Evaluation guideline

Guideline for the evaluators

Below you will find all the questions. It’s important to read the comment section to ask the questions in such a way that you don’t influence the answer. Since it’s qualitative research, don’t be afraid to continuously throw the “Why?” question. It’s advised that you **record** and **immediately write** down the answers.

Questions

|  |  |  |
| --- | --- | --- |
| **INTRODUCTION** | | |
| Get to know about person’s expertise in IoT and Access Control Security models | | |
| Intro | What is your education level? (BSc, MSc, PhD) |  |
| Intro | How do you describe your knowledge level on IoT and Access Control security models? |  |
| **Adding New Device / User** | | |
| 1. Requirement 1: New users or devices can be easily added to be IoT solution | | |
| Scenario | Questions | Comments |
| Adding a new user pauline | * Is it easy to add a new user? * Is it easy to add capabilities to new user? * Any improvement points? |  |
| Adding new devices: Button1 and Ringer | * Is it easy to add a new device? * Any improvement points? |  |
| **Delegation & Transfer**   1. Requirement 2 : Access control can be delegated from the owner to others | | |
| Scenario | Questions | Comments |
| Login as admin.  Go to the Capabilities list screen. Delegate the capability to view /data/profiles/Pauline to user Pauline  Login as Pauline, check whether you have access to information in Pauline’s profile.  Transfer a capability to jack. Once transferred, the capability will no longer available to the first owner.  Login as jack, check whether jack is able to access to the information that you just transferred.  Note: Delegation/transfer can only be done for users/devices that are already registered with Igor | * Is it easy to delegate capabilities to new users / devices * Any improvement points? |  |
| **Revocation**   1. Requirement 3 : Access control can be revoked by the owners when not needed anymore | | |
| Scenario | Questions | Comments |
| Login as admin. Revoke the capability that you just delegated to Pauline.  Logout. Login as Pauline. Go to the capabilities screen. Check if the capability is now removed from your capability list.  Note: There are 2 types of capabilities: internal and external capabilities. External capabilities are assigned to external IoT devices. | * Is it useful to revoke access? * Any improvement points? |  |
| **Fine-grained access control**   1. Requirement 4 : Access control of data should be to the detailed level (fine-grained) | | |
| Scenario | Questions | Comments |
| Login as ID: admin  Go to the Capabilities list screen. Grant access to Pauline the information in /data/people/jack/device  Logout. Login as Pauline. Check if Pauline is able to see the information in /data/people/jack/device | * Is the access control model fine-grained enough? * Any improvement points? |  |
| **Easily Managed and Modified**   1. Requirement 5 : Access control should be easily managed and modified for normal users | | |
| Scenario | Questions | Comments |
| Login as ID: pauline  Go to the Capabilities list screen  Try to modify access for the capabilities granted to Jack | * Is it useful to manage and modify access? * Is this function simple and easy? * Any improvement points? |  |
| **Lightweight – suitable for IoT devices**   1. Requirement 6 : Access control solution should be lightweight as IoT devices have low processing power (with acceptable performance) | | |
| Scenario | Questions | Comments |
| The implementation part of the proposed framework is carried out by setting up Igor on a Raspberry Pi 3. The core of Igor are an XML database, a web server and Xpath 1.0. | Is the performance acceptable?  Any improvement points? |  |
| **Other design choices (Optional)** | | |
| Area | Questions | Comments |
| Design choice 1 - Separation of policy administration and checking from the XML database. A separate Python module is used (access.py) to function as the Policy Decision Point (PDP) and PAP (Policy Administration Point). This is to enable better code management and reduce errors when replicating capabilities and performing policy checks | Do you think this design is useful? Why?  Any improvement points? |  |
| Design choice 2 - There are several approaches to perform delegation. Gong’s approach is for the initiator to create a delegation message and pass it to the person who need the capability. He/she will then request access from the access server by presenting the message from the owner and the access server will then generate new capability and assign it to the requestor. We did not go with that approach as it presents too many steps and is cumbersome for the users. Instead, the capability owner can login to Igor using his/her credentials and delegate the capability through a user interface and the receiver of the capability will be notified through email that he/she has been granted access to certain objects. We have also given the choice to the token owner to specify if the newly delegated token can be further delegated by the new owner or not. | Do you think this design is useful? Why?  Any improvement points? |  |
| Design choice 3 - We have also made the choice to enforce that all external tokens have limited lifespan (1 year) and a renewal process is forced upon the token holder after the expiry date. | Do you think this design is useful? Why?  Any improvement points? |  |
| Design choice 4 - Each agent (device, people, plug-ins, other Igors) have their own profiles or listing in the XML database where the assigned capabilities and other credentials (e.g. logon ID, passwords, secret keys, etc.) are stored for ownership tracking purpose | Do you think this design is useful? Why?  Any improvement points? |  |
| Design choice 5 - We have chosen to have two types of capabilities: internal and external. Internal capabilities are used to control access to data, actions and plugins within the Igor’s database. For better performance, internal capabilities do not to use any encryptions. External capabilities are used by Igor to control access for external IoT devices that are requesting services from Igor. For this implementation, Igor and the external IoT devices use secret symmetric keys to establish trust and JWT is used to transfer generated token from the “token generator” in Igor to the external IoT devices | Do you think this design is useful? Why?  Any improvement points? |  |
| **Summary** | | |
| Questions | Comments |  |
| What is your overall feedback about the usefulness of the design of the fine-grained access control for Igor? |  |  |
| How would you rate whether the project meets the overall objectives in address ethical/privacy issues we are facing regarding information generated by IoT devices? | Rating from 1 – 5 (where 1 is the lowest and 5 the highest) |  |
| **Information about the person** | | |
| Age  Gender  In which industry do you work (study)?  What is your job title? (if applicable) |  |  |

**Appendix - Implementation of external devices (Button and Ringer)**

