SMARTBEAR

CALL H2020-SC1-FA-DTS-2018-2020
Trusted digital solutions and Cybersecurity in Health and Care
TOPIC DT-TDS-01-2019
Smart and healthy living at home

SMART BEAR

"Smart Big Data Platform to Offer Evidence-based Personalised Support for Healthy and Independent Living at Home"

D23 (D5.5)– Continuous Security Assurance & Privacy by design - enabling mechanisms v3 Demonstrator

Due date of deliverable: 30/09/2023
Actual submission date: 28/02/2024

Grant agreement number: 857172
Start date of project: 01/09/2019

Lead contractor: CNR
Duration: 60 months

Project funded by the European Commission within the EU Framework Programme for Research and Innovation HORIZON 2020	
Dissemination Level	
PU = Public, fully open, e.g., web	✓
CO = Confidential, restricted under conditions set out in Model Grant Agreement	
CI = Classified, information as referred to in Commission Decision 2001/844/EC.	
Int = Internal Working Document	

D23 (D5.5) – Continuous Security Assurance & Privacy by design - enabling mechanisms v3 Demonstrator

Editors

L. Koumakis, G. Kalogiannis, O. Dountouraki (STS)

Contributors

Athanatos Manos, Barmpaki Anthi, Diamantaris Michalis, Hatzivailis George, Ioannidis Sotiris, Kopanaki Depoina, Lakka Eftychia, Michalodimitrakis Emmanouil, Shetsov Alexander (FORTH)

Reviewers

Dimitrios Boucharas (UOI)
Lisa Gius (2B)

The work described in this document has been conducted within the project SMART BEAR, started in September 2019. This project has received funding from the European Union's Horizon 2020 research and innovation programme under grant agreement No 857172

Executive Summary

This document presents a demonstration of the version 3 of security and privacy mechanisms that were developed during the third period of the Smart Bear (SB) project. It serves as an update of the version 2 (presented in D5.3). As such, it describes changes to the rationale introduced by version 3, as well as updates of the functionality supported by the Security Component to meet new requirements deemed necessary by clinicians utilising the services and providing their feedback (reported in D5.6).

Contents

List of acronyms

API	Application Programming Interface
CCM	Clinical Case Manager
GDPR	General Data Protection Regulation
GUI	Graphical user interface
HDO	Help Desk Operator
IT	information technology
REST API	Representational State Transfer API
RBAC	Role-Based Access Control
SPAP	Security and Privacy Assurance Platform
SB	Smart Bear
SB@Cloud	SMART BEAR Cloud infrastructure
SB@Dashboard	SmartBear dashboard component
SB@SecurityComponent	SmartBear Security Component
STS	Sphynx Technology Solutions
TRL	Technology Readiness Level
UEBA	User and Entity Behavior Analytics
UI	User interface

List of tables

List of figures

1 Introduction

The Smart Bear (SB) architecture, as illustrated in Figure 1, was designed with a core focus on delivering a secure, scalable, and modular environment. The aim of this deliverable and WP5 as a whole is to support established best practices in security and privacy, such as Role-Based Access Control (RBAC), API authentication, secure transmission channels, and adherence to GDPR principles. The overarching goal of the SB project is to provide an affordable, securely accountable, and privacy-preserving platform. This platform is capable of ingesting usage data from commonly available smart and medical devices (e.g., health, wellbeing, and environmental measurements) for subsequent utilization in the realm of machine learning analytics, achieving a Technology Readiness Level (TRL) of 9. Figure 1 shows a high-level software architecture of SB, showcasing interdependencies among the main building blocks. The primary objective is to enhance the quality of life for elderly individuals dealing with five prevalent health-related conditions: Hearing Loss, Cardiovascular Diseases, Cognitive Impairments, Mental Health Issues, and Balance Disorders.

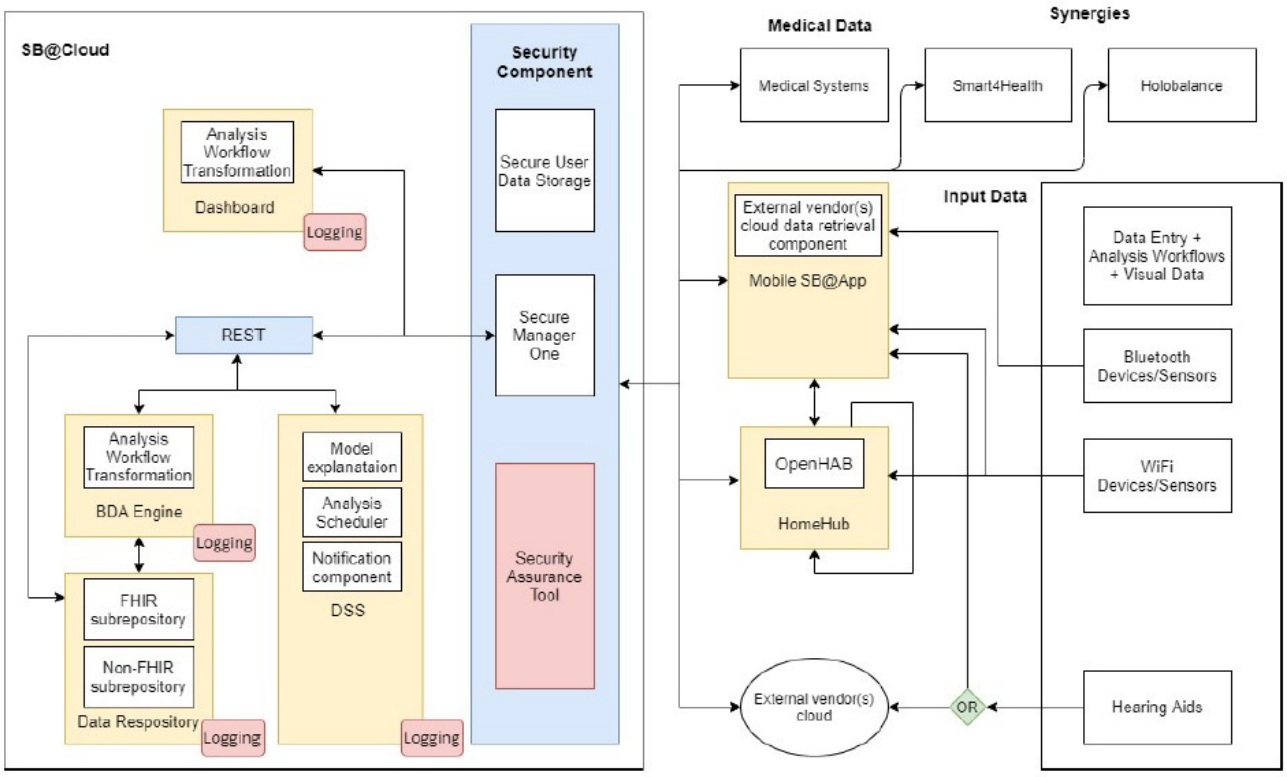


Figure 1: SB final architecture showing the interdependencies among the main building blocks.

The implemented functionality, including that supported by the Security Component (SB@SecurityComponent), has undergone testing since spring 2022. This testing phase involved the monitoring of study participants to evaluate the technical quality and validity of the provided services. Throughout this real-world pilot operation, identified errors were addressed and debugged, and the performance of the mechanisms was assessed to ensure the platform's readiness for the main pilots. Concurrently, feedback from end-users was actively collected to enhance the platform's user experience, leading to the introduction of upgrades that align with the project's objectives.

1.1 Purpose of the document

As outlined in the project's timeline, the SB platform, inclusive of the functionality supported by the SB@SecurityComponent, is slated for rigorous testing and validation through five large-scale pilots. These pilots will be conducted across five distinct countries, involving thousands of individuals. The overarching objective of these pilots is to assess the platform's performance in the realm of healthcare service delivery. This evaluation will encompass both private and public healthcare providers at regional, state, and EU levels. The aim is to demonstrate the efficacy, extensibility, sustainability, and cost-effectiveness of the SB platform, offering valuable insights into its impact on both individuals and the broader healthcare system.

1.2 Delta – Updates from the second iteration (D5.3)

The primary focus of this deliverable is to outline end-user scenarios related to the incorporation of new functionality, which was introduced to address feedback from clinicians and other users. Additionally, the deliverable introduces the Security and Privacy Assurance Platform (SPAP) within SB. The development of the new functionality builds upon existing foundations, with the final architecture outlined in D2.2 and the core components supporting authentication, authorization, and services presented in D5.2, D5.4 and D5.6. Furthermore, the design principles for newly requested services, such as preventing the assignment of the same device ID to two different patients, align with state-of-the-art security and privacy guidelines.

The main update for the security component comes from the API Manager and Identity server. Initially, the deployment utilized WSO2 as a key subcomponent, encompassing the API Manager and Identity server. However, due to technical challenges encountered during the pilot operation, a decision was made to replace the WSO2 subcomponents. Keycloak now serves as the replacement for the Identity server, and KrakenD has taken the place of the API manager—both of which are open-source solutions. This substitution was implemented before the commencement of the main pilots, ensuring continuity in supported functionality despite the change in underlying components.

Pending action: At present, the patient management component is under the administration of the technical partners of SmartBear. In anticipation of the SmartBear platform being delivered as a comprehensive solution post-project completion, a strategic decision has been made to introduce a new, user-friendly interface dedicated to patient management within the SmartBear dashboard UI. This initiative entails updates on the UI of the SmartBear dashboard and creation of new API for the communication with the backend of KeyCloak. These enhancements will be further elaborated upon the forthcoming deliverable (D5.7).

2 Prerequisites for Installation and Configuration of the updated components

2.1 Keycloak and KrakenD Basic Hardware and Software Prerequisites

Prior to installing the Keycloak and KrakenD Identity Server and API Manager, basic subcomponents that support authentication and authorisation services, it is necessary to have the appropriate prerequisite software and hardware installed as presented in the following table (Table 1):

Table 1: Keycloak software and hardware prerequisites for massive traffic

Type	**Min requirement**
CPU	4vCPUs (x86_64 Architecture)
Memory	4 GB RAM
Disk	10 GB disk space, excluding space allocated for log files and databases
Software dependencies	• JDK 8 compliant JDK, available in most common platforms that support Java 11 or Java 8 • Java SE Development Kit (JDK)* • JavaScript enabled Web Browser (e.g., Google Chrome, Microsoft Edge, Mozilla Firefox, Apple Safari)
Databases	PostgreSQL (PostgreSQL License is a liberal Open-Source license, similar to the BSD or MIT licenses)

3 End-User Management (Keycloak version)

3.1 Configuration

3.1.1 End-User Roles Supported

Based on the Role-based access control (RBAC), permissions (i.e., access level to REST API end points) to all Smart Bear Dashboard registered end-users can be assigned based on their role. While each role may be configured having one or more roles, according to the Smart Bear end-user management policy, each end-user can have only associated to one role. The transition from WSO2 to Keycloak didn't affect the SB end-user roles. More details about the management policy can be found in deliverable D5.3.

By design, upon completion of the installation and configuration of the SB@SecurityComponent, the system administrator is created, which contains all the permissions in the permission tree. By this role, different user roles can be defined depending on a project's requirements. This role is responsible for configuring the API management (e.g., defining the list of APIs, manages the API Gateway), for defining access policies, and for creating roles, end-users, assigning them roles. To achieve the goals of the project, the following roles have been defined:

1. System Administrator (SA)
2. Patient (or Study Participant) (P)
3. Clinical Case Manager (or Clinician) (CCM)
4. Caregiver (C)
5. Data Scientist (DS)
6. Auditor (A)
7. Help Desk Operator (or Technical Support) (HDO)

3.1.2 Creation of end-users (Keycloak version)

From the user interface of Keycloak, the administrator can add/edit users.

3.1.2.1 KeyCloak Realm creation

By default, Keycloak is installed with an admin user, and a "master" realm. A realm is the logical collection of a set of users, credentials, roles, and groups. A single user belongs and logs into a realm. For example, the "admin" account, which is present by default, belongs to the "master" realm. To keep the Keycloak administrator and the Smart Bear application's administrator separate, a new realm needs to be created, titled "smartbear". This realm will hold all end-user accounts, credentials, and roles.

To create the new realm, the system administrator:

1. Logs in at the Keycloak administration dashboard (Figure 2).
2. From the menu (on the left) hovers over the "Master" dropdown and clicks on "Add realm" (Figure 3)
3. Inputs the new realm name and clicks "Create"

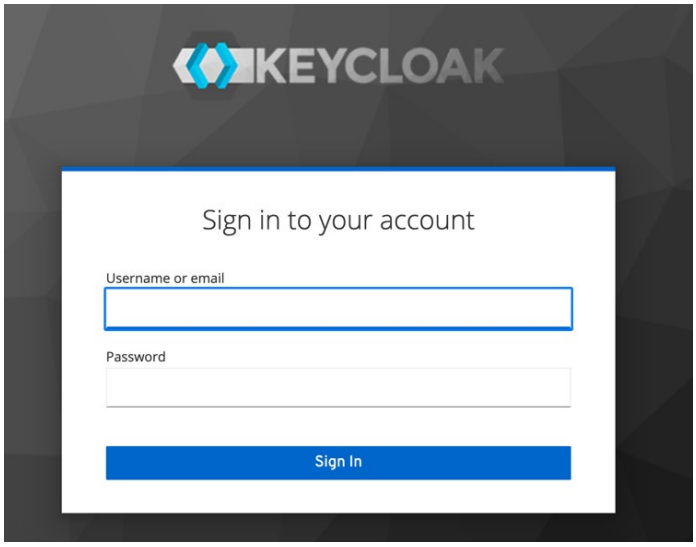


Figure 2: KeyCloak Login page

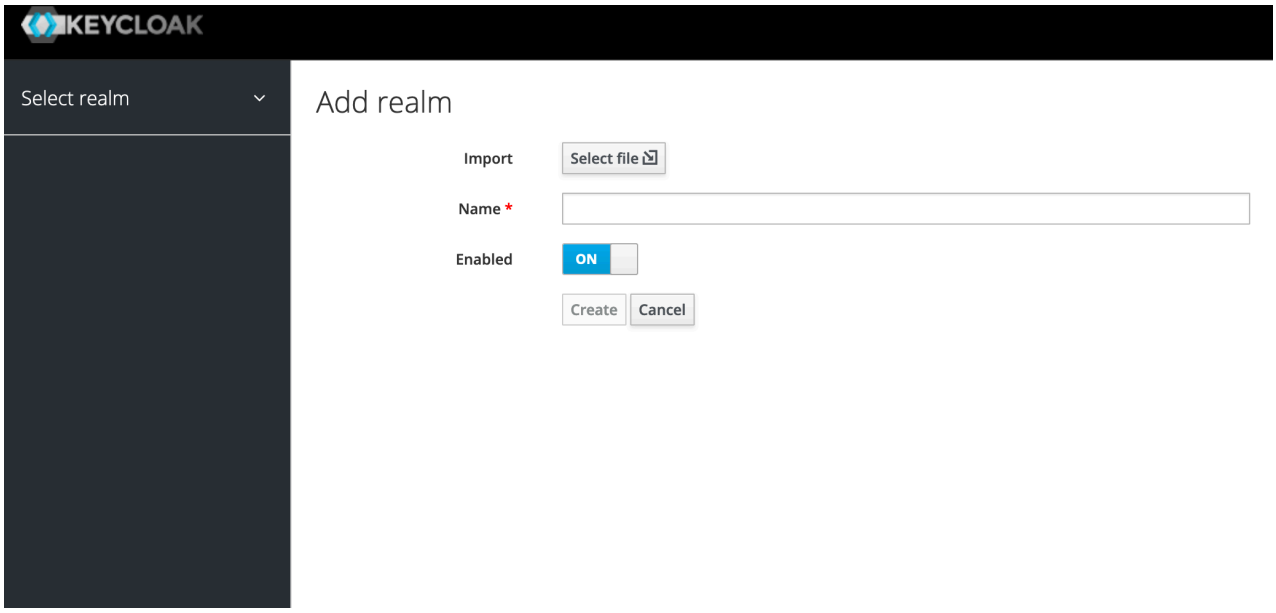


Figure 3: KeyCloak Realm creation.

3.1.2.2 KeyCloak User creation

Following the realm creation, users can be created by selecting “Users” on the sidebar, then clicking on “Add user” (Figure 4).

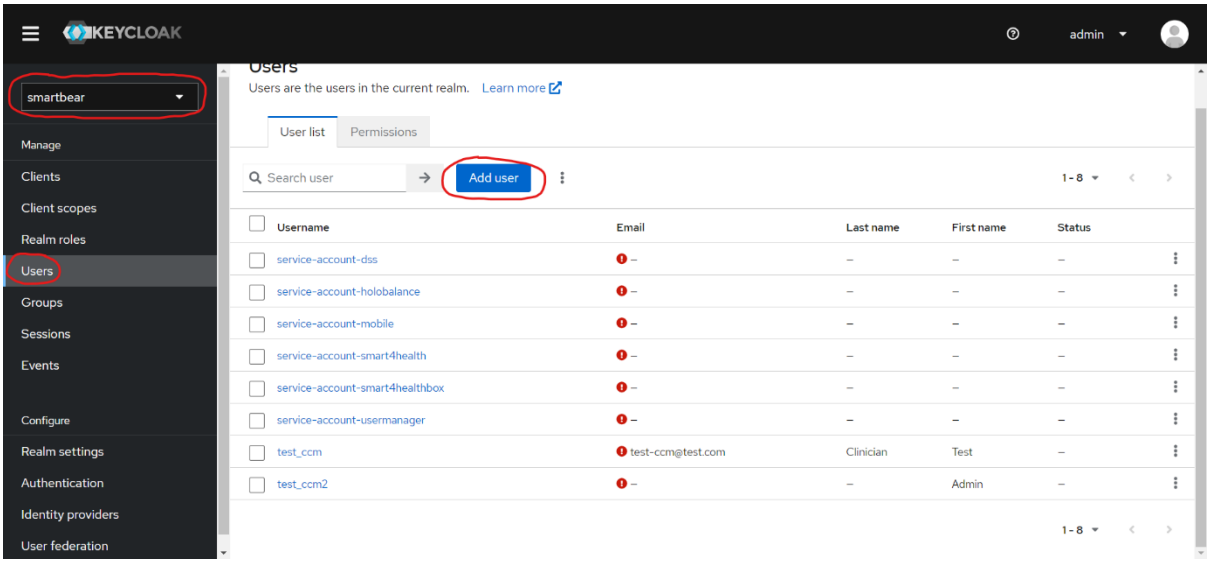


Figure 4:User creation.

Then input user details, whereby email can be verified manually (from UI), or alternatively, an email can be send to the user to verify the email (an additional step after user creation) and click “Create” (Figure 5).

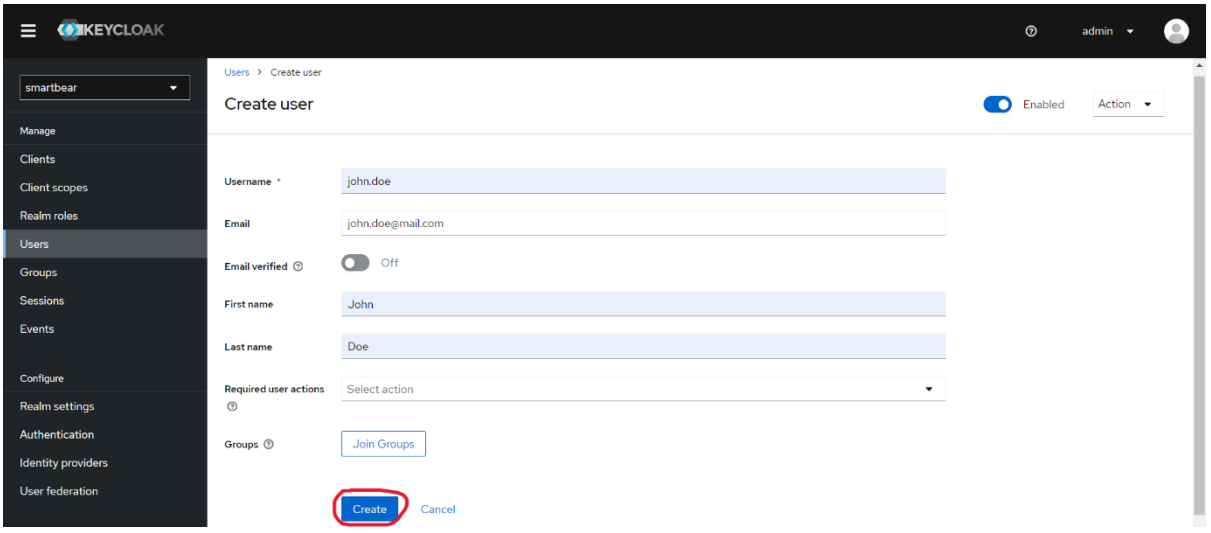


Figure 5: KeyCloak create user details.

If the user has been created successfully, the KeyCloak UI will be redirected to the user “details” tab (Figure 6).

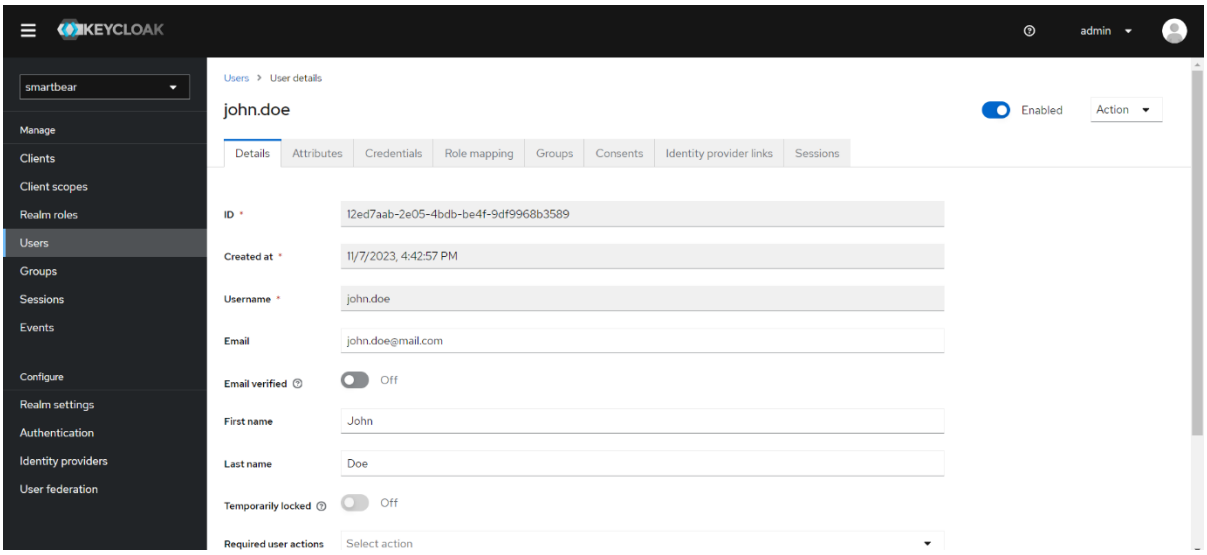


Figure 6:KeyCloak view user details

To add a user organization, one must go to the “attributes” tab. In this example organisation=STS (Note: make sure strings typed correctly (e.g. not organization with ‘z’, use ‘s’)) and click ‘save’ (Figure 7).

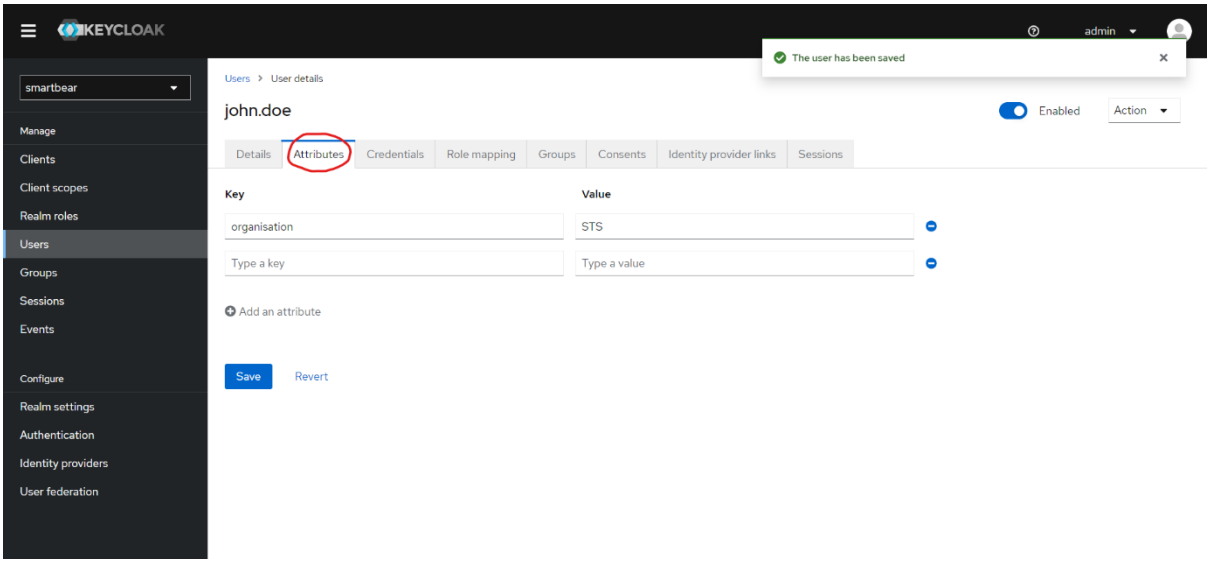


Figure 7:KeyCloak user organization

Then, head to ‘credentials’ tab, click ‘set password' as shown in Figure 8.

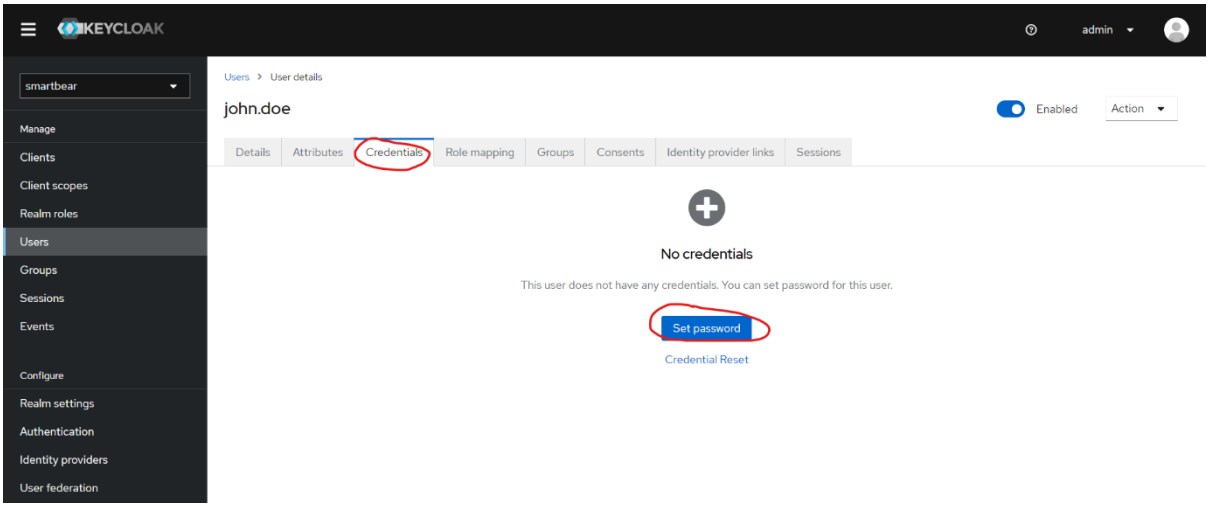


Figure 8: KeyCloak set credentials for user.

Type the password. If temporary option is selected, then when the user will be prompted to change the password upon their initial login. Subsequently, click on the ‘save’ button (refer to Figure 9).

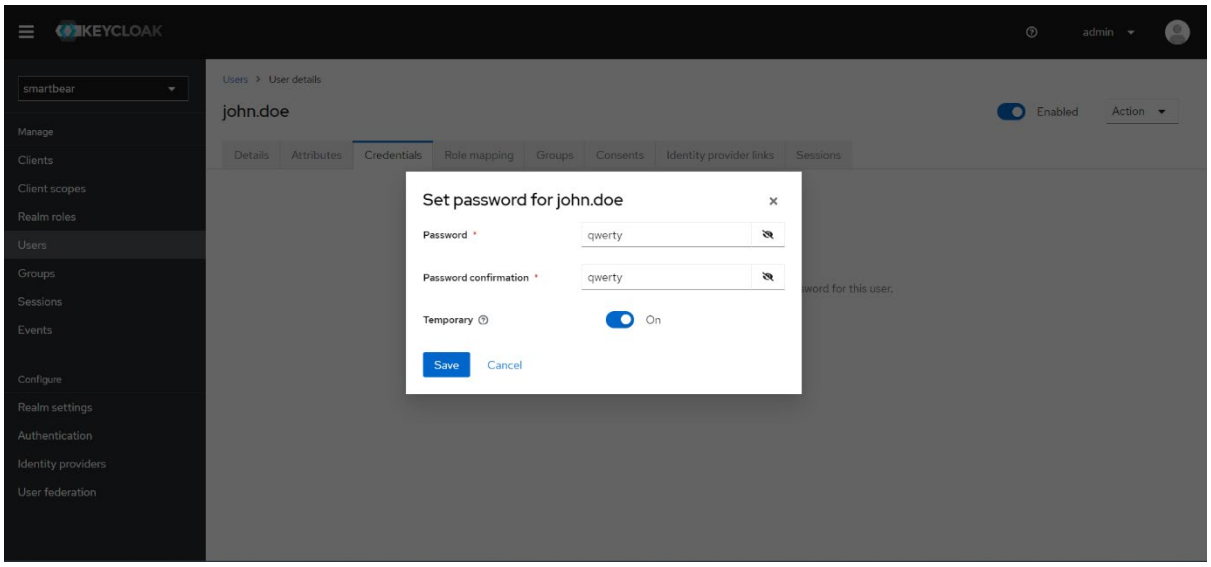


Figure 9: KeyCloak set user password.

Go to the ‘role mapping’ tab and click on ‘assign role’ (Figure 10).

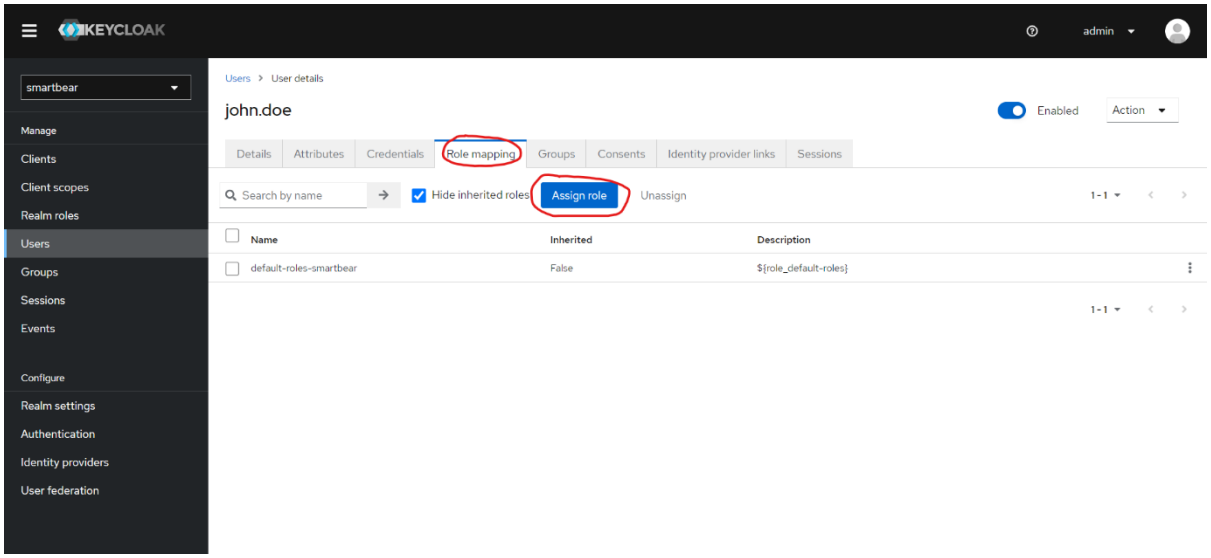


Figure 10: KeyCloak assign role to user.

Ensure that the option “filter by realm roles”is selected, proceed to choose the roles to be added to the user, and then click on the ‘assign’ button (Figure 11).

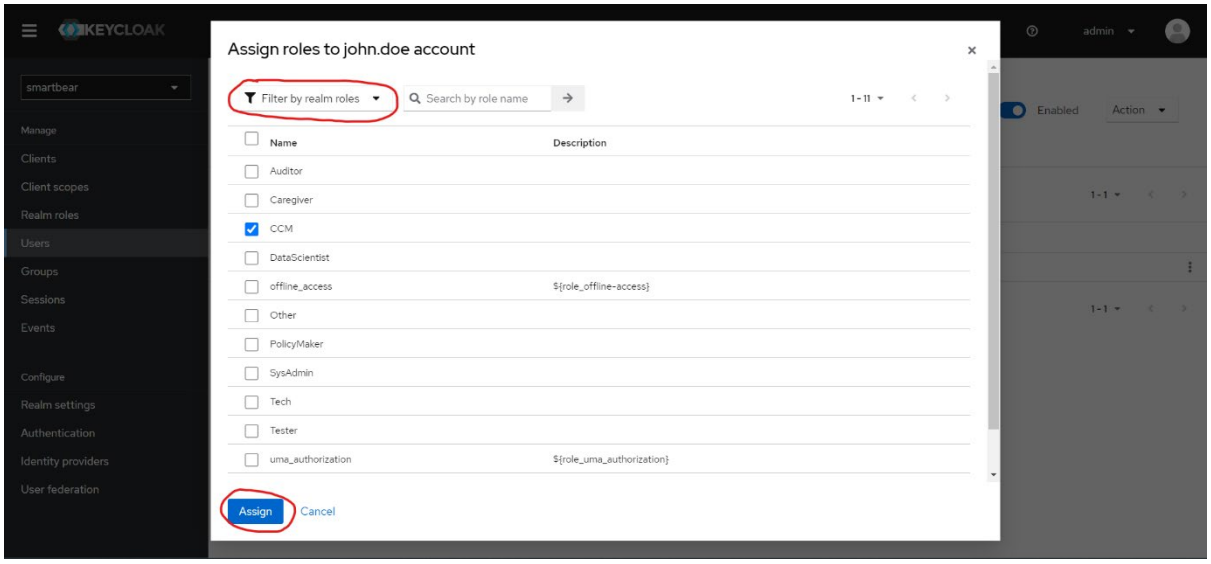


Figure 11: KeyCloak select role to assign.

If the role was successfully assigned, you should be able to view it on the role ‘mapping tab’ (Figure 12).

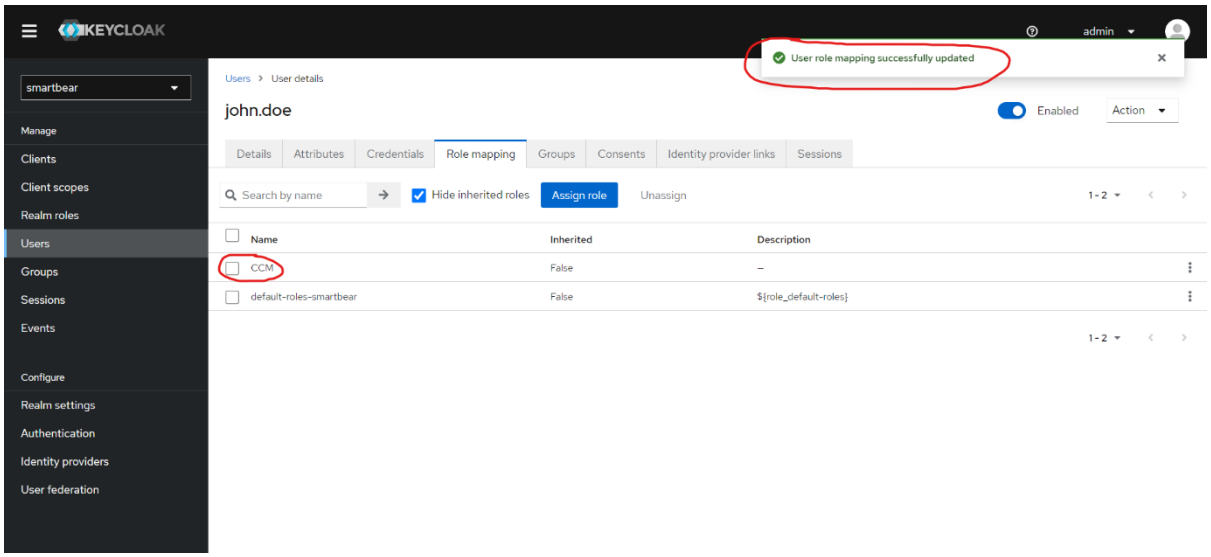


Figure 12: Confirmation of role assignment.

3.1.2.3 KeyCloak reset credentials and send email to user

Keycloak offers the ability to send emails to the user. Navigateto the 'Credentials' tab of the user, then click on 'credential reset'(Figure 13).

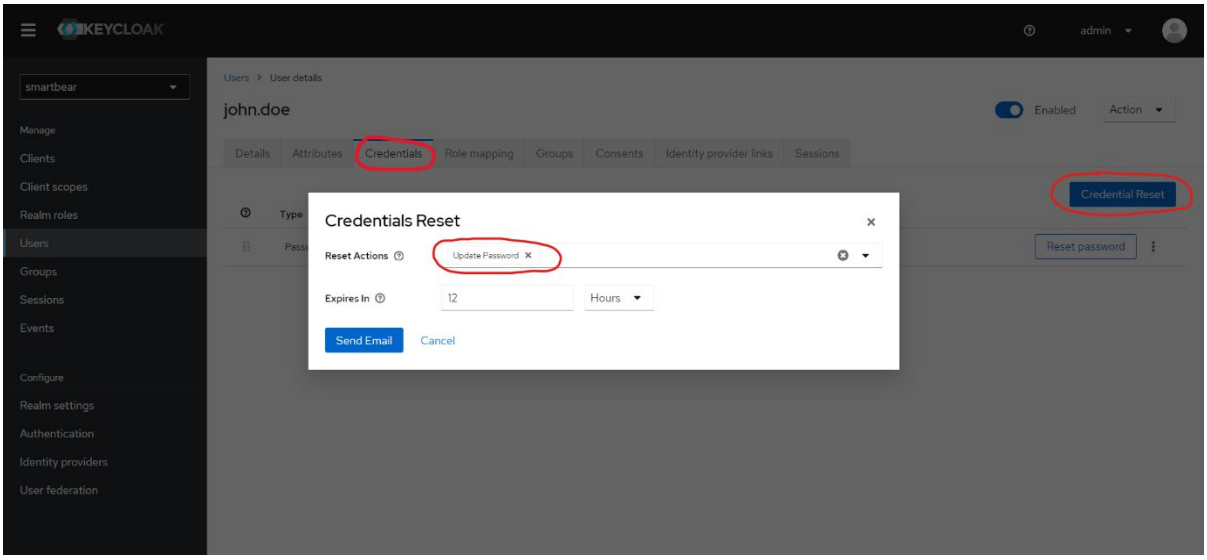


Figure 13: KeyCloak credentials reset

Select the "reset actions" from the dropdown menu. In this case 'update password' has been selected and click 'send email' (Figure 14).

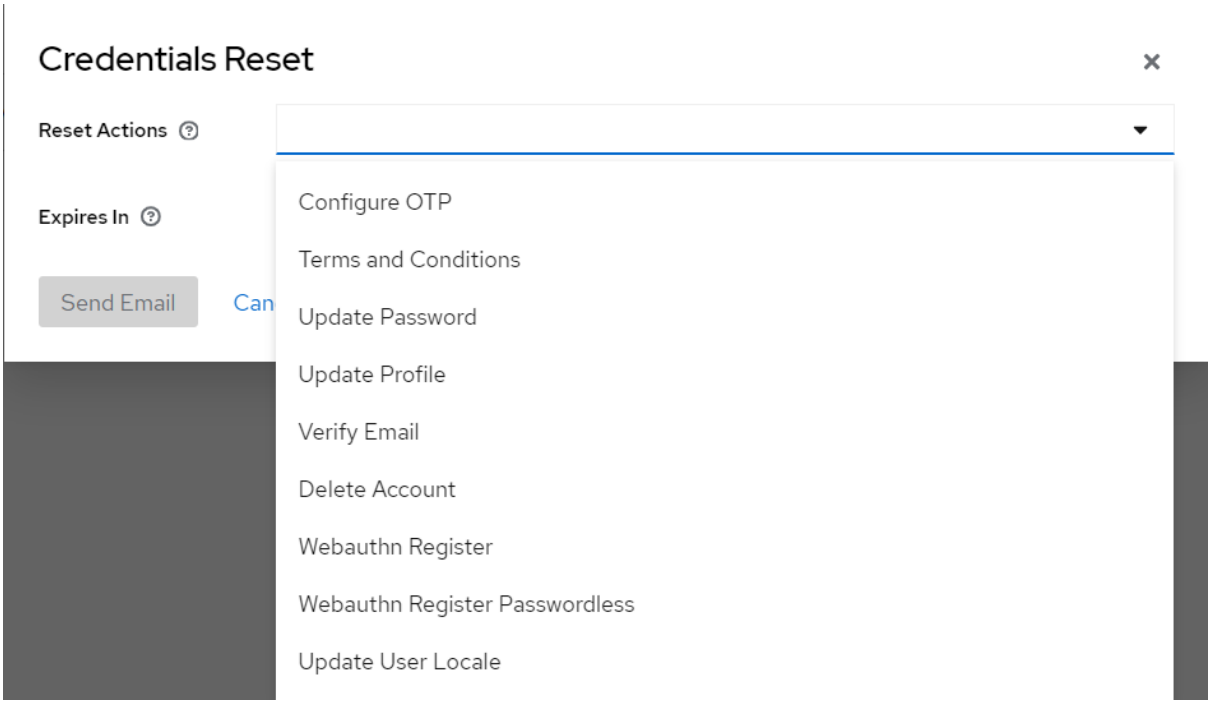
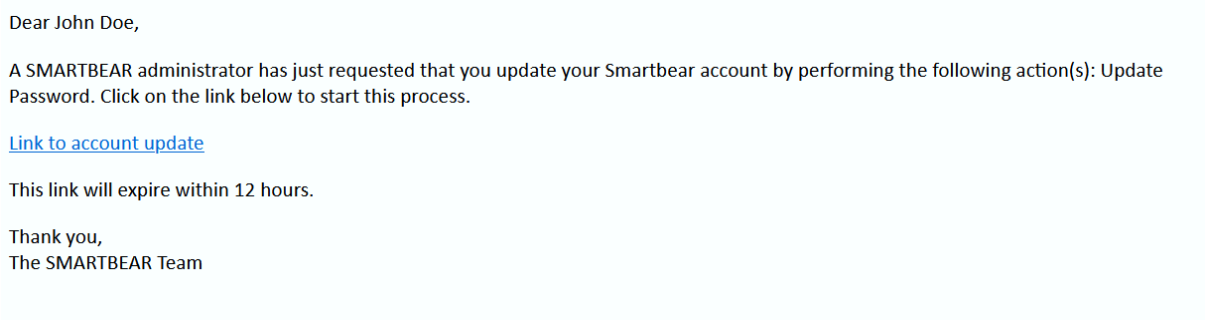


Figure 14: KeyCloak update password.

Upon completion of the previous step, an email containing a link will be sent to the user. Upon clicking the link, the user will be directed to a prompt where they can proceed to change their password as it is shown in Figure 15.


Dear John Doe,

A SMARTBEAR administrator has just requested that you update your Smartbear account by performing the following action(s): Update Password. Click on the link below to start this process.

Link to account update

This link will expire within 12 hours.

Thank you,
The SMARTBEAR Team

Figure 15: KeyCloak sample email for password update.

3.1.2.4 KeyCloak clients

For a user to login, however, one needs to create the appropriate clients. Clients are entities that can request Keycloak to authenticate a user; as such, the mobile application or the web dashboard are clients and need to be created. To create such clients:

1. The Keycloak administrator clicks on “Clients” in the sidebar.
2. Clicks on “Create”.
3. Inputs a client id (e.g. “dashboard” or “mobile”) and clicks on “Save”.
4. Additional settings may be configured. For example, Access Token lifespan, or Access Type.

Now, the exact realm and client combination can be used to integrate Keycloak’s provided login page to any GUI.

3.1.2.5 KeyCloak roles

The following images (Figure 16 and Figure 17) depict the process by which end-user roles are created. To create a role, the system administrator has to follow the following steps:

1. Log in at the Keycloak administration dashboard;
2. From the menu (on the left) select “Roles” and then “Add Role”;
3. Insert the role’s name and optionally a description;
4. Repeat steps 2 and 3 until all roles are created (Figure 18).

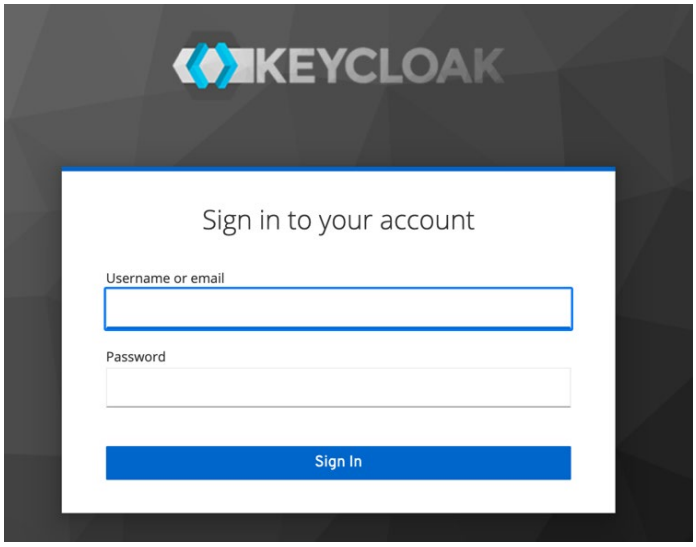


Figure 16: Step 1/3: Keycloak login.

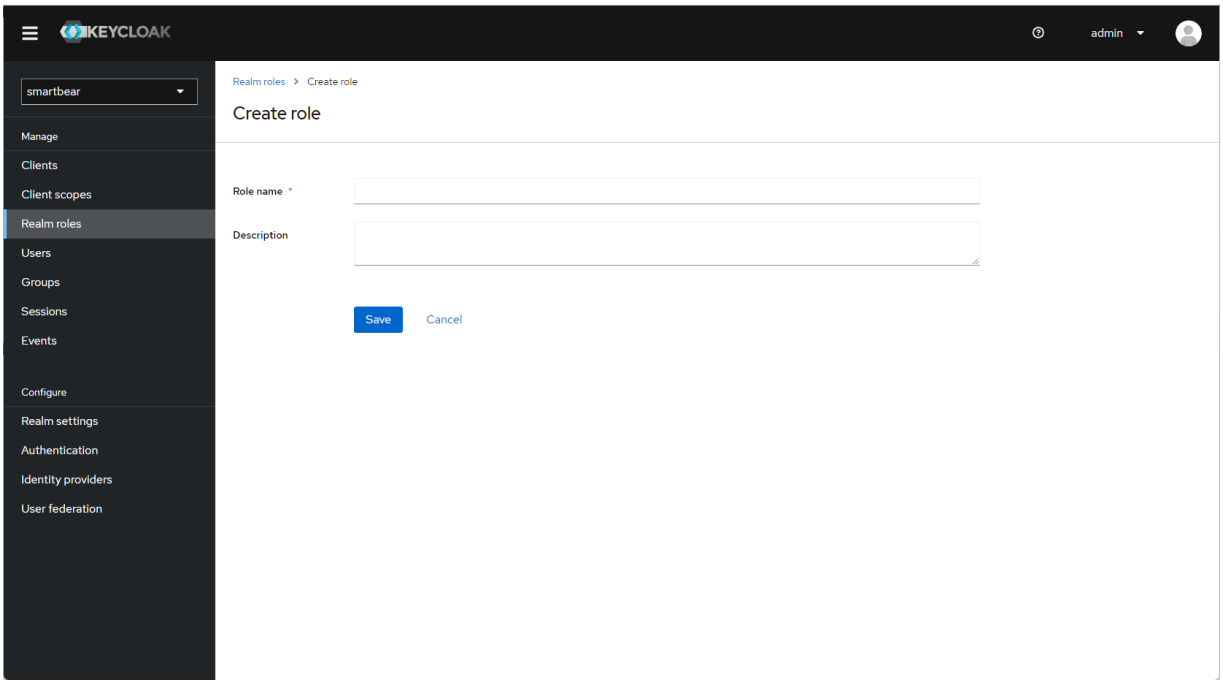


Figure 17: Step 2/3: Create Role

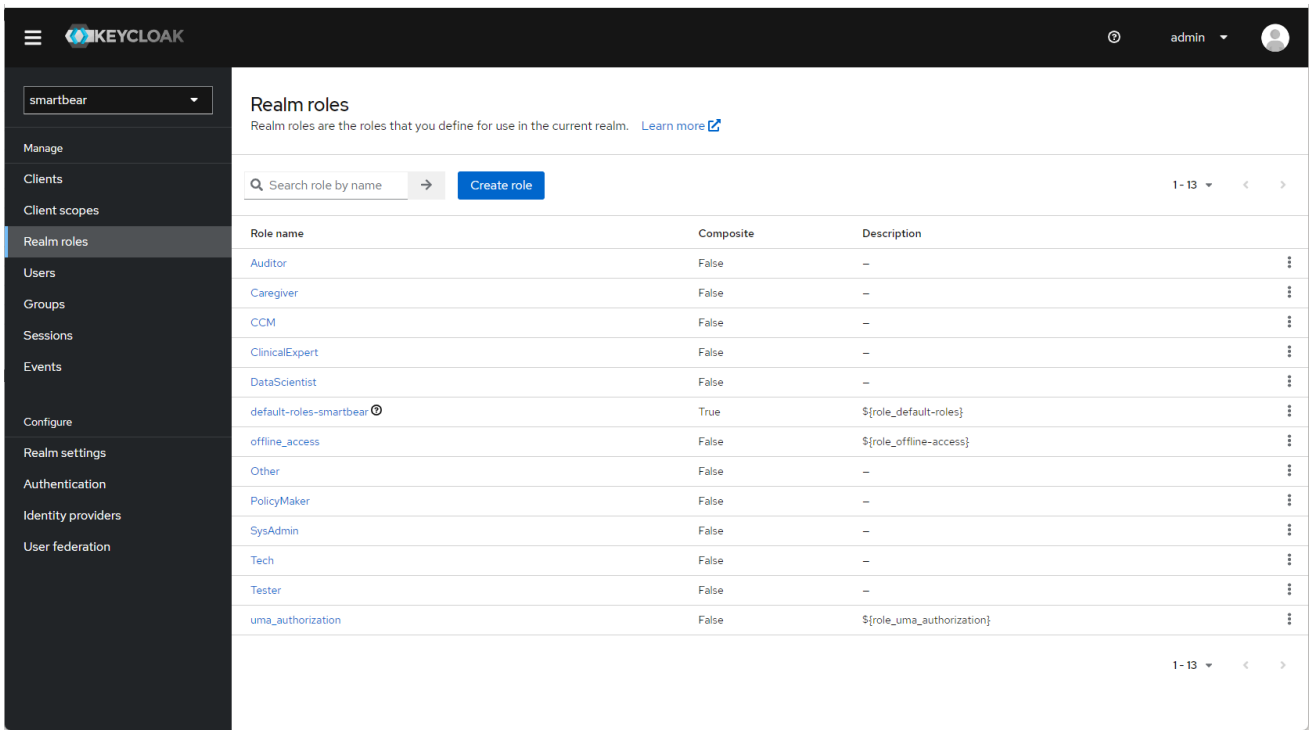


Figure 18: Step 3/3: Role added.

3.1.3 SB@Dashboard End-User Management

The end-user management of SB@Dashboard has not been affected by the transition of the API mechanism from WSO2 to Keycloak. The following figures (Figure 19, Figure 20, Figure 21, Figure 22) provide an overview of the procedures for the end-user management from the dashboard.

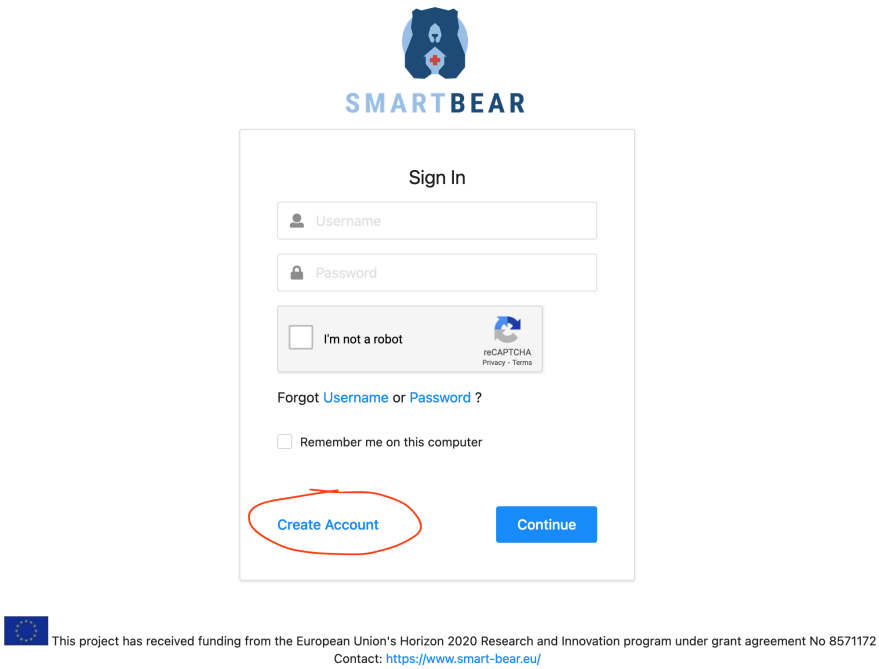


Figure 19: Login page of the SmartBear platform. Unregistered users can use "Create Account" to sign up.

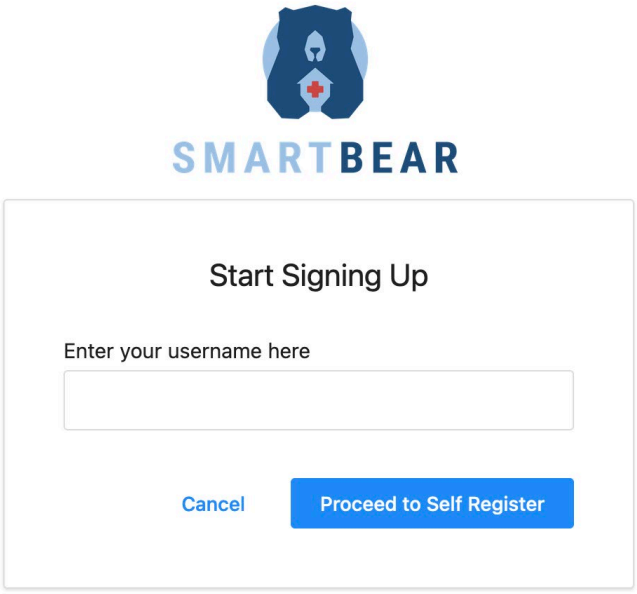


Figure 20: The first page following the "Create Account" option. The user needs to set their username. If the username already exists, they are prompted to use a different one before they proceed.

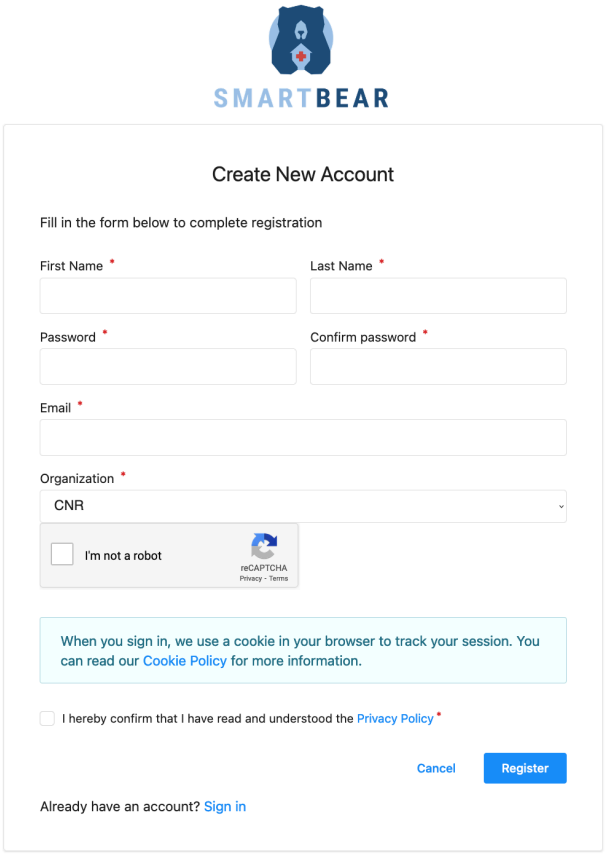


Figure 21: Further details the user needs to add to proceed with the self-sign up.

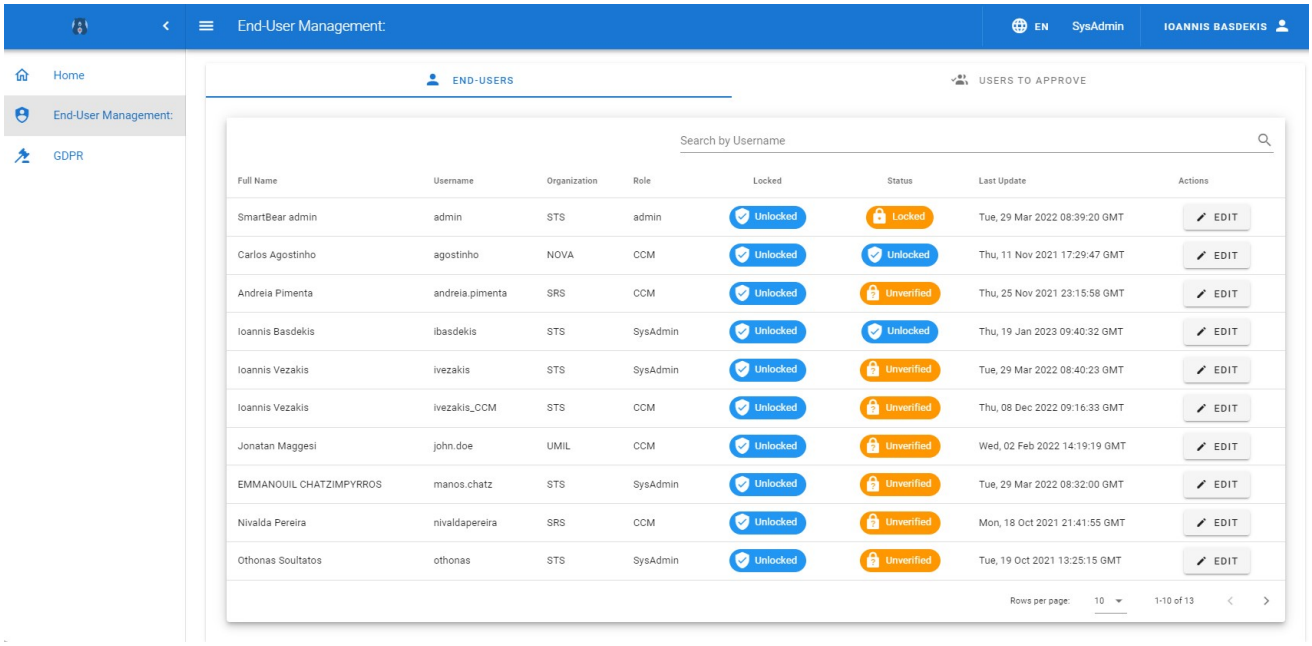


Figure 22: SB@Dashboard displays all registered end-users of the platform. Here, the admin can approve or reject new sign-ups.

3.2 Patients' management

The functionality of the patient's management has been reported in D5.3. Transition from WSO2 to Keycloak does not affect the user interface and the functionality of the patients' management system in the SB@Dashboard.

4 Security and Privacy Assurance Platform

The SB platform's real-time operational compliance with the security requirements is realized through integration with the SB Security and Privacy Assurance Platform (SPAP). This integration facilitates security and privacy assessments using penetration testing and continuous monitoring of SB@Cloud assets. The SPAP, designed to monitor various SB platform components, operates as a model-driven platform capable of conducting hybrid security and privacy assessments. Adapted from the Assurance Platform of STS, this platform has been deployed and customized into the SB platform to enable real-time, continuous assessment of security and privacy posture. Its purpose is to support management procedures for securing IT infrastructure from a technical standpoint, tailored specifically to the healthcare ecosystem.

The SPAP serves as a comprehensive security assurance platform, offering continuous monitoring for SB@Cloud assets. With its capabilities, SPAP can accommodate various types of assessments and allows security experts to develop new and customized assessments dynamically continuously or on demand for a specific period. The management of the platform is overseen by STS. It's crucial to emphasize that SPAP doesn't function as an incident response tool for SB, it serves primarily as a monitoring tool to ensure the smooth operation of the SB platform. In the event of anomaly detection, STS will promptly notify technical partners within 24 hours to conduct further investigation into the incident.

Pending action: Currently, the SPAP provides support for a range of availability assessments concerning critical components of SB and conducts IP profiling for the SB production environment. Furthermore, additional assessments, including GDPR and User and Entity Behavior Analytics (UEBA), are scheduled for integration in the near future and will be documented in the forthcoming deliverable (D5.7).

This section documents the creation of an availability assessment for the SB@Dashboard. Other assessments can be created on demand by the security experts with access to the SPAP platform.

The SPAP has its own authentication mechanism since it is independent from the SB tools that support the clinical trials. The user has to login to the system as shown in Figure 23.

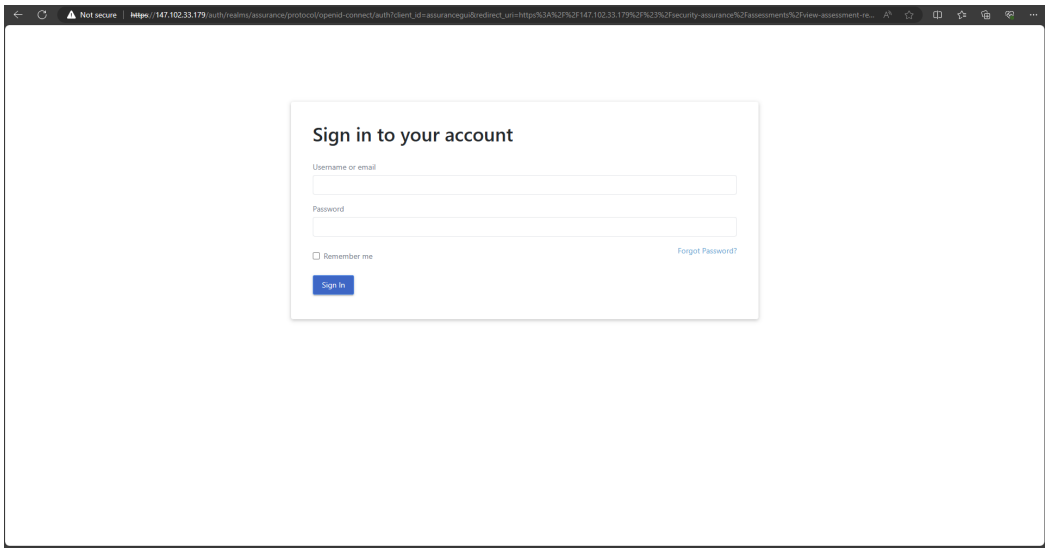


Figure 23: SPAP login.

From the main menu, the user selects Security Assurance, then Assessments and Initiate Assessments as shown in Figure 24. The user has to provide the asset's URL. In our case, the assessment point to the SB dashboard (https://cloud.smart-bear.eu/) for the availability assessment profile. For a continuous assessment, the execution type must be set to "continually".

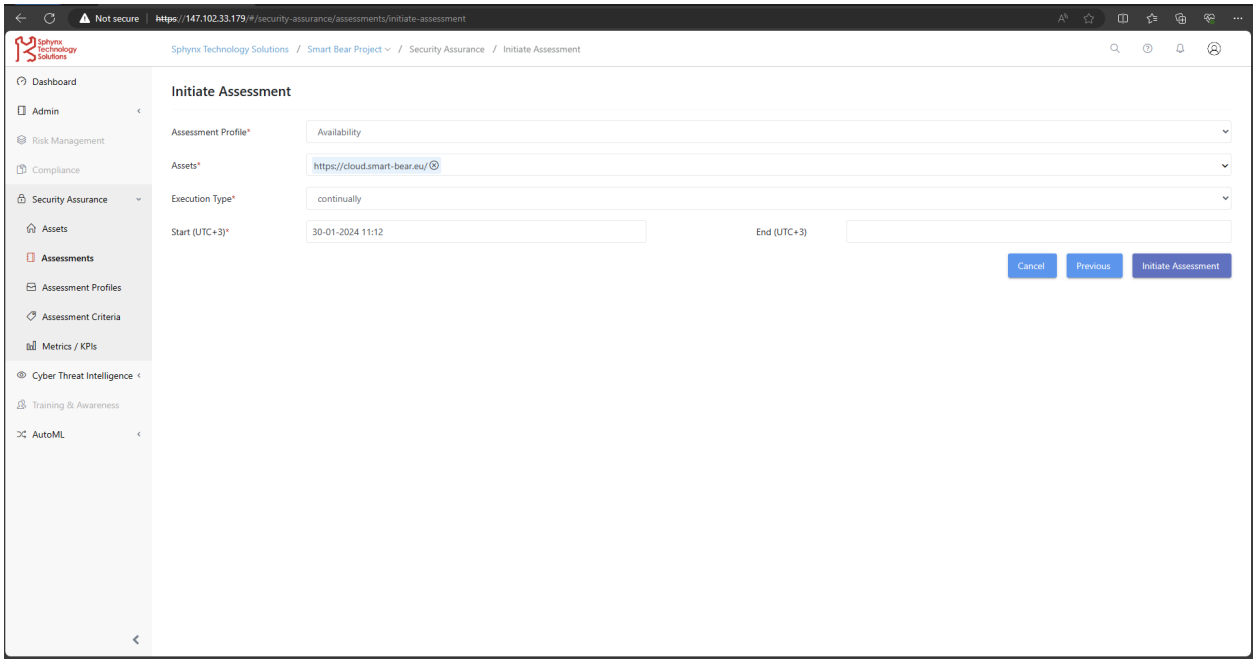


Figure 24: SPAP Initiate Assessment

Upon successful completion of the assessment, the system returns a confirmation message (Figure 25) and the assessment starts.

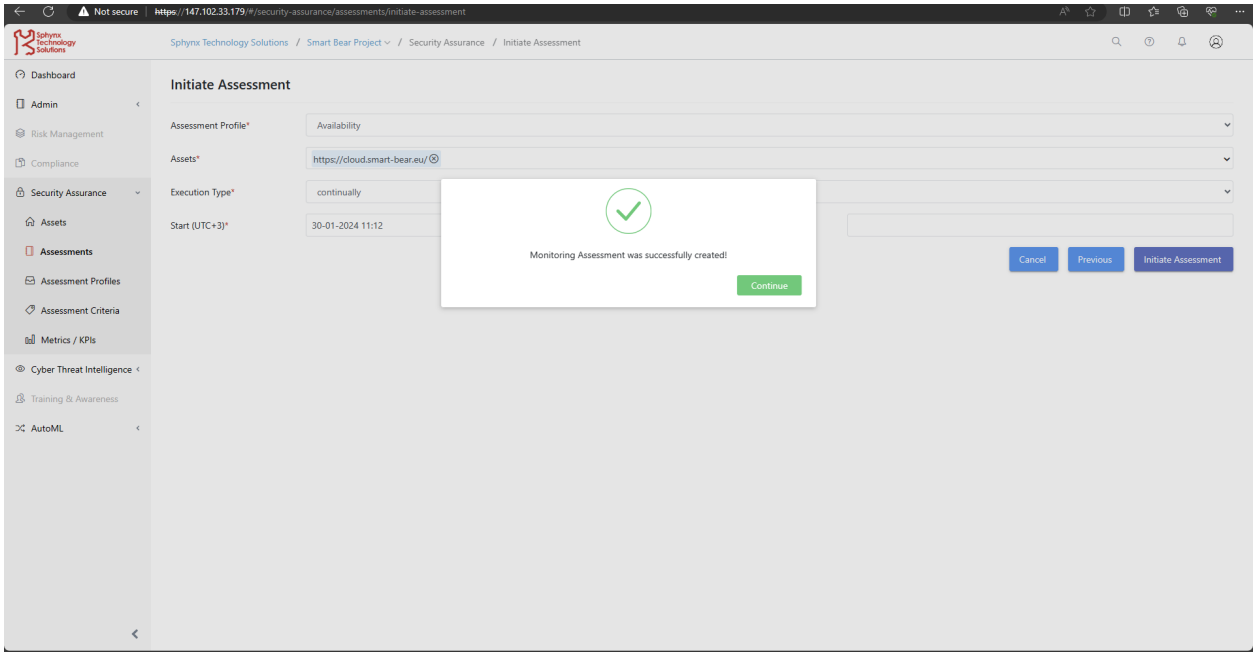


Figure 25: SPAP Initiate Assessment success confirmation.

The security expert can monitor the status of the assessments from the Assessment Results dashboard as shown in Figure 26. Furthermore, the security expert can view more details about the assessment results such as the criterion and details about the assessment as shown in Figure 27.

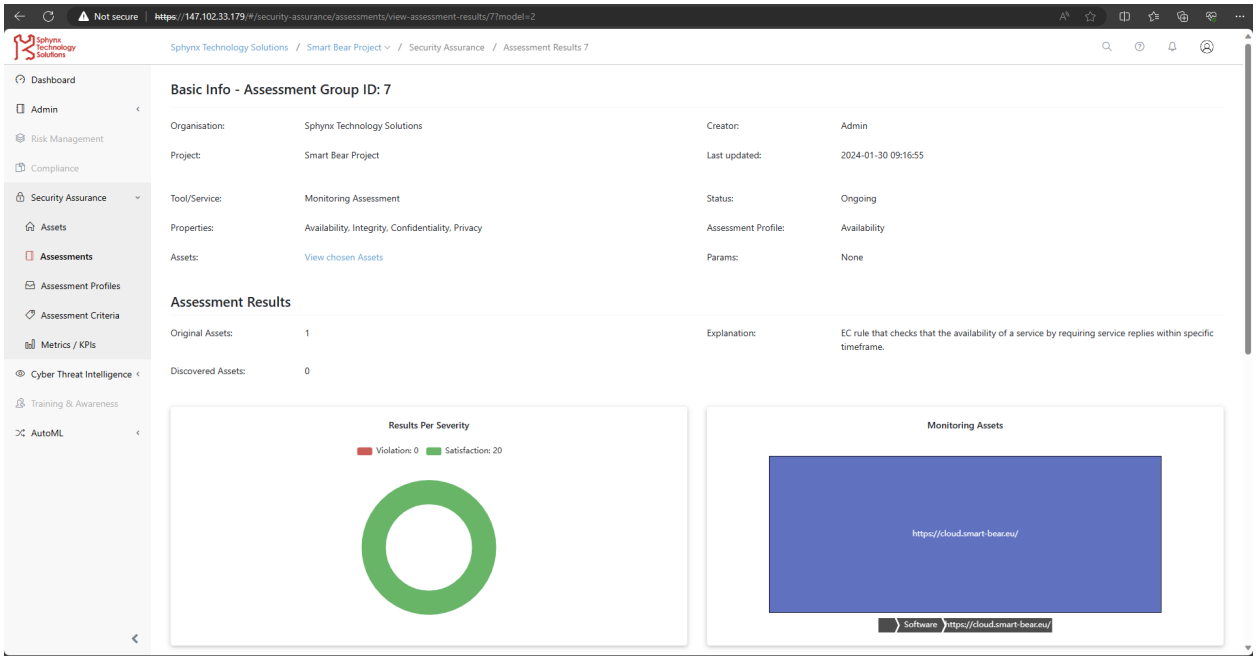


Figure 26: SPAP Assessment results dashboard.

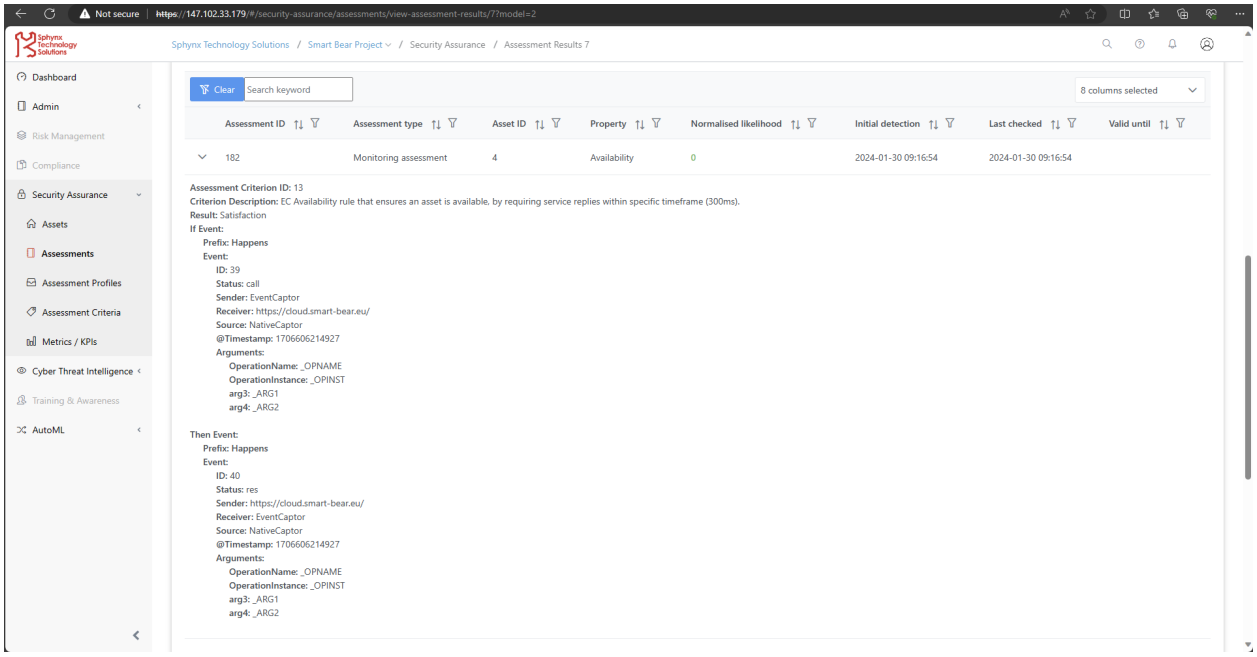


Figure 27: SPAP Assessment results criterion.

5 Conclusion

This document provides an updated version of the privacy and security mechanisms implemented within the SB@Cloud platform, focusing specifically on the third iteration of the SB@SecurityComponent. It highlights the progress and effort undertaken by the consortium to meet the project objectives related to the corresponding component (as defined in T5.1). The enhancements and additional functionality integrated into the SB@SecurityComponent have been informed by the updated requirements derived from extensive evaluations conducted at the pilot sites. Furthermore, this document includes an illustration of the SPAP platform and the introduction of a new assessment.