**Project 4: Denial of Service Attacks**

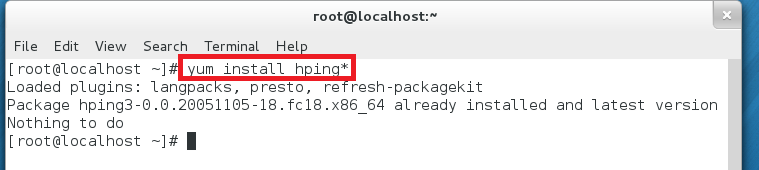
**TCSS 431: Network Security**

**Professor Wei Cheng**

**By Eduard Klimenko & Kyle Beveridge**

**On your FC client, as the root, type the following command: (3 points)**

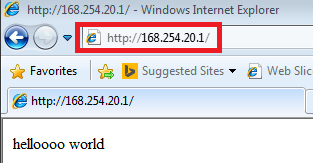
**# yum install hping\***

****

**From your Windows machine, open a web browser and type in the following: (3 points)**

**http://FC\_server\_IP**

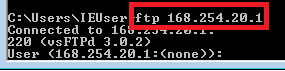
**Observe the time it takes to respond to your http request.**

****

**Start the command line prompt on your Windows, type the following**

**# ftp FC\_server\_IP**

**Observe the time it takes to respond to your ftp connection request. (3 points)**

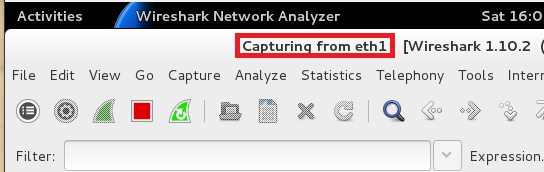
****

The request basically took no time, I did not see any loading or buffer screen. Very fast.

**On your FC server, as the root, start the Wireshark by typing the following command:**

**# wireshark &**

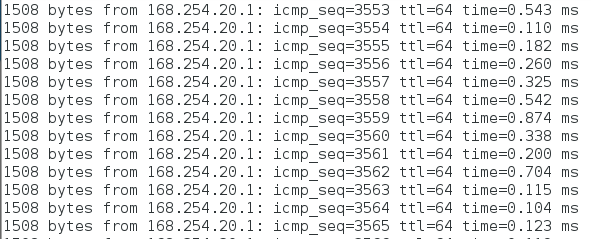
**Start data capture using the right network interface. (3 points)**

****

**We will launch the ping flooding attack from both the FC client and CentOS machines.**

**From both your FC client and CentOS, as the root user, issue the following command: (6 points)**

**# ping your\_FCserver -i 0.001 -s 1500**

****

**Explain the meaning of this command (3 points)**

Pings the ip address provided with an interval of .001 seconds and a packet size of 1500 (doesn’t include ICMP header which is 8 bytes).

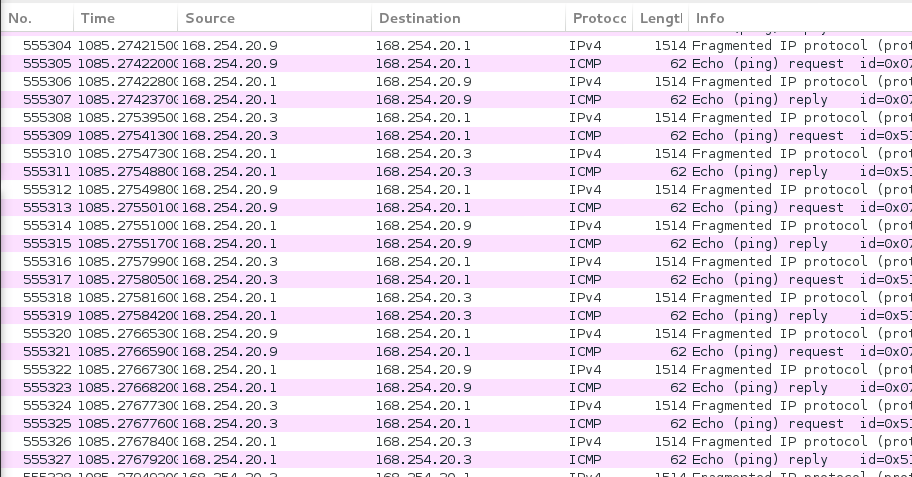
**Now back on your Windows machine, try to ftp into the server and reload the HTTP page. Observe the time it takes to answer your FTP and HTTP request. (3 points)**

There was some slight delay logging into the browser/ftp connection but nothing significant that one might think the server is being DoS’d.

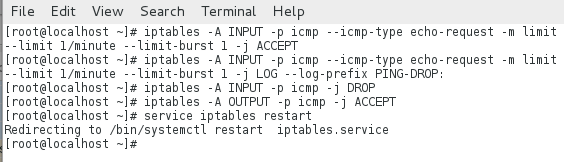
**Were you able to FTP into the server and reload the web page? Did you notice significant delay? Why? (3 points)**

I was able to log into the web page and the ftp server. I notice a very slight delay compared to the initial login. This is due to the fact that the server is being overloaded with many ping requests. The server is busy replying to the ping requests which slows legitimate traffic down. However, with only two machines pinging, DoS isn’t all that effective.

**Provide a screenshot of your Wireshark showing the ping flood attack. (3 points)**

****

**On your FC server, as the root user, add the following firewall rules: (3 points)**

****

**Explain the meaning of these rules. (3 points)**

Appends to input chain, specifies the icmp protocol, specifies type (echo-request), allows 1 packet per minute, number of matches that are allowed through before the 1/m limit kicks in (so right away in this case), accept packet and stop processing rules in this chain.

For the second rule, do the same as first, but log the packet and add “PING-DROP” in front of the line in the log.

Third rule drops incoming icmp packets.

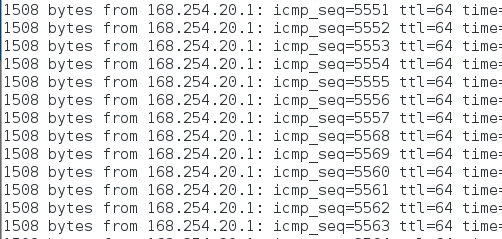
Fourth rule allows outgoing icmp packets.

**After you have finished the modifications and restart the firewall service by typing: (2 points)**

**# service iptables restart**

In screenshot above.

**Once again, launch the ping flooding attack from both the FC client and CentOS machines. (6 points)**



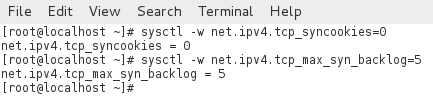
**Explain what happened. Did the ping rate limiting technique work? Why or why not? (3 points)**

The pings were still sent every .001 however many packets were dropped, only a fraction of the requests came back.

**On your FC server, as the root, type the following command: (3 points)**

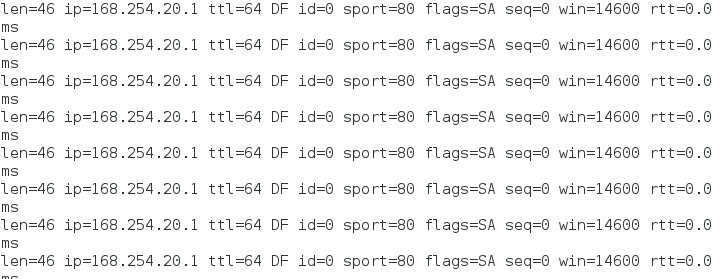
**# sysctl -w net.ipv4.tcp\_syncookies = 0**

**# sysctl -w net.ipv4.tcp\_max\_syn\_backlog = 5**

****

**From your FC client, as the root user, issue either of the following commands: (3 points)**

**# hping -i u1 -S -p 80 your-server-ip**

****

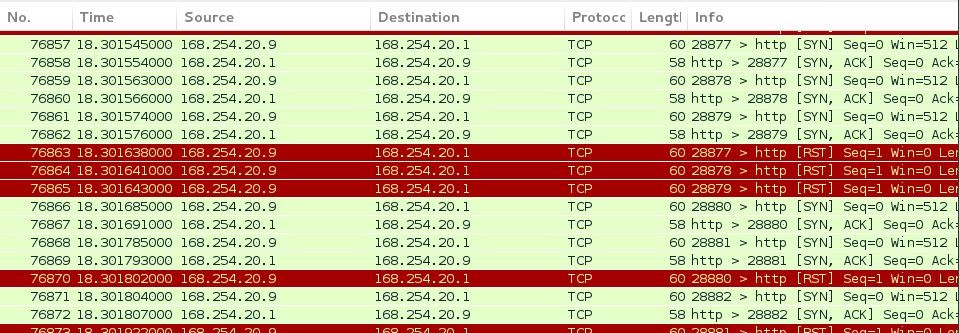
**Observe the time it takes to answer your FTP or HTTP request. (3 points)**

There was some delay but still was eventually able to log into the page and connect to ftp.

**Were you able to FTP into the server and reload the web page? Did you notice significant delay? Why it takes fewer resources (only one client) to achieve DoS compared to the ping flood attack? (3 points)**

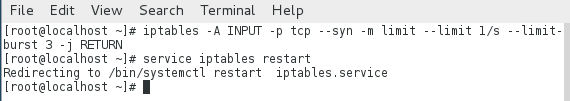
Yes, the page reloaded. The delay was longer than that of the DoS attack from two machines. The attacker sends several packets but does not send the "ACK" back to the server. The connections are hence half-opened and consuming server resources.

**Provide a screenshot of your Wireshark showing the SYN flood attack. (3 points)**

****

**On your FC server, as the root user, add the following firewall rules: (3 points)**

**iptables -A INPUT -p tcp --syn -m limit --limit 1/s --limit-burst 3 -j RETURN**

****

**Explain the meaning of these rules. (3 points)**

Limits incoming tcp syn protocols to 1 per second and sets the bucket to 3 packets.

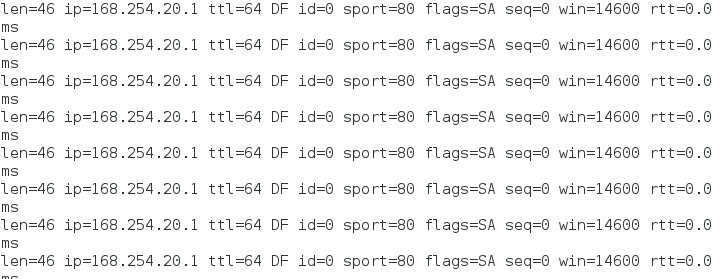
**After you have finished the modifications and restart the firewall service by typing: (2 points)**

**# service iptables restart**

In screenshot above.

**From your FC client, as the root user, issue either of the following commands: (3 points)**

**# hping -i u1 -S -p 80 your-server-ip**

****

**Now back on your Windows machine, try to ftp into the server and reload the HTTP page. Observe the time it takes to answer your FTP and HTTP request. (3 points)**

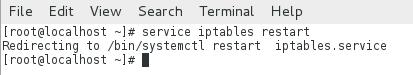
The webpage login was faster than before, the same goes for the ftp connection.

**Were you able to FTP into the server and reload the web page? Did you notice significant delay? (3 points)**

There was delay but I was still able to get into the webpage and ftp.

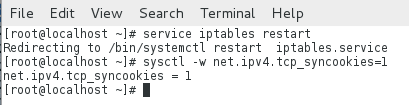
**Once you have accomplished this section, go back to your Fc server. Type: (2 points)**

**# service iptables restart**

****

**Enable SYN Cookies on your FC server. As the root user, type: (2 points)**

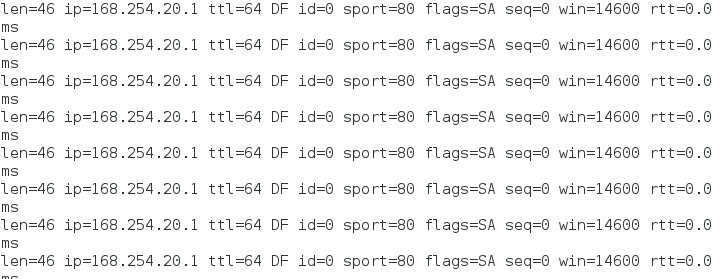
**# sysctl -w net.ipv4.tcp\_syncookies = 1**

****

**Now with SYN COOKIES enabled, start Wireshark on your FC server. Look at Wireshark and wait until the attack gets started (You will see a huge increase in TCP packets). (3 points)**

**From your FC client, as the root user, issue either of the following commands: (3 points)**

**# hping -i u1 -S -p 80 your-server-ip**

****

**Observe the time it takes to answer your FTP or HTTP request. (2 points)**

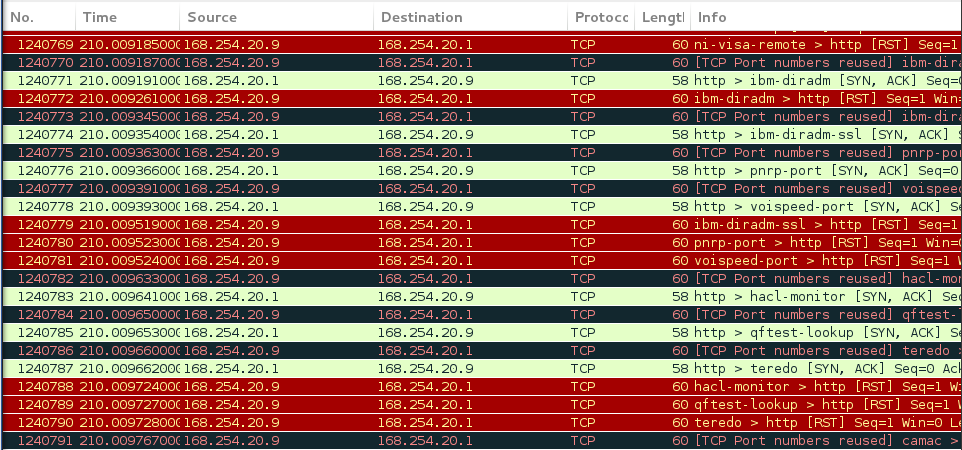
Access to the server took less time than the previous attempt (without cookies).

**Were you able to FTP into the server and reload the web page? Why or why not? Did you notice significant delay? (3 points)**

Yes I was able to load into the page and server. The delay wasn’t so bad because SYN cookies were enabled which are resistant to SYN flood attacks.

**Can you see any difference in the TCP packets? (Hint: Look at the sequence number) Provide a screenshot of your Wireshark capture. (3 points)**

The server sends back the appropriate SYN+ACK response to the client but discards the SYN queue entry. If the server then receives a subsequent ACK response from the client, the server is able to reconstruct the SYN queue entry using information encoded in the TCP sequence number.

****