Transparent AAA Security Design for Low-Latency MEC-Integrated Cellular Networks

Chi-Yu Li, Ying-Dar Lin, *Fellow, IEEE*, Yuan-Cheng Lai, Hsu-Tung Chien, Yu-Sheng Huang, Po-Hao Huang, and Hsueh-Yang Liu

***Abstract*—Multi-access edge computing (MEC) is a key enabler for low-latency services in the cellular network. It enables service requests to be served at the edge without reaching the Internet. However, this service model allows data traffic to bypass conventional security functions deployed at the core network, and may pose security threats. To examine its security impact, we analyze current security functions that span authentication, authorization, accounting (AAA), and access control, and then identify two major issues. First, conventional user authentication methods prevent MEC applications from achieving low-latency service offering. Second, current cellular authorization, accounting, and access control mechanisms hardly secure MEC traffic. We thus propose a transparent security design called MECsec to secure the MEC with low latency in the cellular network. It contains three main components: cellular-based OpenID Connect (OIDC) authentication, bitmap-based authorization/accounting, and two-tier hash-based access control. Especially, its transparent design does not need any changes on current cellular operations, and is standard-compliant. We implement and evaluate the MECsec prototype on an MEC-integrated LTE network architecture developed based on the OpenAirInterface (OAI) cellular platform. Our results show that the cellular-based OIDC can reduce delays of current authentication methods by up to 88.3%, and the other components can successfully defend against possible threats with negligible overhead.**

***Index Terms*—Edge computing, MEC security, network security, 4G LTE network, cellular network.**

I. Introduction

MULTI-ACCESS Edge Computing (MEC), formerly Mobile Edge Computing, is a new concept that deploys cloud computing systems at network edges. It can reduce application latency by running application servers closer to end devices. It has been determined as a key feature for the cellular network by both ETSI [1] and 3GPP [2] standards. It can benefit many latency-sensitive applications, such as virtual reality (VR) and vehicle-to-everything (V2X), for cellular users in 4G/5G networks. A research study [3] shows that its global market size is projected to reach USD 3.24 billion by 2025 with a compound annual growth rate of 41.0% during the forecast period.

In this work, we focus on the MEC platform deployed next to a base station in the cellular network. It hosts application servers to provide services while redirecting service flows from end devices to them. It prevents the service flows from traversing the core network to reach Internet servers, so they can achieve low latency without experiencing network congestion or long propagation delay. However, such service model skips the conventional security functions deployed at the core network. It leaves MEC components unprotected to the attacks coming from end devices. Even when some of current security functions are applicable to the MEC components, they may need to interact with Internet servers (e.g., user authentication) and thus offset low-latency gains from the MEC deployment.

To this end, we take a systematic analysis for the MEC on current security functions that span authentication, authorization, accounting (AAA), and access control, from three aspects: application, control plane, and user plane. We discover two major issues. First, current user authentication methods prevent MEC applications from achieving low-latency service offering, since they need to involve interactions with Internet servers, which may experience network congestion and long propagation delays. For example, the application servers based on the password-based authentication or OpenID Connect (OIDC) [4] methods need to interact with their original or third-party authentication servers respectively on the Internet. Second, current cellular authorization, accounting, and access control mechanisms are not applicable to the MEC, because they are deployed at the core network. It may expose the MEC or the cellular network to security threats from malicious end devices. For example, end device users may generate unauthorized traffic, which does not reach the core network, to abuse MEC and cellular network resources.

We then propose a transparent security design called MECsec to secure the MEC-integrated cellular network with low latency. Both of its security design and underlying MEC deployment are transparent to the cellular network infrastructure. They are standard-compliant and thus do not require any changes on current cellular operations. To address the above two issues, we introduce three main components in the MECsec design: cellular-based OIDC authentication, bitmap-based authorization/accounting, and two-tier hash-based access control. Specifically, the first component reduces the delay of the user authentication for MEC applications by leveraging the OIDC concept

Manuscript received March 9, 2019; revised August 22, 2019 and December 7, 2019; accepted December 8, 2019. Date of publication January 7, 2020; date of current version March 12, 2020. This work was supported in part by the Ministry of Science and Technology, Taiwan, under Grants MOST-106-2628-E-009-003-MY3 and MOST-107-2923-E-009-005. The review of this article was coordinated by Dr. J. Liu. *(Corresponding author: Chi-Yu Li.)*

C.-Y. Li, Y.-D. Lin, H.-T. Chien, Y.-S. Huang, P.-H. Huang, and H.-Y. Liu are with the Department of Computer Science, National Chiao Tung University, Hsinchu 300, Taiwan (e-mail: chiyuli@cs.nctu.edu.tw; ydlin@cs.nctu.edu.tw; hsutung@cs.nctu.edu.tw; yushuang@cs.nctu.edu.tw; huangpoh@cs.nctu.edu.tw; whistlewickystar@cs.nctu.edu.tw).

Y.-C. Lai is with the Department of Information Management, National Taiwan University of Science and Technology, Taipei 106, Taiwan (e-mail: laiyc@cs.ntust.edu.tw).

Digital Object Identifier 10.1109/TVT.2020.2964596

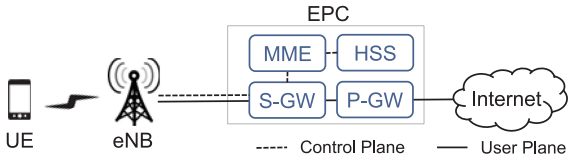

Fig. 1. 4G LTE network architecture.

and an existing cellular authenticator to keep its whole procedure at the edge. The second one enables authorization/accounting functions with fast policy propagation and alteration for the MEC. The third one identifies various traffic types in $O(1)$ time to defend against unauthorized resource access with low overhead.

We implement and evaluate the MECsec design on an MEC-integrated LTE network prototype [5] based on the OpenAirInterface (OAI) cellular platform [6]. The evaluation covers both delay and security performance, as well as performance overheads. The cellular-based OIDC can provide the user authentication with up to 88.3% reduction at median delays, compared with conventional authentication methods. By randomly generating various benign and malicious traffic types, we validate that the authorization/accounting and access control mechanisms can successfully defend against unauthorized resource access from both MEC and non-MEC devices. The result also shows that they impose negligible overheads on both the MEC platform's forwarding bandwidths and the MEC devices' service delays.

The rest of this paper is organized as follows. Section II introduces the 4 G LTE network and presents related work. In Section III, we present the MEC-integrated cellular architecture and analyze its security issues. We conduct a case study on the MEC security functions in Section IV. Sections V, VI, and VII respectively design, implement, and evaluate MECsec. Section VIII discusses several practical issues and Section IX concludes the paper.

II. Background and Related Work

A. 4 G LTE Network Architecture

The 4 G LTE network consists of two major parts, the radio access network and the core network, as shown in Fig. 1. The radio access network includes user equipments (UEs) and base stations, which are named as evolved node B (eNB). The core network called evolved packet core (EPC) embraces four major components: mobility management entity (MME), home subscriber server (HSS), serving gateway (S-GW), and packet data network gateway (P-GW). Both control and user planes span the UE, the eNB and the EPC. In the control plane, the MME provides mobility support, whereas the HSS stores user profiles and does user authentication/authorization. In the user plane, the eNB, the S-GW, and the P-GW forward data packets between the UE and the Internet. The interface between the eNB and the S-GW in the core network is called S1 [7] and consists of two parts: S1-MME [8] and S1-U [9]. They are respectively used to delivery control messages and user-plane packets. The transport of data packets on the S1-U interface relies on the GPRS tunneling protocol (GTP) [10]. A GTP tunnel between two ends is built for each UE's unidirectional Internet traffic. The uplink and downlink tunnels of the UE are assigned different tunnel IDs (TEIDs). This pair of TEIDs is uniquely associated with the UE's IP address.

B. Related Work

Many research studies have examined security issues of 4 G/5 G networks. Two recent studies [11], [12] take systematic analyses on security aspects of 4 G/5 G networks. Specifically, LTEInspector [11] combines a symbolic model checker and a cryptographic protocol verifier to systematically diagnose vulnerabilities of LTE networks, whereas the other [12] formulates a formal model, together with a security protocol verification tool, to identify missing security goals of 5 G authentication protocols. Some of the studies discover practical threats of LTE access networks from privacy [13], [14] and availability [14], [15] aspects, whereas the others focus on security vulnerabilities of IP multimedia subsystem (IMS) applications/systems [16]–[19] and charging architecture [20] in the LTE core network. However, neither of them considers security issues of the MEC-integrated cellular network.

There have been many survey studies about the MEC from various aspects. Some of them focus on its security research, which includes Internet of things (IoT) security on the MEC [21], as well as data security and privacy for the MEC [22]. The others study application scenarios and research challenges [23], applied technologies (e.g., network virtualization and software defined networking) [24], architecture and computation offloading [25], and communication issues [26]. Current MEC research studies for cellular networks can be classified into three categories: architecture and deployment, resource offloading and allocation, and security.

For the architecture and deployment, the studies respectively implement an MEC cache system [27] and an MEC-integrated LTE network architecture using a middlebox approach [5]. For the resource offloading and allocation, Guo *et al.* [28] provide collaborative offloading solutions of the cloud and edge computation with fiber-wireless networks; Tran *et al.* [29] jointly optimize task offloading, transmission power, and computing resource allocation at MEC servers; Guo *et al.* [30] study the energy-aware computation offloading over edge servers in ultradense IoT networks; Wang *et al.* [31] introduce an integrated framework for MEC computation offloading and interference management in cellular networks. The MECsec solution can complement these studies from the security perspective. It can serve as an AAA security service for the MEC systems, and protect MEC resources throughout their offloading and allocation services.

For the security studies, Amadeo *et al.* [32] design a protocol for privacy-friendly MEC service discovery and access based on the named data networking (NDN) paradigm. It supports the scenarios where the identity/location information of each service provider is not priori known, and dynamically builds trust between devices and edge servers. The NDN architecture is different from current client-server communication paradigm, so deploying this solution requires to change service operations

of both applications and edge servers. However, our transparent solution does not require those changes at two communication ends. Wong *et al.* [33] seek to provide AAA security for 5 G systems to support multi-tenancy, multi-network slicing, and multi-level services. It replaces the traditional centrally-governed AAA mechanism with a new hierarchical and distributed AAA approach. It requires current cellular architecture and operations to be changed, but our transparent solution does not have this requirement.

III. Overview

In this work, we focus on low-cost MEC deployment solutions, which can interest carriers most. They should be transparent to existing cellular networks or have minimal changes; that is, no modifications are needed on current cellular facilities and operations (e.g., mobility management and Internet access). Also, they need to have backward compatibility that conventional cellular UEs can still access the Internet as usual without subscribing to the MEC service.[1]

We consider the two most low-cost solutions among the four deployment methods introduced by ETSI [34]: `Bump in the Wire` and `Distributed S-GW`. In the former, the MEC is located at the eNB or deployed next to it by sitting on the S1 interface. It handles GTP-encapsulated packets and routes plain IP packets to/from MEC application servers; meanwhile, it forwards the GTP-encapsulated packets to/from the S-GW for control-plane and regular Internet traffic. In the latter, another local S-GW entity and the MEC are deployed next to the eNB, but the EPC still contains all the components. To connect the eNB to the local S-GW, the MME needs to perform an existing function called local S-GW selection [35].

Both of them can retain current 4 G operations[2] with minimal impact. For the control plane, the EPC still takes a full centralized control so that no operations need be changed. For the user plane, conventional UEs are allowed to consume data services through the EPC as usual. They empower carriers to have the flexibility that offers MEC only as a service, but not a mandatory feature, to UEs. We then develop the MECsec design based on an MEC-integrated cellular architecture that abstracts common essential features from them.

We next present the MEC-integrated cellular architecture, analyze its security issues, and introduce the threat model.

A. MEC-Integrated Cellular Architecture

We consider an MEC platform with three common essential features from the `Bump in the Wire` and `Distributed S-GW` solutions [5]: a traffic shunt, virtualized application servers, and a local DNS server. Fig. 2 shows the MEC-integrated LTE network architecture with them. The traffic shunt, which resides at the MEC platform or the S-GW, is introduced to steer traffic to the MEC. In the uplink direction, the Internet

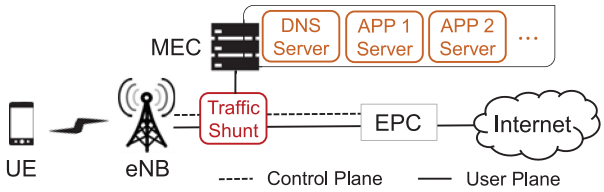

Fig. 2. The MEC-integrated LTE network architecture.

and control-plane packets are forwarded to the EPC, whereas the others go to the MEC. On the other hand, all the downlink packets from both the MEC and the EPC ought to be forwarded to the eNB. To support various applications, the MEC platform should virtualize and host the application servers. The local DNS server is used to redirect traffic to the MEC application servers. Specifically, the DNS server responds to UEs with the servers' local IPs for the MEC application traffic, and then the traffic shunt can steer the traffic to the MEC based on the local IPs.

B. Security Issues

The transition from the conventional LTE network architecture to the new MEC-integrated one may cause issues from current security functions. We examine common AAA and access control functions, in terms of security and latency performance. We below focus on three essential aspects: MEC applications, control plane, and user plane.

MEC Applications: As normal Internet applications, the major security function of the MEC applications is user authentication. Since they work as usual without any changes, the same end-to-end services and security operations remain. No new security issues are expected to arise. However, current authentication methods may have negative impact on the latency performance of service offering. Since storing security context (e.g., user credentials and attributes) locally at the edge for user authentication can hardly support user mobility across different eNBs, they may still rely on authentication servers in the Internet cloud. The required handshakes with the Internet servers may offset the MEC's low-latency benefit, which comes from keeping communication at the edge.

Control Plane: The EPC remains the same in the MEC-integrated cellular architecture, and still takes full control over the control plane. Thus, current authentication method between the UE and the cellular network, i.e., AKA (Authentication and Key Agreement), can work well and be secure as usual. It is one of the advantages of such transparent deployment methods. However, the infrastructure lacks for an authorization mechanism of the new MEC service and its various application servers. It should restrict each MEC resource only to its subscribers and protect it from unauthorized UEs.

User Plane: The MEC traffic, which flows between UEs and the MEC platform, does not traverse the P-GW, where current access control and accounting functions are deployed. So, it can bypass those security functions. Without the access control, malicious UEs may abuse MEC resources and the cellular network resources between the UE and the P-GW. In addition, the absence of the accounting function prevents carriers from having

[1] In this work, the MEC service denotes the access of the MEC platform, and the MEC applications represent application servers hosted by the MEC.

[2] We focus on the 4 G network in this work, but the transparent security design can be easily extended to 5 G networks (See Section VIII).

charging and anomaly detection functions on the MEC traffic. Note that we assume all the MEC applications are benign in this work.

To protect the MEC-integrated architecture with low latency, we need to address three main issues: user authentication of the MEC applications and access authorization of the MEC service/applications, as well as access control and accounting of the MEC traffic.

C. Threat Model

Adversaries are malicious cellular users who attack the MEC-integrated cellular network. They control only their own UEs, but do not have any access to other cellular network components. The cellular and MEC components are not compromised. The MEC platform, which is deployed by the carrier, does not threaten cellular network operations. Adversaries can fabricate packets on their UEs with root access, and employ them to have unauthorized access that abuses MEC or/and cellular network resources (e.g., flooding attacks). They can know whether their accessed eNBs support the MEC and obtain MEC-related information (e.g., from other benign UEs).

IV. Case Study on MEC Security Functions

We study the user authentication of the MEC applications empirically on the latency performance, as well as conduct a security analysis on authorization, accounting, and access control functions for the MEC.

A. User Authentication for MEC Applications

We here consider the user authentication of the MEC applications whose servers migrate from the Internet cloud to the MEC.[3] They mainly have two conventional authentication methods. One is to rely on their original authentication servers on the Internet. They need to be still hosted in the Internet cloud to support user mobility, which requires authentication from many locations. The other is to employ the authentication services offered by OIDC service providers (e.g., Facebook and Google), whose servers are also located in the Internet cloud. From the security perspective, the authentication process does not change and can thus keep the same security assurance. However, both of them involve handshakes between the MEC application servers and the authentication servers on the Internet. It causes the user authentication to experience usual delays without benefiting from the MEC, thereby delaying the service offering.

We conduct experiments to examine authentication delays by varying locations of the application and authentication servers, as shown in Table I. S1 and S2 respectively represent the aforementioned two methods for the MEC application servers. S3 represents conventional cases where both application and authentication servers are located on the Internet. We set up an authentication server in the Google cloud, and two web servers including one at the MEC and the other in the cloud. In S1 and S2, we enable the MEC web server to do user authentication with the authentication server and the Facebook's OIDC service, respectively. In S3, the cloud web server does user authentication with the OIDC service. We have 20 runs for each scenario. The other experimental settings are described in Section VI. Note that the delays do not cover the times of typing usernames and passwords.

[3]Some MEC applications which are designed for local services and can do user authentication without reaching the Internet are not our focus.

TABLE I
Three Authentication Scenarios With Different Locations of Application and Authentication Servers

Scenario	Application server at	Authentication server at
S1	MEC	Internet cloud
S2	MEC	OIDC provider
S3	Internet cloud	OIDC provider

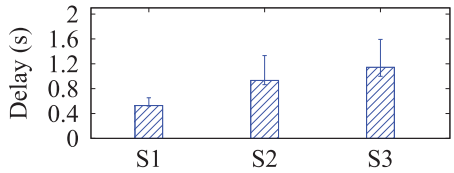

Fig. 3. Authentication delays (min/med/max) for the three scenarios in Table I.

TABLE II
Delay Analysis of the OIDC Authentication Process, Which is Decomposed Into Four Phases, for S2 and S3

Authentication Phase	S2 (ms)	S3 (ms)
Access Service	166	327
Authorization Request	290	300
ID Token Authorization	180	183
User Profile Retrieval	266	272

Fig. 3 shows authentication delays for those three scenarios. We have two observations. First, S2 and S3 have much longer median delays than S1 with 2.2 and 1.8 times, respectively. The reason is that they require several handshakes between the application server and the Facebook's OIDC server on the Internet [4]. But, S1 just needs to reach the Internet once for the user credential verification. Second, S2 saves only 0.21 s from the conventional OIDC scenario S3 by moving its application server to the MEC. We further decompose the authentication process into four phases and analyze the delay of each phase by using the Google Chrome developer tool [36]. Table II shows one trace result of each of S2 and S3, and the other traces have similar trends. We observe that S2 has smaller delay of the first phase, access service, because its UE can access the application server at the MEC instead of the Internet. However, S2 and S3 have similar delays in the other three phases: authorization request, ID token authorization, and user profile retrieval. The UEs in these phases need to communicate with the OIDC server on the Internet.

As a result, the service offering delay of an MEC application server may not benefit much from the MEC architecture when it relies on a user authentication service on the Internet.

B. Authorization for MEC Service

The MEC service is regarded as a new cellular service that offers users various MEC applications, e.g., gaming, virtual

reality, and social network. As conventional cellular application services, carriers should manage user subscriptions and access rights of the MEC applications. Current cellular networks use the policy and charging control (PCC) function [37] as its authorization mechanism. A set of PCC rules for each cellular user is generated based on his/her authorized services. The control plane imposes the PCC rules on the user plane to do access control. However, the PCC rules are applied in the P-GW, which the MEC traffic does not reach. So, it requires a new authorization mechanism for the MEC service.

The new authorization method should address three issues: user diversity, unauthorized access of MEC resources, and attacks from authorized access. First, the user diversity means that users are allowed to subscribe to different sets of MEC applications. Therefore, authorization policies should be specified at the application granularity to identify which applications are authorized to each user. Second, the unauthorized access includes two types: (1) a non-MEC UE without any MEC subscriptions seeks to access any MEC resources; (2) an MEC UE with an MEC subscription set attempts to access any MEC resources which are not authorized to it. Third, malicious UEs may attack the MEC application servers that are authorized to them, so the authorization method should support on-demand de-authorization to stop the attack by canceling the authorization.

The transparent MEC service calls for a new authorization method that handles user diversity, unauthorized access, and on-demand de-authorization.

C. Access Control and Accounting for MEC Traffic

The MEC platform should prevent unauthorized traffic from abusing its resources, as well as do anomaly detection and charging on authorized MEC traffic. It requires access control and accounting functions at the MEC to filter incoming traffic, block unauthorized packets, and collect traffic statistics. Although the user-plane PCC component at the P-GW takes care of similar functions, the MEC traffic does not reach the P-GW. There are two requirements for a new design of those two functions. First, it should be able to fulfill authorization policies, which are at the application granularity, with small overhead. Its traffic inspection needs to consider whether users are accessing their authorized applications. Moreover, any changes of authorization policies can be easily made to the access control at run time. There could be new authorization or de-authorization policies because of new subscribers or malicious events, respectively.

Second, the design should ensure that the user ID on which the authorization relies is authentic; otherwise, the access control and accounting functions can be breached by spoofed user IDs. From the MEC platform's perspective, the user ID is the source IP address of data packets in the GTP payload. However, it could be spoofed by malicious cellular users [20]. In the cellular network, the network segment between the eNB and the P-GW understands only GTP-level information, and the IP information of data packets is checked at the P-GW. That is, spoofed source IP addresses can be detected only at the P-GW, but the MEC traffic does not reach it. It can lead to two spoofing cases that threaten the MEC service: a non-MEC UE spoofs an MEC UE's IP and an MEC UE spoofs another MEC UE's IP. Both cases can breach the accounting and access control functions.

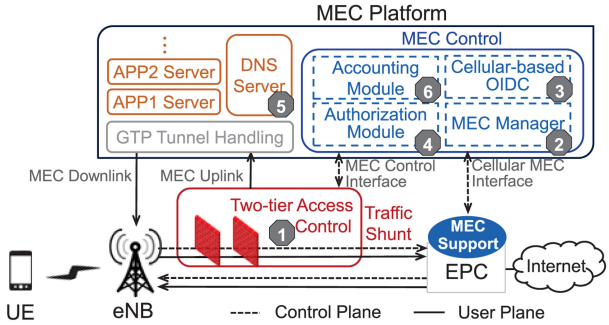

Fig. 4. The MECsec design architecture in the 4G LTE network.

A new design for access control and accounting functions is required for MEC traffic to not only enforce authorization policies at run time but also prevent IP spoofing.

V. MECsec: MEC Security Design for Cellular Networks

We propose a transparent security design called MECsec to secure the MEC-integrated cellular network. It protects the MEC platform and its application servers against malicious cellular users while keeping the MEC service low latency. It covers AAA and access control with three security functions, namely cellular-based OIDC authentication, bitmap-based authorization and accounting, and two-tier hash-based access control. All of them are designed with low-latency operations. Specifically, the cellular-based OIDC shortens the delay of the user authentication by providing an OIDC service based on an authentic cellular identifier. For the authorization and accounting, we develop a bitmap-based approach that specifies authorization policies in bitmaps and enables fast policy propagation among MEC components. For the access control, we use hash functions to identify various traffic types (e.g., MEC, Internet, and illegal ones) in $O(1)$ time, while adopting a two-tier operation to defend against the IP spoofing and the unauthorized MEC access in sequence.

Fig. 4 shows the MECsec design architecture in the 4 G LTE network. We elaborate on the design components below with six parts. The access control module (Part 1) at the traffic shunt filters out illegal or unauthorized packets from the uplink traffic, but steers the others to the MEC or the EPC. It bypasses all the downlink traffic, since the MEC application servers are assumed to be benign and the existing security functions at the P-GW can defend against malicious traffic from the Internet. We next introduce the MEC control framework that takes care of the AAA functions. The MEC manager (Part 2) detects the status of each MEC UE and (de)activates its MEC service accordingly. To activate the MEC service, it fetches the UE's cellular credential/attributes[4] from the MEC support

[4]The UE's cellular credential and attributes include IP, TEIDs, phone number, authorization bitmap, etc.

module through the cellular MEC interface, and then passes them to the cellular-based OIDC and authorization modules. The cellular-based OIDC module (Part 3) uses the uplink TEID as an authenticator for the UE to do the OIDC-based authentication. The authorization module (Part 4) propagates authorization bitmaps to both the DNS server and the access control module. The DNS server (Part 5) responds local IP addresses to only the DNS queries with authorized applications specified in the bitmaps; the other queries are forwarded to the Internet. The access control module produces hash values of authorized traffic flows as permission rules based on the bitmaps, as well as records usage statistics and reports them to the accounting module (Part 6) periodically.

Note that the only support from the current cellular infrastructure is that an MEC support module at the EPC needs to maintain an authorization bitmap for each MEC UE and provide the cellular MEC interface for the MEC manager to query the UE's credential and attributes. This query can be easily supported by the HSS. This add-on module neither interferes with any cellular operations nor requires any changes of current architecture. Carriers should prepare each MEC UE's authorization bitmap based on its MEC subscription.

A. Cellular-Based OIDC Authentication

The cellular-based OIDC module provides the OIDC service based on each UE's authenticated cellular token. It can enable application servers to skip the access of authentication servers on the Internet. Before consuming MEC applications, the UE has been authenticated by the HSS for its cellular network access. The OIDC service can thus leverage an authenticated token which has been bound to the UE. After the authentication, the servers can fetch the UE's application data locally and start to serve it. However, some non-location-based applications may keep user data in their Internet servers. To prevent their servers at the MEC from reaching the Internet during user authentication, we enable the OIDC module to do *data prefetching* for the UE from the Internet servers during its MEC service activation, if there are any. Note that the data prefetching function is configurable for each MEC application.

The uplink TEID can be used as the authenticated token for the OIDC authentication, since it is uniquely associated with each UE. It is attached to each uplink packet by the eNB, so it accompanies each authentication message sent by the UE. Since the eNB is not compromised, as well as the mappings between UE packets and TEIDs are determined based on low-level information derived from SIM cards, the TEIDs are always authentic. This module can thus authenticate users based on the TEIDs of the GTP tunnels that carry their authentication messages.

However, one issue arises for the requirement that the OIDC module needs to check TEIDs: the GTP headers of the authentication packets forwarded to the OIDC module have been stripped off by the GTP tunnel handling module. It prevents the module from observing any GTP information including TEIDs on its received packets. We propose to rely on a *secure binding* between the source IP address and the uplink TEID; once the binding exists, then the source IPs of authentication packets are authentic. This secure binding is also the solution that prevents IP spoofing for the uplink packets in Section IV-C, so we execute it at the access control module. It enables the OIDC module to authenticate MEC users based on the source IPs of their packets.

We next introduce the activation and deactivation of the MEC service, as well as the authentication procedure of the cellular-based OIDC.

MEC Service Activation: The service activation consists of two steps for each MEC UE: detection and initialization. The MEC manager detects new cellular UEs by monitoring new TEIDs from the S1 traffic with the help of the access control module, and then obtains their cellular credentials/attributes from the MEC support module based on their TEIDs. It can check whether a cellular UE subscribes to the MEC service based on its cellular credential, and initialize its MEC service if the subscription exists. Afterwards, it does the MEC initialization for the UE by passing its authorization bitmap to the authorization module and notifying the OIDC module of its credential.

For the UE detection, the MEC manager can inspect the control plane's S1-MME interface or the user plane's S1-U interface. From the S1-MME interface, the MEC manager can inspect a GTP-C packet called `UE Context Setup Request`, which includes both IP address and TEIDs, in the attach procedure. It is sent to the eNB by the MME. From the S1-U interface, it can monitor some initial GTP-U packets (e.g., the device's system traffic) from the UE after its GTP tunnel is built. These two inspection methods are respectively used during and after the attach procedure. Note that the traffic shunt forwards copies of downlink GTP-C packets to the MEC manager for the UE detection.

In the MEC service initialization of a UE, the authorization module enables the DNS server to respond to the UE's DNS queries on its authorized applications with their local IP addresses. It also enables the access control module to permit the UE's authorized MEC traffic flows. The accounting module starts to collect usage statistics, which are reported by the access control module, for the UE based on its bitmap. The OIDC module prefetches the UE's application data based on its unique phone number for the applications that support the data prefetching function. This function should be built based on the collaboration between the MEC and application providers.

MEC Service Deactivation: The UE's MEC service should be deactivated when it stops accessing the MEC. It happens when the UE powers off, deactivates its cellular data usage, or leaves the eNB. These cases can make the UE's GTP-U tunnel be removed. Upon the removal, a data packet that contains a flag called `End Marker` [9] is sent to notify the eNB that the tunnel is going to be torn down right after this packet. Once the MEC manager detects such packet for a UE, it will announce the UE's deactivation to other MEC components.

Authentication Procedure: We devise the authentication procedure of the cellular-based OIDC with two major ideas. First, we enable the OIDC module to serve as an OIDC provider for MEC applications. As shown in Fig. 5, the MEC application does user authentication for the cellular user through the OIDC module. It can limit the authentication to proceed only at the edge without the need of any communication reaching the Internet. Since all the cellular UEs have had their credentials in the EPC and they can be obtained by the OIDC module through the MEC

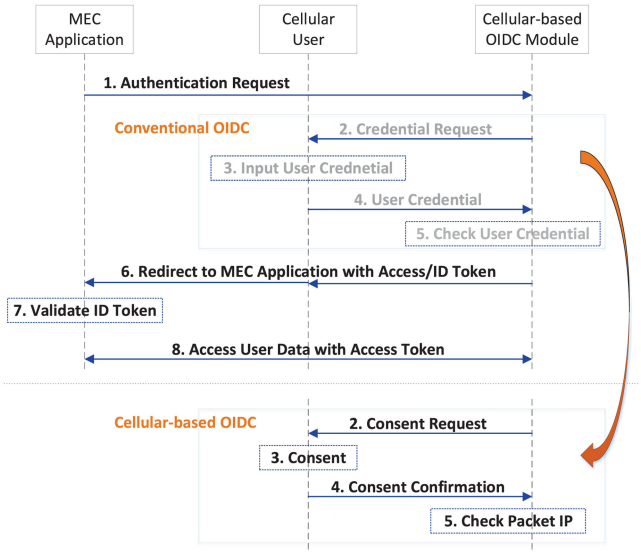

Fig. 5. The general authentication procedure of the cellular-based OIDC.

service activation, they need not do registration with the module before using it for their authentications. So, it can not only shorten the delay but also have high availability that it works for all the cellular UEs.

Second, we omit the input requirement of username and password (Step 3) for the cellular user, but rely on his/her UE IP for the authentication. The input requirement usually takes a long time (e.g., several seconds) to delay service delivery. We replace the input request (i.e., the steps in the conventional OIDC block of Fig. 5) with the user consent (i.e., those in the cellular-based OIDC block). It can let the user click a consent button or trigger any other confirmation action, but can be also skipped based on the user's trust on the MEC application. At Step 5, the OIDC module authenticates the user based on the confirmation packet's source IP, which is assured to be authentic by the access control module. Based on the user credential corresponding to the IP, it checks whether the user subscribes to the application or not. If the subscription exists, the OIDC module generates access and ID tokens, and associates the user's application data and cellular attributes with the access token. Otherwise, it rejects the authentication request from the application.

B. Bitmap-Based Authorization and Accounting

We design a bitmap-based approach to carry out authorization policies at various MEC components, because bitmaps are lightweight to support not only the user diversity of different MEC subscription sets, but also the propagation of authorization polices towards multiple MEC components. Each MEC UE is associated with one bitmap where each bit represents the subscription status of one MEC application. A propagation message of the UE's authorization policy only needs to include the identifier (i.e., IP) and the bitmap in several bytes.[5] Such bitmap-based approach also benefits the operations of different components. They can rely on bitwise operators to check the UE's authorized applications on the bitmaps at run time and then act accordingly to prevent unauthorized access.

Each UE's authorization bitmap is one of its cellular attributes stored at the HSS to represent its MEC subscription, whereas it is maintained by the authorization module for its runtime use. The carrier updates it based on the UE's subscription, whereas the authorization module obtains it during the MEC service activation and allows an anomaly detection module to dynamically update it for de-authorization. Whenever any (de)authorization update happens at the authorization module, it just needs to send few bytes to overwrite old bitmaps at other components.

The authorization module maintains each application's profile and associates it with each bit. The profile includes its domain name, the local IP of its server at the MEC, and the service port. The DNS server requires the first two items to answer the queries of authorized applications, whereas the access control module based on the last two items permits only authorized traffic to reach the MEC platform. At these components, each application's profile information is linked to the location of its represented bit on the bitmap so that it can be accessed without any search once the concerned bit is identified. It empowers the authorization module to commit UEs' authorization polices only based on bitmaps. Note that the update of the application profiles should be sent to all the above components, but they do not change frequently.

The accounting module keeps the bitmap of each active MEC UE, and allocates a space that associates with it to record the UE's usage statistics. The statistics can be in the units of bits, bytes, or/and seconds (here, bytes are used). It generates a data record, which is similar to 3GPP charging data record (CDR) [38], for each user periodically. The data record contains the UE's bitmap and the statistics associated with each bit. When a UE's bitmap changes, the module generates a data record with the old bitmap for the UE, and then starts a new record for the new bitmap. Otherwise, some bits that represent unauthorized applications in the new bitmap may still have usage statistics due to the old authorization policy. The authorization policy/bitmap and its associated usage statistics should be made consistent, since the data records are collected for charging and anomaly detection.

C. Two-Tier Hash-Based Access Control

There are two major goals for the access control. One is to enforce secure binding between the source IP and the uplink TEID for the GTP-U traffic, whereas the other is to ensure authorized access of the MEC platform. It should steer valid MEC packets to the MEC, as well as forward GTP-C and non-spoofing Internet packets to the EPC. Meanwhile, it needs to filter out illegal packets, and identify which attacks (i.e., IP spoofing and unauthorized MEC access) they belong to and record the attack information. A naive approach is to enumerate all the permitted entries with various valid combinations of IPs and TEIDs in the access control filter. However, it can result in too much overhead that includes a large size of permission entries, huge search space, and the analysis of illegal packets on attack types.

[5]The message has only 6 bytes with 4-byte IP and 2-byte bitmap, which can support 16 MEC applications, in the implementation.

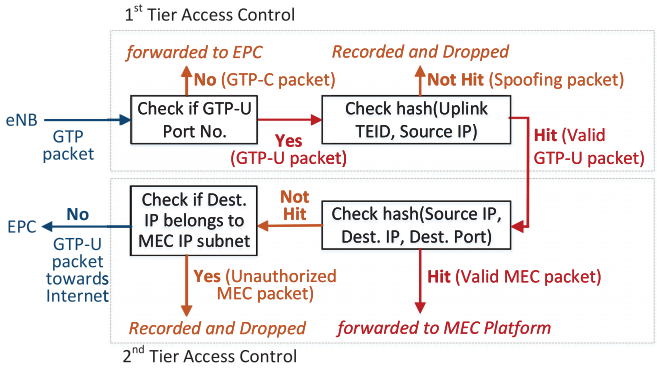

Fig. 6. The overall operation of the two-tier hash-based access control.

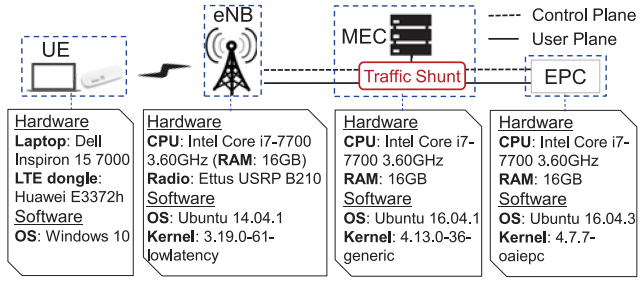

Fig. 7. The prototype of the MEC-integrated LTE network and the software/hardware information of its components.

To this end, we design a novel two-tier hash-based approach for the access control with two major merits. First, it divides the process into two tiers from the verification of GTP-U packets to that of authorized MEC packets. It can thus reduce the size of permission entries and identify the attack type of illegal packets from each tier. Second, it uses hash functions to achieve $O(1)$ search time for packet filtering. As shown in Fig. 6, the first tier forwards GTP-C packets and valid GTP-U ones to the EPC and the second tier respectively, while filtering out IP spoofing packets. The second tier steers the GTP-U packets towards the Internet and valid MEC packets to the EPC and the MEC respectively, while filtering off unauthorized MEC packets. Note that the access control focuses on only the uplink traffic from UEs.

The first-tier access control first differentiates GTP-C and GTP-U packets based on different port numbers of the UDP sessions that transport them on the S1 interface (they are ports 2123 and 2152, respectively) [9]. The GTP-C packets are directly forwarded to the EPC, whereas the remaining GTP-U ones are checked on the hash values of the concatenation of the uplink TEID and the source IP address. If the values of the indexes that the hash values are used as in the first-tier hash table are positive, they are valid GTP-U packets; otherwise, they are spoofing ones. Note that an index-value entry is added to the hash table, when the MEC manager detects a new cellular UE and notifies the access control module of the UE's uplink TEID and IP address. An entry is removed when the MEC manager detects a leaving cellular UE based on the `End Marker`. This secure binding against spoofing packets considers all the cellular UEs, but not only MEC ones.

The second-tier access control first checks the valid GTP-U packets for their hash values of the concatenation of the source/destination IP addresses and the destination port number. If the values of their indexes in the second-tier hash table are positive, they are valid MEC packets and forwarded to the MEC platform; otherwise, they can be Internet packets or unauthorized MEC ones. They can be further differentiated based on whether their destination IP addresses belong to the MEC IP subnet or not, since the attempt destination of the unauthorized MEC ones is the MEC platform. Note that an index-value entry for each authorized traffic flow is added based on the bitmap from the authorization module. The entries of one MEC UE's authorized flows are created based on its IP address and the IP/port pairs that associate with bit 1 s (i.e., authorized applications) in the bitmap.

D. Complexity Analysis

We analyze time complexity on the above three functions by assuming that there are k UEs attaching to the network, m control-plane packets, and n user-plane packets at a time. First, the MEC service activation consists of two steps: detection and initialization. The detection is based on the monitoring of control-plane or data-plane packets. Each packet requires only $O(1)$ processing time, so the complexity is $O(m)$ or $O(n)$. The initialization is done for each UE which is attaching to the network, so its complexity is $O(k)$ when it takes $O(1)$ for each UE. Second, the cellular-based OIDC authentication is needed whenever a UE needs to authenticate with an MEC application (assume q requests at a time). Each authentication process takes $O(1)$ time, so the complexity is $O(q)$. Third, the user-plane traffic protection based on two-tier access control takes $O(1)$ time for each user-plane packet with the hash-based approach, so its complexity is $O(n)$.

VI. Prototype and Implementation

We implement the MECsec design on an MEC-integrated LTE network prototype [5], [39], which is built based on the OAI cellular platform [6]. Fig. 7 shows the prototype and its software/hardware details. It includes three PCs with the same hardware for the eNB, the MEC platform plus the traffic shunt, and the EPC. The eNB's radio hardware is Ettus USRP B210, and the UE is a laptop equipped with an LTE dongle, Huawei E3372 h. For the MEC prototype, we develop a traffic dispatcher at the traffic shunt. It is a `Python` program that forwards uplink GTP packets to the EPC or the MEC, but skips downlink packets. Since the GTP packets are originally forwarded between the eNB and the EPC, we redirect them to the dispatcher using `Proxy ARP` and `iptables`. The former advertises the MEC's MAC address to both the eNB and the EPC, so they consider that the next hops are the MEC. The latter routes GTP packets to the dispatcher. The dispatcher forwards the MEC's GTP packets to its GTP tunnel handling module. We implement this module using the `OsmoGGSN` open source project [40]. It creates a GTP

TABLE III
EVENTS OF THE MEC SERVICE ACTIVATION FOR AN MEC UE

Event	Description
Attach	The MEC UE successfully attaches to the EPC through the eNB next to the MEC platform.
E1	The MEC manager detects the MEC UE from the inspection of control or user plane.
E2	The MEC manager obtains the cellular credential of the MEC UE from the MEC support module.
E3	The cellular-based OIDC module fetches the UE's some application data from the Internet, and the activation completes.

virtual interface to handle GTP encapsulation and decapsulation for all the application and DNS servers.

For the MECsec implementation, we use `Python` to carry out the MEC control plane including the MEC manager, the cellular-based OIDC, and the authorization/accounting modules, as shown in Fig. 4. The MEC support at the EPC, which can communicate with the MEC manager and the HSS, is implemented in `C`. Specifically, we develop the two-tier hash-based access control using `Python` and its hash function based on the `Python Dictionary`. We develop the cellular-based OIDC module using the `pyoidc` library [41], which is a certified OIDC library in `Python`, and customize a DNS server to operate based on the authorization bitmap. For the experiments on the MECsec, we generate computing loads and network traffic loads using two tools, `stress-ng` and `GTP-generator`, respectively. We use `GTP-generator` to emulate a large number of GTP packets from the eNB to the EPC without connecting many UEs to the network.

VII. EVALUATION

We evaluate the MECsec design in two ways. First, we consider the existing security mechanisms for comparisons if there are any. Specifically, we compare the cellular-based OIDC authentication with two possible MEC authentication methods, password-based and OIDC, in terms of the authentication delay. Second, we validate the other components' security performance by randomly generating malicious and benign traffic patterns, and also examine their overheads on the LTE network testbed. Note that current MEC security studies [32], [33] are not standard-compliant and cannot be used for the LTE testbed. In the following, we evaluate the MEC service activation, the cellular-based OIDC authentication, and the user-plane traffic protection.

A. MEC Service Activation

We examine the delays of the MEC service activation, which follows the detection of each MEC UE. We implement both control-plane and user-plane detection methods, which are described in Section V-A, to show their effectiveness and compare their performance. We consider the delays of three events involved in the service activation from the beginning of the UE's attach procedure: a new MEC UE is detected (Event 1); the UE's cellular credential is obtained (Event 2); the UE's application data are prefetched and the activation completes (Event 3). They are summarized in Table III, together with the event that the

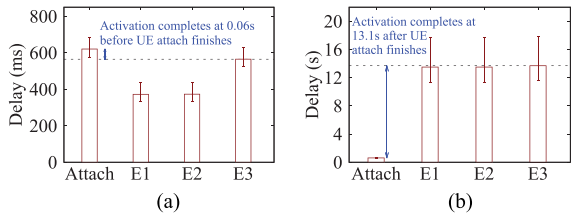

Fig. 8. Min/med/max delays of the MEC service activation, which are time periods between the time that the attach request is sent by the UE and the completion time of each event. Table III shows the description of each event. (a) Inspection of control plane. (b) Inspection of user plane.

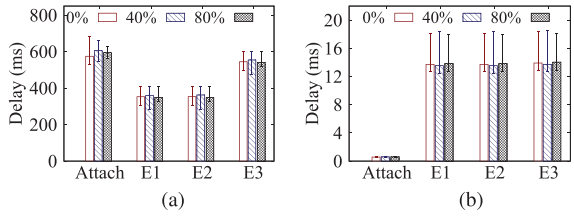

Fig. 9. Min/med/max delays of the MEC service activation vary with the CPU loading. (a) Inspection of control plane. (b) Inspection of user plane.

attach procedure completes. Note that the initialization also involves the transit of the authorization bitmap and the setup of other modules based on the bitmap, but these operations happening at the edge are always done earlier than Event 3.

Figs. 8(a) and 8(b) show the delays of different events for the MEC service activation based on the control-plane and user-plane methods, respectively. With the control-plane method, the median delays of those three events are 372.7 ms, 373.4 ms, and 565.4 ms, respectively. The service activation (E3) completes earlier than the finish time of the attach procedure, 620.9 ms. The reason is that it starts right after the GTP-C packet, `UE Context Setup Request`, is inspected, but there are still many attach steps following the delivery of the packet. E2 proceeds only at the MEC, whereas E3 requires the Internet access. So, E2 takes much shorter time than E3. With the user-plane method, the delays are much longer than those of the control-plane method, since the service activation cannot start until initial data packets are observed. The events take more than 13.5 s at the median. The big gap between the completion of the attach procedure and the delivery of initial packets lies in that the UE spends time on its network/system initialization after getting network settings from the attach procedure.

We next vary the CPU loading at the MEC and examine the effect of computing resource on the delays. As shown in Fig. 9, we observe the same trends for both control-plane and user-plane methods. Since the service activation is very lightweight, the delays do not increase obviously. As the analysis in Section V-D, more connecting UEs should require more resources, thereby possibly affecting the delays, but the software-defined radio platform cannot accommodate too many UEs for the test. However, to guarantee the delay performance, we can secure sufficient resource for this activation function with a virtual machine at the MEC.

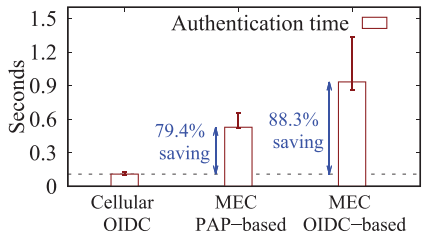

Fig. 10. Min/med/max delays of the user authentication for the MEC application vary with three methods.

B. User Authentication Based on Cellular-Based OIDC

We implement a UE application that can do user authentication with an MEC application server using different methods. We examine the viability of the cellular-based OIDC service, and compare it with another two MEC authentication methods. One is the PAP (Password Authentication Protocol) based authentication where we install an authentication server in the Google cloud and enable user authentication for the UE application based on it. The other is the conventional OIDC-based authentication where we empower the MEC application server to work with the OIDC service offered by Facebook. We have 20 runs for each scenario and plot the min/med/max delays in Fig. 10. We note two things for this experiment. First, the delays do not include the time of typing user credentials, which is not required by the cellular-based OIDC method but is needed by conventional authentication methods. The cellular-based OIDC method does authentication directly based on the cellular security context. Second, we always give correct user credentials and stable network connections for the tests.

We validate that the cellular-based OIDC service functions as expected that the UE always successfully does user authentication with the MEC application server. The server then obtains the UE's cellular credential (e.g., phone number) and uses it to identify the user's application data locally. Since the OIDC service is implemented based on the well-tested Python OIDC library, it can always give successful authentication without other user or network factors (e.g., incorrect user credentials and unstable network connections).

For the delay performance, we observe that the cellular-based OIDC service can reduce median delays of the other two authentication methods by 79.4% and 88.3%, respectively. Specifically, it takes only 0.11 s, since all the authentication handshakes happen at the edge. The others require handshakes with the servers on the Internet, so they need 0.53 s and 0.93 s, respectively. The delay variance of the cellular-based OIDC is much smaller than the others. Moreover, it can gain more when the input time of username/password is considered. For example, given that the input time required for the input is 5 seconds, it can save more than 98.2% time than the others in this case. It can also give user convenience for automatic login without user intervention.

C. User-Plane Traffic Protection

We generate data traffic to examine the performance of the authorization/accounting modules and the two-tier hash-based access control. We evaluate the accuracy of their security operations, and then assess their overheads on the packet forwarding bandwidth and delay.

TABLE IV
THE TRAFFIC CASES THAT MAY HAPPEN FOR BENIGN (B) AND MALICIOUS (M) MEC/NON-MEC UES

Case	Traffic Type	MEC UE		non-MEC UEs	
		B	M	B	M
C1	DNS traffic [Setting] Destination IP/port: 8.8.8.8/80	V		V	
C2	Internet traffic [Setting] Destination IP/port: public servers	V		V	
C3	MEC traffic [Setting] Destination IP/port: MEC servers	V			V
C4	IP spoofing [Setting] Source IP: another UEs' IPs		V		V
C5	Unauthorized access of MEC services [Setting] Destination IP/port: MEC local servers that are not authorized to the UE		V		

Security Accuracy: We emulate a scenario that 10 MEC UEs and 10 non-MEC UEs are connecting to the LTE platform, and the MEC UEs have subscribed to different sets of MEC applications, the total number of which is 10. We assign an IP address and a pair of TEIDs to each of those 20 UEs, and randomly choose a subscription set of MEC applications, which we keep based on a 16-bit bitmap, for each MEC UE. We assign a pair of local MEC IP address and port number to each MEC application server. We emulate the connection of each UE by letting the MEC manager detect the UE's control-plane message. For each UE, the security binding information is passed to the access control module. The authorization bitmaps of the MEC UEs are also committed to the modules of access control, authorization, and accounting.

We further emulate the UEs' traffic by randomly generating GTP traffic from the eNB towards the MEC or the EPC. To validate the security accuracy, we assume that some of the UEs may do IP spoofing or unauthorized access of MEC application servers. We consider 9 difference traffic cases shown in Table IV. Specifically, both MEC and non-MEC UEs can have DNS and Internet traffic, but they are considered to be malicious when generating IP spoofing packets. MEC UEs can have MEC traffic, but non-MEC UEs who generate MEC traffic are malicious. Malicious MEC UEs might also send traffic towards MEC application servers which are not authorized to them. In the emulation, we generate 10000 GTP packets from the eNB. To generate a packet, we first select one case randomly from those 9 cases, and then fill in the packet with the randomly generated information according to the case. For example, given the IP spoofing case for a malicious MEC UE, we randomly select two MEC UEs, malicious and spoofed ones, and then fill the packet information with the former's TEIDs and the latter's IP.

We log the number of packets generated for each case at the eNB, and verify packet statistics of different cases from the accounting module, two tiers of the access control module, and the EPC. Fig. 11 compares these two sets of traffic statistics: generated and verified ones. We observe that for each case, the numbers of generated and verified packets are the same; that is, the detection rate is 100%. It shows that the access

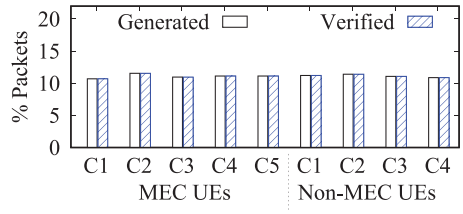

Fig. 11. Security accuracy of user-plane traffic protection is evaluated based on whether the randomly generated packets of various cases can be correctly identified. Table IV describes those cases.

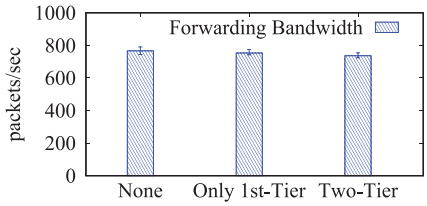

Fig. 12. Min/avg/max forwarding bandwidths for three access control modes.

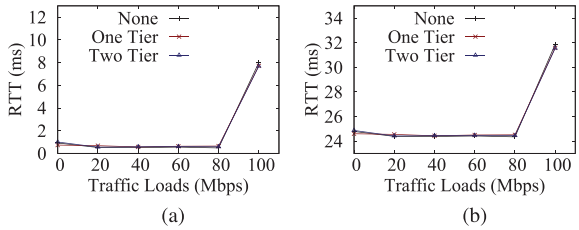

Fig. 13. Average RTTs vary with traffic loads for the user-plane traffic. (a) RTT between eNB and MEC. (b) RTT between UE and MEC.

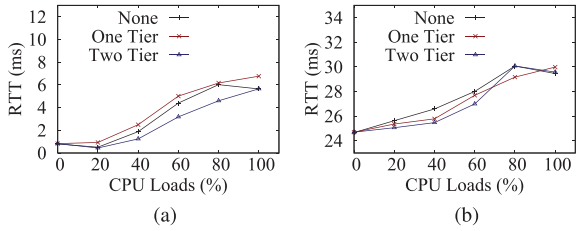

Fig. 14. Average RTTs vary with CPU loads for the user-plane traffic. (a) RTT between eNB and MEC. (b) RTT between UE and MEC.

control can successfully filter out malicious packets based on the authorization maps prepared by the authorization module, and the accounting module can maintain accurate traffic statistics.

Overhead Assessment: We next examine how much overhead the access control can impose on the forwarding bandwidth and the service delay. For the forwarding bandwidth, we generate a large amount of GTP packets (i.e., 100 per millisecond) from the eNB towards the traffic shunt for stress testing. The GTP packets do not have IP payloads but only GTP and IP headers, since we aim to test the processing overhead by skipping network bandwidth constraints. We have 5 runs for each case and each run takes 2 minutes. Fig. 12 shows minimum, average, and maximum forwarding bandwidths for three access control modes: none of access control, only the first tier, and two-tier. We observe that the first tier and two-tier access control modes reduce the average bandwidth of the normal mode by only 1.74% and 3.93%, respectively. It can be attributed to the hash-based approach with low overhead.

We investigate how much overhead the access control can impose on the service delay, where the round trip times (RTTs) of the packets sent between the UE and the MEC are considered. We measure one RTT every 100 ms by sending one packet from the UE to the MEC and then receiving its response at the UE. In the experiment, we generate various traffic loads from the eNB to the EPC to observe the changes of the delays. For each case, we run 3 times with 2 minutes each and take the average value over the collected RTTs. Fig. 13 shows the average RTTs varying with traffic loads. Since the delay variances of wireless transmissions between the UE and the eNB are much larger than the delays of wired networks, we first exclude the wireless delays in Fig. 13(a), and then consider them as a constant that was the average over all the cases in Fig. 13(b). We observe that the access control mechanism has negligible increases on the RTTs of the normal mode. Specifically, it increases the RTTs by only up to 0.06% and 1.63% at 80 Mbps and 100 Mbps, respectively.

We next vary the CPU loading at the MEC to examine the RTTs varying with unavailable computing resources. As shown in Fig. 14, the RTTs gradually increase with the loading from 20% to 100%. We note two things. First, the differences between the delays from those three cases do not exceed 1.5 ms, which is in the error range caused by platform dynamics, so their delays are comparable. Second, when the loading is 100%, the delays do not dramatically increase because the access control module with relatively less computing load can gain higher priority to be scheduled than the `stress-ng` program, which generates the dummy CPU loading.

VIII. Discussion

In this section, we discuss four major issues for the MEC platform and the MECsec design.

Concerns of Cellular Security Violation: Although the MEC platform is enabled to steer and inspect cellular traffic without awareness of the existing cellular infrastructure, it does not violate the cellular security from three aspects: technology, standard, and deployment. First, from the technology aspect, the cellular network has two major security mechanisms, non-access stratum (NAS) security and access stratum (AS) security, which protect the control-plane messages between the UE and the MME, and the wireless traffic between the UE and the eNB, respectively. They do not prevent the MEC platform from steering or inspecting cellular traffic. Second, the ETSI standard [34] also considers the concept of the traffic redirection/inspection as one deployment solution of the MEC platform. Third, when the MEC platform is deployed by the carrier itself, the deployment is like the case that the carrier installs an additional network

device to do traffic engineering and provide the MEC service in its own network.

Tradeoffs of the MECsec Design: The transparent design does not come without cost. It needs to have a graceful interaction with cellular protocols. Compared with normal security functions, it requires extra processing overhead to handle the protocols, e.g., the traffic engineering on GTP packets and the traffic monitoring for GTP session maintenance. Moreover, once the protocols have any changes, the design needs to be modified accordingly. However, the basic cellular protocols and operations are rarely changed. Even for the future 5 G, the underlying GTP protocol still remains.

Securing MEC in 5 G Networks: The MECsec design can be easily applied to future 5 G networks, since it is transparent to the cellular network architecture. It requires only upgrade of the MEC support module for the 5 G core network, whereas the other parts of the design can remain unchanged.

Concerns of Energy Consumption: Energy consumption is a key factor for the MEC end devices with battery constraints. However, the MECsec solution is transparent to end devices without changing their operations but only giving low-latency response to their services, so there is little impact on their energy consumption. As for the MEC platform deployed in the cellular infrastructure and with power supply, its energy consumption is not an important concern.

IX. Conclusion

The MEC has been deemed as an essential 5 G feature, and is being developed in full swing. Carriers are anticipated to roll out the MEC deployment in the near future, but its security can be a major concern. It lies in the requirement that a new component, the MEC platform, has to be inserted into the cellular network, which is a closed system with high security assurance. To examine its security impact, we analyze AAA and access control security functions on the MEC-integrated cellular architecture. The analysis shows that conventional security functions may offset the low-latency benefits of the MEC and expose the MEC or the cellular network to security threats from malicious UEs. We thus propose and prototype a security design, MECsec, to protect the MEC and the cellular network while keeping low latency for the MEC service. Its transparent design not only eases the deployment in 4 G networks by making existing cellular operations remain unchanged, but can be also easily applied to future 5 G networks. We hope that MECsec can secure and facilitate upcoming deployment of the MEC technology in 4 G and 5 G networks.

References

[1] ETSI, "Mobile edge computing: A key technology towards 5G," ETSI White Paper No. 11, 2015.

[2] 3GPP, *Technical Specification Group Services and System Aspects; System Architectuer for the 5G Systems: Stage 2 (Release 15)*, 3GPP Standard TS23.501 V0.4.0, 2017.

[3] Grand View Research Inc., *Edge Computing Market Worth $3.24 Billion By 2025*, 2018.

[4] "OpenID Connect: A Simple Identity Layer on top of the OAuth 2.0 Protocol," 2019. [Online]. Available: https://openid.net/connect

[5] C.-Y. Li *et al.*, "Mobile edge computing platform deployment in 4 G LTE networks: A middlebox approach," in *Proc. USENIX Workshop Hot Topics Edge Comput.*, Boston, MA, USA, Jul. 2018, pp. 1–6.

[6] "OAI (OpenAirInterface): A platform with open source software for the core network, access network and user equipment of 3GPP cellular networks," 2018. [Online]. Available: https://www.openairinterface.org

[7] 3GPP, *TS23.002: Network Arhcitecture*, 3GPP Standard TS23.002 V14.1.0, 2017.

[8] 3GPP, *TS36.413: Evolved Universal Terrestrial Radio Access Network (E-UTRAN); S1 Application Protocol (S1AP)*, 3GPP Standard TS36.413 V14.1.0, 2017.

[9] 3GPP, *TS29.281: General Packet Radio System (GPRS) Tunnelling Protocol User Plane (GTPv1-U)*, 3GPP Standard TS29.281 V14.1.0, 2017.

[10] 3GPP, *TS29.060: General Packet Radio Service (GPRS); GPRS Tunneling Protocol (GTP) Across the Gn and Gp Interface*, 3GPP Standard TS29.060 V14.1.0, 2016.

[11] S. R. Hussain, O. Chowdhury, S. Mehnaz, and E. Bertino, "LTEInspector: A systematic approach for adversarial testing of 4G LTE," in *Proc. Netw. Distrib. Syst. Secur. Symp.*, San Diego, CA, USA, Feb. 2018, pp. 1–15.

[12] D. Basin, J. Dreier, L. Hirschi, S. Radomirovic, R. Sasse, and V. Stettler, "A formal analysis of 5G authentication," in *Proc. ACM SIGSAC Conf. Comput. Commun. Secur.*, Toronto, Canada, Oct. 2018, pp. 1383–396.

[13] A. Dabrowski, N. Pianta, T. Klepp, M. Mulazzani, and E. Weippl, "IMSI-catch me if you can: IMSI-catcher-catchers," in *Proc. ACM 30th Annu. Comput. Secur. Appl. Conf.*, New Orleans, Louisiana, USA, Dec. 2014, pp. 246–255.

[14] A. Shaik, R. Borgaonkar, N. Asokan, V. Niemi, and J.-P. Seifert, "Practical attacks against privacy and availability in 4 G/LTE mobile communication systems," in *Proc. Netw. Distrib. Syst. Secur. Symp.*, San Diego, CA, USA, Feb. 2016, pp. 1–15.

[15] R. P. Jover, "Security attacks against the availability of LTE mobility networks: Overview and research directions," in *Proc. IEEE 16th Int. Symp. Wireless Pers. Multimedia Commun.*, Atlantic City, NJ, USA, Jun. 2013, pp. 1–9.

[16] T. Xie, G.-H. Tu, C.-Y. Li, C. Peng, J. Li, and M. Zhang, "The dark side of operational Wi-Fi calling services," in *Proc. IEEE Conf. Commun. Netw. Secur.*, Beijing, China, May 2018, pp. 1–9.

[17] J. Baek *et al.*, "Wi not calling: Practical privacy and availability Attacks in Wi-Fi calling," in *Proc. ACM 34th Annu. Comput. Secur. Appl. Conf.*, San Juan, PR, USA, Dec. 2018, pp. 1–11.

[18] C.-Y. Li *et al.*, "Insecurity of voice solution VoLTE in LTE mobile networks," in *Proc. ACM Conf. Comput. Commun. Secur.*, Denver, Colorado, USA, Oct. 2015, pp. 316–327.

[19] G.-H. Tu, C.-Y. Li, C. Peng, Y. Li, and S. Lu, "New security threats caused by IMS-based SMS service in 4G LTE networks," in *Proc. ACM Conf. Comput. Commun. Secur.*, Vienna, Austria, Oct. 2016, pp. 1118–1130.

[20] C. Peng, C.-Y. Li, H. Wang, G. Tu, and S. Lu, "Real threats to your data bills: Security loopholes and defense in mobile data charging," in *Proc. ACM Conf. Comput. Commun. Secur.*, Scottsdale, Arizona, USA, Nov. 2014, pp. 727–738.

[21] D. He, S. Chan, and M. Guizani, "Security in the internet of things supported by mobile edge computing," *IEEE Commun. Mag.*, vol. 56, no. 8, pp. 56–61, Aug. 2018.

[22] J. Zhang, B. Chen, Y. Zhao, X. Cheng, and F. Hu, "Data security and privacy-preserving in edge computing paradigm: Survey and open issues," *IEEE Access*, vol. 6, pp. 18209–18237, 2018.

[23] A. Ahmed and E. Ahmed, "A survey on mobile edge computing," in *Proc. IEEE 10th Int. Conf. Intell. Syst. Control*, Coimbatore, India, Jan. 2016, pp. 1–8.

[24] T. Taleb, K. Samdanis, B. Mada, H. Flinck, S. Dutta, and D. Sabella, "On multi-access edge computing: A survey of the emerging 5G network edge cloud architecture and orchestration," *IEEE Commun. Surv. Tut.*, vol. 19, no. 3, pp. 1657–1681, Jul.–Sep. 2017.

[25] P. Mach and Z. Becvar, "Mobile edge computing: A survey on architecture and computation offloading," *IEEE Commun. Surv. Tut.*, vol. 19, no. 3, pp. 1628–1656, Jul.–Sep. 2018.

[26] Y. Mao, C. You, J. Zhang, K. Huang, and K. B. Letaief, "A survey on mobile edge computing: The communication perspective," *IEEE Commun. Surv. Tut.*, vol. 19, no. 4, pp. 2322–2358, Oct.–Dec. 2017.

[27] J. Poderys, M. Artuso, C. M. O. Lensbol, H. L. Christiansen, and J. Soler, "Caching at the mobile edge: A practical implementation," *IEEE Access*, vol. 6, pp. 8630–8637, 2018.

[28] H. Guo and J. Liu, "Collaborative computation offloading for multiaccess edge computing over fiber-wireless networks," *IEEE Trans. Veh. Technol.*, vol. 67, no. 5, pp. 4514–4526, May 2018.

[29] T. X. Tran and D. Pompili, "Joint task offloading and resource allocation for multi-server mobile-edge computing networks," *IEEE Trans. Veh. Technol.*, vol. 68, no. 1, pp. 856–868, Jan. 2019.
[30] H. Guo, J. Zhang, J. Liu, and H. Zhang, "Energy-aware computation offloading and transmit power allocation in ultradense IoT networks," *IEEE Internet Things J.*, vol. 6, no. 3, pp. 4317–4329, Jun. 2019.
[31] C. Wang, F. R. Yu, C. Liang, Q. Chen, and L. Tang, "Joint computation offloading and interference management in wireless cellular networks with mobile edge computing," *IEEE Trans. Veh. Technol.*, vol. 66, no. 8, pp. 7432–7445, Aug. 2017.
[32] M. Amadeo, C. Campolo, A. Molinaro, C. Rottondi, and G. Verticale, "Securing the mobile edge through named data networking," in *Proc. IEEE 4th World Forum Internet Things*, Singapore, Singapore, Feb. 2018, pp. 80–85.
[33] S. Wong, N. Sastry, O. Holland, V. Friderikos, M. Dohler, and H. Aghvami, "Virtualized authentication, authorization and accounting (V-AAA) in 5G networks," in *Proc. IEEE Conf. Standards Commun. Netw.*, Helsinki, Finland, Sep. 2017, pp. 175–180.
[34] ETSI, "MEC deployment in 4G and evolution towards 5G," ETSI White Paper No. 24, 2018.
[35] 3GPP, *TS29.303: Domain Name System Procedures; Stage 3*, 3GPP Standard TS29.303 V14.1.0, 2016.
[36] "Chrome DevTools: A set of web developer tools built directly into the Google Chrome browser," 2019. [Online]. Available: https://developers.google.com/web/tools/chrome-devtools/
[37] 3GPP, *TS29.212: Policy and Charging Control (PCC); Reference Points*, 3GPP Standard TS29.212 V14.1.0, 2016.
[38] 3GPP, *TS32.297: Telecommunication Management; Charging Management; Charging Data Record (CDR) File Format and Transfer*, 3GPP Standard TS32.297 V14.0.0, 2017.
[39] NEMS Lab, NCTU. "MEC middlebox solution," 2019. [Online]. Available: http://nems.cs.nctu.edu.tw/release/
[40] "OsmoGGSN is an open source implementation of a GGSN (Gateway GPRS Support Node)," 2019. [Online]. Available: https://osmocom.org/projects/openggsn
[41] "Pyoidc: A complete OIDC library that can be used to build OIDC OPs or RPs," 2019. [Online]. Available: https://openid.net/developers/certified/

Chi-Yu Li received bachelor's and master's degrees from National Chiao Tung University (NCTU), Hsinchu, Taiwan, and the Ph.D. degree in computer science from the University of California, Los Angeles (UCLA), Los Angeles, CA, USA, in 2015. He is currently an Assistant Professor with the Department of Computer Science, NCTU. His research interests include wireless networking, mobile networks and systems, and network security. He was the recipient of the Award of MTK Young Chair Professor in 2016, MOST Young Scholar Research Award for 2017–2020, and Best Paper Award in IEEE Conference on Communications and Network Security in 2018.

Ying-Dar Lin (Fellow, IEEE) received the Ph.D. degree in computer science from the University of California, Los Angeles, Los Angeles, CA, USA, in 1993. He is currently a Distinguished Professor with the Department of Computer Science, National Chiao Tung University, Hsinchu, Taiwan. He was a Visiting Scholar with Cisco Systems in San Jose during 2007–2008, the CEO of Telecom Technology Center, Taiwan, during 2010–2011, and the Vice-President of National Applied Research Laboratories, Taipei, Taiwan, during 2017–2018. Since 2002, he has been the Founder and Director of Network Benchmarking Lab, which reviews network products with real-traffic and automated tools, and has been an approved test lab of the Open Networking Foundation since July 2014. He also cofounded L7 Networks Inc. in 2002, later acquired by D-Link Corporation, and OPrueba, Inc., in 2018. His research interests include network security, wireless communications, and network softwarization. He was the recipient of the 2017 Research Excellence Award and K. T. Li Breakthrough Award.

Yuan-Cheng Lai received the Ph.D. degree with the Department of Computer and Information Science from the National Chiao Tung University, Hsinchu, Taiwan, in 1997. He joined the Faculty of the Department of Information Management, National Taiwan University of Science and Technology in August 2001 and has been a distinguished professor since June 2012. His research interests include performance analysis, software-defined networking, wireless networks, and IoT security.

Hsu-Tung Chien received the M.S. and Ph.D. degrees in computer science from the National Chiao Tung University, Hsinchu, Taiwan, in 2017 and 2019, respectively. His research interests include wireless networks, mobile networks, and protocol designs. He participated in several H2020 projects including 5G-CORAL, Crosshaul, and Transformer. They aim to respectively develop an integrated edge/fog system, fronthaul/backhaul networks, and a vertical slicer for 5G networks.

Yu-Sheng Huang is currently working toward the master's degree with the Department of Computer Science, National Chiao Tung University, Hsinchu, Taiwan. His research interests include mobile networks and systems.

Po-Hao Huang received the B.S. and M.S. degrees in computer science from the National Chiao Tung University, Hsinchu, Taiwan, in 2015 and 2019, respectively. His research interests include mobile networks and systems.

Hsueh-Yang Liu received the B.S. degree in computer science from Chung Yuan Christian University, Taoyuan City, Taiwan, in 2016, and the M.S. degree in computer science from the National Chiao Tung University, Hsinchu, Taiwan, in 2018. His research interests include mobile networks and systems.