Requirements Document

`Jammed` Alpha Release (1.0)

3.20.15

I. Personnel

Megan Carpenter (`mnc29`)
Daniel Etter (`dje67`)
John Hohm (`jh836`)
Marcos Pedreiro (`mvp34`)

II. System Purpose

A. The purpose of this system is to securely manage and store a users login information for various services. This login information comprises of usernames and passwords, and will be referred to as user data within this document. The system also ensures that only an authenticated user *a* is permitted to modify *a*'s user data. The system will be comprised of two main components, the server application (*sa*) and the client application (*ca*). The *sa* is responsible for maintaining user data, verifying and authenticating users, sending user data to the appropriate user, and performing any updates to the user data. The *ca* is responsible for transmitting a users login to the server to be authenticated, and then either alerting the user that the authentication failed, or receiving the user data to be displayed to the user upon successful verification. These two applications will be responsible for maintaining a secure connection, no data sent or received should be unencrypted, and also maintaining integrity to ensure the recipient of transmitted data is the expected and appropriate recipient.

III. System Backlog

Completed Tasks			
User type	**Assets**	**Importance**	**User story**
System Admin (SysAdmin)	Server Installation Server Configuration Files Server Private Key	M	As a *SysAdmin*, I install the server software onto a machine and configure it to listen for connections.
SysAdmin	Server Key Server Master Log	W	As a *SysAdmin*, I can review the log of all activity associated with the server database accesses and network connections.
User	User Login User Key	S	As a *User*, I can request access to my password data from the server.
User	User Login User Key User Data	S	As a *User*, I can modify my data and send the edited data back to the server for storage.

Incomplete Items			
User Type	**Assets**	**Importance**	**User Story**
Normal User (User)	User Login User Key	M	As a *User*, I can create a new account on the service. I can create a username and password, and provide a locally generated key pair.
User	User Login User Key Server Access Logs	W	As a *User*, I can request the log file associated with my data from the server.
User	User Login	W	As a *User,* I can change the password associated with my account on the server.

User	User Login	M	As a *User*, I can delete my account on the server.
User	User Data User Key	W	As a *User*, I can change the key associated with encrypting my data.
User	User Data	M	As a *User*, I can delete a password from the server

IV. Threat Analysis

A. We will not be addressing physical server security, we will assume physical safety.

B. We are also not responsible for a user who does not password protect their own machine with a password, although we can prompt a reminder for user to password protect their machine at install.

C. Adversaries

1. Outside Attacker
    a) Motivation
        (1) Gain access to user data to gain access to various login accounts.
        (2) Delete or modify a user's data in order to prevent them from accessing it.
        (3) Modifying a user's login information to prevent the user from accessing their data.
    b) Resources
        (1) Powerful computer.
        (2) Network connection.
        (3) Automated bot to perform an arbitrary attack.
    c) Capabilities
        (1) Ability to monitor traffic between *sa* and *ca*.
        (2) Ability to intercept and modify packets sent between *sa* and *ca*.
        (3) Ability to send requests directly to the *sa* and *ca*
        (4) Ability to run any given crypto attack they can think of
2. System Administrator
    a) Motivation
        (1) Gain access to user data to gain access to various login accounts.
        (2) Delete or modify a user's data in order to prevent them from accessing it.
        (3) Modifying a user's login information to prevent the user from accessing their data.
    b) Resources
        (1) Powerful computer.
        (2) Network connection.
        (3) Automated bot to perform an arbitrary attack.
        (4) User usernames and securely stored passwords.
        (5) The system log.

c) Capabilities
    (1) Ability to monitor traffic between *sa* and *ca*.
    (2) Ability to intercept and modify packets sent between *sa* and *ca*.
    (3) Ability to send requests directly to the *sa* and *ca*.
    (4) Ability to run any given crypto attack.
    (5) Ability to establish a secure connection to the server.
    (6) Ability to act as root on the server.
d) Non-Threat
    (1) We will assume that the system administrator is a benevolent actor. However, the design of the system prevents any administrator from decrypting user data, as the user's private keys are not known, and the login passwords will be stored securely.

3. Normal User
    a) Motivation
        (1) Gain access to other user's data to gain access to various login accounts.
        (2) Delete or modify another user's data in order to prevent them from accessing it.
        (3) Modifying another user's login information to prevent the user from accessing their data.
    b) Resources
        (1) Powerful computer.
        (2) Network connection.
        (3) Automated bot to perform an arbitrary attack.
        (4) Access to the system via their own account.
    c) Capabilities
        (1) Ability to monitor traffic between *sa* and *ca*.
        (2) Ability to intercept and modify packets sent between *sa* and *ca*.
        (3) Ability to send requests directly to the *sa* and *ca*.
        (4) Ability to run any given crypto attack.
        (5) Ability to establish a secure connection to the server.

V. Security Goals

A. Assets
   1. User data
      a) Information stored within the application.
   2. User login
      a) Username and user password, used for logging into an account.
   3. User security keys
      a) Public and Private Keys, the latter of which will be stored offline on the user's machine.
   4. Server database
      a) File structure and encrypted user data.
   5. Physical server
   6. User computer/terminal
   7. Server Private key

B. Stakeholders
   1. The Administrators of the system
   2. The Users of the system

C. Harm Analysis
   1. Unauthorized access to users *a* data could result in usernames and passwords being stolen and used to access user *a*'s other accounts.
      a) The system shall prevent unauthorized access to unencrypted user data. **C**
   2. An attacker with access to user login creds could download user data they do not have authorization for.
      a) The system shall prevent login creds from being accessed by unauthorized users. **C**
      b) All stored user data on server is encrypted. **C**
   3. An attacker with the users secret key could decrypt the users data
      a) the system shall never transmit the secret or any private key over network, private/secret keys shall be only stored locally on users machines. (maybe: the keys will be stored locally in an encrypted state, only to be decrypted upon successful user login). **C**
   4. Someone intercepting a user login request could gain knowledge of that users login credentials
      a) Login requests shall be sent in an encrypted state to the server. **C**
   5. An attacker with a users login credentials could submit an unauthorized request for the users data.
      a) The system shall use digital signatures to verify that requests and messages come from the specified recipient. **I**

6. An attacker who has gained system admin privileges could have access to the entire database.
    a) The system shall store user data in an encrypted state that only an authorized user can read it. **C**
    b) Note: Currently no plan to counter deletion/corruption of the database.
    c) System shall detect if user data has been modified while on the database by anyone who's not the user. **I**
7. An attacker with a user account *a* might be able to submit a request for a separate user *b*'s data causing a breach of confidentiality.
    a) The system shall only supply user *a* with user *a*'s data. **C**
8. An attacker could tap into the session connection and submit an unauthorized request to the users machine for data.
    a) Messages from the server shall be verified via digital signature. **I**
    b) The server shall not initiate requests of the client. **C**
9. DY attacker could intercept and modify transmitted data thereby corrupting files being transferred.
    a) The system shall maintain integrity by signing and verifying all messages sent between the *sa* and the *ca*. **I**
10. DY attacker could intercept and not forward the message on which could cause loss of data for an update, or unavailability of service.
    a) The system shall rely on signed confirmation messages to verify user/server receipt of messages. **I**
11. An attacker could send an executable file as the userdata and datamine or destroy the database.
    a) The system shall disallow the execution of any arbitrary code potentially found within user files. **I**
    b) The system shall not use files or their contents to determine system behavior. **I**
    c) The system shall not allow one users upload to affect another user's data. **I**
12. An attacker could try to deny service to the server by bringing the network down
    a) Availability issues are beyond the scope of this course, and will not be handled. **A**
13. An attacker might try to submit a package request to the server without validating their user identity.
    a) The server shall only handle requests from authorized users in a verified session. **I**

14. The server could accept an unencrypted packet from the user to store on the database violating the confidentiality of the user data.
    a) The *ca* shall only ever transmit encrypted data. **C**
15. A user could apply changes to their data and wish to submit changes to the server but never receive an upload confirmation message, which will signify that they will be unable to save their changes.
    a) The application shall provide a user with a warning after *n* failed attempts that their data was not saved. **A**
16. An attacker intercepting a message packet between the *sa* and *ca* could breach the confidentiality and/or integrity of that message.
    a) The system shall only transmit encrypted and signed messages between the *sa* and *ca* (which will then be verified upon arrival). **I**

VI. Essential Security Elements

A. *Authorization*

1. This is essential to our system because our system relies on the fact that every user is authorized to access their data, and only a user *a* is authorized to access *a*'s data. If this security element were not in place, then the system would not function for secure storage of sensitive data.

B. *Authentication*

1. The system must authenticate connections between the *sa* and the *ca* for each user to ensure that no data is transmitted to an unauthorized principal. Otherwise there would be no way to ensure the principal requesting the data is actually authorized to access it.

C. *Audit*

1. The system shall maintain a log of all transactions on the system so that the system administrator can review the log for possible discrepancies as needed. This log shall consist of both the network connections and the access log for each user's data.

D. *Confidentiality*

1. The system must maintain confidentiality of the user data such that it is inaccessible by unauthorized principals. Otherwise the system would not be able to perform its intended role as a storage space for sensitive data.

E. *Integrity*

1. The system shall check to make sure that all messages are received as they were sent, and also that the data stored in the server is not corrupted by any outside action. This is a necessary feature of the system because it ensures that the data which the users have entrusted to the system is maintained in the correct state and can continue to be useful to the user.