Chapter 5

IoT Network Risk Assessment and Mitigation: The SerIoT Approach

By Gianmarco Baldini, Piotr Fröhlich, Erol Gelenbe, Jose Luis Hernandez-Ramos, Mateusz Nowak, Slawek Nowak, Stavros Papadopoulos, Anastasis Drosou and Dimitrios Tzovaras

DOI: 10.1561/9781680836837.ch5

Published in *Security Risk Management for the Internet of Things: Technologies and Techniques for IoT Security, Privacy and Data Protection* by John Soldatos (ed.). 2020. ISBN 978-1-68083-682-0. E-ISBN 978-1-68083-683-7.

Suggested citation: Gianmarco Baldini *et al.*. 2020. "IoT Network Risk Assessment and Mitigation: The SerIoT Approach" in *Security Risk Management for the Internet of Things: Technologies and Techniques for IoT Security, Privacy and Data Protection*. Edited by John Soldatos. pp. 88–104. Now Publishers. DOI: 10.1561/9781680836837.ch5.

now
the essence of knowledge

Cyberattacks on the Internet of Things (IoT) can be the source of major economic damage. They can disrupt production lines, manufacturing processes, and supply chains. They can adversely impact the physical safety of vehicles and transportation systems, and damage the health of living beings both through supply chains for food, medicines, and other vital items, as well as through direct attacks on sensors and actuators that may be connected to vital functions. Thus, securing the IoT is of primary importance to our societies. This paper describes the technical approach that we adopt for IoT security in the SerIoT Research and Innovation Project that is funded by the European Commission. We first discuss the risk scenario for the IoT and briefly review approaches that have been developed to mitigate such risks. Then, we discuss a policy-based lightweight approach that mitigates risks at the level of the attachment of IoT devices to a network. We follow this with a detailed proposal based on using a distributed Machine Learning approach to risk and attack detection in real time, as well as suggestions for future work.

5.1 Introduction

The IoT may extend to billions of objects and sensors connected to Clouds, databases, decision systems, and actuators [3]. It has the potential to improve the critical processes that are at the heart of our socio-economic systems [9, 31] composing a data-driven society [42]. However, the pervasive nature of the IoT raises risks that go way beyond the individual technologies such as the internet or wireless networks [16] and machine-to-machine systems [44]. In addition to risks related to system malfunctions, Quality of Service (QoS) failures, and excessive energy consumption, they also include the theft and tampering of data, conventional network attacks, and other attacks that attempt to deplete the energy of autonomous sensors and actuators [6, 17, 20, 22, 25, 37].

In Information and Communication Technology (ICT), risk management embraces different processes intended to deal with the identification, assessment, and mitigation of risks, which are derived from vulnerable ICT systems or potential cybersecurity attacks [38]. However, the scale and heterogeneity of IoT systems sets out significant challenges for the implementation of a risk management approach. On the one hand, IoT represents the current trend to hyperconnected systems; therefore, risk management aspects must consider the relationships and dependencies among different devices that could affect to the security level of a certain system or deployment. On the other hand, IoT deployments are usually composed by components with resource constraints, which are installed in uncontrolled environments with default security configurations. These constraints also make IoT devices and systems an attractive target for possible attacks. Furthermore, due to the potential huge number of devices in IoT deployments, there is a real need to consider risk management approaches with a high degree of automation to efficiently identify and mitigate new risks affecting such systems. These approaches should be able to represent the relationships among devices, vulnerabilities, and attacks of the overall deployment.

To create a suitable risk management methodology for the IoT, holistic approaches are required to understand the probability and impact of potential risks affecting devices and systems. As a core process of risk management, risk assessment has been strongly considered in recent years by IoT researchers [36]. However, assessing risk is in itself insufficient, and we must design IoT networks that can detect and then mitigate security risks, but also mitigate other risks that may arise regarding the overall performance of the system by preserving QoS and offering energy efficient operation of the system [47]. Thus, a continuing overall evaluation of the risks introduced by IoT systems and their networks is needed, and this paper offers a view of risk identification and mitigation as applied to IoT systems, based on our work in the SerIoT Project supported by the

European Commission [15]. In particular, we propose a system called *Autopolicy*, which is intended to enforce security profiles according to the intended communications of a certain IoT system with other devices or systems. This approach reduces the attack surface of the system and, consequently, the probability of cybersecurity risks. Furthermore, we complement this mechanism with a detection approach, which uses a distributed anomaly detection scheme based on deep learning (DL) and graph networks [2, 53]. Localized anomaly detection methods at IoT ports and network routers can also be investigated [4]. The mitigation method we proposed in this paper exploits network Self-Awareness [18, 29] centered on Software Defined Networks [11] that can achieve secure and QoS-based routing of significant traffic flows using machine learning and adaptivity [8, 12, 45, 50].

The structure of the chapter is as follows: Section 5.2 analyzes the main processes composing a risk management and the challenges associated to the IoT paradigm. Then, the description of the Autopolicy approach is described in Section 5.3. Furthermore, Section 5.4 provides a detailed description of our approach for the detection of attacks and anomalies in IoT-based DL and graph networks.

5.2 Risk Management in IoT

The process of risk management is composed of different elements [38] including risk assessment and the definition of risk mitigation solutions. In particular, according to the definition provided by the National Institute of Standards and Technology (NIST), the risk management includes *"(i) the conduct of a risk assessment; (ii) the implementation of a risk mitigation strategy; and (iii) employment of techniques and procedures for the continuous monitoring of the security state of the information system"* [21]. Therefore, risk management implies the assessment, mitigation, and monitoring of risks.

Risk assessment is defined in CNSSI-4009 [51] as the process of *"identifying, prioritizing, and estimating risks"*. This includes determining the extent to which adverse circumstances or events could affect an enterprise. However, as described by [36, 41], risk assessment in IoT could be more difficult to implement in comparison to traditional ICT systems because of the pervasive connectivity of the IoT devices and the different potential interfaces (e.g., different wireless standards or middlewares), which can increase the attack surface. The assessment of the impact can also be critical in specific categories of IoT devices like the cyberphysical systems (e.g., an automated vehicle) or healthcare (e.g., insulin distributor) as a cybersecurity threat can have quite damaging consequences. Another challenge of the IoT domain for risk assessment is that IoT systems (e.g., a smart home or a smart

vehicle) can be composed of many different devices that, in turn, could be comprised of several components with a different security level. This means that the risk assessment of a certain system will depend on the security level associated to each part of the system. For this aspect, the main challenge is related to the way in which each risk value could be aggregated to provide a reliable value for the whole system.

A number of risk assessment methodologies have been developed in the literature and applied in some cases to the IoT domain. For example, the Common Weakness Scoring System (CWSS) [26] is a methodology to prioritize software weaknesses. The main motivation is to provide means to different stakeholders (e.g., software testers, or manufacturers) for quantifying the risks associated to a specific weakness. This way, the corresponding stakeholder can prioritize the weakness to be solved based on the estimated risk. Various metrics (i.e., Base Finding, Attack Surface and Environmental metric groups, which in turn contain different metric to quantify the CWSS score associated to a certain weakness) are used to produce a final CWSS score between 0 and 100. The main challenge is obviously the quantification process as cybersecurity risk is more difficult to quantify especially in the IoT domain [36]. A similar approach is also used in the Common Vulnerability Scoring System collection (CVSS) [32], which is based on three group metrics: Base, Temporal, and Environmental. Unlike CWSS, CVSS is used in discovered vulnerabilities (i.e., the known known). The CVSS is widely used today as it is used in the National Vulnerability Database (NVD) created by the NIST. In the IoT literature, CVSS and CWSS schemes are used in combination to assess the risks in Bluetooth technology, where the authors extended the authentication metric to include new security factors inherent to the Bluetooth technology [40]. CVSS and CWSS have also been used as an input in the process of cybersecurity certification of IoT devices in [27, 28]. Finally, another quantitative risk assessment scheme is DREAD, which is used to compute a risk value associated to a certain threat or vulnerability based on the use of five categories: Damage potential, Reproducibility, Exploitability, Affected users, and Discoverability. It was used at Microsoft, and currently, it is used by OpenStack. In [5], DREAD is applied to the mobile healthcare domain due to its simplicity.

As already mentioned, *risk identification* is a core process of a risk assessment approach. In this direction, threat and vulnerability assessments are usually employed to identify risks to organizational operations, and the evaluation of those risks in terms of likelihood of occurrence and impact if they occur. In turn, for the identification of threats and vulnerabilities, Intrusion Detection Systems (IDS) have been widely used [7] by monitoring and analyzing the network and/or system events. However, previous considerations make the application of well-known IDSs more challenging into IoT systems. Indeed, one of the popular approaches in literature is to analyze the data produced by IoT device to identify potential

anomalies. A potential weakness of this approach is the difficulty to anticipate the attack as the anomaly is created when the attack is already ongoing or it is completed. Moreover, machine learning capabilities have been increasingly become more powerful in recent times, and a complex security attack may be composed by a sequence of attacks steps. Then, the research objective is to the detect the initials steps in the attack.

These aspects are already highlighted by [52], which provides a comprehensive taxonomy of IDSs for IoT. Authors classify IDSs according to different parameters, such as the *placement strategy* (distributed, centralized, and hybrid) and *detection method* (signature-based, anomaly based, specification-based, and hybrid). In particular, anomaly based detection techniques are used to detect new attacks by comparing the normal behavior of a certain system with its actual activity. This approach could leverage the inherent nature of IoT systems, which are usually composed of special purpose devices. This aspect is considered by the recent Manufacturer Usage Description (MUD) standard [24], which is aimed to restrict the communication to/from a certain IoT device. To generate such intended behavior, anomaly detection techniques have widely used machine learning techniques, and in [49], a distributed anomaly detection approach in which each node monitors its neighbors is proposed, where monitoring nodes inform other nodes about security problems. In [30], deep autoencoders to detect anomalous network traffic of IoT devices are discussed and tested for different commercial IoT devices in the presence of IoT botnets such as Mirai. In our case, we describe a risk monitoring approach based on a multi-agent system and DL techniques for automatic feature extraction and anomaly detection that is described in Section 5.4.

As the next step of a risk management approach, *risk mitigation* focuses on the goal to mitigate the risks through different means: (a) avoiding the risk (e.g., not using a specific interface), (b) reducing the risk by implementing specific countermeasures (e.g., intrusion detection), or (c) transfer the risk to some other entity (e.g., an insurance company). In the IoT domain, the mitigation of risk is more difficult to achieve because of the limited computing capabilities of IoT devices, which makes more difficult the implementation of sophisticated countermeasures. The transfer of the risks is also difficult as IoT devices are often standalone systems (e.g., a sensor). A potential approach to mitigate IoT security risks is the integration of Software-defined Networks (SDNs), in order to restrict communications from/to IoT devices. Indeed, the application of SDN techniques into IoT scenarios has attracted a significant interest from academia in recent years. For example, [13] describes an architecture for managing the obtaining and enforcement of MUD restrictions, which are enforced by SDN switches. In this direction, our approach is based on an architecture to identify and mitigate security risks based on the communication profiles of IoT devices that is described in the next section.

5.3 Autopolicy System

The deployment of IoT scenarios requires new approaches to manage cybersecurity risks throughout the life cycle of devices composing such scenarios. In recent years, diverse proposals have been developed to address risk management aspects, in order to realize the best security practices for IoT (such as those defined by the (ENISA) [10]). In this direction, [33] proposed a system for automatically identifying the type of IoT devices that are connected to a network and enforcing security rules to restrict the communication of potentially vulnerable devices. Indeed, authors proposed the use of SDN techniques for the enforcement of such rules. Furthermore, [1] designed an automated approach to derive network security policies, as well as a multi-layered policy enforcement architecture.

Based on the considerations from previous works, our approach (Autopolicy) addresses several aspects of risk management in IoT scenarios. On the one hand, it reduces the attack surface and, consequently, the potential security risks associated to IoT devices by enforcing traffic profiles to be defined either by the device's manufacturer, or automatically. On the other hand, it links the obtaining of such profile to the authentication process of the IoT device, so that only traffic profiles from legitimate devices are obtained and enforced. Furthermore, it integrates a decentralized mechanism based on a Multi-agent System (MAS) for risk monitoring that is described in the next section. This way, Autopolicy helps to identify and monitor risks in a certain IoT network through traffic profiles, in order to mitigate the impact and likelihood associated to well-known threats, such as DoS attacks.

Autopolicy follows a similar approach to the recent IETF standard Manufacturer Usage Description (MUD) [24], which defines an architecture and data format to obtain and define network profiles for IoT devices. Indeed, MUD has received an increasing interest from other standardization organizations, such as the NIST, which recently published a Cybersecurity Practice Guideline [35] describing the advantages provided by MUD to reduce the potential harm of compromised devices. It follows a simple data model to generate network traffic rule allowing/denying the communication to/from a certain IoT device. In our case, we consider additional aspects, such as the maximum number of connections or restrictions on the message size, in order to properly react against DoS attacks.

An overview of the Autopolicy design is presented in Figure 5.1. The basic flow of information starts when a new *IoT device* joins a network, and it is detected by a switch (or controller) acting as *Device Authenticator*, which is responsible for granting/denying the access to such device. This initial authentication process is usually called *bootstrapping* [14], and it is used to exchange the corresponding cryptographic material between the IoT device and controller for authentication purposes.

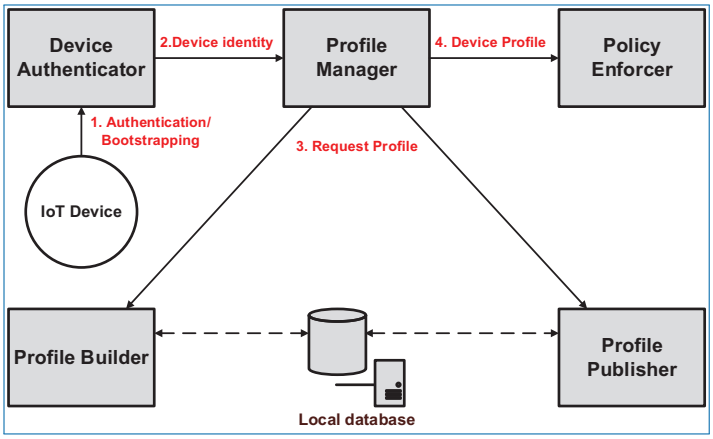

Figure 5.1. Functional diagram of Autopolicy and of its information flows.

For this purpose, there is a plethora of mechanisms that could be considered to instantiate this process, for example, based on the use of the Extensible Authentication Protocol (EAP) [48], which is widely considered in the scope of 5G networks. An alternative approach could be based on the OMA LwM2M specification [43], which explicitly defines a *Bootstrap Interface*, so that a LwM2M Client can be registered into the corresponding LwM2M Server. OMA defines four bootstrapping modes (factory, smartcard, client-initiated and server-initiated), and the use of transport layer security (i.e., TLS/DTLS) and application layer security based on the recent Object Security for Constrained RESTful Environments (OSCORE) [46].

After the bootstrapping process, the entity acting as Device Authenticator sends the authenticated device identity to the *Profile Manager*, which uses the identity to request the associated traffic profile. For example, following the MUD approach, the identity could be represented by a X.509 certificate in which a URL is included to get the traffic profile. In our case, the Profile Manager contacts the *Profile Publisher*, which manages the traffic profiles of IoT devices. Indeed, this entity follows a similar role to the MUD File Server [24], which is an entity provided by the manufacturer of the device. An alternative approach for sharing traffic profiles is represented by the use of blockchain as a trusted, distributed and transparent repository of traffic profiles. This way, manufacturers are enabled to implement smart contracts to manage the creation and update of traffic profiles. It should be noted that the use of blockchain is also considered in [34] to store cybersecurity information of IoT devices, including MUD profiles. In case there is not an associated profile to the device, the Profile Manager contacts the *Profile Builder* entity to create such profile based on IP traffic statistics and, optionally, additional information from other repositories where that device is already deployed.

Listing 5.1 Traffic profile example.

```
{       ''from_device'': {
                ''rate'': 0.1,
                ''allowed_dst'': [
                        ''91.200.1.0/24 tcp/80,443'',
                ],
                ''connections'':''10''
        }
        ''to_device'': {
                ''rate'': 0.1,
                ''blocked_dst'': [
                        ''0/0'',
                ],
}}
```

The obtained traffic profile is then sent to the *Policy Enforcer*, which only allows IP traffic under strict rules defined by the profile. The role of the Policy Enforcer could be embedded in the controller (or switch) acting as Device Authenticator. Listing 5.1 shows a simple example of traffic profile to define restrictions on the communications to/from the device. In particular, *rate* specifies the maximum bandwidth (in Mbps); *allowed_dst* and *blocked_dst* indicate the IP address, protocol, and port of allowed/denied communications. Furthermore, *connections* represents the maximum number of connections allowed with a certain device.

While this represents a simple example of traffic profile, we plan to extend this initial data model and align it with the MUD standard. The described mechanism was designed for the needs of the SerIoT project as one of the components to prevent and mitigate potential risks in IoT systems. Its role is to secure the network from the malicious endpoints and ensure that IoT devices behave according to specific traffic rules. It should be noted that the described mechanism is intended to reduce the attack surface by enforcing the rules associated to the intended behavior of IoT devices. However, it still requires a dynamic approach to continuously monitor the risks that can be identified through traffic analysis. For this purpose, next section provides an overview of the risk monitoring approach that is being implemented in the scope of the SerIoT project.

5.4 Towards Distributed Attack Detection

The recent proliferation of IoT technologies has given rise to a dramatic increase in the number of edge devices. More devices not only introduce more security vulnerabilities but also greatly increase the volume of transferred data that must be analyzed in order to detect and mitigate network anomalies. When undetected,

such anomalies can have a great impact on the robustness and Quality of Service (QoS) of the attacked IoT infrastructure. The main challenge for anomaly detection methods in today's IoT network is the analysis of the big amounts of data that arise from the growing number of IoT devices. Such large volumes of information, as well as the distributed, large-scale, and heterogeneous nature of the IoT network, render the goal of (near) real-time anomaly detection difficult to accomplish using traditional centralized monitoring approaches [23].

In this respect, this work proposes the use of Multi-agent Systems (MAS) for processing the large amounts of data exchanged by the IoT devices in a distributed manner. The MAS allows for monitoring of the network traffic using parallel computation, resulting in faster detection times, and thus, improving the security and QoS of the IoT network. Additionally, when the agents have redundant roles (i.e., detection of anomalies regarding the same IoT device by different agents), the MAS system offers robustness, since it can tolerate failures of one or multiple agents [39]. Finally, MAS also offers scalability, since due to the modular nature of the approach, adding new agents to the network is easy and straightforward.

The main challenge when implementing MAS is how the agents will exchange information in order to solve a problem in a cooperative manner [19]. In this respect, this work utilizes recent advances in AI and Deep Learning (DL) in order to enable automatic feature extraction and anomaly detection by the agents of the MAS. Specifically, due to the graph structure of the network communication events, where nodes represent devices and edges communication, it is natural to consider DL approaches that deal with graph structured datasets, i.e., Graph Neural Networks (GNN) [53]. This work utilizes as an inspiration the more generic GNN formulation, proposed by DeepMind [2].

The IoT network is defined as a graph $G(V, E, f_v, f_e)$, where each node $v_i \in V$ represents either an IoT device or a router, and each directed edge $e_j \in E$ represents a directed communication between nodes. Each node and edge is augmented with a feature vector through functions $f_v : V \rightarrow \mathbb{R}^{N_v}$ and $f_e : E \rightarrow \mathbb{R}^{N_e}$, respectively, where N_v and N_e, the size of node and edge feature vectors, respectively. These features are calculated using network traffic over a specific time window ΔT. The list of utilized features is presented in Table 5.1.

The agents of the MAS are installed in a subset of the nodes $V' \subseteq V$ and exchange information using the same set of communication edges E. Each agent comprised of two networks utilized for updating the feature representation of the adjacent edges and the node they are installed on. These updated features are afterwards utilized for multi-class anomaly detection through another pair of deep learning networks. An overview of this formulation for updating feature representation through information exchange between the different agents is illustrated in Figure 5.3. Under this consideration, the role of the edge Deep Neural Network (DNN)

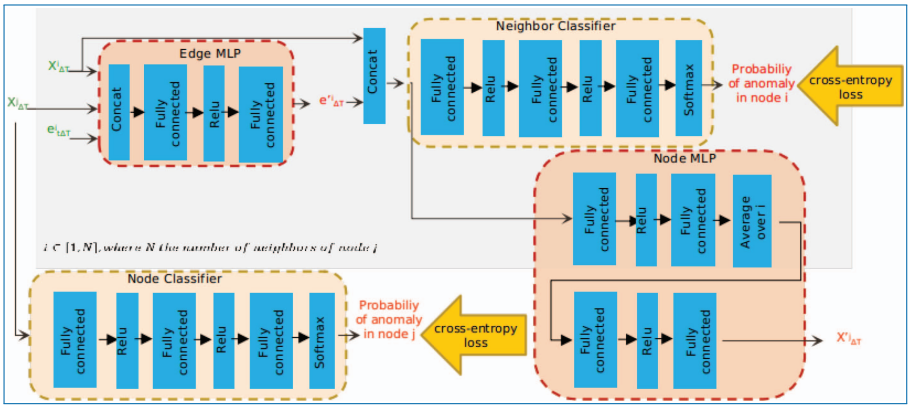

Figure 5.2. The architecture of each DNN agent in the multi-agent anomaly detection system. The edge Multi-layer Perception (MLP) takes information from the N neighboring agents and updated the values of the edge features, while the node MLP combines the updated edge feature values and updates the feature values of the specific node (agent). Edge features are used for predicting anomalies in the neighborhood, while node features are used for detecting anomalies in the specific node, using the classifier networks. The training is performed in a supervised manner using back-propagation with cross-entropy loss.

Table 5.1. The list of network features used for anomaly detection. The number of features for each node is $N_v = 5$ and for each edge is $N_e = 3$.

#	Target	Feature description
1	edge/node	Average number of packets sent
2	node	Average number of packets received
3	edge/node	Average number of bytes sent
4	node	Average number of bytes received
5	edge/node	Average connection duration

is to integrate information from adjacent nodes, as well as the corresponding edge, and update this edge's feature representation. These updated feature representations for all adjacent edges are afterwards taken as input by the node DNN that updates the feature representation of the corresponding node. The Node and Edge DNNs are the same for all agents (i.e., have shared weights), and thus, each agent is able to learn from events happening to other agents.

Based on the previous definitions, there are in total four DNNs, two used for updating node and edge feature representations, namely MLP_n and MLP_e, and two used for classification of neighbor nodes and the node in which the agent is installed, namely $Classifier_{neighbor}$ and $Classifier_{node}$.

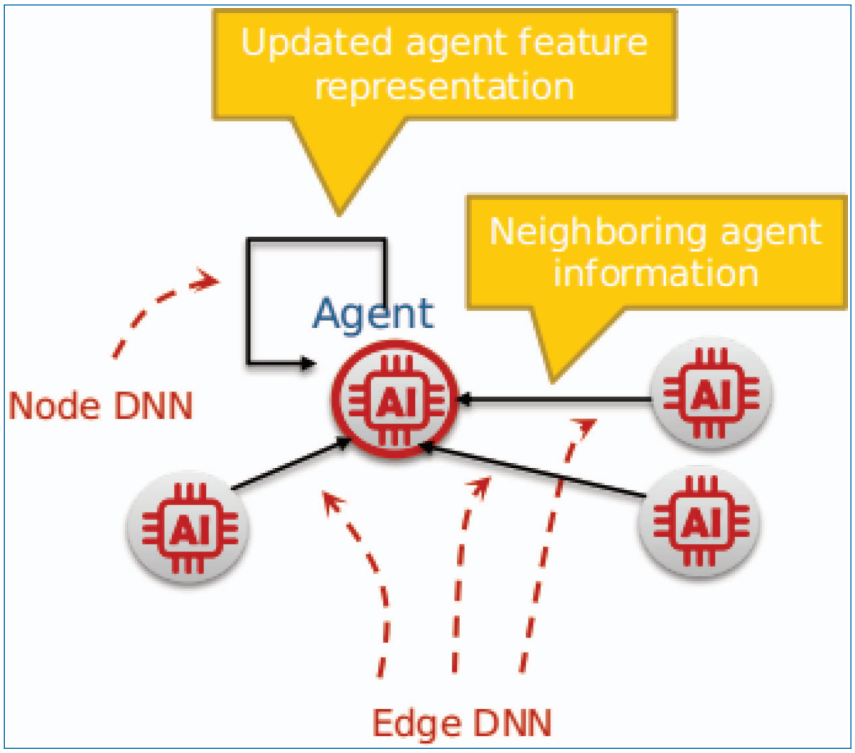

Figure 5.3. The procedure for updating the feature representations of the edges and nodes visible by each agent. The edge Deep Neural Network (DNN) takes as input the previous edge feature values and the adjacent agent features, and updates the feature representation of each edge. The node DNN aggregates the information from the updated edge feature values, and updates the feature representation of the specific node.

Given an agent installed in node v_j, the procedure followed for updating the features of this node and the adjacent edges, as well as detecting anomalies in the entire neighborhood is formulated as follows:

$$\begin{aligned} e'^{i}_{\Delta T} &= MLP_e(x^{j}_{\Delta T}, x^{i}_{\Delta T}, e^{i}_{\Delta T}) \\ x'^{j}_{\Delta T} &= MLP_n\left(\frac{1}{N}\sum_{i=1}^{N} concat(e'^{i}_{\Delta T}, x^{i}_{\Delta T})\right) \\ p_i &= Classifier_{neighbor}(e'^{i}_{\Delta T}, x^{i}_{\Delta T}) \\ p_j &= Classifier_{node}(e'^{i}_{\Delta T}, x^{i}_{\Delta T}) \end{aligned} \tag{5.1}$$

where x^j is the feature vector of node v_j where the agent is installed, x^i is the feature of the neighboring nodes in the graph, e^i is the feature vector of the adjacent

edges, and N is the number of neighboring nodes. This computation is performed with traffic data from a specific window ΔT. The MLP_n network is applied on the average of the updated features of each adjacent edge and node. The $concant(.)$ function takes as input two vectors and returns a larger vector which represents the concatenation of the two vectors. Equation (5.1) represents a single iteration of information exchange between agents. Stacking multiple blocks of such operations, the agents can exchange information multiple times within each time period ΔT. The architecture of this procedure is illustrated in Figure 5.2. The entire network is trained in a supervised manner using the back-propagation algorithm with cross-entropy as loss for multi-class classification.

In this formulation of MAS, each agent is responsible for performing anomaly detection not only on itself but also on its neighboring nodes. This redundancy of the role of the agents enables the system to be robust in cases where one or multiple agents may fail, since other agents will take over the role of detecting anomalies in neighboring nodes instead of them. The issue that arises, however, is how to combine the overlapping decisions of the different agents into a single decision for each node in the IoT network. This work utilizes a simple aggregation method, where a node is considered anomalous if at least one agent reported it as anomalous. Alternative more sophisticated aggregation schemes will be considered in future work.

5.5 Conclusions

The implementation of a suitable risk management approach demands for holistic approaches for the assessment, mitigation, and monitoring of security risks in IoT systems. Toward this end, we have provided an initial description of a policy-based system that is being defined in the scope of the EU SerIoT project. Our approach is based on the use of network traffic profiles, which specify the intended communications of IoT devices. Such effort is aligned with the recent MUD standard, which has received an increasing interest from academia and industry during last year. Autopolicy defines an architecture for obtaining and enforcing traffic profiles, in order to mitigate potential security risks in IoT systems. Furthermore, it integrates a distributed machine learning approach for risk monitoring by analyzing the network traffic. As a future work, we plan to adopt the MUD standard as the baseline to define and extend our traffic profiles to be enforced through the components defined in the SerIoT SDN architecture. Furthermore, we will provide an implementation and detailed description of our adaptive machine learning approach to reroute IoT traffic according to the identification of compromised nodes of a certain network.

Acknowledgments

This research is supported by the European Commission H2020-IOT-2016-2017 (H2020-IOT-2017) Program under Grant Agreement 780139 for the SerIoT Research and Innovation Action.

References

[1] Barrera, D., Molloy, I., and Huang, H.: Standardizing IoT network security policy enforcement. In: Workshop on Decentralized IoT Security and Standards 2018 (01 2018). https://doi.org/10.14722/diss.2018.23007, http://wp.internetsociety.org/ndss/wp-content/uploads/sites/25/2018/07/diss2018_7_Barrera_paper.pdf

[2] Battaglia, P., Hamrick, J.B.C., Bapst, V., Sanchez, A., Zambaldi, V., Malinowski, M., Tacchetti, A., Raposo, D., Santoro, A., Faulkner, R., Gulcehre, C., Song, F., Ballard, A., Gilmer, J., Dahl, G.E., Vaswani, A., Allen, K., Nash, C., Langston, V.J., Dyer, C., Heess, N., Wierstra, D., Kohli, P., Botvinick, M., Vinyals, O., Li, Y., and Pascanu, R.: Relational inductive biases, deep learning, and graph networks. arXiv (2018), https://arxiv.org/pdf/1806.01261.pdf

[3] Bera, S., Misra, S., and Vasilakos, A.V.: Software-defined networking for internet of things: A survey. IEEE Internet of Things Journal **4**(6), 1994–2008 (2017)

[4] Brun, O., Yin, Y., and Gelenbe, E.: Deep learning with dense random neural network for detecting attacks against IoT-connected home environments. Procedia Computer Science **134**, 458–463 (2018)

[5] Cagnazzo, M., Hertlein, M., Holz, T., and Pohlmann, N.: Threat modeling for mobile health systems. In: 2018 IEEE Wireless Communications and Networking Conference Workshops (WCNCW). pp. 314–319. IEEE, Barcelona (Apr 2018). https://doi.org/10.1109/WCNCW.2018.8369033, https://ieeexplore.ieee.org/document/8369033/

[6] Collen, A. *et al.*: Ghost: Safeguarding home IoT environments with personalised real-time risk control. In: Recent Cybersecurity Research in Europe: Proceedings of the 2018 ISCIS Security Workshop, Imperial College London. vol. LNCCIS 821. Springer Verlag (2018)

[7] Debar, H., Dacier, M., and Wespi, A.: Towards a taxonomy of intrusion-detection systems. Computer Networks **31**(8), 805–822 (1999)

[8] Dobson, S. *et al.*: A survey of autonomic communications. ACM Transactions on Autonomous and Adaptive Systems (TAAS) **1**(2), 223–259 (2006)

[9] Elhammouti, H., Sabir, E., Benjillali, M., Echabbi, L., and Tembine, H.: Self-organized connected objects: Rethinking QoS provisioning for IoT services. IEEE Communications Magazine **55**(9), 41–47 (2017)
[10] ENISA: Good Practices for Security of Internet of Things in the context of Smart Manufacturing (2018), https://www.enisa.europa.eu/publications/good-practices-for-security-of-IoT
[11] Francois, F. and Gelenbe, E.: Towards a cognitive routing engine for software defined networks. In: Communications (ICC), 2016 IEEE International Conference on. pp. 1–6. IEEE (2016)
[12] Galis, A., Denazis, S., Brou, C., and Klein, C.: Programmable Networks for IP Service Deployment. Artech House Inc. (2004)
[13] García, S.N.M., Molina Zarca, A., Hernández-Ramos, J.L., Bernabé, J.B., and Gómez, A.S.: Enforcing behavioral profiles through software-defined networks in the industrial internet of things. Applied Sciences **9**(21), 4576 (2019)
[14] Garcia-Morchon, O., Kumar, S.S., and Sethi, M.: Internet of Things Security: State of the Art and Challenges (RFC 8576) (2019)
[15] Gelenbe, E., Domanska, J., Czachórski, T., Drosou, A., and Tzovaras, D.: Security for internet of things: The SerIoT project. In: 2018 International Symposium on Networks, Computers and Communications, ISNCC 2018, Rome, Italy, June 19-21, 2018. pp. 1–5. IEEEXplore (2018), https://doi.org/10.1109/ISNCC.2018.8531004
[16] Gelenbe, E., Gorbil, G., Tzovaras, D., Liebergeld, S., Garcia, D., Baltatu, M., and Lyberopoulos, G.: Security for smart mobile networks: The nemesys approach. In: Privacy and Security in Mobile Systems (PRISMS), 2013 International Conference on. pp. 1–8. IEEE (2013)
[17] Gelenbe, E. and Kadioglu, Y.M.: Energy life-time of wireless nodes with and without energy harvesting under network attacks. In: Advances in Cyber-Security: An ISCIS International Workshop. Springer (2018)
[18] Gelenbe, E., Liu, P., and Lainé, J.: Genetic algorithms for route discovery. IEEE Transactions on Systems, Man, and Cybernetics, Part B (Cybernetics) **36**(6), 1247–1254 (2006)
[19] Gupta, J.K., Egorov, M., and Kochenderfer, M.: Cooperative multi-agent control using deep reinforcement learning. In: International Conference on Autonomous Agents and Multiagent Systems. pp. 66–83. Springer (2017)
[20] He, D., Chan, S., Qiao, Y., and Guizani, N.: Imminent communication security for smart communities. IEEE Communications Magazine **56**(1), 99–103 (Jan 2018). https://doi.org/10.1109/MCOM.2018.1700587

[21] Joint Task Force Transformation Initiative: Guide for applying the risk management framework to federal information systems: a security life cycle approach. Tech. Rep. NIST SP 800-37r1, National Institute of Standards and Technology (Jun 2014). https://doi.org/10.6028/NIST.SP.800-37r1, https://nvlpubs.nist.gov/nistpubs/SpecialPublications/NIST.SP.800-37r1.pdf

[22] Kalkan, K. and Zeadally, S.: Securing internet of things (IoT) with software defined networking (sdn). IEEE Communications Magazine (2017). https://doi.org/10.1109/MCOM.2017.1700714

[23] Khodadadi, F., Dastjerdi, A.V., and Buyya, R.: Internet of things: an overview. In: Internet of Things, pp. 3–27. Elsevier (2016)

[24] Lear, E., Romascanu, D., and Droms, R.: Manufacturer Usage Description Specification (RFC 8520) (2019), https://tools.ietf.org/html/rfc8520

[25] Lu, X., Spear, M., Levitt, K., Matloff, N.S., and Wu, S.F.: A synchronization attack and defense in energy-efficient listen-sleep slotted MAC protocols. In: Emerging Security Information, Systems and Technologies, 2008. SECURWARE'08. Second International Conference on. pp. 403–411. IEEE (2008)

[26] Martin, B. and Coley, S.: Common weakness scoring system (CWSS). Internet, http://cwe.mitre.org/cwss (2014)

[27] Matheu-Garcia, S.N., Hernandez-Ramos, J.L., and Skarmeta, A.F.: Test-based risk assessment and security certification proposal for the Internet of Things. In: 2018 IEEE 4th World Forum on Internet of Things (WF-IoT). pp. 641–646. IEEE, Singapore (Feb 2018). https://doi.org/10.1109/WF-IoT.2018.8355193, https://ieeexplore.ieee.org/document/8355193/

[28] Matheu-Garcia, S.N., Hernandez-Ramos, J.L., Skarmeta, A.F., and Baldini, G.: Risk-based automated assessment and testing for the cybersecurity certification and labelling of IoT devices. Computer Standards & Interfaces **62**, 64–83 (Feb 2019). https://doi.org/10.1016/j.csi.2018.08.003, https://www.sciencedirect.com/science/article/abs/pii/S0920548918301375?via%3Dihub

[29] Mehmood, Y., Ahmad, F., Yaqoob, I., Adnane, A., Imran, M., and Guizani, S.: Internet-of-things-based smart cities: Recent advances and challenges. IEEE Communications Magazine **55**(9), 16–24 (2017)

[30] Meidan, Y., Bohadana, M., Mathov, Y., Mirsky, Y., Shabtai, A., Breitenbacher, D., and Elovici, Y.: N-BaIoT network-based detection of IoT botnet attacks using deep autoencoders. IEEE Pervasive Computing **17**(3), 12–22 (2018)

[31] Melcherts, H.E.: The internet of everything and beyond. Human Bond Communication: The Holy Grail of Holistic Communication and Immersive Experience p. 173 (2017)

[32] Mell, P., Scarfone, K., and Romanosky, S.: Common vulnerability scoring system. IEEE Security & Privacy **4**(6), 85–89 (2006)

[33] Miettinen, M., Marchal, S., Hafeez, I., Asokan, N., Sadeghi, A.R., and Tarkoma, S.: IoT sentinel: Automated device-type identification for security enforcement in IoT. In: 2017 IEEE 37th International Conference on Distributed Computing Systems (ICDCS), 5–8 June 2017. pp. 2177–2184. IEEE (2017). https://doi.org/10.1109/ICDCS.2017.284

[34] Neisse, R., Hernández-Ramos, J.L., Matheu, S.N., Baldini, G., and Skarmeta, A.: Toward a blockchain-based platform to manage cybersecurity certification of IoT devices. In: 2019 IEEE Conference on Standards for Communications and Networking (CSCN). pp. 1–6. IEEE (2019)

[35] NIST: Securing Small-Business and Home Internet of Things Devices: NIST SP 1800-15 (2019)

[36] Nurse, J.R., Creese, S., and De Roure, D.: Security risk assessment in internet of things systems. IT Professional **19**(5), 20–26 (2017)

[37] Pirretti, M., Zhu, S., Vijaykrishnan, N., McDaniel, P., Kandemir, M., and Brooks, R.: The sleep deprivation attack in sensor networks: Analysis and methods of defense. International Journal of Distributed Sensor Networks **2**(3), 267–287 (2006)

[38] Purdy, G.: Iso 31000: 2009 — setting a new standard for risk management. Risk Analysis: An International Journal **30**(6), 881–886 (2010)

[39] Qin, J., Ma, Q., Shi, and Y., Wang, L.: Recent advances in consensus of multi-agent systems: A brief survey. IEEE Transactions on Industrial Electronics **64**(6), 4972–4983 (2016)

[40] Qu, Y. and Chan, P.: Assessing Vulnerabilities in Bluetooth Low Energy (BLE) Wireless Network Based IoT Systems. In: 2016 IEEE 2nd International Conference on Big Data Security on Cloud (BigDataSecurity), IEEE International Conference on High Performance and Smart Computing (HPSC), and IEEE International Conference on Intelligent Data and Security (IDS). pp. 42–48. IEEE, New York, NY, USA (Apr 2016). https://doi.org/10.1109/BigDataSecurity-HPSC-IDS.2016.63, http://ieeexplore.ieee.org/document/7502262/

[41] Radanliev, P., De Roure, D.C., Nicolescu, R., Huth, M., Montalvo, R.M., Cannady, S., and Burnap, P.: Future developments in cyber risk assessment for the internet of things. Computers in Industry **102**, 14–22 (2018)

[42] Ramos, J.L.H., Geneiatakis, D., Kounelis, I., Steri, G., and Fovino, I.N.: Toward a data-driven society: A technological perspective on the development of cybersecurity and data protection policies. IEEE Security & Privacy (2019)

[43] Rao, S., Chendanda, D., Deshpande, C., and Lakkundi, V.: Implementing lwm2m in constrained IoT devices. In: 2015 IEEE Conference on Wireless Sensors (ICWiSe). pp. 52–57. IEEE (2015)

[44] Ratasuk, R., Prasad, A., Li, Z., Ghosh, A., and Uusitalo, M.A.: Recent advancements in M2M communications in 4G networks and evolution towards 5G. In: Proc. 18th IEEE International Conference Intelligence in Next Generation Networks (ICIN). pp. 52–57. Paris, France (Feb 2015). https://doi.org/10.1109/ICIN.2015.7073806

[45] Rubio-Loyola, J., Astorga, A., Serrat, J., Chai, W.K., Mamatas, L., Galis, A., Clayman, S., Cheniour, A., Lefevre, L., Mornard, O., Fischer, A., Paler, A., and Meer, H.D.: Platforms and software systems for an autonomic internet. In: 2010 IEEE Global Telecommunications Conference GLOBECOM. IEEE (2010)

[46] Selander, G., Mattsson, J., Palombini, F., and Seitz, L.: Object security for constrained restful environments (oscore) (July 2019), https://tools.ietf.org/html/rfc8613

[47] Sen, S., Koo, J., and Bagchi, S.: Trifecta: Security, energy-efficiency, and communication capacity comparison for wireless IoT devices. IEEE Internet Computing **22**(1), 74–81 (2018)

[48] Simon, D., Ph.D., D.B.D.A., and Eronen, P.: Extensible Authentication Protocol (EAP) Key Management Framework. RFC 5247 (Aug 2008). https://doi.org/10.17487/RFC5247, https://rfc-editor.org/rfc/rfc5247.txt

[49] Thanigaivelan, N.K., Nigussie, E., Kanth, R.K., Virtanen, S., and Isoaho, J.: Distributed internal anomaly detection system for internet-of-things. In: 2016 13th IEEE Annual Consumer Communications & Networking Conference (CCNC). pp. 319–320. IEEE (2016)

[50] Tsarouchis, C., Denazis, S., Kitahara, C., Vivero, J., Salamanca, E., Magana, E., Galis, A., Manas, J.L., Carlinet, L., Mathieu, B., and Koufopavlou, O.: A policy-based management architecture for active and programmable networks. IEEE Network **17**(3), 22–28 (2003)

[51] USC, S.: 3502. CNSSI-4009

[52] Zarpelao, B.B., Miani, R.S., Kawakani, C.T., and de Alvarenga, S.C.: A survey of intrusion detection in internet of things. Journal of Network and Computer Applications **84**, 25–37 (2017)

[53] Zhang, Z., Cui, P., and Zhu, W.: Deep learning on graphs: A survey. arXiv preprint arXiv:1812.04202 (2018)