Instituto Superior Técnico  
Universidade de Lisboa

**Smart Restaurant**

Network and Computer Security, Alameda, Group 1

|  |  |  |
| --- | --- | --- |
| 77900 | Luís Filipe Abreu dos Santos | luis.jpg |
| 77961 | Pedro Filipe Ferreira Fernandes | pedroboavida.jpg |
| 77966 | Catarina Alexandra Pereira Cepeda | catarinamartins.jpg |

# 1. Problem

Given the restaurant scenario we have # key issues to address in terms of security. The first being client privacy, we need to guarantee that any personal information like account and orders information cannot be accessed by unauthorized parties doing this by separating sensitive data from common information. Another issue is ensuring that all orders can be properly tracked back to a given individual to minimize the risk the restaurant has to assume.

(Given the chosen scenario, why is security necessary? What is the main problem being solved? Use around 200 words)

# 2. Requirements

* Ensure that all client information is kept private;
* Ensure client privacy during the order process (so as to maintain orders cannot be traced back to certain clients by unauthorized personnel or bellow a given k-anonymity)
* Authenticate orders coming from the client and guaranty that orders can’t be repudiated;
* Guarantee authenticity of the restaurant when the app connects to the server;
* The system must be able to locate the position of the client in the restaurant (e.g., table number);
* Encrypt data flows between the network and the outside (eg secure channels)
* Maintain public network separate from local private network;
* Maintain two separate databases to certify that invoices are kept separate from user information;
* Guaranty K-anonymity for the clients (e.g., when the restaurant collects information about its meals for statistical analysis, prohibit queries that return a dataset that is too small);
* Protect the client information stored on the mobile device (eg using trust storage in android and password locking the app);

# 3. Proposed Solution

SIRS_system.png

The above diagram highlights key components of the proposed architecture for the system. We will assume that all key-pairs have been previously installed and distributed between the internal components and that the external clients use an authentication protocol like token passing or Oauth2.

**Basic:**

Ensure the authenticity of all connected components and make sure that any messages sent are clearly identified and confidential in the case of communication between the clients and the Server.

**Intermediate:**

Separate data of incoming requests so that personal information cannot be retrieved unless by a trusted party (ie Manager). Ensure that the information in the user database is confidential.

Isolate the internal network and implement a application level firewall for incoming requests from the outside.

**Advanced:**

Protect payment transactions??? K-anonimity

# 4. Tool Preferences

Android Studio – tested

MySQL Database – tested

Webservice (which one? jax-ws, java-remoting, restful api)???

found/installed/tested/well-tested

# 5. Work Plan

|  |  |  |  |
| --- | --- | --- | --- |
| **Week** | Catarina | Luis | Pedro |
| 10-31\_11-04 |  |  |  |
| 11-07\_11 |  |  |  |
| 11-14\_18 |  |  |  |
| 11-21\_25 |  |  |  |
| 11-28\_12-02 |  |  |  |
| 12-05\_09 |  |  |  |