Design Document: Team Alpha

Security I

December 29, 2020

Daria Manea, Qiran (Tony) Li, Jacob Jordan

Table of Contents

Source Tree Content

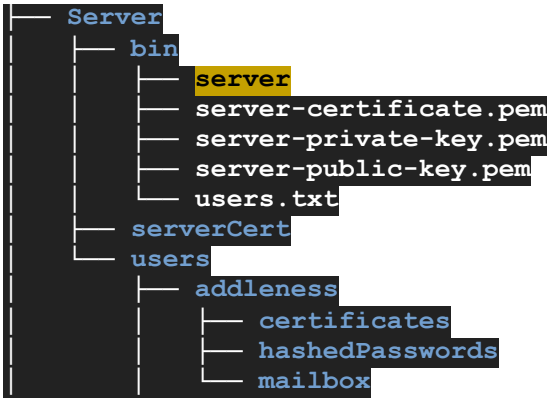
1. install-priv.sh: the main script to install the application
2. getcert.cpp: a cpp file that client would use to get cert from server
3. changepw.cpp: a cpp file that client would use to change password
4. sendmsg.cpp: a cpp file that client would use to send messages
5. recvmsg.cpp: a cpp file that client would use to receive messages
6. server.cpp: a cpp file that supports all features and functionalities of a server
7. users.txt: a txt file that stores users' names, hash passwords and passwords
8. Makefile: a Makefile that compiles all the cpp files
9. create-server-structure.sh: a script to create server's file structure
10. create-client-structure.sh: a script to create client's file structure

System Architecture

Operating System: Ubuntu 20.04 LTS
Development Platform: C/C++, OpenSSL

Server Side Program:

Sample tree structure:



- bin:
  - server: contains the main functionality of the server including launch the server, check password, get message, store message and send message
  - CA: store the certificates
  - scripts: a few bash scripts for creating the certificates
  - users.txt: a txt file that stores users' names, hash passwords and passwords
- users: contains username directories, each has the following directory::
  - Certificates: store user's certificate
  - hashedPasswords: store hashed passwords
  - Mailbox: user's mailbox
- serverCert: store certificates

Client Side Program:

- bin
    - getcert: A user logs in with a username and password, sends a public key, and receives a certificate. The server must also store this certificate.
    - changepw: A user logs in with a username and password, and supplies a new password. The user then receives a new certificate. Since this would render older messages unreadable, the request must be rejected if there are any messages pending for that user.
    - sendmsg: A user logs in with a client-side certificate, sends a list of recipient names, receives their certificates, encrypts the message to those certificates, digitally signs it, and uploads it.
    - recvmsg: A user logs in with a client-side certificate and receives a single encrypted message which is then deleted from the server. The signature on the message is verified and the message is displayed.
- users: contains username directories, each user has the following two directories:
    - Certificate
    - PublicKey

File Layout and Permissions

>Client: rwxrwx--x
    >bin: r-xr-xr-x
        >getcert: --x—x—x
        >sendmsg: --x—x—x
        >recvmsg: --x—x—x
    >Users: rwx------ user: clientcert group: clientcert
        >Certificate: rwx------ user: clientcert group: clientcert
        >PublicKey: rwx------ user: clientcert group: clientcert
>Server: rwxrwx---
    >bin: r-xr-x---
        >server: --x—x---
        >CA: rwxrwx--- user: ca group: ca
            >certs: rwxrwx--- user: ca group: ca
            >crl: rwxrwx--- user: ca group: ca
            >newcerts: rwxrwx--- user: ca group: ca
            >private: rwxrwx--- user: ca group: ca
        >server.sh:--x------ user: root group: root
        >ca.sh:--x------ user: root group: root
        >ca_interm.sh:--x------ user: root group: root
        >interm.sh:--x------ user: root group: root
        >user.txt:--x—x---
    >Users:
        >example user
            >Mailbox: mail will be read only
            >Certificates: read only

Testing

Functionality testing was conducted to ensure a user can send and receive messages. There was also testing to make sure that a user password could not be read by others.