CSC/ECE 773 Advanced Internet Protocols
EB2 1021, TTH 3:50 – 5:05pm

Instructor: Injong Rhee, Professor in Computer Science, North Carolina State University, Office: EB II 3268, Phone: 515-3305, Email: rhee@ncsu.edu

Short Bio
Injong Rhee is Professor of Computer Science at North Carolina State University and runs Networking Research Lab (NRL). He works primarily on network protocols for the Internet. His major contributions in the field include the development of congestion control protocols, called BIC and CUBIC. Since 2004, these protocols have been the default TCP algorithms for Linux and are currently being used by more than 40% of Internet servers around the world and by several tens millions Linux users for daily Internet communication. He also has invented several multimedia streaming and multicast technologies licensed to companies for commercial applications. He started a company based on these technologies in 2000 where he developed and launched the world’s first video streaming products and push-to-talk (PTT) VoIP products for cell phones. His recent research topics include mobile ad hoc networks, delay/disruption tolerant networks, and P2P systems. He has been consulting for companies including Boeing, Lucent Technologies, CISCO, Korea Telecom, LG Electronics, and LG Datacom. He received NSF Career Award in 1999 and NCSU New Inventor's award in 2000.

Office Hours: Every Monday 10am – 11am.
Office hours are kept only if there is a request from students to meet during the office hours. Students must send email indicating the intention to come to the office hour at least 24 hours prior to the office hour. You do not need to receive a confirmation on the email request to come to the office hour (though I typically confirm it). Students are also encouraged to meet me outside the classroom or office hours. For those ad hoc meetings, students need to receive a confirmation on the meeting time from me.

TA: Sungro Yoon (syoon4@ncsu.edu), Office Hour: TBA, Office: EBII 2240

Textbook: None. Most reading materials are research papers from conference proceedings and journals that can be downloaded from the Internet.

Synopsis
In CSC/ECE 573, we learned about various topics related to Internet protocols including Internet architecture, applications, TCP/IP, and routing in general. As a sequel to CSC/ECE573 Internet Protocols, CSC/ECE773 provides students with opportunities for learning and practicing the state-of-the art research work in the areas of Computer Networking. To prepare students for research work, we will first discuss a few HOW-TOs on conducting research, writing papers and presenting research work. Then we will review and discuss recent research papers presented in networking conferences and journals. Through this review, students are exposed to current hot research problems and topics in Computer Networking. This process eventually provides guidelines and examples for students to conduct their own research projects.

In the beginning of the course, students will select research topics and define research problems for their projects, and spend a majority of the class time on discussing and presenting the progress of their projects through face-to-face meetings with the teaching staff. The project is expected to be a team project. I will provide several suggestions for research projects and also encourage students to propose their own. Based on mutually agreed projects, individual research teams and the teaching staff will meet regularly (either outside or during the class time) to check the project progress. In the end of the course, students are responsible for writing a term paper on their research projects and demonstrating and
delivering their research products. The grading will be strictly based on student presentations, paper reviews and research projects.

Paper presentation:
Throughout the course instructor and students will work together on selecting research papers to review. Each student will be assigned to 4-6 research papers to read and will briefly (15-20 min/paper) present and introduce their assigned papers to the class. The presentation will focus on defining the problems solved, their motivations and how the authors approach to solve the problems. The presenter is expected to answer questions from the audience. Each presentation is evaluated by the class and the evaluation score is counted toward the final grade. All the members in the same team will receive the same grade.

Research Projects:
I will provide a list of suggested research projects. Most of these projects involve some system design, implementation, evaluation and writing. By the first two weeks of the course, students are expected to form teams (2-4 members depending on the extent of the projects) and select or propose a research project. The project will be very closely monitored by the instructor and TA as each team is expected to meet a teaching staff once every one or two weeks to submit a progress report and discuss the progress and next action items. The project will be conducted in stages: requirement definition, literature reviews, designs, prototypes, implementation, evaluation and demonstration. The following deliverables are required:

1. Project requirement document: it must describe the definition of the problem being solved, expected outcome, testing and evaluation methods and final deliverables for the project. This document is an introduction of your final term paper.

2. Weekly progress report: independent of whether having a research meeting, described below, or not, each team needs to summarize in document the project activities of the past week and submit it to the TA.

3. Research survey: it must describe in an organized manner any existing and past work related to your project. This document is the related work section of your paper.

4. Design document: it must describe the solutions and algorithms you are proposing and its analysis. This will be the body of your term paper.

5. Software prototype: it should provide some minimal functionality of your software system (if applicable) that you implement as part of the project. The minimal functionality may include user interface and functional operations of your software (may be not necessarily the most efficient version of it). You have to write an implementation document discussing lessons learned from the implementation and any non-trivial design choices you have made in your implementation. This document is the implementation section of your paper.

6. Evaluation: it must describe the performance evaluation setup and methodology, and its result. A bunch of graphs and their descriptions go here. This is the evaluation section of your paper.

7. Conclusion: it summarizes the work and provides the conclusion learned from the project. This is your conclusion section.

Weekly Research Meetings:
Each research team will have a bi-weekly meeting with the teaching staff (me or the TA) to discuss the progress of its project. Some extra meetings can be arranged by the request of students or the instructor. In this meeting, we will check the current due milestones and decide on the milestone for the next week and action items for the team. The meeting lasts about 20 to 30 minutes. Because of the time constraint,
each team will meet the teaching staff probably once in every two weeks. The meeting needs to be
attended by all the members and the instructor may check the progress of each member. Some
deliverables are expected in each meeting that the team needs to provide, such as documents, software,
and demonstrations. Weekly progress reports must be turned every week independent of whether you
have a research meeting or not.

Grading:
Quiz: 10%
Presentation: 30%
Reviews: 5%
Project: 50%
Class participation: 5% (the TA will assign this participation grades to students who are asking questions
during the paper presentation in each class period).

There is no exam in this course.

Suggested Projects

High Performance FTP Server and Client
Design and implement a file transfer server and client for Windows (XP or Vista) using UDP. The FTP
system can still use TCP for control, but for data transfer, you will need to develop your own packet
recovery protocol and congestion control protocol. It does not have to follow the standard IETF FTP
protocol. The keys for this project are efficiency (speed), ease of use, and security. Since existing FTP
systems use TCP, their performance is not more than what TCP can offer. For instance, it does not work
well in wireless network environments and it frequently does not use all the full capacity of network
bandwidth (because of window scaling).

Your transport protocol must provide high throughput in diverse network conditions and must also be fair
at least to its own flows (both throughput-fair and RTT-fair). The protocol must utilize the full capacity of
the network. A desirable feature is that if there are any other existing TCP flows over a path, it uses only
the leftover bandwidth. The transport protocol must provide a well-defined and easy-to-use socket
interface so that many network applications can be built using the protocol. In this project, you will be
developing an FTP application that uses this protocol as a demonstration of the performance of your
protocol.

The client must have a GUI with convenient drag and drop options. It needs to support all the
functionalities of standard FTP clients. Additionally, it needs to support a transfer of directories without
requiring users for tarring and compressing the directory.

The FTP system must be secure. It requires a login process and control must be supported on top of SSL
(the login process and control signals can still be supported using TCP, but your data transfer must be on
UDP). It also needs to be functional over NAT.

Language of implementation: Java
Recommended group size: 3.
QoS for SCTP
Refer to the attached document for this project.
Recommended group size: 3

TCP Performance Fine-tuning for 10GE NIC
10GEs are regularly used for data centers and storage servers. But the current TCP implementation in Linux cannot fully utilize the 10Gbps capacity (typically achieving only 2-3 Gbps). This has been a motivation for storage servers to use an off-loading TCP NIC card which supports TCP operation in hardware. This project involves profiling the TCP stack and identifying the system bottleneck for the low performance of TCP. You will need to design solutions for removing the bottleneck, if possible, to achieve near 10Gbps performance in software. We will provide you with two 10 GE NIC cards and an access to a server for this project.
Recommended group size: 2

A Measurement Study of End Point Redundancy (Sankar)
The Internet was built for communication and is now used predominantly for content distribution. How can we redesign the Internet targeting content distribution? One of the key elements of redesign is to eliminate redundancy in content distribution and harness caches present locally (could be either end points or routers). The goal of this project is to identify the potential for eliminating redundancy at end points. To understand the effectiveness of end point techniques in eliminating redundancy in content distribution, conduct a measurement study of packet content from end hosts (e.g., laptops). Capture packets (with content) at network level and identify opportunities for collaboration within hosts and across hosts. Also, examine long term patterns in redundancy.
Related work:

Efficient P2P Network Delay and Loss Measurement System (Alphonse)
Overlay Networks such as RON [1] are used for detour routing to overcome Layer 3 failures efficiently to avoid disruptions in communications (e.g., TCP flows). The design of the existing overlay networks requires a full mesh \((N \times N)\) probing to calculate the (1) loss rate and (2) round trip time (RTT). As the number of nodes in the overlay increases, the overheads increase in the order of \(N^2\). Instead, small sized overlays are being used to help in reducing these overheads and this choice directly affects the available alternative path choices. This project requires you to design and implement an overlay network that could scale with a large number of nodes while meeting the following requirements

(1) Network & Calculation overhead should be low compared to RON [1].
(2) The design technique should involve the use of clusters formed by CDN [ 2] (Akamai) redirection.

An obvious choice of design for the overlay is use of hierarchy. The approach taken in [2] can be a good place to start.
References:
Non-intrusive TCP Congestion Control Identification (Sangtae)

Internet servers get started using enhanced TCP congestion control algorithms (We call “TCP Variants”) to improve their TCP performance in the Internet. For instance, Windows Vista/7 turn on Compound TCP as their default congestion control algorithm while Linux runs BIC/CUBIC as its default TCP algorithm since 2004. The goal of this project is to identify the TCP congestion control algorithms used by Internet servers today. This project involves setting up a small testbed, where you can control the bandwidth and delay to the traffic (using Nettem in the Linux kernel), and implementing a non-intrusive probe that can identify the congestion control algorithm of the server with high accuracy. You will need to design an identification algorithm, by examining the packets sent by the remote server, in response to your ‘forged’ request packets. Your identification algorithm can use any methodology, including, but not limited to, the slope of congestion control window growth, passive/active measurements and the use of decision branches. At the end of this project, you must show the accuracy of your identification algorithm on the testbed as well as on the Internet.

Recommended group size: 3

Mobile Phone Location Service Using Compass, Accelerometer, Google Maps and GPS

Smart phones are equipped with GPS. They rely on GPS to provide location information for the users. But GPS is highly battery-power hungry; with GPS on, phones typically do not last longer than two hours. This becomes a road block to applications that require frequent location information. This project is to design and implement a system that can still provide GPS-like location information, but with only minimal use of GPS. Smart phones typically come with accelerometer, electronic compass and access to the Internet. By using these sensors, we can track the movement of the phone user – direction and speed. Using Google maps downloaded from the Internet, the sensor information can be applied to track the location of the user. With intermittent GPS accesses, the location information can provide usefully accurate location information. This team needs to work with an Android phone (provided) and needs to develop a power-efficient and accurate location system.

Recommended group size: 2

Improving the performance of wireless AP networks (Sungro)

Design and implementation of P2P push and pull protocol for video content distribution.

Push: Multicast is an important tool for content distribution to a large base of users. But IP multicast requires a lot of support from the network to be successful. An alternative is to use a peer to peer overlay network for content distribution. You have N peers who are interested in a file download. Then construct a source-based efficient multicast overlay tree using the peers. You may use TCP for actual data transfer between two peers for the initial implementation. Your multicast protocol must handle any unexpected disruption of transfer caused by premature leave or failure of peer nodes. The main application for this protocol is streaming of live video content, such as sport events, to a large set of users.

Pull: In a P2P BitTorrent system, when a peer downloads a file, it gets its file segments in arbitrary order. Thus, it is not suitable for real-time video streaming. Your job is to modify a BitTorrent client to enable video streaming. Since each peer has a limited amount of upload bandwidth, to match the bit rate suitable for streaming, you need to download from many peers at the same time. Furthermore, as for streaming,
you will need to receive the video file in sequence, streaming requires data striping among multiple uploading peers.

For both protocols, you will need to implement your protocol in an open source BitTorrent client, called VOZE, and demonstrate the performance of your protocol in the Planet lab using 100s nodes. Later on, when the FTP team has their protocol implementation, you have to replace your TCP calls to the fast UDP transport calls developed by the FTP team for improved performance.

Programming Language: Java

Recommended group size: 4

**Tentative Schedule**

**January**
12: Course introduction/syllabus and overview of projects
14: Short review paper assignment finished by this time. How to do research.
19: How to write and present a research paper.
21: Short review
26: Short reviews
28: Short reviews. Research project assignment and team formation

**February**
2: Short reviews
4: Short reviews
9: Short reviews
11: Research meeting. Project requirement due.
16: Research meeting
18: Long review
23: Long review
25: Research meeting. Research survey due.

**March**
2: Long review
4: Research meeting
9: Long review
11: Research meeting. Design document due.
16: Spring break
18: Spring break
23: Long review
25: Research meeting.
30: Long review

April
1: Spring holiday
6: Long review
8: Research meeting. Implementation document due.
13: Long review
15: Research meeting
20: Long review
22: Research meeting. Evaluation and conclusion due.
29: Final term paper due.