Federal Agency Compromised by Malicious Cyber Actor | CISA

Published: 2020-10-24 · Archived: 2026-04-05 13:37:30 UTC

CISA became aware—via EINSTEIN, CISA's intrusion detection system that monitors federal civilian networks—of a potential compromise of a federal agency's network. In coordination with the affected agency, CISA conducted an incident response engagement, confirming malicious activity. The following information is derived exclusively from the incident response engagement and provides the threat actor's tactics, techniques, and procedures as well as indicators of compromise that CISA observed as part of the engagement.

Threat Actor Activity

The cyber threat actor had valid access credentials for multiple users' Microsoft Office 365 (O365) accounts and domain administrator accounts, which they leveraged for *Initial Access* [TA0001] to the agency's network (*Valid Accounts* [T1078]). First the threat actor logged into a user's O365 account from Internet Protocol (IP) address `91.219.236[.]166` and then browsed pages on a SharePoint site and downloaded a file (*Data from Information Repositories: SharePoint* [T1213.002]). The cyber threat actor connected multiple times by Transmission Control Protocol (TCP) from IP address `185.86.151[.]223` to the victim organization's virtual private network (VPN) server (*Exploit Public-Facing Application* [T1190]).

CISA analysts were not able to determine how the cyber threat actor initially obtained the credentials. It is possible the cyber actor obtained the credentials from an unpatched agency VPN server by exploiting a known vulnerability—CVE-2019-11510—in Pulse Secure (*Exploitation for Credential Access* [T1212]). In April 2019, Pulse Secure released patches for several critical vulnerabilities—including CVE-2019-11510, which allows the remote, unauthenticated retrieval of files, including passwords.[1] CISA has observed wide exploitation of CVE-2019-11510 across the federal government.[2]

After initial access, the threat actor performed *Discovery* [TA0007] by logging into an agency O365 email account from `91.219.236[.]166` and viewing and downloading help desk email attachments with "Intranet access" and "VPN passwords" in the subject line, despite already having privileged access *(Email Collection* [T1114], *Unsecured Credentials: Credentials In Files* [T1552.001]). (Note: these emails did not contain any passwords.) The actor logged into the same email account via Remote Desktop Protocol (RDP) from IP address `207.220.1[.]3` (*External Remote Services* [T1133]). The actor enumerated the Active Directory and Group Policy key and changed a registry key for the Group Policy (*Account Manipulation* [T1098]). Immediately afterward, the threat actor used common Microsoft Windows command line processes— `conhost` , `ipconfig` , `net` , `query` , `netstat` , `ping` , and `whoami` , `plink.exe` —to enumerate the compromised system and network (*Command and Scripting Interpreter* [T1059], *System Network Configuration Discovery* [T1016]).

The cyber threat actor then attempted multiple times to connect to virtual private server (VPS) IP `185.86.151[.]223` through a Windows Server Message Block (SMB) client. Although they connected and disconnected multiple times, the connections were ultimately successful. During the same period, the actor used an alias secure identifier account they had previously created to log into VPS `185.86.151[.]223` via an SMB

share. The attacker then executed `plink.exe` on a victim file server (*Command and Scripting Interpreter* [T1059]). (`plink.exe` is a command-line version of PuTTy that is used for remote administration.)

The cyber threat actor established *Persistence* [TA0003] and *Command and Control* [TA0011] on the victim network by (1) creating a persistent Secure Socket Shell (SSH) tunnel/reverse SOCKS proxy, (2) running `inetinfo.exe` (a unique, multi-stage malware used to drop files), and (3) setting up a locally mounted remote share on IP address `78.27.70[.]237` (*Proxy* [T1090]). The mounted file share allowed the actor to freely move during its operations while leaving fewer artifacts for forensic analysis. Refer to Threat Actor Malware section for more information about the SSH Tunnel/reverse SOCKS proxy and `inetinfo.exe`.

The cyber threat actor created a local account, which they used for data *Collection* [TA0009], *Exfiltration* [TA0010], *Persistence* [TA0003], and *Command and Control* [TA0011] (*Create Account* [T1136]). The cyber threat actor used the local account to:

- Browse directories on a victim file server (*Data from Shared Network Drive* [T1039]).
- Copy a file from a user's home directory to their locally mounted remote share (*Data Staged* [T1074]).
    - CISA analysts detected the cyber threat actor interacting with other files on users' home directories but could not confirm whether they were exfiltrated.
- Create a reverse SMB SOCKS proxy that allowed connection between an cyber threat actor-controlled VPS and the victim organization's file server (refer to Threat Actor Malware section for more information) (*Proxy* [T1090]).
- Interact with PowerShell module `Invoke-TmpDavFS.psm` (refer to Threat Actor Malware section for more information).
- Exfiltrate data from an account directory and file server directory using `tsclient` (`tsclient` is a Microsoft Windows Terminal Services client) (*Data from Local System* [T1005], *Data from Network Shared Drive* [T1039]).
- Create two compressed Zip files with several files and directories on them (*Archive Collected Data* [T1560]); it is likely that the cyber threat actor exfiltrated these Zip files, but this cannot be confirmed because the actor masked their activity.

See figure 1 for the sequence of the cyber threat actor's tactics and techniques.

Figure 1: Cyber threat actor tactics and techniques

Threat Actor Malware

Persistent SSH Tunnel/Reverse SOCKS Proxy

While logged in as "Administrator," the cyber threat actor created two Scheduled Tasks (see table 1) that worked in concert to establish a persistent SSH tunnel and reverse SOCKS proxy. The proxy allowed connections between an attacker-controlled remote server and one of the victim organization's file servers (*Scheduled Task/Job* [T1053], *Proxy* [T1090]). The Reverse SOCKS Proxy communicated through port 8100 (*Non-Standard Port* [T1571]). This port is normally closed, but the attacker's malware opened it.

Table 1: Scheduled Tasks composing SSH tunnel and reverse SOCKS proxy

Scheduled Task	Description
`ShellExperienceHost.exe`	This task created a persistent SSH tunnel to attacker-controlled remote server `206.189.18[.]189` and employed port forwarding to allow connections from the remote server port 39999 to the victim file server through port 8100. This task was run daily. `ShellExperienceHost.exe` is a version of `plink.exe`, a command-line version of PuTTy that is used for remote administration.
`WinDiag.exe`	This task is a reverse SOCKS proxy that is preconfigured to bind to and listen on TCP port 8100. `WinDiag.exe` received responses through the SSH tunnel and forwarded the responses through port 8100 to the VPS IP address `185.193.127[.]17` over port 443. This task was run on boot. `WinDiag.exe` had compile information that matched the VPS login name

Dropper Malware: inetinfo.exe

The threat actor created a Scheduled Task to run inetinfo.exe (*Scheduled Task/Job* [T1053]). `inetinfo.exe` is a unique, multi-stage malware used to drop files (figure 2). It dropped `system.dll` and `363691858` files and a second instance of `inetinfo.exe`. The `system.dll` from the second instance of `inetinfo.exe` decrypted `363691858` as binary from the first instance of `inetinfo.exe`. The decrypted `363691858` binary was injected into the second instance of `inetinfo.exe` to create and connect to a locally named tunnel. The injected binary then executed shellcode in memory that connected to IP address `185.142.236[.]198`, which resulted in download and execution of a payload.

Figure 2: Dropper malware `inetinfo.exe`

The cyber threat actor was able to overcome the agency's anti-malware protection, and `inetinfo.exe` escaped quarantine. CISA analysts determined that the cyber threat actor accessed the anti-malware product's software license key and installation guide and then visited a directory used by the product for temporary file analysis. After accessing this directory, the cyber threat actor was able to run `inetinfo.exe` (*Impair Defenses: Disable or Modify Tools* [T1562.001]).

Reverse SMB SOCKS Proxy

PowerShell script `HardwareEnumeration.ps1` created a reverse SMB SOCKS proxy that allowed connection between attacker-controlled VPS IP `185.193.127[.]18` and the victim organization's file server over port 443

(*Command and Scripting Interpreter: Power Shell* [T1059.001], *Proxy* [T1090]). PowerShell script `HardwareEnumeration.ps1` was executed daily via a Scheduled Task (*Scheduled Task/Job* [T1053]).

`HardwareEnumeration.ps1` is a copy of `Invoke-SocksProxy.ps1`, a free tool created and distributed by a security researcher on GitHub.[3] `Invoke-SocksProxy.ps1` creates a reverse proxy from the local machine to attacker infrastructure through SMB TCP port 445 (*Non-Standard Port* [T1571]). The script was likely altered with the cyber threat actor's configuration needs.

PowerShell Module: `invoke-TmpDavFS.psm`

`invoke-TmpDavFS.psm` is a PowerShell module that creates a Web Distributed Authoring and Versioning (WebDAV) server that can be mounted as a file system and communicates over TCP port 443 and TCP port 80. `invoke-TmpDavFS.psm` is distributed on GitHub.[4]

Summary

This Analysis Report uses the MITRE Adversarial Tactics, Techniques, and Common Knowledge (ATT&CK®) framework. See the ATT&CK for Enterprise framework for all referenced threat actor tactics and techniques.

The Cybersecurity and Infrastructure Security Agency (CISA) responded to a recent threat actor's cyberattack on a federal agency's enterprise network. By leveraging compromised credentials, the cyber threat actor implanted sophisticated malware—including multi-stage malware that evaded the affected agency's anti-malware protection—and gained persistent access through two reverse Socket Secure (SOCKS) proxies that exploited weaknesses in the agency's firewall.

For a downloadable copy of IOCs, see: AA20-268A.stix.

Solution

Indicators of Compromise

CISA analysts identified several IP addresses involved in the multiple stages of the outlined attack.

- `185.86.151[.]223` – Command and Control (C2)
- `91.219.236[.]166` – C2
- `207.220.1[.]3` – C2
- `78.27.70[.]237` – Data Exfiltration
- `185.193.127[.]18` – Persistence

Monitor Network Traffic for Unusual Activity

CISA recommends organizations monitor network traffic for the following unusual activity.

- Unusual open ports (e.g., port 8100)
- Large outbound files
- Unexpected and unapproved protocols, especially outbound to the internet (e.g., SSH, SMB, RDP)

If network defenders note any of the above activity, they should investigate.

Prevention

CISA recommends organizations implement the following recommendations to protect against activity identified in this report.

Deploy an Enterprise Firewall

Organizations should deploy an enterprise firewall to control what is allowed in and out of their network.

If the organization chooses not to deploy an enterprise firewall, they should work with their internet service provider to ensure the firewall is configured properly.

Block Unused Ports

Organizations should conduct a survey of the traffic in and out of their enterprise to determine the ports needed for organizational functions. They should then configure their firewall to block unnecessary ports. Organization should develop a change control process to make control changes to those rules. Of special note, unused SMB, SSH, and FTP ports should be blocked.

Additional Recommendations

CISA recommends organizations implement the following best practices.

- Implement multi-factor authentication, especially for privileged accounts.
- Use separate administrative accounts on separate administration workstations.
- Implement the principle of least privilege on data access.
- Secure RDP and other remote access solutions using multifactor authentication and “jump boxes” for access.
- Deploy and maintain endpoint defense tools on all endpoints.
- Keep software up to date.

References

[1] Pulse Secure Security Advisory SA44101 Out-of-Cycle Advisory: Multiple vulnerabilities resolved in Pulse Connect Secure / Pulse Policy Secure 9.0RX

[3] GitHub Repository for Invoke-SocksProxy

[4] GitHub Repository for Invoke-TmpDavFS

Revisions

September 24, 2020: Initial Version