EUSEC

EU SECURITY CERTIFICATION

EUROPEAN SECURITY CERTIFICATION FRAMEWORK

TECHNICAL REPORT ON PILOT INTEGRATION FOR PROVIDER SELECTION AND CONTINUOUS CERTIFICATION

DRAFT

PROJECT NUMBER: 731845

PROJECT TITLE: EU-SEC

AUTHOR:
RAMON MARTIN DE POZUELO,
MARIO MAAWAD.

PARTNERS CONTRIBUTED:
CAIXA, CSA, FRAUNHOFER, SIXSQ, NIXU,
FABASOFT.

*PU = Public, CO = Confidential

**R = Report, P = Prototype, D = Demonstrator, O = Other

This project has received funding from
the European Union's HORIZON Framework
Program for research, technological development and
demonstration under grant agreement no 731845

EXECUTIVE SUMMARY

This document is part of WP5 in the EU-SEC project.

First, it presents the development and integration of the technical architecture for the EU-SEC Continuous Auditing-based Certification (CAC) pilot. More concretely, the picture of the architecture for the pilot deployment is shown, the different modules of the architecture are summarized and the functions of the EU-SEC CAC API detailed. It also specifies the use cases tested in the pilot in order to validate the proposed CAC approach. In particular, the use case that motivates the EU-SEC pilot in the financial sector project is "financial information sharing" (FISH). That is, the management and exchange of sensitive documents among financial institutions (e.g., banks, insurance companies) and regulatory authorities, which is becoming increasingly relevant in the recent years. The objective of this pilot is to allow us to perform continuous auditing of a financial information sharing application in the Cloud to simplify life to involved parties, while having guarantees that the Cloud provider continuously meets with the requirements to run such a service. Based on this use case, the pilot is tested, taking into account the technical architecture security recommendations and the validation of the different modules.

Disclaimer: The information and views set out in this publication are those of the author(s) and do not necessarily reflect the official opinion of the European Communities. Neither the European Union institutions and bodies nor any person acting on their behalf may be held responsible for the use which may be made of the information contained therein.

ABBREVIATIONS

Abbreviation	Description
API	Application Programming Interface
AWS	Amazon Web Services
CAC	Continuous Auditing-based Certification
CCM	Cloud Control Matrix
CIMI	Cloud Infrastructure Management Interface
CSP	Cloud Service Provider
FISH	Financial Information Sharing
GDPR	General Data Protection Regulation
ISO	International Organization for Standardization
PCI DSS	Payment Card Industry Data Security Standard
SLO	Service Level Objective
SQO	Service Quality Objective
URL	Uniform Resource Locator

TABLE OF CONTENTS

LIST OF TABLES

LIST OF FIGURES

1 INTRODUCTION

This deliverable continues the work done in EU-SEC deliverable *D5.1 – Pilot Definition*, summarizing the architecture deployed for the Continuous Auditing-based Certification (CAC) pilot, and detailing technical validation of the use cases under testing.

The use cases for testing the CAC approach proposed in EU-SEC focus on the financial sector. This is a very regulated sector and still reluctant to move their core services to the Cloud. For this reason, and taking into account the participation of CaixaBank in the EU-SEC project, the pilot showcases the deployment of a Financial Information Sharing (FISH) application in the cloud, allowing financial institutions such as banks, insurance companies, regulators, etc. to exchange information in a secured but agile way. Hence, this pilot represents a proof-of-concept example of the CAC approach and reference architecture, testing it with a highly demanding case of the bank sector.

1.1 OBJECTIVES AND SCOPE

The objective of this deliverable is to report about the deployment of EU-SEC CAC pilot. More concretely, it aims at detailing the final deployment of the pilot, updating the definition specified in D5.1, validating and evaluating the framework and reference architecture developed in EU-SEC for CAC.

To achieve this objective, the deliverable reports the outputs from tools integration, pilot deployment and technical testing phases of the WP5 (Figure 1-1). Further analysis of the pilot results and the evaluation of it from the business perspective will be provided in subsequent WP5 deliverable *D5.3 - Requirements and validation criteria – Pilot results*.

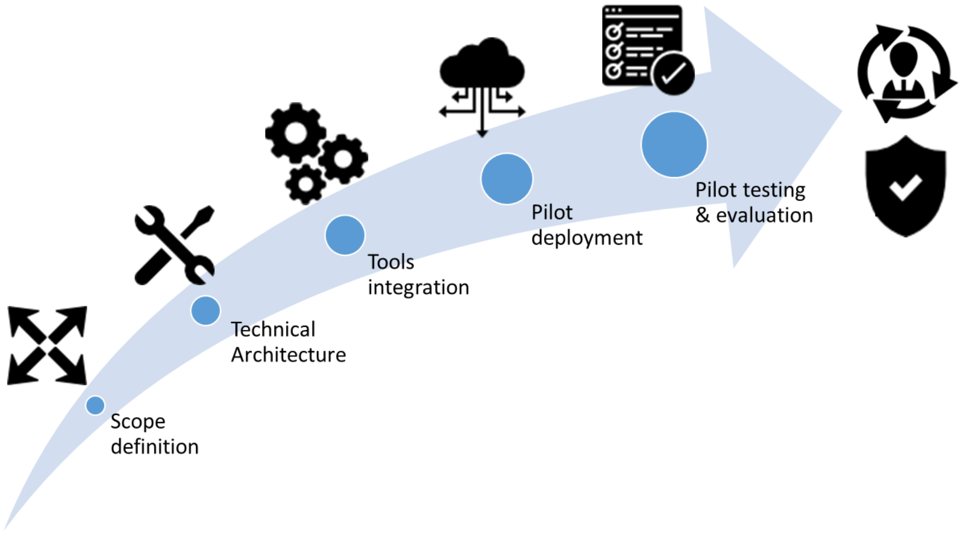


Figure 1-1 WP5 Roadmap: Pilot phases.

1.2 ORGANIZATION OF THIS WORK

The rest of the document is organized with the following structure. In Section 2 the pilot implementation is described, summarizing the configurations and integration of the technical architecture several modules and API at the time of the pilot deployment. Section 3 details the pilot scenario and use cases, as well as explaining how the different high-level requirements defined in D5.1 map to specific SQOs/SLOs (Service Qualitative Objectives / Service Level Objectives). In Section 4, we provide the pilot testing, divided three subsections: (i) the security recommendations for the EU-SEC reference architecture and pilot deployed, (ii) use case testing, (iii) Non-functional aspects evaluation. Finally, we conclude in Section 5.

1.3 WORKPACKAGE DEPENDENCIES

In the following, we describe the dependencies of deliverable with other work packages of this project. This deliverable depends on the definition of the technical architecture (e.g. continuous audit, evidence storage tools) and therefore is strongly linked to all WP3 tasks and deliverables. Moreover, the CAC approach followed in the pilot relies on the certification scheme defined in T2.2.

2 PILOT IMPLEMENTATION

This section summarizes the implementation, features and roles of each module of the EU-SEC CAC technical architecture. It also describes the need of a standardized API in the cloud application interaction with the rest of the architecture and enumerate the list functions used in the execution of the pilot.

2.1 CONTINUOUS AUDITING-BASED CERTIFICATION ARCHITECTURE

As further detailed in deliverables *D5.1 – Pilot Definition* and *D3.5 – Architecture and Tools Integration Framework,* the technical architecture proposed in EU-SEC is composed by several modules. The final picture of the reference architecture that is used in the pilot deployment is shown in Figure 2-1.

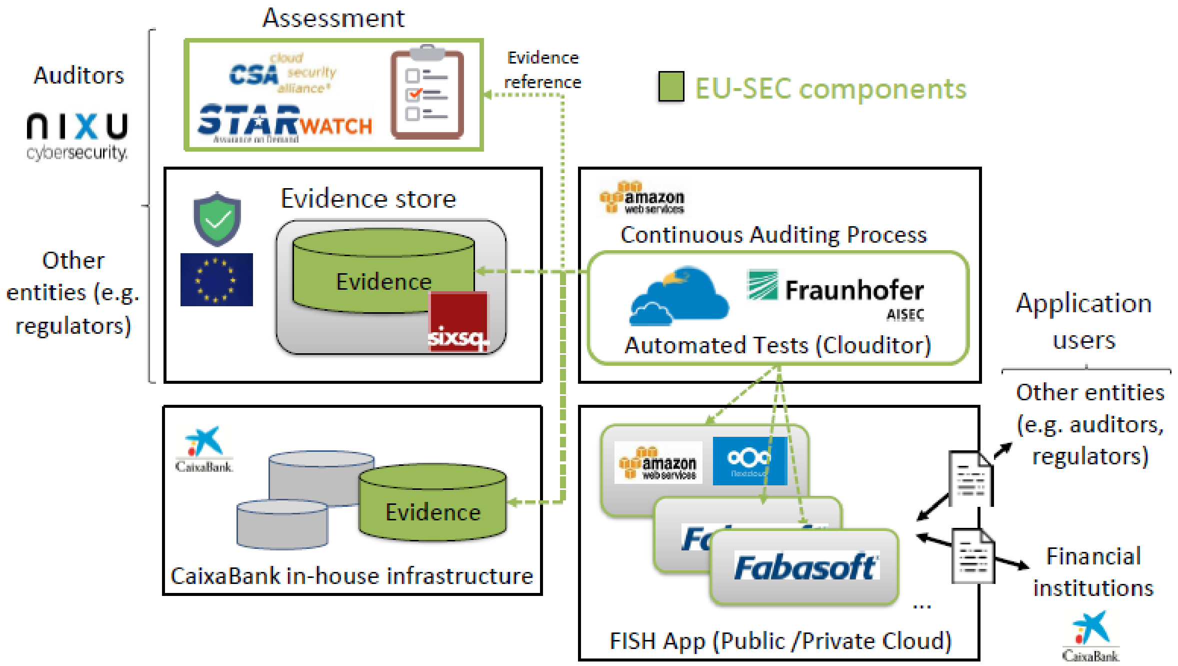


Figure 2-1 EU-SEC Continuous Auditing-based Certification technical architecture.

The rest of this section describes briefly these modules and how are they configured, integrated and used in the pilot.

2.1.1 MODULES AND TOOLS

FISH APPLICATION

In this pilot, a Financial Information SHaring cloud service (FISH) is the subject of certification. Two different approaches have been tested in the deployment of the pilot. In the first approach, a FISH application has been deployed over an AWS (IaaS). In the second approach, a FISH application has been provided based on Fabasoft cloud services, illustrating a SaaS deployment.

IaaS approach

This service is implemented on top of the open-source project Nextcloud and is deployed on Amazon AWS. It allows for regulators to request information by opening up a case with a bank. In a following chat, messages and documents can be exchanged. All required security means that are in the scope of this pilot have been implemented using features of AWS or in the application itself, often both. The EU-SEC CAC API is implemented for this service and delivers evidence records from the AWS services as well as from the application layer.

SaaS approach

Modern cloud services often provide multiple environments, to support different phases of the development process, such as testing, staging and production. Fabasoft follows this approach by providing multiple environments for development/testing and for production usage. The environments for development/testing are called Fabasoft VDE (Virtual Development Environment). The production environment is the Fabasoft Cloud. For the purposes of the EU-SEC project, Fabasoft provided a dedicated FISH VDE (dev4/vm114)[1] for pilot testing & development purposes. The EU-SEC CAC API is also implemented by Fabasoft, validating the usage of the API with both implementation/ deployment approaches.

CLOUDITOR

The Clouditor toolbox[2] consists of five main components which are shown in Figure 2-2. It can be used to design and execute continuous test-based assurance tests. The test results serve as input to compute test metrics which, in turn, can be used as evidence to support validation of controls.

[1] Dedicated VDE URL : `https://vde.fabasoft.com/dev4/vm114/folio`.

[2] For a comprehensive introduction to the Clouditor Toolbox see https://www.aisec.fraunhofer.de/content/dam/aisec/Dokumente/Publikationen/Studien_TechReports/englisch/Whitepaper_Clouditor_Feb2017.pdf.

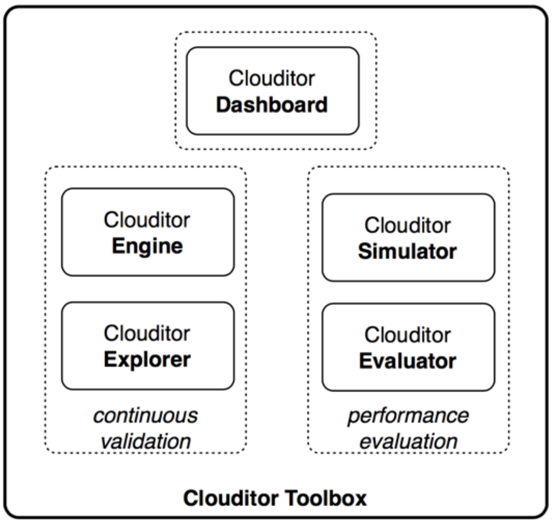


Figure 2-2: Tools of the Clouditor Ecosystem

The Clouditor Engine implements and deploys test-based assurance tests. Discovering a cloud-based application's interfaces and configuring the selected assurance technique is the task of the *Clouditor Explorer*. To that end, the Explorer discovers cloud services' composition and interfaces at runtime as well as automatically generates and adapts test configurations.

NUVLA AS THE EVIDENCE STORE

As described in the pilot architecture, Nuvla provides the Evidence Store. Nuvla is a smart application management service for cloud, edge and hybrid environments, provided by SixSq. This tool is open-source, and its ecosystem was previously defined in in *D3.5 – Architecture and Tools Integration Framework*. Since then, Nuvla has significantly evolved, redefining the way user applications are orchestrated, thus simplifying their management in multi-cloud and edge infrastructures, while keeping the same benefits as before. The previous ecosystem was composed of SlipStream and Nuvla, which now have merge into a single software stack - Nuvla. Its new architecture is show in Figure 2-3.

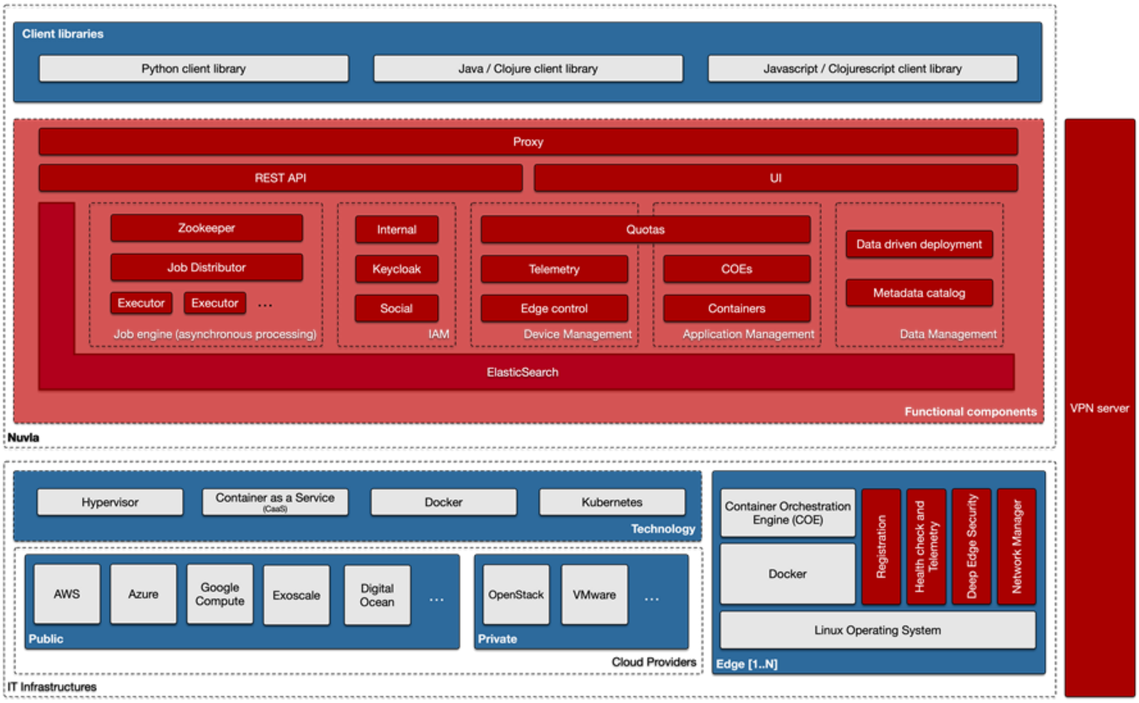


Figure 2-3 Evidence Store architecture

This new decoupled and micro-service based architecture is highly beneficial for the pilot as it allows the pilot participants to pinpoint which Nuvla components need to be installed, thus optimizing the amount of consumed computing resources, without compromising the functionality of the Evidence Store.

For the Evidence Store, Nuvla is mainly contributing with the following components:

- ***proxy:*** a reverse proxy which sits on top of the API server, with automatic generation of Let's Encrypt certificates, providing TLS encryption of data in traffic
- ***REST API:*** a CIMI compliant API server exposing a RESTful interface with rich filtering and aggregation capabilities
- ***UI:*** a simple graphical interface for the API server, which in this pilot can be seen as the audit portal, allowing users to easily manage their evidence records
- ***ElasticSearch:*** provides the storage for the evidence records and other system resources which are managed via the API
- ***IAM:*** the authentication and authorization services, which support internal login, social login and federated login (from interfederation services like eduGAIN)
- ***client libraries:*** programmatic libraries which can be used by any Nuvla user, including Clouditor.

Also, in order to facilitate the spin-off of new evidence stores, Nuvla has been re-factored and packaged as Docker images, which are public and can be deployed by anyone. This way, any party involved in the CAC process has the ability to deploy its own evidence store and request Clouditor to publish copies of the evidence record into it.

STARWATCH

STARwatch is a SaaS application created by CSA enabling users to perform self-assessment using the CAIQ, a questionnaire derived from the CCM. STARwatch has been extended to act as a repository for continuous certifications in the context of the EU-SEC project.

Principles

As described in much more detail in section 2.2 of EU-SEC *D3.5 – Architecture and Tools Integration Framework*, STARwatch acts as a "certification authority" in the context of CAC through a process that is divided in two main phases:

I. Initialization. A certification target (JSON file) is created and uploaded to STARwtach. The certification target contains the list of security objectives that the cloud service provider will aim to attain, as well as a start and end date. Each security objective is associated with an evaluation frequency: failure to confirm an objective in due time can result in the suspension of the certificate, and ultimately its revocation.

II. Iteration: STARwatch offers and API that Clouditor (or any similar tool) uses to update the status of each objective described in the certification target. STARwatch monitors these updates and maintains a public repository with the state of the certification of the relevant cloud service:

- If all objectives initially declared in the certification target are confirmed in due time, the certification of the cloud service is considered as "valid".
- On the other hand, if an objective is not confirmed in due time, the certification is considered as "suspended" until the situation is corrected.
- If the certificate remains "suspended" for too long, the certificate becomes revoked and gets removed from the public registry.

The iteration phase starts at the date indicated in the certification target and ends when either the end date is reached or when the certificate is revoked.

Changes made during the pilot

In the context of integration work conducted in order to run the pilot, a few minor changes were made to the certification target JSON format defined in EU-SEC Deliverable D3.1:

- A "description" field was added to assessments.
- Renamed "frequency" to "period" in assessments.

These changes were only for clarification purposes and had no impact on the technical architecture of the pilot.

Under the hood, the development team found some clever optimizations that facilitate certification state monitoring during the iteration phase described previously. While the description of the iteration phase seems to suggest a continuously running process that could become a bottleneck with a large number of certificates, in practice it's possible to implement a "just-in-time" approach that uses much less resources. In short, the principle of this approach is as follows:

- Collect "events" each time the updates are made.
- Calculate the certification status only when requested by a public query to the registry or by an internal request.

Extensive tests made in preparation for the pilot showed that this approach works well.

2.1.2 INTEGRATION

In EU-SEC deliverable *D3.5 – Architecture and Tools Integration Framework*, section 3 and 4, the technical integration of the architecture and the cloud services is detailed. However, some additional details related to the deployment of the pilot are explained below. More concretely how the interaction of Clouditor, Nuvla and Starwatch are finally configured in the pilot.

After deploying each tool individually, we create an account in the FISH applications, Clouditor, Nuvla and Starwatch. The account created in Nuvla allows Clouditor to publish the evidence records it generates during the continuous auditing process. Upon saving a new evidence record in the Evidence Store, Nuvla's API server will automatically generate a random unique identifier which is synchronously given back to Clouditor, so that it can append it as a tracking reference to the data which is pushed into StarWatch. Figure 2-4 describes all the interactions between the Evidence Store (Nuvla) and the remaining components in the architecture.

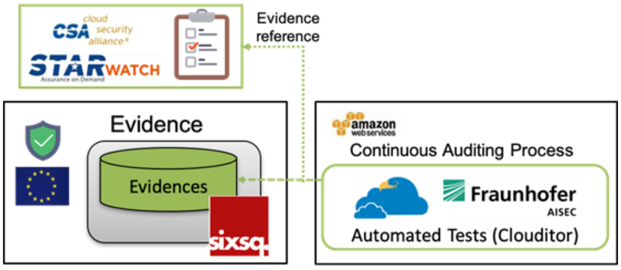


Figure 2-4 Evidence Store interactions with other modules

For an easier integration of Clouditor with the Evidence Store, the *evidence-record* resource has been given the following open schema:

```
{
    "endTime": timestamp,
    "class" : string,
    "startTime" : timestamp,
    "planID" : string,
    "passed" : boolean,
    "log": [string, string],
    "<namespace>.whatever": anything
}
```

Having an open schema gives Clouditor enough flexibility to generate and publish any evidence record, no matter how heterogeneous the tests and checks are. The <namespace> must be registered in Nuvla in advance, and once this is done, Clouditor can define attributes that are not explicitly defined in the data model, as long as they are prepended with that <namespace>.

Finally, Nuvla's API server also provides a very fine grained ACL-based authorization model for all its resources, which means it is also possible for Clouditor, as the owner of the published evidence records, to grant any kind of access level (view, edit, manage, delete, etc.) to any Nuvla user or even unregistered users, if desired. This feature is particularly useful for scenarios where Clouditor might want to share its findings with other actors in the continuous auditing process, like the auditor and/or auditee.

2.2 CONTINUOUS AUDITING-BASED CERTIFICATION API

In traditional point-in-time audits the auditor requests evidences for the claims and uses them to verify if if the requirements are fulfilled. The CAC API facilitates this action of handing over evidences in an organized way. It is the interface between the CSP and the the auditing entity; in the case of EU-SEC this is Clouditor. Clouditor asks for specific evidence by calling the defined REST-Interfaces. Each type of evidence has its own Interface, but similar type of evidence are grouped by their corresponding domain. The API-Calls are generic enough to fit for all types of cloud services by targeting general goals like encryption or identity management. It's the obligation of the cloud service provider to implement the audit API and assure that the correct evidence is provided. The core functionality of the API is to map existing live data from the IT-infrastructure, like logs, database-entries, configuration-files or 3rd party API calls to the Audit-API REST-calls. Often times this involves minor preparation of the original sources, like a log file maybe has to be loaded into a database to allow efficient searching.

The set of calls and therefore the set of possible evidence records is currently designed to fit the purpose of the EU-SEC pilot. This means that the API in its current state is not reflecting the needs of the common security standards, because currently it cannot provide evidence records to cover the automated evaluation of all the controls. Addressing the issue of completeness is beyond the scope and resources of the EU-SEC Project. That's why the API Specification is open sourced. Only a community effort can lead to a comprehensive coverage of possible evidence and the API calls to retrieve them. Table 2-1 features all REST-Resquests used in the Pilot.

Table 2-1 API functions used in the pilot

API ENDPOINT	DESCRIPTION
/scopes/	Cloud services are realised by different technologies often arranged in architectural layers and scopes. This call return all scopes used by the service.
/{scope}/objects/	Returns Object ids of all objects that are in the scope of the audit.
/{scope}/persistence/{objectId}/storage/	Returns persistence information for a particular data object by its Id.
/{scope}/persistence/{objectId}/location/	Returns location the ISO 3166-1 alpha-2 country code of the location of the data of the object.
/{scope}/persistence/{objectId}/encryption	Retrieves the encryption info of an object.
/{scope}/identityfederation/admins/	Returns a list of administrators.
/{scope}/identityfederation/{userId}/groups	Returns the groups of a user.
/{scope}/identityfederation/{userId}/auth	Returns the authentication type of a user. E.g password, two-factor.
/{scope}/identityfederation/{userId}/logins	Returns a list of the last logins of a user.
/{scope}/identityfederation/data/access	Checks whether a user has a certain access to an object.
/{scope}/identityfederation/{userId}/passwordRequirements	Returns the password requirements for a specific user.
/{scope}/meta/submitted	Gives information on when certain documents have been pushed to dedicated endpoints of the customer.

3 PILOT DEPLOYMENT AND USE CASES

This section briefly explains the pilot use cases rationale and details the use cases scenario, actors and interactions. It also includes the analysis of the pilot high-level requirements defined in EU-SEC deliverable *D5.1 – Pilot Definition* and the mapping to more concrete SQOs and SLOs that can be technically and autonomously checked by the tools of the technical architecture.

3.1 FINANCIAL INFORMATION SHARING USE CASES

For the sake of testing the reference architecture and the CAC approach described in previous section, a sample scenario was defined in *D5.1 – Pilot Definition*. In that sense, EU-SEC partners together with the External Advisory Board found a case that was especially interesting for the financial sector. The increasing need for exchanging sensitive financial information between entities is leading to bad practices, unfriendly or unsecure proprietary tools or/and arduous tasks for sharing documentation. The possibility to find ways to share information in the cloud in a secure and friendly way is moving this type of services to the cloud. However, how can a user from those entities assess that the cloud service provider (CSP) offering this service/application is achieving the restrictive requirements that they need to comply with. Indeed, how can the user assess that those requirements are not achieved only at the moment of a point-in-time audit, but they are accomplishing the required levels of security at all the time. EU-SEC CAC provides a methodology and a reference implementation of a technical architecture to configure and automate the audit and certification processes in order to certify periodically (every hour, day, week, month, etc.) if the application is compliant with the specified requirements. This approach applies to other sectors and cases, and it is also interesting when there is no dependency on an external CSP, where one entity is providing a service or application to another entity, and a high level of assurance is needed.

The use cases cover the main features that EU-SEC CAC can offer and is highly representative of a need in the financial sector (although they could be easily generalized to other sectors or situations). More concretely, the use cases detailed below emulate a general current-day situation in which a regulator is periodically asking a bank to report about the evolution of their security and privacy internal projects, or if they want to collect more information about any specific incident related to the bank.

3.1.1 SCENARIO

The pilot consists in two similar but conceptually different scenarios (Figure 3-1) in which the EU-SEC reference architecture is tested. In the first one, the service under test is a custom-tailored FISH application that was built as a proof-of-concept over a IaaS (Amazon AWS). The application was designed to be simple while still offering the main functionalities needed in the sensitive information exchange between the user and the regulator. In the second case, we test the certification of a FISH SaaS application that allows more advanced actions between users. As an example, this second use case approach shows not only involves bilateral communication but multiparty interactions and document management.

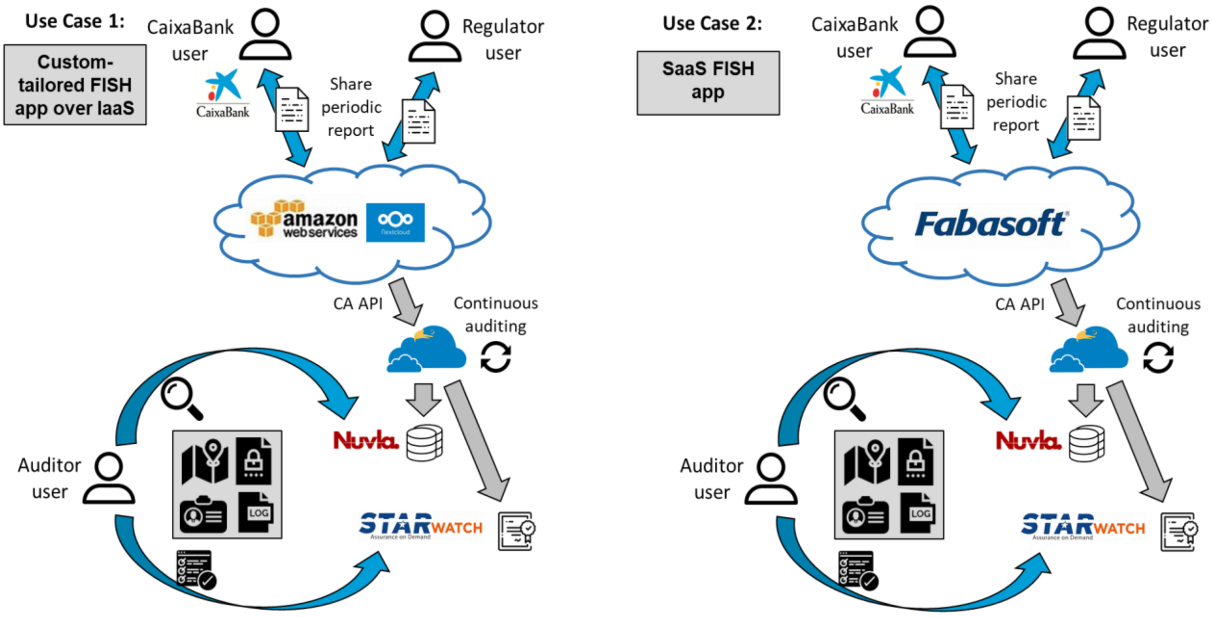


Figure 3-1 Use cases scenario

3.1.2 USE CASE DEFINITION

The following tables describe the different steps executed by the use cases' actors and modules.

USE CASE 1: CUSTOM-TAILORED FISH APP OVER IAAS

Table 3-1 Use Case 1 definition

Use Case Identification and History			
Use Case ID:	EU-SEC.FISH.IAAS.v1		
Use Case Name:	Financial Information reporting to regulator - FISH app approach over IAAS.	Version No:	1.1
End Objective:	To validate the CAC framework proposed and reference architecture developed in EU-SEC project, testing it with the assessment of a sensitive information sharing application in the cloud. More concretely, the application is a proof-of-concept built for Financial Information Sharing (FISH) and will be tested emulating the scenario in which a financial institution has to report to the regulator entity about an specific case or security project in development at the institution. It should validate the CAC assessment that all the requirements specified by the financial institution (CaixaBank) are accomplished in the exchange of information between the entity and a regulator that has claimed for a report about certain operations of the bank.		
Created by:	CAIXA	On (date):	01/09/2018
Last Update by:	CAIXA	On (date):	25/02/2019
Users/Actors:	• **Financial Institution (FI):** An employee of a financial institution dedicated to the financial sector. • **Regulator User (R):** An employee of the entity that regulates the financial sector. • **Cloud Service Provider (CSP):** Cloud provider and FISH application provider. • **Auditor (AU):** Auditor or an external entity employee that is in charge of validating the performance and trustworthiness of the FISH application in the cloud.		
Trigger:	Regulator wants to claim specific sensitive financial information from a financial institution (e.g. periodic reporting of the security projects that are being implemented in the institution).		
Frequency of Use:	Weekly/Bi-weekly.		

Preconditions
- EU-SEC architecture, all the components and the FISH application are already deployed. That means, among other aspects, that: ○ **AU** continuous assessement has already created and configured in Starwatch. ○ **CSP** and **A** configured Clouditor checking the results of controls of the FISH application and sending collected evidence to a trusted external evidence store, to the FI on-premise store and a reference of the evidence record to Starwatch v2. Clouditor is configured to continuously check the configured controls, particularly to ensure that **FI** requirements are achieved (location, encryption, user management, etc.). ○ All the information from the continuous auditing has to be stored as logs and evidence, particularly when there are indicators of non-compliance with the requirements. - **R** and **FI** have their own credentials to access the FISH app, provided by **CSP**. Strong authentication mechanisms are provided by **CSP**. - Cryptographic key management policies and procedures are already defined by **CSP** and AU. Asymetric cryptography with private and public keys RSA-2048 algorithm will be set up in the exchange of messages and document between the two parties. Since Files are usually too big for the browsers internal RSA implementation AES is used for the encryption. The AES-CBC 256 key as well as the initialisation vector are getting generated newly for each file. The cryptographic material for files is being exchanged in the same way as the messages and RSA encrypted.

Basic Flow		
Step	User Actions	System Actions
1	**R** logs into the FISH Application continuously audited by the EU-SEC platform using its own credentials (two-factor authentication).	FISH application shows a LOGON interface and authenticates the credentials of **R**. Clouditor monitors the access of **R**.
2	**R** wants to ask for specific information from **FI**. Selects to open a new case.	FISH application starts showing the initial window, allowing **R** to select a former case or open a new case with a specific group of users.
3	**R** creates a new pair of keys (private and public) for the communication of this case with **FI**.	FISH application opens a window for the generation of the new pair of keys, and they are automatically created when the user presses the “generate” button.

Basic Flow		
Step	User Actions	System Actions
4	**R** sends a message asking for the required documentation from FI.	Once the message is sent by **R**, it is enreypted locally (in the client browser) and sent to the FISH application. Clouditor monitors the message sent and received, including encryption level and ciphers used.
5	**FI** logs into the EU-SEC platform using its own credentials (two-factor authentication).	FISH application authenticates **FI**. Clouditor monitors the access of **FI**.
6	**FI** opens its workspace, visualizes the message of **R**.	FISH application opens the workspace of **FI** and shows an alert indicating the incoming message from **R**. When the user access the message, it is decrypted locally (on the client side) and visualized by **FI**. Clouditor monitors the reception of the message by **FI**.
7	**FI** locally encrypts the document demanded by **R** and uploads the documentation to the FISH application.	FISH application shows to **FI** the option to send a file to **R**. The document is encrypted locally (in the browser of **FI**). Clouditor tool is activated and starts to collect evidence records about data location and encryption related to this asset (the shared document).
8	**R** receives the the document. It accesses and visualizes the documentation received.	Clouditor monitors and reports the access to the documentation.
9	**AU** logs on Starwatch.	Starwatch shows login window and verifies the credentials of **AU**.
10	**AU** access the assessment case, checks the certification status, the controls tested and selects a specific evidence record of a control.	Starwatch show the list of assessment cases and when **AU** selects a case it lists the controls that are being checked in that assessment, and subsequentially the list of reference to the different evidence records when a control is selected.
11	**AU** logs on Nuvla (Evidence Store)	Nuvla checks the credentials of **AU** and gives it access.
12	**AU** introduces the reference of the evidence record and checks everything is ok.	Nuvla shows the details of the evidence record.

Alternate Flow		
Step	User Actions	System Actions
7.1	**FI** argues that they cannot accept the request by **R** and justifies that the information cannot be shared.	FISH application shows an interface to offer the option to send a negative message back to **R**.

USE CASE 2: SAAS FISH APP

Table 3-2 Use Case 2 definition

Use Case Identification and History			
Use Case ID:	EU-SEC.FISH.SAAS.v1		
Use Case Name:	Financial Information reporting to regulator - FISH app approach over SAAS.	Version No:	1.1
End Objective:	To validate the integration of existing or upcoming SAAS applications with the CAC reference architecture developed in EU-SEC project. It tries to extend the scope of use case 1, by undertaking the same process of Financial Information Sharing from the users side, but using the SaaS approach and the adaptation of an already mature application on the cloud.		
Created by:	CAIXA	On (date):	03/12/2018
Last Update by:	CAIXA	On (date):	25/02/2019
Users/Actors:	• **Financial Institution (FI):** An employee of a financial institution dedicated to the financial sector. • **Regulator User (R):** An employee of the entity that regulates the financial sector. • **Additional Regulator User (R2):** Another employee of a regulator entity. • **Cloud Service Provider (CSP):** Cloud provider and FISH application provider. • **Auditor (AU):** Auditor or an external entity employee that is in charge of validating the performance and trustworthiness of the FISH application in the cloud.		
Trigger:	Regulator wants to claim specific sensitive financial information from a financial institution (e.g. periodic reporting of the security projects that are being implemented in the institution).		
Frequency of Use:	Weekly/Bi-weekly.		

Preconditions
• EU-SEC architecture, all the components and the FISH application are already deployed. That means, among other aspects, that: ○ **AU** has already created and configured an assessment case in Starwatch. ○ **CSP** and **AU** configured Clouditor checking the results of controls of FISH application and sending collected evidence to a trusted external evidence store, to the FI on-premise store and a reference of the evidence record to Starwatch v2. Clouditor is configured to continuously check the configured controls, particularly to ensure that FI requirements are achieved (location, encryption, user management, etc.). ○ All the information from the continuous auditing has to be stored as logs and evidence, particularly when there are indicators of non-compliance with the requirements. • **R**, **R2** and **FI** have their own credentials to access the FISH app, provided by **CSP**. Strong authentication mechanisms are provided by **CSP**. • There is a common and secure workspace between **R**, **R2** and **FI** already created in the FISH application by CSP.

Basic Flow		
Step	User Actions	System Actions
1	**FI** logs on the FISH application.	FISH application shows a LOGON interface and authenticates the credentials of **FI**. Clouditor monitors the access of **FI**.
2	**FI** navigates to the Teamroom "Security Project Reports" and create a new report. It fills the mandatory fields and attaches a report by uploading it from the computer.	FISH application shows the navigation menu and context menu or dashboard of tools enabling the option to submit a new report.
3	**FI** starts a process to validate the report form and assigns the task to team *Regulators*, selecting that the approval is required by **R** and **R2**.	FISH application shows the tools menu when **FI** clicks the updated report, and the option to start a new review and approval process.
4	**R** logs on the FISH application.	FISH application shows a LOGON interface and authenticates the credentials of **R**. Clouditor monitors the access of **R**.
5	**R** identifies that it has received a notification of work to be done and access the worklist	FISH application shows the main window with the notification of a new work to be done in the worklist.

Basic Flow		
Step	User Actions	System Actions
6	**R** clicks on the report to be approved, visualize it and approve it.	FISH application shows the report to the approved to **R** and the context menu with the option to approve the report.
7	**R2** logs on the FISH application.	FISH application shows a LOGON interface and authenticates the credentials of **R2**. Clouditor monitors the access of R2.
8	**R2** identifies that it has received a notification of work to be done and access the worklist	FISH application shows the main window with the notification of a new work to be done in the worklist.
9	**R2** clicks on the report to be approved, visualize it and approve it.	FISH application shows the report to the approved to **R** and the context menu with the option to approve the report. A notification is sent to **FI** informing it that the report is fully approved.
10	**AU** logs on Starwatch.	Starwatch shows login window and verifies the credentials of **AU**.
11	**AU** accesses the assessment case, checks the certification status, the controls tested and selects a specific evidence record of a control.	Starwatch show the list of assessment cases and when **AU** select a case it lists the controls that are being checked in that assessment, and subsequentially the list of reference to the different evidence records when a control is selected.
12	**AU** logs on Nuvla (Evidence Store)	Nuvla checks the credentials of **AU** and gives it access.
13	**AU** introduces the reference of the evidence record and checks everything is ok.	Nuvla shows the details of the evidence record.

Alternate Flow		
Step	User Actions	System Actions
6.1	**R** rejects the report.	FISH application sends a notification to **FI**, informing the report has been rejected, and the comments from **R** that needs to be reviewed before submitting it again.
9.1	**R2** rejects the report.	FISH application sends a notification to **FI**, informing the report has been rejected, and the comments from **R2** that needs to be reviewed before submitting it again.

3.2 MAPPING PILOT REQUIREMENTS TO SLOS/SQOS

The security standard schemes, national and international legislation and other relevant guidelines define the security requirements and controls used in cloud security environments. The problem with these type of requirements and controls is that they are designed to be used in point-in-time audits which usually occur every six months or annually. These audits are also designed to be conducted manually by an auditor which means that the requirements and controls are quite generic and allow auditor interpretation. In the context of CAC, this type of approach is not possible when automated and more frequent checks are involved. Therefore to be able to apply the existing requirements and controls to the CAC model, we need to identify a method to break down the controls to measurable objectives that can be easily validated. The method is called SLO and SQO definition.

SLO’s and SQO’s are ways to define measurable values or characteristics which can be used in the CAC scheme. SLO stands for Service Level Objective and SLQ for Service Qualitative Objective. SLO is a commitment which a cloud service provider makes for a specific, quantitative characteristic of a cloud service, where the value follows the interval scale or ratio scale. Correspondingly SQO is a commitment which a cloud service provider makes for a specific, qualitative characteristic of a cloud service where the value follows the nominal scale or ordinal scale. By using both SLO’s and SQO’s we can efficiently define the relevant requirements and controls in a measurable form which can be utilized in the pilot exercise. The definition of SLO’s and SQO’s is done by using auditor’s professional judgement of the requirements and the environment in scope. This process was previously further detailed in EU-SEC *Deliverable 2.2 – Continuous Auditing Certification Scheme*.

In EU-SEC *Deliverable 5.1 - Pilot Preparation* (D5.1), four main categories for requirements were selected. These categories consist of 9 controls which are mapped to relevant CSA CCM controls. The categories are Data Location, Encryption, Identity Federation and Evidence Security. The selection of categories has been defined in D5.1. The selected controls are presented in Table 3-3.

Table 3-3 Selected pilot requirements

Data Location	Type	Control	CCM Code
Local VM data	Platform	Location of all sensible data and its usage by applications and databases should be known	CCM-GRM-02

Persistent Data Storage	Platform	All data should be located within European Economic Space	CCM-STA-05
Encryption	**Type**	**Control**	**CCM Code**
Encryption on data transfers and data at rest	Application	All data stored on Cloud should be encrypted in rest and in transit	CCM-EKM-04
Key management	Application	Cryptographic key management policy and procedures should be defined.	CCM-EKM-02
Secure ciphers	Application	AES-256 should be used and only CaixaBank should be the owner of cryptographic keys	CCM-EKM-04
Identity Federation	**Type**	**Control**	**CCM Code**
VM access control	Platform	Identity administration federated to the administrator of CaixaBank	CCM-IAM-12
Application authentication	Application	Strong authentication of admin users	CCM-IAM-12
Application access control	Application	Access control and admin profiles should be defined	CCM-IAM-12
Evidence Security	**Type**	**Control**	**CCM Code**
Store evidences in CaixaBank	Client	All critical logs should be send to the SIEM of CaixaBank	CCM-IVS-01

3.2.1 DATA LOCATION

To define the metrics on how a certain control is measured, we need to first define the objectives of the controls to quantify the measurable attributes which can be utilized in the CAC. The CCM controls selected for this purpose are GRM-02 and STA-05 which define the data location requirements for all softwares and databases that are used in the environment.

In the financial world, the data in use is sensitive, and therefore it is subject to very strict regulation, in which data location is an important part. For instance, information shared among regulatory authorities and financial institutions should be stored always inside the physical borders of the EU. However, there is no mechanism to ensure that this condition is always enforced by the Cloud provider. The easy solution would be to make sure that the physical location of data can be controlled with the physical placement of all the hardware which handles any data in scope. Still this does not provide full confidence that the data would not leak outside acceptable locations. Therefore it is essential to check the location of data in software level to make sure that the data is not leaking anywhere. There are a couple of reasons to control the location of the data. The most obvious reason being the legislation (i.e GDPR) and another being human error. In a situation where an international organization has multiple data locations around the world. Of course location checks might also prevent malicious activities but primarily data location controls are designed to ensure that the assessed entity complies with legislation.

Location of persistent data storage is actually a yes/no -type query (the data is located in an accepted country/location or it is not). If the answer is no, then the check fails and appropriate actions should be taken to correct the issue. The approach to address this requirement is first to define the data under the scope of the audit, which shall be all sensitive data shared between entities. Also the location of all entities that access the unencrypted data have to be included in the scope of the requirements. To verify the location, we need an attribute to track the location in which the data is located, which will be used to confirm that the data is in an acceptable location.

SLA/SQO Definition:

The following definitions apply to all platforms in scope.

Persistent data storage:

- For all applicable sensitive data in scope, it shall be checked every 60 minutes that the persistent data location is known and trusted.
  - Evidence: Location attribute
  - Metric: Whitelisted locations
  - Result: Pass/Fail

Location of VM data:

- Upon request of sensitive data by a software or database, it shall be verified that the delivery/processing location is known and trusted.
  - Evidence: Location attribute
  - Metric: Whitelisted locations
  - Result: Pass/Fail

3.2.2 ENCRYPTION

This category consists of three different controls that are mapped to 2 CCM contols EKM-02 and EKM-04. The controls aim to make sure that the data is encrypted both in transit and at rest, there are secure key management procedures and that only secure ciphers are used. Encryption is used to provide security to sensitive data while it is at rest or in transit. It is beneficial to divide these to two different objectives because there are different objectives to be met. Correspondingly, the two other controls shall have their own objectives.

Objective one considers the data at rest. All sensitive data that is not being used or transmitted, shall be considered to be at rest. Using this approach it can be made sure, that all sensitive data is encrypted when it is not needed. To quantify the objective to measurable metrics, the used attributes must be defined. In this case it is simple: It shall be validated whether the data in scope is encrypted or not with appropriate encryption method (e.g. AES-256). The appropriate encryption methods must be defined, and the audit check must be in line with the defined encryption methods. This will result in a SLO/SQO definition in which there are only two options, yes and no, which are simple to measure.

Objective two is about encryption data in transit. This covers all sensitive data which is shared over network between applications. There are basically two important aspects to take into consideration. First of all, the connection must be configured correctly, and secondly, all sensitive data must be encrypted properly while in transit between applications.

The objective of secure key management is to ensure that the encryption keys used for encryption are stored securely to prevent unauthorized use. To meet these requirements, the ownership and location of encryption keys must be verified. It is also important that the keys are not stored in the cloud to prevent them from leaking outside and be potentially in possession of any other party than their owner. Ideally, the only owner of symmetric encryption keys shall be the financial institution (e.g. CaixaBank). Secure ciphers play a key role in encryption in assuring that the connections are truly secure and encrypted messages are kept secret. To assure that only secure ciphers are used, the allowed ciphers must be defined and any others must be denied.

SLA/SQO Definition:

Data at rest:

- It shall be verified that the data at rest is encrypted at all times with acceptable encryption method (AES-256). These checks shall be done in 5 minute intervals. (yes/no)
    - Evidence: Encryption method used
    - Metric: Acceptable encryption methods
    - Result: Pass/Fail

Data in transit:

- When establishing new connections between applications, it shall be ensured that the HTTPS (TLS) connection is configured correctly according to industry best-practices.
  - Evidence: Connection information
  - Metric: Best-practice configuration
  - Result: Pass/Fail
- Whenever sensitive data is transferred between applications and/or databases it shall be verified that the application encrypts all of the sensitive data with appropriate encryption methods.
  - Evidence: Encryption method and related information
  - Metric: Whitelisted encryption methods
  - Result: Pass/Fail

Key Management

- Encryption keys shall not be stored in cloud. Verify that encryption keys are not stored in cloud (Yes/no)
  - Evidence: URL of the storage location where the keys
  - Metric: Storage location of the keys
  - Result: Pass/Fail
- Verify that the keys for data in rest symmetric encryption are in possession of owner (Cloud Customer) (yes/no)
  - Evidence: Verification and answer by Cloud Customer (SQO out of the scope of the automated tests of controls)
  - Metric: Storage location of the keys
  - Result: Pass/Fail
- Verify that encryption keys are stored in an accepted location
  - Evidence: Location attribute
  - Metric: Allowed location attribute list
  - Result: Pass/Fail

Secure Ciphers

- Verify that all encryption procedures are done with predefined and accepted ciphers (yes/no)
  - Evidence: Used cipher
  - Metric: Allowed cipher list
  - Result: Pass/Fail

3.2.3 IDENTITY FEDERATION

Identity federation requirements cover the identity and access management of the FISH application. There are 3 control objectives which are linked to CCM control IAM-12: VM access control, application authentication, application access control. The primary objective is that CaixaBank has to make sure that all applications in use are known and verified by CaixaBank. How can this be achieved then? The application's name and domain should be checked to meet known and approved applications. Also it should be checked that the user is active and its access has not been revoked. On the platform level the same kind of information should be retrievable.

SLA/SQO Definition:

Application level:

Authorization of applications shall be checked when access to sensitive data is requested.

- Evidence: Application name and domain**.**
- Metric: Access list.
- Result: Pass/fail**.**

Platform level:

Used platform shall be checked upon request to sensitive data:

- Evidence: Platform name and domain**.**
- Metric: White list of permitted platforms.
- Result: Pass/fail**.**

3.2.4 EVIDENCE SECURITY

Evidence security aims to assure that the audit evidence collected is securely stored so that unauthorized modification is not possible. There also has to be appropriate management procedures and policies to ensure the confidentiality and integrity of audit logs. The CCM control mapped for this purpose is IVS-01. Due to requirements from banks, the evidence must be stored in two different locations, in the CAC evidence service, and in-house inside the bank's own databases. This arrangement assures that the bank always has backups available if evidence is not retrievable from the auditing service. Both locations have to be included in the scope of evidence security requirements to assure secure storage of evidence.

Evidence is mostly collected in a digital form (logs and measurements). . The access to audit evidence shall be restricted so that nobody without authorized need has access to the data. For example, in the pilot exercise, apart from the auditor, only CaixaBank shall have the access to the evidence storage besides the auditor. The evidence storage shall also be located in a controlled location where other outsider users cannot access. These requirements are handled with the identity and access management controls defined above. To meet the controls' requirements, several objectives have to be defined.

SLA/SQO Definition:

- All critical data must be logged in realtime.
  - Evidence: Check the last time the application and evidence records are collected.
  - Metric: Grace period from present time to last recorded log timestamp**.**
  - Result: Pass/Fail.
- Logfile of information pushed to CaixaBank must be updated whenever such information transfer is done.
  - Evidence: Check the connection of the different modules pushing evidence records into CaixaBank SIEM.
  - Metric: Grace period from present time to last recorded evidence record timestamp**.**
  - Result: Pass/Fail.
- Location of logfile
  - Evidence: Location attribute**.**
  - Metric: Allowed location attribute list.
  - Result: Pass/Fail**.**

4 PILOT VALIDATION AND TESTING

This section explains the different tasks done for the validation of the deployed pilot. First, the architecture security recommendations are specified, defining a set of high-level best practices that were taken into account in the deployment of the CAC technical reference architecture and the EU-SEC CAC Pilot. They should be also considered as a guidance in the case of alternative implementation of the CAC architecture. The main responsible of this task was NIXU, taking the role of an auditor, with the help of the technology providers and developers of the different tools, and CaixaBank as a financial institution and customer of the FISH applications proposed in the use cases.

After that, the use case validation process is shown going through each of the steps of the previously detailed use cases, showing the actions of the actors and the response of the different tools in the interaction of them with the CAC reference architecture.

At the end of the section, some preliminary non-functional aspects are analysed.

4.1 ARCHITECTURE SECURITY RECOMMENDATIONS

4.1.1 INFORMATION SECURITY MANAGEMENT AND ARCHITECHTURE

The fundamental basis for security in an application such as the FISH application, is that the application itself is secure and managed properly to be able to truly safeguard the confidentiality, integrity and availability of data it is processing, storing and transmitting. To help to achieve these objectives, the cyber security industry has over the years developed best practice solutions for information security management. Best practice solutions have evolved to industry accepted standards and guidelines that are a good way of enhancing the security posture of applications and organizations. Security management is also an essential part of managing such a complex architecture containing sensitive data, because inefficient management will have negative concequences in many forms, foremost security. It must also be remembered that the application does not consist only of technologies but also includes people and procedures in the background that make the operational functions of the application possible.

To manage the architecture, it is recommended to use industry accepted standards such as ISO 27001 or more cloud specific standards such as CSA CCM, ISO 27017 or ISO 27018 as a basis.

The most important aspects being that the framework provides efficient management and supportive documentation. Good documentation is the backbone of every management system since it defines how the procedures are supposed to be working and therefore it has a direct effect on security. There are a few objectives that the documentation shall meet to achieve its purpose:

- Documentation has to provide detailed information of its applicability to provided task.
- Documentation has to provide detailed usage instructions / principles.
- Documentation has to provide detailed information about security of the environment and control/management of the environment.

Documentation itself is nevertheless not enough. The management of any information security system is based on knowing the threats towards the system and assessing the known risks. To identify the relevant risks, the use of known risk assessment methods is recommended. Luckily, there are multiple risk analysis methods which could be utilized. For example, ISO-based standards (ISO 27000) provide great examples for risk analysis.

The risk analysis should cover the whole architecture in scope of the application. Overall, the risk analysis is used to justify the architecture model selected for use. The risk analysis should include all different modules and integrations of the architecture to analyse risks. When evaluating the architecture model from a risk-based perspective, different cloud hosting options must be evaluated according to the risk level selected. For example, the cloud can be hosted in-house in a private cloud, or in public/hybrid cloud. Of course, the security level of these different hosting options is varying. With a well-prepared team with the expertise to define and maintain it up, private architecture is the safest option, but might require more resources to operate. It still has great benefits, since private cloud enables the use of sensitive data securely because exposure to public networks is not desired. Public cloud as an option is not very capable for such applications which handle sensitive data, e.g. data under GDPR regulation.

Nevertheless, all of the data in scope is not equally critical, therefore risk analysis has to be made on all of the use cases for different types of data and the application that is processing it. For example, sensitive data must be handled with more caution than less important data. This leads to a solution where some data must most likely be stored in a private cloud whereas some other data can be stored in public cloud. Based on the risk analysis, the correct security assessment can be made to evaluate the feasibility of the solutions. But it shall be taken into consideration, that all data should always be handled with care, and encryption and identity and access management shall be used efficiently to prevent any misuse of data.

There are multiple APIs which are used in the CAC architecture. All of the APIs also have to be evaluated with risk analysis based on industry standards. To be able to do this, each API requires documentation that clearly defines the features and procedures on how the API is working. Each API and application used in the CaC process must also be audited based on industry accepted standards to verify their security. This is especially important in this case to verify that the continuous auditing can be trusted.

It should be noted, that the standardized security of the environment used for CAC Architechture is highly important for the auditors, who are expected to use the tools specified during the EU-SEC project. And as such the CAC Architecture should be built and tested against well-known industry standards.

4.1.2 TECHNICAL SECURITY REQUIREMENTS

In CAC process the security of the technical components used to audit the target environment is important. All technical components addressing the audit process must be built securely. This can be archieved by following exiting international standards for applications and services.

CAC Architecture must be compliant with security standards such as ISO27001, ISO27002, ISO27018, CSA Star (CCM). Compliance and certification acquired for the environment is dependant on the implementation method of the environment (private cloud, public cloud, hybrid cloud). Each of the EU-SEC components shown in picture below must be tested and if and it the end certified with well-known industry standards.

1. Technical Architecture

As shown in the Figure 2-1, architecture descriptions must include the division of the communications network into separate network areas. Each of the network areas used to provide CAC service must be included in audit scope of the CAC Architecture audit. Each of these network areas must be separately audited and/or current status of their audit must be confirmed. Level of assurance must be based on level of detailed information stored in each segment.

Information Communications Networks used in Contiuous Auditing Certification Architecture can only be connected when using encryption approved by Crypto Approval Authority (CAA).

2. Data Centers

All Data Processing Facilities used in CAC must be audited using in the well-known industry standards. Data centers used for Continuous Auditing Process, Assessment and Evidence Storage must be secured and certified.

NOTE: Application audited (in this case FISH App) and In-house evidence collection should also be secured based on auditee risk assessment.

3. Infrastructure

All systems used to provide the Continuous Auditing Service must be hardened using well-known industry standards. Systems used, must be installed and configured systematically, resulting in a hardened installation. Systems referred here are all servers, workstations, active components of the network and aquivalent devices (such as firewalls, routers, switches and equivalent devices).

4. Application Security

All applications used to provide the CAC must be audited using well-known industry standards. Application security must include the development processes of the applications in use.

All applications used in CAC Architecture, shall be subject to security testing to ensure that the appropriate level of assurance is obtained and to verify that they are correctly implemented, integrated and configured. Appropriate level of authentication and authorization must be implemented in each application component implemented in CAC Architecture.

All Application Application Programming Interfaces (API) used to connect different components of the CAC Architecture must be audited using the well-known industry standards.

5. Data Security

Data classification must be conducted. This includes what type of data is transmitted, what type of data is stored (data in-rest). All necessary protection to data in use must be applied as defined in well-known industry standards.

Minimum requirements for data encryption must be applied:

- Requirements of the National Communications Security Authority (NCSA) and Crypto Approval Authority must be followed.

4.2 USE CASES TESTING AND SQO/SLOS VALIDATION

Detailed script of the use cases, showing the interfaces of the different modules of the technical architecture and the interactions with the actors.

Actors:

- **Financial Institution (FI):** An employee of a financial institution dedicated to the financial sector.
- **Financial Institution (FI2):** Another employee of a financial institution dedicated to the financial sector.
- **Regulator User (R):** An employee of the entity that regulates the financial sector.
- **Cloud Service Provider (CSP):** Cloud provider and FISH application provider.
- **Auditor (AU):** Auditor or an external entity employee that is in charge of validating the performance and trustworthiness of the FISH application in the cloud.

Preconditions to take into account prior to the users entering the FISH app (either use case 1 or 2):

AU should enter[3] Clouditor and configure the controls for the FISH app according to the ones defined together with CSP and the requirements specified by the cloud customers (in this case, FI).

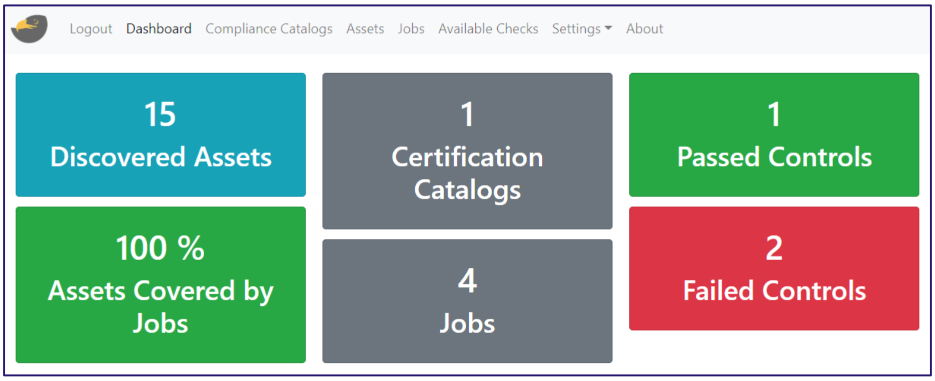
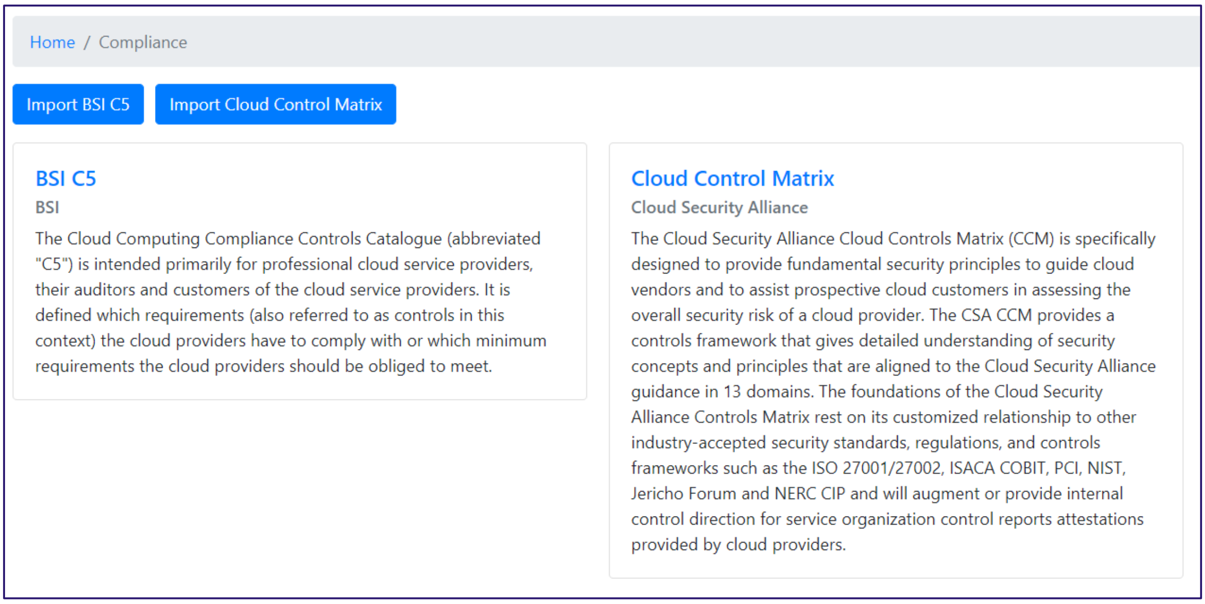


Figure 4-1 Clouditor main window

AU selects which catalog of controls (Figure 4-2) want to take into account in the auditing process. The version of Clouditor used in the pilot allows to define controls from Cloud Control

[3] Clouditor URL for the pilot: https://eusec.clouditor.io

Matrix (CCM) and BSI C5. In the future, additional catalogues will be able to be imported. After selecting the catalogue, A can choose which controls to be applied.



BSI C5

BSI

The Cloud Computing Compliance Controls Catalogue (abbreviated "C5") is intended primarily for professional cloud service providers, their auditors and customers of the cloud service providers. It is defined which requirements (also referred to as controls in this context) the cloud providers have to comply with or which minimum requirements the cloud providers should be obliged to meet.

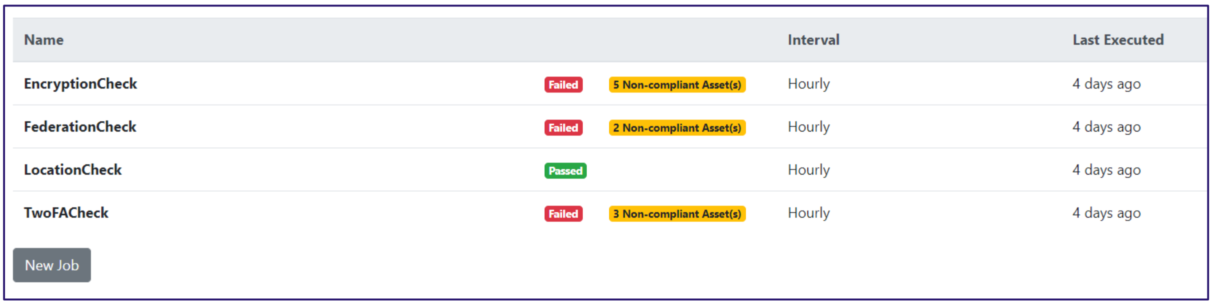
Cloud Control Matrix

Cloud Security Alliance

The Cloud Security Alliance Cloud Controls Matrix (CCM) is specifically designed to provide fundamental security principles to guide cloud vendors and to assist prospective cloud customers in assessing the overall security risk of a cloud provider. The CSA CCM provides a controls framework that gives detailed understanding of security concepts and principles that are aligned to the Cloud Security Alliance guidance in 13 domains. The foundations of the Cloud Security Alliance Controls Matrix rest on its customized relationship to other industry-accepted security standards, regulations, and controls frameworks such as the ISO 27001/27002, ISACA COBIT, PCI, NIST, Jericho Forum and NERC CIP and will augment or provide internal control direction for service organization control reports attestations provided by cloud providers.

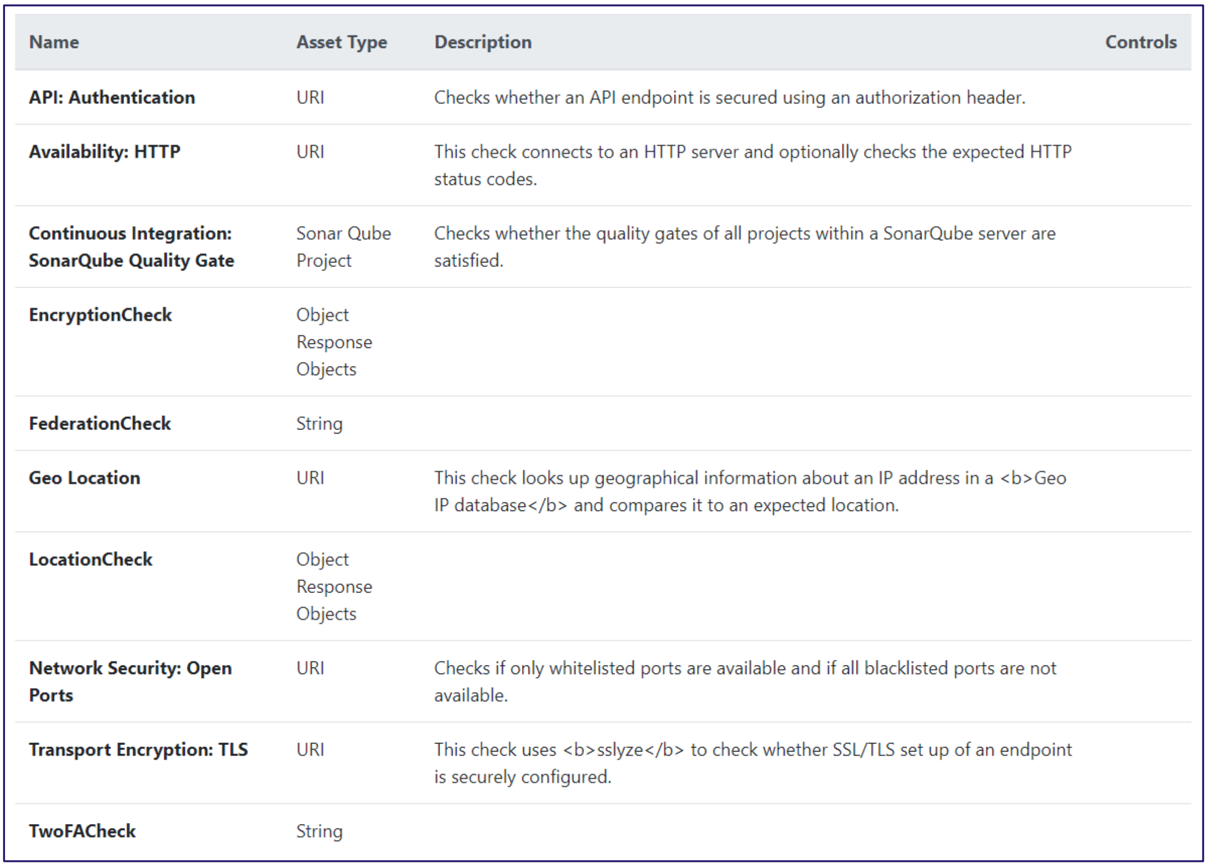
Figure 4-2 Clouditor control catalogs

AU configures four jobs (Figure 4-3) that check a set of controls according to the requirements previously defined by CaixaBank and the mapping task summarized in section 3.2. Figure 4-4 shows all available checks that the can be used to schedule jobs. Currently a total of 10 checks are available.



Name			Interval	Last Executed
EncryptionCheck	Failed	5 Non-compliant Asset(s)	Hourly	4 days ago
FederationCheck	Failed	2 Non-compliant Asset(s)	Hourly	4 days ago
LocationCheck	Passed		Hourly	4 days ago
TwoFACheck	Failed	3 Non-compliant Asset(s)	Hourly	4 days ago

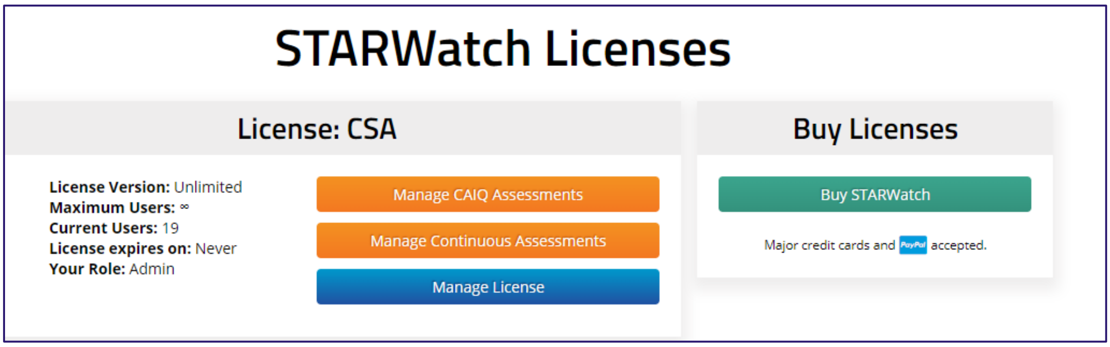
Figure 4-3 Jobs configured in clouditor for control checking of cases 1 and 2



Name	Asset Type	Description	Controls
API: Authentication	URI	Checks whether an API endpoint is secured using an authorization header.	
Availability: HTTP	URI	This check connects to an HTTP server and optionally checks the expected HTTP status codes.	
Continuous Integration: SonarQube Quality Gate	Sonar Qube Project	Checks whether the quality gates of all projects within a SonarQube server are satisfied.	
EncryptionCheck	Object Response Objects		
FederationCheck	String		
Geo Location	URI	This check looks up geographical information about an IP address in a <b>Geo IP database</b> and compares it to an expected location.	
LocationCheck	Object Response Objects		
Network Security: Open Ports	URI	Checks if only whitelisted ports are available and if all blacklisted ports are not available.	
Transport Encryption: TLS	URI	This check uses <b>sslyze</b> to check whether SSL/TLS set up of an endpoint is securely configured.	
TwoFACheck	String		

Figure 4-4 Available control checks in Clouditor

AU accesses StarWatch[4] main window and configures a continuous auditing assessment (Figure 4-5).



STARWatch Licenses

License: CSA

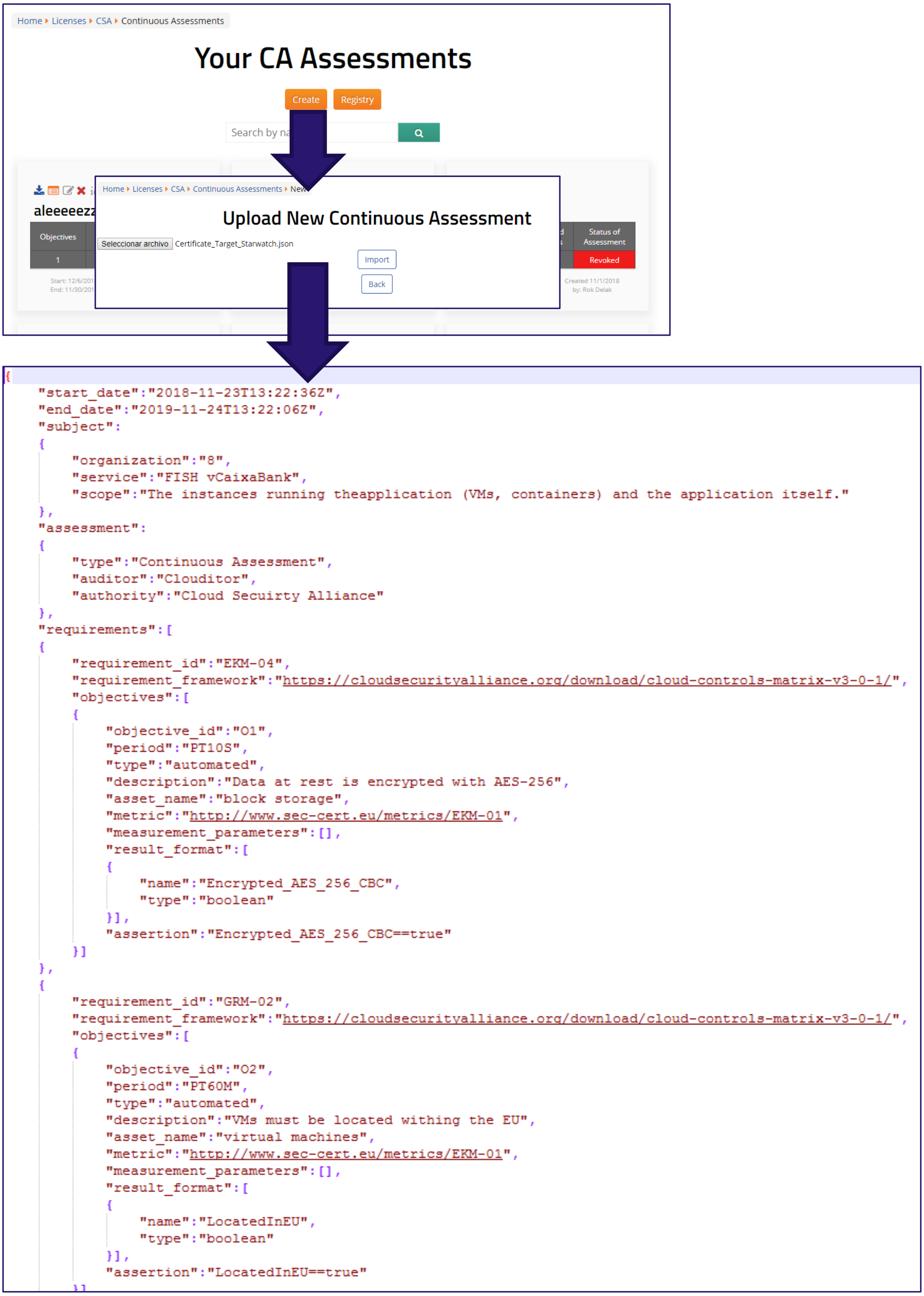
License Version: Unlimited
Maximum Users: ∞
Current Users: 19
License expires on: Never
Your Role: Admin

Manage CAIQ Assessments
Manage Continuous Assessments
Manage License
Buy Licenses
Buy STARWatch
Major credit cards and PayPal accepted.

Figure 4-5 StarWatch starting window

For creating a new assessment, **AU** uses a configuration JSON file with the specification of the assessment. In the example shown in Figure 4-6, the controls are defined according to the requirements of CaixaBank.

[4] Pilot URL of StarWatch: https://eusec-dev-9ejkbcabjw.star.watch/

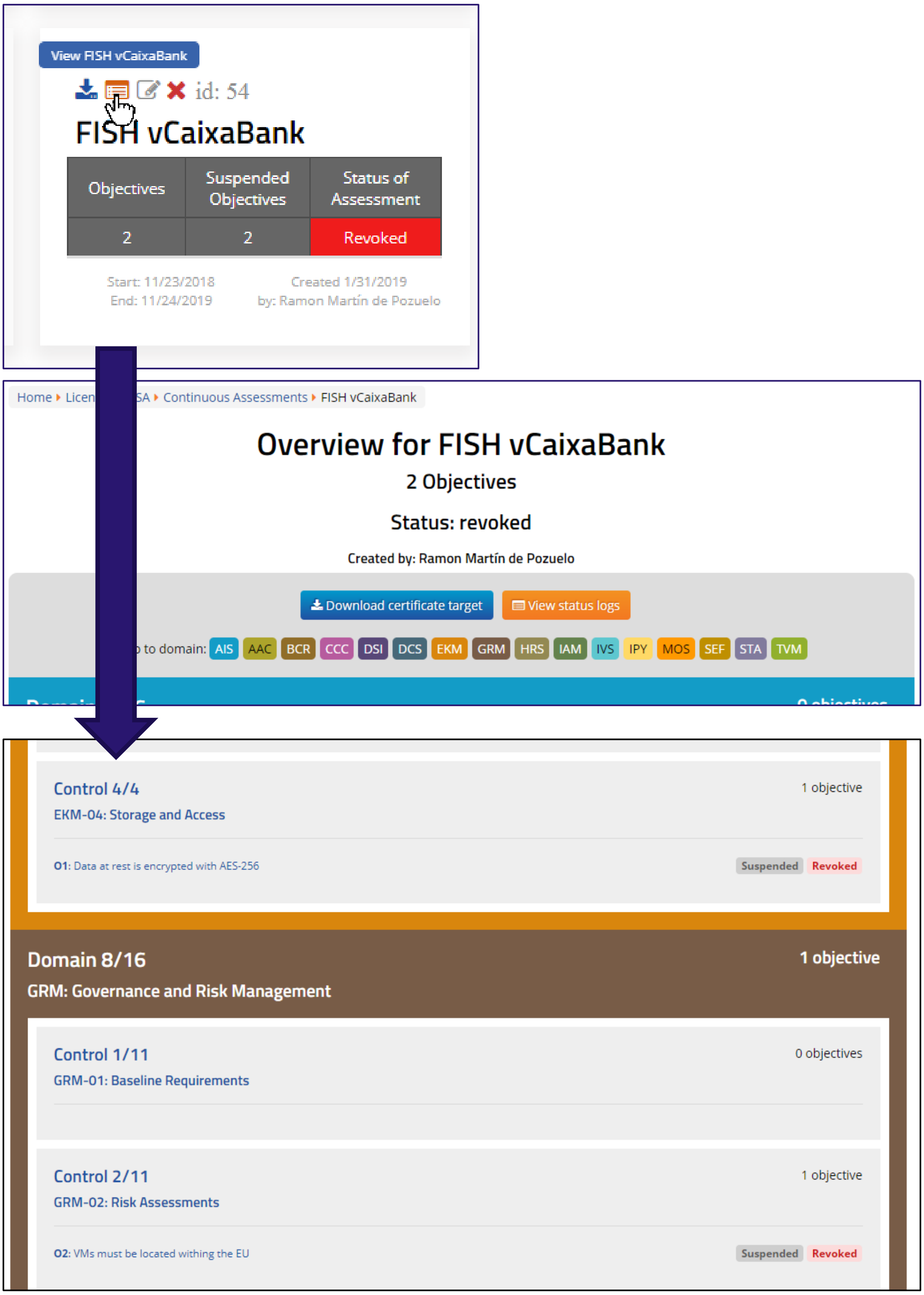


Certificate_Target_Starwatch.json

```
{
    "start_date":"2018-11-23T13:22:36Z",
    "end_date":"2019-11-24T13:22:06Z",
    "subject":
    {
        "organization":"8",
        "service":"FISH vCaixaBank",
        "scope":"The instances running theapplication (VMs, containers) and the application itself."
    },
    "assessment":
    {
        "type":"Continuous Assessment",
        "auditor":"Clouditor",
        "authority":"Cloud Secuirty Alliance"
    },
    "requirements":[
    {
        "requirement_id":"EKM-04",
        "requirement_framework":"https://cloudsecurityalliance.org/download/cloud-controls-matrix-v3-0-1/",
        "objectives":[
        {
            "objective_id":"O1",
            "period":"PT10S",
            "type":"automated",
            "description":"Data at rest is encrypted with AES-256",
            "asset_name":"block storage",
            "metric":"http://www.sec-cert.eu/metrics/EKM-01",
            "measurement_parameters":[],
            "result_format":[
            {
                "name":"Encrypted_AES_256_CBC",
                "type":"boolean"
            }],
            "assertion":"Encrypted_AES_256_CBC==true"
        }]
    },
    {
        "requirement_id":"GRM-02",
        "requirement_framework":"https://cloudsecurityalliance.org/download/cloud-controls-matrix-v3-0-1/",
        "objectives":[
        {
            "objective_id":"O2",
            "period":"PT60M",
            "type":"automated",
            "description":"VMs must be located withing the EU",
            "asset_name":"virtual machines",
            "metric":"http://www.sec-cert.eu/metrics/EKM-01",
            "measurement_parameters":[],
            "result_format":[
            {
                "name":"LocatedInEU",
                "type":"boolean"
            }],
            "assertion":"LocatedInEU==true"
        }]
```

Figure 4-6 Example of new assessment process

After creating this assessment shown in Figure 4-6, A can check the status of the assessment and its evaluation (certified, suspended or revoked).



id: 54

FISH vCaixaBank

Objectives	Suspended Objectives	Status of Assessment
2	2	Revoked

Start: 11/23/2018
End: 11/24/2019

Created 1/31/2019
by: Ramon Martín de Pozuelo

Home ▸ Licen SA ▸ Continuous Assessments ▸ FISH vCaixaBank

Overview for FISH vCaixaBank

2 Objectives

Status: revoked

Created by: Ramon Martín de Pozuelo

Download certificate target
View status logs
to domain: AIS AAC BCR CCC DSI DCS EKM GRM HRS IAM IVS IPY MOS SEF STA TVM

Control 4/4 — 1 objective
EKM-04: Storage and Access

O1: Data at rest is encrypted with AES-256 — Suspended Revoked

Domain 8/16 — 1 objective
GRM: Governance and Risk Management

Control 1/11 — 0 objectives
GRM-01: Baseline Requirements

Control 2/11 — 1 objective
GRM-02: Risk Assessments

O2: VMs must be located withing the EU — Suspended Revoked

Figure 4-7 Assessment overview in StarWatch.

USE CASE 1

R logs into the FISH app over IaaS[5] with its credentials (previously provided by **CSP**).

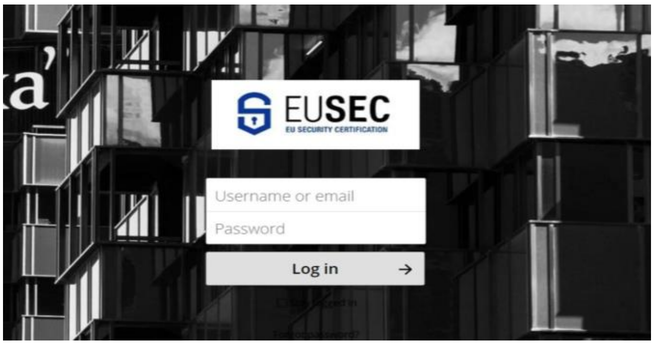
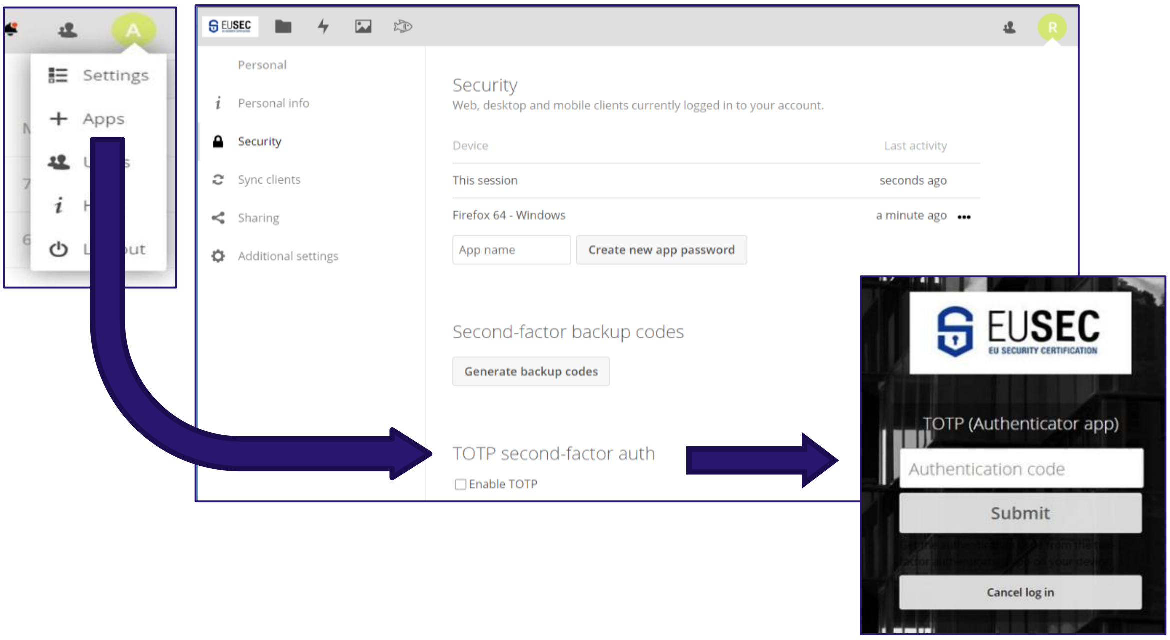


Figure 4-8 FISH application log in window

The second factor autenthication is also tested. As shown in Figure 4-9, it can be activated or deactivated on the user settings (right-top corner) of the FISH app.



Security

Web, desktop and mobile clients currently logged in to your account.

Device	Last activity
This session	seconds ago
Firefox 64 - Windows	a minute ago

App name | Create new app password

Second-factor backup codes

Generate backup codes

TOTP second-factor auth

☐ Enable TOTP

Figure 4-9 Activation of second factor authentication

R opens the main window of the application, selects an institution to open a new case of reporting (Figure 4-10)

[5] FISH App (Use Case 1) for the pilot: https://ec2-18-197-203-65.eu-central-1.compute.amazonaws.com/index.php/login

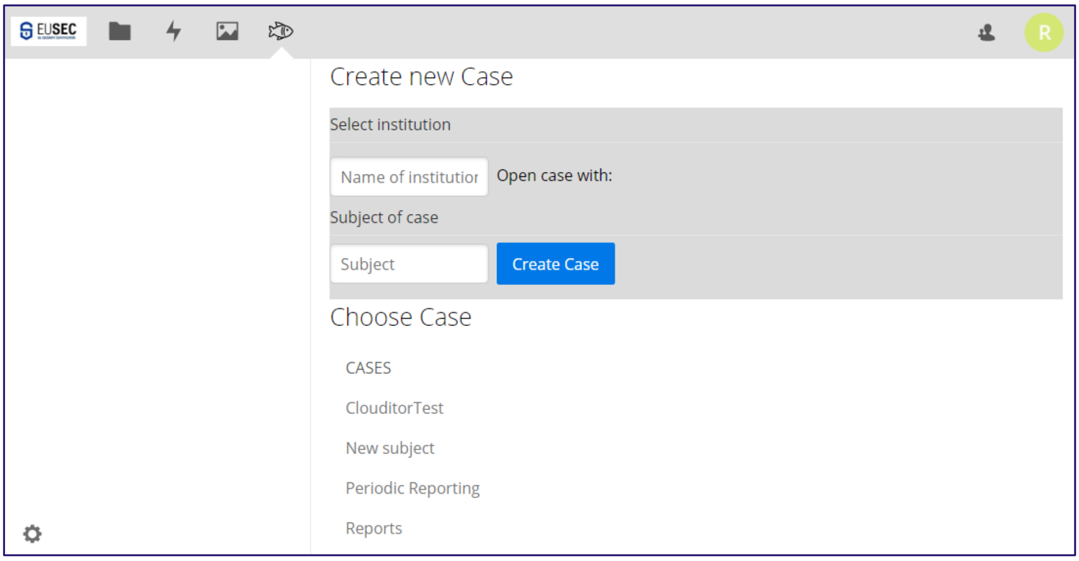
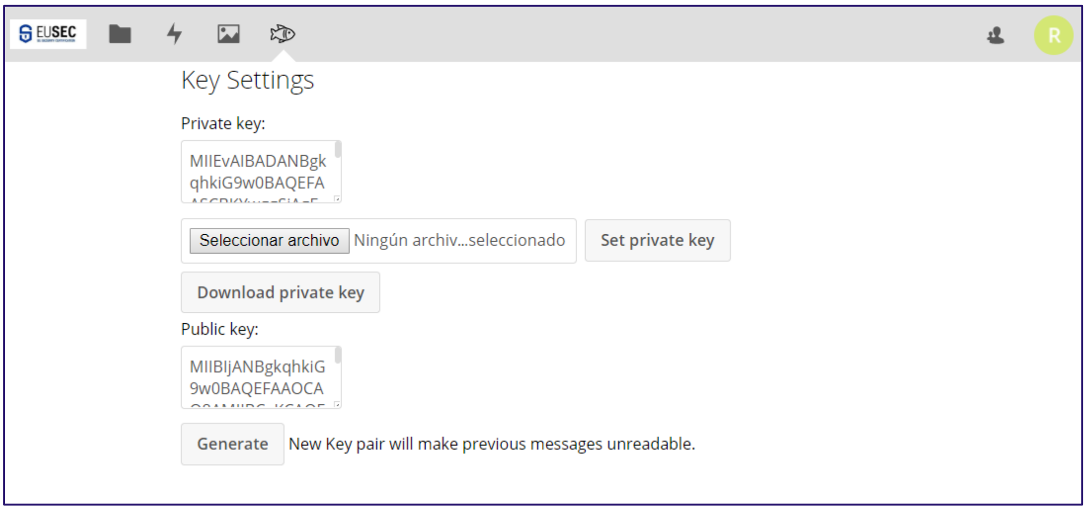


Figure 4-10 FISH app window to open a new case between regulator and finance institution.

R fills the blank fields and press “Create Case”:

- Select Institution: Name of institution to which you want to establish a new case (e.g. CaixaBank).
- Subject of the Case: Name of the case (e.g. CaixaBank reports 2019).

A screen will appear for setting up the public and private key of the user (Figure 4-11).



Private key: MIIEvAIBADANBgk qhkiG9w0BAQEFA

Public key: MIIBIjANBgkqhkiG 9w0BAQEFAAOCA

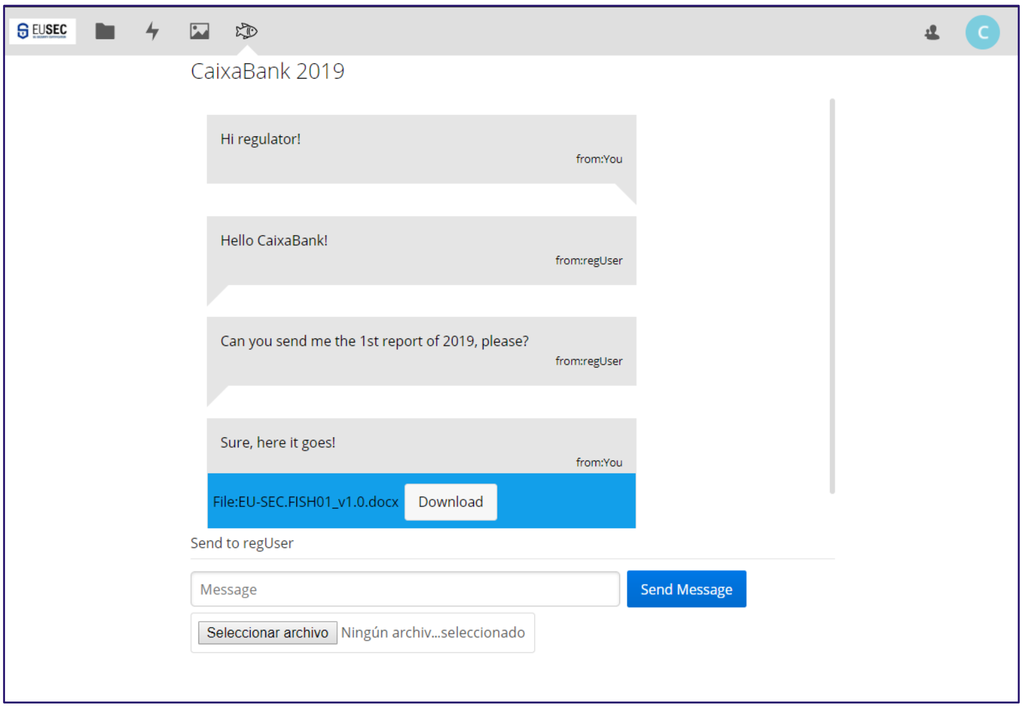
New Key pair will make previous messages unreadable.

Figure 4-11 FISH app public/private key generation window

R presses the button “Set Private Key” in order to save that key in the LocalStorage of the browser. It can also download the private key and save it in your computer.

FI follows the same process executed by **R**. It logs into the application and select the created case by **R** and assign itself to that case. The key set up screen will appear when accessing to it for the first time.

R and **FI** starts a conversation via chat the interface and **R** ask for a specific report.



CaixaBank 2019

Hi regulator!
from:You

Hello CaixaBank!
from:regUser

Can you send me the 1st report of 2019, please?
from:regUser

Sure, here it goes!
from:You

File:EU-SEC.FISH01_v1.0.docx

Send to regUser

Message

Ningún archiv...seleccionado

Figure 4-12 FISH app chat window between R and FI

After the exchange of information is

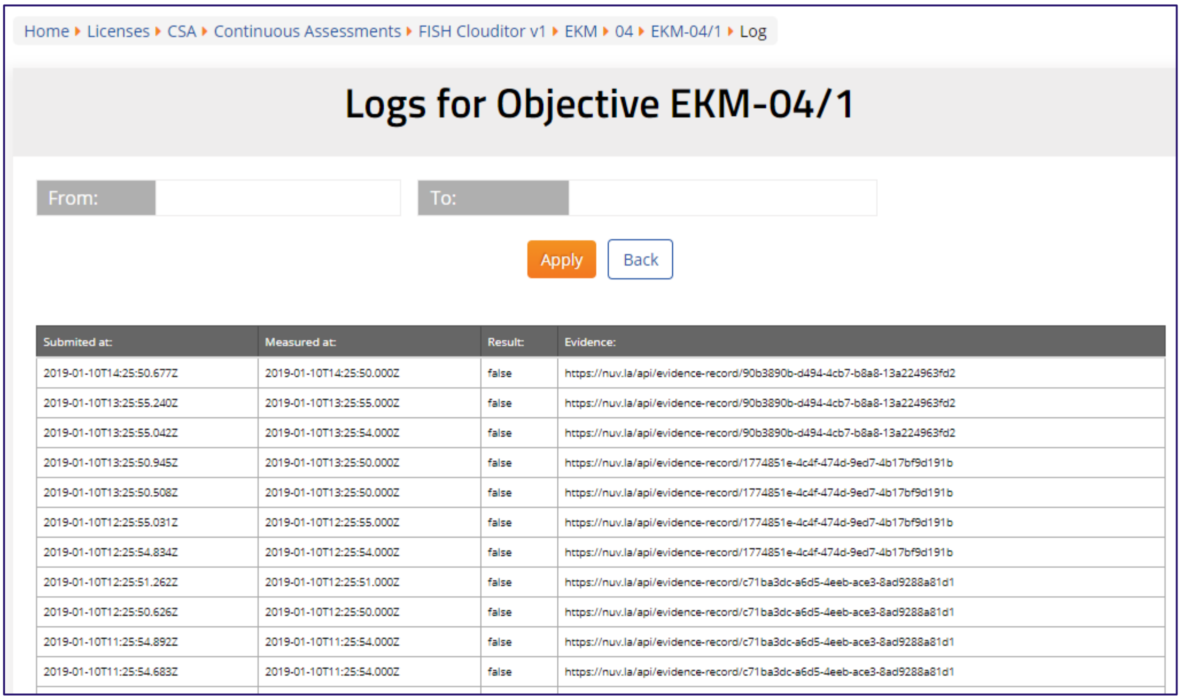
AU enters Starwatch, checks that the certification is active in the FISH app assessment and verifies that the different controls defined in the assessment are achieved:

- *Identity Federation*:
  - Checks that the users are valid and they are authenticated with two-factor authentication.
- *Data Location*:
  - All the shared documents between **FI** and **R** are stored in a location inside the while list of countries inside the European Economic Area.
- Encryption:
  - The encryption algorithm of all the shared between **FI** and **R** is in the white list according to the requirements of CaixaBank, such as AES-256 for data at rest.

- Data-in-transit security is established by means of a protocol is in the white list according to the requirements of CaixaBank.

- Critical logs owned by CaixaBank:
  - Accessibility to evidence store logs by CaixaBank users.

A selects a specific control and analyse the evidence records of the control checks sent by Clouditor (Figure 4-13).

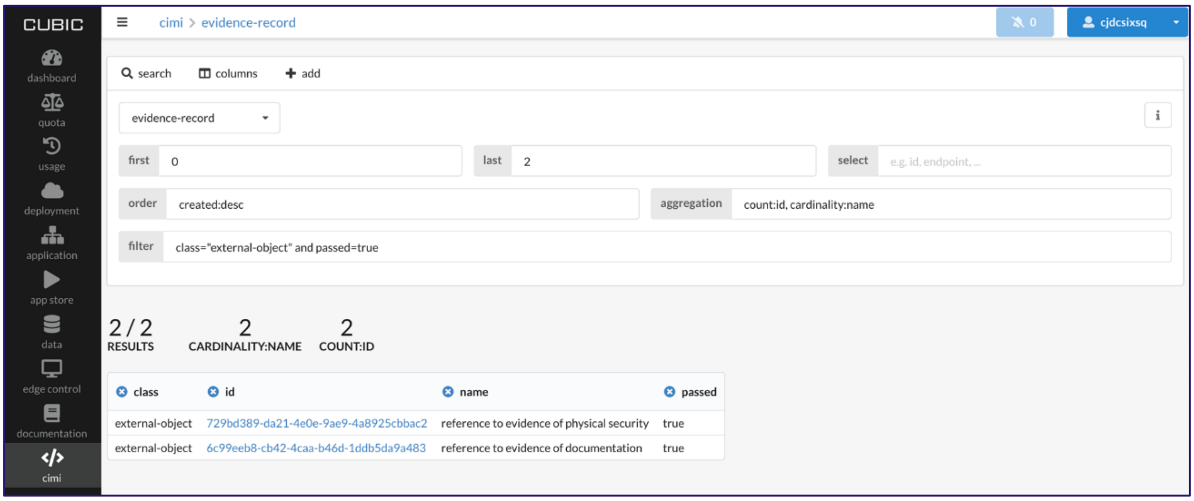

Home ▸ Licenses ▸ CSA ▸ Continuous Assessments ▸ FISH Clouditor v1 ▸ EKM ▸ 04 ▸ EKM-04/1 ▸ Log
From:
To:
Apply
Back

Logs for Objective EKM-04/1

Submited at:	Measured at:	Result:	Evidence:
2019-01-10T14:25:50.677Z	2019-01-10T14:25:50.000Z	false	https://nuv.la/api/evidence-record/90b3890b-d494-4cb7-b8a8-13a224963fd2
2019-01-10T13:25:55.240Z	2019-01-10T13:25:55.000Z	false	https://nuv.la/api/evidence-record/90b3890b-d494-4cb7-b8a8-13a224963fd2
2019-01-10T13:25:55.042Z	2019-01-10T13:25:54.000Z	false	https://nuv.la/api/evidence-record/90b3890b-d494-4cb7-b8a8-13a224963fd2
2019-01-10T13:25:50.945Z	2019-01-10T13:25:50.000Z	false	https://nuv.la/api/evidence-record/1774851e-4c4f-474d-9ed7-4b17bf9d191b
2019-01-10T13:25:50.508Z	2019-01-10T13:25:50.000Z	false	https://nuv.la/api/evidence-record/1774851e-4c4f-474d-9ed7-4b17bf9d191b
2019-01-10T12:25:55.031Z	2019-01-10T12:25:55.000Z	false	https://nuv.la/api/evidence-record/1774851e-4c4f-474d-9ed7-4b17bf9d191b
2019-01-10T12:25:54.834Z	2019-01-10T12:25:54.000Z	false	https://nuv.la/api/evidence-record/1774851e-4c4f-474d-9ed7-4b17bf9d191b
2019-01-10T12:25:51.262Z	2019-01-10T12:25:51.000Z	false	https://nuv.la/api/evidence-record/c71ba3dc-a6d5-4eeb-ace3-8ad9288a81d1
2019-01-10T12:25:50.626Z	2019-01-10T12:25:50.000Z	false	https://nuv.la/api/evidence-record/c71ba3dc-a6d5-4eeb-ace3-8ad9288a81d1
2019-01-10T11:25:54.892Z	2019-01-10T11:25:54.000Z	false	https://nuv.la/api/evidence-record/c71ba3dc-a6d5-4eeb-ace3-8ad9288a81d1
2019-01-10T11:25:54.683Z	2019-01-10T11:25:54.000Z	false	https://nuv.la/api/evidence-record/c71ba3dc-a6d5-4eeb-ace3-8ad9288a81d1

Figure 4-13 Evidence records collected from control EKM-04

AU Nuvla (Slipstream) and review one of the evidence records (Figure 4-14).



evidence-record

first 0

last 2

select e.g. id, endpoint, ...

order created:desc

aggregation count:id, cardinality:name

filter class="external-object" and passed=true

2 / 2 RESULTS

2 CARDINALITY:NAME

2 COUNT:ID

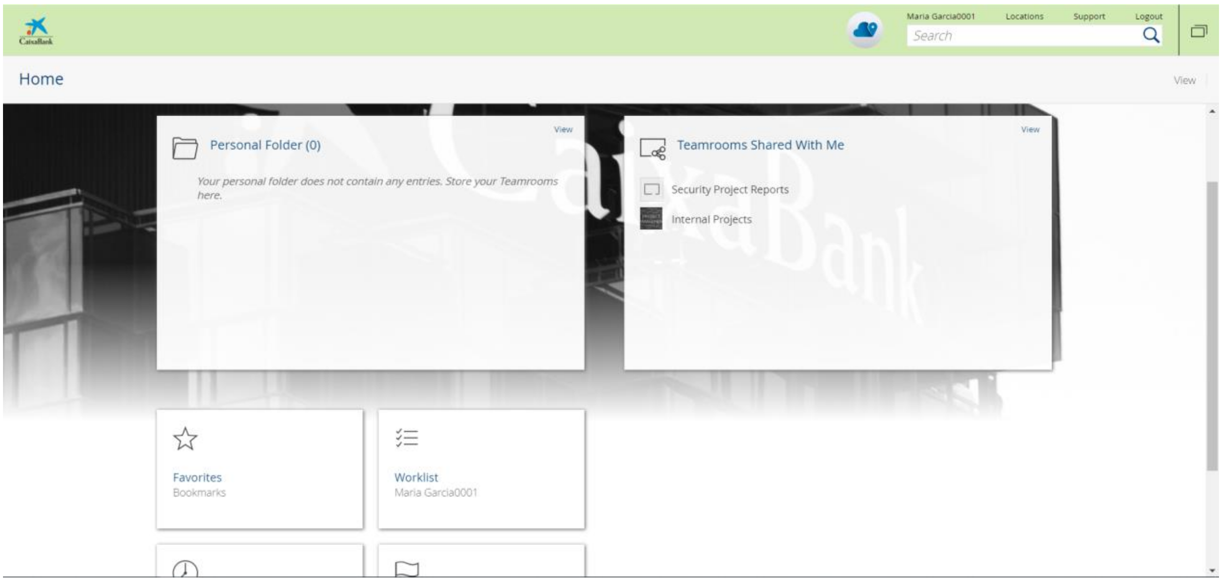
class	id	name	passed
external-object	729bd389-da21-4e0e-9ae9-4a8925cbbac2	reference to evidence of physical security	true
external-object	6c99eeb8-cb42-4caa-b46d-1ddb5da9a483	reference to evidence of documentation	true

Figure 4-14 View of evidence record sample in Nuvla

USE CASE 2

In the second use case, the SaaS FISH app approach will be tested. The objective of the financial institution employee (**FI**) is the same of use case 1: to share a report with the regulator entity and get it validated. However, the workflow of this process may vary depending on the application and the requirements. In this case, the approach with multiple approval from two different profiles (two different users from regulatory entity, **R** and **R2**) is also validated.

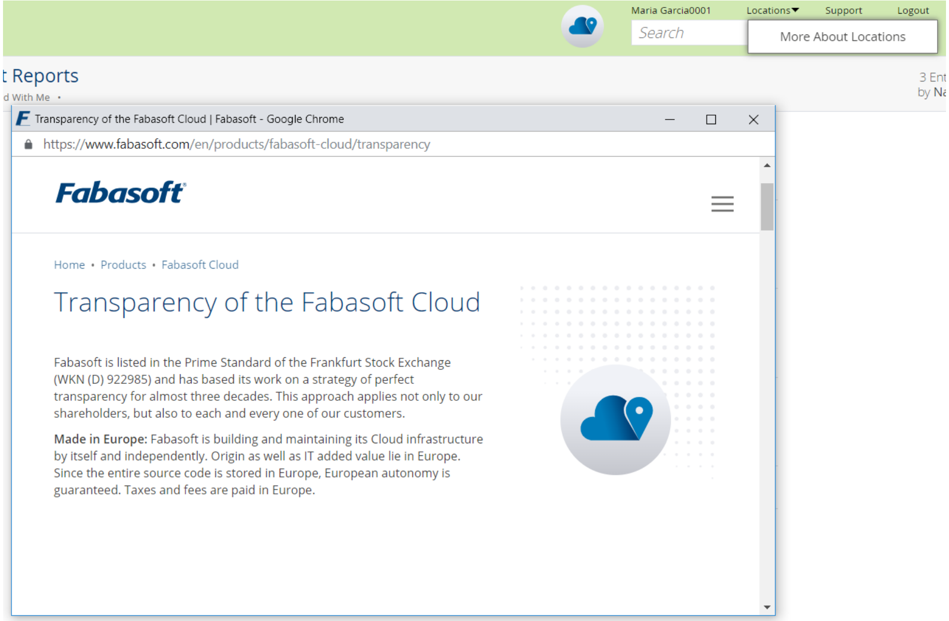
The process starts with **FI** logging in the FISH app with the credentials provided by **CSP**. The application provides the possibility of two-factor authentication configuration, according to the requirement of a strong authentication access control. **FI** enters the FISH application main window (Figure 4-15).



Your personal folder does not contain any entries. Store your Teamrooms here.

Figure 4-15 FISH SaaS application main window

FI can also check the location of the cloud service provided by **CSP**, looking at the locations of the FISH SaaS app provided by Fabasoft (Figure 4-16). The application states that all the infrastructure is based in Europe and hence, all information is stored in Europe as well, according to the mandatory Data Location requirement defined.


Maria Garcia0001
Locations
Support
Logout
Search
More About Locations
t Reports
d With Me •
3 Ent
by Na
Transparency of the Fabasoft Cloud | Fabasoft - Google Chrome
https://www.fabasoft.com/en/products/fabasoft-cloud/transparency
Fabasoft
Home • Products • Fabasoft Cloud

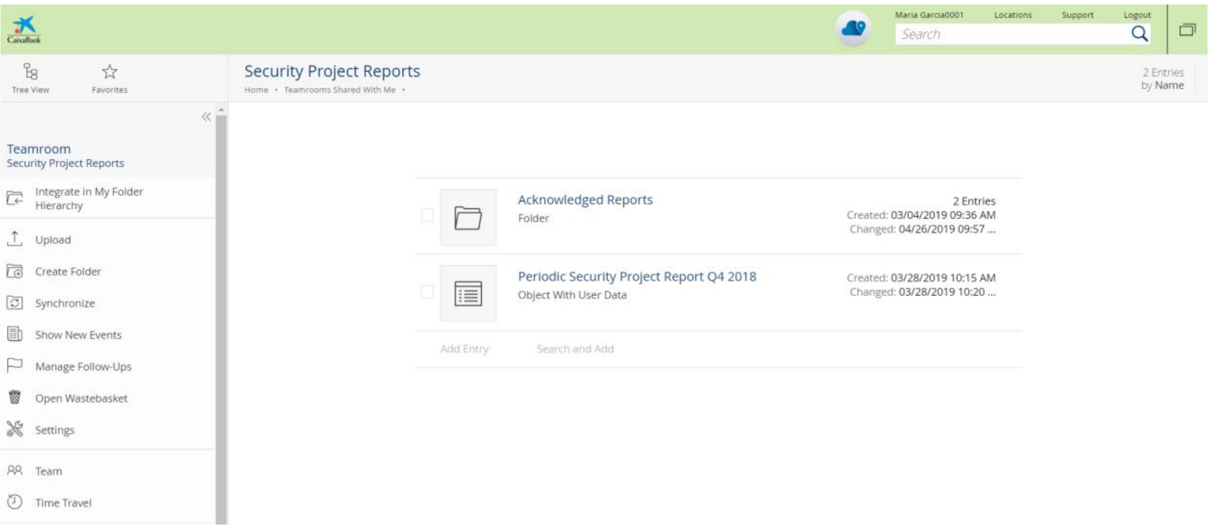
Transparency of the Fabasoft Cloud

Fabasoft is listed in the Prime Standard of the Frankfurt Stock Exchange (WKN (D) 922985) and has based its work on a strategy of perfect transparency for almost three decades. This approach applies not only to our shareholders, but also to each and every one of our customers.

Made in Europe: Fabasoft is building and maintaining its Cloud infrastructure by itself and independently. Origin as well as IT added value lie in Europe. Since the entire source code is stored in Europe, European autonomy is guaranteed. Taxes and fees are paid in Europe.

Figure 4-16 Locations information from Fabasoft FISH app

Following action by **FI** is to create a folder called "Security Project Reports" and configure it as a shared repository (Figure 4-17). FI provides reading grants to the users of "Regulatory Entites" group.

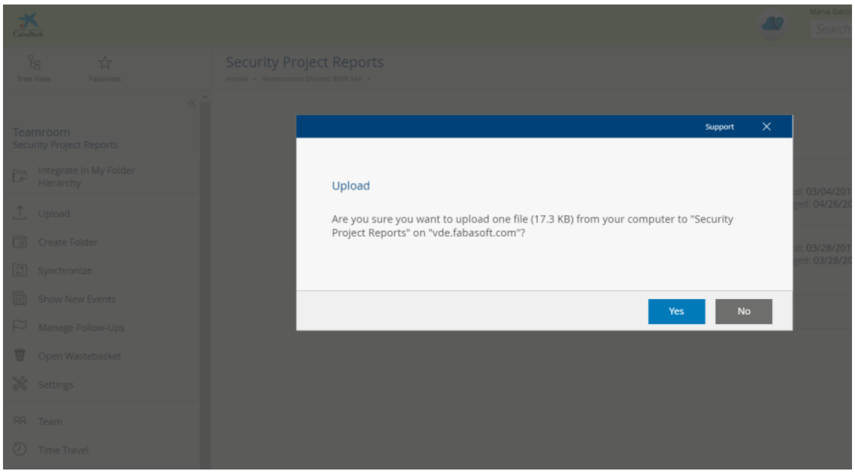


Acknowledged Reports
Folder
2 Entries
Created: 03/04/2019 09:36 AM
Changed: 04/26/2019 09:57 ...

Periodic Security Project Report Q4 2018
Object With User Data
Created: 03/28/2019 10:15 AM
Changed: 03/28/2019 10:20 ...

Figure 4-17 Creating a new folder "Security Project Reports"

At this point, **FI** can upload a new file with the report that it wants to share with the regulatory entity into the folder (Figure 4-18).

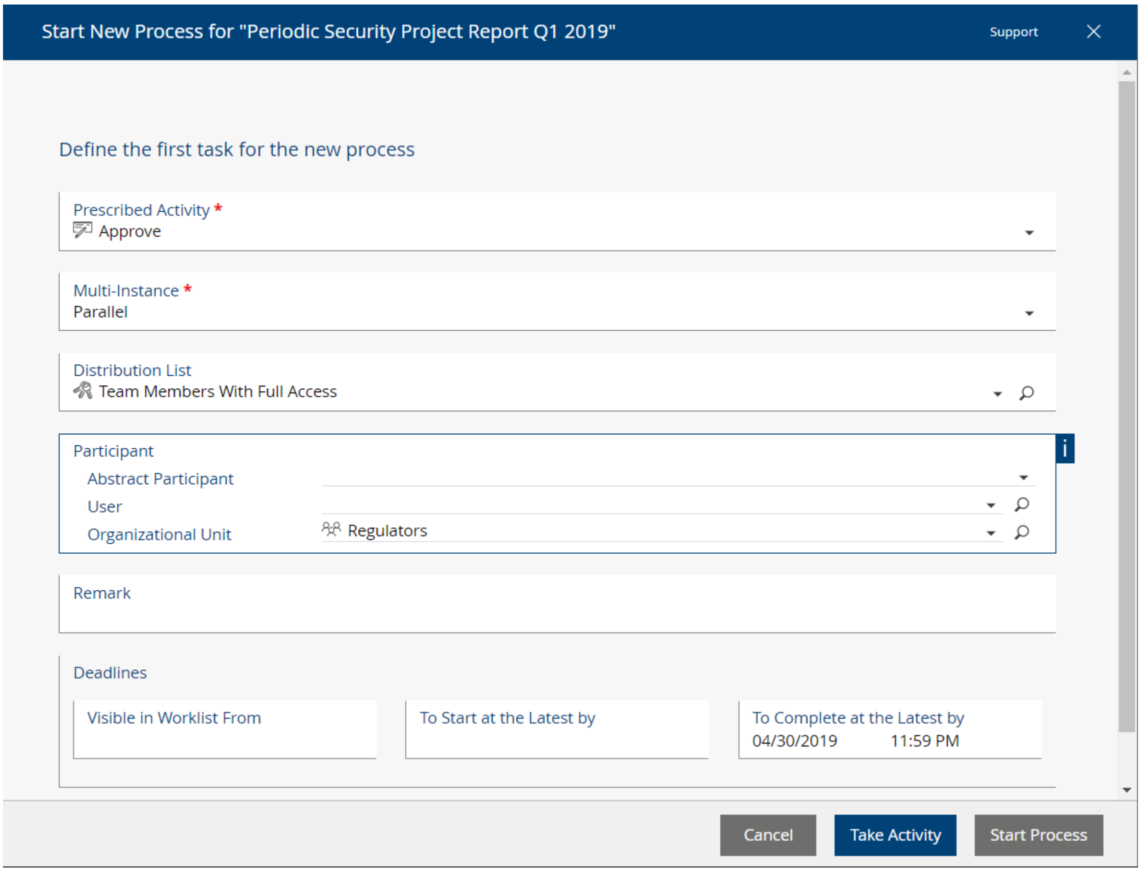


Upload

Are you sure you want to upload one file (17.3 KB) from your computer to "Security Project Reports" on "vde.fabasoft.com"?

Figure 4-18 Uploading a new file to the shared repository

By means of a right click on the file recently uploaded, **FI** opens the contextual menu and selects "Tools>Strart New Process". A new window will be opened and **FI** selects the "Approve" process, "Parallel" approval required from multiple insances, the organization unit from which the approval is needed (in this case "Regulators") and the deadline for this task (Figure 4-19).



Start New Process for "Periodic Security Project Report Q1 2019"

Define the first task for the new process

Prescribed Activity *
Approve

Multi-Instance *
Parallel

Distribution List
Team Members With Full Access

Participant
Abstract Participant
User
Organizational Unit: Regulators

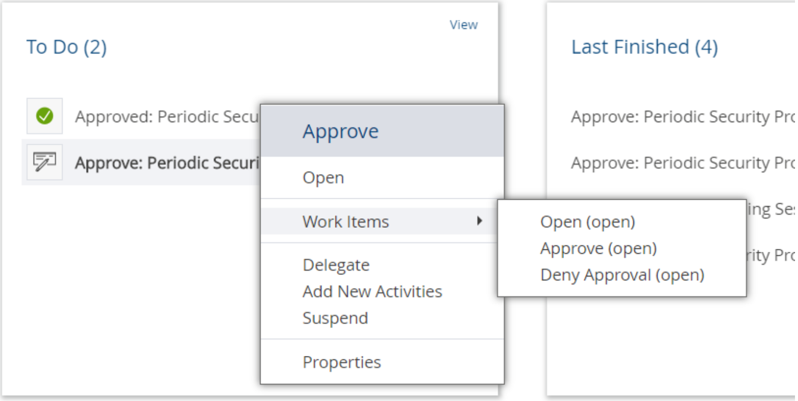
Remark

Deadlines
Visible in Worklist From
To Start at the Latest by
To Complete at the Latest by: 04/30/2019 11:59 PM

Figure 4-19 Definition of an "approval request" process

After this process is completed by the **FI**, users from “Regulators” group will receive a notification in the “To Do” list of the SaaS FISH app main window.

Subsequently **R** and **R2** enter the FISH app with their own credentials and the two-factor authentication. They select the “Approve” task to be done, first open the file report (Figure 4-20), and after their review they select to approve it.



To Do (2)

✓ Approved: Periodic Secu

Approve: Periodic Securi

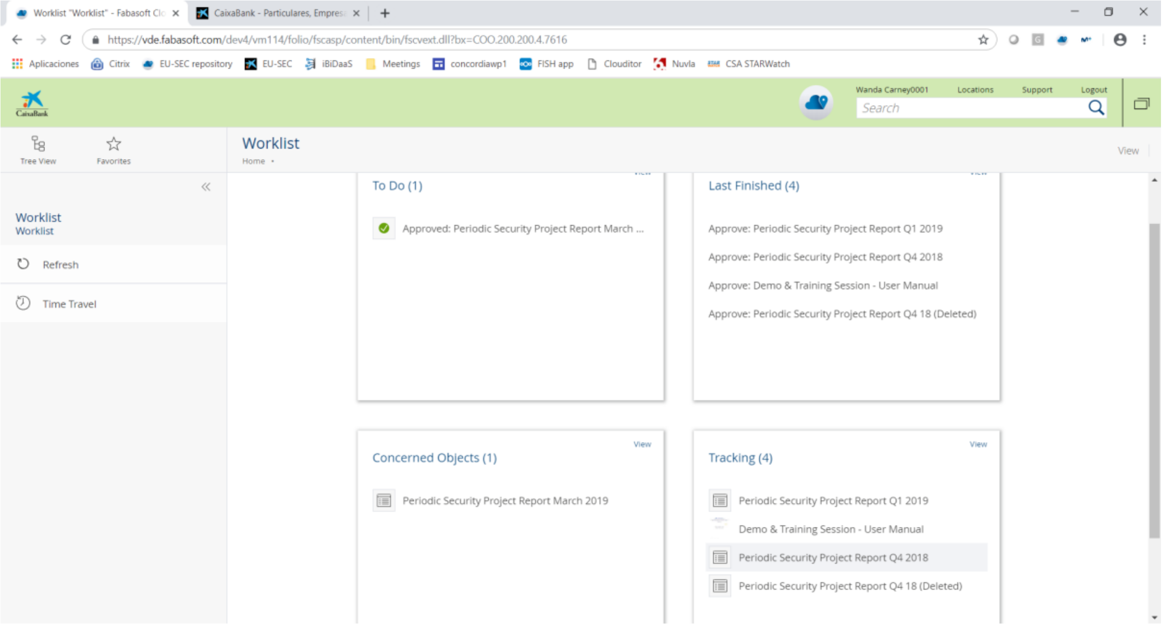
Last Finished (4)

Approve: Periodic Security Prc

Approve: Periodic Security Prc

Figure 4-20 Report approval from regulator profiles

After **R** and **R2** have accomplished the task, **FI** receive the change in the process of the documents, and it appear as approved (Figure 4-21).



https://vde.fabasoft.com/dev4/vm114/folio/fscasp/content/bin/fscvext.dll?bx=COO.200.200.4.7616

Worklist

To Do (1)

✓ Approved: Periodic Security Project Report March ...

Last Finished (4)

Approve: Periodic Security Project Report Q1 2019

Approve: Periodic Security Project Report Q4 2018

Approve: Demo & Training Session - User Manual

Approve: Periodic Security Project Report Q4 18 (Deleted)

Concerned Objects (1)

Periodic Security Project Report March 2019

Tracking (4)

Periodic Security Project Report Q1 2019

Demo & Training Session - User Manual

Periodic Security Project Report Q4 2018

Periodic Security Project Report Q4 18 (Deleted)

Figure 4-21 Notification of approved report

A enters StarWatch and accesses the assessment case of the SaaS FISH app CAC. Analogously to Use Case 1 it verifies the certification and analyses the evidence records through StarWatch first, and via Nuvla afterwards for collecting more details of any specific evidence record.

4.3 NON-FUNCTIONAL ASPECTS EVALUATION

An extended analysis of the pilot results from the business perspective will be provided in EU-SEC deliverable *D5.3 - Requirements and validation criteria – Pilot results*. However, some preliminary non-functional aspects identified during the pilot implementation and technical testing are commented below.

The pilot aimed at demonstrating that the EU-SEC framework can remove many of the compliance-related problems that prevent companies under stringent regulations to move data and computing services to the Cloud. However, one important aspect that has been identified to be consided is how a CAC process should be implemented within a company. It is undeniable that Cloud services should be audited and certified in a continuous way, and the reasons have been presented all along this document, but it is also necessary to take into account that apart from the technical tools, a formal process has to be implemented to accomplish the final goal. It should be considered that the internal process includes a formal approval from a security governance department of the financial institution who evaluates the risks and controls to be implemented, and it may require that the National Regulator approves it too. But this approval can be assigned to a Cloud service which will be changing during its time life, and the risks, controls and requirements, could change in the same way.

In addition, we note that the approach used in the EU-SEC pilot relies on the assumption that the responses provided by the CSP's information system to audir API calls are trusted. To assure this trust, we cannot escape the necessity of a traditional "point-in-time" certification, at least for the purpose of verifying that the audit API is correctly implemented. As such, CAC does not aim to replace point-in-time certification but rather to extend it in order to achieve a higher level of assurance.

Apart from the Cloud Service itself, even regulation or internal risk appetite could change as well and together with all these changes, the implementation of a correct continuous auditing should allow a convenient follow up and adaptation of the continuous audit to the Service which should be considered as a Service in continuous change. For this reason, besides the technical implementation and use of tools such as the ones presented and developed in this pilot, companies and industrial adopters have to be aware that Cloud Services have to be continued evaluated and internal procedures should be implemented to accomplish this goal. In particular, we aim at demonstrating that the EU-SEC framework can remove many of the compliance-related problems that prevent companies under stringent regulations to move data and computing services to the Cloud.

5 CONCLUSIONS

In this deliverable, the description of the CAC pilot of EU-SEC project is provided. In particular, presents the picture of the technical architecture, modules and API used in the context of the pilot.

It presents some technical recommendations about security aspects of the architecture are extracted from the experience of the pilot deployment.Furthermore, the high-level requirements and controls defined in *D5.1 – Pilot Definition* are also mapped into more concrete SLOs/SQOs attributes and metrics to be tested in an automated way by Clouditor.

Last but not least, the details of the use cases are also specified, defining the two approaches tested and showing the steps and scripts followed by the different actors in the emulation of a real scenario in which a financial information sharing application in the cloud is continuously audited and certified.

Further analysis from the pilot results from the business perspective of the different parters and external stakeholders profiles will be provided in EU-SEC deliverable *D5.3 - Requirements and validation criteria – Pilot results.*