SLA-driven trust and reputation management framework for 5G distributed service marketplaces

José María Jorquera Valero, Vasileios Theodorou, Manuel Gil Pérez, and Gregorio Martínez Pérez, *Member, IEEE*

***Abstract*—The fifth generation (5G) of mobile telecommunications is characterized by massive growth in the number of stakeholders, interconnected devices, and available services distributed under different administrative domains. Distributed marketplaces aim at facilitating stakeholders in the quest and hiring of third party resources and services. Establishing trustworthiness in such an open ecosystem is a cornerstone for the final deployment of these marketplaces in 5G networks and beyond. Hence, building trust management systems that ensure the selection of reliable parties or assets in 5G distributed marketplaces is essential. Thus, a reputation-based trust management framework is proposed to analyze stakeholder behavior patterns and predict trust scores to establish trustworthy relationships across domains. Furthermore, an Service Level Agreement (SLA)-driven reward and punishment mechanism is designed and developed on top of the reputation-based trust framework. Such a mechanism enables continuously adapting trust scores by gathering breach predictions, breach detections, and SLA violations in real time. Furthermore, an edge-based use case is presented to contextualize our reputation-based framework in a tangible enforcement scenario. In conclusion, three experiments were conducted on real-life testbeds demonstrating that our framework fairly distinguishes bad-mouthing attacks with 67% accuracy, when 50% recommenders are corrupted, and is resilient to continuous misbehavior bursts.**

***Index Terms*—Trust framework, reputation, SLA-driven, 5G, distributed marketplace.**

I. Introduction

5G and beyond networks envision enforcement scenarios where stakeholders intend to maximize their business profits while ensuring a high level of Quality of Service (QoS) and Quality of Experience (QoE) to consumers and end users. Yet, stakeholders are sometimes not capable of meeting the stated requirements themselves, and therefore, third parties are necessary to guarantee the signed QoS [1]. By means of such third parties, stakeholders can cover certain peak workloads by hiring or purchasing on-demand services and resources, e.g., computing or network resources. Thereby, billions of business relations across operators belonging to different domains are conceived in the foreseeable future, in which the selection of trustworthy third parties is a capital decision [2].

To deal with on-demand service, resource, and infrastructure provisioning, distributed marketplaces present a fruitful solution since they enable to assemble, through a cross-domain platform, both providers who desire to offer their capabilities and consumers who look for purchasing or hiring available services or resources to satisfy a contract [3]. Thence, distributed marketplaces aim to enable the secure and trustworthy trading of heterogeneous resources in dynamic 5G ecosystems and facilitate cross-domain and multi-party collaborations. Conventionally, marketplaces permitted users to apply several filters in order to encounter a subset of stakeholders who comply with imperative constraints and considerations such as category, geographic location, price, and hardware or software requirements, among others. Nevertheless, trust has not normally been considered as a dimension to filter or rank potential candidates in distributed marketplaces. Because trust is one of the fundamental pillars for building 5G networks [4], a dominant challenge is to determine which stakeholders are trustworthy and reliable from an initial set of candidates who previously met the basic constraints to provision 5G services and resources between different domains [5].

Trust models are one of the utmost important approaches considered in the literature to cope with a trustworthy third party selection because they can profile stakeholders to determine a trust level. Nowadays, other approaches such as Distributed Ledger Technologies (DLTs) [6] and Trusted Execution Environments (TEEs) [7] are also being contemplated as a root of trust for 5G scenarios. Nonetheless, they do not generally analyze real-time stakeholders' behavior to enable or disable given actions based on their trust levels, as some trust models normally do. On the contrary, trust approaches based on DLT and TEE solutions tend to guarantee characteristics such as non-repudiation, runtime isolation, tamper-resistant, etc., which are mostly linked to the intrinsic characteristics of the hardware or software of such technologies and not on stakeholders' behavior. In this way, reputation-based trust models accomplish the above statements as they allow estimating future stakeholder behaviors from historical data and reliable recommendations from third parties [8].

Because trust is a long-term concept, reputation-based trust models should enable not only evaluating a set of candidates before starting a business relationship but also adapting trust levels once relationships are in progress. In this vein, trust models usually consider a continuous update module that is in charge of identifying events in real time and triggering the proper decisions to adapt current trust scores. Thus, the update module is totally aligned with the dynamism idea of many 5G ecosystems as distributed marketplaces also follow [9]. In cross-domain and multi-stakeholder scenarios, as the representative one under 5G marketplaces, Service Level Agreements (SLAs) need to be signed in order to legalize settlements among

José María Jorquera Valero, Manuel Gil Pérez, and Gregorio Martínez Pérez are with the Department of Information and Communications Engineering, University of Murcia, 30100 Murcia, Spain (e-mail: josemaria.jorquera@um.es; mgilperez@um.es; gregorio@um.es) *(Corresponding author: José María Jorquera Valero)*

Vasileios Theodorou is with the Research & Development Unit, Intracom S.A. Telecom Solutions, 19002 Peania, Greece (e-mail: theovas@intracom-telecom.com).

stakeholders. As a result of such agreements, multiple tasks are initialized across domains to monitor the Service Level Indicators (SLIs), forecast possible breach predictions, and identify SLA violations [10], to name but a few. In this sense, events generated during the whole SLA life-cycle management, for instance, breach predictions, breach detections, and SLA violations, are really meaningful for updating a previously computed trust score. The principal reasons are: (i) they are produced after starting a business relationship; (ii) they are generated in real time since other components are continuously monitoring them; and (iii) they are linked to the current stakeholder's behaviors so these may be used to characterize them.

Few trust management models considered performance measurements, related to SLA settlements, as the principal dimension in recent years to determine a starting trust score on a target stakeholder [11], [12]. Yet, there is an absence of reputation-based trust frameworks supporting multi-party collaborations for distributed marketplaces where real-time breach predictions and breach detections are contemplated as dimensions to readjust ongoing trust relationships (as the next section underlines). Therefore, this article at hand proposes a trust and reputation management framework for 5G distributed marketplaces which additionally describes in detail a statistics mechanism to adapt trust scores considering historical stakeholders' behaviors as well as the current breach prediction and detection events.

In order to enhance the development of trustworthy communications in 5G networks and to cover the gaps mentioned above, the principal contributions of this article are:

- A reputation-based trust management framework to ensure a reliable ecosystem for distributed marketplaces in which stakeholders look for trustworthy resource and service providers. The framework analyzes not only service providers but also their services and resources offered, which bring us to detect whether a service or resource started to act strangely. Besides, the reputation-based trust management framework fulfills the zero trust principle [13] since trust should not be taken for granted regardless of whether a business relationship is established with a stakeholder who belongs to our same domain (intra-domain) or an external one (inter-domain). Note that the zero trust principle also entails avoiding the fact of assigning an outdated trust value to the same stakeholder if the previous relationship ended and we are going to start a new one.
- An SLA-driven reward and punishment mechanism has been designed and developed as part of the continuous update module. Such a fully automatic mechanism leverages breach predictions and breach detections, appearing in real time, together with SLA violations and the impact of trust as main features to adapt trust scores in an ongoing relationship. By means of such an SLA-driven mechanism, we intend to enhance their trust model, which is based on historical interactions and recommendations, via objective features as well as take advantage of a gap in the literature not explored in depth.
- A real use case (UC) covered by the 5GZORRO H2020 European project [14] has been presented. The UC showcases through an architecture design how the reputation-based trust framework can be smoothly integrated. Such a framework is contextualized in an edge scenario, in particular, the 5GZORRO distributed marketplace. In addition to that, multiple experiments have been performed to investigate accuracy, performance, and resilience in real infrastructures such as 5GBarcelona and 5TONIC.

The remainder of this article can be outlined as follows. Section II reviews the current SLA-based trust models in the literature as well as solutions employing SLA events for creating reward and punishment mechanisms. Section III describes the four modules of our reputation-based trust management framework spotlighting how the framework introduces a novel SLA-driven mechanism to update ongoing trust relationships. Section IV presents the integration and experiments of our framework in the 5GZORRO distributed marketplace. Finally, Section V recaps the main conclusions of the present work and future research lines.

II. Related Work

This section analyzes the literature dealing with SLA-based trust management models as solutions to guarantee a trustworthy ecosystem for large 5G provisioning scenarios. Furthermore, it also reviews trust models which considered SLA events (i.e., SLA violations, breach predictions or detections, etc.) for elaborating reward and punishment mechanisms to continuously update trust scores.

Regarding SLA-based trust models, Li et al. [15] leveraged trust credit as a mechanism to rank service providers (SPs) before negotiating SLAs. The trust credit measured how SPs behaved (*competence*) and how they are behaving (*integrity*). Regarding competence, Rough Set theory was used to forecast the negotiation success rate. In addition, predicted QoS values were used to detect degradation and determine integrity via Bayesian Networks, once an SLA is terminated. The outputs enhanced the SLA compliance by about 34.5% compared to matchmaking-based ranking. Also dealing with checking real network behavior, Ma et al. [16] proposed a time-dependent and deep learning-powered trust evaluation method. Thus, the similarity between predicted and real follow-up behaviors is considered as a trust value, being measured as the distance between the central points of two clusters. Such a distance was also useful for creating a reward and penalty mechanism based on network behaviors, where a significant deviation between central points entailed a trust decay based on a hyperbolic tangent function. On the contrary, nearly identical behaviors ameliorated devices' trust. Experiments displayed that the long short-term memory (LSTM) algorithm achieved a 0.008 mean squared error (MSE) and 96.4% accuracy as well as provided stable trust predictions. In [11], Aslam et al. presented a trustworthiness assessment mechanism to analyze the service trust of Social Internet of Things (SIoT) instead of provider trust. Thereby, service trust was an aggregated parameter from transaction and execution times plus availability. Besides, a social relationship factor considered the degree of intimacy between the service requester and provider. Experiments

TABLE I: Comparison of SLA-driven trust management models.

Solution	Year	Environment	Metrics	SLA-based Reward & Punishment	Cross-domain	Real-time Making-decisions	Zero Trust	Open Source
[17]	2019	Cloud	Reliability, satisfaction, and recommendations	✗	✓	✓	✗	✗
[11]	2020	SIoT	Transaction, availability, execution time, social relationship factor	✗	✗	✗	✗	✗
[12]	2020	Cloud	Response and execution times, availability, bandwidth	✓	✗	✓	✗	✗
[18]	2020	Cloud	Deviation and nearness degrees, user satisfaction	✓	✗	✓	✗	✗
[16]	2021	IoT	Network behavior similarities	✓	✗	✓	✗	✗
[15]	2022	IoT	Competence, integrity	✗	✓	✓	✗	✗
[19]	2022	MEC	Processing success, incompliance, and user termination ratios and throughput	✓	✗	✓	✗	✗
[20]	2022	Cloud	Service availability, response time, bandwidth, throughput, compliance, etc.	✗	✓	✓	✗	✗
[21]	2023	Cloud	Reliability, transitivity, dynamics, service quality, cost, flexibility, accuracy	✓	✓	✓	✗	✗
Our	2023	Cloud	Breach prediction and SLA violation rates, trust impact	✓	✓	✓	✓	✓

showcased that an increase in QoS entailed an increase in service trust and the relationship followed a partially linear nature.

From a different domain, Li et al. [17] supported users' decision-making in a cloud service marketplace via a three-layered trust model. Such a model established end-to-end trust relationships as well as asymmetric evaluations between layers. To measure trust, reliability and satisfaction were inferred from the direct and indirect trust. Provider and user's trust were updated based on performance and feedback, respectively, during transactions. Regarding accuracy, their solution achieved a 70% satisfactory transaction rate in cloud marketplace. Similarly, Muralidharan and Anitha [20] proposed a reputation-based mechanism to supervise that Cloud Providers (CPs) meet the QoS levels declared in SLAs. To this end, the authors defined multiple performance levels, based on technical and non-technical parameters, using fuzzy sets to weigh them. Afterward, a broker contrasted consumer rating and its own rating about the CP to estimate multi-criteria trust on them. The proposed trust model achieved more accurate reputation scores, using identical metrics, than the cloud service trust evaluation model (CSTEM) but no more detailed experiments were performed. Guo et al. [21] presented a trust model for cloud environments based on characteristic factors and SLAs. This model enhances the precision of service cost and quality assessments, as well as the identification of malicious entities, through a negotiation and monitoring mechanism. It effectively combats spoofing, coordination, and defamation attacks, leading to a high trade success rate. By leveraging self-recommended trust and SLAs, it fosters trust relationships between entities, thereby improving the efficiency of selecting the best providers. Compared to MDTES, TrueTrust, and CSRTM models, as described in [21], it has proven to be more effective in resisting attacks from various dishonest entities and in identifying dishonest providers.

When it comes to SLA violations, they usually entail cost penalization and business termination, in the worst case. Yet, Badshah et al. [12] introduced a performance-based SLA framework to maximize provider revenue and customer satisfaction. Prior to finishing an SLA, the authors proposed an adaptive penalization approach for proportionally diminishing provider reputation and helping future customers. Three thresholds were set up to apply penalties based on the percentage exceeded, always less than the initially agreed 10%. Similarly, Zhang et al. [18] designed a trust model to select trustworthy cloud providers without abnormal behaviors at any specific time. Concretely, the authors measured user satisfaction through the nearness degree and the deviation between QoS declared in the SLA and the current performance. Besides, an adaptive weighting method was formulated from the fluctuation of QoS metrics which dwindled the impact of subjective factors on the trust evaluations. In [19], Monir et al. evaluated the SP compliance to SLAs in Mobile Edge Computing (MEC). To this end, they defined four thresholds which assigned a trust status to the SP via the processing performance during service provisioning. Furthermore, a punishment mechanism accordingly dwindled trust status whether SP intended not to send all rated SLAs, register with a new identity, and exceed the computation time. The simulations displayed an efficient and low time-consuming trust evaluation scheme.

TABLE I shows a comparison between the different SLA-driven trust management models in the literature and our proposal. From the analysis performed in this section, it is noticed that there are two principal gaps regarding solutions focused on an open source approach and the zero trust principle. The former is a weak point when it comes to reusing models or checking accuracy results, being one of the downsides found during our initial research. The latter may be conditioned by the fact that it is a cutting-edge principle introduced by NIST [13], and in consequence, it has not been considered and described by the latest proposals discovered in the literature review. In our proposal, we tackle both characteristics by publishing our source code in an open repository [22] and not granting trust scores to any stakeholder regardless of its origin domain. Additionally, our proposal also recomputes trust scores in case two or more stakeholders had a relationship in the past that finished.

Despite four solutions considered SLA-based trust score update mechanisms (see SLA-based Reward & Punishment column in TABLE I), most of the recent solutions contemplated SLA events as the capital information source to calculate an initial trust score and not as a mechanism to continuously update trust scores in ongoing relationships (after computing the first value). Besides, no solution considered SLA-based

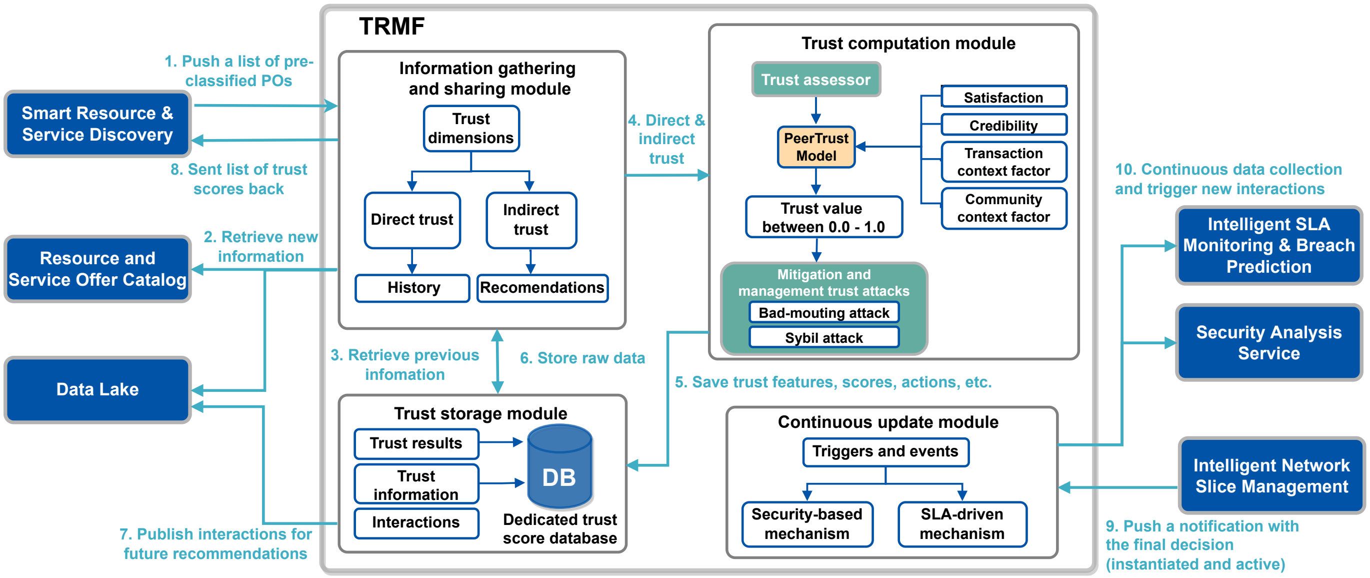
TRMF
1. Push a list of pre-classified POs
Smart Resource & Service Discovery
8. Sent list of trust scores back
2. Retrieve new information
Resource and Service Offer Catalog
Data Lake
7. Publish interactions for future recommendations
Information gathering and sharing module
Trust dimensions
Direct trust
Indirect trust
History
Recomendations
4. Direct & indirect trust
3. Retrieve previous infomation
6. Store raw data
Trust storage module
Trust results
Trust information
Interactions
DB
Dedicated trust score database
Trust computation module
Trust assessor
PeerTrust Model
Satisfaction
Credibility
Transaction context factor
Community context factor
Trust value between 0.0 - 1.0
Mitigation and management trust attacks
Bad-mouting attack
Sybil attack
5. Save trust features, scores, actions, etc.
Continuous update module
Triggers and events
Security-based mechanism
SLA-driven mechanism
10. Continuous data collection and trigger new interactions
Intelligent SLA Monitoring & Breach Prediction
Security Analysis Service
Intelligent Network Slice Management
9. Push a notification with the final decision (instantiated and active)

Fig. 1: Trust and reputation management framework life-cycle

reward and punishment mechanisms which can be applied to cross-domain scenarios. Similarly, there is still a way to go in the research of trust models for decentralized markets since only [17], among the investigations analyzed, tackled the topic. Therefore, our SLA-driven trust and reputation management framework intends to cover the aforementioned gaps for 5G distributed marketplaces.

III. Trust and Reputation Management Framework

This section contextualizes the proposed trust and reputation management framework (TRMF) by providing a high level description of its sub-modules: the *Information gathering and sharing*, the *Trust computation*, the *Trust storage*, and the *Continuous update* (see Fig. 1). Yet, we spotlight on the latter module as it encapsulates the proposed SLA-driven reward and punishment mechanism. Furthermore, the rest of modules, as well as their corresponding equations were previously described in [23].

A. Information gathering and sharing module

The trust and reputation management life-cycle commences through the *Information gathering and sharing* module when the Smart Resource & Service Discovery (SRSD) [24] wishes to analyze a list of available services and resources offered in a distributed marketplace (step 1 in Fig. 1) so as to recognize the most reliable ones (step 8). To this end, this module firstly retrieves raw data from the Product Offers (POs), published in the Resource and Service Offer Catalog [5], to analyze them (step 2). Among raw data, it can be underlined decentralized identifiers (DIDs) of providers and resources, coordinates, current life-cycle status, service specifications, etc. Afterward, it also collects historical information from a dedicated trust database when consumers and providers have had previous trust relationships (step 3). Therefore, they leverage prior interactions as direct trust to predict future behaviors and not as the current value to be assigned directly to the providers (zero trust principle). In addition, consumers can also gather recommendations about specific targets from trustworthy third parties. Note that this capability is enabled thanks to the Data Lake platform [24], which acts as a shared repository where interactions among stakeholders can be openly published and consulted by others. Thereby, consumers should decide whether it is worth asking for feedback, depending on the level of consumers' belief in recommenders. Yet, consumers might still receive dishonest recommendations so mechanisms to mitigate trust attacks should be considered, as described in the following section.

B. Trust computation module

Once all information has been collected, such data are directly shared toward the *Trust computation* module (step 4). The principal goal of this module is to find out a trust score per each PO to be analyzed as well as ease potential trust attacks during the computation steps. When it comes to trust computation, an adapted PeerTrust model is considered as a statistical algorithm since it is principally centered on distributed scenarios where peer-to-peer connections are considered [25]. Besides, the PeerTrust model also brings huge flexibility to researchers due to the fact that they need to figure out how the four main dimensions; satisfaction (S), credibility (Cr), transaction context factor (TF), and community context factor (CF), are going to be designed (see Eq. 1). As a result, it allows adjusting the algorithm to the vast majority of final scenarios.

$$T(u) = \alpha \cdot \left(\sum_{i=1}^{I(u)} S(u,i) \cdot Cr(p(u,i)) \cdot TF(u,i) \right) + \beta \cdot CF(u), \tag{1}$$

where u is the provider for whom wants to determine the trust score $T(u) \in [0,1]$ on the *i-th* interaction; α and β are the weights of each dimension, satisfying that $\alpha + \beta = 1$; and

$I(u)$ is the maximum number of interactions. Due to the fact that the authors of this research work at hand have thoroughly described how all dimensions have been formulated in [23] and [26], we honestly believe such equations should not be reintroduced again as the principal aim is to spotlight a new SLA-driven reward and punishment mechanism. Nonetheless, a high level description of the four dimension objectives is going to be reported so as to comprehend how our adapted PeerTrust model works.

With respect to the satisfaction (S), it measures the acceptance degree of stakeholder u after finishing the *i-th* interaction. This dimension is in turn composed of provider's satisfaction and offer's satisfaction, which estimate provider and offer reputations and returns a value ranging from 0.0 to 1.0. Note that the TRMF is capable of handling 7 types of offers: cloud, edge, radio access network (RAN), spectrum, virtual network function (VNF), network service, and slice. Another dimension of the PeerTrust model is the credibility (Cr) which determines how analogous two entities are when assessing a similar set of stakeholders $(p(u,i))$. To deal with it, a Personalized Similarity Metric (PSM) is utilized. By means of PSM, it is possible both to contrast opinions and to determine the distance of credibility about a set of stakeholders assessed by both stakeholders. Thereby, the lower credibility distance after evaluating the set of stakeholders, the most credible the opinion. It should be pointed out that this mechanism supports the idea of a non-transitive trust model [27], which entails greater disbelief when two stakeholders have not previously interacted with each other. Therefore, it is necessary to identify a set of common stakeholders to find out the belief [28] of such a new stakeholder. Finally, the TRMF also considers two context factors. On the one hand, the TF determines stakeholder's participation associated with the number of PO and provider feedback published in the Data Lake platform through multiple time windows. In this way, the TF gratified stakeholders who divulge their interactions with others in the Data Lake since such an action boosts future stakeholders to look for new interactions at the Data Lake platform, request recommendations from other trustworthy stakeholders, and enlarge the community. On the other hand, the CF gathers feedback from a continuously updated list of trustworthy recommenders who had previous interaction with the target stakeholder u. Additionally, the CF observes untrustworthy recommendations via a recommendation trust mechanism [26]. This mechanism is in charge of evaluating the certainty of recommendations according to the trust in that recommender and the recommendation trust. Thus, CF can come up against traditional attacks in trust models such as bad-mouthing attacks.

C. Trust storage module

Since data privacy-preserving plays a critical role for 5G and beyond 5G networks (B5G), it is really crucial to define how data are going to be handled by the TRMF. As it can be observed in Fig. 1, there are two main information storage sources. When it comes to the private database, it mainly contains personal data, inferred information from raw data, scores, or actions to be addressed (steps 5 and 6). There are two main reasons why this type of information is stored in a private database. First of all, this database is only consulted by the TRMF of its own domain, and in consequence, potential malicious behavior by other stakeholders can be avoided. On another hand, the Data Lake platform is shared across stakeholders, and consequently, stakeholders do not warehouse backups of information in their local domains but they launch on-demand requests since owners can continuously update information. Therefore, each request entails traffic network outside of the TRMF domain. In consequence, the time needed to calculate trust scores could be affected as personal data, inferred information from raw data, and previous scores would be constantly requested. In the case of data to be shared and requested by any stakeholder being registered in the ecosystem, the TRMF leverages the Data Lake platform as a storage source. Interactions among stakeholders are pushed to Data Lake since other stakeholders subsequently employ them to directly request recommendations (step 7).

D. Continuous update module

Owing to the fact that trust changes over time, the continuous update module has a pivotal role in ensuring consistency between real-time events and trust scores (step 9). 5G and B5G networks entail cross-domain environments where stakeholders cooperate under the principle of maximizing their benefits and ameliorating customers' QoS. To enable quality-aware resource and service provision, SLAs are widely utilized as legally binding contracts to commit providers to fulfill the pre-negotiated performance metrics as well as to form a trustworthy provider-consumer relationship. In this vein, SLAs, Service Level Objectives (SLOs), and SLIs enable declaring, defining, and measuring the fulfillment of agreements and indirectly generating events that allow for recomputing trust scores in real time.

Thereby, an SLA-driven reward and punishment mechanism is going to be introduced as an enabler to continuously update trust scores in an ongoing relationship. Besides, such a reward and punishment mechanism is also utilized to determine when a service or resource under a specific trust level should not be able to participate in a relationship anymore due to subsequent misbehaviors. The mechanism follows an agnostic approach so as to be considered by other trust and reputation models. Similarly, the mechanism is not directly related to the types of metrics to check in the SLOs. Hence, it is not necessary to normalize the equations for each possible type of metric, but rather it covers the entire spectrum. Note that the proposed mechanism does not directly leverage the performance measurements provided by a run-time QoS monitoring engine, but it defines a set of statistic features from events generated by breach prediction and detection services [29] and SLA Violation Manager. These statistic features are based on truthful information backed up by a Governance service responsible for defining, validating, and operating the identities, the certificates, and the permissions for all 5GZORRO stakeholders according to Self-sovereign Identity principles [30].

Before thoroughly describing the features and equations of our reward and punishment mechanism based on SLA events (step 10), it should be pointed out that another reward and punishment mechanism was previously defined in [26]. In this case, the mechanism gathered security-based network monitoring events following a time-driven approach. By means of such a mechanism, the trust and reputation management framework was able to early identify feasible threats as well as enhance the security capabilities for network services.

Akin to the security-based mechanism, our SLA-based mechanism follows a time-driven approach to readjust active and trustworthy relationships in real time. In particular, the proposed punishment method $Pu(v,u) \in [0,1]$ on a provider u on whom computations are performed by a consumer v is mainly composed of three dimensions: the *Breach Prediction Rate* ($BPRate$), the *Impact of Trust* over upcoming events ($ITrust$), and the *historical SLA Violation Rate* ($SLAVRate$) (see Eq. 2).

$$Pu(v,u) = \sum_{m=1}^{n} \frac{BPRate(u,m) + ITrust(v,u) \cdot SLAVRate(u,m)}{2}, \tag{2}$$

where v denotes the consumer who updates a trust score; u represents the provider on whom computations are performed; m is the type of SLO metric measured, for example, functionality, availability, performance, requests per minute, etc.; and n is the maximum number of metrics. This method aims at evaluating the impact of breach predictions and detections on a provider's reputation.

When it comes to $BPRate \in [0,1]$, it measures the probability of having SLA violations if a provider continues its actual behavior over time (see Eq 3). Particularly, $BPRate$ determines the percentage of breach predictions ($SLOBP$) for a given metric m over the total breach predictions $k \in [1,n]$ on a target stakeholder u as well as the accuracy level of the intelligent algorithm to prognosticate a prediction ($CertaintyBP$). Note that the algorithm in charge of performing breach predictions, which is part of the 5GZORRO Breach Predictor, which is outside the scope of this paper [29].

$$BPRate(u,m) = \frac{SLOBP(u,m)}{\sum_{k=1}^{n} SLOBP(u,k)} \cdot CertaintyBP(u,m) \tag{3}$$

Another paramount dimension to compute the punishment value is the impact degree of trust on SLA events ($ITrust \in [0,1]$). In this vein, Eq. 4 bears in mind the current trust score between the consumer v and the provider u as well as a trapezoidal fuzzy model $\mu_{trust}(v,u)$. Concerning the trust score, we assume a higher trust score will entail a greater impact on punishment. This assumption comes from the fact that a *fully trustworthy* level can be only reached whether a provider had reliable behavior during a long period, therefore negative events are barely expected.

$$ITrust(v,u) = \left(1 - \frac{1 - T(v,u)}{1 + T(v,u)}\right) \cdot \mu_{trust}(v,u) \tag{4}$$

Regarding the fuzzy model, the principal objective is to determine the membership degree of trust and reputation values with respect to the multiple trust levels defined (see Fig. 2). To this end, fuzzy sets are leveraged as they enable finding out a direct correspondence between reputation values and the impact of the SLA violations on the asset. So as to define the membership degree, a trapezoidal function has been selected since it utilizes linear interpolation to obtain both endpoints of the interval [31], being triangular membership a subcase of this one.

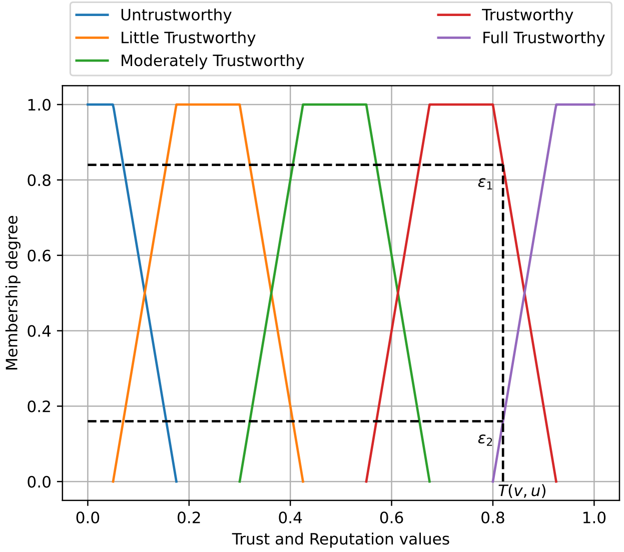

Fig. 2: Trust levels based on the impact degree of trust and reputation values.

As an example, Fig. 2 shows that the likelihood of provider u based on its trust and reputation $T(v,u)$ is directly proportional to the membership degree of provider u for each trust level of a consumer v. According to the likelihood, a consumer v can set up a *trustworthy* or *full trustworthy* levels. Due to the fact, we follow the principle of selecting the highest membership degree between levels involved, the ε_1 linked to the *trustworthy* level would be the option to be selected as a fuzzy set.

Last but not least, we have the last dimension denoted as $SLAVRate^{(t)}(u,m) \in [0,1]$. This dimension defines the growth of SLA violation figures leveraging sliding time windows. In particular, Eq. 5 computes a penalization score ranging from 0 to 1 which determines the deviation between the history and the current SLA violation rates at a given time t.

In Eq. 5, a forgetting factor (ξ) has been contemplated so as to handle the repercussions of time passage over SLA violations. In general terms, the forgetting factor allows utilizing aging functions to gradually adapt to the oblivion of past interactions $SLAVRate^{(t-1)}(u,m)$. Furthermore, an increase or decrease in violation number ($Increment(u,m)$) together with the occurrence level of violations have been also considered. Note that the violation notifications are generated by an SLA Monitoring module, which is also part of the 5GZORRO project [29], via run-time QoS measurement metrics and assessment intervals specified in the SLA settlement.

$$\begin{aligned} SLAVRate^{(t)}(u,m) = SLAVRate^{(t-1)}(u,m) + \\ \xi \cdot Increment(u,m) \cdot \mu_{vio}(u,m) \end{aligned} \quad (5)$$

With respect to the $Increment(u,m)$, we settle to apply penalizations, if and only if new violations appear in the relationship. In this vein, $Increment(u,m)$ defines the growth of the SLA violations over past interactions. Otherwise, the $Increment(u,m)$ is settled to 0 (see Eq. 6) and a reward is applied on the provider u trust score, as depicted in Eq. 7.

$$Increment(u,m) = \begin{cases} \frac{SLAVRate^{(t)}(u,m)}{SLAVRate^{(t-1)}(u,m)}, & \text{if } new\_violation \\ 0 & \text{otherwise} \end{cases} \quad (6)$$

In the case of being applied a reward (Re), its value is directly proportional to the forgetting factor (ξ) applied over the last trust score computed (O_{ts}), as illustrated in Eq. 7. Thus, the greater ξ, the higher the recovery speed of our SLA-driven mechanism.

$$Re(v,u) = \xi \cdot (1 - O_{ts}(v,u)) \quad (7)$$

When it comes to the violation increase, another fuzzy set ($\mu_{vio}(u,m)$) is employed to assess the occurrence level of such violations in the last time window. In this way, three occurrence levels are established: momentary, recurrent, and persistent based on the percentage of increase with respect to the past violation rate $SLAVRate^{(t-1)}(u,m)$, as depicted in Fig. 3. It is worth mentioning that n is always the past violation rate, being updated after new interactions. Contrary to Fig. 2, this fuzzy set selects the highest occurrence level although the membership degree may be the lower one. The reason is that the authors intend to maximize the penalization of an SLA violation rate increment. Thus, a persistent level through ε_4 would be elected as a fuzzy set in Fig. 3.

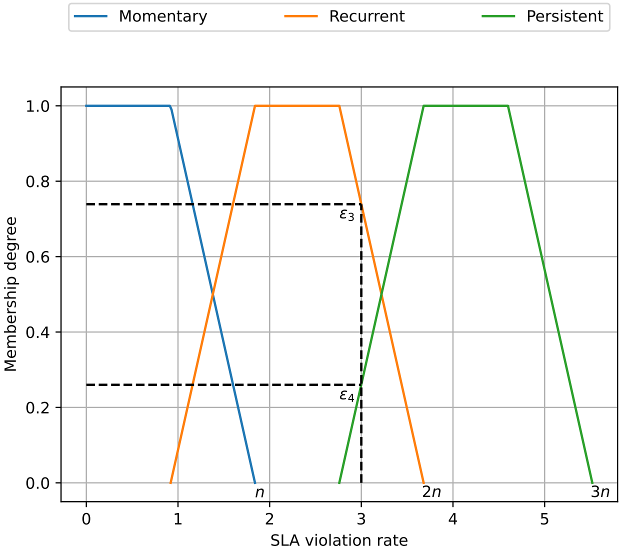

Fig. 3: Occurrence levels based on SLA violation rate values.

Once the three dimensions have been calculated, our SLA-driven reward and punishment mechanism returns a final score $Re(v,u)$ or $Pu(v,u)$ between 0.0 and 1.0, which entails a decrease or increase percentage n on the new trust score (N_{ts}). Afterward, such a reward and punishment is subsequently used by the trust and reputation management framework to accordingly update the last trust score $O_{ts}(v,u)$ and forward the present one to the consumer v, as given in Eq. 8.

$$N_{ts}(v,u) = \begin{cases} O_{ts}(v,u) - Pu(v,u) \cdot \frac{\left(1 - O_{ts}(v,u)\right)}{n}, & \text{if } new\_violation \\ O_{ts}(v,u) + \frac{Re(v,u)}{n} & \text{otherwise} \end{cases} \quad (8)$$

Owing to the fact that this reward and punishment mechanism follows a time-driven approach, its whole life-cycle is constantly triggered when new breach predictions or detections arise in an ongoing business relationship. Therefore, a trust and reputation score could be dwindled at a certain level in which the consumer decides to conclude the current relation, and in consequence, discover new trustworthy providers.

IV. Use Case

This section introduces an emerging paradigm on which trust and reputation are fundamental pillars to optimize orchestration phases. Especially, Section IV-A describes the integration of trust and reputation into a 5G distributed service marketplace use case supported by the 5GZORRO H2020 European project [3]. In addition, this edge-based paradigm also displays a set of experiments to measure both the performance of our proposed SLA-driven reward and punishment mechanism and the whole reputation-based trust framework (see Section IV-C).

A. 5G distributed service marketplace

As previously stated, marketplaces have an important role in 5G networks when stakeholders need to extend their current resource and service capabilities in order to cover peak workloads through third party infrastructure providers. Due to the absence of distributed solutions covering on-demand resource and service provisioning, the 5GZORRO project delineates solutions for secure, trustworthy, automated, and intelligent resource discovery and selection, operating with SLAs to facilitate workload offloading to third party resources across multiple domains. In particular, Fig. 4 showcases an overview of how trust and reputation management is integrated with other primary layers such as *Analytics and Intelligence* and *Resource and Service Trading*. The 5GZORRO architecture design is mainly composed of four layers, being three of them involved in the process of guaranteeing a reliable ecosystem. Firstly, the blue color represents the *Analytics and Intelligence Layer* whose objective is to provide data persistence, data sharing, and data analytics for the 5GZORRO platform across domains. Secondly, the *Security and Trust Layer* aims to guarantee intra- and inter-domain security capabilities, trustworthy establishments across multiple domains as well as the identification and authorization of stakeholders. Lastly, the *Resource and Service Trading Layer* does business

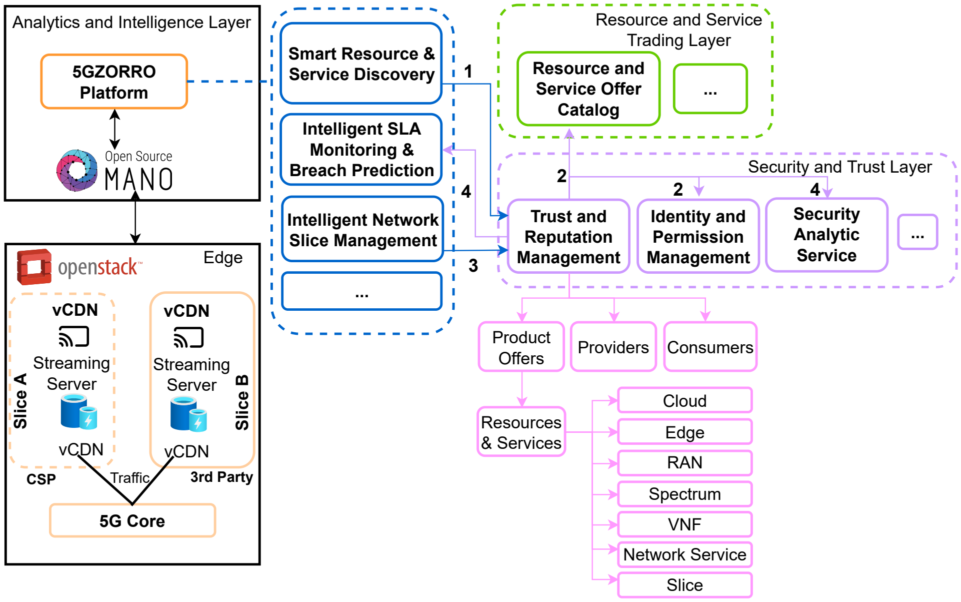
Analytics and Intelligence Layer
5GZORRO Platform
Open Source MANO
Edge
openstack
vCDN
Streaming Server
vCDN
Slice A
Slice B
CSP
Traffic
3rd Party
5G Core
Smart Resource & Service Discovery
Intelligent SLA Monitoring & Breach Prediction
Intelligent Network Slice Management
...
1
2
3
4
Resource and Service Trading Layer
Resource and Service Offer Catalog
Security and Trust Layer
Trust and Reputation Management
Identity and Permission Management
Security Analytic Service
Product Offers
Providers
Consumers
Resources & Services
Cloud
Edge
RAN
Spectrum
VNF
Network Service
Slice

Fig. 4: Trust and reputation integration into a 5G distributed service marketplace

with 5G resources across different domains by utilizing SCs and a dedicated Marketplace DLT platform.

As well, Fig. 4 also presents an edge scenario based on a virtual Content Delivery Network (vCDN) paradigm in which we can underline how trust is applied to optimize the orchestration part as well as ensure a trustworthy slice selection. In this vein, our use case (UC) describes how a stakeholder can elect a reliable provider to cover a load of a streaming service. Especially, the stakeholder has the need for slice expansion due to the impending overload of its vCDN server located on the Content Service Provider (CSP) Edge server. As a result, the stakeholder looks for compute resources, namely a slice instance, at the Edge to hire it and bypass the traffic routing through the network core.

When it comes to slice extension, our UC considers an advanced auto-scaling policy to trigger the resource discovery process that aims at identifying potential usable 3rd party edge resources. Such a discovery process identifies the candidate product offers (POs) and rates them based on how much they satisfy the offer request as well as on profile information related to the resource, e.g., trust properties, pricing, etc. To achieve this objective, the *Analytics and Intelligence Layer* introduces the Smart Resource & Service Discovery (SRSD) that allows obtaining a customized subset of resources and services that best satisfy the consumer expectations. Especially, one of the SRSD sub-steps is to determine a trust score for each available candidate to rank them.

In this sense, the SRSD sends a set of POs to be thoroughly analyzed by the TRMF (step 1). At this point, the TRMF begins a data-related gathering process through different information sources such as the Resource and Service Offer Catalog (step 2) which enables obtaining information about the geolocation of resources and services, current life-cycle status, service specification as well as deriving statistical features. Besides, the TRMF also makes use of Decentralized Identifiers (DIDs) to authenticate stakeholders in the 5GZORRO Marketplace and to identify offers registered in the Catalog (step 2). Afterward, the trust and reputation management framework begins its *Trust computation* module defined in Section III-B so as to find out a trust score per PO. Lastly, the TRMF sends a list of trust scores and POs back to the SRSD to classify candidates.

As the distributed 5GZORRO Marketplace is willing to facilitate stakeholders' interaction during the resource and service discovery stage, a graphical user interface (GUI) is also contemplated to showcase the ranking of trustworthy candidates for each type of offer and for a smoother user experience. Thence, once the stakeholder determined the compute resource to be consumed, he/she is able to visualize all available offers ranked by the highest trust score, together with other characteristics such as price, provider, location, etc. Upon selecting the offer with the highest score, the stakeholder orders it from the Marketplace and starts orchestration steps. In this final stage, the network section is expanded to the 3rd party infrastructure. As a result, a secure connection is carried out between the CSP Edge server and the new infrastructure site as well as the instantiation of the service components on the new resources.

Last but not least, the orchestration part is in charge of notifying the TRMF which offer was finally selected by means of the Intelligent and Automated Slice & Service Management (ISSM) (step 3). This action triggers multiple actions in the TRMF since it should monitor relevant metrics to continuously adjust the trust score of an ongoing relationship. Hence, as the last trust-related step of our UC, the TRMF makes use of two modules, one comes from the *Security and Trust Layer* and the other from the *Analytics and Intelligence Layer*. The former is called security analytic service (SAS), whose aim is to analyze the network traffic and notify potential threats. The latter is the Intelligent SLA Monitoring & Breach Prediction (ISBP) module which recaps breach predictions and detections and

SLA violations (step 4). As we previously described in Section III-D and in [23], the TRMF leverages such information to reassess an ongoing trust establishment in real time.

Note that some steps, which do not directly impact on trust and reputation, have been omitted to simplify the understanding of both this subsection and Fig. 4.

B. Findings from the TRMF fine tuning process

Prior to analyzing the effectiveness of our trust and reputation management framework (TRMF), it is necessary to determine the proper values for those parameters which may be adjusted by users before launching it. Such parameters usually allow us to shape up a solution based on the intrinsic characteristics of a final enforcement scenario.

Particularly, our SLA-driven reward and punishment mechanism presents two parameters to be investigated. First and foremost, we have the forgetting factor ($\xi \in [0, 1]$) symbolized in Eq. 5. By means of the forgetting factor, a user can configure our mechanism to set how many iterations would be needed to equate a historical SLA Violation Rate $SLAVRate^{(t)}(u, m)$ to a sudden SLA Violation Rate increase. In other words, ξ establishes how much time is required for a trust score to converge to a target value. Thereby, the convergence speed of final trust scores may be adapted taking into account the number of interactions that a use case tends to habitually manage.

Also aligned with ξ, the TRMF also introduces the parameter n depicted in Eq. 8. In this case, n enables determining the decrease or increase percentage of punishment and reward mechanisms, respectively. Therefore, n plays a pivotal role to find out how much a trust score can be reset when a sudden increase $Increment(u, m)$ in SLA violation is maintained over time until the $SLAVRate^{(t)}(u, m)$ matches the sudden increase.

In order to figure out the best configuration for ξ and n, Fig. 5 displays a set of charts with multiple parameter combinations for a punishment scenario. Note that we are outlining the behavior of our punishment mechanism since such a mechanism tends to be more important in trust models than the reward mechanism. For this fine tuning process, we have arbitrarily fixed some parameters for all charts, for example, we set a $SLAVRate^{(t)}(u, m) = 2.456$ and a $Increment(u, m) = 4$, which is the target to achieve. Besides, the initial trust score $T(v, u)$ was set to 0.749, which defines a reliable behavior, and the system has previously carried out 100 iterations. Once we have established the first set of parameters, the next step was to discover proper values for adjustable parameters ξ and n. In the case of the forgetting factor, we first leveraged intermediate values ranging from 0.2 to 0.8 so as to avoid extreme results. In fact, the values related to $\xi = 0.2$ have been narrowed down in the graphs as they made it difficult to read the graphs when cutting with the x-axis too far to the right (see the black point to know the cut value). For the n parameter, we analyzed the behavior of our framework utilizing values ranging from 1 to 10 as we identified that values higher than 10 will imply a negligible decrease in trust scores.

Looking at Fig. 5, we can observe the two main conclusions previously introduced. On the one hand, whether the value of n is getting bigger, the amount of punishment to be applied on trust scores will be limited in comparison with a n with a lower value, for instance, $n = 1$ vs. $n = 8$. On the other hand, if we leverage a forgetting factor closer to 0.2, we will need a higher number of interactions to equal a $SLAVRate^{(t)}(u, m)$ to a repeated $Increment(u, m)$ over the time. It is worth mentioning that these statements are fulfilled regardless of fixed parameters selected at the beginning of fine tuning process. Therefore, bearing in mind our use case in which we have a huge number of interactions across multiple domains, we think the best value for n is 3 and for ξ is a value between 0.4 and 0.8. This statement is also supported by the fact that the 5G distributed service marketplace handles thousands of transactions per hour, and in consequence, stakeholders should not undergo drastic variations for events not repeated over time. Nonetheless, SLA violations are a type of unusual occurrence so it is important to uncover a repeated increase above the average at an early stage. Because of that, $n = 3$ allows us to drastically penalize to stakeholders who consecutively had unexpected behaviors based on their historical SLA Violation Rate. In the case of ξ, we have enlarged values between 0.4 and 0.8 so as to discover a forgetting factor that enables us to point out a change of unusual behavior, but not in a short time window. In this vein, the top side of Fig. 6 showcases a specific example based on previous patterns presented in Fig. 5 where values from $\xi = 0.5$ (120 iterations) to $\xi = 0.65$ (93 iterations) can contribute the equilibrium between a quick reaction and a minimum figure of iterations that our 5G distributed service marketplace requires to sharply dwindle a trust score.

When it comes to the fine tuning process for a reward mechanism, the recovery pace should be slower than the punishment mechanism since trust and reputation models conventionally put the focus on fingering stakeholders' misbehaviors rather than good ones. For this reason, after suffering a punishment due to inappropriate actions, our SLA-driven trust and reputation management framework should need a higher number of interactions to return to the pre-penalty trust value (0.749). On the bottom side of Fig. 6, we can observe a feasible combination of $n = 9$ and $\xi = 0.03$ *or* 0.04 for our mechanism. These values will allow us to get back to the normal state but without having a faster recovery process than the sanctioning process.

C. Experiments

Once the UC has been contextualized, this subsection recaps a set of tests to adjust pivotal parameters of our SLA-driven reward and punishment mechanism, checks its suitable behavior as well as that of the TRMF in general, and resilience to multiple attacks.

- *Experiment 1 – Bootstrap time for different amounts of events*: After adjusting the utmost important configurable parameters, our next goal is to analyze the necessary time to process different amounts of SLA Violation and Breach Prediction events by the SLA-driven reward and punishment mechanism. To this end, a time window was set to 5 minutes, therefore our mechanism gathered all events published by 5GZORRO ISBP and SLA Monitoring modules in two different Kafka buses. Due to the fact that our mechanism may receive

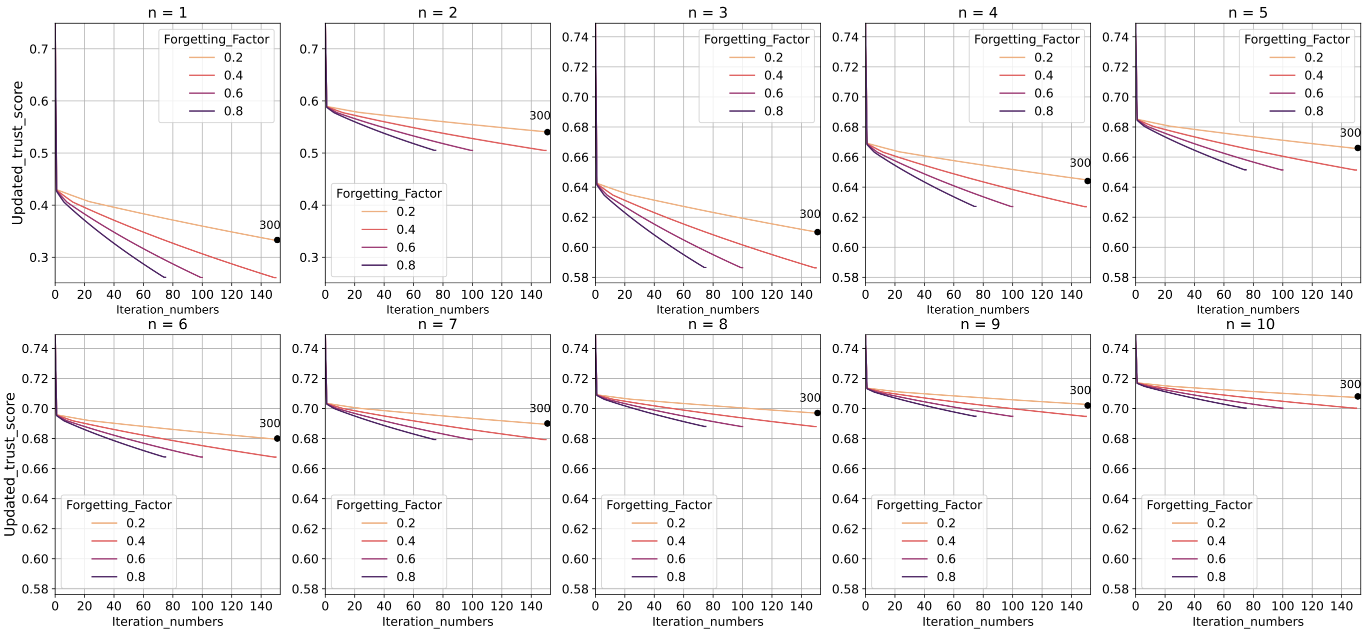

Fig. 5: Iteration number for equaling the $SLAVRate^{(t)}(u, m)$ to a sudden $Increment(u, m)$

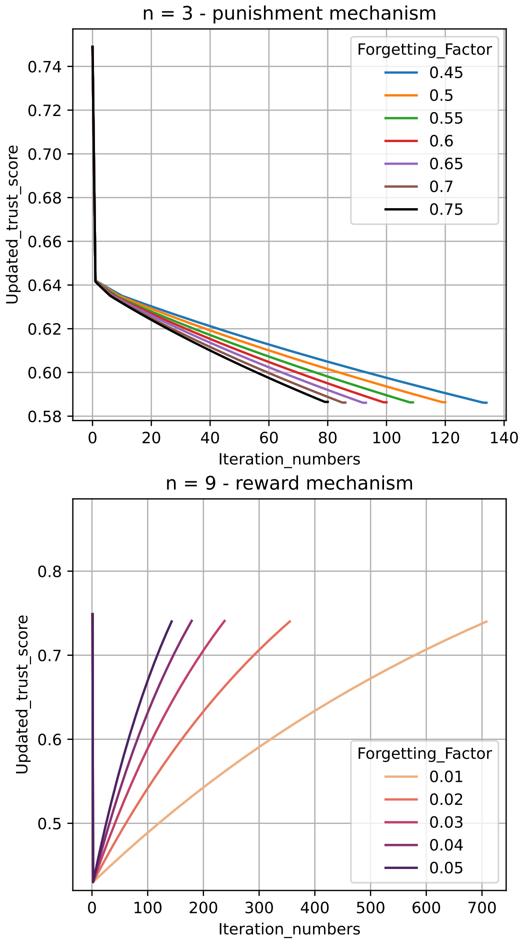

Fig. 6: Setting the best forgetting factor for reward and punishment mechanisms.

events related to violations, predictions, or both, we are going to study the three feasible combinations. Fig. 7 plots the time consumed by the SLA-driven reward and punishment mechanism when only SLA Violations are generated during the current time window (blue bar), only Breach Predictions (orange bar), or when there are 50% of events of each type (black line).

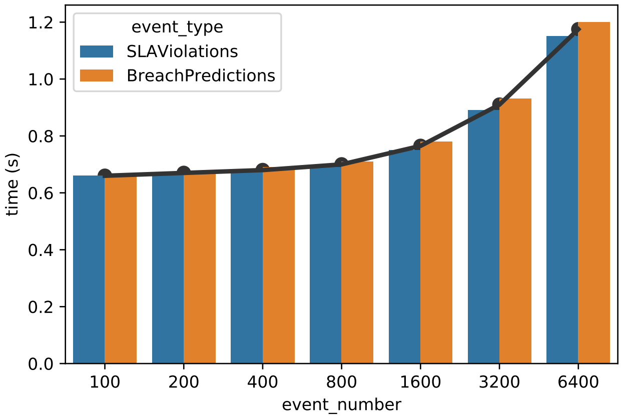

Fig. 7: Time consumption required to analyze SLA Violation and Breach Prediction events for a 5-minute time window

As it can be observed in Fig. 7, our mechanism did not introduce a high delay on the continuous update module as 1.2 seconds are only required to recompute a trust score in the worst-case scenario. Note that the event numbers defined in the x-axis were linked to a specific trust relationship thence, we do not expect to receive a number of events exceeding 3600 in our application scenario, or at least for such a small time window. In addition, we can only notice how time consumed is stable for the first three even numbers and is slightly increased for the four one. For the rest of combinations (1600, 3200, and

6400), the total time consumed is increased to a larger extent but being it still negligible. This behavior is mainly caused because the maximum number of records returned in a single Kafka call is 500, so the need to request multiple calls entail a slight increase in time. Concerning the time required to perform the mathematical operations, there is no significant increase therefore, the proposed mechanism enables its scalability to larger scenarios.

- *Experiment 2 – Continuous misbehavior bursts over time*: This experiment aims at verifying the proper behavior of our SLA-driven reward and punishment mechanism when multiple waves of malicious behaviors tamper a reliable flow, also known as an on-off attack. As we previously stated, SLA violations are a type of unusual occurrence so it is important to uncover a repeated increase above the average at an early stage. Following configuration parameters settled on previous experiments, we have leveraged a 5-minute time window to gather SLA Violations and Breach Prediction events. In addition, we have also set multiple behavior bursts to analyze how rewards and punishments affect trust scores. First, on the one hand, we establish a fixed parameter, *misbehavior*, to 2, 4, or 8 which does not change for each plot in Fig. 8. Such a parameter describes the figure of consecutive iterations that a stakeholder had unusual behaviors. On the other hand, we also depict three different bursts of good behaviors for each plot in Fig. 8. Thus, we try to visualize whether our mechanism quickly forgets misbehaviors.

There are two main conclusions that can be derived from analyzing such results. First, the reward mechanism does not allow stakeholders to overcome consecutive misbehaviors (2, 4, or 8) in a short period. Hence, our prior statement about negative events had a higher impact than positive ones on trust scores is being fulfilled. In the best case (2 consecutive misbehaviors), the model requires 76 iterations or 380 minutes with consecutive good behaviors to restore the initial trust score (0.729). It is worth mentioning that each iteration (x-axis) entails 5 minutes. Second, the punishment enables dwindling a high trust score without drastically setting it to 0, if it has been a one-time setback and the stakeholder has been able to recover. This statement can be observed in any of the plots in Fig. 8 after the first wave of misbehaviors. From the left plot to the right one, we can notice how a higher consecutive number of misbehaviors entails a higher punishment as well as a lower iteration number to get a 0 trust score. In the worst case (8 consecutive misbehaviors) in Fig. 8, 25 iterations or 125 minutes are necessary to reduce a stakeholder's reputation to 0. Thence, our SLA-driven reward and punishment mechanism meets the expected behavior because it allows us to identify the change of behavior of a malicious user in 1 hour, being his/her trust score dwindled to 0.

- *Experiment 3 – Bad-mouthing attack resilience*: One of the most customary recommendation-related attacks of reputation-based trust models is the bad-mouthing attack [32]. By means of it, an attacker intends to dwindle the trustworthiness of honest entities, or the reverse, through deceptive recommendations [27]. In particular, the bad-mouthing attack to be tested follows the collusive bad-mouthing paradigm [33], so malicious nodes are colluded and intend to give hostile feedback about a targeted node. To cope with it, our TRMF introduces a mechanism to deal with such attacks as part of the community factor (CF) dimension. Concretely, our mechanism considers two key factors to detect bad-mouthing attacks [32]: (i) the trust of recommender's feedback and (ii) the recommendation deviation [23]. Therefore, the main objective of our mechanism is to identify the collusion of malicious recommenders, among the total percentage of recommenders, so as to minimize the decrease in the trust score that our TRMF model would suffer if it considers recommendations from third parties.

In order to demonstrate the resilience of our TRMF to the bad-mouthing attack, Fig. 9 displays the likelihood of electing reliable recommenders when a percentage of the population has been corrupted and Fig. 10 plots the impact of such disrupted recommendations over final trust score. Concerning Fig. 9 and Fig. 10, we have evaluated our TRMF in different environments with up to 90% of malicious recommendations, although we considered that a percentage greater than 50% is, in straightforward terms, unrealistic since the disrupted feedback would become the majority and the method could understand malicious recommendations as good because they are the majority.

When it comes to Fig. 9, we can observe a detriment in the accuracy of the resilience mechanism when the amount of malicious recommenders increments. In the worst case, where 90% population is corrupted, our TRMF model is capable of fairly distinguishing misbehaviors in around 33% of the cases. Yet, as above-mentioned, we consider a percentage of malicious recommenders greater than 50% as unrealistic. Bearing such a percentage in mind, our TRMF is able to achieve an accuracy of 67%, reaching a maximum of 93% when only 30% of recommenders behave spitefully. It is worth mentioning that we always select the number of recommenders positioned lower for all statistics in this section. Regarding Fig. 10, we can visualize how a trust score dwindled an 8.6% when our TRMF reached a 33% accuracy to identify misbehaviors (90% malicious population and 100/150 recommenders). On the contrary, only a 2.1% trust score is decreased when only 30% of recommenders behave spitefully. As a result, the TRMF is capable of slightly mitigating the impact of misbehaviors on trust scores when the malicious percentage is lower than 40%, hence the trust scores dwindle from 3.1% to 8.6%.

V. Conclusions and Future Work

This paper presents a trust and reputation management framework to boost trustworthy stakeholder selection in a 5G distributed marketplace. In particular, we have proposed a reputation-based trust model composed of four modules: *Information gathering and sharing*, *Trust computation*, *Trust storage*, and *Continuous update*. By means of these modules, we describe the principal actions of our trust and reputation management framework life-cycle. Specially, for the *Continuous update module*, we propound an SLA-driven reward and punishment mechanism which allows adjusting the trust score of an ongoing relationship through SLA events, i.e., breach predictions and detections and SLA violations. By employing fuzzy models, our reward and punishment mechanism can

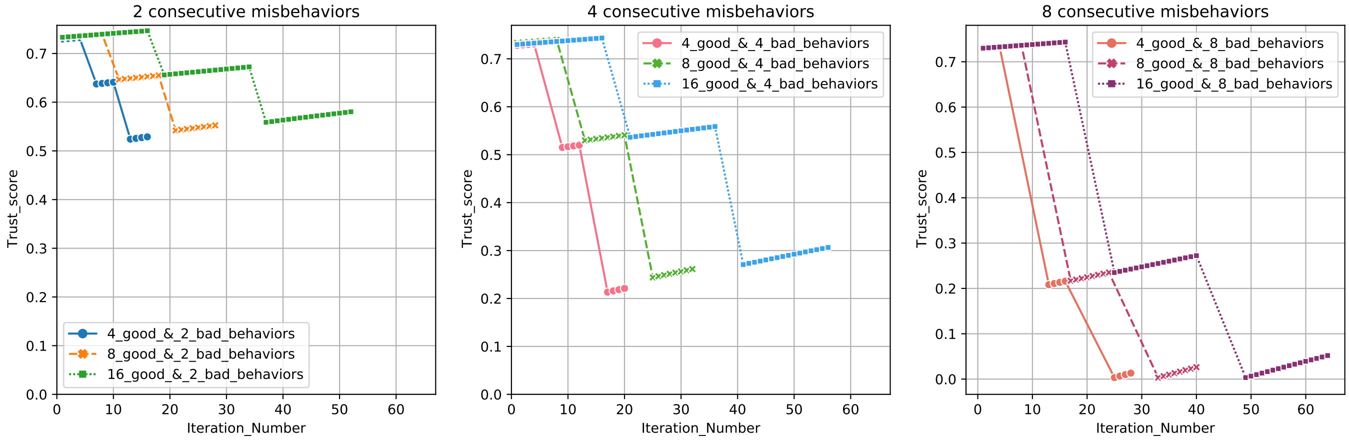

Fig. 8: Misbehavior bursts and their impact on trust scores

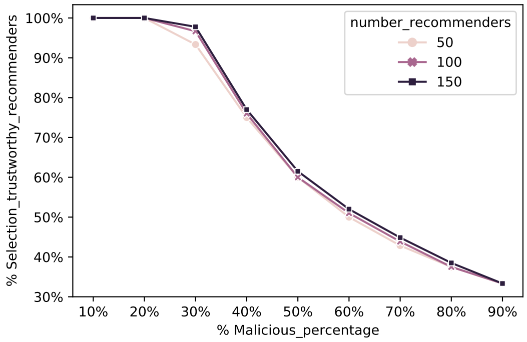

Fig. 9: Likelihood of electing reliable recommenders

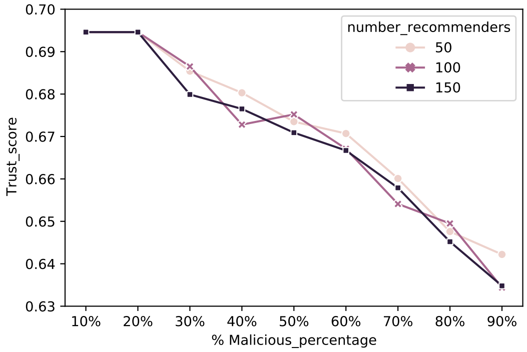

Fig. 10: Disrupted recommendation impact over a trust score

also determine the membership degree of trust and reputation values concerning the declared trust levels. Lastly, we have introduced a real use case covered by the 5GZORRO H2020 European project to verify the behavior and the accuracy of our SLA-driven trust and reputation management framework. Experimental results carried out in real infrastructures like 5GBarcelona and 5TONIC demonstrate that our proposal can deal with conventional trust attacks such as on-off and bad-mouthing. As well, the framework can ensure a reliable recommender selection with an accuracy of 94% and 67% when 30% and 50% population are corrupted.

As future work, we plan to enhance the resilience of our trust framework by considering other well-known trust attacks such as shilling, collusion, or ballot stuffing. Therefore, new resilient mechanisms will be designed and developed together with the current ones. Besides, the authors aim to extend the research field in order to support trustworthy on-demand service provisioning systems for cloud and edge computing. Such an effort entails finding out similarities and divergences of trust models for on-demand service and resource provisioning scenarios and, consequently, designing and developing two new trust models. Last but not least, the authors aim at moving their reputation-based trust framework from a pure statistical approach (PeerTrust model) to Artificial Intelligence (AI)-driven model, as AI-based solutions make it easier to compare the effectiveness of the solution with other algorithms or methodologies, without the need to redesign or define a new set of equations for each purely statistical algorithm to be leveraged.

Acknowledgments

This work has been supported by the European Commission through 5GZORRO project (grant no. 871533) part of the 5G PPP in Horizon 2020.

References

[1] V. S. Mai, R. J. La, T. Zhang, and A. Battou, "End-to-end quality-of-service assurance with autonomous systems: 5G/6G case study," in *2022 IEEE 19th Annual Consumer Communications & Networking Conference (CCNC)*. IEEE, 2022, pp. 644–651.

[2] M. F. M. Firdhous and R. Budiarto, "An enhanced quality of service-based trust model for cloud computing," in *7th International Conference on Mathematics and Computing*. Springer, 2022, pp. 525–538.

[3] G. Carrozzo, M. S. Siddiqui, A. Betzler, J. Bonnet, G. Martínez Pérez, A. Ramos, and T. Subramanya, "AI-driven zero-touch operations, security and trust in multi-operator 5G networks: a conceptual architecture," in *2020 European Conference on Networks and Communications (EuCNC)*. IEEE, 2020, pp. 254–258.

[4] C. Benzaïd, T. Taleb, and M. Z. Farooqi, "Trust in 5G and beyond networks," *IEEE Network*, vol. 35, no. 3, pp. 212–222, 2021.

[5] A. Fernández-Fernández, M. De Angelis, P. G. Giardina, J. Taylor, P. Chainho, J. M. Jorquera Valero, L. Ochoa-Aday, D. R. López, G. Carrozzo, and M. S. Siddiqui, "Multi-party collaboration in 5G networks via DLT-enabled marketplaces: A pragmatic approach," in *2021 Joint European Conference on Networks and Communications & 6G Summit (EuCNC/6G Summit)*. IEEE, 2021, pp. 550–555.

[6] T. Ranathunga, R. Marfievici, A. McGibney, and S. Rea, “A DLT-based trust framework for IoT ecosystems,” in *2020 International Conference on Cyber Security and Protection of Digital Services (Cyber Security)*. IEEE, 2020, pp. 1–8.

[7] M. Yuan, X. Li, X. Li, H. Tan, and J. Xu, “Trust hardware based secured privacy preserving computation system for three-dimensional data,” *Electronics*, vol. 10, no. 13, pp. 1–29, 2021.

[8] D. D. S. Braga, M. Niemann, B. Hellingrath, and F. B. D. L. Neto, “Survey on computational trust and reputation models,” *ACM Computing Surveys*, vol. 51, no. 5, pp. 1–40, 2019.

[9] N. Hashemipour, P. C. del Granado, and J. Aghaei, “Dynamic allocation of peer-to-peer clusters in virtual local electricity markets: A marketplace for EV flexibility,” *Energy*, vol. 236, pp. 1–14, 2021.

[10] A. Terra, R. Inam, P. Batista, and E. Fersman, “Using counterfactuals to proactively solve service level agreement violations in 5G networks,” in *2022 IEEE 20th International Conference on Industrial Informatics (INDIN)*. IEEE, 2022, pp. 552–559.

[11] M. J. Aslam, S. Din, J. J. Rodrigues, A. Ahmad, and G. S. Choi, “Defining service-oriented trust assessment for social internet of things,” *IEEE Access*, vol. 8, pp. 206 459–206 473, 2020.

[12] A. Badshah, A. Ghani, S. Shamshirband, G. Aceto, and A. Pescapè, “Performance-based service-level agreement in cloud computing to optimise penalties and revenue,” *IET Communications*, vol. 14, no. 7, pp. 1102–1112, 2020.

[13] S. W. Rose, O. Borchert, S. Mitchell, and S. Connelly, “Zero trust architecture,” NIST Special Publication 800-207, 2020.

[14] European Commission, “5GZORRO: Zero-touch security and trust for ubiquitous computing and connectivity in 5G networks,” https://doi.org/10.3030/871533, 2019-2022, [Online; accessed 07-May-2023].

[15] F. Li, G. White, and S. Clarke, “A trust model for SLA negotiation candidates selection in a dynamic IoT environment,” *IEEE Transactions on Services Computing*, vol. 15, no. 5, pp. 2565–2578, 2022.

[16] W. Ma, X. Wang, M. Hu, and Q. Zhou, “Machine learning empowered trust evaluation method for IoT devices,” *IEEE Access*, vol. 9, pp. 65 066–65 077, 2021.

[17] W. Li, J. Cao, S. Qian, and R. Buyya, “TSLAM: A trust-enabled self-learning agent model for service matching in the cloud market,” vol. 13, no. 4, pp. 1–41, 2019.

[18] R. Zhang, Q. Wang, J. Cui, and X. Wu, “A QoS&SLA-driven multifaceted trust model for cloud computing,” in *13th Chinese Conference on Trusted Computing and Information Security*. Springer, 2020, pp. 281–295.

[19] M. B. Monir Mansour, T. Abdelkader, M. H. AbdelAziz, and E.-S. M. EI-Horbaty, “A trust evaluation scheme of service providers in mobile edge computing,” *International Journal of Electrical & Computer Engineering*, vol. 12, no. 2, pp. 2121–2138, 2022.

[20] C. Muralidharan and R. Anitha, “Trusted cloud broker for estimating the reputation of cloud providers in federated cloud environment,” *Concurrency and Computation: Practice and Experience*, vol. 34, no. 1, p. e6537, 2022.

[21] L. Guo, H. Yang, K. Luan, L. sun, Y. Luo, and L. Sun, “A trust model based on characteristic factors and slas for cloud environments,” *IEEE Transactions on Network and Service Management*, Early Access, 2023.

[22] J. M. Jorquera Valero, “5G-enabled trust and reputation management framework (5G-TRMF),” https://github.com/5GZORRO/5G-TRMF, 2022, [Online; accessed 07-May-2023].

[23] J. Jorquera Valero, P. Sánchez Sánchez, M. Gil Pérez, A. Huertas Celdrán, and G. Martínez Pérez, “Trust-as-a-Service: A reputation-enabled trust framework for 5G networks,” arXiv 2210.11501, 2022.

[24] D. Breitgand, A. Lekidis, R. Behravesh, A. Weit, P. Giardina, V. Theodorou, C. E. Costa, and K. Barabash, “Dynamic slice scaling mechanisms for 5G multi-domain environments,” in *2021 IEEE 7th International Conference on Network Softwarization (NetSoft)*. IEEE, 2021, pp. 56–62.

[25] L. Xiong and L. Liu, “PeerTrust: Supporting reputation-based trust for peer-to-peer electronic communities,” *IEEE Transactions on Knowledge and Data Engineering*, vol. 16, no. 7, pp. 843–857, 2004.

[26] J. Jorquera Valero, M. Gil Pérez, and G. Martínez Pérez, “A security and trust framework for decentralized 5G marketplaces,” in *VII National Cybersecurity Research Conference (JNIC)*, 2022, pp. 237–240.

[27] J. M. Jorquera Valero, P. M. Sánchez Sánchez, A. Lekidis, J. Fernandez Hidalgo, M. Gil Pérez, M. S. Siddiqui, A. Huertas Celdrán, and

[28] A. Jøsang, J. Diaz, and M. Rifqi, “Cumulative and averaging fusion of beliefs,” *Information Fusion*, vol. 11, no. 2, pp. 192–200, 2010.

G. Martínez Pérez, “Design of a security and trust framework for 5G multi-domain scenarios,” *Journal of Network and Systems Management*, vol. 30, pp. 1–35, 2022.

[29] V. Theodorou, A. Lekidis, T. Bozios, K. Meth, A. Fernández-Fernández, J. Tavlor, P. Diogo, P. Martins, and R. Behravesh, “Blockchain-based zero touch service assurance in cross-domain network slicing,” in *2021 Joint European Conference on Networks and Communications & 6G Summit (EuCNC/6G Summit)*. IEEE, 2021, pp. 395–400.

[30] A. Fernandez-Fernandez, E. Coronado, A. Erspamer, G. Samaras, V. Theodorou, and S. Siddiqui, “Unlocking the path towards intelligent telecom marketplaces for beyond 5G and 6G networks,” *IEEE Communications Magazine*, vol. 61, no. 3, pp. 28–34, 2023.

[31] A. Barua, L. S. Mudunuri, and O. Kosheleva, “Why trapezoidal and triangular membership functions work so well: Towards a theoretical explanation,” *Journal of Uncertain Systems*, pp. 164–168, 2013.

[32] J. Wang, Z. Yan, H. Wang, T. Li, and W. Pedrycz, “A survey on trust models in heterogeneous networks,” *IEEE Communications Surveys & Tutorials*, vol. 24, no. 4, pp. 2127–2162, 2022.

[33] N. K. Saini, A. Chaturvedi, and R. Yadav, “Identifying collusion attacks in P2P trust and reputation systems,” *International Journal of Computer Applications*, vol. 2, pp. 36–41, 2014.

José María Jorquera Valero is a Ph.D. student in Computer Science at the University of Murcia. Jorquera Valero received the M.Sc. degree in Computer Science from the University of Murcia, Spain. His scientific interests include trust, security, 5G, data privacy, continuous authentication, and cybersecurity.

Vasileios Theodorou is an R&D engineer at Intracom S.A. Telecom Solutions, working on NFV and edge computing. He was a research associate at the Polytechnic University of Catalonia and a research assistant at York University of Toronto, Canada.

Manuel Gil Pérez is Associate Professor in the Department of Information and Communication Engineering of the 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.

Gregorio Martínez Pérez is Full Professor in the Department of Information and Communications Engineering of the University of Murcia, Spain. His scientific activity is mainly devoted to cybersecurity and networking, where he has published 160+ papers.