
Advanced Computer Networks

Project 1: Implementation of A IP Router

Assigned: April 5, 2013

Due: May 10, 2013

1 Overview

In this assignment, you will implement a software IP router, using the UCLA Click Modular Router. The module in Click is well-defined and you should study the tutorial and write a IP router with the specific language. You are required to test your software router and try to optimize its performance.

2 Project Outline

During the course of this project, you will do the following:

- Implement a software IP router using Click Modular Router.
- Implement a simple routing protocol.

3 Project specification

3.1 Scenario

This project is based on a proven software router solution. Using the given elements of Click, one can turn its PC/laptop into a router. The major purpose of this project is to give you an experience of investigating the inside of a real router, although our task is limited to a simple IP router. You are required to study the mechanism of the cooperation of different elements. Besides, you have to learn the well-define “CLICK” language, which is used to write a script to “define” your router. Since Click itself is opensourced and written in C++, you can modify its elements or even create your own elements!

The scenario of which your router works is a simple network, consisting of two routers and two hosts behind them, see Figure 1. R1 and R2 are capable of forwarding IP packets. As a simple case, they can be configured with static routing, i.e., the routing is set with configuration file and remains the same unless the configuration is changed. In this case, R1 forwards the packet form H1 to R2, and R2 forwards the packets from R1 to H2. In this way, the single-direction communication form H1 to H2 is established. Similarly, the reverse path

can be set up by static configuration, too.

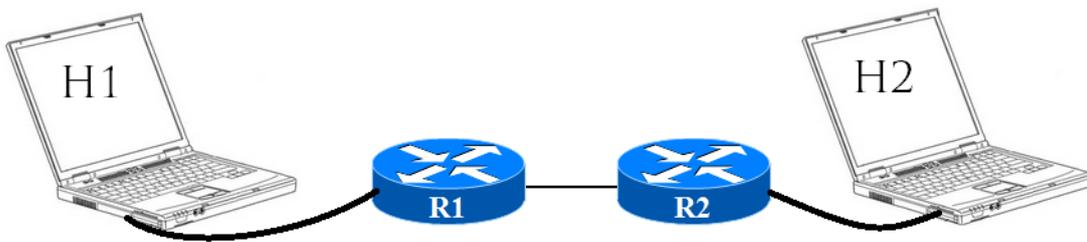


Figure 1 : the simple network scenario for evaluation

3.2 Programming Guidelines

Since your software router is built up with Click module, you should read the tutorial of Click carefully, and be familiar with the elements, which are relevant to your design. Click is available only for linux, so you should install linux on your machine. Besides, since you need at least four machines to set up the topology, we recommend that you install a virtual machine application on your PC/laptop, then use the virtual machines to be the hosts, and your physic machine act as the router. In this case, only two PC/laptops are required.

3.3 Project Tasks

There several tasks to evaluate your application's performance:

3.3.1 Task 1 – Reachability and Throughput

The first task is to implement a router with static routing, using Click Modular Router. As explained above, the two router bridge the two hosts, so that the two host can communicate with each other. We will examine two major aspects of this system: the reachability and throughput. First, to test the reachability, one can use the ping command in the terminal of one of the hosts. If this host receives reply from the other host normally, that means both the basic function of the Click router and the static routing configuration are correct. Second, to test the actual performance of the Click router, one can use the Iperf tool to test the maximum throughput of this network. You are required to optimize your design of router and try to speed up the packet forwarding rate.

3.3.2 Task 2 – Routing Protocol:

Based on the static routing case, you should turn your router into a real one by add the routing-relevant elements. Since only small network is required, you can use some light-weighted protocol, like RIP, or even write your own element with the offered API. An acceptable routing protocol should be like this:

- Neighbor discovery: a router should be aware of its waken neighbor, this neighbor may be another router or a host.
- Routing information spread: a router should spread its own routing information and learn the routing information from its neighbor to construct a full routing table.
- Failure detection: one should be aware of its neighbor’s crash and change the routing table accordingly.

4 Grading

This information is subject to change, but will give you a high-level view of how points will be allocated when grading this assignment. Notice that many of the points are for basic reachable functionality. Make sure these work well before moving to more advanced functionality or worrying about corner-cases.

- **Basic reachability [50 points]:** The end-host should be reachable to each other correctly. We will examine this using ping.
- **Throughput [20 points]:** Your design of Click router should be effective in processing the incoming packets, that is, forwarding it to the correct port as soon as possible.
- **Routing [up to 40 points]:** You should implement a simple routing protocol that will help your router to discover its neighbors and learn the routing information. The routing table is calculated dynamically.

5 Hand-in

Code submission for the final deadline will be done through your subversion repositories. You will receive an email with your Team#, Person#, and associated password soon after the assignment is posted. You can check out your subversion repository with the following command where you must change your Team# to “Team1” for instance, and your P# to the correct number such as “P1”:

The grader will check directories in your repository for grading, which can be created with an “svn copy”:

- Final Handin – YOUR REPOSITORY/tags/final
- Contest Handin (optional) – YOUR REPOSITORY/tags/contest
- Extra Credit (optional) – YOUR REPOSITORY/tags/extracredit