Ira A. Fulton Schools of Engineering
Arizona State University

CSE 548: Advanced Computer Network Security

Packet Filter Firewall (iptables) Project

Purpose

The purpose of this project is to provide students with hands-on experience in setting up and configuring a packet filter firewall, testing network connectivity, and ensuring the proper functioning of the firewall, installing software and services on VMs. This project enables students to grasp fundamental concepts like packet filtering and NAT. Through simulation of real-world scenarios, students learn to implement security policies effectively, troubleshoot issues, and optimize firewall configurations. This project fosters practical skills in translating security policies into iptables rules, preparing students for real-world network security challenges, and emphasizing the importance of firewall management in cyber security.

Objectives

Learners will be able to:

- Explain pros and cons of Iptables, a packet filtering firewall.
- Setup a stateless firewall and be able to differentiate between stateful and stateless firewalls
- Devise effective packet filtering firewall like Iptables
- Setup packet filter firewall (iptables) to allow and block network traffic
- Setup Network Address Translation (NAT) service
- Use basic networking and diagnostic tools such as ifconfig, ip, route, netstat, ping, traceroute, and tcpdump
- Use iptables to regulate network traffic and enable services such as Web service based on provided traffic policies

Technology Requirements

- Hardware

  - Intel or AMD based computer with 8GB or more memory.
  - NOTE: 6GB is technically sufficient, but performance will be sluggish.

- Software
  - VirtualBox 5.0 or Newer
  - Windows 10, Mac OS X, or Linux 64-bit as the base operating system.
  - Access to Apporto through a web browser
  - Linux: Ubuntu 18.04 LTS (Bionic Beaver)

Project Description

In this lab, you will explore the packet filter firewall by using Linux firewall iptables. The first part of the lab will set up necessary iptables running environments; the second part of the lab specify the requirements to implement firewall filtering rules to enable and disable network traffic.

The assessment of the lab is based on the completeness of implementing firewall filtering rules that satisfy the required firewall security policies. Students need to submit a sequence of screenshots and corresponding illustrations to demonstrate how they fulfilled the firewall's packet filtering requirements.

In summary, students will do:

- Set up network packet forwarding and inspect traffic using tools such as ifconfig, route, ip, ping, traceroute, and tcpdump

- Check services setup of apache2 web service

- Use and test iptables-based packet filter firewall to enable and disable access to the established services

Directions

You may work on your project locally or through Apporto. Follow the access guidelines provided and then review the Lab Preparation steps to complete your work. Headings or titles labeled with "Local only" are specific to the local setup, otherwise you may complete your work through Apporto as well. Please make sure your final deliverable follows the requirements.

Project Files

The project files include the VM image file, the assigned project, and associated background labs. These can be found in your course on the Project Overview page.

Local Setup

Setup your local environment based on the directions provided below:

1. Download VirtualBox 5.0 or newer.

2. Download the Virtual Machine Image File mentioned in the Project Files section

3. Extract the .vdi file from the downloaded zip file. Note that the image sizes are large and sometimes you may not be able to decompress to get your image file. In such case follow the instructions below:

    a. Go to your terminal and do: `unzip file_name -d /path/to/directory`

4. After extracting the image files, follow the instructions in the background lab **CS-SYS-00101 (VM in VirtualBox)** available in the **Associated Background Labs** zip file to set up the Virtual Machine.

5. Server credentials:
    a. Username: ubuntu
    b. Password: 123456

Accessing Apporto

For this project, you will work through Apporto's modular cyberlabs available through the course. Review all the project directions before starting so you know how to submit your work correctly:

1. In the course, click "**Apporto Virtual Lab – App Store**" (be patient while the lab loads)

2. In the App Store, select the "**CSE 548 Modular Cyberlab**" lab. Please be patient as the lab initializes, which may take a few moments.

3. Once the lab environment is ready, you will be on a Linux machine and you need to click on the "**GNS3**" app from the Activities Menu.

4. In the pop-up window, choose "Open a project from disk". Navigate to the path Home → Labs → projects → CSE548 Project 1 and then select the "**Project 1.gns3**" file.

5. You will see the following components:

    a. **Client** connected to a switch "**Switch2(10.0.2.0/24)**"

    b. **Gateway** connected to Switch2 and another switch "**Switch1**" which is connected to **Nat(10.0.1.0)**

6. Right-click on the Client/Gateway server and select “**Start”**, to start the server. You will know it’s running when the red block next to the server changes to green.

7. Again, right-click on the Client/Gateway server and select “**console**” to access the Linux desktop.

8. Server credentials:

    a. Username: ubuntu

    b. Password: 123456

9. You are all set! You can now start working on your project.

Note*: Learners should refrain from using the virtual machine (VM) for personal purposes. This VM is provided solely for academic and course-related activities. Any personal activities may interfere with its intended purpose and impact the learning environment for yourself and your peers.*

Lab Preparation

Assigned Project Lab and Associated Background Labs

- Unzip the zip file for the labs using: `unzip file_name -d /path/to/directory`

Background Labs

Preparatory "background" labs are available to help you develop proficiencies that you may not yet have but need to complete each project. Each project will list the recommended background. These are not graded and will not count towards your final grade. The project page within the Welcome & Start Here section of the course provides an overview of the background labs and course projects.

Associated Background Labs

- CS-SYS-00001 (Linux foundation);
- CS-SYS-00003 (Apache Web Service);
- CS-NET-00001 (Network Setup);
- CS-NET-00002 (Gateway-Setup)

Task 1 Preparation of Setting Up Lab Environment

Preparation:

1. **(Local only**) Review and exercise the CS-SYS-00101 (Virtual Box) to create a Client and Gateway/Server VM in VirtualBox on your computer.
2. **(Local or Apporto)** Review and exercise the following labs CS-SYS-00001 (Linux tutorial), CS-NET-00001 (Network setup) and CS-NET-00002(Gateway setup) on both Client and Gateway/Server VM before you do Task 1.1. (Set Gateway/Server VM as the Gateway for the Client VM)
3. **(Local or Apporto)** Review and exercise the Web Service Lab (CS-SYS-00003) on the Gateway/Server VM before you do Task 1.2.

Project Directions

In this lab, an iptables firewall running script template is provided, which allows you to manage and run your iptables rules easier. You can download from the Project Overview page in your course and put it on the Gateway/Server VM. The firewall script template file is a shell script. To make it executable, you can change its permission by (refer to more details in the lab CS-SYS-00001 on Linux file permission):

`$ sudo chmod 755 rc.firewall` this will change the file to green when you show `'ls -l'` command

The rc.firewall is a shell script to help you manage your firewall rules easier. It only includes some basic setup. To fulfill the overall goal of this lab, you need to add and update firewall rules in it. To edit and run the script file, you can:

`$ gedit rc.firewall` # use vim to edit the script. The script has comments that are sufficient to self-explain.
`$ sudo ./rc.firewall` # run the script

Task 1.1 Test network connectivity

The first step is to check the connectivity among VMs.

1. use ping to check connectivity (if this step is successful, please skip to Task 1.2):

   `$ ping ip_address` # you need to perform a mutual ping between any pair of VMs that are connected on the same local networks

Usually, unsuccessful ping responses can result in the following cases:

Ping response	**Possible Reason**
Request timed outs	timeout exceeds, e.g., windows default time out is 4s
No reply from <destination>	no response from the destination, the routers along the path working properly.
<destination> is unreachable	no response from the destination, the routers along the path are working properly.
ICMP host unreachable from gateway	the gateway/router forward you packets is improperly setup

2. Check if your default gateway is properly set up on your client VM (Don't change the gateway setting in the Gateway/Server VM). The default gateway should be set to the gateway/server's IP address. For example:

   ```
   $ route -n # check default gw setup
   $ sudo route add default gw <default_gw_ip> <interface_to_gw_net> # set
   default gw to the default gw IP through the directly connected interface
   ```

   For details on how to check the default gateway setup, you should refer to the lab **CS-NET-00002.** After checking/setting the default gateway configuration, perform ping to each other again.

3. If you can still not ping the gateway from the client or server, you may want to check if the firewall setup on the gateway blocked the ping. You may want to disable the firewall on the gateway and try to ping again:

   ```
   $ sudo ufw disable # disable the firewall
   ```

4. After checking the connectivity, you should check the packet forwarding setup on the Gateway/Server VM:

   (a) Enable packet forwarding on the gateway

   ```
   $ sudo echo "1" > /proc/sys/net/ipv4/ip_forward
   ```

   (b) To check your current iptables rules setup, you can issue the following command:

```
$ sudo iptables -L # display the filter table policies. For
whitelist, the default policy for INPUT, OUTPUT and FORWARD
chains should be DROP.
```

(c) On the gateway, clean up all existing iptables rules (used in care if you have already established iptables rules):

```
$ sudo iptables -F   #  flush all existing chains
$ sudo iptables -X   #  delete all user-defined chains
```

(d) Set iptables default policies to blacklist1. After the following iptables setup, you should be able to ping from any VM to other VMs.

```
$ sudo iptables -P INPUT ACCEPT #  option -P means default policy
$ sudo iptables -P OUTPUT ACCEPT
$ sudo iptables -P FORWARD ACCEPT
```

After the presented steps, your firewall rules are flushed, and no restriction to sending packets among VMS. Thus, you should be able to ping between any pair of VMs.

Task 1.2 Test installed software and services

The second step is to make sure the project-required software packages are installed properly on given VMs. Note that you may need to adjust the configurations of apache2 and make it accommodate to requirements presented below.

1. On the Gateway/Server, test the web server and make sure it is working properly:
   (a) Test Apaches server running the following command:

   ```
   $ service apache2 status
   ```

   (b) Establish a demo website by editing the file /var/www/html/index.html and add a statement such as “Welcome to Demo and Test!”. For more information about how to set up a web service, please refer to the system lab CS-SYS-00003 (Basic Web Service (Apache) Setup on Linux).

Task 1.3 Reset firewall to whitelist

After ensuring the network connectivity is good and the client accesses all the services described in Task 1.2, you need to enforce the whitelist firewall policy as the start point of your lab setup for the next task and flush out all existing firewall rules and chains on the Gateway/Server VM. The firewall whitelist policy means that the firewall only allows known legitimate traffic to pass through and it will drop all unspecified/unknown network traffic. After setting up the whitelist policy, you SHOULD NOT ping between the VMs and you should not be able to access the web service established on the Gateway/Server VM from the Client VM.

On the Gateway/Server VM, first, flush iptables chains and delete all user-defined chains:

```
$ sudo iptables -F #  flush ipables rules
$ sudo iptables -X # delete user-defined chains
```

Second, set the default iptables policy to whitelist:

```
$ sudo iptables -P INPUT DROP # option -P means default policy
$ sudo iptables -P OUTPUT DROP
$ sudo iptables -P FORWARD DROP
```

Now, you should not be able to ping between VMs and you cannot access web service from the client VM.

Task 2 Requirements for setting up a Stateless Packet filter firewall

On the Gateway/Server VM, please set up the following packet filtering rules.

1. Check and set the default iptables policies to DROP for INPUT, OUTPUT, and FORWARD chains. This setup basically implements a whitelist policy, i.e., only allowing specific network traffic as “good” traffic to pass through, and thus disable all other non-specified traffic. Note that only allow the required traffic flow and connectivity described in below, and drop all other network traffic and access.

2. Allow the client to access the web page (http) on the server by ip address of the Gateway/Server VM. The demo web page should contain the keyword “Welcome”, such as “Welcome to the demo and test web page!”

3. Stop the client from pinging the Gateway/Server VM’s IP address.

4. Allow the client to ping 8.8.8.8 (a public IP address on the internet).

Submission Directions for Project Deliverables

The deliverable for this project is a final report that contains a sequence of screenshots and explanations to show that they achieved the requirements described in the Lab Assessment section. A project report template can be found on the Project Overview page in your course.

You are given an unlimited number of attempts to submit your best work. The number of attempts is given to anticipate any submission errors you may have in regards to properly submitting your best work within the deadline (e.g., accidentally submitting the wrong paper). It is not meant for you to receive multiple rounds of feedback and then one (1) final submission. Only your most recent submission will be assessed.

You must submit your Packet Filter Firewall Project deliverables using the submission space in your course. Learners may not email or use other means to submit any assignment or project for review, including feedback, and grading.

The Packet Filter Firewall Project includes one (1) deliverable:

- **Project Report:** Use the Project Report template to submit your report as a DOC or PDF that includes your screenshots. Your file should be named using the following format: [Your Name_CSE 548_Name of Project or Assignment]

Report Submission

1. Navigate to your course and click on the "**Submission: Packet Filter Firewall**" submission space
2. Click "**Start Attempt**" in the upper right corner
3. Click "**Upload File**" and add "doc, pdf"
4. Locate and select your report file
5. When finished, click "**Submit Assignment**"
6. If needed: to resubmit the assignment
    a. Click "**New Attempt**" and add your file submission again
    b. When finished, click "**Submit Assignment**"

Evaluation

Lab assessment for accomplishing Task 1 and Task 2 depends on the following factors:

1. The client
    - can not ping the Gateway/Server VM IP address
    - can access the demo webpage on Gateway/Server VM by accessing the IP address of Gateway/Server VM in the browser (the returning page must contain "Welcome ....", you can also use a web browser)
    - can ping 8.8.8.8.
2. The Gateway/Server VM should

- set up the HTTP (webpage) service to its own IP address (with the demo page available).
- enable POSTROUTING to allow the client to access outside the network (8.8.8.8) and change their source IP addresses.

3. Additional requirements:

- You should set the default firewall policy to DROP for INPUT, OUTPUT, and FORWARD
- Chains.
- Besides the allowed network access described above, you should not allow any other network access. Provide screenshots for the following results:

On client VM

```
$          sudo nmap -sT -p- 10.0.2.x % x is the value of your
              Gateway/Server VM’s IP address
$          sudo nmap -sU -p- 10.0.2.x % x is the value of your
               Gateway/Server VM’s IP address
$          ping 8.8.8.8 % This should be working
$          ping 8.8.4.4 % This should be not working, as you should drop
                          all traffic that is not required in the
                          requirement.
$           ping 10.0.2.x % x is the value of your Gateway/Server VM’s IP
               address, This should be not working
```

On Gateway/Server VM

```
$          ping localhost % This should be not working
$          ping 10.0.2.y % y is the value of your client’s IP address,
           this should be not working
$          ping 8.8.8.8 % This should be not working
```