BehavAuth

Human behaviour based authentication for Internet of Things

Open Call Project for TagItSmart: Grant agreement 688061

WP2: D2.1 Final software tool

Deliverable ID:	**D 2.1**
Deliverable Title:	**Final software tool**
Revision #:	**1.0**
Dissemination Level:	**Public**
Responsible beneficiary:	QBE
Contributing beneficiaries:	All
Contractual date of delivery:	28/02/2018
Actual submission date:	16/3/2018

Start Date of the Project: 1 September 2017 Duration: 6 Months

Document history

Revision	Date	Modification	Authors
1.0	16/03/2018	Draft ready for review	N. Palaghias

Means of delivery

The main objective of this document is to provide the means to the deliverable D2.1 (Final software tool) and to guide the reader on how to get access to the software components and the source code of these components that were developed during the BehavAuth project and through the support of the TagItSmart project.

During the BehavAuth project, two major software components were developed in order to comply with the Description of Work. The first software tool is a mobile app that performs continuous authentication of the user based on behavioural patterns without considering any username and password. The second software tool is a mobile app that receives authentication estimations from the different authentication tools of the TagItSmart platform and merges these estimations into an authentication result that is forwarded to the User Repository component of the TagItSmart platform.

BehavAuth is a mobile app/library that performs authentication of the user by leveraging sensor data generated by the integrated sensors of the smartphone device. The mobile app/library collects sensor data, extracts behavioural cues that will allow the derivation of behavioural patterns of the user and further will enable the validation of the identity of the user. The sensor data are initially pre-processed in order to convert them in the appropriate format. Following, the normalised sensor data are forwarded to the behavioural cues extraction components. Each of the behavioural cues component processes the sensor data that are related to the particular module. The data stream is fed in the behaviour extraction module that learns and authenticates the user. The results from the learning and authentication process are forwarded to the authentication inference component that fuses the results from the different authentication components. Once the authentication inference component fuses the different authentication results, an overall authentication decision is generated and propagated to the interface that the BehavAuth library exposes to third party apps such as the Polling Manager.

The BehavAuth app has been released as open-source and can be accessed from Quadible's publicly accessible repository at https://bitbucket.org/quadible/behavauth. The source code for the BehavAuth app is released under the GNU General Public License 3 software license.

The ***TagItSmart Polling Manager app*** is an accompany app that enables the integration of the different authentication components such as BehavAuth, location based authentication (provided by the University of Surrey) and keystroke based authentication (provided by the University of Padova). The Polling Manager operates mainly in an opportunistic manner. This means that the users just need to install the mobile app. After the installation process, the mobile app operates on its own without the needs of the users input. From the point that the app will be installed, it will monitor the communication bus about any incoming messages and in case it receives any messages it will forward them to the authentication voting mechanism and the result of it will be propagated to the User Repository of the TagItSmart platform.

The Polling Manager has been released as open-source and can be accessed from Quadible's publicly accessible repository at https://bitbucket.org/quadible/polling-manager. The source code for the Polling Manager app is released under the Apache 2.0 software license.

Both repositories contain the source code and are accompanied by a short wiki to enable the users to deploy the app and the developers to extend and reuse the source code.