Exercise for Lecture Software Defined Networking

Prof. Dr. David Hausheer
Julius Rückert, Leonhard Nobach

Winter Term 2015/16
Exercise No. 7
Published at: 19.01.2016
Submission exclusively via Moodle, Deadline: 02.02.2016
Contact: `[rueckert|lnobach]@ps.tu-darmstadt.de`
Web: `http://www.ps.tu-darmstadt.de/teaching/ws1516/sdn/`
Submission: `https://moodle.tu-darmstadt.de/enrol/index.php?id=6349`

Surname (Nachname):	
First name (Vorname):	
ID# (Matrikelnummer):	

Connect via SSH to the host **studlab.ps.tu-darmstadt.de**, but use the alternative port **22XX** where XX is your group number (For example, Group 02 uses port 2202). You will obtain access if you have submitted a valid SSH key to us.

Solve the tasks by editing `lab_task.c` in the directory `~/nfvlab` (`cd ~/nfvlab`). You can edit the file locally with `nano` (`nano lab_task.c`) or copy the file temporarily to your machine using an SCP client.

If you have finished editing, compile with `make`. Run your application with `sudo sh ./run.sh`. If you reboot your VM, you will have to enter `sudo sh ./configure-before-run.sh` once.

In the lab, there is a packet generator, which continuously sends packets to your VM. It also tests the packets which are sent back from you VM. Under `http://studlab.ps.e-technik.tu-darmstadt.de/lab/report_XX.html`, you can see if you have solved all assignments correctly. You agree that you will use the VM **only for solving the lab tasks**, and for nothing else. Access to the VMs and operations on them are monitored.

Addendum: Access is only possible from within the university network (starting with 130.83...). Please use the TUD VPN, if you are away from the campus.I totally forgot to write anything about usernames.

Addendum 2: The username is "labuser" for everyone of you. For running the application, you need root privileges. The password for sudo is placed in a small file in your home directory.

Don't always work with root privileges, you could accidentally break something. Likely not the infrastructure, but your VM. In this case, we have to reinit it, which deletes your current solution. So, please make backups of your solution (lab_task.c).

Problem 7.1 - Layer 2 Echo

Whenever a packet is sent to your VM and destined to **00:15:17:03:00:01**, echo it back to the originating MAC address.

Problem 7.2 - IP Rewriting

Whenever a packet is sent to your VM and destined to **00:15:17:03:00:02**, change the destination MAC address to **00:15:17:04:00:02**. The source MAC address should become the *original* destination address.

Furthermore, if it is a packet destined for `192.168.0.X` and uses the UDP destination port 80, rewrite it to `192.168.1.X` and change the destination port to 8080. Remember to recalculate the checksum!

Problem 7.3 - Stateful Tunneling VNF

Whenever a packet comes into the VM destined to **00:15:17:03:00:03**, encapsulate the *whole* Layer 2 frame into a custom tunneling protocol on top of an additional encapsulating Ethernet header.

The encapsulating Ethernet header should have the destination address **00:15:17:04:00:03**, and the source address **00:15:17:03:00:03**. Your custom tunneling protocol should use the ethertype `0x0803` in the encapsulating Ethernet header. The packet header of your tunneling protocol is comprised of the following (all *network byte order*):

- A 16-bit preamble: `0x3434`
- A 64-bit counter, increasing by 1 for every packet. The starting value of this counter does not matter.

The original Ethernet header should follow on top of this header:

```
[Ethernet][0x3434(16bit)|Counter(64bit)][Payload WITH original ethernet header]
```