

METU CENG491 2015 FALL

START-UP DOCUMENT

G01P11

Group Name: *Codebenders*

Project Name: *Linux Password Vault*

1. System Architecture

- *Draw the overall system architecture diagram. This should include (but it is not limited to) the components of the system, the interactions among the components and their dependencies.*
- *Identify and describe each component (including subcomponents if any), their interactions and dependencies clearly.*
- *Specify the user interaction model.*

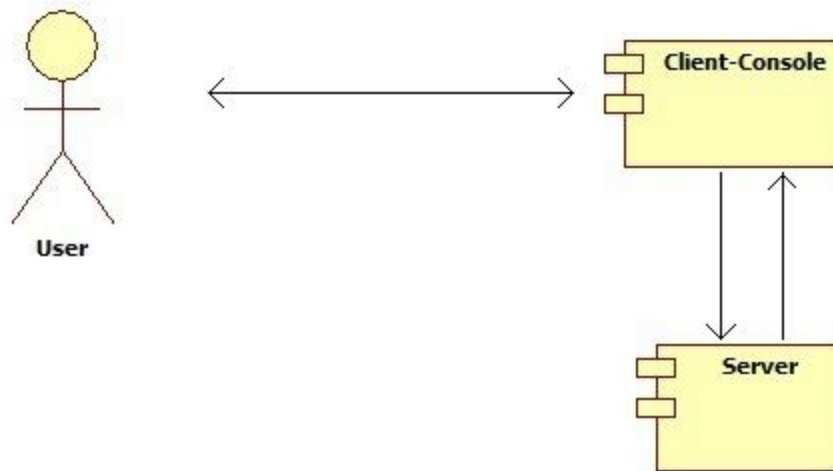


Figure 1: System architecture diagram

There will be two main components in the project: server (password daemon) and client (console based). It could be a viable choice to add a web service to the project at a later point after the full implementation which at the moment is completely optional.

The project is primarily aimed at commercial use. It could also be benefited by end users with optional interfaces.

The server's job is to store the passwords of the company. It will ensure secure access to the large number of passwords used by its employees within the company. It will be robust and viable. It will receive the saved passwords and encrypt them for further security. It will also provide the client with a stored password when the correctly authorized client requests its own password.

The client's job is to securely send the password to be stored to the server and to receive it back from the server when in need. It will not store any passwords but transfer them directly from the password daemon. It will not try to breach other's passwords on the server and will only authorize itself to receive its own password using keys and authorization methods.

The optional web interface would provide end users with saving, restoring and managing tasks of the server and the client using a graphical user interface.

2. Tentative Time Plan

- Identify and itemize all tasks to be performed as a team in the first semester. Assign a unique TaskID for each task. Give a short name and brief description for each identified task.

TaskID	Short Name	Description
T1	Role Assignment	Assign roles for team members, and distribute tasks.
T2	Decide PL	Gather info about possible programming languages which can be used for development, and choose the most appropriate one for our project.
T3	Learn PL	Study and learn the chosen PL for development.
T4	Learn Git	Study and learn git, and choose an appropriate development model to apply using git.
T5	Security Research1	Gather info about possible models of 2-factor authentication, and decide on the most appropriate one for our case (to be used between client and server).
T6	Security Research2	Gather info about possibilities to securely store the data on server (using plain files or a DB?).
T7	Early Implementation	Implementation of the data resource according to security researches.

- Construct your time plan as a simplified Gantt chart, as shown in the following table.

	Iteration1	Iteration2	Iteration3
T1	■	■	
T2	■	■	
T3	■	■	
T4		■	
T5		■	■
T6			■
T7			■

3. Deliverables

- *Identify and list all deliverables of your project for the first 3 sprints.*
- *A deliverable is some component or sub-component, which is running and demonstrable to your assistant and your supervisor. That deliverable is of course subject to improvement over time.*
- *Fill in the following table:*

Deliverable	Description	When? (Sprint#)
D1	Role Assignment	1
D2	Decide PL	2
D3	Learn PL	2
D4	Learn Git	1, 2
D5	Security Research1	2, 3
D6	Security Research2	2, 3
D7	Early Implementation	3

4. Workload Distribution

Fill in the following table to distribute the workload for the first semester among your team members.

	Sprint - I	Sprint - II	Sprint - III
Ali Can	T1, D1, T4, D4	T2, D2, T3, D3, T5	T5, D5, T7, D7
Gökhan	T1, D1, T4, D4	T2, D2, T3, D3, T5	T5, D5, T7, D7
Mehmet Sait	T1, D1, T4, D4	T2, D2, T3, D3, T6	T6, D6, T7, D7
Muhammet	T1, D1, T4, D4	T2, D2, T3, D3, T6	T6, D6, T7, D7