Jun 28 2024

NETMANAGEIT

Intelligence Report

Examining Water Infection Routine Leading to an XMRig Cryptominer

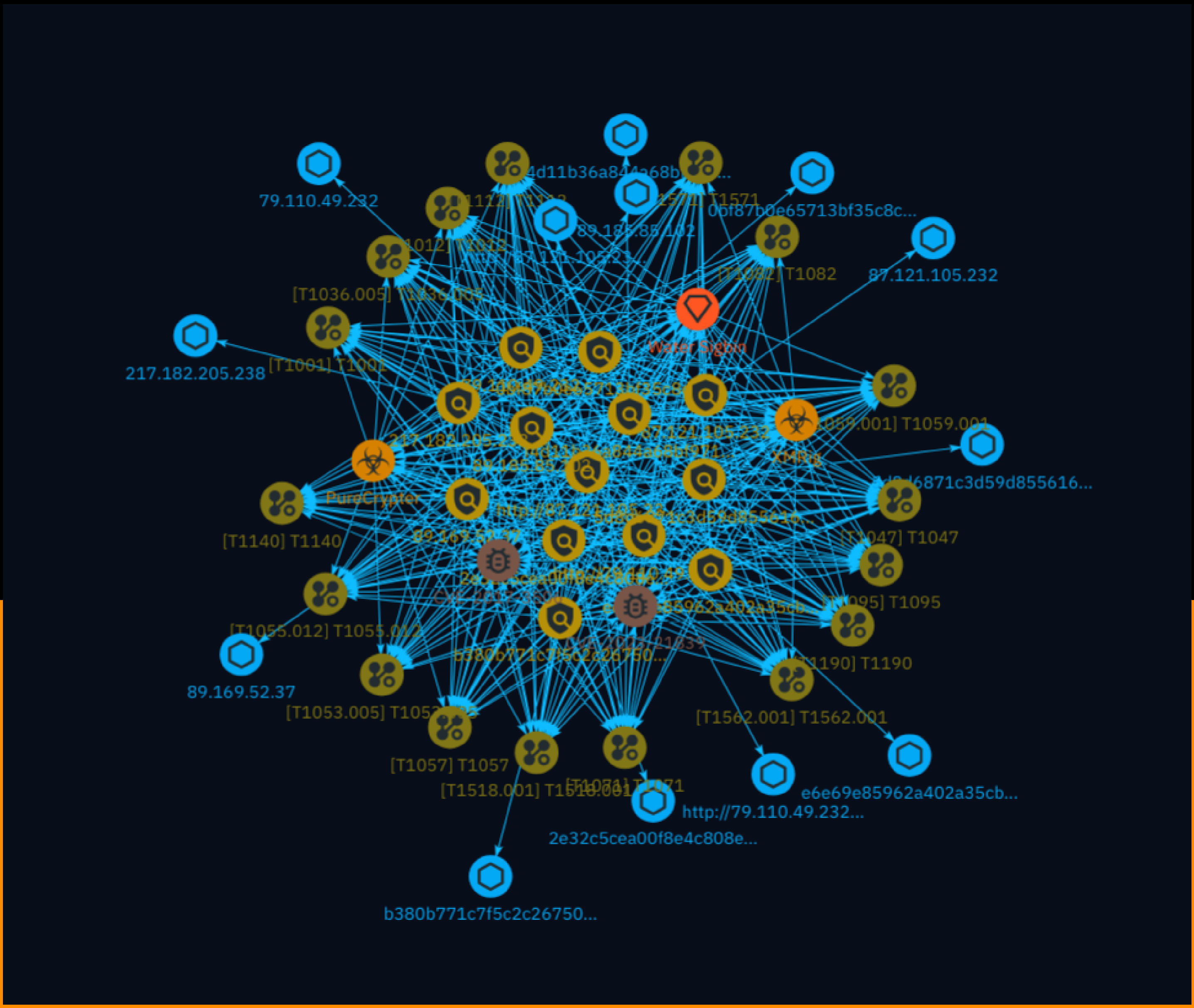

Table of contents

Overview

Entities

Observables

External References

Overview

Description

This report details the multi-stage loading technique utilized by the threat actor Water Sigbin to deliver the PureCrypter loader and XMRig cryptocurrency miner. The actor exploits vulnerabilities in Oracle WebLogic servers, employing fileless execution tactics like DLL reflective and process injection to evade disk-based detection mechanisms. The malware uses code protection software and anti-debugging techniques for obfuscation, making analysis challenging.

Confidence

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

100 / 100

Content

N/A

Attack-Pattern

Name
T1057
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. Administrator or otherwise elevated access may provide better process details. 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
T1047

ID

T1047

Description

Adversaries may abuse Windows Management Instrumentation (WMI) to execute malicious commands and payloads. WMI is designed for programmers and is the infrastructure for management data and operations on Windows systems.(Citation: WMI 1-3) 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) and [Windows Remote Management](https://attack.mitre.org/techniques/T1021/006).(Citation: WMI 1-3) Remote WMI over DCOM operates using port 135, whereas WMI over WinRM operates over port 5985 when using HTTP and 5986 for HTTPS.(Citation: WMI 1-3) (Citation: Mandiant WMI) 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](https://attack.mitre.org/tactics/TA0007) as well as [Execution](https://attack.mitre.org/tactics/TA0002) of commands and payloads.(Citation: Mandiant WMI) For example, `wmic.exe` can be abused by an adversary to delete shadow copies with the command `wmic.exe Shadowcopy Delete` (i.e., [Inhibit System Recovery](https://attack.mitre.org/techniques/T1490)).(Citation: WMI 6) **Note:** `wmic.exe` is deprecated as of January of 2024, with the WMIC feature being “disabled by default” on Windows 11+. WMIC will be removed from subsequent Windows releases and replaced by [PowerShell](https://attack.mitre.org/techniques/T1059/001) as the primary WMI interface.(Citation: WMI 7,8) In addition to PowerShell and tools like `wbemtool.exe`, COM APIs can also be used to programmatically interact with WMI via C++, .NET, VBScript, etc.(Citation: WMI 7,8)

Name

T1036.005

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

T1518.001

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 cloud monitoring agents 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 the [Cloud API](https://attack.mitre.org/techniques/T1059/009) to discover cloud-native security software installed on compute infrastructure, such as the AWS CloudWatch agent, Azure VM Agent, and Google Cloud Monitor agent. These agents may collect metrics and logs from the VM, which may be centrally aggregated in a cloud-based monitoring platform.

Name

T1012

ID

T1012

Description

Adversaries may interact with the Windows Registry to gather information about the system, configuration, and installed software. The Registry contains a significant amount of information about the operating system, configuration, software, and security.(Citation: Wikipedia Windows Registry) Information can easily be queried using the [Reg](https://attack.mitre.org/software/S0075) utility, though other means to access the Registry exist. Some of the information may help adversaries to further their operation within a network. Adversaries may use the information from [Query Registry](https://attack.mitre.org/techniques/T1012) during automated discovery to shape follow-on behaviors, including whether or not the adversary fully infects the target and/or attempts specific actions.

Name

T1571

ID

T1571

Description

Adversaries may communicate using a protocol and port pairing that are typically not associated. For example, HTTPS over port 8088(Citation: Symantec Elfin Mar 2019) or port 587(Citation: Fortinet Agent Tesla April 2018) as opposed to the traditional port 443. Adversaries may make changes to the standard port used by a protocol to bypass filtering or muddle analysis/parsing of network data. Adversaries may also make changes to victim systems to abuse non-standard ports. For example, Registry keys and other configuration settings can be used to modify protocol and port pairings.(Citation: change_rdp_port_conti)

Name

T1055.012

ID

T1055.012

Description

Adversaries may inject malicious code into suspended and hollowed processes in order to evade process-based defenses. Process hollowing is a method of executing arbitrary code in the address space of a separate live process. Process hollowing is commonly performed by creating a process in a suspended state then unmapping/hollowing its memory, which can then be replaced with malicious code. A victim process can be created with native Windows API calls such as `CreateProcess`, which includes a flag to suspend the processes primary thread. At this point the process can be unmapped using APIs calls such as `ZwUnmapViewOfSection` or `NtUnmapViewOfSection` before being written to, realigned to the injected code, and resumed via `VirtualAllocEx`, `WriteProcessMemory`, `SetThreadContext`, then `ResumeThread` respectively.(Citation: Leitch Hollowing)(Citation: Elastic Process Injection July 2017) This is very similar to [Thread Local Storage](https://attack.mitre.org/techniques/T1055/005) but creates a new process rather than targeting an existing process. This behavior will likely not result in elevated privileges since the injected process was spawned from (and thus inherits the security context) of the injecting process. However, execution via process hollowing may also evade detection from security products since the execution is masked under a legitimate process.

Name

T1059.001

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

T1001

ID

T1001

Description

Adversaries may obfuscate command and control traffic to make it more difficult to detect.(Citation: Bitdefender FunnyDream Campaign November 2020) Command and control (C2) communications are hidden (but not necessarily encrypted) in an attempt to make the content more difficult to discover or decipher and to make the communication less conspicuous and hide commands from being seen. This encompasses many methods, such as adding junk data to protocol traffic, using steganography, or impersonating legitimate protocols.

Name

T1112

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

T1053.005

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

T1190

ID

T1190

Description

Adversaries may attempt to exploit a weakness in an Internet-facing host or system to initially access a network. The weakness in the system can be a software bug, a temporary glitch, or a misconfiguration. Exploited applications are often websites/web servers, but can also include databases (like SQL), standard services (like SMB or SSH), network device administration and management protocols (like SNMP and Smart Install), and any other system with Internet accessible open sockets.(Citation: NVD CVE-2016-6662)(Citation: CIS Multiple SMB Vulnerabilities)(Citation: US-CERT TA18-106A Network Infrastructure Devices 2018)(Citation: Cisco Blog Legacy Device Attacks)(Citation: NVD CVE-2014-7169) Depending on the flaw being exploited this may also involve [Exploitation for Defense Evasion] (https://attack.mitre.org/techniques/T1211) or [Exploitation for Client Execution](https://attack.mitre.org/techniques/T1203). If an application is hosted on cloud-based infrastructure and/or is containerized, then exploiting it may lead to compromise of the underlying instance or container. This can allow an adversary a path to access the cloud or container APIs, exploit container host access via [Escape to Host](https://attack.mitre.org/techniques/T1611), or take advantage of weak identity and access management policies. Adversaries may also exploit edge network infrastructure and related appliances, specifically targeting devices that do not support robust host-based defenses.(Citation: Mandiant Fortinet Zero Day)(Citation: Wired Russia Cyberwar) For websites and databases, the OWASP top 10 and CWE top 25 highlight the most common web-based vulnerabilities.(Citation: OWASP Top 10)(Citation: CWE top 25)

Name

T1095

ID

T1095

Description

Adversaries may use an OSI non-application layer protocol for communication between host and C2 server or among infected hosts within a network. The list of possible protocols is extensive.(Citation: Wikipedia OSI) Specific examples include use of network layer protocols, such as the Internet Control Message Protocol (ICMP), transport layer protocols, such as the User Datagram Protocol (UDP), session layer protocols, such as Socket Secure (SOCKS), as well as redirected/tunneled protocols, such as Serial over LAN (SOL). ICMP communication between hosts is one example.(Citation: Cisco Synful Knock Evolution) Because ICMP is part of the Internet Protocol Suite, it is required to be implemented by all IP-compatible hosts.(Citation: Microsoft ICMP) However, it is not as commonly monitored as other Internet Protocols such as TCP or UDP and may be used by adversaries to hide communications.

Name

T1562.001

ID

T1562.001

Description

Adversaries may modify and/or disable security tools to avoid possible detection of their malware/tools and activities. This may take many forms, such as killing security software processes or services, modifying / deleting Registry keys or configuration files so that tools do not operate properly, or other methods to interfere with security tools scanning or reporting information. Adversaries may also disable updates to prevent the latest security patches from reaching tools on victim systems.(Citation: SCADAfence_ransomware) Adversaries may also tamper with artifacts deployed and utilized by security tools. Security tools may make dynamic changes to system components in order to maintain visibility into specific events. For example, security products may load their own modules and/or modify those loaded by processes to facilitate data collection. Similar to [Indicator Blocking](https://attack.mitre.org/techniques/T1562/006), adversaries may unhook or otherwise modify these features added by tools (especially those that exist in userland or are otherwise potentially accessible to adversaries) to avoid detection.(Citation: OutFlank System Calls)(Citation: MDSec System Calls) Adversaries may also focus on specific applications such as Sysmon. For example, the “Start” and “Enable” values in `HKEY_LOCAL_MACHINE\SYSTEM\CurrentControlSet\Control\WMI\Autologger\EventLog-Microsoft-Windows-Sysmon-Operational` may be modified to tamper with and potentially disable Sysmon logging.(Citation: disable_win_evt_logging) On network devices, adversaries may attempt to skip digital signature verification checks by altering startup configuration files and effectively disabling firmware verification that typically occurs at boot.(Citation: Fortinet Zero-Day and Custom Malware Used by Suspected Chinese Actor in Espionage Operation)(Citation: Analysis of FG-IR-22-369) In cloud environments, tools disabled by adversaries may include cloud monitoring agents that report back to services such as AWS CloudWatch or Google Cloud Monitor. Furthermore, although defensive tools may have anti-tampering mechanisms, adversaries may abuse tools such as legitimate rootkit removal kits to impair and/or disable these tools.(Citation: chasing_avaddon_ransomware)(Citation: dharma_ransomware)(Citation: demystifying_ryuk)(Citation: doppelpaymer_crowdstrike) For example, adversaries have used tools such as GMER to find and shut down hidden processes and antivirus software on infected systems.(Citation: demystifying_ryuk) Additionally, adversaries may exploit legitimate drivers from anti-virus software to gain access to kernel space (i.e. [Exploitation for Privilege Escalation](https://attack.mitre.org/techniques/T1068)), which may lead to bypassing anti-tampering features.(Citation: avoslocker_ransomware)

Name

T1071

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.(Citation: Mandiant APT29 Eye Spy Email Nov 22)

Name

T1140

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

T1082

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)

Indicator

Name

b380b771c7f5c2c26750e281101873772e10c8c1a0d2a2ff0aff1912b569ab93

Pattern Type

stix

Pattern

[file:hashes.'SHA-256' = 'b380b771c7f5c2c26750e281101873772e10c8c1a0d2a2ff0aff1912b569ab93']

Name

89.185.85.102

Description

ISP: AEZA INTERNATIONAL LTD **OS:** - -------------------------- Services: **135:** ``` Microsoft RPC Endpoint Mapper d95afe70-a6d5-4259-822e-2c84da1ddb0d version: v1.0 protocol: [MS-RSP]: Remote Shutdown Protocol provider: wininit.exe ncacn_ip_tcp: 89.185.85.102:49152 ncalrpc: WindowsShutdown ncacn_np: \\WIN-TTIVDJM46CK\PIPE\InitShutdown ncalrpc: WMsgKRpc049ED0 76f226c3-ec14-4325-8a99-6a46348418af version: v1.0 provider: winlogon.exe ncalrpc: WindowsShutdown ncacn_np: \\WIN-TTIVDJM46CK\PIPE\InitShutdown ncalrpc: WMsgKRpc049ED0 ncalrpc: WMsgKRpc0571D1 ncalrpc: WMsgKRpc031A4A2 9b008953-f195-4bf9-bde0-4471971e58ed version: v1.0 ncalrpc: LRPC-1042774e2c41aee5f9 ncacn_np: \\WIN-TTIVDJM46CK\pipe\LSM_API_service ncalrpc: LSMApi ncalrpc:

```
LRPC-5793125041c2e7d11e ncalrpc: actkernel ncalrpc: umpo 697dcda9-3ba9-4eb2-9247-
e11f1901b0d2 version: v1.0 ncalrpc: LRPC-1042774e2c41aee5f9 ncacn_np: \\WIN-
TTIVDJM46CK\pipe\LSM_API_service ncalrpc: LSMApi ncalrpc: LRPC-5793125041c2e7d11e
ncalrpc: actkernel ncalrpc: umpo c9ac6db5-82b7-4e55-ae8a-e464ed7b4277 version: v1.0
annotation: Impl friendly name provider: sysntfy.dll ncalrpc: LRPC-5793125041c2e7d11e
ncalrpc: actkernel ncalrpc: umpo ncalrpc: LRPC-17d1e0d6146d21a2b4 ncacn_np: \\WIN-
TTIVDJM46CK\PIPE\srvsvc ncacn_ip_tcp: 89.185.85.102:49154 ncalrpc: ubpmtaskhostchannel
ncacn_np: \\WIN-TTIVDJM46CK\PIPE\atsvc ncalrpc: senssvc ncalrpc:
OLE929717B9DD7482D1CF5CE46F6F5F ncalrpc: IUserProfile2 ncalrpc: senssvc ncalrpc:
OLE929717B9DD7482D1CF5CE46F6F5F ncalrpc: IUserProfile2 ncalrpc: IUserProfile2 ncalrpc:
IUserProfile2 0d3e2735-cea0-4ecc-a9e2-41a2d81aed4e version: v1.0 ncalrpc: actkernel
ncalrpc: umpo c605f9fb-f0a3-4e2a-a073-73560f8d9e3e version: v1.0 ncalrpc: actkernel
ncalrpc: umpo 1b37ca91-76b1-4f5e-a3c7-2abfc61f2bb0 version: v1.0 ncalrpc: actkernel
ncalrpc: umpo 8bfc3be1-6def-4e2d-af74-7c47cd0ade4a version: v1.0 ncalrpc: actkernel
ncalrpc: umpo 2d98a740-581d-41b9-aa0d-a88b9d5ce938 version: v1.0 ncalrpc: actkernel
ncalrpc: umpo bdaa0970-413b-4a3e-9e5d-f6dc9d7e0760 version: v1.0 ncalrpc: actkernel
ncalrpc: umpo 3b338d89-6cfa-44b8-847e-531531bc9992 version: v1.0 ncalrpc: actkernel
ncalrpc: umpo 8782d3b9-ebbd-4644-a3d8-e8725381919b version: v1.0 ncalrpc: actkernel
ncalrpc: umpo 085b0334-e454-4d91-9b8c-4134f9e793f3 version: v1.0 ncalrpc: actkernel
ncalrpc: umpo 4bec6bb8-b5c2-4b6f-b2c1-5da5cf92d0d9 version: v1.0 ncalrpc: actkernel
ncalrpc: umpo 3c4728c5-f0ab-448b-bda1-6ce01eb0a6d5 version: v1.0 annotation: DHCP
Client LRPC Endpoint provider: dhcpcsvc.dll ncalrpc: dhcpcsvc ncalrpc: dhcpcsvc6 ncalrpc:
LRPC-463d682ac5ec598521 ncacn_ip_tcp: 89.185.85.102:49153 ncacn_np: \\WIN-
TTIVDJM46CK\pipe\eventlog ncalrpc: eventlog 3c4728c5-f0ab-448b-bda1-6ce01eb0a6d6
version: v1.0 annotation: DHCPv6 Client LRPC Endpoint provider: dhcpcsvc6.dll ncalrpc:
dhcpcsvc6 ncalrpc: LRPC-463d682ac5ec598521 ncacn_ip_tcp: 89.185.85.102:49153 ncacn_np: \
\WIN-TTIVDJM46CK\pipe\eventlog ncalrpc: eventlog
abfb6ca3-0c5e-4734-9285-0aee72fe8d1c version: v1.0 annotation: Wcm Service ncalrpc:
LRPC-463d682ac5ec598521 ncacn_ip_tcp: 89.185.85.102:49153 ncacn_np: \\WIN-
TTIVDJM46CK\pipe\eventlog ncalrpc: eventlog 30adc50c-5cbc-46ce-9a0e-91914789e23c
version: v1.0 annotation: NRP server endpoint provider: nrpsrv.dll ncalrpc:
LRPC-463d682ac5ec598521 ncacn_ip_tcp: 89.185.85.102:49153 ncacn_np: \\WIN-
TTIVDJM46CK\pipe\eventlog ncalrpc: eventlog f6beaff7-1e19-4fbb-9f8f-b89e2018337c
version: v1.0 annotation: Event log TCPIP protocol: [MS-EVEN6]: EventLog Remoting Protocol
provider: wevtsvc.dll ncacn_ip_tcp: 89.185.85.102:49153 ncacn_np: \\WIN-
TTIVDJM46CK\pipe\eventlog ncalrpc: eventlog 30b044a5-a225-43f0-b3a4-e060df91f9c1
version: v1.0 provider: certprop.dll ncalrpc: LRPC-17d1e0d6146d21a2b4 ncacn_np: \\WIN-
TTIVDJM46CK\PIPE\srvsvc ncacn_ip_tcp: 89.185.85.102:49154 ncalrpc: ubpmtaskhostchannel
ncacn_np: \\WIN-TTIVDJM46CK\PIPE\atsvc ncalrpc: senssvc ncalrpc:
OLE929717B9DD7482D1CF5CE46F6F5F ncalrpc: IUserProfile2 1a0d010f-1c33-432c-
b0f5-8cf4e8053099 version: v1.0 annotation: IdSegSrv service ncacn_ip_tcp:
89.185.85.102:49154 ncalrpc: ubpmtaskhostchannel ncacn_np: \\WIN-
TTIVDJM46CK\PIPE\atsvc ncalrpc: senssvc ncalrpc: OLE929717B9DD7482D1CF5CE46F6F5F
ncalrpc: IUserProfile2 98716d03-89ac-44c7-bb8c-285824e51c4a version: v1.0 annotation:
```

```
XactSrv service provider: srvsvc.dll ncacn_ip_tcp: 89.185.85.102:49154 ncalrpc:
ubpmtaskhostchannel ncacn_np: \\WIN-TTIVDJM46CK\PIPE\atsvc ncalrpc: senssvc ncalrpc:
OLE929717B9DD7482D1CF5CE46F6F5F ncalrpc: IUserProfile2 c49a5a70-8a7f-4e70-
ba16-1e8f1f193ef1 version: v1.0 annotation: Adh APIs ncacn_ip_tcp: 89.185.85.102:49154
ncalrpc: ubpmtaskhostchannel ncacn_np: \\WIN-TTIVDJM46CK\PIPE\atsvc ncalrpc: senssvc
ncalrpc: OLE929717B9DD7482D1CF5CE46F6F5F ncalrpc: IUserProfile2 c36be077-
e14b-4fe9-8abc-e856ef4f048b version: v1.0 annotation: Proxy Manager client server
endpoint ncacn_ip_tcp: 89.185.85.102:49154 ncalrpc: ubpmtaskhostchannel ncacn_np: \
\WIN-TTIVDJM46CK\PIPE\atsvc ncalrpc: senssvc ncalrpc:
OLE929717B9DD7482D1CF5CE46F6F5F ncalrpc: IUserProfile2 2e6035b2-e8f1-41a7-
a044-656b439c4c34 version: v1.0 annotation: Proxy Manager provider server endpoint
ncacn_ip_tcp: 89.185.85.102:49154 ncalrpc: ubpmtaskhostchannel ncacn_np: \\WIN-
TTIVDJM46CK\PIPE\atsvc ncalrpc: senssvc ncalrpc: OLE929717B9DD7482D1CF5CE46F6F5F
ncalrpc: IUserProfile2 552d076a-cb29-4e44-8b6a-d15e59e2c0af version: v1.0 annotation: IP
Transition Configuration endpoint provider: iphlpsvc.dll ncacn_ip_tcp: 89.185.85.102:49154
ncalrpc: ubpmtaskhostchannel ncacn_np: \\WIN-TTIVDJM46CK\PIPE\atsvc ncalrpc: senssvc
ncalrpc: OLE929717B9DD7482D1CF5CE46F6F5F ncalrpc: IUserProfile2 a398e520-d59a-4bdd-
aa7a-3c1e0303a511 version: v1.0 annotation: IKE/Authip API provider: IKEEXT.DLL
ncacn_ip_tcp: 89.185.85.102:49154 ncalrpc: ubpmtaskhostchannel ncacn_np: \\WIN-
TTIVDJM46CK\PIPE\atsvc ncalrpc: senssvc ncalrpc: OLE929717B9DD7482D1CF5CE46F6F5F
ncalrpc: IUserProfile2 3a9ef155-691d-4449-8d05-09ad57031823 version: v1.0 ncacn_ip_tcp:
89.185.85.102:49154 ncalrpc: ubpmtaskhostchannel ncacn_np: \\WIN-
TTIVDJM46CK\PIPE\atsvc ncalrpc: senssvc ncalrpc: OLE929717B9DD7482D1CF5CE46F6F5F
ncalrpc: IUserProfile2 86d35949-83c9-4044-b424-db363231fd0c version: v1.0 protocol: [MS-
TSCH]: Task Scheduler Service Remoting Protocol provider: schedsvc.dll ncacn_ip_tcp:
89.185.85.102:49154 ncalrpc: ubpmtaskhostchannel ncacn_np: \\WIN-
TTIVDJM46CK\PIPE\atsvc ncalrpc: senssvc ncalrpc: OLE929717B9DD7482D1CF5CE46F6F5F
ncalrpc: IUserProfile2 378e52b0-c0a9-11cf-822d-00aa0051e40f version: v1.0 protocol: [MS-
TSCH]: Task Scheduler Service Remoting Protocol provider: taskcomp.dll ncacn_np: \\WIN-
TTIVDJM46CK\PIPE\atsvc ncalrpc: senssvc ncalrpc: OLE929717B9DD7482D1CF5CE46F6F5F
ncalrpc: IUserProfile2 1ff70682-0a51-30e8-076d-740be8cee98b version: v1.0 protocol: [MS-
TSCH]: Task Scheduler Service Remoting Protocol provider: taskcomp.dll ncacn_np: \\WIN-
TTIVDJM46CK\PIPE\atsvc ncalrpc: senssvc ncalrpc: OLE929717B9DD7482D1CF5CE46F6F5F
ncalrpc: IUserProfile2 0a74ef1c-41a4-4e06-83ae-dc74fb1cdd53 version: v1.0 provider:
schedsvc.dll ncalrpc: senssvc ncalrpc: OLE929717B9DD7482D1CF5CE46F6F5F ncalrpc:
IUserProfile2 2eb08e3e-639f-4fba-97b1-14f878961076 version: v1.0 annotation: Group Policy
RPC Interface provider: gpsvc.dll ncalrpc: LRPC-87cf5b49fe3d867a31
3473dd4d-2e88-4006-9cba-22570909dd10 version: v5.256 annotation: WinHttp Auto-Proxy
Service ncalrpc: LRPC-9c314c15d1dc571082 ncalrpc: OLE22A36144A20675E392F11CCB068F
7ea70bcf-48af-4f6a-8968-6a440754d5fa version: v1.0 annotation: NSI server endpoint
provider: nsisvc.dll ncalrpc: LRPC-9c314c15d1dc571082 ncalrpc:
OLE22A36144A20675E392F11CCB068F b2507c30-b126-494a-92ac-ee32b6eeb039 version: v1.0
ncalrpc: LRPC-a4349ec9d52f3be344 2fb92682-6599-42dc-ae13-bd2ca89bd11c version: v1.0
annotation: Fw APIs provider: MPSSVC.dll ncalrpc: LRPC-2e031641597c6590cf ncalrpc:
```

~~~~
LRPC-9e7d5185ed5853a09b f47433c3-3e9d-4157-aad4-83aa1f5c2d4c version: v1.0 annotation:
Fw APIs ncalrpc: LRPC-2e031641597c6590cf ncalrpc: LRPC-9e7d5185ed5853a09b
7f9d11bf-7fb9-436b-a812-b2d50c5d4c03 version: v1.0 annotation: Fw APIs provider:
MPSSVC.dll ncalrpc: LRPC-2e031641597c6590cf ncalrpc: LRPC-9e7d5185ed5853a09b
dd490425-5325-4565-b774-7e27d6c09c24 version: v1.0 annotation: Base Firewall Engine API
provider: BFE.DLL ncalrpc: LRPC-9e7d5185ed5853a09b 7f1343fe-50a9-4927-a778-0c5859517bac
version: v1.0 annotation: DfsDs service ncacn_np: \\WIN-TTIVDJM46CK\PIPE\wkssvc ncalrpc:
LRPC-0bb1275dbf969a338d ncalrpc: DNSResolver eb081a0d-10ee-478a-a1dd-50995283e7a8
version: v3.0 annotation: Witness Client Test Interface ncalrpc: LRPC-0bb1275dbf969a338d
ncalrpc: DNSResolver f2c9b409-c1c9-4100-8639-d8ab1486694a version: v1.0 annotation:
Witness Client Upcall Server ncalrpc: LRPC-0bb1275dbf969a338d ncalrpc: DNSResolver
76f03f96-cdfd-44fc-a22c-64950a001209 version: v1.0 protocol: [MS-PAR]: Print System
Asynchronous Remote Protocol provider: spoolsv.exe ncacn_ip_tcp: 89.185.85.102:49155
ncalrpc: LRPC-f09c38084aeaa23c8b 4a452661-8290-4b36-8fbe-7f4093a94978 version: v1.0
provider: spoolsv.exe ncacn_ip_tcp: 89.185.85.102:49155 ncalrpc: LRPC-f09c38084aeaa23c8b
ae33069b-a2a8-46ee-a235-ddfd339be281 version: v1.0 protocol: [MS-PAN]: Print System
Asynchronous Notification Protocol provider: spoolsv.exe ncacn_ip_tcp: 89.185.85.102:49155
ncalrpc: LRPC-f09c38084aeaa23c8b 0b6edbfa-4a24-4fc6-8a23-942b1eca65d1 version: v1.0
protocol: [MS-PAN]: Print System Asynchronous Notification Protocol provider: spoolsv.exe
ncacn_ip_tcp: 89.185.85.102:49155 ncalrpc: LRPC-f09c38084aeaa23c8b 12345678-1234-abcd-
ef00-0123456789ab version: v1.0 protocol: [MS-RPRN]: Print System Remote Protocol
provider: spoolsv.exe ncacn_ip_tcp: 89.185.85.102:49155 ncalrpc: LRPC-f09c38084aeaa23c8b
367abb81-9844-35f1-ad32-98f038001003 version: v2.0 protocol: [MS-SCMR]: Service Control
Manager Remote Protocol provider: services.exe ncacn_ip_tcp: 89.185.85.102:49156
6b5bdd1e-528c-422c-af8c-a4079be4fe48 version: v1.0 annotation: Remote Fw APIs protocol:
[MS-FASP]: Firewall and Advanced Security Protocol provider: FwRemoteSvr.dll
ncacn_ip_tcp: 89.185.85.102:49157 12345778-1234-abcd-ef00-0123456789ac version: v1.0
protocol: [MS-SAMR]: Security Account Manager (SAM) Remote Protocol provider: samsrv.dll
ncacn_ip_tcp: 89.185.85.102:49161 ncalrpc: samss lpc ncalrpc: SidKey Local End Point ncalrpc:
protected_storage ncalrpc: lsasspirpc ncalrpc: lsapolicylookup ncalrpc: LSA_EAS_ENDPOINT
ncalrpc: lsacap ncalrpc: LSARPC_ENDPOINT ncalrpc: securityevent ncalrpc: audit ncacn_np:
\\WIN-TTIVDJM46CK\pipe\lsass 906b0ce0-c70b-1067-b317-00dd010662da version: v1.0
protocol: [MS-CMPO]: MSDTC Connection Manager: provider: msdtcprx.dll ncalrpc:
LRPC-27ca4eb415e36ad718 ncalrpc: LRPC-27ca4eb415e36ad718 ncalrpc:
LRPC-27ca4eb415e36ad718 12e65dd8-887f-41ef-91bf-8d816c42c2e7 version: v1.0 annotation:
Secure Desktop LRPC interface provider: winlogon.exe ncalrpc: WMsgKRpc031A4A2 ```
------------------ **443:** ``` ``` HEARTBLEED: 2024/06/27 12:04:56 89.185.85.102:443 - SAFE
------------------ **445:** ``` SMB Status: Authentication: enabled SMB Version: 1 OS:
Windows Server 2012 R2 Standard 9600 Software: Windows Server 2012 R2 Standard 6.3
Capabilities: extended-security, infolevel-passthru, large-files, large-readx, large-writex,
level2-oplocks, lock-and-read, lwio, nt-find, nt-smb, nt-status, rpc-remote-api, unicode ```
------------------ **3389:** ``` Remote Desktop Protocol
\x03\x00\x00\x13\x0e\xd0\x00\x00\x124\x00\x02\x0f\x08\x00\x02\x00\x00\x00 Remote
Desktop Protocol NTLM Info: OS: Windows 8.1/Windows Server 2012 R2 OS Build: 6.3.9600
~~~~

Target Name: WIN-TTIVDJM46CK NetBIOS Domain Name: WIN-TTIVDJM46CK NetBIOS Computer Name: WIN-TTIVDJM46CK DNS Domain Name: WIN-TTIVDJM46CK FQDN: WIN-TTIVDJM46CK ``` ------------------ **5985:** ``` HTTP/1.1 404 Not Found Content-Type: text/html; charset=us-ascii Server: Microsoft-HTTPAPI/2.0 Date: Sun, 23 Jun 2024 20:30:35 GMT Connection: close Content-Length: 315 WinRM NTLM Info: OS: Windows Server 2012 R2 OS Build: 6.3.9600 Target Name: WIN-TTIVDJM46CK NetBIOS Domain Name: WIN-TTIVDJM46CK NetBIOS Computer Name: WIN-TTIVDJM46CK DNS Domain Name: WIN-TTIVDJM46CK FQDN: WIN-TTIVDJM46CK ``` ------------------ **8080:** ``` ``` ------------------

Pattern Type

stix

Pattern

[ipv4-addr:value = '89.185.85.102']

Name

2e32c5cea00f8e4c808eae806b14585e8672385df7449d2f6575927537ce8884

Pattern Type

stix

Pattern

[file:hashes.'SHA-256' = '2e32c5cea00f8e4c808eae806b14585e8672385df7449d2f6575927537ce8884']

Name

http://87.121.105.232/bin.ps1

Pattern Type

stix

Pattern

[url:value = 'http://87.121.105.232/bin.ps1']

Name

http://79.110.49.232/plugin3.dll

Pattern Type

stix

Pattern

[url:value = 'http://79.110.49.232/plugin3.dll']

Name

5d8d6871c3d59d855616603f686713ac48bf2351f6182ea282e1d84cbb15b94f

Pattern Type

stix

Pattern

[file:hashes.'SHA-256' = '5d8d6871c3d59d855616603f686713ac48bf2351f6182ea282e1d84cbb15b94f']

Name

0bf87b0e65713bf35c8cf54c9fa0015fa629624fd590cb4ba941cd7cdeda8050

Pattern Type

stix

Pattern

[file:hashes.'SHA-256' = '0bf87b0e65713bf35c8cf54c9fa0015fa629624fd590cb4ba941cd7cdeda8050']

Name

e6e69e85962a402a35cbc5b75571dab3739c0b2f3861ba5853dbd140bae4e4da

Pattern Type

stix

Pattern

[file:hashes.'SHA-256' = 'e6e69e85962a402a35cbc5b75571dab3739c0b2f3861ba5853dbd140bae4e4da']

Name

f4d11b36a844a68bf9718cf720984468583efa6664fc99966115a44b9a20aa33

Pattern Type

stix

Pattern

[file:hashes.'SHA-256' = 'f4d11b36a844a68bf9718cf720984468583efa6664fc99966115a44b9a20aa33']

Name

87.121.105.232

Description

ISP: Constant MOULIN **OS:** - ------------------------- Services: **21:** ``` 220 (vsFTPd 3.0.5) 230 Login successful. 214-The following commands are recognized. ABOR ACCT ALLO APPE CDUP CWD DELE EPRT EPSV FEAT HELP LIST MDTM MKD MODE NLST NOOP OPTS PASS PASV PORT PWD QUIT REIN REST RETR RMD RNFR RNTO SITE SIZE SMNT STAT STOR STOU STRU SYST TYPE USER XCUP XCWD XMKD XPWD XRMD 214 Help OK. 211-Features: EPRT EPSV MDTM PASV REST STREAM SIZE TVFS 211 End ``` ------------------ **22:** ``` SSH-2.0-OpenSSH_8.9p1 Ubuntu-3ubuntu0.7 Key type: ecdsa-sha2-nistp256 Key: AAAAE2VjZHNhLXNoYTItbmlzdHAyNTYAAAAIbmlzdHAyNTYAAABBBLjPdgLmaToiTvcTflp09HC+b/hNzF7QocMNe51BHth0uePsQeVHYennlbxhDYKFTO4sEQmfE/UTUCezwpemzr0= Fingerprint: 16:7a:70:82:1f:bd:40:4c:20:b7:08:8f:b8:c7:a6:e7 Kex Algorithms: curve25519-sha256 curve25519-sha256@libssh.org ecdh-sha2-nistp256 ecdh-sha2-nistp384 ecdh-sha2-nistp521 sntrup761x25519-sha512@openssh.com diffie-hellman-group-exchange-sha256 diffie-hellman-group16-sha512 diffie-hellman-group18-sha512 diffie-hellman-group14-sha256 kex-strict-s-v00@openssh.com 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 ``` ------------------ **80:** ``` HTTP/1.1 200 OK Server: nginx/1.18.0 (Ubuntu) Date: Thu, 09 May 2024 06:26:29 GMT Content-Type: text/html Content-Length: 612 Last-Modified: Mon, 22 Apr 2024 08:40:42 GMT Connection: keep-alive Vary: Accept-Encoding ETag: "6626228a-264" Accept-Ranges: bytes ``` ------------------

Pattern Type

stix

Pattern

[ipv4-addr:value = '87.121.105.232']

Name

79.110.49.232

Description

ISP: 12651980 CANADA INC. **OS:** Ubuntu -------------------------- Services: **21:** ``` 220 (vsFTPd 3.0.5) 230 Login successful. 214-The following commands are recognized. ABOR ACCT ALLO APPE CDUP CWD DELE EPRT EPSV FEAT HELP LIST MDTM MKD MODE NLST NOOP OPTS PASS PASV PORT PWD QUIT REIN REST RETR RMD RNFR RNTO SITE SIZE SMNT STAT STOR STOU STRU SYST TYPE USER XCUP XCWD XMKD XPWD XRMD 214 Help OK. 211-Features: EPRT EPSV MDTM PASV REST STREAM SIZE TVFS 211 End ``` ------------------ **22:** ``` SSH-2.0-OpenSSH_8.9p1 Ubuntu-3ubuntu0.7 Key type: ecdsa-sha2-nistp256 Key: AAAAE2VjZHNhLXNoYTItbmlzdHAyNTYAAAAIbmlzdHAyNTYAAABBBIoQ2OXdOdXjSJ0Ski6bgme5/yKeyo6tq0jFW1Kv3jHVRpHo1naM6c/wvl+7svM7H6HYJznKSpOPFbrb0nqYeEs= Fingerprint: 64:03:13:28:30:6c:dc:f0:5b:b0:b6:4c:26:f4:a1:92 Kex Algorithms: curve25519-sha256 curve25519-sha256@libssh.org ecdh-sha2-nistp256 ecdh-sha2-nistp384 ecdh-sha2-nistp521 sntrup761x25519-sha512@openssh.com diffie-hellman-group-exchange-sha256 diffie-hellman-group16-sha512 diffie-hellman-group18-sha512 diffie-hellman-group14-sha256 kex-strict-s-v00@openssh.com 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 ``` ------------------ **80:** ``` HTTP/1.1 200 OK Server: nginx/1.18.0 (Ubuntu) Date: Fri, 21 Jun 2024 10:51:21 GMT Content-Type: text/html Content-Length: 612 Last-Modified: Mon, 22 Apr 2024 08:40:42 GMT Connection: keep-alive Vary: Accept-Encoding ETag: "6626228a-264" Accept-Ranges: bytes ``` ------------------

Pattern Type

stix

Pattern

[ipv4-addr:value = '79.110.49.232']

Name

89.169.52.37

Description

ISP: AEZA INTERNATIONAL LTD **OS:** - ------------------------- Services: **22:** ``` SSH-2.0-OpenSSH_9.6p1 Ubuntu-3ubuntu13 Key type: ecdsa-sha2-nistp256 Key: AAAAE2VjZHNhLXNoYTItbmlzdHAyNTYAAAAIbmlzdHAyNTYAAABBBP6/CaTdPDZ0rSeivoxAY+vzMJqhhDg7OgGB8t9h1HoObMneDaAezLwInppk5gg5XTccQB9l/wxYBGzFLJdu9nA= Fingerprint: 85:a9:9b:b3:8b:eb:41:94:9a:b0:a3:30:b7:9a:2b:0a 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 ext-info-s kex-strict-s-v00@openssh.com 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 ``` ------------------ **80:** ``` 0$ \x02\x01\x00x\x1f\n\x01T\x04\x00\x04\x00\x8a\x161.3.6.1.4.1.1466.20036 ``` ------------------ **81:** ``` 0$ \x02\x01\x00x\x1f\n\x01T\x04\x00\x04\x00\x8a\x161.3.6.1.4.1.1466.20036 ``` ------------------ **8000:** ``` HTTP/1.1 200 OK Date: Wed, 22 May 2024 11:26:45 GMT Transfer-encoding: chunked ``` ------------------ **8080:** ``` HTTP/1.1 200 OK Date: Wed, 15 May 2024 15:02:42 GMT Transfer-encoding: chunked ``` ------------------

Pattern Type

stix

Pattern

[ipv4-addr:value = '89.169.52.37']

Name

217.182.205.238

Description

ISP: OVH SAS **OS:** - -------------------------- Services: **22:** ``` SSH-2.0-OpenSSH_9.0p1 Ubuntu-1ubuntu8.7 Key type: ecdsa-sha2-nistp256 Key: AAAAE2VjZHNhLXNoYTItbmlzdHAyNTYAAAAIbmlzdHAyNTYAAABBBGEwcd3i2kR6FZH2/fFGMc7FVgArTYJ+WS1OD2INcWbp+A0SAaHHKxxDNYDFt0DFCjitx6wYgN5OoaJBGHHf88k= Fingerprint: ec:ff:55:c5:95:0f:0c:e8:5f:c7:98:5d:f3:80:8a:b0 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 kex-strict-s-v00@openssh.com 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 ``` ------------------ **80:** ``` ``` ------------------ **8080:** ``` ``` ------------------

Pattern Type

stix

Pattern

[ipv4-addr:value = '217.182.205.238']

Intrusion-Set

Name
Water Sigbin

Malware

Name
PureCrypter

Name
XMRig

uses

Name
Name
Name
Name
Name
Name
Name
Name
Name

based-on

Name

Name

indicates

Name

Name

Name

Name

Name

Name

Name

Name

Name

Name

Name

Name

Name

Name

Name

Name

Name

Name

Name

Name

Name

Name

Name

Name

Name

Name

Name

Name

Name

Name

Name

Name
Name
Name
Name
Name
Name
Name
Name
Name
Name
Name

targets

Name

Name

Name

StixFile

Value
2e32c5cea00f8e4c808eae806b14585e8672385df7449d2f6575927537ce8884
b380b771c7f5c2c26750e281101873772e10c8c1a0d2a2ff0aff1912b569ab93
5d8d6871c3d59d855616603f686713ac48bf2351f6182ea282e1d84cbb15b94f
e6e69e85962a402a35cbc5b75571dab3739c0b2f3861ba5853dbd140bae4e4da
f4d11b36a844a68bf9718cf720984468583efa6664fc99966115a44b9a20aa33
0bf87b0e65713bf35c8cf54c9fa0015fa629624fd590cb4ba941cd7cdeda8050

IPv4-Addr

Value
87.121.105.232
217.182.205.238
89.169.52.37
79.110.49.232
89.185.85.102

External References

- https://www.trendmicro.com/content/dam/trendmicro/global/en/research/24/f/water-sigbin-xmrig/ioc-examining-water-sigbin-Infection-routine-leading-to-an-xmrig-cryptominer.txt
- https://www.trendmicro.com/en_us/research/24/f/water-sigbin-xmrig.html
- https://otx.alienvault.com/pulse/667e68ceab594ce0b8387ecc