

CO302: Computer Networks Lab

Tentative weekly schedule

(July - December 2019)

Week 0 (Socket Programming)

- Create a TCP/IP client-server program in which a single client connects to the server. The server then should send a greeting to the client. Repeat the experiment using UDP.
- Create a TCP/IP client-server program. The client should send a message asking for the current date. The server should respond with the current date that will be displayed on the client.
- Create a TCP/IP server-client program in which both client and server can send and receive messages (Chat application). Repeat the experiment using UDP.
- Create a chat room such that multiple clients can connect to a server. The clients must be able to both send and receive messages to each other via the server. Create chat rooms using TCP and UDP.

Week 1 (Basics of ns-3)

- What is simulator?
 - Types of simulators real virtual constructive
 - Why is a network simulator needed?
 - Why ns-3? packet analysis plot the graphs
 - What are Wireshark, TCPDUMP, gnuplot, NetAnim, PyViz?
 - What is WAF? waf is a python-based build tool. network animation
 - What is wscript? integrate files
 - Helper Classes vs Containers in ns-3
 - What are callbacks in general? How they are used in ns-3?
 - What are smart pointers? deallocated the memory when not in use
 - What is a Flow Monitor?
 - Different topologies in ns-3
 - Different modules of ns-3
 - Download and install ns-3
 - Run examples of ns-3
- Mesh Topology :
Star Topology :
Bus Topology :
Ring Topology
Hybrid Topology :
- Helpers and Containers, both are high level APIs, and are used to make programming in ns-3 easier (as you said in case of helper). The difference is that helper APIs create a single object of a class whereas container APIs can create multiple objects of same type. For example, when you want to create multiple nodes in ns-3, you use NodeContainer API which internally creates the specified number of nodes. Similarly, when you want to make a CSMA channel, you use a CsmaHelper which creates a single channel object.

Week 2 (Beginner tasks in ns-3)

- How to install specific modules of ns-3?
- Visualize the output of ns-3 examples by using **PyViz** and NetAnim
- **What** are test cases and test suites in ns-3? How to run and write test cases in ns-3?
- Ping a real host by using the emulation feature of ns-3
- Plot a Line graph for TCP congestion window (cwnd) using gnuplot

.tr file

- Enable support of ASCII traces in first.cc and collect the statistics
- Write your own packet sink, on-off, bulk send, UDP echo server applications. Get creative. Make sure the example highlights the type of application you're using

Week 3 (Intermediate tasks in ns-3)

- Use Valgrind to understand memory leaks using first.cc
- Set breakpoints at three places in star.cc and list all of them using gdb (GNU Debugger)
- Print the data type of the variables in star.cc using gdb (GNU Debugger)
- Modify first.cc to support IPv6 addressing
- Enable support of **flow monitor** in tcp-bulk-send.cc
- Compare the congestion window plots for TCP Newreno and TCP Highspeed
- Install ns-3 Direct Code Execution (DCE) in advanced mode and run dce-iperf.cc example
- Add comments in at least 3 classes or examples of your choice. Get creative. Make sure somebody who is not an expert in ns-3 is able to understand the comments.

Week 4 (Advanced tasks in ns-3)

- Write a single flow ns-3 example program with different-
 - TCP Extensions
 - Advertised window sizes
 - Queue limits on the bottleneck link
 - TCP Segment sizesPlot graphs using gnuplot
- Convert the first program to use multiple flows and compare the results. Does the use of multiple flows increase network throughput? If yes, how or why?
- Write an ns-3 example program to illustrate the impact of Window Scaling.
- Create a dumbbell topology with 5 flows (3 TCP and 2 UDP). User should be able to interactively change TCP variant, TCP Window Scaling (On/Off), queue limits on the bottleneck link.

The veth devices are virtual Ethernet devices. They can act as tunnels between network namespaces to create a bridge to a physical network device in another namespace, but can also be used as standalone network devices.

Week 5 (Network Namespaces)

Reading Material:

- What are containers?
- What is the difference between a container and a virtual machine?
- What are Linux namespaces?
- What is the difference between a Linux namespace and a container?
- What is a network namespace in Linux?

Hands-on:

- Install, Setup and run an LXC container on your system.
- Manage named network namespaces using iproute2:

- Create a network namespace.
- Run a network application inside the network namespace.
- Move the physical NIC interface inside the network namespace.
- Destroy an existing network namespace after killing the processes running inside it.

Week 6 (Virtual Networks)

Reading Material:

- What is the difference between a hub, a switch and a router?
- What is a Linux bridge?
- What is Open vSwitch?
- What is a veth interface?
- What is a route?
- What is a routing table?

Hands-on:

- Ping a network namespace from the default namespace.
- Ping a network namespace from another network namespace.
- Access the Internet from inside a network namespace (without moving any physical NIC interface inside it) by using:
 - A Linux bridge
 - Open vSwitch
- Ping a network namespace from another network namespace via a Linux bridge.
- Ping a network namespace from another network namespace via a third network namespace.
- Create a network of 3 network namespaces in a star topology (with a Linux bridge at the centre).
- Create a network of 4 network namespaces in a star topology (with one of the network namespaces at the centre).
- Create a network by connecting 3 Linux bridges (each connected to a different network namespace) in a linear fashion (1--2--3).
- Create a network by connecting 3 Linux bridges (each connected to a different network namespace) in a circular fashion (1--2--3--1).
- In the above topology replace the Linux bridges with network namespaces.

Week 7 (Linux Traffic Control)

Reading Material:

- What is traffic shaping?
- What is traffic policing and how is it different from traffic shaping?
- What is a qdisc in Linux?
- What is a Token Bucket Filter (TBF)?

Hands-on:

- Connect two named network namespaces:
 - Change the bandwidth between them using the TBF qdisc.
 - Change the bandwidth between them using the netem qdisc.
 - Change the delay between them using the netem qdisc.
- Connect two network namespaces via a third network namespace (Call them Sender, Receiver and Router). Run 10 TCP streams from the sender to the receiver and measure the ping.
 - Scenario 1: Keep the delay and bandwidth on every interface the same.
 - Scenario 2: Change only the bandwidth (both ways) of the link between the Receiver and Router.
 - Scenario 3: Same as scenario 2 except, in the Router, double the delay on the interface connected to the Sender, and add an AQM qdisc (RED, PIE, or CoDel) on the interface connected to the Receiver.

Week 8 (Flent)

Reading Material:

- What is netperf?
- What is flent?

Hands-on:

- Install flent on your system.
- Create a dumbbell topology using network namespaces.
- Run the following tests using flent for different AQMs (RED, PIE, CoDel, fq_codel) at the bottleneck interface.
 - tcp_1up
 - tcp_2up
 - tcp_4up
 - tcp_8up

- rrul

Week 9 (DPDK Concepts)

Reading Material:

- What is fast packet processing?
- What is emulation? How is it different from simulation? How does ns-3 support emulation?
- What is NFV?
- Compare Virtual Machines vs Container
- What is paravirtualization?
- What are hugepages? Setup 256 hugepages of 2Mb size.
- Which one is the default Ethernet driver provided by Linux Kernel and dpdk?
- What is NUMA architecture? How is it different from non-NUMA architecture?
- What is zero copy? Why DPDK is called zero copy? Compare between 3 copy and zero copy.
- What is mbuf?
- What is memcpy? How DPDK doesn't use memcpy?
- What is EAL in DPDK?
- Briefly explain ethtool.
- What is promiscuous mode? When is it used? How is it used?

Hands-on:

- Check your NIC capability with DPDK.
- What are the bandwidth limit your NIC support or upto what bandwidth your NIC can operate?
- Play with ethtool and show different options related to it

Week 10 (DPDK installation)

Reading Material:

- Sign up for the course and go through the course
<https://builders.intel.com/university/networkbuilders>

Hands-on:

- Download dpdk from dpdk website, compile and install it.
- Run the DPDK HelloWorld example and explain the output briefly.

Week 11 (l2fwd)

Reading Material:

- What is l2fwd?
- What is the traffic/packet generator? Get familiar with few traffic generators (like MoonGen packet generator), find out more open source traffic generator.

Hands-on:

- Run the l2fwd program in dpdk.
- Try running it on the same PC with the help of you went through the intel network builders course.
- Use a traffic generator to generate the output of the l2fwd.