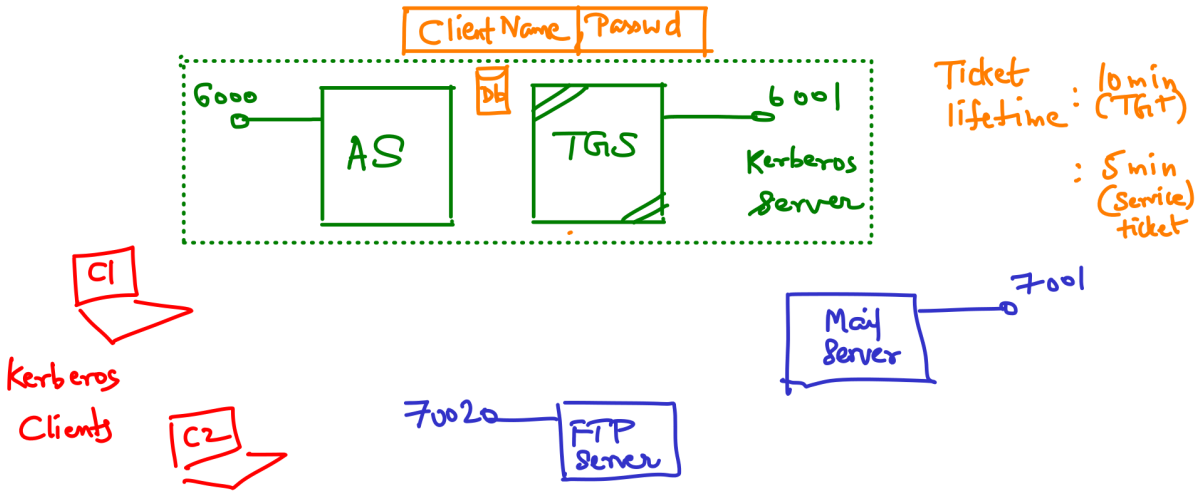
Network Security Laboratory Assignment: Kerberos Authentication System

**Objective:**

- To emulate the Kerberos Protocol

**Assignment Description:**

In this assignment, you will create a simple Kerberos Authentication System where a client can be authenticated and send requests to different servers securely. The Kerberos system will authenticate the client using a Key Distribution Center (KDC). After successful authentication, the client will be able to securely communicate with the server.




Refer to the skeleton code in python from
https://medium.com/@jeffreyleeon/kerberos-a-simple-python-simulation-of-network-authentication-560f4418dae3

You need to extend the above implementation with socket communication and emulate 2 clients and 2 servers as shown in the figure above. Further, you need to implement the authentication database that will contain authentication details of the client, <clientName, passwd> as well as details of the server, <serverName:port, passwd>. This authentication database will be accessed by the Authentication Server as well as the Ticket Granting Server. Recall from the class discussions, that the Kerberos 4 protocol includes Timestamps / lifetime for each ticket. To ease the clock implementation, we suggest the following: assume that the initial wall clock time is provided as an input. Also assume that the clock on each of these clients and servers is synchronized. Then, the current timestamp can be calculated as the difference between the current clock time and the initial wall clock time in minutes. Assume that the default lifetime is 5 min for service tickets and 10 min for the TGT.

Steps to Perform the Task:

1. Set Up Kerberos Infrastructure:
    - Install and configure the Kerberos server (KDC) on a machine: it should have the two modules, authentication server running on port 6000 and ticket granting server running on port 6001.
    - Set up Kerberos client on the client machine.
2. Kerberos Server Configuration (KDC):
    - Define server and Kerberos database. Typically, the combination of the host name and password.
    - Configure Ticket Granting Ticket (TGT): Ensure the KDC issues tickets for authentication.
3. Client-Server Communication Process:
    - Client Requests Authentication: The client will initiate a request to the KDC for a Ticket Granting Ticket (TGT).
    - KDC Authentication: The KDC will verify the client's credentials and send a TGT and a session key.
    - Client Requests Service Ticket: The client will use the TGT to request a service ticket from the KDC for accessing the server.
    - TGT issues a service ticket along with a session key to the client.
    - Server Verifies Service Ticket: The server will validate the service ticket sent by the client, and if valid, it will grant access.
4. Demonstrate Communication:
    - Once the client is authenticated, allow the client to send a secure message to the server.
    - The server should respond, acknowledging the secure communication.

5. Illustrate the working of timestamps and the reuse of the tickets during the lifetime of the ticket.