- **Problem 1** (6 points) **Due by 17:00, Feb. 16**

  1. Use the packet analyzing tool `tcpdump` to capture an ICMP destination unreachable message from one of the four networks `172.x.0.0/16` and show the captured message.
  2. Use the packet analyzing tool `tcpdump` to capture an ICMP redirect message from one of the four networks `172.x.0.0/16`; show the captured ICMP redirect messages; and show the changes of the routing cache and the routing paths caused by the ICMP redirect message.
  3. Select a pair of hosts A and B, one in network `net17` and one in network `net18`. Use `tracepath6` on A to test the reachability of B. Use the packet analyzing tool `tcpdump` to capture an IPv6 datagram in `net17`, `net18` or `net19` which carries the data from your `tracepath6` request, and capture an IPv4 datagram in `net16` which carries an IPv6 datagram for the `tracepath6`. Show the captured datagrams.

  Hints:

  You need to use the packet analyzing tool `tcpdump` to capture the targeted messages. You may need to read the manuals for `tcpdump` before you start working on the assignment. The following web sites might be helpful: `www.tcpdump.org`. To run `tcpdump`, you need to use `sudo` command (e.g., `sudo tcpdump`). Put your commands for the testing into scripts. You probably can not run `tcpdump` on any of the four routers. So please test your program on a client machine which is not a router.

  A graphic user interface (GUI) is supported in the packet analyzing tool `wireshark`. So you may get a better intuition on how to capture a data packet by `wireshark` first. To run `wireshark` on a client machine, a GUI on the client machine is needed. You can use `ssh -Y` to logon to the client machine to have the GUI. Although you are encouraged to try both `tcpdump` and `wireshark`, your submitted work should use `tcpdump`.

  A router R issues an ICMP redirect message to a host H if R forwards a packet P sent by H to the same interface from which P is received. For example, the next hop router in the routing table of router `january` for the `default route` (destinations in `net18` and `net19`) is `december.net16`. If router `january` receives a packet P sent by H from interface `eth1` (`net16`) and P's destination is in `net18` (say `may.net18`) or `net19` (say `year.net19`) then router `january` will forward P to `december.net16` through interface `eth1` and issues an ICMP redirect message to H.

  You need to modify the routing table on a host H to cause the above problem. For example, the next hop router in the routing table of machine `april` (and every client machine connected to `net16`) for the `default route` (destinations in `net17`, `net18` and `net19`) is `december.net16`. If you change the the next hop router in the routing table of `april` for the `default route` from `december.net16` to `january.net16` and send a data packet from `april` to a destination in `net17`, `net18` or `net19` then you will have the above problem. You can use `if route del` and `if route add` commands to modify the routing table of a client machine (you need to use `sudo` to run these commands). You need to observe and report the changes of the routing cache at H and routing path caused by the

ICMP redirect message. You can use `ip route` or `route` command (with `-C` option) to view the contents of a routing cache.

To show the changes of the routing cache in a client machine H clearly, you may need to restore the routing table of H to its original configuration by commands `ifdown eth1; ifup eth1` first and then modify the routing table. You may use `ip route flush cache` to clear the routing cache and use `ip route get xxxx` (`xxxx` is the IP address of a destination) to initialize the routing cache to the destination `xxxx` with the contents of the routing table. You need to use `sudo` for the above changes. To show a routing path, you may use `tracepath` or `ping -R` commands. After you capture the ICMP redirect message and observe the changes of routing cache and routing path caused by the ICMP redirect message, **use `ifdown eth1; ifup eth1` to restore H into its original configuration.**

Notice that `ifdown eth1` turns down interface `eth1` and `ifup eth1` turns up `eth1`.

WARNING: never turn down interface `eth0` of any machine; if you do this in a machine H, you disconnect H from network `192.168.0.0/24` and may not be able to resume the connection.

Networks `net17`, `net18` and `net19` support both IPv4 and IPv6, while network `net16` supports IPv4 only. Each interface of a client machine connected to `net18` is assigned two IPv6 ULA addresses, one with `netid 18` and one with `netid 8018`. An IPv6 datagram from `net17` to a destination with `netid 18` is routed through `net19` and an IPv6 datagram from `net17` to a destination with `netid 8018` is routed through `net16`.

Consult with `man` pages for the related commands.

Your scripts must work in the virtual network and the output must be the data from the virtual network.

You need to submit your programs to the directory `/Assignment/2/yourid` (`yourid` is your login name to the VNL) in the gateway machine (`cs-vnl-01.csil.sfu.ca`, `cs-vnl-02.csil.sfu.ca`, `cs-vnl-03.csil.sfu.ca` or `cs-vnl-04.csil.sfu.ca`) by the due time. Please have clear comments in your shell scripts to explain how they work. In addition to your programs, please also submit a text file to explain how did you conduct your experiments and how to view your results.

Problem 2: (0 points) Solving the following problems will help you to understand the masterials we have discussed. You do not need to submit your answers. Please ask if you have any question.

1. Find the IPv6 ULA address and the link local address assigned to interface `eth0` of machine `may` in the VNL. Each of the IPv6 addresses can be partitioned into two parts: part I for identifying network `admin` and part II for identifying interface `eth0`. Give the uncompressed hexadecimal notations for part I and part II of the ULA address and link local address.

2. Find the Ethernet address assigned to `eth0` of machine `may`. Is the Ethernet address embedded in part II of the IPv6 ULA address for `eth0` of `may`? Is the Ethernet address embedded in part II of the IPv6 link local address for `eth0` of `may`? If your answer is yes for one of the above questions, explain how the Ethernet address is embedded.

3. Binary trie is a data structure used for efficient address lookup at a routing table. Assume that a routing table has the following network addresses: `00010`, `10010`, `00100`, `01010`, `11010`, and `10110`. Draw a binary trie for looking up the network addresses in this routing table.

4. Assume that an IPv4 datagram D with 1120 bytes of data is fragmented at a router into two datagrams, D_1 with the 1st 640 bytes of the data in D and D_2 with the rest data. Let M_1 and M_2 be the *the more fragment bits* in the IP headers of D_1 and D_2, respectively. Let Off_1 and Off_2 be *the fragmentation offsets* in the IP headers of D_1 and D_2, respectively. Give the values of M_1, M_2, Off_1, Off_2.

5. Describe briefly how ARP works and give the ARP message format.

6. Assume that networks N_1 and N_2 are connected by router R which performs Proxy ARP to allow N_1 and N_2 to use a same network address. Describe how a host A connected to network N_1 sends a datagram to a host B connected to network B_2 via Proxy ARP.

7. Company A has a naming hierarchy `xxxx.A.com` and Company B has a naming hierarchy `yyyy.B.com`. The names `xxxx` and `yyyy` are given independently (i.e., a popular name can be used in both `xxxx` and `yyyy`. Now B becomes part of A. Give a naming hierachy for all machines in A and B without changing `xxxx` nor `yyyy`.

8. Find the global IP address of a gateway to the virtual network lab. Draw the subtrees of the tree that define the full domain name and the inverse domain name for the gateway.

   In the virtual networks, `seasons` is the only machine that runs DNS server. The DNS server related files can be found at `/etc/bind/` and `/var/cache/bind/` at `seasons`. Study these files and give the tree which defines the hierarchical domain space and machine names connected to networks `admin, net16, net17, net18, net19`.

9. TCP uses positive acknowledgment and retransmission for reliable transmissions. Does a lost data segment always cause a retransmission? Does a lost ACK message always cause a retransmission? Explain briefly your answers.

10. TCP uses 32 bits for data stream sequence numbers. How does this allow a stream of arbitrary length transmitted? How many bytes can a stream have if each byte in the stream must be assigned a distinct sequence number?

11. Assume that both the sender and the receiver in a TCP connection has a window size of k bytes. What is the minimum number of distinct sequence numbers for the stream in the TCP connection such that every byte in the stream can be uniquely identified at both sides?

12. The sender in a TCP connection is using a window size of 1000 and the previous ACK number is 2000. Now the sender receives a segment with ACK number 2500 and window size 800. Show the changes of the sender's window by figures. Show the changes of the window if the window size in the received ACK segment is 1200.