INXU: A Flow-Based Intrusion Prevention System for Home IoT Networks

Sávyo V. Morais[1], **Claudio M. Farias**[1]

[1]Universidade Federal do Rio de Janeiro (UFRJ)
Rio de Janeiro – RJ – Brasil

savyo.morais@labnet.nce.ufrj.br, cmicelifarias@cos.ufrj.br

Abstract. *Due to the low resources and maintainability in home Internet of Things (IoT) devices, they can represent a risk to end-user's security and privacy. Several proposals tried to manage new vulnerabilities in this scenario, but it is difficult to keep signatures updated or identify anomalous traffic. To reinforce home IoT security, we propose INXU, a flow-based Intrusion Prevention System that protects home IoT devices by blocking traffic related to well known malicious activities. INXU introduces the concept of Malicious Traffic Description (MTD), a data-model to describe traffic related to malicious activities that enables Security Experts to protect home networks and keeps end-user's privacy. Experiments using Mirai botnet have shown the efficacy of our solution.*

1. Introduction

The Internet of Things (IoT) is a socio-technological phenomenon that arises from the human need to monitor and control the environment in which they are inserted, combined with the growing development of Information and Communication Technologies during the last two decades [Kramp et al. 2013]. Thus, IoT assumes its role in transforming initially disconnected devices – such as fridges, doors, cars, other everyday objects, or even environments with sensors and actuators – into connected devices accessible from any part of the world through the Internet. This technology enables the automation of tasks by replacing manual activities and accelerating the growth of the number of devices connected to the Internet.

Currently, one of the most common uses of IoT is in home environments. In this context, a large amount of data about end-users daily lives can be extracted from IoT devices, putting their privacy at risk if an attacker gets access to the devices. Besides privacy, exposure to cyber-physical systems can incur physical harm, such as if a denial of service attack disables a smoke alarm during a fire [Habibi Gharakheili et al. 2019].

The problems caused by these devices' vulnerabilities go beyond their end-users, affecting the whole Internet ecosystem when infected and incorporated into botnets [Marzano et al. 2018] to be used for interrupting online services by carrying Distributed Denial of Service (DDoS) attacks. These attacks affect the Internet's stability, as they commonly take advantage of the Domain Name System (DNS) infrastructure to amplify attacks [Schutijser 2018].

Most of the current weaknesses of home IoT systems are caused by end-users and manufacturers. By their low expertise in configuring and operating IoT devices, end-users commonly create security breaches [Goutam 2019, Schutijser 2018]. Manufacturers, on the other hand, often due to budget constraints or inexperience with secure development [Garcia-Morchon et al. 2019], commonly release IoT devices with serious security breaches, such as the hard-coding of weak access credentials or the usage of insecure or outdated software components [OWASP 2018].

To enhance IoT devices' security, the Internet Engineering Task Force (IETF) released RFC 8520, named Manufacturer Usage Description (MUD) Specification. MUD is an Internet Standard that allows the manufacturer to describe the minimal network configuration that an IoT device needs to work appropriately [Lear et al. 2019]. MUD describes the device's network communications with Access Control Lists (ACLs), specifying the connections' header information of the Network and Transport layers of the TCP/IP stack.

Despite MUD's focus on operations, the standard reinforces security by reducing the device's threat surface. It does this by defining that only the described traffic is allowed. Otherwise, it is dropped. This resource prevents the exploitation of vulnerabilities in services not designed for the device. On the other hand, MUD does not protect against attacks on services implemented on the device. Furthermore, as manufacturers are the only security authority involved in MUD's process, they can become a threat due to the possibility of implementing and describing in MUD a backdoor without the proper disclosure for end-user.

Therefore, after the literature review described in the Dissertation's Chapter 3, it was possible to identify a lack of solutions that allows specialized support in the decision-making process for security measures, preserve the end-users privacy, and enable the collective mitigation of damage from recently discovered attacks on distinct networks. Therefore, envisioning to overcome this gap, this work proposes INXU (Intra Network eXposure analyzer Utility). INXU takes advantage of the MUD-based network communication graph to prevent the exploitation of well-known vulnerabilities. To do this, INXU blocks threats on the home network after identifying them by comparing the signature of well-known malicious activities with the traffic flow allowed by the MUD. We will call flow any continuous communication between two endpoints.

The core component of INXU is Malicious Traffic Description (MTD), a document produced by a security specialist that describes ongoing malicious activities and well-known vulnerabilities and helps INXU find chains of connected IoT devices that can expose them to these threats. On top of MUD's threat surface reduction, INXU adds another security layer that enables protection against incidents not addressed or even caused by the manufacturers.

Another relevant feature of INXU is its architecture that enables a Security Operation Center (SOC) to protect multiple distinct networks by sharing MTDs that can be easily interpreted inside the local network. This feature makes INXU a tool to protect end-users and the entire Internet ecosystem by making the operation of botnets and other attacks that affect the Internet's stability more difficult.

2. Related Work

This section discusses the solutions proposed by other authors to strengthen the security of home IoT networks by detecting and/or preventing intrusions. We highlighted proposals based on anomaly detection or others that explore access control to mitigate known risks.

Some studies are trying to protect the home IoT ecosystem by refining MUD rules.

The proposal in [Goutam 2019], intending to prevent the spreading of malware into the local network, blocks any communication between IoT devices in the same network, allowing only connections to Internet hosts. Besides its protection, this approach blocks legitimate communications between local IoT devices and keeps allowing infection by Internet communications.

[Jonsdottir et al. 2017] proposes an approach that combines anomaly detection and penetration tests to identify incidents and threats in the network. On the other hand, it reduces the protection effectiveness by not accepting updates on the penetration tests to prevent recently discovered attacks.

The work in [Schutijser 2018] proposes an algorithm to trace the network communication profile of each connected IoT device, blocking the traffic that falls outside of the device's typical behavior – regardless of whether the traffic is allowed by MUD. The main issue with this proposal is the possibility of building a profile in the course of malicious activities. Another point to consider is that this proposal does not provide the means to share knowledge about the detected malicious activities.

Similarly, the proposal in [Wan et al. 2020] traces the IoT devices' common behavior to detect outlier traffic. Besides the profile generation, the authors complement the solution with training machine-learning models to detect well-known attacks and apply various anomaly detection methods to identify new attacks, both using network information. Unlike the proposal in [Schutijser 2018], the proposal in [Wan et al. 2020] applies for both internal and Internet traffic.

In [Al-Shaboti et al. 2018], a security framework for home IoT networks is proposed. It combines Mandatory Access Control (MAC) – a concept similar to MUD – with Discretionary Access Control (DAC), which enables customization of network access control. The proposal, however, only specifies means for the end-user to manage DAC. This point affects the effectiveness of security measures, as the user is potentially inexperienced and may not understand the risks to which the network is exposed. The authors also suggest outsourcing DAC to a third party, but there are no further details or mention of privacy protection mechanisms.

3. Malicious Traffic Description

The data model for describing malicious traffic has to enable defining traffic so that distinct networks can interpret and implement security measures, no matter the connected IoT devices or network topology. Another critical feature to be addressed by the data model is to allow the association between the detected exposure and the malicious activity that exploits it and the grouping of vulnerabilities related to the same malicious activity. Since, as far as we know, there is no other data model to address these requirements, we designed the data model described below for the MTD.

The MTD data model uses the ACLs under YANG language to describe the malicious traffic, addressing the classification feature. Furthermore, such as in MUD, we defined two network address abstractions to describe the traffic so that different networks can adapt the description to its context: one abstraction for addresses in the local networks, and the other for using domain names to hosts on the Internet. The data model also includes control fields that support the manageability of the *MTD File*, so the contained data can be categorized into control data and description data.

The traffic description fields are divided in *attack-descriptions* and *malware-descriptions* containers. These two categories are needed because one malware description must aggregate multiple different attacks and can also use other traffic - here called not attack traffic - related to the malware operation. This aggregation is important for the security measures decision-making process, as sometimes only a traffic combination makes the malware effective or blocking just one type of traffic can almost disable a malware, such as the Mirai's Command and Control traffic [Kolias et al. 2017].

Moreover, we also included context information in the MTD data model to specify the correlation between the described traffic, determine the combinations of exposures that become a risk, and suggest the action to be taken with each detected risk. So, based on the defined in [Mozzaquatro et al. 2015], this work establishes a threat as an effective risk of one or more vulnerabilities exposure being exploited by an attacker. Another concept we consider from the ontology is vulnerability, which, besides having an associate severity, does not directly represent a risk because of the possibility of hiding it behind security mechanisms, such as blocking its exposure. So, in short words, an asset is under threat only when an attacker can exploit one or more vulnerabilities to take advantage of it. Thus, merging the concepts from the ontology and this works, we defined the following statements:

- Each Access Control Entry (ACE) has an associated severity defined by the unsigned integer field named risk. When exposure to the ACE is detected, its risk is considered part of its ACL's vulnerability classification;
- Each ACL has alert-threshold and risk-threshold fields, both represented by unsigned integer values. When the sum of the exposed ACEs risks reaches the risk threshold, the exposure to the ACL is considered a vulnerability;
- Under the attack category, ACLs that expose vulnerabilities are considered threats and should be blocked;
- In the malware category, each described malware contains a list of critical ACL sets. A malware is classified as a threat when at least one set of critical ACLs contains all its ACLs classified as a vulnerability exposure. When one set's condition is satisfied, it's associated action to take has to be triggered. The three possible actions to be taken are listed below:
  - *block-all*: blocks all ACLs that expose vulnerabilities related to the malware. Expected to be used when any traffic associated with the malware threatens the IoT device;
  - *block-attack*: blocks all ACLs that expose vulnerabilities under the malware's attack-traffic group. Expected to be used when only risky ACLs that are associated with attacks that threaten the IoT device;
  - *block-not-attack*: blocks all ACLs that expose vulnerabilities under the malware's not-attack-traffic group, plus all the alert ACLs under the malware's attack traffic group. Expected to be used when just blocking the operation traffic of the malware prevents exploitation.

A more detailed description of the data model is available in Chapter 4 of the Dissertation.

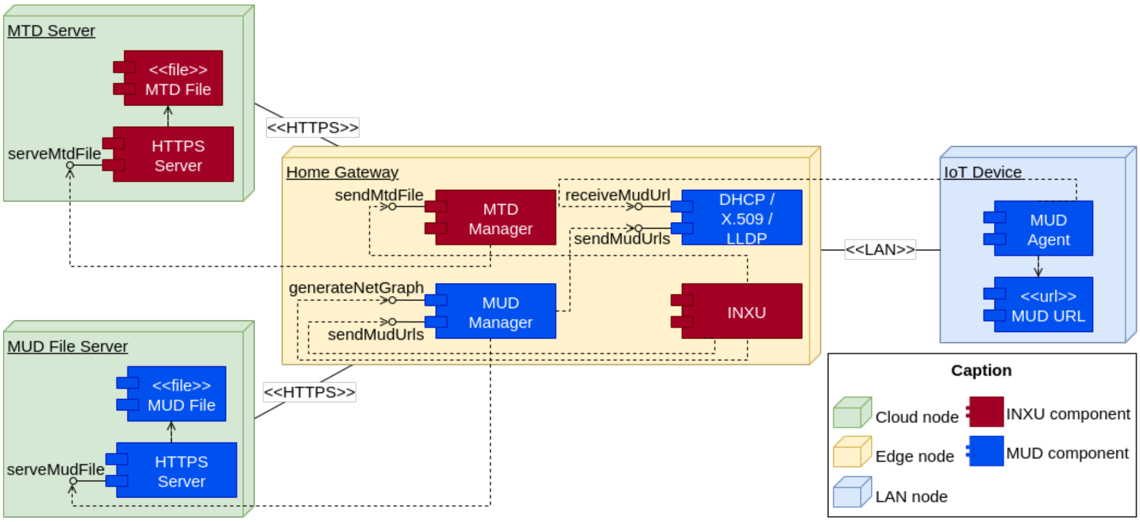

Figure 1. INXU Architecture

4. Intra-Network eXposure analyzer Utility

In this section, we present the proposal of INXU: a security tool to give fast responses to new vulnerabilities in home IoT networks. INXU was designed to have as main features: (i) enable quick responses to new vulnerabilities; (ii) allow mitigation of the damages of a new vulnerability, simultaneously in multiple distinct networks; and (iii) enable a decision-making process about security measures on the network edge, avoiding the disclosure of private information to third parties. The lack of solutions that provide means to share information about vulnerabilities, associated with the difficulty of generating generic security countermeasures to multiple distinct networks justifies features (i) and (ii), and the principle of preserving privacy supports (iii).

In an overview, while MUD builds a network access allowlist based on the connected IoT devices, INXU creates a blocklist over MUD's allowlist to protect the network from malicious activities. To do this, INXU enables a security experts team to describe the traffic of ongoing malicious activities using the data model defined in Section 3. With this, the security experts can use the INXU to protect multiple distinct networks when releasing new MTD Files for every new malicious activity discovered, in a process similar to the antivirus programs vaccines. In the home network, the network manager configures the MTD URL into the home gateway to receive the MTD Files and process them on edge, comparing them with the network graph generated by the MUD manager. Finally, INXU can identify and block possible threats.

4.1. Architecture

The architecture of INXU is illustrated in Figure 1, with the components distributed between the following nodes: *MTD Server*, *MUD file server*, *Home Gateway*, and *IoT Device*.

The *Home Gateway* is the main node, placed on the network edge, and has the responsibility of collecting MUD-related data, which includes receiving the *MUD URLs* and collecting the *MUD files*. It is also responsible for managing MTD-related information, such as collecting *MTD file* and processing it to identify exposure to vulnerabilities.

The *Home Gateway* contains the following software components: *MTD manager*, *MUD manager* and *INXU*.

The *IoT Device* node, situated on the Local Area Network (LAN), represents the IoT devices connected to the home LAN. In the context of INXU, the *IoT Device* is responsible for informing the *MUD URL* of its respective *MUD file* to the *MUD manager* using the extensions to DHCP, X.509, or LLDP created in RFC 8520 to support MUD operation. The *IoT Device* is composed of the software component *MUD Agent* and the data component *MUD URL*.

The *MUD file server* is placed on the cloud. It is maintained by the IoT device manufacturer and is responsible for responding to requests made by the *MUD manager* looking for the *IoT Devices MUD files*. These components interact with the *MUD managers*, serving the *MUD files* and assuring the verifiability of its authenticity. The *MUD file server* is composed by the data component *MUD file* and by the software component *HTTPS Server*.

Also placed on the cloud, the *MTD Server* is responsible for storing and delivering the malicious traffic descriptions made by a security expert. This component was designed to enable trusted third-party specialists to share knowledge about well-known malicious activities affecting home IoT and allow home IoT networks to make use of this knowledge to protect themselves. The *MTD Server* is composed by the software component *HTTPS Server* and by the data component *MTD File*.

4.2. Exposure Analysis Algorithm

The exposure analysis algorithm of INXU uses malicious traffic descriptions from *MTD file* to compare with the MUD-based communication graph and tries to detect vulnerabilities in the network. In this context, INXU identifies one exposure when some graph edge matches with any entry of the *MTD file*.

Based on the *MUD files*, the hosts are represented by nodes on the network communication graph generated by *MUD manager*. The host network address represents the nodes. The graph edges represent TCP, UDP, or ICMP communications, where a directed edge represents a communication path.

The algorithm iterates over the graph edges and the *MTD File* ACEs to compare all the edge-ACE couples. When doing this, it compares the source and destination addresses, the protocol used, and the source and destination ports (or message type and code in the case of ICMP).

After identifying the matches between ACEs and graph edges, the algorithm verifies if the combination of exposures in a device may become risky based on the ACL thresholds. Finally, the algorithm blocks risky ACLs related to *attack-descriptions* and assesses threats related *malware-descriptions*. The threat assessment in the context of *malware-descriptions* verifies if a set of critical ACLs is classified as risky, and if true, takes the action defined in the MTD file.

5. Experiments

To validate and demonstrate the proposal, we carried out experiments that exposed an IoT network to the botnet Mirai's action to assess the degree of protection provided by

INXU. This experiment compares INXU's ability to mitigate DDoS attacks on a network, comparing its performance with MUD's protection and an unprotected network. The full description of the experiment environment and scenarios is in Dissertation's Chapter 6.

As a result of the experiments, we identified that INXU provides good protection against Mirai's activities when compared with MUD's protection. This is evidenced by the substantive reduction of new Mirai infections, scans reported, and the number of controllable bots in many experiment scenarios when we compared INXU and MUD results. Given these results, we believe that this solution can protect home IoT networks against other families of botnet, ransomware, or worm malware.

6. Conclusion

This work was defended in March 2021. The main contributions of this work can be divided into conceptual and specific contributions. The conceptual contributions were found due to the investigations in the published works and the experience gained during the development of this work. The specific contributions are as follows.

- Development of a flow-based IPS for home IoT networks that allows SOCs to protect multiple distinct networks, preserving end-users privacy on home networks;
- The proposal of MTD as a data model to describe malicious traffic to the Internet of Things. The proposed model allows data portability between different networks without significant loss of information. The model also allows the grouping of traffic related to the same malicious activity to assist in decision-making on security measures.

For these contributions, several articles were published and public presentations were held. The following are these directly related to this work:

- **FULL-PAPER** in *VII Workshop pré-IETF 2020* – INXU - A Security Extension for RFC 8520 to Give Fast Response to New Vulnerabilities on Domestic IoT Networks. **Best paper award**;
- **ROUND TABLE** in *X Fórum da Internet no Brasil* – Internet das Tretas: como a Internet das Coisas afeta nossa segurança e privacidade?. Available at: `https://forumdainternet.cgi.br/workshop/detalhe/227/`, accessed on 06/17/2022;
- **ROUND TABLE** in *United Nations Internet Governance Forum 2020* – Internet of Things: Trust, Trick or Threats?. Available at: `https://www.intgovforum.org/en/content/igf-2020-ws-325-internet-of-things-trust-trick-or-threats`, accessed on 06/17/2022;
- **JOURNAL ARTICLE** in *IEEE Communications Standards Magazine* – Malicious Traffic Description: Towards a Data Model for Mitigating Security Threats to Home IoT;
- **ROUND TABLE** in *United Nations Internet Governance Forum 2021* – The Internet of Things is a Ticking Clock: Secure Design Now. Available at: `http://www.intgovforum.org/en/content/igf-2021-ws-239-the-internet-of-things-is-a-ticking-clock-secure-`
accessed on 06/17/2022;

- **INTERNET-DRAFT** in *IETF* – Intra-Network eXposure analyzer Utility Specification. Available at: `https://datatracker.ietf.org/doc/draft-morais-iotops-inxu/`, accessed on 06/17/2022.

References

Al-Shaboti, M., Welch, I., Chen, A., and Mahmood, M. A. (2018). Towards secure smart home iot: Manufacturer and user network access control framework. In *2018 IEEE 32nd International Conference on Advanced Information Networking and Applications (AINA)*, pages 892–899.

Garcia-Morchon, O., Kumar, S., and Sethi, M. (2019). Internet of Things (IoT) Security: State of the Art and Challenges. RFC 8576.

Goutam, S. (2019). Hestia: Simple least privilege network policies for smart homes. Master's thesis, North Carolina State University.

Habibi Gharakheili, H., Sivanathan, A., Hamza, A., and Sivaraman, V. (2019). Network-level security for the internet of things: Opportunities and challenges. *Computer*, 52(8):58–62.

Jonsdottir, G., Wood, D., and Doshi, R. (2017). IoT network monitor. In *2017 IEEE MIT Undergraduate Research Technology Conference (URTC)*, pages 1–5.

Kolias, C., Kambourakis, G., Stavrou, A., and Voas, J. (2017). DDoS in the IoT: Mirai and Other Botnets. *Computer*, 50(7):80–84.

Kramp, T., Van Kranenburg, R., and Lange, S. (2013). Introduction to the internet of things. In *Enabling Things to Talk*, pages 1–10. Springer, Berlin, Heidelberg, Berlin, Heidelberg.

Lear, E., Droms, R., and Romascanu, D. (2019). Manufacturer Usage Description Specification. RFC 8520.

Marzano, A., Alexander, D., Fazzion, E., Fonseca, O., Cunha, I., Hoepers, C., Steding-Jessen, K., Chaves, M. H. P. C., Guedes, D., and Jr., W. M. (2018). Monitoramento e caracterização de botnets bashlite em dispositivos iot. In *Anais do XXXVI Simpósio Brasileiro de Redes de Computadores e Sistemas Distribuídos*, Porto Alegre, RS, Brasil. SBC.

Mozzaquatro, B. A., Jardim-Goncalves, R., and Agostinho, C. (2015). Towards a reference ontology for security in the Internet of Things. In *2015 IEEE International Workshop on Measurements Networking (M N)*, pages 1–6.

OWASP (2018). Owasp top 10 internet of things 2018. `https://www.owasp.org/index.php/OWASP_Internet_of_Things_Project`. Acesso em 10/01/2020.

Schutijser, C. (2018). Towards automated ddos abuse protection using mud device profiles. Master's thesis, University of Twente.

Wan, Y., Xu, K., Xue, G., and Wang, F. (2020). IoTArgos: A Multi-Layer Security Monitoring System for Internet-of-Things in Smart Homes. In *IEEE INFOCOM 2020 - IEEE Conference on Computer Communications*, pages 874–883.