ITOCHU
Cyber &
Intelligence

# Into the Vapor to Tracking Down Unknown Panda's Claw Marks


Suguru_Ishimaru
Yusuke_Niwa
Motohiko_sato

# Who are we?

Company: ITOCHU Cyber & Intelligence Inc.


Suguru Ishimaru

Sr. Cybersecurity Researcher
Botconf2019
Presentation, Workshop


Yusuke Niwa

Lead Cybersecurity Researcher
VB2023, JSAC etc.

Motohiko Sato

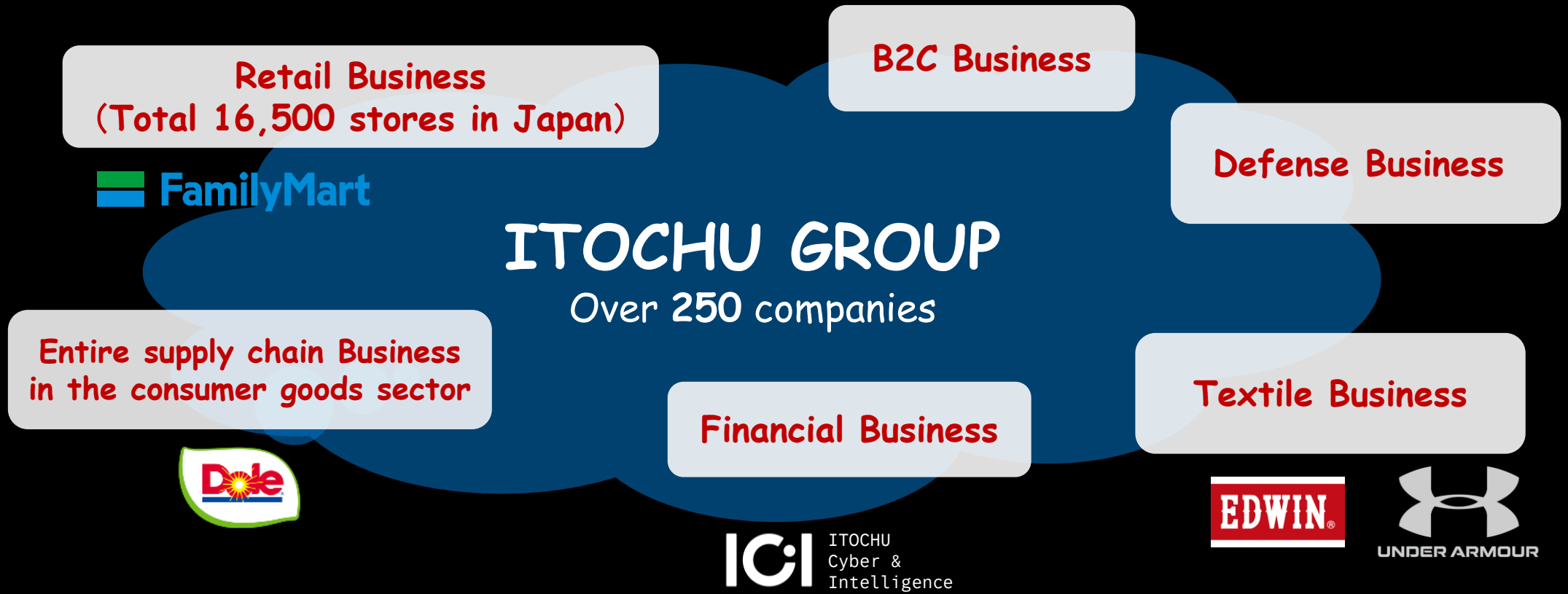
Exec. Cybersecurity Researcher
World Class Sinkiholer

X: @58_158_177_102

ITOCHU Cyber & Intelligence

# About ICI (ITOCHU Cyber & Intelligence Inc.)

- ICI is a BlueTeam company to protect ITOCHU Group's business
- **ITOCHU** is a conglomerate company with a wide range of businesses.



# Agenda

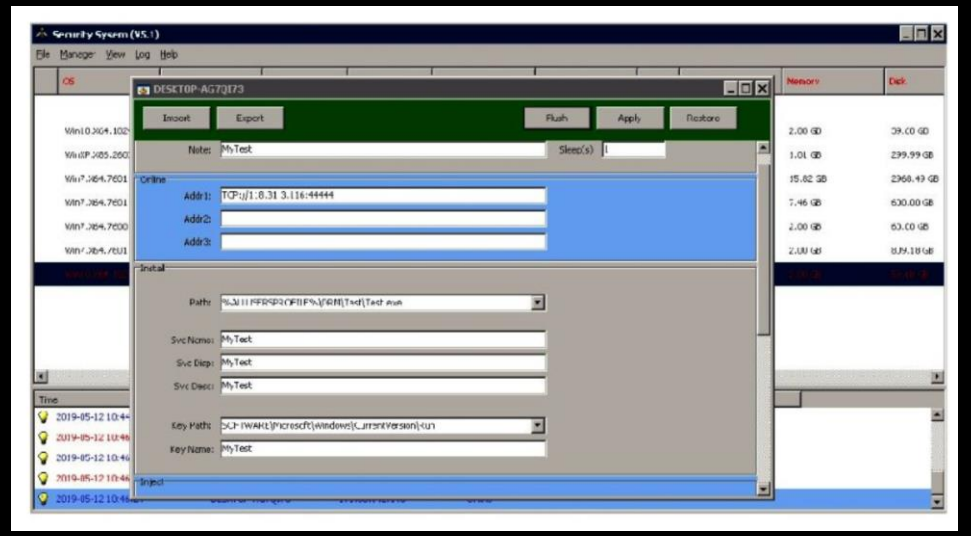
1. Do you know Shadowpad/DeedRAT?
2. Discovered Unknown Claw marks
3. Analyzing BloodAlchemy
4. Code Similarities
5. Infrastructure
6. Conclusions

# Do you know ShadowPad?

# About ShadowPad?

- ShadowPad is a modular Remote Access Trojan (RAT) discovered in 2017 for supply chain cyberattacks.

- It is also known to be shared by several APT actors such as APT41 (a.k.a. Winnti), Tick, Tonto team and TropicTrooper.


https://www.sentinelone.com/labs/shadowpad-a-masterpiece-of-privately-sold-malware-in-chinese-espionage/

# How about DeedRAT?

# What is DeedRAT?

- DeedRAT is not as well-known as ShadowPad.
- This malware family was first discovered around 2019 by Positive Technologies and is said to be a successor malware to ShadowPad.
- At present, its usage has only been observed by a specific threat actor group which is called "Space Pirate" group.

https://www.ptsecurity.com/ww-en/analytics/pt-esc-threat-intelligence/space-pirates-a-look-into-the-group-s-unconventional-techniques-new-attack-vectors-and-tools/#id2-1

# Timeline of Public Incident Case with ShadowPad and DeedRAT

## ShadowPad

- July 2017 APT41 NetSarang supply chain attack
- January 2019 APT41 ASUS LiveUpdate supply chain attack
- Late 2019 Tick Chemicals in Japan
- January 2020 Tonto Team Abusing Trend Micro's vulnerabilities
- July 2020 LuckyMouse Supply chain attack on Able Desktop
- August 2020 Tonto Team Manufacturing in Japan
- June 2021 Earth Lusca Research Institute etc. in Taiwan
- September 2022 Unkown Supply chain attack on gov's E-Office in Pakistan
- February, May 2023 APT41 The entire national grid in Asia


## DeedRAT

- Late 2019 Space Pirates A Russian aerospace company
- Summer 2021 Space Pirates A Russian aerospace company
- July 2023 Space Pirates Government etc. in Russia and Serbia

ITOCHU Cyber & Intelligence

# Why mentioned ShadowPad and DeedRAT?

- "Unknown RAT" file set was discovered through one of our research.
- The newly discovered unknown RAT was turn out to be **BloodAlchemy**.

***NEW Findings***

- The detailed function of BloodAlchemy by in-depth analysis.
- We believe it has a strong correlation with DeedRAT.
  - ShadowPad -> DeedRAT -> BloodAlchemy

https://www.elastic.co/security-labs/disclosing-the-bloodalchemy-backdoor

# Timeline of Public Incident Case with ShadowPad and DeedRAT

## ShadowPad

July 2017 APT41
NetSarang supply chain attack

January 2019 APT41
ASUS LiveUpdate supply chain attack

Late 2019 Tick
Chemicals in Japan

January 2020 Tonto Team
Abusing Trend Micro's vulnerabilities

July 2020 LuckyMouse
Supply chain attack on Able Desktop

August 202
Manufactu

June 2021 Earth Lusca
Research Institute etc. in Taiwan

September 2022 Unkown
Supply chain attack on gov's E-Office
in Pakistan

We observed the existence of
BloodAlchemy in **May 2023**

2017 2019 2020 2021 2022 2023

## DeedRAT

Late 2019 Space Pirates
A Russian aerospace company

Summer 2021 Space Pirates
A Russian aerospace company

July 2023 Space Pirates
Government etc. in Russia and Serbia

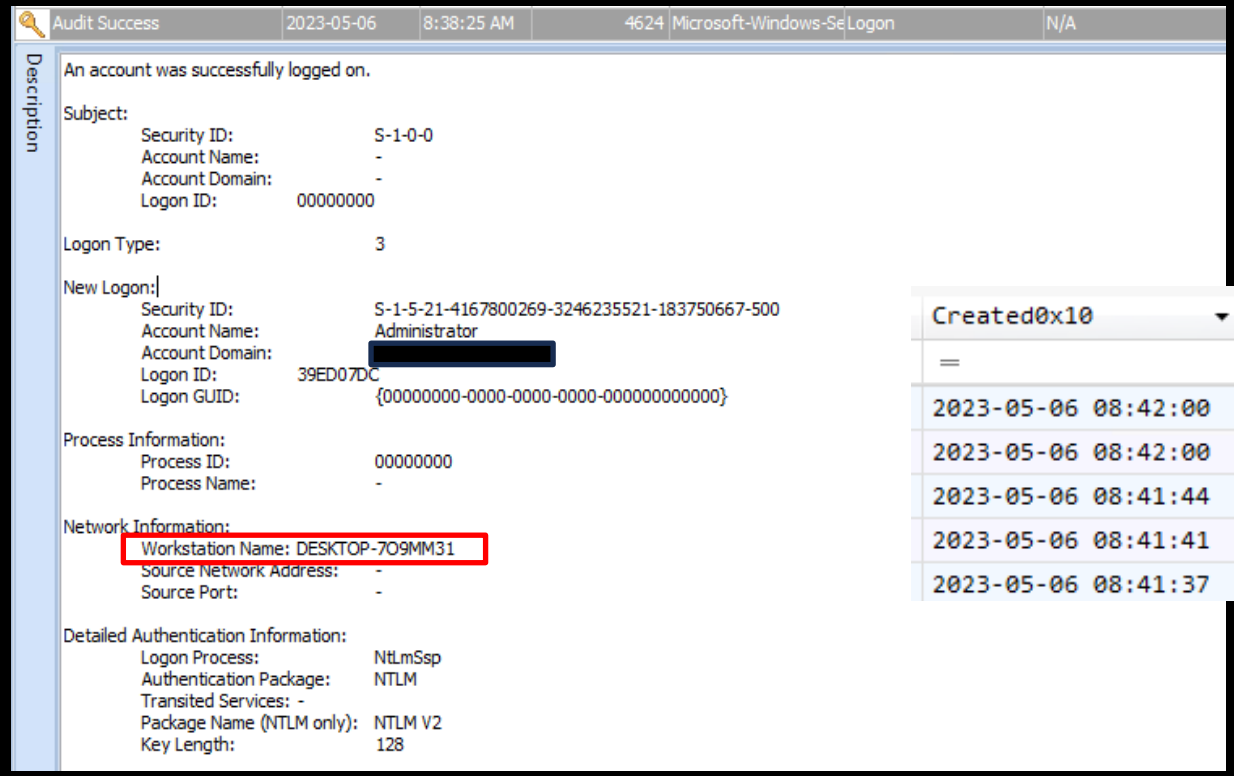
## BloodAlchemy

October 2023 BloodAlchemy / Unkown
Compromised ASEAN member countries and Mongolia's Ministry of Foreign Affairs

Cyber &
Intelligence

# How discovered Unknown Malware (=BloodAlchemy) ?

- Once unauthorized access by unknown host, unknown file set was created and executed on the one of the compromised host.



Audit Success | 2023-05-06 | 8:38:25 AM | 4624 | Microsoft-Windows-Se | Logon | N/A

Description

An account was successfully logged on.

Subject:
- Security ID: S-1-0-0
- Account Name: -
- Account Domain: -
- Logon ID: 00000000

Logon Type: 3

New Logon:
- Security ID: S-1-5-21-4167800269-3246235521-183750667-500
- Account Name: Administrator
- Account Domain: [redacted]
- Logon ID: 39ED07DC
- Logon GUID: {00000000-0000-0000-0000-000000000000}

Process Information:
- Process ID: 00000000
- Process Name: -

Network Information:
- Workstation Name: DESKTOP-7O9MM31
- Source Network Address: -
- Source Port: -

Detailed Authentication Information:
- Logon Process: NtLmSsp
- Authentication Package: NTLM
- Transited Services: -
- Package Name (NTLM only): NTLM V2
- Key Length: 128

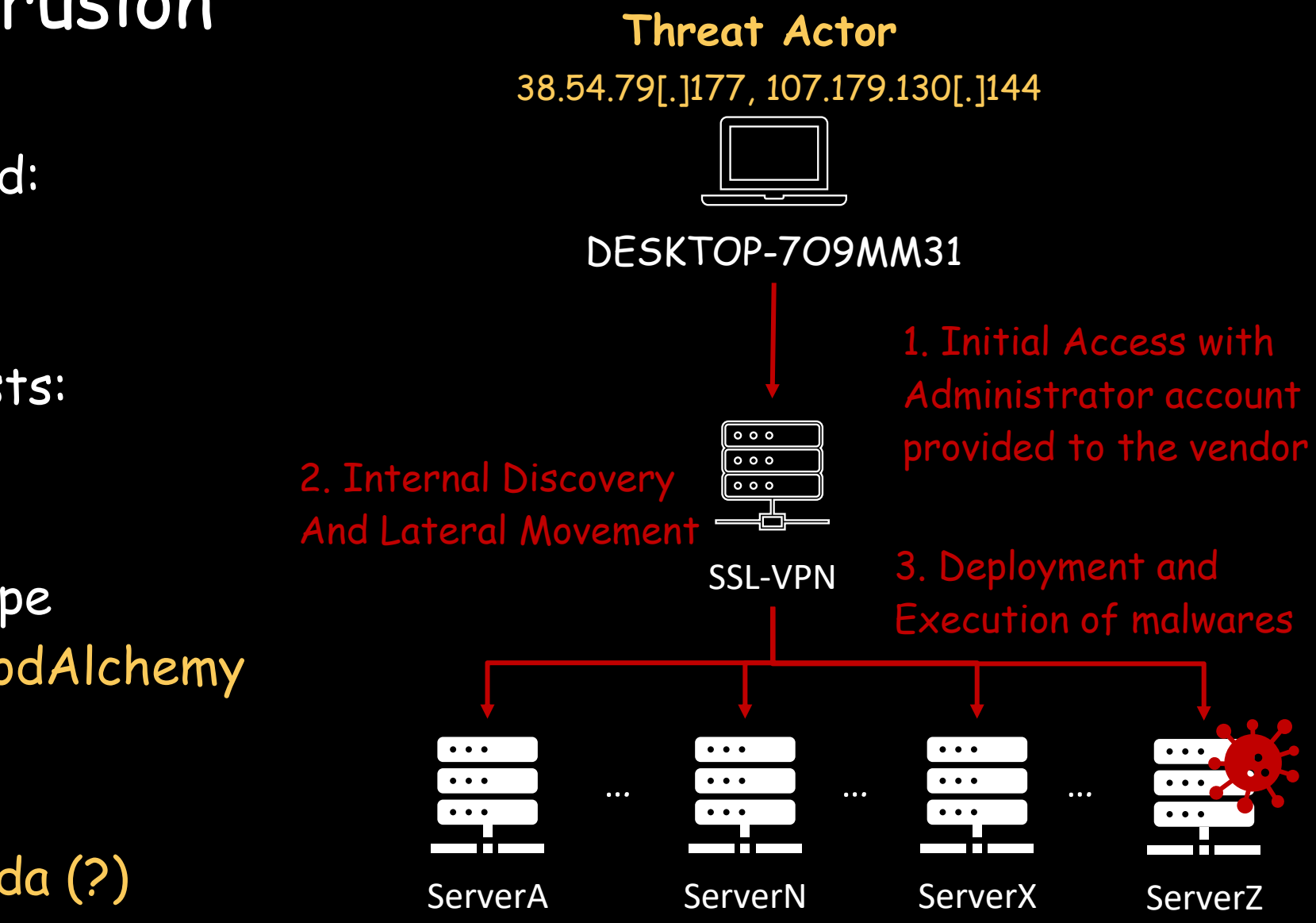
| Created0x10 | Create… | File Size | Parent Path | File Name |
|---|---|---|---|---|
| 2023-05-06 08:42:00 | | 3220 | .\Windows\System32\Tasks\Dell | BrDifxapi |
| 2023-05-06 08:42:00 | | 0 | .\Windows\System32\Tasks | Dell |
| 2023-05-06 08:41:44 | | 66112 | .\Windows | DIFX |
| 2023-05-06 08:41:41 | | 129536 | .\Windows | BrLogAPI.dll |
| 2023-05-06 08:41:37 | | 111472 | .\Windows | BrDifxapi.exe |

# Infection Vector

- Connected via RDP from IP address without abusing any vulnerability.
  - The account is "legitimate", but unfortunately, the access was unauthorized.
- The VPN account was only used for system maintenance by a vender
  - The vendor had inexplicably disabled MFA only for the maintenance account.
- The source IP address of VPN connections were
  38.54.79[.]177(HK) and 107.179.130[.]144 (CA)

# Overview of Intrusion

- The compromised found:
  May 2023
- Total compromised hosts:
  8 hosts
- Discovered malware type
  CobaltStrike and BloodAlchemy
- Threat Actor
  Suspected Vapor Panda (?)



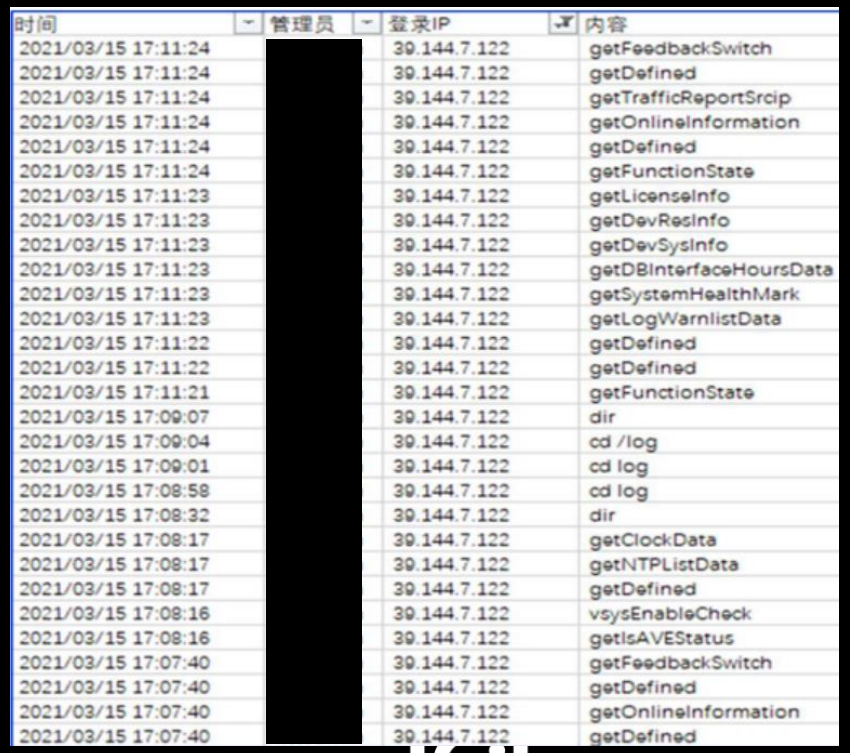

ITOCHU Cyber & Intelligence

*A part of period in this incident

# Additional Findings

Although the root cause of the credential leak remains unclear…

- Unauthorized access has been ongoing **since 2021**
- Attacker had access to credentials for other several accounts
- Attacker also stole VPN configuration information on the console.



| 时间 | 管理员 | 登录IP | 内容 |
|---|---|---|---|
| 2021/03/15 17:11:24 | [redacted] | 39.144.7.122 | getFeedbackSwitch |
| 2021/03/15 17:11:24 | | 39.144.7.122 | getDefined |
| 2021/03/15 17:11:24 | | 39.144.7.122 | getTrafficReportSrcip |
| 2021/03/15 17:11:24 | | 39.144.7.122 | getOnlineInformation |
| 2021/03/15 17:11:24 | | 39.144.7.122 | getDefined |
| 2021/03/15 17:11:24 | | 39.144.7.122 | getFunctionState |
| 2021/03/15 17:11:23 | | 39.144.7.122 | getLicenseInfo |
| 2021/03/15 17:11:23 | | 39.144.7.122 | getDevResInfo |
| 2021/03/15 17:11:23 | | 39.144.7.122 | getDevSysInfo |
| 2021/03/15 17:11:23 | | 39.144.7.122 | getDBInterfaceHoursData |
| 2021/03/15 17:11:23 | | 39.144.7.122 | getSystemHealthMark |
| 2021/03/15 17:11:23 | | 39.144.7.122 | getLogWarnlistData |
| 2021/03/15 17:11:22 | | 39.144.7.122 | getDefined |
| 2021/03/15 17:11:22 | | 39.144.7.122 | getDefined |
| 2021/03/15 17:11:21 | | 39.144.7.122 | getFunctionState |
| 2021/03/15 17:09:07 | | 39.144.7.122 | dir |
| 2021/03/15 17:09:04 | | 39.144.7.122 | cd /log |
| 2021/03/15 17:09:01 | | 39.144.7.122 | cd log |
| 2021/03/15 17:08:58 | | 39.144.7.122 | cd log |
| 2021/03/15 17:08:32 | | 39.144.7.122 | dir |
| 2021/03/15 17:08:17 | | 39.144.7.122 | getClockData |
| 2021/03/15 17:08:17 | | 39.144.7.122 | getNTPListData |
| 2021/03/15 17:08:17 | | 39.144.7.122 | getDefined |
| 2021/03/15 17:08:16 | | 39.144.7.122 | vsysEnableCheck |
| 2021/03/15 17:08:16 | | 39.144.7.122 | getIsAVEStatus |
| 2021/03/15 17:07:40 | | 39.144.7.122 | getFeedbackSwitch |
| 2021/03/15 17:07:40 | | 39.144.7.122 | getDefined |
| 2021/03/15 17:07:40 | | 39.144.7.122 | getOnlineInformation |
| 2021/03/15 17:07:40 | | 39.144.7.122 | getDefined |

Cyber & Intelligence

*A part of period in this incident

# Discovered 3 Malware Samples

| | Sample1 | Sample2 | Sample3 |
|---|---|---|---|
| **Malware Type** | Cobalt Strike | Cobalt Strike | BloodAlchemy |
| **Infection Vector** | Via SSL-VPN with legitimate vender account | | |
| **Artifacts** | No configuration in memory | Memory, malware set (sfc.exe, dxgi.dll, dxgi.cfg) | Memory, malware set (BrDifxapi.exe, BrLogAPI.dll DIFX) |
| **Config Info** | | | |
| C2Server | 121.41.35[.]65,/_/scs/mail-static/_/js/ | cdn39a700bb.jptomorrow[.]com,/search | cdn1ac7bdd3.jptomorrow[.]com,/search |
| Port | 9192 | 443 | 443 |
| Watermark | 426352781 | 2029527128 | - |

ITOCHU Cyber & Intelligence

# Before diving into an analysis in BloodAlchemy...

- We discovered an unidentified malware in this compromised case. This malware, known as BloodAlchemy, is a RAT.
- We believe BloodAlchemy is successor to ShadowPad/DeedRAT.

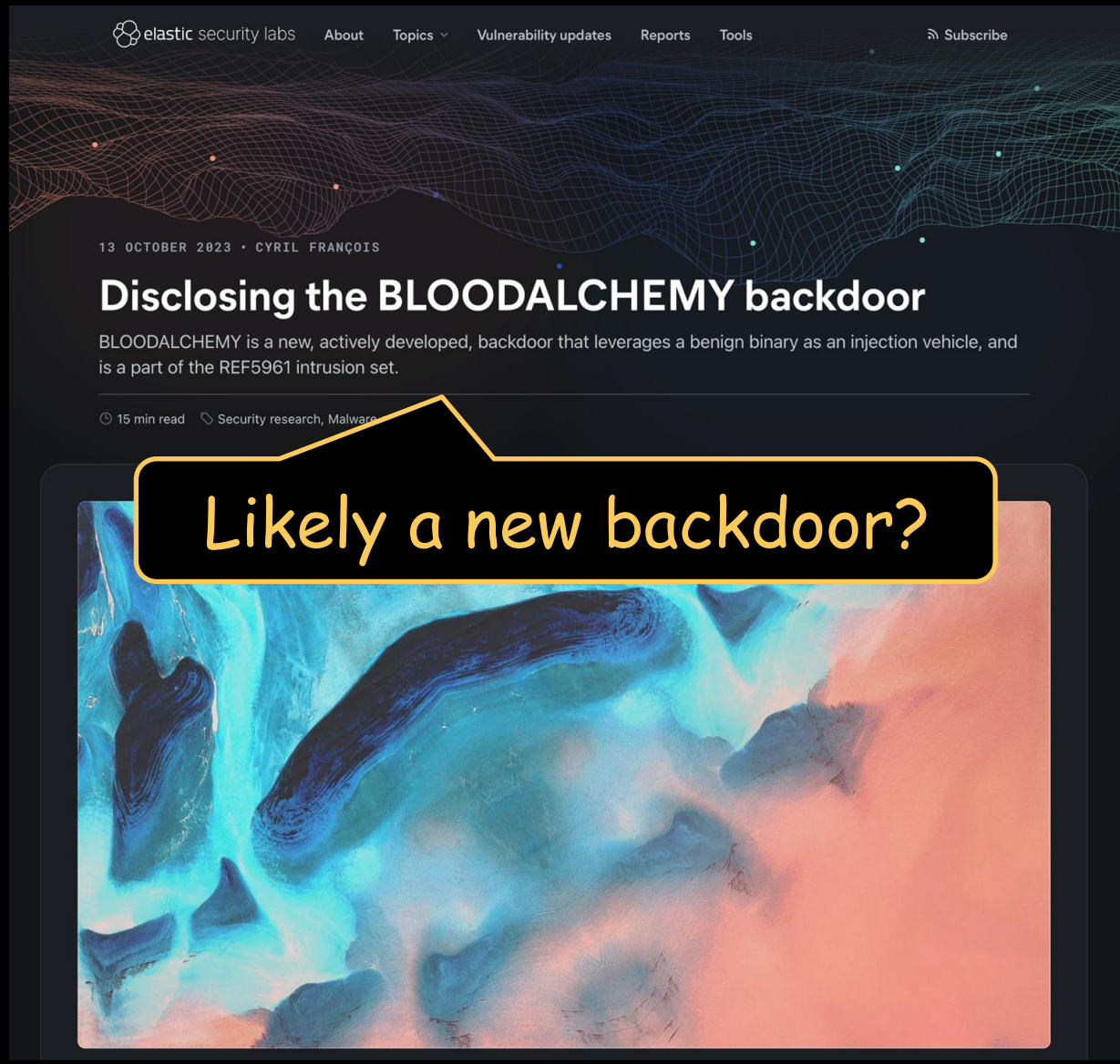
ITOCHU Cyber & Intelligence

What's NEXT?? »

# Analyzing BloodAlchemy

# BloodAlchemy

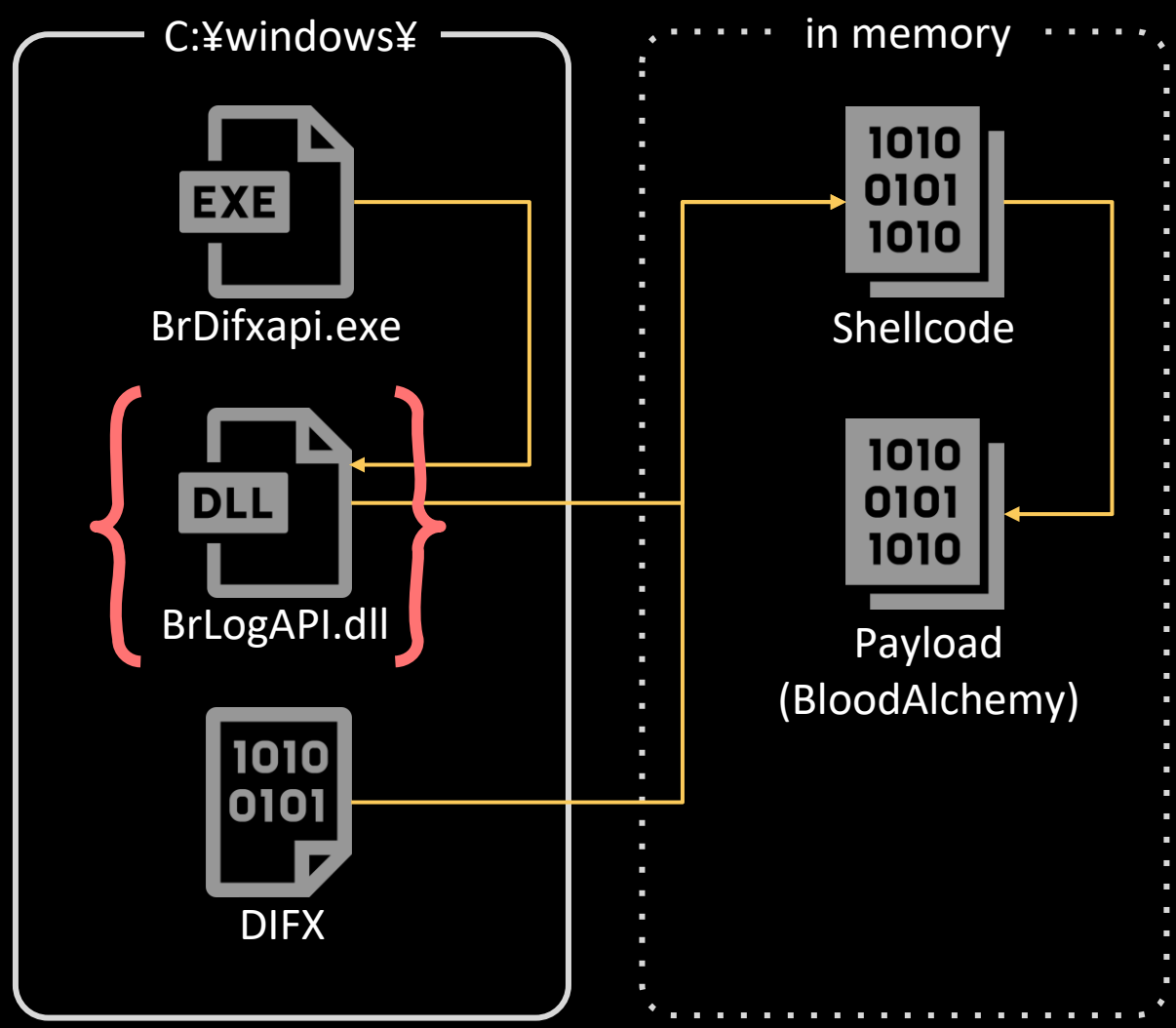
- In October 2023, the elastic security labs published about the BLOODALCHEMY backdoor analysis.
- A very sophisticated modular types of fileless malware like ShadowPad.
- Code and design are similar to Deed RAT and ShadowPad.




https://www.elastic.co/security-labs/disclosing-the-bloodalchemy-backdoor

# Infection Flow

- BrDifxapi.exe loads BrLogAPI.dll using side-loading
- BrLogAPI.dll reads DFIX as a BLOB and decrypts Shellcode
- Shellcode decrypts Payload from an embedded data in itself
- Payload infects in memory




ITOCHU Cyber & Intelligence

# BloodAlchemy: BrLogAPI.dll

- BrLogAPI.dll is loader module
- Reads a BLOB file (DIFX)
- Decrypts the BLOB using AES128 CBC mode with aes_key: BLOB[:0x10] to next Shellcode

```
43    strcpy(String2, "DIFX");
44    lstrcatA(Filename, String2);
45    FileA = CreateFileA(Filename, 0x80000000, 1u, 0, 3u, 0, 0);
46    hFile = FileA;
47    v15 = FileA;
48    if ( FileA == -1
49      || (FileSize = GetFileSize(FileA, 0), nNumberOfBytesToRead = FileSize, Fil
50      || (dec_shellcode = VirtualAlloc(0, FileSize, 0x3000u, 4u)) == 0 )
51    {
52      LastError = GetLastError();
53    }
54    else
55    {
56      NumberOfBytesRead = 0;
57      if ( ReadFile(hFile, dec_shellcode, nNumberOfBytesToRead, &NumberOfBytesRe
58      {
59        enc_data = NumberOfBytesRead;
60        v17[72] = 0;
61        v19 = 0i64;
62        aes_init(v8, &v19);
63        enc_data -= 16;
64        aes_dec(dec_shellcode + 16, enc_data);
65        v10 = enc_data - dec_shellcode[enc_data - 1];
66        v17[0] = 0;
67        NumberOfBytesRead = v10;
68        ModuleHandleA = GetModuleHandleA("kernel32.dll");
69        VirtualProtect = GetProcAddress(ModuleHandleA, "VirtualProtect");
70        if ( VirtualProtect(dec_shellcode, 4096, 32, v17) )
71          LastError = (dec_shellcode)(0);
72        else
```

# BloodAlchemy: BrLogAPI.dll

- BrLogAPI.dll is loader module
- Reads a BLOB file (DIFX)
- Decrypts the BLOB using AES128 CBC mode with aes_key: BLOB[:0x10] to next Shellcode

before

```
0000000000: 46 5C 45 00 7A 66 C4 DC | DD C9 27 A8 26 8B C6 26
0000000010: 61 36 8A FA FA D4 56 91 | 48 88 35 DD 07 82 43 6D
0000000020: 76 EC 43 12 6A 13 28 F8 | 4A 4F 63 F4 20 20 FA 4C
0000000030: 56 BA 85 63 27 95 85 23 | 8A F9 D5 61 DE F5 99 48
0000000040: B5 2B 4A CD 95 58 05 69 | 88 86 AC E2 C4 BB 7A D2
0000000050: 4F DB A2 30 FC EF F4 B3 | 62 83 5D E0 71 79 67 12
0000000060: D5 F6 B5 C1 94 29 B3 65 | BA F3 B3 E7 76 29 E9 44
0000000070: 31 7F 82 6F F6 A1 DC F3 | 6E A5 47 6B A1 7A 7C 29
0000000080: 5B BE ED 9C 50 47 6E F5 | 8B 34 F7 8D D5 D0 96 40
0000000090: A8 AE 51 91 19 39 9E 9C | 71 45 2F 42 26 27 3B CB
```

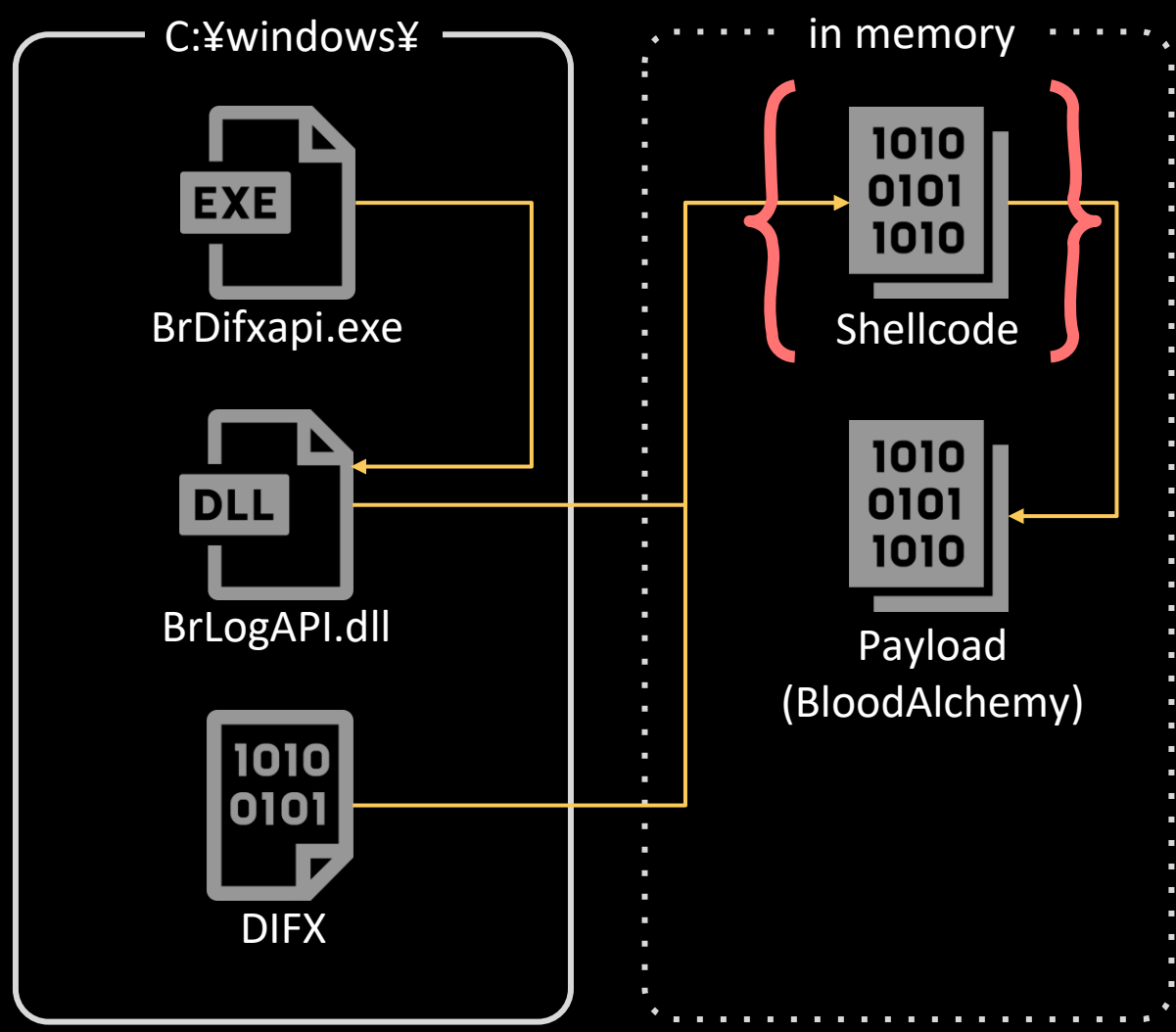
after

```
0000000000: E8 00 00 00 00 58 8D 40 | FB 8B 54 24 04 68 3A 04
0000000010: 00 00 68 74 F8 00 00 68 | 75 05 00 00 50 52 E8 03
0000000020: 00 00 00 C2 04 00 55 8B | EC 83 EC 34 33 C0 53 8B
0000000030: D8 89 45 E4 89 45 E0 89 | 45 DC 89 45 D8 89 45 E8
0000000040: 89 45 D4 64 A1 30 00 00 | 00 56 57 89 5D FC 8B 40
0000000050: 0C 8B 78 14 E9 F3 [illegible] 47 28 33 DB 89 45
0000000060: F4 8B CB 8B F0 8A 00 C1 | C9 07 0F B6 D0 3C 61 72
0000000070: 03 83 C1 E0 8D 46 02 03 | CA 89 45 F4 8B F0 66 39
0000000080: 18 75 E2 8B 5D FC 81 F9 | 18 42 8C 22 0F 85 EC 00
0000000090: 00 00 8B 77 10 C7 45 F0 | 05 00 00 00 8B 46 3C 8B
```

Shellcode

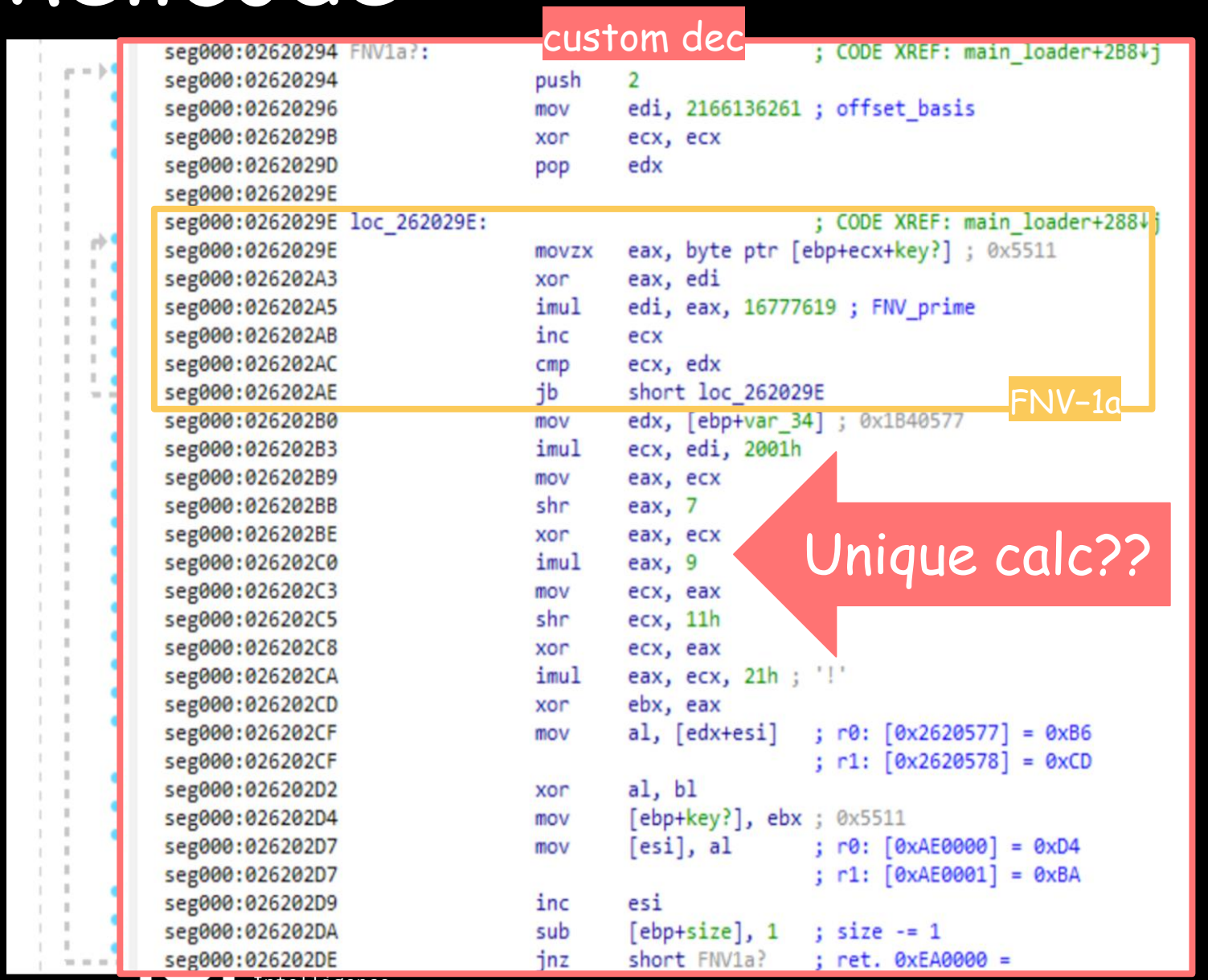
# Infection Flow

- BrDifxapi.exe loads BrLogAPI.dll using side-loading
- BrLogAPI.dll reads DFIX as a BLOB and decrypts Shellcode
- Shellcode decrypts Payload from an embedded data in itself
- Payload infects in memory



# BloodAlchemy: Shellcode

- Shellcode also loader module in memory
- Decrypts an embedded data using a custom decryption based on FNV-1a
- Decompress the decrypted data by lznt1 to next step Payload
- Executes the Payload



# BloodAlchemy: Shellcode

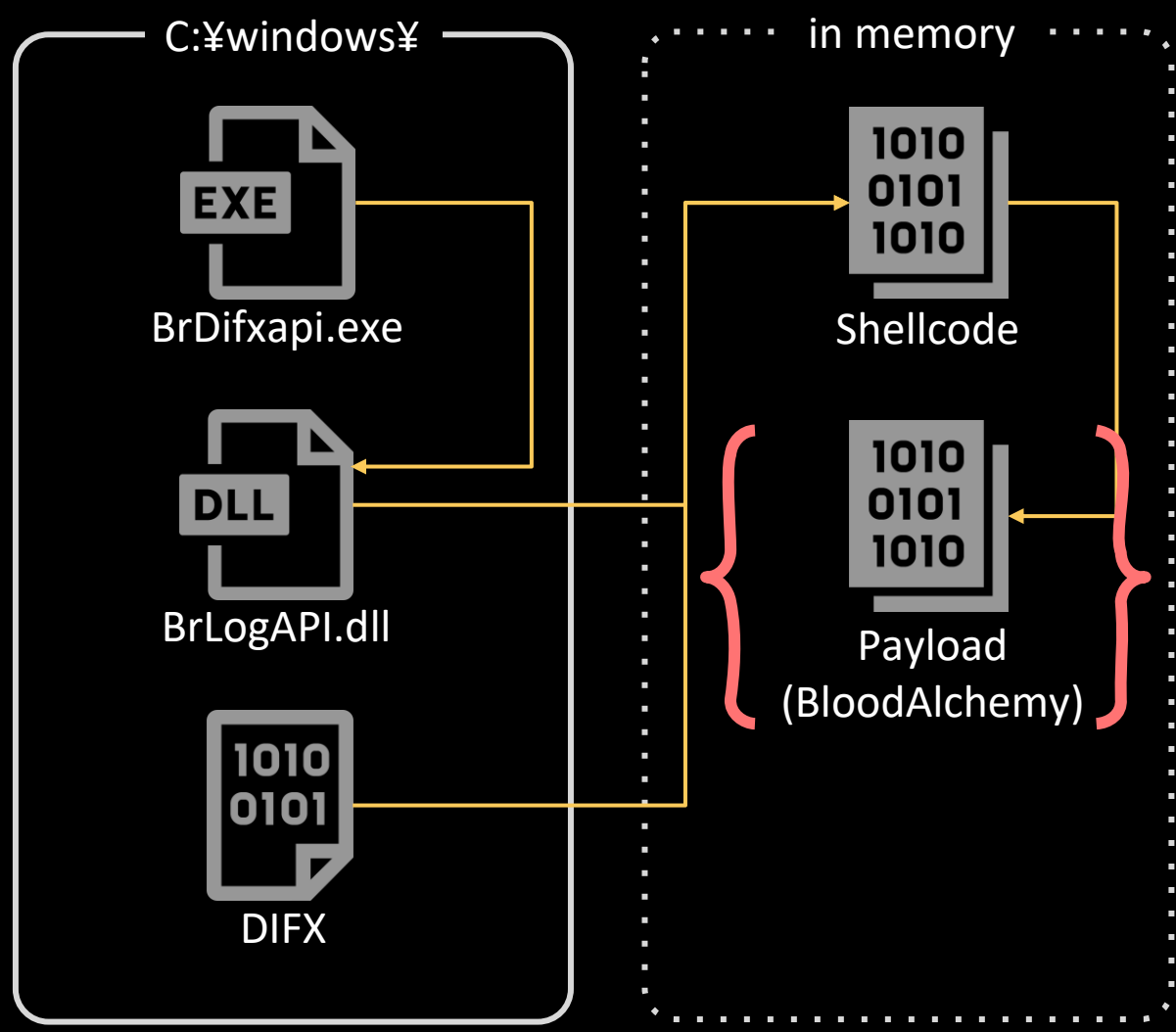
- The Payload has a unique data structure likely a custom PE header for loading the next payload in memory

```
call    [ebp+RtlDecompressBuffer] ; ret.
                        ; 02780000  45 AB 45 AB 10 00 00 00 8C 69 00 00 00 00 40 00
                        ; 02780010  00 00 00 00 00 70 01 00 AB 6F 01 00 BC 63 01 00
                        ; 02780020  00 10 00 00 00 00 00 00 50 00 00 00 AD 0F 01 00
                        ; 02780030  00 10 01 00 FD 0F 01 00 E0 39 00 00 00 50 01 00
push    4
push    edi             ; 0x3000
push    dword ptr [ebx+14h] ; 0x17000
push    0
call    _VirtualAlloc
mov     esi, [ebx+28h]  ; raw_address1 = 0x50
mov     edi, [ebx+24h]  ; virtual_address1 = 0
add     esi, ebx        ; src_imagebaase + raw_address1 = 0x02780050
mov     ecx, [ebx+2Ch]  ; virtual_size1 = 0x10FAD
add     edi, eax        ; dst_imagebase + virtual_address1 = 0x2640000
rep movsb
mov     esi, [ebx+34h]  ; raw_address2 = 0x10FFD
mov     edi, [ebx+30h]  ; virtual_address2 = 0x11000
add     esi, ebx        ; src_imagebaase + raw_address2 = 0x2790FFD
mov     ecx, [ebx+38h]  ; virtual_size2 = 0x39E0
add     edi, eax        ; dst_imagebase + virtual_address2 = 0x2651000
```

| offset | descriptions | data |
|---|---|---|
| 0x00 | magic number | 45 AB 45 AB |
| 0x04 | plugin id | 0x10 |
| 0x08 | entry point | 0x698c |
| 0x0c | original base | 0x400000 |
| 0x10 | absolute offset | 0 |
| 0x14 | size of virtualalloc | 0x17000 |
| 0x18 | size of raw data | 0x16fab |
| 0x1c | size of unknown | 0x163bc |
| 0x20 | base of code? | 0x1000 |
| 0x24 | section1: virtual addr | 0x0 |
| 0x28 | section1: raw data addr | 0x50 |
| 0x2c | section1: size of raw data | 0x10fa0 |
| 0x30 | section2: virtual addr | 0x11000 |
| 0x34 | etc.. | |

# Infection Flow

- BrDifxapi.exe loads BrLogAPI.dll using side-loading
- BrLogAPI.dll reads DFIX as a BLOB and decrypts Shellcode
- Shellcode decrypts Payload from an embedded data in itself
- Payload infects in memory




ITOCHU Cyber & Intelligence

# BloodAlchemy: Payload Run mode

- BloodAlchemy has a run mode which is hardcoded in the previous shellcode.
- The run mode:

0: call main function + process creation and injection + anti debug + anti sandbox + persistence + process injection
1: call main function
2: create thread for main function
3: call main function + anti debug + anti sandbox + persistence + process injection
4: process creation and injection
5: create named pipe
6: install malware

# BloodAlchemy: Payload Configuration

Medium confidence

- The configuration contains many information such as flags of each features, various values, encrypted data and offset of the data.
- The structure is very similar to Deed RAT and ShadowPad

```
02652068 offset_config   dd 534h                 ; DATA XREF: load_and_check_file+3
02652068                                         ; get_value_from_config_by_type+...
02652068                                         ; conf_size
0265206C unkown          dd 0BC67AD09h
02652070 unkown_0        dd 5B0369Ah
02652074 createmutex_flag dd 1                   ; 0: off
02652074                                         ; 1: on
02652078 mutex_value     dd 158h                 ; 0x26521c0 -> DFYNBEDKJHGAFSTIJECYUKFDEUJH
0265207C selefdelete_flag dd 0
02652080 antidbg_flag    dd 1
02652084 checksandbox_flag dd 0
02652088 install_reg     dd 176h                 ; 0x26521de -> SOFTWARE\Microsoft\Store
0265208C install_dir     dd 190h                 ; 0x26521f8 -> %ALLUSERSPROFILE%\Store
02652090 leg_exe         dd 1A9h                 ; 0x2652211 -> %AUTOPATH%\Test\test.exe
02652094 mal_dll         dd 1C3h                 ; 0x265222b -> BrLogAPI.dll
02652098 blob            dd 1D1h                 ; 0x2652239 -> DIFX
0265209C persistence_flag dd 0                   ; 0: off
0265209C                                         ; 1: service + startup + taskschd
0265209C                                         ; 2: service
0265209C                                         ; 3: startup
0265209C                                         ; 4: taskschd
```

```
02652198 c2              dd 50Eh                 ; 02652576 -> TCP://cdn1ac7bdd3.jptomorrow.com:443
026521A0 c2_1            dd 0
026521A4 c2_2            dd 0
026521A8 c2_3            dd 0
026521AC c2_4            dd 0
026521B0 c2_5            dd 0
```

```
02652576 e_c2_size       db 25h                  ; DATA XREF: seg000:c2↑?
02652577 e_c2_key        db 4Ah
02652578 e_c2_data       db 1Eh,9Dh,9,19h,'zH',9Dh,0A6h,'e',0BFh,0F8h,'>3',2,'V',0A5h,0DEh,1Dh
0265258B                 db 9Eh,0BFh,0Dh,86h,'%',9Ch,0B9h,0Dh,9Ch,'d',8Dh,0A4h,0Fh,0D1h,7Eh,0DA
```

```
026520DC p_processhollowing_1 dd 312h            ; 0x265237a -> %windir%\system32\taskhost.exe
026520E0 p_processhollowing_2 dd 332h            ; 0x265239a -> %windir%\system32\svchost.exe
```

# BloodAlchemy: Payload Encoded str/data

- Some configuration strings/data (not only config) were encrypted
- Following idapython can help your RE

| offset | descriptions | data |
|---|---|---|
| 0x00 | size of data | 0x25 |
| 0x01 | a byte key | 0x41 |
| 0x02 | encrypted data | 1E 9D 09 19 7A D0 9D 9D … |

```
1   def dec_obf(offset, s):
2       data = ida_bytes.get_bytes(offset, s+1)
3       iv = data[0]
4       enc = data[1:s+1]
5       dec = ""
6       for i in range(s):
7           dec += chr(iv ^ enc[i] & 0xFF)
8           ku0 = iv << (i % 5 +1) & 0xFF
9           ku1 = iv >> (7 - i % 5) & 0xFF
10          iv = (iv + (ku0 | ku1)) & 0xFF
11      idc.set_cmt(offset, dec[:-1], 1)
12      return dec[:-1]
```

```
026521C0 mutex          db 1Dh                  ; DATA XREF: seg000:mutex_value↑?
026521C0                                        ; DFYNBEDKJHGAFSTIJECYUKFDEUJH
026521C1                db 69h
026521C2                db 2Dh, 7Dh, 7Eh, 2Eh, 24h, 77h, 0D2h, 0BBh, 3Dh, 0A6h
026521CC                db 8Ch, 23h, 0ADh, 19h, 0BAh, 82h, 28h, 0AEh, 9, 0B7h
026521D6                db 9Eh, 29h, 0ADh, 0Eh, 0ABh, 9Eh, 28h, 0A3h
026521DE registry       db 19h                  ; DATA XREF: seg000:install_reg↑?
026521DE                                        ; SOFTWARE\Microsoft\Store
026521DF                db 13h
026521E0                db 40h, 76h, 5Bh, 51h, 2, 0BEh, 0ACh, 0BCh, 94h, 19h, 0B7h
026521EB                db 0F8h, 7Bh, 3Eh, 15h, 5Dh, 0F0h, 84h, 2Bh, 0BDh, 0BFh
026521F5                db 0Dh, 99h, 2Fh
026521F8 dir            db 18h                  ; DATA XREF: seg000:install_dir↑?
026521F8                                        ; %ALLUSERSPROFILE%\Store
026521F9                db 13h
026521FA                db 36h, 78h, 51h, 49h, 0, 0ACh, 0BBh, 0ABh, 9Bh, 4, 8Ch
02652205                db 0D4h, 4Fh, 18h, 2Ah, 77h, 0B3h, 0ACh, 24h, 9Ah, 0A4h
0265220F                db 10h, 8Eh
```

# BloodAlchemy: Payload Persistence

Medium confidence

- If run_mode = 0 or 3 and current_exe_path != persistence_dir¥test.exe and persistence_flag(config + 0x34) != 0, creates persistence depending on the flag 1 : 4.
  - 1: service + startup + taskschd(COM obj)
  - 2: service
  - 3: startup
  - 4: taskschd(COM obj)

Persistence_dir:

- %AUTOPATH%¥Test¥
- %LocalAppData%¥Programs¥Test¥
- %ProgramFiles%¥Test¥
- %ProgramFiles(x86)%¥Test¥

Deed RAT

Queries like these are a likely sign of Deed RAT infection.

Unlike the sample described above, the backdoor contains the environment pseudovariable %AUTOPATH%, u in the configuration field InstallationPath and, depending on backdoor permissions and system bitness, resolv as follows:

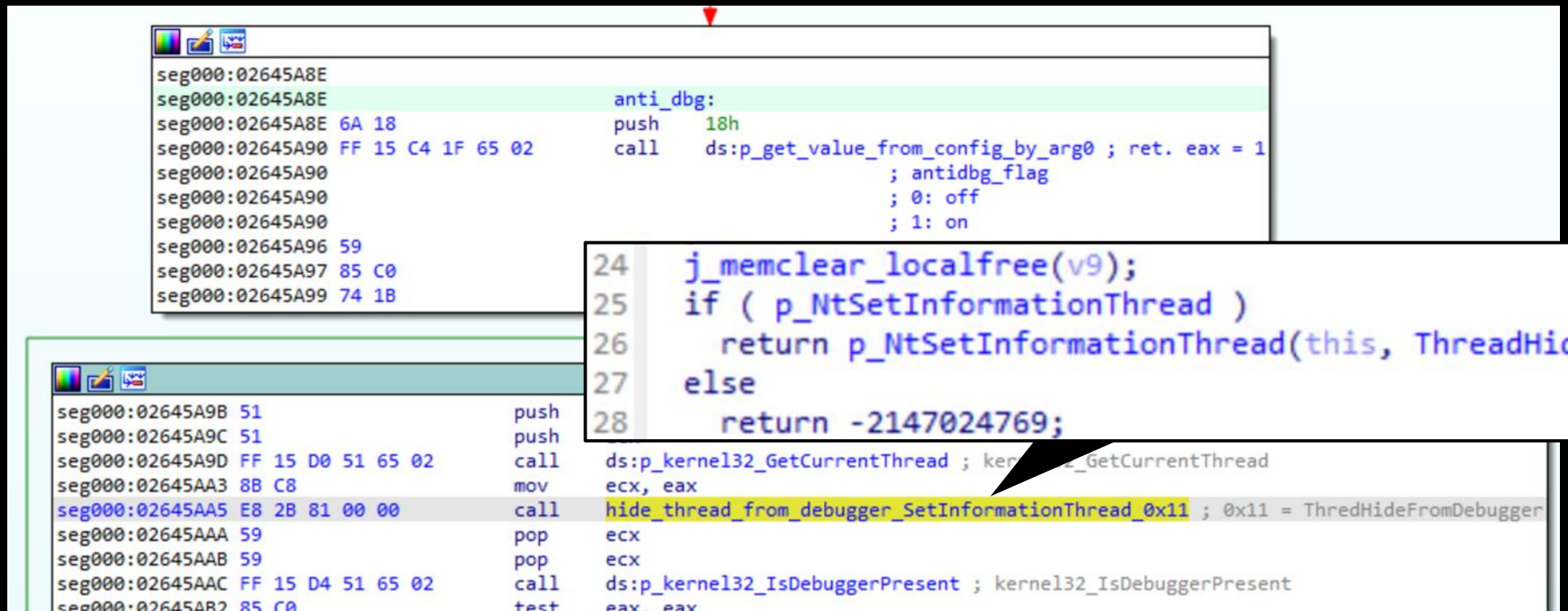
- %AppData% if the backdoor is missing administrator permissions
- %ProgramFiles(x86)% if the backdoor has administrator permissions and the system is 64-bit Windov
- %ProgramFiles% if the backdoor has administrator permissions and the system is 32-bit Windows

We have seen a similar implementation in PlugX, which used the variable %AUTO%.

https://www.ptsecurity.com/ww-en/analytics/pt-esc-threat-intelligence/space-pirates-a-look-into-the-group-s-unconventional-techniques-new-attack-vectors-and-tools/#id2-1

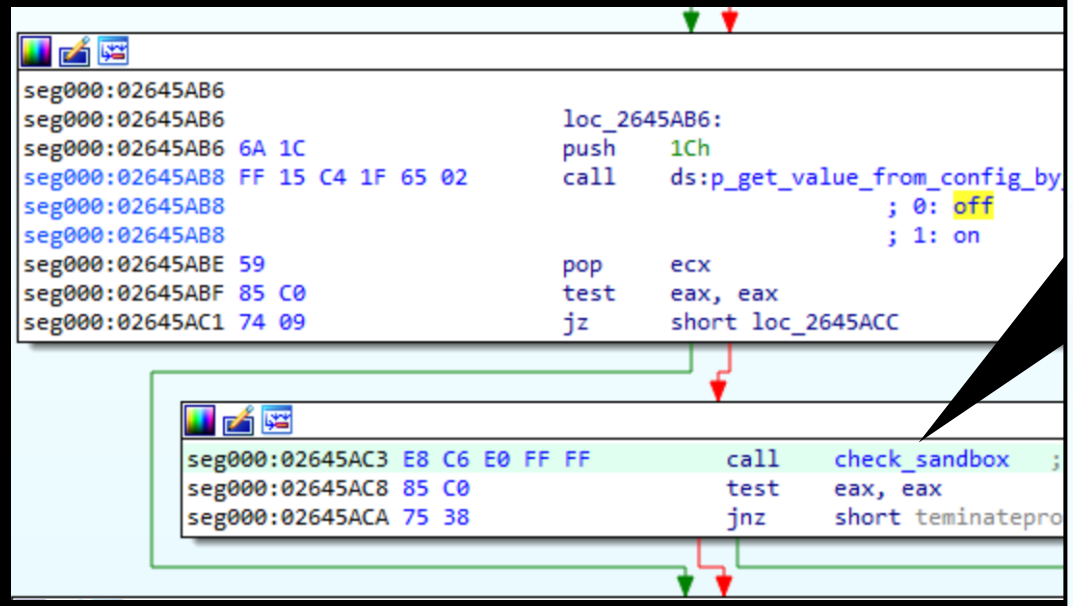
# BloodAlchemy: Payload
# Anti debug

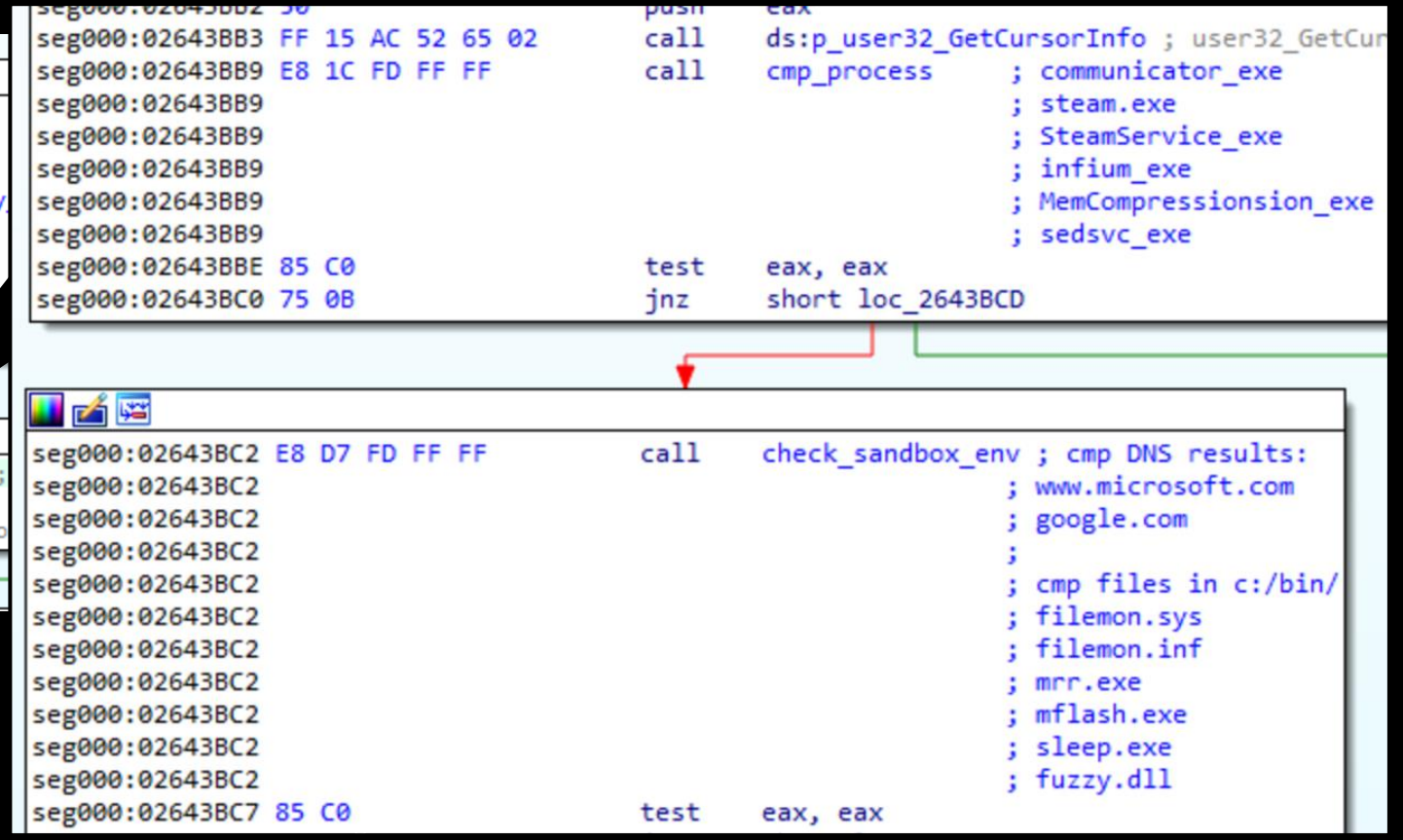
- If run_mode = 0 or 3 and current_exe_path != persistence_dir¥Test¥test.exe and anti_debug_flag(config + 0x18) = 1, calls NtSetInformationThread() with ThreadHideFromDebugger (0x11) to hide thread from debugger



# BloodAlchemy: Payload Anti sandbox

- If run_mode = 0 or 3 and current_exe_path != persistence_dir¥test.exe and anti_sandbox_flag(config + 0x1c) = 1, checks, process_name, files and results of DNS.




- Evades FireEye sandbox?

# BloodAlchemy: Payload Process injection

- If run_mode = 0 or 3 and processinjection_flag(config + 0x54) = 1, try to inject the previous shellcode into several hardcoded processes in config + 0x58 : 0x64:
  - %windir%¥system32¥SearchIndexer.exe
  - %windir%¥system32¥wininit.exe
  - %windir%¥system32¥taskhost.exe
  - %windir%¥system32¥svchost.exe

- Using Early Bard Injection: QueueUserAPC() for setting shellcode as a Queue of APC

```
 9  v8 = size_of_prv_shellcode + size_of_prv_shellcode
10  buf = p_kernel32_VirtualAllocEx(a1, 0, v8, 12288,
11  if ( !buf )
12    return p_kernel32_GetLastError();
13  if ( !p_kernel32_WriteProcessMemory(a1, buf, 39976
14    goto LABEL_9;
15  v5 = size_of_prv_shellcode;
16  if ( size_of_prv_shellcode <= 0x1000 )
17    v5 = 4096;
18  if ( !p_kernel32_VirtualProtectEx(a1, buf, v5, 32,
19    || !p_kernel32_QueueUserAPC(buf, a2, a3)
20    || p_kernel32_ResumeThread(a2) == -1 )
21  {
```

# BloodAlchemy: Payload Process creation + injection

- If run_mode = 0 or 4 and processcreationinjection_flag(config + 0x68) = 1, try to create following processes from config + 0x6c : 0x74, and injects the previous shellcode into these process:
  - %windir%¥system32¥wininit.exe
  - %windir%¥system32¥taskeng.exe
  - %windir%¥system32¥taskhost.exe
  - %windir%¥system32¥svchost.exe

- Creates targeted process and calls the function of Early Bard Injection

```
index = 0x6C;
while ( 1 )
{
  if ( !p_get_value_from_config_by_arg0(index) )
    goto LABEL_8;
  p_dec_procname_from_conf_0x24_0x58_0x6C(index, target_proc);// target process:
                                                 // %windir%\system32\wininit.exe
                                                 // %windir%\system32\taskeng.exe
                                                 // %windir%\system32\taskhost.exe
                                                 // %windir%\system32\svchost.exe
  j_memset(v3, 0, 68);
  v3[0] = 68;
  v4 = 0;
  v5 = 0;
  v6 = 0;
  v7 = 0;
  p_kernel32_GetStartupInfoW(v3);
  v3[11] = 1;
  LOWORD(v3[12]) = 0;
  if ( p_kernel32_CreateProcessW(0, target_proc[0], 0, 0, 0, 532, 0, 0, v3, &v4)
  {
    v0 = early_bard_injection_loader_shellcode(v4, v5, 6);
    v9 = v0;
```

# BloodAlchemy: Payload Import corresponding func

Low confidence

- Set protocol id from the C2 destination:
  - TCP = 1
  - HTTP = 2
  - HTTPS = 3
  - UDP = 4
  - DNS = 5
  - PIPE = 6
  - SMB = 7
  - MUX = 8

- Import corresponding function of?

  protocol id like ShadowPad's

  protocol plugins

```
17  switch ( protocol_id )
18  {
19    case 1:
20      if ( p_get_value_from_config_by_arg0(0x98) )// 1
21      {
22        v4 = v10;
23        return TCP_MUX_protocol(v4);
24      }
25  LABEL_7:
26      v4 = 0;
27      return TCP_MUX_protoco  v4);
```

```
41  inited->field_0 = 1;
42  result = inited;
43  inited->create_socket_connect = tcp_create_socket_connect;
44  inited->connect_send_recv = tcp_connect_send_recv;
45  inited->recv_data = tcp_recv_data;
46  inited->shutdown = tcp_shutdown;
47  inited->shutdonwn_closesocket = tcp_shutdonwn_closesocket;
48  inited->wsarecv = tcp_wsarecv;
49  inited->wsagetoverlappedresult = tcp_wsagetoverlappedresult;
50  inited->wsasend = tcp_wsasend;
51  inited->wsagetoverlappedresult_ = tcp_wsagetoverlappedresult;
52  inited->getsockname = tcp_getsockname;
53  inited->getsockname_ = tcp_getsockname_;
54  return result;
```

```
47      p_get_value_from_config_by_arg0(0xAC);    // ONLOGON
48      return DNS_protocol(v9);
```

# BloodAlchemy: Payload Backdoor commands

- 15 backdoor commands were implemented

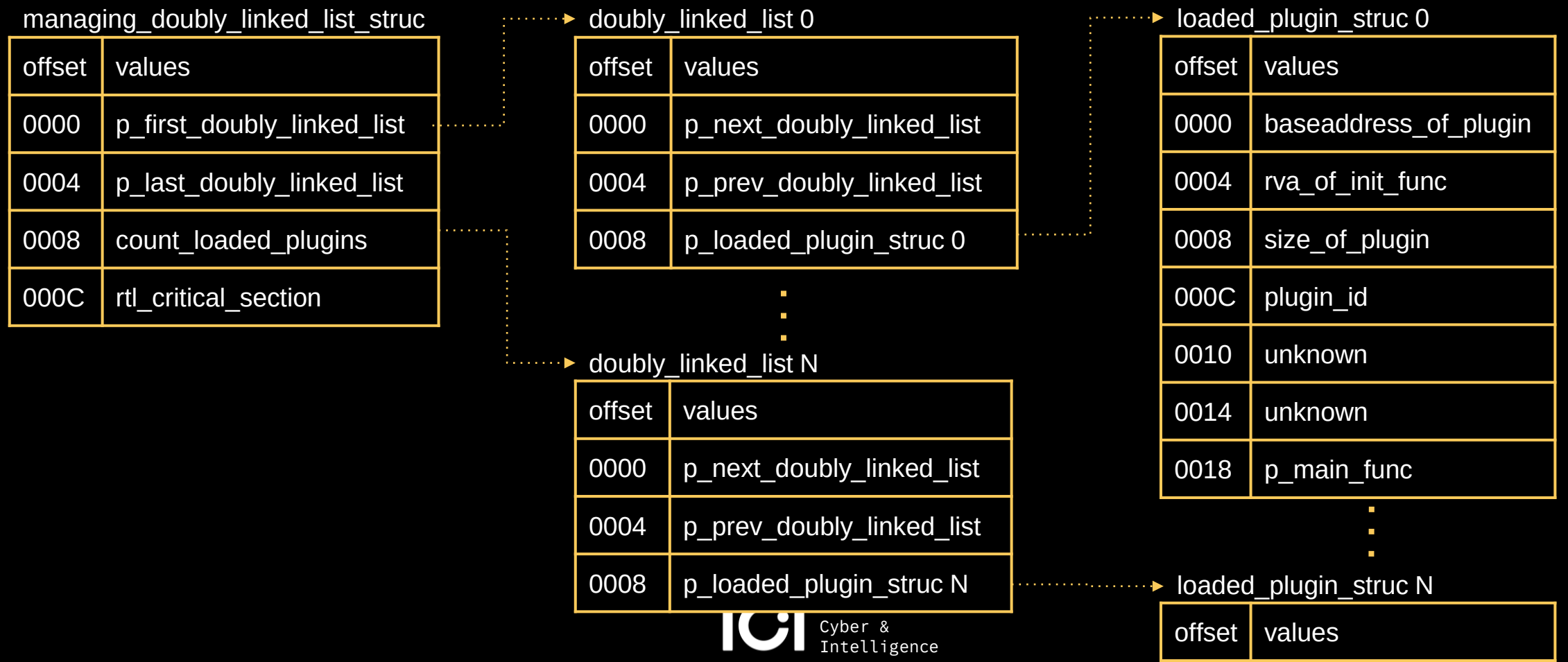
```
21  command_id = *(a2 + 12);
22  if ( command_id <= 0x1301 )
23  {
24    if ( command_id == 0x1301 )
25      return bc_1301(buf, a2);
26    command_1101_1 = command_id - 0x1101;
27    if ( !command_1101_1 )
28      return bc_1101_update_config_(buf, a2);
29    command_1102_1 = command_1101_1 - 1;
30    if ( !command_1102_1 )
31    {
32      p_ntdll_RtlEnterCriticalSection(&dword_2652644);
33      v10 = bc_1102get_current_config_(buf, 4354, &off
34      p_ntdll_RtlLeaveCriticalSection(&dword_265
35      return v10;
40    v6 = v5 - 1;
41    if ( !v6 )
42      return bc_1202_update_BrLogAPI_dll(buf);
43    v7 = v6 - 1;
44    if ( !v7 )
45      return bc_1203_update_DIFX(buf);
```

Forces on loading next plugins?

| command | descriptions |
|---|---|
| 0x1101 | update config |
| 0x1102 | get current config |
| 0x1201 | update test.exe |
| 0x1202 | update BrLogAPI.dll |
| 0x1203 | update DIFX |
| 0x1204 | uninstall and terminated |
| 0x1205 | lunch persistence_dir¥test.exe |
| 0x1301 | unknown |
| 0x1302 | load received payload, stores into registry value |
| 0x1303 | delete registry value |
| 0x1304 | unknown |
| 0x1401 | get proxy info |
| 0x1402 | update proxy info |
| 0x1501 | gather victim info |
| 0x1502 | echo 0x1502 |

# BloodAlchemy: Payload linked list structure

- Malware creates linked list structure to manage loaded plugins.




Cyber & Intelligence

# BloodAlchemy: Payload linked list structure

- Malware creates linked list structure to manage loaded plugins.

```
mov     ecx, offset init_plugin ; flag(arg0):
                                ; 1: p_commands_1101_13?? <- command
                                ; 2: arg_8 = 0x10
                                ; 3: arg_8 = e_baseapi
                                ; 4: arg_8 = e_config__install__plugin__proxy__network
                                ; 5: arg_8 = command function
                                ; 6: 0
sub     ecx, esi                ; rva of init_func of plugin -> 0x1e1d
mov     [eax+loaded_plugin_struc.size_of_plugin], edi ; 0xf874
pop     edi                     ; previos_shellcode = 0x1200000
mov     [eax+loaded_plugin_struc.baseaddress_of_plugin], esi ; 01230000
mov     [eax+loaded_plugin_struc.rva_of_init_func], ecx ; 0x1e1d
mov     ecx, eax                ; loaded_plugin_struc = 00DD9928
mov     [eax+loaded_plugin_struc.v_unknown_], 1 ; 1
pop     esi                     ; mode_flag
jmp     mod_doubly_linked_list ; 0
```

# Code similarities

# Similarities: Deed RAT cmp DLL file

Low confidence

- DLL side-loading, Simple loader func, API order is similar
- Filename, decode, decryption, junk code are different

Deed RAT

```
139     {
140       filename[v
141       break;
142     }
143   }
144   debug_log[0] = 0xD8EBE8E9;
145   debug_log[1] = 0xD6E197DE;
146   v27 = 17886;
147   dec_string(debug_log, 10);
148   (junk)();
149   (lstrcatA)(filename, debug_log);
150   file = (CreateFileA_)(filename, 0x80000000, 1
151   hFile = file;
152   if ( file == -1 )
153     return 0;
154   v17 = (GetFileSize_)(file, 0);
155   dec_shellcode = (VirtualAlloc_)(0, v17, 4096,
156   if ( !dec_shellcode )
157   {
158     (CloseHandle_)(hFile);
159     return 0;
160   }
161   if ( !(ReadFile_)(hFile, dec_shellcode, v17,
162   {
163     (CloseHandle_)(hFile);
164 LABEL_22:
165     (v43)(dec_shellcode, 0, 0x8000);
166     return 0;
167   }
168   (junk)();
169   (CloseHandle_)(hFile);
170   v19 = v48;
171   v20 = 0;
172   for ( i = 29514; v20 < v19; ++v20 )
173   {
174     *(dec_shellcode + v20) = (i ^ *(dec_shellco
175     i = (11700 * i + 371768) >> 3;
176   }
177   junk(i);
178   if ( !(VirtualProtect_)(dec_shellcode, 1536,
179     goto LABEL_22;
180   dword_74373000 = dec_shellcode;
181   dec_shellcode();
```

Blood Alchemy

```
36      }
37      v6 = v4 + 1;
38      if ( v6 >= 0x104 )
39        __report_rangecheckfailure();
40      Filename[v6] = 0;
41    }
42 LABEL_6:
43    strcpy(blob_file, "DIFX");
44    lstrcatA(Filename, blob_file);
45    FileA = CreateFileA(Filename, 0x80000000, 1u, 0, 3u, 0, 0);
46    hFile = FileA;
47    v15 = FileA;
48    if ( FileA == -1
49      || (FileSize = GetFileSize(FileA, 0), nNumberOfBytesToRead = FileSiz
50      || (dec_shellcode = VirtualAlloc(0, FileSize, 0x3000u, 4u)) == 0 )
51    {
52      LastError = GetLastError();
53    }
54    else
55    {
56      NumberOfBytesRead = 0;
57      if ( ReadFile(hFile, dec_shellcode, nNumberOfBytesToRead, &NumberOfB
58      {
59        enc_data = NumberOfBytesRead;
60        v17[72] = 0;
61        v19 = 0i64;
62        aes_init(v8, &v19);
63        enc_data -= 16;
64        aes_dec(dec_shellcode + 16, enc_data);
65        v10 = enc_data - dec_shellcode[enc_data - 1];
66        v17[0] = 0;
67        NumberOfBytesRead = v10;
68        ModuleHandleA = GetModuleHandleA("kernel32.dll");
69        VirtualProtect = GetProcAddress(ModuleHandleA, "VirtualProtect");
70        if ( VirtualProtect(dec_shellcode, 4096, 32, v17) )
71          LastError = (dec_shellcode)(0);
72        else
73          LastError = GetLastError();
74        hFile = v15;
75      }
76      else
77      {
78        LastError = GetLastError();
```

# Similarities: Deed RAT cmp Shellcode

Low confidence

- Shellcode loading process is also similar to Deed RAT
  - getapibyhash > decryption > decompress > extract data from structure

Deed RAT

```
qmemcpy(v11, (shellbase_offset17 + 3
v12 = 0;
v13 = v34;
v14 = 87;
while ( !CryptAcquireContextA(&v51, 0, *(v13 - 1), *v13, v13[1]) )
{
  ++v12;
  v13 += 3;
  if ( v12 >= 4 )
  {
    v10 = v53;
    goto LABEL_33;
  }
}
v28 = 520;
v29 = 26126;
v30 = 16;
qmemcpy(v31, (shellbase_offset17 + 1792), sizeof(v31));
v10 = 0;
if ( !CryptImportKey(v51, &v28, 28, 0, 0, &v52) )
  goto LABEL_33;
v40 = 2;
if ( !CryptSetKeyParam(v52, 4, &v40, 0) )
  goto LABEL_33;
buf = blob;
if ( !CryptDecrypt(v52, 0, 1, 0, blob, &v49) )
  goto LABEL_33;
VirtualAlloc_1 = VirtualAlloc;
size_ = 2 * v49;
v17 = VirtualAlloc(0, 2 * v49, 12288, 4);
v53 = v17;
if ( !v17 )
  goto LABEL_25;
if ( RtlDecompressBuffer(2, v17, size_, buf, v49, &size_) >= 0 && *v17 == -554875564 )// ret. dec =
                                        // 00CC0000  54 45 ED DE 20 00 00 00 40 17 00 00 00 00 40 00
                                        // 00CC0010  00 10 00 00 00 50 00 00 00 50 00 00 00 10 00 00
                                        // 00CC0020  00 0A 00 00 00 20 01 00 8D 15 01 00 00 06 00 00
```

Crypto algorithm was changed

Blood Alchemy

```
buf_1 = VirtualAlloc_1(0, a4, 12288, 4)
buf = buf_1;
key_update = *(a2 + a3);
v31 = a4 - 2;
key_ = key_update;
dst_imagebase = a4 - 2;
if ( a4 != 2 )

  InstructionCachea = a4 - 2;
  uf_1;
  v63 = a2 + a3 + 2 - buf_1;
  do
  {
    v_FMV1a = 2166136261;
    for ( j = 0; j < 2; ++j )
      v_FMV1a = 16777619 * (v_FMV1a ^ *(&key_ + j));
    key_update ^= 0x21
                * ((9 * ((0x2001 * v_FMV1a) ^ ((0x2001 * v_FMV1a) >> 7))) ^ ((9
                                                                             * ((0x2001 * v_FMV1a) ^ ((0x200
    v35 = key_update ^ v32[v63];
    key_ = key_update;
    *v32++ = v35;
    --NtFlushInstructionCachea;
  }
  while ( NtFlushInstructionCachea );
  v31 = dst_imagebase;
}
VirtualAlloc_ = VirtualAlloc;
EP_payload = v31;
size_1 = (5 * v31 + 4096);
v37 = VirtualAlloc(0, size_1, 12288, 4);
key_ = v37;
RtlDecompressBuffer(2, v37, size_1, buf, EP_payload, &size_1);// ret.
                                        // 02780000  45 AB 45 AB 10 00 00 00 8C 69 00 00 00 00 40 00
                                        // 02780010  00 00 00 00 00 70 01 00 AB 6F 01 00 BC 63 01 00
```

# Similarities: Deed RAT cmp Shellcode

High confidence

- Decrypted data structure(like PE?) of next payload is also similar
- Magic number has been modified.

Deed RAT

```
if ( !v17 )
  goto LABEL_25;
if ( RtlDecompressBuffer(2, v17, size_, buf, v49, &size_) >= 0 && *v17 == -554875564 )// ret. dec =
                                                // 00CC0000  54 45 ED DE 20 00 00 00 40 17 00 00 00 00 40 00
                                                // 00CC0010  00 10 00 00 00 50 00 00 00 50 00 00 00 10 00 00
                                                // 00CC0020  00 0A 00 00 00 20 01 00 8D 1 00 00 06 00 00
{
```

| offset | descriptions | data |
|---|---|---|
| 0x00 | magic number | 54 45 ED DE |
| 0x04 | plugin id | 0x20 |
| 0x08 | entry point | 0x1740 |
| 0x0c | original base | 0x400000 |
| 0x10 | absolute offset | 0x1000 |
| 0x14 | size of virtualalloc | 0x5000 |
| 0x18 | size of raw data | 0x5000 |
| 0x1c | ..etc | |

Blood Alchemy

```
key_ = v37;
RtlDecompressBuffer(2, v37, size_1, buf, EP_payload, &size_1);// ret.
                                                // 02780000  45 AB 45 AB 10 00 00 00 8C 69 00 00 00 00 40 00
                                                // 02780010  00 00 00 00 00 70 01 00 AB 6F 01 00 BC 63 01 00
                                                // 02780020  00 10 00 00 00 00 00 0 00 00 00 AD 0F 01 00
v38 = VirtualAlloc_(0, *(v37 + 5), 12288, 4);
```

| offset | descriptions | data |
|---|---|---|
| 0x00 | magic number | 45 AB 45 AB |
| 0x04 | plugin id? | 0x10 |
| 0x08 | entry point | 0x698c |
| 0x0c | original base? | 0x400000 |
| 0x10 | absolute offset? | 0 |
| 0x14 | size of virtualalloc | 0x17000 |
| 0x18 | size of raw data | 0x16fab |
| 0x1c | ..etc | |

# Similarities: Deed RAT cmp Payload

High confidence

- After both EntryPoint is almost same for setting an exception to write %ALLUSERPROFILE%error.log with the same format

Deed RAT

```
47    v5 = v3;
48    v6 = v4;
49    shellbase_offset17 = v3;
50    if ( getprocaddrbyhash(&setunhandledexceptionfilter, &dword_CF51A4) )
51      setunhandledexceptionfilter(c_write_error_log);
52    v7 = 0;
```

```
// %ALLUSERSPROFILE%\error.log
// fmt:
// %4.4d-%2.2d-%2.2d %2.2d:%2.2d:%2.2d Exception Address: 0x%p, Code: 0x%8.8x
int __stdcall c_write_error_log(_DWORD **a1)
{
  void (*v1)(void); // eax
  char v3[1024]; // [esp+0h] [ebp-418h] BYREF
  __int16 v4[8]; // [esp+400h] [ebp-18h] BYREF
  int v5[2]; // [esp+410h] [ebp-8h] BYREF

  v5[0] = 0;
  v5[1] = 0;
  if ( getprocaddrbyhash(&p_GetLocalTime, &e_GetLocalTime) )
    p_GetLocalTime(v4);
  dec(&fomat__4_4d__2_2d___);
  v1 = p_user32_wsprintfA(v3, v5[0], v4[0], v4[1], v4[3], v4[4], v4[5], v4[6], (
  v1();
  if ( getprocaddrbyhash(&kernel32_OutputDebugStringA, &e_outputdebugstringA) )
    kernel32_OutputDebugStringA(v3);
  write_error_log(v3);                          // %ALLUSERSPROFILE%\error.log
  c_c_localfree(v5);
  return 0;
```

Blood Alchemy

```
21    p_kernel32_SetU(c_write_error_log);
22    c_AddVectoredContinueHandler();
23    v15 = 0;
24    v16 = 0;
25    v17 = 0;
26    dec_edx(15);
```

```
// %ALLUSERSPROFILE%\error.log
// fmt:
// %4.4d-%2.2d-%2.2d %2.2d:%2.2d:%2.2d Exception Address: 0x%p, Code: 0x%8.8x
int __stdcall c_write_error_log(int **arg0)
{
  int ModuleHandleA; // eax
  int v3; // [esp-4h] [ebp-424h]
  char v4[1024]; // [esp+4h] [ebp-41Ch] BYREF
  int a1[3]; // [esp+404h] [ebp-1Ch] BYREF
  __int16 v6[8]; // [esp+410h] [ebp-10h] BYREF

  memset(a1, 0, sizeof(a1));
  dec__0(a1, &fomat__4_4d__2_2d___, 0x62u);
  p_kernel32_GetLocalTime(v6);
  j_memset(v4, 0, sizeof(v4));
  v3 = **arg0;
  ModuleHandleA = kernel32_GetModuleHandleA(0, (*arg0)[3]);
  p_user32_wsprintfA(v4, a1[0], v6[0], v6[1], v6[3], v6[4], v6[5], v6[6], Modul
  kernel32_OutputDebugStringA(v4, v3);
  write_error_log(v4);                          // %ALLUSERSPROFILE%\error.log
  j_memclear_localfree(a1);
```

ITOCHU Cyber & Intelligence

# Similarities: ShadowPad cmp Payload

Medium confidence

- Plugin names are similar to ShadowPad
- Plugin loading scheme is also similar

ShadowPad

| Module ID | Module name | Module timestamp |
|---|---|---|
| 58338 | DNS | Fri 15 Mar 2019 05:22:19 PM UTC |
| 3331 | TCP | Fri 15 Mar 2019 05:21:40 PM UTC |
| 22707 | UDP | Fri 15 Mar 2019 05:22:13 PM UTC |
| 48503 | HTTP | Fri 15 Mar 2019 05:21:48 PM UTC |
| 33173 | HTTPS | Fri 15 Mar 2019 05:21:56 PM UTC |
| 4626 | Root | Fri 15 Mar 2019 05:23:15 PM UTC |
| 12996 | Config | Fri 15 Mar 2019 05:23:26 PM UTC |
| 61013 | Plugins | Fri 15 Mar 2019 05:23:21 PM UTC |
| 5176 | Online | Fri 15 Mar 2019 05:23:49 PM UTC |
| 35573 | Install | Fri 15 Mar 2019 05:23:43 PM UTC |

https://web-assets.esetstatic.com/wls/2019/10/ESET_Winnti.pdf

Blood Alchemy

```
switch ( flag )
{
  case 1:
    p_commands_1101_13__ = backdoor_command;
    return 0;
  case 2:
    *a3 = 0x10;                                 // plugin id?
    return 0;
  case 3:
    enc_index = 8;                              // baseapi
    goto LABEL_10;
  case 4:
    enc_index = 0x28;                           // config, install, plugin, proxy, network
LABEL_10:
    memset(v5, 0, sizeof(v5));
    dec_edx(enc_index);
    p_kernel32_lstrcpyW(a3, 0);
    j_memset_localfree_0(v5);
    return 0;
  case 5:
    *a3 = &p_commands_1101_13__;
```

ITOCHU Cyber & Intelligence

# Similarities: Deed RAT

Summary of code Similarities:

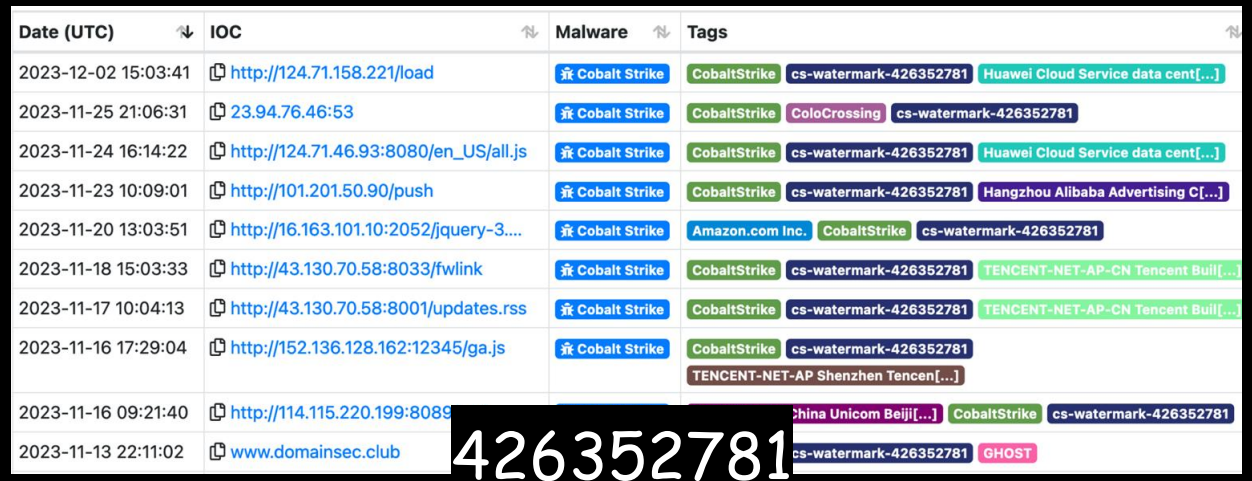
- Payload header structure
- Plugins loading scheme
- Plugin names
- Linked list structure for plugins
- Config structure
- Persistence dir and conditions
- Error msg fmt and file of exception
- DLL loader and loading Shellcode

BloodAlchemy is a new variant of Deed RAT and the origin was Shadowpad

# Infrastructure

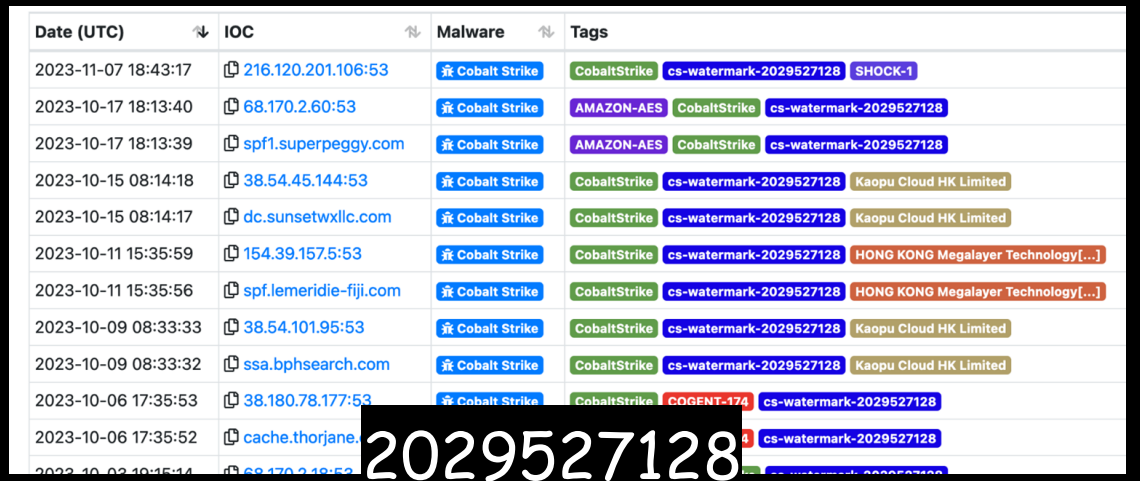
# Extracted C2s

- Infrastructures of Cobalt Strike Beacon and BloodAlchemy
- Cobalt Strike Beacon I: 121.41.35[.]65 + watermark 426352781
- Cobalt Strike Beacon II: cdn39a700bb.jptomorrow[.]com + watermark 2029527128
- BloodAlchemy I: TCP://cdn1ac7bdd3.jptomorrow[.]com:443
- BloodAlchemy II: HTTPS://cdn1ac7bdd3.jptomorrow[.]com:443
- BloodAlchemy (Elastic): HTTPS://cdn-hk-6dc8.bogotatrade[.]co:443



| Date (UTC) | IOC | Malware | Tags |
|---|---|---|---|
| 2023-12-02 15:03:41 | http://124.71.158.221/load | Cobalt Strike | CobaltStrike, cs-watermark-426352781, Huawei Cloud Service data cent[...] |
| 2023-11-25 21:06:31 | 23.94.76.46:53 | Cobalt Strike | CobaltStrike, ColoCrossing, cs-watermark-426352781 |
| 2023-11-24 16:14:22 | http://124.71.46.93:8080/en_US/all.js | Cobalt Strike | CobaltStrike, cs-watermark-426352781, Huawei Cloud Service data cent[...] |
| 2023-11-23 10:09:01 | http://101.201.50.90/push | Cobalt Strike | CobaltStrike, cs-watermark-426352781, Hangzhou Alibaba Advertising C[...] |
| 2023-11-20 13:03:51 | http://16.163.101.10:2052/jquery-3.... | Cobalt Strike | Amazon.com Inc., CobaltStrike, cs-watermark-426352781 |
| 2023-11-18 15:03:33 | http://43.130.70.58:8033/fwlink | Cobalt Strike | CobaltStrike, cs-watermark-426352781, TENCENT-NET-AP-CN Tencent Buil[...] |
| 2023-11-17 10:04:13 | http://43.130.70.58:8001/updates.rss | Cobalt Strike | CobaltStrike, cs-watermark-426352781, TENCENT-NET-AP-CN Tencent Buil[...] |
| 2023-11-16 17:29:04 | http://152.136.128.162:12345/ga.js | Cobalt Strike | CobaltStrike, cs-watermark-426352781, TENCENT-NET-AP Shenzhen Tencen[...] |
| 2023-11-16 09:21:40 | http://114.115.220.199:8089 | | hina Unicom Beiji[...], CobaltStrike, cs-watermark-426352781 |
| 2023-11-13 22:11:02 | www.domainsec.club | | cs-watermark-426352781, GHOST |

426352781



| Date (UTC) | IOC | Malware | Tags |
|---|---|---|---|
| 2023-11-07 18:43:17 | 216.120.201.106:53 | Cobalt Strike | CobaltStrike, cs-watermark-2029527128, SHOCK-1 |
| 2023-10-17 18:13:40 | 68.170.2.60:53 | Cobalt Strike | AMAZON-AES, CobaltStrike, cs-watermark-2029527128 |
| 2023-10-17 18:13:39 | spf1.superpeggy.com | Cobalt Strike | AMAZON-AES, CobaltStrike, cs-watermark-2029527128 |
| 2023-10-15 08:14:18 | 38.54.45.144:53 | Cobalt Strike | CobaltStrike, cs-watermark-2029527128, Kaopu Cloud HK Limited |
| 2023-10-15 08:14:17 | dc.sunsetwxllc.com | Cobalt Strike | CobaltStrike, cs-watermark-2029527128, Kaopu Cloud HK Limited |
| 2023-10-11 15:35:59 | 154.39.157.5:53 | Cobalt Strike | CobaltStrike, cs-watermark-2029527128, HONG KONG Megalayer Technology[...] |
| 2023-10-11 15:35:56 | spf.lemeridie-fiji.com | Cobalt Strike | CobaltStrike, cs-watermark-2029527128, HONG KONG Megalayer Technology[...] |
| 2023-10-09 08:33:33 | 38.54.101.95:53 | Cobalt Strike | CobaltStrike, cs-watermark-2029527128, Kaopu Cloud HK Limited |
| 2023-10-09 08:33:32 | ssa.bphsearch.com | Cobalt Strike | CobaltStrike, cs-watermark-2029527128, Kaopu Cloud HK Limited |
| 2023-10-06 17:35:53 | 38.180.78.177:53 | Cobalt Strike | CobaltStrike, COGENT-174, cs-watermark-2029527128 |
| 2023-10-06 17:35:52 | cache.thorjane. | | cs-watermark-2029527128 |

2029527128

ITOCHU
Cyber &
Intelligence

# jptomorrow[.]com - whois

```
Registrar:
  Handle: 2774153340_DOMAIN_COM-VRSN
  LDH Name: jptomorrow.com
  Nameserver:
    LDH Name: control.dnspod.net
      Status: Active
  Nameserver:
    LDH Name: scheat.dnspod.net
      Status: Active
  SecureDNS:
    Delegation Signed: false
  Event:
    Action: registration
    Date: 2023-04-19T07:01:25Z
  Event:
    Action: expiration
    Date: 2024-04-19T07:01:26Z
  Event:
    Action: last changed
    Date: 2023-04-19T07:01:26.752Z
  Event:
    Action: last update of RDAP database
    Date: 2023-04-20T12:12:59.348Z
    Status: client transfer prohibited
```

```
Entity:
  Role:
    REGISTRANT
    ADMIN
    TECHNICAL
  VCard:
    Kind: individual
    Formatted Name: Huang Shan
    Address:
      Street:


        LianJinCun17
        Locale: Gao An
        Region: JIANGXI
        Postal Code: 336200
        Country: CN
    Email: 3280132818@qq.com
        TYPE: voice
    Status: Active
```

- Whois data

ITOCHU Cyber & Intelligence

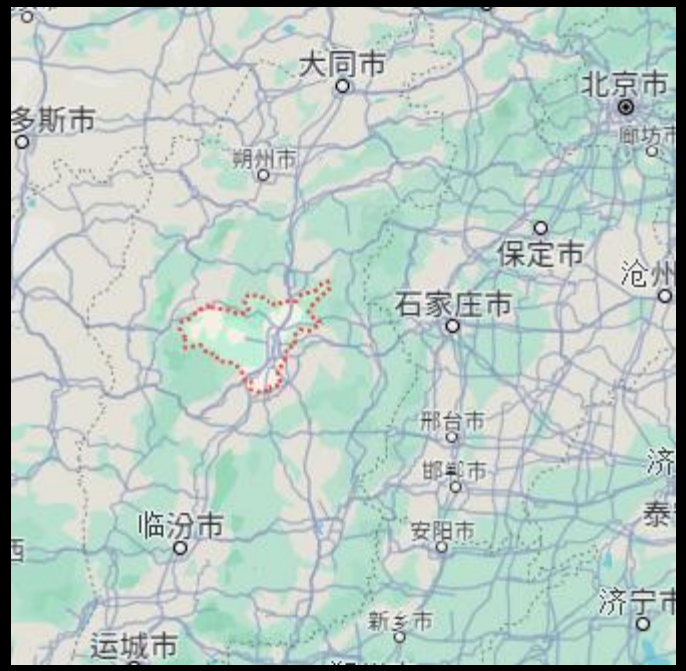
# 3280132818 @ qq[.]com

| Domain | registration | Registrar | Email | Name | Organization | Street | City | State | Postal Code | Country | Phone |
|---|---|---|---|---|---|---|---|---|---|---|---|
| plam2023[.]com | 2023-07-17T15:52:44Z | PDR Ltd. d/b/a PublicDomainRegistry[.]com | 3280132818@qq[.]com | Huang Shan | | | Gao An | JIANGXI | 336200 | CN | +86.62406075 |
| mncdntech[.]com | 2023-07-04T09:13:21Z | PDR Ltd. d/b/a PublicDomainRegistry[.]com | 3280132818@qq[.]com | Huang Shan | | | Gao An | JIANGXI | 336200 | CN | +86.62406075 |
| vultr-dns[.]com | 2023-06-10T15:15:13Z | PDR Ltd. d/b/a PublicDomainRegistry[.]com | 3280132818@qq[.]com | Huang Shan | | | Gao An | JIANGXI | 336200 | CN | +86.62406075 |
| jptomorrow[.]com | 2023-04-19T07:01:25Z | PDR Ltd. d/b/a PublicDomainRegistry[.]com | 3280132818@qq[.]com | Huang Shan | | | Gao An | JIANGXI | 336200 | CN | +86.62406075 |
| jttoday[.]net | 2023-03-20T06:31:21Z | PDR Ltd. d/b/a PublicDomainRegistry[.]com | 3280132818@qq[.]com | Huang Shan | | | Gao An | JIANGXI | 336200 | CN | +86.62406075 |
| nttbusinessdaily[.]com | 2021-03-02T03:56:59Z | PDR Ltd. d/b/a PublicDomainRegistry[.]com | 3280132818@qq[.]com | li bin | | | tai yuan | SHANXI | 30000 | CN | +86.2406075 |
| substantialeconomy[.]com | 2020-05-25T01:15:42Z | PDR Ltd. d/b/a PublicDomainRegistry[.]com | 3280132818@qq[.]com | li bin | | hepingbeilu38hao | tai yuan | SHANXI | 30000 | CN | +86.2406075 |
| sweetdatepalm[.]com | 2019-08-16T05:45:25Z | PDR Ltd. d/b/a PublicDomainRegistry[.]com | 3280132818@qq[.]com | li bin | li bin | hepingbeilu38hao | tai yuan | SHANXI | 030000 | CHINA | 862406075 |

# Address Information


Gao An　JIANGXI

高安市　江西省




tai yuan　SHANXI

太原市　山西省


hepingbeilu　和平北路

ITOCHU
Cyber &
Intelligence

# Related Domain

- 1 more domain is also hit. (no e-mail address but other information is similar)

| | | | | | | | | | | | |
|---|---|---|---|---|---|---|---|---|---|---|---|
| chickenchickengo[.]com | 2023-06-17T06:32:00Z | GMO Internet Group, Inc. d/b/a Onamae[.]com | | Huang Shan | Huang Shan | LianJinCun 17 | Gao An | JIANGXI | 336200 | CN | +86.62406075 |


LianJinCun　Gao An　JIANGXI

连锦村　**高安市**　**江西省**

ITOCHU Cyber & Intelligence

# Earth Estries ?

- Trend Micro published an APT actor report on August, 2023.
- Collected domain list matches some domains in their IoCs.
- They named this Threat Actor as "Earth Estries".
- The infrastructure was almost same, but malware set is different.

Table 6. History of registered domains following the keyword "3280132818@qq.com"

| Domain keyword search: "3280132818@qq.com" | | |
|---|---|---|
| **Domain** | **Registers** | **Expires** |
| *mncdntech[.]com* | Jul 4, 2023 | Jul 4, 2024 |
| *substantialeconomy[.]com* | Jun 30, 2023 | May 25, 2024 |
| *jptomorrow[.]com* | Jun 19, 2023 | Apr 19, 2024 |
| *vultr-dns[.]com* | Jun 10, 2023 | Jun 10, 2024 |
| *jttoday[.]net* | May 21, 2023 | Mar 21, 2024 |

# From this report

- Related domains based on their investigation result.

- smartlinkcorp[.]net
- oxcdntech[.]com

Table 4. Information on C&C activities referenced with WHOIS protocol

| Domain | Registrant information |
|---|---|
| • *nx2.microware-help[.]com*<br>• *east.smartpisang[.]com* | • Registrar: Xin Net Technology Company<br>• Registrar: Bizcn, Inc. |
| *cdn728a66b0.smartlinkcorp[.]net* | • Organization: De Wang Mao Yi You Xian Gong Si (De Wang 貿易有限公司)<br>• City: Qinyuanshi (清遠市) |
| *cdn-6dd0035.oxcdntech[.]com* | Organizaton: De Wang Mao Yi You Xian Gong Si (De Wang 貿易有限公司) |
| *vultr-dns[.]com* | Email: 3280132818@qq.com |

# smartlinkcorp[.]net - whois

```
Domain Name: SMARTLINKCORP.NET
Registry Domain ID: 2687395568_DOMAIN_NET-VRSN
Registrar WHOIS Server: whois.internet.bs
Registrar URL: http://www.internetbs.net
Updated Date: 2022-04-30T11:23:23Z
Creation Date: 2022-04-06T22:47:55Z
Registrar Registration Expiration Date: 2023-04-06T22:47:55Z
Registrar: Internet Domain Service BS Corp.
Registrar IANA ID: 2487
Registrar Abuse Contact Email: abuse@internet.bs
Registrar Abuse Contact Phone: +1.5163015301
```

```
Registry Registrant ID: Not disclosed
Registrant Name: Not disclosed Not disclosed
Registrant Organization: de wang mao yi you xian gong si
Registrant Street: guang dong sheng qing yuan shi qing xin qu huan cheng lu 391hao
Registrant City: qingyuanshi
Registrant State/Province:
Registrant Postal Code: 511500
Registrant Country: CN
Registrant Phone: +86.07634320322
Registrant Phone Ext:
Registrant Fax:
Registrant Fax Ext:
Registrant Email: b7c3172628238d44113db94a72dfce7d.gdrp@customers.whoisprivacycorp.com
```

# oxcdntech[.]com - whois

```
Domain Name: OXCDNTECH.COM
   Registry Domain ID: 2755832342_DOMAIN_COM-VRSN
   Registrar WHOIS Server: whois.internet.bs
   Registrar URL: http://www.internet.bs
   Updated Date: 2023-02-14T02:21:19Z
   Creation Date: 2023-02-03T07:56:58Z
   Registry Expiry Date: 2024-02-03T07:56:58Z
   Registrar: Internet Domain Service BS Corp
   Registrar IANA ID: 2487
   Registrar Abuse Contact Email: abuse@internet.bs
   Registrar Abuse Contact Phone: +1.5163015301
```

```
Registry Registrant ID: Not disclosed
Registrant Name: Not disclosed Not disclosed
Registrant Organization: de wang mao yi you xian gong si
Registrant Street: guang dong sheng qing yuan shi qing xin qu huan cheng lu 391hao
Registrant City: qingyuanshi
Registrant State/Province:
Registrant Postal Code: 511500
Registrant Country: CN
Registrant Phone: +86.07634320322
Registrant Phone Ext:
Registrant Fax:
Registrant Fax Ext:
Registrant Email: b7c3172628238d44113db94a72dfce7d.gdrp@customers.whoisprivacycorp.com
```

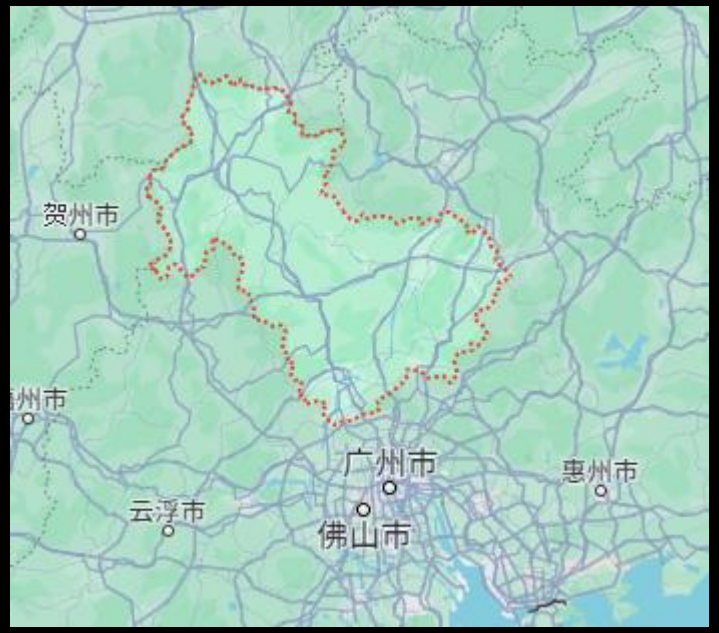
ITOCHU Cyber & Intelligence

# another domains

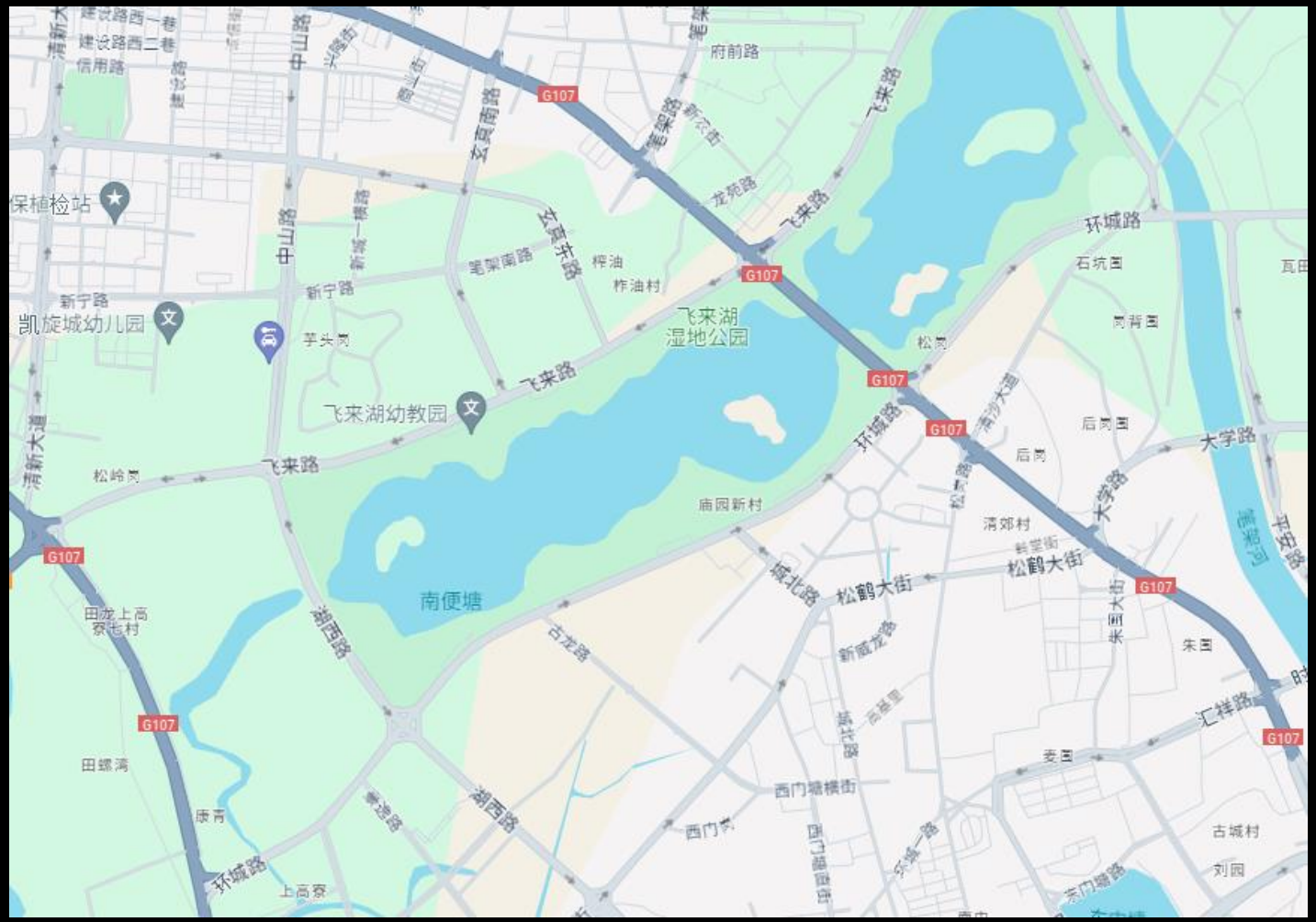
b7c3172628238d44113db94a72dfce7d.gdrp @ customers.whoisprivacycorp[.]com

| Domain | registration | Registrar | PrivacyMail | Name | Organization | Street | City | State | PostalCode | Country | Phone |
|---|---|---|---|---|---|---|---|---|---|---|---|
| protoncdn[.]net | 2023-05-10T02:19:20Z | Internet Domain Service BS Corp. | b7c3172628238d44113db94a72dfce7d.gdrp@customers.whoisprivacycorp[.]com | Not disclosed Not disclosed | li cai ling | mang shi nan bang lu 111 hao 2309 | de hong | | 678400 | CN | +86.06926176602 |
| oxcdntech[.]com | 2023-02-03T07:56:58Z | Internet Domain Service BS Corp | b7c3172628238d44113db94a72dfce7d.gdrp@customers.whoisprivacycorp[.]com | | de wang mao yi you xian gong si | guang dong sheng qing yuan shi qing xin qu huan cheng lu 391hao | qingyuanshi | | 511500 | CHINA | |
| rtsafetech[.]com | 2022-10-08T04:02:17Z | Internet Domain Service BS Corp | b7c3172628238d44113db94a72dfce7d.gdrp@customers.whoisprivacycorp[.]com | | de wang mao yi you xian gong si | guang dong sheng qing yuan shi qing xin qu huan cheng lu 391hao | qingyuanshi | | 511500 | CHINA | '8607634320322 |
| trhammer[.]com | 2022-09-03T15:34:54Z | Internet Domain Service BS Corp | b7c3172628238d44113db94a72dfce7d.gdrp@customers.whoisprivacycorp[.]com | | de wang mao yi you xian gong si | guang dong sheng qing yuan shi qing xin qu huan cheng lu 391hao | qingyuanshi | | 511500 | CHINA | |
| smartlinkcorp[.]net | 2022-04-06T22:47:55Z | Internet Domain Service BS Corp | b7c3172628238d44113db94a72dfce7d.gdrp@customers.whoisprivacycorp[.]com | | | | qingyuanshi | | 511500 | CHINA | 8607634320322 |
| keyplancorp[.]com | 2021-12-16T03:20:00Z | Internet Domain Service BS Corp | b7c3172628238d44113db94a72dfce7d.gdrp@customers.whoisprivacycorp[.]com | | de wang mao yi you xian gong si | guang dong sheng qing yuan shi qing xin qu huan cheng lu 391hao | qingyuanshi | | 511500 | CHINA | |
| rthtrade[.]com | 2021-11-23T02:15:11Z | Internet Domain Service BS Corp | b7c3172628238d44113db94a72dfce7d.gdrp@customers.whoisprivacycorp[.]com | | de wang mao yi you xian gong si | guang dong sheng qing yuan shi qing xin qu huan cheng lu 391hao | qingyuanshi | | 511500 | CHINA | |
| rtwebmaster[.]com | 2021-11-17T09:22:57Z | Internet Domain Service BS Corp | b7c3172628238d44113db94a72dfce7d.gdrp@customers.whoisprivacycorp[.]com | | de wang mao yi you xian gong si | guang dong sheng qing yuan shi qing xin qu huan cheng lu 391hao | qingyuanshi | | 511500 | CHINA | |

# address




guang dong sheng qing yuan shi
清远市 **広東省**




qing xin qu huan cheng lu
**清新県**　环城路

ITOCHU
Cyber &
Intelligence

# address


de hong mang shi
雲南省 德宏 芒市


nan bang lu
南蚌路

ITOCHU
Cyber &
Intelligence

# cdn83a2481b.smartlinkcorp[.]net

- 216.238.85.128

| | | | | | | |
|---|---|---|---|---|---|---|
| smartlinkcorp.net | | cdn728a66b0.smartlinkcorp.net | 139.84.232.200 | 2023/7/30 | NET-139-84-232-0-23 | Vultr Holdings, LLC (VHL-176) |
| | ■ | cdn728a66b0.smartlinkcorp.net | 38.54.115.128 | ? | COGENT-A | PSINet, Inc. (PSI) |
| | | cdn83a2481b.smartlinkcorp.net | 216.238.85.128 | 2023/7/30 | VULTR | VULTR-MEXICO (C07991065) |
| | ■ | cdn83a2481b.smartlinkcorp.net | 45.32.90.211 | ? | CONSTANT | The Constant Company, LLC (CHOOP-1) |
| | | cdn7ac0278b.smartlinkcorp.net | 103.151.229.198 | 2023/4/4 | BDTCL-CN | Beijing Dingbei Technology Co., Ltd. |
| | ■ | cdn7ac0278b.smartlinkcorp.net | 43.129.188.223 | ? | ACEVILLEPTELTD-SG | ACEVILLEPTELTD-SG |

# digitelela[.]com used this IP

- 216.238.85.128

| | | | | | | |
|---|---|---|---|---|---|---|
| digitelela.com | | cdn-e01adc24.digitelela.com | 178.157.63.86 | 2023/7/30 | CA-CL-20100625 | Cluster Logic Inc |
| | ■ | cdn-e01adc24.digitelela.com | 45.76.148.187 | ? | CONSTANT | The Constant Company, LLC (CHOOP-1) |
| | | cdn-8a891bac.digitelela.com | 64.176.7.118 | 2023/7/30 | NET-64-176-6-0-23 | Vultr Holdings, LLC (VHL-176) |
| | | cdn-8a891bac.digitelela.com | 155.138.137.138 | 2023/6/5 | NET-155-138-136-0-23 | Vultr Holdings, LLC (VHL-176) |
| | | cdn-8a891bac.digitelela.com | 216.238.85.128 | 2023/4/3 | VULTR | VULTR-MEXICO (C07991065) |
| | ■ | cdn-8a891bac.digitelela.com | 45.32.201.100 | ? | CONSTANT | The Constant Company, LLC (CHOOP-1) |
| | | cache.digitelela.com | 64.176.2.58 | 2023/2/23 | CHOOP-1 | The Constant Company, LLC (CHOOP-1) |
| | | cache.digitelela.com | 154.201.144.38 | 2022/11/26 | Digital_Core_Technology_Co_Ltd | Digital Core Technology Co., Ltd |
| | | cache.digitelela.com | 67.219.97.188 | 2022/4/11 | CONSTANT | The Constant Company, LLC (CHOOP-1) |
| | ■ | cache.digitelela.com | 14.128.37.27 | ? | CTG128-32-HK | CTG Server Ltd. |
| | | root.digitelela.com | 64.176.7.118 | 2023/7/30 | NET-64-176-6-0-23 | Vultr Holdings, LLC (VHL-176) |
| | | root.digitelela.com | 104.238.153.140 | 2023/5/28 | CONSTANT | The Constant Company, LLC (CHOOP-1) |
| | | root.digitelela.com | 103.133.137.74 | 2022/11/26 | ZNDATA-CN | xiamen zhongheng Technology Ltd |
| | | root.digitelela.com | 163.197.34.221 | 2022/5/14 | ANGLONET | Anglo American South Africa (Pty) Ltd |
| | | root.digitelela.com | 45.134.1.161 | 2021/12/14 | ACEVILLEPTELTD-SG | ACEVILLEPTELTD-SG |
| | ■ | root.digitelela.com | 103.43.188.63 | ? | WEST263GO-HK | West263 International Limited |
| | | web.digitelela.com | 103.139.3.23 | 2022/1/30 | yuquwangluo | Cloud Yuqu LLC |
| | | ssl.digitelela.com | 45.158.35.32 | 2021/12/13 | Prager-Connect-GmbH | Prager-Connect-GmbH |
| | ■ | nsroot.digitelela.com | 101.78.177.248 | ? | HKBNES-HK | HKBN Enterprise Solutions HK Limited |
| | | ns4.digitelela.com | 8.8.4.4 | 2023/10/5 | GOGL | Google LLC (GOGL) |
| | | ns4.digitelela.com | 8.8.8.8 | 2023/4/18 | GOGL | Google LLC (GOGL) |
| | | ns1.digitelela.com | 8.8.4.4 | 2022/8/26 | GOGL | Google LLC (GOGL) |
| | ■ | ns1.digitelela.com | 101.78.177.227 | ? | HKBNES-HK | HKBN Enterprise Solutions HK Limited |

# digitelela[.]com - whois

```
Registrar:
  Handle: 2645196483_DOMAIN_COM-VRSN
  LDH Name: digitelela.com
  Nameserver:
    LDH Name: control.dnspod.net
      Status: Active
  Nameserver:
    LDH Name: scheat.dnspod.net
      Status: Active
  SecureDNS:
    Delegation Signed: false
  Event:
    Action: registration
    Date: 2021-10-03T01:00:40Z
  Event:
    Action: expiration
    Date: 2022-10-03T01:00:40Z
  Event:
    Action: last changed
    Date: 2021-10-03T01:00:40.679Z
  Event:
    Action: last update of RDAP database
    Date: 2021-10-05T02:39:14.781Z
    Status: client transfer prohibited
```

```
Entity:
  Role:
    REGISTRANT
    ADMIN
    TECHNICAL
  VCard:
    Kind: individual
    Formatted Name: Li Caiwang
    Address:
      Street:


        tianjin
        Locale: tianjin
        Region: TIANJIN
        Postal Code: 301600
        Country: CN
    Email: 3087384364@qq.com
        TYPE: voice
    Status: Active
```

3087384364 @ qq[.]com

ITOCHU Cyber & Intelligence

# 3087384364@qq[.]com

| Domain | registration | Registrar | Email | Name | Organization | Street | City | State | PostalCode | Country | Phone |
|---|---|---|---|---|---|---|---|---|---|---|---|
| sprintappcdn[.]com | 2023-07-28T09:20:28+02:00 | PSI-USA, Inc. dba Domain Robot | 3087384364@qq[.]com | Huang Junxi | Huang Junxi | yunnan | yunnan | YUNNAN | 666100 | CN | +86.8712571 |
| hammercdntech[.]com | 2023-02-01T09:10:52Z | PDR Ltd. d/b/a PublicDomainRegistry[.]com | 3087384364@qq[.]com | Li Caiwang | | | tianjin | TIANJIN | 301600 | CN | +86.85730333 |
| z7-tech[.]com | 2022-05-07T13:12:12Z | PDR Ltd. d/b/a PublicDomainRegistry[.]com | 3087384364@qq[.]com | Li Caiwang | | | tianjin | TIANJIN | 301600 | CN | +86.85730333 |
| linkaircdn[.]com | 2022-04-06T14:56:21Z | PDR Ltd. d/b/a PublicDomainRegistry[.]com | 3087384364@qq[.]com | Li Caiwang | | | tianjin | TIANJIN | 301600 | CN | +86.85730333 |
| rtsoftcorp[.]com | 2022-03-14T01:31:21Z | PDR Ltd. d/b/a PublicDomainRegistry[.]com | 3087384364@qq[.]com | Li Caiwang | | | tianjin | TIANJIN | 301600 | CN | +86.85730333 |
| publicdnsau[.]com | 2022-03-08T02:11:58Z | PDR Ltd. d/b/a PublicDomainRegistry[.]com | 3087384364@qq[.]com | Li Caiwang | | | tianjin | TIANJIN | 301600 | CN | +86.85730333 |
| uswatchcorp[.]com | 2022-01-30T09:08:52Z | PDR Ltd. d/b/a PublicDomainRegistry[.]com | 3087384364@qq[.]com | Li Caiwang | | | tianjin | TIANJIN | 301600 | CN | +86.85730333 |
| anynucleus[.]com | 2021-11-16T07:12:23Z | PDR Ltd. d/b/a PublicDomainRegistry[.]com | 3087384364@qq[.]com | Li Caiwang | Li Caiwang | tianjin | tianjin | TIANJIN | 301600 | CHINA | 8685730333 |
| digitelela[.]com | 2021-10-03T01:00:40Z | PDR Ltd. d/b/a PublicDomainRegistry[.]com | 3087384364@qq[.]com | Li Caiwang | | | tianjin | TIANJIN | 301600 | CN | +86.85730333 |
| dns2021[.]net | 2021-02-27T15:59:16Z | PDR Ltd. d/b/a PublicDomainRegistry[.]com | 3087384364@qq[.]com | Li Caiwang | | | tianjin | TIANJIN | 301600 | CN | +86.85730333 |
| lyncidc[.]com | 2019-08-19T08:00:32Z | NameSilo, LLC | 3087384364@qq[.]com | Li Caiwang | | South Haiwan Road | Zhanjiang | guangdong | 524000 | CHINA | 8675983913440 |
| ayeglobal[.]com | 2019-02-15T17:28:41Z | NameSilo, LLC | 3087384364@qq[.]com | Li Caiwang | | South Haiwan Road | Zhanjiang | guangdong | 524000 | CN | 8675983913440 |
| ayugroup[.]com | 2019-02-15T07:23:59Z | NameSilo, LLC | 3087384364@qq[.]com | Li Caiwang | | South Haiwan Road | Zhanjiang | guangdong | 524000 | CN | 8675983913440 |

# Related Domains

- 2 more domains are also discovered. (another e-mail address found 3563759251 @qq[.]com)

| | | | | | | | | | | | |
|---|---|---|---|---|---|---|---|---|---|---|---|
| trade2021[.]net | 2021-02-23T07:21:19Z | PDR Ltd. d/b/a PublicDomainRegistry[.]com | 3563759251 @qq[.]com | Li Caiwang | Li Caiwang | tianjin | tianjin | TIANJIN | 301600 | CHINA | 86857330333 |
| ipv6to4dns[.]com | 2019-07-30T01:15:54Z | PDR Ltd. d/b/a PublicDomainRegistry[.]com | 3563759251 @qq[.]com | Li Caiwang | Li Caiwang | tianjin | tianjin | TIANJIN | 301600 | CHINA | 86857330333 |

# Characteristics of infra

- The attacker set up IP address resolution for only subdomains.

```
> chickenchickengo.com
Server:  UnKnown
Address:  192.168.201.10

*** No internal type for both IPv4 and IPv6 Addresses (A+AAAA) records available for chickenchickengo.com
> cdn-7391.chickenchickengo.com
Server:  UnKnown
Address:  192.168.201.10

Non-authoritative answer:
Name:    cdn-7391.chickenchickengo.com
Address:  43.245.198.161

> cdn-7391.chickenchickengo.com
Server:  UnKnown
Address:  192.168.201.10

Non-authoritative answer:
Name:    cdn-7391.chickenchickengo.com
Address:  43.245.198.161
```

# Characteristics of infra

- Most of the C2 servers were used for a single DNS resolution.

| subdomain | IP | Latest | NetName | Org. |
|---|---|---|---|---|
| cdn19b0.truecdnnetwork[.]com | 38.54.20[.]12 | 2023/11/30 | COGENT-A | PSINet, Inc. (PSI) |
| cdn4b67.sprintappcdn[.]com | 158.247.230[.]187 | 2023/7/30 | CONSTANT-AP | The Constant Company, LLC |
| host0249.mncdntech[.]com | 135.181.91[.]249 | 2023/9/15 | DE-HETZNER-19931109 | Hetzner Online GmbH |
| cdn-7391.chickenchickengo[.]com | 64.176.224[.]148 | 2023/8/14 | CHOOP-1 | The Constant Company, LLC (CHOOP-1) |
| cdn-7391.chickenchickengo[.]com | 38.60.146[.]188 | 2023/9/3 | KAOPUCLOUD-SG | KaopuCloud-SG (C09203248) |
| cdn-7391.chickenchickengo[.]com | 43.245.198[.]161 | 2023/10/4 | IPTELECOM-SG | IPTELECOM Singapore Network |
| cdn-82d1.vultr-dns[.]com | 154.39.157[.]152 | 2023/7/7 | MEGALAYER-GCNT-NET-1 | HONG KONG Megalayer Technology Co.,Limited (C07949252) |
| cdn-7a3d.vultr-dns[.]com | 64.176.7[.]172 | 2023/6/19 | NET-64-176-6-0-23 | Vultr Holdings, LLC (VHL-176) |
| cdn09198213.protoncdn[.]net | 149.28.159[.]228 | 2023/8/29 | CONSTANT | The Constant Company, LLC (CHOOP-1) |
| cdn09198213.protoncdn[.]net | 23.105.214[.]237 | ? | IN-27 | IT7 Networks Inc (IN-27) |
| cdn44820125.protoncdn[.]net | 64.176.2[.]236 | 2023/9/17 | CHOOP-1 | The Constant Company, LLC (CHOOP-1) |
| cdn44820125.protoncdn[.]net | 37.235.53[.]164 | ? | EDIS-ES-NET | EDIS Infrastructure in Spain |
| cdn39a700bb.jptomorrow[.]com | 154.39.137[.]42 | 2023/7/27 | MEGALAYER-GCNT-NET-1 | HONG KONG Megalayer Technology Co.,Limited (C07949252) |
| cdn39a700bb.jptomorrow[.]com | 27.124.24[.]182 | ? | CTG124-24-HK | CTG Server Ltd. |
| cdn1ac7bdd3.jptomorrow[.]com | 103.43.188[.]63 | 2023/9/13 | WEST263GO-HK | West263 International Limited |
| cdn1ac7bdd3.jptomorrow[.]com | 27.124.24[.]147 | ? | CTG124-24-HK | CTG Server Ltd. |

# Characteristics of infra

Set NS* : 8.8.8.8 or 8.8.4.4

| | | |
|---|---|---|
| ns1.chickenchickengo[.]com | 8.8.8.8 | 2023/9/6 |
| ns01.vultr-dns[.]com | 8.8.8.8 | 2023/9/11 |
| ns09198213.protoncdn[.]net | 8.8.8.8 | 2023/8/15 |
| ns1.hammercdntech[.]com | 8.8.8.8 | 2023/6/9 |
| ns2.rtsafetech[.]com | 8.8.8.8 | 2023/6/13 |
| ns1.trhammer[.]com | 8.8.4.4 | 2022/10/24 |
| ns1.bogotatrade.co | 8.8.4.4 | 2023/9/22 |
| ns.z7-tech[.]com | 8.8.8.8 | 2023/6/5 |
| ns4.digitelela[.]com | 8.8.4.4 | 2023/10/5 |
| ns4.digitelela[.]com | 8.8.8.8 | 2023/4/18 |
| ns1.digitelela[.]com | 8.8.4.4 | 2022/8/26 |
| ns1.ipv6to4dns[.]com | 8.8.4.4 | 2022/2/10 |

# from 2019

- We suspect this threat actor works more than 5 years.
- We compare ShdowPad C2 IoC list by Carbonblack and some IPs are found their list.
- And 2021 to 2022, they may use ShadowPad for their attacks.

| c2_ip | first_seen | last_seen | protocol | port | version | | MATCH | |
|---|---|---|---|---|---|---|---|---|
| 15 | 43.129.188[.]223 | 2021/10/17 | 2022/10/16 | TCP | 443 | Variant1 | 65 | cdn7ac0278b.smartlinkcorp[.]net |
| 36 | 149.127.176[.]12 | 2022/6/22 | 2022/6/22 | HTTP | 80 | Variant3 | 77 | nsroot.uswatchcorp[.]com |
| 47 | 103.209.233[.]172 | 2022/1/30 | 2022/1/30 | HTTP | 443 | Variant3 | 131 | pro.smartpisang[.]com |
| 56 | 149.127.176[.]12 | 2022/6/10 | 2022/6/10 | HTTP | 443 | Variant3 | 77 | nsroot.uswatchcorp[.]com |

# Conclusions

# Conclusions

- Please, manage all accounts carefully including providing outside
- BloodAlchemy is a new variant of Deed RAT and the origin was Shadowpad
- Infra has many characteristics, when you find suspicious please check it!


ITOCHU
Cyber &
Intelligence

# IoCs

| File Name | MD5 | Malware Type |
|---|---|---|
| BrDifxapi.exe | fb1edee3efa630a4617f20c7b47844b5 | BloodAlchemy |
| BrLogAPI.dll | 24a208ef3a8cc1c41ec8f5dce8915730 | |
| DIFX | 2c485c80588d71e799d50b497e0723c9 | |
| dxgi.cfg | 5576c0993488db967c298af9a84b0c3d | CobaltStrike |
| dxgi.dll | 30427ce36107594f187ea9821b553423 | |
| sfc.exe | 6c3045560465c27cd845b004dde11c5e | |

| IP/Domain | Watermark | Malware Type |
|---|---|---|
| cdn1ac7bdd3.jptomorrow[.]com | - | BloodAlchemy |
| cdn-hk-6dc8.bogotatrade[.]co | - | BloodAlchemy<br>By Elastic |
| 121.41.35[.]65 | 426352781 | CobaltStrike |
| cdn39a700bb.jptomorrow[.]com | 2029527128 | |


Thanks for Listening ☺
ishimaru-suguru☺itochu.co.jp
niwa-yus☺itochu.co.jp
motohiko-sato☺itochu.co.jp

# References

https://www.macnica.co.jp/business/security/manufacturers/files/mpressioncss_ta_report_2020_5_en.pdf
https://www.macnica.co.jp/business/security/manufacturers/files/mpressioncss_ta_report_2019_4_en.pdf
https://www.uscc.gov/sites/default/files/2022-11/Chapter_3_Section_2--Chinas_Cyber_Capabilities.pdf
https://symantec-enterprise-blogs.security.com/blogs/threat-intelligence/critical-infrastructure-attacks
https://go.recordedfuture.com/hubfs/reports/cta-2023-0808.pdf
https://go.recordedfuture.com/hubfs/reports/cta-2021-0228.pdf
https://www.welivesecurity.com/2020/12/10/luckymouse-ta428-compromise-able-desktop/
https://conference.hitb.org/hitbsecconf2021sin/materials/D1T1%20-%20%20ShadowPad%20-%20A%20Masterpiece%20of%20Privately%20Sold%20Malware%20in%20Chinese%20Espionage%20-%20Yi-Jhen%20Hsieh%20&%20Joey%20Chen.pdf