HOW DDoS ATTACKS ARE LAUNCHED

Just about anyone with even a slightly above average understanding of computers, networks, and the Internet has heard of a DDoS attack. Even users who don't understand what they are or how they work have likely been exposed to them in the movies. The truth is that they can be incredibly effective and disruptive, and white hat hackers are always on the lookout for ways to prevent themselves from becoming the next victim. But DDoS attacks are shrouded in a cloud of mystery. After all, what are they, and how do they work? Well, we are going to answer these questions with a brief tutorial.

What is a DoS Attack?

A DDoS attack is a Distributed Denial of Service attack, but first let's define a DoS attack. In a DoS attack, flaws and code vulnerabilities are exploited with one main goal: to overwhelm a resource so drastically that it ceases to function correctly. For example, if an attacker wanted to initiate a DoS attack against a website, the attacker might generate so many requests that the web server couldn't possibly keep up with the demands, making it unavailable for other web surfers.

However, a *distributed* DoS attack is a little different in that it is not initiated by a single host. Instead, the attack is carried out by a network of different machines that make it hard to pinpoint the origin of the attack. Humorously enough, I have seen gamers resort to using a DDoS or DoS attack to target other online gamers when their tempers flared or they just wanted to boot the winning player from the game. However, even though this is admittedly amusing, it is also illegal.

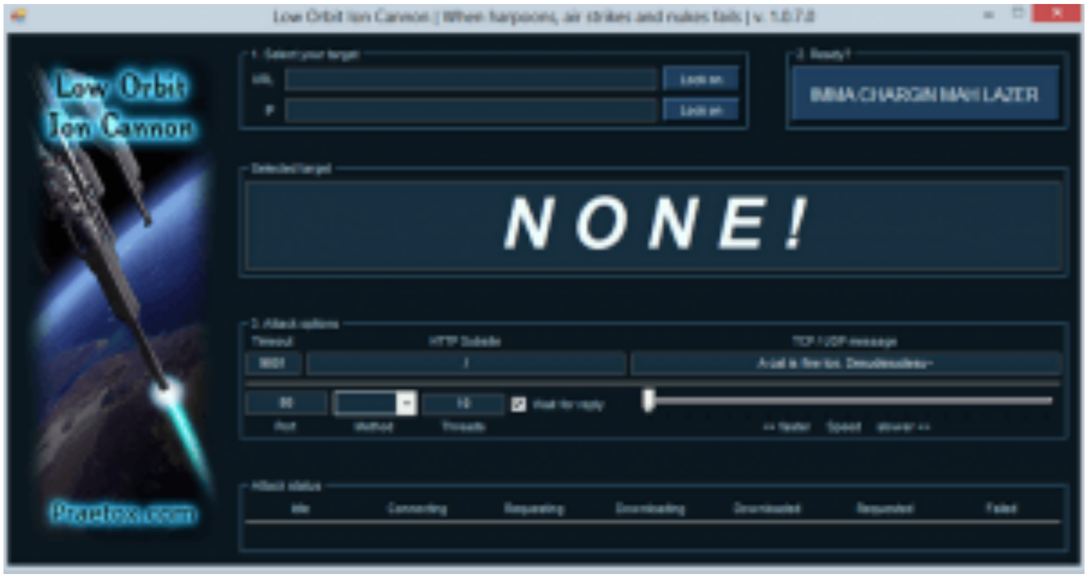
This guide is intended as a way to show you how DDoS attacks work, not to give you a tool to attempt to take down a website or mess with your neighbor. If you have dreams of taking down a website, watch out. There are legal implications and negative consequences if you get caught. So I would advise you to try this out on a network or entity you control. For example, if you host your own Minecraft server, that would be a perfectly acceptable test – but only if *you* own, control, and manage the server.

The truth is that there are many sophisticated an advanced ways of initiating these types of attacks. Hping3 being one of the more advanced techniques. However, today we are going to be looking at how to DDoS an individual IP address using an extremely simple piece of software.

Note that my antivirus software raised some flags after the download, but it only appears that it recognized this code was developed to initiate attacks. After scanning my computer post install, I didn't find any viruses, and it only labeled the software as a PUP (potentially unwanted program). You might have a similar experience, but the code seems fine to run.

The steps are as follows:

Step 1: Download and Install the Application



Timeout: 9001
HTTP Subsite: /
TCP / UDP message: A cat is fine too. Desudesudesu~
Port: 80
Threads: 10
☑ Wait for reply

The first thing you need is the software. We'll be using LOIC (Low Orbit Ion Cannon – I know, sounds like something from a Sci-Fi movie, right?) to carry out the attack. Proceed by downloading the file and extracting it in a location of your preference, though I just did it on my desktop.

Step 2: Target an IP Address of a Website

Go ahead and fire up the software after you have installed it. The first things you should notice at the top are the fields that are labeled as your target. If you want to test out this software on a website, go ahead and enter the URL. If you want to try this on another type of server, perhaps a game console on your local LAN while it is playing an online game, go ahead and enter its IP addresses. Whichever you chose, go ahead and click the “lock-on” button next to the field.

Step 3: Setting Up the Attack

Now we want to focus on the portion of the interface that shows different options and settings. Leave the timeout, HTTP subsite, and speed at the default settings (you can play around with these later, if you want). However, in the TCP/UDP field go ahead and enter some random message. Furthermore, pay attention to the port field. If you are attacking a common HTTP web server, this port will be port 80. However, this could change depending on the service you are trying to overwhelm. Lastly, disable the “wait for reply” and keep the thread count at 10.

Step 4: Attack!

The only thing left to do is to actually run the attack. Though I can’t say that I really like the language used in this application, it is one of the simplest ways to run a denial of service attack. After the fields have been filled in, simply click the “IMMA CHARGIN MAH LAZER” button (ugh). Then the attack status should be displayed. It will essentially show you the number of requests that have been made for a given service. For example, if you are attacking a website (that you own, of course), it will show you the number of HTTP requests that have been generated.

Final Thoughts on LOIC

Admittedly, there are much more sophisticated ways to run this type of attack and we will talk about that below, but again, this tutorial was aimed at simplicity. A lot of beginners want instant gratification, which this simple utility delivers. However, the ugly truth is that to become truly competent, a burgeoning hacker needs to invest countless hours studying their craft.

So Now lets look at some more advanced ways to run denial of service attacks.

Lets talk about DoS attacks and **hping3**!

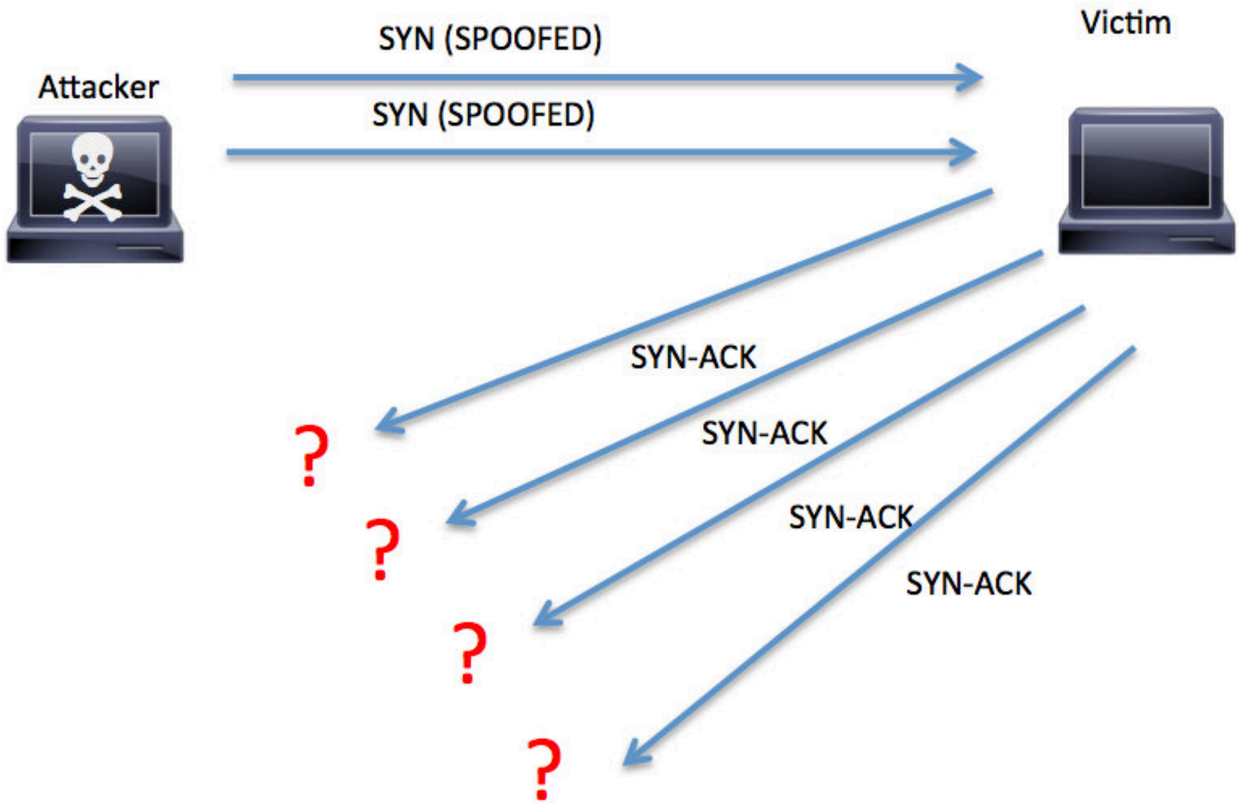
There is a tool by the name of hping3 that allows the attacker to craft and send custom packets. This allows us to do many things with it including recon, possibly some basic exploitation, but for now we're going to use it to launch a DoS attack. There are mutliple kinds of DoS attacks, but today we're going to launching a SYN flood. This sends requests to a server as fast as it can. When these requests are processed, it will take up the server's resources, and will render it unable to respond to any actual users trying to use it.

The problem with DoS attacks is that when we send all these packets the server, it has our address in it. All the administrator has to do is look at the logs and turn our address into the authorities, then an attacker is behind bars in a matter of days. We're not only going to be launching a SYN flood, but we're going to spoof our address so we don't get thrown in the big house! Before we launch the attack, let's deeper discuss the concept of SYN flooding.

SYN Flooding

As we previously stated, a SYN flood is sending an insane amount of requests to a server in order to use up all it's resources. But you may be asking "*What does SYN have to do with using up resources?*". Well, it's all about the TCP three-way handshake.

Remember, SYN stands for **syn**chronize. When we send a SYN packet, we're requesting to establish a connection.

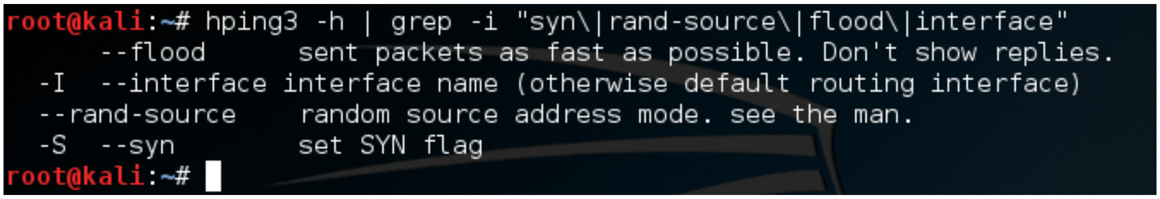


We can see that the attacker sent many SYN packets (with spoofed addresses) to the victim. The victim responded with a SYN-ACK to confirm the connection, but since there was no response, it sends it again and again, using up all it's resources! Also, since the attacker used a fake address, the administrator will have a much more difficult time tracing the source of the attack.

Now that we know how SYN floods work, let's get to launching the attack!

Launching the DoS Attack

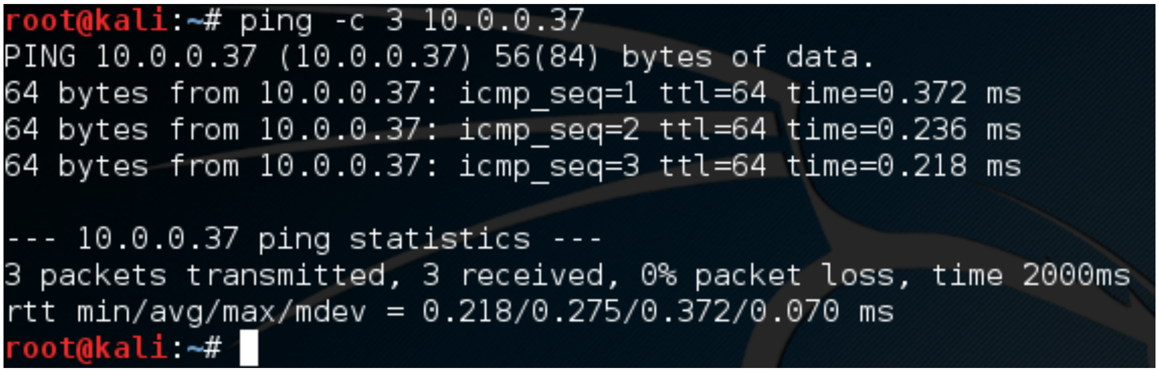
First things first, we'll need to look at the help page for hping3. In order to condense the output, I'm going to grep the lines that are essential. Let's see the flags we need to use:



```
root@kali:~# hping3 -h | grep -i "syn\|rand-source\|flood\|interface"
     --flood      sent packets as fast as possible. Don't show replies.
  -I  --interface interface name (otherwise default routing interface)
  --rand-source    random source address mode. see the man.
  -S  --syn        set SYN flag
root@kali:~# 
```

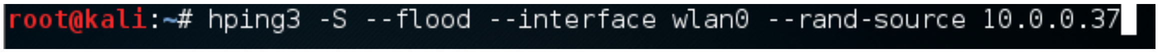
We can see here that we need to use **--flood**, **--interface**, **-S**, and **--rand-source**. These flags are fairly self-explanatory, but let's run through them. Using --flood will set hping3 into flood mode. This is the **flood** part of our SYN flood. Then we have --interface, so we can decide which network interface to send our packets out of. Finally we have --rand-source, this will randomize the source address of each packet. Not only will source not point back to us, but it will appear to come from a wide range of addresses, this increases the trace difficulty even *further*.

Now that we know what flags we're going to use, let's launch our attack. I'm going to be launching this attack against a VM I've set up, Metasploitable 2. First, let's ping the Metasploitable VM to make sure it's up and running, then we'll ping it again when we launch our attack to see the effect. Let's ping it now:



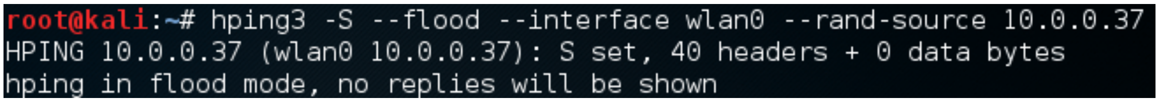
```
root@kali:~# ping -c 3 10.0.0.37
PING 10.0.0.37 (10.0.0.37) 56(84) bytes of data.
64 bytes from 10.0.0.37: icmp_seq=1 ttl=64 time=0.372 ms
64 bytes from 10.0.0.37: icmp_seq=2 ttl=64 time=0.236 ms
64 bytes from 10.0.0.37: icmp_seq=3 ttl=64 time=0.218 ms

--- 10.0.0.37 ping statistics ---
3 packets transmitted, 3 received, 0% packet loss, time 2000ms
rtt min/avg/max/mdev = 0.218/0.275/0.372/0.070 ms
root@kali:~# 
```

Alright, our VM is up and running. Now let's take a look at the command we'll use to launch our attack before we do it:



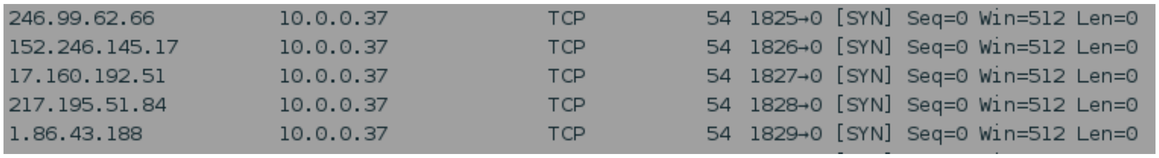
```
root@kali:~# hping3 -S --flood --interface wlan0 --rand-source 10.0.0.37
```

Alright, now that we have our command let's execute it. Now that we've started the attack we should see some output like this:



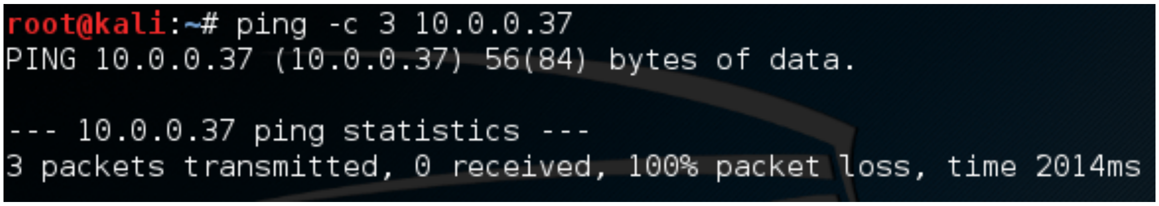
```
root@kali:~# hping3 -S --flood --interface wlan0 --rand-source 10.0.0.37
HPING 10.0.0.37 (wlan0 10.0.0.37): S set, 40 headers + 0 data bytes
hping in flood mode, no replies will be shown
```

There we go! Now we're flooding the target. To see our spoofed packets in action, let's open up one of the best network sniffers out there, wireshark. We should be able to see packets from multiple addresses being flooded towards the same address. Let's take a look at the packets the wireshark has captured:



```
246.99.62.66        10.0.0.37       TCP       54  1825→0 [SYN] Seq=0 Win=512 Len=0
152.246.145.17      10.0.0.37       TCP       54  1826→0 [SYN] Seq=0 Win=512 Len=0
17.160.192.51       10.0.0.37       TCP       54  1827→0 [SYN] Seq=0 Win=512 Len=0
217.195.51.84       10.0.0.37       TCP       54  1828→0 [SYN] Seq=0 Win=512 Len=0
1.86.43.188         10.0.0.37       TCP       54  1829→0 [SYN] Seq=0 Win=512 Len=0
```

Here we can see 5 packets, each with it's own unique source address! We can see that they are being send to our target at the IP 10.0.0.37, with the SYN flag set. Now that we're attacking our target, let's retry pinging the target and see what happens:



```
root@kali:~# ping -c 3 10.0.0.37
PING 10.0.0.37 (10.0.0.37) 56(84) bytes of data.

--- 10.0.0.37 ping statistics ---
3 packets transmitted, 0 received, 100% packet loss, time 2014ms
```

We can see by this ping tool output that our pings failed, we can't reach the server anymore! This proves that our attack was effective in that the server spent all it's resources responding to our attack instead of the real users, we've successfully DoS'd our target!

Since we've randomized the source of every packet, it will be *much* more difficult for an administrator. Now we can launch DoS attack without landing ourselves a seat in prison!

Final Warning

It cannot be stressed enough. Do not use this software in the real world to carry out an attack. That is illegal, and it can land you in a boatload of trouble. If you want to test a DoS attack to see how they function (as well as how to prevent them with different security defenses), only try them on your home network!