Data Link Layer (L2 Routing) Implementation (TCP/ IP Stack Functionality)

A Synopsis Report
in Partial Fulfilment of the Requirements
for the Course of
Minor Project - I
In
Third year – Fifth Semester of
Bachelor of Technology
Specialization in
DevOps and CCVT.

Under the guidance of

Ms. Avita Katal, Department of Virtualization

By

500067035	**R171218058**	**Kshitiz Saini**
500067409	**R110218131**	**Saloni Saxena**
500068293	**R110218107**	**Pratyusha Agarwal**

Department of Cybernetics and Virtualization
SCHOOL OF COMPUTER SCIENCE
UNIVERSITY OF PETROLEUM AND ENERGY STUDIES
Bidholi Campus, Energy Acres, Dehradun – 248007

July,2020

School of Computer Science

University of Petroleum & Energy Studies, Dehradun

Project Proposal Approval Form (2019-2020)

Minor 1

Project Title:
Data Link Layer (L2 Routing) Implementation (TCP/ IP Stack Functionality)

Abstract

In today's world, fast and efficient communication on the network between the sender and receiver is very important. For this communication, data is converted into packets and sent over the network using various routing algorithms. In a network or over multiple networks, routing refers to the process of determining a path for a packet to travel from. L2 Routing is based on the concept of Data Link layer and happens when data is to be transferred between the same subnet. In this project, we aim to implement the functionality of Data Link Layer like creating ARP tables, L2 Routing, MAC learning, L2 Switching & implementing VLAN forwarding.

Keywords: Network Routing, TCP/IP, Network Graph, ARP, L2 Routing, L2 Switching, Packet, Socket Programming, Topology, VLAN, MAC Learning

TABLE OF CONTENT

INTRODUCTION

This project focuses on implementing Layer 2 i.e. Data-Link Layer of TCP/IP Stack. The project mainly focuses on Graph Data Structures highlighting the features of **Generic Graph Topology.** The project will use routers, switches, links as well as nodes to depict the overall functionality of network topology. The entire packet journey (on sending and receiving machines) through the Data Link Layer of TCP/IP Model along with a minimal use of **Socket Programming** will be shown in this project.

The project focuses on the practical implementation of L2 Routing , L2 Switching [2] and VLAN based forwarding [2] where the concepts of ARP Resolution, MAC Learning and Forwarding will be used. Alongside, implementing the concept of Routing and Switching Algorithms, GL Threads, Timers, CLI Integration will also be in consideration.

Networks are treated as graphs G (V, E) consisting of V vertices/nodes and E edges.

- ➔ The graphs are bi-directional (undirected) because data can flow on both sides.
- ➔ Weighted graphs are used where weight indicates distance/cost etc.
- ➔ Connected graph is used as all nodes in a network are connected with each other.

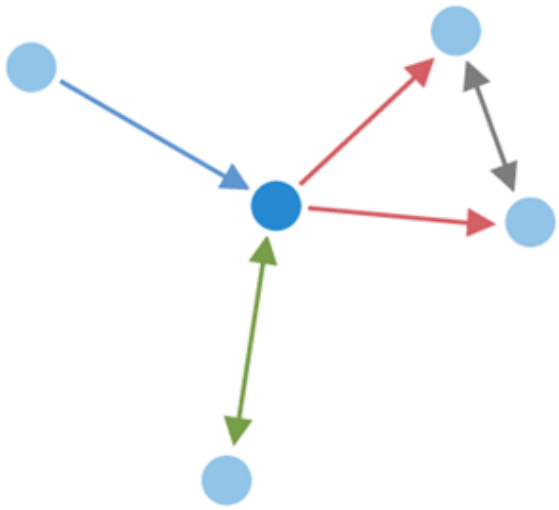


Figure 1 [4] Network G (V, E) with V = 5 and E =5

Figure 1 depicts a graph containing 5 vertices and 5 edges. It is a connected and undirected graph that can be used to represent a network.

Routing Algorithm [1] determines the path selected by a packet to reach its destination. It must decide within each intermediate router which output channel/s are to be selected for incoming packets.

Switching Algorithm [3] determines how network resources are allocated for data transmission, i.e. how and when the input channel is connected to the output channel selected by the routing algorithm. It is the actual mechanism that removes data from input channels and places them on output channels.

LITERATURE REVIEW

1) In [1], the author has focused on packet generation and reception and how ARP is important for their transmission. The paper contains the example demonstration Address Resolution protocol.

2) In [2], the author depicts how the implementation and design of policy based VLAN is carried out. It also focuses on the optimization of network resources.

3) In [3], the authors explained the complete process of how L2 switching works including MAC Address Learning, Look up table entries, Port migration and Hash code and table management.

PROBLEM STATEMENT

The transmission of data from one device to another without the help of TCP/IP stack is only feasible for a geographically small area but when it comes to worldwide solutions, it is not very appropriate. The problem without the TCP/IP Layer is to transmit data through the shortest path using features like creating ARP Tables and MAC Learning for an optimal resolution. Another problem is to create a network to share the data and a new medium has to be used for transmission every time.

When a packet is transferred from one machine to another in a subnet, the source machine requires the mac address of the destination machine for the frame delivery. Also, when the source machine sends the frame out of it then not only the destination machine but all the machines in that subnet gets the frame and later discards it. This process is called thrashing.

This project deals with sending frames from one machine to another in the same subnet reducing the problem of thrashing with the help of L2 switching. ARP protocol is implemented to get the MAC addresses corresponding to their IP addresses.

OBJECTIVES

The main objective is to implement the data link layer.

Sub-objectives for the project are:

- ➔ Developing a generic graph topology
- ➔ Tackling the issues of thrashing, collision and reducing broadcast domain.
- ➔ Implementing ARP, L2 routing, MAC learning and L2 switching.
- ➔ VLAN forwarding.

METHODOLOGY

Agile methodology of software development will be followed for the proposed project. The project is divided into 12 sprints where the sprint 7 and 8 will consist of parallel development by different members of the team. Each sprint is provided ample time to complete itself as well as to maintain the product's backlog (if any). The project can accommodate changes if required at any stage of the project. The sprints 1, 2 and 3 are specifically for requirement analysis and designing of the project. One sprint is specifically designed for setting the environment like maintaining the Version Control (Git in our case) and MakeFile. Each development sprint is followed by Unit Testing and an Integration Testing at the end. Sprints are also designed for the reviewing as well as retrospection part. After all the sprints the project is expected to complete by November 30.
The algorithms of sub-processes are defined below:

ARP Resolution:

1) ARP Request message is broadcasted to all devices on the network.
2) Nodes determine if they are the intended target.
    a) If nodes are the intended target, they proceed to STEP 3
    b) If they are not the intended target, they discard the request and proceed to STEP 4
3) Intended device sends ARP Response to the sender in unicast mode.
4) END

L2 Switching:

1) When the frame reaches the L2 switch, it updates its table with the MAC address of the source machine corresponding to the port number at which the frame is received.
2) L2 switch forwards the packet.
    a) If it does not contain the MAC address of the destination machine corresponding to any port, it floods the packet.
    b) If it contains the MAC address of the destination machine, step 3 is followed.
3) Frame is received only to the intended machine.
4) END

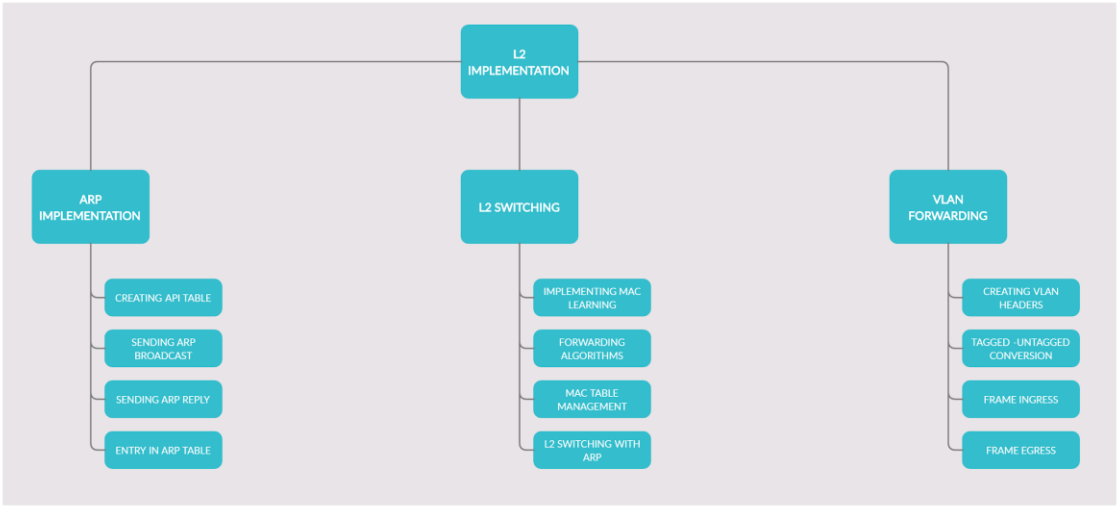


Figure 2 Different phases of the development part

SYSTEM REQUIREMENTS

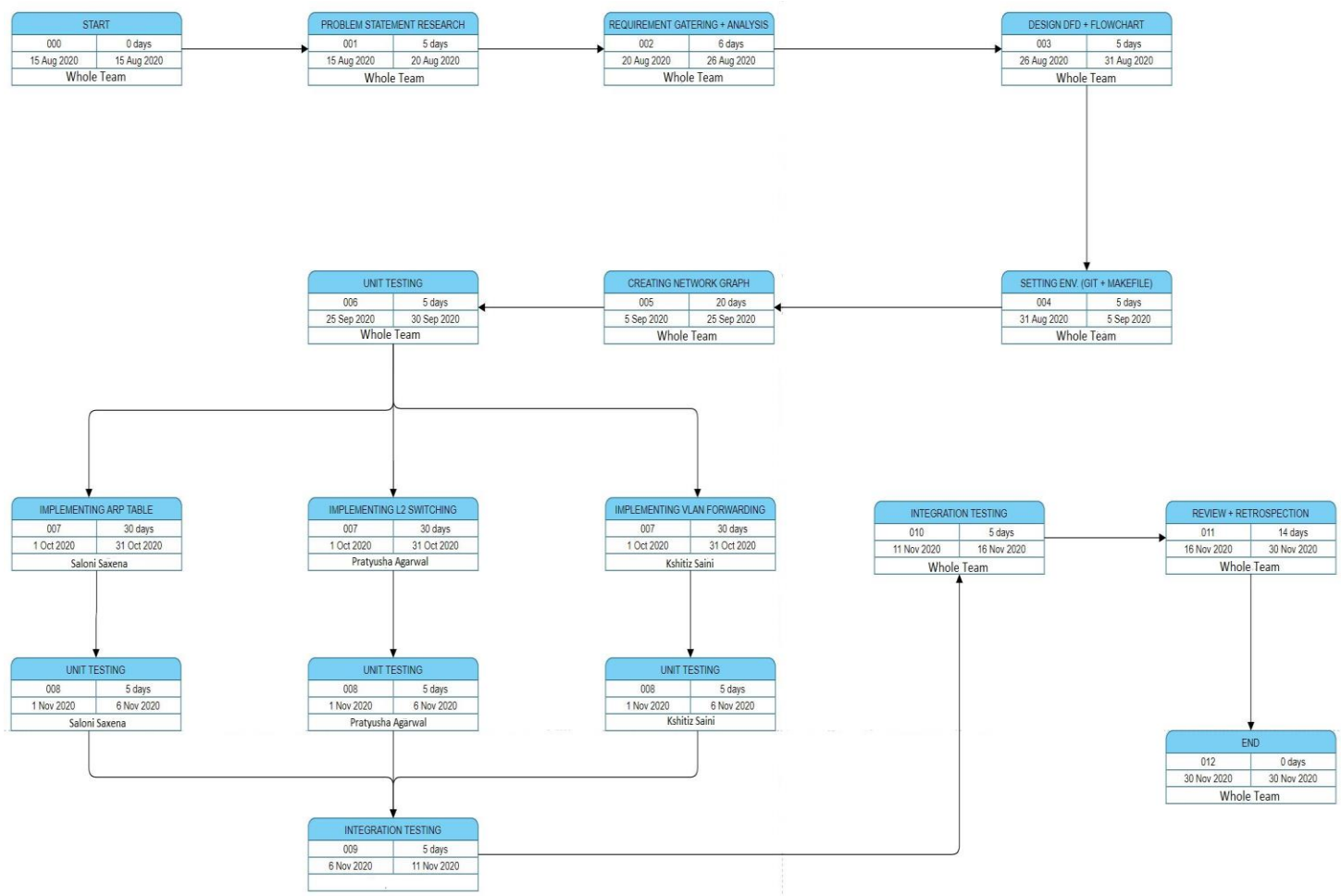
1. Software

- GCC Compiler
- Any flavour of Linux

2. Hardware

- 512 MB RAM
- i3 5th Generation or above processor

PERT CHART



REFERENCES

[1] C Plummer, **"**An Ethernet Address Resolution Protocol**"** : Massachusetts Institute of Technology (MIT), Internet Engineering Task Force, 1982

[2] M. Ernesto Gomez-Romero, M. Reyes-Ayala, E. Alejandro, Aandrade-Gonzalez, J. Alfredo and Tirade-Mendez , "Design and Implementation of a VLAN*"* : Metropolitan Autonomous University, Cinvestav Applied Computer Conference, 2011

[3] F. Balay and MorethanIP , *"*L2 Switch Implementation with programmable logic devices*"* : Germany

[4] Socilyzer, "Interpreting Social Network Diagram": Graph Interpretation, Socilyzer

Synopsis Draft verified by

Project Guide
(Ms. Avita Katal)

HOD
(Dr. Monit Kapoor)