Sep 01 2023

NETMANAGEIT

Intelligence Report

Earth Estries Targets Government, Tech for Cyberespionage

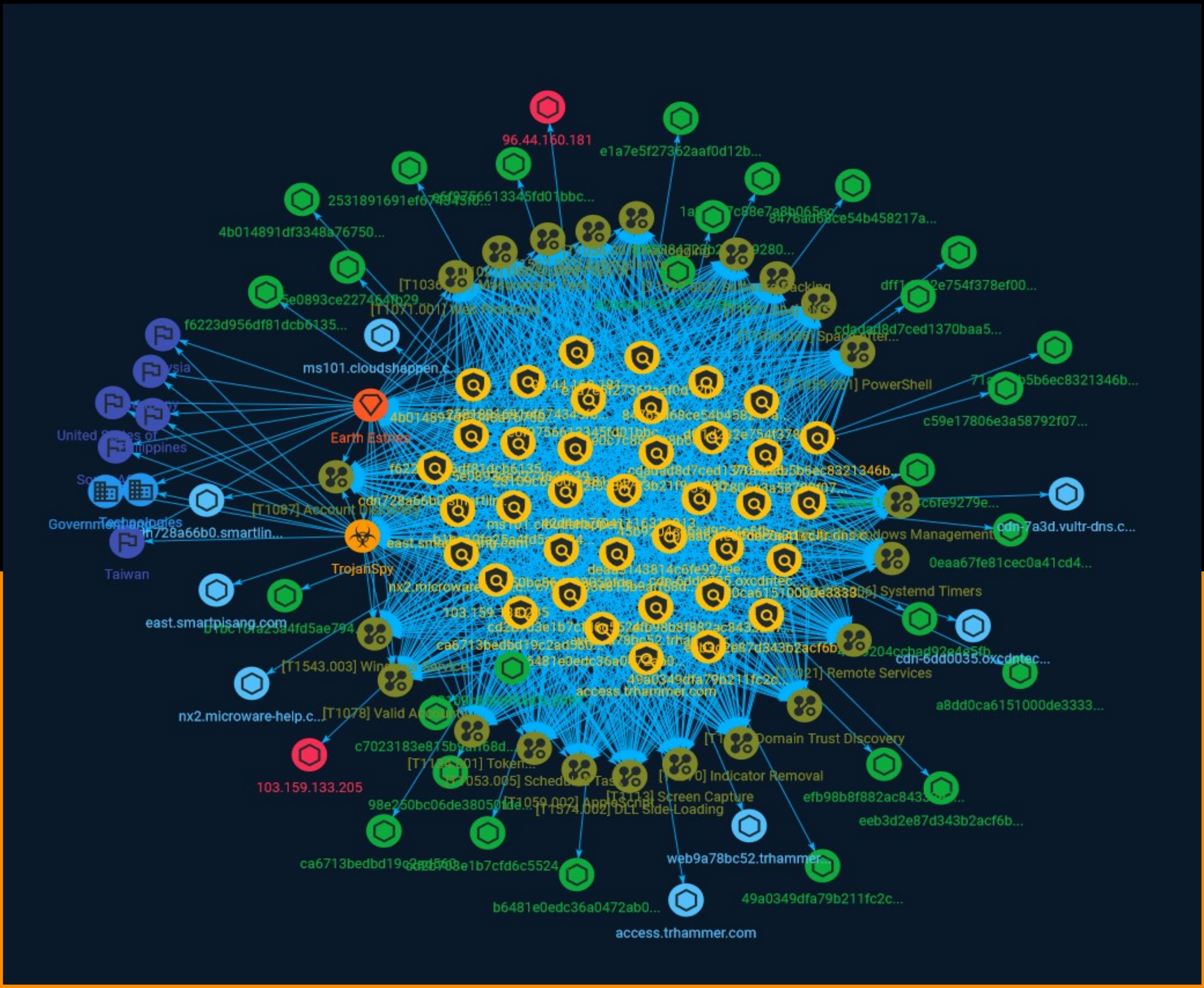

Table of contents

Overview

Entities

Observables

External References

Overview

Description

Earth Estries is working with high-level resources and functioning with sophisticated skills and experience in cyberespionage and illicit activities. The threat actors also use multiple backdoors and hacking tools to enhance intrusion vectors. To leave as little footprint as possible, they use PowerShell downgrade attacks to avoid detection from Windows Antimalware Scan Interface's (AMSI) logging mechanism. In addition, the actors abuse public services such as Github, Gmail, AnonFiles, and File.io to exchange or transfer commands and stolen data.

Confidence

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

15 / 100

Attack-Pattern

Name

Systemd Timers

ID

T1053.006

Description

Adversaries may abuse systemd timers to perform task scheduling for initial or recurring execution of malicious code. Systemd timers are unit files with file extension `.timer` that control services. Timers can be set to run on a calendar event or after a time span relative to a starting point. They can be used as an alternative to [Cron](https://attack.mitre.org/techniques/T1053/003) in Linux environments.(Citation: archlinux Systemd Timers Aug 2020) Systemd timers may be activated remotely via the `systemctl` command line utility, which operates over [SSH](https://attack.mitre.org/techniques/T1021/004).(Citation: Systemd Remote Control) Each `.timer` file must have a corresponding `.service` file with the same name, e.g., `example.timer` and `example.service`. `.service` files are [Systemd Service](https://attack.mitre.org/techniques/T1543/002) unit files that are managed by the systemd system and service manager.(Citation: Linux man-pages: systemd January 2014) Privileged timers are written to `/etc/systemd/system/` and `/usr/lib/systemd/system` while user level are written to `~/.config/systemd/user/`. An adversary may use systemd timers to execute malicious code at system startup or on a scheduled basis for persistence.(Citation: Arch Linux Package Systemd Compromise BleepingComputer 10JUL2018)(Citation: gist Arch package compromise 10JUL2018)(Citation: acroread package compromised Arch Linux Mail 8JUL2018) Timers installed using privileged paths may be used to maintain root level persistence. Adversaries may also install user level timers to achieve user level persistence.

Name
Space after Filename
ID
T1036.006
Description
Adversaries can hide a program's true filetype by changing the extension of a file. With certain file types (specifically this does not work with .app extensions), appending a space to the end of a filename will change how the file is processed by the operating system. For example, if there is a Mach-O executable file called `evil.bin`, when it is double clicked by a user, it will launch Terminal.app and execute. If this file is renamed to `evil.txt`, then when double clicked by a user, it will launch with the default text editing application (not executing the binary). However, if the file is renamed to `evil.txt ` (note the space at the end), then when double clicked by a user, the true file type is determined by the OS and handled appropriately and the binary will be executed (Citation: Mac Backdoors are back). Adversaries can use this feature to trick users into double clicking benign-looking files of any format and ultimately executing something malicious.
Name
AppleScript
ID
T1059.002
Description
Adversaries may abuse AppleScript for execution. AppleScript is a macOS scripting language designed to control applications and parts of the OS via inter-application messages called AppleEvents.(Citation: Apple AppleScript) These AppleEvent messages can be sent independently or easily scripted with AppleScript. These events can locate open windows, send keystrokes, and interact with almost any open application locally or remotely. Scripts can be run from the command-line via `osascript /path/to/script` or

`osascript -e "script here"`. Aside from the command line, scripts can be executed in numerous ways including Mail rules, Calendar.app alarms, and Automator workflows. AppleScripts can also be executed as plain text shell scripts by adding `#!/usr/bin/osascript` to the start of the script file.(Citation: SentinelOne AppleScript) AppleScripts do not need to call `osascript` to execute. However, they may be executed from within mach-O binaries by using the macOS [Native API](https://attack.mitre.org/techniques/T1106)s `NSAppleScript` or `OSAScript`, both of which execute code independent of the `/usr/bin/osascript` command line utility. Adversaries may abuse AppleScript to execute various behaviors, such as interacting with an open SSH connection, moving to remote machines, and even presenting users with fake dialog boxes. These events cannot start applications remotely (they can start them locally), but they can interact with applications if they're already running remotely. On macOS 10.10 Yosemite and higher, AppleScript has the ability to execute [Native API](https://attack.mitre.org/techniques/T1106)s, which otherwise would require compilation and execution in a mach-O binary file format.(Citation: SentinelOne macOS Red Team) Since this is a scripting language, it can be used to launch more common techniques as well such as a reverse shell via [Python](https://attack.mitre.org/techniques/T1059/006).(Citation: Macro Malware Targets Macs)

Name

Token Impersonation/Theft

ID

T1134.001

Description

Adversaries may duplicate then impersonate another user's existing token to escalate privileges and bypass access controls. For example, an adversary can duplicate an existing token using `DuplicateToken` or `DuplicateTokenEx`. The token can then be used with `ImpersonateLoggedOnUser` to allow the calling thread to impersonate a logged on user's security context, or with `SetThreadToken` to assign the impersonated token to a thread. An adversary may perform [Token Impersonation/Theft](https://attack.mitre.org/techniques/T1134/001) when they have a specific, existing process they want to assign the duplicated token to. For example, this may be useful for when the target user has a non-network logon session on the system. When an adversary would instead use a duplicated token to create a new process rather than attaching to an existing process, they can additionally [Create Process with Token](https://attack.mitre.org/techniques/T1134/002) using `CreateProcessWithTokenW` or `CreateProcessAsUserW`. [Token Impersonation/Theft](https://attack.mitre.org/techniques/T1134/001) is also distinct from [Make and

Impersonate Token](https://attack.mitre.org/techniques/T1134/003) in that it refers to duplicating an existing token, rather than creating a new one.

Name

Exfiltration to Cloud Storage

ID

T1567.002

Description

Adversaries may exfiltrate data to a cloud storage service rather than over their primary command and control channel. Cloud storage services allow for the storage, edit, and retrieval of data from a remote cloud storage server over the Internet. Examples of cloud storage services include Dropbox and Google Docs. Exfiltration to these cloud storage services can provide a significant amount of cover to the adversary if hosts within the network are already communicating with the service.

Name

Domain Trust Discovery

ID

T1482

Description

Adversaries may attempt to gather information on domain trust relationships that may be used to identify lateral movement opportunities in Windows multi-domain/forest environments. Domain trusts provide a mechanism for a domain to allow access to resources based on the authentication procedures of another domain.(Citation: Microsoft Trusts) Domain trusts allow the users of the trusted domain to access resources in the trusting domain. The information discovered may help the adversary conduct [SID-History Injection](https://attack.mitre.org/techniques/T1134/005), [Pass the Ticket](https://attack.mitre.org/techniques/T1550/003), and [Kerberoasting](https://attack.mitre.org/

techniques/T1558/003).(Citation: AdSecurity Forging Trust Tickets)(Citation: Harmj0y Domain Trusts) Domain trusts can be enumerated using the `DSEnumerateDomainTrusts()` Win32 API call, .NET methods, and LDAP.(Citation: Harmj0y Domain Trusts) The Windows utility [Nltest](https://attack.mitre.org/software/S0359) is known to be used by adversaries to enumerate domain trusts.(Citation: Microsoft Operation Wilysupply)
Name
Account Discovery
ID
T1087
Description
Adversaries may attempt to get a listing of valid accounts, usernames, or email addresses on a system or within a compromised environment. This information can help adversaries determine which accounts exist, which can aid in follow-on behavior such as brute-forcing, spear-phishing attacks, or account takeovers (e.g., [Valid Accounts](https://attack.mitre.org/techniques/T1078)). Adversaries may use several methods to enumerate accounts, including abuse of existing tools, built-in commands, and potential misconfigurations that leak account names and roles or permissions in the targeted environment. For examples, cloud environments typically provide easily accessible interfaces to obtain user lists. On hosts, adversaries can use default [PowerShell](https://attack.mitre.org/techniques/T1059/001) and other command line functionality to identify accounts. Information about email addresses and accounts may also be extracted by searching an infected system's files.
Name
DNS
ID
T1071.004
Description

Adversaries may communicate using the Domain Name System (DNS) application layer protocol 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. The DNS protocol serves an administrative function in computer networking and thus may be very common in environments. DNS traffic may also be allowed even before network authentication is completed. DNS packets contain many fields and headers in which data can be concealed. Often known as DNS tunneling, adversaries may abuse DNS to communicate with systems under their control within a victim network while also mimicking normal, expected traffic.(Citation: PAN DNS Tunneling)(Citation: Medium DnsTunneling)

Name

DLL Side-Loading

ID

T1574.002

Description

Adversaries may execute their own malicious payloads by side-loading DLLs. Similar to [DLL Search Order Hijacking](https://attack.mitre.org/techniques/T1574/001), side-loading involves hijacking which DLL a program loads. But rather than just planting the DLL within the search order of a program then waiting for the victim application to be invoked, adversaries may directly side-load their payloads by planting then invoking a legitimate application that executes their payload(s). Side-loading takes advantage of the DLL search order used by the loader by positioning both the victim application and malicious payload(s) alongside each other. Adversaries likely use side-loading as a means of masking actions they perform under a legitimate, trusted, and potentially elevated system or software process. Benign executables used to side-load payloads may not be flagged during delivery and/or execution. Adversary payloads may also be encrypted/packed or otherwise obfuscated until loaded into the memory of the trusted process.(Citation: FireEye DLL Side-Loading)

Name

Masquerade Task or Service

ID

T1036.004

Description

Adversaries may attempt to manipulate the name of a task or service to make it appear legitimate or benign. Tasks/services executed by the Task Scheduler or systemd will typically be given a name and/or description.(Citation: TechNet Schtasks)(Citation: Systemd Service Units) Windows services will have a service name as well as a display name. Many benign tasks and services exist that have commonly associated names. Adversaries may give tasks or services names that are similar or identical to those of legitimate ones. Tasks or services contain other fields, such as a description, that adversaries may attempt to make appear legitimate.(Citation: Palo Alto Shamoon Nov 2016)(Citation: Fysbis Dr Web Analysis)

Name

Windows Management Instrumentation

ID

T1047

Description

Adversaries may abuse Windows Management Instrumentation (WMI) to execute malicious commands and payloads. WMI is an administration feature that provides a uniform environment to access Windows system components. The WMI service enables both local and remote access, though the latter is facilitated by [Remote Services](https://attack.mitre.org/techniques/T1021) such as [Distributed Component Object Model](https://attack.mitre.org/techniques/T1021/003) (DCOM) and [Windows Remote Management](https://attack.mitre.org/techniques/T1021/006) (WinRM).(Citation: MSDN WMI) Remote WMI over DCOM operates using port 135, whereas WMI over WinRM operates over port 5985 when using HTTP and 5986 for HTTPS.(Citation: MSDN WMI)(Citation: FireEye WMI 2015) An adversary can use WMI to interact with local and remote systems and use it as a means to execute various behaviors, such as gathering information for Discovery as well as remote

Execution of files as part of Lateral Movement. (Citation: FireEye WMI SANS 2015) (Citation: FireEye WMI 2015)

Name

Valid Accounts

ID

T1078

Description

Adversaries may obtain and abuse credentials of existing accounts as a means of gaining Initial Access, Persistence, Privilege Escalation, or Defense Evasion. Compromised credentials may be used to bypass access controls placed on various resources on systems within the network and may even be used for persistent access to remote systems and externally available services, such as VPNs, Outlook Web Access, network devices, and remote desktop.(Citation: volexity_0day_sophos_FW) Compromised credentials may also grant an adversary increased privilege to specific systems or access to restricted areas of the network. Adversaries may choose not to use malware or tools in conjunction with the legitimate access those credentials provide to make it harder to detect their presence. In some cases, adversaries may abuse inactive accounts: for example, those belonging to individuals who are no longer part of an organization. Using these accounts may allow the adversary to evade detection, as the original account user will not be present to identify any anomalous activity taking place on their account.(Citation: CISA MFA PrintNightmare) The overlap of permissions for local, domain, and cloud accounts across a network of systems is of concern because the adversary may be able to pivot across accounts and systems to reach a high level of access (i.e., domain or enterprise administrator) to bypass access controls set within the enterprise.(Citation: TechNet Credential Theft)

Name

Keylogging

ID

T1056.001

Description

Adversaries may log user keystrokes to intercept credentials as the user types them. Keylogging is likely to be used to acquire credentials for new access opportunities when [OS Credential Dumping](https://attack.mitre.org/techniques/T1003) efforts are not effective, and may require an adversary to intercept keystrokes on a system for a substantial period of time before credentials can be successfully captured. Keylogging is the most prevalent type of input capture, with many different ways of intercepting keystrokes.(Citation: Adventures of a Keystroke) Some methods include: * Hooking API callbacks used for processing keystrokes. Unlike [Credential API Hooking](https://attack.mitre.org/techniques/T1056/004), this focuses solely on API functions intended for processing keystroke data. * Reading raw keystroke data from the hardware buffer. * Windows Registry modifications. * Custom drivers. * [Modify System Image](https://attack.mitre.org/techniques/T1601) may provide adversaries with hooks into the operating system of network devices to read raw keystrokes for login sessions.(Citation: Cisco Blog Legacy Device Attacks)

Name

Indicator Removal

ID

T1070

Description

Adversaries may delete or modify artifacts generated within systems to remove evidence of their presence or hinder defenses. Various artifacts may be created by an adversary or something that can be attributed to an adversary’s actions. Typically these artifacts are used as defensive indicators related to monitored events, such as strings from downloaded files, logs that are generated from user actions, and other data analyzed by defenders. Location, format, and type of artifact (such as command or login history) are often specific to each platform. Removal of these indicators may interfere with event collection, reporting, or other processes used to detect intrusion activity. This may compromise the integrity of security solutions by causing notable events to go unreported. This activity may also impede forensic analysis and incident response, due to lack of sufficient data to determine what occurred.

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

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

Scheduled Task

ID

T1053.005

Description

Adversaries may abuse the Windows Task Scheduler to perform task scheduling for initial or recurring execution of malicious code. There are multiple ways to access the Task Scheduler in Windows. The [schtasks](https://attack.mitre.org/software/S0111) utility can be run directly on the command line, or the Task Scheduler can be opened through the GUI within the Administrator Tools section of the Control Panel. In some cases, adversaries have used a .NET wrapper for the Windows Task Scheduler, and alternatively, adversaries have used the Windows netapi32 library to create a scheduled task. The deprecated [at](https://attack.mitre.org/software/S0110) utility could also be abused by adversaries (ex: [At](https://attack.mitre.org/techniques/T1053/002)), though `at.exe` can not access tasks created with `schtasks` or the Control Panel. An adversary may use Windows Task Scheduler to execute programs at system startup or on a scheduled basis for persistence. The Windows Task Scheduler can also be abused to conduct remote Execution as part of Lateral Movement and/or to run a process under the context of a specified account (such as SYSTEM). Similar to [System Binary Proxy Execution](https://attack.mitre.org/techniques/T1218), adversaries have also abused the Windows Task Scheduler to potentially mask one-time execution under signed/trusted system processes.(Citation: ProofPoint Serpent) Adversaries may also create "hidden" scheduled tasks (i.e. [Hide Artifacts](https://attack.mitre.org/techniques/T1564)) that may not be visible to defender tools and manual queries used to enumerate tasks. Specifically, an adversary may hide a task from `schtasks /query` and the Task Scheduler by deleting the associated Security Descriptor (SD) registry value (where deletion of this value must be completed using

SYSTEM permissions).(Citation: SigmaHQ)(Citation: Tarrask scheduled task) Adversaries may also employ alternate methods to hide tasks, such as altering the metadata (e.g., `Index` value) within associated registry keys.(Citation: Defending Against Scheduled Task Attacks in Windows Environments)

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

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

Windows Service

ID

T1543.003

Description

Adversaries may create or modify Windows services to repeatedly execute malicious payloads as part of persistence. When Windows boots up, it starts programs or applications called services that perform background system functions.(Citation: TechNet Services) Windows service configuration information, including the file path to the service's executable or recovery programs/commands, is stored in the Windows Registry. Adversaries may install a new service or modify an existing service to execute at startup in order to persist on a system. Service configurations can be set or modified using system utilities (such as sc.exe), by directly modifying the Registry, or by interacting directly with the Windows API. Adversaries may also use services to install and execute malicious drivers. For example, after dropping a driver file (ex: `.sys`) to disk, the payload can be loaded and registered via [Native API](https://attack.mitre.org/techniques/T1106) functions such as `CreateServiceW()` (or manually via functions such as `ZwLoadDriver()` and `ZwSetValueKey()`), by creating the required service Registry values (i.e. [Modify Registry](https://attack.mitre.org/techniques/T1112)), or by using command-line utilities such as `PnPUtil.exe`.(Citation: Symantec W.32 Stuxnet Dossier)(Citation: Crowdstrike DriveSlayer February 2022)(Citation: Unit42 AcidBox June 2020) Adversaries may leverage these drivers as [Rootkit](https://attack.mitre.org/techniques/T1014)s to hide the presence of malicious activity on a system. Adversaries may also load a signed yet vulnerable driver onto a compromised machine (known as "Bring Your Own Vulnerable Driver" (BYOVD)) as part of [Exploitation for Privilege Escalation](https://attack.mitre.org/techniques/T1068).(Citation: ESET InvisiMole June 2020)(Citation: Unit42 AcidBox June 2020) Services may be created with administrator privileges but are executed under SYSTEM privileges, so an adversary may also use a service to escalate privileges. Adversaries may also directly start services through [Service Execution](https://attack.mitre.org/techniques/T1569/002). To make detection analysis more challenging, malicious services may also incorporate [Masquerade Task or Service](https://attack.mitre.org/techniques/T1036/004) (ex: using a service and/or payload name related to a legitimate OS or benign software component).

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

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)

Indicator

Name

42d4eb7f04111631891379c5cce55480d2d9d2ef8feaf1075e1aed0c52df4bb9

Pattern Type

stix

Pattern

[file:hashes.'SHA-256' =
'42d4eb7f04111631891379c5cce55480d2d9d2ef8feaf1075e1aed0c52df4bb9']

Name

1a9e0c7c88e7a8b065ec88809187f67d920e7845350d94098645e592ec5534f6

Pattern Type

stix

Pattern

[file:hashes.'SHA-256' =
'1a9e0c7c88e7a8b065ec88809187f67d920e7845350d94098645e592ec5534f6']

Name

deaa3143814c6fe9279e8bc0706df22d63ef197af980d8feae9a8468f441efec

Pattern Type

stix

Pattern

[file:hashes.'SHA-256' =
'deaa3143814c6fe9279e8bc0706df22d63ef197af980d8feae9a8468f441efec']

Name

b1bc10fa25a4fd5ae7948c6523eb975be8d0f52d1572c57a7ef736134b996586

Pattern Type

stix

Pattern

[file:hashes.'SHA-256' =
'b1bc10fa25a4fd5ae7948c6523eb975be8d0f52d1572c57a7ef736134b996586']

Name

82f3384723b21f9a928029bb3ee116f9adbc4f7ec66d5a856e817c3dc16d149d

Pattern Type

stix

Pattern

[file:hashes.'SHA-256' = '82f3384723b21f9a928029bb3ee116f9adbc4f7ec66d5a856e817c3dc16d149d']

Name

c7023183e815b9aff68d3eba6c2ca105dbe0a9b05cd209908dcee907a64ce80b

Pattern Type

stix

Pattern

[file:hashes.'SHA-256' = 'c7023183e815b9aff68d3eba6c2ca105dbe0a9b05cd209908dcee907a64ce80b']

Name

ms101.cloudshappen.com

Pattern Type

stix

Pattern

[hostname:value = 'ms101.cloudshappen.com']

Name

cdn-6dd0035.oxcdntech.com

Pattern Type

stix

Pattern

[hostname:value = 'cdn-6dd0035.oxcdntech.com']

Name

98e250bc06de38050fdeab9b1e2ef7e4d8c401b33fd5478f3b85197112858f4e

Pattern Type

stix

Pattern

[file:hashes.'SHA-256' = '98e250bc06de38050fdeab9b1e2ef7e4d8c401b33fd5478f3b85197112858f4e']

Name

8476ad68ce54b458217ab165d66a899d764eae3ad30196f35d2ff20d3f398523

Pattern Type

stix

Pattern

[file:hashes.'SHA-256' = '8476ad68ce54b458217ab165d66a899d764eae3ad30196f35d2ff20d3f398523']

Name

access.trhammer.com

Pattern Type

stix

Pattern

[hostname:value = 'access.trhammer.com']

Name

415e0893ce227464fb29d76e0500c518935d11379d17fb14effaef82e962ff76

Pattern Type

stix

Pattern

[file:hashes.'SHA-256' = '415e0893ce227464fb29d76e0500c518935d11379d17fb14effaef82e962ff76']

Name

45b9204ccbad92e4e5fb9e31aab683eb5221eb5f5688b1aae98d9c0f1c920227

Pattern Type

stix

Pattern

[file:hashes.'SHA-256' = '45b9204ccbad92e4e5fb9e31aab683eb5221eb5f5688b1aae98d9c0f1c920227']

Name

web9a78bc52.trhammer.com

Pattern Type

stix

Pattern

[hostname:value = 'web9a78bc52.trhammer.com']

Name

e6f9756613345fd01bbcf28eba15d52705ef4d144c275b8cfe868a5d28c24140

Pattern Type

stix

Pattern

[file:hashes.'SHA-256' = 'e6f9756613345fd01bbcf28eba15d52705ef4d144c275b8cfe868a5d28c24140']

Name

a8dd0ca6151000de33335f48a832d24412de13ce05ea6f279bf4aaaa2e5aaecb

Pattern Type

stix

Pattern

[file:hashes.'SHA-256' = 'a8dd0ca6151000de33335f48a832d24412de13ce05ea6f279bf4aaaa2e5aaecb']

Name

east.smartpisang.com

Pattern Type

stix

Pattern

[hostname:value = 'east.smartpisang.com']

Name

cdadad8d7ced1370baa5d1ffe435bed78c2d58ed4cda364b8a7484e3c7cdac98

Pattern Type

stix

Pattern

[file:hashes.'SHA-256' =
'cdadad8d7ced1370baa5d1ffe435bed78c2d58ed4cda364b8a7484e3c7cdac98']

Name

b6481e0edc36a0472ab0ce7d0817f1773c4af9307ae60890a667930558a762ff

Pattern Type

stix

Pattern

[file:hashes.'SHA-256' =
'b6481e0edc36a0472ab0ce7d0817f1773c4af9307ae60890a667930558a762ff']

Name
2531891691ef674345f098ef18b274091acdf3f2808cca753674599c043ccd7d

Pattern Type
stix

Pattern
[file:hashes.'SHA-256' = '2531891691ef674345f098ef18b274091acdf3f2808cca753674599c043ccd7d']

Name
0eaa67fe81cec0a41cd42866df1223cb7d2b5659ab295dffe64fe9c3b76720aa

Pattern Type
stix

Pattern
[file:hashes.'SHA-256' = '0eaa67fe81cec0a41cd42866df1223cb7d2b5659ab295dffe64fe9c3b76720aa']

Name
cd2b703e1b7cfd6c552406f44ec05480209003789ad4fbba4d4cffd4f104b0a0

Pattern Type
stix

Pattern

[file:hashes.'SHA-256' = 'cd2b703e1b7cfd6c552406f44ec05480209003789ad4fbba4d4cffd4f104b0a0']

Name

e1a7e5f27362aaf0d12b58b96a816ef61a2a498def9805297aa81f6f83729230

Pattern Type

stix

Pattern

[file:hashes.'SHA-256' = 'e1a7e5f27362aaf0d12b58b96a816ef61a2a498def9805297aa81f6f83729230']

Name

efb98b8f882ac84332e7dfdc996a081d1c5e6189ad726f8f8afec5d36a20a730

Pattern Type

stix

Pattern

[file:hashes.'SHA-256' = 'efb98b8f882ac84332e7dfdc996a081d1c5e6189ad726f8f8afec5d36a20a730']

Name

f6223d956df81dcb6135c6ce00ee14d0efede9fb399b56d2ee95b7b0538fe12c

Pattern Type

stix

Pattern

[file:hashes.'SHA-256' =
'f6223d956df81dcb6135c6ce00ee14d0efede9fb399b56d2ee95b7b0538fe12c']

Name

c59e17806e3a58792f07662b4985119252c8221688084d20b599699bfdb272d8

Pattern Type

stix

Pattern

[file:hashes.'SHA-256' =
'c59e17806e3a58792f07662b4985119252c8221688084d20b599699bfdb272d8']

Name

ca6713bedbd19c2ad560700b41774825615b0fe80bf61751177ffbc26c77aa30

Pattern Type

stix

Pattern

[file:hashes.'SHA-256' =
'ca6713bedbd19c2ad560700b41774825615b0fe80bf61751177ffbc26c77aa30']

Name

28109c650df5481c3997b720bf8ce09e7472d9cdb3f02dd844783fd2b1400c72

Pattern Type

stix

Pattern

[file:hashes.'SHA-256' = '28109c650df5481c3997b720bf8ce09e7472d9cdb3f02dd844783fd2b1400c72']

Name

cdn-7a3d.vultr-dns.com

Pattern Type

stix

Pattern

[hostname:value = 'cdn-7a3d.vultr-dns.com']

Name

nx2.microware-help.com

Pattern Type

stix

Pattern

[hostname:value = 'nx2.microware-help.com']

Name
dff1d282e754f378ef00fb6ebe9944fee6607d9ee24ec3ca643da27f27520ac3
Pattern Type
stix
Pattern
[file:hashes.'SHA-256' = 'dff1d282e754f378ef00fb6ebe9944fee6607d9ee24ec3ca643da27f27520ac3']
Name
49a0349dfa79b211fc2c5753a9b87f8cd2e9a42e55eca6f350f30c60de2866ce
Pattern Type
stix
Pattern
[file:hashes.'SHA-256' = '49a0349dfa79b211fc2c5753a9b87f8cd2e9a42e55eca6f350f30c60de2866ce']
Name
4b014891df3348a76750563ae10b70721e028381f3964930d2dd49b9597ffac3
Pattern Type
stix
Pattern

[file:hashes.'SHA-256' = '4b014891df3348a76750563ae10b70721e028381f3964930d2dd49b9597ffac3']

Name

71a503b5b6ec8321346bee3f6129af0b8ad490a36092488d085085cdc0fc6b9d

Pattern Type

stix

Pattern

[file:hashes.'SHA-256' = '71a503b5b6ec8321346bee3f6129af0b8ad490a36092488d085085cdc0fc6b9d']

Name

103.159.133.205

Description

ISP: Gigabit Hosting Sdn Bhd **OS:** None ------------------------- Hostnames: ------------------------- Domains: ------------------------- Services: **80:** ``` HTTP/1.1 200 OK Content-Type: text/html Last-Modified: Thu, 30 Apr 2020 03:00:07 GMT Accept-Ranges: bytes ETag: "fbb72e749b1ed61:0" Server: Microsoft-IIS/8.5 Date: Mon, 14 Aug 2023 13:33:06 GMT Content-Length: 30646 ``` ------------------ **8081:** ``` ``` HEARTBLEED: 2023/08/31 09:05:45 103.159.133.205:8081 - SAFE ------------------

Pattern Type

stix

Pattern

[ipv4-addr:value = '103.159.133.205']

Name

96.44.160.181

Description

ISP: QuadraNet Enterprises LLC **OS:** None ------------------------- Hostnames: - 96.44.160.181.static.quadranet.com ------------------------- Domains: - quadranet.com ------------------------- Services: **80:** ``` HTTP/1.1 200 OK Content-Type: text/html Last-Modified: Sat, 30 Jul 2016 05:56:40 GMT Accept-Ranges: bytes ETag: "60212a2427ead11:0" Server: Microsoft-IIS/7.5 X-Powered-By: ASP.NET Date: Wed, 30 Aug 2023 17:05:18 GMT Content-Length: 689 ``` ------------------

Pattern Type

stix

Pattern

[ipv4-addr:value = '96.44.160.181']

Name

eeb3d2e87d343b2acf6bc8e4e4122d76a9ad200ae52340c61e537a80666705ed

Pattern Type

stix

Pattern

[file:hashes.'SHA-256' = 'eeb3d2e87d343b2acf6bc8e4e4122d76a9ad200ae52340c61e537a80666705ed']

Name

cdn728a66b0.smartlinkcorp.net
Pattern Type
stix
Pattern
[hostname:value = 'cdn728a66b0.smartlinkcorp.net']

Intrusion-Set

Name
Earth Estries

Country

Name
South Africa

Name
Malaysia

Name
Philippines

Name
Taiwan

Name
Germany

Name
United States of America

Malware

Name
TrojanSpy

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.

StixFile

Value
71a503b5b6ec8321346bee3f6129af0b8ad490a36092488d085085cdc0fc6b9d
1a9e0c7c88e7a8b065ec88809187f67d920e7845350d94098645e592ec5534f6
45b9204ccbad92e4e5fb9e31aab683eb5221eb5f5688b1aae98d9c0f1c920227
42d4eb7f04111631891379c5cce55480d2d9d2ef8feaf1075e1aed0c52df4bb9
8476ad68ce54b458217ab165d66a899d764eae3ad30196f35d2ff20d3f398523
82f3384723b21f9a928029bb3ee116f9adbc4f7ec66d5a856e817c3dc16d149d
a8dd0ca6151000de33335f48a832d24412de13ce05ea6f279bf4aaaa2e5aaecb
c59e17806e3a58792f07662b4985119252c8221688084d20b599699bfdb272d8
49a0349dfa79b211fc2c5753a9b87f8cd2e9a42e55eca6f350f30c60de2866ce
eeb3d2e87d343b2acf6bc8e4e4122d76a9ad200ae52340c61e537a80666705ed
0eaa67fe81cec0a41cd42866df1223cb7d2b5659ab295dffe64fe9c3b76720aa
deaa3143814c6fe9279e8bc0706df22d63ef197af980d8feae9a8468f441efec
dff1d282e754f378ef00fb6ebe9944fee6607d9ee24ec3ca643da27f27520ac3

e6f9756613345fd01bbcf28eba15d52705ef4d144c275b8cfe868a5d28c24140
415e0893ce227464fb29d76e0500c518935d11379d17fb14effaef82e962ff76
ca6713bedbd19c2ad560700b41774825615b0fe80bf61751177ffbc26c77aa30
2531891691ef674345f098ef18b274091acdf3f2808cca753674599c043ccd7d
98e250bc06de38050fdeab9b1e2ef7e4d8c401b33fd5478f3b85197112858f4e
b1bc10fa25a4fd5ae7948c6523eb975be8d0f52d1572c57a7ef736134b996586
cdadad8d7ced1370baa5d1ffe435bed78c2d58ed4cda364b8a7484e3c7cdac98
b6481e0edc36a0472ab0ce7d0817f1773c4af9307ae60890a667930558a762ff
efb98b8f882ac84332e7dfdc996a081d1c5e6189ad726f8f8afec5d36a20a730
4b014891df3348a76750563ae10b70721e028381f3964930d2dd49b9597ffac3
e1a7e5f27362aaf0d12b58b96a816ef61a2a498def9805297aa81f6f83729230
f6223d956df81dcb6135c6ce00ee14d0efede9fb399b56d2ee95b7b0538fe12c
28109c650df5481c3997b720bf8ce09e7472d9cdb3f02dd844783fd2b1400c72
c7023183e815b9aff68d3eba6c2ca105dbe0a9b05cd209908dcee907a64ce80b
cd2b703e1b7cfd6c552406f44ec05480209003789ad4fbba4d4cffd4f104b0a0

Hostname

Value
cdn-6dd0035.oxcdntech.com
ms101.cloudshappen.com
web9a78bc52.trhammer.com
cdn728a66b0.smartlinkcorp.net
east.smartpisang.com
nx2.microware-help.com
access.trhammer.com
cdn-7a3d.vultr-dns.com

IPv4-Addr

Value
96.44.160.181
103.159.133.205

External References

- https://otx.alienvault.com/pulse/64f09792546303290ab09c15
- https://www.trendmicro.com/content/dam/trendmicro/global/en/research/23/h/earth-estries-targets-government-tech-for-cyberespionage/IOCs-earth-estries-targets-government-tech-for-cyberespionage.txt
- https://www.trendmicro.com/en_us/research/23/h/earth-estries-targets-government-tech-for-cyberespionage.html