Towards Security-as-a-Service in Multi-Access Edge

Reza Tourani
Saint Louis University
reza.tourani@slu.edu

Austin Bos
New Mexico State University
abos@nmsu.edu

Satyajayant Misra
New Mexico State University
misra@cs.nmsu.edu

Flavio Esposito
Saint Louis University
flavio.esposito@slu.edu

ABSTRACT

The prevailing network security measures are often implemented on proprietary appliances that are deployed at fixed network locations with constant capacity. Such a rigid deployment is sometimes necessary, but undermines the flexibility of security services in meeting the demands of emerging applications, such as augmented/virtual reality, autonomous driving, and 5G for industry 4.0, which are provoked by the evolution of connected and smart devices, their heterogeneity, and integration with cloud and edge computing infrastructures.

To loosen these rigid security deployments, in this paper, we propose a data-centric SECurity-as-a-Service (SECaaS) framework for elastic deployment and provisioning of security services at the Multi-Access Edge Computing (MEC) infrastructure. In particular, we discuss three security services that are suitable for edge deployment: *(i)* an intrusion detection and prevention system (IDPS), *(ii)* an access control enforcement system (ACE), and *(iii)* a communication anonymization service (CA). We benchmark the common security microservices along with the design and implementation of a proof of concept communication anonymization application.

CCS CONCEPTS

• **Security and privacy** → **Security services**; **Virtualization and security**; **Access control**; **Privacy-preserving protocols**; **Intrusion/anomaly detection and malware mitigation**; **Trusted computing**; **Denial-of-service attacks**; **Firewalls**; **Security protocols**; • **Networks** → **Network security**; **In-network processing**; *Network protocol design*; *Network simulations.*

KEYWORDS

Security, virtualization, microservice, NDN, edge computing

ACM Reference Format:
Reza Tourani, Austin Bos, Satyajayant Misra, and Flavio Esposito. 2019. Towards Security-as-a-Service in Multi-Access Edge. In *The Fourth ACM/IEEE Symposium on Edge Computing (SEC 2019), November 7–9, 2019, Arlington, VA, USA.* ACM, New York, NY, USA, 6 pages. https://doi.org/10.1145/3318216.3363335

SEC 2019, November 7–9, 2019, Arlington, VA, USA

ACM ISBN 978-1-4503-6733-2/19/11…$15.00
https://doi.org/10.1145/3318216.3363335

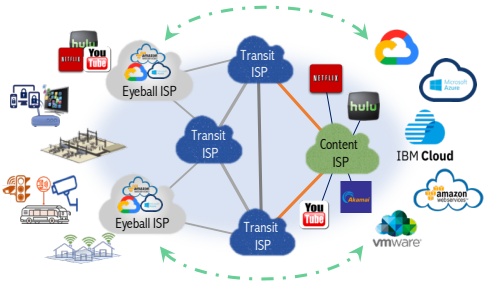


Figure 1: MEC's multi-stakeholder and -tenant ecosystem.

1 INTRODUCTION

The advent of Industrial Internet of Things (IoT), autonomous driving, and smart home applications are the major contributors to the fast growth of intelligent and connected devices, which is expected to outnumber the global population[1]. These devices enter the market with hardware and software vulnerabilities and onboard into networks without proper configuration and commissioning, extending the attack surface. We are witnessing the common trend of compromised IoT devices, forming botnets for massive Distributed Denial of Service (DDoS) attacks orchestration. For instance, Mirai Botnet attack on Dyn DNS used 600, 000 compromised CCTV cameras and routers to take down Netflix and Twitter services [1].

The existing countermeasures against these vulnerabilities and the emerging Cyber threats are designed for vendor-specific proprietary middleboxes with fixed processing capacity–undermining their elasticity to meet the peak traffic demand. Often, these measures are deployed at the service providers or the distant cloud (*e.g.,* Netflix's cloud-based user authentication and authorization and cloud-based DDoS mitigation solutions). Placing these services at fixed network locations inevitably introduces long delays due to routing detour, path stretch, communication overhead, and requires absolute trust in the centralized services [2]. Such a rigid design makes these security services incompetent in providing network safety and users' privacy against the evolving attack vector, which is growing with the infusion of devices at the edge and the increasing hardware and software vulnerabilities. It has been argued that these security services cannot meet the expectations of the new breed of applications, which are dynamically running on virtualized environments across multiple servers and data centers [3, 4].

Motivated by these observations, in this paper, we sketch the design of a trustworthy edge-centric security service implementation and provisioning in a multi-tenant environment–SECurity-as-a-Service (SECaaS) in Multi-Access Edge Computing (MEC). Figure 1 illustrates a multi-stakeholder and multi-tenant architecture, composed of competing stakeholders (*e.g.,* Google and Amazon), placed at the edge of Eyeball ISPs (the edge ISPs provide users'

[1]Cisco Visual Networking Index: Forecast and Trends, 2017–2022 White Paper.

connectivity), serving a diverse set of tenants (*e.g.,* smart houses, Industrial IoT, and data publishers). Considering the rigidity and inefficiency of the existing monolithic security measures for edge deployment [5], we envision a microservice architecture for our proposed SECaaS framework that can be devised by chaining primitive cryptographic microservices (*e.g.,* hashing and symmetric ciphers) to promote agile and flexible service orchestration.

In designing SECaaS, we employ a data-centric substrate, which inherently supports the distributed nature of edge computing by decoupling data from locations [6]. Such a data-centric paradigm also fosters in-network processing by providing applications' semantics to the network layer (enabling informed traffic processing by the network entities) and satisfies the data-centric security demand of our SECaaS framework, in which data is expected to be secured rather than the communication channels. In particular, we use the Named-Data Networking (NDN) [7] architecture, an emerging Information-Centric Networking (ICN) realization, to support the edge computing constraints. NDN uses unique content naming, name-based routing, and built-in security solutions, such as data integrity and provenance as well as producer's trust assessment via digital signatures. We argue that adopting NDN's principles in our edge-based SECaaS framework leads to economic advantages for the stakeholders, tenants, and service providers by reducing the downlink bandwidth via pervasive caching.

Migrating the security measures to the edge, closer to attacks' sources, is a compelling solution in dealing with the evolving attack surface. Such an edge-based security model requires an innovative service design to fully embrace the distributed nature of edge computing and a data-centric networking model to promote in-network processing. We believe the fusion between two emerging technologies, such as microservice architecture and NDN can facilitate the development, deployment, and orchestration of security services at the multi-access edge infrastructure.

In the rest of the paper, we review preliminary definitions in Section 2 and discuss the TCP/IP limitations and NDN's benefits for our SECaaS framework in Section 3. Section 4 includes our envisioned principles for the deployment of security services at the edge. We then detail our edge-centric SECaaS design in Section 5 and perform a proof of concept evaluation in Section 6. In Section 7, we draw our conclusion along with the scope of the future work.

2 BACKGROUND AND RELATED WORK

Named-data Networking Overview: Different from the contemporary networks, which use IP addresses to identify the servers that host the data, the novel Named-Data Networking (NDN) architecture [7] allows data retrieval through unique data naming, data caching, and name-based routing. Each NDN router is equipped with a Content Store (CS), a Pending Interest Table (PIT), and a Forwarding Information Base (FIB). The CS, analogous to the buffer memory in IP routers, is the temporary data cache. The FIB plays a similar role to the IP routers' forwarding tables and similarly will be populated via a routing algorithm. PIT is unique to NDN, which keeps track of on-flight requests and allows request aggregation via stateful forwarding plane.

For retrieving a data chunk, the requester sends a request (*Interest*) to the network by including the data name. Requesting data by name allows the intermediate routers (middleboxes) to perform a CS lookup on the requested data name. On a failed CS lookup, the router performs a PIT lookup to check whether there is an existing entry for the requested data. A successful PIT lookup causes the router to drop the request and add the incoming interface to the existing PIT entry–request aggregation. On a failed PIT lookup, the router creates a new PIT entry and forwards the request, by consulting with the FIB, towards the data source(s). The data takes the request's reverse path to the requester–stateful forwarding–whether it is satisfied by an intermediate router or the source. Other distinctive NDN features are the strategy layer, which allows multiple simultaneous packet forwarding, and its built-in security, including data integrity, provenance, and producer's trust assessment.

Related Work: The state-of-the-art in NDN edge computing focuses on the networking aspects, including task offloading [8, 9], resource discovery [6], and dynamic code execution using lightweight VMs [10]. Recent initiatives used virtualization techniques to implement NDN's forwarding logic as a collection of virtualized functions or microservices [11, 12] for the progressive deployment of NDN islands [13], and a virtualized ICN-based wireless network to tackle resource allocation and caching problems [14]. However, these approaches have neglected to fully utilize NDN's potential in building a secure and resilience MEC.

In the IP domain, the need for modeling, evaluation, and analyses of the security services resulted in conceptual models for support and integration of security service in the cloud [15]. These efforts led to the exploration of a cooperative security service chaining model in multi-domain environments, in which administrative domains negotiate the service duration and resource dedication for a "best-effort" cooperation [16]. In the industry domain, Netflix is one of the pioneers of transforming its monolithic application to a microservice architecture hosted on Amazon Cloud [17]. However, the limitations of the distant cloud have motivated the edge-based deployment of virtualized security services [18]. Given the nascency of edge-centric SECaaS, deeper exploration is needed to shed light on different concerns including interoperability, scalability, and economic aspects of this design.

3 HOST-CENTRIC VERSUS DATA-CENTRIC SUBSTRATE

In this section, we compare the host-centric and data-centric networking paradigms as the SECaaS communication substrate.

3.1 TCP/IP Network for Edge-centric SECaaS

In today's host-centric TCP/IP network, service providers deploy security services in-house or at distant clouds (e.g., provider-based user authentication and authorization and cloud-based DDoS mitigation solutions). These approaches cannot compete with the evolving attack vector, which is growing with the infusion of devices at the edge (e.g., Internet of Things, edge computing, and autonomous driving) and the increasing hardware and software vulnerabilities. Deployment of security services at the network edge can be a more effective countermeasure [2, 18], in which traffic monitoring and filtering happen before the traffic enters the core network.

Providing security services at the edge of TCP/IP networks requires interactions between the edge servers and content providers

for a comprehensive end-to-end service–authentication at the edge and data access at the provider. Such interactions introduce communication overhead and potential path stretch, especially when the edge servers are not on the path between the users to providers. Furthermore, mobile users moving to other domains need to discover available edge servers, establish new sessions, and resume their services, which degrade users' Quality-of-Experience (QoE) due to higher power consumption and latency.

3.2 NDN for Edge-centric SECaaS

NDN's content naming, pervasive caching, and built-in security eliminate the end-to-end connectivity requirement of TCP/IP networks, promoting trustworthy content retrieval from the network's entities (the provider signs all content). With the network caching the popular content and the edge infrastructure providing security services, users experience low latency communication with high service/data availability (higher users' QoE). Content caching also reduces the uplink and downlink traffic to the core network, contributing to lower bandwidth utilization and ISP's transit costs.

In contrast to TCP/IP networks, NDN's native mobility support simplifies the discovery of new edge servers and the service re-establishment of mobile users moving to new edge domains (each edge domain is an instance of a stakeholder's infrastructure in an ISP). Such benefits are even more significant in highly mobile scenarios like vehicular networks, in which users continuously move across edge domains. NDN, by design, augments the network layer with the applications' semantics through content naming. It helps the network to forward the users' requests to the corresponding MEC server, for service execution, based on their names. Moreover, the network entities can make dynamic forwarding decisions with high granularity, which promotes preferential treatment of traffic flows. For instance, routers can decide whether to unicast or multicast the request to the edge server or the provider.

4 SECURITY SERVICES

In general, the majority of the security mechanisms, from the cryptographic primitives such as symmetric cryptosystem to more complex systems like intrusion detection and prevention system are viable to be deployed at MEC. However, we consider those services, which: *(i)* their edge deployment prevents malicious traffic from entering the core network, *(ii)* serve a wide range of tenants, and *(iii)* do not require end-to-end encryption or violate tenants' privacy. Thus, we select three security services, namely *Intrusion Detection and Prevention System (IDPS)*, *Access Control enforcement (ACE)*, and *Communication Anonymization (CA)* due to their critical role in preventing and mitigating recent DDoS attacks. In what follows, we review these security services in the context of the NDN edge.

Intrusion Detection & Prevention System (IDPS). The evolving cyber attacks motivated the development of the Next-Generation Firewalls (NGFWs) and Intrusion Detection and Prevention Systems (IDPS), which are designed to persistently monitor the network and bypassing traffic for detection and prevention of anomalies and well-known attacks [19]. In an NDN network, an intrusion detection and prevention system can be used to detect and prevent DDoS and content poisoning attacks as well as malicious and abnormal traffic identification [20].

The existing cloud-based IDPS solutions suffer from privacy violation, high communication overhead caused by transiting the suspicious traffic to the distant cloud, and their off-premise placement [2, 21]. To address these limitations, an IDPS service should be deployed at the network's edge, closer to the attacks' sources, to process the low-volume local traffic compared to the high-volume traffic that will be aggregated in the core. Such a deployment also protects the network's core from malicious traffic and reduces the uplink bandwidth of the Eyeball ISPs, which results in lower transit cost. Potential tenants of an IDPS service are Internet Service Providers (ISPs), enterprise networks, and content providers, which are in need of protecting their networks.

Access Control Enforcement (ACE). In today's Internet, major content providers such as Netflix employ cloud-based access control, where users authenticate themselves to the authentication server deployed at the distant cloud to get the service from the provider. However, such an access control delegation introduces path stretch and additional latency. While this is a bearable degradation in users' QoE for the traditional applications, the new breed of applications, such as augmented reality/virtual reality, autonomous driving, and live video analytics have more stringent latency requirements. Such expectations call for the deployment of the access control enforcement in the vicinity of the data, computation, and user–the network edge. Delegating access control enforcement to the MEC improves data and service availability as well as reducing the communication latency, which subsequently improves the expected users' QoE. Moreover, an edge-based ACE service can help reduce the impacts of DDoS attacks at the edge by preventing unauthorized traffic from entering the network's core.

An NDN-based ACE mechanism, to be viable for edge deployment, should effectively prevent unauthorized access to the data, avoid per-request interaction with the providers, incur minimal computation on the MEC servers, and allow user mobility. Among the existing data-centric ACE frameworks, the closest to our vision are the proposed approaches in [22, 23], in which the authentication and authorization tasks have been delegation to trusted edge routers. However, further exploration is needed to design a full-fledged ACE framework to consider authentication delegation to untrusted MEC infrastructure with secure accountability. Deployment of such an ACE service at MEC can serve a wide range of tenants from multimedia providers, such as Netflix and YouTube, to enterprise and IoT networks.

Communication Anonymization (CA). In today's TCP/IP networks, to provide communication anonymity and bypassing censorship, users securely tunnel their traffic to the trusted proxies or anonymization networks (*e.g.,* Tor), which are placed outside the censoring domain. However, the use of IP addresses for routing and forwarding allows the powerful filtering authorities to backtrace traffic to their sources–user linkability–even when leveraging robust anonymization tools [24]. The NDN's stateful forwarding plane, however, solves the TCP/IP's linkability problem by eliminating the need for IP addresses in the communication. Thus, the majority of data-centric anonymization frameworks focused on evading name-based traffic filtering by adopting proxy-based secure tunneling [25, 26].

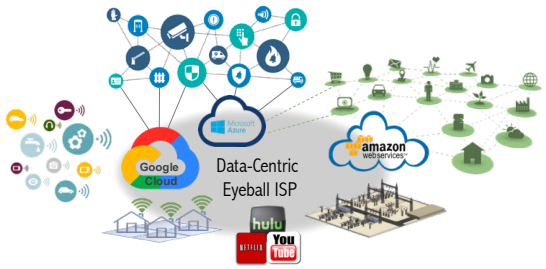


Figure 2: An edge-centric multi-tenant and stakeholder SECaaS architecture.

For a truly unlinkable and secure edge-based CA service, the anonymizing proxies should be deployed across multiple MEC servers (preferably belonging to different stakeholders). Optimizing the proxy placement helps to minimize communication overhead and maximizing users' privacy. Anonymization service deployment at the edge can serve tenants, such as users in need of private communication, critical cyber-physical infrastructure, and businesses protecting their intellectual properties.

5 DATA-CENTRIC SECaaS ARCHITECTURE: DESIGN, DEPLOYMENT, AND CHALLENGES

5.1 SECaaS Architecture Overview

In our envisioned data-centric SECaaS design (illustrated in Figure 2), multiple stakeholders, such as Google, Amazon, and IBM place their MEC infrastructure at the edge of Eyeball ISPs to deliver security services to a diverse range of tenants like content providers, IoT networks, and smart homes. Within an Eyeball ISP, each stakeholder is connected to *(i)* its cloud provider via high bandwidth links, and *(ii)* other stakeholders. We assume that security services are consistent across all stakeholders to promote intra-stakeholder service handover for the mobile tenant moving across multiple domains and ISPs as well as inter-stakeholder collaborations. The Eyeball and content ISPs (those ISPs that serve content providers, such as Akamai and Youtube) are connected via transit ISPs (*e.g.,* Vodafone and Easynet).

For service registration, a tenant obtains the service catalog including service description, service level agreement (SLA), and QoE guarantees from all the available stakeholders. Despite service consistency, factors such as stakeholders' SLAs, QoE guarantees, and service fees may affect the tenants' decisions. Upon stakeholder selection, the tenant's service information (*e.g.,* capacity, SLA, and service fee) will be stored at the stakeholder's edge and cloud infrastructure for seamless service connectivity across multiple domains. In the case of inter-stakeholder collaborations, such information will be shared with the peering stakeholders.

Once the tenant's service registration is confirmed, the tenant's Eyeball ISP should route the tenants' traffic to the corresponding MEC infrastructure. In our design, this is achieved by utilizing the content naming and name-based routing of the ISPs' data-centric communication model [7]–one of the major advantages of the NDN architecture compared to the existing TCP/IP model. Each service includes a lightweight counterpart application that runs on the tenant side. Such a counterpart application–analogous to the existing client-side media players–facilitates service utilization by integrating service information (*e.g.,* stakeholder and service IDs) into tenants' traffic. For *Direct* services (those that their tenants are the actual service users, such as CA service), the tenant's application explicitly modify the tenants' requests. For instance, the counterpart application for Amazon's CA service running on the tenant' host will transform the request name of a data chunk from "`/YouTube /music/jazz/ch_1`" to "`/AWS/CA/fLwlhmlXYU/.`" The transformed name indicates the stakeholder, requested service, and the encrypted content name, respectively, to augment tenant's anonymity and packet forwarding to the CA service deployed at the AWS MEC. As for *Indirect* services (those that their tenants register the services for their subscribers, such as Netflix registering an ACE service for its subscribers' authentication) the tenants either use the counterpart applications to onboard their subscribers or delegate this task to MEC.

5.2 Security Service Development and Deployment

We use microservice architecture in developing the SECaaS services, which is inline with the premise of Network Function Virtualization (NFV)–decoupling network services from the proprietary hardware appliances to facilitate service deployment, provisioning, placement, and utilization [27] while significantly reducing operating expenses (OPEX) and capital expenses (CAPEX). It has been shown that deployment of virtualized network functions (VNFs) at the edge improves scalability via edge analytic, reduces the response time, and augments user and data privacy [28, 29].

We envision the advantages of implementing security VNFs as collections of microservices to be twofold. First, the majority of security VNFs share similar building blocks (microservices), such as signature verification, (a)symmetric cipher, and cryptographic hashing, which can be used to promote agile service development via microservice chaining. Second, microservicing these building blocks promotes assets re-usability in developing various services, which subsequently reduces the service provisioning complexity.

5.3 Secure Accountability of the MEC

Despite MEC benefits, such a distributed ecosystem includes trusted and partially trusted entities performing computation tasks for the deployed applications, implying the coexistence of malicious, compromised, and trusted entities [2]. Thus, giving rise to a major concern–how to trust the results of the offloaded task (service) to the third-party or the integrity of data stored, in the presence of compromised infrastructure. This calls for an accountable and trustworthy computation offloading framework to keep the MEC infrastructure accountable.

We envision an edge-driven cooperative auditing framework, in which MEC servers voluntarily audit each other. The auditing server (*i.e.,* verifier) offloads a task, with known correct result, to a peering server (*i.e.,* prover). On receiving the task's result, the verifier validates its correctness and reports the prover if the result is incorrect. This framework will be accompanied by a reputation system to update the MEC servers' reputations based on their audit histories. These reputation scores will be utilized to adjust the frequency of the auditing process, the peer selection, and further prevent service offloading to the disreputable entities.

Table 1: Throughput measurement of security microservices on various hosts (Mbps)

	Pi		Laptop		Server	
	Bare-Metal	Docker	Bare-Metal	Docker	Bare-Metal	Docker
Ed25519 Sig.	12.8	12.7	163.8	142.6	227.6	207.7
Ed25519 Ver.	4.6	5.1	57.8	54.9	82.5	79.1
AES256 GCM Enc.	18.5	22.6	216.7	145.2	291.6	182.1
AES256 GCM Dec.	19.1	23.87	236.6	154	304.8	191.5
RSA4096 Enc.	0.7	0.7	9.6	8.0	12	10.3
RSA4096 Dec.	0.01	0.01	0.19	0.18	0.24	0.23

5.4 Open Challenges

Trust Relationships. The trust between the tenants and stakeholders allow the tenants of Indirect services to trust stakeholders with their subscribers' information, access pattern, and service consumption. The challenge is for the stakeholders to guarantee that such information is treated privately without being disclosed to competitors; how to keep stakeholders accountable for such information usage? Another concern is, how do stakeholders provide accounting information for tenants like Netflix that lose track of their subscribers' preferences and service utilization due to in-network processing and caching.

Optimal Microservice Chaining. Service function chaining allows for fast and efficient service development. However, it is challenging to achieve optimal microservice chaining while satisfying the expected throughput and latency demands of the applications. From the security standpoint, chaining increases the attack surface caused by interactions among microservices. Thus, a fundamental challenge faced by the community is the design and implementation of secure APIs that can provide an isolated and secure edge environment for microservices inter-process communication.

Algorithmic and Data. Edge computing and microservice technologies have enabled new data and algorithmic approaches to be applied to the deployment and operations of telecommunications networks. For example, how do we design network mechanisms to ensure that data-driven network solutions, such as learning-based routing or transport are compliant with confidentiality, integrity and availability metrics, in combination with performance constraints such as bandwidth pricing, latency, or resilience?

Network Virtualization and Interoperability. Network function virtualization and software-defined networking technologies have enabled new types of network monitoring and surveillance techniques to predict and rapidly adapt to network events such as congestion, node failures, and detection or isolation of security threats. Bringing those services at the edge is challenging. How do we securely stitch slices across different domain or providers without releasing sensitive information? As it is hard to secure or ensure adoption of secure BGP solutions, we predict that it will be challenging to design inter-provider microservice solutions that span different federated service or infrastructure providers.

System and Performance. Edge computing and microservice technologies present significant opportunities but also new operational and security challenges for network operators. The difficulties of specifying and integrating these new systems were foreseen but have not yet been fully solved. How do we deliver deterministic performance, security, and reliability as technologies and markets change, and regulations evolve in different jurisdictions?

6 EVALUATION

We implemented six fundamental security microservices, using Python's cryptography library version 2.7, including *digital signature and verification (Ed25519), symmetric encryption and decryption (AES256 in GCM mode), asymmetric encryption and decryption (RSA4096)*. Initially, we unit test the microservices to benchmark their performance in two settings: execution on the host machine and inside a Docker container (version 18.09). We use a server-class machine (Intel Core-i7, 4.0 GHz processor), a laptop (Intel Core-i7, 2.4 GHz processor) with both running Ubuntu 19.04, and a Raspberry Pi-3 running Raspbian. As shown in Table 1, the majority of microservices show similar performance on the container and bare-metal except for AES microservices–AES running in the container (both laptop and server) resulted in lower performance when compared with the bare-metal execution. One possibility for this discrepancy is the CPU acceleration that Advanced Encryption Standard Instruction Set (AES-NI) provides for AES operations running on Intel processors. In Raspberry Pi-3 experiments, a few containerized microservices outperforming their bare-metal executions. We believe this is due to containers running Ubuntu, which is more optimized, compared to Raspbian running for Pi's bare-metal.

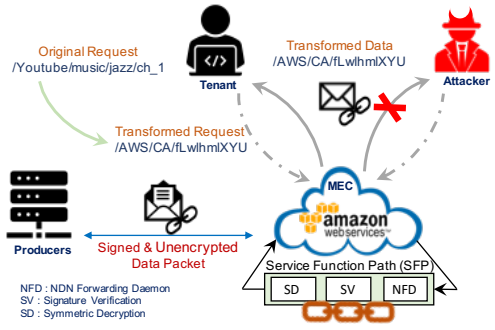
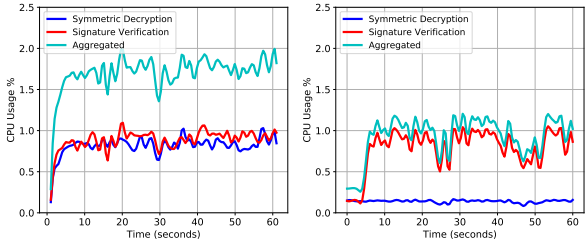


Figure 3: Anonymization service evaluation setup.

We also implemented a proof of concept anonymization service, similar to [25], by chaining signature verification (SV), symmetric decryption (SD), and NDN's forwarding daemon (NFD) microservices to enable data-centric communication (Figure 3). The NFD microservice processes the requests and dispatches them to the SV and SD microservices. To evaluate our application's performance, we experimented its resource utilization when: *(i)* a legitimate user anonymously requesting data with valid signature; and *(ii)* a malicious user requesting data with invalid signatures to orchestrate a DDoS attack on the edge's resources.

The user anonymously requests data by eliciting eight requests per second. The edge server uses the SV and SD microservices to



(a) Legitimate User (b) Malicious User

Figure 4: CPU utilization of the our application running on the server for (a) legitimate user and (b) malicious user.

verify the user's signature on the request and decrypt the requested name if the signature is valid. If the SV microservice fails, the edge server drops the request without passing it to the SD microservice. As shown in Figure 4(a), the SV and SD microservices utilize roughly 1% of the CPU time when the edge server is processing the legitimate traffic. However, when the application processes the malicious traffic (Figure 4(b)), the SD microservice does not utilize the CPU since the SV microservice discards the malicious traffic. This result demonstrates the capability of our application in protecting the edge's resources when the server is under attack.

7 CONCLUSIONS

The infirmity of the contemporary security measures in coping with the evolving attack surface calls for the design of advanced security services. A viable solution is virtualization and deployment of novel defense mechanisms at the network edge for serving a diverse set of tenants. In this paper, we proposed a SECaaS framework for security service deployment at the edge of data-centric Eyeball ISPs. We envision agile and economical service development and provisioning by utilizing virtualization technologies, such as microservicing and service chaining. In the future, we plan to build a prototype of the IDPS and access control enforcement services with dynamic service orchestration.

8 ACKNOWLEDGMENTS

Research supported by NSF awards #1800088; #1719342; #1345232; #1647084; #1836906, and Intel grant #34627535. Any opinions, conclusions or recommendations in this material are those of the authors and do not necessarily reflect the views of the federal government and other funding agencies.

REFERENCES

[1] M. Antonakakis, T. April, M. Bailey, M. Bernhard, E. Bursztein, J. Cochran, Z. Durumeric, A. Halderman, L. Invernizzi, and M. Kallitsis. Understanding the mirai botnet. In *USENIX Security Symposium*, pages 1092–1110, 2017.

[2] P. Garcia Lopez, A. Montresor, D. Epema, A. Datta, T. Higashino, A. Iamnitchi, M. Barcellos, P. Felber, and E. Riviere. Edge-centric computing: Vision and challenges. *ACM SIGCOMM Computer Communication Review*, 45(5):37–42, 2015.

[3] T. Yu, S. K. Fayaz, M. P. Collins, V. Sekar, and S. Seshan. Psi: Precise security instrumentation for enterprise networks. In *Network and Distributed System Security Symposium*, 2017.

[4] H. Li, H. Hu, G. Gu, G. J. Ahn, and F. Zhang. vnids: Towards elastic security with safe and efficient virtualization of network intrus ion detection systems. In *Proceedings of ACM SIGSAC Conference on Computer and Communications Security*, pages 17–34, 2018.

[5] K. Bhardwaj, J. C. Miranda, and A. Gavrilovska. Towards iot-ddos prevention using edge computing. In {*USENIX*} *Workshop on Hot Topics in Edge Computing*, 2018.

[6] A. Mtibaa, R. Tourani, S. Misra, J. Burke, and L. Zhang. Towards edge computing over named data networking. In *International Conference on Edge Computing (EDGE)*, pages 117–120. IEEE, 2018.

[7] L. Zhang, A. Afanasyev, J. Burke, V. Jacobson, P. Crowley, C. Papadopoulos, L. Wang, B. Zhang, et al. Named data networking. *ACM SIGCOMM CCR*, 44(3):66–73, 2014.

[8] C. Tschudin and M. Sifalakis. Named functions and cached computations. In *Consumer Communications and Networking Conference (CCNC)*, pages 851–857. IEEE, 2014.

[9] M. Sifalakis, B. Kohler, C. Scherb, and C. Tschudin. An information centric network for computing the distribution of computations. In *Proceedings of the Conference on Information-Centric Networking*, pages 137–146. ACM, 2014.

[10] M. Król and I. Psaras. Nfaas: named function as a service. In *Proceedings of the Conference on Information-Centric Networking*, pages 134–144. ACM, 2017.

[11] X. Marchal, T. Cholez, and O. Festor. μ NDN: an orchestrated microservice architecture for named data networking. In *ACM Conference on Information-Centric Networking*.

[12] M. Sardara, L. Muscariello, J. Augé, M. Enguehard, A. Compagno, and G. Carofiglio. Virtualized icn (vicn): towards a unified network virtualization framework for icn experimentation. In *Conference on Information-Centric Networking*, pages 109–115. ACM, 2017.

[13] T. Combe, W. Mallouli, T. Cholez, G. Doyen, B. Mathieu, and E. M. De Oca. An sdn and nfv use case: Ndn implementation and security monitoring. In *Guide to Security in SDN and NFV*, pages 299–321. Springer, 2017.

[14] C. Liang, F. Yu, H. Yao, and Z. Han. Virtual resource allocation in information-centric wireless networks with virtualization. *IEEE Transactions on Vehicular Technology*, 65(12):9902–9914, 2016.

[15] V. Varadharajan and U. Tupakula. Security as a service model for cloud environment. *IEEE Transactions on network and Service management*, 11(1):60–75, 2014.

[16] D. Migault, M. Simplicio, B. Barros, M. Pourzandi, T. Almeida, E. Andrade, and T. Carvalho. A framework for enabling security services collaboration across multiple domains. In *International Conference on Distributed Computing Systems (ICDCS)*, pages 999–1010. IEEE, 2017.

[17] T. Mauro. Adopting microservices at netflix: Lessons for architectural design. [online], 2015. https://www.nginx.com/blog/microservices-at-netflix-architectural-best-practices/.

[18] A. Boudi, I. Farris, M. Bagaa, and T. Taleb. Lightweight virtualization based security framework for network edge. In *Conference on Standards for Communications and Networking (CSCN)*, pages 1–6. IEEE, 2018.

[19] A. Karami and M. Guerrero-Zapata. A fuzzy anomaly detection system based on hybrid pso-kmeans algorithm in content-centric networks. *Neurocomputing*, 149:1253–1269, 2015.

[20] R. Tourani, S. Misra, T. Mick, and G. Panwar. Security, privacy, and access control in information-centric networking: A survey. *IEEE Communications Surveys & Tutorials*, 20(1):566–600, 2018.

[21] L. Deri and A. Del Soldato. An architecture for distributing and enforcing iot security at the network edge. In *International Conference on Internet of Things (iThings) and IEEE Green Computing and Communications (GreenCom) and IEEE Cyber, Physical and Social Computing (CPSCom) and IEEE Smart Data (SmartData)*, pages 211–218. IEEE, 2018.

[22] R. S. Da Silva and S. D. Zorzo. An access control mechanism to ensure privacy in named data networking using attribute-based encryption with immediate revocation of privileges. In *Consumer Communications and Networking Conference*, pages 128–133. IEEE, 2015.

[23] R. Tourani, R. Stubbs, and S. Misra. TACTIC: Tag-based access control framework for the information-centric wireless edge networks. In *International Conference on Distributed Computing Systems*, pages 456–466. IEEE, 2018.

[24] E. Erdin, C. Zachor, and M. H. Gunes. How to find hidden users: A survey of attacks on anonymity networks. *Communications Surveys & Tutorials*, 17(4):2296–2316, 2015.

[25] R. Tourani, S. Misra, J. Kliewer, S. Ortegel, and T. Mick. Catch Me If You Can: A Practical Framework to Evade Censorship in Information-Centric Networks. In *Proceedings of the International Conference on Information-Centric Networking*, pages 167–176. ACM, 2015.

[26] S. DiBenedetto, P. Gasti, G. Tsudik, and E. Uzun. Andana: Anonymous named data networking application. *Arxiv preprint arXiv:1112.2205*, 2011.

[27] S. Yi, C. Li, and Q. Li. A survey of fog computing: concepts, applications and issues. In *Proceedings of the 2015 workshop on mobile big data*, pages 37–42. ACM, 2015.

[28] M. Satyanarayanan. The emergence of edge computing. *Computer*, 50(1):30–39, 2017.

[29] N. Akhtar, I. Matta, A. Raza, L. Goratti, T. Braun, and F. Esposito. Virtual function placement and traffic steering over 5g multi-technology networks. In *Conference on Network Softwarization and Workshops (NetSoft)*, pages 114–122. IEEE, 2018.