

NETMANAGEIT

Intelligence Report

A cascade of compromise: unveiling Lazarus' new campaign

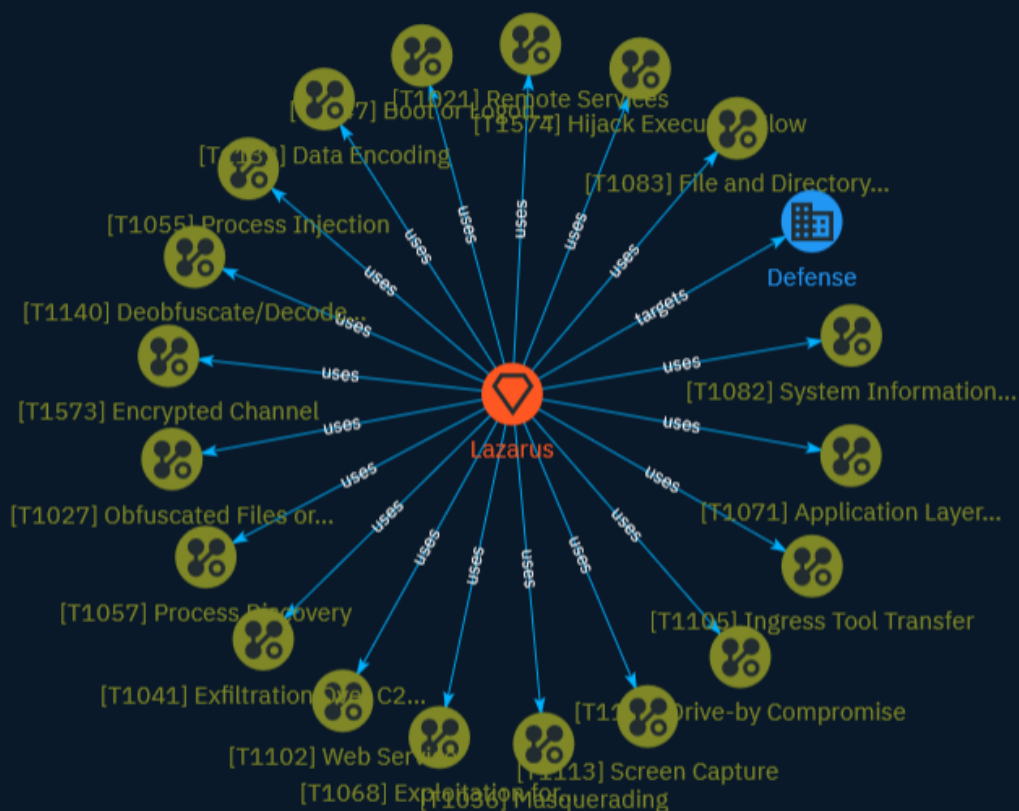


Table of contents

Overview

● Description	4
● Confidence	4
● Content	5

Entities

● Attack-Pattern	6
● Sector	21
● Intrusion-Set	22

Observables

● Url	23
-------	----



External References

- External References

26

Overview

Description

Security firm Kaspersky has identified and identified the malware used in a series of attacks carried out by the group known as Lazarus, which has previously targeted high-profile companies and the cryptocurrency industry.

Confidence

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

15 / 100

Content

N/A

Attack-Pattern

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

OS Credential Dumping

ID

T1003

Description

Adversaries may attempt to dump credentials to obtain account login and credential material, normally in the form of a hash or a clear text password, from the operating system and software. Credentials can then be used to perform [Lateral Movement](<https://attack.mitre.org/tactics/TA0008>) and access restricted information. Several of the tools mentioned in associated sub-techniques may be used by both adversaries and professional security testers. Additional custom tools likely exist as well.

Name

Boot or Logon Autostart Execution

ID

T1547

Description

Adversaries may configure system settings to automatically execute a program during system boot or logon to maintain persistence or gain higher-level privileges on compromised systems. Operating systems may have mechanisms for automatically running a program on system boot or account logon.(Citation: Microsoft Run Key)(Citation: MSDN Authentication Packages)(Citation: Microsoft TimeProvider)(Citation: Cylance Reg Persistence Sept 2013)(Citation: Linux Kernel Programming) These mechanisms may include automatically executing programs that are placed in specially designated directories or are referenced by repositories that store configuration information, such as the Windows Registry. An adversary may achieve the same goal by modifying or extending features of the kernel. Since some boot or logon autostart programs run with higher privileges, an adversary may leverage these to elevate privileges.

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 abusable system utilities to evade security monitoring is also a form of [Masquerading](<https://attack.mitre.org/techniques/T1036>). (Citation: LOLBAS Main Site)

Name

Process Injection

ID

T1055

Description

Adversaries may inject code into processes in order to evade process-based defenses as well as possibly elevate privileges. Process injection is a method of executing arbitrary code in the address space of a separate live process. Running code in the context of another process may allow access to the process's memory, system/network resources, and possibly elevated privileges. Execution via process injection may also evade detection from security products since the execution is masked under a legitimate process. There are many different ways to inject code into a process, many of which abuse legitimate functionalities. These implementations exist for every major OS but are typically platform specific. More sophisticated samples may perform multiple process injections to segment modules and further evade detection, utilizing named pipes or other inter-process communication (IPC) mechanisms as a communication channel.

Name

Encrypted Channel

ID

T1573

Description

Adversaries may employ a known encryption algorithm to conceal command and control traffic rather than relying on any inherent protections provided by a communication protocol. Despite the use of a secure algorithm, these implementations may be vulnerable to reverse engineering if secret keys are encoded and/or generated within malware samples/configuration files.

Name

Exploitation for Privilege Escalation

ID

T1068

Description

Adversaries may exploit software vulnerabilities in an attempt to elevate privileges. Exploitation of a software vulnerability occurs when an adversary takes advantage of a programming error in a program, service, or within the operating system software or kernel itself to execute adversary-controlled code. Security constructs such as permission levels will often hinder access to information and use of certain techniques, so adversaries will likely need to perform privilege escalation to include use of software exploitation to circumvent those restrictions. When initially gaining access to a system, an adversary may be operating within a lower privileged process which will prevent them from accessing certain resources on the system. Vulnerabilities may exist, usually in operating system components and software commonly running at higher permissions, that can be exploited to gain higher levels of access on the system. This could enable someone to move from unprivileged or user level permissions to SYSTEM or root permissions depending on the

component that is vulnerable. This could also enable an adversary to move from a virtualized environment, such as within a virtual machine or container, onto the underlying host. This may be a necessary step for an adversary compromising an endpoint system that has been properly configured and limits other privilege escalation methods. Adversaries may bring a signed vulnerable driver onto a compromised machine so that they can exploit the vulnerability to execute code in kernel mode. This process is sometimes referred to as Bring Your Own Vulnerable Driver (BYOVD).(Citation: ESET InvisiMole June 2020)(Citation: Unit42 AcidBox June 2020) Adversaries may include the vulnerable driver with files delivered during Initial Access or download it to a compromised system via [Ingress Tool Transfer](<https://attack.mitre.org/techniques/T1105>) or [Lateral Tool Transfer](<https://attack.mitre.org/techniques/T1570>).

Name

Data Encoding

ID

T1132

Description

Adversaries may encode data to make the content of command and control traffic more difficult to detect. Command and control (C2) information can be encoded using a standard data encoding system. Use of data encoding may adhere to existing protocol specifications and includes use of ASCII, Unicode, Base64, MIME, or other binary-to-text and character encoding systems.(Citation: Wikipedia Binary-to-text Encoding) (Citation: Wikipedia Character Encoding) Some data encoding systems may also result in data compression, such as gzip.

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

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

Hijack Execution Flow

ID

T1574

Description

Adversaries may execute their own malicious payloads by hijacking the way operating systems run programs. Hijacking execution flow can be for the purposes of persistence, since this hijacked execution may reoccur over time. Adversaries may also use these mechanisms to elevate privileges or evade defenses, such as application control or other restrictions on execution. There are many ways an adversary may hijack the flow of execution, including by manipulating how the operating system locates programs to be executed. How the operating system locates libraries to be used by a program can also be intercepted. Locations where the operating system looks for programs/resources, such as file directories and in the case of Windows the Registry, could also be poisoned to include malicious payloads.

Name

Ingress Tool Transfer

ID

T1105

Description

Adversaries may transfer tools or other files from an external system into a compromised environment. Tools or files may be copied from an external adversary-controlled system to the victim network through the command and control channel or through alternate protocols such as [ftp](https://attack.mitre.org/software/S0095). Once present, adversaries may also transfer/spread tools between victim devices within a compromised environment (i.e. [Lateral Tool Transfer](https://attack.mitre.org/techniques/T1570)). Files can also be transferred using various [Web Service](https://attack.mitre.org/techniques/T1102)s as well as native or otherwise present tools on the victim system.(Citation: PTSecurity Cobalt Dec 2016) On Windows, adversaries may use various utilities to download tools, such as `copy`, `finger`, [certutil](https://attack.mitre.org/software/S0160), and [PowerShell](https://attack.mitre.org/techniques/T1059/001) commands such as `EX(New-Object Net.WebClient).downloadString()` and `Invoke-WebRequest`. On Linux and macOS systems, a variety of utilities also exist, such as `curl`, `scp`, `sftp`, `tftp`, `rsync`, `finger`, and `wget`. (Citation: t1105_lolbas)

Name

Supply Chain Compromise

ID

T1195

Description

Adversaries may manipulate products or product delivery mechanisms prior to receipt by a final consumer for the purpose of data or system compromise. Supply chain compromise can take place at any stage of the supply chain including: * Manipulation of development tools * Manipulation of a development environment * Manipulation of source code repositories (public or private) * Manipulation of source code in open-source dependencies * Manipulation of software update/distribution mechanisms * Compromised/infected system images (multiple cases of removable media infected at the factory)(Citation: IBM Storwize)(Citation: Schneider Electric USB Malware) * Replacement of legitimate software with modified versions * Sales of modified/counterfeit products to legitimate distributors * Shipment interdiction While supply chain compromise can impact

any component of hardware or software, adversaries looking to gain execution have often focused on malicious additions to legitimate software in software distribution or update channels.(Citation: Avast CCleaner3 2018)(Citation: Microsoft Dofail 2018)(Citation: Command Five SK 2011) Targeting may be specific to a desired victim set or malicious software may be distributed to a broad set of consumers but only move on to additional tactics on specific victims.(Citation: Symantec Elderwood Sept 2012)(Citation: Avast CCleaner3 2018)(Citation: Command Five SK 2011) Popular open source projects that are used as dependencies in many applications may also be targeted as a means to add malicious code to users of the dependency.(Citation: Trendmicro NPM Compromise)

Name

Drive-by Compromise

ID

T1189

Description

Adversaries may gain access to a system through a user visiting a website over the normal course of browsing. With this technique, the user's web browser is typically targeted for exploitation, but adversaries may also use compromised websites for non-exploitation behavior such as acquiring [Application Access Token](<https://attack.mitre.org/techniques/T1550/001>). Multiple ways of delivering exploit code to a browser exist (i.e., [Drive-by Target](<https://attack.mitre.org/techniques/T1608/004>)), including: * A legitimate website is compromised where adversaries have injected some form of malicious code such as JavaScript, iFrames, and cross-site scripting * Script files served to a legitimate website from a publicly writeable cloud storage bucket are modified by an adversary * Malicious ads are paid for and served through legitimate ad providers (i.e., [Malvertising](<https://attack.mitre.org/techniques/T1583/008>)) * Built-in web application interfaces are leveraged for the insertion of any other kind of object that can be used to display web content or contain a script that executes on the visiting client (e.g. forum posts, comments, and other user controllable web content). Often the website used by an adversary is one visited by a specific community, such as government, a particular industry, or region, where the goal is to compromise a specific user or set of users based on a shared interest. This kind of targeted campaign is often referred to a strategic web compromise or watering hole attack. There are several known examples of this occurring.(Citation: Shadowserver Strategic Web Compromise) Typical drive-by compromise process: 1. A user visits a website that is used to host the adversary controlled content. 2. Scripts automatically execute, typically searching versions of the browser and plugins for a potentially vulnerable

version. * The user may be required to assist in this process by enabling scripting or active website components and ignoring warning dialog boxes. 3. Upon finding a vulnerable version, exploit code is delivered to the browser. 4. If exploitation is successful, then it will give the adversary code execution on the user's system unless other protections are in place. * In some cases a second visit to the website after the initial scan is required before exploit code is delivered. Unlike [Exploit Public-Facing Application](<https://attack.mitre.org/techniques/T1190>), the focus of this technique is to exploit software on a client endpoint upon visiting a website. This will commonly give an adversary access to systems on the internal network instead of external systems that may be in a DMZ. Adversaries may also use compromised websites to deliver a user to a malicious application designed to [Steal Application Access Token](<https://attack.mitre.org/techniques/T1528>), like OAuth tokens, to gain access to protected applications and information. These malicious applications have been delivered through popups on legitimate websites.(Citation: Volexity OceanLotus Nov 2017)

Name

Web Service

ID

T1102

Description

Adversaries may use an existing, legitimate external Web service as a means for relaying data to/from a compromised system. Popular websites and social media acting as a mechanism for C2 may give a significant amount of cover due to the likelihood that hosts within a network are already communicating with them prior to a compromise. Using common services, such as those offered by Google or Twitter, makes it easier for adversaries to hide in expected noise. Web service providers commonly use SSL/TLS encryption, giving adversaries an added level of protection. Use of Web services may also protect back-end C2 infrastructure from discovery through malware binary analysis while also enabling operational resiliency (since this infrastructure may be dynamically changed).

Name

Remote Services

ID

T1021

Description

Adversaries may use [Valid Accounts](<https://attack.mitre.org/techniques/T1078>) to log into a service that accepts remote connections, such as telnet, SSH, and VNC. The adversary may then perform actions as the logged-on user. In an enterprise environment, servers and workstations can be organized into domains. Domains provide centralized identity management, allowing users to login using one set of credentials across the entire network. If an adversary is able to obtain a set of valid domain credentials, they could login to many different machines using remote access protocols such as secure shell (SSH) or remote desktop protocol (RDP). (Citation: SSH Secure Shell) (Citation: TechNet Remote Desktop Services) They could also login to accessible SaaS or IaaS services, such as those that federate their identities to the domain. Legitimate applications (such as [Software Deployment Tools](<https://attack.mitre.org/techniques/T1072>) and other administrative programs) may utilize [Remote Services](<https://attack.mitre.org/techniques/T1021>) to access remote hosts. For example, Apple Remote Desktop (ARD) on macOS is native software used for remote management. ARD leverages a blend of protocols, including [VNC](<https://attack.mitre.org/techniques/T1021/005>) to send the screen and control buffers and [SSH](<https://attack.mitre.org/techniques/T1021/004>) for secure file transfer. (Citation: Remote Management MDM macOS) (Citation: Kickstart Apple Remote Desktop commands) (Citation: Apple Remote Desktop Admin Guide 3.3) Adversaries can abuse applications such as ARD to gain remote code execution and perform lateral movement. In versions of macOS prior to 10.14, an adversary can escalate an SSH session to an ARD session which enables an adversary to accept TCC (Transparency, Consent, and Control) prompts without user interaction and gain access to data. (Citation: FireEye 2019 Apple Remote Desktop) (Citation: Lockboxx ARD 2019) (Citation: Kickstart Apple Remote Desktop commands)

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

Deobfuscate/Decode Files or Information

ID

T1140

Description

Adversaries may use [Obfuscated Files or Information](<https://attack.mitre.org/techniques/T1027>) to hide artifacts of an intrusion from analysis. They may require separate mechanisms to decode or deobfuscate that information depending on how they intend to use it. Methods for doing that include built-in functionality of malware or by using utilities present on the system. One such example is the use of [certutil](<https://attack.mitre.org/software/S0160>) to decode a remote access tool portable executable file that has been hidden inside a certificate file.(Citation: Malwarebytes Targeted Attack against Saudi Arabia) Another example is using the Windows `copy /b`` command to reassemble binary fragments into a malicious payload.(Citation: Carbon Black Obfuscation Sept 2016) Sometimes a user's action may be required to open it for deobfuscation or decryption as part of [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)

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 `nvr`). (Citation: US-CERT-TA18-106A)

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

Exploitation for Client Execution

ID

T1203

Description

Adversaries may exploit software vulnerabilities in client applications to execute code. Vulnerabilities can exist in software due to unsecure coding practices that can lead to unanticipated behavior. Adversaries can take advantage of certain vulnerabilities through targeted exploitation for the purpose of arbitrary code execution. Oftentimes the most valuable exploits to an offensive toolkit are those that can be used to obtain code execution on a remote system because they can be used to gain access to that system. Users will expect to see files related to the applications they commonly used to do work, so they are a useful target for exploit research and development because of their high utility. Several types exist: ### Browser-based Exploitation Web browsers are a common target through [Drive-by Compromise](https://attack.mitre.org/techniques/T1189) and [Spearphishing Link](https://attack.mitre.org/techniques/T1566/002). Endpoint systems may be compromised through normal web browsing or from certain users being targeted by links in spearphishing emails to adversary controlled sites used to exploit the web browser. These often do not require an action by the user for the exploit to be executed. ### Office Applications Common office and productivity applications such as Microsoft Office are also targeted through [Phishing](https://attack.mitre.org/techniques/T1566). Malicious files will be transmitted directly as attachments or through links to download them. These require the user to open the document or file for the exploit to run. ### Common Third-party Applications Other applications that are commonly seen or are part of the software deployed in a target network may also be used for exploitation. Applications such as Adobe Reader and Flash, which are common in enterprise environments, have been routinely targeted by adversaries attempting to gain access to systems. Depending on the software and nature of the vulnerability, some may be exploited in the browser or require the user to open a file. For instance, some Flash exploits have been delivered as objects within Microsoft Office documents.

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

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.

Sector

Name

Defense

Description

Public and private entities involved in the conception and production of weapons and the planning and conducting of military operations.

Name

Technologies

Description

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

Name

Nuclear power (civilian use)

Description

Public and private entities involved in the extraction and transport of nuclear fuel, operating nuclear plants for civilian purposes and entities processing nuclear waste.

Intrusion-Set

Name

Lazarus

Url

Value

<http://www.vietjetairkorea.com/INFO/info.asp>

<https://www.blastedlevels.com/levels4SqR8/measure.asp>

<https://www.hanlasangjo.com/editor/pages/page.asp>

<https://little-pet.com/web/board/skin/default/read.php>

<https://safemotors.co.kr/daumeditor/pages/template/template.asp>

https://mainbiz.or.kr/SmartEditor2/photo_uploader/popup/edit.asp

https://pediatrics.or.kr/PubReader/build_css.php

<https://www.medric.or.kr/Controls/Board/certificate.cer>

<http://www.friendmc.com/upload/board/asp20062107.asp>

<http://yoohannet.kr/min/tmp/process/proc.php>

<https://www.droof.kr/Board/htmlEdit/PopupWin/Editor.asp>

<http://www.khmcpharm.com/Lib/Modules/HtmlEditor/Util/read.cer>

https://www.healthpro.or.kr/upload/naver_editor/subview/view.inc

<https://swt-keystonevalve.com/data/editor/index.php>

<https://vnfmal2022.com/niabbs5/upload/gongji/index.php>

<https://www.friendmc.com:80/upload/board/asp20062107.asp>

<http://www.hankooktop.com/ko/company/info.asp>

http://ictm.or.kr/UPLOAD_file/board/free/edit/index.php

<https://admin.esangedu.kr/XPaySample/submit.php>

https://kscmfs.or.kr/member/handle/log_proc.php

https://pms.nninc.co.kr/app/content/board/inc_list.asp

<https://www.happinesscc.com/mobile/include/func.asp>

<http://samwoosystem.co.kr/board/list/write.asp>

<https://www.nonstopexpress.com/community/include/index.asp>

<https://yoohannet.kr/min/tmp/process/proc.php>

https://api.shw.kr/login_admin/member/login_fail.php

<http://theorigin.co.kr:443/admin/management/index.php>

<https://mainbiz.or.kr/include/common.asp>

https://www.seouldementia.or.kr/_manage/inc/bbs/jiyek1_ok.asp

https://warevalley.com/en/common/include/page_tab.asp

https://www.seoulanesthesia.or.kr/mail/mail_211230.html

https://www.siriuskorea.co.kr/mall/community/bbs_read.asp

https://kstr.radiology.or.kr/upload/schedule/29431_1687715624.inc

<https://www.muijae.com/daumeditor/pages/template/template.asp>

<https://www.muijae.com/daumeditor/pages/template/simple.asp>

<http://ucware.net/skins/PHPMailer-master/index.php>

<https://new-q-cells.com/upload/newsletter/cn/frame.php>

<https://hicar.kalo.kr/data/rental/Coupon/include/inc.asp>

https://hspje.com:80/menu6/teacher_qna.asp

External References

-
- <https://otx.alienvault.com/pulse/653c0681ae38ba0d7d84e538>
-
- <https://securelist.com/unveiling-lazarus-new-campaign/110888/>