Fleming College

LEARN | BELONG | BECOME

OWASP Vulnerability Assessment Report

Team 5 - Red Team

Ranvijay Singh

Tuan Khanh Vu

Jaspreet Singh Saini

Vishavjit Singh

Karan Tank

Contents

Target(s) Scanned:	Team 6 (Blue Team) Client Machine – 172.31.9.206 Domain Controller – 172.31.9.216 Log server – 172.31.9.193
Report Generated:	12/10/2019

Introduction

Students studying in Sir Sandford Fleming college located in Peterborough, Ontario enrolled in Computer Investigation and Security (CSI) program, developed this document in furtherance of its statutory responsibilities under the course of Advanced Penetration Testing and Firewall and Intrusion Detection System taught by Professor Charles Baker from September 2019 to December 2019.

Team 5's Red Team was asked to conduct penetration testing and vulnerability assessment on a network established by Team 6's Blue Team consisting of a domain controller, a client workstation and a log sever. The following activities took place under full consent and authorization of the Blue Team and Course Tutor, Prof. Charles Baker.

A vulnerability assessment followed by a penetration test was conducted by Team 5's Red Team against the network setup by Team6's Blue Team. The purpose of this assessment was to identify and quantify vulnerabilities or potential threats in the systems and attempt to penetrate before they are exploited by attackers.

Executive Summary

This section provides an overview of the vulnerability assessment results and shows the distribution of vulnerabilities by severity level and by category.

Security Threat Level

This graph presents the security threat level based on the vulnerabilities identified by Red Team. The "Threat Level" is classified as being of Informational, Low, Medium or High severity.

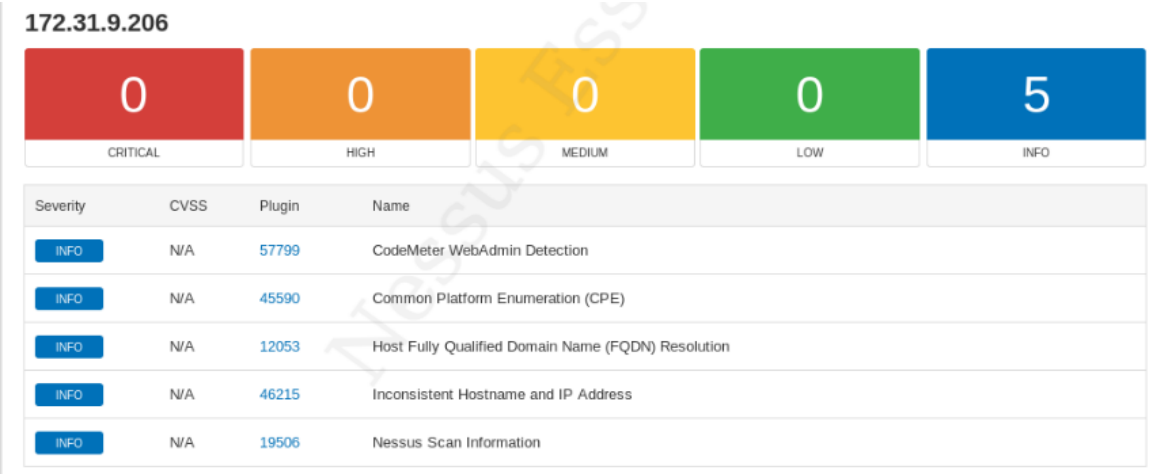
Targets	Information	Low	Medium	High
Client Machine – 172.31.9.206	5	0	0	0
Domain Controller – 172.31.9.216	32	0	0	0
Log server – 172.31.9.193	11	0	0	0

Technical Management Overview

Nessus scan report:

Below is a scan conduced on the Client machine and Domain Controller.

- Executive report for client.



CodeMeter WebAdmin Detection

Info Nessus Plugin ID 57799

Synopsis

The remote web server hosts a copy protection application.

Description

The remote web server hosts CodeMeter WebAdmin, a web-based tool for working with CodeMeter hardware and software-based copy protection technology.

Inconsistent Hostname and IP Address

Info Nessus Plugin ID 46215

Synopsis

The remote host's hostname is not consistent with DNS information.

Description

The name of this machine either does not resolve or resolves to a different IP address. This may come from a badly configured reverse DNS or from a host file in use on the Nessus scanning host. As a result, URLs in plugin output may not be directly usable in a web browser.

Solution

Fix the reverse DNS or host file.

- Executive report for the Domain controller.

172.31.9.216

CRITICAL	HIGH	MEDIUM	LOW	INFO
0	0	0	0	32

Severity	CVSS	Plugin	Name
INFO	N/A	45590	Common Platform Enumeration (CPE)
INFO	N/A	10736	DCE Services Enumeration
INFO	N/A	11002	DNS Server Detection
INFO	N/A	54615	Device Type
INFO	N/A	35716	Ethernet Card Manufacturer Detection
INFO	N/A	86420	Ethernet MAC Addresses
INFO	N/A	43111	HTTP Methods Allowed (per directory)
INFO	N/A	10107	HTTP Server Type and Version
INFO	N/A	12053	Host Fully Qualified Domain Name (FQDN) Resolution
INFO	N/A	24260	HyperText Transfer Protocol (HTTP) Information
INFO	N/A	46215	Inconsistent Hostname and IP Address
INFO	N/A	43829	Kerberos Information Disclosure
INFO	N/A	25701	LDAP Crafted Search Request Server Information Disclosure
INFO	N/A	20870	LDAP Server Detection
INFO	N/A	117886	Local Checks Not Enabled (info)

HyperText Transfer Protocol (HTTP) Information

Info Nessus Plugin ID 24260

Synopsis

Some information about the remote HTTP configuration can be extracted.

Description

This test gives some information about the remote HTTP protocol - the version used, whether HTTP Keep-Alive and HTTP pipelining are enabled, etc...

This test is informational only and does not denote any security problem.

LDAP Crafted Search Request Server Information Disclosure

Info Nessus Plugin ID 25701

Synopsis

It is possible to discover information about the remote LDAP server.

Description

By sending a search request with a filter set to 'objectClass=*', it is possible to extract information about the remote LDAP server.

DCE Services Enumeration

Info Nessus Plugin ID 10736

Synopsis

A DCE/RPC service is running on the remote host.

Description

By sending a Lookup request to the portmapper (TCP 135 or epmapper PIPE) it was possible to enumerate the Distributed Computing Environment (DCE) services running on the remote port. Using this information, it is possible to connect and bind to each service by sending an RPC request to the remote port/pipe.

Kerberos Information Disclosure

Info Nessus Plugin ID 43829

Synopsis

The remote Kerberos server is leaking information.

Description

Nessus was able to retrieve the realm name and/or server time of the remote Kerberos server.

HTTP Methods Allowed (per directory)

Info Nessus Plugin ID 43111

Synopsis

This plugin determines which HTTP methods are allowed on various CGI directories.

Description

By calling the OPTIONS method, it is possible to determine which HTTP methods are allowed on each directory.

The following HTTP methods are considered insecure:

PUT, DELETE, CONNECT, TRACE, HEAD

Many frameworks and languages treat 'HEAD' as a 'GET' request, albeit one without any body in the response. If a security constraint was set on 'GET' requests such that only 'authenticatedUsers' could access GET requests for a particular servlet or resource, it would be bypassed for the 'HEAD' version. This allowed unauthorized blind submission of any privileged GET request.

As this list may be incomplete, the plugin also tests - if 'Thorough tests' are enabled or 'Enable web applications tests' is set to 'yes' in the scan policy - various known HTTP methods on each directory and considers them as unsupported if it receives a response code of 400, 403, 405, or 501.

Note that the plugin output is only informational and does not necessarily indicate the presence of any security vulnerabilities.

- Executive report for Log Server

Ip address – 172.31.9.193

Nessus SYN scanner

Info Nessus Plugin ID 11219

Synopsis

It is possible to determine which TCP ports are open.

Description

This plugin is a SYN 'half-open' port scanner. It shall be reasonably quick even against a firewalled target.

Note that SYN scans are less intrusive than TCP (full connect) scans against broken services, but they might cause problems for less robust firewalls and also leave unclosed connections on the remote target, if the network is loaded.

Solution

Protect your target with an IP filter.

Traceroute Information

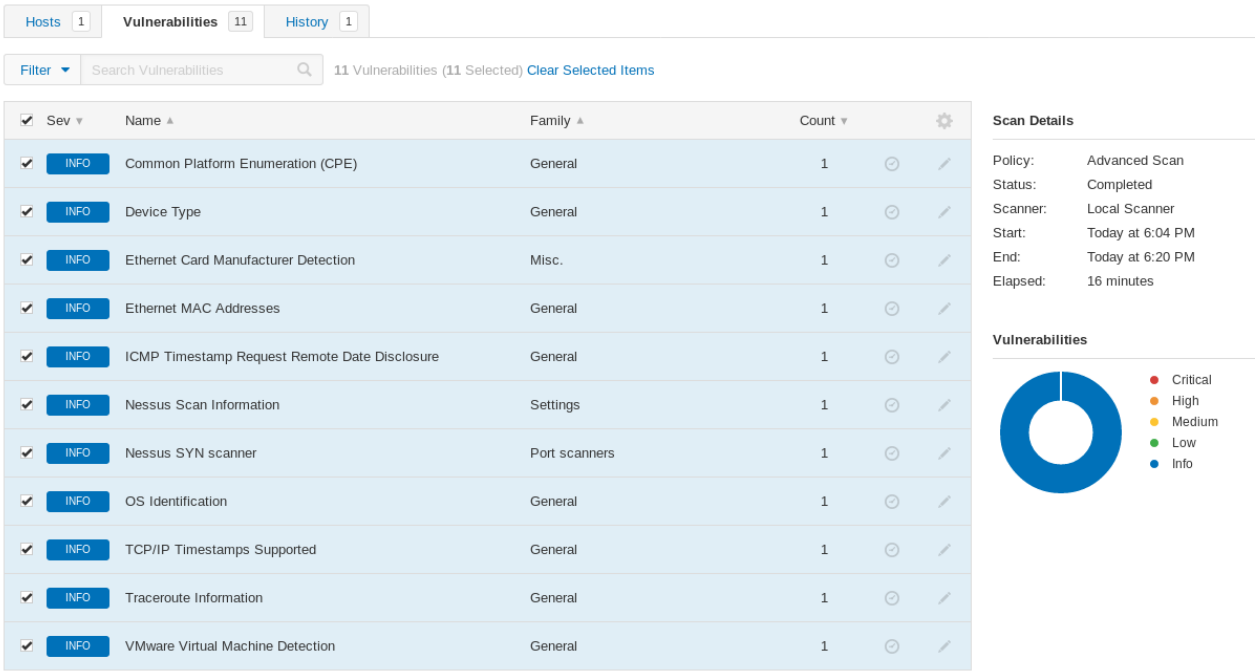
Info Nessus Plugin ID 10287

Synopsis

It was possible to obtain traceroute information.

Description

Makes a traceroute to the remote host.



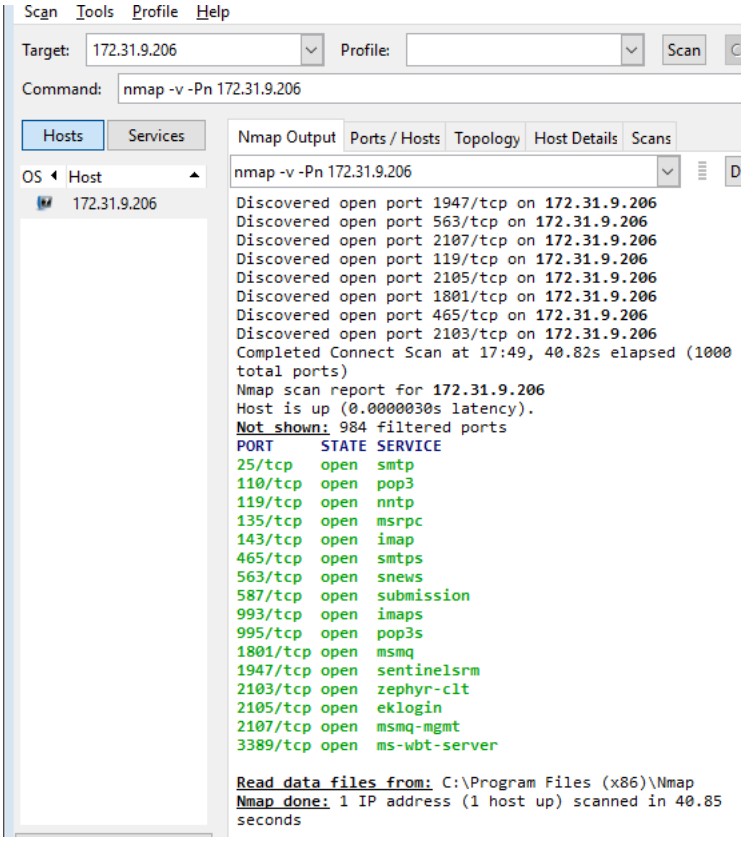
Detailed scanned report for the log server.

Plugin ID	CVE	CVSS	Risk	Host	Protocol	Port	Name	Synopsis	Description	Solution	See Also	Plugin Output
10114	CVE-1999-0524	0	None	172.31.9.193	icmp	0	ICMP Timestamp Request Remote Date Disclosure	It is possible to determine the exact time set on the remote host.	The remote host answers to an ICMP timestamp request. This allows an attacker to know the date that is set on the targeted machine, which may assist an unauthenticated, remote attacker in defeating time-based authentication protocols. Timestamps returned from machines running Windows Vista / 7 / 2008 / 2008 R2 are deliberately incorrect, but usually within 1000 seconds of the actual system time.	Filter out the ICMP timestamp requests (13), and the outgoing ICMP timestamp replies (14).		The remote clock is synchronized with the local clock.
10287			None	172.31.9.193	udp	0	Traceroute Information	It was possible to obtain traceroute information.	Makes a traceroute to the remote host.	n/a		For your information, here is the traceroute from 172.31.9.120 to 172.31.9.193 : 172.31.9.120 172.31.9.193 Hop Count: 1
11219			None	172.31.9.193	tcp	514	Nessus SYN scanner	It is possible to determine which TCP ports are open.	This plugin is a SYN 'half-open' port scanner. It shall be reasonably quick even against a firewalled target. Note that SYN scans are less intrusive than TCP (full connect) scans against broken services, but they might cause problems for less robust firewalls and also leave unclosed connections on the remote target, if the network is loaded.	Protect your target with an IP filter.		Port 514/tcp was found to be open
11936			None	172.31.9.193	tcp	0	OS Identification	It is possible to guess the remote operating system.	Using a combination of remote probes (e.g., TCP/IP, SMB, HTTP, NTP, SNMP, etc.), it is possible to guess the name of the remote operating system in use. It is also possible sometimes to guess the version of the operating system.	n/a		Remote operating system : Linux Kernel 2.6 Confidence level : 65 Method : SinFP The remote host is running Linux Kernel 2.6
19506			None	172.31.9.193	tcp	0	Nessus Scan Information	This plugin displays information about the Nessus scan.	This plugin displays, for each tested host, information about the scan itself : - The version of the plugin set. - The type of scanner (Nessus or Nessus Home). - The version of the Nessus Engine. - The port scanner(s) used. - The port range scanned. - Whether credentialed or third-party patch management checks are possible. - The date of the scan. - The duration of the scan. - The number of hosts scanned in parallel. - The number of checks done in parallel.	n/a		Information about this scan : Nessus version : 8.8.0 Plugin feed version : 201912031940 Scanner edition used : Nessus Home Scan type : Normal Scan policy used : Advanced Scan Scanner IP : 172.31.9.120 Port scanner(s) : nessus_syn_scanner Port range : default Thorough tests : no Experimental tests : no Paranoia level : 1 Report verbosity : 1 Safe checks : yes Optimize the test : yes Credentialed checks : no Patch management checks : None CGI scanning : disabled Web application tests : disabled Max hosts : 100 Max checks : 5 Recv timeout : 5 Backports : None Allow post-scan editing: Yes Scan Start Date : 2019/12/3 18:18 EST Scan duration : 130 sec
20094			None	172.31.9.193	tcp	0	VMware Virtual Machine Detection	The remote host is a VMware virtual machine.	According to the MAC address of its network adapter, the remote host is a VMware virtual machine.	Since it is physically accessible through the network, ensure that its configuration matches your organization's security policy.		The remote host is a VMware virtual machine.
25220			None	172.31.9.193	tcp	0	TCP/IP Timestamps Supported	The remote service implements TCP timestamps.	The remote host implements TCP timestamps, as defined by RFC1323. A side effect of this feature is that the uptime of the remote host can sometimes be computed.	n/a	http://www.ietf.org/rfc/rfc1323.txt	
35716			None	172.31.9.193	tcp	0	Ethernet Card Manufacturer Detection	The manufacturer can be identified from the Ethernet OUI.	Each ethernet MAC address starts with a 24-bit Organizationally Unique Identifier (OUI). These OUIs are registered by IEEE.	n/a	https://standards.ieee.org/faqs/regauth.html http://www.nessus.org/u?794673b4	The following card manufacturers were identified : 00:0C:29:6D:87:5E : VMware, Inc.
45590			None	172.31.9.193	tcp	0	Common Platform Enumeration (CPE)	It was possible to enumerate CPE names that matched on the remote system.	By using information obtained from a Nessus scan, this plugin reports CPE (Common Platform Enumeration) matches for various hardware and software products found on a host. Note that if an official CPE is not available for the product, this plugin computes the best possible CPE based on the information available from the scan.	n/a	http://cpe.mitre.org/ https://nvd.nist.gov/products/cpe	The remote operating system matched the following CPE : cpe:/o:linux:linux_kernel:2.6
54615			None	172.31.9.193	tcp	0	Device Type	It is possible to guess the remote device type.	Based on the remote operating system, it is possible to determine what the remote system type is (eg: a printer, router, general-purpose computer, etc).	n/a		Remote device type : general-purpose Confidence level : 65
86420			None	172.31.9.193	tcp	0	Ethernet MAC Addresses	This plugin gathers MAC addresses from various sources and consolidates them into a list.	This plugin gathers MAC addresses discovered from both remote probing of the host (e.g. SNMP and Netbios) and from running local checks (e.g. ifconfig). It then consolidates the MAC addresses into a single, unique, and uniform list.	n/a		The following is a consolidated list of detected MAC addresses: - 00:0C:29:6D:87:5E

NMAP scan report

Client Machine

Ports picked up upon a simple scan.



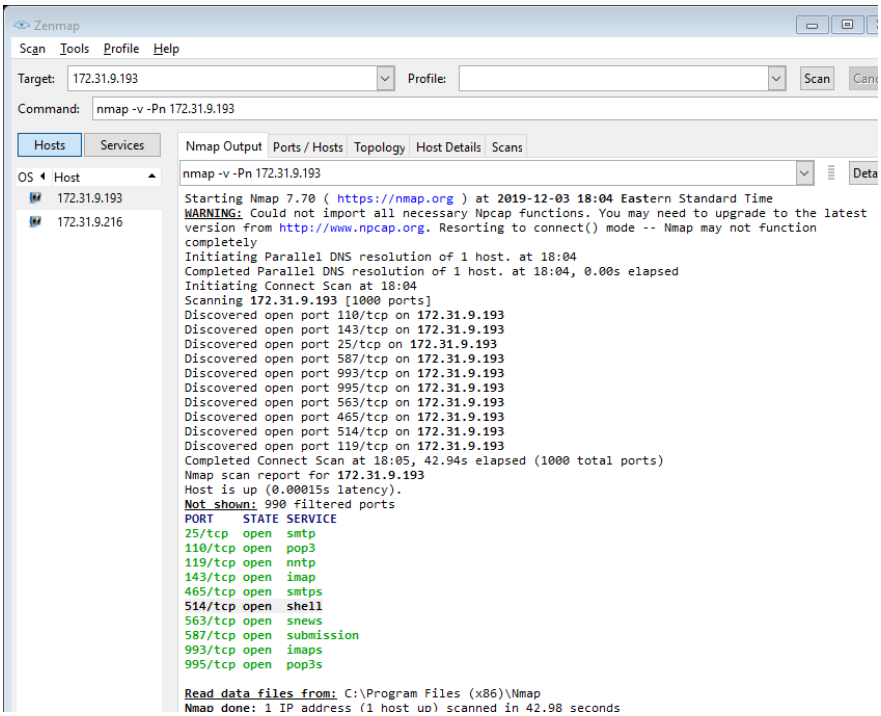
Suspicious ports:

TCP/563 - snews

The ports open were only basic mail services. TCP port 563 is commonly used, or at least was commonly used once, for NNTP (USENET news transfer) over SSL. Most likely, the reason it's open on your particular machine has nothing to do with that though, and you should actually check what's using the port on your specific machine.

Log Server and Domain Controller

Ports picked up on simple scan



Suspicious ports:

TCP/514 – shell

Since syslog's port 514 operates with UDP protocol and receives messages silently (returning no confirmation of their receipt), an open syslog port is not readily visible.

The two potential vulnerabilities of exposing a syslog server to the Internet exist:

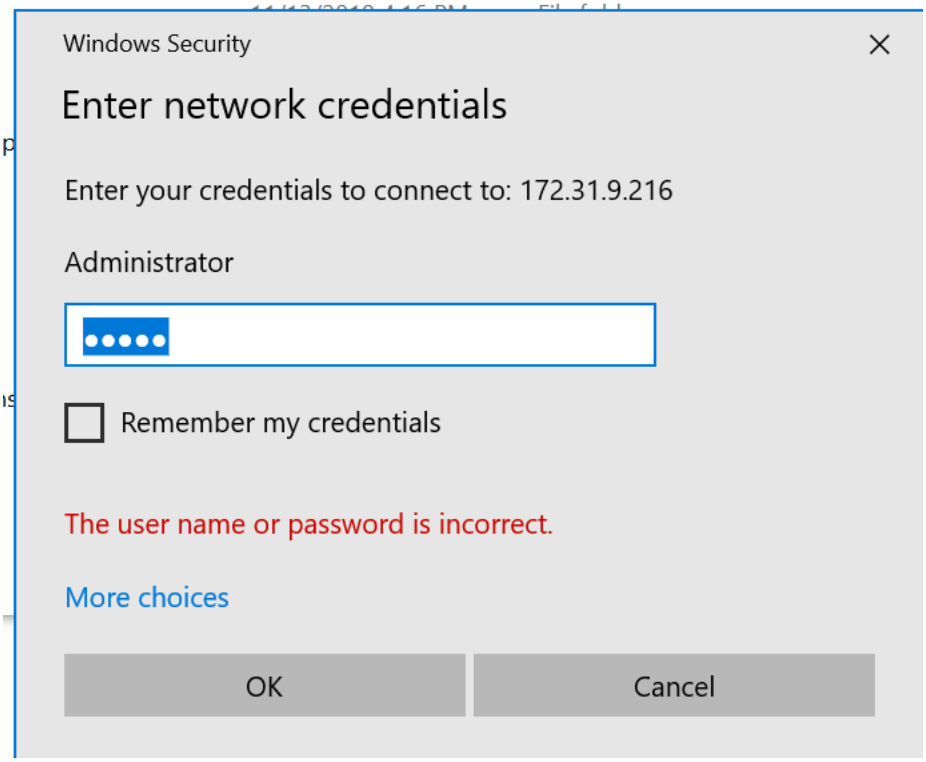
- The first would be someone determining that an exposed syslog service was present and maliciously flooding that log with erroneous messages.
- Secondarily, if the specific syslog server in use was known to have exploitable security vulnerabilities, those could be exploited by random Internet-wide scans.

Risk - since syslog is generally only used within controlled, local network boundaries, corporate and ISP networks may wish to block incoming UDP traffic destined to port 514 of any internal machines.

Open Domain

The domain controller was open to connect to.

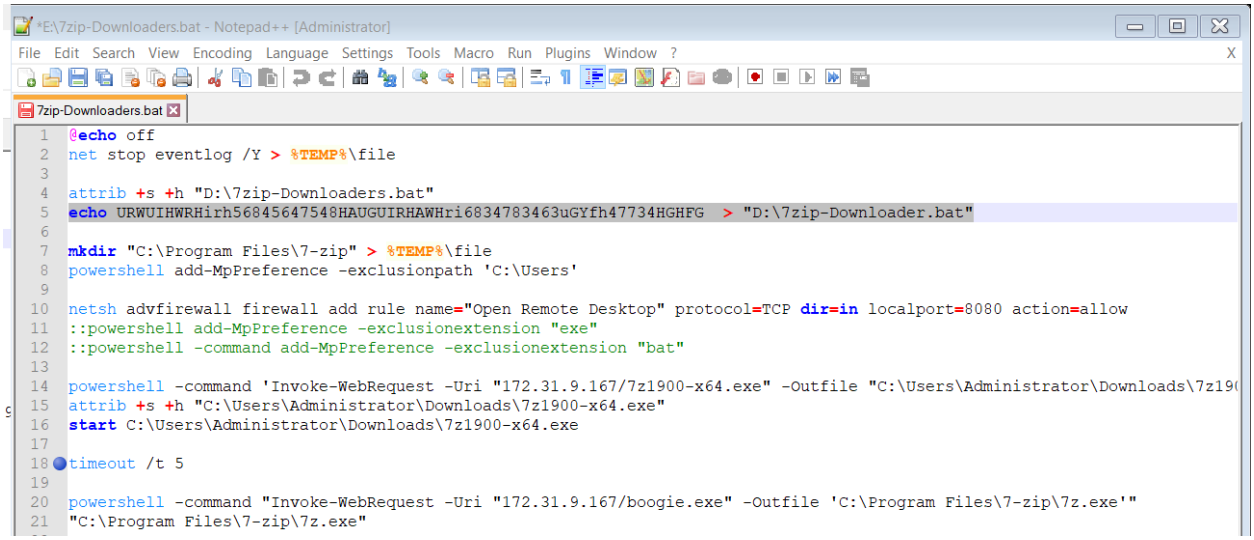
Since a domain was established we tried to connect to the domain using dictionary attacks to brute force. The password set is random, and we couldn't connect to domain.



Phishing attack

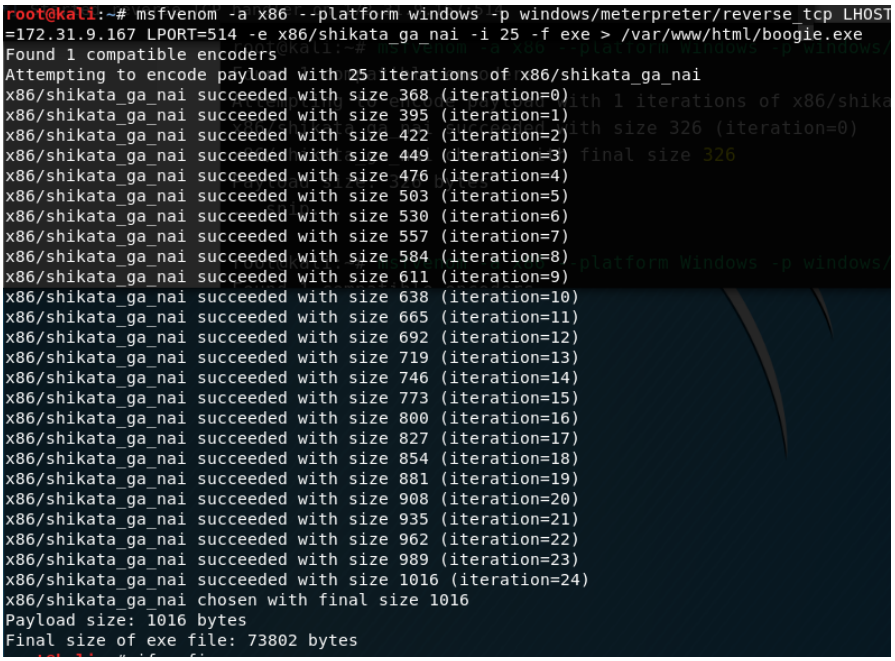
Delivery:

This is a batch the red team crafted that will delivered to the target as a false email saying that the company needs them to install 7zip for the future push installs for upcoming projects by the company. This batch file will stop the event logs. And exclude the C:\Users directory when the antivirus is running system check, followed by downloading a 7zip.exe the actual executable and download a payload that will create a reverse tcp shell to my machine through port 8080. Once the purpose of the batch file is completed it will change its contents to cover footprints.

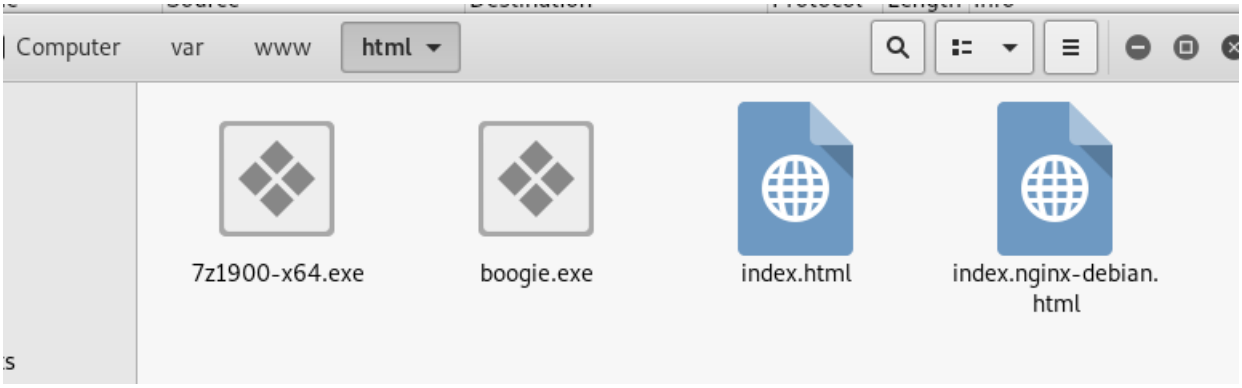


Malware:

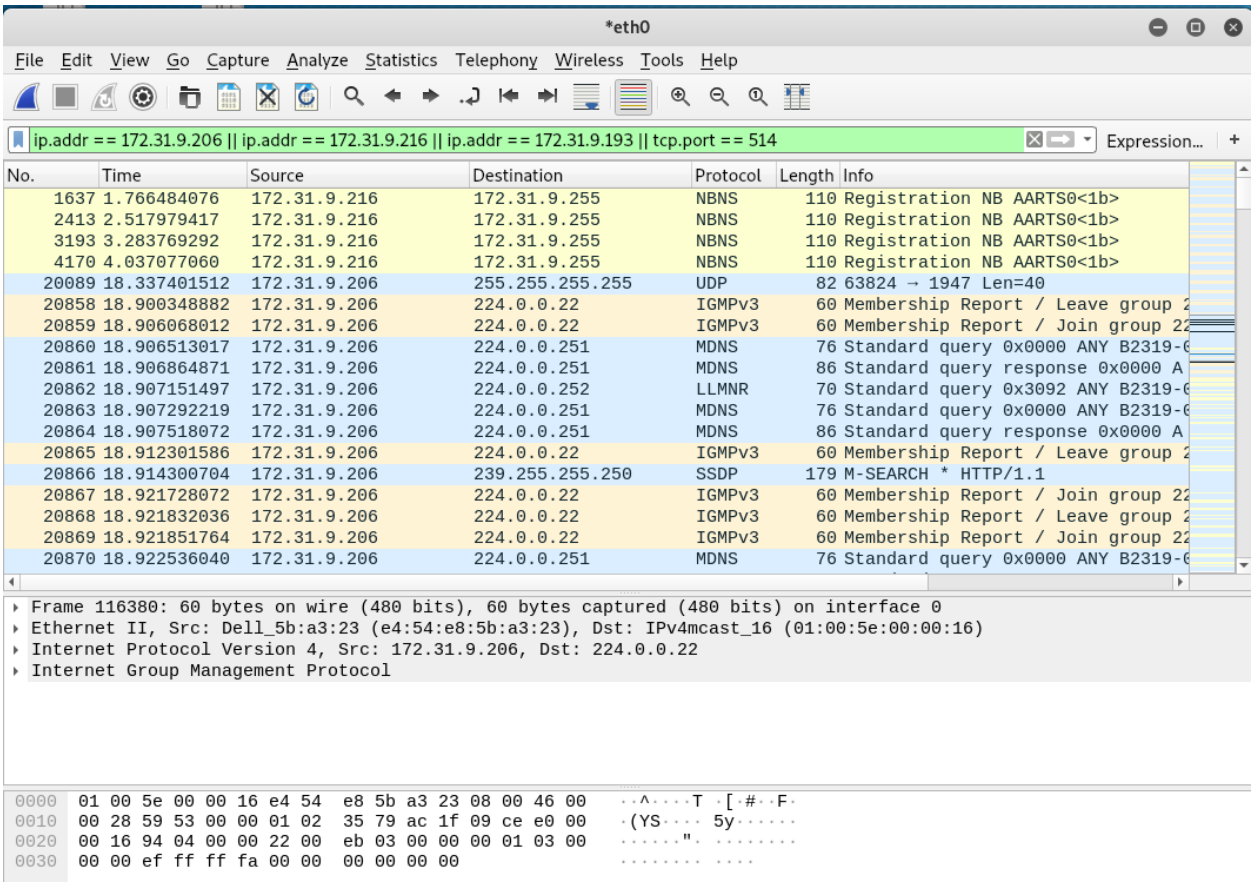
We used msfvenom to create the payload encoded with 25 iterations of shikata ga nai encoder to evade antivirus as well. Since, powershell command execution was disabled we found port 514 open shown in the Nmap scan, so payload was to connect through port 514.



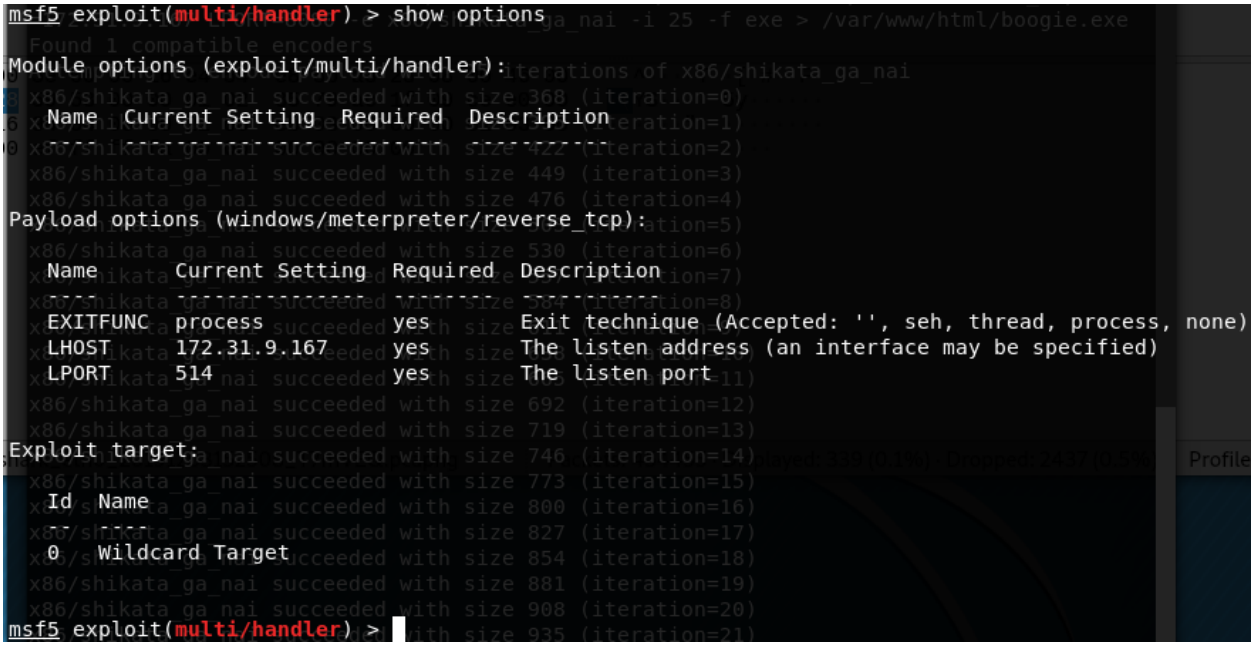
Server folder of our attacking machine.



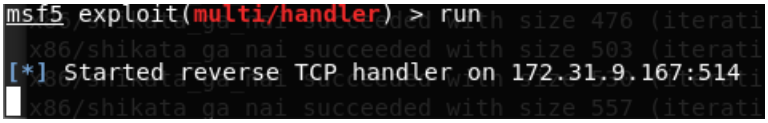
Wireshark monitoring targeted systems and information flowing.



Multi handler exploit listening for connections.



No shell connection was made.



Conclusion

The targeted system seems to be blocking the payload from executing. Therefore, a shell cannot be created.

SYN Flooding

A SYN flood is a form of denial-of-service attack in which we send a succession of SYN requests to the target's system in an attempt to consume enough server resources to make the system unresponsive to legitimate traffic.

Port 514 was also open on ip address 172.31.9.193, we tried to flood the logging servers.

```
msf5 auxiliary(dos/tcp/synflood) > options

Module options (auxiliary/dos/tcp/synflood):

   Name       Current Setting  Required  Description
   ----       ---------------  --------  -----------
   INTERFACE                   no        The name of the interface
   NUM                         no        Number of SYNs to send (else unlimited)
   RHOSTS     172.31.9.193     yes       The target address range or CIDR identifier
   RPORT      514              yes       The target port
   SHOST                       no        The spoofable source address (else randomizes)
   SNAPLEN    65535            yes       The number of bytes to capture
   SPORT                       no        The source port (else randomizes)
   TIMEOUT    500              yes       The number of seconds to wait for new data

msf5 auxiliary(dos/tcp/synflood) >
```

INFERENCE – this port is being used for central logging.

Logs on the target machine showing being flooded. Results of the SYN flood attack.

```
18:11:14.263246 IP (tos 0x0, ttl 187, id 36947, offset 0, flags [none], proto TCP (6), length 40)
    5.86.239.59.13990 > 172.31.9.193.shell: Flags [S], cksum 0xff7a (correct), seq 1169392457, win 1105, length 0
18:11:14.263265 IP (tos 0x0, ttl 64, id 0, offset 0, flags [DF], proto TCP (6), length 44)
    172.31.9.193.shell > 5.86.239.59.13990: Flags [S.], cksum 0xaa90 (incorrect → 0xd20b), seq 2253166773, ack 1169392458, win 64240, options [mss 1460], length 0
18:11:14.263270 IP (tos 0x0, ttl 144, id 36947, offset 0, flags [none], proto TCP (6), length 40)
    5.86.239.59.20449 > 172.31.9.193.shell: Flags [S], cksum 0x2a6e (correct), seq 3845103671, win 952, length 0
18:11:14.263273 IP (tos 0x0, ttl 64, id 0, offset 0, flags [DF], proto TCP (6), length 44)
    172.31.9.193.shell > 5.86.239.59.20449: Flags [S.], cksum 0xaa90 (incorrect → 0x3f7b), seq 2438285975, ack 3845103672, win 64240, options [mss 1460], length 0
18:11:14.263793 IP (tos 0x0, ttl 219, id 36947, offset 0, flags [none], proto TCP (6), length 40)
    5.86.239.59.23889 > 172.31.9.193.shell: Flags [S], cksum 0xecc7 (correct), seq 916028035, win 2105, length 0
18:11:14.263813 IP (tos 0x0, ttl 64, id 0, offset 0, flags [DF], proto TCP (6), length 44)
    172.31.9.193.shell > 5.86.239.59.23889: Flags [S.], cksum 0xaa90 (incorrect → 0x2eae), seq 4113219177, ack 916028036, win 64240, options [mss 1460], length 0
18:11:14.263817 IP (tos 0x0, ttl 184, id 36947, offset 0, flags [none], proto TCP (6), length 40)
    5.86.239.59.43280 > 172.31.9.193.shell: Flags [S], cksum 0xb9a2 (correct), seq 3182484434, win 1848, length 0
18:11:14.263820 IP (tos 0x0, ttl 64, id 0, offset 0, flags [DF], proto TCP (6), length 44)
    172.31.9.193.shell > 5.86.239.59.43280: Flags [S.], cksum 0xaa90 (incorrect → 0xf9e4), seq 1165651645, ack 3182484435, win 64240, options [mss 1460], length 0
18:11:14.264404 IP (tos 0x0, ttl 254, id 36947, offset 0, flags [none], proto TCP (6), length 40)
    5.86.239.59.15520 > 172.31.9.193.shell: Flags [S], cksum 0xd21f (correct), seq 3722318191, win 1377, length 0
18:11:14.264423 IP (tos 0x0, ttl 64, id 0, offset 0, flags [DF], proto TCP (6), length 44)
    172.31.9.193.shell > 5.86.239.59.15520: Flags [S.], cksum 0xaa90 (incorrect → 0x72d4), seq 1626468604, ack 3722318192, win 64240, options [mss 1460], length 0
18:11:14.264428 IP (tos 0x0, ttl 138, id 36947, offset 0, flags [none], proto TCP (6), length 40)
    5.86.239.59.45057 > 172.31.9.193.shell: Flags [S], cksum 0xe18a (correct), seq 618349206, win 2673, length 0
18:11:14.264430 IP (tos 0x0, ttl 64, id 0, offset 0, flags [DF], proto TCP (6), length 44)
    172.31.9.193.shell > 5.86.239.59.45057: Flags [S.], cksum 0xaa90 (incorrect → 0xf56e), seq 1928949973, ack 618349207, win 64240, options [mss 1460], length 0
18:11:14.264953 IP (tos 0x0, ttl 210, id 36947, offset 0, flags [none], proto TCP (6), length 40)
    5.86.239.59.22249 > 172.31.9.193.shell: Flags [S], cksum 0xb8bc (correct), seq 704757111, win 3152, length 0
18:11:14.264973 IP (tos 0x0, ttl 64, id 0, offset 0, flags [DF], proto TCP (6), length 44)
    172.31.9.193.shell > 5.86.239.59.22249: Flags [S.], cksum 0xaa90 (incorrect → 0x6eb9), seq 4014036051, ack 704757112, win 64240, options [mss 1460], length 0
18:11:14.264977 IP (tos 0x0, ttl 232, id 36947, offset 0, flags [none], proto TCP (6), length 40)
    5.86.239.59.58797 > 172.31.9.193.shell: Flags [S], cksum 0xa64b (correct), seq 3996682265, win 291, length 0
18:11:14.264980 IP (tos 0x0, ttl 64, id 0, offset 0, flags [DF], proto TCP (6), length 44)
    172.31.9.193.shell > 5.86.239.59.58797: Flags [S.], cksum 0xaa90 (incorrect → 0xf7b0), seq 1878404361, ack 3996682266, win 64240, options [mss 1460], length 0
18:11:14.265574 IP (tos 0x0, ttl 249, id 36947, offset 0, flags [none], proto TCP (6), length 40)
    5.86.239.59.26784 > 172.31.9.193.shell: Flags [S], cksum 0xc9aa (correct), seq 1527213976, win 1668, length 0
18:11:14.265593 IP (tos 0x0, ttl 64, id 0, offset 0, flags [DF], proto TCP (6), length 44)
    172.31.9.193.shell > 5.86.239.59.26784: Flags [S.], cksum 0xaa90 (incorrect → 0xd943), seq 499959392, ack 1527213977, win 64240, options [mss 1460], length 0
18:11:14.265597 IP (tos 0x0, ttl 185, id 36947, offset 0, flags [none], proto TCP (6), length 40)
    5.86.239.59.49516 > 172.31.9.193.shell: Flags [S], cksum 0x5871 (correct), seq 2671593158, win 909, length 0
18:11:14.265600 IP (tos 0x0, ttl 64, id 0, offset 0, flags [DF], proto TCP (6), length 44)
    172.31.9.193.shell > 5.86.239.59.49516: Flags [S.], cksum 0xaa90 (incorrect → 0x4086), seq 733535489, ack 2671593159, win 64240, options [mss 1460], length 0
18:11:14.266113 IP (tos 0x0, ttl 241, id 36947, offset 0, flags [none], proto TCP (6), length 40)
    5.86.239.59.47301 > 172.31.9.193.shell: Flags [S], cksum 0x0513 (correct), seq 3904786145, win 3574, length 0
18:11:14.266133 IP (tos 0x0, ttl 64, id 0, offset 0, flags [DF], proto TCP (6), length 44)
    172.31.9.193.shell > 5.86.239.59.47301: Flags [S.], cksum 0xaa90 (incorrect → 0x26c1), seq 2951031204, ack 3904786146, win 64240, options [mss 1460], length 0
18:11:14.266137 IP (tos 0x0, ttl 241, id 36947, offset 0, flags [none], proto TCP (6), length 40)
    5.86.239.59.12178 > 172.31.9.193.shell: Flags [S], cksum 0xe5ca (correct), seq 139319059, win 3761, length 0
18:11:14.266140 IP (tos 0x0, ttl 64, id 0, offset 0, flags [DF], proto TCP (6), length 44)
    172.31.9.193.shell > 5.86.239.59.12178: Flags [S.], cksum 0xaa90 (incorrect → 0xe2eb), seq 2032895398, ack 139319060, win 64240, options [mss 1460], length 0
18:11:14.266710 IP (tos 0x0, ttl 242, id 36947, offset 0, flags [none], proto TCP (6), length 40)
    5.86.239.59.49278 > 172.31.9.193.shell: Flags [S], cksum 0x4641 (correct), seq 983152817, win 2404, length 0
18:11:14.266729 IP (tos 0x0, ttl 64, id 0, offset 0, flags [DF], proto TCP (6), length 44)
    172.31.9.193.shell > 5.86.239.59.49278: Flags [S.], cksum 0xaa90 (incorrect → 0x91e5), seq 3552106312, ack 983152818, win 64240, options [mss 1460], length 0
18:11:14.266733 IP (tos 0x0, ttl 244, id 36947, offset 0, flags [none], proto TCP (6), length 40)
    5.86.239.59.2454 > 172.31.9.193.shell: Flags [S], cksum 0x38c1 (correct), seq 3282566341, win 170, length 0
18:11:14.266736 IP (tos 0x0, ttl 64, id 0, offset 0, flags [DF], proto TCP (6), length 44)
    172.31.9.193.shell > 5.86.239.59.2454: Flags [S.], cksum 0xaa90 (incorrect → 0xaa3c), seq 2656558616, ack 3282566342, win 64240, options [mss 1460], length 0
18:11:14.267330 IP (tos 0x0, ttl 163, id 36947, offset 0, flags [none], proto TCP (6), length 40)
    5.86.239.59.1837 > 172.31.9.193.shell: Flags [S], cksum 0xcb12 (correct), seq 1196351318, win 2698, length 0
18:11:14.267349 IP (tos 0x0, ttl 64, id 0, offset 0, flags [DF], proto TCP (6), length 44)
    172.31.9.193.shell > 5.86.239.59.1837: Flags [S.], cksum 0xaa90 (incorrect → 0x3908), seq 1719214941, ack 1196351319, win 64240, options [mss 1460], length 0
```

Solution

here are a number of well-known countermeasures listed in RFC 4987 including:

- Filtering
- Increasing backlog
- Reducing SYN-RECEIVED timer
- Recycling the oldest half-open TCP
- SYN cache
- SYN cookies
- Hybrid approaches
- Firewalls and proxies