Computer Networks CSE 4344
Project 3
Simulation of TCP

Instructor: **Sajib Datta**
GTA: Saipriya Thirvakadu, Harmanjot Kaur

Spring 2018

"A wise man can learn more from a foolish question than a fool can learn from a wise answer" *Bruce Lee*

Objectives

- To understand the Transport Layer Protocol
- To simulate the working of TCP

The division of network routing functionality into software layers had enabled network engineers to write modules that enable faster development, easier testing and smoother interfaces. Communications go across layers serving each neighbor with necessary details in a predetermined message format. Transport layer support multiple protocols including TCP, UDP, SCTP, FCP, etc.

Among the layers Transport Layer enables end-to-end data transfer by delivering data from an application to its remote peer. With its unique design feature it can support multiple applications and multiple users simultaneously. The most-used transport layer protocol is the *Transmission Control Protocol (TCP)*, which provides connection-oriented reliable data delivery, duplicate data suppression, congestion control, and flow control.

With this assignment you will make up a three host system which are connected to each other using different links. Once the connections are established you have to communicate with each other and once the communication is over we will have to terminate the connection as well.

Due Date

May 1, 2018 (Tuesday) 11:59 PM[1]

[1] All Submissions should be completed through BlackBoard

Submission Guidelines

- This is a group project with a team size not exceeding 2 per team.
- Please pair up and pick up your wing-man by the end of this week, Friday.
- Please use the following Link to register your team before March 30, 2018 (Friday).
- Submit a single zipped file with the naming convention,

< missionTCP your Team id > .zip

- Your submission should have the following items to be considered for evaluation,
  (a) You submit the source code.
  (b) **Clear instructions** on how to execute your code on any given system.
  (c) a *readme* file, which clearly mentions the underlying assumptions and system requirements.
- Make sure you write your **names** and your **UTA IDs** in the beginning of the source code that you are submitting.
- Make sure that submissions of the zipped file is through *UTA BlackBoard*[2].
- There will be a reduction of 10 points for the first day and 5 pointers each for each subsequent day.
- You will have to register for a demo in first week of May 2017. Once you pick your teams, demo dates will be announced.

Section 1: Agent Ann & *missionTCP*

The story rolls you back to year 2000. Three secret service agents (Ann-111, Chan-001, Jan-100) are on a very sensitive mission to take over and destroy an enemy hideout. Given the sensitivity of the mission they have to establish a new communication channel amongst them. They wanted to make an ad-hoc

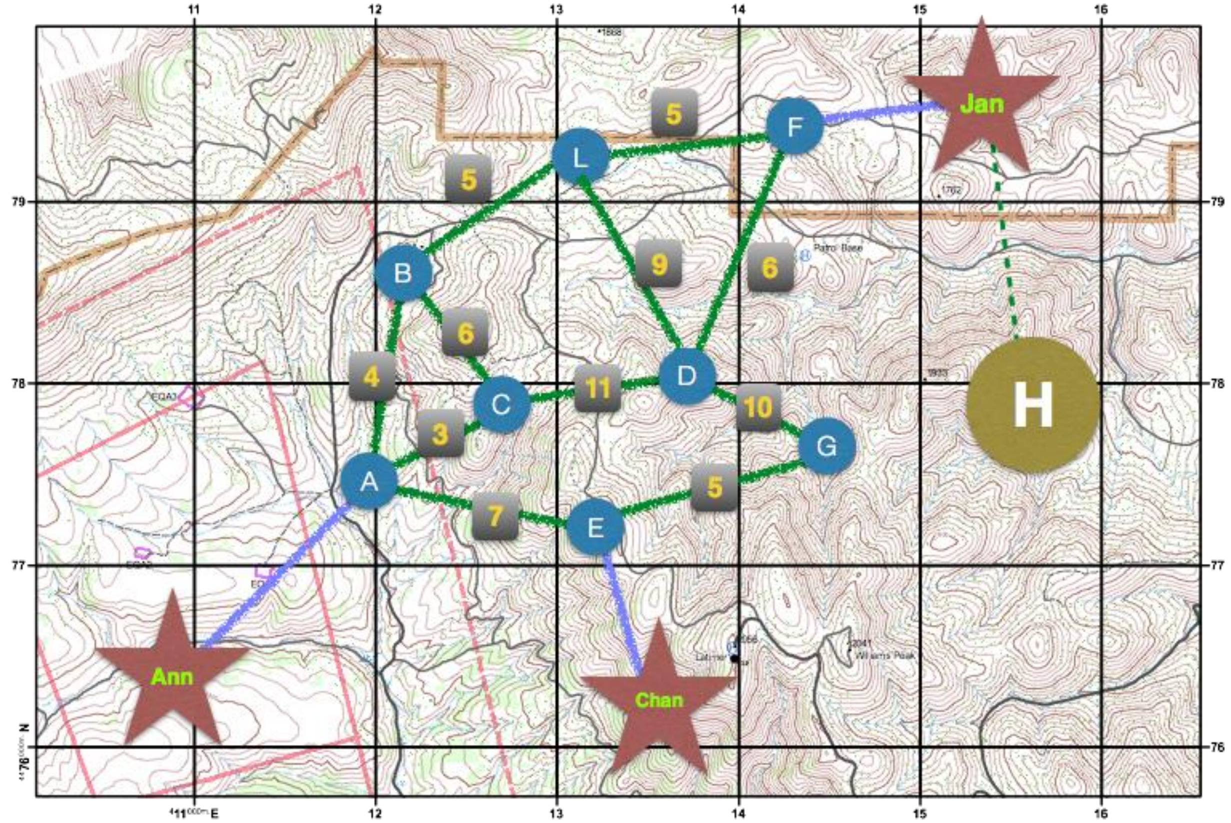


Figure1:Mapofthemission

TCP to make sure that no one else is able to snoop into the conversation. We call this modified TCP as *missionTCP* for this mission.

The mission is supposed to start at 0500Hrs and should end before 0700Hrs (Its real hard for someone to know about this attack and make a counter attack is the assumption). The agents are given two weeks time to come up with a communication plan and to execute the mission on August 20, 2000 0500Hrs. Agents have planned the logistics of the mission and had deployed **eight** routers in place (Mission Map is given in Figure 1) which runs your network algorithm. Now you have been recruited to develop the *missionTCP* for the agents. The specifications are as given below. The path lengths are given near the links between the routers. There are three agents and we should have separate individual connection between all the three agents. There is a Air-Force base *Given in H within a golden circle* to which agent Jan can connect privately to issue commands. For your visualization there is a *missionTCP* packet structure is given pictorially below in Figure 2.

- Your source ports # and destination port # has become the corresponding agent IDs. (111, 001, 100). *Watch out for the final length of the fields according to the packet specification given*.

- Sequence number is a random number that you generate for your communication. Once decided you will use this field to keep track of the message lengths sent for the variable part data based on the TCP principles. • A three way handshake must be performed to establish the sequence number and acknowledgement number.

- You should use your knowledge on *DRP, TER, URG, ACK, RST, SYN, FIN* to use it properly during the communication. Refer to textbook for details.

- You can use the receiver window to rate limit the other user in communication

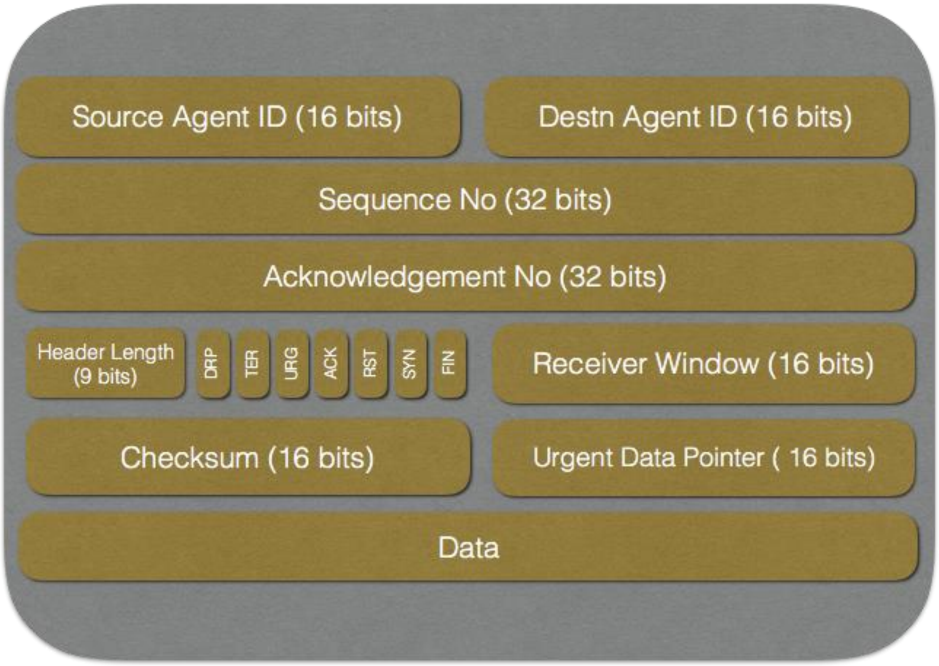
- checksum should be calculated as per the Internet checksum and should be calculated on first 16 Bytes.

- Urgent point could be used to point to anywhere when the *URG* is set.

Mission Objective #1: Mission On Station

"Everyone imposes his own system as far as his army can reach" Joseph Stalin

[2] Please strictly follow the naming convention of the zipped file



- *Use the map given to calculate the shortest path between the routers and establish a miniature routing table within each router.* You may have to do it for each pair as Dijkstra's algorithm calculates the shortest path between pairs of destinations.

- *Establish port based communication between the source and destination.* You remember one communication channel that you have established for Project 1 using original TCP port and socket. Use the same to establish end-to-end connection between the agents.

- *Each other agent should establish communication channel with agent Ann-111.* Challenge here is to have a simulated 3-way handshake to establish communication channel between agents over the pre-established processes.

- *Start the communication*[5] *between the Agents by sending messages.* You can sent any detail as communication between agents.

- *Keep of log of the communication*. Each agents have to log the messages in the order in which they are transacted. Each agent has to keep a separate log file for each of the communication opened. Eg. if agent Chan is communicating with agent Jan and agent Ann he has to have two log files one for each communication.

- *wait for an ack.* Once you sent a message you will have to wait until you get the ack back. I have simplified by completely removing the window concept. So there should be a one-to-one mapping between a message and an ack. You won't sent a message until you get back an ack back.

- *Reliability: one in 5 packets gets lost between everyone, so you may have to recover it by correctly identifying it using the ack and retransmitting it*. This is one of your key challenges to ensure the agents gets all the messages in order. They can't a↵ord to miss any of the messages. You will have to run a timer at each agent side to estimate the loss of packet. You can set the timer to a value not less than 20 seconds. For this getting done you have field in the packet header called *DRP* which the router will check to decide whether to forward the packet or not.

Mission Objective #2: Asset Targeted

"In preparing for battle I have always found that plans are useless, but planning is indispensable" Dwight D. Eisenhower

Never a mission goes as planned. During the mission agent Ann suddenly find something fishy with agent Dans communication. Further intelligence suggested a possible physical intervention by the enemy to agent Chans location and an encounter which possibly could have killed Chan in action.

Your task is to eliminate the communication session with agent Chan and Download the transcript of communication for reporting to family and superiors.

- After the 5*th* message from agent Chan, agent Ann realizes something odd had happened and goes eerie about continuing communication with agent Chan.

- Agent Ann has to communicate this information with agent Jan. with an *URG* pointer on.

- Agent Ann has to download the complete set of communication transaction with agent Chan and terminate all the connections with agent Chan using the *RST* flag in TCP.

- There is pointer in the header field called the *TER* which enabled will terminate the process and you will exit out of the client program.

plays.com/comedies/battle of the bards.html

Mission Objective #3: Mission Completion

"It is fatal to enter an war without the will to win it" Douglas MacArthur

Once there are only two agents who are available now the mission is in a very delicate position. Jan working diligently with the secret service had established the position of target to be destroyed now. Now agent Jan has to get it approved by agent Ann and execute the mission. He reports the location to agent Ann using the *URG* pointer in the header of the packet and include the position coordinates (*32° 43' 22.77" N,97° 9' 7.53" W*) in the urgent field. Agent Ann confirms the location and identity of the target and orders to execute to agent Jan. Agent Jan establishes with the Air-force Headquarters and orders a air-strike. The mission goes well and now the target is destroyed. Now agent Jan has to terminate connection as per TCP norms and signal agent Ann about the success. Agent Ann congratulates agent Jan and request him to meet at coordinates (*32.76" N, -97.07" W*) to inform the mission head about the proceedings.

- Agent Jan signals agent Ann about his discovery of the location of the target using *URG* pointer. Request for a mission execution. • Agent Ann confirms the target by replying 'Execute' and gives out the authorization code for the Airforce Headquarters, 'PEPPER THE PEPPER'. • Agent Jan establishes a mission TCP connection with Airforce Headquarters directly and gives the co-ordinates. Asks them to eliminate the target along with the authorization code of 'PEPPER THE PEPPER' with the URG pointer set.
- Airforce Headquarters gets the code in data field and initiates the mission.
- Airforce Headquarters replies agent Jan about the success of the mission.
- Agent Jan informs agent Ann about the success with a code 'CONGRATULATIONS WE FRIED DRY GREEN LEAVES' with *URG* pointer set.
- Agent Ann congratulates agent Jan and sends a message to meet agent at (*32.76" N, -97.07" W*) to give the mission detail to mission head. • Agent Jan agrees and request for a finish mission-TCP request and terminates the session using *FIN*, and head out to the location.

- Agent Ann now gives the *FIN, ACK* and terminates her connection as well and leaves for the destination.