ARIZONA STATE UNIVERSITY

CSE 434, SLN 70516 — **Computer Networks** — Fall 2022

Lab #1

Lab report due electronically on Canvas before 11:59pm on Sunday, 09/11/2022

This lab has two major parts:

1. An introduction to the racks of networking equipment in BYENG 217 in §2 through exercises and experiments with a single-segment IP network in §3. Be sure to take a USB flash drive to the lab with you to save data from your experimentation.
2. An introduction to CloudLab in §4 through an experiment to set up and test your `ssh` keys, and solving a problem in designing subnets on CloudLab with exercises and experiments in §5. Be sure to solve the subnet design problem prior to trying to implement it on CloudLab.

A supplementary document for Lab #1 is provided in case you are not familiar with Linux commands such as `ifconfig`, `ping`, and the Wireshark application.

Immediate "To do."

1. Join a group and reserve a timeslot before Sunday, 09/04/2022; see §1.1 for instructions.
2. Join the CloudLab project for this class before 09/10/2022. If you do not, *your request will not be approved!* See §4 for instructions.

1 Preliminaries

1.1 Join a Group on Canvas, Reserve a Timeslot on a Rack in BYENG 217

Labs may be completed individually, or in a group of size two; either way, everyone must join a group so that grades are recorded correctly to members in a group. On Canvas, go to People/Groups and choose the `Lab Groups` tab. Self sign-up is enabled. Please lock the group once it includes who you want in it.

Once you've created a group, reserve a three-hour slot on one of the six racks in BYENG 217 using this sign-up sheet.

See also the thread on Ed Discussion on Canvas in case you need help with either of these two steps.

1.2 Lab Report Format

For all lab reports, be sure to include your group number, and the names of the member(s) in your group on the first page of your report.

The exercises to include in your lab report are found in boxes labelled with an exercise number. (This lab has a total of six exercises.)

For each exercise:

1. Give the exercise number.
2. Include the output and/or screen shots as requested in the exercise. Be sure to configure the prompt on the hosts on the racks in BYENG 217, and on CloudLab, to include the ASUrite id of a group member and include it as part of any screen shots you are asked to take to show that you generated the output/screen shot. (This is *not* your student id number; this is the id you use to login to, e.g., `myASU`.)
3. Provide explanations and/or answer any questions, as requested in the exercise.

As you prepare your report, extract the portion of your output (saved to a file, or screen capture) that relates directly to the exercise. Do not just copy and paste in pages of output hoping the grader can find what's relevant; that is your determination.

2 Introduction to the Racks in BYENG 217

In this part of Lab #1, you will acquaint yourself with the racks of equipment in BYENG 217, the Linux operating system, and some traffic measurement tools.

Becoming Familiar with the Equipment

The equipment that you are working with in the lab has a setup similar to that in Figure 1.

Figure 1: Equipment in a rack in BYENG 217.

Take a few minutes to compare the following description with the actual equipment:

- A 19 inch rack that houses most of the equipment.
- Four Linux PCs, which are labelled *A*, *B*, *C*, and *D*. The PCs run the Redhat Linux operating system. Each PC has two Ethernet network interface cards (NICs), labelled *p2p1* and *p2p2*.
- Four Cisco routers (silver, near top of rack beneath the SDN switch) are labelled as *A*, *B*, *C*, and *D*.
- Four Ethernet switches (blue, near middle of rack), each with eight ports. The data rates of the ports are 10 Mbps, 100 Mbps, or a dual speed at 10/100 Mbps.
- Two Ethernet hubs (white, near bottom of rack), each with 5 ports.
- A monitor, a keyboard, a mouse, and a KVM (keyboard-video-mouse) switch. The KVM switch connects the keyboard, monitor, and mouse to a PC. The KVM switch gives you control over all four PCs from one keyboard, monitor, and mouse, but you can access only one computer at a time.
- Ethernet cables (hanging over top rack). Note that there are two kinds: straight-through Ethernet cables and crossover Ethernet cables. The crossover cables should be colour-coded or labelled. In this lab, only straight-through Ethernet cables are used.
- The figure shows light-blue coloured serial cables from the routers to the PCs; these are not used in this lab.

Using the KVM switch, logging in to a Linux PC, and exploring the desktop.

1. Set the KVM switch to computer A (the first light labelled A should light up). Log in as `root`. To do so, select "other" then enter `root` as the account you want to login to; no password is needed.
2. Use the KVM switch to switch to computer C and log in as root.
3. Explore the desktop environment of computer C.
4. Create a terminal window by selecting `Terminal` under the `Applications/Systems Tools` menu. Recall that all Linux commands can be typed from a terminal window.
5. Set the KVM switch to computer A and reboot it by typing `reboot` on the command line at the prompt in the terminal window.
   ```
   [root@hostA ~]# reboot
   ```

Set-up of the network.

In this part of the lab, all four Linux PCs are attached to the same Ethernet switch. (You may need to power-up the equipment!)

1. Attach each Linux PC to the *same* Ethernet switch with (straight-through) Ethernet cables. That is, for each computer, connect one end of an Ethernet cable to the interface with label *p2p1* and the other end of the cable into an Ethernet switch.
2. When you reboot the Linux PCs, the IP addresses of the computers should be configured as shown in Table 1. The IP addresses listed in the table are associated with the Ethernet card labelled *p2p1*. In this lab, the second Ethernet card of the Linux PCs, labelled *p2p2*, is not used.

Table 1: Default IP Address Set-up

Linux PC	IP Address and Mask of Interface *p2p1*
Computer A	`10.0.1.11/24`
Computer B	`10.0.1.12/24`
Computer C	`10.0.1.13/24`
Computer D	`10.0.1.14/24`

Testing connectivity between computers.

After connecting the four Linux PCs to the Ethernet switch, all four computers should be able to communicate with one another. The following steps verify that the Linux PCs are properly connected. The test consists of running a `ping` test between two Linux PCs. (If you are not familiar with the `ping` command see the section `Using Ping` in the Lab #1 Supplementary document.)

1. On each PC, first verify that the IP address and mask are set according to Table 1 by issuing the following command on each PC. Among other things, the output shows you the IP address and mask of the `p2p1` interface on the computer on which it is issued:
   ```
   ifconfig p2p1
   ```
   If the machines are not configured as expected, see §3.2 to reconfigure the IP addresses and masks.
2. If the machines are configured according to Table 1, then you should be able to ping each host from any other one as they are all on the same /24 network, *e.g.*, from computer A, B, or C, the following command `ping`s computer D:
   ```
   ping -c 5 10.0.1.14
   ```

3 Single-Segment IP Networks

In this section of the lab, you become acquainted with IP configuration issues on a single Ethernet segment using the equipment in BYENG 217.

- If you have not already done so, before you get started, it is a good idea to reboot the Linux PCs.
- During the lab, you need to save data to files. Save all files in the directory `/root`.
- Copy your files to a USB flash drive before the end of the lab. You will need the files when you prepare your lab report.

3.1 Set-up

- The set-up for this section of the lab is identical to that described in §2. All Linux PCs are connected to the same Ethernet segment by a single switch.
- The IP addresses for the Linux PCs should be configured as shown in Table 1. Whenever a Linux PC is rebooted, the IP addresses should be set to the values displayed in the table; if not, see §3.2.

3.2 Configuring IP Interfaces in Linux

The `ifconfig` command is used to configure parameters of network interfaces on a Linux system, such as enabling and disabling of interfaces, and setting the IP address. The `ifconfig` is usually run when a system boots up. Once the Linux system is running, the `ifconfig` command can be used to modify the network configuration parameters. See the `man` page for `ifconfig` for details.

This list shows how `ifconfig` is used to query the status of network interfaces.

- `ifconfig`
  Displays the configuration parameters of all active interfaces.
- `ifconfig -a`
  Displays the configuration parameters of all network interfaces, including the inactive interfaces.
- `ifconfig interface`
  Displays the configuration parameters of a single interface. For example, `ifconfig p2p1` displays information on interface `p2p1`.

There are numerous options for configuring a network interface with `ifconfig`. The examples in this list shows how to enable and disable an interface and how to change the IP configuration.

- `ifconfig p2p1 down`
  Disables the `p2p1` interface. No traffic is sent or received on a disabled interface.
- `ifconfig p2p1 up`
  Enables the `p2p1` interface.
- `ifconfig p2p1 10.0.1.8 netmask 255.255.255.0 broadcast 10.0.1.255`
  Assigns interface `p2p1` the IP address 10.0.1.8/24 and a broadcast address of 10.0.1.255.

Exercise 1: Change the IP address of an interface. Use the `ifconfig` command to modify the IP address of the `p2p1` interface of computer D.

1. On computer D, run `ifconfig -a` and save the output.
2. Change the IP address of interface `p2p1` of computer D to 10.0.1.16/24.
3. Run `ifconfig -a` again and save the output.

Include the saved output in your report and explain the fields of the `ifconfig` output.

3.3 Changing Netmasks

Now, you will test the effects of changing the IP addresses and netmasks of a network configuration. If you are not familiar with using Wireshark, be sure to read through the *Basics of Wireshark* in the Lab #1 Supplementary document.

Table 2: IP addresses for Exercise 2

Linux PC	IP Address of Ethernet Interface *p2p1*	Network Mask
Computer A	`128.143.71.201/16`	`255.255.0.0`
Computer B	`128.143.71.21/24`	`255.255.255.0`
Computer C	`128.143.137.144/26`	`255.255.255.192`
Computer D	`128.143.137.32/26`	`255.255.255.192`

Exercise 2: Changing netmasks.

1. Configure the interfaces of the hosts to those given in Table 2.
2. Run Wireshark on computer A and capture the packets for the following `ping` commands (running each on the "from" computer). Save the Wireshark output to a text file (clearing the *Packet details* option), and save the output of the `ping` commands, including any error messages.
   (a) From A to C: `ping -c 3 128.143.137.144`
   (b) From A to B: `ping -c 3 128.143.71.21`
   (c) From A to D: `ping -c 3 128.143.137.32`
   (d) From D to A: `ping -c 3 128.143.71.201`
   (e) From B to D: `ping -c 3 128.143.137.32`
   (f) From B to C: `ping -c 3 128.143.137.144`

Include selected output from each `ping` in your report and briefly explain the `ping` result.

IMPORTANT: Before you leave the lab:

1. **Make a copy of all files you've created on each computer onto your USB flash drive and then delete the files!**
2. **Reset the interfaces to their original values given in Table 1.**
3. **Disconnect the Ethernet cables from the Ethernet switch and from the PC interfaces.**

4 An Introduction to CloudLab

4.1 Join the CloudLab Project

Before you can run lab experiments on CloudLab, you will need to set up an account. **You must set up you account before 09/10/2022,** *i.e.*, **Friday, 09/09/2022 or earlier.** Follow the instructions given in section *Exercise - Create an account* of your first CloudLab experiment.

Fill in your personal information, being sure to give your ASU e-mail address. Join the project named `ASU-CSE-434` that exists for this course. If you have a public key, upload it here. If you've never set up `ssh` keys follow the instructions given in section *Exercise - Set up SSH keys* of the first CloudLab experiment to set up your `ssh` keys. `ssh` keys are required to be able to add any resources into a CloudLab slice.

WAIT FOR YOUR REQUEST TO BE APPROVED; this may take up to 24 hours.

In general, CloudLab prefers the Chrome and Firefox browsers. Be sure to whitelist `cloudlab.us` if you run an ad blocker.

4.2 A First Experiment on CloudLab

Once your request to join the `ASU-CSE-434` CloudLab project has approved, you can move on to section *Reserve and log in to resources on CloudLab* in your first CloudLab experiment. The goal of this experiment is simply to test that your `ssh` keys are set up properly. If they are, you will be be able to login to a virtual machine in your slice without having to provide a password. If they are not set up properly, you will not be able to `ssh` successfully into any virtual machine. If something has gone wrong, try again, *i.e.*, work through the steps to set up your `ssh` keys again until you are able to successfully complete Exercise 3.

Exercise 3: Testing `ssh` keys. *As stated in §1.2, be sure that all screen shots include the ASUrite id of a group member, i.e., set your login prompt when you `ssh` into a VM.*

1. Once your topology is ready (*i.e.*, in the Topology View, your VM will be green and have a "⊙" icon in the top right corner), take a screen shot of your topology and include it in your report.
2. Continue to follow the instructions, logging into your VM using the terminal in the CloudLab web portal. Take a screen shot of this terminal shell after executing the `echo` command.
3. You should also try using your own terminal application, and using VNC.
4. IMPORTANT: Follow the instructions in the section *Release resources* to return the resources used by your VM in this slice when you are finished.

5 Designing Subnets

Do not proceed with this part of the lab if you have not completed the first experiment in §4.2.

Follow the instructions in Designing Subnets, always following the CloudLab specific instructions. In this experiment, your task is to set up subnets in a few small *local area networks* (LANs) to meet given design requirements.

Read the section *Background* and work through the detailed example provided. If you understand this example, you should be able to solve the subnet design problem that is the goal of the lab.

Be sure to solve the subnet design problem (Exercise 4) on paper before moving on to implementing your design on CloudLab (Exercise 5).

Exercise 4: Design Subnets. Solve the subnet design problem posed in the section **Challenge: Design Subnets** that satisfies the requirements given.

For each of the three subnets in the design problem, provide a table in your report giving:

1. The subnet mask.
2. The network address.
3. The smallest IPv4 address that may be assigned to a host in the subnet.
4. The broadcast address for the subnet.
5. The highest IPv4 address that may be assigned to a host in the subnet.

Be sure to follow the conventions on addressing given, *i.e.*, assign the lowest IP address in the subnet to the LAN-facing interface of the router, assign the highest IP address in the subnet to one node, and any other IP address in the subnet to the other node.

In the section **Implement your Design on each LAN**, in each LAN, use `ifconfig` command to configure the IP address and subnet masks of each host in the LAN. After configuration, each host should be able to reach every other host in the *same* LAN.

Exercise 5: Implement your Design. *Remember that all screen shots must include the ASUrite id of a group member, i.e., set your login prompt when you* `ssh` *into a VM.*

Configure the IPv4 address and subnet mask of each host in each LAN using your solution to Exercise 4. Recall the instructions on configuring IP addresses from §3.2.

1. Take a screen shot showing the output of the `ping -c 5 IP` command between two hosts in the same LAN, *e.g.*, between `romeo` and `juliet` on LAN A, and include each one in your report. (You should have 3 screen shots, one for each LAN.)
2. Take a screen shot showing the output of the `ping -c 5 IP` between a host in LAN A and a host in LAN B (*e.g.*, between `romeo` and `othello`), between a host in LAN B and a host in LAN C, and between a host in LAN C and a host in LAN A, and include them in your report. (Again, you should have 3 screen shots.)

If your `ping` output is not as expected, you may need to review your solution to the design problem (Exercise 4) and/or check your node configurations.

In the section **Adding Routing Rules**, on each host, follow the instructions for adding a rule that describes how to reach the other two LANs. Use the `route add` command for this purpose and follow instructions very carefully. (Do not delete any existing rules!)

In addition to adding routing rules to every host, you also need to add a rule on each router.

If you make a mistake you can use the `route del` command and try again. When you have set up these rules correctly, every host in every LAN should now also be able to `ping` every host in *every other LAN* in your topology.

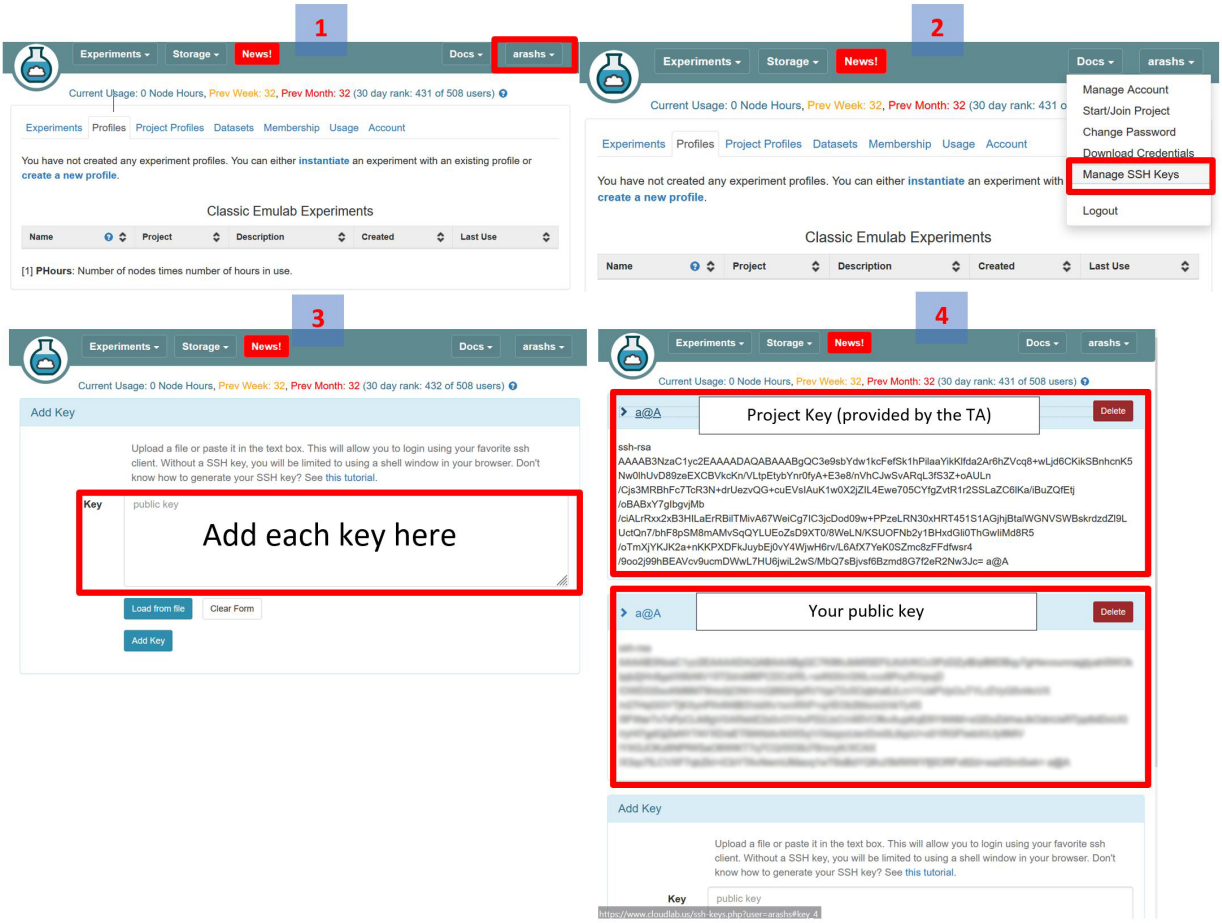
Exercise 6: Add Routing Rules. Configure the route on each host in each LAN, and also on each router.

1. Take a screen shot of the output of the `route -n` command on each node and on each router and include it in your report. (You should have 9 small screen shots.)
2. Take a screen shot of the output of the `ping -c 5 IP` command between a host in LAN A and a host in LAN B, between a host in LAN B and a host in LAN C, and between a host in LAN C and a host in LAN A and include them in your report. (You should have 3 screen shots.)
3. Follow the instructions to configure the routing rules for the 10.10.100.0/24 network. Now, take a screen shot of the output of the `traceroute IP` command between two nodes in different LANs and include it in your report. (The `traceroute` should be able to trace the route taken.)

Once you have completed all exercises in the *Designing Subnets* portion of this lab, and you no longer wish to experiment in your slice, delete the resources in the slice to free them up for other experimenters.

Appendix

If you want our TA to be able to help you debug your CloudLab slices, follow the four steps in the figure to upload his public key. It is provided in the file `ara.pub` on Canvas under `Modules/Lab #1`.


1
Experiments
Storage
News!
Docs
arashs
Current Usage: 0 Node Hours, Prev Week: 32, Prev Month: 32 (30 day rank: 431 of 508 users)
Experiments
Profiles
Project Profiles
Datasets
Membership
Usage
Account
You have not created any experiment profiles. You can either instantiate an experiment with an existing profile or create a new profile.
Classic Emulab Experiments
Name
Project
Description
Created
Last Use
[1] PHours: Number of nodes times number of hours in use.
2
Manage Account
Start/Join Project
Change Password
Download Credentials
Manage SSH Keys
Logout
3
Add Key
Upload a file or paste it in the text box. This will allow you to login using your favorite ssh client. Without a SSH key, you will be limited to using a shell window in your browser. Don't know how to generate your SSH key? See this tutorial.
Key
public key
Add each key here
Load from file
Clear Form
Add Key
4
a@A
Project Key (provided by the TA)
Delete
ssh-rsa
AAAAB3NzaC1yc2EAAAADAQABAAABgQC3e9sbYdw1kcFefSk1hPilaaYikKlfda2Ar6hZVcq8+wLjd6CKikSBnhcnK5
Nw0lhUvD89zeEXCBVkcKn/VLtpEtybYnr0fyA+E3e8/nVhCJwSvARqL3fS3Z+oAULn
/Cjs3MRBhFc7TcR3N+drUezvQG+cuEVslAuK1w0X2jZIL4Ewe705CYfgZvtR1r2SSLaZC6lKa/iBuZQfEtj
/oBABxY7glbgvjMb
/ciALrRxx2xB3HILaErRBilTMivA67WeiCg7IC3jcDod09w+PPzeLRN30xHRT451S1AGjhjBtalWGNVSWBskrdzdZl9L
UctQn7/bhF8pSM8mAMvSqQYLUEoZsD9XT0/8WeLN/KSUOFNb2y1BHxdGli0ThGwliMd8R5
/oTmXjYKJK2a+nKKPXDFkJuybEj0vY4WjwH6rv/L6AfX7YeK0SZmc8zFFdfwsr4
/9oo2j99hBEAVcv9ucmDWwL7HU6jwiL2wS/MbQ7sBjvsf6Bzmd8G7f2eR2Nw3Jc= a@A
Your public key
Delete
Add Key
Upload a file or paste it in the text box. This will allow you to login using your favorite ssh client. Without a SSH key, you will be limited to using a shell window in your browser. Don't know how to generate your SSH key? See this tutorial.
Key
public key
https://www.cloudlab.us/ssh-keys.php?user=arashs#key_4