vEPC-sec: Securing LTE Network Functions Virtualization on Public Cloud

Preprint · November 2018

DOI: 10.13140/RG.2.2.32623.48800

2 authors:

Muhammad Taqi Raza
University of California, Los Angeles
24 PUBLICATIONS **161** CITATIONS
SEE PROFILE

Songwu Lu
University of California, Los Angeles
189 PUBLICATIONS **15,270** CITATIONS
SEE PROFILE

vEPC-sec: Securing LTE Network Functions Virtualization on Public Cloud

Muhammad Taqi Raza and Songwu Lu [UCLA]

ABSTRACT

Public cloud offers economy of scale to adapt workload changes in an autonomic manner, maximizing the use of resources. Through Network Function Virtualization (NFV), network operators can move LTE core to the cloud; hence removing their dependency on carrier-grade LTE network functions. Recent research efforts discuss performance, latency, and fault tolerance of LTE NFV, largely ignoring the security aspects. In this paper, we discover new vulnerabilities that LTE NFV face today with no standard solutions to address them. These vulnerabilities span at both LTE control and user planes. To address them, we propose vEPC-sec that cryptographically secures LTE control-plane signaling messages in the cloud. It provides distributed key management and key derivation schemes to derive shared-symmetric keys for securing the communication between any two network functions. Our approach provides encryption and integrity protection to the messages even during virtual machines scalability and failure recovery scenarios. vEPC-sec also prevents user-plane vulnerabilities by ensuring that LTE routing modules should faithfully forward the LTE subscriber packets.

I. INTRODUCTION

LTE Network Function Virtualization (NFV) is a new trend that replaces carrier grade LTE core network functions with software running on commercial off-the-shelf servers in a cloud data center. On the one hand, NFV reduces operational and capital expenditure at traditional LTE network operators; on the other hand, it opens the cellular network business to small network operators. Network operators can take advantage of dynamic load balancing, the resource elasticity and scalability that the cloud offers. This is a popular trend where a number companies [1], [2], [3], [4] are offering multi-tenant LTE public cloud service following Amazon EC2 and Microsoft Azure style of business model. In a multi-tenant public cloud architecture, LTE network operators are cloud tenants that share compute, storage and network resources with each other. This fact has motivated us to study LTE core network security on public cloud.

In our study, we find that available solutions provide detailed security guidelines to cryptographically secure both LTE signaling messages and data packets over the radio network [5], [6], [7]. They do not discuss, however, secure communication inside the LTE core network. Up till now, every network operator has privately operated its LTE packet core, shielding the backend packets processing and messages exchange from the outside world. In the age of multi-tenant LTE public cloud, LTE core traffic – not ciphered and transported as "clear text" – provides the adversary an opportunity to inspect subscriber traffic and to inject malicious network traffic.

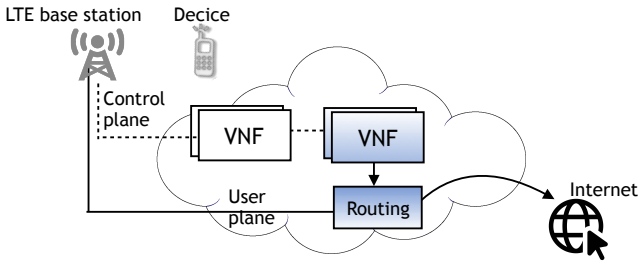

Fig. 1: LTE core NFs are moved to public cloud.

Cloud service providers provision a number of virtual machines to host LTE Network Functions (NFs). These Virtualized Network Functions (VNFs) from different network operators share the same physical infrastructure. They communicate with LTE radio network via two different channels of control and user planes, as shown in Figure 1. Although, cloud service providers logically isolate traffic from different tenants, they cannot guarantee that LTE VNF selection procedure always chains a VNF to the same tenant. This motivates an attacker to hijack VNF selection procedure to get associated with victim tenant's network. After that he gets control over the behavior of victim tenant's VNFs.

We outline that an adversary can bring four different types of vulnerabilities and can launch a number of attacks in LTE NFV. First, the attacker can lie about the status of one-hop away neighbor and tricks the victim VNF to delete all associated subscribers records. Second, it can sniff device master session key during device intra-system switch. As a result, the attacker can decrypt the encrypted packets and even can derive future session keys using the leaked master key. Third, the adversary can put memory pressure by simply sending one false LTE paging notification message. This tricks LTE VNF to reserve memory space for tens of hundreds of devices, and disrupts the memory resource allocation scheduling at victim VNF. Fourth, an adversary can inject fake IP packets into neighboring NF's user-plane module. This renders victim NF's data forwarding module to process fake IP packets that impacts the performance of other IP data packets flows.

To address these vulnerabilities, we put forward vEPC-sec. It is a solution that provides ciphering and integrity protection to LTE control-plane messages, and prevents fake IP packets injection into user-plane. It provides shared-symmetric keys to VNFs to cryptographically protect their control-plane messages in multi-tenant public cloud environment. vEPC-sec is designed to meet cloud requirements of scalability and fault tolerance. The VNF might have lost the shared symmetric-keys during failure recovery or scaling to a new instance. vEPC-sec detects such a scenario through messages exchange with the peer VNF.

It assigns a fresh shared-symmetric key to the recovered or scaled VNF. It also performs *key change on the fly* re-keying procedure with the VNFs whose keys need to be updated as well.

Our solution ensures that any attempt to inject fake IP packets should be detected and blocked. To achieve this, we add default packets forwarding policy as to 'drop' the packet. Further, through vEPC−sec, we can detect replaying of IP packets by malicious user-plane module that results in subscriber overbilling issue[8]. Our solution also identifies if data packets are illegally throttled at malicious forwarding module by delaying their delivery. To achieve these, our idea is to map radio data packets sent at LTE base station with the IP data packet received at LTE core. Because, the LTE core network forwards the same packet that it has received from the base station, any missing/duplicate number of packets can be detected.
Finally, through security analysis, we show that vEPC−sec can guard both LTE control and data planes over public cloud.

In summary, this paper makes the following contributions:

- We propose vEPC−sec that cryptographically secures LTE control-plane transmission at LTE core. The heart of the vEPC−sec is a distributed key management and key derivation scheme that functions even during VNF scalability and failure recovery scenarios.
- vEPC−sec puts user-plane traffic forwarding behavior in check. It ensures no fake IP packet is injected in forwarding plane, data packets are not forwarded twice, and user packets are not intentionally delayed by adversary forwarding module.
- We provide security analysis experiments and show that our solution addresses the discussed vulnerabilities as well as few others.
- vEPC−sec is a plug-and-play component with existing LTE protocols. It provides secrecy to LTE cloud tenants by requiring few lines of code changes.

To the best of our knowledge, this is the first attempt to provide LTE core network security on public cloud. vEPC−sec is a step forward to foster healthy competition among cellular network providers (both small and major operators) by taking away their worry of securing LTE core on public cloud.

II. LTE – NFV IN A NUTSHELL

LTE network consists of three main components: LTE device, LTE base station and LTE core, as shown in Figure 2. LTE NFV architecture virtualizes LTE core network functions over the cloud and eliminates reliance on vendor specific proprietary hardwares. Softwarization of LTE NFs accelerates the innovation by lowering operational and capital expenditures [9], [10]. LTE core (also known as Evolved Packet Core (EPC)) is composed of a number of Network Functions (NFs): the Serving Gateway (SGW), the PDN Gateway (PGW), the Mobility Management Entity (MME), the Home Subscriber Server (HSS), and a few others. These LTE EPC NFs (implemented as virtualized NF (VNFS) over cloud) handle control-plane and data-plane traffic through separate network interfaces and protocols. Cloud providers host EPC NFs on separate virtual machines (VMs) for scalability and flexibility purpose [11], [12], [13].

As shown in Figure 2, LTE control-plane traffic from radio network is sent to MME VNF. MME acts as a central management entity that authenticates and authorizes the device, handles device procedures (such as device registration, handover, location update, and service provisioning). It is also responsible of setting-up device data channel (i.e. data bearers) with SGW and PGW VNFs. In a virtualized environment, both SGW and PGW are divided into control-plane and user-plane modules. The control-plane modules are responsible of assigning IP address(es) for device and creating packet forwarding rules. These packet forwarding rules are sent to corresponding SGW and PGW user-plane modules that enforce the data packets forwarding policy for that device. Such decoupling of SGW and PGW into control and user planes is important for LTE data service performance that allows data packets to be forwarded without going through virtualization layer. This is also a common design approach in Software Defined Networking [14], [15].

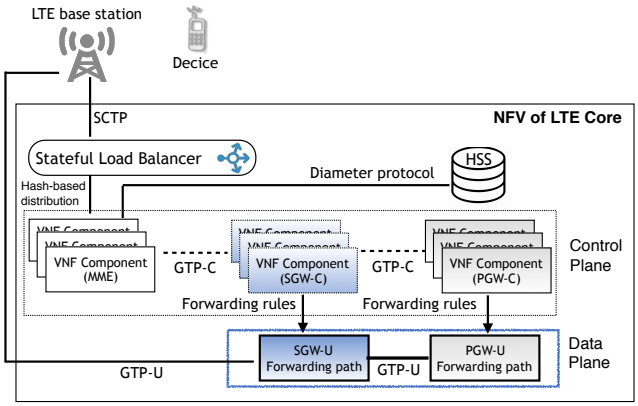

Fig. 2: LTE architecture over NFV: an overview.

VNFs selection in LTE: LTE network operators want that the appropriate EPC VNFs are selected to serve their subscribers according to device geographical area (known as tracking area in LTE), and type of radio network (macro/micro base station) it uses. To achieve this, network operators configure a number of EPC VNFs and create a pool of these VNFs. The best available VNFs – closer to the device and not heavily loaded – are selected to serve the subscriber device during its registration procedure with LTE network. This VNFs selection can be achieved either through stateful load balancer or through LTE standardized procedure [16]. In the first approach, stateful load balancer sends a query to VNF pool database and gets the IP addresses of MME, SGW and PGW VNFs to serve the subscriber. This is a standard cloud based approach implemented in today's public clouds. Examples include, Microsoft Azure's backend pool [17] and Amazon EC2 spot fleet [18]. In the second approach, configured LTE VNFs are registered at Domain Name Server (DNS). During device registration procedure MME VNF makes a DNS query to select best possible SGW and PGW VNFs instances to serve the subscriber. These DNS queries are made using UDP as transport protocol, as the standard states "DNS resolvers in EPC core network nodes shall support recursive queries and responses over UDP transport as specified in IETF RFC 1035" [16]. This selection of VNFs is vulnerable especially when query request/response are not cryptographically protected.

III. LTE SECURITY OVER PUBLIC CLOUD

LTE standard secures device communication with LTE base station, and EPC through symmetric keys. On receiving device registration request, MME contacts HSS and retrieves the device symmetric session key (known as K_{ASME} key). MME further derives separate ciphering and integrity keys to secure the device connection with radio network and LTE core [5]. MME secures its communication with LTE base station and HSS through secure SCTP and diameter protocols, respectively [19], [20]. However, SGW communication with MME, PGW and LTE base station is carried through unsecured IP/UDP based GPRS Tunneling protocol (GTPs). LTE system security [5] and LTE network domain security [7], [6] standards do not discuss securing GTP control and user plane protocols. In this paper, we first show new LTE vulnerabilities that unsecured GTP protocols bring in public cloud, and then provide a framework to secure GTP protocols communication.

Goals We aim to evaluate LTE-NFV security from two aspects: (1) identifying control-plane vulnerabilities that disrupt VNF operations, and (2) disclosing data forwarding policies misuses that impact subscribers data performance.

Threat model In our threat model, we consider a cloud service provider that hosts multiple LTE network operators (i.e. tenants). These tenants serve in a competitive LTE market where multiple tenants compete by providing LTE service in a similar geographical areas. To gain competitive edge, a malicious tenant has benefit to attack other tenants' LTE VNFs. The first step the malicious tenant takes is to trick victim tenant's VNF to get associated with one of malicious tenant's VNF. By doing so, it gets control over the behaviour of victim tenant. This is challenging, especially, when cloud service provider isolates traffic from different tenants through virtual LANs and/or source/destination addresses hash based forwarding. To solve this challenge, the malicious tenant exploits the fact that a VNFs selection query is made to select the best available VNFs during device registration procedure (§II). The malicious tenant can hijack the response of that query by replacing victim tenant's SGW IP address to one of its SGW IP address. He does not need to hijack every VNFs selection response, rather, hijacking one out of few thousands responses is sufficient. Further, malicious tenant can also control the number of VNFs selection requests. It can do so by first becoming the customer of victim tenant (by purchasing LTE service plan under victim tenant), and then sending a number of device registration requests to trigger SGW selection procedure at cloud.
The malicious SGW that associates itself with victim tenant network strictly follows LTE standard operations to avoid being detected through cloud intrusion detection box. The threats it can bring include: (a) sending wrong status information to MME regarding PGW, (b) sniffing unprotected GTP messages exchanged between source and destination MMEs, (d) putting memory pressure throug false paging notification message(s), and (d) injecting fake IP packets to impact the performance of other IP flows. We assume that victim tenant VNFs are not compromised and function according to LTE standard protocols.

IV. LTE – NFV VULNERABILITIES

A. Purging subscribers' context from MME

The malicious SGW can remove all subscribers' context from MME by sending PGW restart notification message. LTE EPC NFs employ a mechanism, known as *path management* [21], in which the availability of directly connected peer NFs can be determined for reliability purpose. A NF sends *echo request* message to its peer NF, and on receiving the *echo response* message it determines the reachability of peer NF. These periodic heart beat messages are also used to adjust the retry timer value for lost signaling messages. Once a NF is detected to be non-responding (i.e. no *echo response* message is received for certain number of tries), it is marked as failed. The failure indication is also sent to next hop NFs which are not directly connected with non-responding NF. We take an example of PGW failure. The PGW is directly connected to SGW, and its connection with MME goes through SGW. When SGW determines that the PGW has failed, it sends failure notification signaling message to MME (refer to section 7.9.5 PGW Restart Notification in TS 29.274[22], and 16.1A.2 PGW Failure in TS 23.007[23] for detailed procedure). On receiving the PGW failure notification, MME clears all those subscribers records which are served by failed PGW. MME then sends *Implicit Detach Request* message to all these subscriber devices. On receiving *Implicit Detach Request*, devices first locally deregister from LTE network and then re-initiate the registration procedure (i.e. Attach Request procedure). As new registration requests (i.e. *Attach Request* messages) from these subscriber devices arrive at EPC, a different PGW is selected (either by stateful load balancer or through DNS resolution).

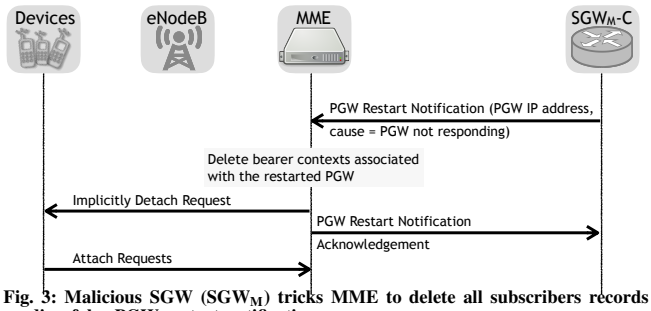

Fig. 3: Malicious SGW (SGW_M) tricks MME to delete all subscribers records by sending false PGW restart notification message.

Malicious SGW adopts LTE failure recovery procedure in its advantage. It sends *PGW Restart Notification* message to MME, as shown in Figure 3. On receiving *PGW Restart Notification* message, MME sends *Implicit Detach Request* message to all those subscribers which are connected to the reported PGW. Thereafter, all these subscribers will send *Attach Request* message to MME. MME will authenticate these devices and will assign them new PGW that will assign IP addresses to these subscribers.
This vulnerability is quite powerful in two aspects. First, when malicious SGW reports PGW failure to MME then this failure is cascaded to other SGWs too, as MME clears all subscribers' contexts related to the reported PGW. LTE design choice of associating multiple SGWs with one MME and a PGW is due to avoid IP address[1] change during device mobility. When

[1]PGW assigns IP address(s) to every subscriber device.

the device moves around, the SGW is relocated by keeping the PGW unchanged, hence the device keeps the same IP address. Second, this vulnerability brings incast micro-burst (signaling spikes) at stateful load balancer and MME VNF that may render them non-response for short period of time, as shown in Figure 4. This is because on receiving the *Implicit Detach Request* message, all devices (associated with different LTE base stations but one MME VNF[2]) initiate the LTE Attach procedure at roughly the same time. These signaling messages, arriving from distributed LTE base stations (i.e. eNodeBs), are received by stateful load balancer that forwards to MME VNF (approximately at same messages arrival rate). Note that the incast micro-burst problem is known in public cloud due to TCP incast throughput collapse [24], [25], [26]. In this paper, we reveal that the standardized PGW failure recovery procedure also introduces incast micro-burst in public cloud. An attacker can benefit by triggering incast micro-bursts through false failure recovery signaling messages.

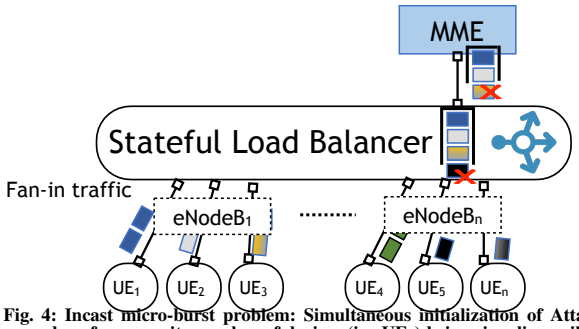

Fig. 4: Incast micro-burst problem: Simultaneous initialization of Attach Request procedure from a quite number of devices (i.e. UEs) bring signaling spikes at cloud.

Vulnerability 1: Malicious SGW can disrupt LTE service provided by victim tenant. The exploit of this vulnerability can be cascaded to other victim's SGWs, as MME clears device sessions spanned over multiple SGWs. Also, this vulnerability inadvertently creates micro-burst at cloud when distributed LTE devices try to re-attach with MME.

B. Master key exposure during device intra-system switch

We find that the device master key is exposed during intra-system switch – when MME is relocated during device mobility. We recall that during device registration procedure, the MME receives master key (i.e. K_{ASME} key) from HSS. MME then uses K_{ASME} to derive K_{eNB} (the key to secure radio communication), and a couple of other keys to cryptographically secure the connection between device and MME. This K_{ASME} key does not change until the user removes and re-inserts the SIM card [27]. The same K_{ASME} is used to derive future K_{eNB} in a chain as the user performs handover from one LTE base station (also known as eNodeB) to the other.

As LTE subscriber moves, the subscriber device sessions may need to be transferred from serving MME to target MME. This process is known as LTE S1-handover procedure. During device mobility, the serving MME sends K_{ASME} key to the target MME in the *S10 Forward Relocation Request* message. The source MME also sends Next hop Chaining Counter (NCC) and Next Hop (NH) values, which are used to derive K_{eNB} in a key chain, to target MME. The rationale of sending K_{ASME} from source MME to target MME is to avoid device re-authentication procedure. Such design choice has merit since traditionally LTE EPC nodes are implemented over carrier grade boxes making a private LTE core network. However, in public cloud, a malicious EPC node can infiltrate into other tenant's network and K_{ASME} transfer in plain text is vulnerable. Cloud service providers virtually partition physical network infrastructure into a number of virtual networks. This network partitioning ensures that intra-tenant communication is not exposed to other tenants. Using our threat model, malicious SGW VNF becomes the part of victim tenant's network. Being the member of victim's operational network, malicious SGW can sniff those packets which are exchanged within victim tenant's network boundaries. There are a number of network packets sniffing tools, such as wireshark [28], Vmware's [29], and Citrix's [30] sniffers, that the attacker can use to sniff intra-tenant VNFs network communication. By passively sniffing the network traffic, the attacker can capture K_{ASME} values belonging to all those devices that performed intra-system switch. In Figure 5, we show a wireshark trace that captures the K_{ASME} value. The attacker can send these keys to outside collaborators to launch a number of attacks (including ciphering/deciphering radio packets, controlling the behavior of compromised subscribers, and many more) which were never possible before in LTE.

Fig. 5: KASME key can be captured through wireshark on intra-system switch

Vulnerability 2: K_{ASME} key value is captured by malicious SGW VNF when device performs intra-system switch during its mobility.

C. Memory pressure through a false downlink notification message

A false downlink notification message from malicious SGW VNF renders MME VNF to reserve the memory for hundreds of devices. The downlink notification message enables the device to re-establish the session with LTE network that it has been teared down while entering into low power idle state. The device enters into idle state when it has no data to send or receive. In the idle state, the device releases its radio connection with eNodeB to conserve the battery. The device periodically listens the broadcast paging message (which is a downlink notification message for device) to check if there is any incoming data waiting to be transmitted at EPC. The paging message is initiated by MME when it receives a downlink data notification signal for particular device from SGW. On receiving the paging message, the device establishes the radio connection with eNodeB followed by *Service Request* initial NAS message. On receiving the *Service Request* message from device, the MME authenticates the subscriber and modifies the data bearer at SGW and PGW.

An attacker can exploit this LTE feature to put memory pressure at MME VNF. In his approach, the malicious SGW VNF sends a false downlink paging notification message to MME VNF, as shown in Figure 6. In that message, it puts the message cause as paging message and provides bearer identities for

[2]During device mobility, device changes LTE base station by performing X2 handover, but keeps same MME [13].

up to one thousand devices[3]. On receiving the paging message notification from SGW VNF, MME VNF reserves the memory for every device addressed in downlink notification message. It then initiates the paging procedure through eNodeBs[4] that send broadcast paging messages addressing multiple devices. On receiving the paging message, the subscriber devices first initiate the radio connection with eNodeB and transition into connected state. They send *Service Request* messages to MME VNF to establish their data channel. MME VNF authenticates these devices and establishes their bearer resources for uplink/downlink transfer of data packets. Because these *Service Request* messages were initiated due to false downlink notification from malicious SGW VNF, there exists no data activity from/to devices. The MME awaits for device inactivity timer to expire (usually set as 11-12 seconds [31]) before releasing the connections for devices, and hence clearing the memory.

Through this vulnerability, malicious SGW VNF can keep MME VNF memory occupied by periodically (at an interval of 12 seconds) sending downlink notification message. As a result, the attacker can slow down messages processing at victim VNF and incur control-plane latencies [32], [33]. The increased memory pressure on MME also impacts co-located VMs instances (that share the same physical memory) [34], [35] that may impact their performance as well. Furthermore, this vulnerability also silently drains the battery of victim devices. It has been shown in [36] that the power consumption in device connected state is 3×-4× higher compared to its idle state.

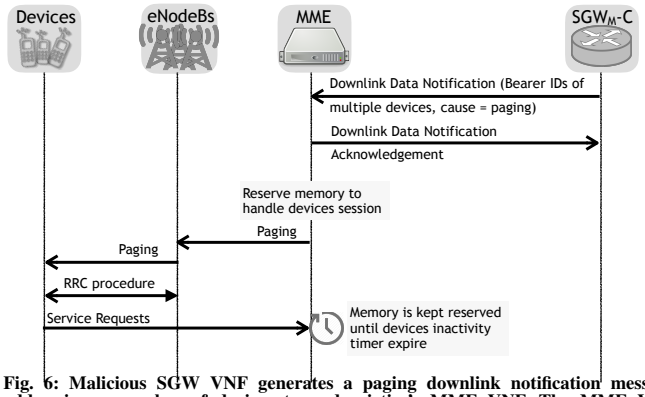

Fig. 6: Malicious SGW VNF generates a paging downlink notification message addressing a number of devices towards victim's MME VNF. The MME VNF reserves the memory and initiates the paging procedure towards target devices. Through this procedure, malicious SGW VNF can increase memory usage at victim's MME VNF.

Vulnerability 3: False downlink data notification signaling message puts memory pressure on MME VNF that lets victim VNF to reserve memory space for a number of devices. It also impacts victim tenant's subscribers devices that end up consuming significantly higher battery power.

D. Slowing GTP forwarding plane by injecting fake IP packets

We find that a malicious SGW VNF can throttle the victim tenant's user-plane traffic by simply sending fake IP packets to victim tenant's forwarding plane. On device registration, once the user is authenticated and authorized by MME, PGW control plane assigns the IP addresses and packet forwarding precedence priority[5]. It also disseminates these policies to SGW control plane VNF. Both SGW and PGW apply IP packet forwarding rules at their user-plane forwarding engine, as *¡rule, action¿* tuples. The rule represents the matching of the different packets according to policy, and the action refers to the basic operation to be carried out over the incoming packets. Much like OpenFlow switching tables[37] and Service Data Flow Templates in LTE (Figure 6.5 in LTE policy and charging control architecture specification[38]), these rules are installed in forwarding tables of SGW-U and PGW-U, as shown in Figure 7. The tables closer to ingress ports store high precedence rules compared to the tables which are closer to egress ports. If the incoming packet does not match any rule at all the tables, it is dropped.

An attacker (SGW-C VNF) exploits the fact that incoming packet rule is searched at all forwarding tables before taking the action of dropping the packet. It first installs few fake IP packets forwarding rules as the highest precedence at its user-plane and then starts injecting these packets. When the fake IP packets arrive SGW-U from SGW-C, they are matched at 1st forwarding table and are sent to PGW-U. The PGW-U does not contain any entry of these fake IP packets as these IP addresses were never assigned by PGW-C. However, PGW-U needs to find the match of all fake IP packets at all of its forwarding table before discarding these packets. This process of matching of IP packets at all forwarding tables introduce extra packet-processing overhead that slows down other legitimate IP packet flows sharing the common hardware resource. Note that, the attacker is not flooding PGW-U with fake IP packets which is equivalent to denial of service attack and can easily be detected. Here the attacker's goal is to send only those number of packets which relatively slow down the victim tenant's forwarding plane performance compared to the performance that the attacker tenant is providing to its subscribers.

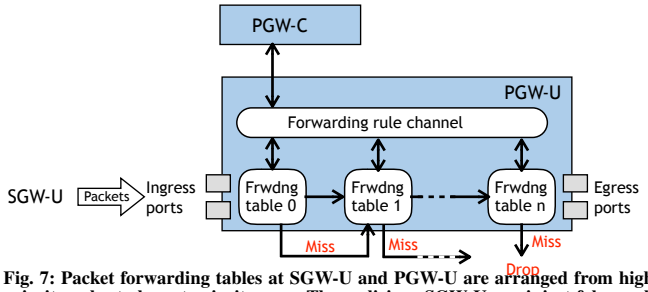

Fig. 7: Packet forwarding tables at SGW-U and PGW-U are arranged from highest priority rules to lowest priority ones. The malicious SGW-U can inject fake packets towards PGW-U that force PGW-U to search the rule at all of its forwarding tables. This procedure slows down the packet forwarding of legitimate IP packets.

Vulnerability 4: Injecting fake IP packets slow down the forwarding plane performance of victim tenant.

V. Solution Goals and Overview

Goals: We want to achieve the following two goals in our solution.

1) GTP-C ciphering and authenticity: We want to cryptographically secure GTP-C communication.

[3]One GTP-C payload message size is 64KB, whereas device identifier length is 64 bytes. LTE standard allows SGW to include multiple devices bearer identities in single downlink notification message [22]

[4]Paging message is sent in registered tracking area of the device. This tracking area spans over multiple eNodeBs.

[5]For example, IP address assigned for voice traffic has higher packet forwarding priority than default IP address assigned to access the Internet.

2) GTP-U faithful packets forwarding: We want to ensure that the filtered packets reach PGW-U from SGW-U. Moreover, SGW-U should not be able to replay or delay packets.

Architecture and `vEPC-sec` component: Figure 8 provides an overview of our architecture. We propose a distributed architecture in which an LTE–NFV over cloud is decomposed into several LTE–NFV subnets. Dividing vEPC into subnets ensure fault tolerant and scalable network design. Our solution introduces `vEPC-sec` component, a central entity for providing key management to GTP-C traffic. It also ensures that only legitimate data packets are forwarded from SGW-U to PGW-U. We assume that `vEPC-sec` component is highly reliable with 1:1 redundancy [39], and communicates with EPC VNFs over secure channels only.

Solution overview: We propose `vEPC-sec` that (1) cryptographically secures communication over GTP-C, and (2) prevents illegitimate packets injection at GTP-U.

At GTP-C, our idea is to provide distributed key management scheme from which LTE EPC VNFs derive ciphering and integrity keys to encrypt and integrity protect their messages. When EPC VNF is selected to serve the subscriber, it connects with `vEPC-sec` over a secure interface (shown as double dotted lines in Figure 8) and requests the shared symmetric keys to communicate with other EPC VNFs. In the request message it includes the VNF identities with which it wants to communicate, as well as its own identity. `vEPC-sec` first checks whether all these virtualized EPC (vEPC) instances are part of same tenant by contacting local database. If the answer is positive then it runs Key Derivation Function (KDF) and generates 3 pairs of keys so that MME, SGW and PGW VNFs can independently communicate with each other. `vEPC-sec` then sends these keys to corresponding EPC VNFs. Every VNF then locally derives integrity and ciphering keys against both keys it has received from `vEPC-sec`. Thereafter, the signaling messages between a pair of VNFs are ciphered and integrity protected. Our solution addresses **vulnerability 1** (§IV-A) and **vulnerability 3** (§IV-B) when MME only accepts integrity protected and ciphered messages from PGW-U (sent via SGW-U) using derived shared keys between MME and PGW VNFs. Therefore, SGW-U cannot lie that the message is originated from PGW-U. Further, it addresses **vulnerability 2** (§IV-B), as K_{ASME} is transferred over encrypted GTP-C between MME_{source} and MME_{target}.

At GTP-U, we introduce the concepts of assigning SGW-U the role of firewall, and correlating data packets received at LTE radio network and PGW-U. SGW-U plays the role of a firewall when PGW-C assigns the default packet forwarding policy to drop the packet. As a result, SGW-U only allows those IP packets whose addresses are assigned by PGW-C (via SGW-C); hence addressing **vulnerability 4** (§IV-D). We further ensure that only those packets should reach PGW-U which are sent by the legitimate subscriber. That is, SGW-U should not be able to replay IP packets towards PGW-U. We achieve this by matching packets sent from LTE base station to `vEPC-sec`, and packets received at PGW-U from SGW-U. By correlating packets sequence numbers from both entities ensure that they were originated by the device and were not delayed/droped by SGW-U. `vEPC-sec` also correlates IP address with device Cell Radio Network Temporary Identifier (C-RNTI)[6]. C-RNTI and IP address mapping confirm that IP packets are originated by the legitimate device, hence avoid IP spoofing attack reported in LTE [40], [8].

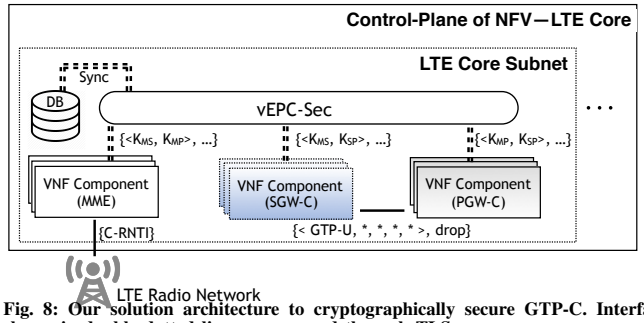

Fig. 8: Our solution architecture to cryptographically secure GTP-C. Interfaces shown in double dotted lines are secured through TLS.

VI. Solution

A. LTE GTP-C confidentiality and integrity protection

We propose distributed security keys derivation and management scheme for integrity protection and ciphering of GTP-C signaling messages.

Distributed security keys derivation and management for GTP–C: Our solution provides a security abstraction module `vEPC-sec`, responsible of providing symmetric keys to LTE VNFs. When a VNF is chosen to serve a subscriber (during device registration procedure), it first checks whether it has the symmetric keys to securely communicate with other selected VNFs or not. If the keys exist then the subscriber signaling messages exchange between these VNFs are ciphered and integrity protected; otherwise, shared symmetric keys are retrieved from `vEPC-sec` over TLS connection[7]. LTE VNF retrieves the keys by sending *Keys Information Request* message, requesting security keys for GTP-C communication. This request includes its VNF identity (which is Universal Unique Identifier (UUID) assigned to VM [41]), as well as identities of other VNFs with which it will communicate. Upon the receipt of the *Keys Information Request* message from the LTE VNF, `vEPC-sec` contacts the database and determines whether all these VNFs belong to same operator or not. If the response is negative then an alarm message will be sent to NFV orchestrator to take further action. This is the first line of defense in which any attempt from malicious tenant to infiltrate into victim tenant network is thwarted. In case all VNFs included in *Keys Information Request* message belong to the same tenant, `vEPC-sec` computes K_{MS}, K_{MP}, and K_{SP} to secure the communication between MME – SGW, MME – PGW, and SGW – PGW VNFs, respectively. Each key is derived from the KDF by using inputs of 256 bits long `vEPC-sec` master key and RAND value, as well as identities of two VNFs with which the key will be shared. We have shown keys derivation steps in Figure 9. `vEPC-sec` applies the H-MAC based KDF, as specified in TS33.220 3GPP specification [42].

After deriving the keys, `vEPC-sec` sends *Keys Information*

[6]C-RNTI uniquely identifies the device over the air. This C-RNTI remains unchanged until the device releases its radio connection (i.e. RRC Connection Release).

[7]LTE VNF and `vEPC-sec` interface is protected using TLS. No message is exchanged until secure tunnel is established

Response message back to the VNF that has requested the keys. It also sends *Keys Allocation Request* message to other two VNFs for whom the keys were derived in the process. These messages contain two keys required to communicate with other two EPC VNFs as well as encryption and integrity algorithm identities[8] for further key derivation. On receiving the message from `vEPC-sec`, every VNF further derives ciphering and integrity keys. It inputs encryption algorithm identity and the received key value to derive the ciphering key. Similarly, it inputs integrity algorithm identity along with the received security key and derives the integrity key. We have shown keys derivation at VNFs in Figure 9. Once both ciphering and integrity keys have been derived, these are truncated and the 128 least significant bits are used. Thereafter, VNF can use these keys to send ciphered and integrity protected messages over GTP-C interface to other paired VNFs.

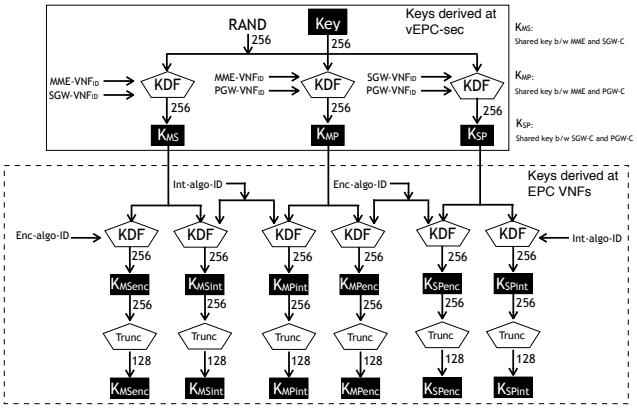

Fig. 9: Keys hierarchy and derivation for securing communication at LTE GTP-C interfaces.

Derivation of MAC, and ciphering the messages: After integrity and encryption keys derivation, we discuss how our solution encrypts messages and calculates their Message Authentication Code (MAC) for integrity check. Let's assume sender wants to send a message to receiver VNF over GTP-C. The sender VNF first calculates the MAC through EPS integrity algorithm [5]. The algorithm takes a number of input parameters: (1) 128 bit shared symmetric key between sender and receiver VNFs, (2) a 32-bit Nonce, (3) 1-bit direction of the transmission, and (4) the GTP-C signaling message itself. The Nonce value is a pseudo-random number to ensure that old messages cannot be replayed. The direction bit is 0 for uplink and 1 for downlink message. After calculating MAC, the sender then ciphers the message by using EPS encryption algorithm [5]. The input values to the algorithm are: (1) 128 bit shared symmetric key between sender and receiver VNFs, (2) a 32-bit Nonce, (3) 1-bit direction of the transmission, and (4) the length of the GTP-C signaling message to be sent. The encryption algorithm outputs keystream block equals to the length of the message. The message is then encrypted using a bit per bit binary addition of the plaintext GTP-C message and the keystream block. The sender sends the encrypted message, MAC and the Nonce value to the receiver. The receiver first ensures that the Nonce value is not the one it has received before. The receiver then calculates the MAC and matches it with received MAC value to ensure the integrity protection of the message. If the integrity check is passed, the receiver decipher the encrypted message. The receiver recovers the message by generating the same keystream using the same input parameters by the sender and applying a bit per bit binary addition with the ciphertext.

Securing communication during device mobility: Up till this end, `vEPC-sec` secures the communication between MME, SGW and PGW VNFs. Due to device mobility, MME_{source} needs to exchange handover signaling messages as well as providing K_{ASME} key to MME_{target}. To meet the security requirement in device mobility, we extend our key management technique for communication between two MME VNFs. On receiving the *Handover Required* message from LTE base station, MME_{source} determines the address of MME_{target} and asks `vEPC-sec` to provide the shared symmetric key to securely communicate with MME_{target}. `vEPC-sec` generates the K_{MM} and gives it to MME_{source} along with integrity and ciphering algorithm identities. It also sends a message *Handover Key Establishment* to MME_{target} that include K_{MM} and integrity and ciphering algorithm identities. The message from `vEPC-sec` explicitly informs MME_{target} that it would receive a ciphered and integrity protected message from MME_{source} which will be decoded using the provided key. On receiving the K_{MM}, MME_{source} proceeds with handover procedure and sends ciphered and integrity protected handover signaling message to MME_{target}. The MME_{target} receives the message from MME_{source} VNF for which it has received the key, and deciphers the message after ensuring the message integrity check. This solution also addresses the **vulnerability 2** (§IV-D), because K_{ASME} is being sent over secured channel between MME_{source} and MME_{target}

B. LTE GTP-U faithful packets forwarding

Our solution ensures that SGW-U (1) does not inject any fake packets, (2) forwards the data packets without delaying, and (3) does not duplicate the packet forwarding. Moreover, `vEPC-sec` also prevents IP packets spoofing by attacker devices.

Making SGW-U the firewall for PGW-U: At the time of device registration, PGW-C VNF assigns the device IP address(es) and applies packet forwarding rules – that it receives from policy and charging LTE NF – at PGW-U. PGW-C also forwards the *¡rule, action¿* pair to SGW-C. SGW-C then installs these rules to its forwarding plane. This means both SGW-U and PGW-U apply identical packets forwarding rules. The SGW-U which receives the device data packets from radio network simply forwards these packets to PGW-U according to data forwarding policy. The vulnerability we discuss in §IV-D arises due to the fact that SGW-U forwards the fake IP packets and exhausts the forwarding table lookup at PGW-U. In principle, SGW-U should never forward a packet whose rule was not defined by PGW-C. It means there exists no practical use case scenario in which SGW-U and PGW-U forwarding policies ever mismatch. PGW-C uses this principle to address **vulnerability 4** (§IV-D). It simply explicitly provides packet drop policy to SGW-C when no rule is found. It sends *¡GTP-U * * * *, drop¿*[9] rule signifying that the default rule is

[8] `vEPC-sec` selects either 00010 or 0010 to represent AES or SNOW 3G algorithm identity, respectively [5], in its response message.

[9] Should be read as: packet from GTP-U protocol with any source address, any source port, any destination address, and any destination port will be dropped.

to drop the packet. We should mention that the default rule must be placed as the last entry of the last table at SGW-U, otherwise legitimate packets will be dropped. This is a common misconfiguration issue reported in packet forwarding middleboxes (liked routers and switches) [43].

Ensuring data packets are not maliciously throttled: Although above solution addresses fake IP packets injection problem, SGW-U can misbehave in a different way. It can replay legitimate data packets to overbill[10] the subscriber [8], and can even delay the data packets forwarding to throttle the end user data throughput. Note that, periodically delaying some TCP packets cause out of order delivery at the receiver. As a result, the TCP at the send side keeps transitioning between fast retransmit/fast recovery and congestion avoidance phases. The application data sending rate is throttled as a consequence.

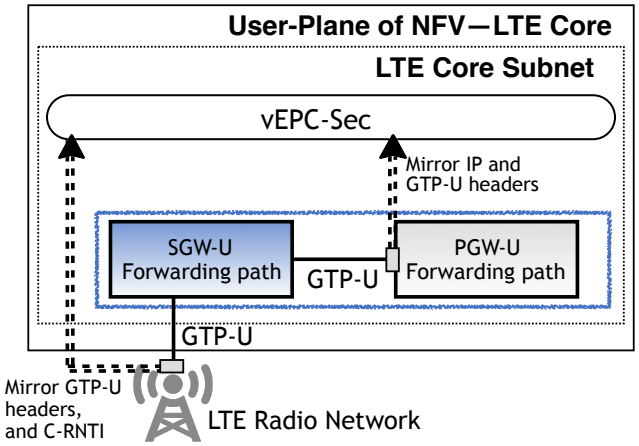

Fig. 10: Our solution to guard GTP-U traffic.

To address these issues, our approach is to enable data packets' headers inspection at `vEPC-sec`. When the device has some data to send, it establishes its data channel with LTE network (i.e. by sending *Service Request* message). When LTE base station receives the data packets from the device, it puts GTP-U header that includes message type, GTP-U tunnel identifier and the packet sequence number. The sequence number field uniquely identifies the packet in an IP flow at LTE network. The value is incremented on every data packet transmission. Our idea is to enable 1:1 mapping between packets sent by LTE base station to SGW-U and the ones received by PGW-U from SGW-U. This approach isolates the malicious acitivity done at SGW-U. As shown in Figure 10, we require that LTE base station should mirror its interface towards SGW-U to `vEPC-sec`. The mirrored packets are only the GTP-U headers and the device radio network identity (i.e. C-RNTI), and does not include packets payload. Similarly, PGW-U mirrors the IP and GTP-U headers of the packets that it has received from SGW-U to `vEPC-sec`. By looking at same headers reported from two different entities, `vEPC-sec` can distinguish any missing packets, out of order packets, and even duplicate packets.

Ensuring data packets are originated by legitimate device: By correlating C-RNTI with IP packets, `vEPC-sec` can also avoid other attacks that have been reported in recent past. These include spoofing of IP packets and injecting data packets by using the IP address of control-plane [44], [45], [8]. These attack mainly occur when the malicious device which is authenticated during connection establishment phase may lie about itself while sending IP packets. To address this issue, `vEPC-sec` binds the C-RNTI with the IP address the P-GW has assigned. In this way, the attacker can only send IP data by using its own data-plane IP address. He can neither use control-plane IP address or spoof IP address of other subscribers.

[10]The subscriber pays for the data packets it has sent/received at PGW-U. So SGW-U can replay subscriber data packets to overcharge the subscriber.

C. Discussion

We briefly discuss how our solution works during VNFs failure recovery and scalability scenarios. We also discuss other implementation related challenges and our solution to address them.

Fault tolerance and scalability: LTE network operators aim to provide all-time service access to their subscribers. Both cloud service providers and LTE standard discuss failure recovery [46], [47], [48], [49] and scalability [50], [51] procedures. In failure recovery procedure, a standby EPC VNF replaces the failed VNF, and failed signaling messages are re-executed. It is possible that during the recovery process the alternative VNF cannot restore the GTP-C security keys (e.g. in fail-stop failure scenario). Similarly, scalability requirements stipulates that a new VNF instance should be prepared to handle increasing subscribers requests. This new VNF does not have the security keys to communicate with its peer VNFs. To address these challenge, we propose that once the new VNF instance becomes active, it first contacts `vEPC-sec` by sending *Keys Information Request due to Recovery and Scalability* message. In this message it includes, its own VNF identity, and the VNF identity of the failed instance – in case of failure, or the identity of the original VNF that is being scaled. On receiving the request, `vEPC-sec` first determines all those VNF instances that have been affected due to failure/scalability. It initiates "key change on the fly" procedure by sending *Re-keying Required* message to all affected VNFs. Once these VNFs receive *Re-keying Required* message from `vEPC-sec`, they suspend their communication and prepare to change the key by responding with *Re-keying Request Acknowledged* message. On receiving the *Re-keying Request Acknowledged* message from all these VNFs, `vEPC-sec` derives and distributes new security keys, according to procedure discussed in §VI-A. In this way, the new VNF as well as other affected VNFs can resume secure messages exchange over GTP-C interface.

Nonce value wrap-around: `vEPC-sec` uses increasing Nonce value to compute MAC and ciphered text. The length which is 32 bit long can generate more than 4 billion unique Nonce values. These number of values although sufficient if they are used per device, are not enough when the Nonce is incremented for every GTP-C signaling message sent/received at VNF. Naturally, the Nonce value wraps around when all unique Nonce values have been used. Due to wrap-around, the GTP-C messages are dropping at the receiver VNF that mistakenly corresponds these messages as a replay attack. To address this issue, we propose of re-keying whenever the wrap around occurs. On wrap-around, the VNF sends

Keys Information Request due to Wrap-Around message to vEPC−sec. In this message it includes, its own VNF identity, and the identities of those VNFs with which it was exchanging GTP-C signaling messages. vEPC−sec then initiates "key change on the fly" procedure as discussed above to update the GTP-C communication keys between these VNFs. Once the new keys are assigned, it is safe to use wrap-around Nonce value as an input to generate MAC and ciphered text block.

Incremental deployment: In designing vEPC−sec, we understand that network operators are interested in NFV for new and expanding deployments but might be less enthusiastic about re-writing their EPC implementations. Our solution acts like plug-and-play and does not conflict with any LTE standard protocol working. We do not make changes to existing LTE interfaces. When LTE VNFs power-on, they setup TLS connection with vEPC−sec. Afterwards, the key derivation is only done once, when a VNF is selected to serve a subscriber. Note that we do not perform GTP-C key derivation for every subscriber registration request, rather it is done once when three VNFs (i.e. MME, SGW and PGW) establish their connections first time to serve a subscriber. In practice hundreds and thousands of devices are assigned to same set of VNFs for which GTP-C key derivation is performed once.
In other words, cloud LTE tenants need to add only few lines of source code at VNF initialization phases (i.e. at VNF bootup and VNFs connection establishment phases). These few changes can ensure network operators that their subscribers GTP-C and GTP-U are well protected.

VII. Security Analysis

We briefly discuss that why available cloud security mechanisms do not protect from LTE NFV security vulnerabilities. Later, we provide security analysis of vEPC−sec.

A. Limitations of cloud security solutions

In cloud, packets exchange can be cryptographically secure either by using TLS at transport layer, or Internet Protocol security (IPsec) at networking layer of the protocol stack. Firewall, Intrusion Detection Systems (IDS), and Intrusion Prevention Systems (IPS) middleboxes block unauthorized access to tenant VNFs, detect and prevent the malicious activities at these VNFs, respectively. We find that all these mechanisms are not sufficient to provide LTE NFV security.
GTP uses UDP/IP protocol to transfer GTP messages[21] and cannot use transport layer security mechanisms. GTP tunnels can use IPsec to secure their messages. However, IPsec does not meet high availability and fault tolerance LTE requirements[52], [53]. It takes more than 15 seconds to re-establish new signaling bearers with the subscribers (which were served by the failed VNF instance)[52]. This delay is 18× more than LTE high availability requirement of five-nines[11] (i.e. VNF downtime should not be greater than 864.3 milliseconds per day). Moreover, IPsec does not protect against fake IP packets injection vulnerability (§IV-D).
Security middleboxes in the cloud have their own limitations. Firewall performs stateful inspection of GTP traffic entering into the tenant network. Its functions protect the mobile packet core from signaling storms and man in the middle attacks. However, it does not guard against insider attacks when an adversary VNF becomes part of victim tenant's network. The purpose of IDS/IPS is to perform signature based packets inspection to find a malicious activities between LTE VNFs. They also fail to detect discussed vulnerabilities when malicious tenant VNF fully obeys the LTE standards to alter the functionalities provided by victim tenant.

[11]Public cloud provides four nines of high availability, that is downtime of 8.64 seconds/day is allowed [12].

B. vEPC−sec security analysis

We are mainly concerned in analyzing vEPC−sec in three dimension: (1) whether an adversarial VNF can get the shared-symmetric keys to communicate with the victim tenant's VNFs? (2) whether it can abuse PGW-U resources by injecting fake IP packets?, and (3) whether the attacker can limit the victim subscriber's packets rate?

On secure communication between malicious and victim tenant's VNFs: Our solution does not allow the malicious VNF to establish a secure GTP-C connection with victim VNFs. We consider an adversarial model in which an adversary can communicate with vEPC−sec to derive the keys. Although, an adversary cannot sniff TLS protected packets between vEPC−sec and victim VNFs, it can get the VNFs identities through other means (e.g. sniffing the ARP packets and decoding UIUD from MAC address[54]). To understand, we take an example of malicious SGW that holds victim tenant's MME and PGW VNF UIUDs and describe it in Analysis 1 pseudocode. Malicious SGW first establishes the TLS connection with vEPC−sec and then sends the *Keys Information Request* message. In the message, it includes UIUDs of its VNF as well as victim MME and PGW VNFs. On receiving the key generation request, vEPC−sec first verifies whether all these VNFs belong to same tenant or not. It contacts cloud database to get an answer. The cloud database has the record of all UIUDs and has mapped these identities against the operator, location, priority and weight factor. It replies vEPC−sec with the operator names that host and manages these VNFs. On receiving the response, vEPC−sec determines that all three VNFs do not belong to the same operator and hence rejects the request by sending *Keys Information Request Rejected* message back to malicious SGW. It can mention the reject cause as: *different operators*. The malicious SGW can try all different combinations of MME and PGW identities and can send *Keys Information Request* as many times as it wants. Every time, it's request will be rejected by vEPC−sec. Note that, we can improve the implementation of vEPC−sec by raising an alarm to NFV orchestrator that can take further action against malicious tenant.

On injecting fake IP packets: Our solution detects the fake IP packets injection by SGW-U. In Analysis 2, we show that the adversary is allowed to inject fake IP packets which is against the policy provided by PGW-C. When these fake IP packets arrive at PGW-U they are marked as resource abuse attempt packets. As there exists no forwarding table entry against these fake IP packets. PGW-U then sends an alarm signal message to PGW-C that takes further action after contacting NFV orchestrator. We should point out, it is not possible for SGW-U to send the IP packets when it recovers from the failure. This is because that the lost data bearers

vEPC−sec Analysis 1 Adversary tries to receive shared symmetric keys to communicate with victim VNFs.

```
Assume an adversary can sniff all VNF identities of victim tenant;
Let SGW_M = Adversarial controlled SGW-C VNF identity;
Let MME_V[n] = VNF identities of victim tenant's MMEs;
Let PGW_V [n]= VNF identities of victim tenant's PGW-C;
for i = 0 to MME_V[n] do
    for j = 0 to PGW_V[n] do
        SendTovEPC−sec (KeysInformationRequest, SGW_M, MME_V[i],
        MME_V[j])
        if KeysInformationResponse == TRUE then
            return 1; // Adversary wins
            else
                return 0;// Adversary loses
            end
        end
    end
end
```

vEPC−sec Analysis 2 Adversary tries to misuse PGW-U resources by sending fake IP packets or delaying packets.

```
Let receiver = Adversarial controlled machine over the Internet;
if SendtoPGW-U(msg, FAKE_src_IP, dest_IP) == SUCCESS &&
ReceivefromPGW-U(msg, FAKE_src_IP, dest_IP)==SUCCESS then
    return 1; // Adversary wins
    else
        return 0;// Adversary loses
    end
end
```

are required to be re-established by SGW-C first, and data forwarding policies are installed afterwards.

On illegal throttling of data packets: We show that an attacker cannot illegally throttle subscriber's data packets by delaying the packets forwarding. In our analysis, as shown in Analysis 3 pseudocode, the adversarial control attacker receives the packets from LTE base station and delays their forwarding to PGW-U. When the PGW-U receives the packets (both delayed and not delayed), it mirrors packets' headers to vEPC−sec (refer to Figure 10) before forwarding them to the Internet. vEPC−sec performs 1:1 mapping of packet sequence numbers that it has received from LTE base station and PGW-U. If the sequence numbers mismatch is consistently observed for a certain period of time (as the attacker periodically delays packets forwarding to achieve throughput throttling), vEPC−sec raises an alarm towards NFV orchestrator. NFV orchestrator then needs to identify that whether the pause in data forwarding by SGW-U is intentional or not. If it is intentional then it takes an action against malicious tenant, otherwise it replaces slow performing SGW-U instance.

Performance: We simulate to determine the performance of vEPC−sec. First, we determine how quickly our solution can detect the throttling of data packets. The challenge we face was to distinguish between slow performing SGW-U with the malicious one. To solve this challenge, we implement a sequence number window at vEPC−sec. Our sequence number window is linearly numbered. When the packet sequence number from LTE base station arrives, we put it in the window and wait for the packet sequence number from LTE PGW-U. On receiving the sequence number from PGW-U, the difference is calculated. If the difference is zero it means there is no packet delay. If the next packet sequence number has arrived from LTE base station while our window was waiting for a packet from PGW-U, we move the window. That is, we do not record the delayed packet. In this way, our final window has churn of readings representing packet sequence numbers.

vEPC−sec Analysis 3 Adversary tries to throttle the victim tenant's subscriber packets.

```
Let receiver = Adversarial controlled machine over the Internet;
while Certain TIME has not not passed do
    if ReceivefromENB(msg, src_IP, dest_IP) == SUCCESS then
        WAIT(timer); //wait for certain time before forwarding to PGW-U
        SendtoPGW-U(msg, src_IP, dest_IP);
        sent_count = sent_count + 1; //count packets sent by SGW-U
        ReceivefromPGW-U(msg, src_IP, dest_IP);
        received_count = received_count + 1; //count packets received at
        reciver
    end
end
if sent_count == received_count then
    return 1; // Adversary wins
    else
        return 0;// Adversary loses
    end
end
```

That is, once the window has skewed, then the this skew will keep increasing over time. In this way we detect the adversarial SGW-U that periodically delays the packets. From the Figure 11, we can see that in just 30 seconds, vEPC−sec can detect packets delaying malicious activity when the gap has largely skewed away from the linear line.

In Figure 12, we show an overhead associated with KDF. We consider a machine with CPU of 2.5GHz and 3GB RAM. Our approach only causes one time overhead of 2.5 seconds. This overhead is also associated with the number of times the failure recovery has occurred. For every failure recovery procedure, vEPC−sec needs to generate fresh pairs of shared-symmetric keys. The re-keying process which explicitly ask VNFs to calculate the key has the lowest overhead. This is mainly due to the fact that these VNFs have to locally run KDF once.

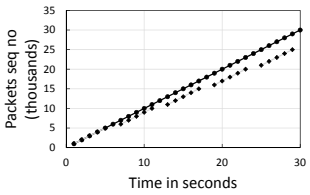

Fig. 11: Data packets throttling detection

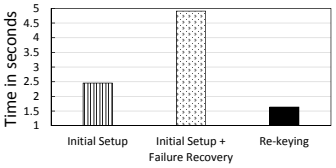

Fig. 12: Key management overhead

VIII. Related Work

The most recent work on NFV security is SafeBricks, published last year [55]. It shields generic NFs from an untrusted cloud and proposes to encrypted all the traffic entering into the cloud. In our work, we did not provide extra encryption of already encrypted traffic (e.g. traffic ciphered by secure DNS and secure SCTP protocols). Rather, our focus is to cryptographically secure the unsecured LTE GTP-C traffic. Other works [56], [57] discuss security issues associated to multi-tenancy and live migration. [58], [59] use Intel Software Guard Extensions (Intel SGX) to securely isolate the states of NFV applications. [60], [61] unveils DDoS attack that comes from flexible and elastic resource provisioning in NFV. Contrary to all these works, this paper presents attacks which are unique to LTE operations. We show how an adversary by sending fake signaling messages can disrupt LTE service, and to be worse, no middlebox signature based vulnerability detection solution can detect these types of attacks. Further, all these previous works have not discussed attacks on user-plane, but our paper addresses.

A number of other works discuss LTE security issues. [62], [36] conduct LTE protocol vulnerability analysis and show real impacts on LTE subscribers. [63] conducts experimental validation to prove that LTE temporary identity can disclose subscriber location. [64] discusses privacy attacks in which signalling information is leveraged to infer user privacy information. [65] shows that current cellular infrastructures exhibit security loopholes (off-path TCP hijacking) due to their NAT/firewall settings. [40], [8] study insecurity in mobile data charging. [44], [45] discuss how a subscriber can inject control-plane traffic into user-plane and can get free data service. Different to all above works, we do not discuss security vulnerabilities originated by an adversarial device. Rather, we present first work that discuss security issues arising from LTE core network implemented over public cloud.

IX. Conclusion

We propose vEPC−sec that secures LTE NFV over public cloud. It cryptographically protects LTE control-plane traffic on virtualized instances, and enforces data forwarding policies at every forwarding module. vEPC−sec enables encryption and integrity protection in LTE core network through distributed key management scheme. It's design ensures that communication between LTE NFs must be secure even during NF scalability and failure recovery scenarios. vEPC−sec provides light weight data forwarding monitoring component that only checks one type of header from two different sources to identify whether subscriber packets were delayed or duplicated. The security analysis confirms that vEPC−sec shields LTE core network traffic from adversarial model over public cloud.

References

[1] HP: Hybrid Cloud Solutions for LTE NFV, note= https://h20195.www2.hpe.com/v2/getpdf.aspx/4aa6-8334enw.pdf.

[2] Affirmed: Mobile Cloud on NFV, note= https://www.affirmednetworks.com/products-solutions/mobile-core/.

[3] Ixia: NFV, SDN and virtual network solutions, note= https://www.ixiacom.com/solutions/segments/service-providers.

[4] Aricent: Evolution of the Digital Network, note= https://www.aricent.com/services/rapid-network-transformation.

[5] 3GPP. TS33.401: 3GPP SAE; Security architecture, Sep. 2013.

[6] 3GPP. TS33.310: Network Domain Security (NDS); Authentication Framework (AF), Dec. 2014.

[7] 3GPP. TS33.210: 3G security; Network Domain Security (NDS); IP network layer security, Dec. 2014.

[8] Peng, Chunyi and Li, Chi-Yu and Wang, Hongyi and Tu, Guan-Hua and Lu, Songwu. Real threats to your data bills: Security loopholes and defenses in mobile data charging. In *Proceedings of the 2014 ACM SIGSAC Conference on Computer and Communications Security*, pages 727–738, 2014.

[9] R. Mijumbi and et al. Network function virtualization: State-of-the-art and research challenges. *IEEE Communications Surveys & Tutorials*, 18(1):236–262, 2016.

[10] Bringing Network Function Virtualization to LTE, November 2014. http://www.4gamericas.org/files/1014/1653/1309/4G_Americas_-_NFV_to_LTE_-_November_2014_-_FINAL.pdf.

[11] Qazi, Zafar Ayyub and et al. A High Performance Packet Core for Next Generation Cellular Networks. In *ACM SIGCOMM*, 2017.

[12] Binh Nguyen and et al. A reliable distributed cellular core network for public clouds. In *Technical Report: Microsoft Research*, 2018.

[13] M. T. Raza and et al. Rethinking lte network functions virtualization. In *IEEE ICNP*, 2017.

[14] Van der Merwe, Jacobus E and Rooney, Sean and Leslie, L and Crosby, Simon. The Tempest-a practical framework for network programmability. *IEEE network*, 12(3):20–28, 1998.

[15] Open vSwitch with DPDK Overview. https://software.intel.com/en-us/articles/open-vswitch-with-dpdk-overview.

[16] 3GPP. TS29.303: Domain Name System Procedures, Release 13, June 2016.

[17] Azure Load Balancer. https://docs.microsoft.com/en-us/azure/load-balancer/load-balancer-overview.

[18] Amazon EC2: Spot Fleet. https://docs.aws.amazon.com/AWSEC2/latest/UserGuide/spot-fleet.html.

[19] Secure SCTP: IETF draft-hohendorf-secure-sctp-25. https://datatracker.ietf.org/doc/draft-hohendorf-secure-sctp/.

[20] V. Fajardo and et.al. Diameter Base Protocol, 2000. RFC 7075.

[21] 3GPP. TS29.281: General Packet Radio System (GPRS) Tunnelling Protocol User Plane (GTPv1-U), 2013.

[22] 3GPP. TS29.274: Tunnelling Protocol for Control plane (GTPv2-C), 2014.

[23] 3GPP. TS23.007: LTE Restoration procedures, 2014.

[24] Chen, Yanpei and Griffith, Rean and Liu, Junda and Katz, Randy H and Joseph, Anthony D. Understanding TCP incast throughput collapse in datacenter networks. In *Proceedings of the 1st ACM workshop on Research on enterprise networking*, pages 73–82, 2009.

[25] Shan, Danfeng and Jiang, Wanchun and Ren, Fengyuan. Absorbing micro-burst traffic by enhancing dynamic threshold policy of data center switches. In *Computer Communications (INFOCOM), 2015 IEEE Conference on*, pages 118–126, 2015.

[26] Shan, Danfeng and Ren, Fengyuan and Cheng, Peng and Shu, Ran. Micro-burst in data centers: Observations, implications, and applications. *arXiv preprint arXiv:1604.07621*, 2016.

[27] 3GPP. TS24.301: Non-Access-Stratum (NAS) protocol for Evolved Packet System (EPS); Stage 3, Jun. 2013.

[28] Monitoring network traffic on VMware vSphere. https://wiki.wireshark.org/CaptureSetup/VLAN.

[29] Monitoring network traffic on VMware vSphere. https://kb.vmware.com/s/article/1038847.

[30] Monitoring network traffic on VMware vSphere. https://kb.vmware.com/s/article/1038847.

[31] Huang, Junxian and Qian, Feng and Gerber, Alexandre and Mao, Z Morley and Sen, Subhabrata and Spatscheck, Oliver. A close examination of performance and power characteristics of 4G LTE networks. In *Proceedings of the 10th international conference on Mobile systems, applications, and services*, pages 225–238, 2012.

[32] Markuze, Alex and Smolyar, Igor and Morrison, Adam and Tsafrir, Dan. DAMN: Overhead-Free IOMMU Protection for Networking. In *Proceedings of the Twenty-Third International Conference on Architectural Support for Programming Languages and Operating Systems*, pages 301–315. ACM, 2018.

[33] Shen, Zhiming and Zhang, Zhe and Kochut, Andrzej and Karve, Alexei and Chen, Han and Kim, Minkyong and Lei, Hui and Fuller, Nicholas. Vmar: Optimizing i/o performance and resource utilization in the cloud. In *ACM/IFIP/USENIX International Conference on Distributed Systems Platforms and Open Distributed Processing*, pages 183–203, 2013.

[34] Ren, Yi and Liu, Ling and Zhang, Qi and Wu, Qingbo and Guan, Jianbo and Kong, Jinzhu and Dai, Huadong and Shao, Lisong. Shared-memory optimizations for inter-virtual-machine communication. *ACM Computing Surveys (CSUR)*, 48(4):49, 2016.

[35] Kim, Jinchun and Fedorov, Viacheslav and Gratz, Paul V and Reddy, AL. Dynamic memory pressure aware ballooning. In *Proceedings of the 2015 International Symposium on Memory Systems*, pages 103–112. ACM, 2015.

[36] Raza, Muhammad Taqi and Anwar, Fatima Muhammad and Lu, Songwu. Exposing LTE Security Weaknesses at Protocol Inter-layer, and Inter-radio Interactions. In *International Conference on Security and Privacy in Communication Systems*, pages 312–338. Springer, 2017.

[37] Chiba, Yasunobu and Shinohara, Yusuke and Shimonishi, Hideyuki. Source flow: Handling millions of flows on flow-based nodes. *ACM SIGCOMM, year=2010*.

[38] 3GPP. TS 23.203: Policy and Charging Control Architecture, 2013.

[39] J. Sherry, P. X. Gao, S. Basu, A. Panda, A. Krishnamurthy, C. Maciocco, M. Manesh, J. Martins, S. Ratnasamy, L. Rizzo, et al. Rollback-recovery for middleboxes. In *ACM SIGCOMM*, August 2015.

[40] Peng, Chunyi and Li, Chi-yu and Tu, Guan-Hua and Lu, Songwu and Zhang, Lixia. Mobile data charging: new attacks and countermeasures. In *Proceedings of the 2012 ACM conference on Computer and communications security*, pages 195–204. ACM, 2012.

[41] VMware Workstation 5.0 Virtual Machine Identifier – UUID. https://www.vmware.com/support/ws5/doc/ws_move_uuid.html.

[42] 3GPP. TS33.220: LTE Generic Authentication Architecture (GAA) and Generic Bootstrapping Architecture (GBA), Sep. 2012.

[43] Fiebig, Tobias and Lichtblau, Franziska and Streibelt, Florian and Krueger, Thorben and Lexis, Pieter and Bush, Randy and Feldmann, Anja. SoK: An Analysis of Protocol Design: Avoiding Traps for Implementation and Deployment. *arXiv preprint arXiv:1610.05531*, 2016.

[44] Kim, Hongil and Kim, Dongkwan and Kwon, Minhee and Han, Hyungseok and Jang, Yeongjin and Han, Dongsu and Kim, Taesoo and Kim, Yongdae. Breaking and fixing volte: Exploiting hidden data channels and mis-implementations. In *Proceedings of the 22nd ACM SIGSAC Conference on Computer and Communications Security*, pages 328–339, 2015.

[45] Li, Chi-Yu and Tu, Guan-Hua and Peng, Chunyi and Yuan, Zengwen and Li, Yuanjie and Lu, Songwu and Wang, Xinbing. Insecurity of voice solution volte in LTE mobile networks. In *Proceedings of the 22nd ACM SIGSAC Conference on Computer and Communications Security*, pages 316–327, 2015.

[46] ALCATEL-LUCENT VIRTUALIZED EPC DELIVERING ON THE PROMISE OF NFV AND SDN. http://www.tmcnet.com/tmc/whitepapers/documents/whitepapers/2014/10743-alcatel-lucent-virtualized-epc-delivering-the-promise-nfv.pdf.

[47] Ericsson Virtual Router. https://archive.ericsson.net/service/internet/picov/get?DocNo=1/28701-FGB1010557&Lang=EN&HighestFree=Y.

[48] OPNFV – Building fault management into NFV deployments . https://www.opnfv.org/wp-content/uploads/sites/12/2016/11/opnfv_faultmgt_final.pdf.

[49] 3GPP. TS23.401: General Packet Radio Service (GPRS) enhancements for Evolved Universal Terrestrial Radio Access Network (E-UTRAN) access, 2011.

[50] Build massively scalable soft real-time systems. http://www.erlang.org/.

[51] 3GPP. TS36.401: LTE Architecture description, 2011.

[52] LTE Security for Mobile Service Provider Networks, note= https://www.juniper.net/us/en/local/pdf/whitepapers/2000536-en.pdf.

[53] High Availability is more than five nines. https://archive.ericsson.net/service/internet/picov/get?DocNo=10/28701-FGB101256&Lang=EN&HighestFree=Y.

[54]

[55] Poddar, Rishabh and Lan, Chang and Popa, Raluca Ada and Ratnasamy, Sylvia. SafeBricks: Shielding Network Functions in the Cloud. In *15th USENIX Symposium on Networked Systems Design and Implementation (NSDI'18), Renton, WA*, 2018.

[56] Firoozjaei, Mahdi Daghmehchi and Jeong, Jaehoon Paul and Ko, Hoon and Kim, Hyoungshick. Security challenges with network functions virtualization. *Future Generation Computer Systems*, 67:315–324, 2017.

[57] Moon, Soo-Jin and Sekar, Vyas and Reiter, Michael K. Nomad: Mitigating arbitrary cloud side channels via provider-assisted migration. In *Proceedings of the 22nd acm sigsac conference on computer and communications security*, pages 1595–1606. ACM, 2015.

[58] Shih, Ming-Wei and Kumar, Mohan and Kim, Taesoo and Gavrilovska, Ada. S-nfv: Securing nfv states by using sgx. In *Proceedings of the 2016 ACM International Workshop on Security in Software Defined Networks & Network Function Virtualization*, pages 45–48. ACM, 2016.

[59] Duan, Huayi and Yuan, Xingliang and Wang, Cong. LightBox: SGX-assisted Secure Network Functions at Near-native Speed. *arXiv preprint arXiv:1706.06261*, 2017.

[60] Fayaz, Seyed Kaveh and Tobioka, Yoshiaki and Sekar, Vyas and Bailey, Michael. Bohatei: Flexible and Elastic DDoS Defense. In *USENIX Security Symposium*, pages 817–832, 2015.

[61] Jakaria, AHM and Yang, Wei and Rashidi, Bahman and Fung, Carol and Rahman, M Ashiqur. VFence: A defense against distributed denial of service attacks using network function virtualization. In *Computer Software and Applications Conference (COMPSAC), 2016 IEEE 40th Annual*, volume 2, pages 431–436, 2016.

[62] Hussain, Syed Rafiul and Chowdhury, Omar and Mehnaz, Shagufta and Bertino, Elisa. LTEInspector: A Systematic Approach for Adversarial Testing of 4G LTE. In *Network and Distributed Systems Security (NDSS) Symposium*, 2018.

[63] Hong, Byeongdo and Bae, Sangwook and Kim, Yongdae. GUTI Reallocation Demystified: Cellular Location Tracking with Changing Temporary Identifier. 2018.

[64] Shaik, Altaf and Borgaonkar, Ravishankar and Asokan, N and Niemi, Valtteri and Seifert, Jean-Pierre. Practical attacks against privacy and availability in 4G/LTE mobile communication systems. 2018.

[65] Qian, Zhiyun and Mao, Z Morley. Off-path TCP sequence number inference attack-how firewall middleboxes reduce security. In *Security and Privacy (SP), IEEE Symposium on*, pages 347–361, 2012.