Unsolvable Solutions

Client: CSIR DSSR

Architecturall Requirements

Eavesdropping Protection in Conclave

Github link: https://github.com/Unsolvable-Solutions/Project-EPIC

Members:	*Student Number:*
Edwin Fullard	12048675
Jaco Bezuidenhoudt	11013878
Jandre Coetzee	10693077
Maret Stoffberg	11071762
Ryno Pierce	12003922

Contents

1 Architecture Requirements

1.1 Access and Itegration Requirements

1.1.1 Human Access Channels

- Via Web_Browser:The user must be able to access the data from the server these standard web browser: Mozilla Firefox, Google Chrome and Microsoft Internet Explorer. It must have a user friendly GUI.

- Via Andro̲id device:This device is used for the malware as well as the protection software.

- Via NFC Nodes̲ and an Edison:This is used outside the meeting for the access control and the protection application.

- The M̲alware :The user access the mobile Malware Application via the server and the server can also be accessed via the Mobile Application.

1.1.2 System Access Channels

- Web̲site: There is interaction with the server via the website.

- Edi̲son: The Edison send requests to the server and receives responses.

- NF̲C: The NFC receive a email address and password from the mobile application and sends the data then to the Edison.

- The M̲alware: The Malware server sends request to the Appication and receive an audio stream.

1.2 Quality Requirements

1.2.1 Scalability

- EPIC : The server must be able to handle a large amount of users and users must be able to use the system simultaneously. However, the controlled access to a meeting, via the Gateway and the Node will only have a limit of one gateway per meeting and at most three nodes per meeting.

- Malware : The server can only handle one live stream recording at a time.

1.2.2 Performance Requirements

The system does not have specific performance requirements, but the application to gateway operations should respond within less than 1 second, because it may delay the meeting unnecessarily. The server query operations may process up to 5 seconds long.

1.2.3 Maintainability

The system must be easily maintainable. The application and server may be updated in time, to fit the latest technologies.

1.2.4 Reliability and Availability

The application must be available from the Google play store, and the access to the web server can be found online at projectepic.info.

1.2.5 Auditability

The system log the entrance and exit of all meetings. For each meeting the log contains the users allowed, the users that attended, their entrance and exit times as well as the initial time and place of the meeting.

1.2.6 Security

The NFC components should not be hackable.

1.2.7 Monitorability

The response of the Nodes is visible with the LED light showing the permission.

1.2.8 Testability

All the components must be tested separate at first, unit testing, and after completion the whole system must be tested.
For each service the pre and post conditios must be met.

1.2.9 Usability

All the components must be intuitive and easy to use. Any user that is Android literate and computer literate must be able to use the system. The physical components must be labelled accordingly.

The User Manual may be used for more information on any of the components.

1.2.10 Integrability

The different components of the system should work together and the system must also be able to handle future additions to it.

1.3 Architecture constraints

- The Mobile application can only be used on Andriod version 4.4 and 5.1.
- The Node can only communite with one device at a time.
- The Node can only work if the specific Mobile Application is installed.
- The Edison can only be connected to three Nodes maximum at a time.
- The Malware application can only be used on Android version 4.4 and higher.
- The Malware server has a minimum Java version 7.
- The Malware may run on any operating system that supports Virtual Machine.

2 Architectural patterns or styles

Different structural patterns are used. Some of these patterns strech over the whole system, while others are just used in parts of the system.

2.1 Layering

The whole system is devided into Layers, and each layer only communicate with its adjacent layers.

2.1.1 Layers:

1. The user
2. The Mobile Application
3. The NFC Node
4. The Edison
5. The webserver
6. The website
7. The user

2.1.2 Advantages

Pluggability One layer can be replaced or changed, and only the adjacent layers will have to be updated. For example if another webserver is needed, or if the Android Mobile Application is not enough and a iOS application is needed.

Reusability Since the layers are developed to function relative separately, they can be reused by another system, for example the webserver or the NFC Node.It is possible for any of the layers to be reused by another system, the code would only have to be adjusted a littleif needed.

Testability Since every layer performs its own methods relatively separate from the rest of the layers, unit testing can be done very easily. Each of these layers can be tested separately with pseudo input values from the adjacent layers. If the layers perform as it should, they can be tested together.

Complexity reduction All the tasks are devided into the different layers, and therefore the system is symplified. Each layer only has to perform its own tasks.

Cohesion There is high-level localization.

Maintainability The layers are seperate, and are therefore easier to manage and update. It can also be developed by different developers simultaneously.

2.1.3 Disadvantages

High Maintenace Cost If a lower level is changed, the higher level sometimes also need to be adjusted.

Communication Overheads The information could be send unnecessary through many layers. For example when the user scans the mobile device, the username and password is sends from the phone to the NFC, then Edison, then webserver, and a respond is send back through the same layers. Now it could be argued that it would be easier for the application to communicate directly with the webserver, but all the layers inbetween performs additional tasks in the process, which is needed.

2.2 Model-View-Controller

Ther server uses a MVC architecture.

Model It is represented by the MongoDB database that the server uses.

Viewl The web interface serves as the view.

Controller All the tasks and functions that the user may do via the weppage, like adding a meeting or changing the user profile are the controller.

The usage of the MVC architecture makes the server more maintainable, testable, reusable and simple.
The **maintainability** can easily be seen in the development. Each of the MVC can be developed separately and just be merged afterwards.
The separations of the different concerns makes the **testability** so much easier. Unit testing can be done and the whole server can then also be tested.

The **reusability** can be seen with the Model, in this case called the database.
The database is also used by the Edison for the access control.
The separation of the different concerns shows the **simplicity** of the server.

3 Architectural tactics or strategies

3.1 Thread pooling:

It is used to improve scalability, by the recording handler of the Malware Application and to create multiple connections to different client phones. The Edison is connected in serial to more than one node to improve scalability(thus physical threads). The server can also handle multiple requests simultaneosly to update the database from different web clients.

3.2 Clustering:

All communication and events are stored on the server. This is used to improve scalability and reliability.

3.3 Interception:

For any event the username and password must be visible and correct for the event to proceed. This ensures security and auditability. The website uses a Log in System, and the user may not proceed to the website if he has not logged in. The Mobile Application only works with the username and password, it is send to the Node, via NFC, in the access permission request.

3.4 Authentication

The NFC module and the mobile application share a unique AID (Application Identifier) which allows only the mobile application with the correct AID to communicate with the NFC Module.

4 Access and integration channels

4.1 EPIC

- The user interact and have access to the system via the website (www.project-epic.info). The user may register as a user, log in, create a meeting,view all upcomming and past meetings and view,delete and edit his own account. The website then sends the requests for these functions to the server. If the user is an administrative user, they may additionally add more administrators, remove meetings, view all user details.

- The Edison sends a request to the server for the attendant list by sending the meeting identification. The Edison then have the list of attendees and updates the list every hour or on a request.

- The Nodes communicate with the Edison via serial port. The Nodes send the email address and password that they received to the Edison to confirm if the user may enter the meeting. The Edison respond with a true or false, if the user may enter or not. The Nodes neopixels will then turn red if access is denied and green if access is granted.

- The Mobile Application communicates with the Node via NFC. It sends the email address and password to the Node and receive a response stating if the user may enter the meeting or not. If access is granted the phones screen turn green and all communication is shut down, otherwise the screen just turns red.

- The user have to install the Mobile Application. If the user wants to scan the phone, he has to open the application before scanning. The first time the user opens the application he has to fill out his details.

4.2 Malware

- The user may acces the server to control the application, the server then communicates with the application via TCP

5 Technologies

- NFC
- NodeJS
- Arduino
- MongoDB
- Passport
- Java
- Android SDK
- TCP
- UDP
- HCE
- INDEF
- JavaScript
- AngularJS