# *IT Security (420-F30-HR)*

# *Lab 07 – Network Security*

Date assigned: Tuesday, March 18

Date Due: Part A, B, Tuesday, March 18, end of lab (mark in lab)

C-F, Friday end of lab

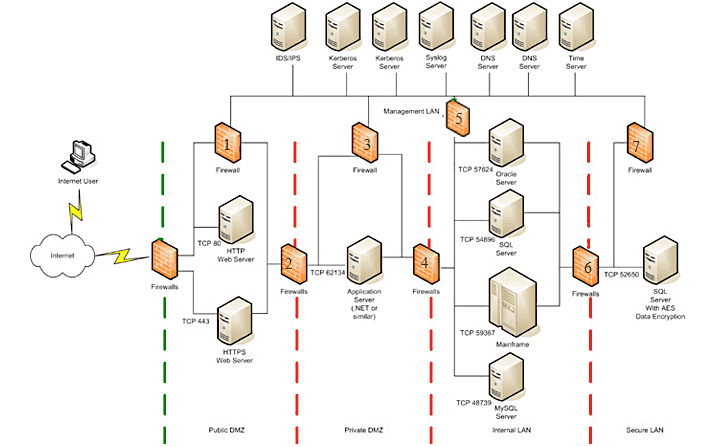
**Objectives:**

Learn:

1. Understand Network vulnerabilities to attack
2. Network Defenses and Counter measures

# Paper firewall

## Before working on real firewalls, refresh yourself on how they work.



In this network the following subnets are defined:

|  |  |  |
| --- | --- | --- |
| Network | Subnet | Host |
| Public DMZ | 13.225.189.112/28 | ~~113 HTTP Web Server~~ (don’t use this one~~)~~  114 HTTPS Web Server |
| Management LAN | 192.168.1.x/24 | 1 DNS1  2 DNS2  3 time server  4 SYS log server  5 kerberos1  6 kerberos2  X IDS/IPS (ignore for now) |
| Private DMZ | 13.255.189.128/28 | 129 Application server |
| Internal LAN | 192.168.2.x/24 | 1 Oracle Server  2 SQL Server  3 MY SQL Server  10 Mainframe |
| Secure LAN | 192.168.4.x/24 | 1 SQL Server |

Flows of traffic:

### Web traffic comes into HTTPS web server (port 443)

### All servers authenticate to Kerberos server (port 88), DNS, and time server

### Web servers connect to Application servers

### Applications servers connect to database servers (Oracle, SQL Server and MySQL) and the Mainframe for services.

### The mainframe connects SQL server in the secure lan

## Fill in the following Firewall 1. Assume no defaults.

(add/remove rows as required)

|  |  |  |
| --- | --- | --- |
| Allow/Deny | Source | Destination |
| Allow | 13.225.189.114 (HTTPS Web Server) | Kerberos 1 192.168.1.5  53 both |
| Allow | 13.225.189.114 (HTTPS Web Server) | Kerberos 2 192.168.1.6  53 both |
| Allow | 13.225.189.114 (HTTPS Web Server) | SYS log server 192.168.1.4  UDP 123 |
| Allow | 13.225.189.114 (HTTPS Web Server) | Time server 192.168.1.3  UDP 514 |
| Allow | 13.225.189.114 (HTTPS Web Server) | DNS 1 192.168.1.1  UDP 88 |
| Allow | 13.225.189.114 (HTTPS Web Server) | DNS 2 192.168.1.2  UDP 88 |
| Deny | Any | Any |

## Fill in the following Firewall 2. Assume no defaults.

(add/remove rows as required)

|  |  |  |
| --- | --- | --- |
| Allow/Deny | Source | Destination |
| Allow | 13.225.189.114 (HTTPS Web Server) | 13.255.189.129 (Application Server) TCP 62134 |
| Deny | Any | Any |

## Fill in the following Firewall 4. Assume no defaults.

(add/remove rows as required)

|  |  |  |
| --- | --- | --- |
| Allow/Deny | Source | Destination |
| Allow | 13.255.189.129 (Application Server) | 192.168.2.1 (Oracle Server)  TCP 57624 |
| Allow | 13.255.189.129 (Application Server) | 192.168.2.2 (SQL Server)  TCP 54896 |
| Allow | 13.255.189.129 (Application Server) | 192.168.2.3 (MySQL Server)  TCP 48739 |
| Allow | 13.255.189.129 (Application Server) | 192.168.2.10 (Mainframe)  TCP 59637 |
| Deny | Any | Any |

## Fill in the following Firewall 6. Assume no defaults.

(add/remove rows as required)

|  |  |  |
| --- | --- | --- |
| Allow/Deny | Source | Destination |
| Allow | 192.168.2.10 (Mainframe) | 192.168.4.1 (Secure SQL Server) TCP 52650 |
| Deny | Any | Any |

# DDOS

## Research how do vendors mitigate DDOS attacks? Read [this](https://www.cloudflare.com/en-gb/learning/ddos/ddos-mitigation/)

## What is a DDOS? How is it carried out and what is the impact?

Distributed Denial of Service. When an attacker uses many (lightly) infected computers to blindly attack an endpoint by making multiple requests, overloading its processing capability, causing real requests to get jammed or lost. The goal is to overwhelm the target and make it unavailable to legitimate users.

How it’s carried out:  
Botnet creation – infecting devices with malware

Launching the attack – attacker commands botnet to flood target system

Overloading the Target – target becomes slow, unresponsive, or crashes.

## What is DDoS mitigation? What equipment is used to do this?

The tools and technique used to detect, filter, and absorb malicious traffic.

Equipment used:

* Firewalls
* Intrusion Detection/Prevention Systems
* Load Balancers
* Web App Firewalls
* Cloud-based DDOS Protection Services

## How much traffic has Cloudflare been able to scale up to in order to handle DDoS?

Cloudflare has been able to mitigate DDoS attacks exceeding 71 million requests per second (RPS) – the largest ever recorded (as of 2023).

This was 35% larger than previous record-breaking attacks.

Cloudflare uses a global network of data centers to distribute and absorb attack traffic.

## Using a cloud service rather than designing your own counter measure is which risk response strategy {avoid/mitigate/accept/transfer}?

Transfer

Instead of handling DDoS protection internally (which is expensive and complex), the cloud provider absorbs the risk and mitigates attacks for you.

# Kali – Attack tool ARP spoofing

Log into your VDI environment. You may need to fix your LinuxBox1

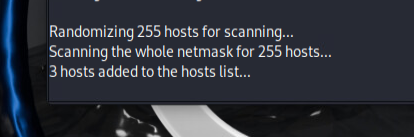
Startup all the machine in Configuration 01.

This could take awhile.

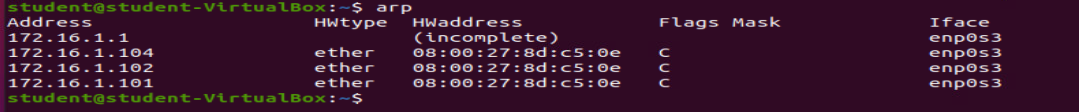
Now you know the hosts, and which ports are open (from earlier labs). You now have targets to do your vulnerability scans.

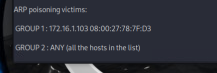
## Another scan tool. (sudo ettercap -G). Start it up and run it.

## Scan for hosts. (Click on the check mark, then magnifying glass). It discovers the 3 other hosts (see host list)

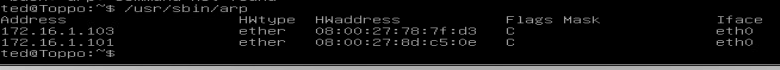


## Select 172.16.1.103, add to target1, run an ARP poisoning attack





## From LinuxBox1 (172.16.1.102), ping all the machines. Run arp. Note the mac address for 172.16.1.103. Which machine does it belong to? What happened here?



# Kali – Vulnerability discovery

Now you know the hosts, and which ports are open. You now have targets to do your vulnerability scans.

## Run skipfish against 172.16.1.103 (cd; skipfish -o ./web\_scan)

## Let it run for a while (timeout after 4-8 minutes, watch the scan time). Break it with ctrl-c (read the instructions)

Find the open the report (use FileManager to open web\_scan/index.html, it tells you this in the output after your broke it with ctrl-c)



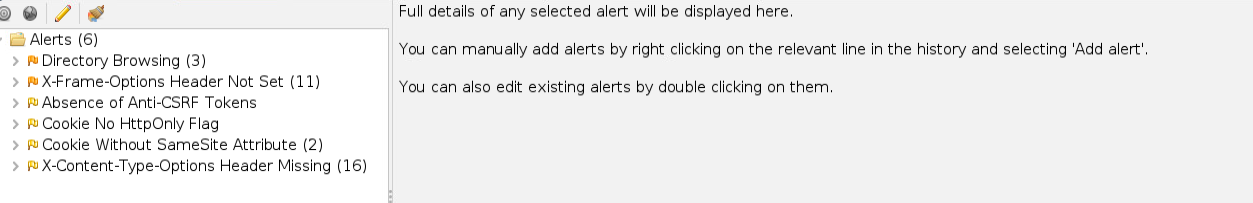
## What did Skipfish do?

Skipfish performed a web app security scan on the URL http://172.16.1.103/

The scan gives a good starting point to patch weak points like:

* Disabling directory listings.
* Securing password forms.
* Ensuring consistent charset and error handling.

## Just for nostalgia, run OWASP zap against 172.16.1.103. Show a summary of the alerts.

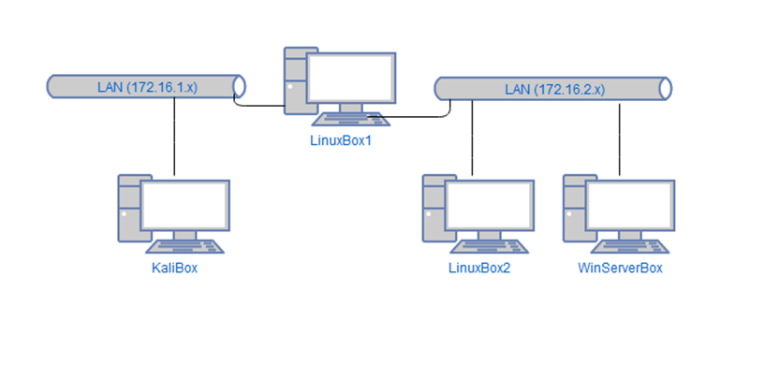


# Network Firewall

Shutdown all your Configuration 1 VMs.

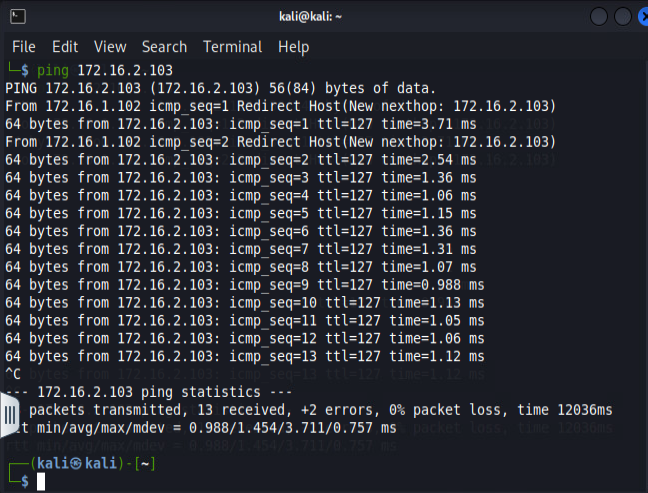
In this exercise, you will:

* Configure a network firewall to block traffic routed through it.

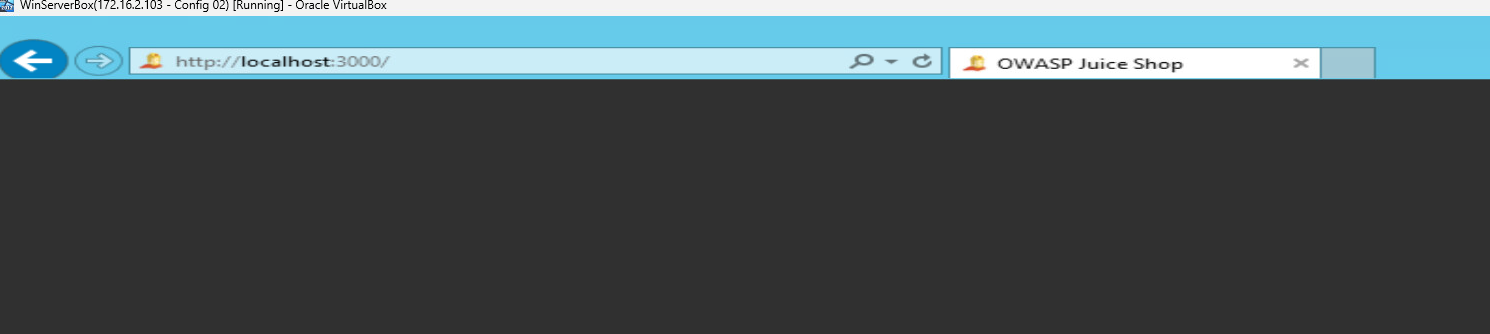


## In the VDI environment. Startup configuration2, KaliBox, LinuxBox1, WinServerBox

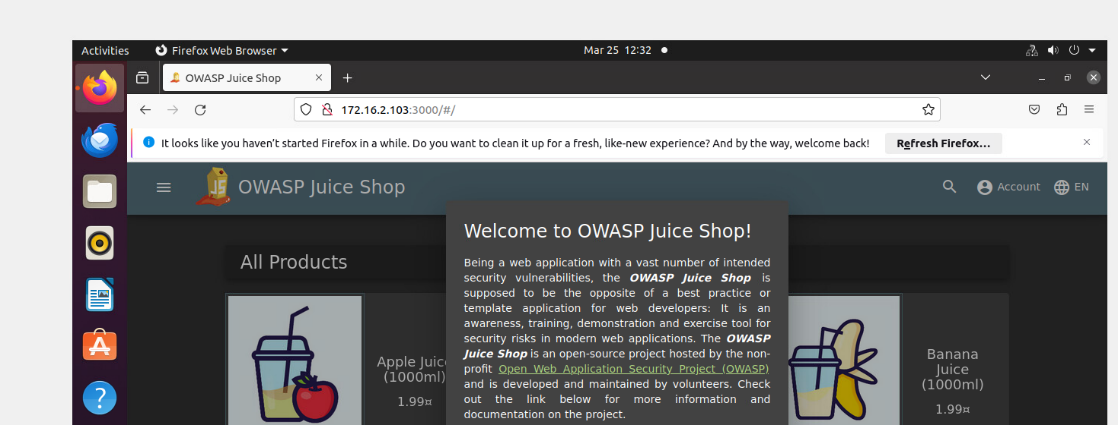
## From Kalibox, ping the WinServerBox.



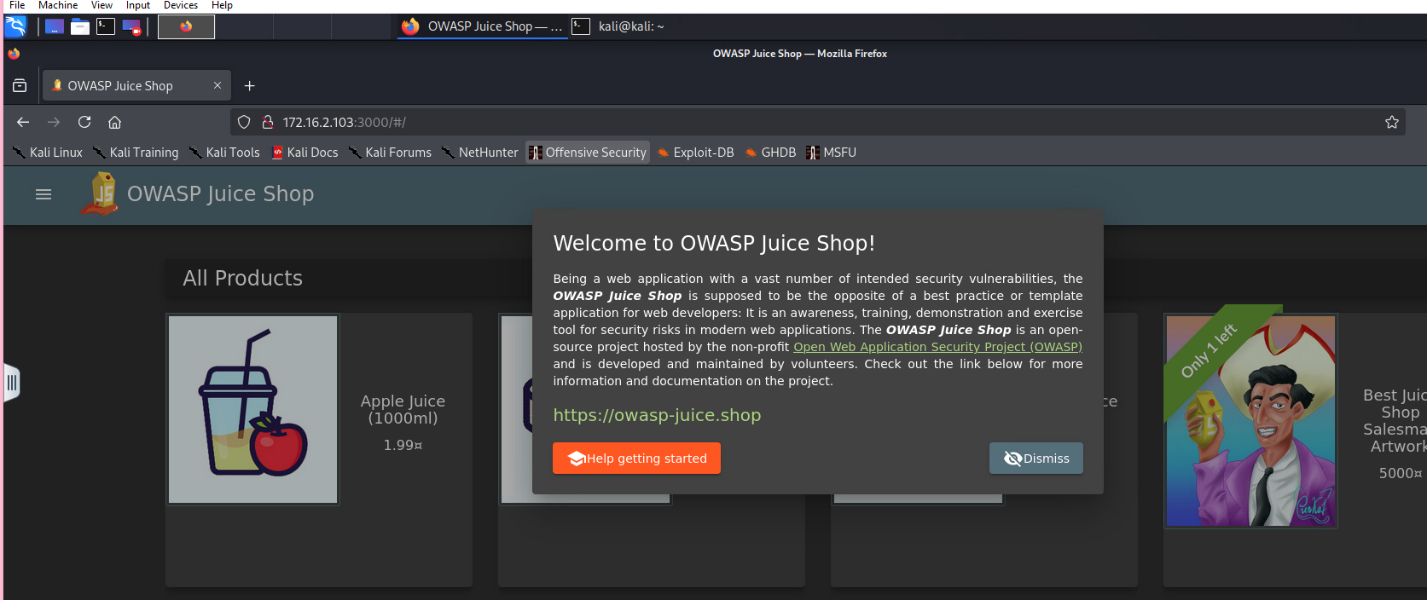
## On WinServerBox, startup JuiceShop (see Lab setup doc appendix)



Bring up the Juice Shop locally on LinuxBox1 browser to convince yourself that it’s up and running.



## On KaliBox, open the browser to the Juicebox site

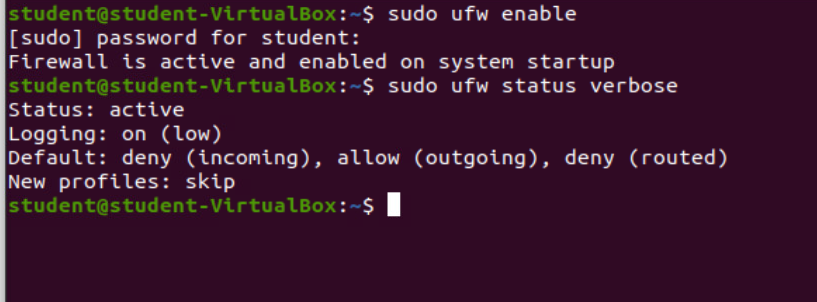
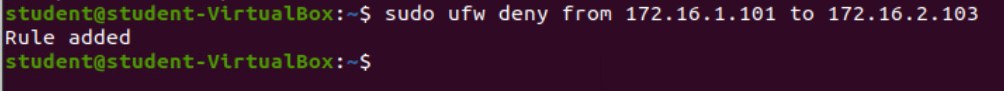


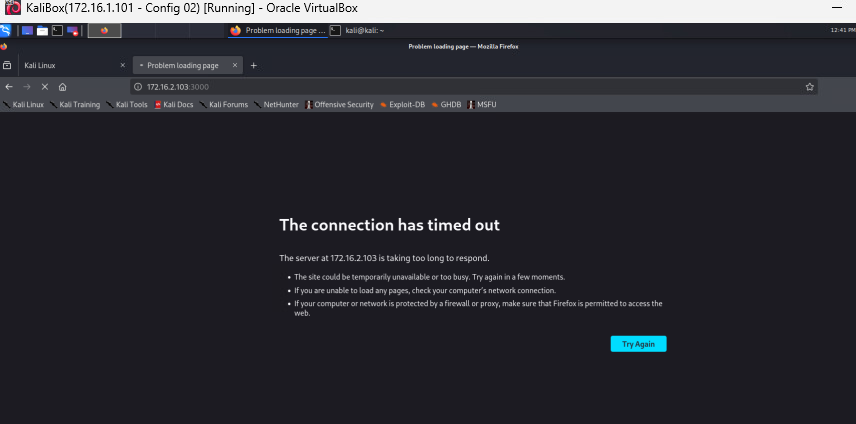
## Also convince yourself that you can open up the JuiceShop from LinuxBox1

## I am convinced.

## Your goal is to stop KaliBox from accessing JuiceShop on WinServerBox. Approach1: Firewall rule at LinuxBox1 to block traffic from KaliBox to WinServerBox port 3000 (or whatever port Juiceshop is running on.

Log onto LinuxBox1 and setup the firewall rule. [Hints](https://ubuntu.com/server/docs/security-firewall) (ufw)

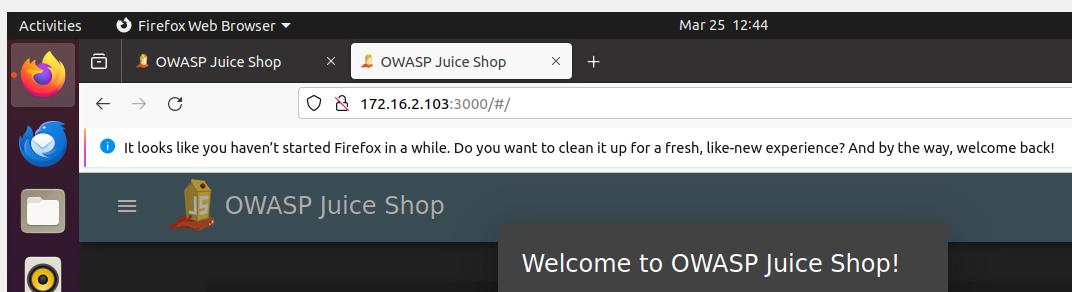
 



## 

## Despite KaliBox not being able to access the JuiceShop on WinServerBox, does LinuxBox1 still have access? Explain why KaliBox can’t access the same URL that LinuxBox1 can

Because we added a rule to the ufw in LinuxBox1. LinuxBox1’s own traffic isn’t blocked because the rule is based on the source IP (KaliBox)

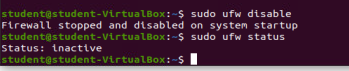


## Disable the firewall on LinuxBox1 and prove that KaliBox can once again access WinServerBox Juice shop

<commands to setup firewall>

sudo ufw disable

sudo ufw status



# Host Firewall

## In Part D above, you blocked traffic using a gateway firewall. What other option could you have done by configuring **only** on WinserverBox?

Instead of using a gateway firewall (LinuxBox1), you could block access directly on WinServerBox using Windows Firewall.

## Explain the steps/configuration needed

1. Open Windows Defender Firewall with Advanced Security
2. Click Inbound Rules → New Rule
3. Rule Type: Port
4. Protocol: TCP, Specific Port: 3000
5. Action: Block the connection
6. Scope: Specify remote IP
   1. Add KaliBox IP
7. Apply the rule.
8. Name it: Block JuiceShop from Kali

**Marking Scheme**

|  |  |
| --- | --- |
|  | **Out of** |
|  |  |
| **Part A: Paper Firewall** |  |
| F1 | 5 |
| F2 | 5 |
| F4 | 5 |
| F7 | 5 |
|  |  |
|  |  |
| **Part B: DDOS** |  |
| What | 4 |
| Mitigation | 4 |
| Traffic limits | 2 |
| Risk Response Strategy | 2 |
|  |  |
| **Part C: I’m having a ARP attack!** |  |
| scan | 2 |
| ARP Poison | 4 |
| LinuxBox2 proof of attack & analysis | 4 |
| Counter measures | 2 |
|  |  |
| **Part D: Vulnerability** |  |
| Skipfish report | 4 |
| Analysis | 2 |
| OWASP Zap | 4 |
|  |  |
|  |  |
| **Part E: Network Firewall** |  |
| Ping works | 2 |
| Juiceshop target started | 2 |
| Works from KaliBox | 2 |
| Works for LinuxBox1 | 2 |
| Block installed | 4 |
| Block confirmed KaliBox | 2 |
| But not for LInuxBox1 | 2 |
| Restore and proof | 2 |
|  |  |
| **Part F: Host Firewall** |  |
| Steps / Analysis | 4 |
|  |  |
| Crosswords | 10 |