Intent-Based Attack Mitigation through Opportunistic Synchronization of Micro-Services

Do Duc Anh Nguyen, Pierre Alain, Fabien Autrel, Ahmed Bouabdallah, Jérôme Franc

Intent-Based Attack Mitigation through Opportunistic Synchronization of Micro-Services

Do Duc Anh Nguyen*, Pierre Alain†, Fabien Autrel*, Ahmed Bouabdallah*, and Jérôme François‡
*SOTERN - IRISA (UMR CNRS 6074), IMT Atlantique,
†SOTERN - IRISA (UMR CNRS 6074), Université de Rennes,
‡SnT, University of Luxembourg and Inria Nancy Grand Est
*firstname.lastname@imt-atlantique.fr, †pierre.alain@irisa.fr, ‡jerome.francois@uni.lu

***Abstract*—The escalating number of cyberattacks poses a significant threat to digital infrastructures. Defining and deploying accurate countermeasures is challenging because of (1) the variety of threats and their possible evolution over time and (2) the need to enforce them as fast as possible, especially for fast-propagating attacks. Intent-Based Networking (IBN) stands for a promising solution for security management, especially to mitigate attacks through the specification of reaction intents, saving time and avoiding error-prone tasks. Nevertheless, most current IBN solutions rely on centralized architectures performing time-consuming operations, which makes them inappropriate to timely deploy countermeasures, especially in the case of fast-propagating attacks spreading large-scale systems. As a solution to shorten the reaction time while supporting scalability, we first consider fast micro-services technologies (e.g., Unikernels) as the substrate of security functions acting as Policy Enforcement Points (PEP). Second, we propose to enable an opportunistic synchronization of those PEPs to react, at least partially but autonomously, against the ongoing attacks in a decentralized fashion. Such a solution raises challenges related to the consistency and performance of the overall enforced reaction policies. This paper presents the early stage of the PhD, outlining the specific challenges, limitations, and research required to leverage decentralized reaction using opportunistic synchronization of micro-services in an IBN framework for security.**

***Index Terms*—Decentralized mitigation, Reaction policy, IBN, micro-services, Opportunistic synchronization**

I. Introduction

In recent years, network infrastructures have been exposed to a wide range of cyber-attacks[1]. Defining and deploying accurate countermeasures is complex due to the heterogeneity of attacks and their potential evolution over time. More specifically, fast-propagating threats such as worms require a fast reaction time to minimize their impact. The Intent-Based Networking (IBN) approach is used to express the desired outcome of a network configuration, called an intent, without specifying the details of the operations to achieve it. By applying the IBN concept to security [1], the deployment and configuration of security functions acting as Policy Enforcement Points (PEP) (e.g., firewall, Intrusion Detection System (IDS)) can be automated.

However, inferring a reaction policy from a set of intents, i.e., computing the PEP low-level configuration, can be far slower than the expected time to efficiently mitigate an ongoing attack. For instance, considering the case of fast-propagating worms such as WannaCry [2] or NotPetya [3], we have empirically assessed that up to 14 hosts can be infected in 100 seconds in a 100-node local network, while the time needed for a centralized policy-based system [4], executing high-level policy optimization operations, to compute a novel global security policy can be twice for similar network sizes, thus making the reaction inefficient to mitigate worm propagation.

To enforce fast reaction policy deployment, we propose to deploy and configure PEPs in an autonomous and decentralized fashion. To that aim, we consider (1) fast micro-services technologies, such as Unikernels, and (2) opportunistic synchronization mechanisms between them. Micro-services are software-based functions that are decomposed from a large, complex application into independent services. They allow the system to deploy and scale functions independently, reducing the risk of system-wide failure and enabling greater agility in changing behavior. Furthermore, by leveraging data-plane packets to synchronize those PEPs regarding the enforcement of reaction mechanisms, we propose an opportunistic approach that provides scalable and extremely fast communication means for PEPs to mitigate attacks. This paper presents the early stage of the PhD, explores the ideas, and raises the following research questions: (1) What are the properties of the propagation mechanisms used by fast-propagating worms? (2) How can we translate intents into reaction policies that can respond to attacks? (3) What methods can be used to efficiently provision micro-services to implement these reaction policies? (4) How can micro-services perform opportunistic synchronization in response to attacks?

The rest of the paper is structured as follows: Section II provides the state of the art in IBN for security, along with contributions on micro-services and signaling methodology. Section III discusses more concrete challenges and some early ideas or directions to answer the above research questions. Section IV presents the current status of our topic. Finally, Section V contains the final remarks of the paper.

II. State of the Art

In this section, we look at the existing contributions of IBN applied to security, the use of micro-services, signaling methods for reactive actions, and finally our conclusion about these contributions.

[1]The 2021 Security Outcomes Study, Cisco

A. Recent Advances in Intent-based Security

According to [1], a typical IBN system handles in total three main processes: translation, activation, and assurance. The translation process consists of translating intents into low-level configurations, while the activation process orchestrates them by provisioning the required services and their actual behavior. The assurance process is required to ensure compliance of intents. In the remainder, we categorize studies integrating IBN for system security into five groups:

1) Cloud Service Management: The authors of [5] propose an implementation of the Interface to Network Security Functions (I2NSF) framework [6], which was introduced by the Internet Engineering Task Force (IETF). The translation process is based on automata theory and then converted to device-level information to configure appropriate NSFs. In addition, the authors of [7] present the Intent-based Cloud Service Management (ICSM) framework, where user requests for cloud services are expressed in natural language and parsed into a structured format.

2) Secure Connection: In [8], the authors identify keywords used to express intents, which are mapped to a list of connectivity services. Then, the IBN framework proposed in [9] allows users to express intents to request a secure connection with some constraints (e.g., bandwidth, latency). By mapping constraints to the list of encryption layers, a suitable encryption layer is selected. The authors of [10], [11] propose the North-Bound Interface (NBI), which provides the abstraction interface for client applications to request connectivity services. In [10], intents are defined with constraints that can be recognized by an ONOS controller, while in [11], they refer more concretely to their implementation, including a proposed architecture.

3) Access Control Management: The system presented in [12] allows users to automatically generate or update Access Control Lists (ACLs) in a declarative manner. User requirements, network configuration, and topology are ingested by a Satisfiability Modulo Theory (SMT) solver to provide the final set of ACL rules. In [13], Business Rule Management Systems (BRMS) syntax is used to express intents that are translated into OpenFlow rules.

4) DDoS Mitigation: The IBN frameworks proposed in [14], [15] allow the management of network behavior. They also enable users to define a threshold to identify a Distributed Denial of Service (DDoS) attack and enforce countermeasures. In [14], user intents in Nile format are parsed into multiple predefined P4 code templates for specific actions. The final P4 programs, containing all the parsed P4 code templates, are then compiled on programmable switches. In [15], the framework captures users' verbal intents, which are then processed by a voice assistant. Instead of expressing intents in advance, the generation of intents is automated in [16] to perform Moving Target Defense (MTD) based on alerts generated by an IDS.

5) Optimal Security Provisioning: A formal approach based on Maximum SMT proposed in [4] can achieve the optimal solution based on high-level policies. From translated policies and a network service graph, this approach can compute the optimal policies and allocation position for firewalls based on some modeled constraints. Following this idea, the authors propose a self-protection system [17] for IoT, where response rules are automatically generated by monitoring tools (e.g., SIEM) and then recomputed with the existing rules to obtain the optimal solution.

B. Micro-services

Micro-services [18] are based on the concept of breaking complex applications into multiple small services. The virtualized IDSs provided by [19] are micro-services that can be customized to handle different policies. For cloud management, the authors of [20] propose to automatically select the appropriate micro-services to secure a web service. Security policies can also be updated and handled separately in security micro-services. In IoT, [21] uses micro-services as intermediate nodes between IoT devices and clients to manage communication. Finally, [22] proposes edge computing architectures where security micro-services are deployed and managed in edge gateways between web clients and local networks of IoT devices.

C. Signaling Methodologies

For opportunistic synchronization among PEPs, effective signaling is essential for information sharing during attacks. We categorize existing contributions into the four groups below. All these groups target DDoS attacks, requiring quick and low-latency responses.

1) Client Host Protection: The authors of [23] propose a defense strategy to mitigate DDoS attack impact using rings of Intrusion Prevention Systems (IPSs) centered around client hosts. When an IPS detects an attack, hop-by-hop communication on the same ring is triggered to infer the plausibility of the attack based on the packet rate. In [24], the authors focus on a specific case, namely Distributed Reflection Denial of Service (DRDoS), where client hosts identify attacks by checking packet routing paths.

2) Domain Collaboration: In [25], the authors choose to mitigate the impact of DDoS by enabling communication between IPSs of multiple domain networks. Similarly, collaboration between Software Defined Networking (SDN) controllers in different domains can be used to communicate with each other about attack details [26].

3) Autonomous System Collaboration: Autonomous systems (ASs) in [27] can mitigate DDoS attacks and report a list of suspicious IP addresses in blockchain-based smart contracts. Authorized ASs near the attack source can access this list to block traffic. In [28], ASs report their flows and actions (e.g., drop, forward) to a centralized ledger. Downstream ASs label these actions through deep analysis or captchas to detect bots, providing feedback to upstream ASs.

4) IP Traceback: Source traceback mechanisms are also investigated to respond to DDoS attacks. In [29], unique marks are requested by routers that detect abnormal traffic to a central server. The authors of [30] enable switches in the network to embed their IDs in the data plane of packets.

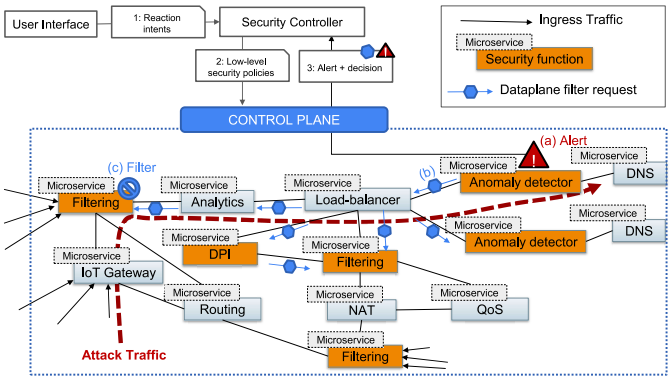

Fig. 1. Target architecture

D. Synthesis

Numerous contributions propose IBN systems to provide reliable protection mechanisms. However, they are inevitably hampered by the latency of complex computations, which makes them unscalable on large networks. Extensive studies of micro-services and signaling methodologies provide us with effective ways to synchronize micro-services for rapid attack response. However, the integration of these two concepts remains an open question. To our knowledge, most contributions use micro-services to enforce security policies, but none of them consider reaction intents in the case of attack events. Leveraging the data plane for message signaling, inspired by overlay protection mechanisms for collaborative attack detection, could facilitate our opportunistic synchronization approach.

III. Challenges and Methodologies

The envisioned target architecture combining IBN and the opportunistic approach and their interactions is illustrated in Figure 1. Users first express reaction intents (1) via the user interface. The security controller provisions appropriate micro-services and their low-level security policies through the control plane (2). When a micro-service detects an attack, it can immediately proceed by configuring a countermeasure or propagating a request to the network to find other micro-services that handle the appropriate security functions. Once the prompt countermeasure is applied, these micro-services may notify the security controller to recompute the accurate one. To pursue this research, this section discusses some of the challenges and limitations as well as the methods planned to address them.

A. Intent Translation and provisioning

The first operation in the intent pipeline is to represent user intents in an intermediate language. This language should be generic enough to express reaction requirements and provide some abstraction independent from the various low-level configuration languages used in the micro-services. This intermediate language is then analyzed and translated into low-level configuration languages. This analysis and translation, being centralized by design, induces a response time that cannot cope with fast-propagating attacks such as worms.

After translation, provisioning micro-services and their security policies presents its own challenges. For example, if solutions provided by IBN systems are not especially designed and optimized to manage micro-services, the complexity of the network can increase due to redundant instances.

B. Opportunistic Decentralized Mitigation

Conducting mitigation without considering complex computation or network topology is challenging. Micro-services need a way to quickly communicate and synchronize with each other to provide prompt countermeasures. Moreover, due to the different capabilities of micro-services, it is quite time-consuming to investigate the appropriate one to counter specific attacks.

To promptly react to attacks, we propose to embed security requests into the data-plane traffic as illustrated in Figure 1. In this example, an attack targets DNS micro-services. Although an anomaly detector on the traffic path detects it (a), it does not have filtering capabilities but can request another micro-service to perform this function. From a general point of view, this request could be sent backward along the attack path. However, attack paths might be multiple in reality, and forwarding rules can change. Besides, micro-service deployments are flexible and can thus rapidly change. We thus adopt an epidemic-like approach to deliver such a request in an opportunistic manner by signaling the request to the different neighbors using existing data packets on the data plane to embed the request (b). Different propagation strategies are possible: broadcast, neighbor sampling, etc. The chosen strategy directly impacts the reaction time and the induced overhead, which also justifies embedding requests in existing packets. In the example of Figure 1, the filter request is broadcasted until micro-services with the appropriate security capabilities (*i.e.*, filtering) can generate and enforce the new security policies. As a result, the attack traffic is filtered (c).

With this opportunistic data-plane-based mechanism, we ensure an initial set of countermeasures to be applied immediately when attacks are detected until an optimal reaction strategy at network scale is derived. Micro-services do not need to consider the entire network topology or wait for the security controller to compute an overall optimal countermeasure deployment strategy.

However, there are still challenges to overcome. We can mention among them: propagation strategy definition depending on the type of request and/or attack to mitigate, efficient data-plane signaling to limit the induced overhead, conflict resolution at a local level since possible non-consistent requests can be signaled and security of the opportunistic signaling to avoid misuse by attackers.

IV. Current Status

At the current stage, we have studied the I2NSF framework implementation proposed by [5] and provided a conflict detection and resolution approach [31] for the I2NSF security

controller after translation, which performs pair-wise comparison for detection and enforces separation constraints together with partial ordering relationships for resolution. A conflict is detected when two rules share the same attributes (e.g., *time*) while their actions are contradictory (*drop* vs. *pass*). Although the algorithm has polynomial complexity, our results show that the system is unscalable for large numbers of policies.

In addition to this first result, we are analyzing the propagation mechanisms of fast-spreading worms such as WannaCry and NotPetya on virtualized networks composed of dozens of Virtual Machines (VMs). They are both using the EternalBlue exploit to achieve fast propagation speeds, but with different spreading strategies. NotPetya first probes its local network and performs a sequential exploitation of each detected VM. Regarding WannaCry, it performs exploitation on VMs detected in parallel. As a result, for a 50-VM scenario, the total propagation time for WannaCry to spread its whole network is 836.11 seconds on average, which is quite shorter than that of NotPetya, which is 1454.08 seconds. It is worth noting that once a VM is infected by these two malware instances, it becomes a new infector also trying to spread. This means that the propagation speed accelerates according to the number of VMs infected in networks. These results confirm that the response time of centralized approaches to the detection of such attacks is slow enough to allow such worms to spread to machines in a subnet while the reaction is being processed.

V. Conclusion

Fast-propagating attacks, such as worms, leverage the need for fast reaction mechanisms. We propose to use IBN for automated deployment of reaction policies based on user intents, coupled with micro-services for policy enforcement. The opportunistic synchronization approach empowers micro-services to autonomously yield prompt responses. Our current work consists of comprehensively assessing to what extent the standard IBN architecture exhibits scalability issues. Following this, we aim at addressing the challenges related to synchronization between micro-services. More concretely, our plan is to design a first opportunistic mechanism to synchronize micro-services enforcing an autonomous reaction policy and compare its performance and reliability to that of the standard IBN architecture.

Acknowledgment

This work has been partially supported by the French National Research Agency under the France 2030 label (Superviz ANR-22-PECY-0008). The views reflected herein do not necessarily reflect the opinion of the French government.

References

[1] A. Leivadeas *et al.*, "A survey on intent-based networking," *IEEE Communications Surveys & Tutorials*, vol. 25, no. 1, pp. 625–655, 2023.

[2] S. Mohurle and M. Patil, "A brief study of wannacry threat: Ransomware attack 2017," *International journal of advanced research in computer science*, vol. 8, no. 5, pp. 1938–1940, 2017.

[3] S. Y. A. Fayi, "What petya/notpetya ransomware is and what its remidiations are," in *15th international conference on information technology*. Springer, 2018, pp. 93–100.

[4] D. Bringhenti, G. Marchetto, R. Sisto, F. Valenza, and J. Yusupov, "Automated optimal firewall orchestration and configuration in virtualized networks," in *IEEE/IFIP NOMS*, 2020, pp. 1–7.

[5] J. Kim *et al.*, "IBCS: Intent-based cloud services for security applications," *IEEE Comm. Mag.*, vol. 58, no. 4, pp. 45–51, 2020.

[6] D. Lopez, E. Lopez, L. Dunbar, J. Strassner, and R. Kumar, "Framework for Interface to Network Security Functions," RFC 8329, Feb. 2018.

[7] W. Chao and S. Horiuchi, "Intent-based cloud service management," in *ICIN*, 2018, pp. 1–5.

[8] M. Toy, "Intent-based networking for connectivity and cloud services," *Advances in Networks*, vol. 9, no. 1, p. 19, 2021.

[9] T. Szyrkowiec *et al.*, "Automatic intent-based secure service creation through a multilayer SDN network orchestration," *Journal of Optical Communications and Networking*, vol. 10, no. 4, pp. 289–297, 2018.

[10] F. Pederzolli *et al.*, "SDN application-centric orchestration for multi-layer transport networks," in *ICTON*. IEEE, 2016, pp. 1–4.

[11] P. Sköldström *et al.*, "Dismi - an intent interface for application-centric transport network services," in *ICTON*, 2017, pp. 1–4.

[12] B. Tian *et al.*, "Safely and automatically updating in-network ACL configurations with intent language," in *Proceedings of the ACM Special Interest Group on Data Communication*, 2019, pp. 214–226.

[13] S. Rivera, J. Griffioen, Z. Fei, and J. H. Hayes, "Expressing and managing network policies for emerging HPC systems," in *PEARC on Rise of the Machines (learning)*, 2019, pp. 1–7.

[14] M. Riftadi and F. Kuipers, "P4I/O: Intent-based networking with P4," in *NetSoft*. IEEE, 2019, pp. 438–443.

[15] M. Jain *et al.*, "Intent-based, voice-assisted, self-healing SDN framework," *JNCET*, vol. 10, no. 2, 2020.

[16] M. F. Hyder and M. A. Ismail, "INMTD: Intent-based moving target defense framework using software defined networks," *ETASR*, vol. 10, no. 1, pp. 5142–5147, 2020.

[17] D. Bringhenti *et al.*, "Automatic, verifiable and optimized policy-based security enforcement for SDN-aware IoT networks," *Computer Networks*, vol. 213, p. 109123, 2022.

[18] I. Nadareishvili *et al.*, *Microservice architecture: aligning principles, practices, and culture*. O'Reilly Media, Inc., 2016.

[19] N. Zhang, H. Li, H. Hu, and Y. Park, "Towards effective virtualization of intrusion detection systems," in *Proceedings of the ACM International Workshop on Security in SDN & NFV*, 2017, pp. 47–50.

[20] S. Abidi *et al.*, "A web service security governance approach based on dedicated micro-services," *Procedia Computer Science*, vol. 159, pp. 372–386, 2019.

[21] D. Lu, D. Huang, A. Walenstein, and D. Medhi, "A secure microservice framework for IoT," in *IEEE SOSE*, 2017, pp. 9–18.

[22] W. Jin, R. Xu, T. You, Y.-G. Hong, and D. Kim, "Secure edge computing management based on independent microservices providers for gateway-centric IoT networks," *IEEE Access*, vol. 8, pp. 187 975–187 990, 2020.

[23] J. François, I. Aib, and R. Boutaba, "Firecol: a collaborative protection network for the detection of flooding DDoS attacks," *IEEE/ACM Transactions on networking*, vol. 20, no. 6, pp. 1828–1841, 2012.

[24] A. Bremler-Barr and M. Sabag, "Preventing the flood: Incentive-based collaborative mitigation for drdos attacks," in *2022 IFIP Networking Conference (IFIP Networking)*. IEEE, 2022, pp. 1–9.

[25] B. Rashidi *et al.*, "Cofence: A collaborative ddos defence using network function virtualization," in *CNSM*. IEEE, 2016, pp. 160–166.

[26] S. Hameed and H. A. Khan, "Leveraging SDN for collaborative DDoS mitigation," in *NetSys*. IEEE, 2017, pp. 1–6.

[27] Z. Abou El Houda, A. Hafid, A. Khoukhi, LyeHafid, and L. Khoukhi, "Co-iot: A collaborative ddos mitigation scheme in iot environment based on blockchain using sdn," in *2019 IEEE Global Communications Conference (GLOBECOM)*. IEEE, 2019, pp. 1–6.

[28] A. Dutta, E. Al-Shaer, and B.-T. Chu, "A collaborative & distributed framework for defending distributed denial of service (ddos) attack," in *Proceedings of the 16th Annual Symposium on Information Assurance (ASIA'21)*, 2021, pp. 62–72.

[29] S. Yu, W. Zhou, S. Guo, and M. Guo, "A feasible IP traceback framework through dynamic deterministic packet marking," *IEEE Transactions on Computers*, vol. 65, no. 5, pp. 1418–1427, 2015.

[30] R. Wang *et al.*, "In-band network telemetry based fine-grained traceability against IP address spooling attack," in *ACM ICEA*, 2021, pp. 229–233.

[31] A. Nguyen, F. Autrel, A. Bouabdallah, and G. Doyen, "A robust approach for the detection and prevention of conflicts in I2NSF security policies," in *IEEE/IFIP NOMS 2023 2023*. IEEE, 2023.