Array 19 (2023) 100304

Contents lists available at ScienceDirect

Array

journal homepage: www.elsevier.com/locate/array

# A data security and privacy scheme for user quality of experience in a Mobile Edge Computing-based network

Miguel Landry Foko Sindjoung [a,*], Mthulisi Velempini [a], Clémentin Tayou Djamegni [b]

[a] *Department of Computer Science, University of Limpopo, Mankweng, South Africa*
[b] *Fotso Victor University Institute of Technology, University of Dschang, Cameroon*

ARTICLE INFO

*Keywords:*
Autonomous vehicular network
Data security and privacy
Mobile edge computing
User authentication
User quality of experience
Quality of service

ABSTRACT

Cloud computing has widely been used for applications that require huge computational and data storage resources. Unfortunately, with the advent of new technologies such as fifth generation of cellular networks that provide new applications like IoT, cloud computing presents many limits among which the End-To-End (E2E) latency is the main challenge. These applications generally degrade scenarios that require low latency. Mobile Edge Computing (MEC) has been proposed to solve this issue. MEC brings computing and storage resources from cloud data center to edge data center, closer to end-user equipment to reduce the E2E latency for request processing. However, MEC is vulnerable to security, data privacy, and authentication that affect the end-user Quality of Experience (QoE). It is therefore fundamental that these challenges are addressed to avoid poor user experience due to the lack of security or data privacy. In this paper, we propose a hybrid cryptographic system that uses the symmetric and asymmetric cryptographic systems, to improve data security, privacy, and user authentication in a MEC-based network. We show that our proposed scheme is secured by validating it with the Automated Validation of Internet Security Protocol and Application tool. Simulation results show that our solution consumes less computing resources.

## 1. Introduction

Mobile Edge Computing (MEC) is a technology that brings computing and data storage to edge servers in order to reduce the workload of network devices in applications such as 5G and 6G networks, Internet of Things (IoT), augmented reality, and big data [1–3]. The increase in end user devices and generated data which is sent to cloud datacenters for processing and storage make it difficult to achieve the goal of deployed networks in different scenarios, due to the End-to-End (E2E) latency that increases when the number of user devices increases [4]. Consequently, the quality of service (QoS) and user quality of experience (QoE) may be affected. Mobile Edge Computing (MEC) technology has emerged in recent years as an alternative to cloud computing. MEC brings computing and storage resource management from the cloud datacenters to the edge datacenters in order to improve the quality of service [5] by reducing the E2E latency of end user requests. Unfortunately, that QoS improvement brings new issues that affect the user QoE. While QoS can be measured by network metrics such as end-to-end delay (E2ED), computing, storage, and bandwidth resources [6], QoE is measured by the user experience and satisfaction [7,8].

MEC infrastructures worsen the security and privacy issues which are experienced in the context of cloud computing [9], which impacts user experience and satisfaction. MEC infrastructure should ensure QoS, security, and data privacy in order to guarantee both the QoS and the user QoE [4]. As stated by Filali et al. [6], MEC solutions should combine software-defined networking (SDN), network function virtualization (NFV), service function chaining (SFC), and network slicing (NS) technologies in order to optimize the QoS. While the combination of SDN and NFV may facilitate the data and control plane separation to improve the data processing within the network, SFC uses a multiple NFV to improve efficiency. The NS is used to manage network slices for real time data forwarding [10]. Many solutions have previously been proposed in the literature for the QoS [6,10–13] and user QoE [5,9,14]. These solutions have limitations that we highlighted in Section 2. Based on these challenges, we propose a scheme designed to secure communication and ensure data privacy for a better QoS and user QoE in the MEC infrastructure. The proposed scheme is a hybrid one that combines two existing secure schemes to improve the security of communications, data security and privacy in a MEC-based network an d that improves QoS and user QoE. By ensuring the data security and

[*] Corresponding author.
*E-mail addresses:* miguel.fokosindjoung@ul.ac.za (M.L. Foko Sindjoung), mthulisi.velempini@ul.ac.za (M. Velempini), dtayou@gmail.com (C. Tayou Djamegni).

https://doi.org/10.1016/j.array.2023.100304
Received 11 February 2023; Received in revised form 23 June 2023; Accepted 24 June 2023
Available online 28 June 2023

privacy within the network, information is secured. There is also no possibility for users to complain about privacy for communications in the network, which results in a great user experience when users are using the network services. This is how the user QoE is ensured in this work. We seek to improve the security of MEC architecture we proposed in [10]. We also provide the proof of concept of our proposed security scheme.

The rest of the paper is organized as follows: In Section 2, we present a literature review on security and data privacy management for MEC infrastructures. After that, we present our proposed solution for data security and privacy in a MEC architecture in Section 3. Section 4 presents the simulation results and finally, in Section 5, we conclude the paper.

## 2. Literature review on security and data privacy in MEC infrastructures

As highlighted by Tasnim et al. [8], the user quality of experience for a network service refers to the experience and satisfaction one experiences when using a given service. A user may not be satisfied if data being sent or received is corrupted. This results in poor decision-making. Furthermore, in regard to applications such as IoT, augmented reality, and AVNET that are mobile ad hoc networks (MANET) [15,16], it is crucial for a user to know the identity of other users since these networks do not require preexisting infrastructures for their deployment. Security threats are more prevalent and they impact negatively on the user quality of experience. Data security and privacy are challenges that need to be solved for user QoE in MEC networks. Indeed, in these networks, end user equipment have the ability to collect users sensitive information such as their identity or their location [4]. It therefore become crucial to protect these information against hackers. In this paper, we focus on data security and privacy in MEC networks optimized for provisioning of QoS and user QoE.

There are many solutions in the literature which were designed for data security and privacy in MEC environments. Kaur et al. [9] proposed a lightweight and efficient mutual authentication protocol for MEC environments that use Elliptic Curve Cryptography (ECC), concatenation operations, and one-way hash functions. The solutions were motivated by higher communication and computational overheads in existing security and data privacy solutions for MEC environments. The security protocol operates in four phases that are set up, user registration, server registration, and authentication. The authors present a detailed security analysis and an overview of security threats. They also present a comparative study that shows that their solution is a better MEC security solution than the one for Jia et al. [17], Tsai et al. [18] and Irshad et al. [19]. Unfortunately, we observed some drawbacks to the proposed security protocol. Firstly, Kaur et al. assume that the communication channel between the registration center (RC) and mobile user, also the communication channel between MEC servers (MS) and the RC are secured, which is not the case since the wireless communication channels used in MEC environments are generally open in nature [20]. Secondly, in the user registration phase, users communicate directly with the RC which incurs more delays and latency.

Mohammad et al. [5] proposed a secure authenticated key agreement protocol for MEC that tolerates security weaknesses of the protocol presented by Jia et al. [17] and also incurs low computational and communication costs in comparison to schemes presented in [17,21–25]. However, the scheme by Jia et al.'s does not define a key registration and revocation mechanism, which makes their scheme less practical. The authors also claim that their protocol is secure against impersonation attacks and that it provides mutual authentication, but Mohammad et al. show that it is not the case. Unfortunately, in Mohammad et al.'s scheme [5], it is the user (U) that requests for its revocation, meaning that an attacker may attack a user and request for its un-subscription near the MS or the RC. Finally, both the scheme by Jia et al. [17] and the one of Mohammad et al. consider the existence of a secure channel for communications between U, MS, and RC which may not be possible.

Hou et al. [26] proposed a Fine-Grained Access Control mechanism (FGAC) that manages user access control for data security in MEC. They noted that access control methods used in MEC infrastructures are essentially role-based access control (RBAC) and attribute-based access control (ABAC) [27]. Their study was motivated by some disadvantages that exist in the previous access control mechanisms. Namely, these solutions are coarse-grain, they are not flexible and accurate and they do not consider internal attacks. Hou et al. [26] then presented FGAC, a solution that can be used in the MEC environment since it is fined-grained, more flexible, and manages the internal attacks that mobile equipment can be subject to. They combined RBAC with meta graph theory based on user grouping strategy and user attributes in order to achieve a fine-grained access control mechanism. Unfortunately, their solution does not consider the case where a user may move within the network, which is a drawback in MEC.

Li et al. [23] showed that security concerns in a MEC environment can be summarized in two categories: data security issues and application security issues. While the first deals with data integrity, confidentiality, and privacy protection, the latter deals with risks that can exist in communication links such as identity authentication and confidentiality protection during communications. The proposed solution considers both categories and takes into account efficient communication between mobile devices and MEC servers as well as the heterogeneity of MEC architecture equipment. The main drawback observed in this scheme is the consideration of an existing secured channel for the registration of mobile users and the MEC server.

## 3. The proposed secured and data privacy scheme for MEC architectures

In this section, we present our solution for data security, data privacy, and user authentication for a MEC infrastructure. This solution aims to solve the gaps we observed in the existing solutions presented in Section 2. We observed that the use of an existing secure channel for user equipment (UE) authentication may not be possible. Moreover, UE mobility that requires a Single Sign-On (SSO) scenario for each of the UEs and MSs is not considered. Finally, these solutions consider that each UE should directly communicate with the RC when joining the network, which may degrade the QoS and increase the latency between the UE and the MS.

We present the MEC architecture on which we apply our proposed secure scheme in Section 3.1. Then, we give the overview of the proposed scheme and the considered prerequisites in Sections 3.2 and 3.3 respectively. The proposed scheme is presented in Section 3.4 and its analysis in Section 3.5.

### *3.1. An SDN-NFS-SFC-NS-based MEC architecture for a better QoS in an Autonomous Vehicular Network (AVNET)*

In this section, we present a MEC architecture based on SDN, NFS, SFC, and NS in an AVNET environment. That architecture is more detailed in our work presented in [10]. The said architecture is presented in Fig. 1 where all the objects denote Autonomous Vehicles.

The core concept of the architecture resides in the MEC server internal architecture presented in Fig. 2. However, except the Network Slicing component that is managed by the cloud server, the SDN, NFV, and SFC are globally managed by the MEC server

In the MEC server-internal architecture, Autonomous Vehicles (AV) send their request through inflow (1). The request sent by the AV is received by the SDN controller (in VM1) which directly forwards it to the VNF Checker using flow (2) (in VM2). The role of the VNF checker is to check if the actual MEC server has enough resources to compute the task, then, dependent on the result it sends the feedback back to the

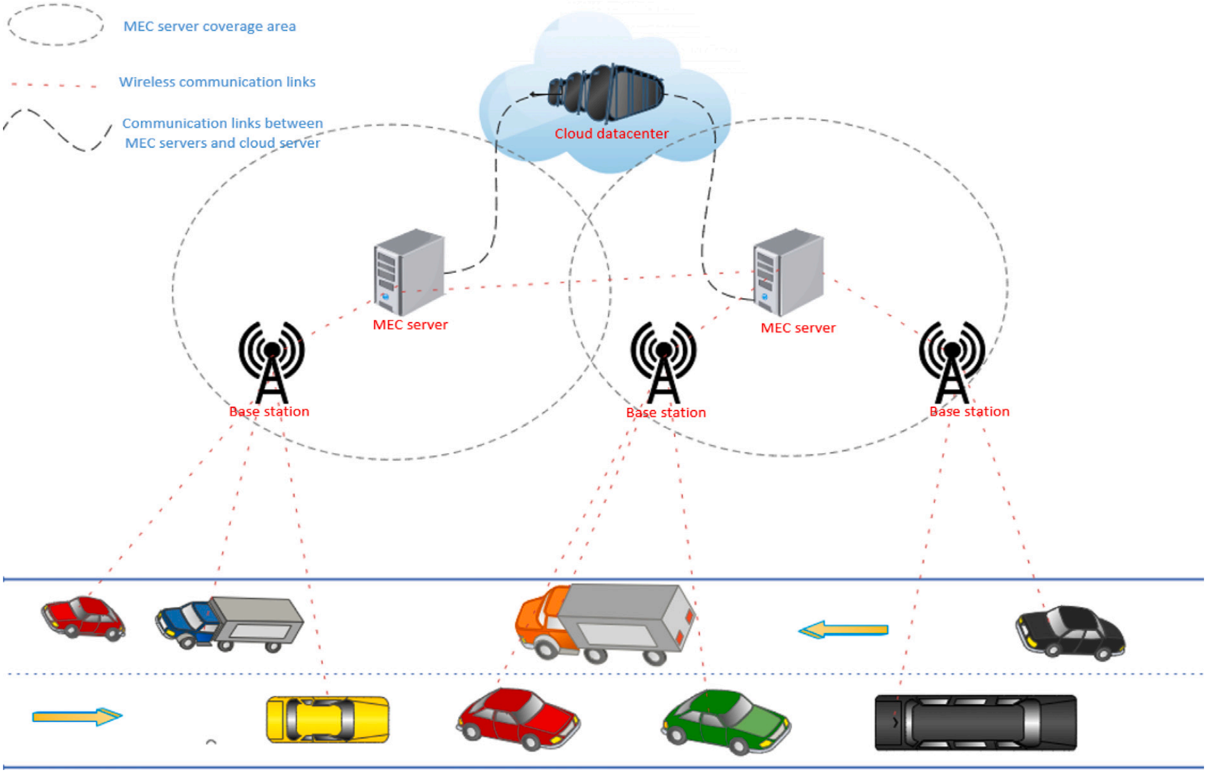


**Fig. 1.** A MEC-based AVNET architecture [10].

SDN Controller using flow (3), the latter decides whether the received request will be computed in the local MEC server by the VNF Processor using flow (4), or if that task needs to be computed elsewhere using flow (7). If the task is computed locally (in VM3), the result is sent back to the SDN controller using flow (5). In this case, the VNF Checker decides that the task should be computed elsewhere.

Regarding the AV movement and its velocity, a given task may be sent by the VNF Sender in VM4 to the cloud server using flow (11) or to the next MEC server in the direction of the requesting AV using flow (9). Results computed by the cloud server are sent back to the SDN controller using flows (12) and (8). In some cases, the MEC server may be the one that is requesting AV and then receives a request for that AV from a previous MEC server through flow (10). In that case, The VNF Receiver sends the received request to the SDN controller using flow (8), and the process restarts at flow (2). Finally, when SDN Controller receives the processing results, it forwards them to the requesting AV using flow (6).

The Service Function Chain graph (SFG) of the architecture is shown in Fig. 3. Based on the specifications described by Medhat et al. [28], the SFC graph should have two components: the SFC data plane and the SFC control plane. The SDN controller of the MEC Server internal architecture also acts as the SFC Controller. There exist two Service Function Forwarders in that SFC graph, SFF1, and SFF2 that are respectively located in VM4 and VM1. SFF1 includes two internal services that are VNF Receiver and VNF Sender.

As mentioned before, the Network Slicing is done by the cloud server, specifically by the cloud SDN. At this level, it is important to note that two slices are created for the network functionalities: the low-latency slices that helps to forward request requiring low latency such as AV request, and the high data rate slice that aims to transport a high volume of data that does not necessarily need low latency.

As can be seen, there is no authentication or security process in the architecture in Fig. 1. In Section 3.2, we present our secure and data privacy scheme for the MEC architecture.

### 3.2. Overview of the proposed hybrid cryptographic system

The hybrid cryptographic system as described in our scheme consists of the use of symmetric and asymmetric cryptography systems. We opted for this approach due to its simplicity and that it consumes fewer resources and the systems are used as and when needed to ensure an efficient use of UE resources. That is, the asymmetric cryptographic system is a robust system in terms of key protection, unfortunately, it consumes more resources because it has to compute more tasks [15,29], this may be a challenge given that UEs and AVNETs have limited resources. Due to this challenge, we use the asymmetric cryptographic system only to register the AVNET equipment (MEC servers and AVs) at a registration center. After registration, communication inside the network is secured using the symmetric cryptographic system. The symmetric cryptographic system does not require a lot of computation resources, but if it is used alone (without the asymmetric system) in a network such as AVNET, it may require a lot of storage from AVs to store private session keys for each network equipment. In our scheme, the symmetric cryptographic system is used for UE authentication after the session keys are generated by the asymmetric cryptographic system in the registration phase. Our solution ensures data privacy, security, and user authentication for all the communications inside the AVNET.

### 3.3. Prerequisites

The following conditions are to be considered in this solution:

- The proposed secure solution is executed in the MEC architecture presented in Section 3.1.
- Three network equipment (actors) are identified in the system: Automated Vehicles denoted AV, MEC servers denoted MS, and Registration Center denoted RC. In Fig. 2, the registration center is considered to be the Cloud Server. Then, MS and RC have the same internal architecture. The AV is the initiator of the message using flow (1) (Fig. 2).

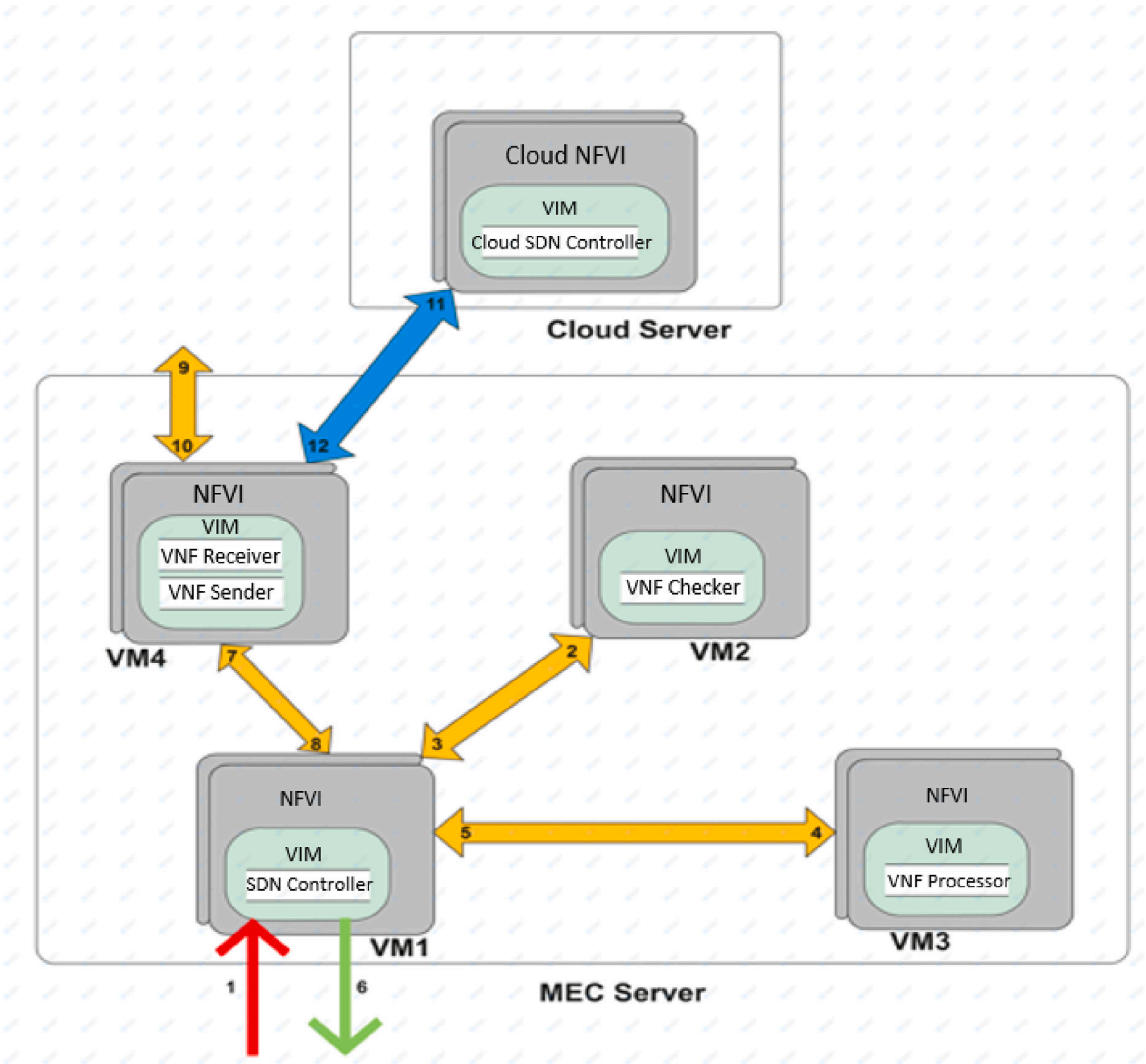


**Fig. 2.** MEC server internal architecture [10].

- Contrary to the security solution presented in [23] that takes place in three phases, our cryptographic scheme consists of four phases: initialization, registration, authentication, and revocation phases.
- Contrary to the assumptions made by [5] and Jia et al. [17], we consider that there is no existing secure communication channel.
- The asymmetric system is based on Elliptic Curve Cryptography (ECC) where G is the public generator point of the elliptic curve [15,29].
- $K_{DH}(X_1, X_2)$ denotes the Diffie–Hellman key exchange without interaction between $X_1$ and $X_2$.
- $pk(X)$ and $sk(X)$ denote the public and the private key of $X$ respectively.
- $SEK(N, M)$ is the session key between $N$ and $M$.

### 3.4. An hybrid cryptographic system for MEC architectures

The objective of the hybrid cryptographic system is to take advantage of the simplicity of the two existing cryptographic systems in a MEC architecture. Our hybrid system compatible with all kinds of end user equipment regardless of their characteristics [30]. The cryptographic system for data security, privacy, and user authentication for MEC architectures that we propose takes place in fourth stages: initialization, registration, authentication, and revocation.

#### 3.4.1. Phase 1: Initialization

The initialization phase loads keys that will be used for the registration phase in the network equipment. The loaded keys are used for the asymmetric cryptographic system. The public keys of MS and AV are loaded in the RC while the RC's public key is loaded in each of the MS and AV just before they are deployed in the network. Then, according to the ECC, $pk(X) = sk(X) \times G$, each of the network equipment $N$ (MS and AV) is able to compute the Diffie–Hellman key exchange between itself and the registration center $RC$ using Eq. (1)[29]. The RC can also compute the key exchanged between itself and the other network's equipment using Eq. (2).

$$K_{DH}(N, RC) = sk(N) \times pk(RC) \tag{1}$$

$$K_{DH}(RC, N) = sk(RC) \times pk(N) \tag{2}$$

It is proven that $K_{DH}(N, RC) = K_{DH}(RC, N)$ in Eq. (3). Then we conclude that the key exchanged by the RC and the MS or AV is the same. The key is used for the registration phase.

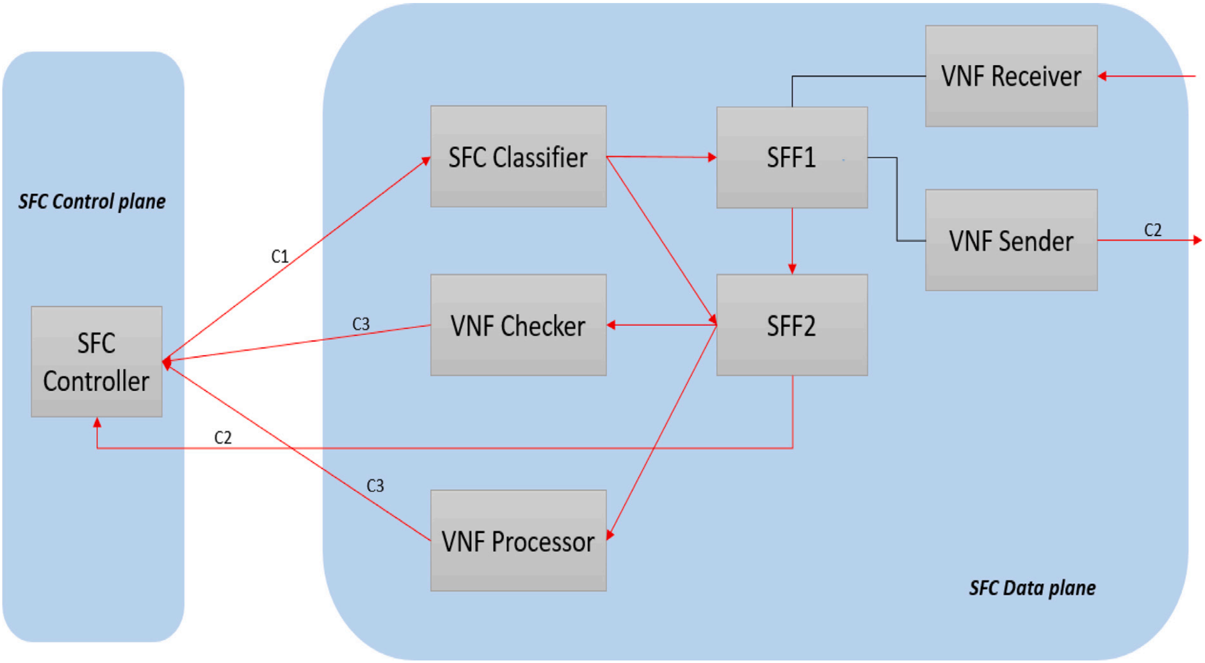


**Fig. 3.** The SFC graph of the used architecture.

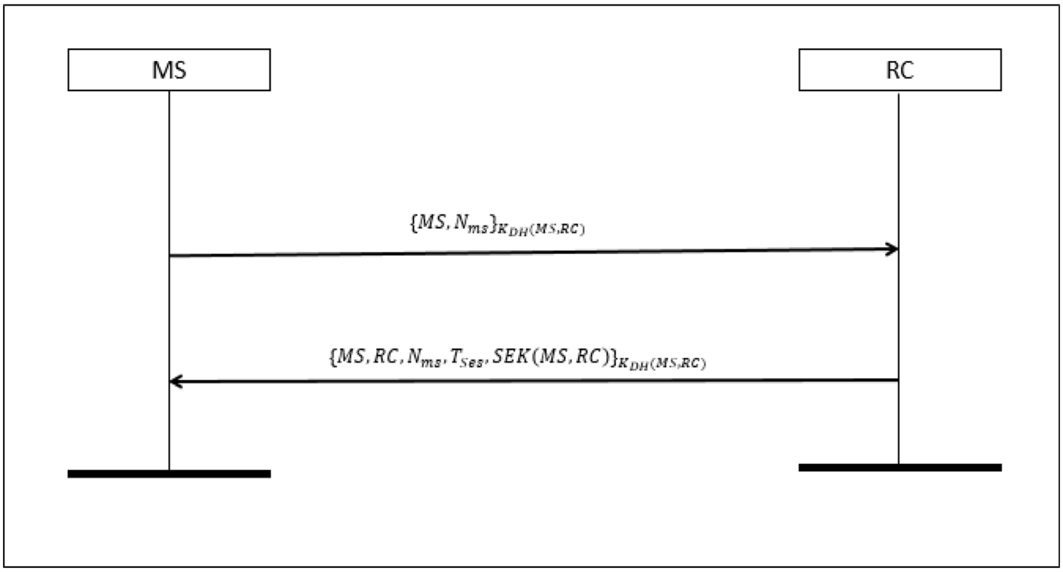

MS
RC
$\{MS, N_{ms}\}_{K_{DH}(MS,RC)}$
$\{MS, RC, N_{ms}, T_{Ses}, SEK(MS,RC)\}_{K_{DH}(MS,RC)}$

**Fig. 4.** MEC server registration process.

$$\begin{aligned} &K_{DH}(RC,N) = sk(RC) \times pk(N) \\ &= sk(RC) \times (sk(N) \times G) = (sk(RC) \times G) \times sk(N) \\ &= pk(RC) \times sk(N) = sk(N) \times pk(RC) \\ &= K_{DH}(N,RC) \end{aligned} \tag{3}$$

*3.4.2. Phase 2: Registration*

The aim of the registration phase is to use the loaded public keys of phase 1 to allow AV and MS to communicate securely in the networks. The registration takes place in two steps: firstly, the MEC servers get registered using the process presented in Fig. 4, and secondly, the AV gets registered using the process in Fig. 5.

Concerning the MS registration, once deployed, each MS sends a request $\{MS, N_{ms}\}$ to RC requesting for registration, where $N_{ms}$ is a nonce generated by MS for authentication purposes. The message is protected by the Diffie–Hellman key exchange without interaction between MS and RC $K_{DH}(MS,RC)$. At the reception of the previous message, RC uses $K_{DH}(RC,MS)$ to decrypt the message, then generates a session key $SEK(MS,RC)$ that will be used for all the future communications between itself and the given MS. It produces the message $\{MS, RC, N_{ms}, T_{Ses}, SEK(MS,RC)\}$, and protects it with $K_{DH}(RC,MS)$ and sends it back to MS. Upon reception, MS decrypts the previous message using $K_{DH}(MS,RC)$, then checks if the received $N_{ms}$ is equal to the one it initially generated and therefore saves the received session key. From this point and until $T_{Ses}$ (The validity delay) is reached, exchanged messages between MS and RC are encrypted by $SEK(MS,RC)$. After the registration of all the MS, the RC generates a list of session keys that each MS may use to communicate with other MS and share the keys with each of the MS.

The AV registration is done using an already registered MS using the process in Fig. 5. Initially, an AV that wishes to join the AVNET sends the message $\{AV, N_{AV}\}$ encrypted by the Diffie–Hellman

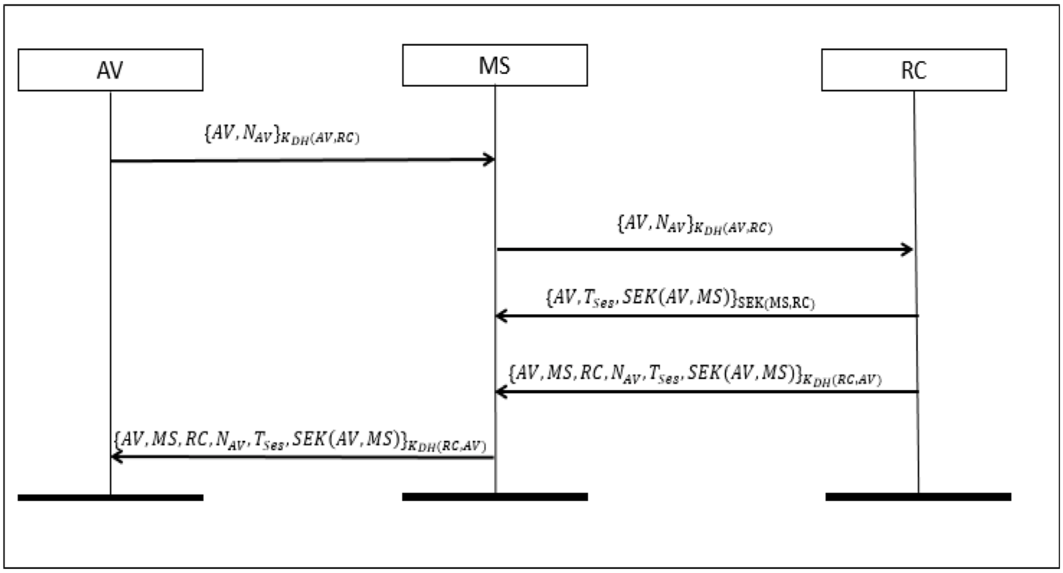

AV
MS
RC
$\{AV, N_{AV}\}_{K_{DH}(AV,RC)}$
$\{AV, N_{AV}\}_{K_{DH}(AV,RC)}$
$\{AV, T_{Ses}, SEK(AV, MS)\}_{SEK(MS,RC)}$
$\{AV, MS, RC, N_{AV}, T_{Ses}, SEK(AV, MS)\}_{K_{DH}(RC,AV)}$
$\{AV, MS, RC, N_{AV}, T_{Ses}, SEK(AV, MS)\}_{KDH(RC,AV)}$

**Fig. 5.** Autonomous vehicle registration process.

key exchange without interaction between itself and $K_{DH}(AV, RC)$ to the MS that covers the area within which it is located ($N_{AV}$ is a nonce generated by AV for authentication purpose). After receiving the message, MS forwards it to the RC since it is not able to decrypt the message. At the reception, RC decrypts the message with $K_{DH}(RC, AV)$, generates the session key $SEK(AV, MS)$ that will be used later for communication between MS and AV, then, shares the session key with MS and AV. When sharing $SEK(AV, MS)$, RC sends the message $\{AV, T_{Ses}, SEK(AV, MS)\}$ encrypted by the session key $SEK(MS, RC)$ it already has with MS (It does not use $K_{DH}(RC, MS)$ because it wants to avoid duplicate processing delays). The message contains AV to allow MS to know that the session key is the one it will share with AV. On the other hand, RC encrypts the message $\{AV, MS, RC, N_{AV}, T_{Ses}, SEK(AV, MS)\}$ with the key $K_{DH}(RC, AV)$ and sends it to AV via MS (Which just forwards the message without decrypting it). Upon receiving the message, AV first decrypts it with $K_{DH}(AV, RC)$, then checks if the received $N_{AV}$ is the same nonce it generated initially, after that, it saves $SEK(AV, MS)$ as the session key it will use for further communications with MS until $T_{Ses}$ is reached.

In Fig. 2, registration requests that arrive at a MEC server from AV are intercepted by the SDN controller(1) and directly forwarded to the cloud server (acting as registration center) through the VNF sender using flows 7 and 11. Inside the cloud server, the cloud SDN controller is responsible for decrypting the received messages, generating session keys for MS and AV communication, encrypting the messages for MS and AV registration requests, and sending back the responses (12). When it receives the registration response from the cloud server (12), the VNF Receiver forwards it to the SDN Controller (8) and depending on the message, forwards it to the requesting AV (if the message is $\{AV, MS, RC, N_{AV}, T_{Ses}, SEK(AV, MS)\}_{K_{DH}(RC,AV)}$) or saves the session key (if the message is $\{MS, RC, N_{ms}, T_{Ses}, SEK(MS, RC)\}_{K_{DH}(MS,RC)}$ or $\{AV, T_{Ses}, SEK(AV, MS)\}_{SEK(MS,RC)}$).

Furthermore, to ensure the AV's Single Sign On (SSO) property, when the Cloud Controller (RC) generates the session key $SEK(AV, MS)$ for communication between AV and MS, it shares the key with all the other MS in the AVNET at the same time that it sends the message $\{AV, T_{Ses}, SEK(AV, MS)\}_{SEK(MS,RC)}$ to MS, meaning that this session key will be known by all the MS. This ensures that when an AV moves from the coverage area of an MS to the coverage area of another one, there is no need to relaunch the registration process. It is registered only once.

### 3.4.3. Phase 3: Authentication

This phase is executed by network equipment when they need to communicate with each other. We distinguish four scenarios: MS-RC communication, MS-AV communication, AV-AV communication, and MS-MS communication. The MS-RC, MS-AV, and MS-MS communications are authenticated with the session keys that have been produced in Section 3.4.2. When a message encrypted by a session key arrives in a network entity (AV, MS, or RC) it uses the corresponding session key for decryption before any processing.

For the AV-AV communication, assume AV1 needs to send a $Message$ to AV2. For the first time that AV1 needs to communicate with AV2, the scheme presented in Fig. 6 is executed. AV1 sends $SEK(AV2, N_{AV})$ encrypted with $SEK(AV1, MS)$ to MS requesting a session key establishment with AV2. When it receives and authenticates the message, MS generates $SEK(AV1, AV2)$ and shares it with AV1 and AV2 respectively with $\{N_{AV}, T_{Ses}, SEK(AV1, AV2)\}_{SEK(MS,AV1)}$ and $\{AV1, AV2, T_{Ses}, SEK(AV1, AV2)\}_{SEK(MS,AV2)}$. At this point, both AV1 and AV2 are aware of their session key $SEK(AV1, AV2)$ with its corresponding validity delay, $T_{Ses}$, then AV1 sends its message to AV2 encrypted by the session key ($Message_{S}EK(AV1, AV2)$). For further direct communication between AV1 and AV2, they use $SEK(AV1, AV2)$ until $T_{Ses}$ are reached.

### 3.4.4. Phase 4: Revocation

To avoid attacks such as node tampering, node compromise, selective forwarding, sinkhole, and sibyl attacks [15], it is important to revoke the established session keys between network equipment during the registration phase. The revocation phase consists of revoking session keys, meaning that network equipment will need to re-execute the registration phase in the RC in order to be authorized to communicate in the AVNET. The revocation may happen in two scenarios. Firstly, it can be done automatically by AV and MS. When the RC generates session keys for AV and MS communication, it accompanies these session keys with validity delays. After the given delays, the session key is automatically deactivated, and the network equipment sends another registration request if it still needs to communicate within the AVNET and the registration process restarts from phase 1. Secondly, a session key revocation may be launched by any network entity after it receives a message from a supposed authenticated network equipment. For instance, assuming an MS receives a message from an AV encrypted by a valid session key, and that the given session key is not intended for the forwarding AV. In this case, MS regards it as a previous AV which

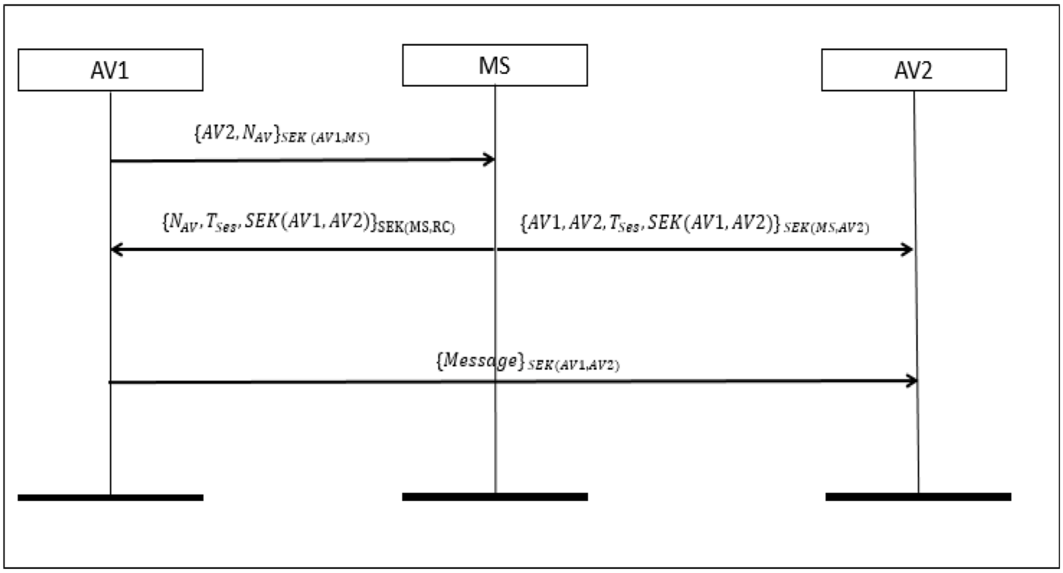

AV1
MS
AV2
$\{AV2, N_{AV}\}_{SEK\,(AV1,MS)}$
$\{N_{AV}, T_{Ses}, SEK(AV1,AV2)\}_{SEK(MS,RC)}$
$\{AV1, AV2, T_{Ses}, SEK(AV1,AV2)\}_{SEK(MS,AV2)}$
$\{Message\}_{SEK(AV1,AV2)}$

**Fig. 6.** AV authentication process.

is corrupted and is using a compromised session key and then, it will inform the RC. The RC in turn will inform all the other AVNET's MS of its decision to revoke the compromised session key. At the reception of the message, the MS will delete the given session keys in the session key's list, meaning the AV that was previously authorized to use a given session key will need to re-authenticate before it can communicate in the AVNET.

### 3.5. Critical analysis of the proposed cryptographic system

The aim of this section is twofold. We first use an intuitive analysis method to show the security properties that our secure scheme has and we compare it to existing secure schemes for MEC architectures, and secondly, we use the Avispa tool [31] to validate our proposed secure scheme.

#### 3.5.1. Security properties of our scheme

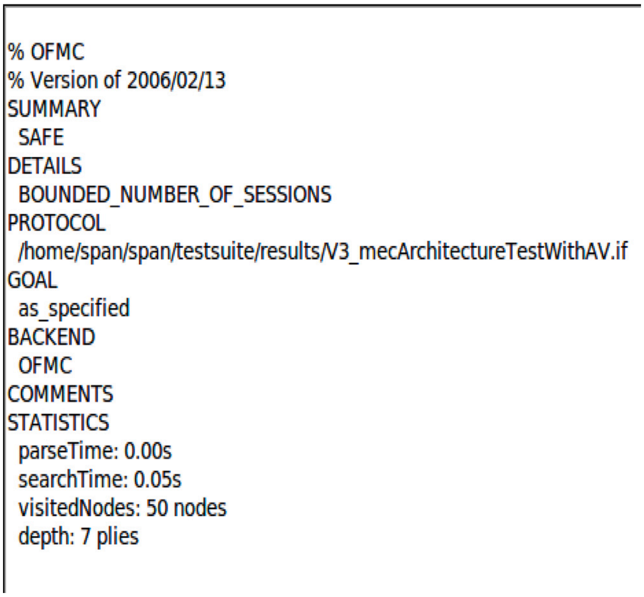
In this section, we present the fundamental security properties that our scheme proposes and prove why they are significant. Our proposed secure scheme has the following properties:

1. **Message authentication**: When a message m comes from a network entity E1 and arrives in a network entity E2, E2 first checks if E1 is an authorized network entity before accepting the message. In our scheme, this is done using either public keys or session keys as we described above, depending on the execution phase (registration or authentication).
2. **Agent authentication**: The agent authentication consists of each network entity E1, to establish whether it can identify network entity E2 if so a successful communication session is established. This is achieved in our scheme, since for each session key establishment, MS or RC shares the session key between E1 and E2 accompanied by the network entities ID, in an encrypted message.
3. **User un-traceability**: the generated session key established between each pair of network equipment is randomly generated during both session key establishment and session key renewal, by doing so, there is no possibility for an intruder to use an obsolete session key to communicate with any given network equipment, then this guarantees that the user is not traceable since each established session corresponds to a new session key generated.
4. **Secrecy**: The secrecy property ensures that an attacker cannot access encrypted messages. This property is achieved in our scheme since each message that circulates in our network is either encrypted by a session key or the Diffie–Hellman key.
5. **Freshness**: the freshness property particularly helps to ensure that during session key establishment, the generated session key is precisely generated for the actual session to prevent an attacker from sending a fake session key to authenticated network equipment. The property is achieved in our scheme by the use of Nonces during the registration phase.
6. **Session key agreement**: each message exchange between network equipment is secured using a session key established during the registration phase or during the authentication phase (in case of communication between two AVs).
7. **Session key renewal**: Our scheme ensures that the session keys are renewed after a given delay. This helps to avoid network equipment impersonation attacks. The property also ensures the security of network equipment during mobility.
8. **Resistance to impersonation attack**: the impersonation attack is avoided in our scheme through the use of a session key establishment and renewal. Before network equipment is authorized to communicate in the network, it should first be authenticated by a third-party authenticated user network equipment (MS or RC).
9. **Automatic provability**: our scheme is evaluated using AVISPA, which is a tool used to validate the correctness of cryptographic schemes designed for network communication protocols.

We compare these security properties for our protocol to those of Mohammad et al. [5], Jia et al. [17], Li et al. [23] and Li et al. [32] in Table 1. In Table 1, we observed that unlike the protocol by Jia et al. [17] where the secrecy and the freshness are not considered, the protocol by Li et al. [32] where the agent authentication, the session key renewal and the mitigation of impersonation attack are not considered, and the protocol by Li et al. [23] where the message authentication, the user untraceability and the session key renewal are not considered, all the security properties that we highlighted are considered in both the protocol by Mohammad et al. [5] and ours. Furthermore, our proposed scheme was tested for vulnerabilities using AVISPA, unlike the ones proposed in [17,23,32] which were not verified.

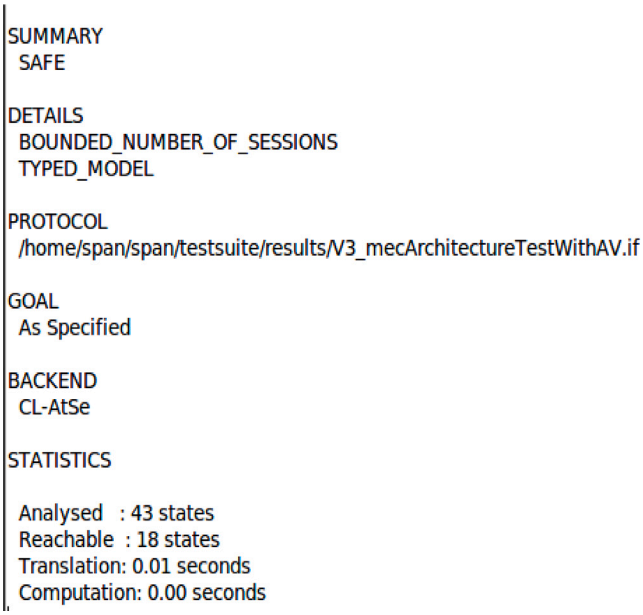
**Table 1**
Comparison of security schemes.

| Security properties | Jia et al. [17] | Li et al. [32] | Li et al. [23] | Mohammad et al. [5] | Our Scheme |
|---|---|---|---|---|---|
| Message authentication | Yes | Yes | No | Yes | Yes |
| Agent authentication | Yes | No | Yes | Yes | Yes |
| User untraceability | Yes | Yes | No | Yes | Yes |
| Secrecy | No | Yes | Yes | Yes | Yes |
| Freshness | No | Yes | Yes | Yes | Yes |
| Session key agreement | Yes | Yes | Yes | Yes | Yes |
| Session key renewal | No | No | No | Yes | Yes |
| Resistance to impersonation attack | Yes | No | Yes | Yes | Yes |
| Single Sign On (SSO) | Yes | Yes | Yes | Yes | Yes |
| Automatic provability | No | No | No | Yes | Yes |



```
% OFMC
% Version of 2006/02/13
SUMMARY
 SAFE
DETAILS
 BOUNDED_NUMBER_OF_SESSIONS
PROTOCOL
 /home/span/span/testsuite/results/V3_mecArchitectureTestWithAV.if
GOAL
 as_specified
BACKEND
 OFMC
COMMENTS
STATISTICS
 parseTime: 0.00s
 searchTime: 0.05s
 visitedNodes: 50 nodes
 depth: 7 plies
```

**Fig. 7.** Result of our secure scheme on the OFMC checker.



```
SUMMARY
 SAFE

DETAILS
 BOUNDED_NUMBER_OF_SESSIONS
 TYPED_MODEL

PROTOCOL
 /home/span/span/testsuite/results/V3_mecArchitectureTestWithAV.if

GOAL
 As Specified

BACKEND
 CL-AtSe

STATISTICS

 Analysed   : 43 states
 Reachable  : 18 states
 Translation: 0.01 seconds
 Computation: 0.00 seconds
```

**Fig. 8.** Result of our secure scheme on the CL-AtSe checker.

### 3.5.2. Validation of our secure scheme with avispa

In order to evaluate the effectiveness of our secured scheme, we use the AVISPA (Automated Validation of Internet Security Protocols and Applications) tool [31]. Formal methods which are validated using Avispa are efficient and mature for the design of secured protocols [22]. It helps to detect attacks such as Man in the Middle attacks (MITM). AVISPA implements multiple back-end techniques which are able to detect attacks on the input protocols. Among these implemented back-end techniques, we have (1) the On-the-fly Model-Checker (OFMC) that performs protocol falsification and bounded verification, (2) the Constraint- Logic-based Attack Searcher (CL-AtSe) that applies straint solving with powerful simplification heuristics and redundancy elimination techniques, (3) the SAT-based Model-Checker (SATMC), and (4) the Tree Automata based on Automatic Approximations for the Analysis of Security Protocols (TA4SP). AVISPA shows its relevance to many protocols that were already standardized by organizations like the Internet Engineering Task Force (IETF).

Figs. 7 and 8 show that our secured scheme is safe and trusted on the OFMC checker and CL-AtSe checker respectively.

Unfortunately, Avispa does not evaluate the computational and communicational properties of the developed protocol. That is why simulations were employed to evaluate the properties during the real execution of our proposal.

## 4. Simulations and analysis

In this section, we present the simulation results of our proposed secured solution. The simulations used the computation cost and the communication cost metrics. Table 2 presents the list of parameters used for comparison. The execution time of different operations was performed by the work in [24]. In order to use these execution times in our simulations, we compute the number of operation of each type, then according to the cost of individual operation obtained in [24], we compute the global operation cost of our solution and we compare it to the existing solutions. Simulations were performed on a desktop with Ubuntu 20.04.4 LTS, 8 GB of RAM, and a Core i5 3.4 GHz Processor. These simulations were written in Python version 3.

### 4.1. Computational cost comparison

Table 3 presents the computational cost of the schemes proposed by Jia et al. [17], Li et al. [23], Mohammad et al. [5], Kaur et al. [9], Yashar et al. [33], Irshad et al. [24] and our scheme. The computational cost represents the required computational aspects required by AV and MS to be authenticated in the AVNET. We did not present the computational cost for RC since RC is located in the cloud data center that does not have computational constraints like MS and AV however, the cost is negligible.

The results presented in Table 3 and Fig. 9 present the computational cost with multiple execution instances in AV and MS. Fig. 9-(a), (b), and (c) show that except for the secure scheme presented by Yashar et al. [33], both AV and MS in our scheme consume less computational time than those of Jia et al. [17], Li et al. [23], Mohammad et al. [5], Kaur et al. [9] and Irsahd et al. [24]. Moreover, all the existing solutions consider the existence of a secure channel for the registration phase,

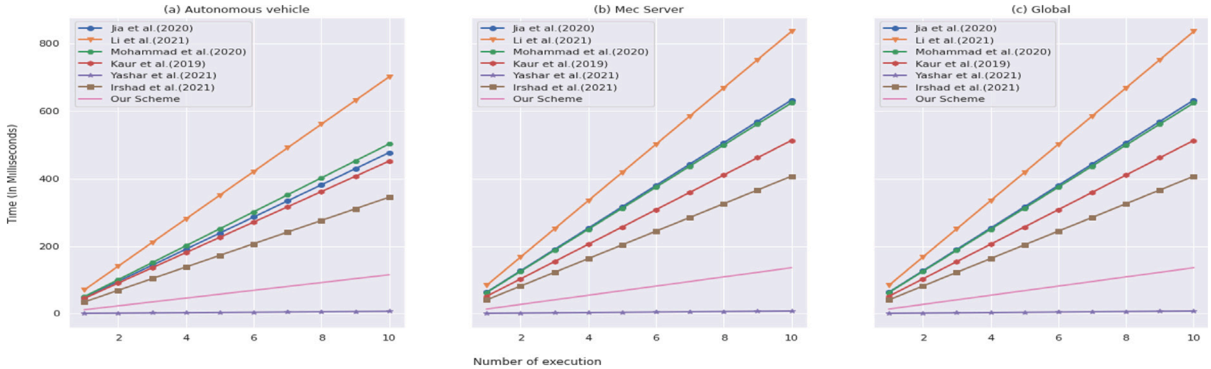


**Fig. 9.** Computational cost.

**Table 2**
Execution time of different operations (in milliseconds).

| Parameter | | Entity | |
|---|---|---|---|
| Symbol | Significance | Autonomous vehicle | MEC server |
| $T_{encPK}$ | Time to encrypt a message with a given key (public or DHWI) | 0.133 | 0.019 |
| $T_{decPK}$ | Time to decrypt a message with given key (private or DHWI) | 0.133 | 0.019 |
| $T_{mtp}$ | Time of map-to-point hash function | 290.433 | 5.388 |
| $T_{exp}$ | Time to perform an exponentiation operation | 2.361 | 0.325 |
| $T_{mul}$ | Time to perform a multiplication operation | 0.013 | 0.003 |
| $T_h$ | Time to perform a general hash operation | 0.067 | 0.010 |
| $T_{pa}$ | Time to perform a point addition operation | 0.079 | 0.024 |
| $T_{sm}$ | Time to perform a scalar multiplication operation | 11.228 | 2.026 |
| $T_{bp}$ | Time to perform a bilinear paring operation | 28.592 | 5.317 |

**Table 3**
Computation cost comparison (in milliseconds).

| Scheme | Autonomous vehicle | MEC server | Total |
|---|---|---|---|
| Jia et al. [17] | $4T_{sm} + T_{exp} + T_{pa} + 5T_h = 47.687$ | $T_{bp} + 5T_{sm} + 3T_{pa} + 5T_h = 15.569$ | 63.256 |
| Li et al. [23] | $6T_{sm} + T_{exp} + T_{pa} + 5T_h = 70.143$ | $T_{pb} + 4T_{sm} + 2T_h = 13.441$ | 83.584 |
| Mohammad et al. [5] | $4T_{sm} + 2T_{exp} + T_{pa} + 8T_h = 50.249$ | $T_{bp} + 3T_{sm} + 2T_{pa} + 2T_{exp} + 5T_h = 12.143$ | 62.392 |
| Kaur et al. [9] | $4T_{sm} + 4T_h = 45.18$ | $3T_{sm} + 4T_h = 6.118$ | 51.298 |
| Yashar et al. [33] | $4T_h + T_{encPK} + 2T_{decPK} = 0.667$ | $4T_h + 2T_{encPK} + T_{decPK} = 0.097$ | 0.764 |
| Irshad et al. [24] | $3T_{sm} + 10T_h + T_{encPK} = 34.489$ | $3T_{sm} + 10T_h + T_{encPK} = 6.197$ | 40.684 |
| Our Scheme | $T_{sm} + T_{encPK} + T_{dekPK} = 11.494$ | $T_{sm} + 2T_{decPK} + T_{enkPK} = 2.083$ | 13.577 |

**Table 4**
Size of operation for communication cost evaluation.

| Parameter | | Value (in bits) |
|---|---|---|
| Symbol | Significance | |
| $\|G\|$ | Size of elements in group $G$ | 1024 |
| $\|SEK\|$ | $Sizeofasessionkey$ | 1024 |
| $\|PUK\|$ | $Sizeofapublickey$ | 1024 |
| $\|Z_q^*\|$ | Size of elements in $Z_q^*$ | 160 |
| $\|H\|$ | Size of Hashing elements of $ID$ | 256 |
| $\|ID\|$ | Size of $ID$ | 256 |
| $\|T\|$ | Size of Timestamp $T$ | 32 |
| $\|N\|$ | Size of Nonce $N$ | 32 |

which is not realistic in regards to the type of network (Mobile ad hoc Network-MANET). Furthermore, except the scheme by Mohammad et al. [5], none of the schemes considered the session key revocation after a given period, which may lead to a node being compromised. We conclude that our security scheme is superior to the evaluated schemes.

### *4.2. Communicational cost comparison*

For the communicational cost evaluation, we used the same metrics that have been used by Kaur et al. [9], Mohammad et al. [5] and Irshad et al. [24]. These metrics and their description are summarized in Table 4.

Table 5 shows the communicational cost (in bits) in comparison with the number of messages exchanged by evaluated schemes.

Fig. 10 shows the communicational cost (in bits) of the compared schemes while the number of execution increase. We observed that our scheme incurred lower communication costs than the one by Yashar et al. [33], but it has a greater communicational cost than the presented in [5,9,17,19,23]. In comparison to the computational cost presented in Section 4.1, we can conclude that the gain in the computational aspect is compensated by a loss in the communicational aspect. However, since our scheme guarantees more security properties than the ones we evaluated and compared our scheme to, in the SDN-NFV-SFC-NS-based MEC architecture, it is considered to be the best scheme.

## 5. Conclusion

The aim of this paper was to propose a data security and privacy scheme for user QoE in a MEC-based autonomous vehicular network (AVNET) that uses the SDN-NFV-SFC-NS technologies as recommended by Filali et al. [6]. To achieve our goal, we presented a MEC architecture using the existing technologies for an AVNET, then we defined a secure scheme that ensures data security and user privacy in the MEC architecture. We then analyzed the properties that our secure scheme provides. Moreover, we used the AVISPA tool to prove the effectiveness of our secure scheme for security and data privacy. Finally, we performed simulations to evaluate the communicational and computational costs of our proposal. The behavior of our simulation

**Table 5**
Communicational cost comparison.

| Scheme | Cost | Number of bits | Number of messages |
|---|---|---|---|
| Jia et al. [17] | $4\|G\|+2\|Z_q^*\|+\|ID\|+2\|T\|$ | 4736 | 2 |
| Li et al. [23] | $4\|G\|+4\|Z_q^*\|+2\|T\|$ | 4800 | 5 |
| Mohammad et al. [5] | $3\|G\|+2\|Z_q^*\|+\|ID\|+2\|T\|$ | 3712 | 2 |
| Kaur et al. [9] | $2\|G\|+2\|T\|+2\|H\|$ | 2624 | 3 |
| Yashar et al. [33] | $2\|G\|+4(\|PUK\|+\|Z_q^*\|+\|H\|+\|ID\|+\|N\|+\|T\|)$ | 9088 | 3 |
| Irshad et al. [24] | $2\|G\|+3\|Z_q^*\|$ | 2528 | 3 |
| Our scheme | $3\|SEK\|+8\|ID\|+4\|N\|+3\|T\|$ | 5344 | 5 |

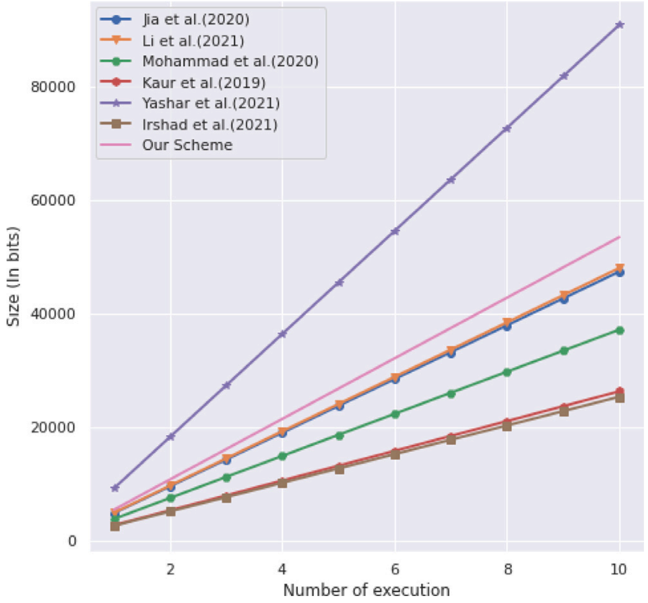


**Fig. 10.** Communicational cost.

results is consistent with those in literature. Furthermore, the simulation results show that our proposed scheme is the best outperforming scheme in terms of computational cost, but the opposite is observed for the communicational scenario. However, we recommend the use of our secured scheme for MEC infrastructures since, contrary to the ones we observed in the literature, our scheme does not consider the existence of a secure channel for the admission of AV and MS into the network, which is more realistic in regard to the type of considered network (MANET).

## CRediT authorship contribution statement

**Miguel Landry Foko Sindjoung:** Conceptualization, Methodology, Software, Validation, Formal analysis, Investigation, Data curation, Writing – original draft, Writing – review & editing, Visualization. **Mthulisi Velempini:** Conceptualization, Methodology, Writing – original draft, Writing – review & editing, Supervision, Project administration, Funding acquisition. **Clémentin Tayou Djamegni:** Conceptualization, Writing – original draft, Writing – review & editing, Supervision.

## Declaration of competing interest

The authors declare that they have no known competing financial interests or personal relationships that could have appeared to influence the work reported in this paper.

## Data availability

Data will be made available on request.

## Funding

This work is based on the research supported wholly by the National Research Foundation of South Africa (Grant Numbers: 141918)

## References

[1] Foko Sindjoung ML, Minet P. Estimating and predicting link quality in wireless IoT networks. Ann Télécommun 2022;77(5–6):253–65. http://dx.doi.org/10.1007/s12243-021-00835-1.

[2] Abbas N, Zhang Y, Taherkordi A, Skeie T. Mobile edge computing: A survey. IEEE Internet Things J 2018;5(1):450–65. http://dx.doi.org/10.1109/JIOT.2017.2750180.

[3] Shirazi SN, Gouglidis A, Farshad A, Hutchison D. The extended cloud: Review and analysis of mobile edge computing and fog from a security and resilience perspective. IEEE J Sel Areas Commun 2017;35(11):2586–95. http://dx.doi.org/10.1109/JSAC.2017.2760478.

[4] Kong X, Wu Y, Wang H, Xia F. Edge computing for internet of everything: A survey. IEEE Internet Things J 2022;1. http://dx.doi.org/10.1109/JIOT.2022.3200431.

[5] Rakeei MA, Moazami F. An efficient and provably secure authenticated key agreement scheme for mobile edge computing. 2020, Cryptology ePrint Archive, Report 2020/1566, https://ia.cr/2020/1566.

[6] Filali A, Abouaomar A, Cherkaoui S, Kobbane A, Guizani M. Multi-access edge computing: A survey. IEEE Access 2020;8:197017–46. http://dx.doi.org/10.1109/ACCESS.2020.3034136.

[7] Buford JF, Yu H, Lua EK. Chapter 9 - peercasting and overlay multicasting. In: Buford JF, Yu H, Lua EK, editors. P2P networking and applications. Boston: Morgan Kaufmann; 2009, p. 203–28. http://dx.doi.org/10.1016/B978-0-12-374214-8.00009-X.

[8] Abar T, Ben Letaifa A, El Asmi S. In: Hurson AR, editor. Chapter five - user behavior-ensemble learning based improving QoE fairness in HTTP adaptive streaming over SDN approach. Advances in computers, vol. 123, Elsevier; 2021, p. 245–69. http://dx.doi.org/10.1016/bs.adcom.2021.01.004.

[9] Kaur K, Garg S, Kaddoum G, Guizani M, Jayakody DNK. A lightweight and privacy-preserving authentication protocol for mobile edge computing. In: 2019 IEEE global communications conference. GLOBECOM, 2019, p. 1–6. http://dx.doi.org/10.1109/GLOBECOM38437.2019.9013856.

[10] Foko Sindjoung ML, Velempini M, Bomgni AB. A MEC architecture for a better quality of service in an autonomous vehicular network. Comput Netw 2022;219:109454. http://dx.doi.org/10.1016/j.comnet.2022.109454.

[11] Yala L, Frangoudis PA, Ksentini A. Latency and availability driven VNF placement in a MEC-NFV environment. In: 2018 IEEE global communications conference. GLOBECOM, 2018, p. 1–7. http://dx.doi.org/10.1109/GLOCOM.2018.8647858.

[12] Cziva R, Anagnostopoulos C, Pezaros DP. Dynamic, latency-optimal vNF placement at the network edge. In: IEEE INFOCOM 2018 - IEEE conference on computer communications. 2018, p. 693–701. http://dx.doi.org/10.1109/INFOCOM.2018.8486021.

[13] Peng H, Ye Q, Shen XS. SDN-based resource management for autonomous vehicular networks: A multi-access edge computing approach. IEEE Wirel Commun 2019;26(4):156–62. http://dx.doi.org/10.1109/MWC.2019.1800371.

[14] Guanwen L, Huachun Z, Bohao F, Guanglei L, Taixin L, Qi X, Wei Q. Fuzzy theory based security service chaining for sustainable mobile-edge computing. Mob Inf Syst 2017;2017. http://dx.doi.org/10.1155/2017/8098394.

[15] Foko Sindjoung ML, Bomgni AB, Tagne Fute E, Chalhoub G, Tayou Djamegni C. ISCP : An instantaneous and secure clustering protocol for wireless sensor networks. Netw Protoc Algorithms 2018;10(1):65–82. http://dx.doi.org/10.5296/npa.v10i1.12574.

[16] Bomgni AB, Foko Sindjoung ML, Kamdem Tchibonsou D, Velempini M, Myoupo JF. NESEPRIN: A new scheme for energy-efficient permutation routing in IoT networks. Comput Netw 2022;214:109162. http://dx.doi.org/10.1016/j.comnet.2022.109162.

[17] Jia X, He D, Kumar N, Choo K-KR. A provably secure and efficient identity-based anonymous authentication scheme for mobile edge computing. IEEE Syst J 2020;14(1):560–71. http://dx.doi.org/10.1109/JSYST.2019.2896064.

[18] Tsai J-L, Lo N-W. A privacy-aware authentication scheme for distributed mobile cloud computing services. IEEE Syst J 2015;9(3):805–15. http://dx.doi.org/10.1109/JSYST.2014.2322973.

[19] Irshad A, Sher M, Ahmad HF, Alzahrani BA, Chaudhry SA, Kumar R. An improved multi-server authentication scheme for distributed mobile cloud computing services. KSII Trans Internet Inf Syst 2016;10(12):6092–115. http://dx.doi.org/10.3837/tiis.2016.12.021.

[20] Ahmad I, Shahabuddin S, Kumar T, Okwuibe J, Gurtov A, Ylianttila M. Security for 5G and beyond. IEEE Commun Surv Tutor 2019;21(4):3682–722. http://dx.doi.org/10.1109/COMST.2019.2916180.

[21] He D, Kumar N, Khan MK, Wang L, Shen J. Efficient privacy-aware authentication scheme for mobile cloud computing services. IEEE Syst J 2018;12(2):1621–31. http://dx.doi.org/10.1109/JSYST.2016.2633809.

[22] Odelu V, Das AK, Kumari S, Huang X, Wazid M. Provably secure authenticated key agreement scheme for distributed mobile cloud computing services. Future Gener Comput Syst 2017;68:74–88. http://dx.doi.org/10.1016/j.future.2016.09.009.

[23] Li Y, Cheng Q, Liu X, Li X. A secure anonymous identity-based scheme in new authentication architecture for mobile edge computing. IEEE Syst J 2021;15(1):935–46. http://dx.doi.org/10.1109/JSYST.2020.2979006.

[24] Irshad A, Chaudhry SA, Alomari OA, Yahya K, Kumar N. A novel pairing-free lightweight authentication protocol for mobile cloud computing framework. IEEE Syst J 2021;15(3):3664–72. http://dx.doi.org/10.1109/JSYST.2020.2998721.

[25] Amin R, Kumar N, Biswas G, Iqbal R, Chang V. A light weight authentication protocol for IoT-enabled devices in distributed cloud computing environment. Future Gener Comput Syst 2018;78:1005–19. http://dx.doi.org/10.1016/j.future.2016.12.028.

[26] Hou Y, Garg S, Hui L, Jayakody DNK, Jin R, Hossain MS. A data security enhanced access control mechanism in mobile edge computing. IEEE Access 2020;8:136119–30. http://dx.doi.org/10.1109/ACCESS.2020.3011477.

[27] Zhang J, Chen B, Zhao Y, Cheng X, Hu F. Data security and privacy-preserving in edge computing paradigm: Survey and open issues. IEEE Access 2018;6:18209–37. http://dx.doi.org/10.1109/ACCESS.2018.2820162.

[28] Medhat AM, Taleb T, Elmangoush A, Carella GA, Covaci S, Magedanz T. Service function chaining in next generation networks: State of the art and research challenges. IEEE Commun Mag 2017;55(2):216–23. http://dx.doi.org/10.1109/MCOM.2016.1600219RP.

[29] Mansour I, Chalhoub G, Lafourcade P. Evaluation of secure multi-hop node authentication and key establishment mechanisms for wireless sensor networks. J Sens Actuat Netw 2014;3(3):224–44. http://dx.doi.org/10.3390/jsan3030224.

[30] Foko Sindjoung ML, Minet P. Wireless link quality prediction in IoT networks. In: The 8th IFIP/IEEE international conference on performance evaluation and modeling in wired and wireless networks - (PEMWN 2019). Paris, France; 2019.

[31] Armando A, Basin D, Boichut Y, Chevalier Y, Compagna L, Cuellar J, Drielsma PH, Heám PC, Kouchnarenko O, Mantovani J, Mödersheim S, von Oheimb D, Rusinowitch M, Santiago J, Turuani M, Viganò L, Vigneron L. The AVISPA tool for the automated validation of internet security protocols and applications. In: Etessami K, Rajamani SK, editors. Computer aided verification. Berlin, Heidelberg: Springer Berlin Heidelberg; 2005, p. 281–5.

[32] Li C-T, Lee C-C, Weng C-Y, Fan C-I. An extended multi-server-based user authentication and key agreement scheme with user anonymity. KSII Trans Internet Inf Syst 2013;7:119–31. http://dx.doi.org/10.3837/tiis.2013.01.008.

[33] Yashar s, Yaser E, Vahid K. CE–SKE: cost–effective secure key exchange scheme in fog federation. Iran J Comput Sci 2021;4:305–17. http://dx.doi.org/10.1007/s42044-021-00086-2.