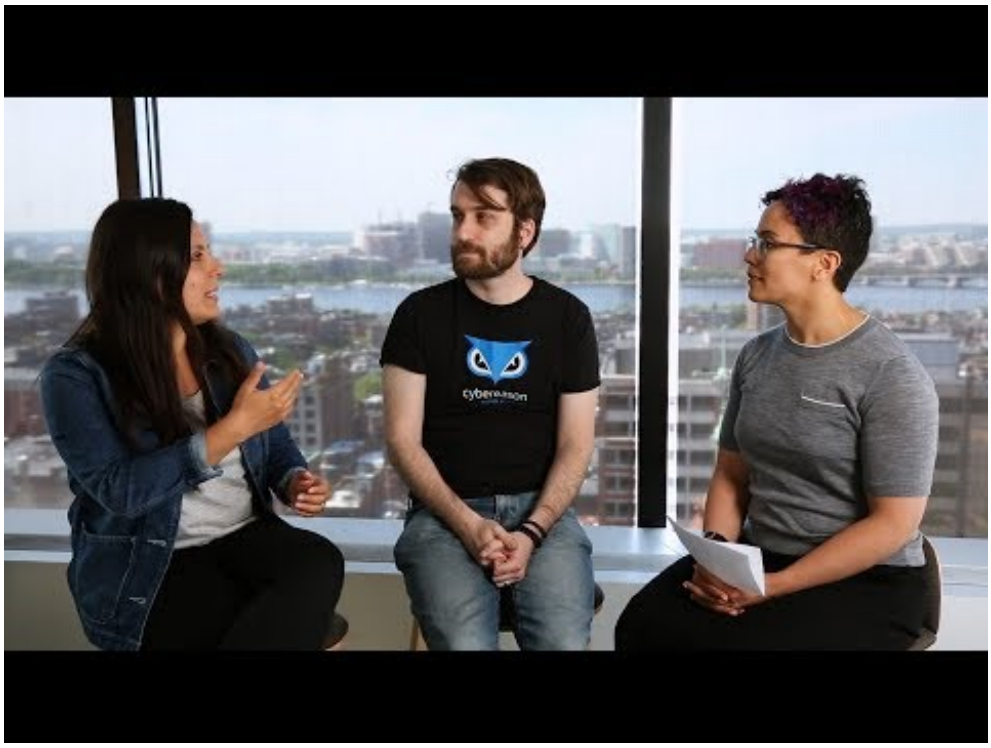
Operation Soft Cell: A Worldwide Campaign Against Telecommunications Providers

cybereason.com/blog/operation-soft-cell-a-worldwide-campaign-against-telecommunications-providers

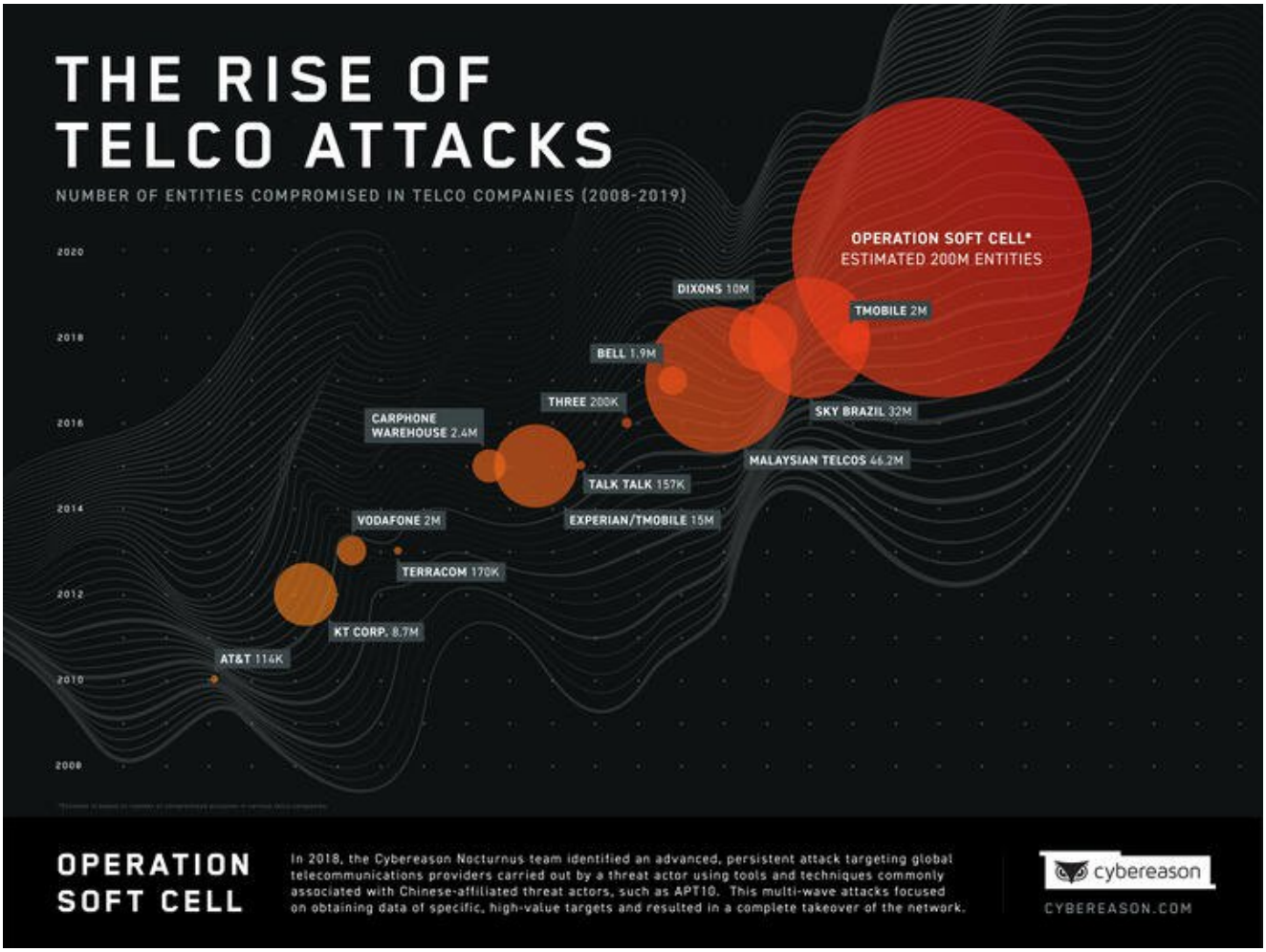


https://youtu.be/Ihpn2-4jTvc

Research by: Mor Levi, Assaf Dahan, and Amit Serper

EXECUTIVE SUMMARY

In 2018, the Cybereason Nocturnus team identified an advanced, persistent attack targeting global telecommunications providers carried out by a threat actor using tools and techniques commonly associated with Chinese-affiliated threat actors, such as APT10. This multi-wave attacks focused on obtaining data of specific, high-value targets and resulted in a complete takeover of the network.



Key Points

- Earlier this year, Cybereason identified an advanced, persistent attack targeting telecommunications providers that has been underway for years, soon after deploying into the environment.
- Cybereason spotted the attack and later supported the telecommunications provider through four more waves of the advanced persistent attack over the course of 6 months.
- Based on the data available to us, Operation Soft Cell has been active since at least 2012, though some evidence suggests even earlier activity by the threat actor against telecommunications providers.
- The attack was aiming to obtain CDR records of a large telecommunications provider.
- The threat actor was attempting to steal all data stored in the active directory, compromising every single username and password in the organization, along with other personally identifiable information, billing data, call detail records, credentials, email servers, geo-location of users, and more.
- The tools and TTPs used are commonly associated with Chinese threat actors
- During the persistent attack, the attackers worked in waves- abandoning one thread

of attack when it was detected and stopped, only to return months later with new tools and techniques.

Security Recommendations

- Add an additional security layer for web servers. For example, use WAF (Web Application FW) to prevent trivial attacks on Internet-facing web servers.
- Expose as few systems or ports to the Internet as possible. Make sure that all web servers and web services that are exposed are patched.
- Use an EDR tool to give visibility and immediate response capabilities when high severity incidents are detected.
- Proactively hunt in your environment for sensitive assets periodically.

Table of Contents

INTRODUCTION

Watch our CEO Lior Div's keynote on the operation.

In 2018, 30% of the telecommunications providers reported sensitive customer information was stolen due to an attack. These telecommunications providers have been expanding in size, to the point where In the past thirteen years, mobile cellular phone subscribers have quadrupled in size and sit at 8 billion subscribers today. Due to their wide availability and the fundamental service they bring, telecommunications providers have become critical infrastructure for the majority of world powers.

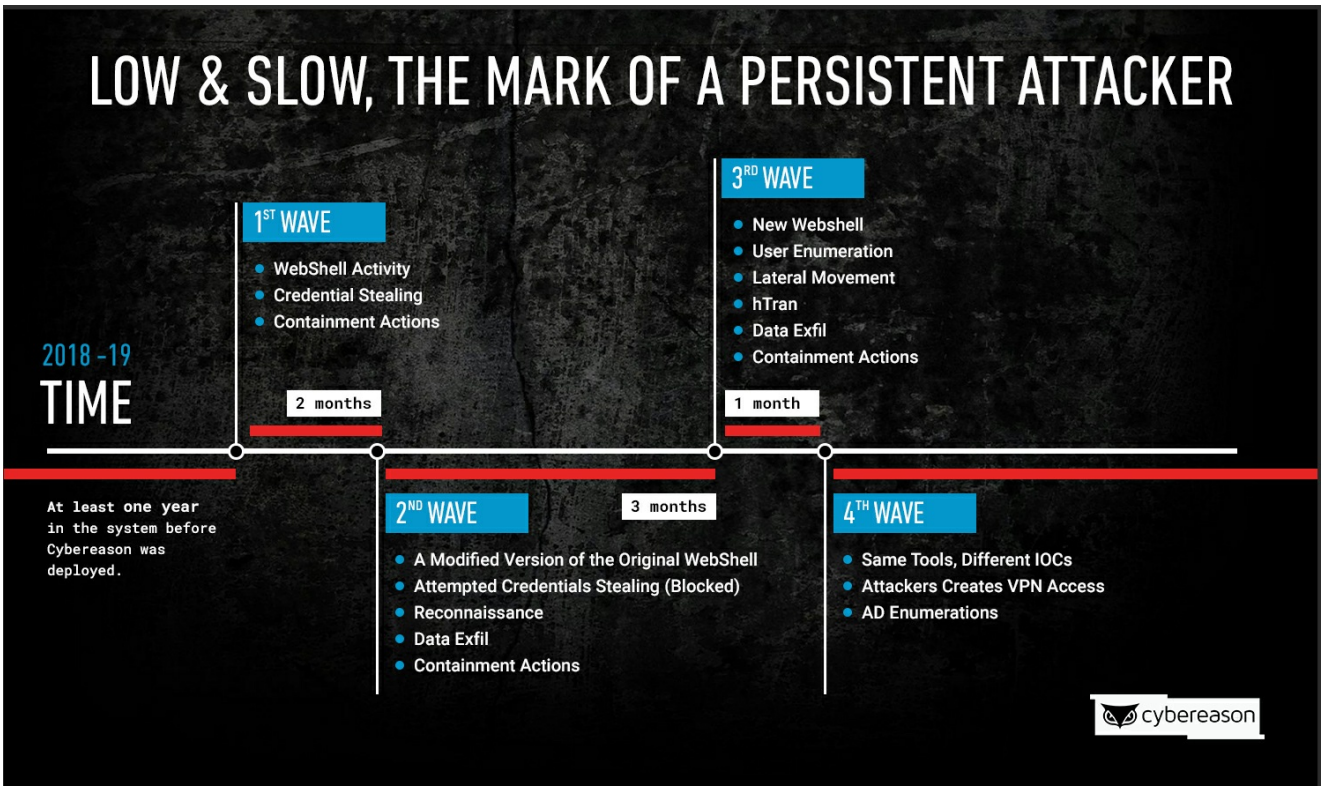
Much like telecommunication providers, many other critical infrastructure organizations provide a valuable targets for nation state threat actors, due to their high impact. In studies, nearly a quarter of critical infrastructure organizations reported they had been hit by nation state attacks and 60% said disruptive cyber attacks are among the threats they are most worried about.

Threat actors, especially those at the level of nation state, are seeking opportunities to attack these organizations, conducting elaborate, advanced operations to gain leverage, seize strategic assets, and collect information. When successful, these attacks often have huge implications.

Last year, we identified a threat actor that has been operating in telecommunications provider environments for at least two years. We performed a post-incident review of the attacks and were able to identify changes in the attack patterns along with new activity every quarter.

The threat actor mainly sought to obtain CDR data (call logs, cell tower locations, etc.) belonging to specific individuals from various countries. This type of targeted cyber espionage is usually the work of nation state threat actors.

We've concluded with a **high level of certainty** that the threat actor is affiliated with China and is likely state sponsored. The tools and techniques used throughout these attacks are consistent with several Chinese threat actors, such as APT10, a threat actor believed to operate on behalf of the Chinese Ministry of State Security (MSS).



The threat actor changed activity every quarter.

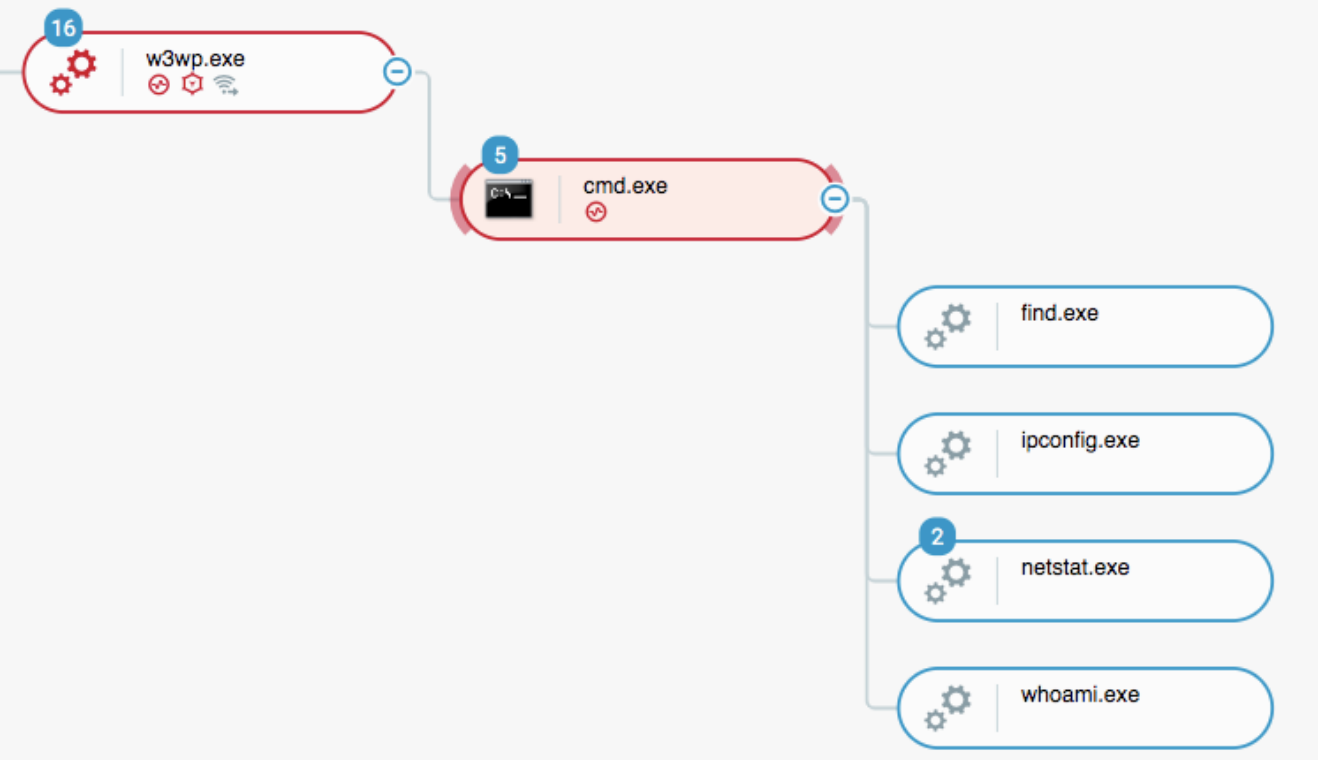
The attack began with a web shell running on a vulnerable, publicly-facing server, from which the attackers gathered information about the network and propagated across the network. The threat actor attempted to compromise critical assets, such as database servers, billing servers, and the active directory. As malicious activity was detected and remediated against, the threat actor stopped the attack.

The second wave of the attack hit several months later with similar infiltration attempts, along with a modified version of the web shell and reconnaissance activities. A game of cat and mouse between the threat actor and the defenders began, as they ceased and resumed their attack 2 more times in the span of a 4 month period.

Anatomy of the Attack

Initial Compromise: the Modified China Chopper Web Shell

The initial indicator of the attack was a malicious web shell that was detected on an IIS server, coming out of the *w3wp.exe* process. An investigation of the web shell, later classified as a modified version of the China Chopper web shell, uncovered several attack phases and TTPs. The threat actor was able to leverage the web shell to run reconnaissance commands, steal credentials, and deploy other tools.



Malicious web shell activity as observed in the Cybereason solution.

```
"cmd" /c cd /d "C:\Program Files\Microsoft\Exchange Server\V15\FrontEnd\HttpProxy\owa\auth\"&whoami&echo [S]&cd&echo [E]
"cmd" /c cd /d "C:\Program Files\Microsoft\Exchange Server\V15\FrontEnd\HttpProxy\owa\auth\"&netstat -oan&echo [S]&cd&echo [E]
"cmd" /c cd /d "C:\Program Files\Microsoft\Exchange Server\V15\FrontEnd\HttpProxy\owa\auth\"&ipconfig /all&echo [S]&cd&echo [E]
```

Commands executed via a modified version of the China Chopper web shell.

China Chopper is a web shell first discovered in 2012 that is commonly used by malicious Chinese actors. It is used to remotely control web servers, and has been used in many attacks against Australian web hosting providers. The web shell parameters in this attack match to the China Chopper parameters, as described in FireEye's analysis of China Chopper. This tool has been used by several Chinese-affiliated threat actors, such as APT 27 and APT 40. It is important to note that this tool is widely available and can be used by other threat actors.

Reconnaissance and Credential Stealing

The threat actor launched a series of reconnaissance commands to try to obtain and enumerate information about the compromised machine, network architecture, users, and active directory enumeration.

```
> "cmd" /c cd /d "C:\Program Files\Microsoft\Exchange Server\V15\FrontEnd\HttpProxy\owa\auth\"&whoami&echo [S]&cd&echo [E]
> "cmd" /c cd /d "C:\Program Files\Microsoft\Exchange Server\V15\FrontEnd\HttpProxy\owa\auth\"&query user&echo [S]&cd&echo [E]
> "cmd" /c cd /d "C:\Program Files\Microsoft\Exchange Server\V15\FrontEnd\HttpProxy\owa\auth\"&netstat -oan&echo [S]&cd&echo [E]
> "cmd" /c cd /d "C:\Program Files\Microsoft\Exchange Server\V15\FrontEnd\HttpProxy\owa\auth\"&ipconfig /all&echo [S]&cd&echo [E]
```

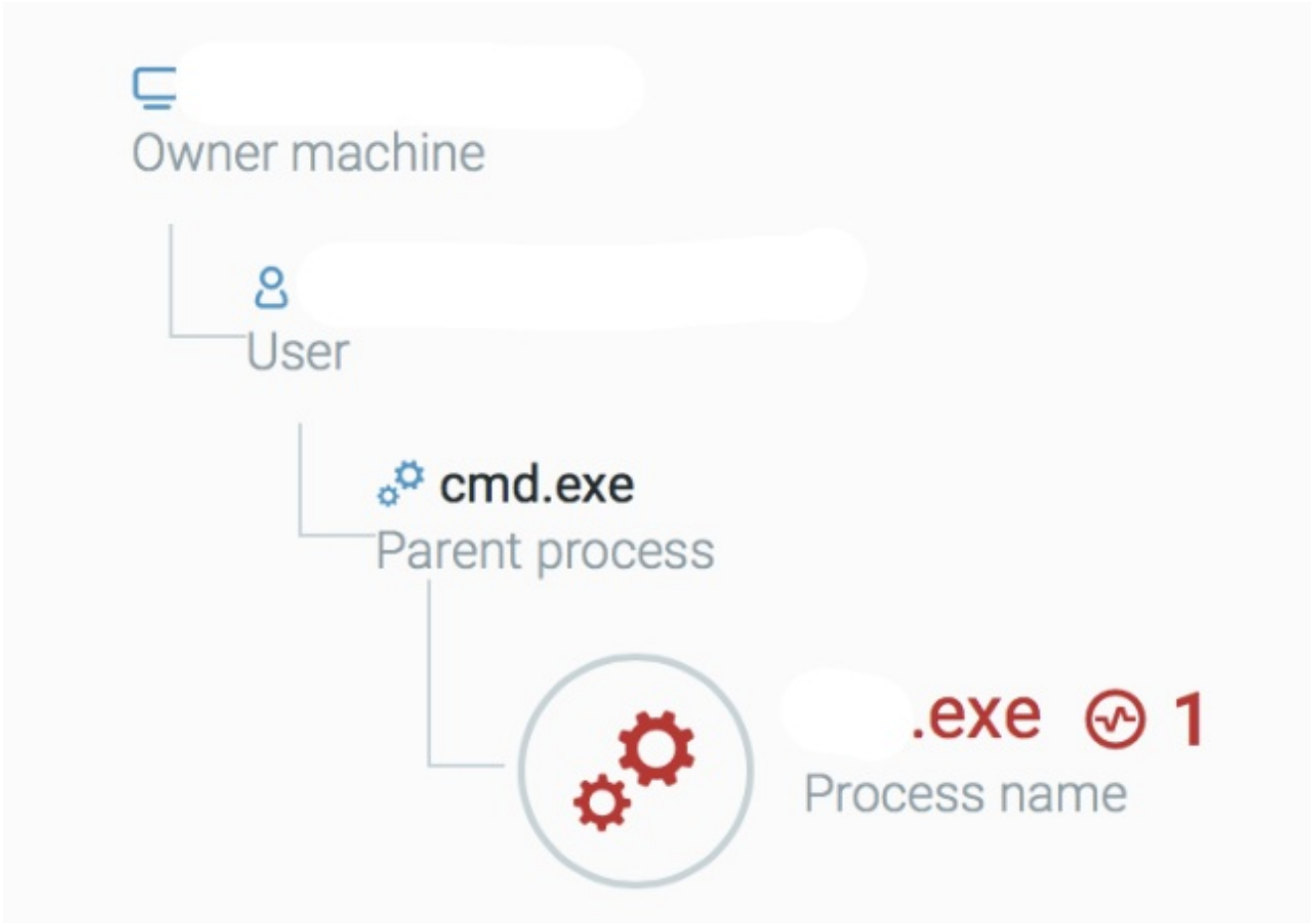
Example 1: Reconnaissance Commands

```
"powershell" /Command "cmd.exe /c 'cd /d C:\inetpub\wwwroot\&whoami&echo [S]&cd&echo [E]"
"powershell" /Command "cmd.exe /c 'cd /d C:\inetpub\wwwroot\&netstat -an | find \"ESTABLISHED\"&echo [S]&cd&echo [E]"
"powershell" /Command "cmd.exe /c 'cd /d C:\inetpub\wwwroot\&ipconfig /all&echo [S]&cd&echo [E]"
```

Example 2: Reconnaissance Commands

Modified “nbtscan”

One of the reconnaissance commands was to run a modified nbtscan tool ("NetBIOS nameserver scanner") to identify available NetBIOS name servers locally or over the network. Nbtscan has been used by APT10 in Operation Cloud Hopper to search for services of interest across the IT estate and footprint endpoints of interest. It is also capable of identifying system information.



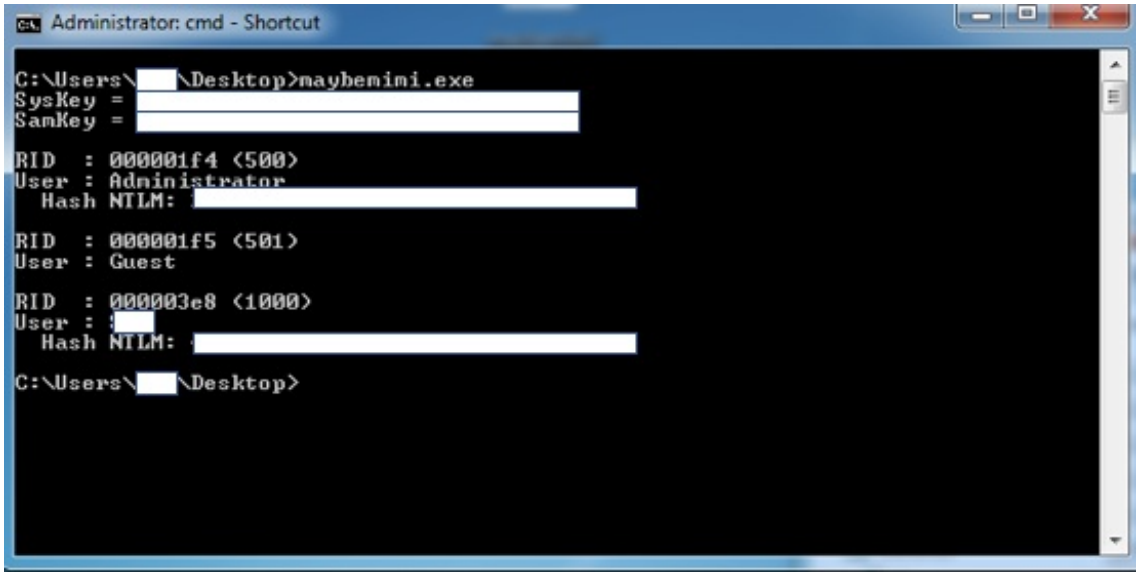
NetBIOS Scanner execution as seen in the Cybereason solution.

```
"cmd" /c cd /d C:\[redacted]&[redacted]10.[redacted].1/24&echo [S]&cd&echo [E]
```

NetBIOS scanner is set to scan an internal IP range.

Modified Mimikatz

Following the reconnaissance phase, the threat actor attempted to dump credentials stored on the compromised machines. The most common credential stealing tool used by the threat actor was a modified mimikatz that dumps NTLM hashes. This version of mimikatz did not require any command line arguments, most likely in an attempt to avoid detection based on command-line auditing. The dumped hashes were used to authenticate to other machines via pass the hash. We renamed this sample to *maybemimi.exe*.



Modified Mimikatz that dumps NTLM hashes.

Reverse engineering shows the similarity between maybemimi.exe and mimikatz.

```
        }
        break;
    default:
        PRINT_ERROR(L"Unknow SAM_HASH revision (%hu)\n", pHash->Revision);
    }
    if(status)
        kuhl_m_lsadump_dcsync_decrypt(cypheredHashBuffer.Buffer, cypheredHashBuffer.Le
    if(cypheredHashBuffer.Buffer)
        LocalFree(cypheredHashBuffer.Buffer);
```

Mimikatz code from GitHub.

```
                align 20h
; char aErrorUnknowSam[]
aErrorUnknowSam:                        ; DATA XREF: sub_140002704+77↑o
                text "UTF-16LE", 'ERROR Unknow SAM_HASH revision (%hu)',0Ah,0
                align 10h
aNtlm:                                  ; DATA XREF: sub_140002704+209↑o
                text "UTF-16LE", 'ntlm',0
                align 20h
aNtlm_0:                                ; DATA XREF: sub_140002704+213↑o
                text "UTF-16LE", 'NTLM',0
                align 10h
aLm_0:                                  ; DATA XREF: sub_140002704+225↑o
                text "UTF-16LE", 'lm  ',0
                align 20h
aLm:                                    ; DATA XREF: sub_140002704+21A↑o
                text "UTF-16LE", 'LM  ',0
                align 10h
```

maybemimi strings.

Dumping the SAM Hive from the Registry

In order to obtain credentials, the threat actor used another technique that can be seen in the below screenshots. They dumped specific hives from the Windows Registry, such as the SAM hive, which contains password hashes.

Reg.exe is being spawned from a shell process.

Command-line arguments indicate SAM hive dumping.

Lateral Movement

Once the threat actor mapped the network and obtained credentials (through net use), they began to move laterally. They were able to compromise critical assets including production servers and database servers, and they even managed to gain full control of the Domain Controller. The threat actor relied on WMI and PsExec to move laterally and install their tools across multiple assets.

The following example demonstrates how the threat actor moved laterally from the first machine, compromised by the modified version of the China Chopper web shell, to other machines inside the network.

/c cd /d "C:\Program Files\Microsoft\Exchange Server\V15\FrontEnd\HttpProxy\ecp\auth\"&wmic /node:[REDACTED] /user:"[REDACTED]" /password:"[REDACTED]" process call create a.bat&echo [S]&cd&echo [E]

WMI command used by the threat actor to move laterally.

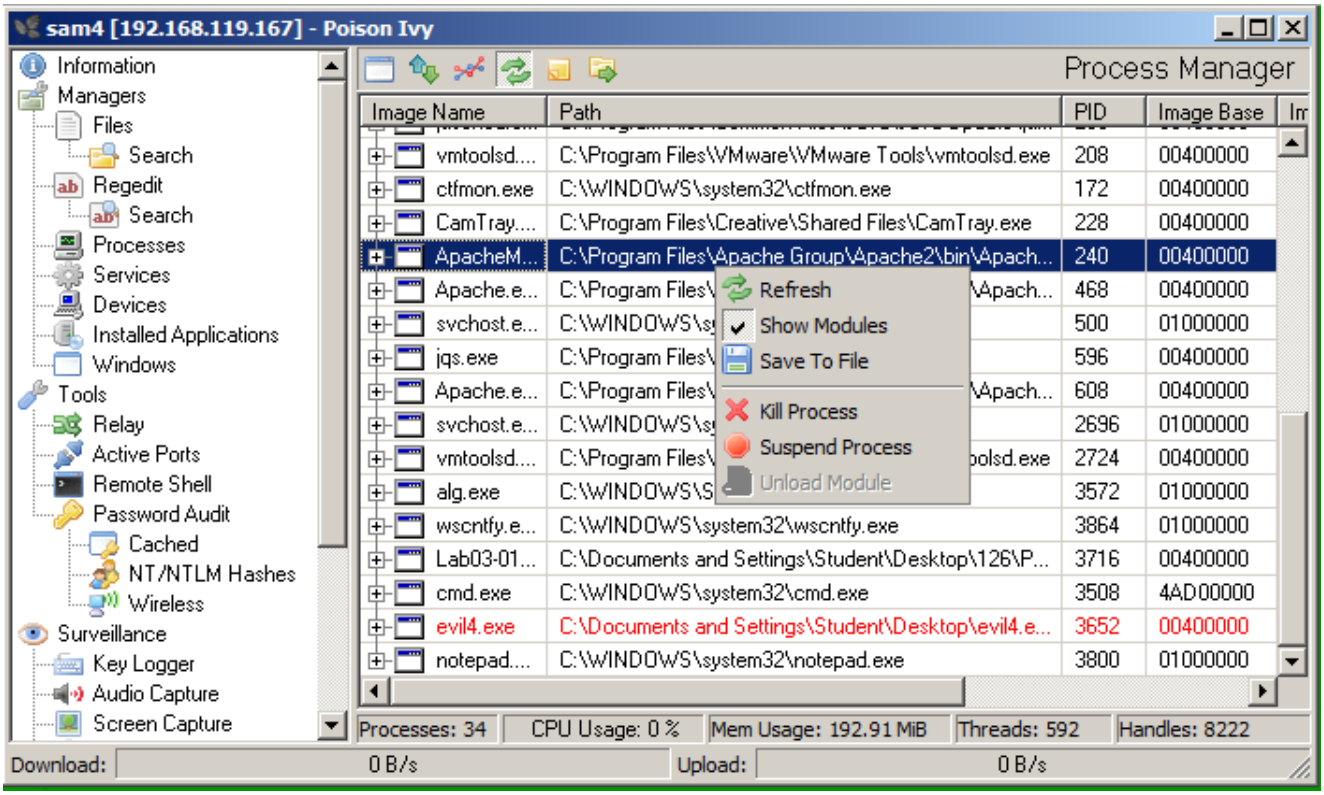
Maintaining a Long-term Foothold and Stealing Data

The threat actor abused the stolen credentials to create rogue, high-privileged domain user accounts which they then used to take malicious action. By creating these accounts, they ensured they would maintain access between different waves of the attack. Once the threat actor regains their foothold, they already have access to a high-privileged domain user account. This significantly reduces the "noise" of having to use credential dumpers repeatedly, which helped them evade detection.

PoisonIvy

A second method the threat actor used to maintain access across the compromised assets was through the deployment of the PoisonIvy RAT (PIVY). This infamous RAT has been associated with many different Chinese threat actors, including APT10, APT1, and DragonOK. It is a powerful, multi-featured RAT that lets a threat actor take total control over a machine. Among its most notable features are:

- Registry Editor
- Screenshot Grabber
- Credential Stealer
- Interactive Shell
- File Manager with Upload and Download Support
- Process Monitor
- Keylogging and Various other Surveillance Features



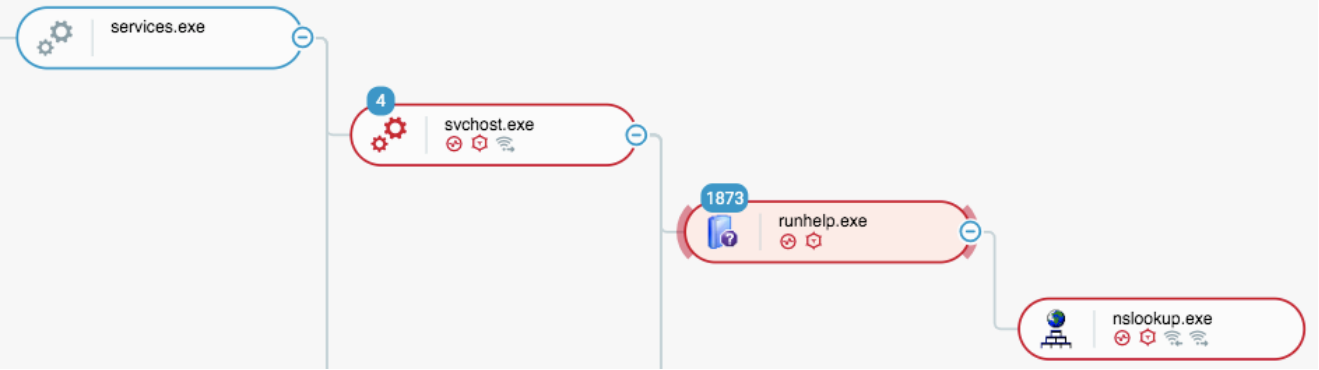
The control panel for PoisonIvy.

Courtesy of Sam Bowne - samsclass.info

We assume the threat actor used PoisonIvy for keylogging and other surveillance features, as they had that functionality available to them as shown in the screenshot above.

The strain of PIVY in this attack used a DLL side-loading technique to stealthily load itself into memory. To accomplish this, it exploited a trusted and signed application. The PIVY payload was dropped along with the trusted and signed Samsung tool (*RunHelp.exe*) in the following manner:

1. A nullsoft installer package (NSIS) was created with a legitimate, signed Samsung tool in it.
2. Once executed, the installer script within the NSIS package extracted the Samsung tool and added a fake DLL with the same name as a legitimate DLL (*ssMUIDLL.dll*), which is required by the application.
3. The DLL contains a PIVY stager, which is then loaded by the Samsung tool.
4. After the fake DLL was loaded by the Samsung tool, it decrypted a blob payload in the same folder, which contains the actual PIVY payload.
5. It was able to achieve persistence by creating a rogue scheduled task.



Post-persistence execution of PIVY, side-loaded into a legitimate Samsung application.

PIVY's use of DLL side-loading to abuse Samsung tools is not new, and has been reported previously by Palo Alto. In 2016 it was used to attack pro-democratic activists in Hong Kong, most probably by Chinese threat actors.

⚠ **Note: Our team has reached out to and advised the targeted organizations on active containment actions.**

Secondary Web Shells

In later stages of the attack, the threat actor deployed two other custom-built web shells. From these web shells, they launched reconnaissance commands, stole data, and dropped additional tools including portqry.exe, renamed cmd.exe, winrar, and the notorious hTran.

aa.exe -n[redacted]-e 3389
at \\[redacted]
wmic /node:[redacted]/user:"[redacted]" /password:"[redacted]" process call create[redacted]bat
[redacted]
nslookup [redacted]
ping [redacted]

Reconnaissance and lateral movement commands launched from the secondary web shell.

Data Exfiltration

The threat actor exfiltrated stolen data using multiple different channels including web shells and hTran.

Compressing the Stolen Data

In an attempt to hide the contents of the stolen data, the threat actor used winrar to compress and password-protect it. The winrar binaries and compressed data were found mostly in the Recycle Bin folder, a TTP that was previously observed in APT10-related attacks, as well as others. This threat actor is known to stage the data in multi-part archives before exfiltration.

The threat actor used the following commands to compress the data.

- rar.exe a -k -r -s -m1 -[password] [REDACTED].rar [REDACTED].temp
- rar.exe a -k -r -s -m1 -[password] [REDACTED].rar [REDACTED].csv
- rar a -r -[password] [REDACTED].rar sam system ntds.dit

"cdm" /c cd /d C:\hp\&copy \\[redacted]\d$\emc\sql\[redacted]A.rar&echo [S]&cd&echo [E]
"cdm" /c cd /d C:\hp\&dir \\[redacted]\d$\emc\sql\&echo [S]&cd&echo [E]
"cdm" /c cd /d C:\hp\&dir \\[redacted]\d$\EMC\sql\[redacted]b.rar&echo [S]&cd&echo [E]
"cdm" /c cd /d C:\hp\&move \\[redacted]\d$\EMC\sql\[redacted]b.rar&echo [S]&cd&echo [E]

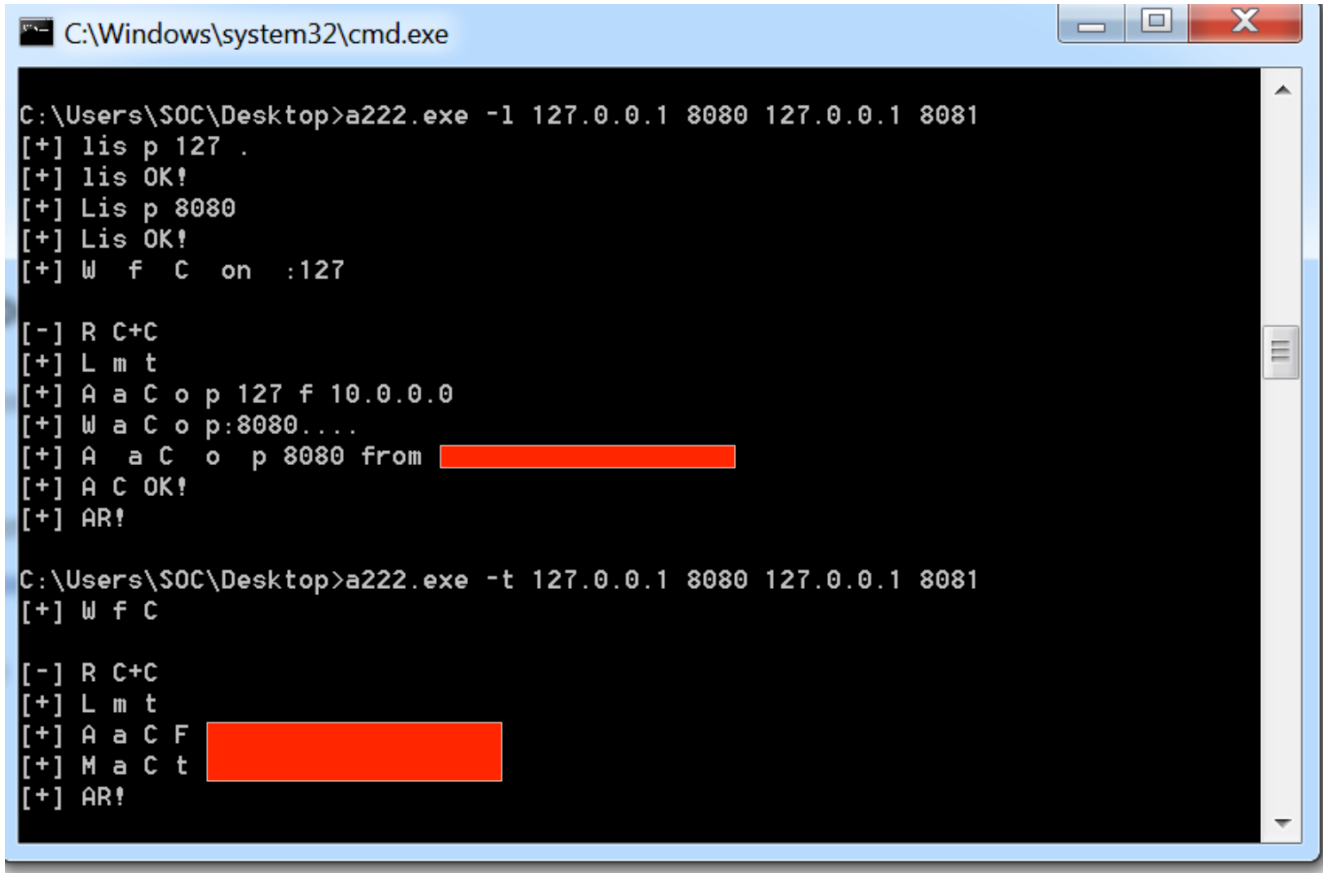
Compressed stolen data exfiltrated via web shell.

The contents of the compressed data was crucial in understanding the threat actor's motivation for the attack, as well as what type of information they were after.

hTran

In order to exfiltrate data from a network segment not connected to the Internet, the threat actor deployed a modified version of hTran. This 'connection bouncer' tool lets the threat actor redirect ports and connections between different networks and obfuscate C2 server traffic. There have been numerous reports of hTran being used by different Chinese threat actors, including: APT3, APT27 and DragonOK.

The threat actor made some modifications to the original source code of hTran. Many strings, including the debug messages, were intentionally changed and obfuscated in an attempt to evade detection and thwart efforts to identify the malware by antivirus and researchers.



Obfuscated debug messages.

Since the original source code for hTran is publicly available, we were able to compare the debug output to the original source code to show that it has indeed been modified.

```
v3 = name;
v4 = s;
v5 = gethostbyname(name);
if ( v5 )
{
  namea = 0i64;
  namea.sa_family = 2;
  *(_WORD *)namea.sa_data = htons(hostshort);
  *(_DWORD *)&namea.sa_data[2] = **(_DWORD **)v5->h_addr_list;
  if ( connect(v4, &namea, 16) >= 0 )
  {
    result = 1;
  }
  else
  {
    looks_like_printf((const wchar_t *)"[-] C e.\r\n");
    result = 0;
  }
}
```

Identifying modifications in a disassembly of the modified hTran.

printf is being called (dubbed by us as “looks_like_printf”) with output “C e.”. By looking at the original source code, we were able to identify that this is supposed to be “Connect error”.

```
    if(connect(sockfd,(struct sockaddr *)&cliaddr,sizeof(struct sockaddr))<0)
    {
            printf("[-] Connect error.\r\n");
            return(0);
    }
    return(1);
}
```

A section of the source code for hTran.

Understanding the Motive

When you think of large breaches to big organizations, the first thing that comes to mind is usually payment data. An organization that provides services to a large customer base has a lot of credit card data, bank account information, and more personal data on its systems. These attacks are usually conducted by a cybercrime group looking to make money.

In contrast, when a nation state threat actor is attacking a big organization, the end goal is typically not financial, but rather intellectual property or sensitive information about their clients.

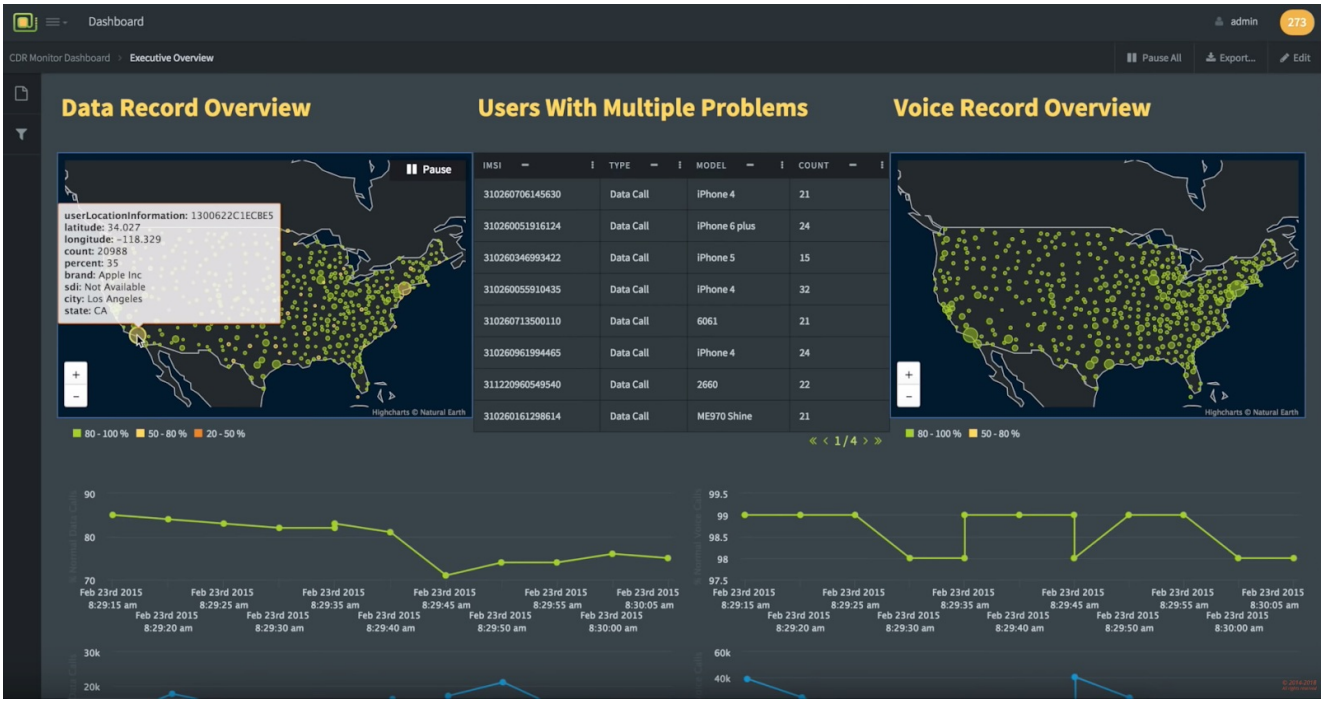
One of the most valuable pieces of data that telecommunications providers hold is Call Detail Records (CDRs). CDRs are a large subset of metadata that contains all details about calls, including:

- Source, Destination, and Duration of a Call
- Device Details
- Physical Location
- Device Vendor and Version

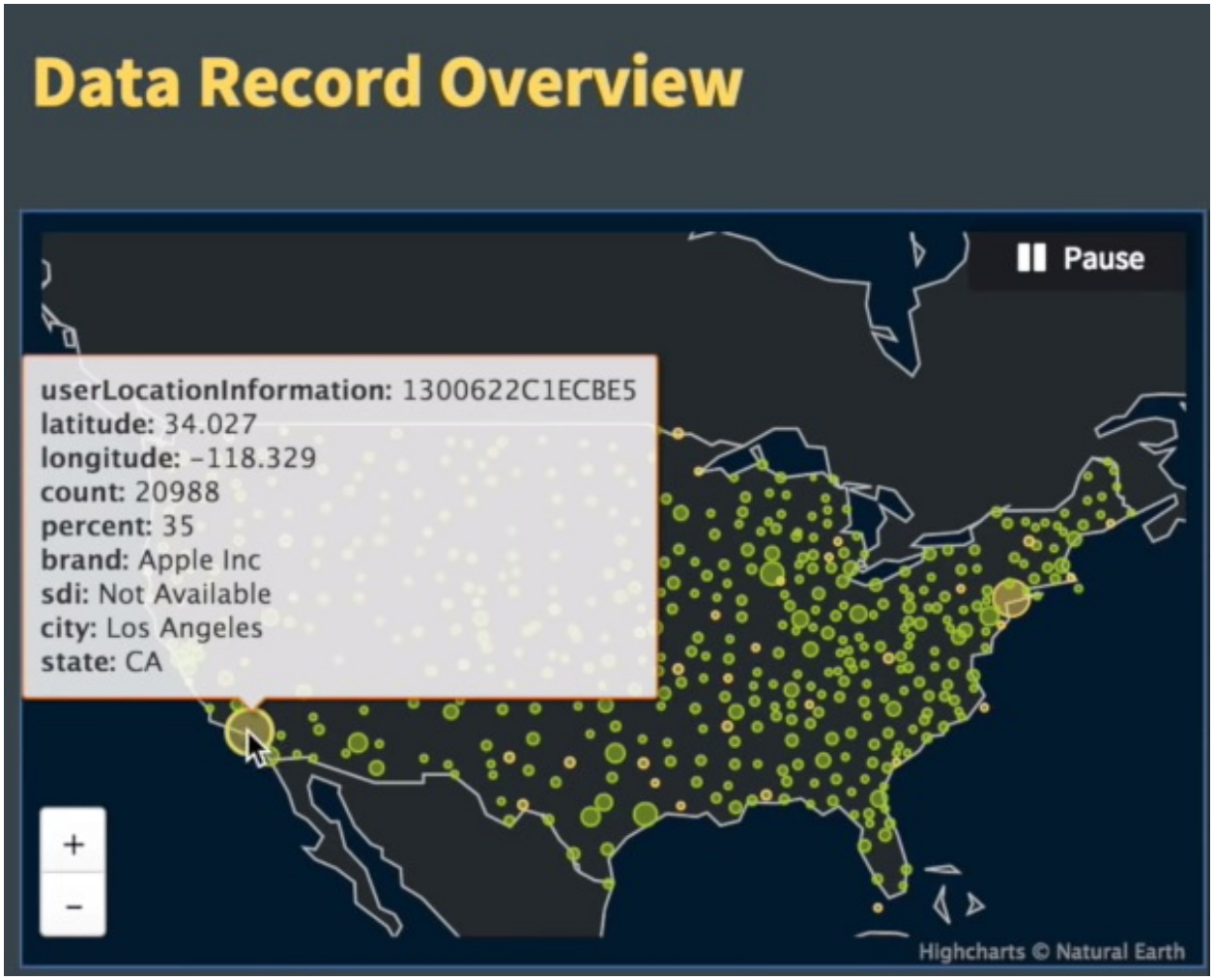
For a nation state threat actor, obtaining access to this data gives them intimate knowledge of any individuals they wish to target on that network. It lets them answer questions like:

- Who are the individuals talking to?
- Which devices are the individuals using?
- Where are the individuals traveling?

Having this information becomes particularly valuable when nation-state threat actors are targeting foreign intelligence agents, politicians, opposition candidates in an election, or even law enforcement.



Example 1: CDR Data



Example 2: CDR Data

IMSI	TYPE	MODEL
310260706145630	Data Call	iPhone 4
310260051916124	Data Call	iPhone 6 plus
310260346993422	Data Call	iPhone 5
310260055910435	Data Call	iPhone 4
310260713500110	Data Call	6061
310260961994465	Data Call	iPhone 4
311220960549540	Data Call	2660
310260161298614	Data Call	ME970 Shine

Example 3: CDR Data

Beyond targeting individual users, this attack is also alarming because of the threat posed by the control of a telecommunications provider. Telecommunications has become critical infrastructure for the majority of world powers. A threat actor with total access to a telecommunications provider, as is the case here, can attack however they want passively and also actively work to sabotage the network.

This attack has widespread implications, not just for individuals, but also for organizations and countries alike. The use of specific tools and the choice to hide ongoing operations for years points to a nation state threat actor, most likely China. This is another form of cyber warfare being used to establish a foothold and gather information undercover until they are ready to strike.

Want to learn about post-incident review?

Read about post-incident review.

Threat Intel Research

The following sections detail the methodology and work process used to piece together the various stages and components of the attack. This work enabled us to not only reconstruct these attacks, but also to find additional artifacts and information regarding the threat actor and its operations.

Methodology

Step 1: Creating and Maintaining an IOC Inventory

The first step in this process was to create a comprehensive list of indicators of compromise (IOCs) observed throughout the different stages of the attack. This list included various indicators, such as file hashes, domains, IP addresses, file names, and registry/service names. In addition to this, our reverse engineers were able to extract further IOCs from the collected samples, which have also been added to the list.

The list of IOCs was periodically updated and fed back into our threat intel engine as more were discovered.

Step 2: Hunting for Known Evil

Equipped with an ever-growing list of known IOCs, our team set out to hunt for “low-hanging fruit” across multiple environments. This step was done by using both internal sources, such as the Cybereason solution, as well as hunting for indicators in the wild.

The hunt for “known evil” yielded interesting results that helped uncover additional compromised assets as well as more parts of the attack infrastructure.

Step 3: Threat Actor’s Arsenal

Perhaps one of the most interesting steps involved identifying and analyzing the tools the threat actor used throughout the attack. The combination of the preference of tools, sequence of use, and specifically how they are used during the attack says a lot about a threat actor, especially when it comes to attribution.

One of the more notable aspects was how the threat actor used mostly known tools that were customized for this specific attack. Each tool was customized differently, and included re-writing the code, stripping debug symbols, string obfuscation, and embedding the victim’s specific information within the tools’ configuration.

However, the threat actor also used tools we were not able to attribute to any known tool. These tools were used in the later stages of the attack, once the operation was already discovered. This was most likely to decrease the risk of exposure or attribution.

Finally, the payloads were almost never repeated. The threat actor made sure that each payload had a unique hash, and some payloads were packed using different types of packers, both known and custom.

The main tools these attacks had in common are:

1. **Web Shells**
    - A modified version of the China Chopper web shell was used for initial compromise.
    - Custom-built web shells were used for later phases of the attack.
2. **Reconnaissance Tools**
    - A modified version of Nbtscan was used to identify available NetBIOS name servers locally or over the network.
    - Multiple Windows built-in tools were used for various tasks, including whoami, net.exe, ipconfig, netstat, portqry, and more.
    - WMI and PowerShell commands were used for various tasks.
3. **RAT**
    - PoisonIvy was used to maintain access across the compromised assets.
    - PlugX was used in some of the instances that we're aware of.
4. **Credential Dumpers**
    - A modified version of Mimikatz was used to dump credentials stored on the compromised machines.
    - A PowerShell-based Mimikatz was also used to dump credentials stored on the compromised machines.
5. **Lateral movement**
    - WMI was used for lateral movement.
    - PsExec was also used for lateral movement.
6. **Connection Proxy**

    A modified version of hTran was used to exfiltrate stolen data.
7. **Compression tool**

    Winrar was used to compress and password-protect stolen data.

Step 4: Creating a TTP-based Behavioral Profile

One of the key components of threat hunting is to create a TTP-based behavioral profile of the threat actor in question. Malware payloads and operational infrastructure can be quickly changed or replaced over time, and as such, the task of tracking a threat actor can become quite difficult.

For that reason, it is crucial to profile the threat actor and study its behavior, the tools it uses, and its techniques. These behavioral-based TTPs are less likely to change drastically, and are\ key factors of any threat hunt or attribution efforts.

The Cybereason solution is compatible with the MITRE ATT&CK framework, which made it easy to keep track of the observed TTPs and correlate the data with known threat actors.

The following chart reflects the behavioral profile of the threat actor based on the most frequently observed techniques used throughout these attacks.

MITRE ATT&CK Techniques Breakdown

Initial Access	Execution	Persistence	Privilege Escalation	Defense Evasion	Credential Access
Exploit Public-Facing Application	Command-line interface	Web Shell	Valid Accounts	DLL-side Loading	Credential Dumping
	Windows Management Instrumentation	Create Account	Web Shell	Indicator Removal from Tools	
	PowerShell			Obfuscated Files or Information	
				Masquerading	

Discovery	Lateral Movement	Collection	Command and Control	Exfiltration	Impact
System Network Configuration Discovery	Data From Local System	Remote File Copy	Data Compressed		
Remote System Discovery	Pass the Hash	Data Staged	Connection Proxy	Exfiltration Over Command and Control Channel	
Account Discovery	Remote File Copy	Input Capture			
Permission Groups Discovery					

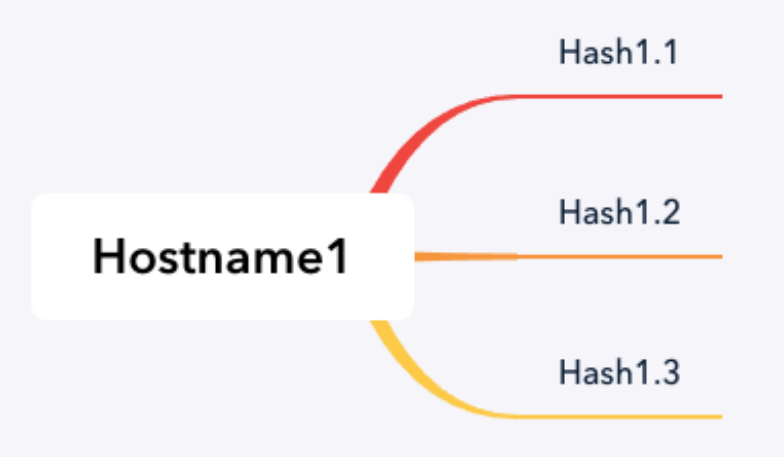
Step 5: Mapping out the Infrastructure and Operational Activity

Reconstructing the Infrastructure

In order to make sense of all the data, we fed it into multiple threat intelligence sources, including our own and third parties.

⚠ **Note: Since we cannot share any IOCs, we will refer to file hashes, hostnames, IP addresses and other IOCs as generic placeholders.**

Hostname1 is the hostname that was used for the C2 server targeting the telecommunications providers.



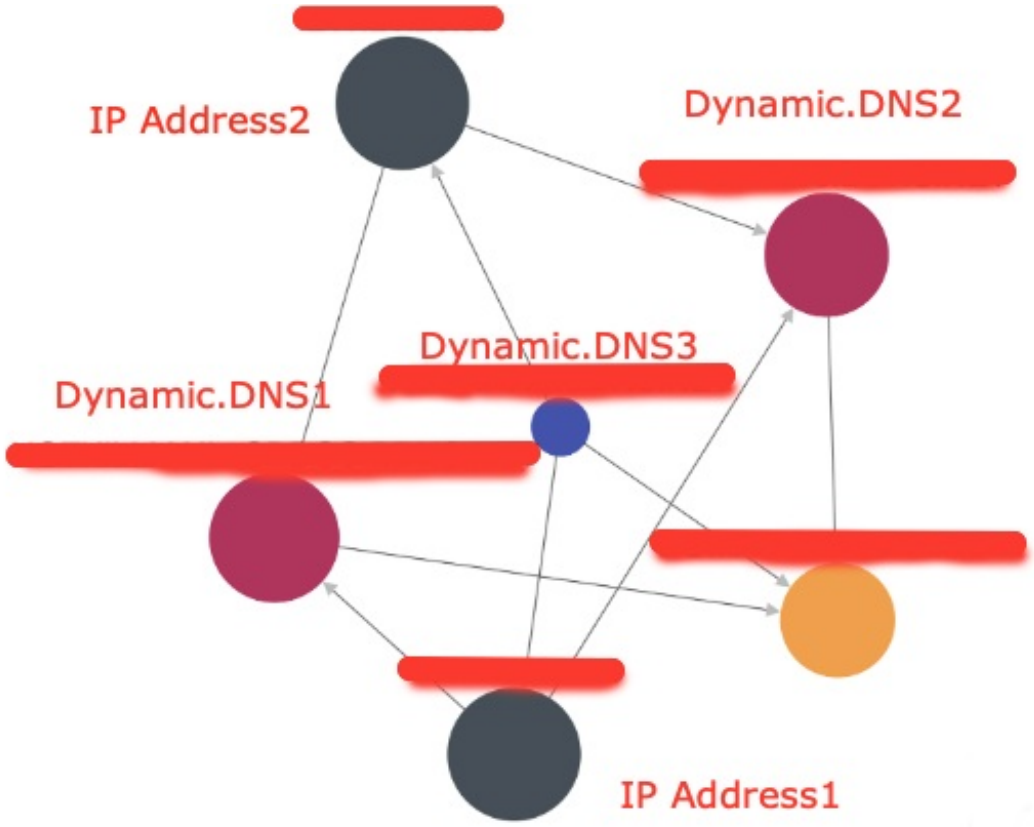
Hostname1 connected to multiple tools.

In analyzing the files, it is clear they are all contacting the same host *hostname1*. *hostname1* was the C2 server that the malware and web shells connected to.

Once we determined the hashes in the scope of the attack were only connecting to *hostname1*, which is a dynamic DNS hostname, we looked to see if we could find more information about the C2 server.

A simple WHOIS query revealed that the IP address was registered to a colocation **hosting company in Asia**, though there was no other publicly available information about this IP address.

By querying all of our threat intel resources about this IP address, we discovered that it was associated with multiple dynamic DNS hostnames.

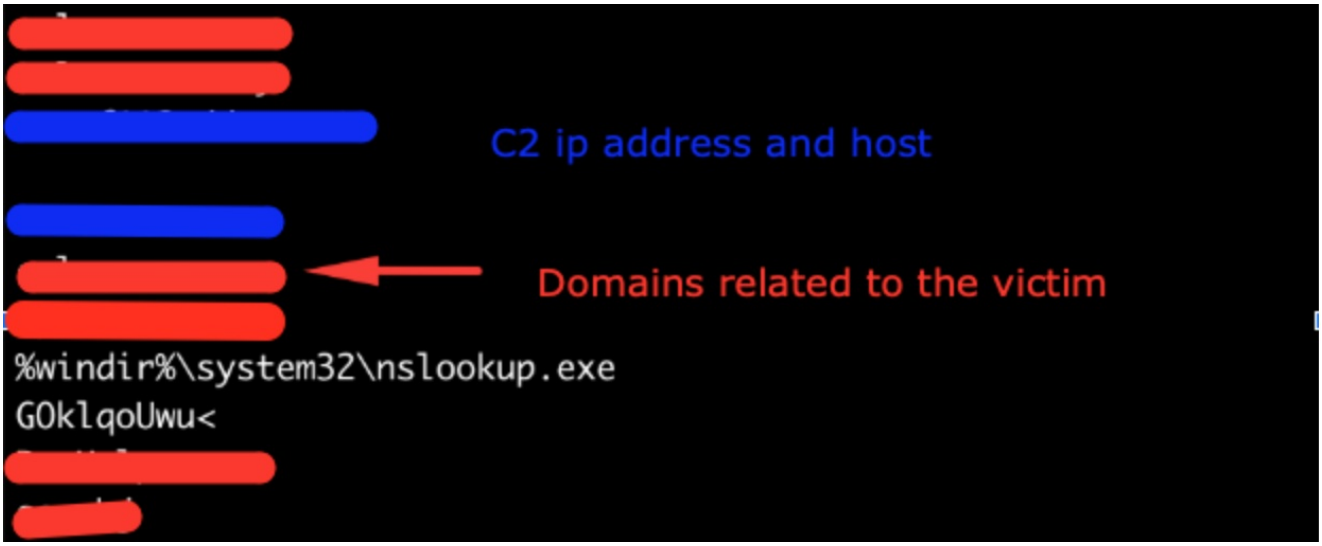


Multiple dynamic DNS hostnames.

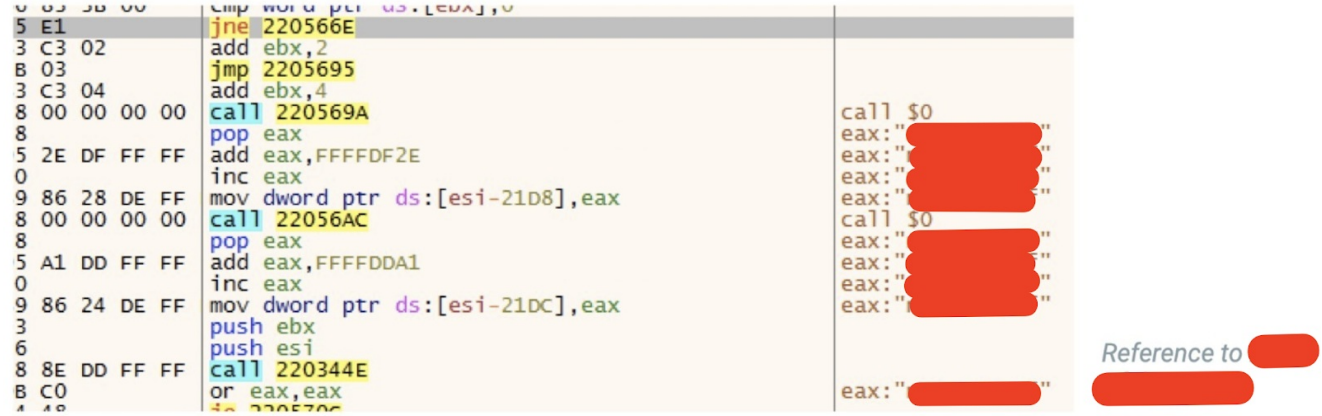
We were unable to find indications of connections to *Dynamic.DNS2* and *Dynamic.DNS3.* However, they were registered and associated with *IP.Address1*.

For the other dynamic DNS hosts, we leveraged various threat intel repositories and crafted queries that searched for executables with these IP addresses and hostnames in their string table. One of the queries returned a few DLLs with identical names to the DLL we had initially investigated. However, the hashes were different. After obtaining the found DLLs, we patched them back into the NSIS installer and detonated the samples in our testing environment. Dynamic analysis of the newly obtained DLLs revealed a new set of domains and IP addresses that were completely different. These domains were actually related to different telecommunications providers.

⚠ **Note: Cybereason immediately reached out to those telecommunications providers and provided them all of the necessary information to handle the incident internally.**

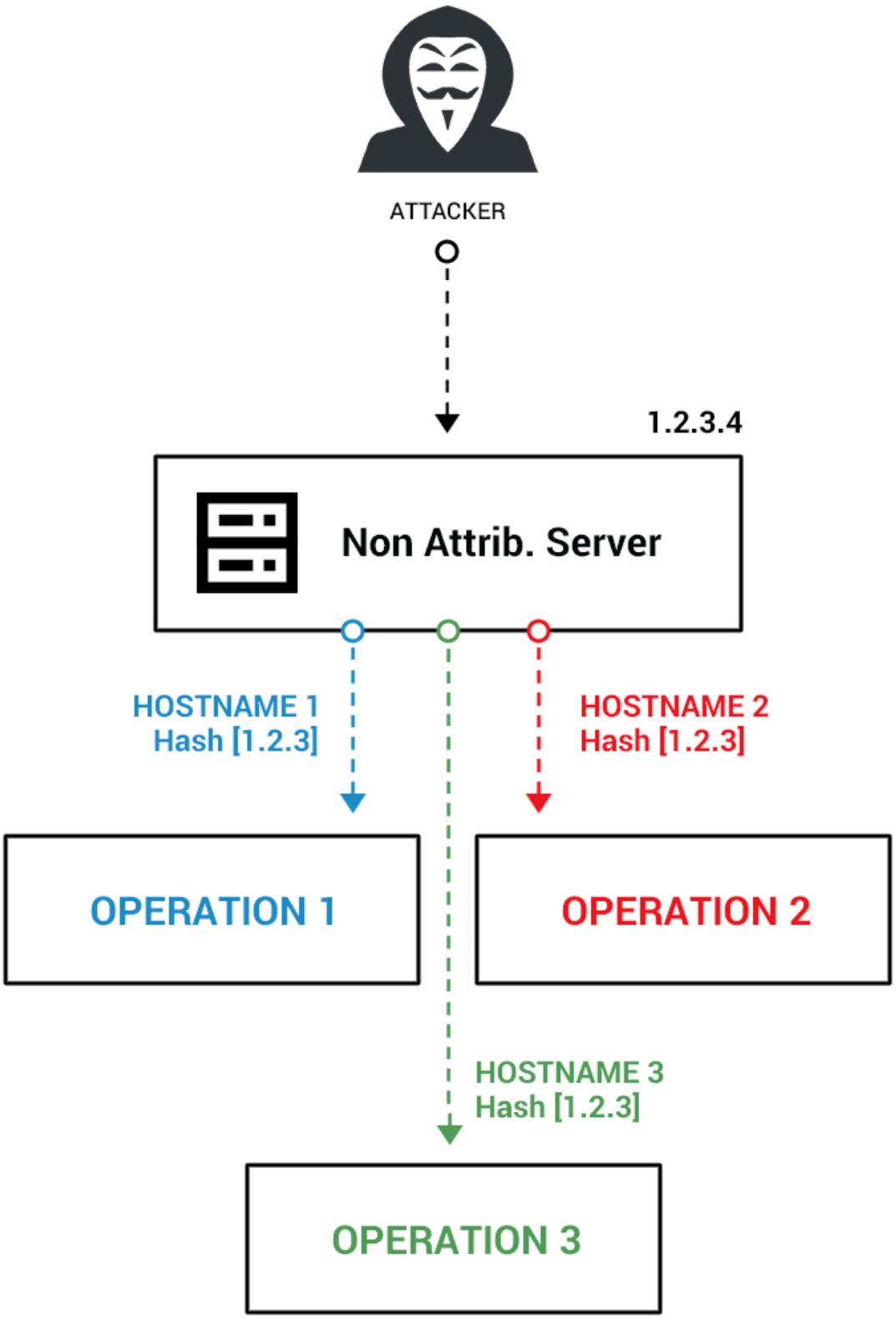


Strings from the dumped memory section of the injected shellcode. We can see many details about the attack including domains and C2 server IP addresses.



Shellcode being unpacked and injected into a remote process. The redacted segments contain the name of the customer, C2 IP addresses, and domains.

Infrastructure Operational Security


ATTACKER
1.2.3.4
Non Attrib. Server
HOSTNAME 1
Hash [1.2.3]
HOSTNAME 2
Hash [1.2.3]
OPERATION 1
OPERATION 2
HOSTNAME 3
Hash [1.2.3]
OPERATION 3

The threat actor's infrastructure.

The threat actor had a specific pattern of behavior that allowed us to understand their modus operandi: they used one server with the same IP address for multiple operations. This server is a key component in their 'non-attributable' infrastructure.

The threat actor separated operations by using different hostnames per operation, though they are hosted on the same server and IP address. The domains and server registration information pointed to three main countries: China, Hong Kong, and Taiwan.

This is cheap and efficient for the threat actor, but is almost transparent for a seasoned researcher with access to the right threat intelligence tools. There are previous reports of threat actors including APT10 and APT1 using dynamic DNS.

Monitoring this infrastructure gave us information about if and when the threat actor was starting new waves of the attack or additional attacks on other providers.

When researching C2 servers, it is important to watch for:

- Association with domains, especially if they are dynamic DNS domains.
- File hashes that are associated with the IP address or the domain of the C2 server.
    Static information and metadata from associated samples that could be used to broaden the search after additional information is gathered.

This demonstrates the importance of proper operational security and a separation between tools and operations for threat actors.

Step 6: Rounding Up Immediate/Potential Suspects

Attribution is a fickle and delicate art. In most cases, it is very difficult to achieve 100% certainty when attributing an attack to a specific threat actor. It can be tempting to attribute an attack to a certain threat actor whenever certain tools-of-the-trade, IP addresses, strings, or "indicative" techniques are observed.

However, it is important to bear in mind that the aforementioned data points are often prone to manipulation and reuse across different threat actors. Further, they are not impervious to psychological warfare, as in, trying to "pin" an operation on a different threat actor to avoid proper attribution.

In order to increase the certainty level when attributing to a specific threat actor, we took the following aspects of the attacks into consideration:

- Indicators of Compromise
- TTPs (Tactics, Techniques and Procedures)

- Threat actor's tools
- Motive behind the attacks
- Regional and industry considerations

Carefully examining each of the different aspects plays an important role in avoiding misattribution. This model offers a more balanced interpretation of the data that is based on a myriad of components. By performing a contextualized review of the data, you are able to yield a more wholesome result with greater certainty.

When it comes to attributing Operation Soft Cell, we are unable to achieve 100% certainty with regard to the identity of the threat actor. However, based on our interpretation of the data, we conclude with a **high level of certainty** that:

- The threat actor behind Operation Soft Cell is likely state-sponsored.
- The threat actor is affiliated with China.

After following the above attribution model and carefully reviewing the data, we are able to narrow down the suspect list to three known APT groups, all of which are known to be linked to China- APT10, APT27, and DragonOK.

Having found multiple similarities to previous attacks, it is our estimation that the threat actor behind these attacks is likely linked to APT10, or at the very least, to a threat actor that shares tools, techniques, motive and infrastructural preferences with those of APT10.

While we cannot completely rule out a “copy-cat” scenario, where another threat actor might masquerade as APT10 to thwart attribution efforts, we find this option to be less likely in light of our analysis of the data.

Conclusion

In this blog, we have described an ongoing global attack against telecommunications providers that has been active since at least 2017. The threat actor managed to infiltrate into the deepest segments of the providers’ network, including some isolated from the internet, as well as compromise critical assets. Our investigation showed that these attacks were targeted, and that the threat actor sought to steal communications data of specific individuals in various countries.

Throughout this investigation, we have uncovered the infrastructure that facilitated the malicious operations taken by this threat actor. The data exfiltrated by this threat actor, in conjunction with the TTPs and tools used, allowed us to determine with a **very high probability** that the threat actor behind these malicious operations is backed by a nation

state, and is affiliated with China. Our contextualized interpretation of the data suggests that the threat actor is likely APT10, or at the very least, a threat actor that shares, or wishes to emulate its methods by using the same tools, techniques, and motives.

It's important to keep in mind that even though the attacks targeted specific individuals, any entity that possesses the power to take over the networks of telecommunications providers can potentially leverage its unlawful access and control of the network to shut down or disrupt an entire cellular network as part of a larger cyber warfare operation.

Due to multiple and various limitations, we cannot disclose all the information we have gathered on the attack in this report. Our team will continue to monitor and track the threat actor's activity in order to identify more tools and compromised organizations.

Ask the researchers questions about this attack during their live webinar.

Closing Notes: This research, which is still ongoing, has been a huge effort for the entire **Cybereason Nocturnus** team. Special thanks goes to **Niv Yona, Noa Pinkas, Josh Trombley, Jakes Jansen, and every single member of the Nocturnus team** for the countless hours and effort that were put into this research. We will continue to monitor and update our blog with more information once available and as our investigation progresses.