Université d'Ottawa
Faculté de génie

École d'ingénierie electrique
et de science informatique

uOttawa
L'Université canadienne
Canada's university

University of Ottawa
Faculty of Engineering

School of Electrical Engineering
And Computer Science

ELG5369

Internetworking Technologies

PROJECT MANUAL

Professor: **Dan Ionescu**

Fall 2017

1. Introduction

This document contains all the information you need in order to start working on the ELG5369 term project.

The most recent version of this document can always be retrieved from the ELG5369 course website, located at:

https://course.ncct.uottawa.ca/course/view.php?id=21

If you have any questions related to the project, you should ask them in the \Project Questions" forum of the ELG5369 course website.

2. Project Description

The term project is a group effort. Each group of three or four students (as previously established) will choose one of the available projects. All projects are research-oriented.

The research projects will require you to gather, study, and analyze documentation on the project topic. The documentation can come in the form of white papers, journal and/or conference papers, as well as vendor-provided resources. It is highly recommended to study both vendor-independent and vendor-provided resources. The one restriction about the documentation is that it needs to be publicly available.

3. Requirements

This section provides details about what results you will be expected to submit and when you will have to submit them.

3.1. Project Methodology

The following items are the deliverables for each of the projects:

- Technology Review. The scenario will lock you into the technologies you must use. You will have to research these technologies and write a document of no more than ten pages describing the main functionality points of the reviewed technologies. The review should contain enough detail to make it meaningful in the context of the given scenario.
- Analysis of the Existing Design. You will have to look at the existing network and identify the key incompatibilities between the design and the given set of your requirements. You will have to write these incompatibilities in the form of a short document outlining where and why the existing approach fails to meet your requirements.
- Proposed Re-design. Based on your analysis of the current design technologies, you should write a design document explaining exactly how the network will be re-engineered and how your proposed solution addresses the requirements.

3.2. Project Deliverables

The following items are the deliverables for your research projects:

1. **A Project Proposal:**

   The proposal has to contain the following items:

   1. **A brief description of the Project Theme** in which you have to describe in your own words the project subject together with a motivation for approaching it.
   2. **Approach:** describe how would you approach the subject and how would you develop it. A short literature review accompanied by your own vision on how to provide a solution for it is a must in this Project Proposal.
   3. **A diagram,** either block or another type

2. **Project Report**

   1. Executive Summary. This is a document of maximum one page that describes the topic and the underlying technologies from a high-level point of view: concept definition; what challenges or limitations created a need for this concept and the appropriate set of supporting technologies? The goal of this portion of the project is to be able to understand and explain what it is that the topic is about.

   2. In-depth Technology Overview. You will have to describe in depth the studied topic, including any supporting technologies (algorithms, protocols, etc.) The goal of this step of the project is to be able to explain how the particular technologies that comprise the topic work, together and separately.

   3. **Challenges and Future Work.** As it is the case with every emerging concept or technology, there are challenges to be overcome and steps to be completed before it becomes “mainstream." You should write a document that answers questions such as: what is the current state of the technologies (definition and standardization of algorithms and protocols)? Are there any real implementations, if applicable? What are the most important challenges the technologies need to overcome in order to become viable? The goal of this part of the project is to define where we are now and where we are going as far as the studied topic is concerned.

   4. Presentation. Before the end of the semester, you will have to present in class your work on the term project, in enough slides to include all the details needed for people to be able to understand your work, but not to exceed twenty minutes of presentation.

5. Final Project Report. At the end of the semester, you will have to submit all the information you have gathered while working on the project (including the previously mentioned documents, amended as needed) in the form of a properly structured project report.

Note that you do not need to submit items 1 to 3 separately. They are part of the final project report. However, the report will have to clearly identify each of the three sections.

4. Presentation Slides :

You will have to submit the slides you used for the oral presentation of your project.

5. Important Dates

Below are the important dates and deadlines for different phases of the term project.

- Selection of a project topic: September 15, 2017
- Oral presentation (in class): November 24, and 28
- Final Exam December 1, 2017
- December 15: Submission of the final project report

4. Project Topics:

The available projects are described throughout the remainder of this document.

1. Congestion Control in IP Networks:

In data networking and queueing theory, network congestion occurs when a link or node is carrying so much data that its quality of service deteriorates. Typical effects include queueing delay, packet loss or the blocking of new connections. A consequence of the latter two effects is that an incremental increase in offered load leads either only to a small increase in network throughput, or to an actual reduction in network throughput.

Network protocols which use aggressive retransmissions to compensate for packet loss tend to keep systems in a state of network congestion, even after the initial load has been reduced to a level which would not normally have induced network congestion. Thus, networks using these protocols can exhibit two stable states under the same level of load. The stable state with low throughput is known as congestive collapse.

Modern networks use congestion control and congestion avoidance techniques to try to avoid congestion collapse. These include: exponential backoff in protocols such as 802.11 CSMA/CA and the original Ethernet, window reduction in TCP, and fair queueing in devices such as routers.

Another method to avoid the negative effects of network congestion is implementing priority schemes, so that some packets are transmitted with higher priority than others. Priority schemes do not solve network congestion by themselves, but they help to alleviate the effects of congestion for some services. An example of this is 802.1p.

A third method to avoid network congestion is the explicit allocation of network resources to specific flows. One example of this is the use of Contention-Free Transmission Opportunities (CFTXOPs) in the ITU-T G.hn standard, which provides high-speed (up to 1 Gbit/s) local area networking over existing home wires (power lines, phone lines and coaxial cables).

The project relates to a synoptic review of various aspects of congestion control in various types of networks, and of remedies of the congestion in each of the above

networks. An example is required to be provided in order to illustrate the theory and its applications.

2. Congestion Control in Wireless Networks

About 90% of connections on the internet use TCP to communicate. Through several upgrades and improvements, such as TCP Tahoe, TCP Reno, XCP, and others, TCP became well optimized for the very reliable wired networks. As a result, TCP considers all packet timeouts in wired networks as due to network congestion and not to bit errors. However, with networking becoming more heterogeneous, providing wired as well as wireless topologies, TCP suffers from performance degradation over error-prone wireless links as it has no mechanism to differentiate error losses from congestion losses. It therefore considers all packet losses as due to congestion and consequently reduces the burst of packet, diminishing at the same time the network throughput.

Algorithmically, when a timeout is detected, the TCP congestion control algorithm reduces dramatically the packet burst to diminish the network load and relieve the congestion. In effect, the TCP congestion algorithm cannot differentiate the loss caused by congestion from the one caused by error. Once a loss occurs, TCP deals with that as a congestion event and halves down the size of its congestion window. This unnecessary slowdown reaction decreases the throughput of TCP and reduces the overall speed of the network.

In practice, the aforementioned behavior of the TCP congestion algorithm is acceptable in wired networks as packet timeouts are most of the time caused by congestion. However, this is totally inappropriate in wireless networks as wireless links are known to experience a lot error bits and packet loss due to fading, interference, hand-off, and other radio effects.

Consequently, packet loss in wireless networks cannot be considered as due to congestion. As a result, the TCP often makes the wrong decision by slowing down the burst of packets while it should instead retransmit lost packets.

This problem is popularly known as the TCP performance problem over wireless network and has been researched and studied by many researchers for many years now . The key in solving this problem is to allow TCP to differentiate between timeouts caused by congestion and those caused by errors and noise in the wireless channel.

This project reviews the proposals for TCP congestion control schemes wich are appropriate for wireless as well as wired networks and is capable of distinguishing congestion losses from error losses.

3. Active Queue Management

Active queue management (AQM) is the intelligent drop of network packets inside a buffer associated with a network interface controller (NIC), when that buffer becomes full or gets close to becoming full, often with the larger goal of reducing network congestion. This task is performed by the network scheduler, which for this purpose uses various algorithms such as random early detection (RED), Explicit Congestion Notification (ECN), or controlled delay (CoDel). RFC 7567 recommends active queue management as a best practice.

An Internet router typically maintains a set of queues, one per interface, that hold packets scheduled to go out on that interface. Historically, such queues use a drop-tail discipline: a packet is put onto the queue if the queue is shorter than its maximum size (measured in packets or in bytes), and dropped otherwise.

Active queue disciplines drop or mark packets before the queue is full. Typically, they operate by maintaining one or more drop/mark probabilities, and probabilistically dropping or marking packets even when the queue is short.

Drop-tail queues have a tendency to penalize bursty flows, and to cause global synchronization between flows. By dropping packets probabilistically, AQM disciplines typically avoid both of these issues.

By providing endpoints with congestion indication before the queue is full, AQM disciplines are able to maintain a shorter queue length than drop-tail queues, which combats buffer bloat and reduces network latency.

Early AQM disciplines (notably RED and SRED) require careful tuning of their parameters in order to provide good performance. Modern AQM disciplines (ARED, Blue, PI) are self-tuning, and can be run with their default parameters in most circumstances.

This project relates to the review of all AQM algorithms presenting all pros and cons for adopting the best one for being implemented in either access or core networks.

4. Active Queue Management for Differentiated Services:

Emerging new service types, such as real-time audio/video applications require QoS support in terms of Internet Service Layer Agreement.

Differentiated services (DiffServ) networks provide coarse QoS control for Internet applications only. Instead of providing a guarantee to individual flows, which may cause scalability problems, DiffSer provides statistical QoS to a few predefined service classes over a long time scale. In a DiffServ network, a service subscriber first sets up a service profile with an Internet service provider (ISP) regarding the desired type of service. At the ingress of the DiffServ network, edge routers classify all packets into several predefined service classes by inspecting them, and mark the packets with different drop precedences (such as in/out packets) according to the subscriber's service profile . The traffic that conforms to the service profile arc marked with low drop priority and will receive better service, while the packets marked as the nonconforming part of the traffic is marked with high drop priority and receive a best effort service.

The token bucket marker is one of the most commonly used markers in edge routers.

The service differentiation of packets is provided by core routers, by using an active queue management scheme, such as RED with in and out (RIO) according to the preassigned service classes and drop precedences carried in the packet header.

The problem of this scheme of DiffServ networks is that there are two kinds of routers: edge routers and core routers for the service subscriber, and the remote control of services is limited to the domain of the DiffServ network, and they cannot be extended to common Internet network.

As such, an AQM congestion control tries to utilize all available bandwidth in a fair and efficient way, while service subscriber has served its purpose exceptionally well and is partly responsible for the communication explosion of the last decade.

This project relates to the analysis of all AQM protocols which are used in Diffserv and to propose the best one, justifying the solution.

5. Software-Defined Networking

Software controlled networking, a subject, methods and technologies were at the attention of network designers, equipment producers, and providers since the time of ATM (Asynchronous Transfer Mode switches) were applied on a large scale.

The Software platform was used to set dynamic Channels and Paths, dynamically and in real-time. In a nutshell the software contained all the needed routines needed to automate the travel of cells and thus of the packets from one end to the other. In this way a separation between the control commands and the data has been realized.

The method of separating the control from the forwarding part of the packets/routing emerged again, especially since virtualizing computer resources was extended to the networking devices such as switches and routers. Under SDN, the control plane is implemented in software in servers separated from the network equipment and the data plane is implemented in commodity network equipment.

The basic approach to achieve that is by applying globally aware and topology decoupled software control at the edges of the network. The assumption is that traditional topology-coupled bridging & routing drives the core of the network so that scalability, interoperability, high-availability, and extensibility of IP networks can be maintained. Using the analogy of a postal service, the way SDN controllers would work is: for any given street location, all the letters from all the tenants would first be aggregated by an SDN edge. This edge function would examine the current location for each of the letter-destinations using a global non-autonomous lookup mechanism.

Based on that global lookup and on other globally defined and globally measured considerations - such as access control or remote location load conditions the SDN edge places one or more of the original letters in an additional envelope addressed to each of the street locations where the destinations currently are. It then uses the normal postal service which works like traditional IP to get these outer envelopes to the remote locations. This is done based on the existing and scalable hop-by-hop forwarding country.state.zip.street postal service. The outer letters are then opened by the remote SDN edge and the original envelopes are delivered to the destinations.
This is a very important feature for the new paradigm of distributed computing called cloud computing.

6. MESH Networking and Internetworking

A mesh network is a network topology in which each node relays data for the network. All mesh nodes cooperate in the distribution of data in the network. Mesh networks are typically wireless. Over the past decade, the size, cost, and power requirements of radios has declined, enabling multiple radios to be contained within a single mesh node, thus allowing for greater modularity; each can handle multiple frequency bands and support a

variety of functions as needed—such as client access, backhaul service, and scanning (required for high-speed handoff in mobile applications)—even customized sets of them.

Mesh nodes are built using small radio transmitters that function in the same way as a wireless router. Nodes use the common WiFi standards known as 802.11a, b and g to communicate wirelessly with users, and, more importantly, with each other.

In a wireless mesh network, only one node needs to be physically wired to a network connection like a DSL Internet modem. That one wired node then shares its Internet connection wirelessly with all other nodes in its vicinity. Those nodes then share the connection wirelessly with the nodes closest to them. The more nodes, the further the connection spreads, creating a wireless "cloud of connectivity" that can serve a small office or a city of millions.

Nodes are programmed with software that tells them how to interact within the larger network. Information travels across the network from point A to point B by hopping wirelessly from one mesh node to the next. The nodes automatically choose the quickest and safest path in a process known as dynamic routing.

Mesh networks can relay messages using either a flooding technique or a routing technique.

With routing, the message is propagated along a path by hopping from node to node until it reaches its destination.

To ensure all its paths' availability, the network must allow for continuous connections and must reconfigure itself around broken paths, using self-healing algorithms such as Shortest Path Bridging.

Self-healing allows a routing-based network to operate when a node breaks down or when a connection becomes unreliable. As a result, the network is typically quite reliable, as there is often more than one path between a source and a destination in the network. Although mostly used in wireless situations, this concept can also apply to wired networks and to software interaction.

A mesh network whose nodes are all connected to each other is a fully connected network. Fully connected wired networks have the advantages of security and reliability: problems in a cable affect only the two nodes attached to it. However, in such networks, the number of cables, and therefore the cost, goes up rapidly as the number of nodes increases.

Mesh networks can be considered a type of an ad-hoc network. Thus, mesh networks are closely related to mobile ad hoc networks (MANETs), although MANETs also must deal with problems introduced by the mobility of the nodes.

7. Network Functions Virtualization (NFV)

Network-Function Virtualization (NFV) is a network architecture concept that uses the technologies of IT virtualization to virtualize entire classes of network node functions into building blocks that may connect, or chain together, to create communication services.

NFV relies upon, but differs from, traditional server-virtualization techniques, such as those used in enterprise IT. A virtualized network function, or VNF, may consist of one or more virtual machines running different software and processes, on top of standard high-volume servers, switches and storage, or even cloud computing infrastructure, instead of having custom hardware appliances for each network function.

For example, a virtual session border controller could be deployed to protect a network without the typical cost and complexity of obtaining and installing physical units. Other examples of NFV include virtualized load balancers, firewalls, intrusion detection devices and WAN accelerators.

The rise of significant competition in communication services from fast-moving organizations operating at large scale on the public Internet (such as Google Talk, Skype, Netflix) has spurred service providers to look for ways to disrupt the status.

The NFV framework consists of three main components:

(a) Virtualized network functions (VNFs) are software implementations of network functions that can be deployed on a network function virtualization infrastructure (NFVI).

(b) Network function virtualization infrastructure (NFVI) is the totality of all hardware and software components that build the environment where VNFs are deployed. The NFV infrastructure can span several locations. The network providing connectivity between these locations is regarded as part of the NFV infrastructure.

(c) Network functions virtualization management and orchestration architectural framework (NFV-MANO Architectural Framework) is the collection of all functional blocks, data repositories used by these blocks, and reference points and interfaces through which these functional blocks exchange information for the purpose of managing and orchestrating NFVI and VNFs.

The building block for both the NFVI and the NFV-MANO is the NFV platform. In the NFVI role, it consists of both virtual and physical processing and storage resources, and virtualization software. In its NFV-MANO role it consists of VNF and NFVI managers and virtualization software operating on a hardware controller. The NFV platform implements carrier-grade features used to manage and monitor the platform components, recover from failures and provide effective security - all required for the public carrier network.

8. IoT (Internet of Things):

IoT refers to uniquely identifiable objects (things) and their virtual representations in an Internet-like structure. The term Internet of Things was first used by Kevin Ashton in 1999. The concept of the Internet of Things first became popular through the Auto-ID Center and related market analysts publications. Radio-frequency identification (RFID) is often seen as a prerequisite for the Internet of Things. If all objects and people in daily life were equipped with radio tags, they could be identified and inventoried by computers. However, unique identification of things may be achieved through other means such as barcodes or 2D-codes as well.

Equipping all objects in the world with minuscule identifying devices could be transformative of daily life. For instance, business may no longer run out of stock or generate waste products, as involved parties would know which products are required and consumed. One's ability to interact with objects could be altered remotely based on immediate or present needs, in accordance with existing end-user agreements.

The original idea of the Auto-ID Center is based on RFID-tags and unique identification through the Electronic Product Code.

An alternative view, from the world of the Semantic Web focuses instead on making all things (not just those electronic, smart, or RFID-enabled) addressable by the existing naming protocols, such as URI. The objects themselves do not converse, but they may now be referred to by other agents, such as powerful centralized servers acting for their human owners.

The next generation of Internet applications using Internet Protocol Version 6 (IPv6) would be able to communicate with devices attached to virtually all human-made objects because of the extremely large address space of the IPv6 protocol. This system would therefore be able to identify any kind of object.

A combination of these ideas can be found in the current GS1/EPCglobal EPC Information Services (EPCIS) specifications. This system is being used to identify objects in industries ranging from Aerospace to Fast Moving Consumer Products and Transportation Logistic

9. Sensor Networks: Routing Techniques for Sensor Networks

Sensor networks use a large number of ultra-small devices, known as sensor nodes, to form a network without the aid of any established infrastructure. In these networks, the individual nodes are capable of sensing their environments and either processing the information locally or sending it to one or more collection points through wireless links.

Through their ability to monitor their surroundings and provide detailed data, sensor networks have tremendous potential to benefit a broad range of sectors. They can be used for many things, including environmental areas such as monitoring for seismic activity and forest fires, creating industrial efficiencies, traffic control, security and military operations and even improved health care.

For obvious reasons, having strong network infrastructure is essential to sensor network technologies.

A SN is built of "nodes" – from a few to several hundreds or even thousands, where each node is connected to one (or sometimes several) sensors. Each such sensor network node has typically several parts: a radio transceiver with an internal antenna or connection to an external antenna, a microcontroller, an electronic circuit for interfacing with the sensors and an energy source, usually a battery or an embedded form of energy harvesting. A sensor node might vary in size from that of a shoebox down to the size of a grain of dust, although functioning "motes" of genuine microscopic dimensions have yet to be created. The cost of sensor nodes is similarly variable, ranging from a few to hundreds of dollars, depending on the complexity of the individual sensor nodes. Size and cost constraints on sensor nodes result in corresponding constraints on resources such as energy, memory, computational speed and communications bandwidth. The topology of the SNs can vary from a simple star network to an advanced multi-hop wireless mesh network. The propagation technique between the hops of the network can be routing or flooding.

To reduce communication costs some algorithms for SNs remove or reduce nodes redundant sensor information and avoid forwarding data that is of no use. As nodes can inspect the data they forward they can measure averages or directionality for example of readings from other nodes. For example, in sensing and monitoring applications, it is generally the case that neighbouring sensor nodes monitoring an environmental feature typically register similar values. This kind of data redundancy due to the spatial correlation between sensor observations inspires the techniques for in-network data aggregation and mining.

The decentralized nature of wireless ad hoc networks makes them suitable for a variety of applications where central nodes can't be relied on, and may improve the scalability of

wireless ad hoc networks compared to wireless managed networks, though theoretical[2] and practical[3] limits to the overall capacity of such networks have been identified.

Minimal configuration and quick deployment make ad hoc networks suitable for SNs and play a central role in emergency situations like natural disasters or military conflicts. The presence of dynamic and adaptive routing protocols enables ad hoc networks to be formed quickly.

10. Routing technologies in Vehicular Ad-Hoc Networking (VANET)

Vehicular Ad Hoc Networks (VANETs) are created by applying the principles of mobile ad hoc networks (MANETs) - the spontaneous creation of a wireless network for data exchange - to the domain of vehicles. They are a key component of intelligent transportation systems (ITS).

At the beginning VANETs were seen as a mere one-to-one application of MANET principles, they have since then developed into a field of research in their own right.

By 2015, the term VANET became mostly synonymous with the more generic term inter-vehicle communication (IVC), although the focus remains on the aspect of spontaneous networking, much less on the use of infrastructure like Road Side Units (RSUs) or cellular networks**.**

There has been growing research interests on vehicular ad hoc networks (VANETs) over the past years due to their ease of deployment and potential support for wide range of applications that can greatly enhance our everyday driving experience on the road.

Multi-hop message dissemination is expected to be the primary mode of communication among vehicles for many VANET applications. High reachability and low end-to-end delays are among the key requirements of any dissemination scheme, especially when handling safety-related messages, which are inherently critical in nature.

Furthermore, any dissemination scheme should ensure efficient utilization of the channel bandwidth, a resource that is often scarce in VANETs. However, guaranteeing good performance levels in reachability, end-to-end delay as well as bandwidth utilization is a challenging issue in message dissemination over VANETs, which has not so far been adequately addressed in the literature. In an effort to fill this gap, this research investigates the above-described performance aspects of message dissemination in VANETs, with a particular focus on improving message reachability over high node density networks.

This project relates to the proposal of a routing technology in the conditions of the short time interactions between cars and their networking device.