EMERALD

Deliverable D2.4

AMOE - v1

Editor(s):	Franz Deimling (FABA)
Responsible Partner:	Fabasoft R&D GmbH
Status-Version:	Final – v1.0
Date:	31.10.2024
Type:	OTHER (SW)
Distribution level:	PU

Project Number:	101120688
Project Title:	EMERALD

Title of Deliverable:	AMOE - v1
Due Date of Delivery to the EC	31.10.2024

Workpackage responsible for the Deliverable:	WP2 - Methodology for knowledge extraction
Editor(s):	Franz Deimling (FABA)
Contributor(s):	Angela Fessl (KNOW)
Reviewer(s):	Angela Fessl (KNOW) Cristina Martínez, Juncal Alonso (TECNALIA)
Approved by:	All Partners
Recommended/mandatory readers:	WP1, WP2, WP3, WP4 and WP5

Abstract:	Interim evidence extraction from policy documents that can be integrated with the certification graph
Keyword List:	Evidence extraction, policy documents, organisational metrics
Licensing information:	This work is licensed under Creative Commons Attribution-ShareAlike 4.0 International (**CC BY-SA 4.0 DEED** https://creativecommons.org/licenses/by-sa/4.0/)
Disclaimer	Funded by the European Union. Views and opinions expressed are however those of the author(s) only and do not necessarily reflect those of the European Union. The European Union cannot be held responsible for them.

Document Description

Version	Date	Modifications Introduced	
		Modification Reason	Modified by
v0.1	04.09.2024	First draft version	Franz Deimling (FABA)
v0.2	03.10.2024	Added text and figures to "Implementation" and "Delivery and usage"	Franz Deimling (FABA)
v0.3	07.10.2024	Updated references and texts	Franz Deimling (FABA)
v0.4	14.10.2024	Updated conclusions, executive summary, future work, references, formatting	Franz Deimling (FABA)
v0.5	15.10.2024	Quality Assurance Review	Angela Fessl (KNOW)
v0.6	28.10.2024	Addressed comments of Quality Assurance review	Franz Deimling (FABA)
v0.7	29.10.2024	Final review	Cristina Martínez/ Juncal Alonso (TECNALIA)
v0.8	30.10.2024	Revised document based on the final review	Franz Deimling (FABA)
v1.0	31.10.2024	Submitted to the European Commission	Cristina Martínez/ Juncal Alonso (TECNALIA)

Table of contents

List of tables

List of figures

Terms and abbreviations

AI	Artificial Intelligence
AMOE	Assessment and Management of Organisational Evidence
API	Application Programming Interface
CPU	Central Processing Unit
CSP	Cloud Service Provider
DB	Database
DoA	Description of Action
EC	European Commission
GA	Grant Agreement to the project
GPU	Graphical Processing Unit
GUI	Graphical User Interface
ID	Identifier
HTML	Hypertext Markup Language
IaaS	Infrastructure as a Service
KPI	Key Performance Indicator
KR	Key Result
MEDINA	Predecessor project of EMERALD
NLP	Natural Language Processing
nDCG	normalized Discounted Cumulative Gain
PaaS	Platform as a Service
PDF	Portable Document Format
QA	Question Answering
RCM	Repository of Controls and Metrics
SW	Software
TRL	Technology Readiness Level
UI	User Interface
WP	Work Package

Executive Summary

This deliverable presents the initial design, architecture, and implementation state of the *Assessment and Management of Organisational Evidence* (*AMOE) component*, an evidence extractor for policy documents. The main contributions are related to the key result KR1-EXTRACT of EMERALD, a framework to continuously extract knowledge from different layers of a cloud service and prepare suitable evidence based on them.

The policy document evidence extractor, developed in Task 2.3 and described in this deliverable, aims at identifying relevant text segments related to security related features, as defined in the respective EMERALD metrics based on specific controls and security requirements of various security schemes. The extracted evidence is stored in the EMERALD *Evidence Store*. Other related deliverables in WP2, all due at project month 12 (October 2024), provide functional and technical details on further evidence extractors from different sources, i.e., D2.2 [1] on source code evidence extraction in Task 2.2, D2.6 [2] on security and privacy preserving evidence extraction in Task 2.4, D2.8 [3] on runtime data extraction in Task 2.5. All these details contributed to D2.1 [4] on the overall information model of the certification graph in Task 2.1.

This document starts by illustrating how the policy document evidence extractor fits into the overall EMERALD architecture. The main part provides functional and technical descriptions of the evidence extractor *AMOE*, including its purpose and scope, the (current and planned) coverage of the EMERALD requirements, the components' internal architecture and their subcomponents. These descriptions are complemented by information on delivery and usage, as well as on limitations and future work. Finally, the document concludes with a short summary.

Based on the work described in this deliverable, the policy document evidence extractor will be further extended and integrated into the EMERALD framework. This is the first iteration of the deliverable coming from Task 2.3. The second and final version of this deliverable (D2.5 [5]) with the updated extractor will be delivered in project month 24 (October 2025).

1 Introduction

EMERALD aims to offer a suite of tools and techniques for evidence collection, leveraging a knowledge graph-based approach. KR1-EXTRACT facilitates a unified, tool-supported methodology for continuously extracting knowledge across various layers of a cloud service—such as infrastructure, platform, runtime data, policy documents, software, and AI models.

The goal of WP2 is to develop a cohesive view of the cloud service being certified by extracting and enriching knowledge from these layers and generating relevant evidence for security metrics. A key focus of this work package is the research and design of tools and techniques to extract knowledge from diverse sources. Central to this is the *Evidence Store*, utilizing a graph-based model that acts as a common structure, populated by all evidence extraction tools with evidence[1].

1.1 About this deliverable

The goal of this deliverable is to present the EMERALD evidence extractor tool *AMOE* and how it is integrated into the EMERALD framework. This report reflects the current prototype of *AMOE*, which was originally launched in MEDINA[2]. In EMERALD, it should be advanced to a higher TRL and improved to verify that the functionality is adapted to the needs of the EMERALD pilot use cases.

EMERALD follows a knowledge graph-based approach to provide a unified view of the cloud service under certification at different layers of the service. The different evidence extraction tools are ranging from the infrastructure layer (e.g., virtual resources), to the business layer (e.g., policies and procedures), to the implementation layer (e.g., source code files), and the data layer (e.g., increasingly used AI models) in cloud applications. *AMOE* focuses on providing evidence based on policy documents which shall be included into the whole automated certification process. This deliverable will give insights into the technical and functional approach that *AMOE* uses to support the key results of the project (e.g. the use cases demonstrated by the pilots as well as the technical integration via the certification-graph and evidence extraction workflows of EMERALD).

1.2 Document structure

The document is structured as follows. In Section 2, the functional and technical descriptions of *AMOE* are described. This covers the requirements for *AMOE* in the EMERALD project as well as the *AMOE* architecture description and how it fits into the whole EMERALD architecture. Furthermore, an overview of the testing and quality management for evidence extraction is described alongside the annotation setup. Additionally, this section includes limitations and future work to be commenced regarding *AMOE*.

Section 3 focuses on the delivery and usage of *AMOE*. First the package and its contents are described, followed by installation and deployment instructions. Second, this section also provides some instructions of use, licensing information, and where to download the current public version.

The deliverable is concluded in Section 4, followed by some references in Section 5.

[1] For details consult the *AMOE* and the *Evidence Store* data model presented in deliverable D1.1 [9]

[2] https://medina-project.eu/

2 Implementation

The following subsections provide functional and technical descriptions of *AMOE*.

2.1 Functional description

Overall purpose. *AMOE* is based on a prototype developed in a previous project called MEDINA[2]. It is designed to extract evidence based on metrics, which target specific parts of policy documents. After the extraction process, the evidence can be inspected in a GUI (Graphical User Interface) that comes with *AMOE* or retrieved via the API. Once the evidence results have been reviewed by a user, they can be forwarded to the EMERALD framework.

Context, scope and motivation. *AMOE* allows to transform the organisational process of checking policy documents for their content into a technical process. Text passages can be checked against predefined goals and target values. As policy documents are rather static, compared to other evidence gathered (e.g. log files, runtime information), the evidence gathering is done once per document for a specific set of metrics and target values. In the case updates are required, the new document is processed, and additional evidence is produced. The policy evidence results are integrated into the EMERALD audit process via submission to the *Evidence Storage* component and subsequent processing in the *Assessment* component.

Requirements. The relevant requirements from D1.3 [6] with their respective implementation state (partially / fully / not implemented) and a brief description of how they are / will be implemented are given in Table 1 to Table 7.

Table 1. AMOE.01 - Upload PDF document

Field	Description
Requirement ID	AMOE.01
Short title	Upload PDF document.
Description	The component shall be able to receive a PDF document via API and process its contents regarding the defined metrics. The PDF shall receive a unique ID so that it can be retrieved and deleted later on.
Status	Work in Progress
Priority	Must
Component	AMOE
Source	Component
Type	Technical
Related KR	KR1_EXTRACT, KR2_CERTGRAPH, KR8_PILOTS
Related KPI	KPI 1.1
Validation acceptance criteria	The user can upload a document via API. The user shall be able to retrieve document meta data by using the unique id that is returned on successful upload. The process shall finish in reasonable time.
Progress	90%
Milestone	MS2: Components V1 (M12)

AMOE provides the functionality to upload a PDF document via its API. At the current implementation status, the processing is started immediately after the upload of the document is completed and a unique file ID is returned. After the metrics have been processed for the document, the extracted results can be retrieved using the unique file ID. If AMOE.05 is implemented, the processing is only done for a set of selected metrics – and started on demand (not directly after the upload).

Table 2. AMOE.02 - Provision of extracted evidence to Evidence Store

Field	Description
Requirement ID	AMOE.02
Short title	Provision of extracted evidence to Evidence Store
Description	The evidence extraction component needs to be able to forward the extracted evidence to the EMERALD Evidence Store, so it can be used for assessment and further audit processes.
Status	Work in Progress
Priority	Must
Component	AMOE, Clouditor-Evidence Store
Source	Component
Type	Technical
Related KR	KR1_EXTRACT, KR2_CERTGRAPH, KR8_PILOTS
Related KPI	KPI 1.1
Validation acceptance criteria	A user with permissions to forward evidence shall be able to use the API to submit the extracted evidence to the evidence store. The process shall finish in reasonable time.
Progress	50% (testing with EMERALD deployment is open, and might also require some additional changes)
Milestone	MS5: Components V2 (M24)

The functionality has been tested in the MEDINA[2] framework. The adjustment to the EMERALD data model and *Evidence Store* API remains to be implemented. Also, the process must be tested within the EMERALD deployment, before it can be considered fully implemented.

Table 3. AMOE.03 - Refine evidence extraction approach

Field	Description
Requirement ID	AMOE.03
Short title	Refine evidence extraction approach
Description	The evidence extraction approach should be refined to the needs of the pilots, so that the tool is able to provide relevant evidence for the metric assessments.
Status	Accepted
Priority	Should
Component	AMOE
Source	Component
Type	Technical
Related KR	KR1_EXTRACT, KR2_CERTGRAPH, KR8_PILOTS
Related KPI	KPI 1.1
Validation acceptance criteria	Users of AMOE should be able to view a documentation text or diagram showing the performance or results of the extraction approach. The performance shall be tuned to metrics and policy documents to be provided by the pilots - including information/annotations of targets in the documents.
Progress	0%
Milestone	MS5: Components V2 (M24)

The refinement of the extraction approach has not been implemented. The extraction processes need to be fine-tuned to data and metrics related to and provided by the pilots. The quality assessment and annotation process have been set up and can be used to progress and evaluate the evidence extraction approach.

Table 4. AMOE.04 - Compare results from multiple documents

Field	Description
Requirement ID	AMOE.04
Short title	Compare results from multiple documents
Description	Results from multiple policy documents shall be comparable using AMOE. A metric can be used to extract evidence from different policy documents. AMOE shall provide the results via API for a metric and given cloud service.
Status	Work in Progress
Priority	Should
Component	AMOE
Source	Component
Type	Technical
Related KR	KR1_EXTRACT, KR2_CERTGRAPH, KR8_PILOTS
Related KPI	KPI 1.1
Validation acceptance criteria	The user can retrieve the extracted evidence on the basis of a metric via API for different uploaded policy files by supplying the metric id as well as a cloud service id in the request.
Progress	70% - testing in EMERALD environment required
Milestone	MS2: Components V1 (M12)

AMOE provides an API to retrieve evidence results for a cloud service and metric id. The testing in the EMERALD environment still has to be conducted before this can be considered fully implemented.

Table 5. AMOE.05 - Select metrics per document.

Field	Description
Requirement ID	AMOE.05
Short title	Select metrics per document
Description	AMOE should offer the possibility to select some metrics before they are extracted for a document. This speeds up the processing time as metrics that are not contained in the document do not need to be checked. Also, it should be more convenient for the user, as the results are more precise and less irrelevant results need to be discarded.
Status	Accepted
Priority	Should
Component	AMOE, EmeraldUI
Source	Component
Type	Technical
Related KR	KR1_EXTRACT, KR8_PILOTS
Related KPI	KPI 1.1

Validation acceptance criteria	The user can send a set of metric ids in the upload request or before the extraction process is started to make sure only those metrics are being processed for the uploaded file.
Progress	0%
Milestone	MS5: Components V2 (M24)

The implementation of this requirement has not started yet.

Table 6. AMOE.06 - Classify document, select respective metrics (optional)

Field	**Description**
Requirement ID	AMOE.06
Short title	Classify document, select respective metrics (optional)
Description	AMOE could use document classification to pre-select some metrics based on the category, text, requirements or other features that would be of use. This could potentially, reduce the manual workload and help to provide only results for metrics that target the specific document.
Status	Accepted
Priority	Could
Component	AMOE
Source	Component
Type	Technical
Related KR	KR1_EXTRACT, KR2_CERTGRAPH, KR8_PILOTS
Related KPI	KPI 1.1
Validation acceptance criteria	The component owner is able to configure this option in AMOE if it is implemented. For the different categories, different metrics need to be defined. The validation steps need to be defined more clearly, once the requirement is implemented.
Progress	0%
Milestone	MS8: Integrated audit suite V3(M34)

The implementation of this requirement has not started yet. The requirement is considered optional and thus has lower priority compared to essential features of *AMOE* and the EMERALD framework in general, as reflected in the other requirements.

Table 7. AMOE.07 - Metric states

Field	**Description**
Requirement ID	AMOE.07
Short title	Metric states
Description	AMOE could add some internal states to the metrics. This should help to visualize the current process for every metric and role. Here is a list of metric flags that could be used: • new: the metric has been successfully extracted • extraction-failed: evidence could not be extracted • internal-started: internal auditor/compliance manager started inspecting the metric • ready-for-audit: internal auditor/compliance manager has finished with the metric, and marked it ready for auditor • revise-policy: auditor sets the metric to be revised

	• audit-finished: auditor is ok with the metric • result-outdated: automatic or manual triggered check if result is outdated
Status	Accepted
Priority	Could
Component	AMOE, EmeraldUI
Source	Component
Type	Technical
Related KR	KR1_EXTRACT, KR8_PILOTS
Related KPI	KPI 1.1
Validation acceptance criteria	The user is able to retrieve the metric state / evidence state information via API. The state should change given the defined strategy - this change could be obtained depending on the actual state on different times using the API.
Progress	0%
Milestone	MS5: Components V2 (M24)

The implementation of this requirement has not started yet. The requirement is considered optional and thus has lower priority compared to essential feature of *AMOE* and the EMERALD framework in general, as reflected in the other requirements.

2.1.1 Fitting into the overall EMERALD Architecture

Figure 1 depicts *AMOE's* connections in the overall EMERALD architecture. *AMOE* provides the functionality to add assessment results of organisational requirements/metrics to the EMERALD framework. It works with metrics from the *RCM (Repository of Controls and Metrics)* and accesses the target values from the *Orchestrator* API (metric configuration, if defined). Alternatively, the metrics can be read from a local file. Once an uploaded file is processed and the evidence is processed and confirmed by a user, it can be forwarded to the *Evidence Store* and handled further by other EMERALD components according to the evidence/assessment pipeline defined. *AMOE* provides its functionalities via an API to the EMERALD UI.

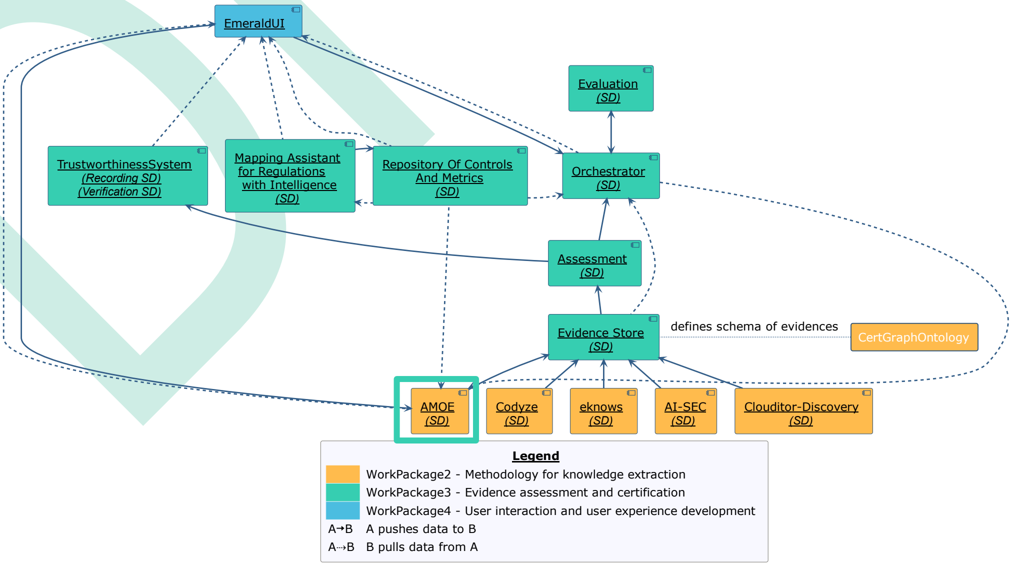

EmeraldUI
Evaluation (SD)
TrustworthinessSystem (Recording SD) (Verification SD)
Mapping Assistant for Regulations with Intelligence (SD)
Repository Of Controls And Metrics (SD)
Orchestrator (SD)
Assessment (SD)
Evidence Store (SD)
defines schema of evidences
CertGraphOntology
AMOE (SD)
Codyze (SD)
eknows (SD)
AI-SEC (SD)
Clouditor-Discovery (SD)
Legend
WorkPackage2 - Methodology for knowledge extraction
WorkPackage3 - Evidence assessment and certification
WorkPackage4 - User interaction and user experience development
A→B A pushes data to B
A⇢B B pulls data from A

Figure 1. EMERALD component overview diagram

2.2 Technical description

This section describes the technical details of the implemented software.

2.2.1 Prototype architecture

Figure 2 depicts the *AMOE* architecture. The main subcomponent is the webservice built on Quart[3]. The UI and API are the main parts hosted by the webservice. The component is connected to a Keycloak[4] instance for authentication and authorization of the users. To access data of the EMERALD framework, *AMOE* utilizes Python clients generated for the different component APIs based on their respective OpenAPI[5] files (*RCM*, *Orchestrator*, *Evidence Store*).

The MongoDB[6] is connected via the db utils wrapper. It consists of three collections, one for storing the user action log data, one for the file metadata and one for the extracted evidence. The Redis[7] instance is storing the session data required for the authentication libraries.

The solid connections in Figure 2 show the components used when deploying *AMOE*. The dashed connections (qa quality checks) reflect offline parts that are run on demand. A previous version of *AMOE*, developed in the MEDINA[8] project, has been described in [7].

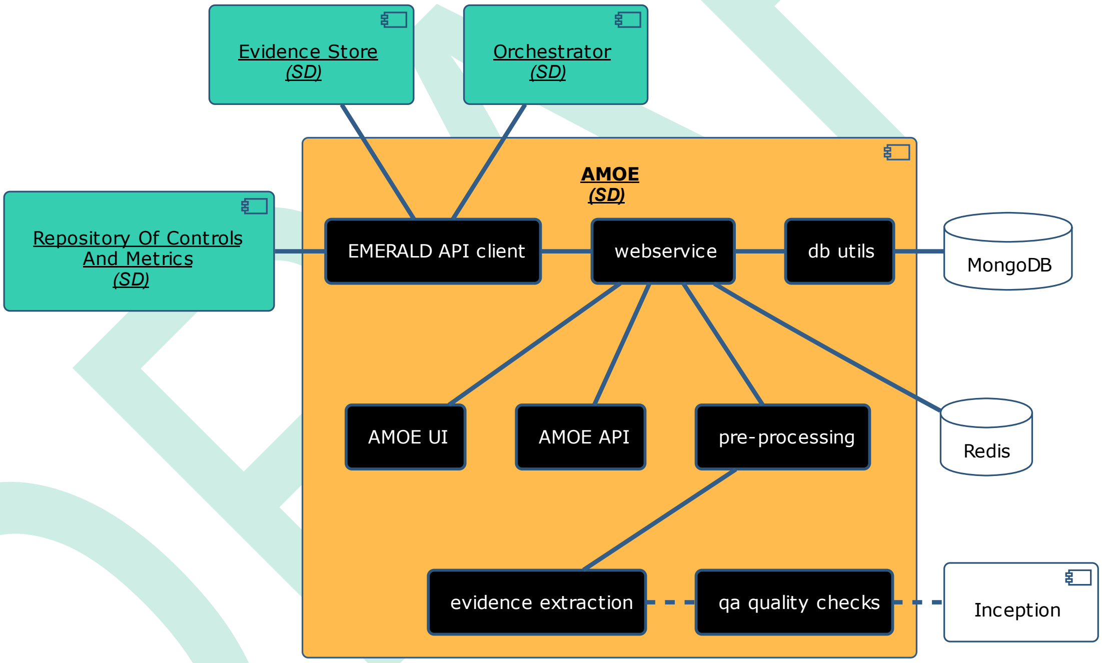


Figure 2. AMOE architecture diagram

[3] https://pypi.org/project/Quart/
[4] https://www.keycloak.org/
[5] https://www.openapis.org/
[6] https://www.mongodb.com/
[7] https://redis.io/
[8] https://medina-project.eu

2.2.1.1 Sub-components description

This section describes the sub-components of AMOE. Figure 2 depicts how the sub-components are related to each other and the lines reflect how they are related / how data flows between the sub-components.

Webservice

This is the core component redirecting the data flow to the relevant subcomponents. It serves the *AMOE* UI and API and verifies authentication via the Keycloak[4] instance from EMERALD.

EMERALD API client

This subcomponent is used to access the different components of the EMERALD framework. It is used to retrieve requirements and metrics from the *Repository of Controls and Metrics* and the metric configuration from the *Orchestrator*. Furthermore, it is used for submitting the assessment results and extracted evidence to the *Evidence Store*.

DB utils

This subcomponent is used to store and access evidence results as well as local assessment results. It is also used to log relevant information such as by whom and when a document has been uploaded or an assessment result has been changed.

AMOE UI

The graphical user interface serves to upload documents as well as to access the processed evidence. It enables the user to search, filter, and manage the organisational evidence. This UI will not be further developed in EMERALD. *AMOE* will be used mainly via its API through the *EMERALD UI*.

AMOE API

The API enables data access for other applications such as the *EMERALD UI*. It can be used to perform the most essential functions of *AMOE*. These include uploading a document, retrieving the processed evidence, setting assessment results, and submitting the evidence results to the *Evidence Store*.

Pre-processing

This subcomponent is triggered in a background process once a document has been uploaded. It performs the necessary transformations (PDF to HTML conversion, removal of header/footer, stop word removal, …) to enable the evidence extraction.

Evidence extraction

This subcomponent is triggered after the pre-processing pipeline is done. In MEDINA[9] different approaches for evidence extraction in AMOE have been tested. The keyword-based approach is currently active due to the best performance in the test results. It uses a set of predefined keywords linked to specific organizational metrics to find relevant sections of documents during a cloud audit. For each metric (e.g., password policy), keywords are identified (e.g., “password,” “age,” “maximum”), and the tool scans policy documents to extract matching sections by using the section headings. After this initial retrieval, the section text serves as input for a question answering (QA) system alongside the metric question. QA is a natural language processing (NLP)

[9] https://medina-project.eu/

task- a trained model of a QA system can provide answers to a question, given a question and text as input. The top answer of the QA is provided by the system. More details and alternative approaches tested (e.g. using cosine-similarity), are described in the paper [8].

QA quality checks

The question answering (QA) quality check subcomponent enables the user to compare the extracted evidence (using e.g. the keyword-based approach) with the annotations exported from the INCEpTION tool. See Section 2.3 for details on the quality management process.

2.2.2 Technical specifications

The *AMOE* tool is written in Python >=3.12. It uses various Python libraries as well as the pdftohtml functionality from poppler utils[10]. The webservice is built on Quart[11], the evidence extraction is based on transformers[12], PyTorch[13] and the roberta-base-squad2[14] model from huggingface.

The component is using MongoDB[15] and Redis[16] to store the data. Evidence and logs are stored in the MongoDB. Redis is used in par with the quart-session library.

2.3 Testing and quality management of evidence extraction method

To test the evidence extraction method used in *AMOE*, the extracted data is compared to some predefined target values. This allows us to compute some scores that can be used to adjust the settings of the approach. The quality management is thus split into two parts: 1) the annotation setup and 2) the execution of tests, evaluation, and analysis of the results.

2.3.1 Annotation setup

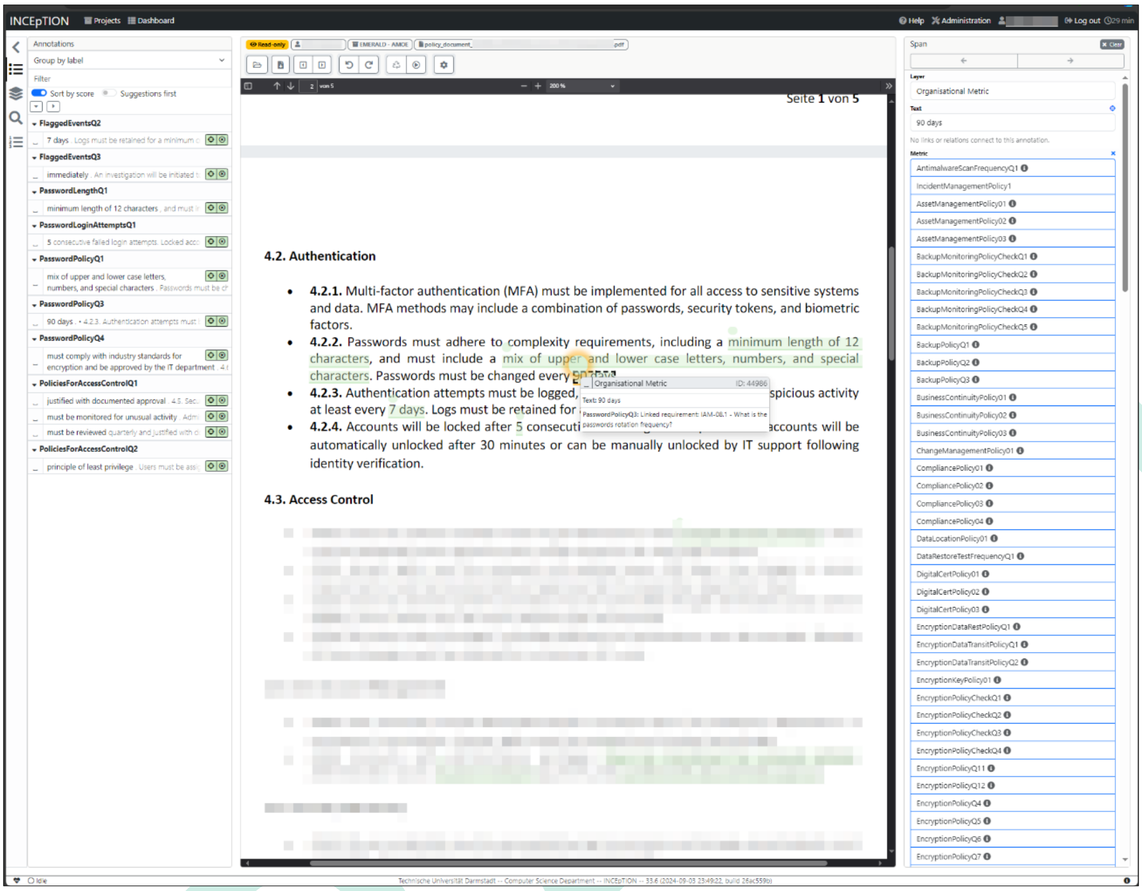
This section describes the data annotation process and preparational steps. First, the policy documents (PDFs) need to be gathered. Also, the annotation software INCEpTION[17] needs to be set up and running. Then an annotation project can be set up by adding the list of metrics to the tag set and configuring the layers to be annotated.

The tag set for the metrics can be generated using an excel list/csv containing the metric ids with the support of the python program supplied in the *AMOE* source code “`src/extract_metric_tag_list.py`”.

After the INCEpTION project is set up, the policy documents can be uploaded, and users can annotate the files. To annotate, the text must be selected and afterwards a metric can be assigned. If the metric or annotation is hovered, more information is displayed such as the description that has been configured in the tag set. Once the document has been annotated, the curation process can be applied to ensure high quality data. In the curation step, multiple annotations by different users are combined into a single source of truth, which can be exported and used for the quality assessment of the evidence extraction approach. Figure 3 depicts a

[10] https://poppler.freedesktop.org/
[11] https://pypi.org/project/quart/
[12] https://github.com/huggingface/transformers
[13] https://pytorch.org/
[14] https://huggingface.co/deepset/roberta-base-squad2
[15] https://www.mongodb.com/
[16] https://redis.io/
[17] https://inception-project.github.io/

screenshot of the annotation view in INCEpTION. In the specific case shown the annotated text samples are highlighted in green.



FlaggedEventsQ2
- 7 days . Logs must be retained for a minimum o

FlaggedEventsQ3
- immediately . An investigation will be initiated t

PasswordLengthQ1
- minimum length of 12 characters , and must i

PasswordLoginAttemptsQ1
- 5 consecutive failed login attempts. Locked acco

PasswordPolicyQ1
- mix of upper and lower case letters, numbers, and special characters . Passwords must be ch

PasswordPolicyQ3
- 90 days . • 4.2.3. Authentication attempts must b

PasswordPolicyQ4
- must comply with industry standards for encryption and be approved by the IT department . 4.

PoliciesForAccessControlQ1
- justified with documented approval . 4.5. Secu
- must be monitored for unusual activity . Admi
- must be reviewed quarterly and justified with d

PoliciesForAccessControlQ2
- principle of least privilege . Users must be assi

Seite 1 von 5

4.2. Authentication

- **4.2.1.** Multi-factor authentication (MFA) must be implemented for all access to sensitive systems and data. MFA methods may include a combination of passwords, security tokens, and biometric factors.
- **4.2.2.** Passwords must adhere to complexity requirements, including a minimum length of 12 characters, and must include a mix of upper and lower case letters, numbers, and special characters. Passwords must be changed every 90 days.
- **4.2.3.** Authentication attempts must be logged, spicious activity at least every 7 days. Logs must be retained for
- **4.2.4.** Accounts will be locked after 5 consecuti accounts will be automatically unlocked after 30 minutes or can be manually unlocked by IT support following identity verification.

Organisational Metric ID: 44986
Text: 90 days
PasswordPolicyQ3: Linked requirement: IAM-08.1 - What is the passwords rotation frequency?

4.3. Access Control

[redacted]

Layer: Organisational Metric
Text: 90 days
No links or relations connect to this annotation.
Metric:
- AntimalwareScanFrequencyQ1
- IncidentManagementPolicy1
- AssetManagementPolicy01
- AssetManagementPolicy02
- AssetManagementPolicy03
- BackupMonitoringPolicyCheckQ1
- BackupMonitoringPolicyCheckQ2
- BackupMonitoringPolicyCheckQ3
- BackupMonitoringPolicyCheckQ4
- BackupMonitoringPolicyCheckQ5
- BackupPolicyQ1
- BackupPolicyQ2
- BackupPolicyQ3
- BusinessContinuityPolicy01
- BusinessContinuityPolicy02
- BusinessContinuityPolicy03
- ChangeManagementPolicy01
- CompliancePolicy01
- CompliancePolicy02
- CompliancePolicy03
- CompliancePolicy04
- DataLocationPolicy01
- DataRestoreTestFrequencyQ1
- DigitalCertPolicy01
- DigitalCertPolicy02
- DigitalCertPolicy03
- EncryptionDataRestPolicyQ1
- EncryptionDataTransitPolicyQ1
- EncryptionDataTransitPolicyQ2
- EncryptionKeyPolicy01
- EncryptionPolicyCheckQ1
- EncryptionPolicyCheckQ2
- EncryptionPolicyCheckQ3
- EncryptionPolicyCheckQ4
- EncryptionPolicyQ11
- EncryptionPolicyQ12
- EncryptionPolicyQ4
- EncryptionPolicyQ5
- EncryptionPolicyQ6
- EncryptionPolicyQ7

Technische Universität Darmstadt -- Computer Science Department -- INCEpTION -- 33.6 (2024-09-03 23:49:22, build 26ac559b)

Figure 3. Screenshot of annotations in INCEpTION

2.3.2 Test setup

This section describes the possible tests to be commenced in the project to check the performance of the evidence extraction approach and tune it to the metrics/documents of the EMERALD project. The exported annotations (ground truth) can be compared to the results of *AMOE*. The most basic score that can be computed is based on the number of matches vs the total number of metrics annotated. The evidence extraction approach shall be evaluated by computing this score per pilot. For example, if for a document 28 metrics have been annotated in INCEpTION (#of annotated evidence) and *AMOE* retrieves the correct answer for 19 metrics (#correctly retrieved evidence), the resulting score would be ca. 0.68.

The pilot documents differ depending on the concrete use case (IaaS, PaaS, …), the language, and the formatting. Different scores will be analysed to get a good overview of the different domains and the performance overall.

The question answering model (roberta-base-squad2) can retrieve multiple answers that can be ranked by the score associated. This score can be interpreted as a kind of probability for how likely the answer is to be correct. These ranked answers can be used to compute scores like the nDCG[18] - normalized Discounted Cumulative Gain, which can help in tuning the approach to rank relevant answers higher. This score has so far not been applied for *AMOE*, as the current

[18] https://en.wikipedia.org/wiki/Discounted_cumulative_gain

implemented approaches provide a single answer. The main benefit of the nDCG will be in the improvement of *AMOE* evidence extraction approaches as some interim results can be tuned.

2.4 Limitations and future work

One key challenge is the limited availability of data, which restricts the system's ability to provide comprehensive answers across diverse domains. Additionally, language-related limitations arise, as the model may struggle to extract relevant responses in languages it has not been trained on. Privacy concerns also play a significant role, particularly since the tool avoids using pre-trained publicly available AI models due to data protection requirements, further constraining the variety and quality of data inputs. The models are selected on the basis of their license as well as local operability – to make sure, the data does not have to leave the premise. The prototype operates under limited GPU and CPU resources, which hinders the processing speed and scalability of the system, affecting its overall efficiency and performance when handling complex queries or large datasets. The processing time of the current evidence extraction approach is dependent on the input size – longer documents potentially take longer to process. However, this is somewhat mitigated by applying keywords to reduce the search space (see also [8]).

AMOE is designed to support in the assessment and management of policy documents, but not to fully automate the assessment. With the current design, no guarantee can be given that the AI models would retrieve the correct evidence 100%. Given the limited data set provided to the project by the different partners, results might be biased for specific use cases/metrics that are only relevant to some of the partners. The performance will be evaluated given the provided resources, since the focus of the project is on innovation rather than research.

At the current state of the project, not all *AMOE* requirements have been implemented. AMOE.07 lists a few possible metric states that could be added to improve the traceability and usability of metrics. Furthermore, at the current state of *AMOE* it is impossible to determine for sure whether the target of a metric is contained in a document. To limit the processing resources and time waiting for results and, most of all, to extract more precise results the implementation of AMOE.05 is planned – selecting metrics per file to avoid processing non-sensical data.

The largest remaining requirement is AMOE.03 – the improvement of the evidence extraction approach. For the remainder of the project AMOE.01, AMOE.02 and AMOE.04 will be worked on and AMOE.06 is considered optional to be addressed if enough resources are available and other requirements are completed. The *AMOE* GUI will not be updated further as it will be integrated into the EMERALD UI.

3 Delivery and usage

The following sections give a short overview of the delivery and usage of the tool.

3.1 Package information

AMOE can be deployed as a Docker container. Table 8 shows an overview of the repository folders and files.

Table 8. Overview of AMOE's source code package contents

Folder	Description
/	The root folder contains some helper scripts and configurations needed to build and run *AMOE* (e.g. Dockerfile).
/kubernetes/	Contains the Kubernetes[19] files for the deployment of *AMOE*.
/metric_data/	Contains the local version of the metrics.
/src	Contains the source code of *AMOE*
/src/paragraph_extraction/	Contains the code for the pre-processing pipeline.
/src/qa/	Contains the code for evidence extraction using the question answering model as well as code to compute quality scores.
/src/static/	Contains the stylesheets and images for the webservice.
/src/templates/	Contains the HTML templates for the webservice.
/src/utils/	Contains code for utility functions of the webservice such as use of other EMERALD component's API, evidence extraction and database management.
/tests/	Contains the tests for the *AMOE* API
/clouditor-evidence-client/	Contains the repo for the generated Python client for the *Evidence Store* API based on their OpenAPI file.
/orchestrator-client/	Contains the repo for the generated Python client for the *Orchestrator* API based on their OpenAPI file.
/rcm-client/	Contains the repo for the generated Python client for the *Repository of Controls and Metrics* API based on their OpenAPI file.

3.2 Installation instructions

Clone the *AMOE* repository. Set up a MongoDB and a Redis instance (see Kubernetes files in the repository).

Set the following environment variables or variables directly in the `config.py`:

- `MONGODB_URL`
- `MONGODB_PORT`
- `MONGODB_USER`
- `MONGODB_PASSWORD`
- `REDIS_SERVICE`
- `REDIS_PASS`

[19] https://kubernetes.io/docs/concepts/workloads/controllers/deployment/

- `REDIS_PORT`
- `KEYCLOAK_URL`
- `KEYCLOAK_REALM`
- `KEYCLOAK_CLIENT_ID`
- `KEYCLOAK_CLIENT_SECRET`
- `KEYCLOAK_USER`
- `KEYCLOAK_PASSWORD`

Optionally set (needed to deploy in production with EMERALD components):

- `CATALOGUE_API_URL`
- `ORCHESTRATOR_API_URL`
- `ALLOWED_ORIGINS`

Run `python3 -m hypercorn -b 0.0.0.0 "src.app:create_app()"` to deploy the service locally, or deploy with Kubernetes.

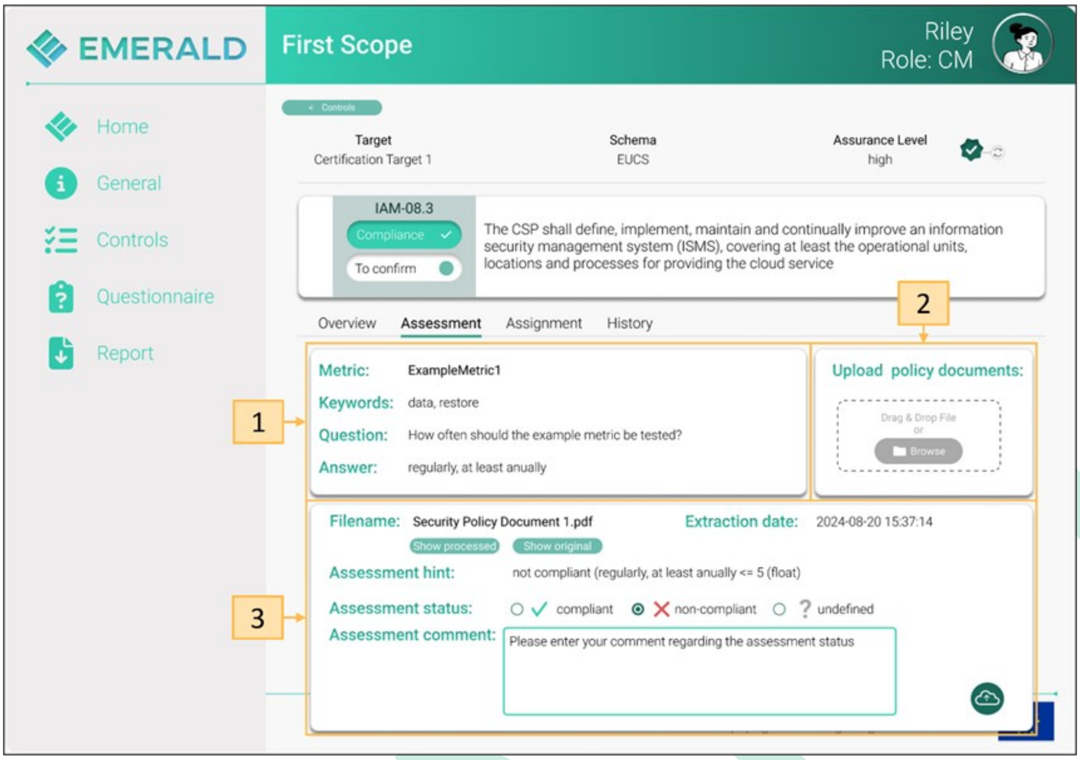
3.3 Instructions for use

Instructions for use are included in the *AMOE*'s public GitLab repository[20]. The OpenAPI documentation can be found in the code repository in GitLab as well as retrieved for every deployment by accessing *<amoe-server-url>/docs.*

Figure 4 and Figure 5 display mock-ups for the EMERALD UI containing information that is or will be provided by *AMOE*. The first depicts how the implementation in the EMERALD UI could look like to upload a policy file to *AMOE* as well as how the evidence data for a certain metric could be displayed. The second figure provides a draft for the functionality of AMOE.05 – which will allow to select a set of metrics to be retrieved for a file in *AMOE*.

For testing purposes of the *AMOE* integration in the EMERALD UI, some pages were recreated like the overview page of uploaded documents (Figure 6), a page listing the metadata of an uploaded file and processed evidences – filtered by search term "password" (Figure 7) and a page allowing to set the assessment status and compliance comment while investigating the extracted evidence (Figure 8).

[20] https://git.code.tecnalia.com/emerald/public/components/amoe-assessment-and-management-of-organizational-evidence



Target: Certification Target 1

Schema: EUCS

Assurance Level: high

IAM-08.3

Compliance ✓

To confirm

The CSP shall define, implement, maintain and continually improve an information security management system (ISMS), covering at least the operational units, locations and processes for providing the cloud service

Metric: ExampleMetric1

Keywords: data, restore

Question: How often should the example metric be tested?

Answer: regularly, at least anually

Upload policy documents:

Filename: Security Policy Document 1.pdf

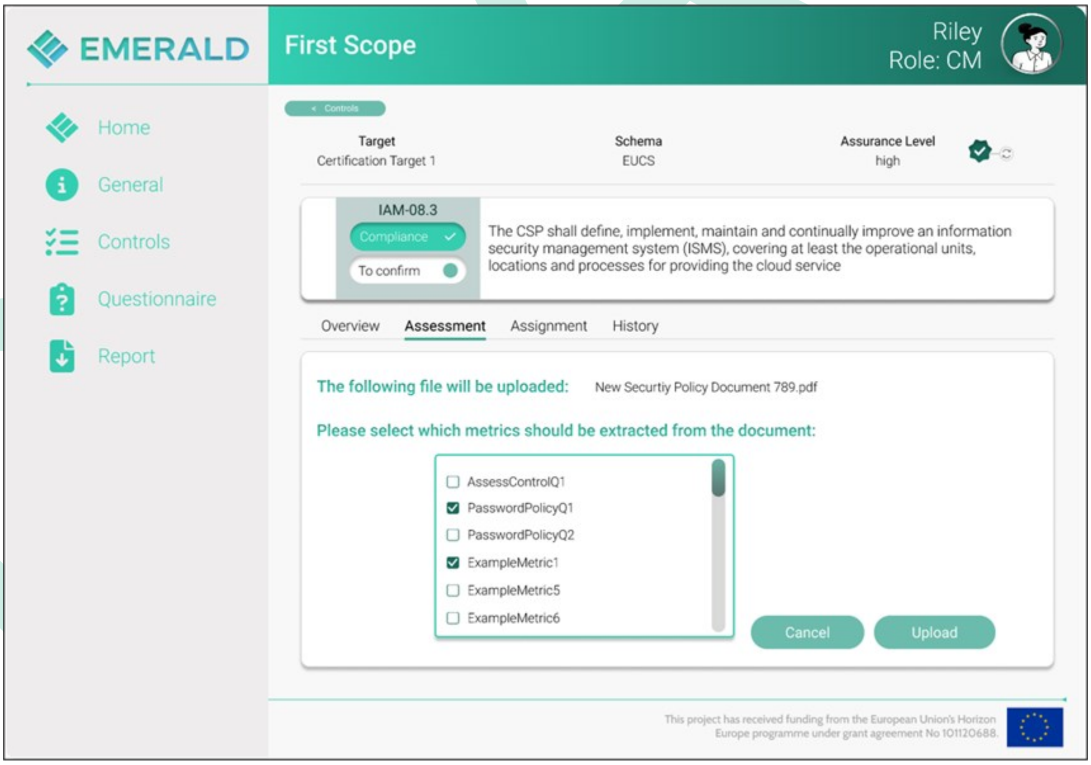
Extraction date: 2024-08-20 15:37:14

Assessment hint: not compliant (regularly, at least anually <= 5 (float)

Assessment status: ○ ✓ compliant ● ✗ non-compliant ○ ? undefined

Assessment comment: Please enter your comment regarding the assessment status

Figure 4. EMERALD UI mock-up containing AMOE evidence data (D4.3 [9])



Target: Certification Target 1

Schema: EUCS

Assurance Level: high

IAM-08.3

Compliance ✓

To confirm

The CSP shall define, implement, maintain and continually improve an information security management system (ISMS), covering at least the operational units, locations and processes for providing the cloud service

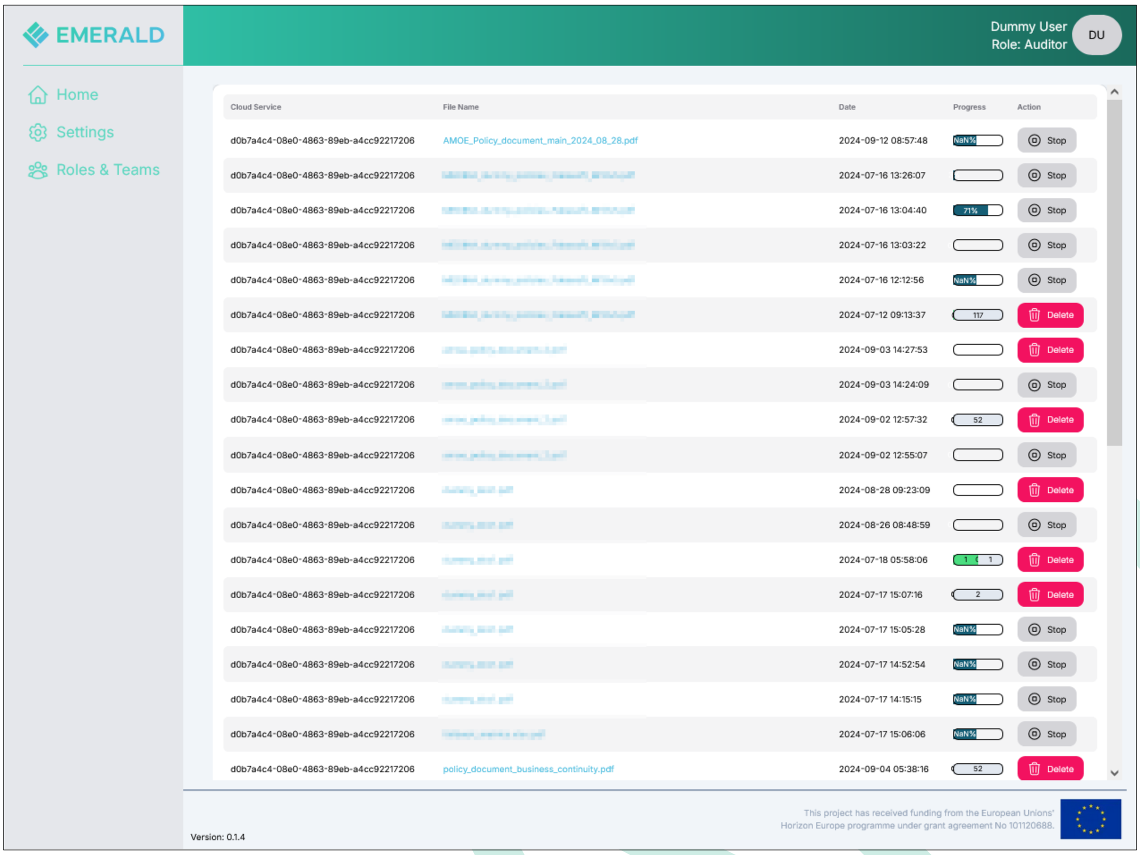
The following file will be uploaded: New Securtiy Policy Document 789.pdf

Please select which metrics should be extracted from the document:

- □ AssessControlQ1
- ☑ PasswordPolicyQ1
- □ PasswordPolicyQ2
- ☑ ExampleMetric1
- □ ExampleMetric5
- □ ExampleMetric6

This project has received funding from the European Union's Horizon Europe programme under grant agreement No 101120688.

Figure 5. EMERALD UI mock-up for AMOE.05 (D4.3 [9])

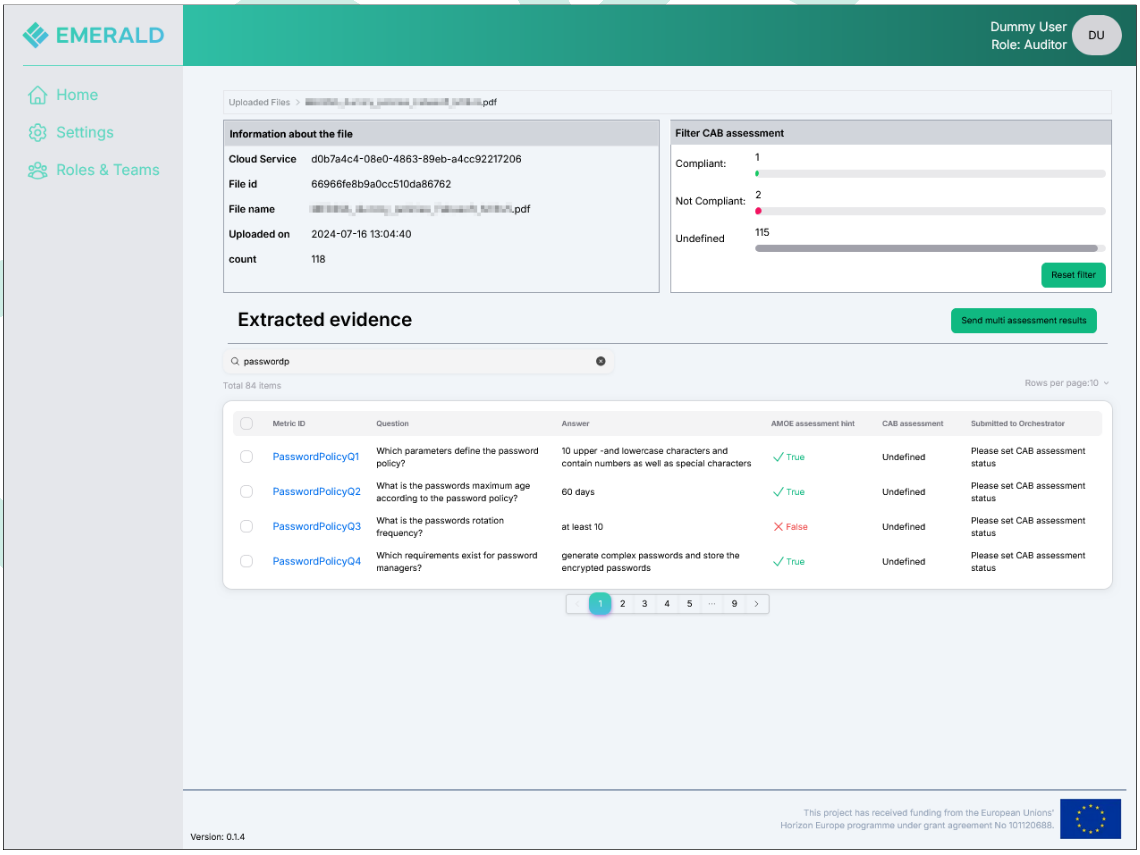


Cloud Service	File Name	Date	Progress	Action
d0b7a4c4-08e0-4863-89eb-a4cc92217206	AMOE_Policy_document_main_2024_08_28.pdf	2024-09-12 08:57:48	NaN%	Stop
d0b7a4c4-08e0-4863-89eb-a4cc92217206	[redacted]	2024-07-16 13:26:07		Stop
d0b7a4c4-08e0-4863-89eb-a4cc92217206	[redacted]	2024-07-16 13:04:40	71%	Stop
d0b7a4c4-08e0-4863-89eb-a4cc92217206	[redacted]	2024-07-16 13:03:22		Stop
d0b7a4c4-08e0-4863-89eb-a4cc92217206	[redacted]	2024-07-16 12:12:56	NaN%	Stop
d0b7a4c4-08e0-4863-89eb-a4cc92217206	[redacted]	2024-07-12 09:13:37	117	Delete
d0b7a4c4-08e0-4863-89eb-a4cc92217206	[redacted]	2024-09-03 14:27:53		Delete
d0b7a4c4-08e0-4863-89eb-a4cc92217206	[redacted]	2024-09-03 14:24:09		Stop
d0b7a4c4-08e0-4863-89eb-a4cc92217206	[redacted]	2024-09-02 12:57:32	52	Delete
d0b7a4c4-08e0-4863-89eb-a4cc92217206	[redacted]	2024-09-02 12:55:07		Stop
d0b7a4c4-08e0-4863-89eb-a4cc92217206	[redacted]	2024-08-28 09:23:09		Delete
d0b7a4c4-08e0-4863-89eb-a4cc92217206	[redacted]	2024-08-26 08:48:59		Stop
d0b7a4c4-08e0-4863-89eb-a4cc92217206	[redacted]	2024-07-18 05:58:06	1 1	Delete
d0b7a4c4-08e0-4863-89eb-a4cc92217206	[redacted]	2024-07-17 15:07:16	2	Delete
d0b7a4c4-08e0-4863-89eb-a4cc92217206	[redacted]	2024-07-17 15:05:28	NaN%	Stop
d0b7a4c4-08e0-4863-89eb-a4cc92217206	[redacted]	2024-07-17 14:52:54	NaN%	Stop
d0b7a4c4-08e0-4863-89eb-a4cc92217206	[redacted]	2024-07-17 14:15:15	NaN%	Stop
d0b7a4c4-08e0-4863-89eb-a4cc92217206	[redacted]	2024-07-17 15:06:06	NaN%	Stop
d0b7a4c4-08e0-4863-89eb-a4cc92217206	policy_document_business_continuity.pdf	2024-09-04 05:38:16	52	Delete

Version: 0.1.4

This project has received funding from the European Unions' Horizon Europe programme under grant agreement No 101120688.

Figure 6. EMERALD UI screenshot depicting a list of AMOE files



Uploaded Files > [redacted].pdf

Information about the file

Cloud Service	d0b7a4c4-08e0-4863-89eb-a4cc92217206
File id	66966fe8b9a0cc510da86762
File name	[redacted].pdf
Uploaded on	2024-07-16 13:04:40
count	118

Filter CAB assessment

Compliant:	1
Not Compliant:	2
Undefined	115

Extracted evidence

passwordp

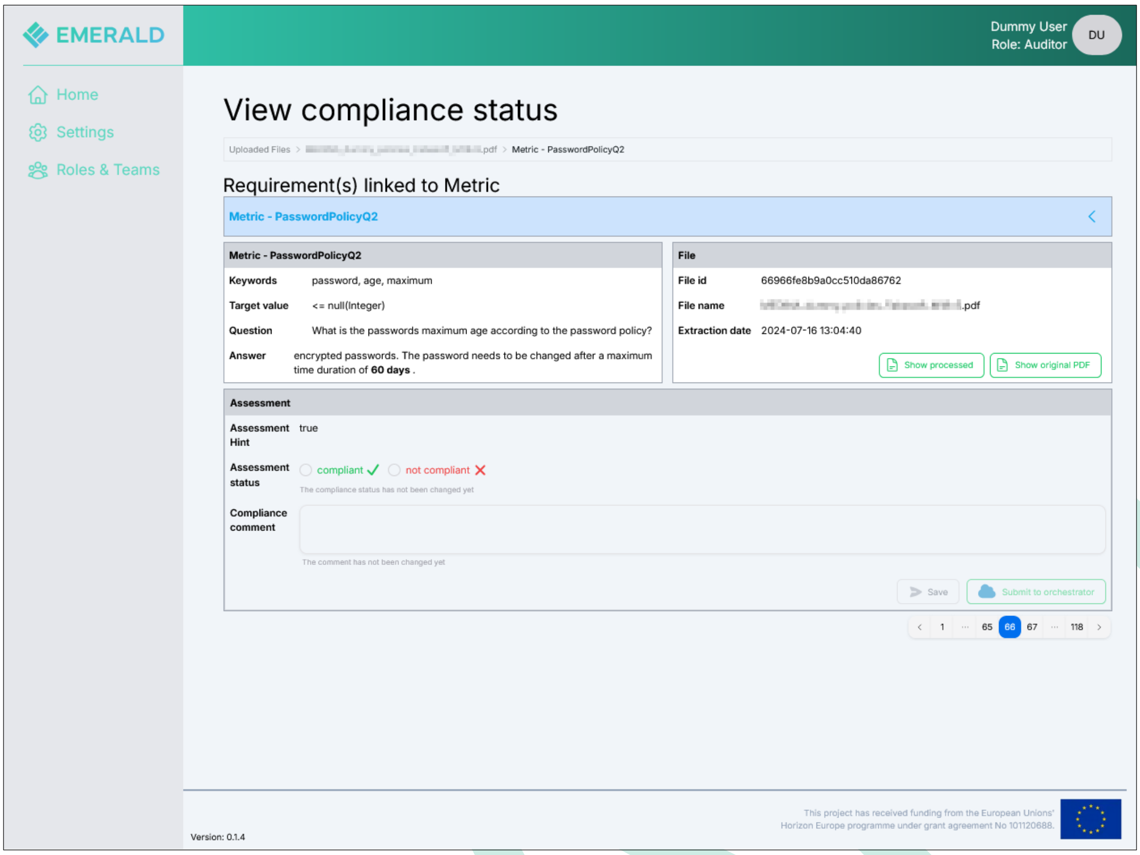
Total 84 items

□	Metric ID	Question	Answer	AMOE assessment hint	CAB assessment	Submitted to Orchestrator
□	PasswordPolicyQ1	Which parameters define the password policy?	10 upper -and lowercase characters and contain numbers as well as special characters	✓ True	Undefined	Please set CAB assessment status
□	PasswordPolicyQ2	What is the passwords maximum age according to the password policy?	60 days	✓ True	Undefined	Please set CAB assessment status
□	PasswordPolicyQ3	What is the passwords rotation frequency?	at least 10	✗ False	Undefined	Please set CAB assessment status
□	PasswordPolicyQ4	Which requirements exist for password managers?	generate complex passwords and store the encrypted passwords	✓ True	Undefined	Please set CAB assessment status

Version: 0.1.4

This project has received funding from the European Unions' Horizon Europe programme under grant agreement No 101120688.

Figure 7. EMERALD UI screenshot depicting a list of extracted AMOE evidence for an uploaded policy file



View compliance status

Uploaded Files > [redacted].pdf > Metric - PasswordPolicyQ2

Requirement(s) linked to Metric

Metric - PasswordPolicyQ2

Metric - PasswordPolicyQ2	
Keywords	password, age, maximum
Target value	<= null(Integer)
Question	What is the passwords maximum age according to the password policy?
Answer	encrypted passwords. The password needs to be changed after a maximum time duration of **60 days** .

File	
File id	66966fe8b9a0cc510da86762
File name	[redacted].pdf
Extraction date	2024-07-16 13:04:40

Assessment

Assessment Hint: true

Assessment status: ○ compliant ✓ ○ not compliant ✗

The compliance status has not been changed yet

Compliance comment:

The comment has not been changed yet

Version: 0.1.4

This project has received funding from the European Unions' Horizon Europe programme under grant agreement No 101120688.

Figure 8. EMERALD UI screenshot depicting current implementation of AMOE evidence view

3.4 Licensing information

The component is planned to be licenced under Apache 2.0.

3.5 Download

The component code can be downloaded from the EMERALD public Gitlab repository[21].

[21] https://git.code.tecnalia.com/emerald/public/components/amoe-assessment-and-management-of-organizational-evidence

4 Conclusions

AMOE is designed to extract relevant information based on EMERALD metrics from different policy documents provided by the project partners. In this deliverable, the technical report of the EMERALD evidence extraction component *AMOE* is presented. The functional description and how *AMOE* fits into the general EMERALD framework is described. Furthermore, a list of sub-components is given and how they interact.

AMOE is using the APIs of other EMERALD components - the *Repository of Controls and Metrics* to retrieve information of the metrics and security schemes, the *Orchestrator* to retrieve specific target values and metric configurations, and the *Evidence Store* to integrate the extracted results into the EMERALD framework. The clients are generated using the respective OpenAPI files.

AMOE is built using Python and different NLP libraries and pre-trained AI models. The basic requirements are under development and the functionalities are offered to the EMERALD UI via a dedicated *AMOE* API. The open requirements are planned for the upcoming project period and described in the second version of this deliverable D2.5 [5] in M24 (October 2025).

5 References

[1] EMERALD Consortium, "D2.2 Source Evidence Extractor – v1: Evidence extraction from source code that can be integrated with the certification graph," 2024.

[2] EMERALD Consortium, "D2.6 ML model certification – v1: Security and privacy preserving evidence that can be integrated with the certification graph," 2024.

[3] EMERALD Consortium, "D2.8 Runtime evidence extractor – v1: Evidence extraction from runtime data that can be integrated with the certification graph," 2024.

[4] EMERALD Consortium, "D2.1 Graph Ontology for Evidence Storage: Description of a uniform schema for storing and linking heterogenous data," 2024.

[5] EMERALD Consortium, "D2.5 AMOE–v2," 2025.

[6] EMERALD Consortium, "D1.3 EMERALD solution architecture - v1," 2024.

[7] MEDINA Consortium, "D3.6 - Tools and techniques for collecting evidence of technical and organisational measures-v3," 2023.

[8] F. Deimling and M. Fazzolari, "AMOE: A Tool to Automatically Extract and Assess Organizational Evidence for Continuous Cloud Audit," In: Atluri, V., Ferrara, A.L. (eds) Data and Applications Security and Privacy XXXVII. DBSec 2023. Lecture Notes in Computer Science, vol 13942. Springer, Cham. https://doi.org/10.1007/978-3-031-37586-6_22, 2023.

[9] EMERALD Consortium, "D4.3 User interaction and user experience concept – v1," 2024.

[10] EMERALD Consortium, "D1.1 Data modelling and interaction mechanisms - v1," 2024.