A Deep Dive into Security Vulnerabilities and Proprietary Protocols: a case study

Victor Goeman

DistriNet

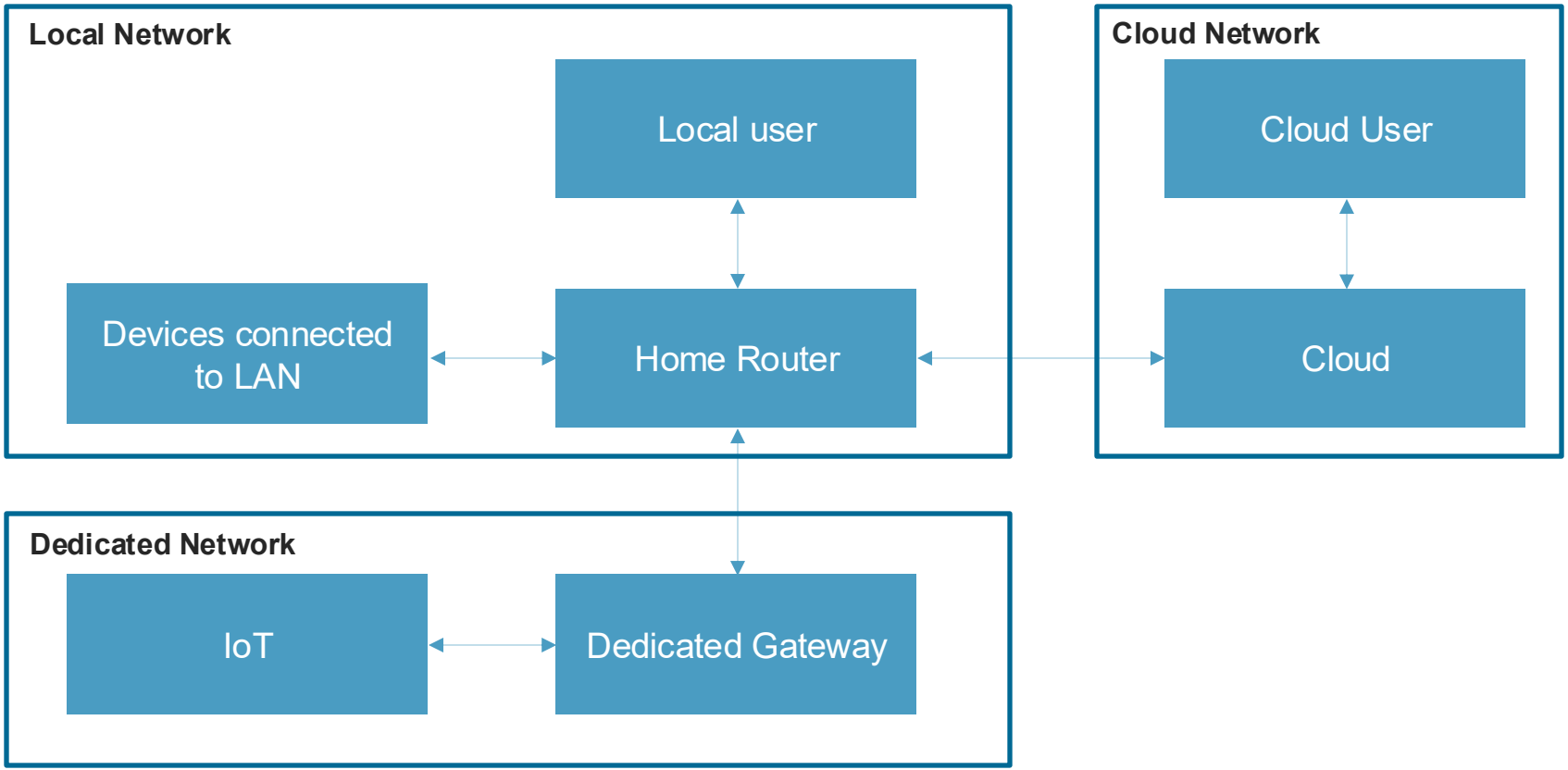
Device Case Studies

- Research: In-depth security analysis on a set of devices
- Research questions:
  - What is the current state of IoT security?
  - What vulnerabilities can we uncover?
  - What can we learn from these vulnerabilities?
  - What vulnerabilities could have been prevented?

Eufy

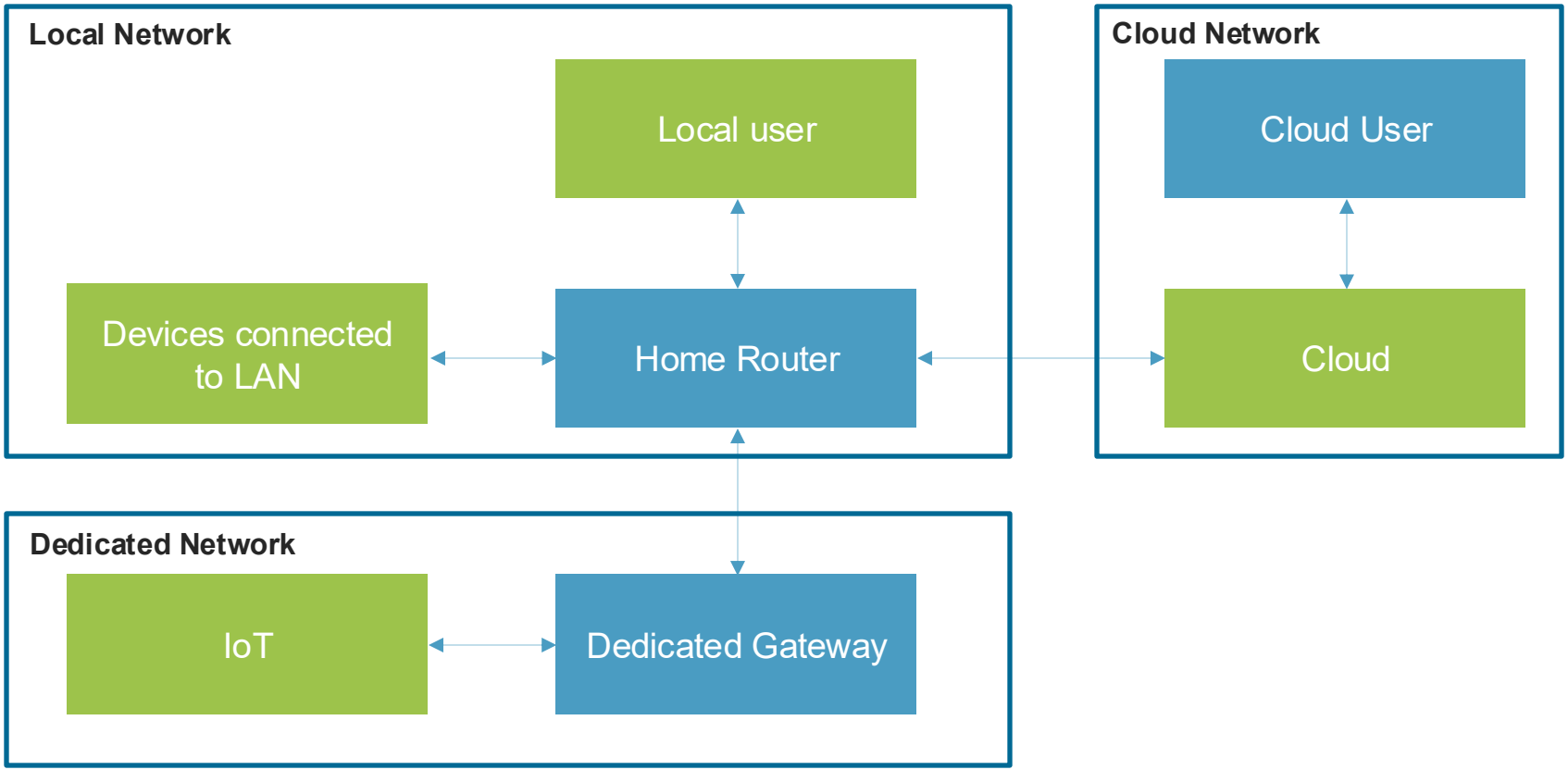
- Founded in 2016
- Already among market leaders
- Focus on security
- 1/334 Households

Eufy - Architecture



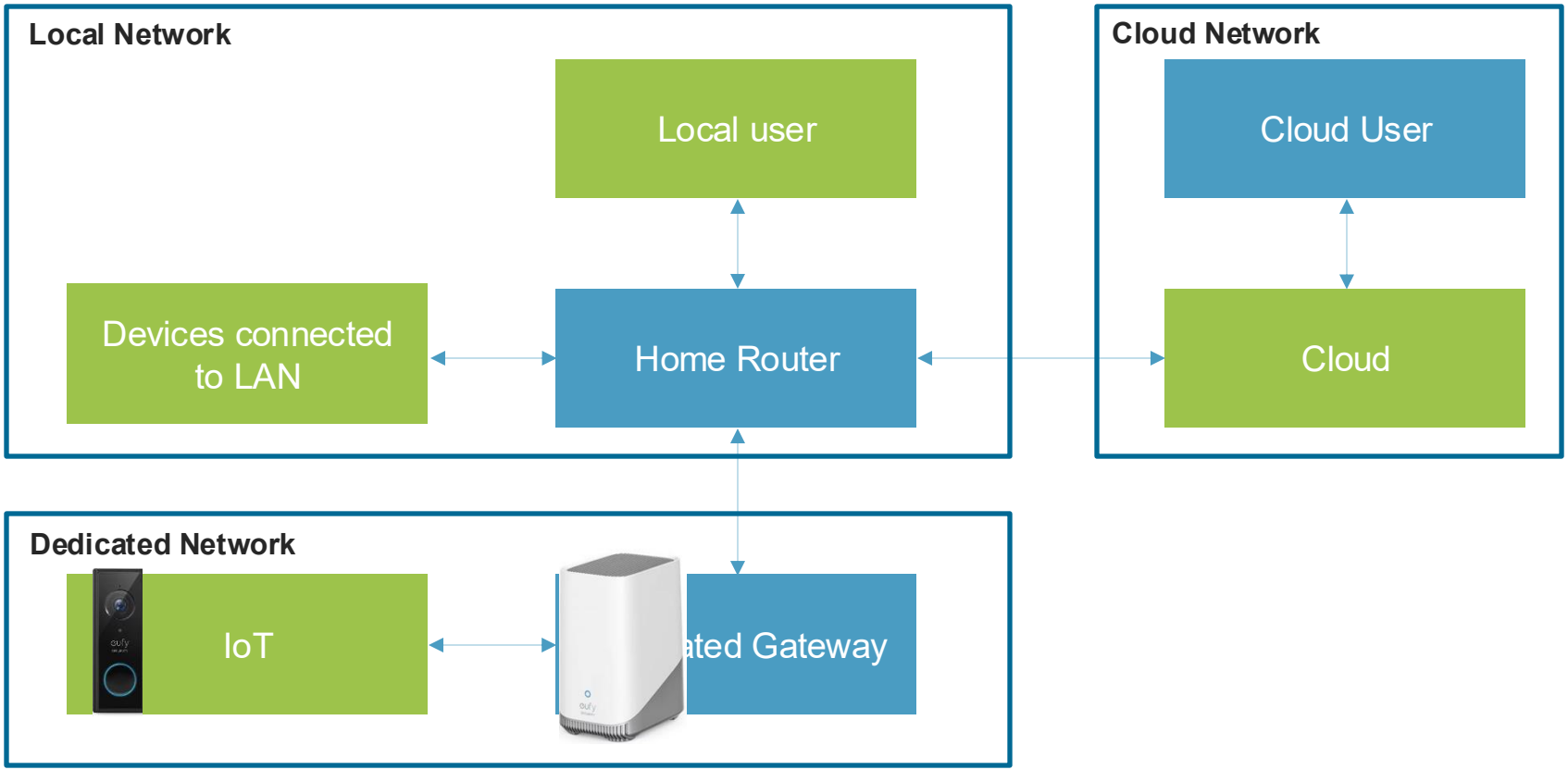
DistriNet

Eufy - Architecture



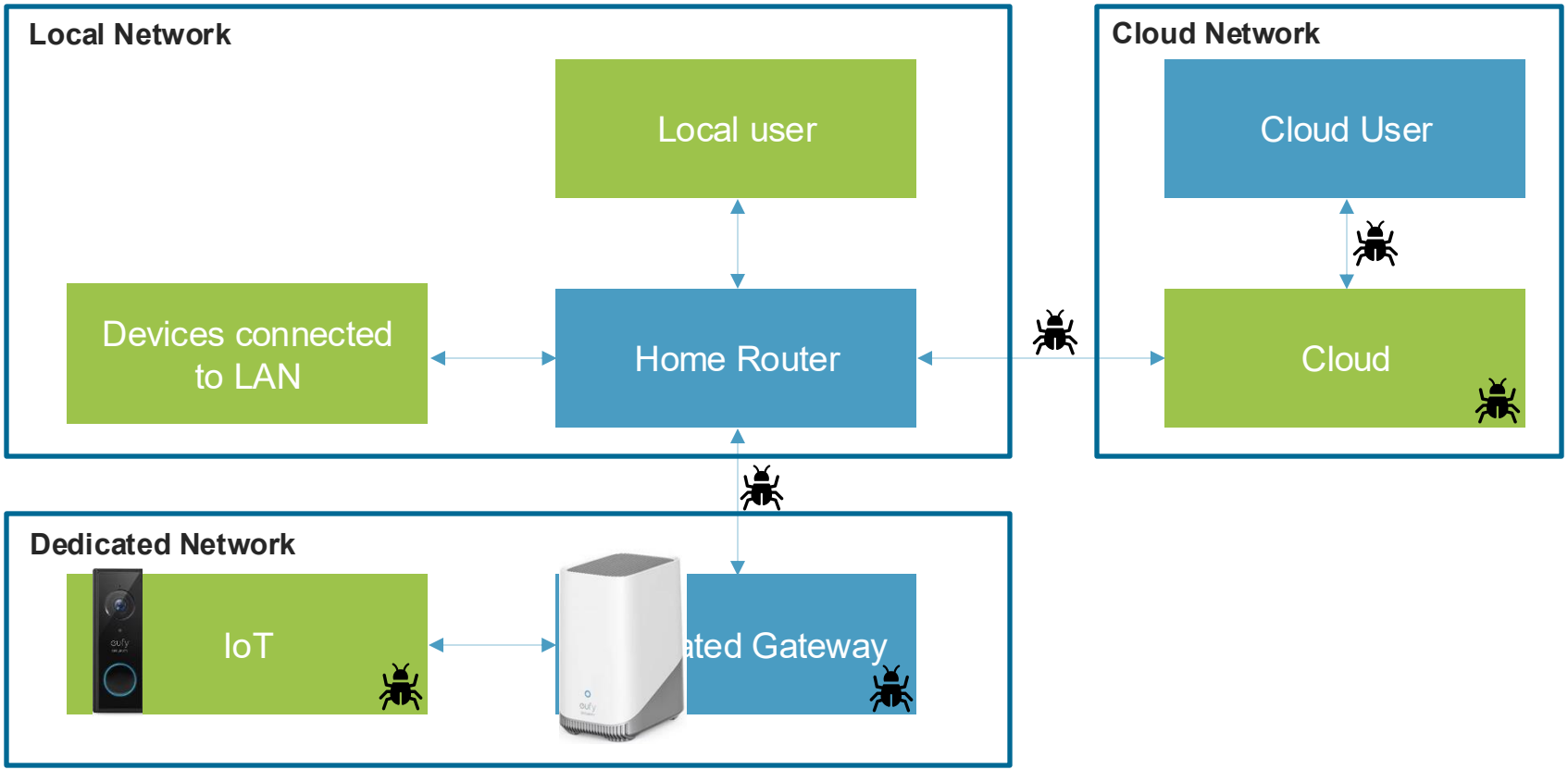
DistriNet

Eufy - Architecture



DistriNet

Eufy - Architecture



DistriNet

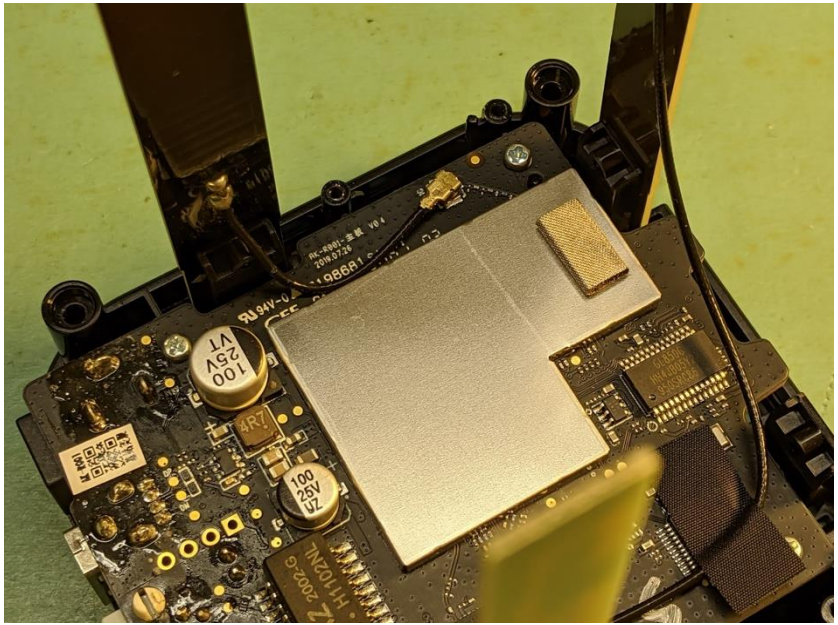
Eufy – Dedicated Network - Setup

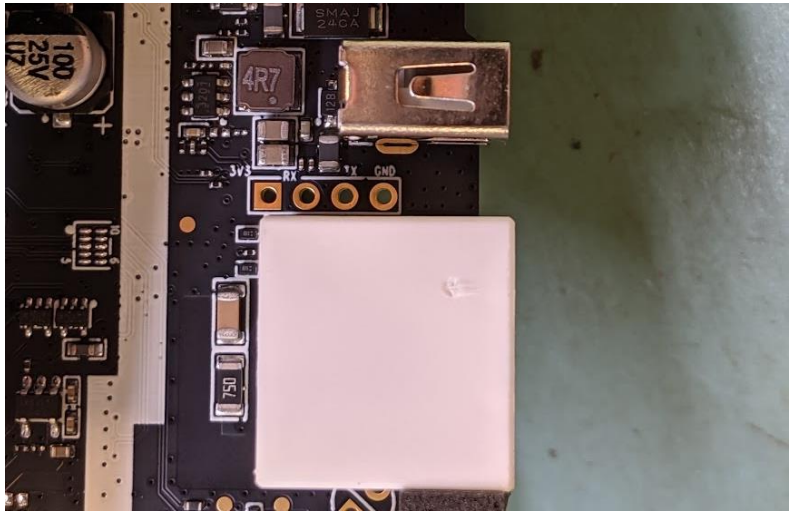
Methods

- Combination of techniques
  1. Network traffic analysis
  2. Firmware analysis
  3. Symbolic analysis

Cracking the WPA2-PSK

Eufy – Firmware acquisition





DistriNet

Eufy – Firmware acquisition

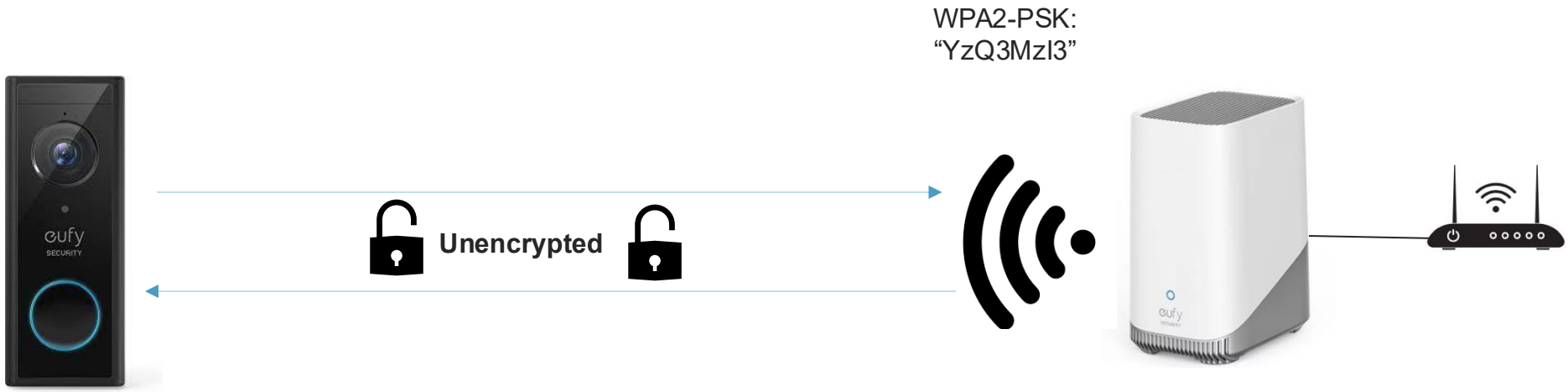
```
  ** send data to uart2 **
FE 01 21 58 00 78


 ++ read from uart2 data:
FE 02 61 58 00 10 2B

 success to get respance.
get response from app. code:0x00.
in normal mode.
  ** send data to uart2 **
FE 01 21 51 00 71


 ++ read from uart2 data:
FE 01 61 51 00 31

 success to get respance.


3: System Boot system code via Flash.
## Booting image at bc050000 ...
raspi_read: from:50000 len:40
   Image Name:   Linux Kernel Image
   Image Type:   MIPS Linux Kernel Image (lzma compressed)
   Data Size:    14616169 Bytes = 13.9 MB
   Load Address: 80000000
   Entry Point:  805e9490
raspi_read: from:50040 len:2906a9
```

```
normal.sh current_year:1970
normal.sh fail to get system time. ignore ...
[   50.488000] ####Set_SignalUserPid_Proc,5897

Login timed out process '/bin/login' (pid 5290) exited. Scheduling for restart.
setrlimit(RLIMIT_CORE, &limit);
starting pid 6382, tty '/dev/ttyS1': '/bin/login'
Eufycamera login:
```

```
 ##### The CPU freq = 575 MHZ ####
 estimate memory size =128 Mbytes
RESET MT7628 PHY!!!!!!
Please choose the operation:
   1: Load system code to SDRAM via TFTP.
   2: Load system code then write to Flash via TFTP.
   3: Boot system code via Flash (default).
   4: Entr boot command line interface.
   7: Load Boot Loader code then write to Flash via Serial.
   9: Load Boot Loader code then write to Flash via TFTP.

   m: Load mini system code then write to Flash via TFTP.
   u: enter mini system for upgrade normal system.
```

DistriNet

Eufy – Authentication bypass - CVE-2023-37822



Eufy – Filesystem analysis

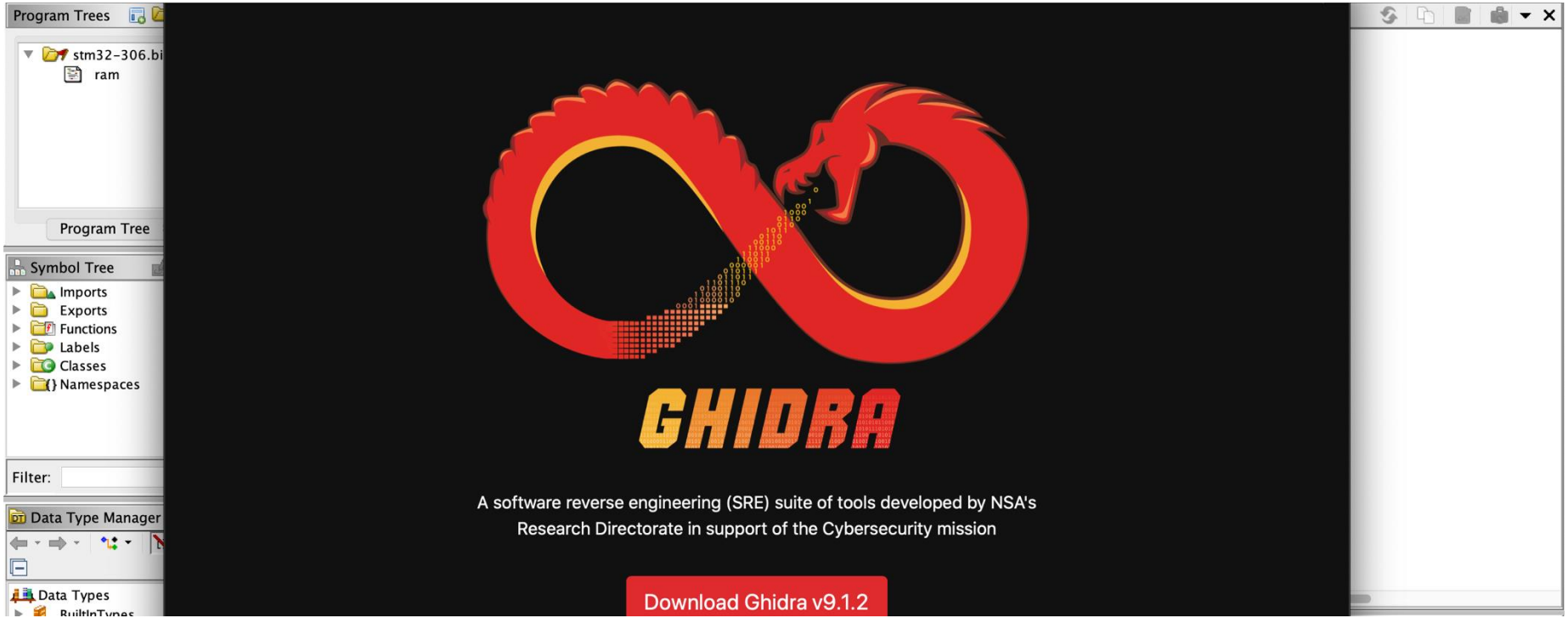
- Suspicious activities
  - Reuse of password
    - UART login
    - WPA2-PSK
  - Password follows a pattern

```
genSysFiles()
    #login=`nvram_get 2860 Login`
    #pass=`nvram_get 2860 Password`
    login="eufycamera"
    pass="`nvram_get WPAPSK1`"
    if [ "$pass" == "" ];then
        pass="Anker39#*"
    #login="admin"
    #pass="admin"
    if [ "$login" != "" -a "$pass" != "" ]; then
    echo "$login::0:0:Adminstrator:/:/bin/sh" > /etc/passwd
    echo "$login:x:0:$login" > /etc/group
        chpasswd.sh $login $pass
    if [ "$CONFIG_PPPOL2TP" == "y" ]; then
    echo "l2tp 1701/tcp l2f" > /etc/services
    echo "l2tp 1701/udp l2f" >> /etc/services
configVIF()
```

Eufy – Filesystem analysis

```
[ERROR][log.c:LOG_vprintf:463] - appClientInit OK
[ERROR][log.c:LOG_vprintf:463] - appClientInit OK, create main thread=0x22c6398!
[INFO][sub1g_interface.c:zx_init_sub1g:4952] - init_sub1g enter,use /etc_ro/appclient.cfg. ret=0
[INFO][sys_interface.c:zx_create_thread:1128] - stacksize = 262144 Byte
[INFO][sys_interface.c:zx_create_thread:1166] - tid:28701, thread_type:1 ok
[INFO][security_main.c:zx_watch_dog:1760] - zx_watch_dog, PID=5839----------
[INFO][security_main.c:zx_watch_dog:1772] - gRecord_pipe:11,12
[INFO][sub1g_interface.c:zx_get_gw_info_cnf:589] - Get Nwk Info Confirmed, Status=1 ,collector extAddr = 0x417b6659c5f,panID = 0x9c5f
[INFO][sub1g_interface.c:zx_get_gw_info_cnf:600] - using anker_mac: 0x4:17
[INFO][remote_system.c:zx_remote_system_accept_node:273] - cmd:/sbin/chpasswd.sh eufycamera YzQ3MzI3 need_response:0
[INFO][remote_system.c:zx_remote_system_cmd:104] - 000 alloc send cmdbuf:/sbin/chpasswd.sh eufycamera YzQ3MzI3, need_response:0, node:0x22d3db8
[INFO][xm_interface.c:zx_init_camera_sdk:5314] - enter......
[INFO][remote_system.c:zx_remote_system_send_server:178] - get socket:13
[INFO][remote_system.c:zx_remote_system_free_node:367] - 222 find delete node:0x22d3db8, cmdbuf:/sbin/chpasswd.sh eufycamera YzQ3MzI3, fd:13
[INFO][xm_interface.c:zx_init_camera_sdk:5358] - XMLIB VERSION: V-3.0.8.8  <==========
[INFO][xm_interface.c:zx_init_camera_sdk:5419] - xmsdk:0, mac:04:17:B6:66:45:C0, status:1, IP:192.168.32.5
[INFO][xm_interface.c:zx_init_camera_sdk:5419] - xmsdk:1, mac:, status:0, IP:
[INFO][xm_interface.c:zx_init_camera_sdk:5419] - xmsdk:2, mac:, status:0, IP:
[INFO][xm_interface.c:zx_init_camera_sdk:5419] - xmsdk:3, mac:, status:0, IP:
[INFO][xm_interface.c:zx_init_camera_sdk:5419] - xmsdk:4, mac:, status:0, IP:
[INFO][xm_interface.c:zx_init_camera_sdk:5419] - xmsdk:5, mac:, status:0, IP:
[INFO][xm_interface.c:zx_init_camera_sdk:5419] - xmsdk:6, mac:, status:0, IP:
[INFO][xm_interface.c:zx_init_camera_sdk:5419] - xmsdk:7, mac:, status:0, IP:
```

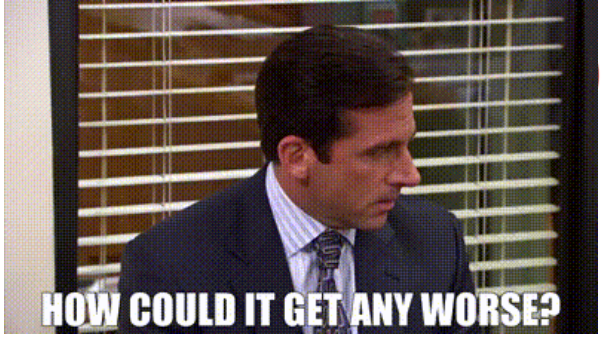
Eufy – Reverse engineering



DistriNet

Eufy – Reverse engineering

- WPA2-PSK directly linked to serial number?!




```
int set_wifi_pwd(char *serialNumber)

{
  char *md5HashSN_hex;
  size_t length;
  char *ActualPassword;
  int iVar1;
  char *passwordPointer;

  if (((serialNumber != (char *)0x0) && (*serialNumber != '\0')) &&
     (md5HashSN_hex = (char *)zx_MD5_HexStr(serialNumber), md5HashSN_hex != (char *)0x0)) {
    length = strlen(md5HashSN_hex);
    ActualPassword = (char *)zx_Memory_Encode(md5HashSN_hex,length);
    if (md5HashSN_hex != (char *)0x0) {
      free(md5HashSN_hex);
    }
    if (ActualPassword != (char *)0x0) {
      passwordPointer = (char *)0x0;
      strncpy((char *)&passwordPointer,ActualPassword,8);
      if (ActualPassword != (char *)0x0) {
        free(ActualPassword);
      }
      zx_do_system("/sbin/chpasswd.sh eufycamera %s",&passwordPointer);
      nvram_init(1);
      md5HashSN_hex = (char *)nvram_bufget(1,"WPAPSK1");
      if ((md5HashSN_hex != (char *)0x0) &&
         (iVar1 = strcmp((char *)&passwordPointer,md5HashSN_hex), iVar1 != 0)) {
        nvram_bufset(1,"WPAPSK1",&passwordPointer);
        nvram_commit(1);
        nvram_close(1);
        zx_do_system("iwpriv ra0 set WPAPSK=%s",&passwordPointer);
        dzlog("src/sys_interface.c",0x13,"set_wifi_pwd",0xc,0x22f,0x28,"set WIFI_PWD successed!");
        return 0;
      }
      nvram_close(1);
    }
  }
  return -1;
```

Eufy – Authentication Bypass

Serial number:
T8010P23224107B0

MD5 hash

c47327

Base64

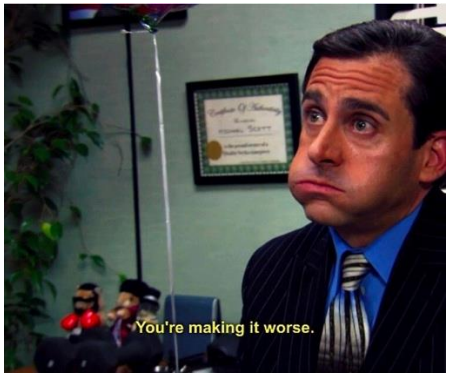
YzQ3MzI3

First 8 bytes

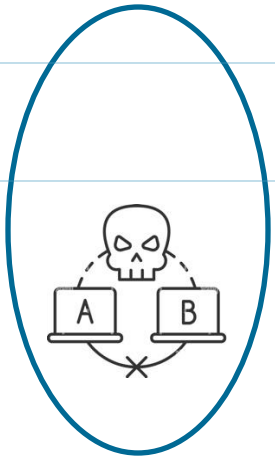
WPA2PSK:
“YzQ3MzI3”

Entropy? #possible combinations?

16^6 = 16.777.216



Eufy – Authentication bypass



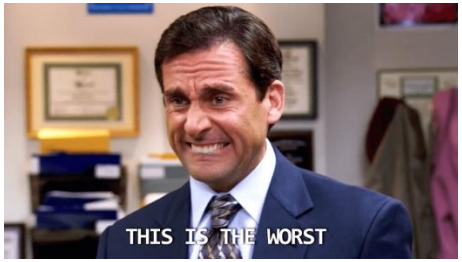
WPA2-PSK:
XXXXXXXX
“YzQ3MzI3”

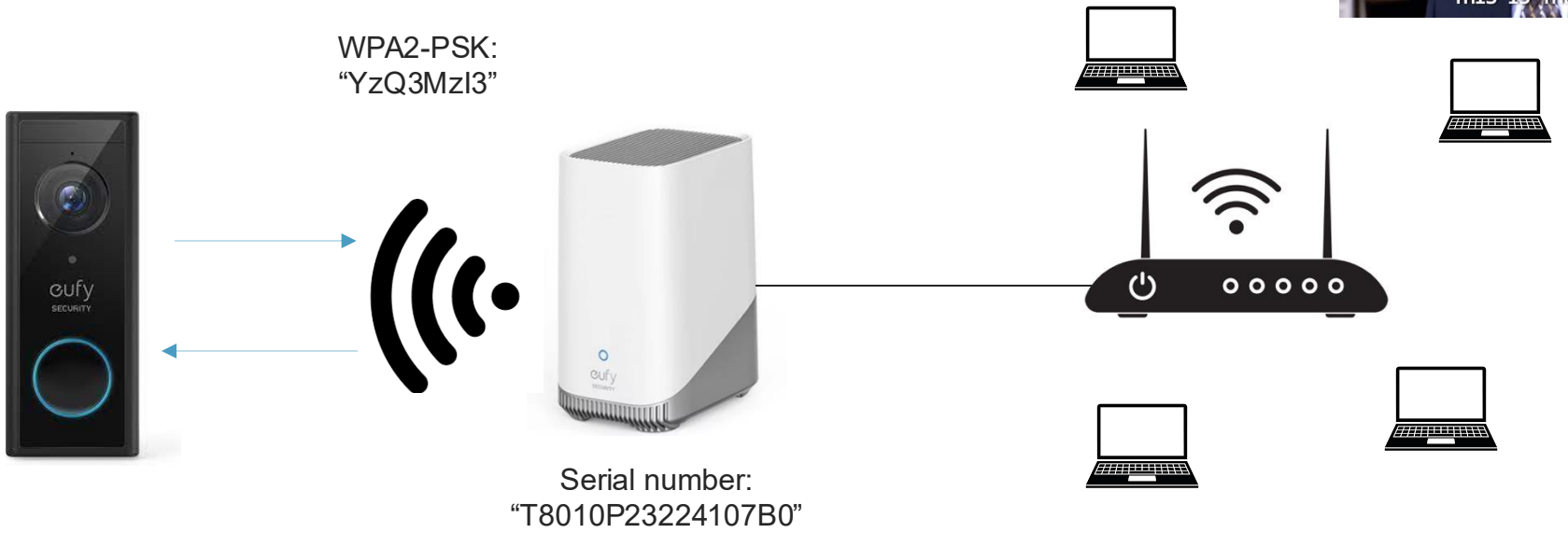
Serial number:
“T8010P23224107B0”

- Steal WPA Handshake
- Crack WPA2-PSK – Brute force
  - 20 seconds

DistriNet

Authentication Bypass - CVE-2023-37822

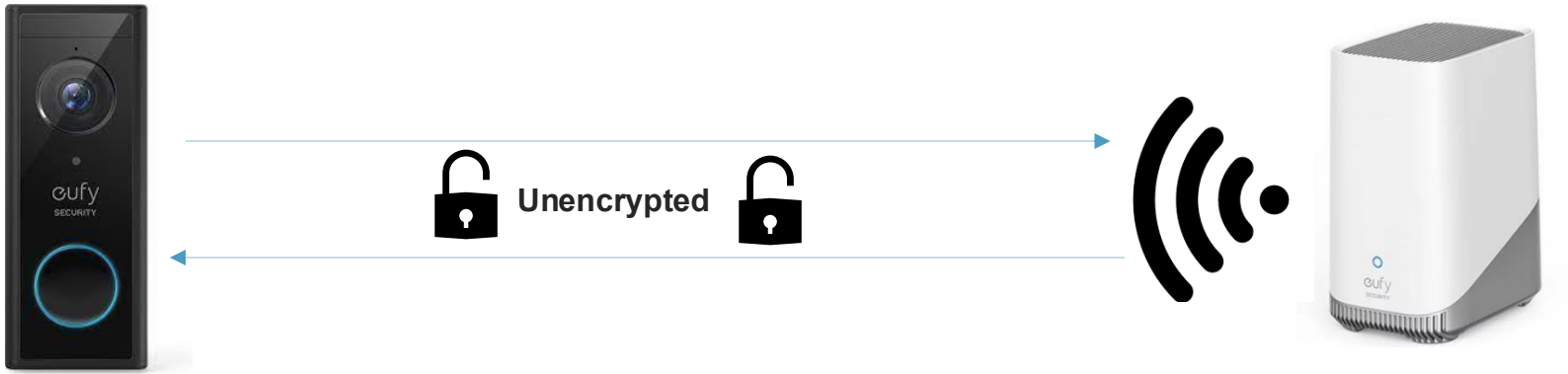




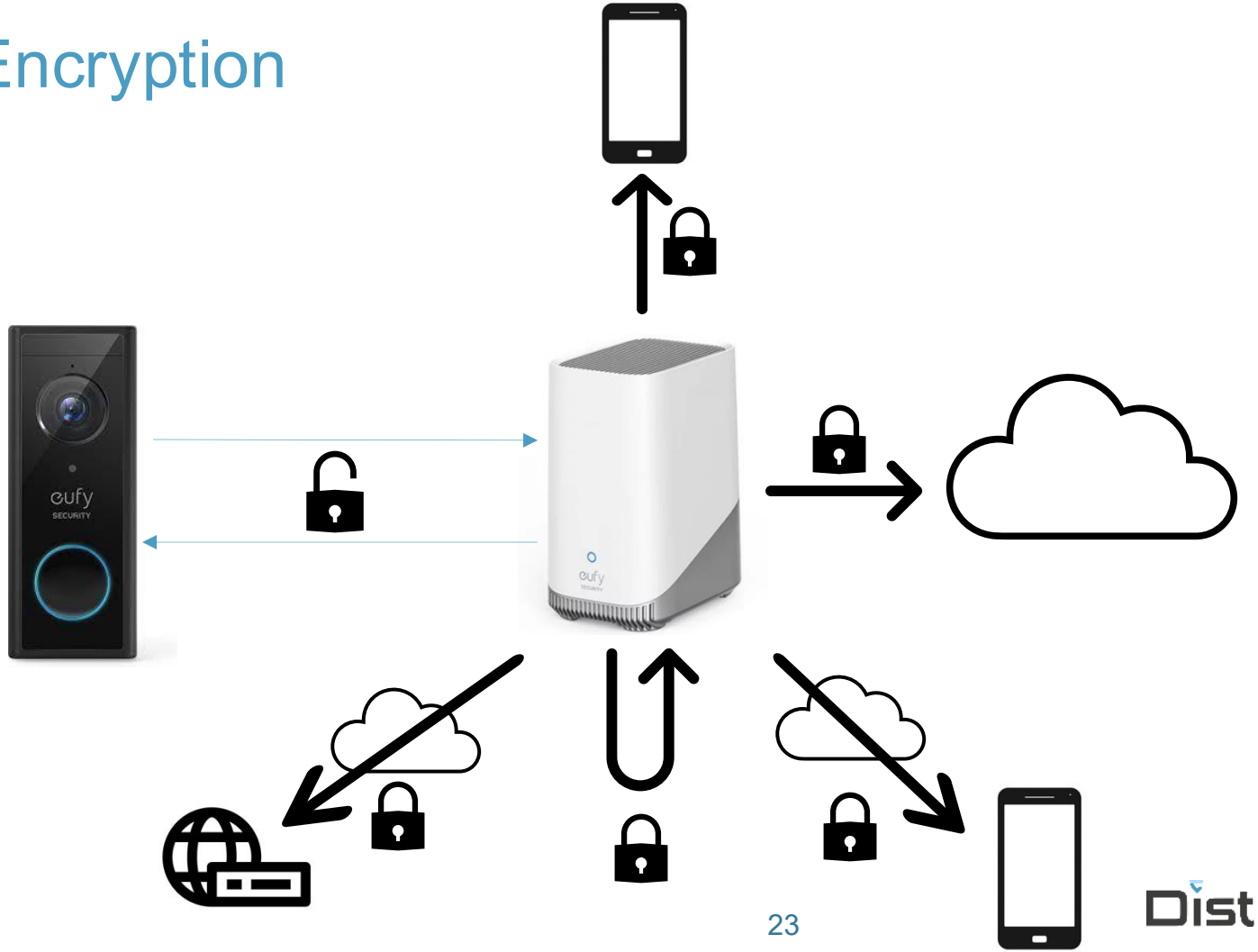
DistriNet

Decrypting the Video Stream

Eufy – Encryption

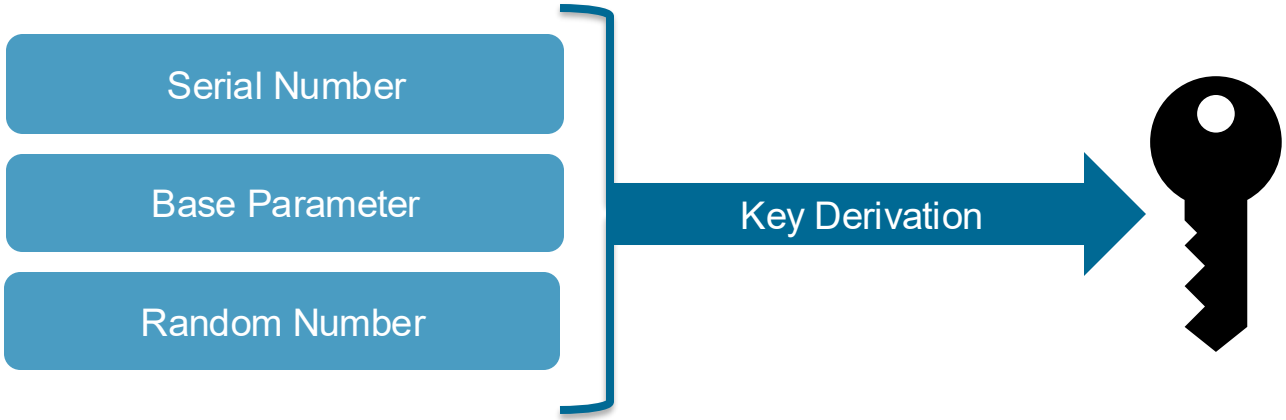


Eufy – Encryption



DistriNet

Eufy - Encryption



Eufy – Encryption

P2P Encryption

$$Key = baseParam[0:15] + serialNumber[9:15]$$

Media Encryption

```
1: function GENPICBASECODE(serialNumber, baseParam)
2:     PPCS_SFX = cal_ppcs_id_suffix(baseParam)  ▷ See Algorithm 2
3:     baseCode = serialNumber[0 : l] + str(PPCS_SFX)
4:     return baseCode
5: end function
6: function GENPICRANDSEED(baseParam)
7:     PPCS_SFX = cal_ppcs_id_suffix(baseParam)
8:     randomValue = random()
9:     randomNum = "01" + str(randomValue)
10:    tmpKey = str(1000 − PPCS_SFX) + randomNum
11:    seed = Obfuscate1(MD5(tmpKey))
12:    return (seed, randomValue)
13: end function
14: function GENCHECKCODE(baseCode, seed)
15:    tmpKey = "01" + baseCode + seed
16:    tmpKey2 = SHA256(tmpKey)
17:    encKey = Obfuscate2(tmpKey2)
18:    return encKey
19: end function
```

Eufy – Encryption

- Manual reverse engineering
  - Complex
  - Fault prone
  - Time consuming

Selective execution

```
var C7 = M7wd_;
var s2 = [arguments];
s2[6] = C7.U1()[0][5];
C7.P1();
for (; s2[6] !== C7.q8()[26][3];) {
  switch (s2[6]) {
    case C7.U1()[3][6]:
      s2[9][C7.F(4)]((function () {
        C7.P1();
        var z2 = [arguments];
        z2[6] = C7.q8()[25][8];
        for (; z2[6] !== C7.U1()[1][7];) {
          switch (z2[6]) {
            case C7.U1()[31][14]:
```

```
      case C7.q8()[18][23]:
        s2[9] = [];
        s2[6] = C7.U1()[28][19];
        break;
      case C7.U1()[8][16]:
        s2[4] = s2[0][0][s2[5]];
        s2[3] = V0oXS$(s2[4]);
        s2[6] = C7.q8()[15][18];
        break;
      case C7.U1()[21][8]:}
}
```

Eufy - Encryption

Selective execution

- Objective: Reconstruct AES key
  - Run cross-architecture binary
  - Run targeted functions
  - Run with concrete inputs
  - Avoid complex setup

Eufy - Encryption

Selective execution

	Debugger (GDB)	Emulation (QEMU)
Targeted Execution	✓	
Cross-architecture		✓
Avoid Complexity	✓	

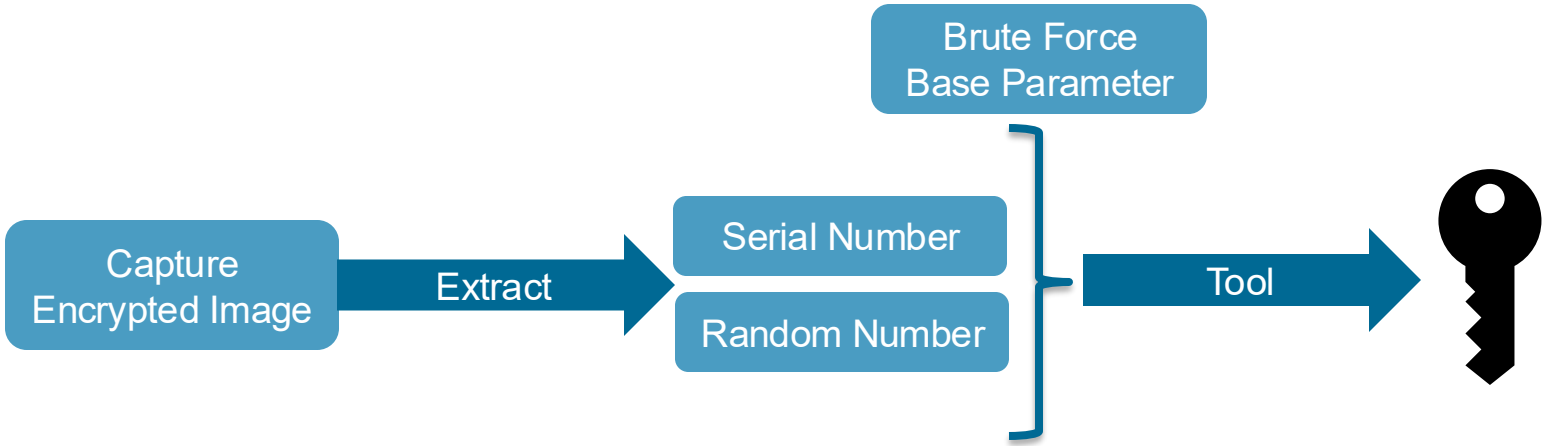
Eufy - Encryption

Selective execution

	Debugger (GDB)	Emulation (QEMU)	Our Tool
Targeted Execution	✓		✓
Cross-architecture		✓	✓
Avoids Complexity	✓		✓

- Our Tool
  - Combines **Binary lifting** and **Debugging**
  - = Cross-architecture symbolic debugger

Eufy - Encryption



Reflection

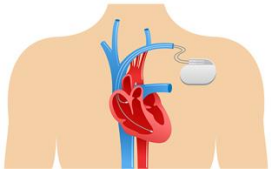
Reflection

- Objective of an attack on an IoT device
  1. Corrupt IoT device
  2. Pivot into device network
  3. Integrate into botnet

Reflection

eufy

- Objective of an attack on an IoT device
  1. **Corrupt IoT device**
  2. Pivot into device network
  3. Integrate into botnet



DistriNet

Reflection

- Objective of an attack on an IoT device
  1. Corrupt IoT device
  2. **Pivot into device network**
  3. Integrate into botnet

Reflection

- Objective of an attack on an IoT device
  1. Corrupt IoT device
  2. Pivot into device network
  3. **Integrate into botnet**

Reflection

- Why this case study
  - Towards industry
    - What vulnerabilities are prevalent
    - Preventing vulnerabilities
  - Towards research
    - Uncovering advanced vulnerabilities
    - Automation

Read the Paper!

DistriNet

Thank you!