BLOG

SECURONIX THREAT LABS SECURITY ADVISORY: NEW SEO#LURKER ATTACK CAMPAIGN: THREAT ACTORS USE SEO POISONING AND FAKE GOOGLE ADS TO LURE VICTIMS INTO INSTALLING MALWARE

THREAT RESEARCH

Share

By Securonix Threat Research: Den Iuzvyk, Tim Peck, Oleg Kolesnikov

tldr:

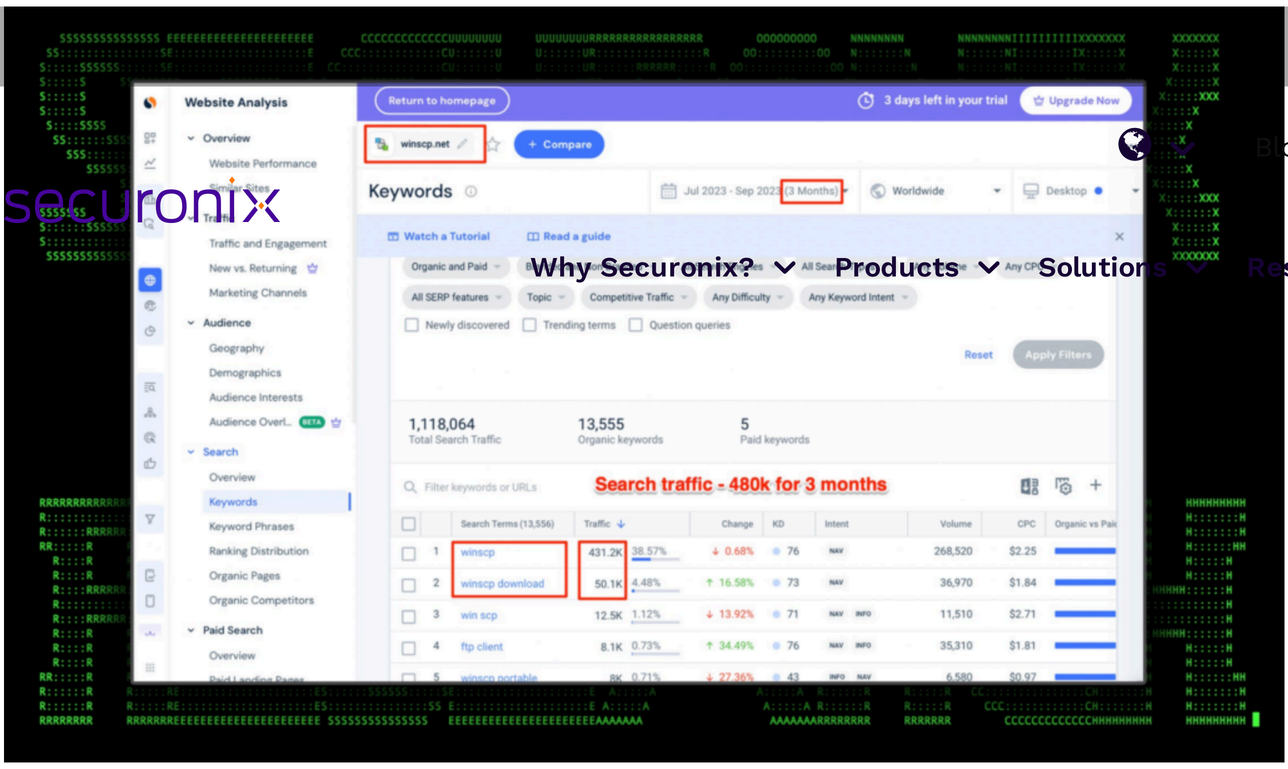
An interesting ongoing SEO poisoning/malvertising campaign leveraging WinSCP lures along with a stealthy infection chain lures victims into installing malware (alongside the legitimate WinSCP software). Attackers are likely leveraging dynamic search ads which let threat actors inject their own malicious code while mimicking legitimate sources like Google search pages.

A rather steep uptick in malicious advertising (malvertising) has been observed, especially in the last year which involves threat actors paying either your favorite **search engine** or **social networking sites** for ad space in order to promote malware in prominent locations.

The Securonix Threat Research team has been tracking an ongoing campaign SEO#LURKER which targets “WinSCP” keywords in Google Search results. WinSCP is a popular SSH/SCP connection platform which has established a huge user base over the years making it a lucrative target for threat actors. It’s highly likely that WinSCP is not the only downloadable software being targeted by these threat actors to serve malicious advertisements.

Given its popularity, WinSCP has been a target in the past, along with a host of other popular software downloads. In July this year, we observed some of these infections resulting in the victim machine being **infected with ransomware** using similar approaches.

Today, malvertising and SEO poisoning continues to remain popular. As seen in the figure below, the homepage for the popular software tracks about 430K searches in the past three months, and another 50k for the download page itself. Another factor to consider is that this method is also especially lucrative as according to the data below, the CPC (cost per click) ranges between $2.25 and $1.84 USD.



□ Newly discovered □ Trending terms □ Question queries

1,118,064 Total Search Traffic
13,555 Organic keywords
5 Paid keywords

Search traffic - 480k for 3 months

	Search Terms (13,556)	Traffic		Change	KD	Intent	Volume	CPC
1	winscp	431.2K	38.57%	↓ 0.68%	76	NAV	268,520	$2.25
2	winscp download	50.1K	4.48%	↑ 16.58%	73	NAV	36,970	$1.84
3	win scp	12.5K	1.12%	↓ 13.92%	71	NAV INFO	11,510	$2.71
4	ftp client	8.1K	0.73%	↑ 34.49%	76	NAV INFO	35,310	$1.81
5	winscp portable	8K	0.71%	↓ 27.36%	43	INFO NAV	6,580	$0.97

Figure 1: Google traffic for winscp.net (three months)

This type of malvertising is extremely effective since a threat actor can place a link masquerading as a legitimate website at the very top of the Google search results, exactly where a user might click after searching for a specific tool to download.

Attack chain overview

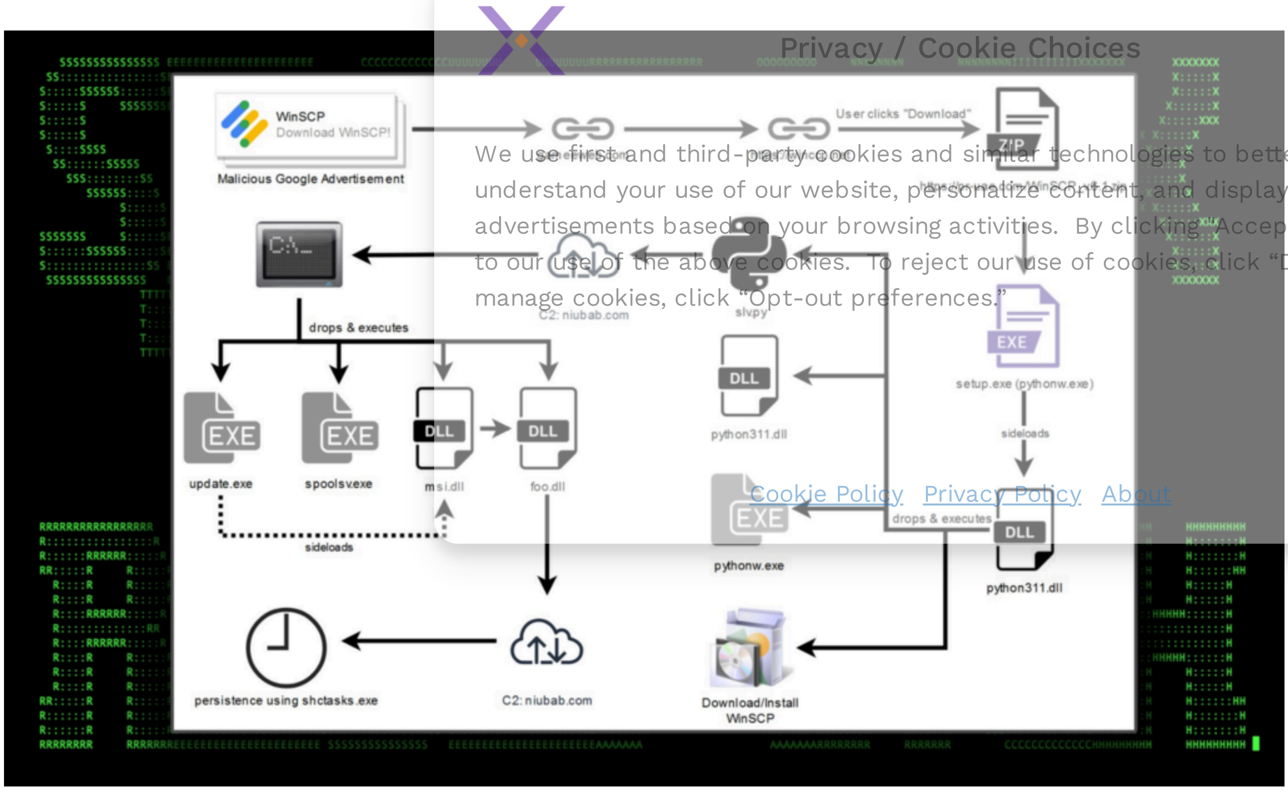


Figure 2: SEO#LURKER attack chain example

Malvertising infrastructure analysis

The attack begins with the user searching for “WinSCP” in Google. The ad appears before the legitimate website for WinSCP which is https://winscp.net. The malicious advertisement directs the user to a compromised WordPress website gameeweb[.]com which redirects the user to an attacker-controlled phishing site. The threat actors registered a similar domain and an almost identical looking website to trick users into downloading their malware.

Malicious DSA ads in Google Search

It’s also highly likely that the attackers leverage dynamic search ads (DSA). This type of advertising allows threat actors to inject their own malicious code into another, say if the site becomes compromised, or create their own based on the content of their own phishing website. These kinds of ads will promote their own malicious software as if they were the legitimate source from Google search pages.

This type of advertising has been used **extensively in the past** by threat actors and will likely continue despite Google's efforts to mitigate it.

Malvertising infrastructure of the SEO#LURKER campaign

The threat actors registered a domain similar to the legitimate winscp.net in order to appear as legitimate as possible. As you can see in the figure below phishing domain hxxps://winccp[.]net (left) and the legitimate website are nearly identical:

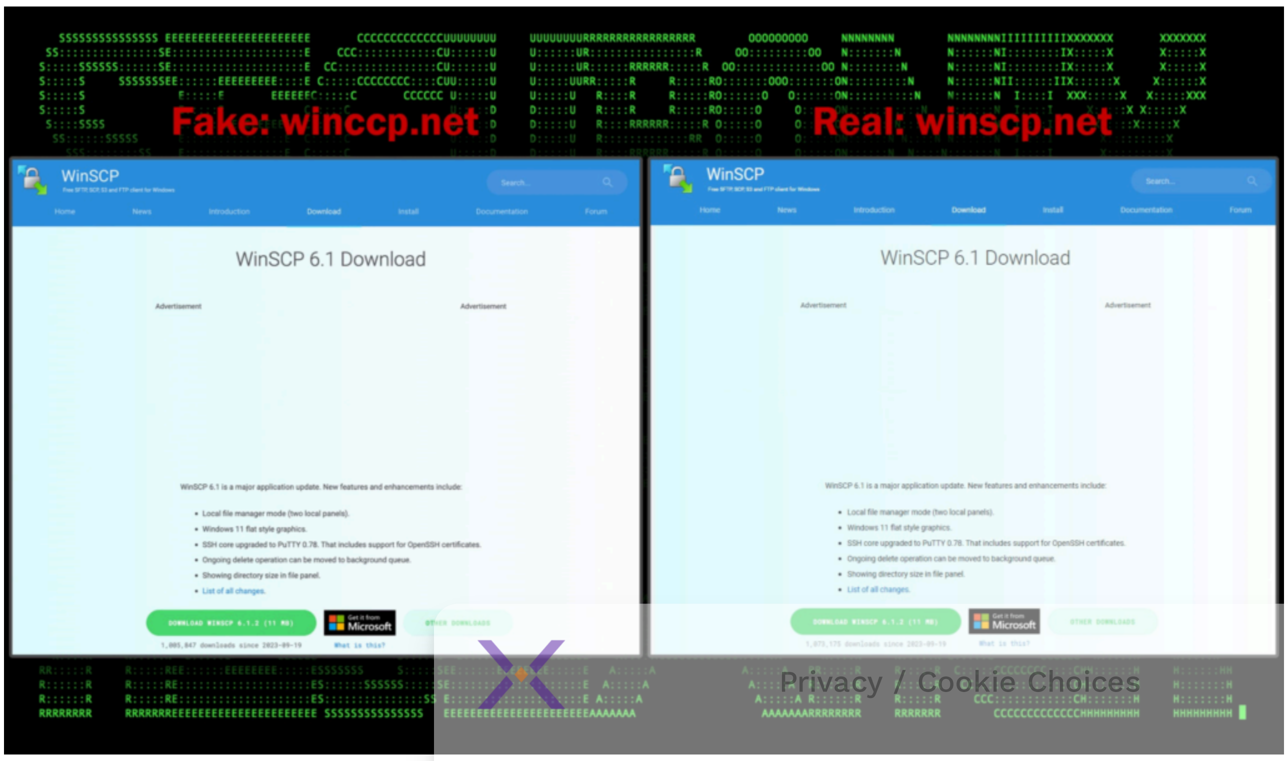
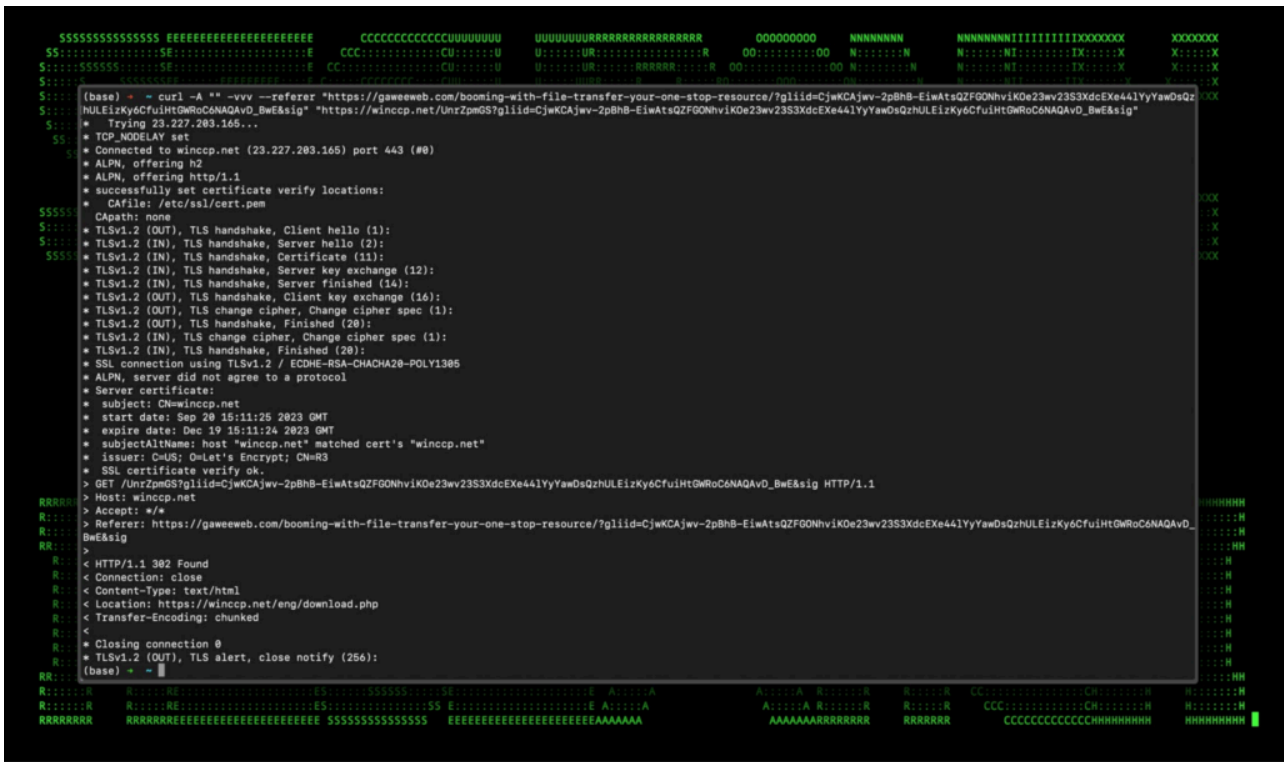


Figure 3: lure website versus legitimate winscp website download page

When the user clicks the malicious advertisement, they are brought to the domain gaweeweb[.]com which immediately redirects to hxxps://winccp[.]net. The fake WinSCP phishing site follows a similar naming to previous documented campaigns which used the domain winsccp[.]net rather than what we are observing currently.

When the "DOWNLOAD WINSCP 6.1.2" link is clicked, the file is downloaded from a seemingly legitimate Japanese website to download the malicious zip package stored at the root of the website: hxxps://pr-uae[.]com/WinSCP_v.6.1.zip.

Traffic from the gaweeweb[.]com website to the fake winsccp[.]net website relies on a correct referrer header being set properly. If the referrer is incorrect, the user is "**Rick Rolled**" and is sent to the infamous Rick Astley YouTube video.



```
(base) → ~ curl -A "" -vvv --referer "https://gaweeweb.com/booming-with-file-transfer-your-one-stop-resource/?gliid=CjwKCAjwv-2pBhB-EiwAtsQZFGONhviKOe23wv23S3XdcEXe44lYyYawDsQz
hULEizKy6CfuiHtGWRoC6NAQAvD_BwE&sig" "https://winccp.net/UnrZpmGS?gliid=CjwKCAjwv-2pBhB-EiwAtsQZFGONhviKOe23wv23S3XdcEXe44lYyYawDsQzhULEizKy6CfuiHtGWRoC6NAQAvD_BwE&sig"
*   Trying 23.227.203.165...
* TCP_NODELAY set
* Connected to winccp.net (23.227.203.165) port 443 (#0)
* ALPN, offering h2
* ALPN, offering http/1.1
* successfully set certificate verify locations:
*   CAfile: /etc/ssl/cert.pem
  CApath: none
* TLSv1.2 (OUT), TLS handshake, Client hello (1):
* TLSv1.2 (IN), TLS handshake, Server hello (2):
* TLSv1.2 (IN), TLS handshake, Certificate (11):
* TLSv1.2 (IN), TLS handshake, Server key exchange (12):
* TLSv1.2 (IN), TLS handshake, Server finished (14):
* TLSv1.2 (OUT), TLS handshake, Client key exchange (16):
* TLSv1.2 (OUT), TLS change cipher, Change cipher spec (1):
* TLSv1.2 (OUT), TLS handshake, Finished (20):
* TLSv1.2 (IN), TLS change cipher, Change cipher spec (1):
* TLSv1.2 (IN), TLS handshake, Finished (20):
* SSL connection using TLSv1.2 / ECDHE-RSA-CHACHA20-POLY1305
* ALPN, server did not agree to a protocol
* Server certificate:
*  subject: CN=winccp.net
*  start date: Sep 20 15:11:25 2023 GMT
*  expire date: Dec 19 15:11:24 2023 GMT
*  subjectAltName: host "winccp.net" matched cert's "winccp.net"
*  issuer: C=US; O=Let's Encrypt; CN=R3
*  SSL certificate verify ok.
> GET /UnrZpmGS?gliid=CjwKCAjwv-2pBhB-EiwAtsQZFGONhviKOe23wv23S3XdcEXe44lYyYawDsQzhULEizKy6CfuiHtGWRoC6NAQAvD_BwE&sig HTTP/1.1
> Host: winccp.net
> Accept: */*
> Referer: https://gaweeweb.com/booming-with-file-transfer-your-one-stop-resource/?gliid=CjwKCAjwv-2pBhB-EiwAtsQZFGONhviKOe23wv23S3XdcEXe44lYyYawDsQzhULEizKy6CfuiHtGWRoC6NAQAvD_
BwE&sig
>
< HTTP/1.1 302 Found
< Connection: close
< Content-Type: text/html
< Location: https://winccp.net/eng/download.php
< Transfer-Encoding: chunked
<
* Closing connection 0
* TLSv1.2 (OUT), TLS alert, close notify (256):
(base) → ~ 
```

Zip file content analysis: WinSCP_v.6.1.zip

As the victim user would suspect, the zip file is downloaded and once opened, the user is presented with a simple setup.exe file, along (along with a bunch of hidden files). It should be noted that the legitimate WinSCP file download does not download a zip package, rather a single installer.

Figure 4: WinSCP_v.6.1.zip file contents (hidden files in red)

Inside the zip file there are several files, several of which serve no real purpose. Setup.exe is a renamed legitimate "pythonw.exe" which is digitally signed. The executable is designed to launch GUI or no-UI-at-all scripts so it will never open a popup window. The second file worth mentioning is python311.dll. This is not the legitimate DLL used by Python processes, rather it is built by the attacker which when executed kicks off the malicious code execution.

When the user runs setup.exe, the hidden python311.dll is sideloaded and its code contents are executed by the setup/python process. No actual python code is executed at this point. Eventually the pythonw.exe process will crash but not before running the malicious code from the DLL.

Code execution: "python311.dll"

When setup.exe is executed and python311.dll is sideloaded, the process will perform several key functions. First, it downloads and executes a legitimate WinSCP installer. The file is saved to the user's downloads folder and executed thus kicking off the WinSCP installation. This reinforces legitimacy along with the user's initial intent of downloading and installing WinSCP.

After the installer executes, the malware will create a bunch of files in the user's %LOCALAPPDATA% directory. The DLL acts as a dropper and loads the following files onto the user's file system:

%LOCALAPPDATA%\Notepad\dllhost.exe

%LOCALAPPDATA%\Notepad\libcrypto-1_1.dll

%LOCALAPPDATA%\Notepad\libffi-7.dll

%LOCALAPPDATA%\Notepad\libssl-1_1.dll

%LOCALAPPDATA%\Notepad\python.exe

%LOCALAPPDATA%\Notepad\python3.dll

%LOCALAPPDATA%\Notepad\python310.dll

%LOCALAPPDATA%\Notepad\pythonw.exe

%LOCALAPPDATA%\Notepad\spoolsv.exe

%LOCALAPPDATA%\Notepad\slv.py

%LOCALAPPDATA%\Notepad\sqlite3.dll

%LOCALAPPDATA%\Notepad\vcruntime140.dll

%LOCALAPPDATA%\Notepad\vcruntime140_1.dll

%LOCALAPPDATA%\Programs\Microsoft\Office\msi.dll

%LOCALAPPDATA%\Programs\Microsoft\Office\update.exe

%LOCALAPPDATA%\Programs\Microsoft\TrustedInstaller\msi.dll

%LOCALAPPDATA%\Programs\Microsoft\TrustedInstaller\update.exe

The files created in the "Notepad" directory are all legitimate Python files and library components for running Python scripts. Setup.exe lastly runs the following command which executes the Python code contained in the "slv.py" file.

C:\Users\[redacted_username]\AppData\Local\Notepad\pythonw.exe

C:\Users\flareon\AppData\Local\Notepad\slv.py

At this point the user is probably in the middle of installing WinSCP, unaware of the malware currently executing in the background on their system.

Code execution: slv.py

Now we're actually executing malicious Python code. The Python file is heavily obfuscated and all but one line of the file are completely useless. These useless lines appear to attempt evading detection by creating a bunch of useless functions and defining unused variables along with randomly inserted comments.

Figure 5: File content sample of the slv.py file

The line of code we're interested in is quite large at over 240,000 characters containing raw Python bytecode. In summary, this line takes the long string contained in single quotes and then decompresses it using the zlib library. Next the decompressed object is marshaled which is done to serialize and deserialize Python code objects. It then executes the results of the payload via exec().

Immediately after executing the contained code within the Python script, it will immediately begin beaconing to niubab[.]com (141.98.6[.]195) on port 8443. At this point the attackers have command and control capabilities over the infected host.

Another similar Python file was then dropped into the %LOCALAPPDATA%\Notepad directory called wo15.py. Once executed, the file began beaconing to 194.180.48[.]42 on 443. Both slv.py and wo15.py appear to function in the same manner.

Observed commands: manual session

With the system compromised and C2 established, we observed the following enumeration commands (spawning from pythonw.exe process):

- whoami /all
- whoami /priv
- net localgroup administrators
- net group “domain controllers” /domain
- net group “domain computers” /domain
- net group “domain admins” /domain

Moving up the infection chain…

Next, three new directories were again created, this time from within the newly created %LOCALAPPDATA%\JetBrains\ directory. These appeared to be randomized containing similar files within:

- \jetbrains\qdslli\update.exe
- \jetbrains\qdslli\spoolsv.exe
- \jetbrains\qdslli\msi.dll
- \jetbrains\qdslli\foo.dll
- \jetbrains\erouud\update.exe
- \jetbrains\erouud\spoolsv.exe
- \jetbrains\erouud\msi.dll
- \jetbrains\erouud\sex.dll
- \jetbrains\pdyjlq\update.exe
- \jetbrains\pdyjlq\spoolsv.exe
- \jetbrains\pdyjlq\mix.dll
- \jetbrains\pdyjlq\sex.dll

The files appear to be used for persistence and function similarly to that of the Python files we covered earlier.

The file update.exe is a legitimate msiexec binary that was renamed. When executed it will side load msi.dll which should normally exist in C:\Windows\System32. The DLL is sideloaded which then is rewritten to load the second 3-letter DLL in the folder which is the malicious payload.

The use of msiexec.exe to sideload msi.dll is not new and has been **documented in the pas** by the process exhibited some classic Meterpreter characteristics and was built using known obfuscation methods. Beaconing to the following IP address was observed after execution:

194.169.175[.]221:8443

Once again we observed similar enumeration commands coming from the “update.exe” process as from the Python beacons earlier.

Persistence

Once infected, persistence on the host was established using scheduled tasks. We observed three unique tasks being created.

Figure 6: scheduled tasks persistence

As you can see in the figure above, each of the scheduled tasks attempts to masquerade as a legitimate scheduled task starting with “onedrive standalone update task” and a security and group identifier that doesn’t exist.

The scheduled task calls update.exe (renamed msiexec.exe) followed by a long string which appears to be ignored by msiexec during our testing. This is likely an attempt to break detections. The process pythonw.exe is also executed along with the slv.py payload.

Figure 7: scheduled task details

Wrapping up…

Given the fact that the attackers were leveraging Google Ads to disperse malware, it can be inferred that the targets are limited to anyone seeking WinSCP software. The geoblocking used on the site hosting the malware suggests that those in the US are victims of this attack.

The overall attack chain is quite unusual and relies heavily upon several rounds of DLL sideloading, and malicious compiled Python files which results in a rather complex attack chain ending with persistence via scheduled tasks.

C2 and infrastructure

The SEO#LURKER consisted of several C2 IP addresses using either port 443 or 8443. The domain pr-uae[.]com appears to be compromised and is currently hosting the malicious WinSCP zip file at the root of the domain.

All three of the IP addresses are hosted in the Netherlands, registered to RIPE NCC.

C2 Address	Description
gaweeweb[.]com	Compromised WordPress website
pr-uae[.]com hxxps://pr-uae[.]com/WinSCP_V.6.1.zip	Download initial lure payload
niubab[.]com	C2 Comms over port 8443/443 by update.exe
141.98.6[.]195	C2 Comms over port 8443 by pythonw.exe
194.180.48[.]42	C2 Comms over port 443 by pythonw.exe
194.169.175[.]221	C2 Comms over port 8443 by update.exe

Securonix recommendations and mitigations

With malvertising becoming more and more popular, it's critical to scrutinize web results thoroughly especially when searching for software to download and install.

- Check that files are downloaded from reputable sites, always check the URL that it matches the intended software
- Verify file download that it matches the checksum provided by the trusted source (**guide**)
- Monitor common malware staging directories, especially the user's "\Appdata\Local" which was used in this attack campaign
- Deploy additional process-level logging such as **Sysmon** and **PowerShell logging** for additional log detection coverage
- Securonix customers can scan endpoints using the Securonix Seeder Hunting Queries below

MITRE ATT&CK matrix

Tactic	Technique

Resource Development	T1583.008: Acquire Infrastructure: Malvertising
Execution	T1204.001: User Execution: Malicious Link T1204.002: User Execution: Malicious File T1059: Command and Scripting Interpreter T1059.006: Command and Scripting Interpreter: Python
Defense Evasion	T1574.002: Hijack Execution Flow: DLL Side-Loading T1036.004: Masquerading: Masquerade Task or Service T1036.005: Masquerading: Match Legitimate Name or Location
Command and Control	T1105: Ingress Tool Transfer T1573.001: Encrypted Channel: Symmetric Cryptography T1219: Remote Access Software
Persistence	T1053: Scheduled Task/Job

Analyzed file hashes

File Name	SHA256 (IoC)
WinSCP_6.1.2-Setup.zip	6EB977F30B1D54E450118381F345DB2546613D1AF5D4D097B0E8D4769962A581
setup.exe	24385D352B83222DC5AB92FA57B6649854ECD74DE378E279D8AC20A0B3B16009
python311.dll	BAFEDBA6E75D64E7820048E7ED6625451D22382A4F5F77F822DAEF225CB6EAE8
msi.dll	EE895FF48DA45393C4573E9E9E5C062DBC0F747BEFF89B7DDC53BDE41F773A41 1CE2B14E35AD00D6029DE24D192CF3CB3DDA09D22CC6851E9EFB0DA1B3EEBC1A EEA0EF246B99590072FCBC004724FC98613E4BD31F05345DE7FBB0EED3B7BBFB
sex.dll	D5A5B4CB023DB243D1A65489B75A3252948252F21D9609E6C65A059D7ACF0566
mix.dll	AE346633270EB0FB0ED97E0B2E840AFD333D2EFA967CF90FF35CF55FD4B3D86E
foo.dll	3164D0A9CB3D1088EA89F1429B51DBEAB4EFA44E200F0CB9F7908D0AA9D8159B
slv.py	D4CEF07C9BA72CD4ED63F6FE7B3C86188CBF7DC9E2988791907186E744A235B8
wo15.py	B663EA82D4BDB6DE13264E637F817F08AD6EB107606E2385485E467EC4F7F54C

Relevant provisional Securonix detections

- EDR-ALL-1100-RU
- EDR-ALL-185-ER
- EDR-ALL-1169-RU
- EDR-ALL-1262-RU

Relevant hunting/Spotter queries (be sure to remove square brackets “[ ]”)

- index = activity AND rg_functionality = “Web Proxy” AND (destinationaddress = “141.98.6[.]195” OR destinationaddress = “194.180.48[.]42” OR destinationaddress = “194.169.175[.]221”) AND (destinationport = “443” OR destinationport = “8443”)

- index = activity AND rg_functionality = “Endpoint Management Systems” AND (deviceaction = “Network connection detected” OR deviceaction = “Network connection detected (rule: NetworkConnect)”) AND (destinationhostname CONTAINS “gaweeweb[.]com” OR destinationhostname CONTAINS “winccp[.]net” OR destinationhostname CONTAINS “niubab[.]com”)
- index = activity AND rg_functionality = “Endpoint Management Systems” AND baseeventid = “7” AND customstring67 ENDS WITH “\msi.dll” AND (customstring67 NOT CONTAINS “\windows\system32” OR customstring67 NOT CONTAINS “\windows\syswow64”)
- index = activity AND rg_functionality = “Endpoint Management Systems” AND (deviceaction = “ProcessCreate” OR deviceaction = “Process Create” OR deviceaction = “Process Create (rule: ProcessCreate)” OR deviceaction = “ProcessRollup2” OR deviceaction = “Procstart” OR deviceaction = “Process” OR deviceaction = “Trace Executed Process”) AND destinationprocessname = “schtasks.exe” AND (resourcecustomfield1 CONTAINS “\Appdata\Local” OR resourcecustomfield1 CONTAINS “\Appdata\Roaming”)

References:

1. Malvertising Used as Entry Vector for BlackCat, Actors Also Leverage SpyBoy Terminator
https://www.trendmicro.com/en_us/research/23/f/malvertising-used-as-entry-vector-for-blackcat-actors-also-lever.html
2. Malvertisers Using Google Ads to Target Users Searching for Popular Software
https://thehackernews.com/2023/10/malvertisers-using-google-ads-to-target.html
3. Malvertising in Facebook: Analysis, Quantification and Solution
https://www.mdpi.com/2079-9292/9/8/1332
4. Malvertising via Dynamic Search Ads delivers malware bonanza
https://www.malwarebytes.com/blog/threat-intelligence/2023/10/malvertising-via-dynamic-search-ads-delivers-malware-bonanza
5. Hijacking DLLs in Windows
https://www.wietzebeukema.nl/blog/hijacking-dlls-in-windows
6. How to Check a File Checksum: A Step-by-Step Guide
https://codesigningstore.com/how-to-check-file-checksum
7. Rickrolling.
https://en.wikipedia.org/wiki/Rickrolling

← PREVIOUS ARTICLE
Securonix Threat Labs Monthly Intelligence Insights – October 2023

NEXT ARTICLE →
The SIEM Alternatives Fallacies

Related Resource

View all →

securonix

Blog Contact Us REQUEST A DEMO

Why Securonix? Products Solutions Resources Partners Company

THREAT RESEARCH

Details and Guidance on New "FortiJump" Vulnerability or CVE-2024-47575

Learn More

THREAT RESEARCH

Securonix Threat Labs Monthly Intelligence Insights – September 2024

Learn More

THREAT RESEARCH

SHROUDED#SLEEP: A Deep Dive into North Korea's Ongoing Campaign Against Southeast Asia

Learn More

THREAT RESEARCH

Secur Labs Summer Intelligence Insights – 2024

Learn More