SMTP Enumeration

Simple Mail Transfer Protocol (SMTP) is the standard protocol used for sending and receiving email messages on the internet. While SMTP is essential for email communication, improper configuration or the use of default settings can leave SMTP servers vulnerable to enumeration and exploitation.

Banner Grabbing with netcat.

Lets start with the basics first. - Banner Grabbing.

We can do this using netcat

```
nc 192.168.29.141 25
```

Detecting SMTP Service using nmap.

As we all know, we can use nmap to discover a service and its version running. Lets find out if SMTP is running on our target and which version.

```
sudo nmap -sS -sV -p 25 192.168.29.141
```

SMTPS information gathering using OpenSSL

If the target is using the secure version of SMTP that is SMTPS. Then we can use the openssl utility to gather information from it.

```
openssl s_client -crlf -connect smtp.mailgun.org:465
```

```
openssl s_client -starttls smtp -crlf -connect smtp.mailgun.org:587
```

Finding MX servers using dig

We have already used dig to find out information in our DNS recon section. However the MX records are now crucial when we are targetting Mail Server of our target.

Lets find out the mail servers of paytm.com

```
dig +short mx paytm.com
```

Discovering smtp commands using Nmap NSE smtp-commands scripts

Lets now user nmap scripts to perform some deeper enumeration into the target SMTP server. We will first start with the smtp-commands scripts. This will tell us which SMTP commands we can run on the target.

Some common useful commands are like:

- **RCPT TO**: This specifies who the email is being sent t
- **VRFY**: Checks if a particular email address is valid
- **EXPN**: Asks for a list of email addresses that a mailing list expands to

```
sudo nmap -sV -p25 --script smtp-commands 192.168.29.141
```

Gathering potential using Nmap NSE smtp-enum-users script

Now that we know which commands we can run on the SMTP server. Lets enumerate some potential usernames with the help of the VRFY command and SMTP-enum-user script.

```
sudo nmap -sV -p25 --script smtp-enum-users --script-args smtp-enum-
users.methods={VRFY} 192.168.29.141
```

There is an alternative way of doing this also. Remember, in ftp enumreration we used a script to enumerate the potential usernames. Like that, in this one we will use smtp-enum perl script to perform the same task.

```
./smtp-user-enum.pl -M VRFY -U ~/Desktop/users.txt -t 192.168.29.141
```

Now that we know about our potential usernames. Lets confirm the email addresses associated with it also.

```
./smtp-user-enum.pl -M VRFY -D metasploitable.localdomain -U
~/Desktop/users.txt -t 192.168.29.141
```

This thing can be also be done manually using telnet. Let see an example.

```
telnet 192.168.29.141 25

VRFY root

VRFY itachi

quit
```

Checking open relays using Nmap NSE open-relay script

Now lets check the target SMTP server for open relays.

But before that, lets understand Open relays.

open relay server is an email server that is misconfigured to allow anyone on the internet to send emails through it to any recipient, even if the recipient is not part of the server's local network

Imagine an open relay server as a mailbox that anyone can drop letters into, and the server will dutifully deliver those letters to the intended recipients, even if the recipients have no connection to the mailbox owner

So if we found a server vulnerable to open-relay then we can send emails from their server to anyone on their behalf. In short, we can use their domain name to send our email like arch1t3ct@zomato.com

Lets see how we can check this using nmap.

```
sudo nmap -sV -p25 --script smtp-open-relay 192.168.29.141
```

Gathering NTLM information from Windows host running SMTP

On Windows, with NTLM authentication enabled, sending a SMTP NTLM authentication request with null credentials will cause the remote service to respond with a NTLMSSP message disclosing information to include NetBIOS, DNS, and OS build version.

```
sudo nmap -sV -p25 --script smtp-ntlm-info 192.168.29.141
```

Checking SMTP server for known vulnerabilities

Now lets check if the target server is vulnerable to any known vulnerabilities using nmap.

```
sudo nmap -sV -p25 --script smtp-vuln-cve2010-4344,smtp-vuln-cve2011-
1720,smtp-vuln-cve2011-1764 192.168.29.141
```

Gathering information with Metasploit

Like every other section, there is a metasploit module for enumeration of SMTP servers. Lets see quickly what we can do with it.

First we can check the smtp version running.

```
use auxiliary/scanner/smtp/smtp_version
set RHOSTS <TARGET>
run
```

And Second we can enumerate SMTP users.

```
use auxiliary/scanner/smtp/smtp_enum
set RHOSTS <TARGET>
set THREADS 250
run
```

Email Headers

Now there is a very interesting technique that we can use to gather juicy information from the target SMTP server.

If we can trick our target to send us an email by any means like for example a web form in a contact page. Then we can understand the internal topology of the organization using the email headers of his email.

One neat trick to do that is to send an email to a non-existent address. The server will respond back with NDR mail stating that the recipient is not available. For example - we send a mail to farzimail@zomato.com now the recipient farzimail does not exist in zomato's mail server list. The Zomato mail server might bounce back our email stating that it did not find the recipient.

One thing to note here is that, we have to send the email from an allowed address as per the SPF records. If we are not allowed in the SPF records then we will not receive the NDR mail from the server.

We can add different contents also while exercising this technique as we might find more interesting information in the headers like: `X-Virus-Scanned: by av.domain.com` indicating the Anti virus in use by the target organization which can help us to develop better payloads later.
