BORDER GATEWAY PROTOCOL

Lab 12: IP Spoofing and Mitigation Techniques

Document Version: 3-5-2020

Award 1829698
“CyberTraining CIP: Cyberinfrastructure Expertise on High-throughput Networks for Big Science Data Transfers”
# Contents

Overview .......................................................................................................................... 3  
Objectives ....................................................................................................................... 3  
Lab settings ..................................................................................................................... 3  
Lab roadmap ..................................................................................................................... 3  
1 Introduction .................................................................................................................. 4  
   1.1 BGP overview ....................................................................................................... 4  
   1.2 IP Spoofing and DoS attacks ................................................................................. 4  
   1.3 Anti-Spoofing techniques .................................................................................... 5  
   1.3.1 Unicast Reverse Path Forwarding (uRPF) ...................................................... 5  
   1.3.2 Route filtering ................................................................................................. 6  
2 Lab topology ................................................................................................................ 6  
   2.1 Lab settings ......................................................................................................... 8  
   2.2 Open topology and load the configuration ......................................................... 8  
   2.3 Load zebra daemon and Verify IP addresses .................................................. 11  
3 Configure BGP on routers ......................................................................................... 16  
   3.1 Configure EBGP on all routers ........................................................................... 16  
   3.2 Configure IBGP on routers r2 and r3 ................................................................. 20  
4 Perform IP spoofing and DoS attack ....................................................................... 23  
5 Mitigate DDoS attach by using IP source filtering ............................................... 24  
References ...................................................................................................................... 26
Overview

This lab introduces Internet Protocol (IP) address spoofing that occurs on the Internet between routers running Border Gateway Protocol (BGP). In this lab, a compromised host will spoof the IP address and launch a Denial of Service (DoS) on a victim, each in a different Autonomous System (AS). The goal of this lab is to configure the Internet Service Provider (ISP) to mitigate IP spoofing attacks by applying the appropriate filters on the network traffic of its customers.

Objectives

By the end of this lab, students should be able to:

1. Configure BGP as the main protocol between ASes.
2. Understand and configure IP spoofing and DoS attack.
3. Understand IP spoofing mitigation techniques.
4. Apply route filters to mitigate IP spoofing.

Lab settings

The information in Table 1 provides the credentials to access Client1 machine.

<table>
<thead>
<tr>
<th>Device</th>
<th>Account</th>
<th>Password</th>
</tr>
</thead>
<tbody>
<tr>
<td>Client1</td>
<td>admin</td>
<td>password</td>
</tr>
</tbody>
</table>

Lab roadmap

This lab is organized as follows:

1. Section 1: Introduction.
2. Section 2: Lab topology.
3. Section 3: Configure BGP on routers.
4. Section 4: Perform IP spoofing and DoS attack.
5. Section 5: Mitigate DDoS attack by using IP source filtering.
1 Introduction

1.1 BGP overview

BGP is an exterior gateway protocol designed to exchange routing and reachability information among ASes on the Internet. BGP is relevant to network administrators of large organizations which connect to one or more ISPs, as well as to ISPs who connect to other network providers. In terms of BGP, an AS is referred to as a routing domain, where all networked systems operate common routing protocols and are under the control of a single administration.

BGP is a form of distance vector protocol. It requires each router to maintain a table, which stores the distance and the output interface (i.e., vector) to remote networks. BGP makes routing decisions based on paths, network policies, or rule set configured by a network administrator and is involved in making core routing decisions.

Two routers that establish a BGP connection are referred to as BGP peers or neighbors. BGP sessions run over Transmission Control Protocol (TCP). If a BGP session is established between two neighbors in different ASes, the session is referred to as an External BGP (EBGP) session. If the session is established between two neighbors in the same AS, the session is referred to as Internal BGP (IBGP). Figure 1 shows a network running BGP protocol. Routers that exchange information within the same AS use IBGP, while routers that exchange information between different ASes use EBGP.

1.2 IP Spoofing and DoS attacks

IP source address spoofing is the process of originating IP packets with source addresses other than those assigned to the origin host. An attacker that spoofs source IP addresses appears as the to be another host. IP spoofing can be exploited in several ways, mainly to launch DoS attacks. The latter is an attack that can exhaust the computing and communication resources of its victim within a short period of time.

Consider Figure 2. Host A (attacker) spoofs the source IP address of host C (victim) and request host B to send 100 GB of data. Host B will receive the request and sends the data to the spoofed source address, i.e., to host C. Thus, consuming the resources of the victim.
1.3 Anti-Spoofing techniques

Mutually Agreed Norms for Routing Security (MANRS) is a global initiative, supported by the Internet Society, that provides crucial fixes to reduce the most common routing threats. MANRS as many recommendations to prevent IP spoofing by ingress filtering, e.g., checking the source addresses of IP datagrams.

1.3.1 Unicast Reverse Path Forwarding (uRPF)

uRPF is one effective method to prevent IP spoofing. uRPF has multiple modes of operation, among them is the uRPF strict mode, in which the router accepts incoming packets on a specific interface if two conditions satisfy:

1. The source IP address of the incoming packet has an entry in the routing table.
2. The router uses the same interface to reach this source as where it received the packet on.

Consider Figure 3. Router r1 uses uRPF strict mode. Incoming packets on interface r1-eth0 that have source IP address of Host B will be dropped, since router r1 uses interface r1-eth1 to reach host B.
1.3.2 Route filtering

Route filtering is a method for selectively identifying routes that are advertised or received from neighbor routers. Route filtering may be used to manipulate traffic flows, reduce memory utilization, or to improve security.

Network operators should apply route filters to prevent spoofed IP packets from their customers. Consider Figure 4. The ISP (router r2) filters inbound network traffic of its customers, i.e., traffic sent from routers r1 and r3, based on their assigned IP addresses. Thus, each customer can’t generate network traffic with spoofed IP addresses.

![Figure 4](image_url)

Figure 4. The ISP drops network traffic sent from Customer 1, since their source IP address corresponds to Customer 2.

In this lab, we will apply the route filters to prevent IP spoofing using netfilter. The latter is a framework for packet filtering built in Linux kernel.

2 Lab topology

Consider Figure 5. The topology consists of three ASes. The ISP, consisting of routers r2 and r3, provides Internet service to the Campus-1 (router r1) and Campus-2 (router r4) networks. The Autonomous System Numbers (ASNs) assigned to Campus-1, ISP, and Campus-2 are 100, 200, and 300, respectively. The ISP communicates with the Campus networks via EBGP routing protocol, and the routers within the ISP communicate using IBGP. Host h1 in Campus-1 spoofs the IP address of host h4 in Campus-2. Consequently, host h1 launches a DoS attack on host h4 using hosts h2 and h3. To mitigate IP spoofing, the ISP (router r2) applies the appropriate route filters on the network traffic generated from Campus-1.
Figure 5. Lab topology.
2.1 Lab settings

Routers and hosts are already configured according to the IP addresses shown in Table 2.

<table>
<thead>
<tr>
<th>Device</th>
<th>Interface</th>
<th>IPv4 Address</th>
<th>Subnet</th>
<th>Default gateway</th>
</tr>
</thead>
<tbody>
<tr>
<td>r1 (Campus-1)</td>
<td>r1-eth0</td>
<td>192.168.1.1</td>
<td>/24</td>
<td>N/A</td>
</tr>
<tr>
<td></td>
<td>r1-eth1</td>
<td>192.168.12.1</td>
<td>/30</td>
<td>N/A</td>
</tr>
<tr>
<td>r2 (ISP)</td>
<td>r2-eth0</td>
<td>192.168.2.1</td>
<td>/24</td>
<td>N/A</td>
</tr>
<tr>
<td></td>
<td>r2-eth1</td>
<td>192.168.12.2</td>
<td>/30</td>
<td>N/A</td>
</tr>
<tr>
<td></td>
<td>r2-eth2</td>
<td>192.168.23.1</td>
<td>/30</td>
<td>N/A</td>
</tr>
<tr>
<td>r3 (ISP)</td>
<td>r3-eth0</td>
<td>192.168.3.1</td>
<td>/24</td>
<td>N/A</td>
</tr>
<tr>
<td></td>
<td>r3-eth1</td>
<td>192.168.23.2</td>
<td>/30</td>
<td>N/A</td>
</tr>
<tr>
<td></td>
<td>r3-eth2</td>
<td>192.168.34.1</td>
<td>/30</td>
<td>N/A</td>
</tr>
<tr>
<td>r4 (Campus-2)</td>
<td>r4-eth0</td>
<td>192.168.4.1</td>
<td>/24</td>
<td>N/A</td>
</tr>
<tr>
<td></td>
<td>r4-eth1</td>
<td>192.168.34.2</td>
<td>/30</td>
<td>N/A</td>
</tr>
<tr>
<td>h1</td>
<td>h1-eth0</td>
<td>192.168.1.10</td>
<td>/24</td>
<td>192.168.1.1</td>
</tr>
<tr>
<td>h2</td>
<td>h2-eth0</td>
<td>192.168.2.10</td>
<td>/24</td>
<td>192.168.2.1</td>
</tr>
<tr>
<td>h3</td>
<td>h3-eth0</td>
<td>192.168.3.10</td>
<td>/24</td>
<td>192.168.3.1</td>
</tr>
<tr>
<td>h4</td>
<td>h4-eth0</td>
<td>192.168.4.10</td>
<td>/24</td>
<td>192.168.4.1</td>
</tr>
</tbody>
</table>

2.2 Open topology and load the configuration

Step 1. Start by launching Miniedit by clicking on Desktop’s shortcut. When prompted for a password, type `password`.

![MiniEdit shortcut](image)
Step 2. On Miniedit’s menu bar, click on File then open to load the lab’s topology. Locate the Lab12.mn topology file in the default directory, /home/frr/BGP_Labs/lab12 and click on Open.

Figure 7. MiniEdit’s Open dialog.

At this point the topology is loaded with all the required network components. You will execute a script that will load the configuration of the routers.

Step 3. Open the Linux terminal.

Figure 8. Opening Linux terminal

Step 4. Click on the Linux’s terminal and navigate into BGP_Labs/lab12 directory by issuing the following command. This folder contains a configuration file and the script responsible for loading the configuration. The configuration file will assign the IP addresses to the routers’ interfaces. The cd command is short for change directory followed by an argument that specifies the destination directory.

```
cd BGP_Labs/lab12
```
Step 5. To execute the shell script, type the following command. The argument of the program corresponds to the configuration zip file that will be loaded in all the routers in the topology.

```
./config_loader.sh lab12_conf.zip
```

Step 6. Type the following command to exit the Linux terminal.

```
exit
```

Step 7. At this point hosts h1, h2, h3 and h4 interfaces are configured. To proceed with the emulation, click on the Run button located in lower left-hand side.

Step 8. Click on Mininet’s terminal, i.e., the one launched when MiniEdit was started.
Step 9. Issue the following command to display the interface names and connections.

```
links
```

In Figure 14, the link displayed within the gray box indicates that interface eth0 of host h1 connects to interface eth1 of switch s1 (i.e., h1-eth0 <-> s1-eth1).

2.3 Load zebra daemon and Verify IP addresses

You will verify the IP addresses listed in Table 2 and inspect the routing table of routers r1, r2, r3 and r4.

Step 1. Hold right-click on host h1 and select Terminal. This opens the terminal of host h1 and allows the execution of commands on that host.
**Step 2.** On host h1 terminal, type the command shown below to verify that the IP address was assigned successfully. You will verify that host h1 has an interface, h1-eth0 configured with the IP address 192.168.1.10 and the subnet mask 255.255.255.0.

```bash
ifconfig
```

![Figure 16. Output of `ifconfig` command.](image)

**Step 3.** On host h1 terminal, type the command shown below to verify that the default gateway IP address is 192.168.1.1.

```bash
route
```
Lab 12: IP spoofing and mitigation techniques

Step 4. In order to verify hosts h2, h3 and h4, proceed similarly by repeating from step 1 to step 3 on host h2, h3 and h4 terminals. Similar results should be observed.

Step 5. You will validate that the router interfaces are configured correctly according to Table 2. In order to verify router r1, hold right-click on router r1 and select Terminal.

Figure 17. Output of `route` command.

Figure 18. Opening terminal on router r1.
Lab 12: IP spoofing and mitigation techniques

**Step 6.** Start zebra daemon, which is a multi-server routing software that provides TCP/IP based routing protocols. The configuration will not be working if you do not enable zebra daemon initially. In order to start the zebra, type the following command:

```bash
zebra
```

![Figure 19. Starting zebra daemon.](image)

**Step 7.** After initializing zebra, vtysh should be started in order to provide all the CLI commands defined by the daemons. To proceed, issue the following command:

```bash
vtysh
```

![Figure 20. Starting vtysh on router r1.](image)

**Step 8.** Type the following command on router r1 terminal to verify the routing table of router r1. It will list all the directly connected networks. The routing table of router r1 does not contain any route to the networks attached to routers r2 (192.168.2.0/24), r3 (192.168.3.0/24) and r4 (192.168.4.0/24) as there is no routing protocol configured yet.

```bash
show ip route
```

![Figure 21. Displaying the routing table of router r1.](image)
**Step 9.** Router r2 is configured similarly to router r1 but, with different IP addresses (see Table 2). Those steps are summarized in the following figure. To proceed, in router r2 terminal issue the commands depicted below. At the end, you will verify all the directly connected networks of router r2.

![show ip route](image_url)

Figure 22. Displaying the routing table of router r2.

**Step 10.** Router r3 is configured similarly to router r1 but, with different IP addresses (see Table 2). Those steps are summarized in the following figure. To proceed, in router r3 terminal issue the commands depicted below. At the end, you will verify all the directly connected networks of router r3.

![show ip route](image_url)

Figure 23. Displaying the routing table of router r3.

**Step 11.** Router r4 is configured similarly to router r1 but, with different IP addresses (see Table 2). Those steps are summarized in the following figure. To proceed, in router r4 terminal issue the commands depicted below. At the end, you will verify all the directly connected networks of router r4.
3 Configure BGP on routers

In this section, you will configure BGP on the routers that are hosted in different ASes. You will assign BGP neighbors to allow the routers to exchange BGP routes. Furthermore, all routers will advertise their Local Area Networks (LANs) via BGP.

3.1 Configure EBGP on routers

In this section, you will configure EBGP on all routers.

Step 1. To configure BGP routing protocol, you need to enable the BGP daemon first. In router r1 terminal, type the following command to exit the vtysh session:

```
exit
```

Figure 25. Exiting the vtysh session.

Step 2. Type the following command on r1 terminal to enable and start BGP routing protocol.

```
bgd
```

Figure 26. Starting BGP daemon.

Step 3. In order to enter to router r1 terminal, type the following command:
Step 4. To enable router r1 into configuration mode, issue the following command:

```
configure terminal
```

Figure 28. Enabling configuration mode in router r1.

Step 5. The ASN assigned for router r1 is 100. In order to configure BGP, type the following command:

```
router bgp 100
```

Figure 29. Configuring BGP on router r1.

Step 6. To configure a BGP neighbor to router r1 (AS 100), type the command shown below. This command specifies the neighbor IP address (192.168.12.2) and the ASN of the remote BGP peer (AS 200).

```
neighbor 192.168.12.2 remote-as 200
```
Step 7. In this step, router r1 will advertise the LAN 192.168.1.0/24 to its BGP peers. To do so, issue the following command:

```bash
network 192.168.1.0/24
```

Figure 31. Advertising local network in router r1.

Step 8. Type the following command to exit from configuration mode.

```bash
end
```

Figure 32.Exiting from configuration mode.

Step 9. Type the following command to verify BGP networks. You will observe the LAN of router r1.
Step 10. Follow from step 1 to step 7 but with different metrics in order to configure BGP on router r2. All the steps are summarized in the following figure.

Step 11. Follow from step 1 to step 8 but with different metrics in order to configure EBGP on router r3 to establish BGP peering with routers r2 and r4. All the steps are summarized in the following figure.
**Step 12.** Follow from step 1 to step 8 but with different metrics in order to configure EBGP on router r4. All the steps are summarized in the following figure.

![Figure 36. Configuring BGP on router r4.](image)

**Step 13.** Type the following command to verify the routing table of router r1. Router r1 has a route to the network 192.168.2.0/24 only since IBGP is not configured between routers r2 and r3 yet.

```
show ip route
```

![Figure 37. Displaying the routing table of router r1.](image)

### 3.2 Configure IBGP on routers r2 and r3

In this section, you will configure IBGP on routers r2 and r3. Furthermore, you will configure BGP next-hop-self on routers r2 and r3 so that these routers have valid routes to the EBGP routes that are advertised by their IBGP neighbors.

**Step 1.** Type the following command on r2 terminal to establish IBGP peering with router r3.

```
neighbor 192.168.23.2 remote-as 200
```
Step 2. Type the following command to verify the BGP table of router r3. Router r3 can’t reach the network 192.168.1.0/24 since the next hop address (192.168.12.1) is not known to router r3.

```
show ip bgp
```

Step 3. Router r2 will configure BGP next-hop-self so that the neighbor 192.168.23.2 (router r3) can reach the EBGP routes advertised by router r2, such as 192.168.1.0/24, through router r2. To do so, type the following command on router r2 terminal.

```
neighbor 192.168.23.2 next-hop-self
```

Step 4. Type the following command to exit from configuration mode.

```
end
```
**Step 5.** In router r3 terminal, configure IBGP to peer with router r3. All the steps are summarized in the following figure.

![Configuring IBGP on router r3](image)

**Figure 42. Configuring IBGP on router r3.**

**Step 6.** Type the following command to verify the routing table of router r1. Router r1 has routes to the networks 192.168.3.0/24 and 192.168.4.0/24.

```
show ip route
```

![Displaying the routing table of router r1](image)

**Figure 43. Displaying the routing table of router r1.**

**Step 7.** In host h1 terminal, perform a connectivity between host h1 and host h4 by issuing the command shown below. To stop the test, press `Ctrl+c`. The result will show a successful connectivity test.

```
ping 192.168.4.10
```

![Connectivity test using ping command](image)

**Figure 44. Connectivity test using `ping` command.**
4 Perform IP spoofing and DoS attack

In this section, host h1 will spoof the IP address of host h4 and perform a DoS attack on it. Host h1 will send network traffic to hosts h2 and h3 with the source IP address of host h4. Thus, when hosts h2 and h3 receive the network traffic, they will reply to the source, i.e., host h4.

**Step 1.** Type the following command on h1 terminal. Host h1 is compromised and it will spoof the IP address of host h4 to perform a DoS against it. To do so, h1 sets an interface to the IP address of h4 first.

```
ifconfig lo 192.168.4.10
```

![Figure 45. Assigning an IP address to the loopback interface.](image)

**Step 2.** Type the following command on host h4 terminal. The command `tcpdump` allows you to capture the network traffic. The `-i` option allows you to specify the interface to be monitored (h4-eth0).

```
tcpdump -i h4-eth0
```

![Figure 46. Capturing packets on interface h4-eth0.](image)

Consider Figure 46. Currently, there is no network traffic directed on interface `eth0` of host h4. After launching the DoS attack from host h1, you will notice the network traffic redirected to host h4 using the `tcpdump` command.

**Step 3.** In host h1 terminal, issue the command shown below. The command used is `fping`. This command allows host h1 to ping multiple hosts. The `--src` option is followed by the source IP address. In this case, host h4 is configured with the source IP address (192.168.4.10). Then, the destinations IP addresses are specified, i.e. host h2 (192.168.2.10) and host h3 (192.168.3.10).

```
fping --src 192.168.4.10 192.168.2.10 192.168.3.10
```
Consider Figure 47. The two hosts h2 and h3 are unreachable since they will not reply to host h1. Instead, they will reply to the source IP address 192.168.4.10 which is host h4. This is how host h1 performs a DoS attack using different hosts.

**Step 4.** In host h4 terminal, observe the network traffic redirected from host h2 (192.168.2.10) and host h3 (192.168.3.10) towards host h4 (192.168.4.10).

To interrupt capturing the network traffic on interface eth0 of host h4 press `Ctrl+C`.

5 Mitigate DDoS attack by using IP source filtering

In this section, you will configure the ISP (router r2) to filter network traffic of Campus-1 based on their source IP addresses. Thus, mitigating IP spoofing and its possible attacks, such as DoS. To filter network traffic based on the source IP address, `iptables` utility will be used as FRR doesn’t support this feature. `iptables` is a command line program used to configure packet filtering on Linux operating systems.

**Step 1.** In router r2 terminal, type the following command to exit the vtysh session:

```
exit
```
Step 2. Type the following command on router r2 terminal to add a filtering rule. The option \texttt{A FORWARD} specifies that the rule added corresponds to incoming connections that are not being delivered locally. The option \texttt{-s} is used to specify the source IP address of the traffic. The option \texttt{-i} is used to specify the input interface receiving the traffic (r2-eth1). The option \texttt{-j} is to specify what to do if the packet matches. Briefly, in this command we are inserting a rule to accept all incoming packets on interface r2-eth1 (facing Campus-1) having a source IP address that belongs to the subnet 192.168.1.0/24.

\begin{verbatim}
iptables -A FORWARD -s 192.168.1.0/24 -i r2-eth1 -j ACCEPT
\end{verbatim}

After adding two filters on router r2, all incoming packets on interface r2-eth1 will be permitted if their IP address belongs to the subnet 192.168.1.0/24. Otherwise, the packets will be dropped and not forwarded to their destination.

Step 3. Similarly, to add another filtering rule on router r2, type the following command. The \texttt{-s 0/0} option matches all IP addresses. Briefly, we are dropping all incoming packets on interface r2-eth1.

\begin{verbatim}
iptables -A FORWARD -s 0/0 -i r2-eth1 -j DROP
\end{verbatim}

Step 4. We will launch another DoS attack from host h1 on host h4 and validate that the attack is mitigated. On host h4 terminal, type the following command to capture the network traffic on interface h4-eth0.

\begin{verbatim}
ttcpdump -i h4-eth0
\end{verbatim}
Step 5. On host h1 terminal, use the command `fping` to ping hosts h2 (192.168.2.10) and h3 (192.168.3.10) from the source IP address 192.168.4.10.

\[ \text{fping --src 192.168.4.10 192.168.2.10 192.168.3.10} \]

Step 6. In host h4 terminal, observe that even after the DoS attack was performed from host h1, host h4 did not receive any packet. Thus, the ISP (router r2) was able to filter the network traffic based on the source IP address and mitigate IP spoofing attacks.

To interrupt capturing the network traffic on interface `eth0` of host h4 press `Ctrl+c`.

This concludes Lab 12. Stop the emulation and then exit out of MiniEdit.

References