IEEE Access
Multidisciplinary | Rapid Review | Open Access Journal

Received March 20, 2017, accepted April 17, 2017, date of publication May 8, 2017, date of current version June 28, 2017.

Digital Object Identifier 10.1109/ACCESS.2017.2701416

Enhancing Security of Software Defined Mobile Networks

MADHUSANKA LIYANAGE[1], (Member, IEEE), IJAZ AHMED[1], (Student Member, IEEE), JUDE OKWUIBE[1], (Student Member, IEEE), MIKA YLIANTTILA[1], (Senior Member, IEEE), HAMMAD KABIR[2], (Student Member, IEEE), JESUS LLORENTE SANTOS[2], (Student Member, IEEE), RAIMO KANTOLA[2], (Member, IEEE), OSCAR LÓPEZ PEREZ[3], MIKEL URIARTE ITZAZELAIA[3], AND EDGARDO MONTES DE OCA[4], (Member, IEEE)

[1]Centre for Wireless Communications, University of Oulu, 90014 Oulu, Finland
[2]Aalto University, 11000 Aalto, Finland
[3]Nextel S.A., 48170 Zamudio, Spain
[4]Montimage, 75013 Paris, France

Corresponding author: Madhusanka Liyanage (madhusanka.liyanage@oulu.fi)

This work was supported in part by Tekes, Finland, and in part by the Academy of Finland.

ABSTRACT Traffic volumes in mobile networks are rising and end-user needs are rapidly changing. Mobile network operators need more flexibility, lower network operating costs, faster service roll-out cycles, and new revenue sources. The 5th Generation (5G) and future networks aim to deliver ultra-fast and ultra-reliable network access capable of supporting the anticipated surge in data traffic and connected nodes in years to come. Several technologies have been developed to meet these emergent demands of future mobile networks, among these are software defined networking, network function virtualization, and cloud computing. In this paper, we discuss the security challenges these new technologies are prone to in the context of the new telecommunication paradigm. We present a multi-tier component-based security architecture to address these challenges and secure 5G software defined mobile network (SDMN), by handling security at different levels to protect the network and its users. The proposed architecture contains five components, i.e., secure communication, policy-based communication, security information and event management, security defined monitoring, and deep packet inspection components for elevated security in the control and the data planes of SDMNs. Finally, the proposed security mechanisms are validated using test bed experiments.

INDEX TERMS 5G, SDN, NFV, security, mobile networks, monitoring.

I. INTRODUCTION

The evolution to 5G and future mobile telecommunication networks is characterized by a significant surge in demands in terms of performance, flexibility, portability, and energy efficiency across all network functions. Software Defined Mobile Network (SDMN) architecture integrates the principles of Software Defined Networking (SDN), Network Function Virtualization (NFV) and cloud computing to telecommunication networks. The SDMN architecture is designed to provide a suitable platform for novel network concepts that can meet the requirements of both evolving and future mobile networks.

The underlying principle of the SDN architecture is the decoupling of the network control and data planes. Using this principle, network control functions are logically centralized and the underlying network infrastructure is abstracted from the control functions. The introduction of NFV offers a new paradigm to design, deploy and manage networking services based on the decoupling of the network functions from proprietary hardware appliances, and providing such services on a software platform. However, the separation of control and data planes as well as the virtualization of network functions and programmability introduce a number of novel use cases and functions on the network. This will further usher in new stakeholders into the networking arena and hence will obviously alter the approach to security management in 5G and future telecommunication networks. Several proposals are available for securing general SDN networks [1]–[6] and SDMNs [7], [8]. However, none of these solutions provide a unified solution to secure future 5G SDMN backhaul

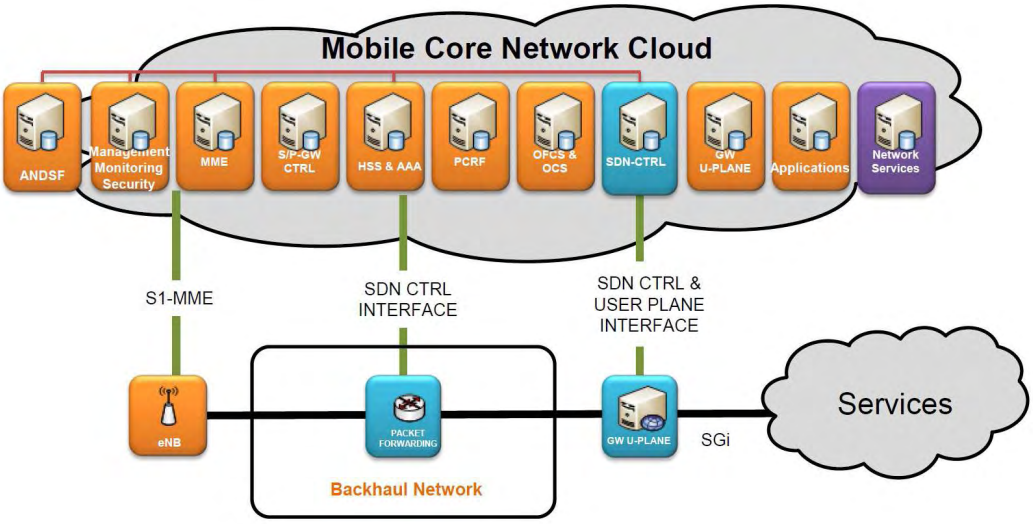

FIGURE 1. The consolidated SDMN architecture.

network. Therefore, it is needed to define a comprehensive security architecture for 5G SDMN networks.

In this article, we study the SDN and NFV based Software Defined Mobile Network (SDMN) architecture. Then, we highlight the topics related to security aspects for future SDN based mobile networks. We analyze the SDN, NFV and cloud technologies and view them as enablers for enhancing security of the new generation mobile networks, where virtual network architectures will be integrated. We further present the security challenges of the new telecommunication architecture which will arise while adapting these new technologies. Then, we propose a multi-tier component based architecture to secure SDMN backhaul network by tackling security issues, thus protecting the network and its users. The architecture defines security solutions to establish secure communication channels between network elements using Host Identity Protocol (HIP) and IPSec tunneling. Additional security mechanisms are implemented to prevent unwanted access to the mobile backhaul network using policy-based communications. Here, we propose that all flows to mobile users are admitted based on policy. The proposed architecture serves to protect the backhaul devices, the air interface and the mobile devices from source address spoofing and Denial of Service (DoS) attacks as well as supports the automation of tracing back attackers. Moreover, security assessment and awareness are gained with Software Defined Monitoring (SDM) and data collection to detect, prevent and react to security threats. Finally, Proof-of-Concepts (PoC) of proposed security components are verified in testbeds.

The paper is organized as follows; Section II presents an overall description of the SDMN architecture, it further provides a general overview of the security challenges in SDMN together with existing security solutions. Section III presents a description of the architectural framework of the proposed security solution. Section IV contains the testbed implementation and experiment results. Section V presents the discussion based on the outcome of the experiments. Finally, Section VI concludes the article and proffers future research directions.

II. BACKGROUND

A. SDMN ARCHITECTURE

SDMN architecture integrates the core principles of SDN, cloud computing, and NFV into a design of programmable flow-centric mobile networks providing high flexibility. This modification is of significant improvement to the current LTE 3GPP (3rd Generation Partnership Project) networks. It offers benefits such as a uniform approach to Best Effort and Carrier Grade services, centralized control for functions that benefit from a network wide view, improvement in flexibility and more efficient segmentation. It also provides an enabling platform for automatic network management, granular network control, elastic resource scaling and cost savings for backhaul devices. With SDMN, resource provisioning is done on-demand, hence allowing elastic resource scaling across the network [9]. With these attributes, SDMN becomes the latest innovation in the field of telecommunication [10], [11]. A consolidated illustration of the SDMN architecture is presented in Figure 1.

In this architecture, traditional legacy control functions which include the MME (Mobility Management Entity), the HSS (Home Subscriber Server), the PCRF (Policy and Charging Rules Function) and the control planes of S/P-GW (Serving/Packet Gateway) are all run as SDN applications atop the mobile network cloud. With this approach, the user plane will consist of SDN enabled switches and devices placed in strategic locations on the network [12].

SDMN applies to both LTE and 5G network. Currently, 5G is planned to meet the needs of both the consumer markets and new massive machine-to-machine communications with tailored support for ultra-high reliability applications. Based on current 5G standardization activities, the assumption is that the 5G core network will be based on SDN. It is also planned that the core network will be sliced for better isolation and tailoring to the particular requirements of the market segments. The exact set of network functions in each slice can vary.

B. SECURITY THREATS IN SDMN

As an ever growing share of Internet use is over mobile networks [13], inherent Internet threats such as ease of Denial of Service (DoS) attacks, source address spoofing and distribution of malware apply to mobile networks as well. Similarly, SDN and NFV have their own security limitations as described in [12] and [14], and deploying these concepts in mobile networks without considering their inherent limitations will further elevate the security challenges. Hence, the separation of planes, aggregating the control functionality to a centralized system and running the control functions in the cloud as in SDN will open new security challenges for SDMNs. For instance, the communication channels between the isolated planes can be targeted to masquerade one plane for attacking the other. The control plane is more vulnerable to security attacks, especially to DoS and DDoS (Distributed DoS) attacks, because of its centralized nature and global visibility and can become a single point of failure [14]. Since the networking paradigm of future mobile networks is converging towards software-based networking, operational malfunctioning or malicious software can compromise the whole network by getting access to the control plane [15]. Some of the known security challenges in SDMN are summarized in Table 1.

C. RELATED WORK

Since, SDN is considered to enable innovation in communication networks, bring flexibility and simplify network management, research efforts are going on for the deployment of its concepts in mobile networks. From security perspectives, SDN will enhance network security for two main reasons. First, it centralizes the network control plane that will provide global visibility of the network state and traffic behavior. Second, SDN brings programmability into communication networks through programmable APIs in the data forwarding elements. These two aspects enable SDNs to facilitate runtime network monitoring with quick threat identification, faster response systems, easy security policy alteration, and fast security service insertion without individual device configurations [14]. Therefore, several security systems development proposals for SDN-based networks are proposed such as FRESCO [1], FSL [16] and splendid isolation [17]. These mechanisms can be used to develop mobile network specific security techniques. Various approaches are also suggested to secure SDNs due to its inherent limitations as discussed in the previous section. These technologies and proposals are listed in Table 2 which presents those security solutions with the type of security and the target SDN plans and interfaces.

There are several proposals for securing SDN based mobile network architectures from a particular security threat. For example, [7] proposes vulnerability assessment methodology

TABLE 1. Summery of security threats in SDMN architecture.

SDMN Layer	Type of Threat	Threat Reason and Description
Application	Lack of authentication and authorization	Possible huge number of (third-party) apps
	Fraudulent rules insertion	Malicious applications generated false flow rules
	Access control and accountability	Lack of binding mechanisms for apps
Control	DoS, DDoS attack, Controller hijacking or compromise	Visible nature of Ctrl-plane
	Unauthorized controller access	No compelling mechanisms for enforcing access ctrl on backhaul devices.
	Privacy of communications	Attacker with access to controller can command to fork any flow at any point to a VNF funtion anywhere where it can analyze the content breaking confidentiality of communications.
	Scalability or availability	Centralized intelligence
Data	Fraudulent flow rules	Lack of intelligence
	Flooding attacks	Limited capacity of flow tables.
	Controller and DP switch masquerading	Lack of strong authentication
Ctrl-Data Interface	TCP-Level attacks	TLS is susceptible to TCP level attacks.
	Man-in-the middle attacks	Optional use of TLS and complexity in configuration of TLS
App-Ctrl Interface	Illegal controller access, policy manipulation and fraudulent rule insertion	Limited secure APIs, lack of binding mechanisms b/w Apps and controller.

TABLE 2. Proposed security mechanism for general SDN networks.

Security Type	Reference	SDN Layer/Interface
Threat detection and mitigation	[1]	Ctrl, App-Ctrl
App debugging, flow rules inspection	[18]	App, Data
Flow rules verification, Configuration verification	[19] [20]	Ctrl, Data
Flow policy verification, catch bugs in OF programs	[21]	App, Ctrl
App testing and debugging	[22]	App
Conflict resolution, authorization, security audit system	[23]	Ctrl, Data, App-Ctrl
DDoS detection, Controller resilience	[24] [25]	Ctrl, Data
Link monitoring	[2]	Data, Ctrl Data
Find contradictions in flow rules, authorize applications	[26]	Ctrl, Data
Controller availability, network monitoring	[3] [4]	Ctrl, Ctrl Data
Access control and dynamic policy enforcement	[5] [6]	Ctrl, Data

for SDN based 5G network architectures. Similarly, [8] proposes leveraging SDN to strengthen authentications security and protect privacy during handovers in 5G networks. The proposed mechanisms in [8] also simplifies the handover authentication in heterogeneous 5G networks leveraging on global visibility attained by the centralized control platform of SDN. However, there is no unified solution to the future 5G mobile networks that provides security to the whole backhaul and the core networks along with the transport channels.

The SIGMONA project [27] proposed SDN-based mobile network architecture. Then telecom-specific security requirements which gathered for a consolidated security architecture that efficiently secures the whole SDN-based mobile network that we call SDMN [11], [28]. This paper presents the SDMN backhaul security architecture proposed in SIGMONA with its validation results.

D. SECURITY REQUIREMENTS FOR SDMN

In addition to challenges from new technologies: SDN and NFV, the growing popularity of smartphones, rising mobile broadband volume and sophistication of malware exposes the mobile-terminals and their networks to the attacks of the fixed networks, such as source address spoofing, unwanted flows, malicious traffic and DDoS. However, compared to their fixed counterparts, i.e. laptops and desktops etc., mobile terminals are constrained by computing resources, storage and battery lifetime. This is even truer for some of the new devices envisioned to connect under 5G, such as sensors etc. which could be even resource constrained [29]. This deters deploying the host-based security solutions on the wireless hosts. Moreover, the host-only security would leave the backhaul network and radio interface unprotected against hacks, malicious flows and unwanted traffic from the Internet, taking a significant toll on network performance.

Taking a fresh look at the end-to-end principle, we state that a function that is not feasible in the end-hosts shall be left to the network. The new technologies and planned enhancements in the core network can significantly contribute to the security of the network as well as its users. For example, relying on the principles of SDN and SDMN can leverage the global visibility of the SDN controller on the underlying network to: a) enforce consistent security policies across the network; b) fine-grain handling of individual user flows and new flows in network; and (c) to dynamically react to evolving threats by forwarding updated firewall rules to the data-plane nodes.

To address these Internet threats, we argue that future mobile networks should:

- Limit the flow acceptance to verifiable sources, to tackle the problem of unwanted traffic, source address spoofing, and thus prevent resource exhaustion.
- Eliminate source address spoofing to attribute the evidence of misbehavior to the sender.
- Make it possible to aggregate misbehavior evidences under a stable source identity, and contribute towards using reputation mechanisms for improving the security of communicating entities.
- Under network stress, grant resources based on source reputation.
- Allow defining dynamic (reachability) policies for hosts, applications and services. The management and control of the policies will be in the cloud while enforcement takes place in standard data-plane nodes on trust boundaries. This is in contrast to the current mobile networks where policies are tightly coupled to physical resources and are not scalable to services/applications.
- Leverage the logically centralized controller to overview, analyse and manage the policy configuration of data-plane elements, in order to deploy a robust and consistent security policy across the network.

In addition, the deployment of existing and new mechanisms to SDMN requires that they are implemented and tested for their compliance to SDN principles, because existing solutions could be difficult to deploy, manage and scale to secure SDMNs. We argue that the security solutions should:

- Optimize the network resource utilization for security functions.
- Leverage the existing research/work in network security to harden SDMNs against classical Internet attacks.
- Limit all the changes to network edges, and not require any mandatory changes in the end-hosts or protocols, to minimize the deployment challenge.

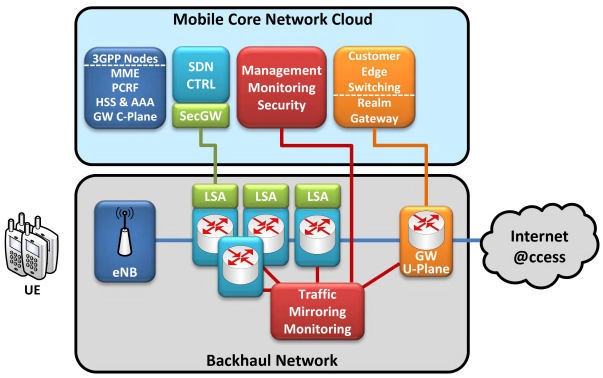

FIGURE 2. The proposed security architecture for SDMN.

III. PROPOSED SECURITY ARCHITECTURE

Given that most of the requirement specific to telecom architectures are tightly coupled with the control and data planes than with the application plane [9], [12], hence, the proposed security architecture is geared towards securing the control plane, data plane and the Ctrl-Data interface (southbound interface). Figure 2 presents the proposed security architecture for SDMN networks.

The proposed SDMN security architecture is a multitier security approach with five components, namely;

1) Secure Communication (SC) Component.
2) Policy Based Communication (PBC) Component.

3) Security Information and Event Management (SIEM) Component.
4) Security Defined Monitoring (SDM) Component.
5) Deep Packet Inspection (DPI) Component.

A. SECURE COMMUNICATION (SC) COMPONENT

The SDMN architecture comprises of two main communication channels, the data and control channels. The data channel handles the transportation of the user communication data while the control channel handles the movement of essential control and signaling data between the data and control planes.

The major security concerns in SDMN communication channels are the lack of IP-level security and weak authentication between backhaul devices as shown in Table 1. Existing SDMN communication channels are heavily reliant on higher layer security mechanisms like TLS (Transport Layer Security) /SSL (Secure Sockets Layer). A typical example is the widely used OpenFlow protocol which runs over a TLS/SSL based control channel [30]. However, such higher layer security mechanisms offer no protection to information at IP levels. This leaves the communication sessions vulnerable to IP based attacks such as TCP SYN DoS, TCP reset attacks and IP spoofing [31], [32]. In addition, the TLS/SSL authentication mechanism is also exposed to IP spoofing and Compression Ratio Info-leak Made Easy (CRIME) attacks [31]. These vulnerabilities buttress the need for secure communication mechanisms in SDMN architecture so as to mitigate against such threats.

To secure the communication channel of the SDMN architecture, we propose a HIP (Host Identity Protocol) based secure IPsec tunnelling architecture, this architecture helps to establish secure HIP tunnels between the controller and the DP (Data Plane) switches. The latest IP based telecommunication network (i.e. LTE/LTE-A) operators are heavily relying on IPsec tunneling and security gateway mechanisms to protect their backhaul traffic. Several versions of IPSec key exchange mechanisms are available such as Internet Key Exchange version 1 (IKEv1), Internet Key Exchange version 2 (IKEv2), IKEv2 Mobility and Multihoming Protocol (MOBIKE), Host Identity Protocol (HIP). However, it is not possible to implement these legacy IPsec tunneling mechanisms in SDMNs due to several limitations, such as distributed tunnel establishment, lack of centralized controlling, Point-to-Point (P2P) tunnel establishment, per tunnel encryption key negotiation, limited security plane scalability, lack of visibility, lack of access control and static tunnel establishment. The propose security mechanisms overcame these identified limitations.

Figure 3 illustrates the secure HIP tunnel establishment under the proposed SC component.

Here we propose three key modifications to existing SDMN architecture. First, we introduce distributed Security Gateways (SecGWs) for securing the controller from outside network and mitigating against the odd of a single point of failure. SecGW is the intermediate gateway device between

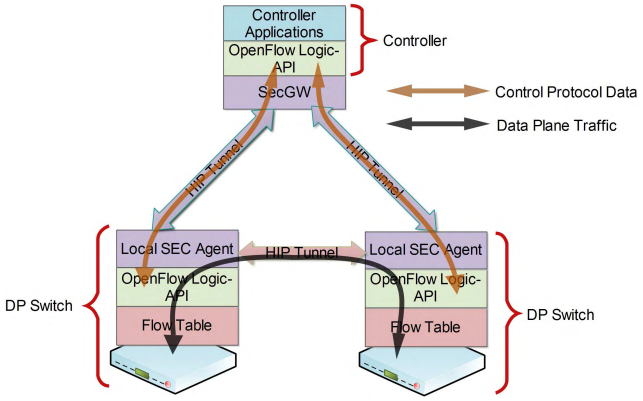

FIGURE 3. Secure communication channel.

the controller and the data plane switches. It not only hides the controller but also reduces the security work load on the controller. Second, we added a new Security Entity (SecE) to control SecGWs and other security functions in SDMN. Third, we installed local Security Agent (LSA) application in each data plane switch to manage security related functions in the switch. The proposed solution is a bump-in-the-wire mechanism and it does not affect the underling control protocols like Openflow nor other user-plane communication protocols. In Table 3, we compare various features of the proposed SC component with other security solutions.

TABLE 3. Comparison of proposed architecture with existing security mechanisms.

Property	TLS/SSL	IPSec with IKEv2	IPSec with HIP	Proposed Architecture
Vulnerability of mutual authentication mechanism	Medium	Medium	Low	Low
DoS attack prevention	No	No	Yes	Yes
Support for seamless mobility of backhaul nodes	No	No	Yes	Yes
Multihomed Support	No	No	Yes	Yes
Centralized Controlling	No	No	No	Yes
Point-to-Multipoint/ Multipoint-to-Multipoint	No	No	No	Yes
Visibility of traffic transportation	No	No	No	Yes
Access Control	No	No	No	Yes
Collaboration with other control entities	No	No	No	Yes

B. POLICY BASED COMMUNICATION (PBC) COMPONENT

The best-effort paradigm of the current Internet allows a host to initiate flows towards any destination address. Hackers often abuse this paradigm to launch attacks to their victims. Hiding under a spoofed address, hackers often bypass security mechanisms and also prevent tracing the attack back to its originator. The best-effort principle attempts its best to deliver the packets of the sender to the destination. However, because interests of the receiver do not always align with the

sender [24], [25], the destination receives unwanted traffic. Since SDMN proposes an all-IP based open network architecture, it is also vulnerable to unwanted traffic, DoS and source address spoofing [15], [33] similar to other IP networks such as Internet [12].

Cooperation is a proven mechanism to effectively curb the antisocial behavior in a population [34]. We propose a two-tier cooperative approach to improve SDMN security and limit the extent of damage from Internet malpractices. The goal is to: 1) mitigate traditional attacks on SDMNs, i.e. DoS and source address spoofing; 2) encourage cooperation of all benevolent entities against the malicious sources; and 3) tracing as well as containing all the resources used by the hacker in attacks. First tier is achieved by establishing the required level of edge-to-edge trust using Customer Edge Switching (CES). The second tier involves the ubiquitous collection and attribution of the attack evidences within a trust domain. CES nodes will then use the consolidated evidences to black and grey list remote entities.

CES allows policy-based communication to mobile hosts. A CES node in principle replaces NAT at the network edge, and extends the classical stateful firewall into cooperative firewall. CES acts as a secure connection broker for hosts located in its network, and in contrast to the classical Internet matches the interest of the sender with the receiver prior to flow admission, or before forwarding a new flow on the Internet. The interests of the end-hosts are expressed as a policy, which could require stable source identity, verification of the sender credentials, i.e. via reverse DNS (Domain Name Server) lookup to more complex certificate validation, as well as the possibility of utilizing the private-transit links, not exposed to the public Internet. The negotiation of interests between CES firewalls for respective hosts effectively extends the classical statefull firewall functionality into cooperative firewall.

For mobile networks, CES offers many advantages: (a) end users will benefit from a network firewall in the cloud, instead of relying only on host-based security solutions on the mobile device for blocking unwanted traffic and common attacks. This (b) saves computing resources of the device; and (c) contributes to battery lifetime of wireless device, by preventing unwanted traffic from reaching to device and disturbing its sleep cycle; besides preventing d) cluttering of air interface and network. CES does not require changes in end protocols, applications, or any explicit signaling from hosts to maintain their network connection: NAT bindings, or connection states. The policy-based communication facilitated by CES means that all flows to the mobile hosts are admitted based on policy. Prior to admitting a flow, the outbound CES (oCES) node and the inbound CES (iCES) node will negotiate policies of the respective hosts via Customer Edge Traversal Protocol (CETP). In case of a policy matching, the connection state is inserted into data-plane to admit the user connection. Figure 4 shows the deployment of proposed PBC component, (i.e. CES) which runs as an SDN application on top of the SDN controller.

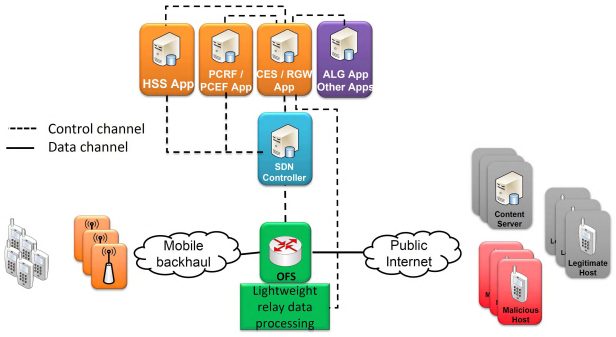

FIGURE 4. SDN oriented customer edge switching.

CES retrieves the necessary user policies and certificates from mobile network components, such as HSS or PCRF on demand. Upon successfully negotiating policy, CES interacts with the data-plane to insert the negotiated flows into the OpenFlow switches, and hence allows the end-to-end user communication.

CES provides backwards compatibility with legacy networks using Realm Gateway (RGW). For outbound connections, the RGW function is similar to NAT, however it admits inbound connection following a domain name query from the Internet towards a private host, leading to creation of a NAT binding and granting connectivity. We implemented a number of heuristic mechanisms to secure the interaction of the legacy Internet with CES/RGW. For DNS, these include classifying DNS servers into white, grey and black lists based on: service-level agreements, influx of DNS floods and resource assigning model. Whitelisting can be based on spoofing-free DNS/TCP channel, SLAs and use of private-links between networks. The address allocation in the realm gateway is controlled by policy to limit or deny the resource allocation to any host, server or a given network. Upon a new valid host name query, the address allocation algorithm creates a half-open state with respect to sender which is elaborated into a full inbound NAT binding upon the first inbound SYN from sender. In the full binding state, the RGW applies the address and port dependent filtering on incoming packets relative to the client. We also define Service-FQDN to address the services running on end-hosts; this not only contributes to security but also increases the scalability of RGW. The naming scheme allows a more strict NAT binding by virtue of adding a port number to the half-open state, making it more resistant to attacks and improving the reuse of the (pool of) inbound public addresses.

RGW employs the TCP-Splicing mechanism to eliminate spoofing in the user connections admitted to the network. For this it leverages the SYN cookie algorithm to postpone the allocation of TCP resources until the sender is determined as non-spoofed. Consequently, the network is protected against spoofed flows, and resources are assigned to valid sources. We also leverage the SYN cookie [35] algorithm to implement a bot-detection method that detects and mitigates SYN floods from non-spoofed malicious sources, repeatedly

targeting the temporary half-open connection states or bindings in RGW.

In summary, the proposed PBC component offers several benefits to its adoption. Since it uses standard DNS requests as communication trigger and it can be deployed transparently to end hosts, i.e. as it requires no changes in end-hosts, protocols or applications. It offers a light-weight, host independent NAT (Network Address Translation) traversal solution to admit the inbound connections. The centralized operations of SDN can contribute to more fine-grained and informed decision making of the heuristic algorithms, contributing to a more secure environment in SDMN.

C. SECURITY INFORMATION AND EVENT MANAGEMENT (SIEM) COMPONENT

Network monitoring solutions come in different variants depending on what they measure and how they collect the data:

1) Active Probing: service-centric approach that collects data based on synthetic measurements such as ICMP Echo Requests, HTTP GET requests or specially crafted packets. Often these measurements attempt to measure properties of the network that would be impossible to capture from pure passive measurements and are arguably the only way to measure service availability.
2) Device Polling: device-centric approach that queries devices typically using SNMP (Simple Network Management Protocol), collecting interface status information, traffic volumes, device load, CPU, etc.
3) Flow Collection: solutions that collect traffic information from network devices such as routers/switches; traffic is aggregated in flows using e.g. Cisco Netflow [36] and stored in disk for post-analysis. Flow data is easier to analyze and process than packet data, but provides less granular information.
4) Packet Analysis: usually involves a SPAN port from a switch or a network tap and extracts information from individual packets, including information from payloads through DPI.
5) Log Analysis: solutions that collect machine generated data typically in the form of log files (e.g. syslog) and present a query interface to correlate events across different types of systems, e.g. routers, web servers, load balancers.

Security Information and Event Management (SIEM) component collects flow information and takes the security ensuring actions such as access control list update, firewall update, flow table modification, rate bound enforcement and so on. SIEM provides Security Information Management (SIM) on the one hand, and Security Event Management (SEM) on the other hand. SIEM component also collects event data from network infrastructures, applications and security devices. Although the SIEM component uses log data as the primary data source, it can also generate other forms of data such as NetFlow and packet capture. Data from such events are combined with other contextual information regarding the users, assets, threats as well as other perceived vulnerabilities. To ensure that data and events are correlated, contextual information from disparate sources are normalized and analyzed for specific purposes, be it monitoring of user activities, monitoring of network security events or compliance reporting. The SIEM component performs real-time security monitoring and historical analysis. It also provides support for investigating incidents and providing reports on performance. It also contains a security monitoring and event management that perform an analysis of security event data in real time focused on network events, and present security information in a consolidated Graphical User Interface (GUI).

Figure 5 shows a high level overview of the SIEM component described above:

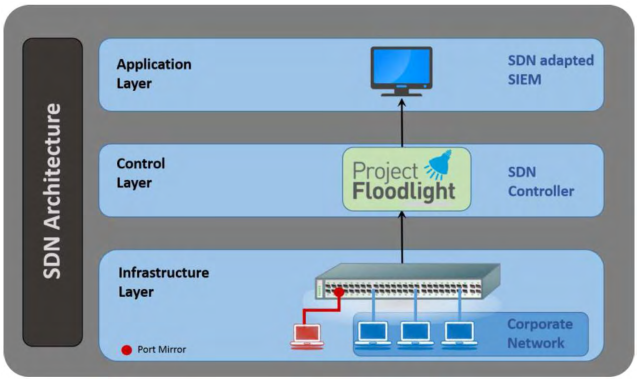

FIGURE 5. Security information and event management (SIEM) component.

It mainly consist two components:

- Security Sensor (Port Mirror): responsible for gathering security information (i.e. through IDS (Intruder Detection System) [37] features for security event detection) and reporting to the security server.
- Security Server (SDN Adapted SIEM): responsible for collecting security information coming from deployed security sensors. The collected information is correlated and validated against predefined security policies for a final decision making. It optionally allows an automatic reaction to detected security events.

Therefore, proposed SIEM component offers the following features:

- Security Management features:
  - Security policies definition
  - Countermeasures definition
- Security Monitoring features:
  - Asset inventory
  - Availability monitoring
  - Network monitoring (usage and latency)
  - Vulnerability discovery
  - Event detection (intrusion, anomalies, etc.)

D. DEEP PACKET INSPECTION (DPI) COMPONENT

SDMN enhances security by making it easier to implement counter-measures and isolate network parts when security

problems are detected. On the other hand, additional software, components and interfaces required in SDMN open new opportunities for attacks by malicious agents. Security needs to be addressed on the network side as well as the mobile device side. Deep Packed Inspection (DPI) as part of Network Intrusion Detection Systems (NIDS) strengthens network security by detecting and tackling harmful traffic flows.

In SDMN, it is crucial that both applications and associated control elements are constantly aware of the conditions of the underlying infrastructure so as to guarantee optimal security at different levels. This is central to the overall performance of the network and can well be handled by the DPI. The DPI can routinely gather network information and channel it back to the control layer.

As illustrated in Figure 6, the proposed DPI component is a part of an active monitoring probe that detects security threads. It is also able to react to detections. Optionally, the DPI component can work with mirrored traffic as a part of network monitoring functions. In both cases, the DPI component can be virtualized. This component will be adapted to analyze and detect diverse security threats related to application flows matching with predefined malware rule databases. The developed solution concentrates on the analysis of HTTP application flows. The detections are written to the local database and optionally provided directly to the other network functions. Currently, the proposed DPI component is not compatible with HTTPS flows. However, we assumed HTTPs traffics are protected at the application plane.

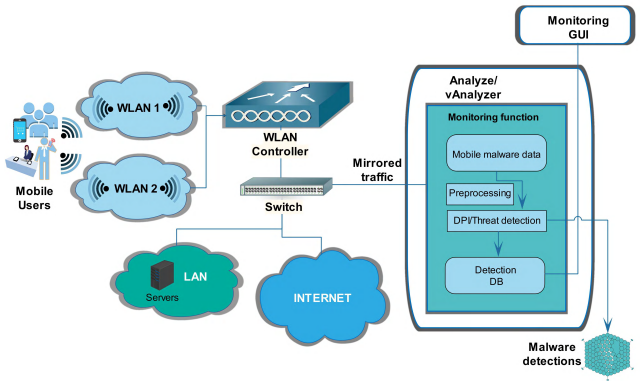

FIGURE 6. Deep packet inspection (DPI) component.

E. SECURITY MANAGEMENT AND MONITORING (SMM) COMPONENT

SDM component is designed to perform monitoring functions in SDN/NFV-based 5G mobile network architectures. It is able to monitor both virtualized and physical network environments in an economical and efficient way. Initially, the proposed SDM architecture is used only to monitor SDMN backhual networks. However, the proposed SDM architecture extend the capabilities of current SDN/OpenFlow features to provide the required level of monitoring capabilities in 5G

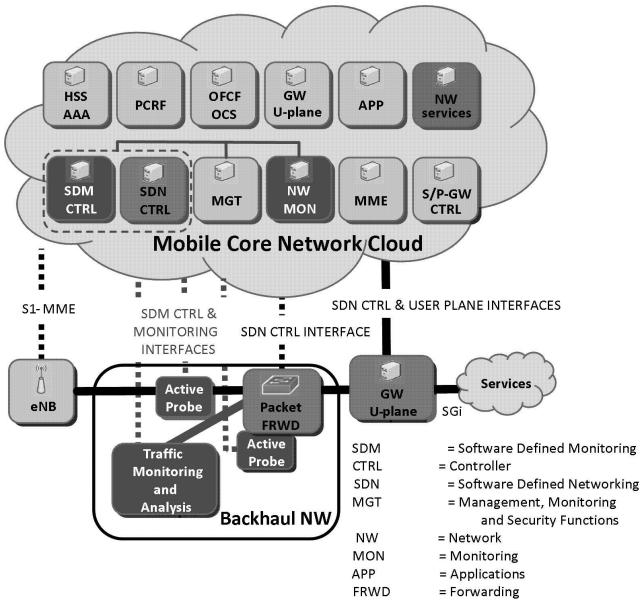

FIGURE 7. Security management and monitoring component.

backhual networks. Figure 7 illustrates how SDM component is implemented in the 5G SDMN architecture.

The following modules and interfaces have been added to the SMM components.

- Modules:
  - Security sensor: an active monitoring probe used to detect information related to security and anomalous behaviors on the network. It also tries to mitigate detected attacks through the use of mechanisms such as filtering. Information collected by the security sensor may include general security properties and attack reports. Security sensor can be installed on the network elements or in network taps (passive network observation points).
  - SDM CTRL: an extension of SDN CTRL which allows the control of monitoring functions such as management of network monitoring appliances, traffic mirroring, traffic load balancing and aggregation. This module also attends to requests from network functions and applications. SDM CTRLs are usually distributed following either a peer-to-peer or hierarchical model. They inter-operate with the management/monitoring/security function and act as distributed analysis or decision points for the defined security policies (security SLAs).
  - Network monitoring: a virtual monitoring module which extends part of the traffic analysis to the cloud.
  - Traffic mirroring and analysis: a passive traffic monitoring device located at the backhaul to monitor variety of network functions.
- Interface:
  - SDM CTRL Interface: controls the use of monitoring resources, recuperating traffic or metadata for analysis. This interface allows monitoring

requests to be sent to ascertain the status of the network, hence enabling applications and network functions to send requests for monitoring-based information, and monitoring functions can send status and recommendations.

The SMM component introduces a dedicated Software Defined Monitoring controller (SDM CTRL) to orchestrate the monitoring activities related to security that are performed by sthe ecurity sensors (i.e., probes) deployed in the network and in the cloud. The SDM and SDN controllers can be separate modules or integrated into one module. The SDN CTRL also interacts with the routers implementing the SDN CTRL interface to manage the traffic (e.g. redirect traffic to the security appliances) and recuperate certain information. The SDM controller interacts with the security appliances or probes implementing the SDM CTRL interface to manage them and recuperate the metadata part of the traffic or verdicts. This information can be used by the network monitoring function in the cloud to perform analysis and trigger mitigation actions; of by the other network functions/services and applications.

Security sensors used in SMM component can be passive (not disrupting traffic) or active (in the data path to perform online countermeasures); and can either be installed in existing network elements or in dedicated security appliances. The probes analyze network traffic, correlate information from different sources and produce meta-data and verdicts that can then be used by a centralized decision point and by the different network functions.

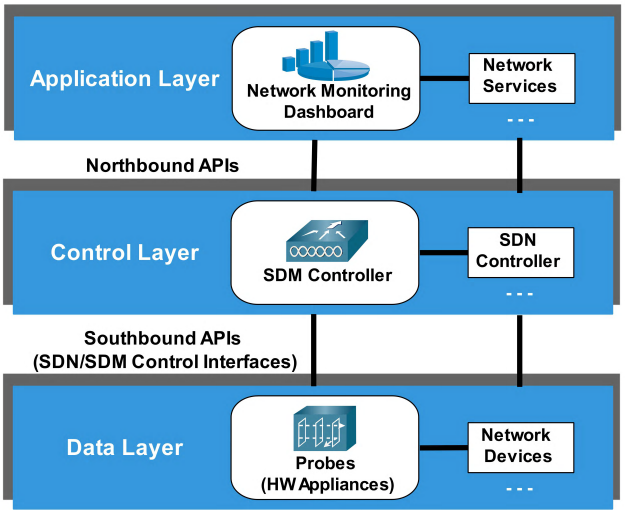

FIGURE 8. Security management and monitoring (SMM) component in three layer SDN architecture.

Figure 8 shows how the components of the proposed architecture map to the three-layer mobile SDN architecture proposed by Open Network Foundation (ONF) [38].

IV. TESTBED IMPLEMENTATIONS AND RESULTS

Here, we implement a proof-of-concept prototype on a testbed for the components of the proposed architecture in four sets of experiments. We then provide a performance evaluation for each component.

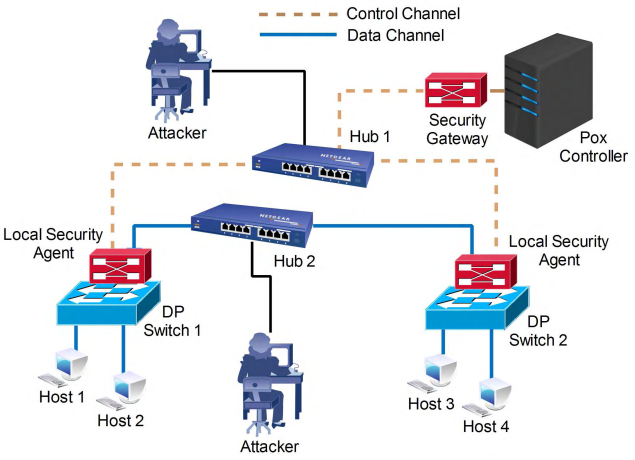

FIGURE 9. The layout of the experimental testbed for secure communication (SC) component.

The first set of experiments was for the secure communication component. In this experiemnt, we evaluated the performance penalty of this component in terms of throughput, jitter and latency. We further measured the capability of the proposed architecture to protect the communication channels against common IP based attacks like TCP SYN DoS and TCP reset attacks. We used OF protocol [39] with TLS/SSL session as reference for the control channel. Figure 9 illustrates the preliminary testbed components for this experiment.

As shown in this figure, the testbed contains two Data Plane(DP) switches, an SDN controller and two hubs. We used the latest version of POX controller [40] as the SDN controller and OpenVswitch (OVS) version 1.10.0 virtual switches [41] as DP switches. We have used four virtual hosts as users. For each OVS, two virtual hosts were connected. We used two D-LINK DSR-250N routers to connect the controller and the switches. For this experiments, we kept out-band control channel. We modeled the Security Gateway and LSAs using OpenHIP [42]. We used IPERF network measurement tool [43] to measure the performance in terms of throughput and latency. We finally connected an attacker to each hub for each scenario of the experiment, the attacker operates from an i5-3210M CPU of 2.5GHz processor laptop.

TABLE 4. The simulation settings for the IPERF.

Parameter	Value	Value
Protocol	UDP	TCP
Port	5004	5004
Buffer size	default (1470 kB)	default (1470 kB)
Packet size	default (1470 B)	default (1470 B)
TCP window size	-	21.0 KByte
Report interval	1 s	1 s

The experiment settings of IPERF testing tool is presented in Table 4.

TABLE 5. Data channel performance without attack (normal operation).

Performance Metric	Existing SDMN Data Channel	Proposed Secure Data Channel
TCP Throughput (Mbps)	93.5514	91.8054
UDP Throughput (Mbps)	95.2845	92.3828
Latency (ms)	36.6514	37.6452
Jitter (ms)	0.34522	0.4651

Table 5 presents the performance of the data plane under each architecture. We ran each experiment for 500 seconds and recorded the average values of the outcome.

The experiment results presented in Table 5 indicated about 2% decrease in TCP and UDP throughputs for the proposed secure channel. In addition, we observed a 3% increase in latency when compared to existing SDMN data channel. This reduced performance of the network is caused by the extra layer of encryption added to the proposed secure channel. Notwithstanding, the addition of IPSec accelerators can help to further boost the performance of this architecture and minimize the deficiency caused by the extra layer of encryption. More recent Intel processors are capable of supporting such IPsec acceleration leveraging on external accelerators and/or using new AES (Advanced Encryption Standard) instruction sets [44].

For the next experiment, we added a TCP SYN DoS attacker to the data channel (Hub2). We ran each experiment for 500 seconds while launching attacks between 100 and 200 seconds time interval. Table 6 shows the average performance of each architecture.

TABLE 6. Data channel performance under TCP DoS attack.

Performance Metric	Existing SDMN Data Channel	Proposed Secure Data Channel
TCP Throughput (Mbps)	72.1945	91.51564
UDP Throughput (Mbps)	74.4656	92.4551
Latency (ms)	548.14854	37.5146
Jitter (ms)	5.1495	0.4301

The outcome of the experiments recorded in Table 6 clearly shows the vulnerability of current SDMN channel to TCP DoS attacks. The experiment results show a 20% drop in throughput for both TCP and UDP in current SDMN data channel. The percentage drop in throughput is directly proportional to the percentage of time during which attacks were launched in reference to the overall experiment duration. We therefore conclude that current SDMN data channel is highly vulnerable to DoS attack, given that the effect of the attack on throughput lasted for the whole duration of the attack. Moreover, our experiment results also showed a 14 times increase in both latency and jitter using current SDMN data channel compared to normal operation. However, using the proposed secure channel, we experienced similar performance as in normal operation. Thus, we verify that the proposed secure channel is capable of securing the SDMN data channel from potential DoS attacks.

TABLE 7. Control channel performance under TCP DoS attack.

Performance Metric	OpenFlow with TLS/SSL	Proposed Secure Control Channel
Connection Establishment Delay (ms)	58.3224	135.4165
Connection Establishment Delay under TCP SYN DoS Attack (ms)	–	135.9145
Flow Table Update Delay (ms)	30.85645	32.1573
Flow Table Update Delay under TCP Reset Attack (ms)	–	32.2472

In the next experiment, we orchestrated a TCP SYN DoS and Reset attack on the control channel (Hub1). We then recorded the connection delay and flow table update delay experienced between the controller and the DP switch 1. We ran each experiment 25 times and recorded the average performance of each architecture. Table 7 shows the outcome of this experiment.

The experiment outcome presented in Table 7 shows a significant increase in connection establishment delay using the proposed secure channel. We observed an additional latency coming from the extra HIP tunnel establishment between LSA and SecGW. We also observed a 4% increase in flow table update delay using the proposed secure channel under a steady state of operation (i.e. after establishing the connection). This deficiency in performance comes from the extra layer of encryption when using the proposed secure channel.

The experiment results shown in Table 7 also shows how vulnerable the existing SDMN control channel is to TCP DoS and reset attacks. We observed that it was not possible to establish connections with the controller during TCP SYN DoS attacks and during the TCP reset attack it was not possible to update the flow tables. However, using the proposed secure channel, we observed consistent performance, hence resistant to those attacks. This confirms the ability of the proposed secure channel to secure the SDMN control channel from IP based attacks.

In the second set of experiments, we aim to measure the performance of the PBC component, as well as determine the strength and effectiveness of its security. The proposed PBC component consists of CES/RGW and is implemented in Python. The prototype is developed as CES proof-of-concept. Figure 10 presents the implementation of our CES testbed, which is built in Linux environment and employs control/data plane split architecture. The testbed has two private networks that are respectively served by CES-A and CES-B. The edge of each network bears a data-path element, which enforces the rules generated by the control plane and forwards the new flows towards the CES function at the control plane, i.e. for connection admission, policy negotiation or security analysis. The nodes in the setup are implemented as linux containers and are connected using basic Linux networking support. Tests were conducted with a total of 8 simulated hosts.

Each network edge has two external interfaces: a) a public interface to receive traffic from the legacy Internet; and

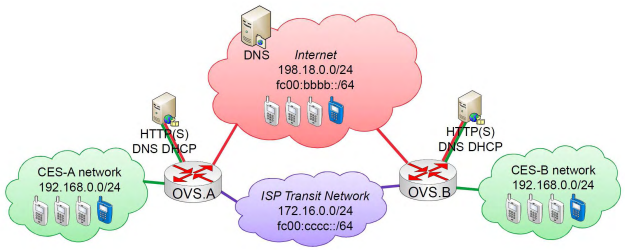

FIGURE 10. Layout of the experimental testbed for policy based communication (PBC) component.

b) a more secure private-transit link to receive flows from hosts in a different CES network. The setup also contains few nodes in the legacy Internet to launch typical Internet attacks or abuses for testing the CES/RGW security. Under this setup, the PBC is tested for both its use cases: 1) policy-based communication between CES nodes; and 2) inter-operability with legacy IP networks, using Realm Gateway (RGW) functions.

TABLE 8. Security testing of CES policy negotiation.

Metrics	Testing Results	
Signaling round trips (RTTs)	1-RTT	2-RTT
Connection establishment delay (msec)	80	145
Proof-of-Work sender's delay (msec)	3	
Proof-of-Work receiver's delay (msec)	0.001	
Certificate/Signature computation (msec) 1st packet	2	
Certificate/Signature verification (msec) 1st packet	1.8	

Table 8 presents results of CES security testing, in terms of the processing delay of the security mechanisms. The use of proof-of-work mechanism allows CES to push the burden of communication to the sender, such that sender invests more computing cycles than the receiver. This also effectively eliminates source address spoofing in the admitted flows, failing spoofed sources from leaking traffic into the private network.

The use of CES certificates (at CETP layer) coupled with signed CETP header is used to authenticate the remote node as CES. The mechanism leverages an object identifier in X.509 certificate to uniquely define CES certificates, and identify the remote node as CES (possessing the private-key) due to signed CETP headers. The mechanism is triggered upon the first flow from a new source and ascertains if the remote node is a valid CES. The testing revealed that only flows from valid CES nodes are admitted into the network. A CES node based on its policies decides whether to accept an inbound flow or request the sender for additional details, which may result in another round of policy exchange. The negotiation of policies completes in either one or two round trips and results in either: a) success; or b) failure depending on policies. The subsequent connections from the sender reutilize this validation result.

Having negotiated the CES policies, the subsequent flows from the sender only undergo one or two round trips of the host-to-host policy negotiation. A typical host-to-host user flow establishes after 80 msec or 145 msec delay incurred by 1-RTT or 2-RTTs of the host-policy negotiation, respectively. However, due to additional round of CES-policy exchange on the first inbound flow from the sender, the first host-to-host flow establishes in 220 msec for 1-RTT and 300 msec for 2-RTTs of the host-policy negotiation. Since we measured the connection setup on zero-latency links, one must add edge-to-edge latency of the real networks to get the actual connection setup delay. To account for network uncertainties, CES state machine can absorb any host retransmissions while the CETP process is still concluding.

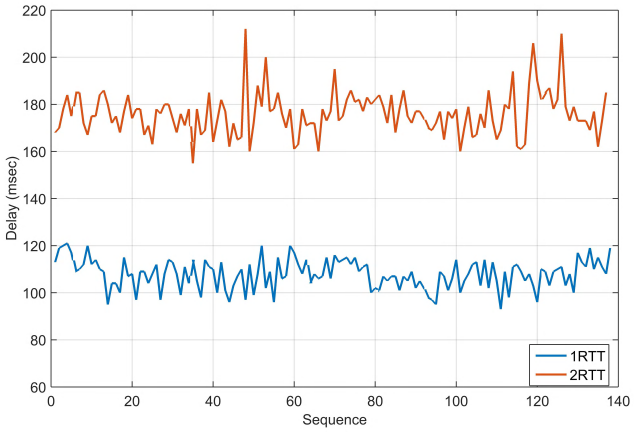

FIGURE 11. Delay induced by CETP policy negotiation on forwarding of the first packet of the user-data connection.

Figure 11 illustrates the connection setup delay of nearly 80 connections, using CETP host policies of varying complexity. The figure reveals that less complex policies are negotiated quicker than more complex policies that result in another round trip. Most of the presented delay in the figure is the due to slow control/data plane interaction, while the policy processing by CES is carried out in the order of milliseconds. In future, we aim to improve the CES-to-CES signaling by direct CES-to-CES control plane communication, and then synchronizing the negotiated user connection to the data-plane.

Figure 12 shows the impact of resource allocation model on RGW on the event of a DNS flood. The model prevents the exhaustion of the address pool resources by rate limiting the DNS sources and by limiting the resources available to grey-listed DNS servers. By default, the servers that do not meet the SLA defined for trusted sources are greylisted. This results in higher availability of address pool resources to legitimate DNS servers and clients, particularly under load conditions. Our testing of RGW revealed that TCP-Splicing completely eliminated spoofing, and no spoofed source could leak traffic into the private network or claim a user connection. A future version of the prototype aims to employ SYN proxies instead of TCP-Splice, since they are optimized to handle millions of packets per second.

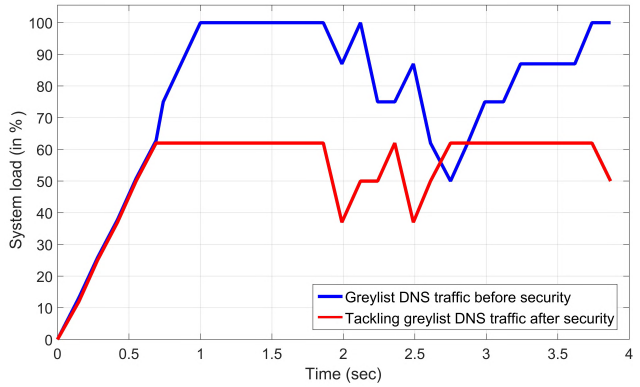

FIGURE 12. Tackling DNS flood from greylisted DNS servers.

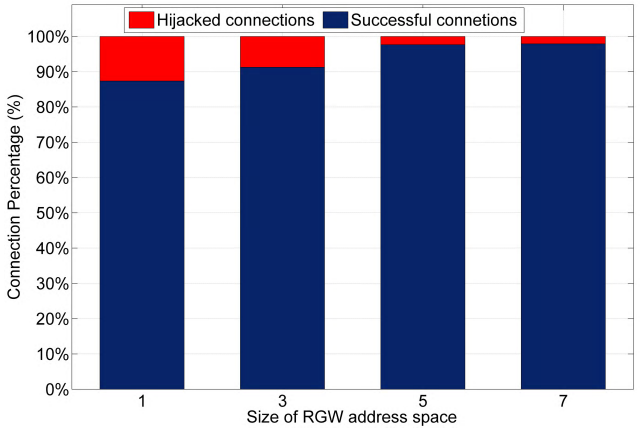

FIGURE 13. RGW security against non-spoofed floods.

Figure 13 presents an evaluation of the bot-detection algorithm which aims to filter floods from non-spoofed sources. The figure shows the impact of increasing the pool of inbound public IP addresses on RGW security. The figure shows that the security offered by the bot-detection algorithm is more effective if the attack surface for the hackers is larger.

In the third set of experiments, we validate the performance of SIEM and SMM components. The experiment testbed is presented in Figure 14. We used Mininet v2.2.1 the network emulation environment and OpenvSwitch v2.3.1 for the deployment of SDN switches. Floodlight v1.1 was used as the SDN controller. Security monitoring and management elements such as SDN adapted SIEM, security sensor and security server were connected via a legacy switch. S1,S2, and S3 were virtual SDN switch which were implemented as OpenvSwitches. RO (Route Optimizer) deals with a virtualized element for routing purposes. The test network had been segmented into four LANs depending on the nature of their services. These segments have different security requirements. Tests were conducted with two users.

- DMZ LAN: It includes services exposed to Internet.
- Security LAN: It includes security services, such as the security sensor.
- Server LAN: It includes internal services.
- Client LAN: This is the end-user network.

In this testbed, a security use case had been defined as a proof of concept to show how SIEM and SMM components help detecting and isolating insecure network devices, before they can negatively affect the rest of the network. Upon discovering a potential threat by SMM, the SIEM identified the problem and automatically performs the previously considered or planned reactions to mitigate it, by interacting with the Northbound API of the SDN controller. After the threat had been resolved the SIEM software allows the affected devices to rejoin the network.

For the purpose of this use case, a VLC server streamed video in the server LAN and several VLC Clients were consuming this video from the Client LAN.

1) The "VLC Client 2" has been compromised by an external attacker.
2) The compromised host tries to extend the attack by launching a network discovery process over the internal networks, the DMZ LAN in this case.
3) The suspicious traffic of the network discovery is detected by the security sensor that is sniffing all the traffic crossing the virtual switch S1.
4) The Security sensor reports this security event to the Security Server, located in the legacy network.
5) The Security Server processes this event matching against a predefined security policy that tells him to immediately block in S1 the connections to this host.
6) Security Server process the event and it is correlated in Security Server. The matching security policy (which is built on the server side) triggers an action that injects from Security Server to the SDN Controller via the NorthBound API. Then, the SDN Controller sends a flow table update message to the S1 to drop all the traffic related to the compromised host.

In the first case, SIEM performed a cyber-attack detection by considering a unique source of information from the security sensor. The test consisted on detecting a port scan followed by sending ten echo requests (pings) that was detected by the security sensor running snort in a virtual machine deployment as described above. Different measurements have shown and compiled to an average to have a perception on the results of the different response times and latencies introduced. Figure 15 represents the delay time, which was measured in seconds, between the attacker began to carry out the attack to the time that the attack had been blocked.

Figure 16 represents the delay time between detection by the correlation engine generating the alert and the attacking device being blocked.

In the second case, SIEM considered different sources of information from the security sensor. In this case, the response times had been evaluated for a case of simulation of a botnet where a host took the control of another host in the network and proceeded to download a malware file. The compromised host and the attacker host generated a network connection and it was identified by the security server

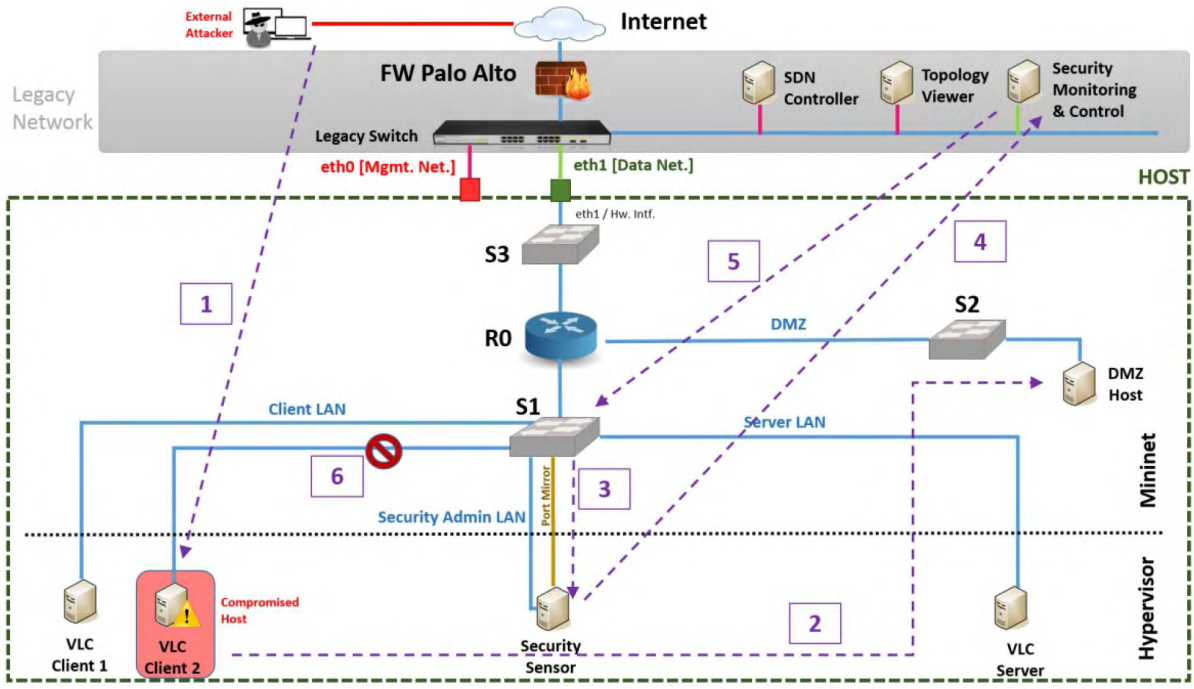

FIGURE 14. The layout of the experimental testbed for security information and event management (SIEM) and security management and monitoring (SMM) components.

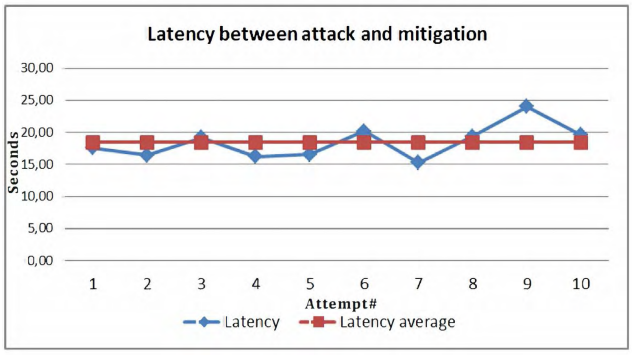

FIGURE 15. Latency between attack and mitigation. Case 1.

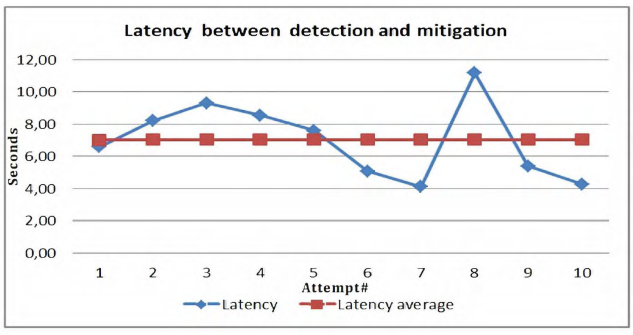

FIGURE 16. Latency between detection and mitigation. Case 1.

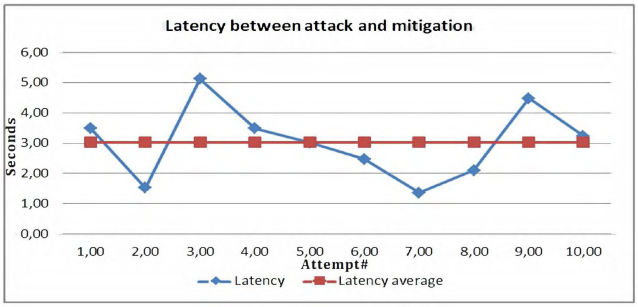

FIGURE 17. Latency between attack and mitigation. Case 2.

detecting an outgoing connection from the botnet web server. The compromised host downloaded a malware file that was also detected by the security sensor. In addition, a malware engine analyzed the downloaded file and assigned a score, in order to determine if it was considered malware.

Figure 17 represents the delay time between the time the attacker began to carry out the attack to the time that the attack had been blocked.

Figure 18 represents the delay time between detection by the correlation engine generating the alert and the attacking device being blocked.

The results of the validation show that it is possible automate mitigation and reaction actions in SDMNs by providing countermeasures and mitigation actions directly using RESTful API in a SDN controller. The result of the validation also evidences that multiples sources of information can be combined and help to provide more accurate and rapid detection on cyber attacks scenarios demonstrated. Improving

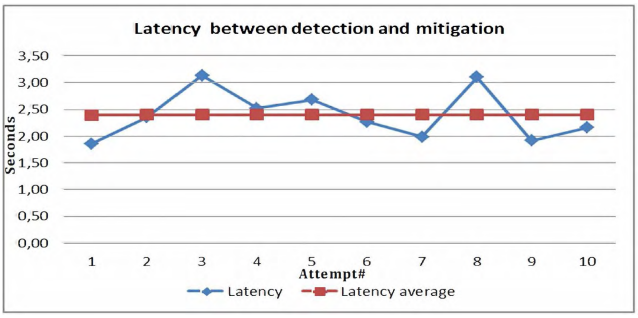

FIGURE 18. Latency between detection and mitigation. Case 2.

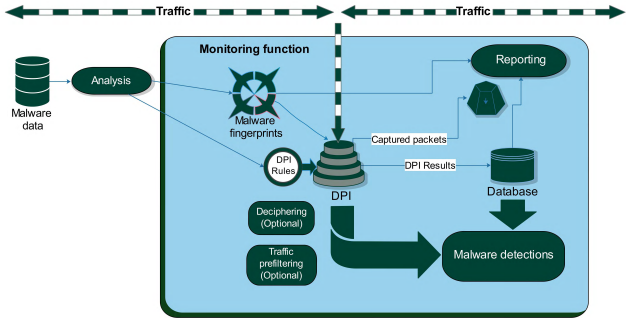

FIGURE 19. The layout of the experimental testbed for deep packet inspection (DPI) component.

performance combining multiple sources will be crucial for future and further work.

In the fourth set of experiments, we validate the performance of DPI component. Figure 19 illustrates main components of the developed monitoring prototype. The threat detection is based on malware fingerprints that are compared with monitored online traffic patterns as part of DPI analysis. Possible malware detections are then written to the local database with the other analysis data produced by the DPI component. Tests were conducted with a total of 10 simulated hosts.

In the evaluation environment about 5 percent of all data flows were interpreted as HTTP application flows by the DPI engine and therefore were compared with fingerprints (i.e., signature detection). In real time analysis we could not measure any increase of CPU usage compared to the reference DPI analysis when the same number of metadata attributes were extracted. It has been found that the actual performance penalty should be measured at high data rates when packet drops may occur due to the additional processing.

The average processing delay was defined as the measure of the delay from the time the flow starts to the time a packet is received that allows the first detection decision to be made. In the test bed environment, the average detection decision delay was 57 ms. The results of the validation show that it is possible to perform DPI in SDN scenarios by using the proposed DPI component.

V. DISCUSSION

Introducing SDN and NFV to networking will be a major game changer to the wireless networking arena. The costs, efficiency and network performance will be the main drivers of the change. There are two notable theories when it comes to network security. First is the idea of centralizing network control to minimize the fragmentation of security mechanisms. However, this inadvertently leads to higher risk of security lapses at a single point of failure, and this gives rise to the second theory which is using SDN to enhance network security by leveraging on its global network visibility feature as well as the centralized control functions. The security can be further improved by moving to a more cooperative approach within large trust alliances where trust evidence or the results of trust processing are shared over the cloud. Naturally, such technological advancements usually come with renewed threat landscape, this paper has highlighted such potential threats for SDMN, it also presented corresponding mitigation techniques together with preliminary test results.

This paper proposes the use of HIP-based IPSec tunneling architecture to secure the communication channel between separated planes. The proposed security gateways in this architecture conceal the actual controller from potential adversaries, thereby, mitigating against possible DoS and DDoS attacks. The network is accessed through policy-based communication that is enforced at the network edges using CES. The CES helps to protect the network against inherent Internet vulnerabilities such as address spoofing and DoS attacks. It is also capable of limiting the communication to only non-spoofed flows or just the authorized hosts using a tool that is capable of implementing Anything-as-a-Service delegations based on the given policy control techniques. CES links data services with the security infrastructure which has always been an essential part of mobile networks; e.g. authentication can be made a precondition of end to end data communication. CES can also enforce black lists when necessary. In addition, SIEM-based security management and real-time sensors-assisted monitoring can also be used to enforce network-wide security.

Notwithstanding the vulnerabilities that have been considered in this work, there are still some gray areas in SDMN security that require further investigations. For instance, the closely inter-linked security and scalability mechanisms in SDMN due to the centralized control plane. Hence, the need for further investigation into the control-data plane intelligence trade-off and the dependencies between security architectures and traffic forwarding mechanisms, this is required in order to minimize the delays incurred in traffic forwarding resulting in variations and multiplicity of purpose in SDMN security architectures. Essentially, security requires a feature rich data plane element at trust domain boundaries because a standard OpenFlow switch does not have all the capabilities for packet filtering or rate limiting that are needed in a proper firewall. To improve performance, many of the security

mechanisms we implemented in the proof-of-concept version of CES can be significantly improved. We are now working on making substantial improvements. Moreover, there is also a need to investigate other identity-location separation architectures beside HIP, this will pave way for higher mobility with security in future wireless networks.

VI. CONCLUSION AND FUTURE WORKS

This paper investigated the security vulnerabilities in SDMN (Software Defined Mobile Networks) and proposed novel security architectures to mitigate them. On the up side, SDMN concepts will improve network security leveraging on it global visibility of the network state in addition to its centralized control and network function softwarization. On the down side,these same attributes also introduce new vulnerabilities that are inherent to software applications, Internet-based systems, and new technologies. This paper presented a comprehensive collection of the pros and cons related to SDMN as well as the state of the art for implementing security architectures in SDMN. Based on the outcome of the experiments in this work, we maintain that security considerations are paramount when relying on SDN and NFV.

Various security methods have been implemented on the SDMN platform. In this work, we presented a multi-tier security architecture based on five key components: (1) secure communication channels leveraging on HIP. This is used to secure both control and data channels; (2) policy based communications. This will serve to mitigate DoS attacks as well as source address spoofing, it will also allow network communications between end hosts only after a successful negotiation of policy between edge nodes. This will effectively tackle the problem of unwanted traffic across the network and managing all flow admissions by policy; (3) security management and monitoring where the security mechanisms implemented are monitored on one hand while detected security threats are isolated using DPI and traffic monitoring techniques on the other hand; (4) Security Defined Monitoring (SDM) to orchestrate the monitoring activities related to security and finally 5) Deep Packet Inspection (DPI) component for improved security threat detection.

In this work, we analyzed the feasibility of implementing these components in a real-world using testbeds. The outcome of these experiments showed that the proposed security architecture can be implemented in real-world and would be able to prevent IP based attacks on SDMNs. The results of the validation also show that it is possible to automate mitigation and reaction actions in SDMNs by providing countermeasures and mitigation actions directly using RESTful API in an SDN controller. The result of the validation shows that multiple sources of information can be combined to provide more accurate and rapid detection of cyber attack.

Notwithstanding, certain elements of these system still needs to be examined in greater detail before integrating these new systems with the existing production environments. We will extend this research to further analyze these requirements and define specific guidelines for the integration of the proposed security components into the SDMN architecture.

ACKNOWLEDGMENT

This work has been performed in the framework of the CELTIC project CP2012 SIGMONA and SECUREConnect (Secure Connectivity of Future Cyber-Physical Systems) Projects. The authors would like to acknowledge the contributions of their colleagues.

REFERENCES

[1] S. Shin, P. A. Porras, V. Yegneswaran, M. W. Fong, G. Gu, and M. Tyson, "FRESCO: Modular composable security services for software-defined networks," in *Proc. NDSS*, Apr. 2013, pp. 1–16.

[2] J. Kempf, E. Bellagamba, A. Kern, D. Jocha, and A. Takacs, and P. Sköldstrom, "Scalable fault management for OpenFlow," in *Proc. IEEE Int. Conf. Commun. (ICC)*, Jun. 2012, pp. 6606–6610.

[3] K. Phemius, M. Bouet, and J. Leguay, "DISCO: Distributed multi-domain SDN controllers," in *Proc. Netw. Oper. Manage. Symp. (NOMS)*, May 2014, pp. 1–4.

[4] Y. Zhang, N. Beheshti, and M. Tatipamula, "On resilience of split-architecture networks," in *Proc. Global Telecommun. Conf. (GLOBECOM)*, Dec. 2011, pp. 1–6.

[5] A. K. Nayak, A. Reimers, N. Feamster, and R. Clark, "Resonance: Dynamic access control for enterprise networks," in *Proc. 1st ACM Workshop Res. Enterprise Netw.*, 2009, pp. 11–18.

[6] H. Hu, W. Han, G.-J. Ahn, and Z. Zhao, "FLOWGUARD: Building robust firewalls for software-defined networks," in *Proc. 3rd Workshop Hot Topics Softw. Defined Netw.*, 2014, pp. 97–102.

[7] S. Luo, J. Wu, J. Li, L. Guo, and B. Pei, "Toward vulnerability assessment for 5G mobile communication networks," in *Proc. IEEE Int. Conf. Smart City/SocialCom/SustainCom (SmartCity)*, Dec. 2015, pp. 72–76.

[8] X. Duan and X. Wang, "Authentication handover and privacy protection in 5G HetNets using software-defined networking," *IEEE Commun. Mag.*, vol. 53, no. 4, pp. 28–35, Apr. 2015.

[9] K. Pentikousis, Y. Wang, and W. Hu, "Mobileflow: Toward software-defined mobile networks," *IEEE Commun. Mag.*, vol. 51, no. 7, pp. 44–53, Jul. 2013.

[10] H. Hawilo, A. Shami, M. Mirahmadi, and R. Asal. (2014). "NFV: State of the art, challenges and implementation in next generation mobile networks (vEPC)." [Online]. Available: https://arxiv.org/abs/1409.4149

[11] M. Liyanage, M. Ylianttila, and A. Gurtov, *Software Defined Mobile Networks (SDMN): Beyond LTE Network Architecture*. Hoboken, NJ, USA: Wiley, 2015.

[12] M. Liyanage, A. Abro, M. Ylianttila, and A. Gurtov, "Opportunities and challenges of software-defined mobile networks in network security," *IEEE Security Privacy*, vol. 14, no. 4, pp. 34–44, Aug. 2016.

[13] ITU-T. (2016). *World Telecommunication/ICT Facts and Figures*. [Online]. Available: http://www.itu.int/en/ITU-D/Statistics/Pages/stat/default.aspx

[14] I. Ahmad, S. Namal, M. Ylianttila, and A. Gurtov, "Security in software defined networks: A survey," *IEEE Commun. Surveys Tuts.*, vol. 17, no. 4, pp. 2317–2346, 4th Quart., 2015.

[15] A. Y. Ding, J. Crowcroft, S. Tarkoma, and H. Flinck, "Software defined networking for security enhancement in wireless mobile networks," *Comput. Netw.*, vol. 66, pp. 94–101, Jun. 2014.

[16] T. Hinrichs, N. Gude, M. Casado, J. Mitchell, and S. Shenker, "Expressing and enforcing flow-based network security policies," Univ. Chicago, Chicago, IL, USA, Tech. Rep., 2008, vol. 9.

[17] S. Gutz, A. Story, C. Schlesinger, and N. Foster, "Splendid isolation: A slice abstraction for software-defined networks," in *Proc. 1st Workshop Hot Topics Softw. Defined Netw.*, 2012, pp. 79–84.

[18] R. Beckett, X. K. Zou, S. Zhang, S. Malik, J. Rexford, and D. Walker, "An assertion language for debugging SDN applications," in *Proc. 3rd Workshop Hot Topics Softw. Defined Netw., Ser. HotSDN*, vol. 14. 2014, pp. 91–96.

[19] A. Khurshid, W. Zhou, M. Caesar, and P. B. Godfrey, "Veriflow: Verifying network-wide invariants in real time," *ACM SIGCOMM Comput. Commun. Rev.*, vol. 42, no. 4, pp. 467–472, 2012.
[20] E. Al-Shaer and S. Al-Haj, "FlowChecker: Configuration Analysis and Verification of Federated OpenFlow Infrastructures," in *Proc. 3rd ACM Workshop Assurable Usable Secur. Configuration*, 2010, pp. 37–44.
[21] S. Son, S. Shin, V. Yegneswaran, P. Porras, and G. Gu, "Model checking invariant security properties in OpenFlow," in *Proc. IEEE Int. Conf. Commun. (ICC)*, Jun. 2013, pp. 1974–1979.
[22] M. Canini, D. Kostic, J. Rexford, and D. Venzano, "Automating the testing of OpenFlow applications," in *Proc. 1st Int. Workshop Rigorous Protocol Eng. (WRiPE)*, Jun. 2011, pp. 1974–1979.
[23] (Oct. 2013). *Security-Enhanced Floodlight*. [Online]. Available: http://www.sdncentral.com/education/toward-secure-sdn-controllayer/
[24] R. Braga, E. Mota, and A. Passito, "Lightweight DDoS flooding attack detection using NOX/OpenFlow," in *Proc. IEEE 35th Conf. Local Comput. Netw. (LCN)*, Oct. 2010, pp. 408–415.
[25] P. Fonseca, R. Bennesby, E. Mota, and A. Passito, "A replication component for resilient OpenFlow-based networking," in *Proc. Netw. Oper. Manage. Symp. (NOMS)*, 2012, pp. 933–939.
[26] P. Porras, S. Shin, V. Yegneswaran, M. Fong, M. Tyson, and G. Gu, "A security enforcement kernel for OpenFlow networks," in *Proc. 1st Workshop Hot Topics Softw. Defined Netw.*, 2012, pp. 121–126.
[27] *SDN Concept in Generalized Mobile Network Architectures (SIGMONA) Project*, accessed on May 1, 2017. [Online]. Available: http://www.sigmona.org/
[28] J. Costa-Requena *et al.*, "SDN and NFV integration in generalized mobile network architecture," in *Proc. Eur. Conf. Netw. Commun. (EUCNC)*, Jul. 2015, pp. 1–5.
[29] Ericsson. *5G Systems*, accessed on May 15, 2017. [Online]. Available: https://www.ericsson.com/assets/local/publications/white-papers/wp-5g-systems.pdf
[30] M. McBride, M. Cohn, S. Deshpande, M. Kaushik, M. Mathews, and S. Nathan, "SDN security considerations in the data center," *Open Netw. Found. Solution*, Oct. 2013, pp. 1–12.
[31] C. Meyer and J. Schwenk, "Lessons learned from previous SSL/TLS attacks- a brief chronology of attacks and weaknesses," *IACR Cryptol. ePrint Arch.*, 2013, p. 49.
[32] D. Kreutz, F. Ramos, and P. Verissimo, "Towards Secure and Dependable Software-Defined Networks," in *Proc. 2nd ACM SIGCOMM Workshop Hot Topics Softw. Defined Netw.*, 2013, pp. 55–60.
[33] M. Liyanage, A. Gurtov, and M. Ylianttila, "IP-based virtual private network implementations in future cellular networks," *Handbook Res. Prog. Trends Wireless Commun. Netw.*, 2014, p. 44.
[34] R. Axelrod, "The evolution of cooperation: Revised edition," *Basic books*, vol. 185, p. 186, Mar. 2006.
[35] W. Eddy, "TCP SYN flooding attacks and common mitigations," in *Proc. RFC*, 2007, 4987,.
[36] *Introduction to Cisco IOS NetFlow-A Technical Overview*, accessed on May 15, 2017. [Online]. Available: http://www.cisco.com/c/en/us/products/collateral/ios-nx-os-software/ios-netflow/prod_white_paper0900aecd80406232.html
[37] G. Folino and P. Sabatino, "Ensemble based collaborative and distributed intrusion detection systems: A survey," *J. Netw. Comput. Appl.*, vol. 66, pp. 1–16, May 2016.
[38] A. Lara, A. Kolasani, and B. Ramamurthy, "Network innovation using OpenFlow: A survey," *IEEE Commun. Surveys Tuts.*, vol. 16, no. 1, pp. 493–512, First Quarter 2014.
[39] N. McKeown *et al.*, "OpenFlow: Enabling innovation in campus networks," *ACM SIGCOMM Comput. Commun. Rev.*, vol. 38, no. 2, pp. 69–74, Apr. 2008.
[40] *About POX*, accessed on May 1, 2017. [Online]. Available: http://www.noxrepo.org/pox/about-pox/
[41] *Open vSwitch: An Open Virtual Switch*, accessed on May 1, 2017. [Online]. Available: http://openvswitch.org/
[42] *The OpenHIP Project*, accessed on May 1, 2017. [Online]. Available: http://www.openhip.org/
[43] *Iperf*, accessed on May 1, 2017. [Online]. Available: http://iperf.sourceforge.net/
[44] "Carrier cloud telecoms-exploring the challenges of deploying virtualisation and SDN in telecom networks," Intel Cooperation, Santa Clara, CA, USA, Tech. Rep., 2013.

MADHUSANKA LIYANAGE (S'07–M'16) received the B.Sc. degree (Hons.) in electronics and telecommunication engineering from the University of Moratuwa, Moratuwa, Sri Lanka, in 2009, the M.Eng. degree from the Asian Institute of Technology, Bangkok, Thailand, in 2011, the M.Sc. degree from the University of Nice Sophia Antipolis, Nice, France, in 2011, and the Ph.D. degree in communication engineering from the University of Oulu, Oulu, Finland, in 2016. From 2011 to 2012, he was a Research Scientist with the I3S Laboratory and Inria, Shopia Antipolis, France. He is currently a Post-Doctoral Researcher and a Project Manager with the Center for Wireless Communications, University of Oulu. He has been a Visiting Research Fellow with the Data61, CSIRO, Sydney Australia, the Infolabs21, Lancaster University, U.K., and Computer Science and Engineering, The University of New South Wales since 2016. He is a Member of the ICT. He has co-authored over 30 publications including one edited book with Wiley. He is also a Management Committee Member of the EU COST Action IC1301, IC1303, CA15107, and CA15127 projects. His research interests are SDN, IoT, block chain, mobile, and virtual network security.

IJAZ AHMED (S'13) received the B.Sc. degree in computer systems engineering from the University of Engineering and Technology, Peshawar, Pakistan, the M.Sc. (Technology) degree in wireless communications engineering with major in telecommunications engineering from the University of Oulu, Finland, in 2012, where he is currently pursuing the Ph.D. degree. He was a Research Assistant with the Center for Wireless Communications. He has received several awards including the 2013 Achievement Award as Inventor from the University of Oulu, and the Nokia Foundation and the Tuano Tönning Foundation Research Grants for his excellent research in University of Oulu. During his Ph.D., he contributed over 15 publications including high impact factor journal articles, conference papers, and book chapters. His research interest includes SDN, SDN-based mobile networks, wireless networks, network security, and network load balancing.

JUDE OKWUIBE (S'16) received the B.Sc. degree in telecommunications and wireless technologies from the American University of Nigeria, Yola, in 2011, the master's degree in wireless communications engineering from the University of Oulu, Finland, in 2015, where he is currently pursuing the Ph.D. degree in communications engineering with the Graduate School. He was a recruitment specialist with the American University of Nigeria for about a year before going for a one year National Service, where he served as an Assistant Instructor teaching computer science. His research interests are 5G and future networks, IoT, SDN, network security, and biometric verifications.

MIKA YLIANTTILA (M'99–SM'08) received the Ph.D. degree on communications engineering from the University of Oulu in 2005. He was an Associate Director of the MediaTeam Research Group from 2009 to 2011, a Professor (pro tem) in information networks from 2005 to 2010, and the Director of the Center for Internet Excellence from 2012 to 2015. He is currently a Full-Time Professor with the Center for Wireless Communications, Faculty of Information Technology and Electrical Engineering, University of Oulu, Finland. He has been an Adjunct Professor in computer science and engineering since 2007. He has co-authored over 100 international peer-reviewed articles on broadband communications networks and systems, including aspects on network security, mobility management, distributed systems, and novel applications. His research interests include 5G applications and services, software-defined networking and edge computing. He is currently an Editor of *Wireless Networks* journal.

HAMMAD KABIR (S'14) is currently pursuing the Ph.D. degree with the Department of Communication and Networking, Aalto University, Finland. His research focuses on intrusion detection, network security, mobile network, SDN, and policy management.

JESUS LLORENTE SANTOS (S'13) is currently pursuing the Ph.D. degree with the Department of Communication and Networking, Aalto University, Finland. His research focuses on mobile networks, software defined networking, and future internet architectures.

RAIMO KANTOLA (M'97) is a Doctor of Science in Technology. He was with Nokia Networks holding positions in research and development and marketing for 15 years. He joined the Helsinki University of Technology as a Professor in 1996 and was tenured in 2006. He is currently a Full, tenured Professor of Networking Technology with the Department of Communications and Networking, Aalto University. His recent research is in trust in networks and customer edge switching. He has held many positions of trust with the Helsinki University of Technology and Aalto University and acted/acts as the General Chair of several conferences including the TrustCom 2015 and the NSS 2017.

OSCAR LÓPEZ PEREZ received the B.Sc. in telecommunication engineering from the the Polytechnic University of Catalonia in 1998. He was with a technical school, where he was involved in teaching different IT subjects in an Associate degree. In 2000, he joined Nextel S.A., covering different stages as technical, auditor, and later providing consultancy services in ICT and cyber security. Since 2008, he has been a Research and Development Researcher, participating in national and European research projects. His research work has been related with the evaluation of the operational security assurance, and in other initiatives such as enforcing security policies and in the result of an adequate security monitoring in different application environments.

MIKEL URIARTE ITZAZELAIA received the B.Sc. and M.Sc. degrees in telecommunication engineering from the University of the Basque Country in 1998. He spent one year in public Research and Development with Tecnalia. From 2001 to 2006, he was the ICT Director and an Information Security Lead Auditor, subsequently becoming the Head of Research and Development Unit. Since 1998, he has been with Nextel S.A., a Telecommunications enterprise providing ICT engineering and consulting services. His research interests include ICT interoperability, resilience, performance, and security in several areas such as identity and access control, networking, wireless sensing, and cloud computing.

EDGARDO MONTES DE OCA (M'16) received the degree in engineering from Paris XI University, Orsay, in 1985. He was Research Engineer with the Alcatel Corporate Research Center, Marcoussis, France, and the Ericsson's Research Center, Massy, France. In 2004, he founded Montimage, where he is currently the CEO. He is the Originator and main Architect of Montimage Monitoring Tool. He has authored many papers and book chapters on SDN/SVN, testing, network monitoring, network security, and performance. His main interests are future networks SDN/NFV, network and application monitoring and security, detection and mitigation of cyber-attacks, and building critical systems that require the use of the state-of-the-art fault-tolerance, testing, and security techniques. He has participated in several EU and French national research projects such as the CelticPlus-MEVICO, SIGMONA, and SENDATE, the H2020-SISSDEN, and the ANR-DOCTOR. He is a member of the NetWorld2020.