Combining AntibIoTic with Fog Computing: AntibIoTic 2.0

Michele De Donno
DTU Compute
Technical University of Denmark (DTU)
mido@dtu.dk

Nicola Dragoni
DTU Compute
Technical University of Denmark (DTU), Denmark
and *AASS, Örebro University, Sweden*
ndra@dtu.dk

***Abstract*—The Internet of Things (IoT) has been one of the key disruptive technologies over the last few years, with its promise of optimizing and automating current manual tasks and evolving existing services. From the security perspective, the increasing adoption of IoT devices in all aspects of our society has exposed businesses and consumers to a number of threats, such as Distributed Denial of Service (DDoS) attacks. To tackle this IoT security problem, we proposed AntibIoTic 1.0 [1]. However, this solution has some limitations that make it difficult (when not impossible) to be implemented in a legal and controlled manner. Along the way, Fog computing was born: a novel paradigm that aims at bridging the gap between IoT and Cloud computing, providing a number of benefits, including security. As a result, in this paper, we present AntibIoTic 2.0, an anti-malware that relies upon Fog computing to secure IoT devices and to overcome the main issues of its predecessor (AntibIoTic 1.0). First, we present AntibIoTic 1.0 and its main problem. Then, after introducing Fog computing, we present AntibIoTic 2.0, showing how it overcomes the main issues of its predecessor by including Fog computing in its design.**

***Index Terms*—Fog Computing, Internet of Things, Security, Distributed Denial of Service, Malware, Anti-Malware**

I. Introduction

Internet of Things (IoT), Industrial Internet of Things (IIoT), and Industry 4.0 are some of the most hyped technologies of recent years. By interconnecting a large number of devices in both industry and consumer environments, these paradigms promise to innovate business models and improve the overall user experience. In 2017, the number of connected IoT devices was estimated around 20 billion and it is predicted that this number will be more than doubled by 2025[1]. Moreover, according to CISCO [2], by 2022 the 81% of the global IP traffic is expected to be driven by non-PC devices.

However, this exciting IoT revolution can soon become a security nightmare. Indeed, IoT security has represented in the last years one of the biggest cybersecurity challenges, and one of the most embarrassing failures of IoT ([3]–[5] to mention only a few examples). In fact, the large number of (I)IoT devices flooding the market are often poorly secured, thus easy prey of different families of malware. As a result, cybersecurity threats such as Distributed Denial of Service (DDoS) attacks have become more dangerous and easy to achieve than ever, since insecure IoT devices can often be used as sources of large attacks [6]. As a matter of fact, 2016 is still remembered as the year of Mirai, the IoT malware able to compromise approximately 500'000 IoT devices to convey one of the largest DDoS attacks ever recorded [7], [8]. After Mirai, the situation has not improved. According to Akamai, the average number of DDoS attacks per target increased of 19% from 2016 to 2017 [9], and the total number of DDoS attacks increased by 16% from 2017 to 2018 [10].

In this critical security situation, we proposed a palliative solution to improve the IoT security [1]: AntibIoTic (addressed in this paper as AntibIoTic 1.0). AntibIoTic 1.0 is a white worm that infects vulnerable devices and creates a botnet of safe systems, protecting them against IoT malware. Even though the solution is promising, it drags some issues that impede it to be used in a legal and controlled manner.

In the meantime, a new distributed computing paradigm has become more and more popular, especially in IIoT: Fog computing. Fog computing was born from the necessity of overcoming the challenges that the IoT evolution has posed to the Cloud, bridging the gap between IoT and Cloud computing [11]. Among others, one of the promises of Fog computing is to improve the security level of IoT and Cloud computing. This paper points to this aspect, focusing on the use of Fog computing as a security solution for Internet of Things.

As a result, we designed a new version of AntibIoTic that preserves the core idea of AntibIoTic 1.0 and overcomes its fundamental limitations by leveraging Fog computing. In this way, we bring the rationale of the AntibIoTic approach to its full potential. To the best of our knowledge, AntibIoTic 2.0 is the first Fog-based anti-malware for IoT.

A number of works in the literature can be related to AntibIoTic 2.0, but they significantly differ from it. Some works, such as [12], [13], present solutions that rely upon Fog computing to protect a specific target (either IoT devices or Cloud systems) against external attacks. However, AntibIoTic 2.0 has a different aim: to tackle the intrinsic insecurity of IoT devices, the root cause of many attacks. Our solution acts directly on IoT devices to secure them from inside and avoid their infection by malware, thus, reducing the possibility of perpetrating large-scale attacks (e.g., DDoS attacks) through IoT devices.

Some other works, such as [14], [15], propose interesting solutions to increase the overall security of the IoT. Although

[1]https://www.statista.com/statistics/471264/iot-number-of-connected-devices-worldwide/ *[Accessed on December 5th, 2018]*

978-1-7281-2365-3/19/$31.00 ©2019 IEEE

these solutions are compatible with ANTIBIOTIC and their integration with our solution is encouraged, they are still in the early stages and significantly differ from ANTIBIOTIC 2.0.

Finally, there are works, such as [16]–[18], that have a similar aim as ANTIBIOTIC 2.0 but do not rely on the Fog computing paradigm.

A. Contribution of the Paper

In this paper, we present ANTIBIOTIC 2.0, an anti-malware that relies upon Fog computing to secure (I)IoT devices. In particular, we introduce ANTIBIOTIC 1.0 [1] (the predecessor) and explain its key limitations. Then, after shortly introducing the Fog computing paradigm, we present ANTIBIOTIC 2.0, showing how the introduction of Fog computing in the picture brings the rationale of the ANTIBIOTIC approach to its full potential and solves the main limitations of ANTIBIOTIC 1.0.

B. Outline of the Paper

The paper is organized as follows. Section II presents ANTIBIOTIC 1.0 and its main shortcomings. Section III briefly introduces Fog computing. Section IV describes ANTIBIOTIC 2.0, showing how Fog computing can play a key role in securing (I)IoT devices. Finally, Section V wraps up the paper.

II. THE PREDECESSOR: ANTIBIOTIC 1.0

The ANTIBIOTIC solution was initially formalized through ANTIBIOTIC 1.0: a white worm against IoT-driven DDoS attacks [1]. Although the idea is promising, it has some issues that make it difficult, if not impossible, to be used in a legal and controlled manner. Nevertheless, ANTIBIOTIC 1.0 represents the origin of ANTIBIOTIC 2.0, thus, we consider relevant to briefly present ANTIBIOTIC 1.0 before introducing ANTIBIOTIC 2.0.

In this section, an overview of ANTIBIOTIC 1.0 is presented, describing the rationale behind it and its main features.

A. The Core

ANTIBIOTIC 1.0 was designed with the belief that the intrinsic vulnerability of IoT devices could be the solution to the IoT security problem rather than the problem itself. Indeed, similarly to the modus operandi of the homonym medications used against bacterial infections of the human body, ANTIBIOTIC 1.0 operates as a white worm that infects vulnerable IoT devices to create a botnet of safe systems, removing them from the clutches of other malware [1]. So, it basically spreads like malicious worms (e.g., Mirai) but, once the control of IoT units is gained, it tries to secure them instead of taking advantage of them. In addition, ANTIBIOTIC 1.0 includes some features to increase the awareness on the IoT security problem, potentially pushing security experts, devices manufacturers, and users to collaborate towards a more secure Internet of Things [1].

To support its rationale, ANTIBIOTIC 1.0 was designed with the infrastructure depicted in Figure 1 and mostly arisen from the Mirai one [8]. The major component of ANTIBIOTIC 1.0 are the Command-and-Control (CNC) Server and the ANTIBIOTIC Bot, each of them composed of several modules [1]. The *CNC Server* is the component interacting with human actors and bots. On the one side, it exposes data and statistics to users and admins, and it supports their interaction with the system. On the other side, the CNC Server interacts with the code running on each IoT device to monitor and control their operation. The ANTIBIOTIC *Bot* is the code running on the vulnerable IoT devices to secure them while scanning for new IoT units to extend the white botnet.

B. Main Features

ANTIBIOTIC 1.0 was designed with a set of features that are presented below divided into *core features* and *additional features*. This distinction will be useful when presenting ANTIBIOTIC 2.0 (Section IV), which inherits all the main features of its predecessor and evolves the additional ones.

The core features of ANTIBIOTIC 1.0 can be summarized as follows [1].

- *Sanitize IoT devices*: once the ANTIBIOTIC bot is running on the vulnerable IoT device, it cleans the device from other possible running malware and secures the perimeter to avoid future intrusion.
- *Secure IoT devices*: the ANTIBIOTIC bot is designed to apply the countermeasures necessary to fix the security vulnerabilities of the hosting IoT device (e.g., change admin credentials, update the firmware).
- *Resist to reboot*: differently from similar solutions, ANTIBIOTIC 1.0 is designed to be resistant to reboot, avoiding to be wiped off from the IoT device memory by simply restarting the system.

The additional features of ANTIBIOTIC 1.0 are briefly described below [1].

- *Publish data and statistics*: with the aim of increasing the awareness on the IoT security problem, ANTIBIOTIC 1.0 is designed to publish data and statistics related to the botnet of vulnerable IoT systems.
- *Expose interactive interfaces*: in order to let anyone join and improve the solution, ANTIBIOTIC 1.0 exposes different interfaces with different privileges.
- *Notify devices owner*: ANTIBIOTIC 1.0 is designed to notify the owner of the vulnerable IoT device, when possible, providing him with some advice to secure the system, avoiding the code to do it for him.

C. The Legal Issues

ANTIBIOTIC 1.0 is designed to act as a white worm to protect vulnerable IoT devices. Although the solution is valid from a technical level, its feasibility is inhibited by some legal issues, mainly arisen by the intent of gaining control and tamper with unsuspecting targets, even if only for security purposes.

Looking at the EU directive on attacks against information systems [19], it is possible to assert that ANTIBIOTIC 1.0 violates at least two articles: *article 3* - illegal access to information systems, *article 5* - illegal data interference. According

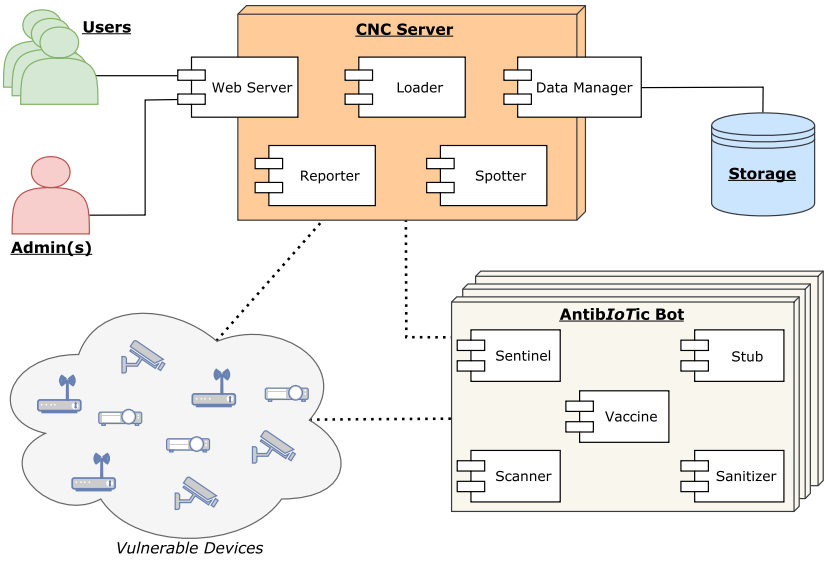

Fig. 1: The predecessor: ANTIBIOTIC 1.0 [1]

to article 3, intentionally gaining access, without right, to an information system is punishable as a criminal offence. Since ANTIBIOTIC 1.0 cannot protect IoT devices without gaining access to them, article 3 would be infringed. Also, in compliance with article 5, intentionally altering data on an information system, without right, is considered a criminal offence. Although the aim is to secure the hosting IoT device, ANTIBIOTIC 1.0 is designed to alter data of the infected device (e.g., changing password or updating the firmware), thus, article 5 would be violated.

The legal issues of ANTIBIOTIC 1.0 along with the potential of Fog computing motivated us in redesigning the system to overcome its legal issues and to enhance its potential, proposing ANTIBIOTIC 2.0.

III. FOG COMPUTING

Fog computing is a relatively new paradigm that promises to bridge the gap between Cloud computing and Internet of Things. In this section, we briefly define Fog computing and locate it in the Cloud-to-Things continuum.

The term Fog computing first appeared in the literature in 2012, when Bonomi et al. defined it as "a highly virtualized platform that provides compute, storage, and networking services between end devices and traditional Cloud Computing Data Centers, typically, but not exclusively located at the edge of network" [11]. Thereafter, several definitions of Fog computing have been proposed in the literature [20]–[27].

To date, we think that the clearest definition of Fog computing is the one provided from the OpenFog Consortium[2]: "Fog computing is a *system-level horizontal* architecture that *distributes* resources and services of computing, storage, control and networking anywhere along the *continuum from Cloud to Things*" [28].

[2]https://www.openfogconsortium.org/

Key concepts can be extrapolated from this definition. First, Fog computing is a *system-level* and *horizontal* architecture: it extends from end-devices, over the network edge, to the Cloud (not just at one side of an end-to-end system), and across multiple protocol layers (not just a specific one), supporting different types of industry and application domains. Secondly, it is a *distributed* approach: resources and services are distributed anywhere between the Cloud and IoT to overcome limitations of the centralized approach of Cloud computing. Finally, the definition includes the *Cloud to Things continuum*: Fog computing is not an alternative to the Cloud, rather a smart extension of Cloud computing, acting as the glue that bridges the gap between the Cloud and IoT.

In this context, a Fog node is "the physical and logical network element that implements Fog computing services" [29].

From an architectural point of view, the most common model for the Fog computing architecture is based on three layers [20], [22], [30], [31]: IoT, Fog computing, and Cloud computing. However, the OpenFog Consortium refined this architecture giving an inner structure to the Fog layer and referring to it as the N-tier architecture [29].

The main idea behind the Fog architecture, depicted in Figure 2, derives from the concept of Fog computing as a non-trivial extension of Cloud computing, in the Cloud to Things continuum. Indeed, there are still three main layers: IoT, Fog, and Cloud. However, the Fog layer is further structured with several tiers of Fog nodes (N-tiers): the farther Fog nodes move away from IoT devices, the more computational capabilities and intelligence they gain. In addition, Fog nodes at each layer can be linked together with the aim of providing additional features (e.g., fault tolerance, load balancing, resilience, etc.). Thus, Fog nodes can communicate both horizontally and vertically within the Fog layer. A detailed description of each layer is presented in [29], [31].

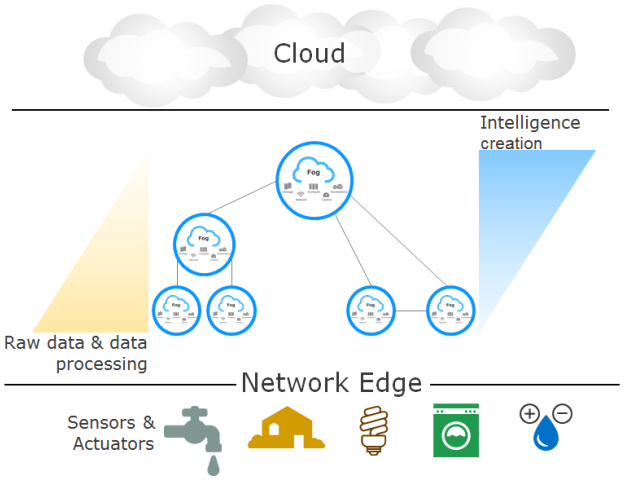

Fig. 2: OpenFog N-tier architecture (adapted from [29])

IV. AntibIoTic 2.0

AntibIoTic 1.0 represents a promising idea to improve the security level of the IoT, mainly fighting against IoT-driven DDoS attacks. However, it has some issues that make it difficult to be used in a legal and controlled way. That is why we introduce AntibIoTic 2.0: an enhanced version of AntibIoTic 1.0 that relies on Fog computing to overcome the issues of its predecessor.

In this section, the rationale behind AntibIoTic 2.0 is presented along with a summary of its main features.

A. The Idea

The aim of AntibIoTic 2.0 is to protect IoT devices against malware, in a legal and controlled manner. To achieve so, AntibIoTic 2.0 inherits the key features of its predecessor but includes Fog computing in the picture to overcome the issues of AntibIoTic 1.0.

The idea behind AntibIoTic 2.0 is to use a Fog node (or a federation of Fog nodes) to monitor and sanitize the IoT devices connected to it, allowing only safe ones to access the Internet. To this aim, the Fog node uploads on each IoT device an "anti-malware" (also addressed as AntibIoTic *Bot*) that sanitizes and secures them, and reports live information back to the Fog node. Then, depending on the information received from each IoT device, as well as the operation mode set for AntibIoTic 2.0, the Fog node decides if the host is allowed to connect to the Internet.

Since AntibIoTic 2.0 involves the use of one or more Fog nodes to protect local IoT devices, the solution can be easily framed in the N-tier architecture typical of Fog computing (presented in section III), as depicted in Figure 3. On the bottom, there is a Local Area Network (LAN) composed of IoT devices that relies on AntibIoTic 2.0 to enforce security. Then, the AntibIoTic Fog node is connected to the Cloud through one or more optional layers of Fog nodes that can provide enhancements and additional services. If desired, the

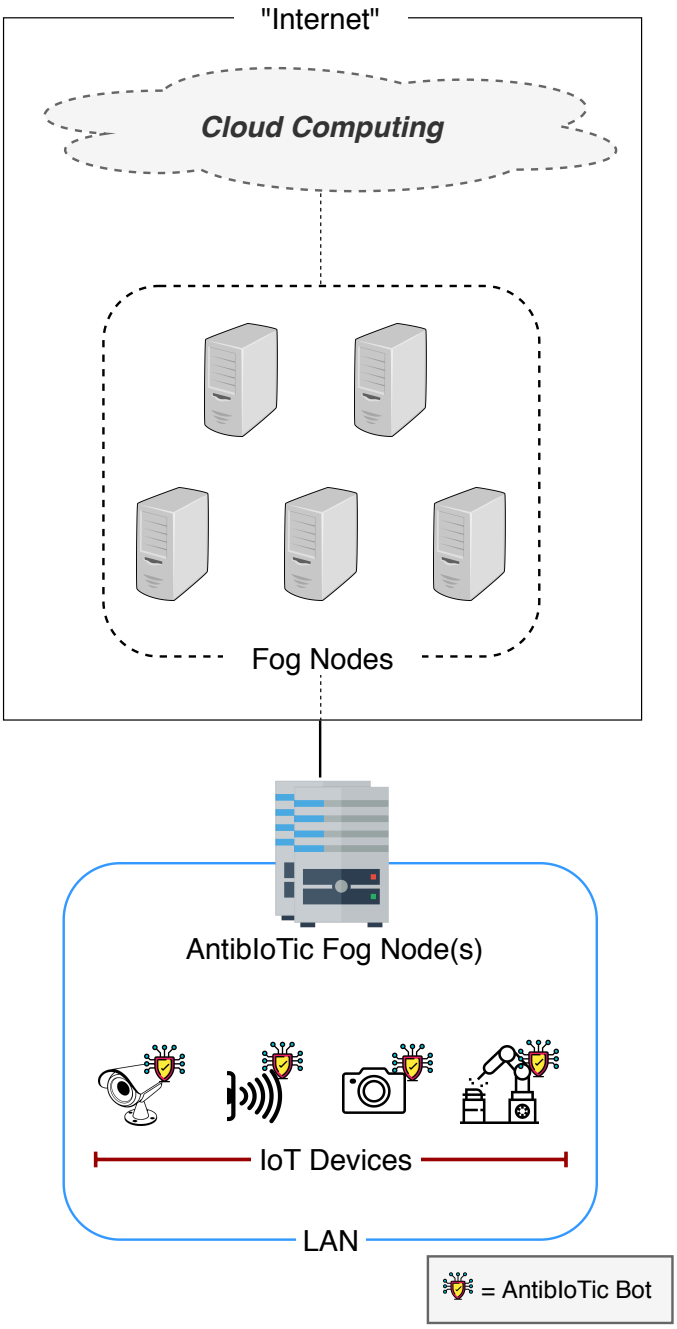

Fig. 3: AntibIoTic 2.0 framed in the N-tier architecture of Fog computing

AntibIoTic Fog node can also directly be connected to the Cloud without requiring any additional Fog layer.

The introduction of Fog computing in AntibIoTic provides a number of benefits. First of all, relying on Fog computing, AntibIoTic 2.0 solves the main problem of its predecessor: the legal issues. Indeed, IoT devices protected from AntibIoTic give prior consensus for it, allowing the Fog node to upload and run code on them. In addition, the

use of a Fog node, both as the gateway of the network and the main component of ANTIBIOTIC, significantly simplifies the system architecture. Finally, the introduction of Fog computing allows for simple improvements and extensions of the solution. Indeed, without tampering with each IoT device, it is possible to add features, improve existing ones, or scale up the solution to a large number of devices just by acting on the Fog node and its interaction with the Internet, without worrying about resources constraints. As an example, the introduction of Fog computing can be used to enable more refined analysis of live data through Machine Learning techniques, which is almost impossible to achieve through a traditional IoT botnet such as the one composing ANTIBIOTIC 1.0.

B. Main Features

ANTIBIOTIC 2.0 inherits the core features of ANTIBIOTIC 1.0 and enhances them with an additional set of features adapted to the new Fog-based design.

The core features of ANTIBIOTIC 2.0 can be summarized as follows.

- *Sanitize and secure IoT devices*: all IoT devices in the scope of ANTIBIOTIC 2.0 run the ANTIBIOTIC *Bot* that is in charge of sanitizing and securing them. First, the Bot *sanitizes* the device, i.e. cleans it up from malware and other possible threats. Then, the Bot *secures* the device, i.e. identifies security vulnerabilities of the device and takes action against them (e.g., close ports, change login credentials, update the firmware, etc.).
- *Persistent protection*: ANTIBIOTIC 2.0 persistently controls and protects IoT devices in its scope. If a device is rebooted or temporary disconnected, it will be automatically protected again as soon as it becomes available, without the need of performing any manual configuration on it.

In addition, thanks to the insertion of Fog computing in the design, ANTIBIOTIC 2.0 presents a number of features that are inspired from ANTIBIOTIC 1.0 but have been enhanced and extended. The additional features of ANTIBIOTIC 2.0 are described below.

- *Easy to install & transparent to use*: ANTIBIOTIC 2.0 is designed to work in different scenarios without the need of manually accessing and configuring each IoT device. It is sufficient to introduce a Fog node (or a federation of Fog nodes) in the desired network and perform the first configuration. Afterwards, the system starts working and securing the IoT network in a transparent way for the user. ANTIBIOTIC 2.0 best fits Industrial Internet of Things environments, where the variety of devices is limited and the number is high. Nevertheless, with a proper initial configuration, it can work properly in any IoT scenario.
- *Collect and process relevant data*: the code running on each IoT device (namely, the *Bot*) periodically reports back to the Fog node information about the device (e.g., technical specifications, discovered vulnerabilities, removed malware, etc.). Data are collected by the Fog node and can be used in a proactive manner with the help of the multi-layer Fog architecture and the Cloud (e.g., to generate statistics, to improve the system, to update the Bot, etc.).
- *Versatile and scalable*: due to the inclusion of Fog computing in its design, ANTIBIOTIC 2.0 is extremely versatile and scalable. It is possible to (horizontally or vertically) connect several Fog nodes to sensitively increase the number of IoT devices supported and the overall intelligence of the system. For instance, possible extensions include the use of Artificial Intelligence, Machine Learning, and Blockchain to upgrade the current solution.

Compared to ANTIBIOTIC 1.0, the core features have been inherited by ANTIBIOTIC 2.0, some features have been removed, others have been updated and adapted to the new Fog-based design. A quick comparison between ANTIBIOTIC 1.0 and ANTIBIOTIC 2.0 is shown in Table I.

The features listed above are only the high-level summary of the ANTIBIOTIC 2.0 functionalities, aimed at giving an idea of the system. Further extension and improvements are foreseen.

V. CONCLUSION AND FUTURE WORK

In this paper, we presented ANTIBIOTIC 2.0, to the best of our knowledge, the first Fog-based anti-malware for IoT systems. Specifically, we have first summarized ANTIBIOTIC 1.0 [1], predecessor of ANTIBIOTIC 2.0, along with its main practical shortcomings. Then, after introducing the Fog computing paradigm, we have described the rationale behind ANTIBIOTIC 2.0 and its main features. The result is a novel solution that, relying on Fog computing, enhances and improves ANTIBIOTIC 1.0, overcoming its main limitations and bringing the rationale of the ANTIBIOTIC approach to its full potential.

Future work will be focused on fully implementing ANTIBIOTIC 2.0, starting from a Proof-of-Concept (PoC), and on constantly improving the design of the system, adding features and improving existing ones. From a theoretical perspective, we will investigate new key concepts for next-generation IoT systems, such as sustainability and self-protection/healing [32].

REFERENCES

[1] M. De Donno, N. Dragoni, A. Giaretta, and M. Mazzara, "AntibIoTic: Protecting IoT Devices Against DDoS Attacks," in *Proceedings of the 5th International Conference in Software Engineering for Defence Applications*, P. Ciancarini, S. Litvinov, A. Messina, A. Sillitti, and G. Succi, Eds. Springer International Publishing, 2018, pp. 59–72.

[2] Cisco, "Cisco Visual Networking Index: Forecast and Trends, 2017-2022," Tech. Rep., November 2018. [Online]. Available: https://www.cisco.com/c/en/us/solutions/collateral/service-provider/visual-networking-index-vni/white-paper-c11-741490.pdf

[3] N. Dragoni, A. Giaretta, and M. Mazzara, "The Internet of Hackable Things," in *Proceedings of the 5th International Conference in Software Engineering for Defence Applications*, P. Ciancarini, S. Litvinov, A. Messina, A. Sillitti, and G. Succi, Eds. Springer, 2017, pp. 129–140.

TABLE I: Key differences between AntibIoTic 1.0 [1] and AntibIoTic 2.0

	AntibIoTic 1.0	**AntibIoTic 2.0**
Configuration	complex	simple
Data & Statistics	published online	collected & processed internally
Target devices	uncontrolled	only compliant IoT devices
Legal issues	Yes	No
Architecture	"white" botnet	Fog-based anti-malware
Short-term security	sanitize & secure	sanitize & secure
Long-term security	persistent protection	persistent protection
Versatility	low	high
Scalability	medium	high

[4] R. Goyal, N. Dragoni, and A. Spognardi, "Mind the tracker you wear: A security analysis of wearable health trackers," in *Proceedings of the 31st Annual ACM Symposium on Applied Computing*, ser. SAC '16. New York, NY, USA: ACM, 2016, pp. 131–136. [Online]. Available: http://doi.acm.org/10.1145/2851613.2851685

[5] M. Favaretto, T. Tran Anh, J. Kavaja, M. De Donno, and N. Dragoni, "When the price is your privacy: A security analysis of two cheap iot devices," in *Proceedings of 6th International Conference in Software Engineering for Defence Applications*, P. Ciancarini, M. Mazzara, A. Messina, A. Sillitti, and G. Succi, Eds. Springer International Publishing, 2020, pp. 55–75.

[6] M. De Donno, N. Dragoni, A. Giaretta, and A. Spognardi, "Analysis of DDoS-Capable IoT Malwares," in *Proceedings of the Federated Conference on Computer Science and Information Systems (FedCSIS)*. IEEE, 2017, pp. 807–816.

[7] K. York. (2016, October) Dyn Statement on 10/21/2016 DDoS Attack. Dyn Blog. Accessed on 2018-11-02. [Online]. Available: http://dyn.com/blog/dyn-statement-on-10212016-ddos-attack/

[8] M. De Donno, N. Dragoni, A. Giaretta, and A. Spognardi, "DDoS-Capable IoT Malwares: Comparative Analysis and Mirai Investigation," *Security and Communication Networks*, vol. 2018, 2018.

[9] Akamai, "Q2 2017 State of the Internet-Security Report," Tech. Rep. 2, 2017. [Online]. Available: https://www.akamai.com/us/en/multimedia/documents/state-of-the-internet/q2-2017-state-of-the-internet-security-report.pdf

[10] ——, "Summer 2018 State of the Internet-Security: Web Attack Report," Tech. Rep., 2018. [Online]. Available: https://content.akamai.com/us-en-PG11224-summer-2018-soti-web-attack-report.html

[11] F. Bonomi, R. Milito, J. Zhu, and S. Addepalli, "Fog Computing and Its Role in the Internet of Things," in *Proceedings of the First Edition of the MCC Workshop on Mobile Cloud Computing*, ser. MCC '12. New York, NY, USA: ACM, 2012, pp. 13–16.

[12] S. Alharbi, P. Rodriguez, R. Maharaja, P. Iyer, N. Bose, and Z. Ye, "FOCUS: A Fog Computing-based Security System for the Internet of Things," in *Proceedings of the 15th Consumer Communications & Networking Conference (CCNC)*. IEEE, 2018, pp. 1–5.

[13] B. Paharia and K. Bhushan, "Fog Computing as a Defensive Approach Against Distributed Denial of Service (DDoS): A Proposed Architecture," in *Proceedings of the 9th International Conference on Computing, Communication and Networking Technologies (ICCCNT)*. IEEE, 2018, pp. 1–7.

[14] E. Lear, R. Droms, and D. Romascanu, "Manufacturer Usage Description Specification," *IETF draft*, June 2018. [Online]. Available: https://tools.ietf.org/pdf/draft-ietf-opsawg-mud-25.pdf

[15] W. Razouk, D. Sgandurra, and K. Sakurai, "A New Security Middleware Architecture Based on Fog Computing and Cloud to Support IoT Constrained Devices," in *Proceedings of the 1st International Conference on Internet of Things and Machine Learning*. ACM, 2017, pp. 35:1–35:8.

[16] M. Serror, M. Henze, S. Hack, M. Schuba, and K. Wehrle, "Towards In-Network Security for Smart Homes," in *Proceedings of the 13th International Conference on Availability, Reliability and Security*, ser. ARES 2018. ACM, 2018, pp. 18:1–18:8.

[17] T. D. Nguyen, S. Marchal, M. Miettinen, M. H. Dang, N. Asokan, and A.-R. Sadeghi, "DIoT: A Crowdsourced Self-learning Approach for Detecting Compromised IoT Devices," *arXiv:1804.07474*, 2018. [Online]. Available: https://arxiv.org/pdf/1804.07474.pdf

[18] H. Sun, X. Wang, R. Buyya, and J. Su, "CloudEyes: Cloud-based Malware Detection with Reversible Sketch for Resource-constrained Internet of Things (IoT) Devices," *Software: Practice and Experience*, vol. 47, no. 3, pp. 421–441, 2017. [Online]. Available: https://onlinelibrary.wiley.com/doi/epdf/10.1002/spe.2420

[19] EU Parliament, Council, "Directive of the European Parliament and of the Council of 12 August 2013 on Attacks Against Information Systems and Replacing Council Framework Decision 2005/222/JHA," 2013, accessed on 2018-11-02. [Online]. Available: https://eur-lex.europa.eu/legal-content/EN/TXT/PDF/?uri=CELEX:32013L0040&from=EN

[20] I. Stojmenovic and S. Wen, "The Fog Computing Paradigm: Scenarios and Security Issues," in *Proceedings of the Federated Conference on Computer Science and Information Systems (FedCSIS)*. IEEE, 2014, pp. 1–8.

[21] 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*. ACM, 2015, pp. 37–42.

[22] S. Yi, Z. Hao, Z. Qin, and Q. Li, "Fog Computing: Platform and Applications," in *Proceedings of the 3rd IEEE Workshop on Hot Topics in Web Systems and Technologies (HotWeb)*, Nov 2015, pp. 73–78.

[23] M. Chiang and T. Zhang, "Fog and IoT: An Overview of Research Opportunities," *IEEE Internet of Things Journal*, vol. 3, no. 6, pp. 854–864, Dec 2016.

[24] L. M. Vaquero and L. Rodero-Merino, "Finding Your Way in the Fog: Towards a Comprehensive Definition of Fog Computing," *SIGCOMM Computer Communication Review*, vol. 44, no. 5, pp. 27–32, October 2014.

[25] S. Chen, T. Zhang, and W. Shi, "Fog Computing," *IEEE Internet Computing*, vol. 21, no. 2, pp. 4–6, 2017.

[26] M. Chiang, B. Balasubramanian, and F. Bonomi, *Fog for 5G and IoT*. John Wiley & Sons, 2017.

[27] M. Iorga, L. Feldman, R. Barton, M. J. Martin, N. Goren, and C. Mahmoudi, "Fog Computing Conceptual Model - Recc," Tech. Rep., 2018.

[28] OpenFog Consortium, "Glossary of Terms Related to Fog Computing," 2018, [Accessed on July 10th, 2018]. [Online]. Available: https://goo.gl/cS7un3

[29] OpenFog Consortium Architecture Working Group and others, "OpenFog Reference Architecture for Fog Computing," *OPFRA001*, vol. 20817, February 2017.

[30] J. Ni, K. Zhang, X. Lin, and X. Shen, "Securing Fog Computing for Internet of Things Applications: Challenges and Solutions," *IEEE Communications Surveys & Tutorials*, 2017.

[31] P. Hu, S. Dhelim, H. Ning, and T. Qiu, "Survey on Fog Computing: Architecture, Key Technologies, Applications and Open Issues," *Journal of Network and Computer Applications*, 2017.

[32] N. Dragoni, F. Massacci, and A. Saidane, "A Self-Protecting and Self-Healing Framework for Negotiating Services and Trust in Autonomic Communication Systems," *Computer Networks*, vol. 53, no. 10, pp. 1628 – 1648, 2009, autonomic and Self-Organising Systems. [Online]. Available: http://www.sciencedirect.com/science/article/pii/S1389128608002855