Feb 09 2024

NETMANAGEIT

Intelligence Report

Kimsuky disguised as a Korean company signed with a valid certificate to distribute Troll Stealer

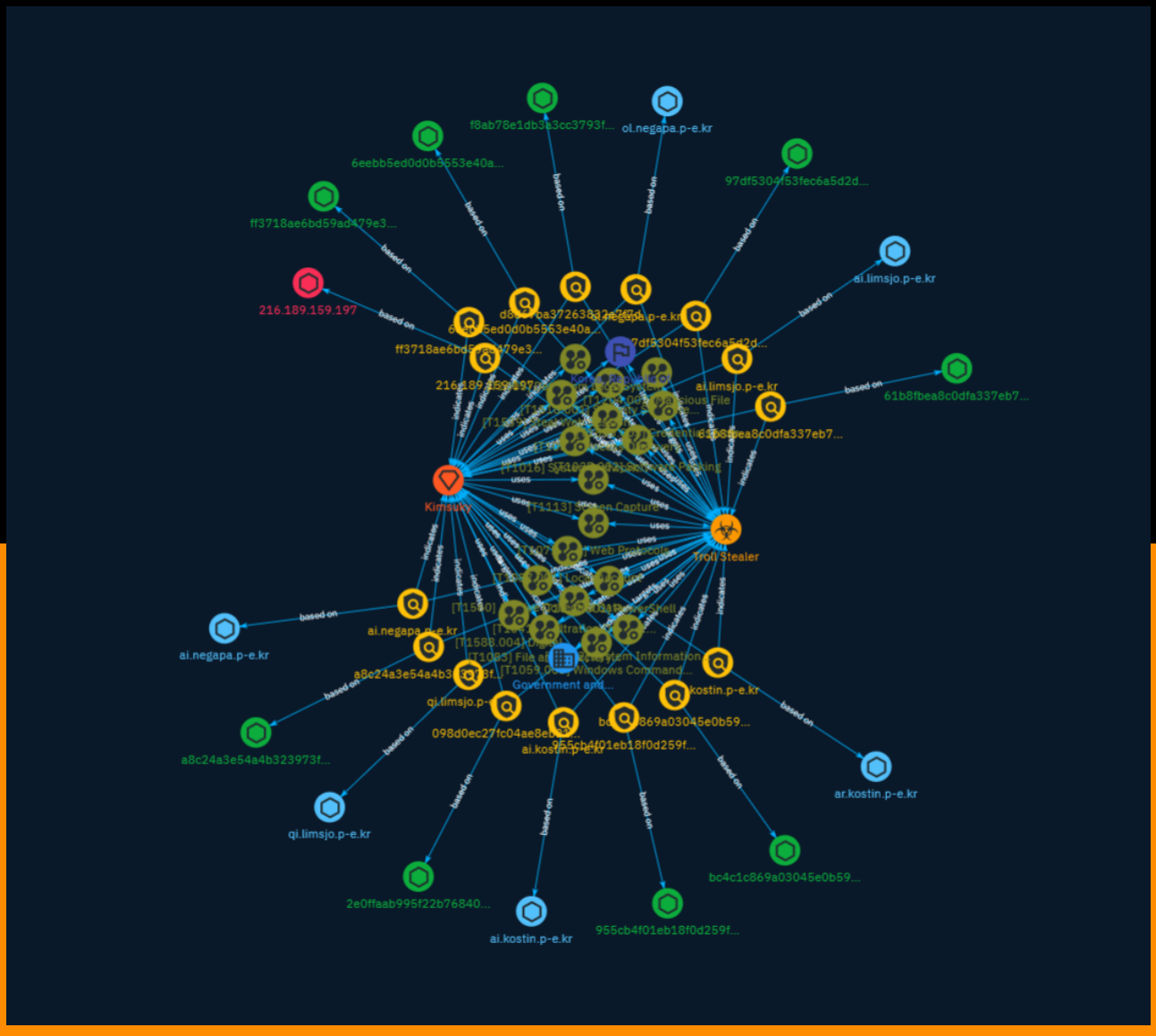

Table of contents

Overview

- Description 4
- Confidence 4
- Content 5

Entities

- Indicator 6
- Malware 19
- Intrusion-Set 20
- Attack-Pattern 22
- Country 45
- Sector 46

Observables

- Hostname 47

- IPv4-Addr 48
- StixFile 49

External References

- External References 51

Overview

Description

A new malware called Troll Stealer has been discovered, which is believed to originate from the North Korean APT group Kimsuky. Troll Stealer is an information-stealing malware written in Go language that exfiltrates data including SSH credentials, FileZilla information, browser data, system info, and screen captures. It is distributed via droppers disguised as Korean security software installers, signed with a stolen certificate from D2innovation Co. LTD. Troll Stealer appears related to previous Kimsuky malware AppleSeed and AlphaSeed based on code similarities, and specifically targets the GPKI certificate folder on systems, suggesting it is aimed at government and administrative organizations in South Korea.

Confidence

This value represents the confidence in the correctness of the data contained within this report.

15 / 100

Content

N/A

Indicator

Name
qi.limsjo.p-e.kr
Pattern Type
stix
Pattern
[hostname:value = 'qi.limsjo.p-e.kr']
Name
ol.negapa.p-e.kr
Pattern Type
stix
Pattern
[hostname:value = 'ol.negapa.p-e.kr']
Name
ar.kostin.p-e.kr

Pattern Type
stix
Pattern
[hostname:value = 'ar.kostin.p-e.kr']
Name
ai.negapa.p-e.kr
Pattern Type
stix
Pattern
[hostname:value = 'ai.negapa.p-e.kr']
Name
ai.limsjo.p-e.kr
Pattern Type
stix
Pattern
[hostname:value = 'ai.limsjo.p-e.kr']
Name
ai.kostin.p-e.kr

Pattern Type

stix

Pattern

[hostname:value = 'ai.kostin.p-e.kr']

Name

216.189.159.197

Pattern Type

stix

Pattern

[ipv4-addr:value = '216.189.159.197']

Name

ff3718ae6bd59ad479e375c602a81811718dfb2669c2d1de497f02baf7b4adca

Pattern Type

stix

Pattern

[file:hashes.'SHA-256' = 'ff3718ae6bd59ad479e375c602a81811718dfb2669c2d1de497f02baf7b4adca']

Name

bc4c1c869a03045e0b594a258ec3801369b0dcabac193e90f0a684900e9a582d

Pattern Type

stix

Pattern

[file:hashes.'SHA-256' = 'bc4c1c869a03045e0b594a258ec3801369b0dcabac193e90f0a684900e9a582d']

Name

a8c24a3e54a4b323973f61630c92ecaad067598ef2547350c9d108bc175774b9

Pattern Type

stix

Pattern

[file:hashes.'SHA-256' = 'a8c24a3e54a4b323973f61630c92ecaad067598ef2547350c9d108bc175774b9']

Name

97df5304f53fec6a5d2d2bd75b9310a3747b681520fe45d2961bc4df86e556d7

Pattern Type

stix

Pattern

[file:hashes.'SHA-256' = '97df5304f53fec6a5d2d2bd75b9310a3747b681520fe45d2961bc4df86e556d7']

Name

955cb4f01eb18f0d259fcb962e36a339e8fe082963dfd9f72d3851210f7d2d3b

Pattern Type

stix

Pattern

[file:hashes.'SHA-256' = '955cb4f01eb18f0d259fcb962e36a339e8fe082963dfd9f72d3851210f7d2d3b']

Name

6eebb5ed0d0b5553e40a7b1ad739589709d077aab4cbea1c64713c48ce9c96f9

Pattern Type

stix

Pattern

[file:hashes.'SHA-256' = '6eebb5ed0d0b5553e40a7b1ad739589709d077aab4cbea1c64713c48ce9c96f9']

Name

61b8fbea8c0dfa337eb7ff978124ddf496d0c5f29bcb5672f3bd3d6bf832ac92

Pattern Type

stix

Pattern

[file:hashes.'SHA-256' = '61b8fbea8c0dfa337eb7ff978124ddf496d0c5f29bcb5672f3bd3d6bf832ac92']

Name

d8d67ba37263832e7f7d0a945a04afe3d9cea24e78a2d82b00463a2ab575ddb0b53f020c9967391c8469a831c3205f68d010d752a17419d7c2bb34ae8dc55384

Description

Created by VirusTotal connector as the positive count was >= 10

Pattern Type

stix

Pattern

[file:hashes.'SHA-256' = 'f8ab78e1db3a3cc3793f7680a90dc1d8ce087226ef59950b7acd6bb1beffd6e3']

Name

098d0ec27fc04ae8eb212efa85bdce3ad36f09c5ed934ac14a2cf120237ed1ca19ece03ca327bb4ce900672ebf0b4f3782e31df704d4b9b1141b059ee64182b9

Description

Created by VirusTotal connector as the positive count was >= 10

Pattern Type

stix

Pattern

[file:hashes.'SHA-256' = '2e0ffaab995f22b7684052e53b8c64b9283b5e81503b88664785fe6d6569a55e']

Name

qi.limsjo.p-e.kr

Pattern Type

stix

Pattern

[hostname:value = 'qi.limsjo.p-e.kr']

Name

ol.negapa.p-e.kr

Pattern Type

stix

Pattern

[hostname:value = 'ol.negapa.p-e.kr']

Name

ar.kostin.p-e.kr

Pattern Type
stix
Pattern
[hostname:value = 'ar.kostin.p-e.kr']
Name
ai.negapa.p-e.kr
Pattern Type
stix
Pattern
[hostname:value = 'ai.negapa.p-e.kr']
Name
ai.limsjo.p-e.kr
Pattern Type
stix
Pattern
[hostname:value = 'ai.limsjo.p-e.kr']
Name
ai.kostin.p-e.kr

Pattern Type
stix
Pattern
[hostname:value = 'ai.kostin.p-e.kr']
Name
216.189.159.197
Pattern Type
stix
Pattern
[ipv4-addr:value = '216.189.159.197']
Name
ff3718ae6bd59ad479e375c602a81811718dfb2669c2d1de497f02baf7b4adca
Pattern Type
stix
Pattern
[file:hashes.'SHA-256' = 'ff3718ae6bd59ad479e375c602a81811718dfb2669c2d1de497f02baf7b4adca']
Name

bc4c1c869a03045e0b594a258ec3801369b0dcabac193e90f0a684900e9a582d

Pattern Type

stix

Pattern

[file:hashes.'SHA-256' = 'bc4c1c869a03045e0b594a258ec3801369b0dcabac193e90f0a684900e9a582d']

Name

a8c24a3e54a4b323973f61630c92ecaad067598ef2547350c9d108bc175774b9

Pattern Type

stix

Pattern

[file:hashes.'SHA-256' = 'a8c24a3e54a4b323973f61630c92ecaad067598ef2547350c9d108bc175774b9']

Name

97df5304f53fec6a5d2d2bd75b9310a3747b681520fe45d2961bc4df86e556d7

Pattern Type

stix

Pattern

[file:hashes.'SHA-256' = '97df5304f53fec6a5d2d2bd75b9310a3747b681520fe45d2961bc4df86e556d7']

Name

955cb4f01eb18f0d259fcb962e36a339e8fe082963dfd9f72d3851210f7d2d3b

Pattern Type

stix

Pattern

[file:hashes.'SHA-256' = '955cb4f01eb18f0d259fcb962e36a339e8fe082963dfd9f72d3851210f7d2d3b']

Name

6eebb5ed0d0b5553e40a7b1ad739589709d077aab4cbea1c64713c48ce9c96f9

Pattern Type

stix

Pattern

[file:hashes.'SHA-256' = '6eebb5ed0d0b5553e40a7b1ad739589709d077aab4cbea1c64713c48ce9c96f9']

Name

61b8fbea8c0dfa337eb7ff978124ddf496d0c5f29bcb5672f3bd3d6bf832ac92

Pattern Type

stix

Pattern

[file:hashes.'SHA-256' = '61b8fbea8c0dfa337eb7ff978124ddf496d0c5f29bcb5672f3bd3d6bf832ac92']

Name

d8d67ba37263832e7f7d0a945a04afe3d9cea24e78a2d82b00463a2ab575ddb0b53f020c9967391c8469a831c3205f68d010d752a17419d7c2bb34ae8dc55384

Description

Created by VirusTotal connector as the positive count was >= 10

Pattern Type

stix

Pattern

[file:hashes.'SHA-256' = 'f8ab78e1db3a3cc3793f7680a90dc1d8ce087226ef59950b7acd6bb1beffd6e3']

Name

098d0ec27fc04ae8eb212efa85bdce3ad36f09c5ed934ac14a2cf120237ed1ca19ece03ca327bb4ce900672ebf0b4f3782e31df704d4b9b1141b059ee64182b9

Description

Created by VirusTotal connector as the positive count was >= 10

Pattern Type

stix

Pattern

[file:hashes.'SHA-256' = '2e0ffaab995f22b7684052e53b8c64b9283b5e81503b88664785fe6d6569a55e']

Malware

Name
Troll Stealer

Name
Troll Stealer

Intrusion-Set

Name

Kimsuky

Description

[Kimsuky](https://attack.mitre.org/groups/G0094) is a North Korea-based cyber espionage group that has been active since at least 2012. The group initially focused on targeting South Korean government entities, think tanks, and individuals identified as experts in various fields, and expanded its operations to include the United States, Russia, Europe, and the UN. [Kimsuky](https://attack.mitre.org/groups/G0094) has focused its intelligence collection activities on foreign policy and national security issues related to the Korean peninsula, nuclear policy, and sanctions.(Citation: EST Kimsuky April 2019)(Citation: BRI Kimsuky April 2019)(Citation: Cybereason Kimsuky November 2020)(Citation: Malwarebytes Kimsuky June 2021)(Citation: CISA AA20-301A Kimsuky) [Kimsuky](https://attack.mitre.org/groups/G0094) was assessed to be responsible for the 2014 Korea Hydro & Nuclear Power Co. compromise; other notable campaigns include Operation STOLEN PENCIL (2018), Operation Kabar Cobra (2019), and Operation Smoke Screen (2019).(Citation: Netscout Stolen Pencil Dec 2018)(Citation: EST Kimsuky SmokeScreen April 2019)(Citation: AhnLab Kimsuky Kabar Cobra Feb 2019) North Korean group definitions are known to have significant overlap, and some security researchers report all North Korean state-sponsored cyber activity under the name [Lazarus Group](https://attack.mitre.org/groups/G0032) instead of tracking clusters or subgroups.

Name

Kimsuky

Description

[Kimsuky](https://attack.mitre.org/groups/G0094) is a North Korea-based cyber espionage group that has been active since at least 2012. The group initially focused on targeting South Korean government entities, think tanks, and individuals identified as experts in various fields, and expanded its operations to include the United States, Russia, Europe, and the UN. [Kimsuky](https://attack.mitre.org/groups/G0094) has focused its intelligence collection activities on foreign policy and national security issues related to the Korean peninsula, nuclear policy, and sanctions.(Citation: EST Kimsuky April 2019)(Citation: BRI Kimsuky April 2019)(Citation: Cybereason Kimsuky November 2020)(Citation: Malwarebytes Kimsuky June 2021)(Citation: CISA AA20-301A Kimsuky) [Kimsuky](https://attack.mitre.org/groups/G0094) was assessed to be responsible for the 2014 Korea Hydro & Nuclear Power Co. compromise; other notable campaigns include Operation STOLEN PENCIL (2018), Operation Kabar Cobra (2019), and Operation Smoke Screen (2019).(Citation: Netscout Stolen Pencil Dec 2018)(Citation: EST Kimsuky SmokeScreen April 2019)(Citation: AhnLab Kimsuky Kabar Cobra Feb 2019) North Korean group definitions are known to have significant overlap, and some security researchers report all North Korean state-sponsored cyber activity under the name [Lazarus Group](https://attack.mitre.org/groups/G0032) instead of tracking clusters or subgroups.

Attack-Pattern

Name
Web Protocols
ID
T1071.001
Description
Adversaries may communicate using application layer protocols associated with web traffic to avoid detection/network filtering by blending in with existing traffic. Commands to the remote system, and often the results of those commands, will be embedded within the protocol traffic between the client and server. Protocols such as HTTP/S(Citation: CrowdStrike Putter Panda) and WebSocket(Citation: Brazking-Websockets) that carry web traffic may be very common in environments. HTTP/S packets have many fields and headers in which data can be concealed. An adversary may abuse these protocols to communicate with systems under their control within a victim network while also mimicking normal, expected traffic.
Name
Software Packing
ID
T1027.002

Description

Adversaries may perform software packing or virtual machine software protection to conceal their code. Software packing is a method of compressing or encrypting an executable. Packing an executable changes the file signature in an attempt to avoid signature-based detection. Most decompression techniques decompress the executable code in memory. Virtual machine software protection translates an executable's original code into a special format that only a special virtual machine can run. A virtual machine is then called to run this code.(Citation: ESET FinFisher Jan 2018) Utilities used to perform software packing are called packers. Example packers are MPRESS and UPX. A more comprehensive list of known packers is available, but adversaries may create their own packing techniques that do not leave the same artifacts as well-known packers to evade defenses.(Citation: Awesome Executable Packing)

Name

Windows Command Shell

ID

T1059.003

Description

Adversaries may abuse the Windows command shell for execution. The Windows command shell ([cmd](https://attack.mitre.org/software/S0106)) is the primary command prompt on Windows systems. The Windows command prompt can be used to control almost any aspect of a system, with various permission levels required for different subsets of commands. The command prompt can be invoked remotely via [Remote Services](https://attack.mitre.org/techniques/T1021) such as [SSH](https://attack.mitre.org/techniques/T1021/004).(Citation: SSH in Windows) Batch files (ex: .bat or .cmd) also provide the shell with a list of sequential commands to run, as well as normal scripting operations such as conditionals and loops. Common uses of batch files include long or repetitive tasks, or the need to run the same set of commands on multiple systems. Adversaries may leverage [cmd](https://attack.mitre.org/software/S0106) to execute various commands and payloads. Common uses include [cmd](https://attack.mitre.org/software/S0106) to execute a single command, or abusing [cmd](https://attack.mitre.org/software/S0106) interactively with input and output forwarded over a command and control channel.

Name

Security Software Discovery

ID

T1518.001

Description

Adversaries may attempt to get a listing of security software, configurations, defensive tools, and sensors that are installed on a system or in a cloud environment. This may include things such as firewall rules and anti-virus. Adversaries may use the information from [Security Software Discovery](https://attack.mitre.org/techniques/T1518/001) during automated discovery to shape follow-on behaviors, including whether or not the adversary fully infects the target and/or attempts specific actions. Example commands that can be used to obtain security software information are [netsh](https://attack.mitre.org/software/S0108), `reg query` with [Reg](https://attack.mitre.org/software/S0075), `dir` with [cmd](https://attack.mitre.org/software/S0106), and [Tasklist](https://attack.mitre.org/software/S0057), but other indicators of discovery behavior may be more specific to the type of software or security system the adversary is looking for. It is becoming more common to see macOS malware perform checks for LittleSnitch and KnockKnock software. Adversaries may also utilize cloud APIs to discover the configurations of firewall rules within an environment.(Citation: Expel IO Evil in AWS) For example, the permitted IP ranges, ports or user accounts for the inbound/outbound rules of security groups, virtual firewalls established within AWS for EC2 and/or VPC instances, can be revealed by the `DescribeSecurityGroups` action with various request parameters. (Citation: DescribeSecurityGroups - Amazon Elastic Compute Cloud)

Name

PowerShell

ID

T1059.001

Description

Adversaries may abuse PowerShell commands and scripts for execution. PowerShell is a powerful interactive command-line interface and scripting environment included in the Windows operating system.(Citation: TechNet PowerShell) Adversaries can use PowerShell to perform a number of actions, including discovery of information and execution of code. Examples include the `Start-Process` cmdlet which can be used to run an executable and the `Invoke-Command` cmdlet which runs a command locally or on a remote computer (though administrator permissions are required to use PowerShell to connect to remote systems). PowerShell may also be used to download and run executables from the Internet, which can be executed from disk or in memory without touching disk. A number of PowerShell-based offensive testing tools are available, including [Empire](https://attack.mitre.org/software/S0363), [PowerSploit](https://attack.mitre.org/software/S0194), [PoshC2](https://attack.mitre.org/software/S0378), and PSAttack.(Citation: Github PSAttack) PowerShell commands/scripts can also be executed without directly invoking the `powershell.exe` binary through interfaces to PowerShell's underlying `System.Management.Automation` assembly DLL exposed through the .NET framework and Windows Common Language Interface (CLI).(Citation: Sixdub PowerPick Jan 2016)(Citation: SilentBreak Offensive PS Dec 2015)(Citation: Microsoft PSfromCsharp APR 2014)

Name

Exfiltration Over C2 Channel

ID

T1041

Description

Adversaries may steal data by exfiltrating it over an existing command and control channel. Stolen data is encoded into the normal communications channel using the same protocol as command and control communications.

Name

Process Discovery

ID

T1057

Description

Adversaries may attempt to get information about running processes on a system. Information obtained could be used to gain an understanding of common software/applications running on systems within the network. Adversaries may use the information from [Process Discovery](https://attack.mitre.org/techniques/T1057) during automated discovery to shape follow-on behaviors, including whether or not the adversary fully infects the target and/or attempts specific actions. In Windows environments, adversaries could obtain details on running processes using the [Tasklist](https://attack.mitre.org/software/S0057) utility via [cmd](https://attack.mitre.org/software/S0106) or `Get-Process` via [PowerShell](https://attack.mitre.org/techniques/T1059/001). Information about processes can also be extracted from the output of [Native API](https://attack.mitre.org/techniques/T1106) calls such as `CreateToolhelp32Snapshot`. In Mac and Linux, this is accomplished with the `ps` command. Adversaries may also opt to enumerate processes via /proc. On network devices, [Network Device CLI](https://attack.mitre.org/techniques/T1059/008) commands such as `show processes` can be used to display current running processes.(Citation: US-CERT-TA18-106A)(Citation: show_processes_cisco_cmd)

Name

File and Directory Discovery

ID

T1083

Description

Adversaries may enumerate files and directories or may search in specific locations of a host or network share for certain information within a file system. Adversaries may use the information from [File and Directory Discovery](https://attack.mitre.org/techniques/T1083) during automated discovery to shape follow-on behaviors, including whether or not the adversary fully infects the target and/or attempts specific actions. Many command shell utilities can be used to obtain this information. Examples include `dir`, `tree`, `ls`, `find`, and `locate`.(Citation: Windows Commands JPCERT) Custom tools may also be used to gather file and directory information and interact with the [Native API](https://attack.mitre.org/techniques/T1106). Adversaries may also leverage a [Network Device CLI]

(https://attack.mitre.org/techniques/T1059/008) on network devices to gather file and directory information (e.g. `dir`, `show flash`, and/or `nvram`).(Citation: US-CERT-TA18-106A)

Name

System Network Configuration Discovery

ID

T1016

Description

Adversaries may look for details about the network configuration and settings, such as IP and/or MAC addresses, of systems they access or through information discovery of remote systems. Several operating system administration utilities exist that can be used to gather this information. Examples include [Arp](https://attack.mitre.org/software/S0099), [ipconfig](https://attack.mitre.org/software/S0100)/[ifconfig](https://attack.mitre.org/software/S0101), [nbtstat](https://attack.mitre.org/software/S0102), and [route](https://attack.mitre.org/software/S0103). Adversaries may also leverage a [Network Device CLI](https://attack.mitre.org/techniques/T1059/008) on network devices to gather information about configurations and settings, such as IP addresses of configured interfaces and static/dynamic routes (e.g. `show ip route`, `show ip interface`).(Citation: US-CERT-TA18-106A)(Citation: Mandiant APT41 Global Intrusion) Adversaries may use the information from [System Network Configuration Discovery](https://attack.mitre.org/techniques/T1016) during automated discovery to shape follow-on behaviors, including determining certain access within the target network and what actions to do next.

Name

Credentials from Web Browsers

ID

T1555.003

Description

Adversaries may acquire credentials from web browsers by reading files specific to the target browser.(Citation: Talos Olympic Destroyer 2018) Web browsers commonly save credentials such as website usernames and passwords so that they do not need to be entered manually in the future. Web browsers typically store the credentials in an encrypted format within a credential store; however, methods exist to extract plaintext credentials from web browsers. For example, on Windows systems, encrypted credentials may be obtained from Google Chrome by reading a database file, `AppData\Local\Google\Chrome\User Data\Default\Login Data` and executing a SQL query: `SELECT action_url, username_value, password_value FROM logins;`. The plaintext password can then be obtained by passing the encrypted credentials to the Windows API function `CryptUnprotectData`, which uses the victim's cached logon credentials as the decryption key.(Citation: Microsoft CryptUnprotectData April 2018) Adversaries have executed similar procedures for common web browsers such as FireFox, Safari, Edge, etc. (Citation: Proofpoint Vega Credential Stealer May 2018)(Citation: FireEye HawkEye Malware July 2017) Windows stores Internet Explorer and Microsoft Edge credentials in Credential Lockers managed by the [Windows Credential Manager](https://attack.mitre.org/techniques/T1555/004). Adversaries may also acquire credentials by searching web browser process memory for patterns that commonly match credentials.(Citation: GitHub Mimikittenz July 2016) After acquiring credentials from web browsers, adversaries may attempt to recycle the credentials across different systems and/or accounts in order to expand access. This can result in significantly furthering an adversary's objective in cases where credentials gained from web browsers overlap with privileged accounts (e.g. domain administrator).

Name

Archive Collected Data

ID

T1560

Description

An adversary may compress and/or encrypt data that is collected prior to exfiltration. Compressing the data can help to obfuscate the collected data and minimize the amount of data sent over the network. Encryption can be used to hide information that is being exfiltrated from detection or make exfiltration less conspicuous upon inspection by a defender. Both compression and encryption are done prior to exfiltration, and can be performed using a utility, 3rd party library, or custom method.

Name

Data from Local System

ID

T1005

Description

Adversaries may search local system sources, such as file systems and configuration files or local databases, to find files of interest and sensitive data prior to Exfiltration. Adversaries may do this using a [Command and Scripting Interpreter](https://attack.mitre.org/techniques/T1059), such as [cmd](https://attack.mitre.org/software/S0106) as well as a [Network Device CLI](https://attack.mitre.org/techniques/T1059/008), which have functionality to interact with the file system to gather information.(Citation: show_run_config_cmd_cisco) Adversaries may also use [Automated Collection](https://attack.mitre.org/techniques/T1119) on the local system.

Name

System Information Discovery

ID

T1082

Description

An adversary may attempt to get detailed information about the operating system and hardware, including version, patches, hotfixes, service packs, and architecture. Adversaries may use the information from [System Information Discovery](https://attack.mitre.org/techniques/T1082) during automated discovery to shape follow-on behaviors, including whether or not the adversary fully infects the target and/or attempts specific actions. Tools such as [Systeminfo](https://attack.mitre.org/software/S0096) can be used to gather detailed system information. If running with privileged access, a breakdown of system data can be gathered through the `systemsetup` configuration tool on macOS. As an example, adversaries with user-level access can execute the `df -aH` command to obtain currently

mounted disks and associated freely available space. Adversaries may also leverage a [Network Device CLI](https://attack.mitre.org/techniques/T1059/008) on network devices to gather detailed system information (e.g. `show version`).(Citation: US-CERT-TA18-106A) [System Information Discovery](https://attack.mitre.org/techniques/T1082) combined with information gathered from other forms of discovery and reconnaissance can drive payload development and concealment.(Citation: OSX.FairyTale)(Citation: 20 macOS Common Tools and Techniques) Infrastructure as a Service (IaaS) cloud providers such as AWS, GCP, and Azure allow access to instance and virtual machine information via APIs. Successful authenticated API calls can return data such as the operating system platform and status of a particular instance or the model view of a virtual machine.(Citation: Amazon Describe Instance)(Citation: Google Instances Resource)(Citation: Microsoft Virutal Machine API)

Name

Local Account

ID

T1087.001

Description

Adversaries may attempt to get a listing of local system accounts. This information can help adversaries determine which local accounts exist on a system to aid in follow-on behavior. Commands such as `net user` and `net localgroup` of the [Net](https://attack.mitre.org/software/S0039) utility and `id` and `groups`on macOS and Linux can list local users and groups. On Linux, local users can also be enumerated through the use of the `/etc/passwd` file. On macOS the `dscl . list /Users` command can be used to enumerate local accounts.

Name

Malicious File

ID

T1204.002

Description

An adversary may rely upon a user opening a malicious file in order to gain execution. Users may be subjected to social engineering to get them to open a file that will lead to code execution. This user action will typically be observed as follow-on behavior from [Spearphishing Attachment](https://attack.mitre.org/techniques/T1566/001). Adversaries may use several types of files that require a user to execute them, including .doc, .pdf, .xls, .rtf, .scr, .exe, .lnk, .pif, and .cpl. Adversaries may employ various forms of [Masquerading](https://attack.mitre.org/techniques/T1036) and [Obfuscated Files or Information](https://attack.mitre.org/techniques/T1027) to increase the likelihood that a user will open and successfully execute a malicious file. These methods may include using a familiar naming convention and/or password protecting the file and supplying instructions to a user on how to open it.(Citation: Password Protected Word Docs) While [Malicious File](https://attack.mitre.org/techniques/T1204/002) frequently occurs shortly after Initial Access it may occur at other phases of an intrusion, such as when an adversary places a file in a shared directory or on a user's desktop hoping that a user will click on it. This activity may also be seen shortly after [Internal Spearphishing](https://attack.mitre.org/techniques/T1534).

Name

Digital Certificates

ID

T1588.004

Description

Adversaries may buy and/or steal SSL/TLS certificates that can be used during targeting. SSL/TLS certificates are designed to instill trust. They include information about the key, information about its owner's identity, and the digital signature of an entity that has verified the certificate's contents are correct. If the signature is valid, and the person examining the certificate trusts the signer, then they know they can use that key to communicate with its owner. Adversaries may purchase or steal SSL/TLS certificates to further their operations, such as encrypting C2 traffic (ex: [Asymmetric Cryptography](https://attack.mitre.org/techniques/T1573/002) with [Web Protocols](https://attack.mitre.org/techniques/T1071/001)) or even enabling [Adversary-in-the-Middle](https://attack.mitre.org/techniques/T1557) if the certificate is trusted or otherwise added to the root of trust (i.e. [Install Root Certificate](https://attack.mitre.org/techniques/

T1553/004)). The purchase of digital certificates may be done using a front organization or using information stolen from a previously compromised entity that allows the adversary to validate to a certificate provider as that entity. Adversaries may also steal certificate materials directly from a compromised third-party, including from certificate authorities. (Citation: DiginotarCompromise) Adversaries may register or hijack domains that they will later purchase an SSL/TLS certificate for. Certificate authorities exist that allow adversaries to acquire SSL/TLS certificates, such as domain validation certificates, for free.(Citation: Let's Encrypt FAQ) After obtaining a digital certificate, an adversary may then install that certificate (see [Install Digital Certificate](https://attack.mitre.org/techniques/T1608/003)) on infrastructure under their control.

Name

Steal Web Session Cookie

ID

T1539

Description

An adversary may steal web application or service session cookies and use them to gain access to web applications or Internet services as an authenticated user without needing credentials. Web applications and services often use session cookies as an authentication token after a user has authenticated to a website. Cookies are often valid for an extended period of time, even if the web application is not actively used. Cookies can be found on disk, in the process memory of the browser, and in network traffic to remote systems. Additionally, other applications on the targets machine might store sensitive authentication cookies in memory (e.g. apps which authenticate to cloud services). Session cookies can be used to bypasses some multi-factor authentication protocols.(Citation: Pass The Cookie) There are several examples of malware targeting cookies from web browsers on the local system.(Citation: Kaspersky TajMahal April 2019)(Citation: Unit 42 Mac Crypto Cookies January 2019) There are also open source frameworks such as `Evilginx2` and `Muraena` that can gather session cookies through a malicious proxy (ex: [Adversary-in-the-Middle](https://attack.mitre.org/techniques/T1557)) that can be set up by an adversary and used in phishing campaigns.(Citation: Github evilginx2)(Citation: GitHub Mauraena) After an adversary acquires a valid cookie, they can then perform a [Web Session Cookie] (https://attack.mitre.org/techniques/T1550/004) technique to login to the corresponding web application.

Name

Screen Capture

ID

T1113

Description

Adversaries may attempt to take screen captures of the desktop to gather information over the course of an operation. Screen capturing functionality may be included as a feature of a remote access tool used in post-compromise operations. Taking a screenshot is also typically possible through native utilities or API calls, such as `CopyFromScreen`, `xwd`, or `screencapture`.(Citation: CopyFromScreen .NET)(Citation: Antiquated Mac Malware)

Name

Web Protocols

ID

T1071.001

Description

Adversaries may communicate using application layer protocols associated with web traffic to avoid detection/network filtering by blending in with existing traffic. Commands to the remote system, and often the results of those commands, will be embedded within the protocol traffic between the client and server. Protocols such as HTTP/S(Citation: CrowdStrike Putter Panda) and WebSocket(Citation: Brazking-Websockets) that carry web traffic may be very common in environments. HTTP/S packets have many fields and headers in which data can be concealed. An adversary may abuse these protocols to communicate with systems under their control within a victim network while also mimicking normal, expected traffic.

Name

Software Packing

ID

T1027.002

Description

Adversaries may perform software packing or virtual machine software protection to conceal their code. Software packing is a method of compressing or encrypting an executable. Packing an executable changes the file signature in an attempt to avoid signature-based detection. Most decompression techniques decompress the executable code in memory. Virtual machine software protection translates an executable's original code into a special format that only a special virtual machine can run. A virtual machine is then called to run this code.(Citation: ESET FinFisher Jan 2018) Utilities used to perform software packing are called packers. Example packers are MPRESS and UPX. A more comprehensive list of known packers is available, but adversaries may create their own packing techniques that do not leave the same artifacts as well-known packers to evade defenses.(Citation: Awesome Executable Packing)

Name

Windows Command Shell

ID

T1059.003

Description

Adversaries may abuse the Windows command shell for execution. The Windows command shell ([cmd](https://attack.mitre.org/software/S0106)) is the primary command prompt on Windows systems. The Windows command prompt can be used to control almost any aspect of a system, with various permission levels required for different subsets of commands. The command prompt can be invoked remotely via [Remote Services](https://attack.mitre.org/techniques/T1021) such as [SSH](https://attack.mitre.org/

techniques/T1021/004).(Citation: SSH in Windows) Batch files (ex: .bat or .cmd) also provide the shell with a list of sequential commands to run, as well as normal scripting operations such as conditionals and loops. Common uses of batch files include long or repetitive tasks, or the need to run the same set of commands on multiple systems. Adversaries may leverage [cmd](https://attack.mitre.org/software/S0106) to execute various commands and payloads. Common uses include [cmd](https://attack.mitre.org/software/S0106) to execute a single command, or abusing [cmd](https://attack.mitre.org/software/S0106) interactively with input and output forwarded over a command and control channel.

Name

Security Software Discovery

ID

T1518.001

Description

Adversaries may attempt to get a listing of security software, configurations, defensive tools, and sensors that are installed on a system or in a cloud environment. This may include things such as firewall rules and anti-virus. Adversaries may use the information from [Security Software Discovery](https://attack.mitre.org/techniques/T1518/001) during automated discovery to shape follow-on behaviors, including whether or not the adversary fully infects the target and/or attempts specific actions. Example commands that can be used to obtain security software information are [netsh](https://attack.mitre.org/software/S0108), `reg query` with [Reg](https://attack.mitre.org/software/S0075), `dir` with [cmd](https://attack.mitre.org/software/S0106), and [Tasklist](https://attack.mitre.org/software/S0057), but other indicators of discovery behavior may be more specific to the type of software or security system the adversary is looking for. It is becoming more common to see macOS malware perform checks for LittleSnitch and KnockKnock software. Adversaries may also utilize cloud APIs to discover the configurations of firewall rules within an environment.(Citation: Expel IO Evil in AWS) For example, the permitted IP ranges, ports or user accounts for the inbound/outbound rules of security groups, virtual firewalls established within AWS for EC2 and/or VPC instances, can be revealed by the `DescribeSecurityGroups` action with various request parameters. (Citation: DescribeSecurityGroups - Amazon Elastic Compute Cloud)

Name

PowerShell

ID

T1059.001

Description

Adversaries may abuse PowerShell commands and scripts for execution. PowerShell is a powerful interactive command-line interface and scripting environment included in the Windows operating system.(Citation: TechNet PowerShell) Adversaries can use PowerShell to perform a number of actions, including discovery of information and execution of code. Examples include the `Start-Process` cmdlet which can be used to run an executable and the `Invoke-Command` cmdlet which runs a command locally or on a remote computer (though administrator permissions are required to use PowerShell to connect to remote systems). PowerShell may also be used to download and run executables from the Internet, which can be executed from disk or in memory without touching disk. A number of PowerShell-based offensive testing tools are available, including [Empire](https://attack.mitre.org/software/S0363), [PowerSploit](https://attack.mitre.org/software/S0194), [PoshC2](https://attack.mitre.org/software/S0378), and PSAttack.(Citation: Github PSAttack) PowerShell commands/scripts can also be executed without directly invoking the `powershell.exe` binary through interfaces to PowerShell's underlying `System.Management.Automation` assembly DLL exposed through the .NET framework and Windows Common Language Interface (CLI).(Citation: Sixdub PowerPick Jan 2016)(Citation: SilentBreak Offensive PS Dec 2015)(Citation: Microsoft PSfromCsharp APR 2014)

Name

Exfiltration Over C2 Channel

ID

T1041

Description

Adversaries may steal data by exfiltrating it over an existing command and control channel. Stolen data is encoded into the normal communications channel using the same protocol as command and control communications.

Name

Process Discovery

ID

T1057

Description

Adversaries may attempt to get information about running processes on a system. Information obtained could be used to gain an understanding of common software/applications running on systems within the network. Adversaries may use the information from [Process Discovery](https://attack.mitre.org/techniques/T1057) during automated discovery to shape follow-on behaviors, including whether or not the adversary fully infects the target and/or attempts specific actions. In Windows environments, adversaries could obtain details on running processes using the [Tasklist](https://attack.mitre.org/software/S0057) utility via [cmd](https://attack.mitre.org/software/S0106) or `Get-Process` via [PowerShell](https://attack.mitre.org/techniques/T1059/001). Information about processes can also be extracted from the output of [Native API](https://attack.mitre.org/techniques/T1106) calls such as `CreateToolhelp32Snapshot`. In Mac and Linux, this is accomplished with the `ps` command. Adversaries may also opt to enumerate processes via /proc. On network devices, [Network Device CLI](https://attack.mitre.org/techniques/T1059/008) commands such as `show processes` can be used to display current running processes.(Citation: US-CERT-TA18-106A)(Citation: show_processes_cisco_cmd)

Name

File and Directory Discovery

ID

T1083

Description

Adversaries may enumerate files and directories or may search in specific locations of a host or network share for certain information within a file system. Adversaries may use the information from [File and Directory Discovery](https://attack.mitre.org/techniques/T1083) during automated discovery to shape follow-on behaviors, including whether or not the adversary fully infects the target and/or attempts specific actions. Many command shell utilities can be used to obtain this information. Examples include `dir`, `tree`, `ls`, `find`, and `locate`.(Citation: Windows Commands JPCERT) Custom tools may also be used to gather file and directory information and interact with the [Native API](https://attack.mitre.org/techniques/T1106). Adversaries may also leverage a [Network Device CLI](https://attack.mitre.org/techniques/T1059/008) on network devices to gather file and directory information (e.g. `dir`, `show flash`, and/or `nvram`).(Citation: US-CERT-TA18-106A)

Name

System Network Configuration Discovery

ID

T1016

Description

Adversaries may look for details about the network configuration and settings, such as IP and/or MAC addresses, of systems they access or through information discovery of remote systems. Several operating system administration utilities exist that can be used to gather this information. Examples include [Arp](https://attack.mitre.org/software/S0099), [ipconfig](https://attack.mitre.org/software/S0100)/[ifconfig](https://attack.mitre.org/software/S0101), [nbtstat](https://attack.mitre.org/software/S0102), and [route](https://attack.mitre.org/software/S0103). Adversaries may also leverage a [Network Device CLI](https://attack.mitre.org/techniques/T1059/008) on network devices to gather information about configurations and settings, such as IP addresses of configured interfaces and static/dynamic routes (e.g. `show ip route`, `show ip interface`).(Citation: US-CERT-TA18-106A)(Citation: Mandiant APT41 Global Intrusion) Adversaries may use the information from [System Network Configuration Discovery](https://attack.mitre.org/techniques/T1016) during automated discovery to shape follow-on behaviors, including determining certain access within the target network and what actions to do next.

Name

Credentials from Web Browsers

ID

T1555.003

Description

Adversaries may acquire credentials from web browsers by reading files specific to the target browser.(Citation: Talos Olympic Destroyer 2018) Web browsers commonly save credentials such as website usernames and passwords so that they do not need to be entered manually in the future. Web browsers typically store the credentials in an encrypted format within a credential store; however, methods exist to extract plaintext credentials from web browsers. For example, on Windows systems, encrypted credentials may be obtained from Google Chrome by reading a database file, `AppData\Local\Google\Chrome\User Data\Default\Login Data` and executing a SQL query: `SELECT action_url, username_value, password_value FROM logins;`. The plaintext password can then be obtained by passing the encrypted credentials to the Windows API function `CryptUnprotectData`, which uses the victim’s cached logon credentials as the decryption key.(Citation: Microsoft CryptUnprotectData April 2018) Adversaries have executed similar procedures for common web browsers such as FireFox, Safari, Edge, etc. (Citation: Proofpoint Vega Credential Stealer May 2018)(Citation: FireEye HawkEye Malware July 2017) Windows stores Internet Explorer and Microsoft Edge credentials in Credential Lockers managed by the [Windows Credential Manager](https://attack.mitre.org/techniques/T1555/004). Adversaries may also acquire credentials by searching web browser process memory for patterns that commonly match credentials.(Citation: GitHub Mimikittenz July 2016) After acquiring credentials from web browsers, adversaries may attempt to recycle the credentials across different systems and/or accounts in order to expand access. This can result in significantly furthering an adversary's objective in cases where credentials gained from web browsers overlap with privileged accounts (e.g. domain administrator).

Name

Archive Collected Data

ID

T1560

Description

An adversary may compress and/or encrypt data that is collected prior to exfiltration. Compressing the data can help to obfuscate the collected data and minimize the amount of data sent over the network. Encryption can be used to hide information that is being exfiltrated from detection or make exfiltration less conspicuous upon inspection by a defender. Both compression and encryption are done prior to exfiltration, and can be performed using a utility, 3rd party library, or custom method.

Name

Data from Local System

ID

T1005

Description

Adversaries may search local system sources, such as file systems and configuration files or local databases, to find files of interest and sensitive data prior to Exfiltration. Adversaries may do this using a [Command and Scripting Interpreter](https://attack.mitre.org/techniques/T1059), such as [cmd](https://attack.mitre.org/software/S0106) as well as a [Network Device CLI](https://attack.mitre.org/techniques/T1059/008), which have functionality to interact with the file system to gather information.(Citation: show_run_config_cmd_cisco) Adversaries may also use [Automated Collection](https://attack.mitre.org/techniques/T1119) on the local system.

Name

System Information Discovery

ID

T1082

Description

An adversary may attempt to get detailed information about the operating system and hardware, including version, patches, hotfixes, service packs, and architecture. Adversaries may use the information from [System Information Discovery](https://attack.mitre.org/techniques/T1082) during automated discovery to shape follow-on behaviors, including whether or not the adversary fully infects the target and/or attempts specific actions. Tools such as [Systeminfo](https://attack.mitre.org/software/S0096) can be used to gather detailed system information. If running with privileged access, a breakdown of system data can be gathered through the `systemsetup` configuration tool on macOS. As an example, adversaries with user-level access can execute the `df -aH` command to obtain currently mounted disks and associated freely available space. Adversaries may also leverage a [Network Device CLI](https://attack.mitre.org/techniques/T1059/008) on network devices to gather detailed system information (e.g. `show version`).(Citation: US-CERT-TA18-106A) [System Information Discovery](https://attack.mitre.org/techniques/T1082) combined with information gathered from other forms of discovery and reconnaissance can drive payload development and concealment.(Citation: OSX.FairyTale)(Citation: 20 macOS Common Tools and Techniques) Infrastructure as a Service (IaaS) cloud providers such as AWS, GCP, and Azure allow access to instance and virtual machine information via APIs. Successful authenticated API calls can return data such as the operating system platform and status of a particular instance or the model view of a virtual machine.(Citation: Amazon Describe Instance)(Citation: Google Instances Resource)(Citation: Microsoft Virutal Machine API)

Name

Local Account

ID

T1087.001

Description

Adversaries may attempt to get a listing of local system accounts. This information can help adversaries determine which local accounts exist on a system to aid in follow-on behavior. Commands such as `net user` and `net localgroup` of the [Net](https://attack.mitre.org/software/S0039) utility and `id` and `groups`on macOS and Linux can list local users and groups. On Linux, local users can also be enumerated through the use of

the `/etc/passwd` file. On macOS the `dscl . list /Users` command can be used to enumerate local accounts.

Name

Malicious File

ID

T1204.002

Description

An adversary may rely upon a user opening a malicious file in order to gain execution. Users may be subjected to social engineering to get them to open a file that will lead to code execution. This user action will typically be observed as follow-on behavior from [Spearphishing Attachment](https://attack.mitre.org/techniques/T1566/001). Adversaries may use several types of files that require a user to execute them, including .doc, .pdf, .xls, .rtf, .scr, .exe, .lnk, .pif, and .cpl. Adversaries may employ various forms of [Masquerading](https://attack.mitre.org/techniques/T1036) and [Obfuscated Files or Information](https://attack.mitre.org/techniques/T1027) to increase the likelihood that a user will open and successfully execute a malicious file. These methods may include using a familiar naming convention and/or password protecting the file and supplying instructions to a user on how to open it.(Citation: Password Protected Word Docs) While [Malicious File](https://attack.mitre.org/techniques/T1204/002) frequently occurs shortly after Initial Access it may occur at other phases of an intrusion, such as when an adversary places a file in a shared directory or on a user's desktop hoping that a user will click on it. This activity may also be seen shortly after [Internal Spearphishing](https://attack.mitre.org/techniques/T1534).

Name

Digital Certificates

ID

T1588.004

Description

Adversaries may buy and/or steal SSL/TLS certificates that can be used during targeting. SSL/TLS certificates are designed to instill trust. They include information about the key, information about its owner's identity, and the digital signature of an entity that has verified the certificate's contents are correct. If the signature is valid, and the person examining the certificate trusts the signer, then they know they can use that key to communicate with its owner. Adversaries may purchase or steal SSL/TLS certificates to further their operations, such as encrypting C2 traffic (ex: [Asymmetric Cryptography](https://attack.mitre.org/techniques/T1573/002) with [Web Protocols](https://attack.mitre.org/techniques/T1071/001)) or even enabling [Adversary-in-the-Middle](https://attack.mitre.org/techniques/T1557) if the certificate is trusted or otherwise added to the root of trust (i.e. [Install Root Certificate](https://attack.mitre.org/techniques/T1553/004)). The purchase of digital certificates may be done using a front organization or using information stolen from a previously compromised entity that allows the adversary to validate to a certificate provider as that entity. Adversaries may also steal certificate materials directly from a compromised third-party, including from certificate authorities. (Citation: DiginotarCompromise) Adversaries may register or hijack domains that they will later purchase an SSL/TLS certificate for. Certificate authorities exist that allow adversaries to acquire SSL/TLS certificates, such as domain validation certificates, for free.(Citation: Let's Encrypt FAQ) After obtaining a digital certificate, an adversary may then install that certificate (see [Install Digital Certificate](https://attack.mitre.org/techniques/T1608/003)) on infrastructure under their control.

Name

Steal Web Session Cookie

ID

T1539

Description

An adversary may steal web application or service session cookies and use them to gain access to web applications or Internet services as an authenticated user without needing credentials. Web applications and services often use session cookies as an authentication token after a user has authenticated to a website. Cookies are often valid for an extended period of time, even if the web application is not actively used. Cookies can be found on disk, in the process memory of the browser, and in network traffic to remote systems.

Additionally, other applications on the targets machine might store sensitive authentication cookies in memory (e.g. apps which authenticate to cloud services). Session cookies can be used to bypasses some multi-factor authentication protocols.(Citation: Pass The Cookie) There are several examples of malware targeting cookies from web browsers on the local system.(Citation: Kaspersky TajMahal April 2019)(Citation: Unit 42 Mac Crypto Cookies January 2019) There are also open source frameworks such as `Evilginx2` and `Muraena` that can gather session cookies through a malicious proxy (ex: [Adversary-in-the-Middle](https://attack.mitre.org/techniques/T1557)) that can be set up by an adversary and used in phishing campaigns.(Citation: Github evilginx2)(Citation: GitHub Mauraena) After an adversary acquires a valid cookie, they can then perform a [Web Session Cookie] (https://attack.mitre.org/techniques/T1550/004) technique to login to the corresponding web application.

Name

Screen Capture

ID

T1113

Description

Adversaries may attempt to take screen captures of the desktop to gather information over the course of an operation. Screen capturing functionality may be included as a feature of a remote access tool used in post-compromise operations. Taking a screenshot is also typically possible through native utilities or API calls, such as `CopyFromScreen`, `xwd`, or `screencapture`.(Citation: CopyFromScreen .NET)(Citation: Antiquated Mac Malware)

Country

Name
Korea, Republic of

Name
Korea, Republic of

Sector

Name
Government and administrations
Description
Civilian government institutions and administrations of the executive and legislative branches. The diplomatic and judicial branches are not included.
Name
Government and administrations
Description
Civilian government institutions and administrations of the executive and legislative branches. The diplomatic and judicial branches are not included.

Hostname

Value
qi.limsjo.p-e.kr
ol.negapa.p-e.kr
ar.kostin.p-e.kr
ai.negapa.p-e.kr
ai.limsjo.p-e.kr
ai.kostin.p-e.kr
qi.limsjo.p-e.kr
ol.negapa.p-e.kr
ar.kostin.p-e.kr
ai.negapa.p-e.kr
ai.limsjo.p-e.kr
ai.kostin.p-e.kr

IPv4-Addr

Value
216.189.159.197
216.189.159.197

StixFile

Value
ff3718ae6bd59ad479e375c602a81811718dfb2669c2d1de497f02baf7b4adca
f8ab78e1db3a3cc3793f7680a90dc1d8ce087226ef59950b7acd6bb1beffd6e3
bc4c1c869a03045e0b594a258ec3801369b0dcabac193e90f0a684900e9a582d
a8c24a3e54a4b323973f61630c92ecaad067598ef2547350c9d108bc175774b9
97df5304f53fec6a5d2d2bd75b9310a3747b681520fe45d2961bc4df86e556d7
955cb4f01eb18f0d259fcb962e36a339e8fe082963dfd9f72d3851210f7d2d3b
6eebb5ed0d0b5553e40a7b1ad739589709d077aab4cbea1c64713c48ce9c96f9
61b8fbea8c0dfa337eb7ff978124ddf496d0c5f29bcb5672f3bd3d6bf832ac92
2e0ffaab995f22b7684052e53b8c64b9283b5e81503b88664785fe6d6569a55e
ff3718ae6bd59ad479e375c602a81811718dfb2669c2d1de497f02baf7b4adca
f8ab78e1db3a3cc3793f7680a90dc1d8ce087226ef59950b7acd6bb1beffd6e3
bc4c1c869a03045e0b594a258ec3801369b0dcabac193e90f0a684900e9a582d
a8c24a3e54a4b323973f61630c92ecaad067598ef2547350c9d108bc175774b9

97df5304f53fec6a5d2d2bd75b9310a3747b681520fe45d2961bc4df86e556d7

955cb4f01eb18f0d259fcb962e36a339e8fe082963dfd9f72d3851210f7d2d3b

6eebb5ed0d0b5553e40a7b1ad739589709d077aab4cbea1c64713c48ce9c96f9

61b8fbea8c0dfa337eb7ff978124ddf496d0c5f29bcb5672f3bd3d6bf832ac92

2e0ffaab995f22b7684052e53b8c64b9283b5e81503b88664785fe6d6569a55e

External References

- https://medium.com/s2wblog/kimsuky-disguised-as-a-korean-company-signed-with-a-valid-certificate-to-distribute-troll-stealer-cfa5d54314e2
- https://otx.alienvault.com/pulse/65c6395cc792b1c60e302242