Continuous Life-Cycle Management of Cloud Security Certifications

(Whitepaper)

Editor(s):	Immanuel Kunz
Responsible Partner:	FhG
Status-Version:	Final
Date:	30.09.2023
Distribution level:	PU

Project Number:	952633
Project Title:	MEDINA

Editor(s):	Immanuel Kunz
Contributor(s):	Hrvoje Ratkajec (XLAB), Artsiom Yautsiukhin (CNR), Tatu Suhonen (NIXU), Cristina Regueiro (TECNALIA)
Reviewer(s):	Cristina Martinez (TECNALIA)
Approved by:	All Partners
Recommended readers:	Cloud Service Providers, Auditors, IT Security-related

Keyword List:	Automated Certification, Continuous Certification
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	24.07.2023	ToC	FhG
v0.2	19.09.2023	First draft of partners' contributions	FhG, XLAB, CNR, NIXU, TECNALIA
v0.3	29.09.2023	Final partners' contributions	FhG, XLAB, CNR, NIXU, TECNALIA
v1.0	30.09.2023	Final revised version	TECNALIA

Table of contents

Terms and abbreviations 5
Executive Summary 6
1 Introduction 7
2 MEDINA Overview 9
3 Certification Life Cycle Management 11
3.1 Continuous Certification Evaluation 11
3.1.1 Overall Component Description 11
3.1.2 CCE's Role in the Continuous Certification Life Cycle Management 12
3.2 Risk Assessment and Optimisation Framework 13
3.2.1 Overall Component Description 13
3.2.2 RAOF's Role in the Continuous Certification Life Cycle Management 15
3.3 Automated Certificate Life Cycle Manager 16
3.3.1 Overall Component Description 16
3.3.2 The LCM's Role in the Continuous Certification Life Cycle Management 18
3.4 Self-Sovereign Identity System for Managing Certificates 18
3.4.1 Overall Component Description 18
3.4.2 The SSI system's Role in the Continuous Certification Life Cycle Management 19
4 Discussion 20
4.1 Potentials and Limitations of Continuous Certificate Life Cycle Management 20
4.2 Continuous Certificate Life Cycle Management from the CSP's Perspective 21
4.3 Continuous Certificate Life Cycle Management from the Auditor's Perspective 21
4.4 Limitations of the Tools 22
4.4.1 Continuous Certification Evaluation 22
4.4.2 Risk Assessment and Optimisation Framework 23
4.4.3 Automated Certificate Life Cycle Manager 23
4.4.4 SSI Framework 24
5 Conclusions 25

List of Figures

FIGURE 1. ARCHITECTURE DIAGRAM OF THE MEDINA FRAMEWORK 10
FIGURE 2. AN EXCERPT OF AN EXAMPLE EVALUATION TREE REPRESENTING (NON-)CONFORMITIES OF STANDARDISATION HIERARCHY ELEMENTS 11
FIGURE 3. CONTINUOUS CERTIFICATION EVALUATION: DIAGRAM OF INTERACTION WITH RELATED COMPONENTS 12
FIGURE 4. PROVISIONING OF INPUT TO SATRA AND CONSUMING ITS RESULTS. 14
FIGURE 5. COMPUTATION OF RISK DURING THE DYNAMIC RISK ASSESSMENT 15
FIGURE 6. NON CONFORMITY ANALYSIS 15

FIGURE 7. THE DECISION FACTORS THAT ARE TAKEN INTO ACCOUNT BY THE LIFE CYCLE MANAGER: OPERATIONAL EFFECTIVENESS DATA IS RETRIEVED FROM THE CCE, TIMING RULES ARE HARDCODED IN THE LCM, WHILE A SERVICE'S RISK VALUE IS CALCULATED BY RAOF 16
FIGURE 8. THE STATE MACHINE THAT ENCODES THE EUCS-DEFINED CERTIFICATE STATES AND THEIR TRANSITIONS 17
FIGURE 9. SSI FRAMEWORK HIGH LEVEL ARCHITECTURE 18

Terms and abbreviations

API	Application Programming Interface
AWS	Amazon Web Services
CAB	Conformance Assessment Body
CISO	Chief Information Security Officer
EUCS	European Cybersecurity Certification Scheme for Cloud Services
CI/CD	Continuous Integration / Continuous Delivery
CNL	Controlled Natural Language
CSA or EU CSA	EU Cybersecurity Act
CSP	Cloud Service Provider
CSPM	Cloud Security Posture Management
DSL	Domain Specific Language
DLT	Distributed Ledger Technology
EC	European Commission
EUCS	European Cybersecurity Certification Scheme for Cloud Services
GUI	Graphical User Interface
HTTP	Hypertext Transfer Protocol
ICO	Internal Control Owner
IoT	Internet of Things
JWT	Json Web Token
KPI	Key Performance Indicator
NCB	National Certification Body
NLP	Natural Language Processing
OPA	Open Policy Agent
PaaS	Platform as a Service
PII	Personally Identifiable Information
RKE	Rancher Kubernetes Engine
SaaS	Software as a Service
SATRA	Self-Assessment Tool for Risk Analysis
SSI	Self-Sovereign Identity
SSO	Single Sign-On
ToC	Target of Certification
TOM	Technical and Organizational Measure
UI	User Interface
VM	Virtual Machine

Executive Summary

Cloud computing has witnessed rapid growth in adoption over the last decade, with prominent public cloud vendors like Amazon Web Services, Microsoft Azure, and Google Cloud offering enticing benefits such as cost savings, efficiency, and reduced security responsibilities. However, despite these advantages, cloud adoption remains constrained by the essential factor of trust. To fully leverage cloud services, users must trust cloud providers with the security and confidentiality of their sensitive data. In response to this challenge, cloud security standards have been introduced. These certifications aim to assure users that cloud providers adhere to robust security standards. However, managing these certifications becomes a complex task due to the dynamic nature of cloud systems, necessitating continuous and automated assessment. This whitepaper explores the challenge of managing cloud security certifications automatically and the complexities involved in deciding certification statuses through automation.

This whitepaper focuses on the final parts of the MEDINA pipeline, i.e., the components that aggregate and evaluate assessment results, aggregate decisive data and translate them into a certificate status, and which publish and secure the certificate.

We provide a comprehensive description of how MEDINA addresses the challenge of continuously and (semi-)automatically managing certificates and their life cycle. To this end, we first describe the MEDINA framework as a whole and then we go into the details of the components that are responsible – directly or indirectly – for the continuous management of certificates.

Also, the whitepaper covers a detailed discussion on the benefits and limitations that a continuous, automated life cycle management of cloud security certifications implies, for example regarding the standardization of life cycle management and the false positive results it can produce. We approach this discussion from the auditor's perspective as well as from the CSP's perspective.

In summary, the automated, continuous life cycle management of certificates, e.g., based on the EUCS, holds great potentials as it enables a standardized, transparent management process which also allows auditors to create new business models. At the same time, it involves risks as it is difficult to create an automated process that is reliable, secure, and precise.

1 Introduction

Cloud computing has changed significantly how organizations manage and provide their data and services to customers. Major public cloud vendors, like Amazon Web Services[1], Microsoft Azure[2], and Google Cloud[3], have played a pivotal role in the widespread adoption of cloud technology, offering compelling advantages in terms of cost-effectiveness, operational efficiency, and offloading security responsibilities.

Despite the potential benefits, the adoption of cloud services remains contingent on establishing trust in cloud providers' security practices and data handling capabilities. Organizations, private users, and government agencies seek assurances that their information is secure and will not be compromised. This trust becomes an indispensable factor in making informed decisions regarding cloud migration.

To address this issue, cloud security certifications have been introduced. They act as a mechanism to instil confidence cloud users, assuring them through independent audits that certified providers have implemented appropriate security controls and best practices.

Various cloud security standards and certification schemes have been established to promote trust and transparency in cloud services. Notable examples include BSI C5[4] and ANSSI NumSecCloud[5]. The European Union Agency for Cybersecurity's (ENISA) EU Cloud Security Certification Scheme (EUCS) aims at providing a unified, EU-wide standard[6]. These standards provide a framework for cloud providers to demonstrate compliance with specified security requirements and regulations. By obtaining these certifications, cloud providers aim to assure users of their commitment to maintaining a robust security posture.

However, managing and maintaining certifications in a constantly evolving cloud environment poses unique challenges, necessitating the adoption of automated and continuous assessment processes. Cloud environments are characterized by their dynamic nature, with frequent changes, updates, and enhancements being the norm. Traditional manual audit-based certification management approaches are not sufficient to cope with the ever-changing cloud landscape. The need for continuous monitoring and assessment has propelled the adoption of automated certification management processes. Automated certification management systems must be capable of real-time monitoring, analysis, and decision-making. The complexity arises from the necessity of not only determining the current compliance status but also predicting future certification statuses based on the cloud system's evolution. Making informed decisions on certification status automatically demands sophisticated algorithms and models that can assess a vast array of security controls and requirements.

MEDINA[7] tackles these challenges by combining a modular framework of components for evidence collection, assessment, aggregation, as well as risk assessment and life cycle management.

[1] https://aws.amazon.com
[2] https://azure.microsoft.com
[3] https://cloud.google.com/
[4] https://www.bsi.bund.de/EN/Themen/Unternehmen-und-Organisationen/Informationen-und-Empfehlungen/Empfehlungen-nach-Angriffszielen/Cloud-Computing/Kriterienkatalog-C5/kriterienkatalog-c5_node.html
[5] See, for example: https://www.ssi.gouv.fr/actualite/lanssi-actualise-le-referentiel-secnumcloud/
[6] https://www.enisa.europa.eu/publications/eucs-cloud-service-scheme
[7] See other publications and general information about the MEDINA project on the project website: https://medina-project.eu/

In this whitepaper, we focus on the final steps of the MEDINA pipeline: The aggregation of assessment results, the risk assessment of a cloud service based on the current assessment results, and the certificate status decision. We also look at how the certificate is issued securely and discuss the implications for CSPs and auditors of managing certificates automatically.

Section 2 first gives an overview of the complete MEDINA framework, providing more context for the understanding of the rest of the whitepaper. Section 3 introduces the different components that form part of the continuous management of certificates: The Continuous Certification Evaluation (Section 3.1), the Risk Assessment and Optimisation Framework (Section 3.2), the Automated Certificate Life Cycle Manager (Section 3.3), and the Self-Sovereign Identity system (Section 3.4). In a discussion section (Section 4), we then discuss the potentials and limitations of an automated, continuous certificate life cycle management from different perspectives.

2 MEDINA Overview

The design of the MEDINA framework has been approached from multiple perspectives. These involve use cases, defining the system's users and the workflows they engage in, a common data model, as well as functional components, each tailored to specific tasks. The following provides an overview of the MEDINA framework.

Figure 1 shows an overview of the complete framework. It distinguishes the components developed in the main technical work packages as follows:

- The yellow components form part of work package 2 which is concerned with developing certification metrics and specification languages. Components in this context include language editors and mappers that are able to translate (semi-)formalized certification requirements into machine-readable code that can be used to assess evidence automatically.
- The components depicted in green have been developed in work package 3 whose purpose is to create methods and tools for the continuous collection, assessment, and management of certification evidence from various sources, such as cloud resources, software assessments, and others. Note that the assessment components use the machine-readable code that is provided by the work package 2 components.
- Finally, the blue components are part of work package 4 which aims at automating the certification life cycle. These are described in more detail in the following sections.

The framework is largely composed of open-source implementations[8] which can freely be adopted and modified.

A core tool in MEDINA is Clouditor. Clouditor is an open-source tool[9] in itself and comprises three modules that have been adapted for use in MEDINA. They are briefly explained in the following (and can also be seen in Figure 1).

- The Cloud Evidence Collector discovers existing resources in a cloud system, such as Microsoft Azure or Amazon Web Services (AWS), and retrieves detailed information about them.
- The Security Assessment obtains the information about existing resources and assesses it.
- The Orchestrator is a central management component that receives the assessment results from the Security Assessment, stores them, forwards them to other components, and offers many more functionalities.

Apart from the evidence collection from cloud systems, MEDINA integrates multiple other evidence collectors, e.g., based on software analysis (Codyze[10]) and malware protection (Wazuh[11]).

[8] https://git.code.tecnalia.com/medina/public
[9] https://github.com/clouditor/clouditor
[10] https://github.com/Fraunhofer-AISEC/codyze
[11] https://wazuh.com

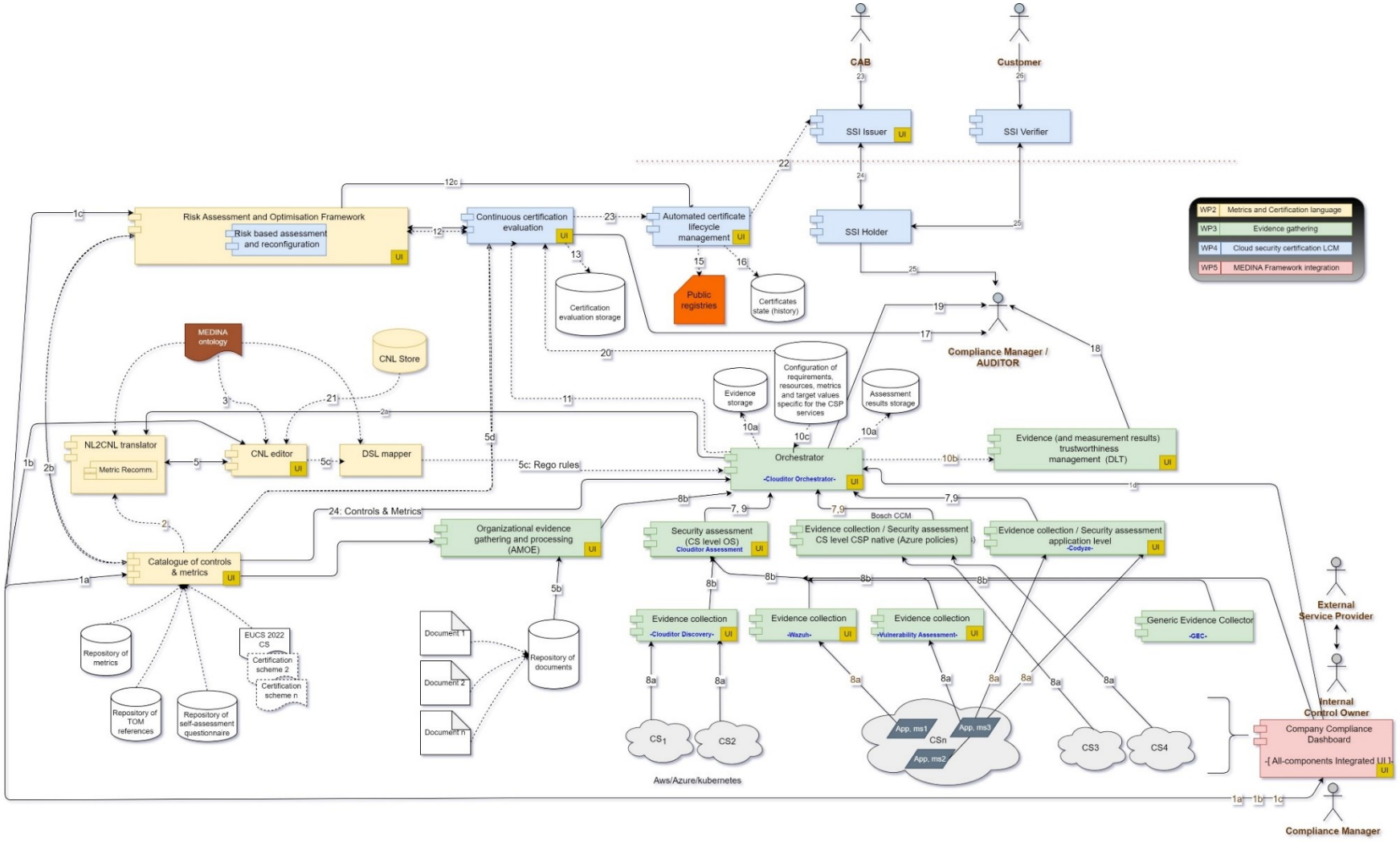

Figure 1. Architecture diagram of the MEDINA framework

3 Certification Life Cycle Management

3.1 Continuous Certification Evaluation

The evaluation of security compliance in MEDINA starts with the gathering of evidence by different tools and techniques. Security assessment components assess this evidence based on the target values as configured for the specific requirement and provide their output (assessment results with the state of fulfilment of a specific metric for a specific monitored resource) to the Continuous Certification Evaluation (CCE) component. If the assessment result value represents the lowest-level information about the certification state, the role of the CCE component is to combine the received assessment results into information about the fulfilment of higher-level certification objects: requirements, controls, control groups, and the selected certificate scheme in its entirety. This information does not directly determine the cloud service's eligibility for a certificate, but serves as input for other components, the Risk Assessment and Optimisation Framework and the Certificate Lifecycle Management, as well as for easy visualisation of the certificate state for the users (Cloud Service Providers – CSPs - and auditors).

3.1.1 Overall Component Description

The method for aggregation of assessment results in the Continuous Certification Evaluation component follows the tree-like hierarchy. Values in the tree are evaluated and aggregated bottom-up: from the leaves that represent assessment results to the root representing the complete certification scheme and thus indicating the fulfilment of the certificate.

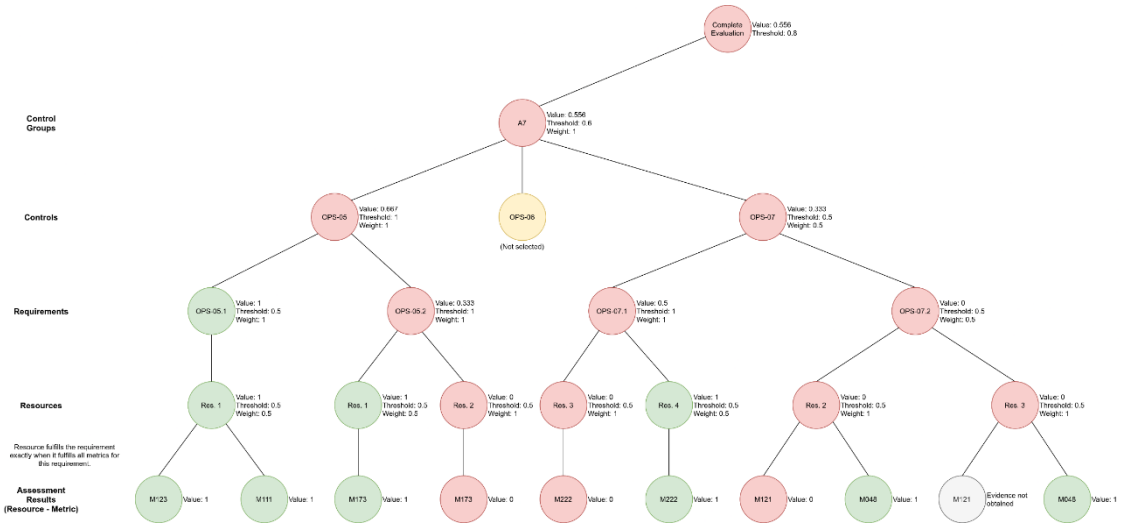

Figure 2. An excerpt of an example evaluation tree representing (non-)conformities of standardisation hierarchy elements

The aggregation can be done with weighted arithmetic means. Additionally, since the goal is to also present intermediate fulfilment values in all levels of the aggregation tree (not only at its root for the entire certification fulfilment), thresholds can be set to determine the fulfilment in individual tree nodes (controls, control groups, etc.). These thresholds and the aggregation weights of the nodes can be set by the user or the auditor (e.g., based on the importance of evaluated resources or controls). On the other hand, the evaluation tree can be easily simplified to an AND tree by setting the thresholds in all nodes to 1, meaning that all the assessment results must indicate fulfilment for the evaluation to be positive, irrespective of the assigned weights (as long as they are positive). This is the current setup used.

Beside the calculation of the current state of the evaluation tree nodes, the CCE also provides information about the evaluation history supported by metrics of operational effectiveness, through the button "Current tree state". These are metrics that measure, in various ways, how well a particular requirement or control was established (fulfilled) in a certain period of time. If a control is unfulfilled for a small amount of time, this is typically not a big issue for the entire certificate state. On the other hand, if the problem has not been mitigated for a long time, the certificate may be revoked.

Regarding connection to other MEDINA components, CCE results are forwarded to the Risk Assessment and Optimisation Framework component to further evaluate them and report possible deviations to the Life-Cycle Manager. The Risk Assessment framework does not consume the entire tree, but only the bottom three levels of nodes (assessment results, resources, and requirements). As an additional metric in evaluating the final certificate state, the Life-Cycle Manager further inspects the operational effectiveness values obtained directly from the CCE.

The Continuous Certification Evaluation component is also linked with the Catalogue of Controls and Metrics (developed in WP2) and the Orchestrator component. The Catalogue provides the structure of the used certification scheme (lists and mappings of metrics, requirements, controls, control groups…), needed to construct the evaluation tree. The Orchestrator is the source of all configurations related to the evaluated service (target of evaluation), including the chosen controls/requirements and a list of monitored resources subject to evaluation.

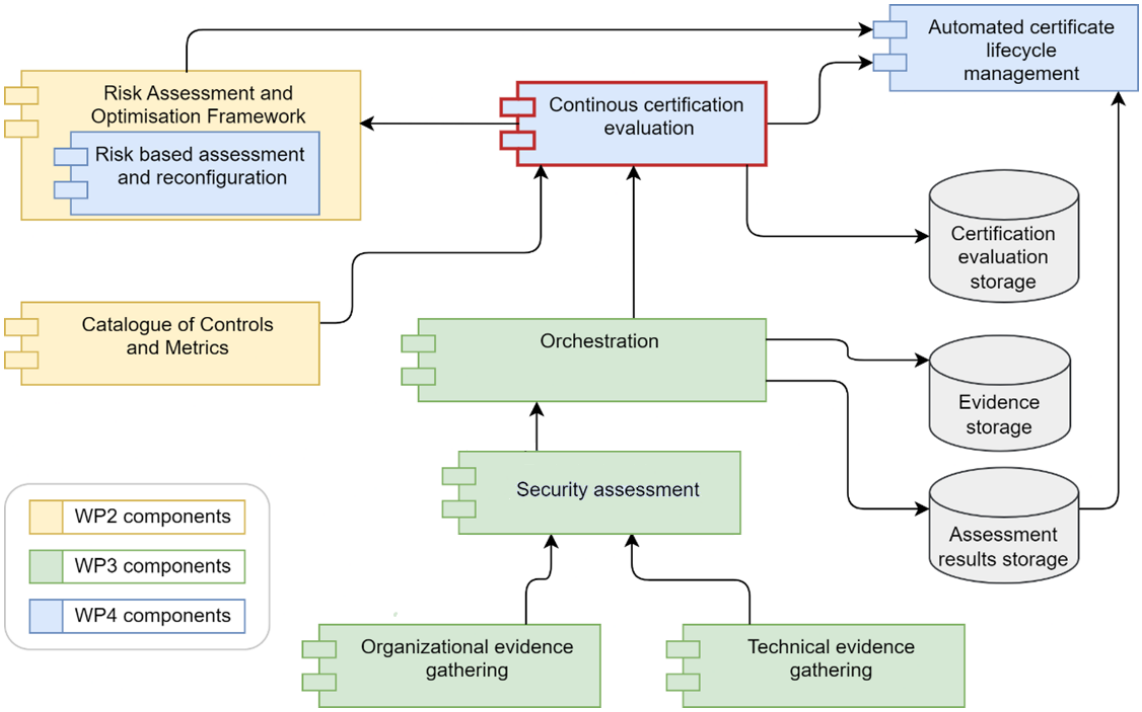

Figure 3. Continuous Certification Evaluation: diagram of interaction with related components

3.1.2 CCE's Role in the Continuous Certification Life Cycle Management

As mentioned in the previous section, the role of the CCE component is to use the received assessment results to evaluate the compliance level on all levels of the certification hierarchy (resources, requirements, controls, control groups, standard) based on the aggregation of assessment results and configuration (weights of individual tree nodes). The main challenge in aggregation was in choosing the most optimum aggregation technique for aggregating assessments results. Three aggregation techniques were considered:

a) directly aggregating assessment results into compliance values of requirements,
b) combining assessment results of different resources into compliance values of metrics, and combining metrics into compliance values of requirements,
c) combining assessment results of different metrics into compliance values of resources and combining resources into compliance values of requirements.

The approach for the initial MEDINA proof-of-concept considers that all metrics for a particular resource need to be evaluated positively to regard the requirement fulfilled. Aggregation of the metrics level is thus made with simple AND rules and weighted aggregation does not apply at this level. On the other hand, configuration of different weights for resources can be desirable from the risk assessment perspective. With aggregation technique b), fulfilment values of metrics are calculated from multiple resources and are thus not Boolean values. If we are to apply AND aggregation on metrics, we could consider the metrics values positive or negative depending on thresholds. Regardless of thresholds though, the weights of resources used on the leaf-level would become irrelevant at the requirement and higher levels of the evaluation tree (Boolean fulfilment values are applied to requirements). With technique c), the assessment results are aggregated into resources' compliance levels using AND, applying Boolean values to resources. The compliance values of resources can therefore be aggregated into requirements' fulfilment levels using their respective weights.

Following the considerations described above, technique c) was chosen for the implementation of the Continuous Certificate Evaluation component and is therefore considered in the tree-building process. Additional, CCE does not receive any weights or thresholds of individual nodes and thus treats all parts of the certification tree equally in the aggregation. The distinction between the importance of various requirements or controls is considered by the Risk Assessment and Optimisation Framework when determining how critical an incompliance is to the overall certification state of a CSP.

3.2 Risk Assessment and Optimisation Framework

The Risk Assessment and Optimisation Framework (RAOF) aims to evaluate the detected non-conformity and define whether it should be treated as minor or major problem. During the continuous monitoring phase, the result of the analysis is to be used by the Life Cycle Manager (see Section 3.3) in its decision making about the state of the certificate.

3.2.1 Overall Component Description

It worth mentioning that as a component, RAOF is used in two phases: before obtaining a certificate (in order to evaluate the state of compliance using the input provided by a compliance manager) and after obtaining it, i.e., during continuous monitoring phase (the goal is to evaluate the current state of compliance using the input information provided automatically by assessment tools). In this whitepaper, RAOF operation is considered only during the latter phase.

The analysis performed by RAOF is based on risk assessment. In a nutshell, the component checks how much risky is the actual security configurations (i.e., how many EUCS requirements are correctly implemented) with respect to the ideal situation, in which all EUCS requirements are addressed. Risk-based analysis allows focusing on the CSP's needs and differentiate between failures in implementing different requirements.

Similar to other risk assessment guidelines, the risk assessment process is based on identification of the three main risk features: assets, threats and vulnerabilities. Assets are to be provided to the tool before starting the monitoring procedure (during the preparation phase). The list of considered threats (selected specifically for CSPs) is contained in the tool. Vulnerabilities represent failures in addressing EUCS requirements. The RAOF defines relations between

possible assets, threats and vulnerabilities required for computing risk values. Thus, during the continuous monitoring phase, only the information on fulfilment of EUCS requirements is needed for risk computation. This information is provided by CCE (see Section 3.1).

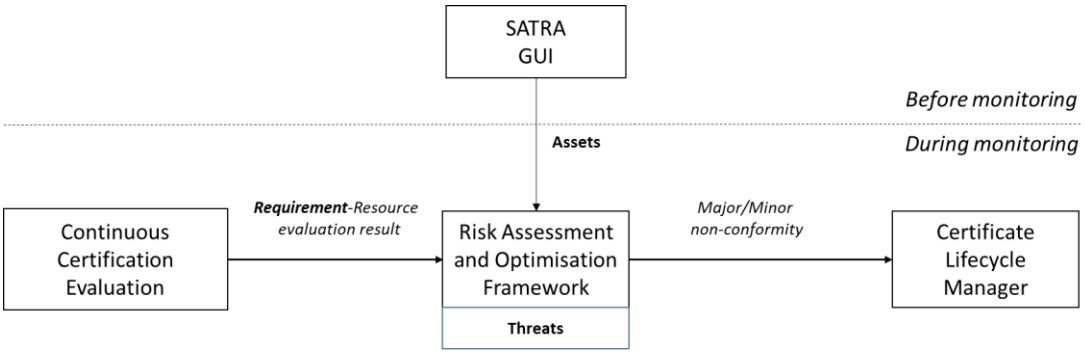

Figure 4. Provisioning of input to SATRA and consuming its results.

Assets are reported by a compliance manager before starting the monitoring. The tool specifies 14 types of possible assets, based on typical Cloud resources (called, *supported* resources). There is also one specific resource, that represents clients' trust, it helps to take into account threats related to incorrect provisioning of resources (e.g., storing data outside of allowed geographical region).

Information about assets includes expected loss of Confidentiality, Integrity, and Availability of these assets due to a threat occurrence. Since a cloud resource may be added or removed dynamically, two mechanisms are provided for reporting these values. First, a compliance manager may set up the required values for a resource type, and all resources of this type will be treated by RAOF as having these values. Second, specific resources (i.e., major assets) can be associated with specific values, to single out them from the rest of resources. The first approach helps to label any resource of a specific type, and the second one ensures that the main resources get more focus in the analysis.

As it is said above, vulnerabilities in our analysis are represented by the inability of the CSP to satisfy EUCS requirements. Various assessment tools measure and evaluate different cyber security metrics, which confirm or disprove that an associated requirement is well addressed or not (the later part is done by CCE). This type of assessment is performed for an associated resource. Having this fine-grained information, risk can be computed for every supported resource and then aggregated.

Since not all resources directly impact risk assessment (i.e., not all of them are treated as assets), RAOF divides reported resources in two sets: supported and non-supported. Logically, non-supported resources do not cause direct damage (e.g., an account or a policy document), but may lead to a loss for any of supported resources (e.g., database or virtual machine). Thus, in order to understand which requirements are satisfied for a specific supported resource, first all reported requirements stated for non-supported resources are defined, and then those that are associated with the supported resources are taken into account. Any requirement which status is not determined in this way, i.e., we have no objective measure to tell if the requirement is satisfied or it is not, is assumed as satisfied[12].

[12] We acknowledge that this is not the most objective approach, but with no measurement available we can only rely on honesty of the CSP. Yet, in the ideal situation, all requirements must be monitored and there must be evidence supporting fulfillment (or non fulfilment) of all requirements.

In this way complete information about satisfaction of EUCS requirements per an asset (i.e., supported resource) is obtained and the corresponding risk is computed.

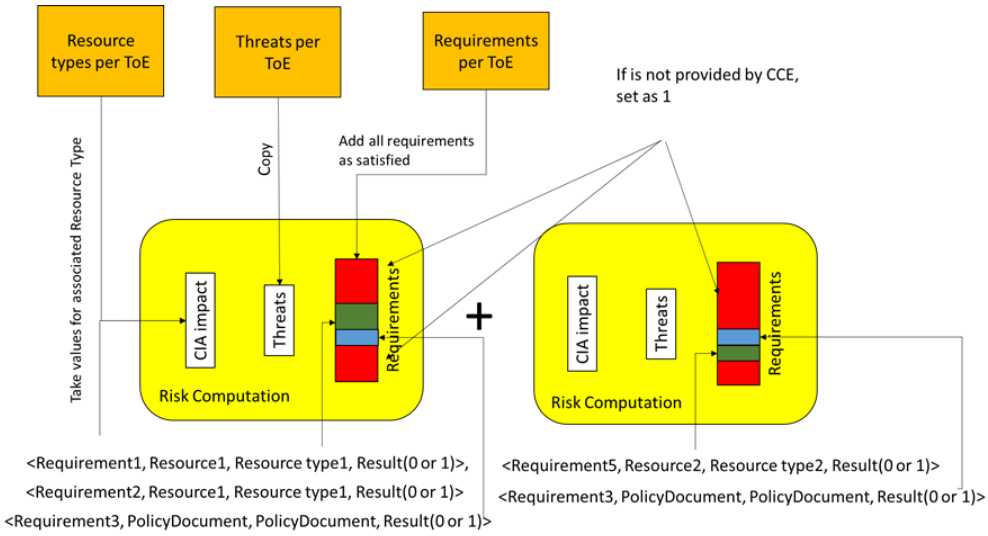

Figure 5. Computation of risk during the dynamic risk assessment

Finally, risks per supported resources are summed up to obtain the overall risk value per service. Note, that although our analysis is quantitative, the final result is transformed to a [0;100] value for simplicity of further usage.

Then, we compute "ideal risk", i.e., the risk level for the service of all requirements are properly addressed. This is the best risk level which can be obtained by the service. Finally, the difference between real and ideal risk is found and compared with a threshold. In case the detected deviation is higher than the threshold, the detected non-conformity is considered as Major (and Minor otherwise).

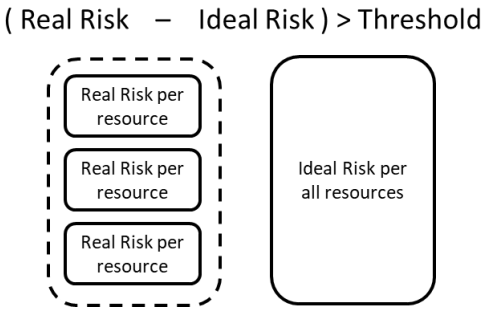

Figure 6. Non conformity analysis

3.2.2 RAOF's Role in the Continuous Certification Life Cycle Management

RAOF is used to evaluate the detected non-conformity and decide if it should be considered as Major or Minor. One of the challenges is to compute risk for a cloud service with resources that could be added or removed dynamically. This challenge is addressed with the possibility to set up labelling values for a resource type, yet allowing to set up different labels for specific resources.

Another challenge is to measure risk for the whole service, while the monitored values are associated with specific resources. This challenge is addressed by splitting all resources in two classes, using the reported values to define the input for risk assessment procedure focusing on supported resources, and then aggregating the values for the whole service.

Finally, it was challenging to use risk for assessment of non-conformity. This challenge is solved by comparing the current risk level and the ideal risk level.

3.3 Automated Certificate Life Cycle Manager

The automated certificate Life Cycle Manager (LCM) is the component that aggregates information about the compliance state of the cloud service as a whole.

3.3.1 Overall Component Description

In Section 3.1, we have already seen how the large amount of assessment results that is continuously generated is aggregated for a cloud service and its certification(s). Depending on how many changes have occurred in the service and its compliance state, new risk values are computed, potentially also very frequently (see Section 3.2).

The purpose of the LCM is to take a decision on the state of a certain certificate. This is a complex decision that is normally taken by a human auditor after carefully examining lots of documents and resources, conducting interviews with employees, and other audit activities. In the LCM component, we try to reduce the complexity of this decision process to make it as simple and transparent, but also as effective and correct as possible.

The EUCS-defined certificate states are as follows:

- **New Certificate:** This is the state of a newly issued certificate.
- **Continued:** Certificates that have been re-assessed after their initial issuance without a significant change in the evaluation result, obtain this state.
- **Renewed:** The *renewed* state is for certificates that have been re-assessed and their validity is extended. Also, updates to its information may be done.
- **Updated:** Certificates that have been re-assessed and remain valid but whose information should be updated have the *updated* state.
- **Suspended:** Certificates that have been re-assessed resulting in the discovery of major deviations with the schema's requirements are *suspended*.
- **Withdrawn:** Certificates that are suspended and are not maintained are finally *withdrawn*.

Figure 7 illustrates the three sources of information for the decision process: there are hardcoded timing rules, as well as historical compliance data (operational effectiveness), and overall risk values for the cloud service in question.

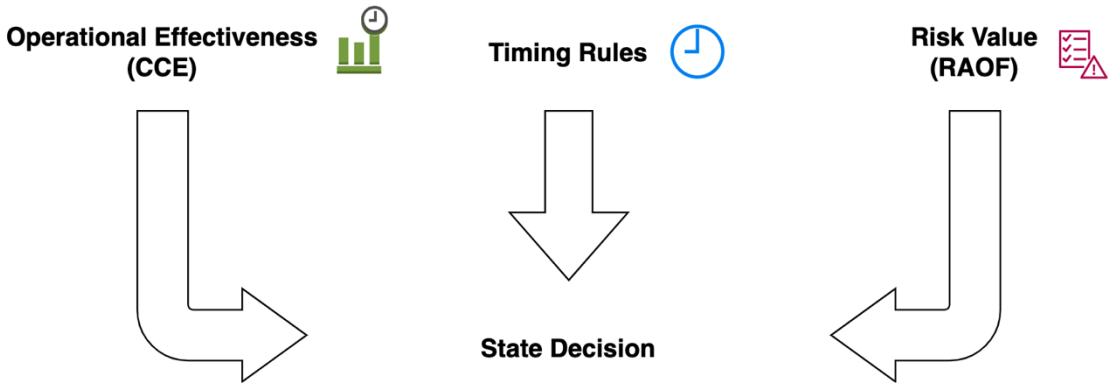

Figure 7. The decision factors that are taken into account by the Life Cycle Manager: Operational effectiveness data is retrieved from the CCE, timing rules are hardcoded in the LCM, while a service's risk value is calculated by RAOF

- **Operational Effectiveness:** The operational effectiveness examines the service's compliance over a certain timeframe. This way several factors can potentially be included in the state decision: First, the ratio of compliance to non-compliance, for example in the past 3 months, can be computed as an indicator for how consistent the compliance (of a certain requirement) is. Second, it can be inferred how long it usually takes to fix a non-compliance once it arises.
- **Timing Rules:** The EUCS defines several certificate state transitions based on time periods, e.g., of inactivity. For example, a certificate may be withdrawn automatically if it is in the *suspended* state and no remediation has been performed during the past three months.
- **Risk Value:** The risk value is determined by the Risk Assessment and Optimisation Framework (see Section 3.2).

Figure 8 shows the certificate life cycle as a state machine.

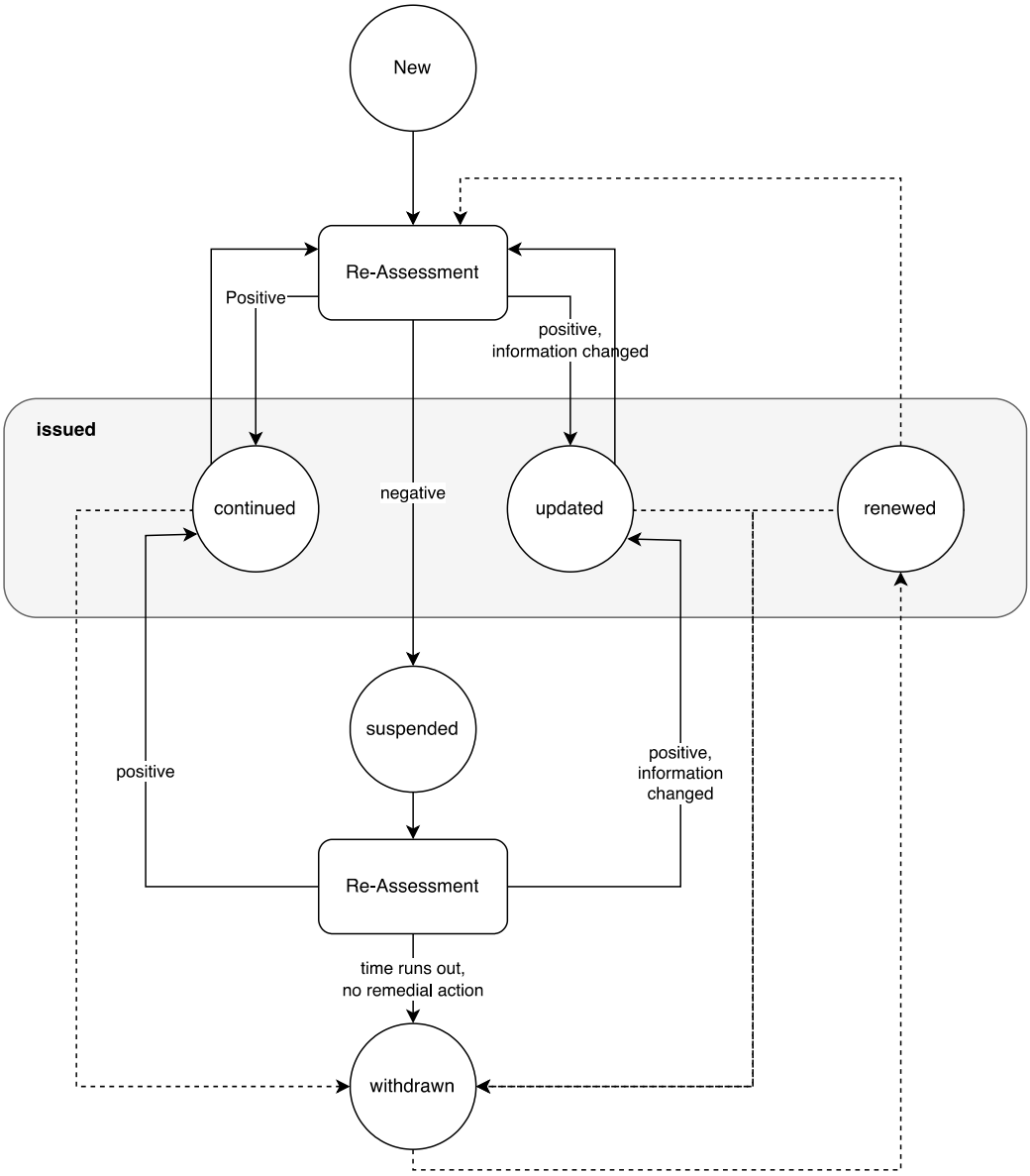

Figure 8. The state machine that encodes the EUCS-defined certificate states and their transitions

3.3.2 The LCM's Role in the Continuous Certification Life Cycle Management

The main challenges in designing and implementing the LCM were to identify the decision factors, defining thresholds for them, and designing a process that integrates and balances the decision factors and ensures a smooth and reliable life cycle management.

The LCM is therefore central in the continuous life cycle management: It does not only decide on the certificate state, but it also stores it permanently, and forwards the decision to the SSI Framework where it can be verified manually by an auditor. The storage of certificates and their states (as well as their deletion) is handled through the Orchestrator, i.e., they are stored in the Orchestrator's database alongside evidence and assessment results. This way, the LCM does not become a single point of failure for the framework. Instead, the Orchestrator's database should be well secured, and regular backups should be created to ensure reliability of certificate information.

3.4 Self-Sovereign Identity System for Managing Certificates

3.4.1 Overall Component Description

The *Self-Sovereign Identity (SSI) Framework* provides CSPs with the capability to manage their own security certificates through verifiable attestations (credentials) that are locally stored, controlled and managed. This refers to the holder component of the SSI Framework. However, there are two additional components to be considered: the issuer, which provides the CAB a way to issue verifiable attestations about the security certificates related to the CSPs updating their state based on the LCM output; and the verifier, which provides customers/clients a way to ask and verify proofs of different security certificates features from a CSP.

These functionalities are provided with a high level of security thanks to the use of Blockchain technology as backbone. Blockchain is used as a global repository of public identifiers of the different actors (CAB, and CSPs) and revocation status of the issued credentials, in order to allow secure verifications. In this sense, Blockchain technologies provide:

- Trust: Blockchain is a decentralized network of nodes from different parties; no trust on any specific party is needed.
- Integrity: this is an inherent property of Blockchain guaranteeing tamper-proofed information.
- Availability: Blockchain is a global network of computers; its down is almost impossible.

Taking everything into consideration, Figure 9 shows the SSI Framework high-level architecture showing all the components.

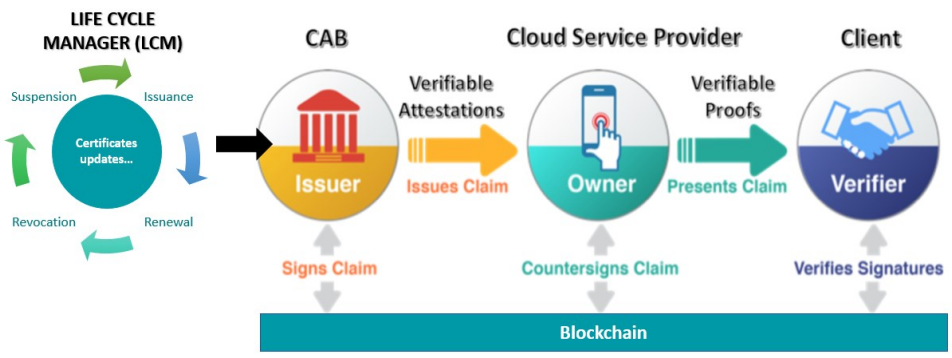

Figure 9. SSI Framework high level architecture

The main components are:

- The Blockchain network, which stores the public cryptographic material and associated metadata. An Hyperledger Indy Blockchain network has been deployed in TECNALIA for demo purposes.
- An SSI-API to be deployed with the issuer to receive the notifications from the LCM about the security certificate state updates.
- Each actor (CAB, CSP and client) has two main subcomponents:
    - SSI agent: it contains all the SSI intelligence, allowing all SSI functionalities and correct communication between them. Hyperledger Aries agents have been considered.
    - SSI controller: it refers to a web application to ease the use of the functionalities provided by SSI-agents.

3.4.2 The SSI system's Role in the Continuous Certification Life Cycle Management

The SSI Framework is considered an extension of the MEDINA framework as it provides a way to give utility and security to the results obtained from MEDINA about the CSPs security certificate states updates.

The issuer of the SSI Framework will be useful for the CAB to automatize and secure the issuance of security certificates. It receives the CSPs security certificate states updates from the LCM and automatically generates a verifiable credential with the updated information that needs to be sign. This verifiable credential is automatically shared with the corresponding CSP (holder), who stores, controls, and manages it as desired. This verifiable credential is trustworthy because it has been signed by the CAB and could not be modified. In addition, the CAB can revoke the validity of this credentials at any time (because the state has changed, for example) as required by continuous certification.

At any time, a potential customer can ask for a proof of the CSP security certificate state. The CSP can provide verifiable proofs (based on the received verifiable credentials) proving the current security certificate state. These proofs validity can be validated in two ways: i) verifying the signature of the CAB in the Blockchain, where public identity material is stored; and ii) verifying the revocation status of the credential in the Blockchain, where the revocation control is also managed.

By this way, the certification life cycle management is extended until its validation by customers.

4 Discussion

Continuous Certificate Life Cycle Management (CCLCM) represents a novel approach in the realm of certification processes, holding promises to streamline and enhance the efficiency and effectiveness of certificate management. In the following, we discuss some potentials and limitations of a continuous, automated certificate life cycle management from different perspectives.

4.1 Potentials and Limitations of Continuous Certificate Life Cycle Management

Support in Audit Preparation: Standardization of Evidence Collection and Visualization

One of the foremost potentials of CCLCM lies in its ability to enhance and standardize the audit preparation process. Traditional audits often require extensive manual effort in collecting and organizing evidence. CCLCM addresses this by standardizing evidence collection and visualization procedures. Automation can significantly expedite the gathering of necessary documentation, ensuring that all relevant artifacts are readily accessible and presented coherently during audits.

Early, Automated Alerts for Potential Problems

A significant advantage of CCLCM is the provision of early, automated alerts for potential issues. By employing real-time monitoring and analysis, deviations from established standards or compliance requirements can be promptly detected. The efficacy of alerting mechanisms, however, hinges on their alignment with the nature of deviations. For instance, in cases of major deviations or non-conformities, auditors can be informed in real time, facilitating swift intervention.

High Degree of Automation Combined with Manual Verification

Perhaps the biggest potential of CCLCM is its fusion of high automation levels with manual verification. This synergy leverages automated systems to monitor, assess, and predict certification statuses. Manual intervention, in turn, ensures the accuracy and validity of automated assessments. This combination addresses the need for human expertise in nuanced scenarios and ensures a comprehensive certification management process.

Simplified Life Cycle Management: Limited Set of Decision Factors

Despite its potentials, CCLCM presents certain limitations. One notable constraint is its reliance on a simplified life cycle management approach, which often entails the consideration of a limited set of decision factors. This simplification, while enhancing automation, may overlook intricacies present in the broader context, potentially affecting the accuracy of certification assessments.

Error Rate: Potential for High Number of False Positives

The nature of CCLCM introduces uncertainty regarding its error rate, particularly concerning the possibility of false positives. Automated systems, by their very nature, can generate alerts that do not necessarily correspond to actual non-conformities. A potential research avenue involves fine-tuning the system by iterating over various boundaries and parameters, striking a balance between identifying actual deviations and minimizing false positives. Otherwise, internal and external auditors monitoring the alerts could quickly become overwhelmed.

Recognition and Standardization Challenges

A substantial limitation of CCLCM is the relatively nascent recognition of this concept by certification scheme owners and other stakeholders in the certification process. This lack of recognition poses challenges in terms of standardization and harmonization of processes. Establishing unified frameworks and protocols is essential for the successful integration of CCLCM into existing certification paradigms.

4.2 Continuous Certificate Life Cycle Management from the CSP's Perspective

Risk of false results

The potentials and limitations of CCLCM from the CSP's perspective are largely similar to the points raised above. One such point mentioned above concerns false results: First, false positives can overwhelm internal and external auditors. An automated remediation may therefore be appropriate to install alongside automated alerts to non-compliances. Second, false negatives are a considerable risk from the CSP's perspective, as in these cases, the non-compliances are not found.

Transparency and traceability

A major issue for CSPs is to be able to trace back the reports of non-compliances and certificate changes to the root cause of these changes, e.g., the specific resource configuration that triggered the change. Missing traceability in this regard would be a major limitation to the usefulness of CCLCM.

Audit once, certify many

A significant potential lies in the standardization of audit materials, such as assessment results. Assuming that they are collected, assessed, and documented in a standardized way, they could be reused many times for different certifications. This presents a considerable potential in cost efficiency for CSPs.

Missing room for interpretation

In traditional audits, human auditors can take into account special circumstances of the CSP and apply the certification requirements to the CSP's environment. In this case, the auditor can take into account special circumstances which is not possible in a fully standardized, automated certification process.

4.3 Continuous Certificate Life Cycle Management from the Auditor's Perspective

Harmonizing Audit Material

From an auditor's perspective, there is significant potential in harmonizing audit materials. While high-level auditing guidelines exist, the absence of strict definitions for the evidence required for audits can lead to inconsistencies. Achieving a standardization necessitates collaboration with certification scheme owners to delineate the audit and CCLCM steps clearly. This includes defining evidence requirements, setting boundaries, and establishing parameters that guide audits and CCLCM.

Potentials and Limitations of Decision Factors: Timing Rules, Risk Value, Operational Effectiveness

Comparing CCLCM to the standard audit process reveals a distinct set of potentials and limitations. Timing rules within CCLCM exhibit a relatively low risk of false positives, as they involve straightforward temporal thresholds for state changes. Risk values, however, depend on maintaining parameters that determine how the Risk Assessment and Optimisation Framework (RAOF) computes risks. Setting boundaries to classify a "low" or "high" risk value demands clarity and transparency in the logic underpinning these decisions. Comprehensively computing an overall risk value contextualizes all conformities and non-conformities, providing a holistic perspective in decision-making. Operational effectiveness similarly requires defining meaningful ratios that indicate major deviations. Establishing boundaries for ratios and time horizons necessitates a deep understanding of different standards' requirements. Notably, while traditional audits emphasize process assessment, the MEDINA CCLCM approach focuses more on non-conformities, altering the auditor's perspective and priorities.

Reliability: Avoiding False Positives and Managing State Changes

Ensuring the reliability of the CCLCM process, particularly in avoiding false positives, becomes a paramount concern from auditors‘ perspective. The risk of erroneously suspending a certificate presents a significant challenge. The potential multitude of state changes necessitates the establishment of well-defined boundaries. These boundaries determine the severity of non-conformities that would trigger certificate suspension and should be as generic as possible.

Currently, certificate management remains predominantly manual due to the lack of standardization in the CCLCM processes. This environment impedes auditors from embracing automatic certificate management, even though automation could lead to efficiency gains. The industry's gradual progress toward continuous auditing further accentuates the cautious approach taken by auditors.

Overall, automation within the CCLCM process offers the potential for auditors to redirect their attention toward more critical tasks, i.e., they can focus on tasks that require human judgment, thereby enhancing the quality of audits. Furthermore, the prospect of automating some aspects of CCLCM opens avenues for auditors to explore innovative business models, such as continuous monitoring of non-conformities.

4.4 Limitations of the Tools

4.4.1 Continuous Certification Evaluation

The evaluation tree built by the CCE component is an enhanced representation of data coming from the evidence gathering and security assessment tools. The confidence of the CCE's outputs thus largely depends on the data provided by those components.

The CCE can be efficiently used to review the state of gathered evidence at the chosen point in time, but a limitation is that no conclusions about the actual risk state or the certification status can be made solely based on the CCE outputs. Other components of the MEDINA solution (Risk Assessment and Optimisation Framework and certificate Life-Cycle Manager) help users to better understand the broader view of their certification state.

Regarding improvements, to enable full and dynamic support for multiple targets of evaluation, it would be necessary to implement changes in CCE according to updates in the common data model as well as optimize the connection with the Orchestrator regarding the exchange of data about targets of evaluation and their configuration (e.g., requirements to be covered).

4.4.2 Risk Assessment and Optimisation Framework

Certainly, our approach has inherent limitations. Ideally, it should be implemented when monitoring covers all resource requirements comprehensively. Regrettably, we cannot guarantee the realism of this assumption. We have diligently crafted a computational model that permits risk assessment, but we must assume that unmonitored requirements are satisfied.

Furthermore, our evaluation is constrained by the resource categories recognized and endorsed by Clouditor, which are subsequently refined to concentrate solely on resources potentially housing sensitive data or playing a key role in core business operations—those whose compromise directly impacts data and processes, considered as the primary assets. We draw upon the expertise of our collaborators who have established an ontology for cloud cybersecurity resources.

Finally, it's worth noting that specifying values individually for each resource, as opposed to applying uniform values for all resources of the same type, could theoretically offer greater precision. However, the sheer variety of potential resource types and the ever-changing nature of the cloud environment, where resources can be added or removed dynamically, render this approach unfeasible. Therefore, after several discussions with our use case owners, we have chosen to adhere to this generalised approach, while retaining the flexibility to address the most critical resources on a case-by-case basis.

Potential directions for enhancing the tool include a range of options, such as:

1. Incorporating consideration of resource interrelationships and the interdependence of security attributes.
2. Establishing predefined states for requirements that prove challenging (or unfeasible) to monitor during the bootstrapping phase.
3. Developing a more efficient method for estimating sensitivity (Confidentiality, Integrity, and Availability impact) for dynamic resources.

4.4.3 Automated Certificate Life Cycle Manager

There are several limitations of the Life Cycle Manager, beginning with its focal point on risk value and operational effectiveness. While this specific focus provide valuable insights, its practical utility necessitates substantiation through empirical studies. Real-world scenarios and use cases are important in demonstrating how this rather narrow scope contributes to informed decision-making about certificate states.

Moreover, an intricate challenge emerges from the possibility of generating frequent changes in certificates due to oscillating assessment results. This phenomenon has the potential to overwhelm auditors, inundating them with notifications and state changes that may not always correspond to substantial deviations. The consequential cognitive load on auditors underscores the importance of refining and fine-tuning the Life Cycle Manager's sensitivity to fluctuations to maintain its effectiveness without generating undue noise.

Additionally, the security of certificates within the Life Cycle Manager's scope is dependent on the security measures implemented by Cloud Service Providers (CSPs). Consequently, while the Life Cycle Manager addresses certification-related security aspects, its effectiveness can be limited by the prevailing measures that secure the entirety of the cloud environment.

Potential future work for the Life Cycle Manager component includes pragmatic experimentation as well as theoretical studies. Practical implementation of the Life Cycle Manager warrants an exploration of how its parameters, such as thresholds for risk value and operational effectiveness, should be configured to strike the optimal balance between

sensitivity and reliability. This entails not only determining the thresholds but also considering their adaptability to varying contexts and use cases. This way, oscillating certificate states, particularly in scenarios where resource configurations undergo frequent changes, may be preventable.

Another direction for future work is the LCM's extension to other certification catalogues. To this end, the certification states and their conditions for state transitions need to be translated to state machines (as seen in Figure 8) and implemented in the LCM.

In conclusion, multiple limitations and opportunities envelop the Life Cycle Manager component. Empirical validation of its scope, addressing challenges related to oscillating assessment results, and fine-tuning its sensitivity are paramount.

4.4.4 SSI Framework

The main limitation in the current development of the *SSI Framework* prototype is related to the simulation of CAB (issuer) and CSP customers (verifier). More validation is still needed including real partners for the credentials' issuance and verification functionalities (new requirements will arise).

Additionally, a prototyping Blockchain network has been deployed for MEDINA. However, in real deployments, this network should be distributed among different organizations, without a central control.

Finally, the information to be included on the credential for the security certificate could be extended with more fields and details provided not only by the LCM but also from other tools or sources of information.

5 Conclusions

The dynamic nature of today's cloud systems both call for thorough security evaluations to ensure the protection of sensitive data as well as the continuous—and thus automated—execution of such evaluations.

In this whitepaper, we have reviewed MEDINA's contribution to a continuous, automated management of certification life cycles. MEDINA's components allow to continuously collect evidence, assess, manage, and evaluate it, and translate it automatically into a certificate state based on the EUCS.

We have reviewed the different tools that make up this process: First, the Continuous Certification Evaluation aggregates assessment results, assigning them to the correct cloud service and certification schema. The results are then used in the Risk Assessment and Optimisation Framework to calculate an overall risk value for the cloud service. This value, alongside other information, is then used by the Life Cycle Manager to decide on the certificate state according to the transitions defined in the EUCS. Finally, the SSI system allows to involve humans in this automated process to ensure a reliable and self-sovereign issuance of certificates.

MEDINA's project lifetime ends in October 2023. The follow-up EU-funded project EMERALD, however, will further develop results from MEDINA, bringing them to a higher technology readiness level. Also, the EU-funded COBALT project will build on results from MEDINA as it aims at creating a common certification model for different domains besides cloud computing.