IP카메라,
환경 구축부터 취약점 분석까지

2022. 7. 4.

발표자 소개

김현식 - Gyul

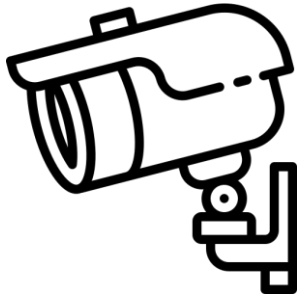
kkhhss0290@gmail.com

관심 분야 : 하드웨어, 임베디드

- BOB 9기 취약점 분석 트랙
- KVE-2022-0004, KVE-2022-0098 …

IP 카메라

IP 카메라
(Internet
Protocol)



IP 카메라, Why?

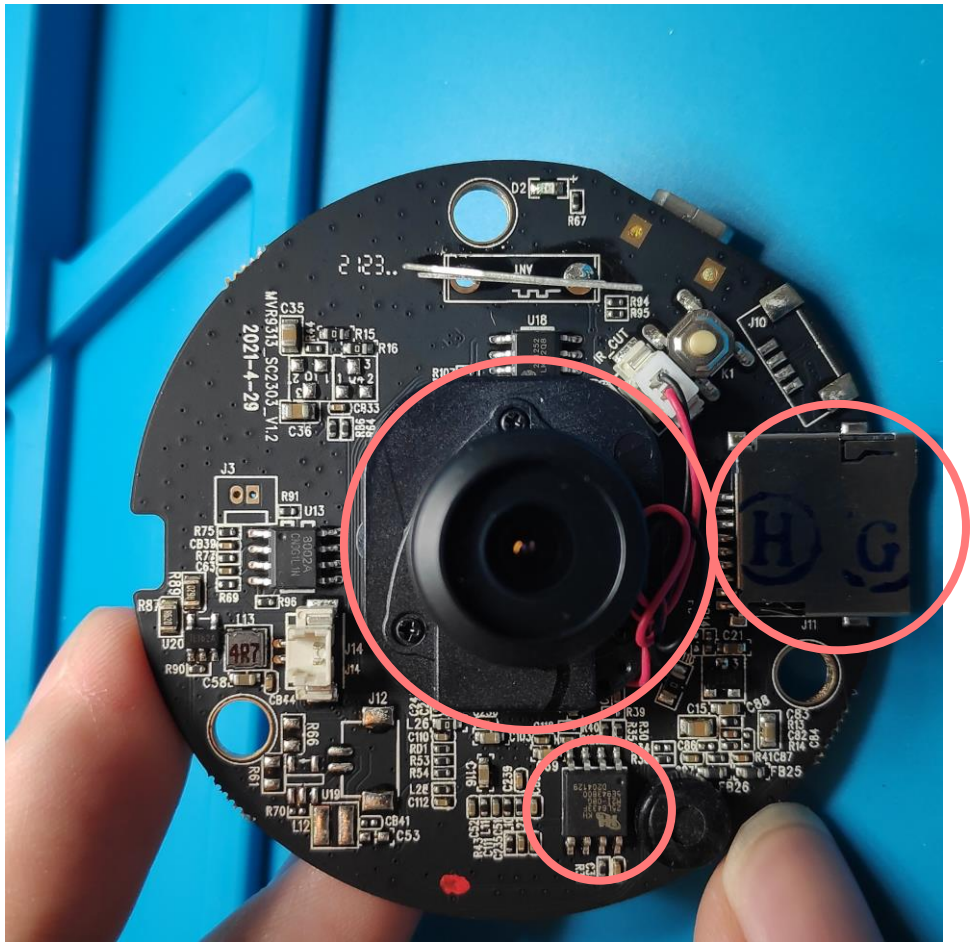
폐쇄성▲ 위험도▲

IP 카메라 구조

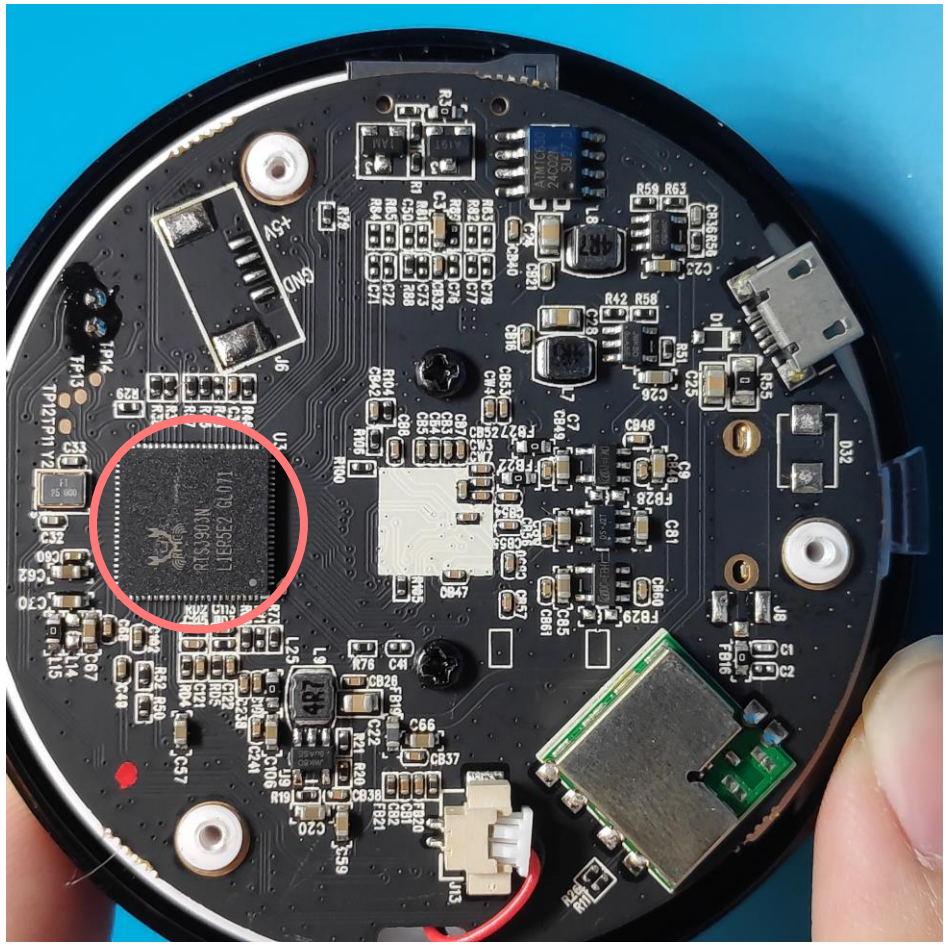
- IP 카메라 보드 구조
- IP 카메라 서비스 구조

IP 카메라 보드 구조

앞면

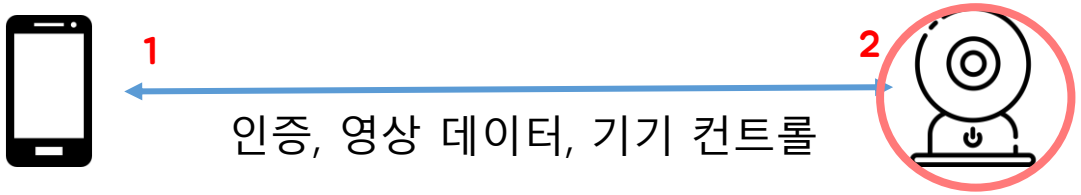


뒷면



IP 카메라 서비스 구조

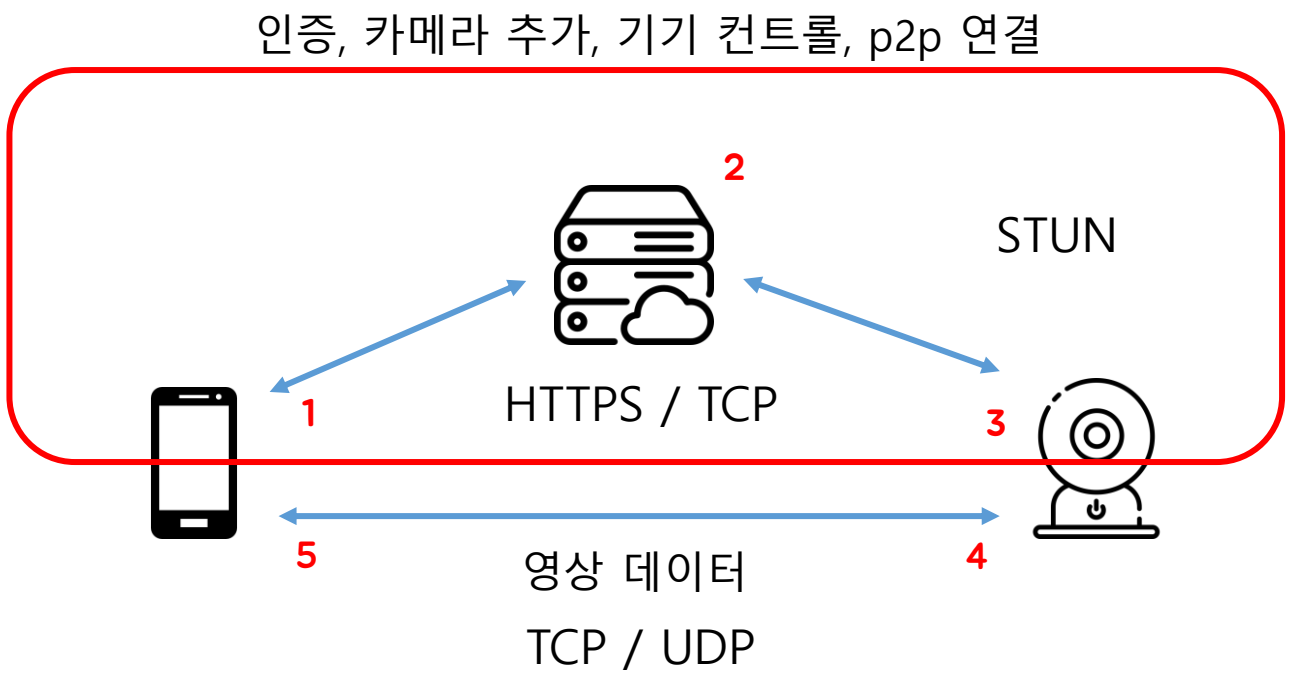
1



like

IP 카메라 서비스 구조

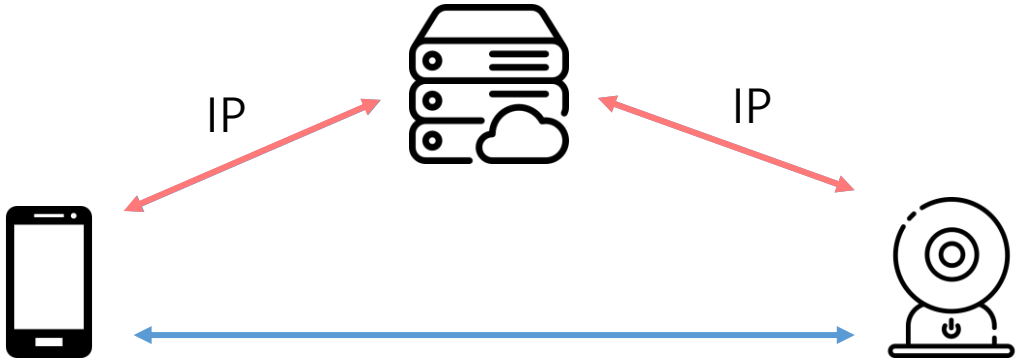
2



IP 카메라 서비스 구조

STUN 프로토콜

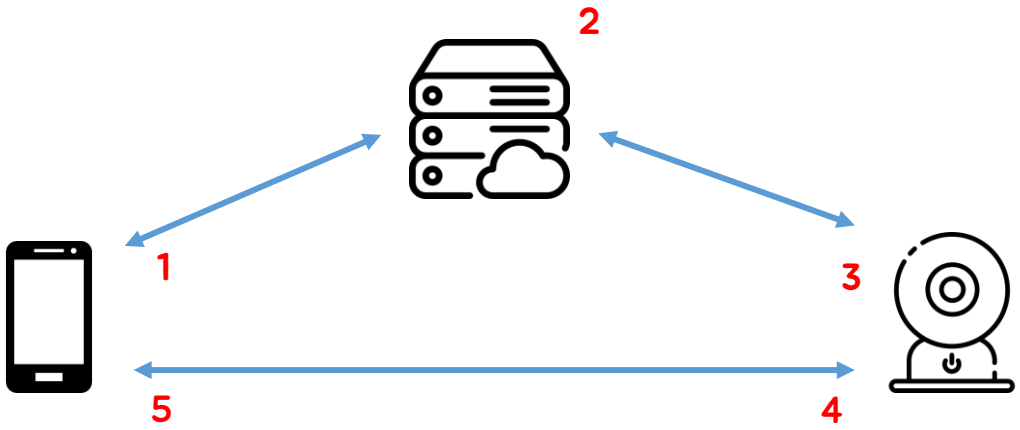
Session Traversal Utilties for NAT



IP 카메라 서비스 구조

1

2



IP 카메라 분석환경 구축

```
int32_t r0_1 = arg3 + ((arg2 + 1) << 2)
*data_98e14 = 0
*data_987e8 = arg11
*data_99e9c = r0_1
int32_t r3_2
do
    r3_2 = *r0_1
    r0_1 = r0_1 + 4
while (r3_2 != 0)
sub_23000(r0_1)
if (*data_98e14 == 0)
    int32_t r0_2 = sub_23b54()
    if (r0_2 s< 0)
        sub_d8a4(0x79740)  {"FATAL: cannot determine kern
        noreturn
    int32_t r3_4 = *data_9ada8
    int32_t r3_5
    if (r3_4 == 0)
```

Static Analysis

- 펌웨어 추출

```
R12  0x400890 ◂— xor    ebp, ebp
R13  0x7fffffffe410 ◂— 0x1
R14  0x0
R15  0x0
RBP  0x7fffffffe330 —▸ 0x400d40 ◂— push   r15
RSP  0x7fffffffe318 —▸ 0x400d30 ◂— mov    eax, 0
RIP  0x7ffff7b04360 (__read_nocancel+7) ◂— cmp    rax, -0xfff

 ► 0x7ffff7b04360 <__read_nocancel+7>     cmp    rax, -0xfff
   0x7ffff7b04366 <__read_nocancel+13>    jae    read+73 <read
    ↓
   0x7ffff7b04399 <read+73>               mov    rcx, qword pt
   0x7ffff7b043a0 <read+80>               neg    eax
   0x7ffff7b043a2 <read+82>               mov    dword ptr fs:
   0x7ffff7b043a5 <read+85>               or     rax, 0xffffff
   0x7ffff7b043a9 <read+89>               ret

   0x7ffff7b043aa                         nop    word ptr [rax
   0x7ffff7b043b0 <write>                 cmp    dword ptr [ri
   0x7ffff7b043b7 <write+7>               jne    write+25 <wri
```

Dynamic Analysis

- IP 카메라 내부 쉘 접근

- gdbserver

IP 카메라 분석환경 구축

```
int32_t r0_1 = arg3 + ((arg2 + 1) << 2)
*data_98e14 = 0
*data_987e8 = arg11
*data_99e9c = r0_1
int32_t r3_2
do
    r3_2 = *r0_1
    r0_1 = r0_1 + 4
while (r3_2 != 0)
sub_23000(r0_1)
if (*data_98e14 == 0)
    int32_t r0_2 = sub_23b54()
    if (r0_2 s< 0)
        sub_d8a4(0x79740)  {"FATAL: cannot determine ker
        noreturn
    int32_t r3_4 = *data_9ada8
    int32_t r3_5
    if (r3_4 == 0)
```

Static Analysis

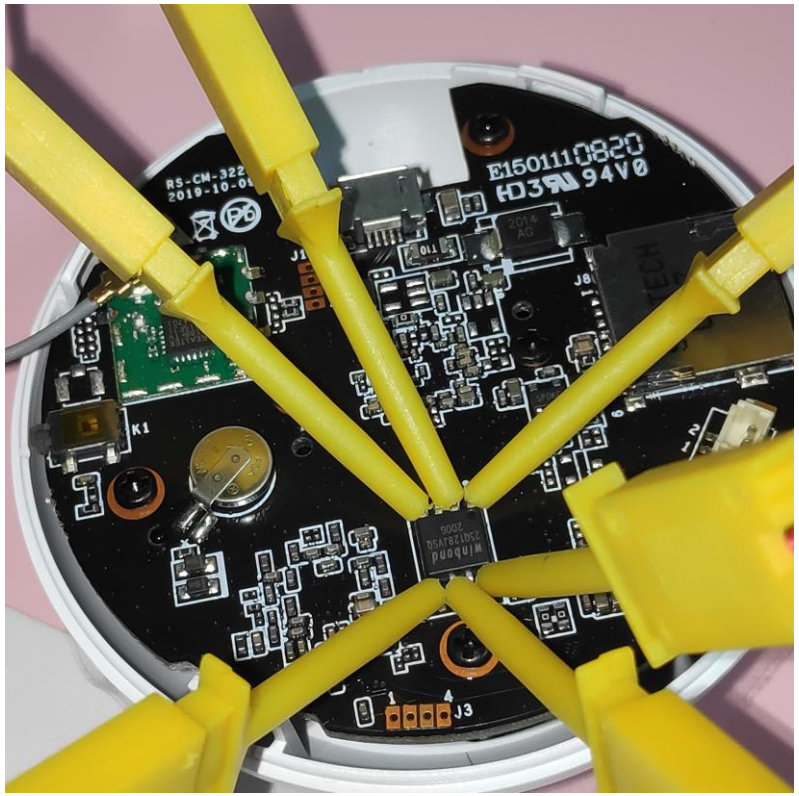
펌웨어 추출

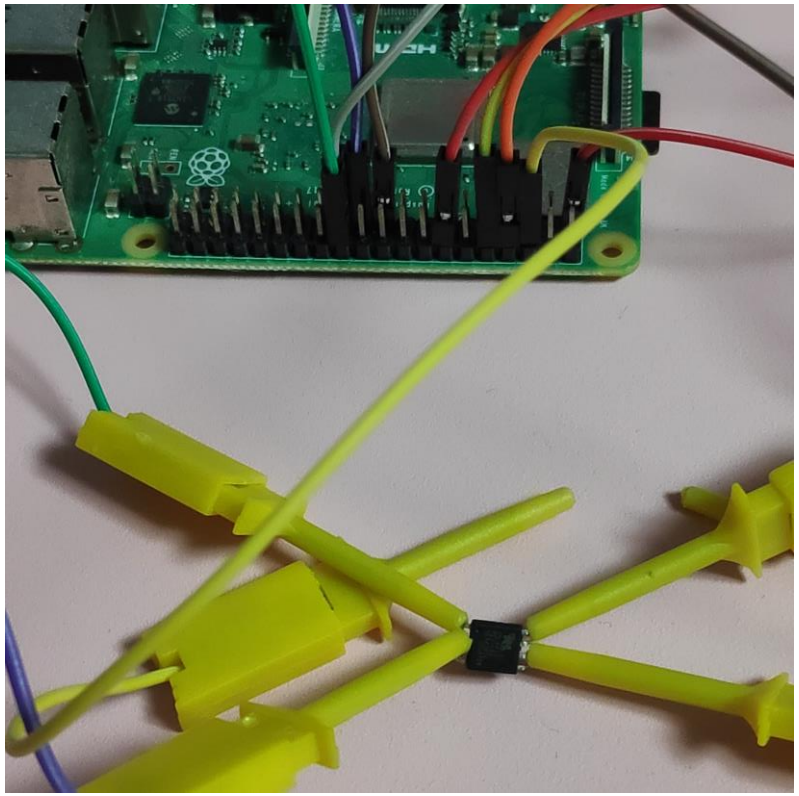
1. 라즈베리 파이와 flash 칩 연결
2. flashrom을 통해 flash 데이터 읽기
3. binwalk를 통해 파일시스템 추출

IP 카메라 분석환경 구축

펌웨어 추출

라즈베리 파이와 flash 칩 연결





IP 카메라 분석환경 구축

펌웨어 추출

3. PACKAGE TYPES AND PIN CONFIGURATIONS

W25Q64FV is offered in an 8-pin SOIC 208-mil (package code SS), an 8-pin VSOP 208-mil (package code ST), an 8-pad WSON 6x5-mm or 8x6-mm (package code ZP & ZE), an 8-pin PDIP 300-mil (package code DA), a 16-pin SOIC 300-mil (package code SF) and a 24-ball (5x5-1 or 6x4 balls) 8x6-mm TFBGA (package code TB & TC) as shown in Figure 1a-e respectively. Package diagrams and dimensions are illustrated at the end of this datasheet.

3.1 Pin Configuration SOIC / VSOP 208-mil

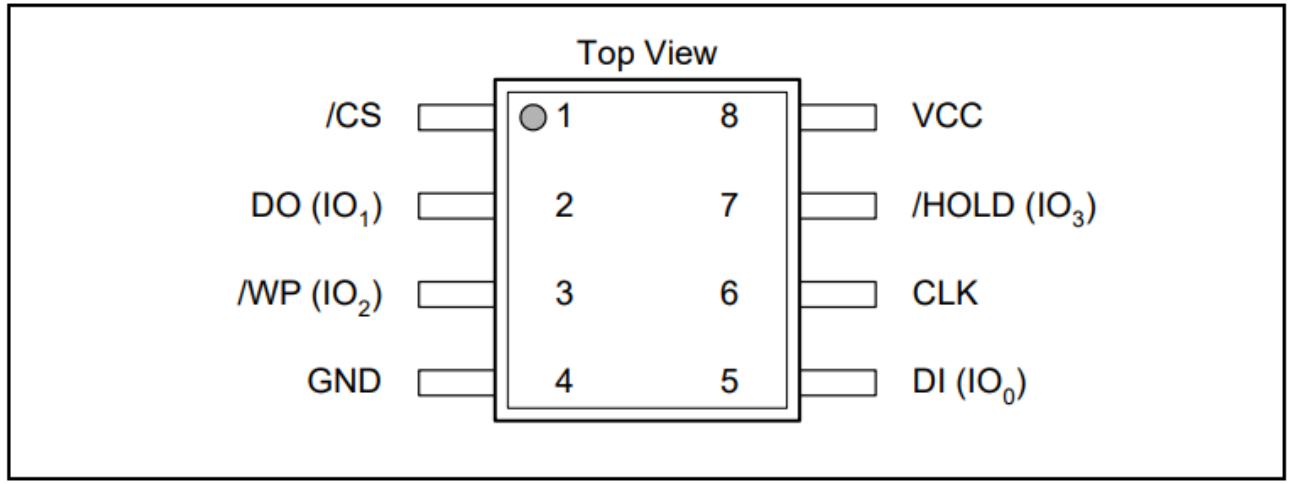
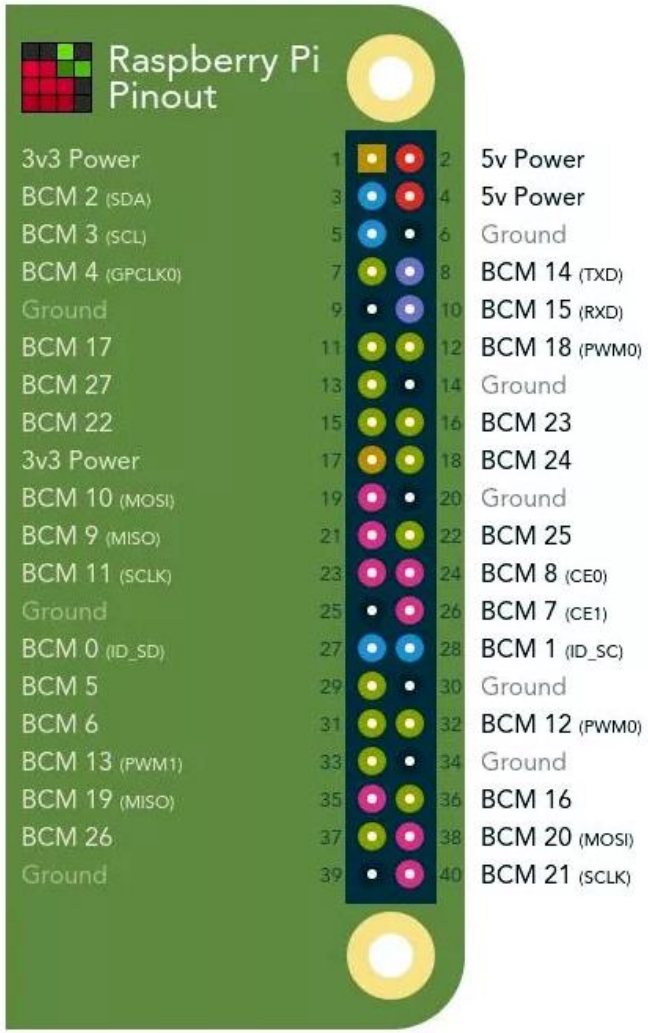


Figure 1a. W25Q64FV Pin Assignments, 8-pin SOIC / VSOP 208-mil (Package Code SS / ST)

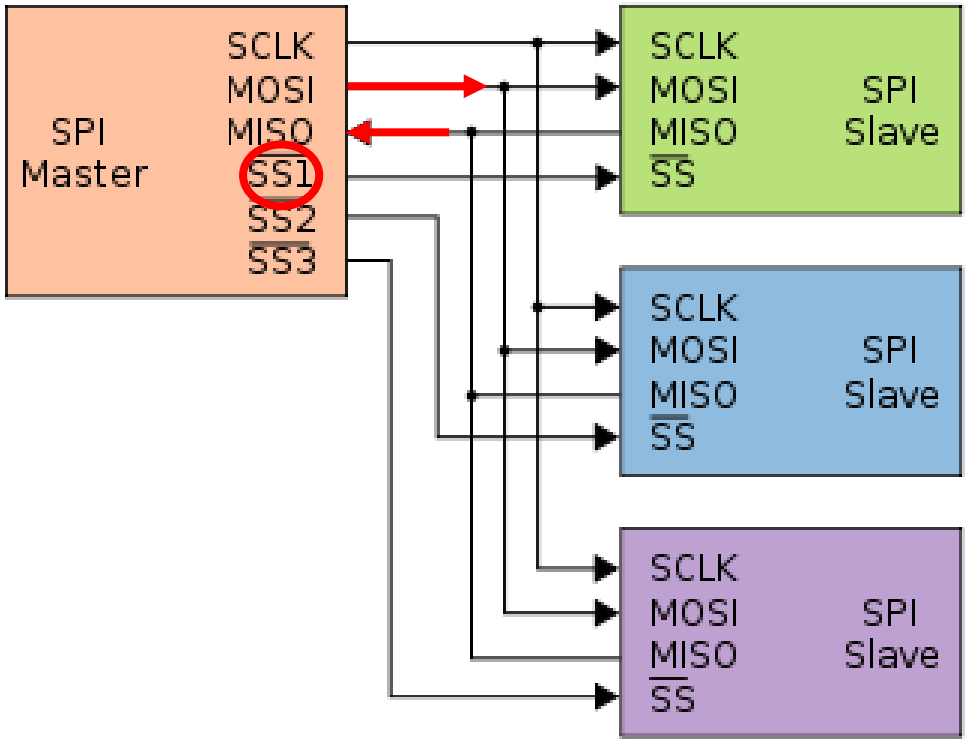


Legend
GPIO (General Purpose IO)
SPI (Serial Peripheral Interface)

IP 카메라 분석환경 구축

SPI

Serial Peripheral Interface



IP 카메라 분석환경 구축

펌웨어 추출

flashrom을 통해 flash 데이터 읽기

```
> flashrom -p linux_spi:dev=/dev/spidev0.0 -c "MX25L6436E/MX25L6445E/MX25L6465E/MX25L6473E/MX25L6473F" \
> -r ./2022_05_10_FIRST_DUMP.dat
flashrom  on Linux 5.10.63-v7+ (armv7l)
flashrom is free software, get the source code at https://flashrom.org

Using clock_gettime for delay loops (clk_id: 1, resolution: 1ns).
Using default 2000kHz clock. Use 'spispeed' parameter to override.
Found Macronix flash chip "MX25L6436E/MX25L6445E/MX25L6465E/MX25L6473E/MX25L6473F" (8192 kB, SPI) on linux_spi.
Reading flash... done.
```

IP 카메라 분석환경 구축

펌웨어 추출

binwalk를 통해 파일시스템 추출

```
> binwalk ./2022_05_10_FIRST_DUMP.dat

DECIMAL       HEXADECIMAL     DESCRIPTION
--------------------------------------------------------------------------------
32768         0x8000          LZMA compressed data, properties: 0x5D, dictionary size: 8388608 bytes, uncompressed size: 234984 bytes
262144        0x40000         uImage header, header size: 64 bytes, header CRC: 0x2A56C21, created: 2021-07-30 03:18:16, image size:
0x33A94717, OS: Linux, CPU: MIPS, image type: OS Kernel Image, compression type: none, image name: "Linux-4.9.145"
266368        0x41080         LZMA compressed data, properties: 0x5D, dictionary size: 33554432 bytes, uncompressed size: -1 bytes
2133872       0x208F70        Flattened device tree, size: 11232 bytes, version: 17
2293760       0x230000        Squashfs filesystem, little endian, version 4.0, compression:xz, size: 900914 bytes, 202 inodes, blocks
3276800       0x320000        Squashfs filesystem, little endian, version 4.0, compression:xz, size: 3110690 bytes, 206 inodes, block
7995392       0x7A0000        JFFS2 filesystem, little endian
```

```
> ls -d */
jffs2-root/    jffs2-root-10/  jffs2-root-13/  jffs2-root-16/  jffs2-root-19/  jffs2-root-21/
jffs2-root-3/  jffs2-root-6/  jffs2-root-9/
jffs2-root-0/  jffs2-root-11/  jffs2-root-14/  jffs2-root-17/  jffs2-root-2/   jffs2-root-22/
jffs2-root-4/  jffs2-root-7/  squashfs-root/
jffs2-root-1/  jffs2-root-12/  jffs2-root-15/  jffs2-root-18/  jffs2-root-20/  jffs2-root-23/
jffs2-root-5/  jffs2-root-8/  squashfs-root-0/
```

IP 카메라 분석환경 구축

펌웨어 추출

binwalk를 통해 파일시스템 추출

```
> ls -al
total 132
drwxr-xr-x 18 gyul gyul 61440 Jun 19 01:33 .
drwxr-xr-x  5 gyul gyul  4096 Apr 26 16:03 ..
drwxr-xr-x  2 gyul gyul  4096 Mar  3 01:22 bin
drwxr-xr-x  2 gyul gyul  4096 Mar  3 01:22 dev
drwxr-xr-x  8 gyul gyul  4096 Mar 29 00:18 etc
drwxr-xr-x  3 gyul gyul  4096 Mar  3 01:22 home
drwxr-xr-x  3 gyul gyul  4096 Mar  3 01:22 include
lrwxrwxrwx  1 gyul gyul    11 Mar  3 01:22 init -> bin/busybox
drwxr-xr-x  4 gyul gyul  4096 Mar  3 01:22 lib
lrwxrwxrwx  1 gyul gyul    11 Mar  3 01:22 linuxrc -> bin/busybox
drwxr-xr-x  2 gyul gyul  4096 Mar  3 01:22 media
drwxr-xr-x  3 gyul gyul  4096 Mar  3 01:22 mnt
drwxr-xr-x  4 gyul gyul  4096 Mar  3 01:22 opt
drwxr-xr-x  2 gyul gyul  4096 Mar  3 01:22 proc
drwxr-xr-x  2 gyul gyul  4096 Mar  3 01:22 root
lrwxrwxrwx  1 gyul gyul     3 Mar  3 01:22 run -> tmp
drwxr-xr-x  2 gyul gyul  4096 Mar  3 01:22 sbin
drwxr-xr-x  2 gyul gyul  4096 Mar  3 01:22 sys
drwxr-xr-x  3 gyul gyul  4096 Mar 14 08:50 tmp
drwxr-xr-x  7 gyul gyul  4096 Mar  3 01:22 usr
drwxr-xr-x  4 gyul gyul  4096 Mar  3 01:22 var
```

IP 카메라 분석환경 구축

펌웨어 추출

TIPS – flashrom

펌웨어 추출

TIPS – binwalk

IP 카메라 분석환경 구축

Dynamic Analysis

IP 카메라 내부 쉘 접근

1. 파일시스템 내에 telnetd 바이너리 삽입
2. rcS 스크립트, shadow 파일 수정
3. 수정한 펌웨어 flash에 쓰기

gdbserver

1. 파일시스템 내에 gdbserver 바이너리 삽입
2. 수정한 펌웨어 flash에 쓰기
3. gdb-multiarch로 디버깅

IP 카메라 분석환경 구축

IP 카메라 내부 쉘 접근

telnetd, gdbserver 빌드

```
> file ./busybox
./busybox: ELF 32-bit LSB executable, MIPS, MIPS-I version 1,
 dynamically linked, interpreter /lib/ld-uClibc.so.0, BuildID
[sha1]=a15a595288fce4cdfc545f3a6ff5dd2d32530d31, stripped
```

BuildRoot
Making Embedded Linux Easy

IP 카메라 내부 쉘 접근

telnetd, gdbserver 빌드

Target options

```
    Target Architecture (MIPS (little endian))  --->
    Target Binary Format (ELF)  --->
    Target Architecture Variant (Generic MIPS32)  --->
[ ] Use soft-float
    FP mode (xx)  --->
```

IP 카메라 내부 쉘 접근

telnetd, gdbserver 빌드

buildroot -> make menuconfig

```
[ ] dt
    *** duma needs a toolchain w/ C++, threads, dy
    *** fio needs a toolchain w/ dynamic library,
[*] gdb
-*-   gdbserver
[ ]   full debugger (NEW)
[ ] google-breakpad
[ ] iozone
[ ] kexec
    *** ktap needs a Linux kernel to be built ***
[ ] latencytop
```

buildroot -> make busybox-menuconfig

```
[ ] tcpsvd (14 kb)
[ ] udpsvd (13 kb)
[*] telnet (8.8 kb)
[*]   Pass TERM type to remote host
[*]   Pass USER type to remote host
[*]   Enable window size autodetection
[ ] telnetd (12 kb)
[*] tftp (11 kb)
[ ]   Enable progress bar
[*]   tftp-hpa compat (support -c get/put FILE)
[ ] tftpd (10 kb)
[*]   Enable 'tftp get' and/or tftpd upload code
```

IP 카메라 내부 쉘 접근

rcS 스크립트, shadow 수정

/etc/init.d/rcS

```
#!/bin/sh

# Set mdev
echo /sbin/mdev > /proc/sys/kernel/hotplug
/sbin/mdev -s && echo "mdev is ok......"

# create console and null node for nfsroot
#mknod -m 600 /dev/console c 5 1
#mknod -m 666 /dev/null c 1 3


# networking
ifconfig lo up
#ifconfig eth0 192.168.1.80

# Start telnet daemon
telnetd &

# Set the system time from the hardware clock
hwclock -s
```

/etc/shadow

```
root:$1$soid...:10933:0:99999:7:::
↓
root::10933:0:99999:7:::
```

IP 카메라 내부 쉘 접근

파일시스템 빌드 후 펌웨어에 덮어쓰기

수정한 파일시스템 빌드

```
> mksquashfs ./first/ ./new_filesystem -comp xz
Parallel mksquashfs: Using 6 processors
Creating 4.0 filesystem on ./new_filesystem, block size 131072.
[=================================================================

Exportable Squashfs 4.0 filesystem, xz compressed, data block size
        compressed data, compressed metadata, compressed fragments
        duplicates are removed
Filesystem size 8041.64 Kbytes (7.85 Mbytes)
        32.73% of uncompressed filesystem size (24570.31 Kbytes)
Inode table size 5750 bytes (5.62 Kbytes)
        17.32% of uncompressed inode table size (33208 bytes)
Directory table size 8080 bytes (7.89 Kbytes)
        52.54% of uncompressed directory table size (15380 bytes)
```

기존 펌웨어에 덮어쓰기

```
005306F0  9B F8 A0 19 3F 94 BA 8F A0 83 B7 0C 8C F1 F0 41  ›ø .?”°. ƒ·.Œñ›A
00530700  ED 15 90 EA F2 04 C3 7B 93 59 93 65 F0 1E 8E FC  í..êò.Ã{“Y“e›.Žü
00530710  7A 45 2A DF 6C 85 E6 0B 81 91 35 BA 2A CB F3 D0  zE*ßl…æ..‘5º*Ëó›
00530720  4A AB CF 04 AD 01 A7 09 75 E3 AD 68 39 C5 81 A9  J«Ï...§.uã.h9Å.©
00530730  C7 C0 A0 0C AD A6 AE 4F 8B 8A 73 36 F2 9E 68 87  ÇÀ ..¦®O‹Šs6òžh‡
00530740  20 FD B7 2F 8D E3 0F 0C D3 AC D7 A5 8A 38 30 A9   ý·/.ã..Ó¬×¥Š80©
00530750  AF 51 0C 01 2A F6 47 47 F9 19 DE 40 29 40 65 7C  ¯Q..*öGGù.Þ@)@e|
00530760  5B 92 E7 02 59 0F 5C 45 C3 B5 9D 9B 3E 58 C5 47  [’ç.Y.\EÃµ.›>XÅG
00530770  AA F5 CA A9 03 D1 91 22 02 EB 35 07 87 5D 32 37  ªõÊ©.Ñ‘".ë5.‡]27
00530780  23 B7 13 C9 96 A6 13 66 2F 61 5F 8E D6 2E A7 13  #·.É–¦.f/a_ŽÖ.§.
00530790  4E ED 65 9C DF 14 AA EB 0D 18 EB 4D 49 DE 61 21  NíeœÞ.ªë..ëMIÞa!
005307A0  21 A5 02 0D 52 0C AF 38 35 9B E7 49 D6 35 F3 C5  !¥..R.¯85›çIÖ5óÅ
005307B0  AD 94 00 98 44 09 27 F8 80 88 3C E8 6E 32 AA 47  .”.˜D.'ø€ˆ<èn2ªG
005307C0  1D 3B B6 25 38 D9 98 A8 1A 61 11 39 C6 1D 70 B0  .;¶%8Ù˜¨.a.9Æ.p°
005307D0  90 A1 FD 7F 45 AF 6A 76 1B 29 C8 80 65 C6 30 F6  .¡ý.E¯jv.)È€eÆ0ö
005307E0  E2 BC 33 69 7B 1E 65 CA F5 EA C7 33 DD D2 F0 61  â¼3i{.eÊõêÇ3ÝÒ›a
005307F0  EF 62 29 E4 DA B0 0F FC BD BF A1 03 93 58 B3 AC  ïb)äÚ°.ü½¿¡.“X³¬
```

IP 카메라 내부 쉘 접근

수정한 펌웨어 flash에 쓰기

```
> flashrom -p linux_spi:dev=/dev/spidev0.0 -c "MX25L6436E/MX25L6445E/MX25L6465E/MX25L6473E/MX25L6473F" -w ./2022_05_13_EDITED.dat
flashrom  on Linux 5.10.63-v7+ (armv7l)
flashrom is free software, get the source code at https://flashrom.org

Using clock_gettime for delay loops (clk_id: 1, resolution: 1ns).
Using default 2000kHz clock. Use 'spispeed' parameter to override.
Found Macronix flash chip "MX25L6436E/MX25L6445E/MX25L6465E/MX25L6473E/MX25L6473F" (8192 kB, SPI) on linux_spi.
Reading old flash chip contents... done.
Erasing and writing flash chip...
Warning: Chip content is identical to the requested image.
Erase/write done.
```

IP 카메라 분석환경 구축

IP 카메라 내부 쉘 접근

gdb-multiarch로 디버깅

```
[root@[redacted]:tmp]# ./gdbserver-7.12-mipsel-i-v1-sysv :8888 --attach 120
Attached; pid = 120
Listening on port 8888
Remote debugging from host 192.168.0.7
```

```
> gdb-multiarch
GNU gdb (Ubuntu 7.11.1-0ubuntu1~16.5) 7.11.1
Copyright (C) 2016 Free Software Foundation, Inc.
License GPLv3+: GNU GPL version 3 or later <http://gn
This is free software: you are free to change and rec
There is NO WARRANTY, to the extent permitted by law.
and "show warranty" for details.
This GDB was configured as "x86_64-linux-gnu".
Type "show configuration" for configuration details.
For bug reporting instructions, please see:
<http://www.gnu.org/software/gdb/bugs/>.
Find the GDB manual and other documentation resources
<http://www.gnu.org/software/gdb/documentation/>.
For help, type "help".
Type "apropos word" to search for commands related to
pwndbg: loaded 194 commands. Type pwndbg [filter] for
pwndbg: created $rebase, $ida gdb functions (can be u
pwndbg> set arch mips
The target architecture is assumed to be mips
pwndbg> set endian little
The target is assumed to be little endian
pwndbg> target remote 192.168.0.27:8888
Remote debugging using 192.168.0.27:8888
```

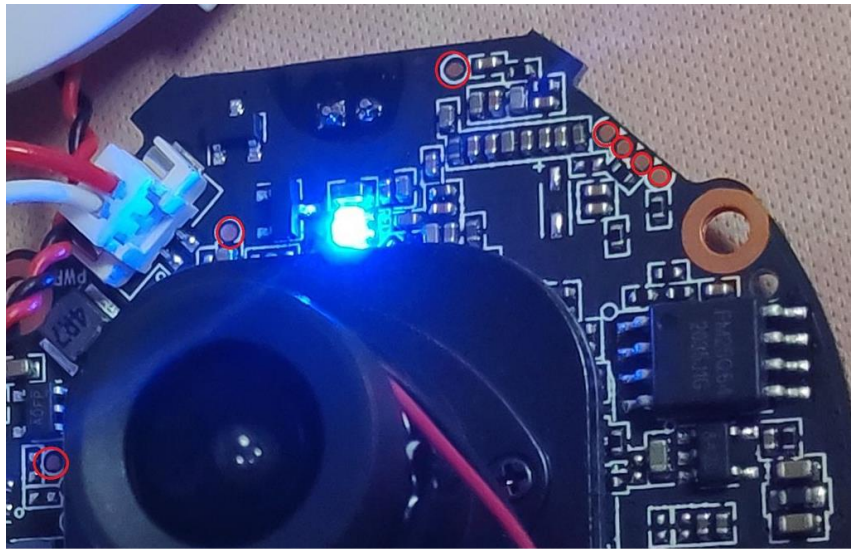
IP 카메라 내부 쉘 접근

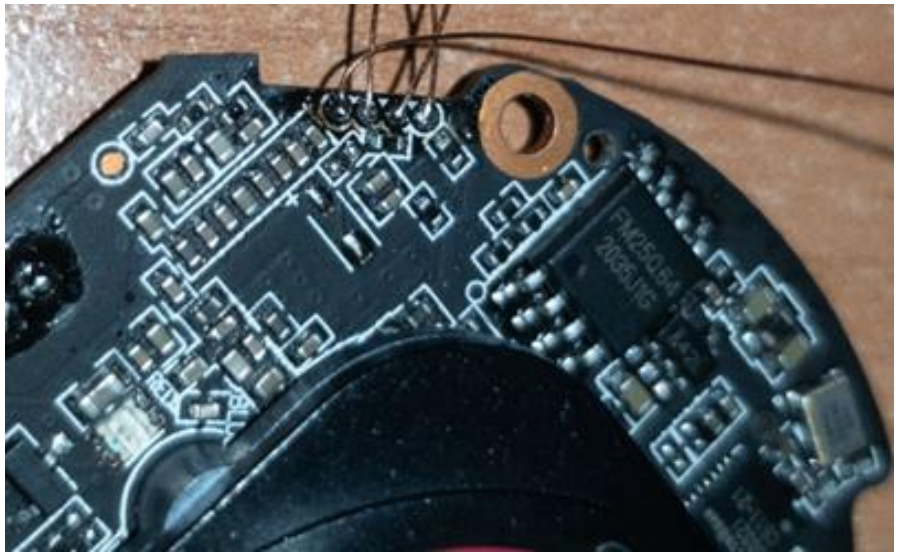
gdb-multiarch로 디버깅

```
T1   0x5000
T2   0x0
T3   0x0
T4   0x0
T5   0x0
T6   0x0
T7   0x710000 ◂— 0x114
T8   0x7163f8 (usleep@got.plt) —▸ 0x7718ab30 ◂— lui    $gp, 5
T9   0x7718b430 ◂— lui    $gp, 5
S0   0x0
S1   0x7fc6a1a0 ◂— 0x0
S2   0x760000 ◂— 0x0
S3   0x1
S4   0x28
S5   0x760000 ◂— 0x0
S6   0x490000 ◂— lw     $s2, -0x7d30($gp)
S7   0x77d6d444
S8   0x48c3e8 (mimo_ak_bc_parse+1676) ◂— b      0x48c12c
FP   0x0
SP   0x7fc6a160 ◂— 0x772f2010
PC   0x77139470 ◂— beqz   $a3, 0x7713948c
──────────────────────────────────────────────[ DISASM ]──────────────
 ► 0x77139470    beqz   $a3, 0x7713948c

   0x77139474    move   $s0, $v0
   0x77139478    lw     $v0, -0x70f4($gp)
```

IP 카메라 내부 쉘 접근

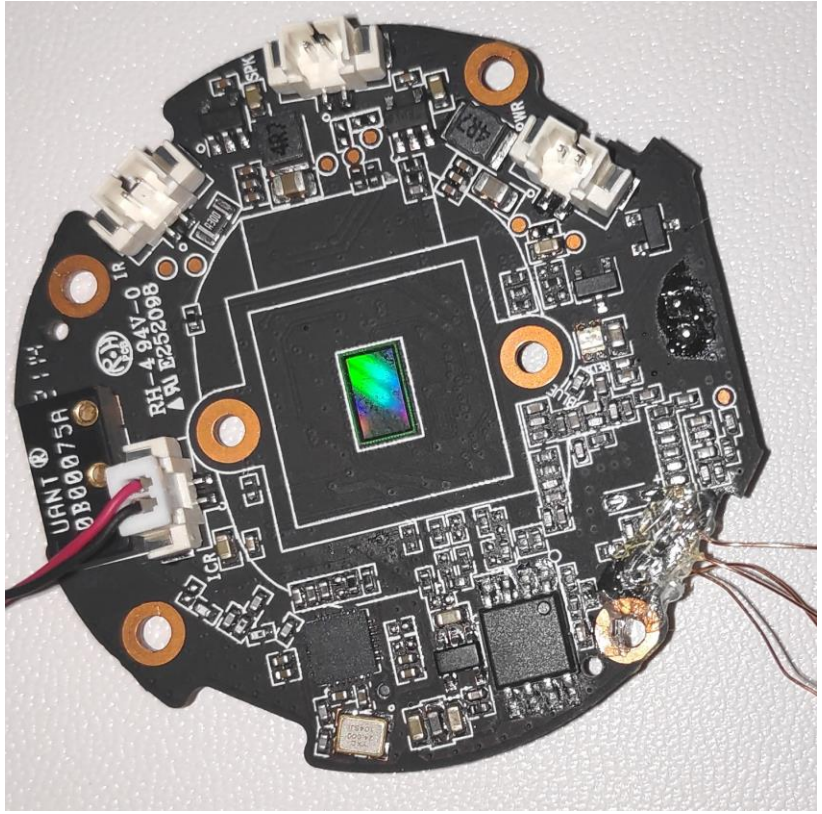
TIPS – UART





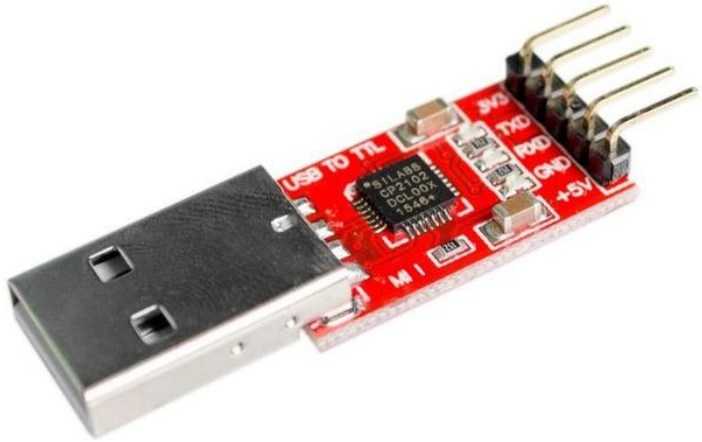
IP 카메라 내부 쉘 접근

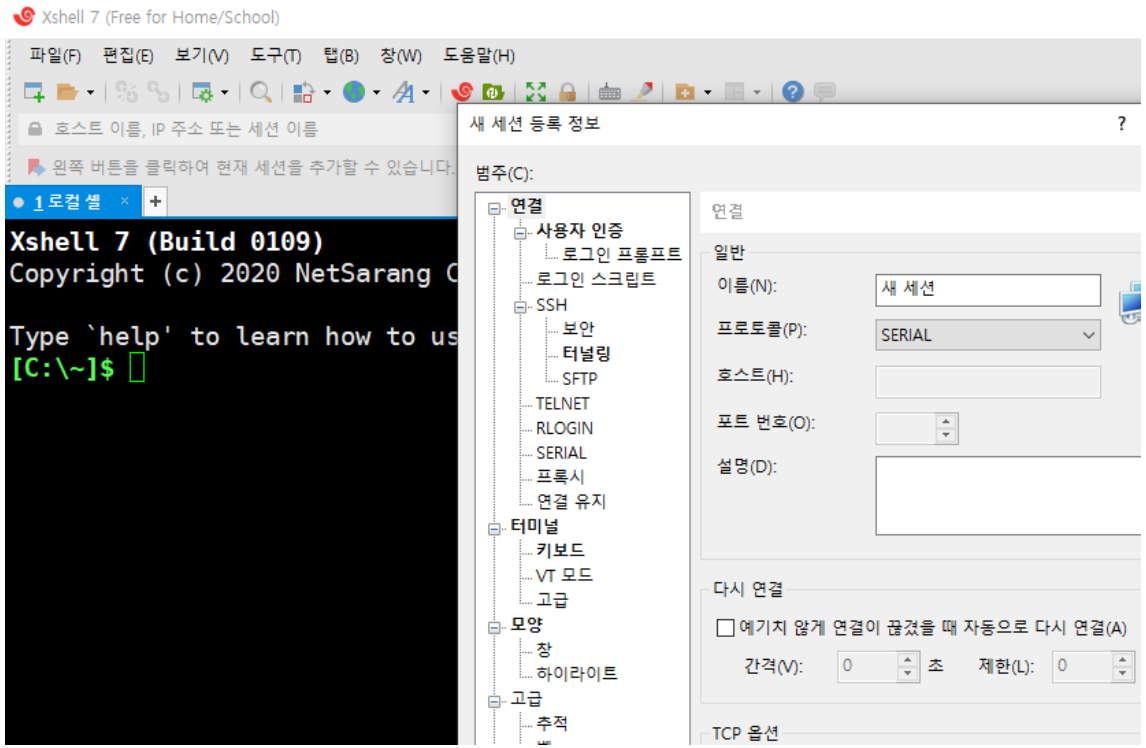
TIPS – UART



IP 카메라 내부 쉘 접근

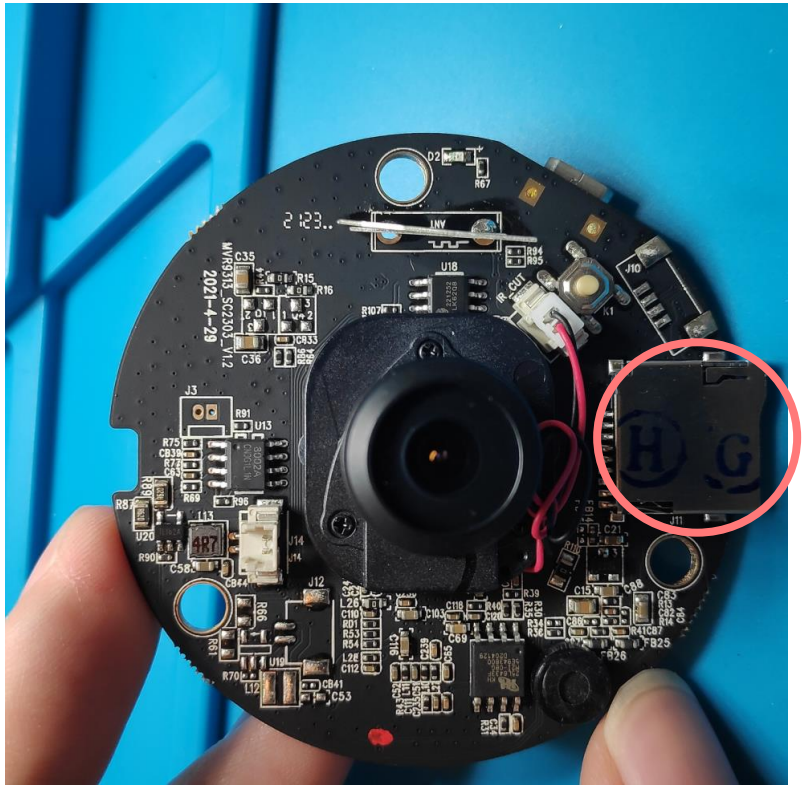
TIPS – UART





IP 카메라 내부 쉘 접근

TIPS – SD카드 사용하기



IP 카메라 내부 쉘 접근

TIPS – SD카드 사용하기

```
[root@[redacted]:tmp]# cd /mnt/sdcard/files/
[root@[redacted]:files]# ls
gdbserver_mipsel_i                tcpdump-mipsel-i-sysv
busybox_edited_2                  gdbserver-7.12-mipsel-i-v1-sysv
busybox-mipsel-mips32-v1-sysv     busybox-mipsel
```

IP 카메라 내부 쉘 접근

TIPS – 프로세스 출력 확인 #1

/etc/init.d/rcS

```
#!/bin/sh

# Set mdev
echo /sbin/mdev > /proc/sys/kernel/hotplug
/sbin/mdev -s && echo "mdev is ok......"

# create console and null node for nfsroot
#mknod -m 600 /dev/console c 5 1
#mknod -m 666 /dev/null c 1 3


# networking
ifconfig lo up
#ifconfig eth0 192.168.1.80

# Start telnet daemon
telnetd &

# Set the system time from the hardware clock
hwclock -s

service_app > /tmp/log &
```

service_app > /tmp/app_log

tail –f /tmp/app_log

IP 카메라 내부 쉘 접근

TIPS – 프로세스 출력 확인 #2

strace –fp[pid] –s 9999 –e write

```
write(2, "gyul@DESKTOP-G0891FA:~$ ", 24) = 24
write(2, "l", 1)                        = 1
write(2, "s", 1)                        = 1
write(2, "\n", 1)                       = 1
strace: Process 30698 attached
[pid 30698] write(1, "\33[0m\33[01;34manaconda3\33[0m      \33[01;3
[0m\n", 169) = 169
[pid 30698] write(1, "\33[01;34mARM_buildroot\33[0m  \33[01;34mfatc
hfs-tools\33[0m\n", 179) = 179
[pid 30698] write(1, "\33[01;34mbinbloom\33[0m       \33[01;34mfirm
62) = 162
[pid 30698] write(1, "\33[01;34mbinwalk\33[0m        \33[01;32mfull
[pid 30698] write(1, "\33[01;34mboofuzz\33[0m        \33[01;34mfuzz
[pid 30698] write(1, "\33[01;34mchecksec\33[0m       \33[01;34mgdb-
[pid 30698] +++ exited with 0 +++
--- SIGCHLD {si_signo=SIGCHLD, si_code=CLD_EXITED, si_pid=30698, si
write(2, "gyul@DESKTOP-G0891FA:~$ ", 24) = 24
```

IP 카메라 분석환경 구축

```
int32_t r0_1 = arg3 + ((arg2 + 1) << 2)
*data_98e14 = 0
*data_987e8 = arg11
*data_99e9c = r0_1
int32_t r3_2
do
    r3_2 = *r0_1
    r0_1 = r0_1 + 4
while (r3_2 != 0)
sub_23000(r0_1)
if (*data_98e14 == 0)
    int32_t r0_2 = sub_23b54()
    if (r0_2 s< 0)
        sub_d8a4(0x79740)  {"FATAL: cannot determine ker
        noreturn
    int32_t r3_4 = *data_9ada8
    int32_t r3_5
    if (r3_4 == 0)
```

Static Analysis

- 펌웨어 추출

```
R12  0x400890 ◂— xor    ebp, ebp
R13  0x7fffffffe410 ◂— 0x1
R14  0x0
R15  0x0
RBP  0x7fffffffe330 —▸ 0x400d40 ◂— push   r15
RSP  0x7fffffffe318 —▸ 0x400d30 ◂— mov    eax, 0
RIP  0x7ffff7b04360 (__read_nocancel+7) ◂— cmp    rax, -0xfff

 ► 0x7ffff7b04360 <__read_nocancel+7>     cmp    rax, -0xfff
   0x7ffff7b04366 <__read_nocancel+13>    jae    read+73 <read
    ↓
   0x7ffff7b04399 <read+73>               mov    rcx, qword pt
   0x7ffff7b043a0 <read+80>               neg    eax
   0x7ffff7b043a2 <read+82>               mov    dword ptr fs:
   0x7ffff7b043a5 <read+85>               or     rax, 0xffffff
   0x7ffff7b043a9 <read+89>               ret

   0x7ffff7b043aa                         nop    word ptr [rax
   0x7ffff7b043b0 <write>                 cmp    dword ptr [ri
   0x7ffff7b043b7 <write+7>               jne    write+25 <wri
```

Dynamic Analysis

- IP 카메라 내부 쉘 접근

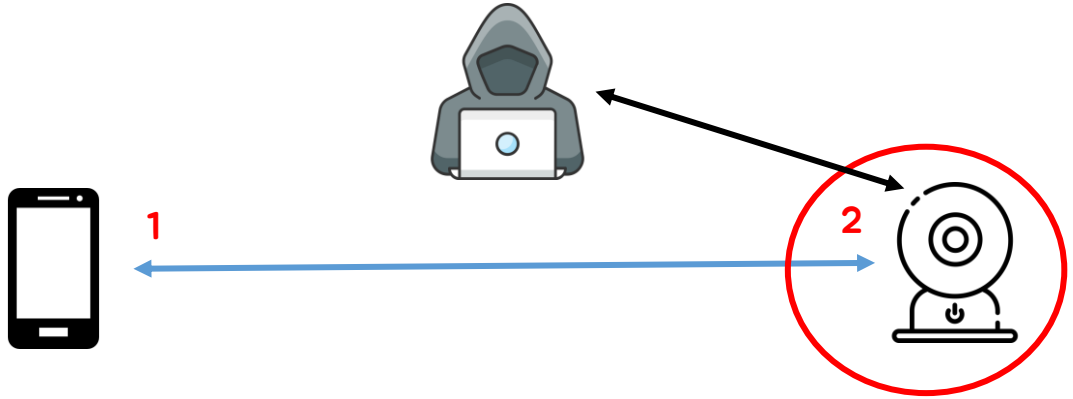
- gdbserver

IP 카메라 취약점 분석

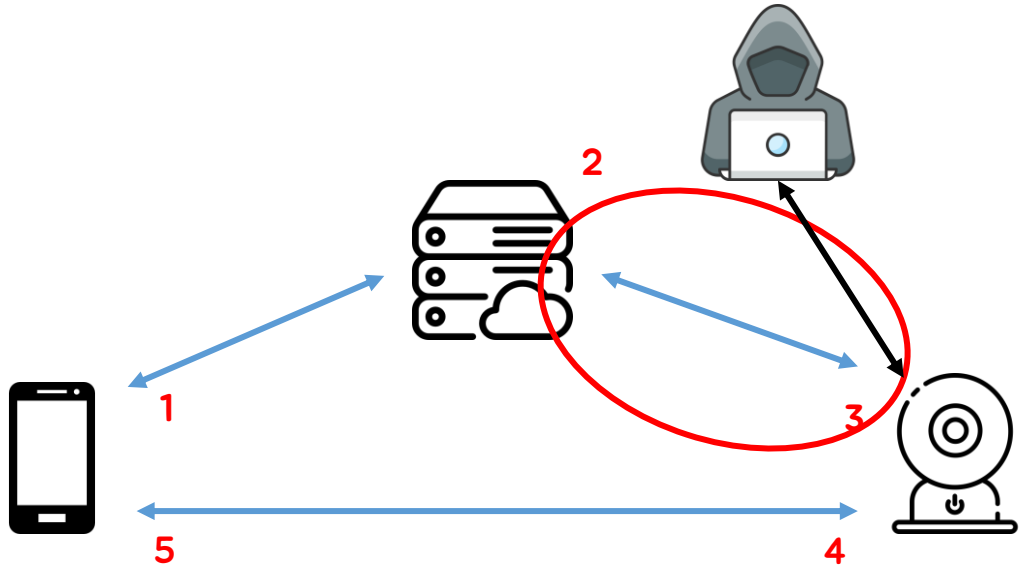

DOS
COMMAND INJECTION
STACK BOF
After all these years.
Finally, I have them all.
imgflip.com

IP 카메라 취약점 분석

1



2



IP 카메라 취약점 분석

분석 프로세스

열려있는 포트 확인

```
> netstat -lan
Active Internet connections (servers and established)
Proto Recv-Q Send-Q Local Address           Foreign Address         State
tcp        0      0 0.0.0.0:6668            0.0.0.0:*               LISTEN
tcp        0      0 192.168.0.27:49628      [redacted]              ESTABLISHED
tcp        0      0 :::23                   :::*                    LISTEN
tcp        0     84 ::ffff:192.168.0.27:23  ::ffff:192.168.0.7:3552 ESTABLISHED
udp        0      0 192.168.0.27:43913      0.0.0.0:*
udp        0      0 127.0.0.1:54193         0.0.0.0:*
raw      576      0 0.0.0.0:1               0.0.0.0:*               1
Active UNIX domain sockets (servers and established)
```

분석 프로세스

```
# /etc/inittab
#
# Copyright (C) 2001 Erik Andersen <andersen@codepoet.org>
#
# Note: BusyBox init doesn't support runlevels.  The runlevels field is
# completely ignored by BusyBox init. If you want runlevels, use
# sysvinit.
#
# Format for each entry: <id>:<runlevels>:<action>:<process>
#
# id        == tty to run on, or empty for /dev/console
# runlevels == ignored
# action    == one of sysinit, respawn, askfirst, wait, and once
# process   == program to run

# Startup the system
::sysinit:/sbin/swapoff -a
::sysinit:/bin/mount -t tmpfs tmpfs /dev
::sysinit:/bin/mkdir -p /dev/pts
::sysinit:/bin/mkdir -p /dev/shm
::sysinit:/bin/mount -a
::sysinit:/bin/hostname -F /etc/hostname

# now run any rc scripts
::sysinit:/etc/init.d/rcS

# Put a getty on the serial port
ttyS1::respawn:/sbin/getty -L ttyS1 57600 vt100 # GENERIC_SERIAL

# Stuff to do for the 3-finger salute
#::ctrlaltdel:/sbin/reboot

# Stuff to do before rebooting
::shutdown:/bin/umount -a -r
```

/etc/inittab

시스템 Init 및 shutdown에 실행되는 명령어가 담긴 스크립트

분석 프로세스

```
#!/bin/sh

# Set mdev
echo /sbin/mdev > /proc/sys/kernel/hotplug
/sbin/mdev -s && echo "mdev is ok......"

# create console and null node for nfsroot
#mknod -m 600 /dev/console c 5 1
#mknod -m 666 /dev/null c 1 3


# networking
ifconfig lo up
#ifconfig eth0 192.168.1.80

# Start telnet daemon
telnetd &

# Set the system time from the hardware clock
hwclock -s

service_app > /tmp/log &

# Mount system partition
mount -t squashfs /dev/mtdblock5 /app
mount -t jffs2 -o sync /dev/mtdblock6 /conf
```

/etc/init.d/rcS

시스템 Init에서 inittab에 의해 실행되는 쉘 스크립트

IP 카메라 취약점 분석

분석 프로세스

init 스크립트 확인

inittab

```
# /etc/inittab
#
# Copyright (C) 2001 Erik Andersen <andersen@cod
#
# Note: BusyBox init doesn't support runlevels.
# completely ignored by BusyBox init. If you wan
# sysvinit.
#
# Format for each entry: <id>:<runlevels>:<actio
#
# id        == tty to run on, or empty for /dev/
# runlevels == ignored
# action    == one of sysinit, respawn, askfirst
# process   == program to run

# Startup the system
::sysinit:/sbin/swapoff -a
::sysinit:/bin/mount -t tmpfs tmpfs /dev
::sysinit:/bin/mkdir -p /dev/pts
::sysinit:/bin/mkdir -p /dev/shm
::sysinit:/bin/mount -a
::sysinit:/bin/hostname -F /etc/hostname

# now run any rc scripts
::sysinit:/etc/init.d/rcS
```

->

rcS

```
#!/bin/sh

# Set mdev
echo /sbin/mdev > /proc/sys/kernel/hotplug
/sbin/mdev -s && echo "mdev is ok......"

# create console and null node for nfsroot
#mknod -m 600 /dev/console c 5 1
#mknod -m 666 /dev/null c 1 3


# networking
ifconfig lo up
#ifconfig eth0 192.168.1.80

# Start telnet daemon
telnetd &

# Set the system time from the hardware clock
hwclock -s

service_app > /tmp/log &
```

->

서비스 바이너리

```
int v12; // r7
int v13; // r9
_DWORD *v14; // r4
int v15; // r3
int (__fastcall *v16)(_DWORD *, int, int); // r3
int v17; // r3

v5 = a3 == 0;
if ( a3 )
  v5 = a2 == 0;
v6 = v5;
if ( !result )
  v6 |= 1u;
if ( !v6 )
{
  v7 = result;
  v10 = sub_10ECC0();
  v11 = *(_DWORD **)(v10 + 48);
  v12 = v10;
  v13 = v11[535];
  if ( !v13 )
```

분석 프로세스

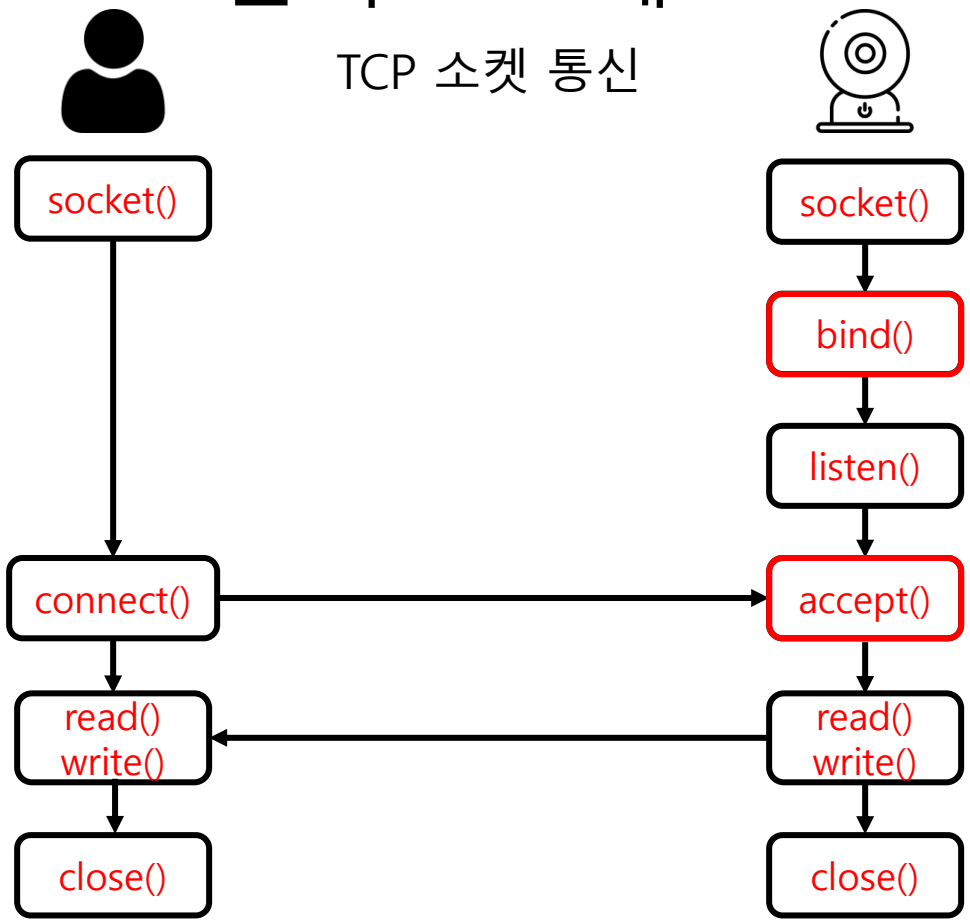
서비스 바이너리 보호기법 확인

```
> checksec --file=./service_app
WARNING: 'openssl' not found! It's required for most checks.

WARNING: Not all necessary commands found. Some tests might not

RELRO           STACK CANARY      NX            PIE
No RELRO        Canary found      NX disabled   No PIE
```

IP 카메라 취약점 분석

분석 프로세스

TCP 소켓 통신



IP 카메라 취약점 분석

분석 프로세스

통신 함수 분석

HTTP
(http://192.168.0.1/attack)

CUSTOM
(192.168.0.1:1234)

서버와 IP카메라가 통신하면서 보내야 하는 데이터들?

IP 카메라 취약점 분석

분석 프로세스

통신 함수 분석

서버와 IP카메라가 통신하면서 보내야 하는 데이터들

COMMAND ID

- 인증
  - ID, PW
- 영상 데이터 요청
  - 전송할 주소 정보
  - 사용자 크레덴셜
- 카메라 조작
  - 조작 명령 데이터
  - 사용자 크레덴셜
- 펌웨어 업데이트
  - 펌웨어 데이터
  - 사용자 크레덴셜
- 내부 시스템 세팅
  - 세팅할 값(ex. 네트워크)
  - 사용자 크레덴셜

취약점 분석

통신 함수 분석

COMMAND ID

- 인증
  ID, PW
- 영상 데이터 요청
  전송할 주소 정보
  사용자 크레덴셜
- 카메라 조작
  조작 명령 데이터
  사용자 크레덴셜
- 펌웨어 업데이트
  펌웨어 데이터
  사용자 크레덴셜
- 내부 시스템 세팅
  세팅할 값(ex. 네트워크)
  사용자 크레덴셜

IP 카메라 취약점 분석

통신 함수 분석

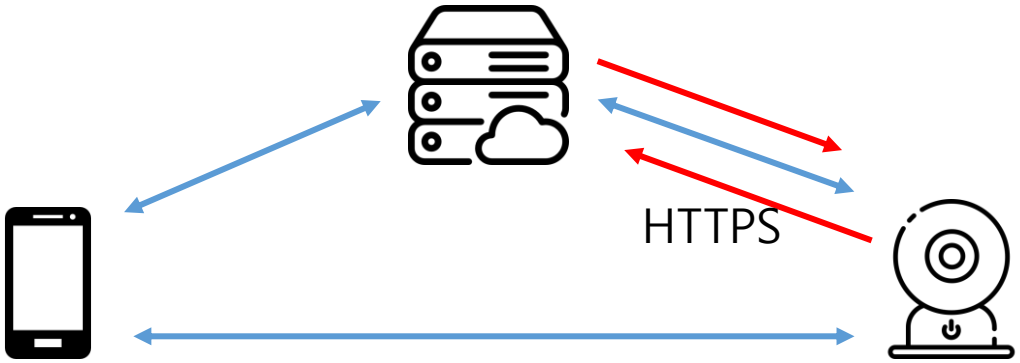
#인증 함수 예시 코드

```
length = recv(fd, data, 1024);
if(length > 0 ){
    command_id[0] = data[0];
    command_id[1] = data[1];

    if(command_id == 0x1234){
      id_pw = strdup(data[10]);
      id = strtok(id_pw, "/");
      pw = strtok(NULL, "/");

      id_length = (pw - id) - 1;
      pw_length = strlen(pw);

      real_id = base64decode(id, id_length);
      real_pw = base64decode(pw, pw_length);

      auth(real_id, real_pw);
    }
}
else {
  // ERROR
}

// auth function
    if(strcmp(real_id, saved_id) == 0){
      if(strcmp(real_pw, saved_pw) == 0){
        // SUCCESS
      }
    }
```

취약점 분석

TIPS – wireshark

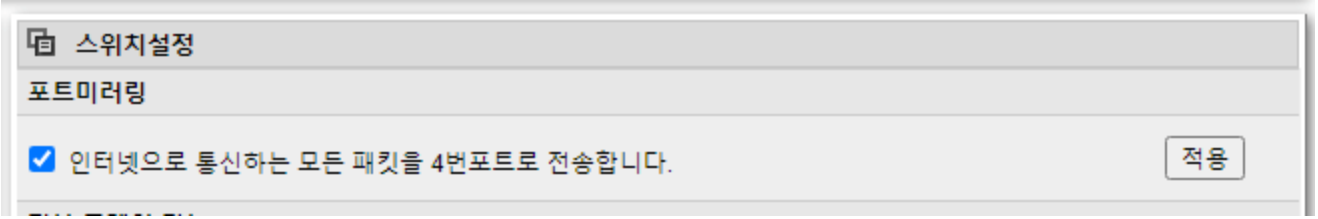
[redacted]	192.168.0.31	TCP	1514
	192.168.0.31	TCP	1514
	192.168.0.31	TCP	1514
	192.168.0.31	TCP	1514
	192.168.0.31	TCP	1514
	192.168.0.31	TCP	1514
	192.168.0.31	TCP	1514
	192.168.0.31	TLSv1.2	128
192.168.0.31	[redacted]	TCP	66



취약점 분석

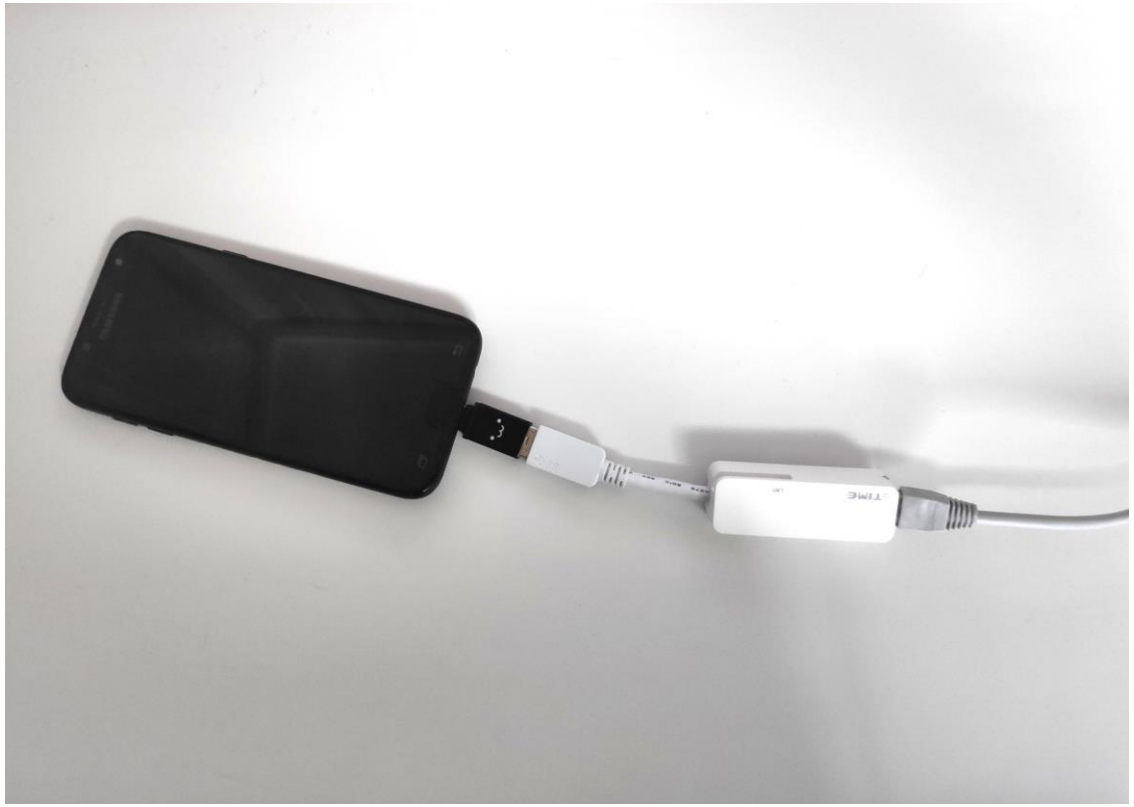
TIPS – wireshark

ipTIME 기준



취약점 분석

TIPS – wireshark



취약점 분석

TIPS – wireshark

```
> ./tcpdump-mipsel-i-sysv -i wlan0 -w ./log.pcap
tcpdump-mipsel-i-sysv: listening on wlan0, link-type EN10MB (Ethernet),
```

취약점 분석

TIPS – Fuzzing

COMMAND ID

- 인증
  - ID, PW
- 영상 데이터 요청
  - 전송할 주소 정보
  - 사용자 크레덴셜
- 카메라 조작
  - 조작 명령 데이터
  - 사용자 크레덴셜
- 펌웨어 업데이트
  - 펌웨어 데이터
  - 사용자 크레덴셜
- 내부 시스템 세팅
  - 세팅할 값(ex. 네트워크)
  - 사용자 크레덴셜

IP 카메라 취약점 분석

취약점 분석

TIPS – Fuzzing

Wireshark로 패킷 구조 분석 – 시드 파일 생성

COMMAND ID

+

Radamsa를 통한 Mutation

인증	영상 데이터 요청	카메라 조작	펌웨어 업데이트	내부 시스템 세팅
ID, PW	전송할 주소 정보 사용자 크레덴셜	조작 명령 데이터 사용자 크레덴셜	펌웨어 데이터 사용자 크레덴셜	세팅할 값(ex. 네트워크) 사용자 크레덴셜

+

느린 속도..😥

Future Work

Emulation

Q&A

kkhhss0290@gmail.com