Holistic Cyber Risk Assessment in the Cloud Continuum: A Multi-Layer, Multi-Domain Approach

IEEE Access
Multidisciplinary ⋮ Rapid Review ⋮ Open Access Journal

Received 23 September 2025, accepted 8 October 2025, date of publication 17 October 2025, date of current version 23 October 2025.

Digital Object Identifier 10.1109/ACCESS.2025.3622915

RESEARCH ARTICLE

Holistic Cyber Risk Assessment in the Cloud Continuum: A Multi-Layer, Multi-Domain Approach

GABRIELE GATTI[1], JOSÉ MARÍA JORQUERA VALERO[2], MANUEL GIL PÉREZ[2], AND CATALDO BASILE[1], (Member, IEEE)

[1]Department of Control and Computer Engineering, Polytechnic University of Turin, 10129 Turin, Italy
[2]Department of Information and Communications Engineering, University of Murcia, 30100 Murcia, Spain

Corresponding author: José MarÍa Jorquera Valero (josemaria.jorquera@um.es)

This work was supported in part by the Smart Networks and Services Joint Undertaking (SNS JU) under European Union's Horizon Europe Research and Innovation Programme, iTrust6G Project under Agreement 101139198; in part by the Strategic Project CDL-TALENTUM, Spanish National Institute of Cybersecurity (INCIBE) by the Recovery, Transformation, and Resilience Plan, Next Generation EU; in part by the MOVING MINDS grant by the Directorate General for Universities and Research of the Regional Ministry of the Environment, Universities, Research, and Mar Menor of the Autonomous Community of the Region of Murcia; and in part by European Commission through the Horizon Europe/JU SNS Project ROBUST-6G under Agreement 101139068. The work of Gabriele Gatti and Cataldo Basile was PARTIALLY supported by the SERICS–Security and Rights in the CyberSpace Project and Received Funding from European Union Next-GenerationEU (PIANO NAZIONALE DI RIPRESA E RESILIENZA (PNRR)—MISSIONE 4 COMPONENTE 2, INVESTIMENTO 1.3—D.D. 1556 11/10/2022) under Grant PE00000014.

ABSTRACT The increasing complexity of modern information technology infrastructures poses new challenges for cyber risk assessment, especially when the boundaries of resources and computing extend across multiple administrative domains and heterogeneous environments, such as those found in the cloud continuum. Traditional frameworks fall short in these dynamic, multi-domain contexts. In this work, we propose a holistic approach that combines trust assessment through continuous monitoring of platform health indicators and service-level security assessment into a comprehensive cyber risk model for quantitative scoring. Our framework leverages a multi-layer architecture featuring automated data collection, semantic enrichment, and advanced risk metrics. It supports intra- and inter-layer anomaly detection and is validated on an experimental, cloud continuum-like deployment spanning multiple administrative domains and mixing physical and virtual environments. Results show that our method outperforms isolated approaches, offering enhanced detection, contextual explanation of threats, and improved risk visibility across the cloud continuum.

INDEX TERMS Cloud continuum, continuous monitoring, cyber risk assessment, dynamic risk modeling, multi-domain security, trust assessment.

I. INTRODUCTION

More and more companies rely on information systems every day to achieve their business objectives. While the advantages of introducing innovations such as software automation, virtualization, broadly available data storage, and high-frequency computing are undeniable, it is also indisputable that new kinds of risk may arise as the surface for threat infiltration increases [1]. This aspect takes on greater importance when adopting new technologies, which becomes more pervasive in the architecture of an organisation. For instance, the integration of highly distributed and virtualized environments such as cloud services, but also physical Internet of Things (IoT) devices in the business processes, together with locally deployed services and devices, determines an engrained reliance on the sharing of resources across different domains (either administrative or logically separated), shaping what could be defined as a (cloud) continuum [2].

The increase in the complexity given by the cloud continuum and the underlying virtualization technologies also

The associate editor coordinating the review of this manuscript and approving it for publication was Barbara Masini.

determines a rising complexity concerning the risk analysis, assessment, and general management of cyber risk. Current risk assessment frameworks and guidelines may provide valuable conceptual insights into managing these complex scenarios, but do not provide the quantitative measures needed to accurately assess the required investments in terms of cybersecurity [3]. Hence, significant efforts are needed to adapt them to assess and measure risks in target organizations [4], [5]. These efforts can be expressed in terms of costs and resources needed to establish internal risk management activities or subsidised to third parties.

An obvious and widely adopted solution to simplify this problem could be separately analysing sub-systems. Target components can be regarded as closed boxes, observing what they expose through the network, as in typical third-party risk assessment approaches based on vulnerability assessment tasks [6], [7], [8]. Still, this approach leaves behind the fundamental concept that risk may propagate across said components, an aspect which cannot be excluded from a comprehensive risk analysis [9], especially when aiming at continuously monitoring risks. It also does not capture well the information related to modern software-defined infrastructures, where, for instance, a high level of observability could be leveraged [10]. Thereby, it is not enough to study the individual components to detect vulnerabilities and potential security issues; the relationships and interactions among them could represent possible opportunities for cyber-attackers. Indeed, vulnerabilities in a component may be leveraged to compromise other elements belonging to completely different domains [11], [12].

Our research goal is to simplify risk assessment tasks in the cloud continuum and other multi-domain environments by automating the collection of risk-relevant data and the computation of quantitative (or semi-quantitative) risk scores and at the same time, increasing the quality of the risk assessment and supporting continuous monitoring. A fundamental assumption in our research is that analysing the platform's health of individual nodes composing the cloud continuum should be an integral part of a risk management framework. Managing potential symptoms may help the risk management framework to prevent the forthcoming spread of threats. Many indicators, such as CPU, memory, network, and processes, can provide useful information regarding active threats that might not have been considered during the risk analysis due to bad practices, lack of information sources on assets, or because of their novelty [13]. The research question that guided this study was whether composing local platform health data with service-level information about vulnerabilities, exposures, and weaknesses in a comprehensive risk model could estimate the risks an organization faces in heterogeneous multi-domain scenarios better than the currently existing approaches, where the two information domains are considered separately. In other words, our objective is to determine if the composition of these data can produce additional knowledge for infrastructure administrators and cyber risk respondents. Therefore, in this article, we propose an approach that provides a holistic overview of cyber risk across complex, distributed architectures such as those found in the cloud continuum. To illustrate and motivate our work, we consider a concrete example scenario for which we developed an ad-hoc testbed: a multi-domain environment that includes IoT and edge servers in one administrative domain (e.g. simulating a smart building located in Spain), and a cloud-native infrastructure in a second domain (e.g. mimicking a data center in Italy). In this setup, resource sharing and service interactions span national borders, heterogeneous technologies, and administrative silos, creating an intricate attack surface.

In such an environment, a vulnerability on a single node may be exploited to compromise a connected layer, increasing in complexity and escalating the severity of the attack while also providing an example of risk propagation across domains. Traditional risk models are insufficient for continuously capturing this type of cascading threats.

In this context, the threat model assumes that adversaries with network-level access may exploit vulnerabilities across IoT, edge, and cloud layers to compromise confidentiality, integrity, and availability. Typical attack vectors include weak authentication mechanisms (e.g. default or guessable credentials on IoT devices) enabling remote command execution, injection attacks (e.g. SQL injection on web-facing services) leading to data exfiltration, and misconfigurations in cloud-native platforms that expose sensitive services. In addition, adversaries may deploy malware or leverage zero-day exploits that affect system stability and performance, often visible through abnormal resource usage such as CPU, memory, or network activity. Overall, the threat model highlights the potential for both localized compromises and cascading attacks that propagate across layers and administrative domains.

Our approach addresses the limitation of the existing frameworks by combining two complementary perspectives: trust assessment, using low-level platform health indicators (e.g. CPU usage, process analysis, memory, network activity) to detect anomalies; and network-based assessment, which uses vulnerability and configuration analysis of deployed services. These are integrated into a unified risk model that quantifies, correlates, and explains risks across the cloud continuum close to real-time. Furthermore, the framework is designed as a foundational layer that can support increasingly sophisticated correlation logic, from static associations, as demonstrated in our proof of concept, to future AI-driven models.

Our contributions are summarized as follows:

- a risk model for multi-domain scenarios comprising multiple metrics that quantitatively and qualitatively synthesize risk exposures, and offer explanatory capabilities to anomalies;
- a multi-layer analyser able to detect available services and evaluate their vulnerabilities by employing open-source tools, both on traditional and cloud infrastructures;

- a trust framework leveraging low-level health indicators of the underlying hardware to detect anomalies and assess trustworthiness.

These contributions have been integrated into an automated framework for assessing and monitoring cybersecurity risks across multi-domain environments, which we have used to validate our approach.

This framework was then instantiated on the devices composing the validation environment and was able to produce novel and useful information for security administrators in the form of detected vulnerabilities and security issues, risk assessment metrics, and explainable real-time anomaly indicators.

Our validation showed that a more integrated approach merging data from different domains, when paired with a comprehensive risk model, allows assessing risks better than simple individual scores and enriches threat detection in multi-domain scenarios in a way that is not obtainable without merging these data through an association model. Indeed, the outcomes presented in this work are currently being employed in EU-funded projects, namely the 6G flagship Hexa-X project [14] and the iTrust6G project [15].

The rest of the paper is structured as follows: Section II presents an analysis of the existing literature related to risk and trust assessment in modern environments, Section III provides details about the solution we propose, with its components, their interactions, and implementation details. Section IV presents how our approach has been validated; Section V highlights the significance of the obtained results and the need for further research in the field, while also addressing the limitations of our approach. Lastly, Section VI draws conclusions and lays down ideas for future work.

II. RELATED WORKS

Cloud continuum is an emerging paradigm that can be defined as an extension of the traditional cloud towards multiple entities, such as Fog, Edge, and IoT nodes, with the aim of extending resources and computational capabilities [2]. Given this concept's novelty, the ideas of trust and risk assessment applied to this context are scarcely investigated in the literature, where the focus is rather on the elements composing cloud continuum architectures.

In fact, in the scope of risk assessment applied at different levels of the cloud continuum architecture, several examples of frameworks have been proposed and investigated. Starting from the IoT layer, the work by Kandasamy et al. [16] consists of one of the most recent and relevant analyses of risk assessment frameworks applied to different types of risk and IoT domains. At the same time, this research proposes a quantitative approach for calculating risk scores for IoT devices (with an example of application in the medical IoT domain) that is based on static factors, in a similar fashion to our initial approach. However, a significant drawback of this methodology is that it does not include real-time parameters from the devices under analysis, thus lacking a precise knowledge of the system's state. In our case, this limitation was overcome by introducing the health information provided by the trust framework in the score computation. Also applicable to the IoT layer and the edge layer is the framework developed by Gatti et al. [4], which aimed at developing a practical risk assessment expert system that could produce scores by analyzing information from various external tools and information sources. Although the research served as one of the main inspirations of this work, it presented several significant obstacles for a comprehensive risk assessment, starting from the analysis point of view, which was external to the organization as it mimicked the perspective of an attacker. This approach heavily constrains the amount of risk-relevant information obtainable and their reliability, reducing the quality of the assessment. The research work presented in this paper assumes complete interaction with the IT infrastructure, completely overcoming the limitation. Regarding cloud systems, the work by Akinrolabu et al. [5] underlines the need for cloud-specific risk assessment frameworks, as traditional ones do not adapt well to the needs of dynamic environments such as cloud computing. Their work aims to fill the gap related to risk assessment from a cloud service provider perspective. Hence, they propose the Cyber Supply Chain Cloud Risk Assessment (CSCCRA) model, which strongly focuses on the risks introduced by supply chain disruptions. As the cloud supply chain closely relates to the cloud continuum due to the significant reliance on resource sharing across different domains, this work can be considered related to ours. However, our approach is considerably different and more comprehensive as we use the combination of platform indicators and service vulnerabilities to manage the risks derived from dependence on other nodes (therefore, including supply chain analysis as another possible application scenario) and the risks of individual elements.

When it comes to trust assessment, several solutions have emerged in the last years to cover the cloud continuum requirements. To begin with, IoT-Edge solutions shall boost lightweight approaches to incorporate trust evaluations close to end-users without overloading resource-constraint devices. In this sense, Din et al. [17] designs a lightweight management framework to ensure trust and security among heterogeneous IoT nodes deployed for healthcare applications. In terms of trust assessment, the authors gather reliability, cooperativeness, and experience. Such features are contemplated to discover events on IoT nodes that may increase or decrease trust levels. Nonetheless, the authors do not detect whether events may entail potential threats or misbehaviours of IoT nodes as the proposed solution of this manuscript does. In terms of performance evaluation, Valero et al. analyze the resilience of their trust framework against well-known trust attacks such as bad-mouthing [18] where an attacker intends to decrease or increase the reputation of a honest entity via deceptive recommendations, self-promoting [19] in which a dishonest entity promotes its own reputation to gain an fair privileges on purpose, or on-off [20] where an insider alternates good behaviors (on) and evil

behaviors (off) to avoid being detected. Results demonstrate a higher resilience, which means higher decreases of trust (0.18) against self-promoting attacks, reiterated over time in comparison to Robust-D (0.6) and SGSQ-TM (0.4) algorithms.

With respect to the edge layer, Dhanapala et al. [21] proposes a performance-based trust mechanism in the context of the edge-cloud continuum to optimize the selection and management of heterogeneous cloud resources, detect anomalies, and support self-healing. To evaluate trustworthiness, the authors employ real-time monitoring parameters such as CPU load, memory consumption, and network bandwidth and use self-learning algorithms to discover anomalies and concept drift. In addition, a reputation mechanism is also contemplated to check the quality of the communication network during task execution. A positive interaction fulfils the required performance; otherwise, the interaction is marked as negative. Besides, both direct and indirect trust are calculated, considering the similarity of the recommenders as a mechanism to weigh feedback. In terms of experiments, the authors evaluate the malicious behaviour of nodes when trying to execute tasks being successful (not detected) only in 19.8%. Yet, the authors do not consider misbehaviours or risks across the edge-to-cloud continuum as our system does.

Concerning the Cloud systems, John and Singh [22] underline a fuzzy logic-based model that evaluates the trustworthiness of Cloud Service Providers (CSPs) by analysing Quality of Service (QoS) parameters such as security, privacy, performance, dynamicity, and data integrity. The authors employ a fuzzy inference system to model uncertainty using linguistic variables (e.g., ''low,'' ''medium,'' ''high''). They define trust parameters with triangular membership functions and compute trust scores through fuzzification, rule-based inference, and defuzzification. Furthermore, they also conduct practical experiments to compare various information extraction modules in CSPs. Concretely, rule-based extraction, machine learning, hybrid approach, or deep learning approaches are compared, with the deep learning approach achieving the highest precision (0.93), recall (0.89), and F1-score (0.91). In terms of execution time, the average time in three different use cases is 71 seconds, which is considered a reasonable time. Compared to ours, a potential weakness of their solution is the need to demonstrate whether it can also be deployed for Edge and IoT scenarios, as our approach supports the entire cloud continuum with a single solution.

To the best of our knowledge, we are the first to explicitly target and propose a comprehensive approach for Cyber Risk Assessment (CRA) that combines trust indicators with risk-related information while simultaneously encompassing all the layers involved in a cloud continuum architecture. Additionally, instead of simply providing general principles and guidelines, an issue highlighted in the literature, we develop actual tools that can be used to automatically perform risk assessment in this context, which can notably benefit the industry [4], [5].

The work presented in this paper can be considered as falling under the scope of Dynamic Risk Assessment, which Cheimonidis and Rantos define as a continuous process aimed at identifying and assessing risks to organizational operation in a near real-time manner, with the idea of supporting prompt decisions of how to mitigate these risks [26].

With this regard, Table 1 presents a comparison of the main aspects of some of the most recent frameworks and approaches to dynamic risk assessment available in the literature. From this comparison, it emerges that most of the existing research is focused on Industrial Control Systems, with a particular interest in validating against SCADA environments. In this context, our work emerges as the first one targeting modern cloud infrastructures, where different administrative domains interact. This distinction is particularly relevant given the growing complexity and heterogeneity of cloud continuum environments, which introduce dynamic and distributed threat surfaces not typically addressed in Industrial Control System settings. Furthermore, unlike prior approaches that often emphasize static indicators or device-level activities (such as in the work by Gonzalez-Granadillo et al. [24], where network state snapshots where used to adapt the assessment based on which devices were available at a given moment), our framework integrates platform-level health indicators, offering a more holistic and timely perspective on risk evolution. In contrast to other models, apart from providing useful risk metrics, our approach also manages to support other phases of the Risk Management process, such as Risk Monitoring and Remediation, by combining the dynamic information domain with the static data, allowing real-time indicators to increase the weight of otherwise minor vulnerabilities or to deprioritise them when platform behaviour is healthy.

For example, in our validation scenario the presence of a SQL injection vulnerability was known, but it alone did not trigger an alert. In practice, this may occur for various reasons, such as limited resources for patching, low perceived asset criticality, technical complexity, or planned patching schedules, all of which are typically captured in the organization's risk framing phase (through risk tolerances, thresholds, and management strategies). Likewise, an isolated and moderate increase in outbound traffic would not have raised an alert. However, when these two conditions co-occurred, our system flagged the situation as an active risk and provided an explanation that linked the anomaly to a specific vulnerability class (information leak) and affected layer (Cloud via Edge). This cross-domain correlation is crucial in dynamic environments such as the cloud continuum.

Lastly, a significant challenge faced by many risk assessment approaches is effectively addressing unknown attacks. While our approach does not offer a complete solution to this problem, it provides partial mitigation by enabling risk metrics to increase dynamically during the execution of such attacks. This is possible because low-level health indicators can be impacted by the anomalous behaviors

TABLE 1. Comparison of dynamic risk assessment approaches.

Category	Our Approach	Cheimonidis et al. [23]	Gonzalez-Granadillo et al. [24]	Zhang et al. [25]
Scenario	Cloud continuum	Industrial Control Systems	Industrial Control Systems	Industrial Control Systems
Dynamicity Source	Platform indicators, EPSS	EPSS	Device activity, Mitigation Rules	Detected Anomalies
Threat Analysis	✗	✓	✓	✗
Asset Prioritization	✓	✓	✓	Partial
Vulnerability Prioritization	✓	✗	✗	✗
Risk Model	Aggregative	Bayesian networks	Attack Graphs	Fuzzy Probability Bayesian network
Metrics	Quantitative and Qualitative	Quantitative	Quantitative	Quantitative
Enhancement to other risk management phases	Monitoring/ Remediation	✗	Mitigation	✗
Resistance to unknown attacks	Partial	✗	✗	✗

typically exhibited by an attacker, allowing the system to register and respond to emerging threats in real time. For instance, consider a data exfiltration attack that exploits an undisclosed vulnerability to achieve remote code execution and leak sensitive information from a database. In this case, vulnerability assessment alone would not reveal any warning signs since the flaw is unknown. However, anomalous platform health indicators such as unusual outbound traffic and abnormal resource utilization would be captured by our system, enabling the detection of the ongoing exploitation in real time.

III. OUR SOLUTION

This section presents the details of the solution we developed to perform a comprehensive risk assessment that leverages network, cloud, and trust assessment and the details of how these two domains were combined into a suitable risk model. Lastly, details concerning the implementation of network and trust assessment tools are also provided.

A. MOTIVATING SCENARIO

To ground our research in a tangible application, we consider the scenario of a multi-institutional sensor analytics platform. IoT temperature sensors deployed across various university buildings in Spain report to an edge server that aggregates the data, preprocesses it, and forwards it to a cloud-native data storage cluster hosted in Italy. The layers (e.g. IoT, Edge, Cloud) are therefore operated under two different administrative domains and are subject to independent risk exposures and trust policies determined by their functions.

In this environment, attacks can originate from any layer and propagate both laterally and vertically. For example, a poorly secured IoT device could be exploited to gain access to an edge node's backend interface, which may contain further vulnerabilities that impact the cloud layer. Traditional risk assessment, being predominantly static, evaluates the risk level of an infrastructure at a specific point in time based solely on asset characteristics and contextual information. This approach is ill-suited to the dynamic and evolving nature of threats, particularly in complex, internet-connected architectures. While periodic assessments may help, they typically overlook the real-time activity of threats, leading to less accurate evaluations.

To address this limitation, it is essential to develop a model that integrates ongoing attack data with conventional risk assessments. Effective cyber risk assessment must move beyond static vulnerability analysis to include real-time monitoring, cross-layer correlation, and business context prioritization. This scenario forms the basis for both our framework design and our validation experiments.

B. REFERENCE ARCHITECTURE

Including different types of elements deployed at various technological levels and multiple domains replicates a complex architecture and highlights the interactions between them, thus providing an excellent ground for showcasing a holistic risk assessment. To achieve this goal, we started by identifying the conceptual elements needed for a versatile analysis.

Figure 1 presents a high-level overview of the architecture we envisioned for the risk assessment system, organized into three distinct layers.

At the lowest layer, the system automates data collection from the target scenario. This layer aims to collect as much pertinent data as possible, leveraging various methods and techniques applied to the system under examination.

Data collection components are categorized based on the operational level at which they function, namely, cloud, network, and platform levels. The cloud and network components are part of what we define as the *Network Assessment* domain, which focuses on acquiring information related to service configurations and vulnerabilities. In contrast, data collected at the platform level belongs to the *Trust Assessment* domain, which consolidates and evaluates the raw data gathered from low-level indicators.

The information collected in these domains is annotated and semantically enriched, enabling the generation of

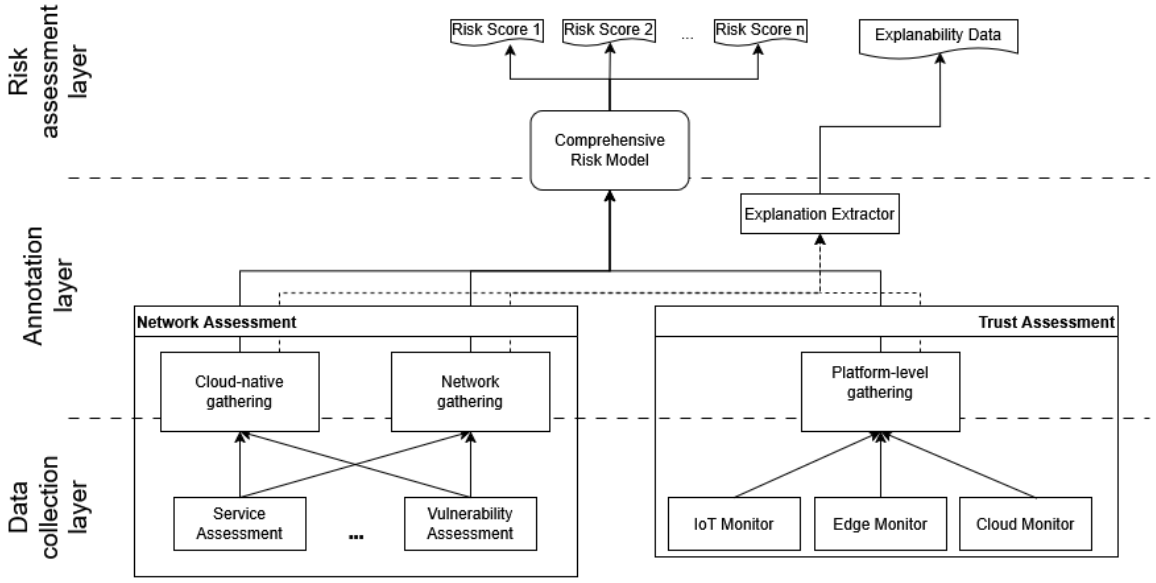


FIGURE 1. High-level representation of the elements needed to perform the comprehensive assessment in the reference architecture.

explainability data that supports and enhances continuous monitoring.

C. DATA COLLECTION LAYER

Data collection is performed by means of a set of specialized modules. These modules are able to cover the most relevant sources of information. This layer can be seen as composed of two separate information domains: the first one is covered by the network assessment process and contains all the data relative to the existing services and the vulnerabilities that they expose. This information is represented by the appropriate information model at each technological layer of the infrastructure (i.e. cloud-native or traditional client-server deployments). The second domain pertains to the platform's health information, which is obtained via trust assessment. This information reflects the real-time state of hardware and resource utilization on the platform. In this context, the data model used is consistent among each technological layer. The following subsections provide more details regarding the different data collection entities.

1) NETWORK GATHERING

The first type of target for network assessment that we identify in our scope of analysis is represented by traditional IT systems. This category includes the components of the architecture that perform specialized tasks and use technologies that are more suited for conventional client-server architectures, where software is executed on bare-metal hardware or with light forms of virtualization, and where the scalability is reduced. In this context, a single machine usually coincides with a specific asset that serves a single purpose and can be uniquely identified by elements such as their IP Address. Referencing the test infrastructure used during validation and presented in Section IV-A, we consider as traditional systems the edge node, which serves the purpose of a web server, and the IoT devices, which act as clients interacting with said web server. For these elements, the collected information is either information that enables the discovery of, or that describes vulnerabilities in the software that is being executed. For this purpose, the data model employed uses five types of entities: IP Addresses, Operating Systems, Software Services, Common Vulnerabilities and Exposures (CVE), and Security Alerts. These specific entities have been selected as they summarize and allow for the discovery of assets that may be relevant to the organization (both physical and intangible assets such as network devices and software). In this context, IP addresses have been chosen as the main identifiers for network entities, as in LAN they generally identify devices univocally. Of course, proper precautions must be taken when analysing particular devices with multiple network interfaces or different domains that may share address ranges, but we consider these as implementation issues that can be easily solved. The information regarding assets serves the purpose of detecting and weighting the presence of risk-relevant entities (i.e. vulnerabilities and security alerts) since it is the only information needed to execute the tools responsible for the vulnerability assessment. For instance, operating system names and versions can be used to retrieve associated CVEs; similarly, a machine's IP address can be

used for service discovery, enabling version fingerprinting and vulnerability identification. Information regarding the services, such as the communication protocols employed and the exposed ports, in turn, may enable more advanced assessments to further yield targeted security alerts. It is important to note that the entities defined in the information models are designed to generalize the components involved in the data collection process. However, many intermediate and more specific data points (e.g. application URLs, software versions, TCP ports and protocols, high-level descriptions, etc.) may still be utilized at the implementation level by the tools performing the actual data gathering. When relevant, this detailed information is captured as attributes of the corresponding entities within the data model.

2) CLOUD-NATIVE GATHERING

The second type of target for the network assessment represents the family of technologies that enable modern, highly scalable, and distributed environments. Specifically, we focus on Kubernetes Systems as they represent a de facto standard for cloud-native applications. Separating these two categories of targets is required since the entities involved in traditional architectures are unsuitable to represent modern ones. For instance, in a Kubernetes environment, the infrastructural elements can not be used to uniquely identify specific assets since multiple, unrelated business-relevant services may be executed on the same platform. In a similar way, IP addresses and similar information can not be used to identify these elements, as this data changes frequently due to the highly scalable and rapidly changing nature of these environments. For this reason, a more abstract data modelling approach is used, which is limited to the following four entities: Components, Packages, CVE, and Misconfigurations. The model of the relationships between these entities is shown in Figure 2. Also in this case, intermediate information is used by the tools involved in the information gathering and stored as attributes of the main entities.

It can be noted that cloud-native environments also include Misconfigurations as possible sources of risk. This is due to the fact that apart from a different data model, we also chose to apply a different information-gathering approach for these targets. Differently from traditional software deployments, the Kubernetes platform offers more readily available insights into the workloads that are being executed, especially with regard to their configuration. Therefore, we decided to integrate also an analysis of this information into the network assessment.

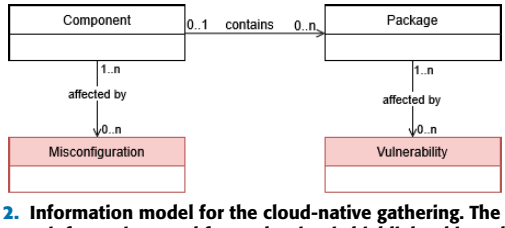


FIGURE 2. Information model for the cloud-native gathering. The risk-relevant information used for evaluation is highlighted in red.

3) PLATFORM-LEVEL GATHERING

The Level of Trust (LoT) is a concept designed to evaluate the reliability of network services in modern networks. Central to this evaluation is the Level of Trust Assessment Function (LoTAF), a component in charge of scrutinizing the implementation of relevant security measures. Acting as an impartial, two-way tool, LoTAF assists both trustors (users) by providing informed decision support and trustees (network providers) by offering feedback on adherence to trust criteria. In terms of trust management, the main functionalities of LoT are organized through the LoTAF components (depicted in Figure 3 as blue boxes).

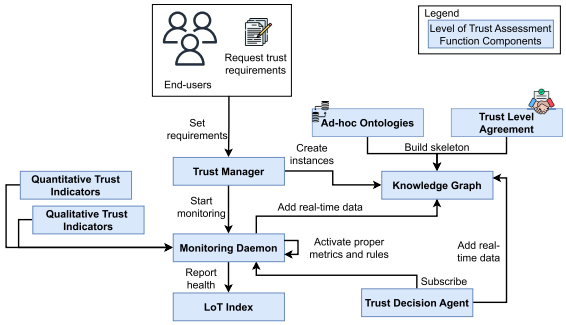


FIGURE 3. Abstraction of the functionalities to support a level of trust assessment.

The generic architecture of the LoTAF needs a *Knowledge Graph* to register evidence and predict deviations from the trust requirements stipulated in the *Trust Level Agreement (TLA)*. The *Knowledge Graph* introduces in turns multiple benefits for LoTAF since it encapsulates an ontology that builds the skeleton of collected trust information, enables applying inference techniques to acquire new knowledge, and facilitates a standardised format to share information across instances. Concerning TLA, this term presents a generic template in which stakeholders like consumers and service providers may declare the terms and conditions that resources or services must fulfill in terms of health indicators officially. LoTAF achieves this thanks to the role of the *Trust Manager* who conducts the main steps. This component must analyze both quantitative (metrics derived from frameworks such as RFC 9417 [27]) and qualitative (third-party recommendations, user experiences, or reputational data) trust indicators. The Trust Manager uses a *Monitoring Daemon* to verify the real-time health of network resources or services settled in the TLA. The *Monitoring Daemon* activation also encompasses the configuration of pre-defined rules to observe specific symptoms on assets, like service assurance rules (see Appendix B to visualize specific examples of rules). It is worth mentioning that the set of predefined rules determines both the subset of objective parameters that are pivotal for auditing and under what thresholds those parameters may reflect probable misbehaviour. Defining these rules is crucial for the correct behaviour of the *Monitoring Daemon*. The *Monitoring Daemon* is also in charge of continuously updating the initial trust index with

real-time information. A set of objective parameters need to be defined for this purpose (see VI). In parallel, the Monitoring Daemon should also store real-time information in a knowledge graph to boost future inference actions from the *Trust Decision Agent*.

In our implementation, the *Monitoring Daemon* gathers features from CPU, sensors, memory, network, process, and disk usage periodically. Therefore, it gathers information both hardware and software characteristics using pre-defined system calls in Linux (see Appendix B to visualize the calls). It compiles this information to deliver a new LoT index that represents the asset's health (more details on the computation are available in Section III-E). Moreover, given the absence of a well-known trust-specific ontology of the cloud continuum, usable for the Trust Decision Agent for inferences purposes, for our prototype, we have designed an *Ad-hoc Ontology* for LoTAF [28].

D. ANNOTATION LAYER

The Annotation Layer produces additional value by leveraging the points of contact and synergies of the two information domains. This value is expressed as additional information (i.e. annotations) that are associated to the data gathered from the assessments. In particular, the trust assessment provides reliable indicators of anomalous behaviour by monitoring the underlying platform (i.e. hardware and kernel level information), and the network assessment gathers high-level information that describes possible enabling causes of anomalous behaviour. The annotation layer correlates these two types of data to achieve detailed explainability of the anomalies that affect the systems, an essential element for guiding the remediation procedures that may be triggered during the monitoring phase. The advantages of this are twofold. First, the selection of the probable cause behind the anomaly can be guided and prioritized by the individual score assigned to the risk-relevant information (as detailed in Section III-E). Therefore, the process will automatically integrate the organization's objectives, risk tolerances, and business priorities. Secondly, because this information is collected across all domains of the architecture and their respective nodes, knowing the connections and interactions between components enables the analysis to be extended. This allows vulnerabilities present in adjacent nodes to be referenced when explaining anomalies on a given node, thereby identifying potential kill chains. Since the current focus of our research is to demonstrate the benefits of such an approach, as a proof of concept we relied on a static categorisation scheme to provide a semantic link between the trust and network assessment domains. In particular, both the risk-relevant data (i.e. vulnerabilities, misconfigurations, and security alerts) and the trust assessment rules have been manually annotated with one of five attack categories: *Command Execution, Denial of Service, Information Leak, Privilege Escalation, and Bypass*. This manual mapping enables correlations between anomalous platform behaviors and potential enabling causes identified at the service or configuration level, thus supporting explainability in the assessment results. While this annotation process is static and handcrafted in the current prototype, it effectively demonstrates how enriched semantics can guide cross-layer reasoning. In future work, we plan to automate this step through Artificial Intelligence (AI)-based techniques, allowing for more scalable, adaptive, and fine-grained classification of threats as new attack patterns emerge. The association process relies on identifying key characteristics in both the trust rules and the risk-relevant data. For trust rules, this involves monitoring symptoms such as abnormal process creation, excessive CPU or memory usage, or unexpected network activity, which can be indicative of specific attack types. For risk-relevant data, descriptive attributes such as CVE entries, misconfiguration patterns, or security alert tags are used to map the vulnerability or weakness to a corresponding category. By aligning these two sources, each anomaly detected at the platform level can be linked to a higher-level attack class (e.g. mapping abnormal process execution to *Command Execution* or anomalous outbound traffic combined with a SQL vulnerability to *Information Leak*), thereby enabling explainable cross-layer analysis.

E. RISK ASSESSMENT LAYER

The key concept of our research work is to combine the information from different assessment sources to produce additional value for security operators. For how they are designed, the Trust Assessment and the Network Assessment have natural synergies, which, when joined, provide a holistic risk assessment of a platform and the services deployed on it. The objective of the Risk Assessment Layer is to synthesize these synergies into a risk measure that clearly represents the state of the system.

As a matter of fact, the trust assessment offers observability over the lower level execution context of the software, analyzing indicators such as network statistics, process counts, and CPU usages, while on the other hand, the network assessment is responsible for discovering and analyzing the software and contextual information of the organization to identify vulnerabilities and possible sources for adversarial actions. Most importantly, the two elements are complementary since they can compensate for the gaps in the other. In particular, while the trust measurements can help detect risks that went unnoticed by the network assessment component, it is also true that the network information gathered can be linked to the trust measurements, allowing for the explanation of anomalous behaviours via direct association with vulnerabilities that may be undergoing exploitation.

For this reason, we conceived a system that is able to leverage all the aspects in play during the two types of analysis.

In the model we have used for our validation, the data collected and annotated by the Network Assessment Tool is evaluated according to the model described in Equation (1);

the higher the value, the riskier the data entity is. Each entity is evaluated individually, yet, for each entity type, a weight field is implemented to adjust the data's importance based on the related asset's business relevance. In fact, during the discovery, the relations between software services or cloud-native components and their risk-relevant information are kept to allow this operation. The weight assignment is currently static, in the sense that before the assessment, the organization is expected to perform an initial risk framing process to define the scope of the analysis and specify said weights, according to their business objectives. The same logic applies to the risk-relevant information, which can also be assigned static scores depending on characteristics such as the type of attack that a vulnerability may enable. In our approach, weights act as a modulation factor for the risk scores, adjusting their impact according to asset relevance. The baseline weights are derived from attack statistics published in annual RiskLens reports [29], which provide distributions of incidents by asset type and industry sector. Assets identified during the network analysis are statically mapped to these categories based on the services and software they execute, and the corresponding weights are applied. When organizations have prior risk assessments, these parameters can be further customized to reflect their specific attack history and risk exposure. This design employs statistical weighting in combination with recognized risk scoring methodologies [30], [31], providing a solid baseline while preserving the flexibility for security experts to adapt the weighting scheme to their specific context and risk models.

$$\mathrm{N} = \text{Weight} \times \text{Parent weight} \times \text{Risk Amplifier} \quad (1)$$

The only information that is assigned a Risk Amplifier (hence, the only entities participating in the score computation) are Vulnerabilities, Alerts, and Misconfigurations since they are unmistakably indicators of risk. Specifically, the risk scores assigned to data of type Vulnerability are calculated following the approach proposed by [30], which utilizes the CVSS properties of the associated CVE to derive a quantitative risk value. The main difference in our method is the use of the EPSS Score[1] in place of the Exploitability parameter from the Temporal CVSS score. The EPSS Score offers frequently updated estimations of the likelihood of exploitation for a given vulnerability, enabling more timely and tailored risk assessments. In contrast, risk scores for Security Alerts and Misconfigurations are based on the severities provided by the detecting tools, which typically draw from vendor advisories and established scoring systems such as CWSS and CCSS [32], [33]. All computed scores are normalized to a 0–100 scale for consistency. The other entities are used to discover more risk-relevant data and propagate business-relevance weights (i.e. through the Parent Weight factor). These weights, expressed on a scale from 0 to 1.0, must be defined at the organizational level and assigned to assets (i.e. devices, services, or any other relevant components of the infrastructure). Similarly, weights can be specified for categories of risk-relevant information that may have varying importance to the organization. For example, a company may choose to prioritize vulnerabilities that enable information disclosure on certain assets, while on other components they may favour those that primarily impact availability.

The individual scores for the risk-relevant entities are then aggregated to produce summarising metrics. Since, during assessments, upper-risk bounds are more significant than lower ones, we used a weighted average that assigns increasing weights to higher scores and produces what we reference as the Overall Network Score ($\mathbf{N_{overall}}$). Additionally, we summarise the upper 1% of scores into an average called the Peak Network Score ($\mathbf{N_{peak}}$).

Concerning the individual scores for the trust assessment T, the Trust Decision Agent calculates a trust score using time windows of five seconds. In this regard, the Monitoring Daemon compiles the average value of each dimension, i.e. CPU, sensors, memory, network, process, and disk, and applies a weighting factor per dimension. Weighting factors can be fixed based on configuration rules during the activation of the Monitoring Daemon. Therefore, depending on enforcement scenarios we can assign a higher impact to certain dimensions. By default, each weighting factor w_x is set to 0.16 (1/6). Nevertheless, weighting factors might be adjusted in real time when system starts gathering recent history data. Methods like Z-score magnitude [34] helps system to find out how unusual the values of a dimension is compared to its recent history. Therefore, Z-score may notice deviation with respect to the mean value or standard deviation of a given dimension and, in consequence, adjust all weighting factors to assign more weight to the dimension spotting potential anomalies. Likewise, the specific features and respective thresholds to be evaluated during the five-second time windows should be declared in the rules based on the whole available low-level indicators that the Monitoring Daemon may collect (see GitHub for in-depth details [35]). Equation (2) presents how the trust score is calculated. Unlike the other scores, a higher value indicates better platform health and, consequently, a lower level of risk.

$$\begin{aligned}\mathrm{T} = {} & w_{\text{cpu}} \cdot \text{CPU} + w_{\text{sen}} \cdot \text{Sensors} + w_{\text{mem}} \cdot \text{Memory} \\ & + w_{\text{net}} \cdot \text{Network} + w_{\text{proc}} \cdot \text{Process} + w_{\text{disk}} \cdot \text{Disk} \quad (2)\end{aligned}$$

where

$$w_{\text{cpu}} + w_{\text{sen}} + w_{\text{mem}} + w_{\text{net}} + w_{\text{proc}} + w_{\text{disk}} = 1$$

As already mentioned, for the combined score computation, we wanted a formulation that could synthesize the two scores into one while simultaneously keeping the information regarding both of them; Equation (3) presents the formulation we decided to adopt, while table 2 provides an overview of the parameters involved. Also in this case, a higher score

[1] https://www.first.org/epss/

indicates a higher risk level.

$$R = \frac{100}{1 + e^{-k(\max(0, T_t - T) + N - c)}} \quad (3)$$

TABLE 2. Description of parameters used in the risk scoring formula.

Parameter	Description
R	Final risk score (ranges from 0 to 100).
T	Assessed trust score of the system (ranges from 0 to 100).
N	Assessed network (& cloud) score of the system (ranges from 0 to 100).
T_t	Predefined trust score threshold introduced to address systems that may have T lower than 100 under normal workloads. T greater than this threshold will not affect R.
k	Scaling factor controlling the logistic function's steepness.
c	Adjustment constant for fine-tuning the function behaviour depending on the organization's risk tolerance. Lower values lead to higher scores.

Equation (3) allows for the computation of a score both using the Overall Network Score, as well as the Peak Network Score; depending on which one has been used we will be referring to the result as either $\mathbf{R_{overall}}$ or $\mathbf{R_{peak}}$. Additionally, qualitative score ranges can be defined depending on organisational needs to more intuitively depict the risk level represented by a score. In the context of this paper, we selected a four-level qualitative representation where scores lower than 30 are considered *Low Risk*, scores between 30 and 60 being *Medium Risk*, *Medium-High Risk* when between 60 and 80, and *High Risk* when greater than 80.

The complexity of the proposed system is not incidental, it is required to accurately represent the interdependencies of a multi-domain architecture. Simpler models that assess risk in isolation (e.g. by statically scoring vulnerabilities or anomalies alone) miss the crucial insight that risk propagates across components. For example, a vulnerability that is not actively exploited may appear low risk, but when paired with an increase in anomalous process activity on the same host, the likelihood of exploitation dramatically increases. Our risk model captures this by combining real-time trust indicators with context-aware vulnerability data, yielding a score that reflects both the state and the exposure of the system.

F. WORKFLOWS

The cyber risk assessment is performed by combining the information obtained by the Network Assessment Tool and the information produced by the Trust Assessment Framework and the underlying LoTAF. The entire process can be seen as separated into two phases: the *Initialization Phase* and the *Monitoring Phase*. Agorithm 1 provides a pseudocode description of the steps involved during the former, while Agorithm 2 shows the actions performed during the latter. While the workflow phases and purposes are generic enough to describe the overall objectives of a general framework, this section reports several details of our implementation.

As the name suggests, the Initialization Phase is executed upon system deployment to establish a baseline that will be used during the system's operation (i.e. the Monitoring Phase).

This phase's purposes are twofold: first, it collects the information needed for the second phase, gathering the data that will be used when anomalous behaviours are detected. The second and more obvious is to provide an initial assessment of the infrastructure upon deployment, leveraging the combination of platform health indicators and detected vulnerabilities: an infrastructure that presents high scores during the initialization phase is likely to require corrective actions to avoid issues during actual operation, either by allocating more resources or by correcting vulnerabilities.

During this phase, the system retrieves the trust score from each node deployed in each of the architecture's layers and then proceeds to discover the services and cloud-native components in execution on the infrastructure. This operation is required so that risk-relevant information can be obtained from the execution of vulnerability assessment tools targeting these entities.

After gathering and storing the risk-relevant information, the computation of a network score takes place: the information regarding vulnerabilities, misconfigurations, and security alerts are evaluated and then aggregated to produce multiple scores. Once all the scores are obtained, the final risk scores are computed. The computation of all the different scores involved is performed according to the methodology defined in Section III-E. At the same time, the collected information is annotated and associated based on their semantics as described in Section III-D, building the explainability data that will be used during the monitoring phase.

It is important to note that any implementation of the data collection phases should follow our proposed order of operation. Indeed, the network assessment process, which may resort to intensive scanning activities, may produce non-trivial stress on the resources of the node under examination, which can, in turn, trigger a decrease in the trust score. Since our objective is to obtain a Trust Score that is as close as possible to the score relative to the normal execution of the node, we require fetching it before introducing this stress.

The Monitoring Phase is performed during the normal operation of the infrastructure. It measures the scores close to real-time, and when significant increases in the scores are detected, it produces explanations of the causes of the variation by leveraging the risk-relevant information from an inter-layer perspective.

By fetching the most up-to-date version of the risk-relevant information from the network assessment and the most recent indicators provided by the trust assessment, it is possible to summarize the overall state of the environment at a specific moment via the derived risk scores. Because

the network assessment process can impose additional load on the platform, the update frequency of the risk-relevant information should be carefully managed, either scheduled periodically during periods of reduced activity or triggered on demand when new entities or significant architectural changes are deployed.

These scores determine whether the system is in an unsafe state based on the risk tolerances of the organization. When a transition from a safe to an unsafe state is detected, automated actions and decisions can be triggered. In the context of this paper, we demonstrate the execution of a workflow that leverages the collected risk-relevant information to automatically provide an explanation for the anomalies that caused the state change. This workflow is based on the assumption that a sudden increase in risk is primarily triggered by real-time platform indicators rather than network assessment data, due to the inherently dynamic nature of the former compared to the more static nature of the latter.

Therefore, this phase begins by periodically fetching the trust score of the monitored nodes and then recomputing the risk scores using the stored information, which can also be updated by re-executing the network assessment. The monitoring phase should refresh the risk-relevant information to obtain updated scores; however, since the process may affect the performance of the infrastructure, it is suggested that its periodicity be fine-tuned according to the operational needs. Incremental updates of the said information (e.g. via Cyber Threat Intelligence (CTI) integration) are left as future work.

Once the needed information is gathered, the scores are recomputed according to the model presented in Section III-E. They are then compared with thresholds defined at an organizational level. If the thresholds are violated, an unsafe state is reached; therefore, the explanatory process is triggered: the service assurance rules (i.e. the *symptoms*, refer to Section III-C3 for more details) that caused the trust score reduction will be fetched and used to identify same-category risk-relevant information, selecting the one with the higher individual risk evaluation as a possible culprit for the node worsened health conditions. In case no associated vulnerability, security alert, or misconfiguration has been detected on the current node, the analysis is repeated based on the risk-relevant information gathered on each node connected to the unhealthy one. This determines an inter-layer risk analysis that enriches anomalous behaviour with relevant risk-aware information.

G. IMPLEMENTATION

The following sections briefly provide lower-level details regarding the choices that have been made to implement the tools to support the types of assessments envisioned, especially regarding the process of gathering information.

1) NETWORK ASSESSMENT TOOL

The Network Assessment Tool used in this research stems from the results presented by Gatti et al. [4]. While the core functioning and principles of the tool remained similar, considerable changes and significant restructuring have been introduced to improve the quality of the assessments and increase the scope of application, by integrating cloud-native information gathering as well as multiple improvements to the scoring system, as described in Sections III-C2 and III-E.

In general, the Network Assessment Tool relies on the CLIPS[2] programming language and reasoning engine to process information and drive the gathering process, which is performed via Python APIs that execute external open source tools such as NMAP,[3] or ZAP[4] or retrieve data from external databases such as the NIST's National Vulnerability Database.[5] For what concerns the cloud-native gathering, the tool relies on the open source vulnerability scanner Trivy[6] to discover the components that constitute the Kubernetes environment as well as detect vulnerabilities and misconfigurations. All the information is processed as facts in the CLIPS knowledge base and used for the discovery of new data. Once the discovery process is completed, meaning that no new facts can be added to the knowledge base, all the information is stored as JSON data to be used during risk evaluation and monitoring.

2) TRUST ASSESSMENT TOOL

Regarding LoTAF, this assessment tool is configured and instantiated on each asset to be monitored in all the administrative domains considered. LoTAF leverages a monitoring agent based on the service assurance for intent-based networking [27]. Thus, LoTAF determines the health score of an asset in a pre-defined time window (e.g. five seconds), leverages an exponential moving average (EMA) to assign a higher weight to recent measurements, and shares real-time data either via a Kafka topic[7] or exports it through JSON files. Python has been used as the main programming language to deploy all the functionalities included in Figure 3. Likewise, Neo4j[8] is leveraged to store knowledge graph information generated by the Monitoring Daemon. Additionally, LoTAF instances provide an HTTP endpoint to look up health scores by filtering the outcomes by timestamps in ISO format or nanoseconds. Similarly, the instances also provide an HTTP endpoint for retrieving the information about the service assurance rules that are currently being triggered. The documentation to interact with LoTAF is delivered via Swagger API [35].

Therefore, it is possible to interact with LoTAF in two ways. The first approach is to subscribe to relevant Kafka topics to access trust values or health scores. The second option is to request trust values and symptoms on demand via the specific HTTP endpoint for a given time period, receiving

[2]https://www.clipsrules.net/
[3]https://nmap.org/
[4]https://www.zaproxy.org/
[5]https://nvd.nist.gov/
[6]https://github.com/aquasecurity/trivy
[7]https://kafka.apache.org/
[8]https://neo4j.com/

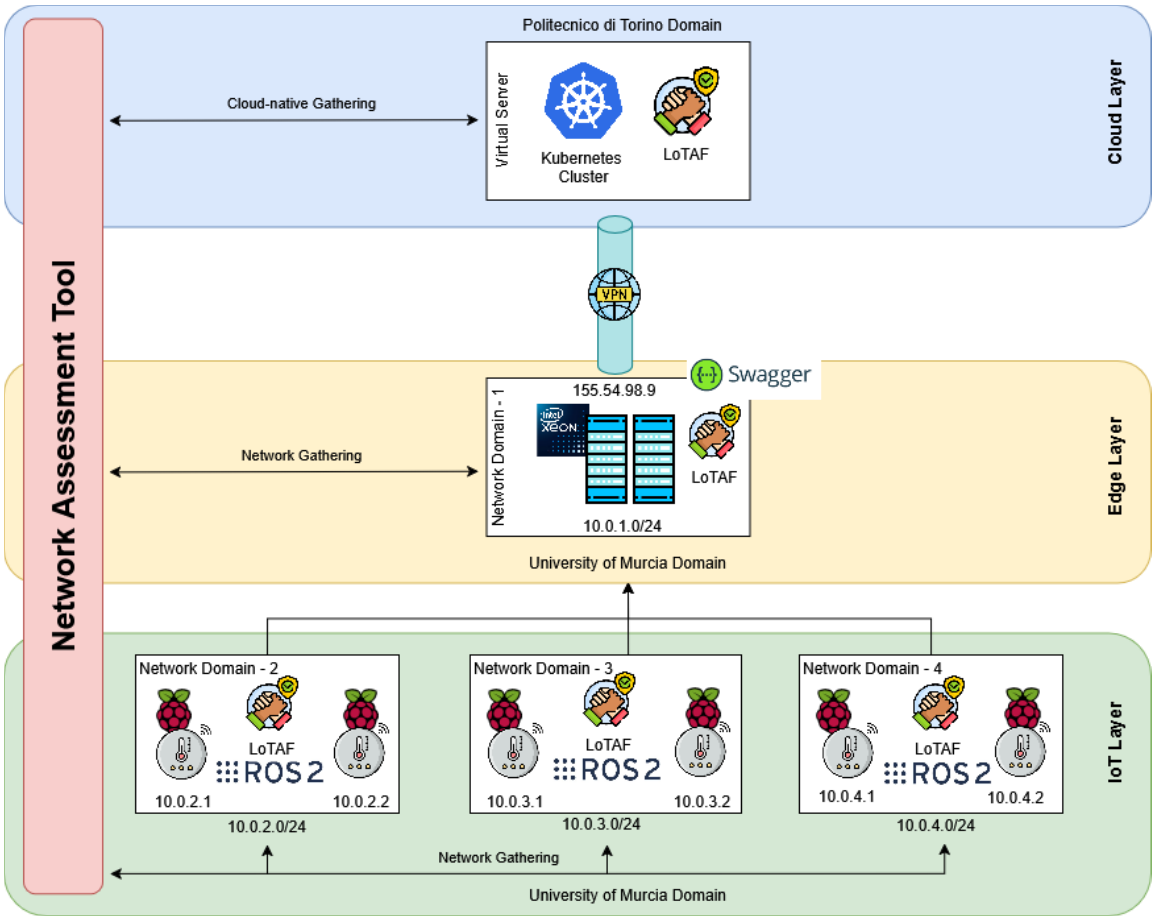


FIGURE 4. Abstraction of the functionalities to support a level of trust assessment and the interactions with the Network Assessment Tool.

the final trust value as a weighted average of all metrics within the selected timeframe. This second option was used in the testing environment presented in Section IV.

IV. VALIDATION

The validation we propose in this research consists of deploying our solution in realistic scenarios to prove how merging information domains with ad hoc risk models benefits the risk assessment process. Moreover, we investigated if combining data from different sources offers better results than considering models independently, both in terms of quality of risk assessment and multi-domain capabilities. To achieve this, we opted to instantiate the developed tools in a realistic environment depicting a cloud continuum architecture, where we simulated attacks from malicious actors.

The following sections, therefore, introduce the contextual setup and the experiments carried out to verify the effectiveness of the conjoined approach.

A. TEST ENVIRONMENT SETUP

This subsection introduces the hardware and software characteristics of the environment where our solution has been validated. Figure 4 displays the configuration at IoT, Edge, and Cloud layers. Following a bottom-up approach, the *IoT Layer* is deployed across three network domains at the University of Murcia, each functioning as an independent subnet. In particular, each network domain contains a cluster of Raspberry Pi 4 boards which simulate IoT devices such as temperature sensors, and transmit their readings to the edge layer for advanced processing. Each Raspberry Pi 4 runs the ROS2 operating system,[9] a Linux distribution oriented to smart devices with resource constraints. Regarding the Edge Layer, the test environment contemplates an Intel Xeon Server, which executes a containerized back-end consisting of a Python web server that supports the layer-specific functions. Such a server carries out close-to-site data processing activities and orchestrates the management and configuration of IoT devices as part of the example application scenario. Last but not least, the Cloud Layer serves as data storage for the IoT sensor readings that are processed and aggregated by the Edge Layer. For this purpose, a simple deployment of a replicated PostgreSQL database has been instantiated on the Kubernetes cluster, offering direct database access to the edge server. Because the enforcement scenario comprises two

[9]https://www.raspberrypi.com/software/

different administrative domains, a VPN tunnel is required to enable secure communication.

Figure 4, shows that the LoTAF component is instantiated on each device across all technological layers to collect quantitative parameters related to both normal and abnormal device behaviors. As previously noted, this monitoring process is guided by a set of statically defined rules; the complete list of the ones employed during the following validation scenarios is provided in Appendix B. It is important to note that the set of metrics reported is not intended to be exhaustive for all possible attack classes. Rather, they were selected to cover the threats considered in our validation scenarios and in the EU-funded projects where our framework has been applied. This necessarily introduces a certain bias, since the metrics emphasize categories such as command execution, information leak, and privilege escalation, which are most relevant to the chosen environments. Nevertheless, this focus is justified because our goal is to demonstrate the feasibility and benefits of combining trust and network-based indicators, not to claim coverage of the entire attack space.

With regard to comprehensiveness, the current metrics provide sufficient sensitivity for the attacks explicitly modeled, but they do not guarantee detection of every possible threat. Similarly, while false positives may arise, our design mitigates this by correlating anomalies across domains (e.g. resource usage anomalies combined with known vulnerabilities), reducing the likelihood of spurious alerts during normal operation.

Lastly, Figure 5 presents a brief overview of the specific tasks performed by the devices in the validation scenarios.

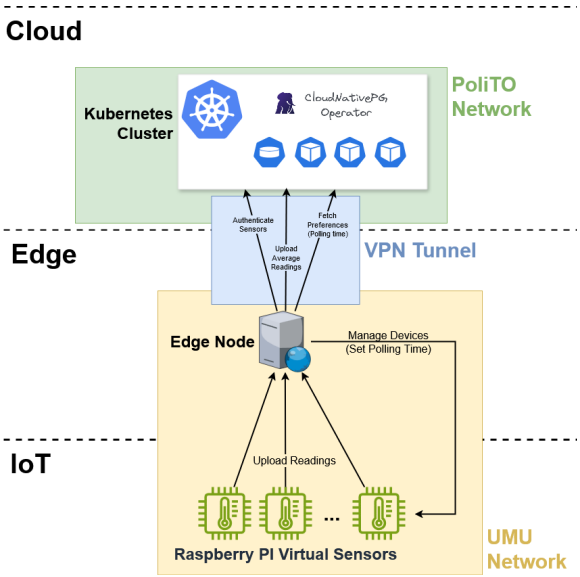


FIGURE 5. The IoT devices produce temperature readings and communicate them to the edge node, which, after authenticating the devices, configures them and periodically uploads the average reading to the cloud layer.

B. VALIDATION SCENARIOS

To verify the effectiveness of the approach we propose, we prepared an environment that is able to highlight the main security issues considered in the data collection layer. Since the two frameworks analyze vulnerabilities (i.e. the Network Assessment Tool) and platform health indicators (i.e. the LoTAF component) that are affected when the system is compromised, it is necessary to introduce vulnerabilities that can be leveraged to perform attacks on the platform.

For this reason, multiple layers, specifically the elements belonging to the IoT Layer and the Edge Layer, have been implemented with exploitable vulnerabilities. The IoT layer nodes are affected by CWE-1391,[10] which consists in the use of weak credentials for the authentication of the services in execution (i.e. the SSH service that is used for device management by the edge server). This weakness enables a scenario where an attacker who has network-level access to the device is able to execute commands on the device itself. The second weakness introduced is on the edge server. The HTTP endpoint used by the IoT layer nodes to upload their data presents an instantiation of CWE-89,[11] better known as SQL Injection, on the parameters used for authentication. This fault allows an attacker that has network-level access to the edge layer (possibly by leveraging the IoT devices compromise) to achieve information leak from the underlying database. Notably, this second vulnerability is instantiated on the edge layer but actually impacts a component (i.e. the database) that is deployed at the cloud layer.

As a preliminary form of validation, it is also important to verify that the vulnerable environments produce higher scores than non-vulnerable environments during the initialization phase as well as the monitoring phase. For this reason, we simulated the deactivation of vulnerabilities. Examining the nodes under the same workload conditions (i.e. the same trust score) shows that the initialization phase indeed produces higher risk scores for the vulnerable environments.

Table 3 provides an overview of the scores obtained across the different layers. Notably, even after deactivating the intended vulnerabilities, certain issues remain. At the Edge Layer, for instance, the absence of specific security headers in HTTP responses is still detected. Similarly, in the Cloud Layer, despite the deployment not being designed to be directly exploitable within our scenarios, the Network Assessment Tool reports several minor CVEs. This is primarily because the deployment is based on an open-source example not intended for production use. While fully securing every portion of our example scenario is out of the scope of this paper, this example is good at showing how our methodology can support addressing security issues during the initial deployment of the infrastructure.

1) SCENARIO 1—INTRA-LAYER DETECTION AND EXPLANATION

The first scenario we propose leverages the comprehensive risk assessment for detecting and explaining anomalies spanning over a single network layer. This scenario is an

[10] https://cwe.mitre.org/data/definitions/1391.html
[11] https://cwe.mitre.org/data/definitions/89.html

TABLE 3. Scores registered at the different layers during the initialization phase. Refer to Table 2 for the parameters descriptions.

Environment	T	$N_{overall}$	N_{peak}	$R_{overall}$ (Quant.)	$R_{overall}$ (Qual.)	R_{peak} (Quant.)	R_{peak} (Qual.)
IoT-vulnerable	93.55	58.0	100.0	59.86	Medium	92.41	High
IoT-non vulnerable	93.55	25.0	25.0	22.27	Low	22.27	Low
EDGE-vulnerable	88.36	49.55	100.0	51.48	Medium	92.96	High
EDGE-non vulnerable	88.36	40.49	50.0	40.28	Medium	52.04	Medium
CLOUD-non vulnerable	99.41	32.14	46.67	29.04	Low	45.84	Medium

"atomic unit" of the networks for which we aim to perform risk computation. Therefore, validating the ability to assess risks precisely in this scenario also allows for inferring the ability to assess risks for larger and more connected networks that span a single domain.

From the risk perspective, the broader implication of this scenario is that by introducing a correlation model (which may consist of inference rules) that bridges the trust assessment and network assessment domains for a single entity, it becomes possible to generate valuable information that would not be available within the information domains when considered independently. The setup begins with the deployment of an IoT device that is subject to standard workloads while in its vulnerable configuration; the device is monitored following the workflow of the corresponding phase. The risk-relevant information was collected during the initialization phase and has not been updated since no relevant changes and updates were applied to the device during the scenario. During the research different organizations with different levels of risk tolerance have been simulated to test the effectiveness of the Risk Models used in the Risk Assessment Layers; however, only numerical tests were conducted as deploying actual infrastructures for all of them was excluded due to the high development efforts needed. The scenarios presented are instantiations of these examples on real-world infrastructure. In the first example, the thresholds for the Overall Risk Score and Peak Risk Score are set to 75.0 and 95.0, respectively, assuming an organization with high-risk tolerances. The initial phase of the experiment shows a baseline Trust Score of 98.37, a reduction caused by low-severity monitoring rules that are expected to be triggered during normal execution. In this state, it registered an Overall Risk Score of 59.86 and a Peak Risk Score of 92.42. Therefore, no explanatory behaviour is triggered. The experiment proceeds with introducing an attacker in the environment, which connects to the IoT device exploiting the weakness on the SSH server and begins the execution of new processes. In a timeframe of 25 seconds (due to the polling time being set to five seconds and the trust score averaging window set to the same amount), the trust score begins to decrease, causing an increase in the risk scores. Ten seconds later, the trust score drops to 71.84, causing both the peak and overall scores to violate the thresholds, determining an unsafe state. As this state is reached, the explanatory behaviour is triggered: the system fetches from the anomalous device the rule that determined the violation of the threshold, which is *"Average process number greater than expected"*. Via the association model, this rule is statically annotated as a rule for monitoring events that initiate *Command execution* attacks on the node. This threat category is obtained from the symptoms and then used while inspecting the Network Assessment's stored knowledge base for vulnerabilities that enable this type of threat. The search identifies the related information with the highest individual risk score, which in the case of the experiment is the alert presented in Listing 1. This data is then sent to the operator as a possible cause of the anomaly.

```
{
    "template": "ALERTEMPLATE",
    "slots": {
        "cwe": "1391",
        "alert": "Found Weak SSH Credentials
            ",
        "description": "The product uses
            weak credentials for securing SSH
             connections, such as default or
            easily guessable passwords. 2
            valid credentials have been found
            : admin:admin, clippy:clippy.
            These credentials have been used
            to establish an SSH connection to
             the device, allowing the
            execution of shell commands.",
        "url": "10.0.2.1:22",
        "category": "Command Execution",
        "cvss": "CVSS:3.1/AV:N/AC:H/PR:N/UI:
            N/S:U/C:H/I:N/A:N",
        "severity": "High",
        "cvss_score": 5.8,
        "impact_score": 3.6,
        "expl_score": 2.2,
        "risk_amplifier": 100.0,
        "weight": 1.0,
        "parent_weight": 1.0
    }
}
```

LISTING 1. Weak credentials vulnerability alert.

Compared to a scenario where only the host-based data are used for detecting anomalies, we report that the anomaly might be detected, but no explanations could

be generated without manual human inspection of the entire system. On the other hand, when only relying on data from vulnerability assessment through state-of-the-art tools, no anomalies would have been detected, even when refreshing the network assessment, as the characteristics of the deployed software, as seen from vulnerability assessment tools, remain unchanged also when under attack. These results show that our approach has a significant advantage.

2) SCENARIO 2—INTER-LAYER DETECTION AND EXPLANATION

The second scenario is designed to take advantage of the capabilities of the comprehensive risk assessment to support the detection and explanation of anomalous behaviour across multiple network layers of the architecture. This scenario represents a set of template network units that span different domains and interact to achieve some business goals in a cloud continuum scenario. While in real-world situations, there can be more heterogeneity (different services, different types of business relations), in the end, all these scenarios can be traced back to inter-layer interactions among different domains, which is what we depict in our example. Hence, this validation in this scenario can be generalized to the multi-domain case.

From a generalized perspective, the scenario shows how, by introducing an association between the two different information models, it is possible to produce additional, novel information. In particular, this scenario offers an even broader example of an application with respect to the first one since it demonstrates how information can be associated across independent technological domains to obtain impactful insights. The setup is similar to the first scenario, with the difference being that the Cloud Layer is under monitoring. Under normal workloads (i.e. receiving from the edge server the aggregated data from two IoT devices), the cloud platform registers a trust score of 99.41, which implies ideal health information. Assuming a higher criticality of the cloud layer for the organization, the thresholds have been lowered to trigger whenever the overall risk score increases over 70.0, that is, the middle point of the *Medium-High* risk level, or when the peak score crosses from *Medium-High* to *High* risk levels (i.e. increases beyond 80.0). The following step of the experiment consists of the execution of an attack to the edge layer, exploiting the introduced vulnerability. The attack employs the open source tool SQLMap[12] and mimics the behaviour of an attacker that is leveraging SQL Injection vulnerabilities for exfiltrating information from a database. Specifically, the tool performs blind time-based SQL injection attacks to extract the names of the database tables as a part of a process that aims at retrieving the entries contained in such tables. This type of attack requires significant amounts of data to be sent to the database, which in turn translates to anomalous quantities of network traffic departing from the cloud layer. Once the attack is launched, the trust score begins to decrease after five polling cycles (corresponding to a 25-second time window). The decrease is again gradual due to the averaging nature of the score. Therefore, three other polling cycles are required before a violation of the Peak Risk Score threshold is achieved since the associated score reached 81.68, crossing into the high-risk level. Once again, the automated explanatory behaviour is triggered by this unsafe state. The system fetches the rule that determined the score reduction from the cloud, obtaining the rule *"Anomalous Outbound Traffic"* which was statically annotated with events that may result in *Information leak* attacks, during the definition of the ruleset. This time, the network assessment knowledge base for the cloud infrastructure does not contain matching risk-relevant information with significant individual scores, due to the lack of impactful vulnerabilities at this layer. The system resorts to analysing the nodes connected (currently, the analysis is limited to the directly connected nodes, i.e. the edge server, but we plan to extend this aspect as future work), applying the same logic to the risk knowledge base of the edge server. The process results in the identification of the security alert shown in Listing 2.

```
{
    "template": "ALERTEMPLATE",
    "slots": {
        "cwe": "89",
        "alert": "SQL Injection",
        "description": "SQL injection may be
             possible.",
        "url": "https://10.0.1.1/
            upload_reading?sensor_api_key=a&
            sensor_id=1+AND+1%3D1+--+&
            temperature=2.2",
        "category": "Information Leak",
        "cvss": "CVSS:3.1/AV:N/AC:H/PR:N/UI:
            N/S:U/C:H/I:N/A:N",
        "severity": "High",
        "score": 5.8,
        "impact_score": 3.6,
        "expl_score": 2.2,
        "risk_amplifier": 100.0,
        "weight": 1.0,
        "parent_weight": 1.0
    }
}
```

LISTING 2. SQL Injection vulnerability alert.

As a last step, the alert is prompted to the monitoring operator, together with the information regarding the node where it was discovered, as a probable cause for the risk level increase on the node under monitoring.

As in Scenario 1, no explanatory action would be available relying only on trust assessment, and no detection would be possible with only network assessment. However, the significance of the validation in this scenario lies in the fact that the layers involved, which are logically interconnected, belong to different administrative domains, demonstrating how our approach is well-suited for applications that rely on

[12]https://sqlmap.org/

the cloud continuum paradigm. In general, the same applies to scenarios where different layers that share resources and data with the organization but belong to separate administrative domains are involved, for example, where a third-party supply chain needs to be considered during risk analysis and monitoring.

V. DISCUSSION

The validation scenarios presented in this paper show that our approach introduces significant benefits to the risk assessment process by combining the capabilities offered by two types of assessments. These improvements are a direct consequence of introducing a comprehensive risk model that merges and evaluates the information domains, and an association model that correlates the data entities pertaining to these domains. This aspect is extremely significant since it highlights how our proposed methodology can be generalized for a broader context of applications and risk assessment scenarios. We demonstrate that once a trust assessment model to detect anomalies and a network assessment model that analyzes security issues are instantiated, adding rules to connect these types of information is enough to introduce noteworthy explanatory capabilities that can benefit security administrators.

While this research focused on the cloud continuum as a specific context due to its novelty and consequent gaps in research, we also highlight that this approach could be applied to various IT infrastructures or, more in general, wherever multi-domain interconnections between IT elements are present. In fact, another particularly relevant example of a context where our research could excel is the analysis of risks related to the supply chain. With our methodology, an automated and all-inclusive risk assessment can be performed. At the same time, anomalies detected in organization systems could be linked with risks and vulnerabilities detected in third-party domains, enabling a much-needed holistic approach for risk management in this area [36].

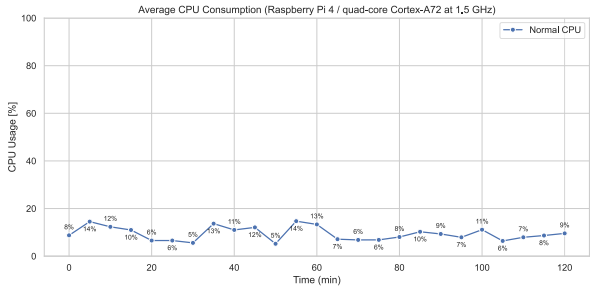
Given the recent trend in cyberattacks that increasingly exploit the weaknesses in the supply chain [37], investigating methods for automating the continuous monitoring of the security posture of organizations and its extension to entire supply chains must be considered highly relevant. Given the impact it can have nowadays, researchers should be incentivized to investigate this research area more. Our work proves that carefully analysing the information that may allow basing risk decisions (and computing risk formulas) and building tools to gather them can improve the possibility of automatically estimating risk exposure.

Moreover, developing reference scenarios of complex multi-domain infrastructures, like in many other fields, may help researchers focus on developing new models and tools for risk management without having to imagine and develop real industrial-grade infrastructures.

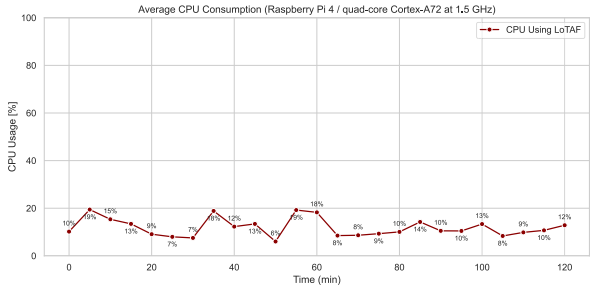
With respect to the complexity and overhead introduced by our approach, several considerations can be made. The complexity of the score computation grows linearly with both the number of discovered risk-relevant entities and the number of defined monitoring rules, although the impact of the latter is negligible compared to the former. Furthermore, since the two components required for the final risk score computation can be independently assessed, these processes can be parallelized. The score computation itself is performed on a device that is external to the infrastructure. This device maintains the knowledge bases containing risk-relevant information and retrieves trust scores from the infrastructure nodes.

The most resource-intensive task in our process is data collection, which involves per-node analysis whereby the external device interacts individually with each node in the infrastructure. The complexity of this analysis is directly related to the number of components present on each node (e.g. software services, cluster pods, software packages). However, since the analyses for different nodes are independent, they can also be parallelized.

Regarding the LoTAF deployment, each node must run an instance of the monitoring daemon required for computing the trust component. Through our experiments, we validated that the overhead introduced on less capable IoT devices is negligible (see Figures 6 and 7). In the case of CPU overhead, on a Raspberry Pi 4 with quad-core Cortex-A72 at 1.5 GHz, the LoTAF increased the average CPU usage of by 2.51%.



(a) Average CPU measurement

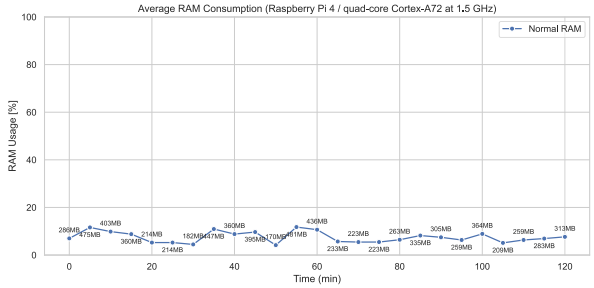


(b) Average CPU measurement using LoTAF

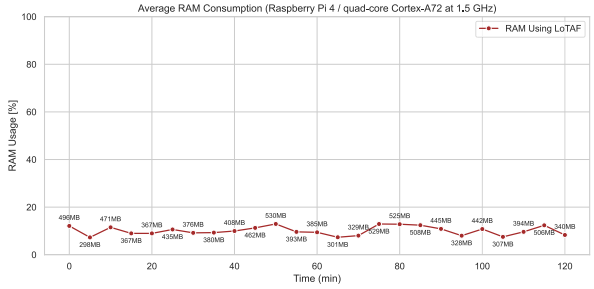
FIGURE 6. CPU Overhead of LoTAF on an idle Raspberry Pi 4 device.

When it comes to the average RAM consumption (see Figure 7), our tests measured usage under normal workloads of 308.28 MB, which corresponds to 7.53% of the total amount of RAM (4 GB). This is due to the fact that the desktop environment and its associated services take up memory. Just like CPU, average RAM usage was slightly

increased to 413.31 MB, which is 10.09% of the total amount of RAM. Therefore, LoTAF did not noticeably affect the performance of the capacity-constrained devices.



(a) Average RAM measurement



(b) Average RAM measurement using LoTAF

FIGURE 7. RAM Overhead of LoTAF on an idle Raspberry Pi 4 device.

Finally, Figure 8 presents the performance of the explanatory process demonstrated during validation, executed on a consumer-grade CPU. The results confirm the linear scalability with respect to both the number of neighboring nodes to inspect and the number of risk-relevant entities associated with each connected node.

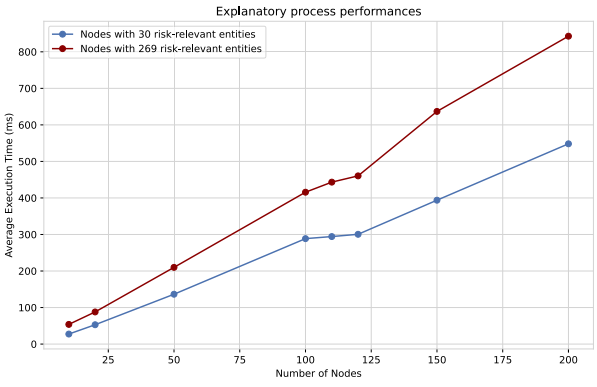


FIGURE 8. Execution times of the explanatory process with respect to the number of nodes and to the number of risk-relevant entities per node.

Nevertheless, our research is not without its limitations. For instance, adversarial behaviors that successfully mimic normal workloads may not trigger a noticeable reduction in platform health. Addressing this issue would require the integration of more advanced anomaly detection mechanisms to complement the low-level indicators we currently use. Nonetheless, the primary objective of this research was to develop an approach that integrates static and dynamic information to enhance risk management, rather than to focus on sophisticated anomaly detection techniques.

Algorithm 1 Initialization Phase: Risk Score Computation Under Normal Workloads

Require: *Nodes* – set of network nodes of the infrastructure
Ensure: Populated risk knowledge base and risk scores under normal workloads

for all $target\_node \in Nodes$ **do**
  initialize_lotaf($target\_node$)
  $T \leftarrow$ retrieve_trust_score($target\_node$)
  if $target\_node \in$ Cloud_layer **then**
    cloud_gathering($target\_node$)
  else
    network_gathering($target\_node$)
  end if
  $N_{overall}, N_{peak} \leftarrow$ compute_network_score($target\_node$)
  $R_{overall}, R_{peak} \leftarrow$ compute_risk_score($T, N_{overall}, N_{peak}$)
end for

Another potentially significant limitation is our reliance on standardised vulnerabilities and weaknesses, which inherently restricts our system's ability to detect unknown zero-day attacks or newly discovered ones. As discussed, we offer partial mitigation for such scenarios, as zero-day exploits can still impact the risk score by degrading platform health indicators. However, the explanatory process triggered by these variations would be unable to specify the root cause of the increased risk, since it is based on known vulnerabilities.

VI. CONCLUSION & FUTURE WORK

In this study, we proposed a holistic cyber risk assessment framework designed to address the challenges posed by the growing complexity of cloud continuum environments. By integrating trust and network assessments across multiple layers and domains via a comprehensive risk model, our approach enables the estimation of risk exposures on different technological layers and the detection and contextual explanation of security threats in close to real-time. The results demonstrate the effectiveness of merging low-level platform health indicators with high-level risk-relevant information, providing security operators with valuable insights.

Regarding the future work that we expect to carry out stemming from this research, we have identified many interesting opportunities for improvement. Since our objective was not to propose an association model between the two information models but rather to show the implications and advantages that this action could bring, we are considering increasing the complexity and capabilities of the association rules that are currently defined statically. One promising path that we could follow to achieve this is leveraging AI and particularly Large Language Models (LLMs) to process risk-relevant information and trust assessment information and

TABLE 4. LoTAF rules deployed at the IoT layer.

Description	Category (Severity Level)	Path	Rule
No free memory available	Denial of service (Orange)	/node/vm/mem	free<10
No free hugepages available	Denial of service (Orange)	/node/bm/mem	pages_free==0
No CPU idle time for 1 min	Denial of service (Orange)	/node/bm/cpus/cpu	1min(idle_time)≤1
No CPU idle time for 1 min	Denial of service (Orange)	/node/vm/cpus/cpu	1min(idle_time)≤1
Sensor reached maximum temperature	Denial of service (Orange)	/node/bm/sensors/sensor	input_temp≥max_temp
Zombie Threads	Denial of service (Orange)	/node/bm/proc	zombie_count >0
Receive Errors Peak	Denial of service (Orange)	/node/bm/net/if	1min(dynamicity(rx_error))>100
Receive Drops Peak	Denial of service (Orange)	/node/bm/net/if	1min(dynamicity(rx_drop))>10000
Transmit Errors Peak	Denial of service (Orange)	/node/bm/net/if	1min(dynamicity(tx_error))>100
Transmit Drops Peak	Denial of service (Orange)	/node/bm/net/if	1min(dynamicity(tx_drop))>100
No Buffers available for GSO	Denial of service (Orange)	/node/kb/net/if	dynamicity(gso_no_buffers)>0
Packet dropped due to missing mbuf	Denial of service (Orange)	/node/kb/net/if	dynamicity(rx_no_buffer)>0
Missing buffer in ip4-input	Denial of service (Orange)	/node/kb/net/if	dynamicity(ip4_input_out_of_buffers)>0
Fragment chain too long	Denial of service (Orange)	/node/kb/net/if	dynamicity(ip4_input_fragment_chain_too_long)>0
Receive Errors Peak	Denial of service (Orange)	/node/kb/net/if	1min(dynamicity(rx_error))>100
Receive Drops Peak	Denial of service (Orange)	/node/kb/net/if	1min(dynamicity(rx_drop))>10000
Transmit Errors Peak	Denial of service (Orange)	/node/kb/net/if	1min(dynamicity(tx_error))>100
Transmit Drops Peak	Denial of service (Orange)	/node/kb/net/if	1min(dynamicity(tx_drop))>100
Low Buffer Availability	Denial of service (Orange)	/node/kb/mem	(buffer_free/buffer_total)<0.1
Average process number greater than expected	Command execution (Red)	/node/bm/proc	dynamicity(sleep_count)+dynamicity(run_count)>85
Bare Metal CPU High Load	Command execution (Red)	/node/bm/cpus/cpu	idle_time<30

TABLE 5. LoTAF rules deployed at the Edge layer. These rules have not been triggered during validation.

Description	Category (Severity Level)	Path	Rule
Anomalous Outbound Traffic	Information Leak (Red)	/node/bm/net/if	1min(dynamicity(tx_packets))>4000

TABLE 6. LoTAF rules deployed at the cloud layer.

Description	Category (Severity Level)	Path	Rule
No free memory available	Denial of service (Orange)	/node/vm/mem	free<10
No free hugepages available	Denial of service (Orange)	/node/bm/mem	pages_free==0
No CPU idle time for 1 min	Denial of service (Orange)	/node/bm/cpus/cpu	1min(idle_time)≤1
No CPU idle time for 1 min	Denial of service (Orange)	/node/vm/cpus/cpu	1min(idle_time)≤1
Sensor reached maximum temperature	Denial of service (Orange)	/node/bm/sensors/sensor	input_temp≥max_temp
Zombie Threads	Denial of service (Orange)	/node/bm/proc	zombie_count >0
Receive Errors Peak	Denial of service (Orange)	/node/bm/net/if	1min(dynamicity(rx_error))>100
Receive Drops Peak	Denial of service (Orange)	/node/bm/net/if	1min(dynamicity(rx_drop))>10000
Transmit Errors Peak	Denial of service (Orange)	/node/bm/net/if	1min(dynamicity(tx_error))>100
Transmit Drops Peak	Denial of service (Orange)	/node/bm/net/if	1min(dynamicity(tx_drop))>100
No Buffers available for GSO	Denial of service (Orange)	/node/kb/net/if	dynamicity(gso_no_buffers)>0
Packet dropped due to missing mbuf	Denial of service (Orange)	/node/kb/net/if	dynamicity(rx_no_buffer)>0
Missing buffer in ip4-input	Denial of service (Orange)	/node/kb/net/if	dynamicity(ip4_input_out_of_buffers)>0
Fragment chain too long	Denial of service (Orange)	/node/kb/net/if	dynamicity(ip4_input_fragment_chain_too_long)>0
Receive Errors Peak	Denial of service (Orange)	/node/kb/net/if	1min(dynamicity(rx_error))>100
Receive Drops Peak	Denial of service (Orange)	/node/kb/net/if	1min(dynamicity(rx_drop))>10000
Transmit Errors Peak	Denial of service (Orange)	/node/kb/net/if	1min(dynamicity(tx_error))>100
Transmit Drops Peak	Denial of service (Orange)	/node/kb/net/if	1min(dynamicity(tx_drop))>100
Low Buffer Availability	Denial of service (Orange)	/node/kb/mem	(buffer_free/buffer_total)<0.1
Anomalous Outbound Traffic	Information leak (Red)	/node/bm/net/if	1min(dynamicity(tx_packets))>2500
Anomalous Outbound Traffic	Information leak (Red)	/node/bm/net	1min(dynamicity(snmp_IpOutRequests))>2000

associate them based on their natural language descriptions, along with high-level contextual information regarding the organization. This will not only enhance scalability by eliminating the need for manual information classification but also enable correlation between detected events, allowing the reconstruction of complex, multi-step attacks. Regarding the contextual information, we also plan to investigate the integration of CTI in the risk assessment process to have real-time capabilities for collecting risk-relevant information, such as active threats targeting the organization's business sector. Likewise, we intend to add automatic trust management in CTI solutions to evaluate the reputation or confidence of shared information and its issuers in cross-domain environments where CTI sharing is carried out across

Algorithm 2 Monitoring Phase: Detection of Anomalous Behaviour and Explanation via Risk-Relevant Information

Require: *Nodes* – set of nodes; *THRESHOLD_OVERALL*, *THRESHOLD_PEAK* – organisational-level constants
Ensure: Explanations for detected anomalies

while True **do**
  for all *target_node* ∈ *Nodes* **do**
    $T \leftarrow$ retrieve_trust_score(*target_node*)
    if *update_risk_kb*(*target_node*) **then**
      if *target_node* ∈ Cloud_layer **then**
        cloud_gathering(*target_node*)
      else
        network_gathering(*target_node*)
      end if
    end if
    $N_{overall}, N_{peak} \leftarrow$ compute_network_score(*target_node*)
    $R_{overall}, R_{peak} \leftarrow$ compute_risk_score($T, N_{overall}, N_{peak}$)
    if $R_{overall} \geq$ *THRESHOLD_OVERALL* $\vee R_{peak} \geq$ *THRESHOLD_PEAK* **then**
      symptoms $\leftarrow$ *fetch_triggered_rules*(*target_node*)
      explanation $\leftarrow$ *inspect_risk_kb*(*target_node*, *symptoms*)
      if *explanation* $= \varnothing$ **then**
        connected_nodes $\leftarrow$ *retrieve_connected_nodes*(*target_node*)
        for all *connected_node* ∈ *connected_nodes* **do**
          explanation $\leftarrow$ *inspect_risk_kb*(*connected_node*, *symptoms*)
          if *explanation* $\neq \varnothing$ **then**
            break
          end if
        end for
      end if
    end if
  end for
end while

multiple stakeholders. Since the current implementation only considers risk-relevant information from the first hop of the anomalous node, we plan to explore more advanced algorithms to incorporate farther nodes in the analysis, aiming to detect and explain more complex, multi-step attack chains. Furthermore, we aim to reduce the impact of the Network and Cloud Information Gathering processes on the trust score so that near real-time vulnerability detection can also be achieved. Lastly, our research outputs provide a first input for a remediation process aimed at suppressing the anomalies. Therefore, future work will also cover investigations regarding integrating methods of automatic enforcement of actionable playbooks and general remediation strategies, including deploying additional security controls, to achieve a complete risk management workflow that begins from the assessment and covers both the monitoring and remediation of risks.a

APPENDIX A
ALGORITHMS

See Algorithms 1 and 2.

APPENDIX B
LoTAF RULES USED DURING THE VALIDATION SCENARIO

See Tables 4–6.

REFERENCES

[1] M. Chernyshev, Z. Baig, and S. Zeadally, "Cloud-native application security: Risks, opportunities, and challenges in securing the evolving attack surface," *Computer*, vol. 54, no. 11, pp. 47–57, Nov. 2021.

[2] S. Moreschini, F. Pecorelli, X. Li, S. Naz, D. Hästbacka, and D. Taibi, "Cloud continuum: The definition," *IEEE Access*, vol. 10, pp. 131876–131886, 2022.

[3] I. Lee, "Cybersecurity: Risk management framework and investment cost analysis," *Bus. Horizons*, vol. 64, no. 5, pp. 659–671, Sep. 2021.

[4] G. Gatti, C. Basile, and G. Perboli, "An expert system for automatic cyber risk assessment and its AI-based improvements," in *Proc. IEEE 47th Annu. Comput., Softw., Appl. Conf. (COMPSAC)*, Jun. 2023, pp. 1434–1440.

[5] O. Akinrolabu, J. R. C. Nurse, A. Martin, and S. New, "Cyber risk assessment in cloud provider environments: Current models and future needs," *Comput. Secur.*, vol. 87, Nov. 2019, Art. no. 101600.

[6] *Guide for Conducting Risk Assessments*, Comput. Secur. Division, Inf. Technol. Lab., Nat. Inst. Standards Technol. (NIST), Gaithersburg, MD, USA, Sep. 2012, doi: 10.6028/NIST.SP.800-30r1.

[7] *Managing Information Security Risk: Organization, Mission, and Information System View*, Comput. Secur. Division, Inf. Technol. Lab., Nat. Inst. Standards Technol. (NIST), Gaithersburg, MD, USA, Mar. 2011, doi: 10.6028/NIST.SP.800-39.

[8] *The NIST Cybersecurity Framework (CSF) 2.0*, NIST, Gaithersburg, MD, USA, Feb. 2024.

[9] M. Fumagalli, G. Engelberg, T. P. Sales, Ì. Oliveira, D. Klein, P. Soffer, R. Baratella, and G. Guizzardi, "On the semantics of risk propagation," in *Research Challenges in Information Science: Information Science and the Connected World*. Cham, Switzerland: Springer, 2023, pp. 69–86.

[10] J. Kosińska, B. Baliś, M. Konieczny, M. Malawski, and S. Zieliński, "Toward the observability of cloud-native applications: The overview of the state-of-the-art," *IEEE Access*, vol. 11, pp. 73036–73052, 2023.

[11] M. De Donno, A. Giaretta, N. Dragoni, A. Bucchiarone, and M. Mazzara, "Cyber-storms come from clouds: Security of cloud computing in the IoT era," *Future Internet*, vol. 11, no. 6, p. 127, Jun. 2019.

[12] I. Stellios, P. Kotzanikolaou, M. Psarakis, C. Alcaraz, and J. Lopez, "A survey of IoT-enabled cyberattacks: Assessing attack paths to critical infrastructures and services," *IEEE Commun. Surveys Tuts.*, vol. 20, no. 4, pp. 3453–3495, 4th Quart., 2018.

[13] C. Vellaithurai, A. Srivastava, S. Zonouz, and R. Berthier, "CPIndex: Cyber-physical vulnerability assessment for power-grid infrastructures," *IEEE Trans. Smart Grid*, vol. 6, no. 2, pp. 566–575, Mar. 2015.

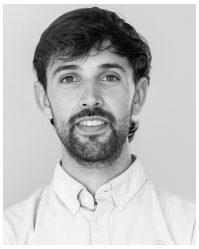
[14] B. M. Khorsandi, M. A. Habibi, G. Avino, S. Barmpounakis, G. Bernini, M. Ericson, B. Han, I. L. Pavón, J. M. J. Valero, D. R. Lopez, B. Richerzhagen, R. B. Rouphael, M. Saimler, L. Scheuvens, C. K. Schindhelm, P. Schneider, T. Svensson, and S. Wunderer, "Enabling Hexa-X 6G vision: An end-to-end architecture," in *Proc. Joint Eur. Conf. Netw. Commun. 6G Summit (EuCNC/6G Summit)*, Jun. 2024, pp. 676–681.

[15] M. Ghoraishi, M. S. Siddiqui, M. Compastié, S. Mhiri, C. Ntanos, M. Kontoulis, D. López, A. Lioy, E. Markakis, and S. B. M. Baskaran, "ITrust6G: Zero-trust security for 6G networks," *Methods*, vol. 5, no. 6, pp. 411–416, 2024.

[16] K. Kandasamy, S. Srinivas, K. Achuthan, and V. P. Rangan, "IoT cyber risk: A holistic analysis of cyber risk assessment frameworks, risk vectors, and risk ranking process," *EURASIP J. Inf. Secur.*, vol. 2020, no. 1, pp. 1–18, Dec. 2020.

[17] I. Ud Din, A. Bano, K. A. Awan, A. Almogren, A. Altameem, and M. Guizani, "LightTrust: Lightweight trust management for edge devices in industrial Internet of Things," *IEEE Internet Things J.*, vol. 10, no. 4, pp. 2776–2783, Feb. 2023.
[18] J. M. J. Valero, V. Theodorou, M. G. Pérez, and G. M. Pérez, "SLA-driven trust and reputation management framework for 5G distributed service marketplaces," *IEEE Trans. Dependable Secure Comput.*, vol. 21, no. 4, pp. 1863–1875, Jul. 2024.
[19] A. B. Can and B. Bhargava, "SORT: A self-organizing trust model for peer-to-peer systems," *IEEE Trans. Dependable Secure Comput.*, vol. 10, no. 1, pp. 14–27, Jan. 2013.
[20] P. Saikia, S. K. Deka, and M. Devi, "A robust trust framework for detecting on–off attacks in cognitive radio sensor networks," in *Proc. IEEE Calcutta Conf. (CALCON)*, Dec. 2024, pp. 1–6.
[21] I. Dhanapala, S. Bharti, A. McGibney, and S. Rea, "Towards a performance-based trustworthy edge-cloud continuum," *IEEE Access*, vol. 12, pp. 99201–99212, 2024.
[22] J. John and K. J. Singh, "Trust value evaluation of cloud service providers using fuzzy inference based analytical process," *Sci. Rep.*, vol. 14, no. 1, p. 18028, Aug. 2024.
[23] P. Cheimonidis and K. Rantos, "A novel proactive and dynamic cyber risk assessment methodology," *Comput. Secur.*, vol. 154, Jul. 2025, Art. no. 104439.
[24] G. Gonzalez-Granadillo, S. Dubus, A. Motzek, J. Garcia-Alfaro, E. Alvarez, M. Merialdo, S. Papillon, and H. Debar, "Dynamic risk management response system to handle cyber threats," *Future Gener. Comput. Syst.*, vol. 83, pp. 535–552, Jun. 2018.
[25] Q. Zhang, C. Zhou, Y.-C. Tian, N. Xiong, Y. Qin, and B. Hu, "A fuzzy probability Bayesian network approach for dynamic cybersecurity risk assessment in industrial control systems," *IEEE Trans. Ind. Informat.*, vol. 14, no. 6, pp. 2497–2506, Jun. 2018.
[26] P. Cheimonidis and K. Rantos, "Dynamic risk assessment in cybersecurity: A systematic literature review," *Future Internet*, vol. 15, no. 10, p. 324, Sep. 2023.
[27] *Service Assurance for Intent-Based Networking Architecture*, document RFC 9417, 2023.
[28] J. M. J. Valero, J. P. Serrano, and A. S. Gil, "Level of trust assessment ontology for computing continuum," Tech. Rep., 2025.
[29] *2023 Cybersecurity Risk Report*, RiskLens, Washington, DC, USA, 2023. [Online]. Available: https://www.risklens.com/hubfs/Content/reports/RISK_RiskLens%20Annual%20Report.pdf?hsLang=en
[30] M. U. Aksu, M. H. Dilek, E. I. Tatli, K. Bicakci, H. I. Dirik, M. U. Demirezen, and T. Aykir, "A quantitative CVSS-based cyber security risk assessment methodology for IT systems," in *Proc. Int. Carnahan Conf. Secur. Technol. (ICCST)*, Oct. 2017, pp. 1–8.
[31] J. Jacobs, S. Romanosky, B. Edwards, I. Adjerid, and M. Roytman, "Exploit prediction scoring system (EPSS)," *Digit. Threats: Res. Pract.*, vol. 2, no. 3, pp. 1–17, Jul. 2021.
[32] P. M. Mell and K. Scarfone, "The common configuration scoring system (CCSS): Metrics for software security configuration vulnerabilities," Comput. Secur. Division, Inf. Technol. Lab., Nat. Inst. Standards Technol. (NIST), Gaithersburg, MD, USA, Tech. Rep., Dec. 2010.
[33] *CWE—Common Weakness Scoring System (CWSS)*, MITRE Corp., Bedford, MA, USA, Jun. 2023. [Online]. Available: https://cwe.mitre.org/cwss/cwss_v1.0.1.html
[34] A. S. Yaro, F. Maly, P. Prazak, and K. Malý, "Outlier detection performance of a modified Z-score method in time-series RSS observation with hybrid scale estimators," *IEEE Access*, vol. 12, pp. 12785–12796, 2024.
[35] J. M. J. Valero, J. P. Serrano, and A. S. Gil, "Level of trust assessment function APIs," Tech. Rep., 2025.
[36] A. Ghadge, M. Weiß, N. D. Caldwell, and R. Wilding, "Managing cyber risk in supply chains: A review and research agenda," *Supply Chain Manag., Int. J.*, vol. 25, no. 2, pp. 223–240, Nov. 2019.
[37] *2024 Verizon Data Breach Investigations Report*, Verizon Bus., Basking Ridge, NJ, USA, 2024. [Online]. Available: https://www.verizon.com/business/resources/T735/reports/2024-dbir-data-breach-investigations-report

GABRIELE GATTI received the M.Sc. degree in cybersecurity, Politecnico di Torino, in 2022, where he is currently pursuing the Ph.D. degree with the Department of Control and Computer Engineering. He is with the TORSEC-Research Group. His research focuses on the automation of cyber risk assessment and management, as well as the integration of cyber risk models into modern cloud infrastructures.



JOSÉ MARÍA JORQUERA VALERO received the M.Sc. and Ph.D. degrees in computer science from the University of Murcia. He is currently a Postdoctoral Researcher with the CyberDataLab, University of Murcia. His scientific research interests include trust management, security, 5G, intent-based management, continuous authentication, and cybersecurity.

MANUEL GIL PÉREZ is currently an Associate Professor with the Department of Information and Communication Engineering, University of Murcia, Spain. His scientific activity is mostly focused on cybersecurity, including intrusion detection systems, trust and reputation management, and security operations in highly dynamic scenarios.

CATALDO BASILE (Member, IEEE) received the M.Sc. and Ph.D. degrees in computer engineering, Politecnico di Torino, in 2001 and 2005, respectively. He is currently an Associate Professor with the Politecnico di Torino. His research interests include policy-based security management, general models for detecting, resolving, and reconciling security policy conflicts, software protection, and software attestation.