Inanc Dokurel 17575

Kamil Doruk Gur 17699

CS432 Project Report I

Description

This program is the first step of the term project consisting of only authentication and disconnection operations between clients and an authentication server. The authentication server and client do an authentication protocol on a predefined port and IP.

In order to function flawlessly, both server and client needs their respective key and info files in the same folder as their executable.

Workflow

The GUI of client and server are as follows:

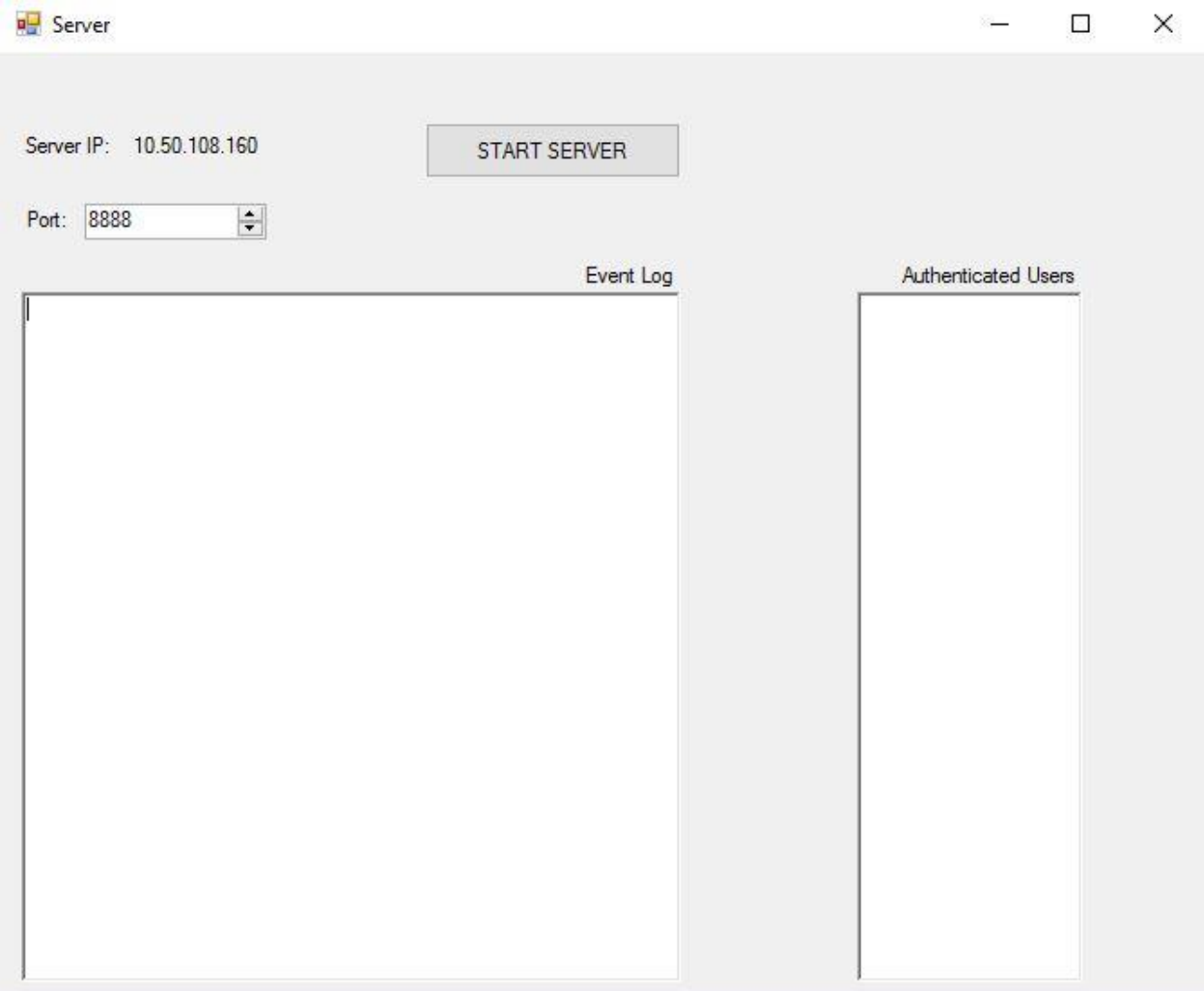

Figure 1.1.Server GUI

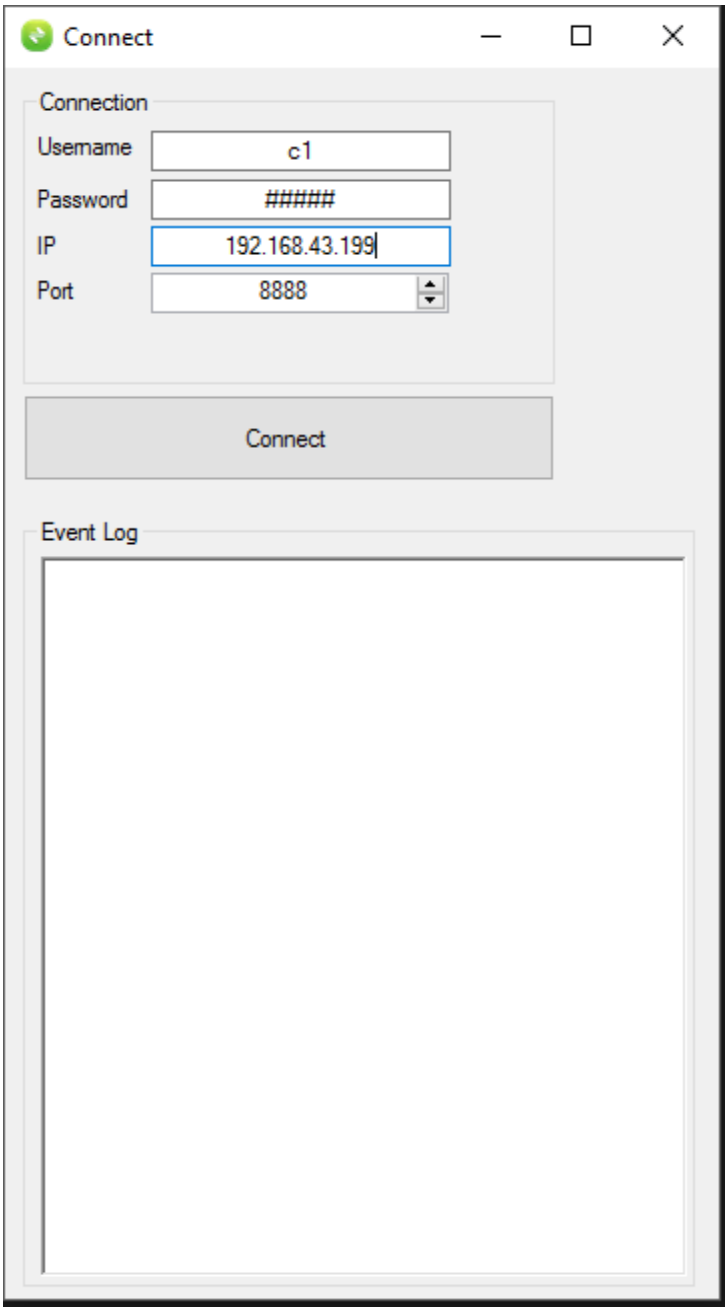

Figure 1.2. Client GUI

The server starts by choosing a port on from the GUI and starts the server with the click of a button. The IP of server is shown in a label, which cannot be changed since it is the IP of the adapter that is connected to the network. When a client wants to connect to the server first the IP and port number of server is entered via GUI, followed by user's username and password. Then password is used to decrypt users private and public RSA-2048 keys on client side with AES128 CFB mode with the SHA256 hash of the password entered, where the first 16 bytes are the key and the rest are the IV. If a wrong password is entered, this operation will fail and an exception will be thrown. If it does not fail however, the authentication operation will begin.

The server uses a header system while communicating with clients to understand which client requests which operation. Each command send by the client starts with the user name and followed by the command and the operation data whereas its server equivalent only uses the command followed by its data. When the password check is successful, client send its username with the command “init”. Server sees this command and generates a 128-bit random number back to client, with the command “init” again preceding. Client then parses this 128-bit number and signs it using RSA-2048 and sends both number and the signature preceded by the command “authenticate”. When server sees this command it tries to validate the signature of the random number with RSA-2048 public key of the client. Depending on the results it sends answer “yes” or “no” with its signature preceded by the command “fb”. If authentication was successful, client is added to the authenticated users list and shown on the GUI. Client parses the command coming from server again, first validating it is coming from server by verifying RSA-2048 with servers public key and then showing an authentication result based on the answer.

Encountered Problems

The project was flowing smoothly with the benefits we had from our previous experiences. All of the group members took the CS408 course in Fall 2015, which means that the term project was a server-client application, so built our project based on the working code of our CS408 project(some strings may even still reference its CS408 correspondence.) The only encountered problem was when it came to the encoding and message parsing since misunderstandings in where to use hexadecimal, byte encoding and the string itself caused us to have various validation errors during development, even though we had fully functioning code.