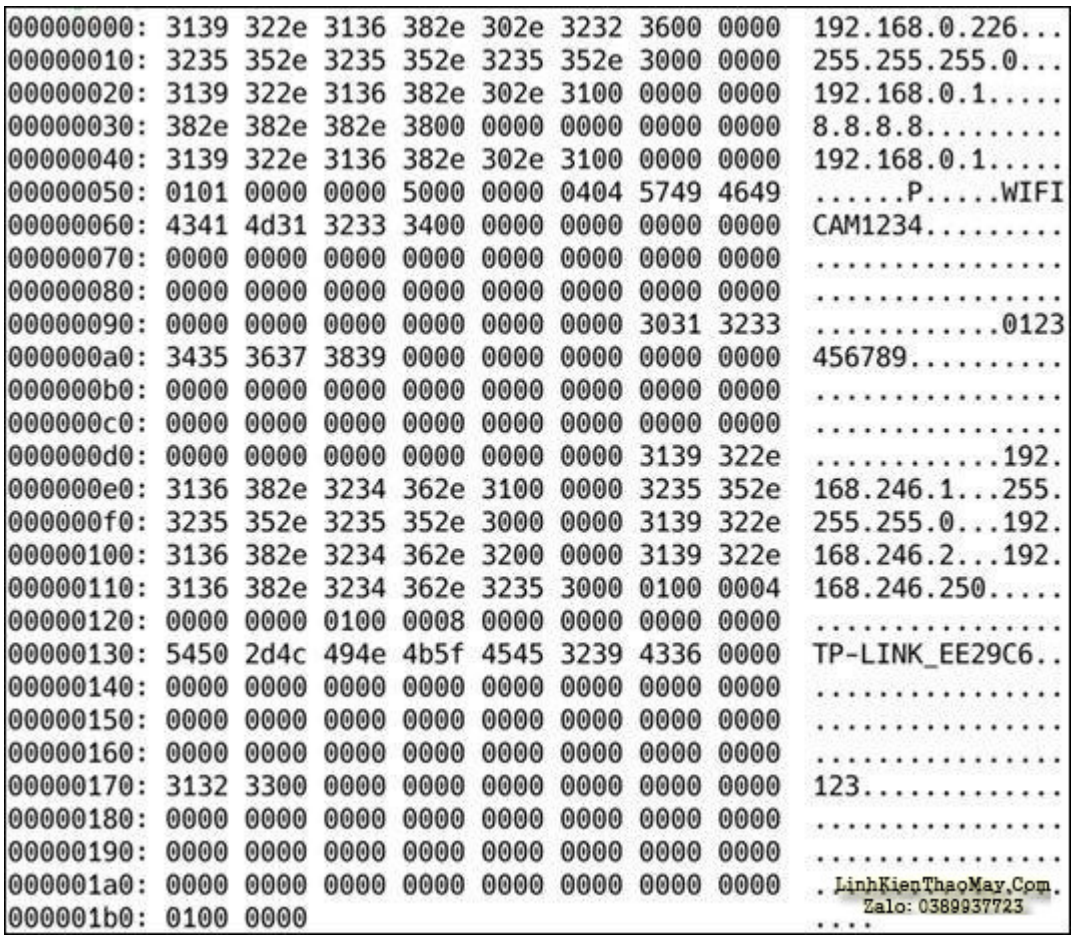

Nếu trong request tồn tại cặp “loginuse” + “loginpas” hoặc “user” + “pwd” (không cần valid). Phần xác thực hoàn toàn bị bỏ qua.

2 file đáng chú ý là:

– system.ini – chứa config camera như **các tài khoản quản trị, tài khoản mail, ftp**



– network.ini – chứa ip, DNS config, **tài khoản wifi**

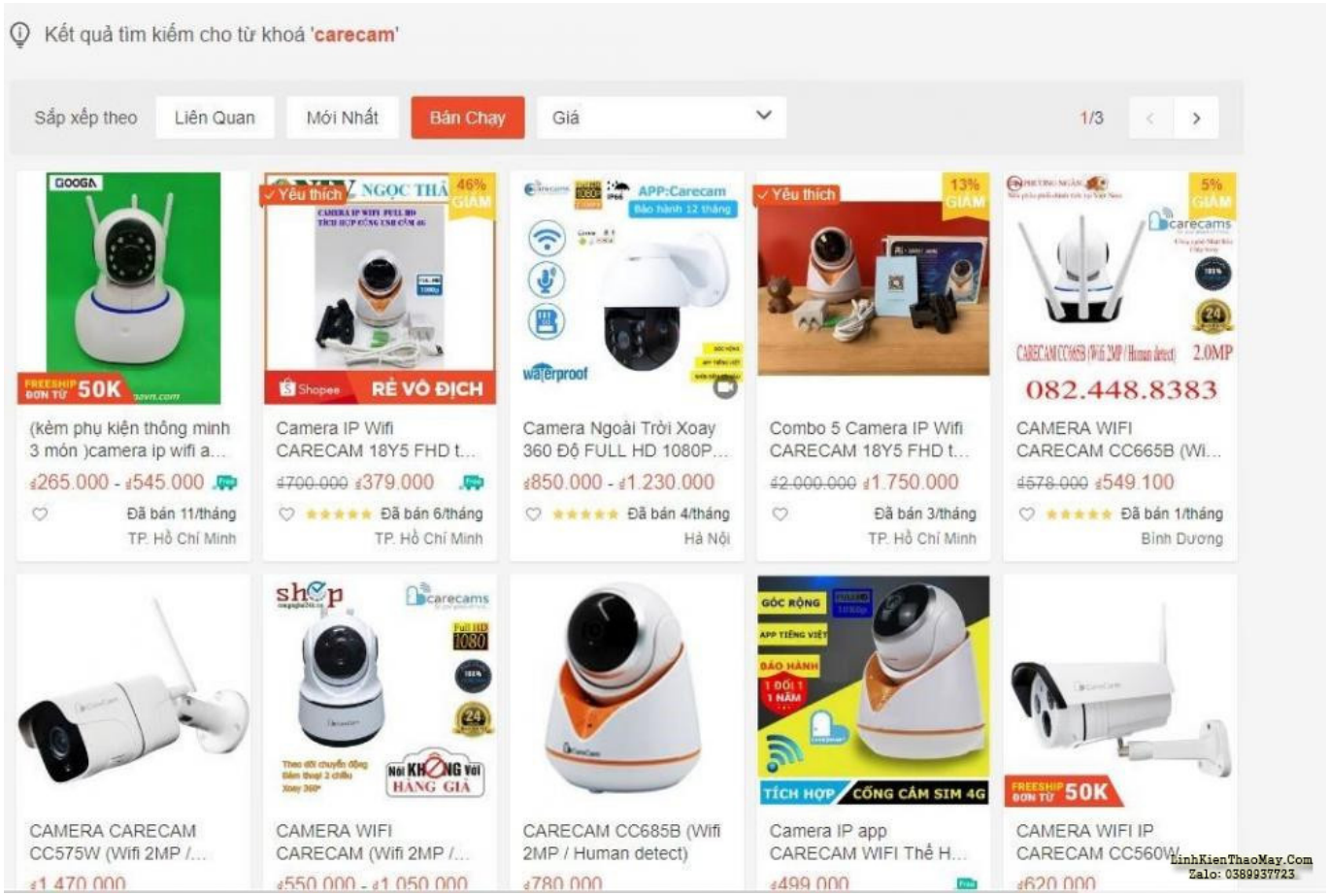


Kết hợp lỗi này lấy tài khoản admin với lỗi chèn lệnh trong chức năng mail hoặc ftp => reliable unauthenticated RCE.

Target 2.

Recon.

Con camera này minh mua trên shopee có tên carecam.



Thiết bị sử dụng app cùng tên để điều khiển. Search một hồi thì ra trang chủ: www.smartcloudcon.com

Port scan ra một loạt các port, từ telnet, http, https tới các port lạ như 843, 1234, 1300...

Do con cam này thiếu dấu hiệu đặc trưng nên khó scan. Mình dùng zmap scan dải của VN thấy 58 thiết bị. nhưng false positive cũng khá nhiều.

```
vmware@ubuntu:~/F_Drive$ wc ./1300_8699_6688_80.txt
 58  58 832 ./1300_8699_6688_80.txt
```

Như con cam trước mình thử tìm đường vào qua telnet => không bruteforce được credential.

Hàn dây vào chân serial của thiết bị, mình lấy được firmware qua uboot, cách lây theo phương pháp này bạn đọc có thể tham khảo tại https://tradahacking.vn/cam9-ph%C3%ADa-sau-m%E1%BB%99t-c%C3%B4-g%C3%A1i-90e5192554ec

Extract qua binwalk. Mình thấy password telnet mặc định trong file shadow trên thiết bị là root – 123. Nhưng sau khi bootscript chạy. File /etc/shadow bị ghi đè với file /app/shadow – root:z1YC93pV6OlQI:17771:0:99999:7:::.

Sửa lại firmware xóa file này đi => cuối cùng cũng có shell.

List các binary đang mở cổng trên camera:

```
Active Internet connections (only servers)
Proto Recv-Q Send-Q Local Address           Foreign Address         State       PID/Program name
tcp        0      0 0.0.0.0:6688            0.0.0.0:*               LISTEN      779/apollo
tcp        0      0 0.0.0.0:6980            0.0.0.0:*               LISTEN      780/cld_upd
tcp        0      0 0.0.0.0:7530            0.0.0.0:*               LISTEN      779/apollo
tcp        0      0 0.0.0.0:8554            0.0.0.0:*               LISTEN      779/apollo
tcp        0      0 0.0.0.0:843             0.0.0.0:*               LISTEN      781/noodles
tcp        0      0 0.0.0.0:80              0.0.0.0:*               LISTEN      779/apollo
tcp        0      0 0.0.0.0:1234            0.0.0.0:*               LISTEN      779/apollo
tcp        0      0 0.0.0.0:1300            0.0.0.0:*               LISTEN      781/noodles
tcp        0      0 0.0.0.0:23              0.0.0.0:*               LISTEN      871/inetd
tcp        0      0 0.0.0.0:443             0.0.0.0:*               LISTEN      779/apollo
tcp        0      0 0.0.0.0:8699            0.0.0.0:*               LISTEN      779/apollo
udp        0      0 0.0.0.0:57095           0.0.0.0:*                           779/apollo
udp        0      0 0.0.0.0:8000            0.0.0.0:*                           779/apollo
udp        0      0 0.0.0.0:8002            0.0.0.0:*                           779/apollo
udp        0      0 0.0.0.0:3702            0.0.0.0:*                           779/apollo
udp        0      0 0.0.0.0:6785            0.0.0.0:*                           780/cld_upd
udp        0      0 0.0.0.0:5012            0.0.0.0:*                           781/noodles
udp        0      0 0.0.0.0:5564            0.0.0.0:*                           779/apollo
```


Mình search thông tin về các binary thì tìm được series này của các bạn bên VNPT sec: https://sec.vnpt.vn/2018/12/tan-man-ve-1-chiec-ip-camera-nao-do/

Hóa ra con cam này với loại camera liveyes clone từ 1 source ra. Tới đây thì confirm lại bug thôi.

RCE as a feature

Port 1300 được binary noodles sử dụng, Chức năng chính của binary này là handle download, upload file cũng như thực hiện update firmware, cụ thể:

```
switch ( tag_mode )
{
  case 1:
    sub_A854();
    SUB_UPGRADE(client_fd_0, buffer);
    sub_A840();
    break;
  case 2:
    SUB_BURNMAC(client_fd_0, buffer);
    break;
  case 3:
    SUB_ELFEXEC(client_fd_0, buffer);
    break;
  case 4:
    SUB_SYSTEM(client_fd_0, (int)buffer);
    break;
  case 5:
    SUB_SYSTEMEX(client_fd_0, (int)buffer);
    break;
  case 6:
    SUB_DOWNLOAD(client_fd_0, buffer);
    break;
  case 7:
    SUB_UPLOAD(client_fd_0, buffer);
    break;
  case 8:
    SUB_FLASHDUMP(client_fd_0, buffer);
    break;
  case 9:
    SUB_BURNSN(client_fd_0, buffer);
    break;
  default:
    puts("Not supported cmd!");
    break;
}
```


Request gửi tới port này được parse theo tag để nhận thông tin file và chọn function xử lý cụ thể theo các case đã nêu, sau đó mới nhận nội dung file nếu có.

Trong các case xử lý có 3 chức năng khá hay là ELFEXEC, SYSTEM, SYSTEMEX trong đó:

– ELFEXEC nhận binary, script,… từ user và thực thi. Quá trình thực thi qua hai bước:

o Nhận thông tin file qua data gửi lên port 1300 dưới format ***'<ELFEXEC><FNAME>' + filename + '</FNAME><FSIZE>' + filesize + '</FSIZE></ELFEXEC>'***

o Nhận file data, ghi ra file và thực thi.

– Chức năng SYSTEM và SYSTEMEX cũng hoạt động tương tự, nhưng nhận lệnh thay vì file.

– Để sử dụng 3 chức năng này không cần qua bất cứ xác thực nào.

Các lỗi tràn bộ đệm và command injection.

Trong chức năng UPGRADE:

Trong đoạn code xử lý của chức năng "UPGRADE", Buffer 's' có độ dài tối đa 0x1cc byte có

thể bị tràn khi hàm sub_F208 sao chép mọi dữ liệu từ giữa hai nhãn <METHOD></METHOD> mà không kiểm tra độ dài, khi request gửi tới có thể dài tới 0x400 byte.

```
v5 = strstr(request_buf_0, "<METHOD>");
if ( v5 )
{
  v6 = strstr(request_buf_0, "</METHOD>");
  if ( v6 )
  {
    memset(&s, 0, 0x100u);
    sub_F208(&s, v5 + 8, v6 - (v5 + 8));      // copy all inside method tag, request_buf can be 0x400 max, s is 0x1cc above $bp
    if ( !strcmp(&s, "all") )
    {
      sub_D254("/home/uboot_up_enable");
      sub_D254("/home/data_up_enable");
      sub_D254("/home/force");
    }
  }
}
```


Trong hàm sub_9DE4, là hàm thực thi download dữ liệu theo yêu cầu và ghi ra file. Tồn tại cả hai lỗi chèn lệnh và tràn bộ đệm do sử dụng hàm sprintf với chuỗi lệnh thực thi nhưng không kiểm tra xác thực và độ dài.

– Đoạn code dưới đây là đoạn code bị lỗi khi “file_name_0” có độ dài tối đa 0x110 (là dữ liệu người dùng gửi tới chức năng qua thẻ <FNAME></FNAME>) được sprintf vào câu lệnh điều khiển tại biến “v25” có độ dài tối đa là 0xc8, lỗi tràn bộ đệm xảy ra và có thể chiếm được con trỏ lệnh, trước đó câu lệnh tại “v25” được thực thi cho phép thực hiện lỗi command injection.

```
if ( CHECKSUM_ptr )
{
  sprintf((char *)&v25, "md5 %s", file_name_0);// bof, filenamemax 0x110, v25 to bp 0xc8
  PROC_execcmd((char *)&v25, (char *)&v23, 256);
  if ( sub_F3D8(CHECKSUM_buf, &v23) )
  {
    fprintf((FILE *)stderr, "md5-org: %s %s md5-cal: %s\n", CHECKSUM_buf, "!=", &v23);
    fwrite("checksum err\n", 1u, 0xDu, (FILE *)stderr);
    file_fd = 0;
    goto LABEL_7;
  }
  fprintf((FILE *)stderr, "md5-org: %s %s md5-cal: %s\n", CHECKSUM_buf, "==", &v23);
}
```


Trong chức năng UPLOAD:

Cũng tương tự lỗi trên tại hàm sub_9B5C được thực thi trong chức năng “UPLOAD”:

– Đoạn code dưới đây sprintf dữ liệu được gửi đến vào biến chứa lệnh ‘s’, sau đó thực thi, dẫn tới khả năng bị command injection.

– Biến ‘s’ có thể chứa tối đa 0xA0 bytes, hàm sprint sử dụng tham biến tag của hàm sub_9B5C lấy trực tiếp dữ liệu từ thẻ <LOCALNAME ></LOCALNAME> có độ dài tối đa 0x100 bytes dẫn đến overflow.

```
signed int __fastcall sub_9B5C(int fd, char *buff, const char *tag)
{
  int fd_0; // r8
  const char *tag_0; // r5
  FILE *v5; // r0
  FILE *v6; // r10
  int v7; // r7
  int v8; // r0
  int v9; // r4
  int v10; // r5
  int v11; // r0
  int v12; // r4
  int v14; // [sp+0h] [bp-5A0h]
  char v15; // [sp+400h] [bp-1A0h]
  char s; // [sp+500h] [bp-A0h]

  fd_0 = fd;
  tag_0 = tag;
  sprintf(&s, "md5 %s");                    // arg tag is sprintfed to s
  PROC_execcmd(&s, &v15, 256);
```


Tuy có thể khai thác bằng buffer overflow. Command injection vẫn là cách hay nhất để khai thác do không gây crash service. Chỉ cần gửi command theo format ***<UPLOAD><LOCALNAME> cmd </UPLOAD></LOCALNAME>*** tới port 1300 là có thể thực thi lệnh trên thiết bị.

Tạm kết

Trong quá trình thực hiện project mình còn làm một con camera khác (yoosee). Nội dung blog cũng khá lan man rồi nên mình tạm dùng bài viết này ở đây. Hẹn các bạn trong bài viết sau (maybe :>)

Các bài viết tương tự:

1. đầu DVD california MP 180 – em cho đĩa vào xong nó bị dắt, không quay. lúc open thì nó chỉ ra chỗ lắp đĩa còn đĩa bị kẹt ở trong k ra. em hì hục mãi lôi dc đĩa kẹt ra.lắp thử đĩa khác vào thì lúc đầu nó kêu to to xong cũng chạy đĩa.nhưng thử mấy cái nữa thì không đọc.cho đĩa vào nó báo không có đĩa.
2. đĩa cài win tự động. 40 nghìn – đĩa sẽ tự động cài win, cài phần mềm cơ bản, cài và nhận đầy đủ driver, dùng cho cả laptop và pc
3. lcd acer v173 – khi e cắm nguồn vào thì vẫn hiện logo,nhưng màn hình chỉ hiện thị khoảng mấy giây rồi tắt,khi rút cáp tín hiệu thì màn hình lại hiên thị không có tín hiệu được kết nối,e không kết nối cáp tín hiệu thì để cả tiếng không vấn đề gì và vẫn hiện thị không có thiết bị đươc kết nối
4. may giat electrolux EWF549 – máy giặt electrolux 5,5kg chỉ có 2 nút ấn là start và nút ấn chọn tốc độ và núm xoay chọn chương trình . máy cấp nước giặt được khoảng 5 đến 7 phút là mất nguôn. rút điên ra căm lại thì lại có điện và giặt được khoảng 5 đến7 phút lại mất điện . chưa thực hiện được 1 chu trình giặt- xả văt thì mất nguồn
5. máy giặt panasonic F70A6 lông đứng – + máy bật nguồn để khoảng 30s máy tự động kéo xả .nhung khi bật chạy thì lại ngát xả và cấp nuocs giạt bình thường nhung đến lân giạt thứ 2 thì lại tụ đọng kéo xả và cáp nuocs nhung khi nhấc canh của hoạc án tạm dùng sau đó bám lai thì lại haotj động bình thường
6. Máy HP 1020 – Khi in hết một tờ giấy thì kéo luôn tờ thứ 2 vào, tờ thứ 2 chỉ ra được 1 nửa thì bị kẹt giấy, tờ thứ 2 không có chữ.
7. Nhà mình có cái loa .hôm nọ cắm điện tự nhưng nổ….mở ra xem cầu chì vỡ nát….loay hoay gắn lại…kết quả lại nổ..các bác cho e hỏi khả năng bị hư. Linh kiện nào vậy….và kiểm tra ntn ạ – Nhà mình có cái loa .hôm nọ cắm điện tự nhưng nổ….mở ra xem cầu chì vỡ nát….loay hoay gắn lại…kết quả lại nổ..các bác cho e hỏi khả năng bị hư. Linh kiện nào vậy….và kiểm tra ntn ạ
8. PC cai win xp sp2 – khi dang cai thi binh thuong nhung khj copy cac file tu dia sang may xong khoi dong lai thi may tat luon.bam nut nguon khoi dong thi den chi chop 1 cai roi tat chu khong len.minh da dem dia win do cai vao may khac thi binh thuong ko bi gi ca
9. Sam sung cs 21z45ml – Khởi động nguồn cho chạy , rít cao áp , nóng sò ngang . E đã kt các tụ và diot xung quanh sò , cũng đã thay thử cao áp và sò , nhưng vẫn vậy .
10. Ti vi samsung slim cs 21z45ml – Co hình trên dưới . E đã kt và thay thử 7845 , diode đường 16v , tụ và các R sung quanh . Nhưng vẫn khôg có gì mới lạ .
11. tivi TCL model kg nhớ rỏ tại gấp quá””’tại lãnh sữa tai nhà – bên thứ cấp ””12v có 24v và 110v kg có .đèn nháy 1 nhịp rồi đi đai.e thây IC giao động 1506 và sới lên hết phân nguồn cũng kg ăn thua gì.e nap card mới đăng tin đc. e mới vào diễn đàn mong ae giúp đở e. e cám ơn ae trên diễn đàn nhiều lăm
12. toi co may in canon2900 khi ket noi may tinh thi bao co nhan USnhung khong ket noi dc voi may in va may tinh khong tim dc thiet bi B nhung khong ket noi dc voi may in va may tinh khong tim dc thiet bi – toi co may in canon2900 khi ket noi may tinh thi bao co nhan USnhung khong ket noi dc voi may in va may tinh khong tim dc thiet bi B nhung khong ket noi dc voi may in va may tinh khong tim dc thiet bi