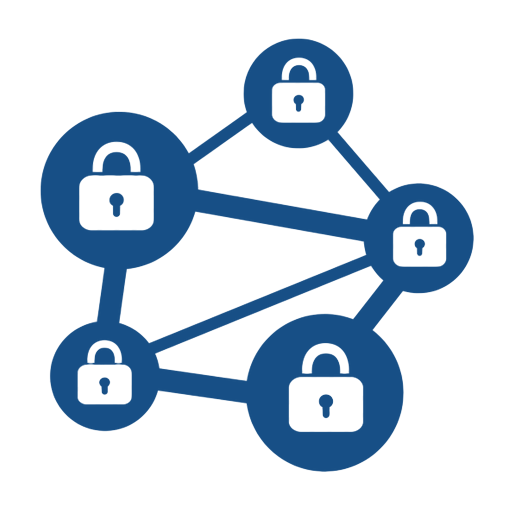


**3D3 COMPUTER NETWORKS**

**REPORT PROJECT 1:**

**IMPLEMENTING CONTROLLED P2P NETWORK**



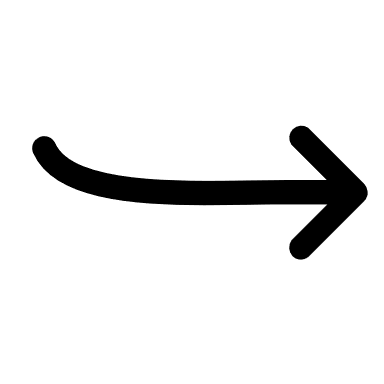
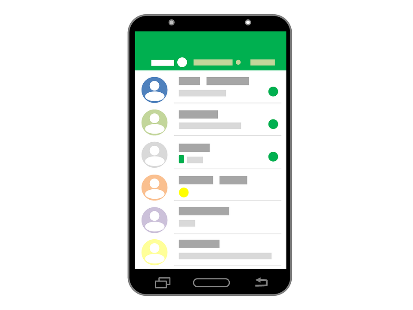
ANURAG JANGHALA

19324037

11TH FEB 2020

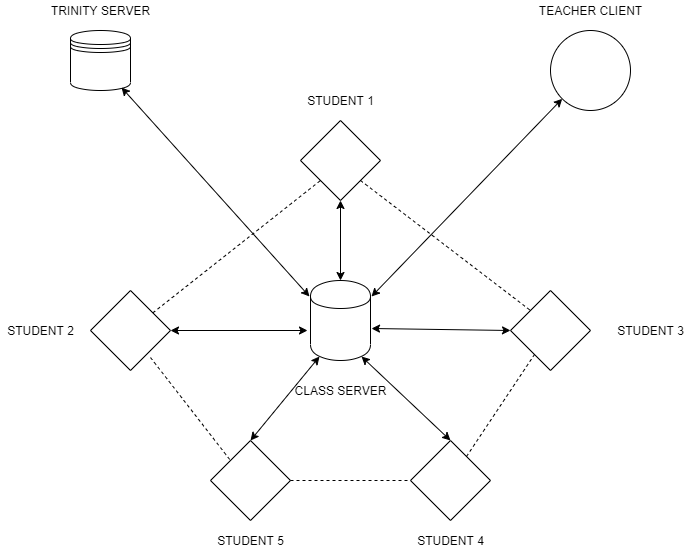
1. USE CASE:

I was working on a decentralised system that was based on blockchain in which I created a network for students, teachers and trinity server. Here, in this network student can record their presence in any activity inside trinity using IOT based device and mobile application, device will scan the data from mobile app of a student and then communicate with the class server and trinity server to verify that student data is right or not tampered on database and to store that presence of student inside a blockchain based database on teachers block. So here I am providing the details on how devices will communicate with each other to send, verify and store that information.



Also, I am only providing the communication between students as a client using IOT device and Class server.

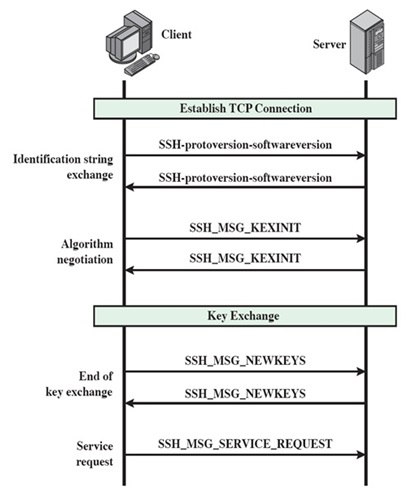
1. LOGICAL DIAGRAM:



1. FUNCTIONALITY:

* The functionality is simple for this networking model, as here the class server will be open for every scheduled timing for each class (1 hour) and if the client (student) can get access through the IOT device via mobile scanner (using Proto-life app), which will connect that client on a class server. This type of connection will be same for all the students (or clients).
* Also, teacher will get access as a client type. But there will be difference between teacher mobile app and student app, teacher client will hold some information of each student of that class in form of hash value. Rest works the same.
* Now, while each student client (say STU1) joins the class server, then the information or message of that student client (STU1) goes to the class server which later will send that information message to all other students of that class and to the teacher assigned for that class.
* Now for the verification of the presence of the student client (STU1) will send a private message or information to the teacher client and two specific students for verification of STU1 id and to check the presence of STU1.
* Here, STU1 will send private message to the student client (STU0 – who came earlier than STU1 in class) and to another student client (STU2 – who will come after STU1 in class) for the purpose of student presence verification.
* This will create a controlled P2P connection as its messaging will be controlled by class server. And all this verification of the data will be done using messages right now (later, it will be done using encoded-decoded hash value of student data).
* When student gets verified than the its presence gets stored in class server (block based on blockchain (yet to implement)) which will later be stored in trinity server database using information verification and later be stored in that trinity server block using blockchain (yet to implement).

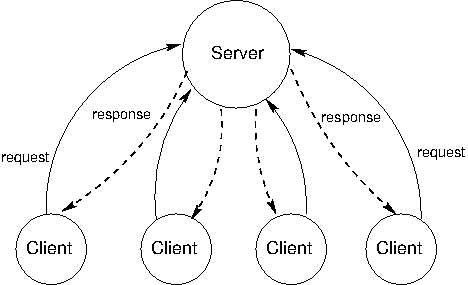
1. LOGICAL THINKING:



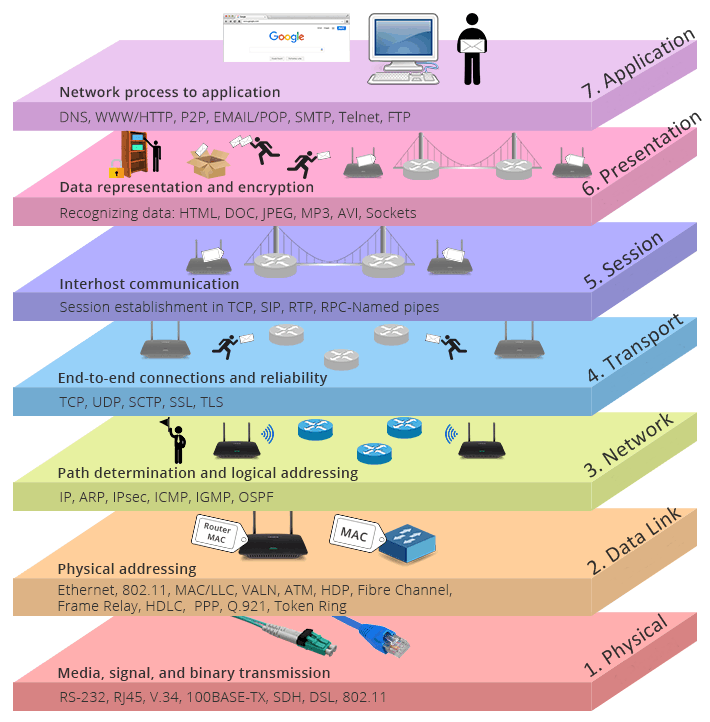
This will be based on hand shaking protocol.

Here, clients will connect to server and each client will connect to each other via server but in private mode so that no other client not involve in the verification will not connect to their conversation.

Also, the server will not record their info but will transfer in private mode to requested client which will maintain privacy between clients.



1. NETWORK MODEL:



In this network, I made the connection in such a way that it is transferring messages and data or information from application layer-> session layer-> transport layer-> session layer-> application layer.

1. OTHER CONSIDERARION:

For security purposes, I will be using the SHA256 for the encryption and decryption of the data/information in form hash value for the verification with the stored hash values for a student in the class server and other student local app-based server on their mobile.

But it is not implemented as of now (will be used in further improvements).

1. IMPLEMENTATION DETAILS:

* I have created this network on python-based socket programming which in scripted in python.
* Here I have used lots of python libraries like: socket, select, sys, argparse, signal, threading, termcolor, pickle ,etc,. in my code.
* Did the implementation of client and server-side code in a single file using the concept of class which will be controlled by select class to run the server and client side separately.
* Used argparse library to get arguments form the used if it is client or server which is setting up on port which will also be provided in argument.
* Other tools and library are used for the creating user friendly interface.

1. HIGHLIGHTS:

* Mobile app based
* Easy to use
* Works on blockchain (specifically Hyperledger technique)
* Verification model is easy to use
* Secure
* Works for reward system