Metric Recommender System and the use of Natural Language Processing (Whitepaper)

Editor(s):	Michela Fazzolari
Responsible Partner:	Consiglio Nazionale delle Ricerche (CNR)
Status-Version:	Final – v1.0
Date:	30.09.2023
Distribution level (CO, PU):	PU

Project Number:	952633
Project Title:	MEDINA

Editor(s):	Michela Fazzolari (CNR)
Contributor(s):	
Reviewer(s):	Marinella Petrocchi (CNR), Cristina Martinez (TECNALIA)
Approved by:	All Partners
Recommended readers:	Cloud Service Providers, Auditors, IT Security-related

Keyword List:	MEDINA Cloud Certification Language, MEDINA CNL, CNL
Licensing information:	This work is licensed under Creative Commons Attribution-ShareAlike 3.0 Unported (CC BY-SA 3.0) http://creativecommons.org/licenses/by-sa/3.0/
Disclaimer	This document reflects only the author's views and neither Agency nor the Commission are responsible for any use that may be made of the information contained therein.

Document Description

Version	Date	Modifications Introduced	
		Modification Reason	Modified by
v0.1	28.08.2023	First draft version	Michela Fazzolari
v0.2	05.09.2023	Add Section 6	Michela Fazzolari
v0.3	06.09.2023	Add Section 7 and Executive Summary	Michela Fazzolari
v0.4	17.09.2023	Revision	Marinella Petrocchi
v0.5	18.09.2023	Second revision	Michela Fazzolari
v1.0	30.09.2023	Final revised version	Cristina Martinez

Table of contents

List of figures

Terms and abbreviations

API	Application Programming Interface
CNL	Controlled Natural Language
CSA or EU CSA	Cybersecurity Act
CSP	Cloud Service Provider
DB	Data Base
DCG	Discounted Cumulative Gain
DSL	Domain Specific Language
EU	European Union
EUCS	European Cybersecurity Certification Scheme for Cloud Services
GA	Grant Agreement to the project
ICT	Information Communications Technology
IDCG	Ideal Discounted Cumulative Gain
IT	Information Technologies
IUI	Integrated User Interface
nDCG	Normalized Discounted Cumulative Gain
NL	Natural Language
NLP	Natural Language Processing
NL2CNL	Natural Language To Controlled Natural Language
P@k	Precision at k
PCA	Principal Component Analysis
REO	Requirement & Obligations
TSNE	T-distributed Stochastic Neighborhood Embedding
TSVD	Truncated Singular Value Decomposition
UI	User Interface
XML	Extensible Markup Language

Executive Summary

This whitepaper provides an overview of the Metric Recommender system, which has been designed and implemented in the framework of the EU MEDINA project. This document highlights its role as a crucial component of the Cloud Security Certification Language toolchain and describes how Natural Language Processing (NLP) techniques are exploited to reach the scope.

The MEDINA project aims to establish a framework for continuous audit-based certification of Cloud Service Providers (CSPs) based on cybersecurity certification schemes. The Metric Recommender plays a pivotal role in automatically associating metrics with security requirements.

This report initially introduces the motivations behind the realisation of this system. Then, it presents the architecture and workflow of the Metric Recommender, explains its role in the MEDINA project, and discusses the data and features it utilizes. Furthermore, the document outlines the proposed solution, focusing on the k-dimensional tree approach, and discusses the performance indicators used to evaluate its effectiveness. Initial validation results, visual analysis, and performance metrics are also provided, demonstrating the system's capability to efficiently recommend metrics for security requirements.

1 Introduction

In recent years, Cloud Computing has emerged as a widespread solution for businesses seeking cost-effective and scalable IT infrastructure. The advantages brought by cloud services, such as flexibility, cost-efficiency and maintenance reduction, made them an attractive option for companies of all sizes.

Nevertheless, the adoption of cloud services also implies moving from direct control and governance of data and applications to an indirect form of control. Thus, several concerns have been raised about security, privacy, transparency and trustworthiness. To address these concerns, Cloud Service Certifications (CSCs) have become an effective solution to increase the level of trust in cloud services. CSCs provide assurance that cloud services are compliant with industry standards and best practices, and that appropriate security controls are in place to protect sensitive data and applications. The goal is to obtain a certificate that proves compliance with one or more security schemes, allowing users and customers to trust the cloud service.

Within the European context, the EU Cybersecurity Act (EU CSA)[1] gives ENISA the task of establishing a security certification scheme for ICT products, processes, and services across three levels of assurance (low, substantial and high). In this context, the MEDINA European research project[2] has been proposed, with the aim of developing a comprehensive framework that promotes customer control and trust in cloud services by supporting CSPs to achieve continuous certification in accordance with the EU CSA. This framework incorporates tools, techniques, and processes for continuous auditing and certification of cloud services with measurable security and accountability.

Continuous auditing involves the collection, analysis, and evaluation of security-related data and events to provide real-time insights into the security posture of cloud services. Continuous auditing is particularly effective when applied to analyse technical/performance aspects, which can be represented with unambiguous and measurable objectives. In most cases, technical/performance aspects are expressed through statements written in Natural Language, and therefore must be associated with measurable metrics that can be automatically assessed. Therefore, one of the first problems addressed in the context of the MEDINA project was to investigate how to render the security requirements of the chosen certification scheme, which are expressed in Natural Language (NL), into a set of measurable metrics. This is necessary because a security requirement expressed in NL is not objectively measurable, whereas metrics can be measured and then assessed automatically.

This whitepaper describes the functioning of a sub-component of the MEDINA framework, namely the Metric Recommender, which precisely aims to suggest a set of metrics useful for measuring a security requirement. Both the requirements and metrics considered in the MEDINA framework are described in NL, so this tool relies on NLP techniques to associate metrics to requirements according to their textual similarity. The remaining part of this document is structured as follows: Section 2 provides some background information. Section 3 presents a description of the problem. Section 4 describes the proposed solution. Section 5 explains the experiments done to validate the proposed solution. Section 6 highlights some benefits and drawbacks. Section 7 concludes the document.

[1] https://eur-lex.europa.eu/legal-content/EN/TXT/HTML/?uri=CELEX:52017PC0477
[2] https://medina-project.eu/

2 Background

This section provides the context and sets the stage to understand the rest of the document. The Metric Recommender is contextualized by giving an overview of the general architecture of the MEDINA framework and of the NL2CNL Translator component, which integrates the Metric Recommender.

2.1 MEDINA Architecture and Overview

The ultimate goal of the MEDINA project is to develop a framework for achieving continuous audit-based certification for Cloud Service Providers based on cybersecurity certification schemes. The definition of the MEDINA framework has been approached from different points of view. Starting with the use cases, the users of the system and the workflows they are involved in have been defined. From the functionality point of view, the whole framework has been divided into eight building blocks, as shown in Figure 1, each of them corresponding to a well differentiated functionality. The architecture of the MEDINA framework is described in [1].

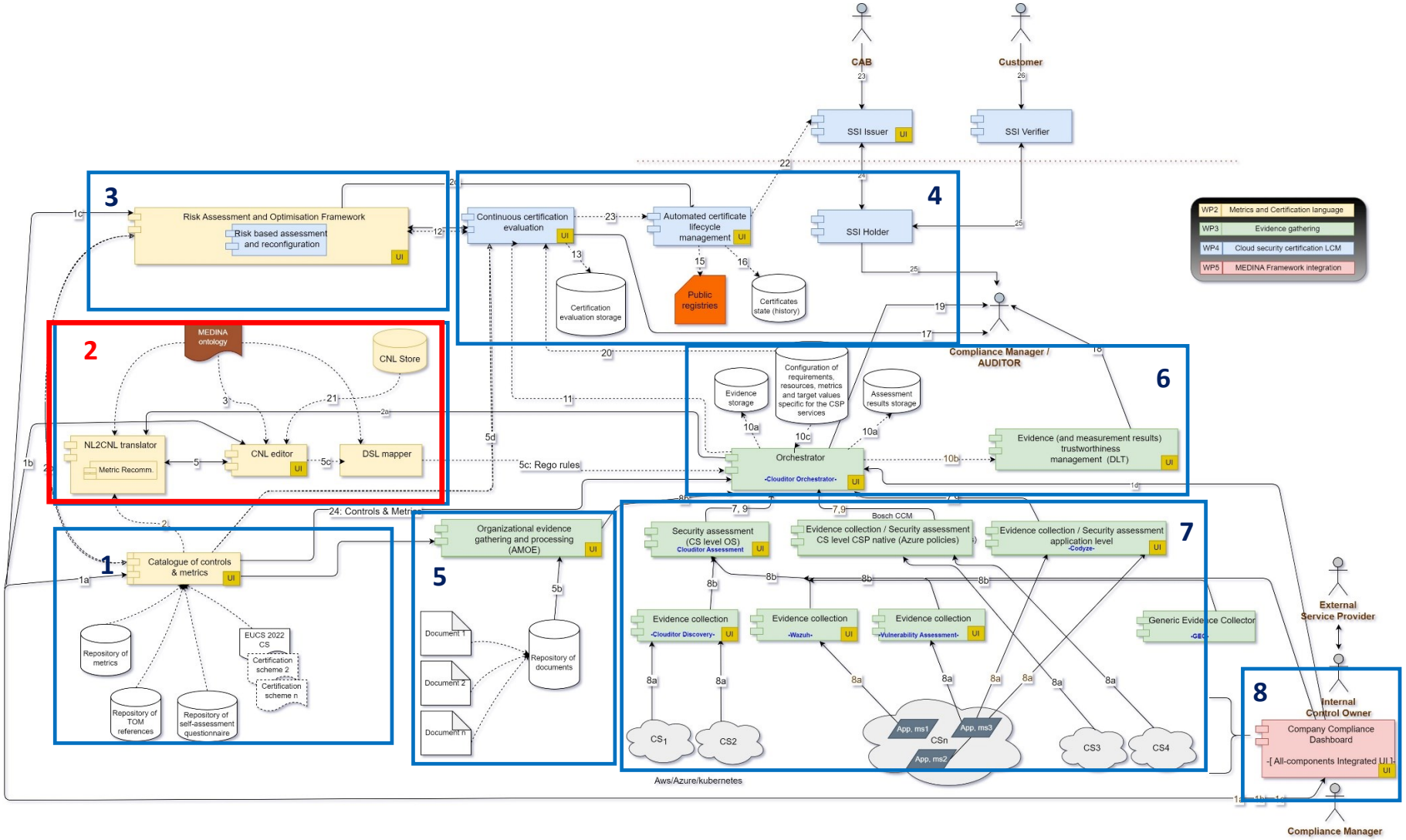

Figure 1. Building blocks view of the MEDINA framework

The Metric Recommender is a subcomponent of the NL2CNL Translator component and both of them are part of the Cloud Security Certification Language functionality, which corresponds to block 2. The Cloud Security Certification Language block involves the following tasks:

1. Definition of Security Requirements and Security Metrics relevant for the continuous certification of Cloud Service Providers (CSP).
2. Definition and implementation of a specification for Cloud Security Certification in a Controlled Natural Language (CNL).
3. Definition and implementation of a tool that allows users to modify part of the policies elicited for security controls.
4. Definition and implementation of a tool that can translate the generated CNL to an enforceable machine-oriented Domain Specification Language (DSL).
5. Development of a framework for defining the risks of the controls previously identified.

The Metric Recommender performs its functions under Task 2. For the sake of clarity, Figure 2 shows a more detailed schema of the components and architecture of block 2.

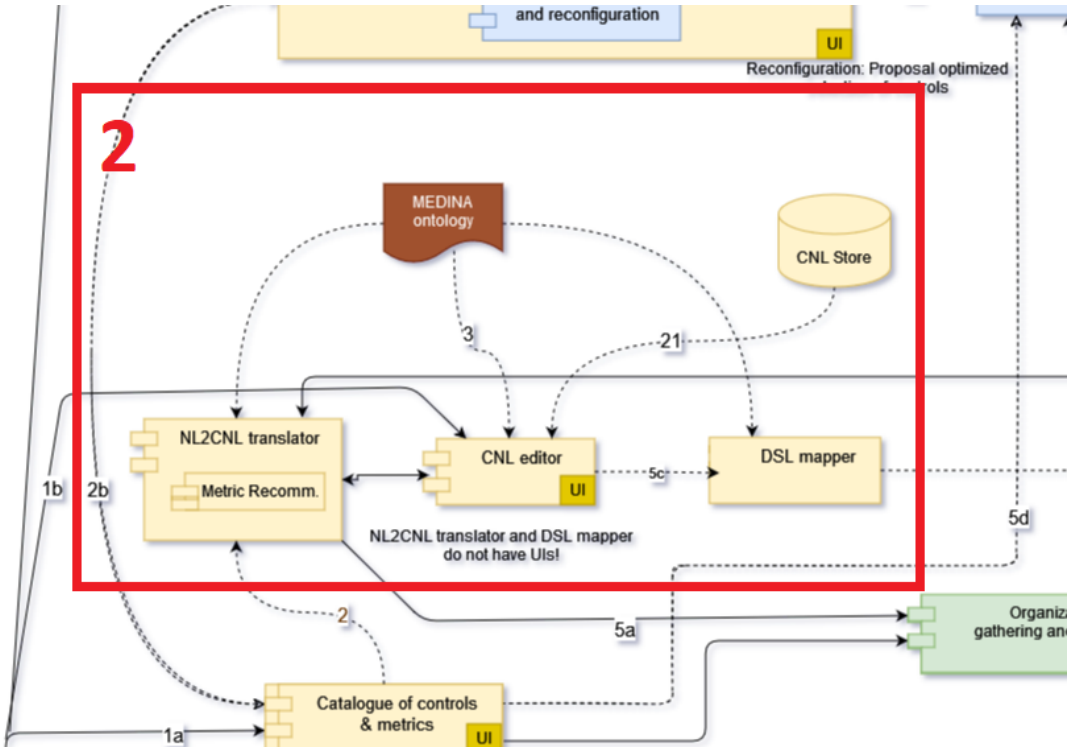

Figure 2. Architecture of the components involved in the Cloud Security Certification Language

The following are the components of the MEDINA architecture involved in block 2 that contribute to the Cloud Security Certification Language:

- The **Metric Recommender** associates a set of metrics to a requirement, by exploiting Natural Language Processing techniques. It is a sub-component of the NL2CNL Translator.
- The **NL2CNL Translator** translates EUCS NL requirements and metrics into their MEDINA CNL representation.
- The **CNL Editor** is the user interface that allows users to visualize and possibly revise the translation of the requirements and metrics into the MEDINA CNL.
- The **CNL Store** is a database that serves as output for the NL2CNL Translator and as input for the CNL Editor. It is controlled by the CNL Editor and can be accessed through its API

to read and write objects in it. The objects stored in this DB are written by the NL2CNL Translator and are read by the CNL Editor and the DSL Mapper.

- The **DSL Mapper** is the MEDINA component that maps the yet not executable MEDINA CNL into the MEDINA Domain Specific Language DSL, whose statements are instead machine-readable. Therefore, the DSL is the Cloud Certification Language.
- The **MEDINA Ontology** is used by all the Cloud Security Certification Language components for the representation of the security requirements and metrics.

The components presented so far interact with the following components in the other blocks of the MEDINA framework to perform their operations:

- The **Catalogue of Controls and Metrics** (a.k.a. Catalogue) belongs to block 1. The interaction with this component is essential since the NL2CNL Translator needs to retrieve the requirements and metrics descriptions from it. The two components connect to each other via their respective APIs.
- The **Orchestrator** begins and ends the entire translation process. In fact, the Orchestrator UI allows the user to select the requirements to be translated and to send them to the NL2CNL Translator. Finally, the result of the translation is sent to the Orchestrator itself through the DSL Mapper. Both the Orchestrator and the DSL Mapper expose an API to perform all necessary operations.

2.2 Sequence diagram

The interaction among the various components of the MEDINA framework follows predefined flows. Among them, the one that involves the components of the Cloud Security Certification Language is described in the sequence diagram depicted in Figure 3.

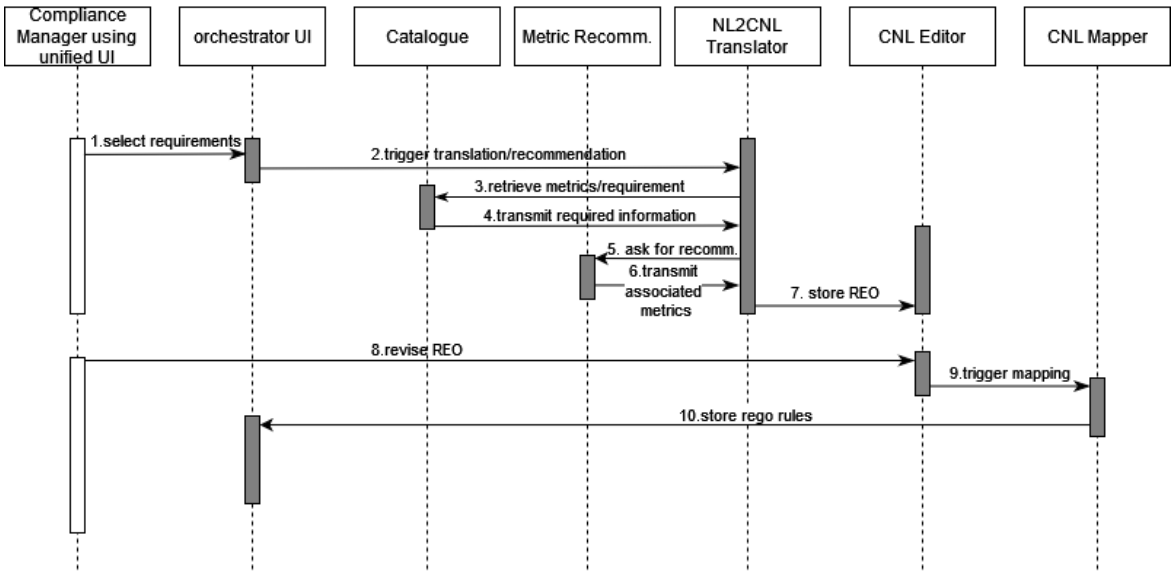

Figure 3. Sequence diagram describing the interaction among the Cloud Security Certification Language components

The sequence diagram includes the following steps:

1. The Cloud Security Certification Language functionality is activated from the Integrated User Interface (IUI) of the MEDINA framework, which allows the user to access the Orchestrator User Interface (UI) and to select a (set of) requirements to be assessed.
2. The chosen requirement is sent from the Orchestrator to the NL2CNL Translator, thus triggering the translation/recommendation.

3. The NL2CNL Translator connects to the Catalogue to retrieve information related to requirements and metrics, in particular, it retrieves, for a certain requirement, all the metrics already associated to it in the Catalogue.
4. The Catalogue responds with the information required, if available.
5. The NL2CNL Translator queries the Metric Recommender with the requirement(s) specified by the user through the Orchestrator UI.
6. The Metric Recommender returns a set of metrics associated to a requirement. In the Catalogue, each metric is already linked to a requirement or to a set of requirements and this association has been made by an expert when constructing the Catalogue. The metrics returned by the Metric Recommender are different from the ones already linked to it in the Catalogue and they are useful for having a larger list of metrics to choose from.
7. The NL2CNL Translator translates all the metrics/requirements pairs into obligations, then it builds an object containing all the information needed to describe a requirement and its associated metrics/obligations. This object is called REO (Requirement&Obligations) and it is represented in XML. The REO object is sent from the NL2CNL Translator to the CNL Editor, which takes care of storing it into the CNL Store.
8. The control returns to the IUI, giving the user the possibility to open the CNL Editor and view/revise the obligations.
9. Once satisfied with the obligations, the user can directly trigger the mapping from the CNL Editor UI, by sending the REO object to be mapped to the DSL Mapper.
10. The DSL Mapper extracts from the REO object the information needed to generate the Rego rules, which are finally sent back to the Orchestrator.

3 Problem definition

Cloud certification schemes consist of a set of rules, technical requirements, standards and procedures to strengthen the cybersecurity of ICT services and products offered to citizens, and their terms and conditions are usually published in NL. This is the case, e.g., of the EUCS draft candidate Cloud Certification Scheme [2], which includes a set of security requirements that CSPs should fulfil in order to be certified. Thus, a rigorous translation procedure is required to produce a machine-readable format out of textual NL requirements. This translation should minimise as much as possible human intervention - which is prone to errors and time consuming.

The main goal of passing from a NL representation to a CNL representation of the security requirements and associated metrics is to help achieving the automatic and continuous monitoring of the EUCS scheme.

This process is carried out through two consecutive steps. The first step consists in associating each requirement with one or more predefined metrics. In fact, to get a compliance status, each requirement needs to be objectively evaluated and thus it should be associated with metrics that reflect the technical details of the requirement itself. The second step is represented by the translation of each requirement/associated metric(s) into a policy, expressed in CNL.

As already introduced, the Metric Recommender is a subcomponent of the NL2CNL Translator tool, which implements the two steps previously described. The NL2CNL Translator relies on the Metric Recommender to perform the first step, i.e., to choose which are the metrics to be associated to a requirement. This association is done by considering the text similarity among the requirement description and the metrics descriptions, both expressed in NL.

The main source of data for the NL2CNL Translator (and thus for the Metric Recommender) is the Catalogue of Controls and Metrics, which contains requirements, metrics descriptions and metadata. The results of the translation are created according to the MEDINA Ontology and then stored in the CNL Store, a database managed by the CNL Editor.

Figure 4 depicts the simplified workflow of Task 3: first, for every requirement, a set of metrics is recommended/predicted, then the set is translated into the defined CNL.

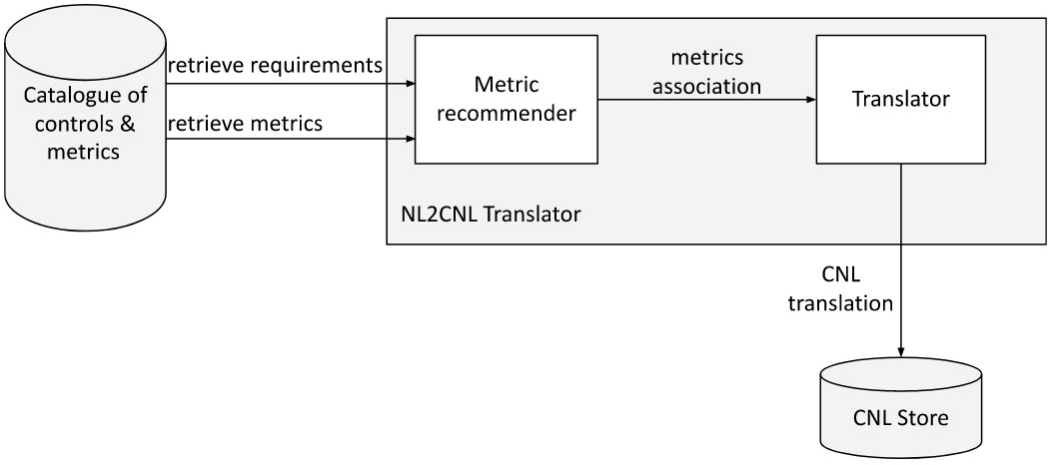

Figure 4. From Natural Language to Controlled Natural Language: Simplified overview Metric association

The following section explains the process carried out to automatically associate metrics to requirements. The idea behind this proposal comes from the fact that the metrics should be reusable when adding new requirements and the manual association between metrics and requirements is a time-consuming process.

4 Proposed solution

The Metric Recommender uses a k-dimensional tree (k-d tree) [3], which is a data structure for organizing points in k-dimensional space. The main idea behind the k-d tree is to repeatedly split the space into two regions, based on the values of the points' coordinates. The algorithm works by recursively partitioning the set of training instances based on a median value of a chosen dimension. At each level of the tree, the splitting is performed along one of the k dimensions, alternating with each level. This creates a hierarchical structure where each node represents a region of the space.

One of the main benefits of the k-d tree is its ability to efficiently search for nearest neighbours of a given point in k-dimensional space. This is achieved by recursively traversing the tree, moving towards the region of the space that is closest to the query point.

In the Metric Recommender, the points to construct the k-d tree are the metrics available in the Catalogue and the dimension are the attributes used to represent each metric. Similarly, the query data are the security requirements, which are represented with the same feature space as the metrics. In fact, both requirements and metrics are expressed in NL. The underlying idea is that the description of a requirement is similar to the description of the metric which can be useful to assess that requirement, thus the Metric Recommender associates metrics to a requirement according to the similarity of their descriptions. This is obtained by performing a nearest neighbour search on the k-d tree, using the requirement description to query the tree.

To do so, both requirements and metrics descriptions need to be represented by numerical features. The transformation between textual description and numerical representation is obtained by relying on NLP techniques, with the hypothesis that similar requirements and features descriptions will reside in the same local area in the feature space. The quality of the process is actively supervised in visual analysis and experiment results are computed using state of the art metrics. The following subsections will give more details about this process.

4.1 Data and features

The input of the metric recommender system is taken from the Catalogue of Controls and Metrics, which currently includes the requirements available in the EUCS scheme [2]. In particular, the first input is represented by the natural language description of the requirements and metrics.

Requirement input data example

The following is an example for a high-level security requirement description in natural language (ReqID=*OIS-02.3H*, category=*Organisation of Information Security*):

```
The CSP introduces and maintains a manually managed
inventory of conflicting roles and enforces the
segregation of duties during the assignment or
modification of roles as part of the role management
process. OIS-02.3H
```

Metric input data example

The following is an example for a metric description in natural language:

```
This metric is used to assess if access monitoring is
enabled
```

4.2 Experimental setup

First, the input texts are processed using a pre-trained network to produce a high dimensional feature vector. This feature vector represents the requirement or metric in a mathematically comparable way, instead of their original textual description. The basis for the recommender system is the hypothesis that requirements having similar descriptions are expected to be closer in the feature space. Since the same features are also computed for the metrics, this should also hold for the metrics. Which means, that, if the features are selected/fine-tuned to this purpose, the metrics, that should be associated with a requirement, are in the same local area in this high dimensional feature space.

4.2.1 Features

As described above, the input for the Metric Recommender system, i.e., the model that associates metrics with a requirement, consists of the computed feature vectors. The features can be computed using knowledge of the domain (e.g., for audio, spectrograms can be used). Alternatively, the features can be learned using machine learning. To train such a model, usually a lot of data is needed (in our case, text). Unfortunately, the requirements introduced in EUCS and the metrics defined in MEDINA are in insufficient numbers for a good training phase. Therefore, we use the output of pre-trained models that have worked well on state-of-the-art NLP tasks.

For our use case, two types of features are considered applicable – the ones related to context-aware models (e.g. BERT [4] and derivatives [5]) and context-free word embeddings (e.g. word2vec [6] or fastText [7]). We applied the two approaches and fastText features (on metric/requirement description text) have shown the most promising results.

The quality of the selected features directly influences the final result; therefore, some data cleaning is needed. In our case, this is mostly based on removing stop words [8]. This means that we remove words that appear often, but do not add useful information to the text.

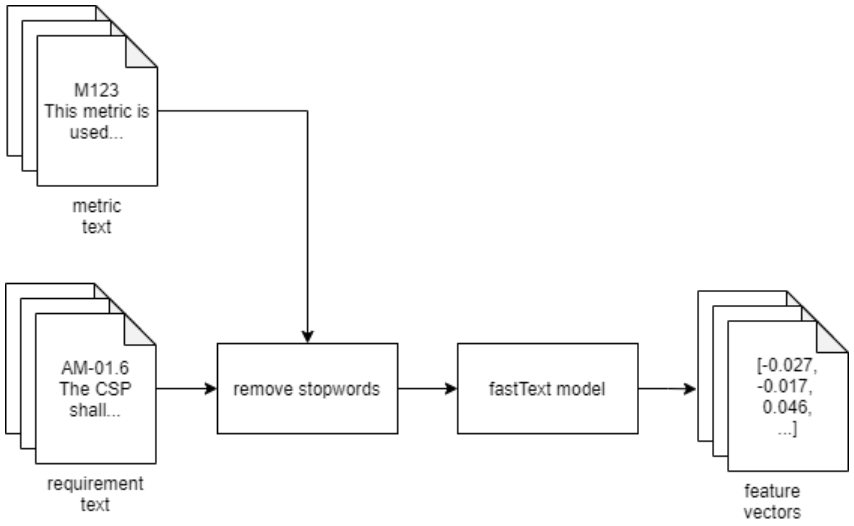

Figure 5. Feature computation workflow

4.2.2 Current approach

Features are computed using the fastText model *cc.en.300* which returns a 300-dimensional vector per requirement/metric (see Figure 5). The fastText model is pre-trained on English texts from Wikipedia and Common Crawl [9]. For visual analysis, this high dimensional vector is

reduced to two dimensions using the principal component analysis (PCA) [10] or the T-distributed Stochastic Neighbour Embedding (TSNE) [11] or the Truncated SVD (TSVD) [12].

A K-d tree is computed on the feature vectors of the metrics, which can be used to select the k closest neighbours of a query vector, based on the shortest Euclidean distance. As a query, we use the feature vector of a requirement. The workflow is depicted in Figure 6, and example results are provided in the following sections.

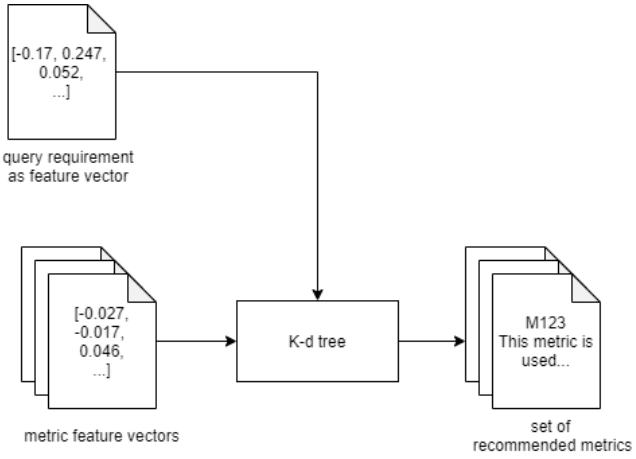

Figure 6. Recommender system workflow

4.2.3 Performance indicators

To analyse the performance of our system, basic indicators can be used. These indicators allow us to compare different approaches and help choosing the most promising one. Precision@K is typically the metric of choice for evaluating the performance of a recommender system. However, additional diagnostic metrics and visualizations can be used, since they can offer deeper insights into a model's performance. Our experimental results are based on the following indicators.

Precision@k (P@k)

A quality metric to evaluate model's performance is *precision@k* [13], e.g., *precision@5* reflects how well the system performs in the top 5 recommendation results. However, this score only considers whether the relevant metrics are in the set of retrieved metrics, without taking into account their rank. Equation 1 shows the adjusted formula to calculate the precision@k score for this task. The numerator is the amount of metrics in the intersection of *relevant metrics* (=metrics associated with a requirement) and *k* is the number of *retrieved metrics* (=recommended metrics for a requirement), the denominator is the amount of *relevant metrics*.

$$precision@k = \frac{|\{relevant\ metrics\} \cap \{k\ retrieved\ metrics\}|}{|\{relevant\ metrics\}|} \qquad \textit{Equation 1}$$

Normalised Discounted Cumulative Gain (nDCG)

An alternative quality metric to *precision@*k is the Discounted Cumulative Gain [14] (DCG), which is an indicator that can be used to measure the quality of our recommender system results, including the rank of relevant documents. The resulting metric list is ranked, and the metrics associated with the query requirement (=relevant documents) receive a relevance score $rel_i=1$, while metrics not relevant are set to $rel_i=0$. The position of the document/metric is denoted by *i*. The DCG is calculated as defined in Equation 2 To make the DCG metric comparable

to other results it needs to be normalized – therefore an ideal DCG (Equation 3) is calculated, reflecting the ideal result – all associated metrics are in the top results of the recommender. The DCG is normalized to a score between 0 and 1 using the ratio of *DCG* to *IDCG* (Equation 4).

$$DCG_p = \sum_{i=1}^{p} \frac{2^{rel_i} - 1}{log_2(i+1)} \qquad \textit{Equation 2}$$

$$IDCG_p = \sum_{i=1}^{|REL_p|} \frac{2^{rel_i} - 1}{log_2(i+1)} \qquad \textit{Equation 3}$$

$$nDCG_p = \frac{DCG_p}{IDCG_p} \qquad \textit{Equation 4}$$

5 Validation

For the purpose of testing the performances of the Metric Recommender we considered 34 EUCS requirements, which have been manually associated with 167 metrics in total. This means that these can be actively used for testing this recommender system prototype. So far, the quality of the approach was visually analysed using interactive plots and filtering of the results and measured using the above defined nDCG.

5.1 Visual analysis

Figure 7 depicts three plots using the first two components of the down-projected features. The reduction has been done using TSNE, PCA and TSVD, from left to right. Each coloured dot represents either a requirement (in blue and black) or a metric (in green). Visual analysis indicates that the features are mostly good, as the distribution of different schemes and metrics are clustered on top of each other. This empirically supports the hypothesis the recommender system is built on.

Especially in the PCA plot, it is noticeable that some metrics are skewed. Using the interactive analysis tool built for examining the data, these outliers can be directly investigated.

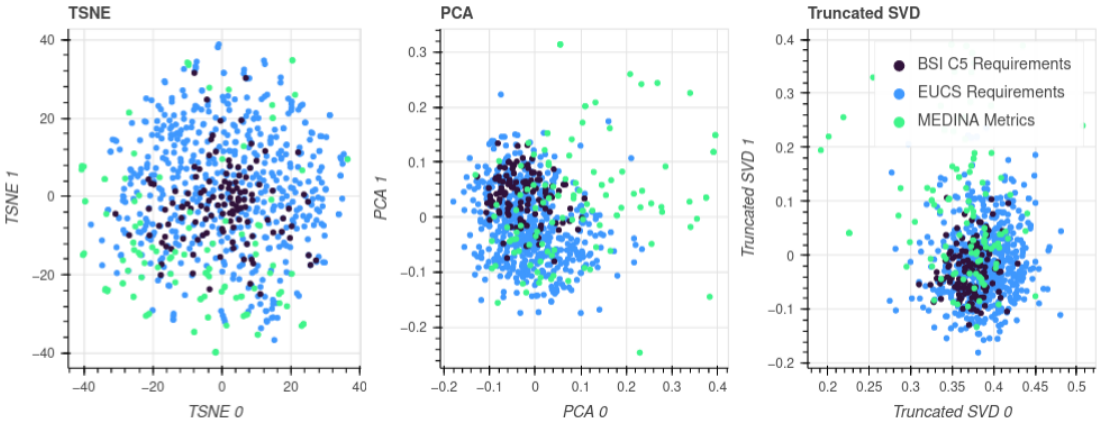

Figure 7. Plot of requirements and metrics using the first two components of the feature vectors, down-projected using TSNE, PCA and TSVD respectively

5.2 Results

For 27 out of 34 EUCS requirements, at least a subset of the linked metrics could be retrieved within the top 10 results (see Figure 8 and Figure 9). For the rest 7 requirements, no metrics could be retrieved (for an example, see Figure 10). In total the mean of the nDCG is 0.41.

Disregarding the 7 requirements, for which no metrics could be retrieved yet, the "corrected" mean nDCG is 0.66. For 12 requirements 100% of the associated metrics were retrieved in the top results. An example of the latter can be seen in Figure 8.

Query:

	ID	ReqID	EUCS Control ID	Description
85	AM-01.6	NaN	AM-01	The CSP shall automatically monitor the inventory of assets to guarantee it is up-to-date

Mapped metrics:

	ID	ReqID	EUCS Control ID	Description	relevance	rank
68	M212	AM-01.6	AM-01	This metric is used to assess if the inventory of assets is regularly monitored	1	1
69	M213	AM-01.6	AM-01	This metric is used to assess if the inventory if assets are regularly monitored against policies	1	2

Recommended:

	ID	ReqID	EUCS Control ID	Description	relevance	rank
69	M213	AM-01.6	AM-01	This metric is used to assess if the inventory if assets are regularly monitored against policies	1	1
68	M212	AM-01.6	AM-01	This metric is used to assess if the inventory of assets is regularly monitored	1	2
7	M155	HR-03.5	HR-03	Check if there is a possibility to monitor the verification of acknowledgement of security policies automatically	0	3
8	M156	HR-04.7	HR-04	Check if exists a possibility to monitor the completion of the security awareness and training program automatically	0	4
78	M222	PM-04.7	PM-04	The check that exists an automatic functionality to monitor compliance	0	5
19	M164	PSS-04.3	PSS-04	Are integrity checks of VM and container images automatically monitored?	0	6
87	M231	ISP-03.7	ISP-03	Check if security approvals and exceptions are automatically monitored	0	7
79	M223	PM-04.7	PM-04	The check that the results of the monitoring automatically use in the listed procedures: • Configuration of system components;\n• Performance and availability of system components;\n• Response time to malfunctions and security incidents; and\n• Recovery time (time until completion of error handling).	0	8
40	M184	OPS-07.2	OPS-07	This metric is used to assess if a self service portal for data backup monitoring is available.	0	9
109	M253	IAM-03.11	IAM-03	Monitoring for log events produced by automated mechanisms to check if they are working properly	0	10

DCG: 2.35. IDCG: 2.35 nDCG: 1.0

Figure 8. Prototypical results for EUCS requirement AM-01.6, optimal results on rank 1 and 2

Query:

	ID	ReqID	EUCS Control ID	Description
94	AM-03.6	NaN	AM-03	The approval of the commissioning and decommissioning of hardware shall be digitally documented and automatically monitored.

Mapped metrics:

	ID	ReqID	EUCS Control ID	Description	relevance	rank
70	M214	AM-03.6	AM-03	This metric is used to assess the existence of digital record of the commissioning requests including the approval or denial	1	1
71	M215	AM-03.6	AM-03	This metric is used to assess the existence of digital record of the decommissioning requests including the approval or denial	1	2

Recommended:

	ID	ReqID	EUCS Control ID	Description	relevance	rank
79	M223	PM-04.7	PM-04	The check that the results of the monitoring automatically use in the listed procedures: • Configuration of system components;\n• Performance and availability of system components;\n• Response time to malfunctions and security incidents; and\n• Recovery time (time until completion of error handling).	0	1
34	M178b	IM-03.4	IM-03	This metric is used to assess if the automated incident remediation mechanism requires user approvals.	0	2
74	M218	AM-04.4	AM-04	This metric is used to assess the existence of the information related to the verification of the secure configuration of the mechanisms for error handling, logging, encryption, authentication and authorisation according to the intended use and based on the applicable policies, before authorization to commission the asset can be granted	0	3
109	M253	IAM-03.11	IAM-03	Monitoring for log events produced by automated mechanisms to check if they are working properly	0	4
19	M164	PSS-04.3	PSS-04	Are integrity checks of VM and container images automatically monitored?	0	5
87	M231	ISP-03.7	ISP-03	Check if security approvals and exceptions are automatically monitored	0	6
71	M215	AM-03.6	AM-03	This metric is used to assess the existence of digital record of the decommissioning requests including the approval or denial	1	7
70	M214	AM-03.6	AM-03	This metric is used to assess the existence of digital record of the commissioning requests including the approval or denial	1	8
108	M252	AM-04.4	AM-04	No. of alerts raised for employees without or outdated acknowledgment record	0	9
10	M157b	HR-05.4	HR-05	Check if exist external employees with accesses granted after termination or change of employment, which should have been revoked according to the outcomes of the decision-making procedure	0	10

DCG: 0.94. IDCG: 2.35 nDCG: 0.4

Figure 9. Prototypical results for EUCS requirement AM-03.6, results on rank 7 and 8

Query:

	ID	ReqID	EUCS Control ID	Description
441	IM-03.4	NaN	IM-03	The CSP shall allow customers to actively approve the solution before automatically approving it after a certain period

Mapped metrics:

	ID	ReqID	EUCS Control ID	Description	relevance	rank
33	M178a	IM-03.4	IM-03	This metric is used to assess if automated incident management (detection, response) and SIEM has been enabled on a cloud service / asset	1	1
34	M178b	IM-03.4	IM-03	This metric is used to assess if the automated incident remediation mechanism requires user approvals.	1	2
89	M233	IM-03.4	IM-03	(BSI-C5 / Sim-04) Check if customers have the ability to review security incident solutions.	1	3
90	M234	IM-03.4	IM-03	(BSI-C5 / Sim-04) Check if security incident solutions are up to date.	1	4

Recommended:

	ID	ReqID	EUCS Control ID	Description	relevance	rank
13	M158	HR-05.4	HR-05	Check if access rights are revoked on contract termination or change according to the decision making procedure automatically	0	1
74	M218	AM-04.4	AM-04	This metric is used to assess the existence of the information related to the verification of the secure configuration of the mechanisms for error handling, logging, encryption, authentication and authorisation according to the intended use and based on the applicable policies, before authorization to commission the asset can be granted	0	2
10	M157b	HR-05.4	HR-05	Check if exist external employees with accesses granted after termination or change of employment, which should have been revoked according to the outcomes of the decision-making procedure	0	3
9	M157a	HR-05.4	HR-05	Check if exist internal employees with accesses granted after termination or change of employment, which should have been revoked according to the outcomes of the decision-making procedure	0	4
7	M155	HR-03.5	HR-03	Check if there is a possibility to monitor the verification of acknowledgement of security policies automatically	0	5
79	M223	PM-04.7	PM-04	The check that the results of the monitoring automatically use in the listed procedures: • Configuration of system components;\n• Performance and availability of system components;\n• Response time to malfunctions and security incidents; and\n• Recovery time (time until completion of error handling).	0	6
8	M156	HR-04.7	HR-04	Check if exists a possibility to monitor the completion of the security awareness and training program automatically	0	7
2	M92	NaN	-	percentage of relevant employees who have received training on the privacy program and policies in place.	0	8
4	M105	NaN	-	the percentage of employees who have certified their acceptance of responsibilities for activities that involve handling of private data.	0	9
87	M231	ISP-03.7	ISP-03	Check if security approvals and exceptions are automatically monitored	0	10

DCG: 0.0. IDCG: 3.7 nDCG: 0.0

Figure 10. Prototypical results for EUCS requirement IM-03.4, no results

6 Benefits and drawbacks

The main innovation realized by this system is the use of NLP techniques to automatically associate a set of metrics to a security requirement. This automation streamlines the association process, which is be time-consuming when done manually. By leveraging NLP-based Metric Recommender, the system enhances efficiency and accuracy in the translation of natural language requirements into a CNL.

Nevertheless, the obtained results may sometimes be inaccurate. This limitation arises from a partial lack of data to work with, specifically due to the project's focus on requirements with high assurance levels related to continuous monitoring (EUCS requirements). Because of this restricted focus, the system may sometimes provide inaccurate associations between metrics and requirements. As a result, manual verification of the output is necessary to ensure the correctness of the generated CNL obligations. This limitation implies that the system's performance may not be fully reliable for all types of requirements, particularly those outside the project's specific scope.

7 Conclusions

The Metric Recommender is a critical component of the MEDINA framework, contributing significantly to the automatic association of metrics with security requirements in the Cloud Security Certification Language toolchain. The system's use of a k-dimensional tree for recommending metrics based on textual similarity has shown promised, with initial validation results indicating its effectiveness in retrieving relevant metrics for security requirements.

The Metric Recommender leverages advanced Natural Language Processing techniques and relies on high-dimensional feature vectors to achieve this goal. Visual analysis of feature representations suggests that the system can successfully cluster related metrics and requirements, supporting our underlying hypothesis.

Performance evaluation, as measured by performance indicators such as normalized nDCG and precision@k, indicates that the Metric Recommender performs well in recommending metrics for a significant portion of requirements, with a mean nDCG of 0.41 across all requirements. After excluding cases where no metrics could be retrieved, the “corrected” mean nDCG increases to 0.66, and in some instances, the system successfully retrieves 100% of associated metrics within the top results.

Future work may involve refining the Metric Recommender to enhance the accuracy of metric associations and expanding the system's capabilities to handle a broader range of requirements, beyond those of high assurance levels, to extend the application of the Metric Recommender outside the MEDINA framework.

8 References

[1] M. L. C. R. I. K. F. W. F. B. M. F. &. D. G. Iñaki Etxaniz, *An architecture proposal for the MEDINA framework (Whitepaper) (1.0),* 2022.

[2] ENISA, "EUCS – Cloud Services Scheme (2022)," Draft version provided by ENISA (August 2022) - not intended for being used outside the context of MEDINA.

[3] J. L. Bentley, "Multidimensional binary search trees used for associative searching," *Communications of the ACM,* vol. 18, no. 9, pp. 509-517, 1975.

[4] "Wikipedia - BERT (language model)," October 2021. [Online]. Available: https://en.wikipedia.org/wiki/BERT_(language_model). [Accessed April 2023].

[5] J. Devlin, M.-W. Chang, K. Lee and K. Toutanova, *BERT: Pre-training of Deep Bidirectional Transformers for Language Understanding,* 2018.

[6] "Wikipedia - Word2vec," April 2023. [Online]. Available: https://en.wikipedia.org/wiki/Word2vec.

[7] "FastText," April 2023. [Online]. Available: https://fasttext.cc/.

[8] "Wikipedia - Stop word," [Online]. Available: https://en.wikipedia.org/wiki/Stop_word. [Accessed April 2023].

[9] "Common crawl," [Online]. Available: https://commoncrawl.org/. [Accessed April 2023].

[10] "Wikipedia - PCA," [Online]. Available: https://en.wikipedia.org/wiki/Principal_component_analysis. [Accessed April 2023].

[11] "Wikipedia - TSNE," [Online]. Available: https://en.wikipedia.org/wiki/T-distributed_stochastic_neighbor_embedding. [Accessed April 2023].

[12] "scikit learn Truncated SVD," [Online]. Available: https://scikit-learn.org/stable/modules/generated/sklearn.decomposition.TruncatedSVD.html. [Accessed April 2023].

[13] "Wikipedia - Evaluation meaures (information retrieval)," [Online]. Available: https://en.wikipedia.org/wiki/Evaluation_measures_(information_retrieval). [Accessed April 2023].

[14] "Wikipedia - DCG," [Online]. Available: https://en.wikipedia.org/wiki/Discounted_cumulative_gain. [Accessed April 2023].

[15] ENISA, "EUCS – Cloud Services Scheme," [Online]. Available: https://www.enisa.europa.eu/publications/eucs-cloud-service-scheme. [Accessed April 2023].