How to check if your Linux WebServer is under a DoS attack

Author : admin

There are few commands I usually use to track if my server is possibly under **a Denial of Service attack** or under **Distributed Denial of Service**

Sys Admins who still have not experienced the terrible times of being under a DoS attack are happy people for sure ...

1. How to Detect a TCP/IP Denial of Service Attack This are the commands I use to find out if a loaded Linux server is under a heavy DoS attack, one of the most essential one is of course **netstat**.
To check if a server is under a **DoS** attack with netstat, it's common to use:

```
linux:~# netstat -ntu | awk '{print $5}' | cut -d: -f1 | sort | uniq -c | sort -n|wc -l
```

If the output of below command returns a result like **2000 or 3000 connections!**, then obviously it's very likely the server is under a DoS attack.

To check all the IPS currently connected to the Apache Webserver and get a very brief statistics on the number of times each of the IPs connected to my server, I use the cmd:

```
linux:~# netstat -ntu | awk '{print $5}' | cut -d: -f1 | sort | uniq -c | sort -n
221 80.143.207.107 233 145.53.103.70 540 82.176.164.36
```

As you could see from the above command output the IP *80.143.207.107* is either connected 221 times to the server or is in state of connecting or disconnecting to the node.

Another possible way to check, if a Linux or BSD server is under a Distributed DoS is with the list open files command **lsof**
Here is how *lsof* can be used to list the approximate number of **ESTABLISHED connections** to port 80.

```
linux:~# lsof -i TCP:80
litespeed 241931 nobody 17u IPv4 18372655 TCP server.pc-freak.net:http (LISTEN)
litespeed 241931 nobody 25u IPv4 18372659 TCP 85.17.159.89:http (LISTEN)
litespeed 241931 nobody 30u IPv4 29149647 TCP server.pc-freak.net:http->83.101.6.41:54565 (ESTABLISHED)
litespeed 241931 nobody 33u IPv4 18372647 TCP 85.17.159.93:http (LISTEN)
litespeed 241931 nobody 34u IPv4 29137514 TCP server.pc-freak.net:http->83.101.6.41:50885 (ESTABLISHED)
litespeed 241931 nobody 35u IPv4 29137831 TCP server.pc-freak.net:http->83.101.6.41:52312 (ESTABLISHED)
litespeed 241931 nobody 37w IPv4 29132085 TCP server.pc-freak.net:http->83.101.6.41:50000 (ESTABLISHED)
```

Another way to get an approximate number of established connections to let's say Apache or LiteSpeed webserver with lsof can be achieved like so:

```
linux:~# lsof -i TCP:80 |wc -l
2100
```

I find it handy to keep track of above **lsof** command output every few secs with *gnu watch* , like so:

```
linux:~# watch "lsof -i TCP:80"
```

2. How to Detect if a Linux server is under an ICMP SMURF attack

ICMP attack is still heavily used, even though it's already old fashioned and there are plenty of other Denial of Service attack types, one of the quickest way to find out if a server is under an ICMP attack is through the command:

```
server:~# while :; do netstat -s| grep -i icmp | egrep 'received|sent' ; sleep 1; done
120026 ICMP messages received
1769507 ICMP messages sent
120026 ICMP messages received
1769507 ICMP messages sent
```

As you can see the above one liner in a loop would check for sent and recieved ICMP packets every few seconds, if there are big difference between in the output returned every few secs by above command, then obviously the server is under an ICMP attack and needs to hardened.

3. How to detect a SYN flood with netstat

```
linux:~# netstat -nap | grep SYN | wc -l
1032
```

1032 SYNs per second is quite a high number and except if the server is not serving let's say 5000 user requests per second, therefore as the above output reveals it's very likely the server is under attack, if however I get results like 100/200 SYNs, then obviously there is no SYN flood targetting the machine ;)

Another two netstat command application, which helps determining if a server is under a Denial of Service attacks are:

```
server:~# netstat -tuna |wc -l
10012
```

and

```
server:~# netstat -tun |wc -l
9606
```

Of course there also some other ways to check the count the IPs who sent SYN to the webserver, for example:

```
server:~# netstat -n | grep :80 | grep SYN |wc -l
```

In many cases of course the **top** or **htop** can be useful to find, if many processes of a certain type are hanging around.

4. Checking if UDP Denial of Service is targetting the server

```
server:~# netstat -nap | grep 'udp' | awk '{print $5}' | cut -d: -f1 | sort |uniq -c |sort -n
```

The above command will list information concerning possible UDP DoS.

The command can easily be accustomed also to check for both possible TCP and UDP denial of service, like so:

```
server:~# netstat -nap | grep 'tcp\|udp' | awk '{print $5}' | cut -d: -f1 | sort |uniq -c |sort -n
104 109.161.198.86
115 112.197.147.216
129 212.10.160.148
227 201.13.27.137
3148 91.121.85.220
```

If after getting an IP that has too many connections to the server and is almost certainly a DoS host you would like to filter this IP.

You can use the **/sbin/route** command to filter it out, using route will probably be a better choice instead of iptables, as iptables would load up the CPU more than simply cutting the route to the server.

Here is how I remove hosts to not be able to route packets to my server:

```
route add 110.92.0.55 reject
```

The above command would null route the access of IP 110.92.0.55 to my server.

Later on to look up for a null routed IP to my host, I use:

```
route -n |grep -i 110.92.0.55
```

Well hopefully this should be enough to give a brief overview on how, one can dig in his server and find if he is under a Distributed Denial of Service, hope it's helpful to somebody out there.
Cheers ;)