

NETMANAGEIT

Intelligence Report

Software Installer

Impersonation Kimsuky

(APT-Q-2) Information

Stealing Operation

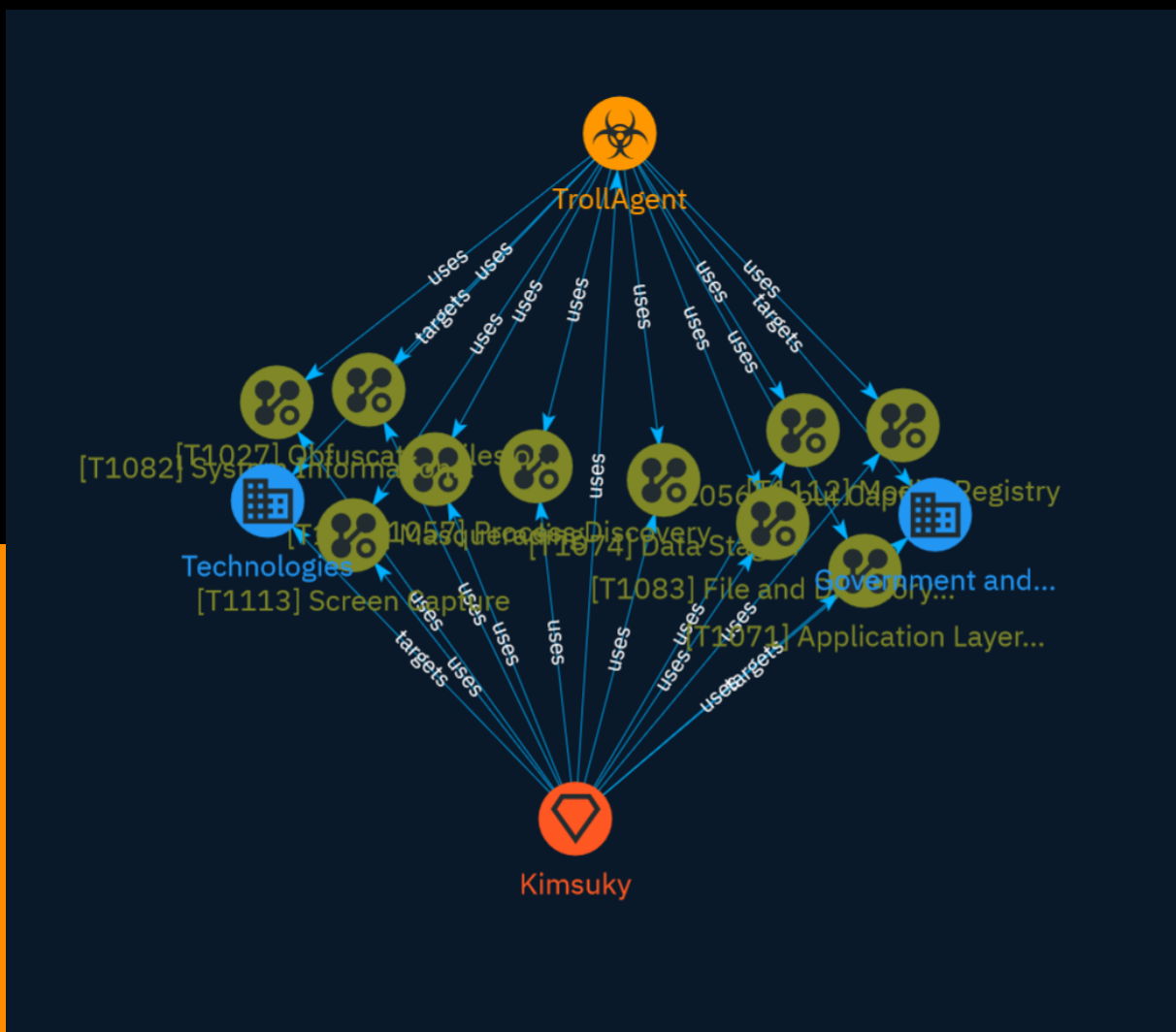


Table of contents

Overview

● Description	3
● Confidence	3
● Content	4

Entities

● Malware	5
● Intrusion-Set	6
● Attack-Pattern	8
● Sector	21

External References

● External References	23
-----------------------	----

Overview

Description

This report describes a recent information stealing operation by the threat actor group Kimsuky, also known as APT-Q-2. The attackers distributed malware impersonating software installers from South Korean companies, which collected sensitive information from compromised devices and exfiltrated it to attacker-controlled servers.

Confidence

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

15 / 100

Content

N/A

Malware

Name

TrollAgent

Name

TrollAgent

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

Input Capture

ID

T1056

Description

Adversaries may use methods of capturing user input to obtain credentials or collect information. During normal system usage, users often provide credentials to various different locations, such as login pages/portals or system dialog boxes. Input capture mechanisms may be transparent to the user (e.g. [Credential API Hooking](https://attack.mitre.org/techniques/T1056/004)) or rely on deceiving the user into providing input into what they believe to be a genuine service (e.g. [Web Portal Capture](https://attack.mitre.org/techniques/T1056/003)).

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

Data Staged

ID

T1074

Description

Adversaries may stage collected data in a central location or directory prior to Exfiltration. Data may be kept in separate files or combined into one file through techniques such as [Archive Collected Data](<https://attack.mitre.org/techniques/T1560>). Interactive command shells may be used, and common functionality within [cmd](<https://attack.mitre.org/software/S0106>) and bash may be used to copy data into a staging location.(Citation: PWC Cloud Hopper April 2017) In cloud environments, adversaries may stage data within a particular instance or virtual machine before exfiltration. An adversary may [Create Cloud Instance](<https://attack.mitre.org/techniques/T1578/002>) and stage data in that instance. (Citation: Mandiant M-Trends 2020) Adversaries may choose to stage data from a victim network in a centralized location prior to Exfiltration to minimize the number of connections made to their C2 server and better evade detection.

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

Obfuscated Files or Information

ID

T1027

Description

Adversaries may attempt to make an executable or file difficult to discover or analyze by encrypting, encoding, or otherwise obfuscating its contents on the system or in transit. This is common behavior that can be used across different platforms and the network to evade defenses. Payloads may be compressed, archived, or encrypted in order to avoid detection. These payloads may be used during Initial Access or later to mitigate detection. Sometimes a user's action may be required to open and [Deobfuscate/Decode Files or Information](<https://attack.mitre.org/techniques/T1140>) for [User Execution](<https://attack.mitre.org/techniques/T1204>). The user may also be required to input a password to

open a password protected compressed/encrypted file that was provided by the adversary. (Citation: Volexity PowerDuke November 2016) Adversaries may also use compressed or archived scripts, such as JavaScript. Portions of files can also be encoded to hide the plain-text strings that would otherwise help defenders with discovery. (Citation: Linux/Cdorked.A We Live Security Analysis) Payloads may also be split into separate, seemingly benign files that only reveal malicious functionality when reassembled. (Citation: Carbon Black Obfuscation Sept 2016) Adversaries may also abuse [Command Obfuscation](<https://attack.mitre.org/techniques/T1027/010>) to obscure commands executed from payloads or directly via [Command and Scripting Interpreter](<https://attack.mitre.org/techniques/T1059>). Environment variables, aliases, characters, and other platform/language specific semantics can be used to evade signature based detections and application control mechanisms. (Citation: FireEye Obfuscation June 2017) (Citation: FireEye Revoke-Obfuscation July 2017)(Citation: PaloAlto EncodedCommand March 2017)

Name

Modify Registry

ID

T1112

Description

Adversaries may interact with the Windows Registry to hide configuration information within Registry keys, remove information as part of cleaning up, or as part of other techniques to aid in persistence and execution. Access to specific areas of the Registry depends on account permissions, some requiring administrator-level access. The built-in Windows command-line utility [Reg](<https://attack.mitre.org/software/S0075>) may be used for local or remote Registry modification. (Citation: Microsoft Reg) Other tools may also be used, such as a remote access tool, which may contain functionality to interact with the Registry through the Windows API. Registry modifications may also include actions to hide keys, such as prepending key names with a null character, which will cause an error and/or be ignored when read via [Reg](<https://attack.mitre.org/software/S0075>) or other utilities using the Win32 API. (Citation: Microsoft Reghide NOV 2006) Adversaries may abuse these pseudo-hidden keys to conceal payloads/commands used to maintain persistence. (Citation: TrendMicro POWELIKS AUG 2014) (Citation: SpectorOps Hiding Reg Jul 2017) The Registry of a remote system may be modified to aid in execution of files as part of lateral movement. It requires the remote Registry service to be running on the target system. (Citation: Microsoft Remote) Often [Valid Accounts](<https://attack.mitre.org/techniques/>

T1078) are required, along with access to the remote system's [SMB/Windows Admin Shares](<https://attack.mitre.org/techniques/T1021/002>) for RPC communication.

Name

Masquerading

ID

T1036

Description

Adversaries may attempt to manipulate features of their artifacts to make them appear legitimate or benign to users and/or security tools. Masquerading occurs when the name or location of an object, legitimate or malicious, is manipulated or abused for the sake of evading defenses and observation. This may include manipulating file metadata, tricking users into misidentifying the file type, and giving legitimate task or service names. Renaming abusible system utilities to evade security monitoring is also a form of [Masquerading](<https://attack.mitre.org/techniques/T1036>). (Citation: LOLBAS Main Site) Masquerading may also include the use of [Proxy](<https://attack.mitre.org/techniques/T1090>) or VPNs to disguise IP addresses, which can allow adversaries to blend in with normal network traffic and bypass conditional access policies or anti-abuse protections.

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 Virtual Machine API)

Name

Application Layer Protocol

ID

T1071

Description

Adversaries may communicate using OSI application layer protocols 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. Adversaries may utilize many different protocols, including those used for web browsing, transferring files, electronic mail, or DNS. For connections that occur internally within an enclave (such as those between a proxy or pivot node and other nodes), commonly used protocols are SMB, SSH, or RDP.

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

Input Capture

ID

T1056

Description

Adversaries may use methods of capturing user input to obtain credentials or collect information. During normal system usage, users often provide credentials to various different locations, such as login pages/portals or system dialog boxes. Input capture mechanisms may be transparent to the user (e.g. [Credential API Hooking](https://attack.mitre.org/techniques/T1056/004)) or rely on deceiving the user into providing input into what they believe to be a genuine service (e.g. [Web Portal Capture](https://attack.mitre.org/techniques/T1056/003)).

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

Data Staged

ID

T1074

Description

Adversaries may stage collected data in a central location or directory prior to Exfiltration. Data may be kept in separate files or combined into one file through techniques such as [Archive Collected Data](<https://attack.mitre.org/techniques/T1560>). Interactive command shells may be used, and common functionality within [cmd](<https://attack.mitre.org/software/S0106>) and bash may be used to copy data into a staging location.(Citation: PWC Cloud Hopper April 2017) In cloud environments, adversaries may stage data within a particular instance or virtual machine before exfiltration. An adversary may [Create Cloud Instance](<https://attack.mitre.org/techniques/T1578/002>) and stage data in that instance. (Citation: Mandiant M-Trends 2020) Adversaries may choose to stage data from a victim

network in a centralized location prior to Exfiltration to minimize the number of connections made to their C2 server and better evade detection.

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

Obfuscated Files or Information

ID

T1027

Description

Adversaries may attempt to make an executable or file difficult to discover or analyze by encrypting, encoding, or otherwise obfuscating its contents on the system or in transit. This is common behavior that can be used across different platforms and the network to

evade defenses. Payloads may be compressed, archived, or encrypted in order to avoid detection. These payloads may be used during Initial Access or later to mitigate detection. Sometimes a user's action may be required to open and [Deobfuscate/Decode Files or Information](<https://attack.mitre.org/techniques/T1140>) for [User Execution](<https://attack.mitre.org/techniques/T1204>). The user may also be required to input a password to open a password protected compressed/encrypted file that was provided by the adversary. (Citation: Volexity PowerDuke November 2016) Adversaries may also use compressed or archived scripts, such as JavaScript. Portions of files can also be encoded to hide the plain-text strings that would otherwise help defenders with discovery. (Citation: Linux/Cdorked.A We Live Security Analysis) Payloads may also be split into separate, seemingly benign files that only reveal malicious functionality when reassembled. (Citation: Carbon Black Obfuscation Sept 2016) Adversaries may also abuse [Command Obfuscation](<https://attack.mitre.org/techniques/T1027/010>) to obscure commands executed from payloads or directly via [Command and Scripting Interpreter](<https://attack.mitre.org/techniques/T1059>). Environment variables, aliases, characters, and other platform/language specific semantics can be used to evade signature based detections and application control mechanisms. (Citation: FireEye Obfuscation June 2017) (Citation: FireEye Revoke-Obfuscation July 2017)(Citation: PaloAlto EncodedCommand March 2017)

Name

Modify Registry

ID

T1112

Description

Adversaries may interact with the Windows Registry to hide configuration information within Registry keys, remove information as part of cleaning up, or as part of other techniques to aid in persistence and execution. Access to specific areas of the Registry depends on account permissions, some requiring administrator-level access. The built-in Windows command-line utility [Reg](<https://attack.mitre.org/software/S0075>) may be used for local or remote Registry modification. (Citation: Microsoft Reg) Other tools may also be used, such as a remote access tool, which may contain functionality to interact with the Registry through the Windows API. Registry modifications may also include actions to hide keys, such as prepending key names with a null character, which will cause an error and/or be ignored when read via [Reg](<https://attack.mitre.org/software/S0075>) or other utilities using the Win32 API. (Citation: Microsoft Reghide NOV 2006) Adversaries may abuse these pseudo-hidden keys to conceal payloads/commands used to maintain persistence.

(Citation: TrendMicro POWELIKS AUG 2014) (Citation: SpectorOps Hiding Reg Jul 2017) The Registry of a remote system may be modified to aid in execution of files as part of lateral movement. It requires the remote Registry service to be running on the target system. (Citation: Microsoft Remote) Often [Valid Accounts](<https://attack.mitre.org/techniques/T1078>) are required, along with access to the remote system's [SMB/Windows Admin Shares](<https://attack.mitre.org/techniques/T1021/002>) for RPC communication.

Name

Masquerading

ID

T1036

Description

Adversaries may attempt to manipulate features of their artifacts to make them appear legitimate or benign to users and/or security tools. Masquerading occurs when the name or location of an object, legitimate or malicious, is manipulated or abused for the sake of evading defenses and observation. This may include manipulating file metadata, tricking users into misidentifying the file type, and giving legitimate task or service names. Renaming abusible system utilities to evade security monitoring is also a form of [Masquerading](<https://attack.mitre.org/techniques/T1036>). (Citation: LOLBAS Main Site) Masquerading may also include the use of [Proxy](<https://attack.mitre.org/techniques/T1090>) or VPNs to disguise IP addresses, which can allow adversaries to blend in with normal network traffic and bypass conditional access policies or anti-abuse protections.

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 Virtual Machine API)

Name

Application Layer Protocol

ID

T1071

Description

Adversaries may communicate using OSI application layer protocols 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. Adversaries may utilize many different protocols, including those used for web browsing, transferring files, electronic mail, or DNS. For connections that occur internally within an enclave (such as those between a proxy or pivot node and other nodes), commonly used protocols are SMB, SSH, or RDP.

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)

Sector

Name

Technologies

Description

Private entities related to the research, development, manufacturing and distribution of electronics, softwares, computers and products related to information technologies.

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

Technologies

Description

Private entities related to the research, development, manufacturing and distribution of electronics, softwares, computers and products related to information technologies.

Name

Government and administrations

Description

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

External References

-
- https://mp.weixin.qq.com/s?__biz=MzI2MDc2MDA4OA%3D%3D&mid=2247509476&idx=1&sn=b0e09436095203b75710836d718d6699
-
- <https://otx.alienvault.com/pulse/65c117536df0c4c6c3cc85e9>