FamilyGuard

Subjects: Computer Science, Cybernetics

Contributor: Pedro H. A. D. de Melo, Rodrigo Sanches Miani, Pedro Frosi Rosa

The residential environment is constantly evolving technologically. With this evolution, sensors have become intelligent interconnecting home appliances, personal computers, and mobile devices. Despite the benefits of this interaction, these devices are also prone to security threats and vulnerabilities. Ensuring the security of smart homes is challenging due to the heterogeneity of applications and protocols involved in this environment. This entry proposes the FamilyGuard architecture to add a new layer of security and simplify management of the home environment by detecting network traffic anomalies.

Keywords: machine learning ; anomaly detection ; network security

1. Introduction

The accelerated growth of applications and devices for the Internet of things (IoT) means excellent amenities for people in diverse areas, such as smart homes, industrial automation, healthcare, electricity control, cities, and smart grids [1]. However, deploying IoT applications in different scenarios can also introduce security threats. An example is the Mirai botnet, first identified in August 2016 by a security research group called MalwareMustDie. Mirai scours the Web for smart home devices that have default usernames and passwords and then enlists the devices in attacks that hurl junk traffic at an online target [2].

IoT security is also affected by software and hardware constraints on such devices. For example, implementing encryption and authentication mechanisms on these devices can be challenging [3]. Another problem is that IoT applications might rely on users' personal information to provide services. However, collecting, transferring, and using this information increases the risk of damaging users' privacy [4][5]. Thus, one of the most prominent research challenges for the networking and security community is providing a cost-effective balance between the use of sensitive data and privacy [6].

There are several types of non-standard IoT devices and protocols in smart home scenarios whose lack of standardization can pose security risks to this environment. For example, the owner of cameras, TVs, and smart sensors must manage the security updates individually, even if they do not have the time or the knowledge to deal with this situation. Moreover, without proper threat monitoring, infected mobiles devices might compromise the smart home environment as they ingress into the home network.

Several works propose using SDN to create a safer environment for smart homes [7][8][9]. When dealing with security solutions for smart homes, researchers should consider several challenges when deploying the solution in the environment: heterogeneity of devices and protocols and resource constraints such as small memory, low power consumption, and low computing power. Based on these security challenges, several researchers pursue solutions that include privacy-based, risk-based, and role-based approaches [10][11][12].

The scientific community offers advances in the security of smart home networks [7][8][9][10][11][12]. However, most solutions do not show how security mechanisms could be deployed to operate in the residential environment. Therefore, there still exist research gaps in security requirements for smart home networks consisting of device authentication, network monitoring, secure session key management, physical protection, information security, and user authentication [13][14]. Another limitation identified in the state of the art is about solutions that detect anomalies. Usually, they do not present the complete structure of the solution; in other words, it is unclear how network traffic would be captured and classified.

Therefore, this work presents FamilyGuard, an architecture that provides a way to define and deploy mechanisms that meet the security requirements of a smart home environment. FamilyGuard uses the SDN paradigm to analyze and manage home area network (HAN) flows, providing flexibility when dealing with security issues, such as traffic monitoring to identify and mitigate threats. The architecture uses machine learning (ML) algorithms to identify anomalous behavior based on network traffic to provide additional protection to residential environments in terms of information security. The

idea is that the proposed architecture would help transform the static residential environment into a dynamic one. In other words, FamilyGuard will provide means to deal with changes in the environment and respond to threats.

2. Background

2.1. Smart Home Concepts

A smart home can be defined based on two perspectives. The first perspective considers a house equipped with Information and Communication Technology (ICT) and with connected devices that can be remotely monitored and controlled to meet the needs of residents. In the second perspective, smart homes and other related buildings are seen as elements of flexibly and interactively connected energy systems on a broader scale. These two perspectives are useful for the families themselves and the electrical system as a whole [15].

In the home environment, “intelligence” can integrate electrical devices and services (e.g., heating, lighting, security, photovoltaic generation, electric vehicle charging) that the occupants or other agents remotely control. In addition, sensors and processors can also obtain and apply knowledge about a house, operating independently of direct human action. Together, intelligent or smart household electrical devices have the potential to help manage the network and promote system efficiency, helping to reduce peak demand and match demand with supply in real time. This assists the integration of more distributed renewable generation into electrical systems [16].

There are questions about what technologies are needed at what times to qualify a home as smart. Smart TVs or smartphones, for example, can be classified as smart devices since they allow communication between the house and the outside world. Researchers consider that a high level of device connectivity inside and outside the residential environment, together with the reliance on this connectivity for daily activities, is relevant to whether a home can be named “smart”. The definition presented here is associated with physical and operational factors and assumes that the functionalities are beyond the typical limits of a traditional house [17].

A smart home is a communication network that connects sensors, appliances, controls, and other devices to enable remote monitoring and control by occupants and others to provide frequent services to residents and the electrical system. Therefore, researchers can say that this concept is closely linked to the term Internet of things (IoT) because smart homes provide facilities for the occupants of the residential environment and interconnect objects and sensors.

2.2. Internet of Things (IoT)

The term Internet of things (IoT) refers to the network interconnection of objects (sensors, actuators, devices, etc.), which allows sharing of data and information to accomplish some task [18]. The term was also defined in the Recommendation ITU-T Y.2060 (June 2012) as “a global infrastructure for the information society, enabling advanced services by interconnecting (physical and virtual) things based on existing and evolving interoperable information and communication technologies” [19].

The advancement of IoT applications and devices is providing progress in many areas, each of them with different requirements and goals [1]. The smart home, for example, aims to provide greater convenience for people, the smart grid seeks to obtain higher efficiency, reliability, and sustainability in energy systems, and smart-agro offers solutions to improve productivity in agriculture.

As the objectives of these areas are entirely different, the security mechanisms employed in these environments need to be adapted to their needs. This work seeks to contribute to the domain of smart homes where there are several challenges, such as interoperability, context-aware middleware, energy-aware/efficient consumption, and security [20].

Since the devices are interconnected using different network standards [21] (Zwave, Insteon, Bluetooth, Zigbee, Ethernet, Wifi), they can be targeted by several types of attackers, from cyber terrorists to script kiddies. Furthermore, smart homes are usually operated by non-expert users in the security of information [22]. Therefore, it is important that the solution involves a high level of automation at the lowest possible cost.

2.3. IoT Security Issues

Smart home environments are full of personal information handled by devices, protocols, and services to provide user convenience. Due to the sensitivity of private information, security requirements become more and more necessary in such a scenario. However, managing security controls in these heterogeneous environments is a significant challenge. According to Bugeja et al. [23], security issues can be divided into three different layers of the generic IoT architecture:

- Device Issues: devices in IoT scenarios have performance constraints such as CPUs with low clock rates, low memory, and low throughput. These hardware limitations make it difficult to implement security mechanisms, such as encryption, which are computationally consuming. Many devices do not have a management interface, making it difficult to create security mechanisms such as authentication. For this reason, users need to trust websites or smartphones to manage their devices and information. Another critical issue is that objects in a smart home are physically accessible and may be subject to physical attacks such as tampering by a visitor in the home or even by the household to reduce the charge for some service that relies on smart meters [23].

- Communication Issues: to interconnect many different devices in a smart home, multiple bridges, hubs, or gateways and many communication protocols are required, which makes it difficult to implement adequate security mechanisms. The smart home environment is highly dynamic, where a device can join or leave the network at any time, reinforcing the idea of developing resilient security mechanisms that can handle asset management. A large number of existing protocols and the restricted capabilities of each device make traditional security mechanisms unsuitable for the smart home environment [21].

- Service Issues: to reduce the number of vulnerabilities, patch fixes need to be installed periodically. However, performing this process on all devices could be problematic since the firmware of these devices and protocols may not support these security updates dynamically [23].

Based on these questions, researchers can conclude that implementing traditional security mechanisms to tackle problems in smart homes may not be the best strategy. Therefore, it is necessary to explore the use of alternative approaches. One of them is using a Software Defined Networking (SDN) that, when applied to the IoT context, will change the network from a static to an adaptive or programmable state, which is a necessary feature in a heterogeneous environment.

Several areas in IoT are gaining benefits with the SDN paradigm, for example, smart cities [24], smart grid [25], smart homes [26], among others. Kalkan and Zeadally [5] conclude that IoT challenges such as security, scalability, and heterogeneity can be solved through the dynamism and flexibility of SDN. However, integrating SDN and IoT environments will open up new security risks, such as attackers granting themselves unsupervised access to SDN elements and exploiting either weak or nonexistent access control mechanisms. Another example is the creation of exploits for vulnerable network components for the purpose of installing them on rogue devices or binding them to remote connections to the network. Researchers must strive to mitigate these threats in the future [27].

2.4. Software Defined Networking (SDN)

With SDN [28], networks became programmable, making it possible to virtualize network functions and manage services and applications through logically centralized platforms. SDN introduces control plane and data plane separation. One of the main goals of this separation is creating an agile and flexible network that is capable of handling rapid changes, supporting some requirements of IoT environments that traditional networks may not be able to provide [5].

The SDN controller is responsible for managing the entire network, while switches are responsible for operating the data plane based on the settings specified by the controller [29]. The Southbound API is offered by the OpenFlow protocol, and its assignment is to enable communication between the SDN controller (control plane) and network switches (data plane), thus allowing the controller to define the network flows and the API Northbound interface between the controller and higher layer applications or programs.

The SDN is materialized through the OpenFlow protocol [30] and has a set of specifications maintained by the Open Networking Foundation (ONF) [31].

3. FamilyGuard

FamilyGuard is a security architecture designed for smart homes equipped with different types of computing devices. The main goal of FamilyGuard is to anticipate and respond to the needs of residents, working to promote their comfort, convenience, safety, and entertainment. According to [32], home environments lack reliable security solutions since households occasionally only have antivirus software installed on computers and rarely have perimeter defenses installed on their networks such as an intrusion detection system (IDS) or a firewall. Besides, the systematic literature survey presented in Section 3 showed that proposed solutions only focus on IoT devices and ignore the other communication devices present in the HAN. The solution serves the entire residential environment, detecting threats in IoT and mobile and traditional devices, such as computers and laptops.

Figure 1 presents a high-level view of the architecture. Security service providers (SSPs) are responsible for providing management and configuration features that help protect existing information on a home area network (HAN), such as machine learning (ML) models to identify anomalies in network behavior. The user can access the services provided by SSPs through an application on their mobile device and hire a Security-as-a-Service (SECaaS) that meets their needs. HANs can send information and alerts to SPPs if they identify anomalous situations in the network. In this way, by managing multiple households, an SPP can work collaboratively to detect threats.

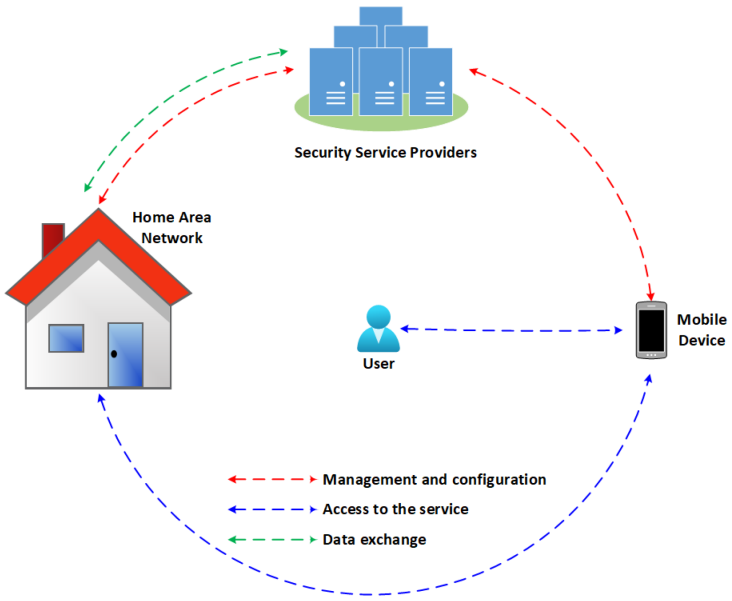

Figure 1. High level architecture. The user can access some of the security services provided by SSPs using mobile devices.

Each family living in a smart home environment has specific needs. Therefore, besides dealing with the heterogeneity of devices, it is necessary to address the specificities of each smart home environment. Researchers will tackle this issue by analyzing the inbound and outbound network traffic. With this in mind, researchers propose the following essential components of FamilyGuard: Home Surveillance Unit (HSU), controller, network flow generator (NFG), and Central Security Assistant (CSA). **Figure 2** shows each of the components in the environment and how they communicate.

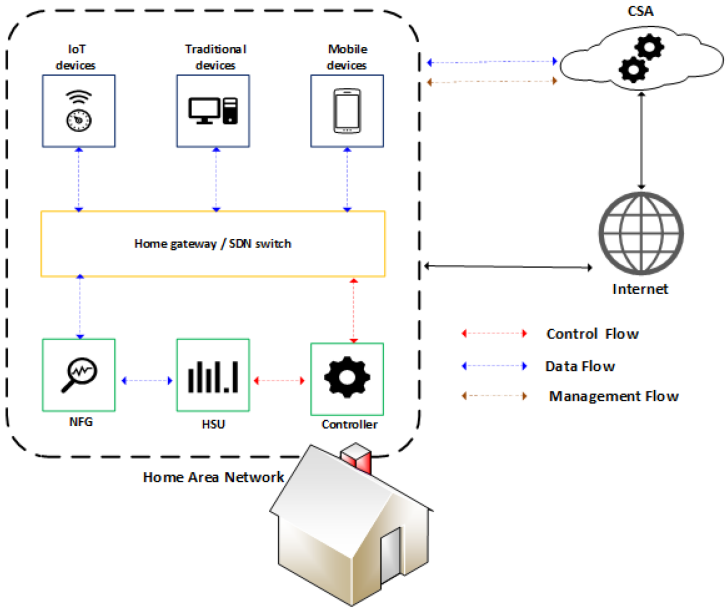

Figure 2. Communication between the architecture components.

The NFG receives network packets transmitted by the home gateway and generates network flows analyzed by the HSU. Upon identifying any anomalous behavior, the controller receives a notification, and the device involved in the anomalous behavior can be blocked to mitigate potential threats. The rules that determine the actions to be taken when identifying a threat depend on the HSU configuration; it may issue only a notification or completely block device communication.

3.1. Home Surveillance Unit (HSU)

The HSU is responsible for receiving flows from the network, performing traffic analysis tasks, and managing the network through the controller interface. The HSU uses a workflow to handle incoming network traffic flows and perform analysis. A workflow can contain multiple artificial intelligence models to handle flows and perform analysis sequentially. For example, the HSU receives a flow and forwards it to Workflow A, which initially prepares the flow by removing unnecessary characteristics; another model identifies the traffic type of the flow and forwards it to the last model to verify if the flow has anomalous behavior. The structure HSU has two main components: the Security Orchestrator and the AI Workflow Orchestrator, shown in **Figure 3** and described below:

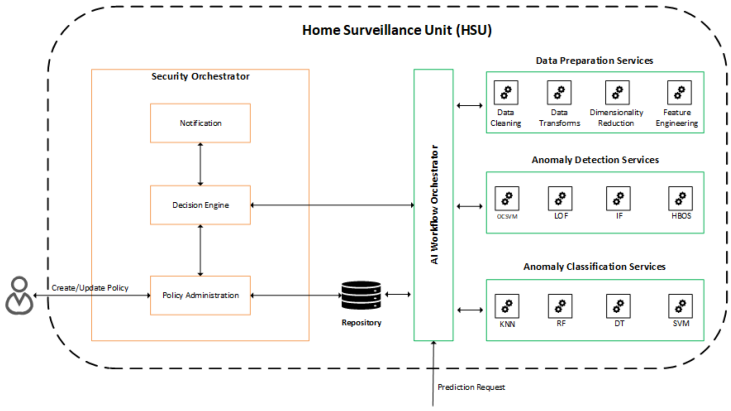

Figure 3. Communication between the services present in the HSU.

- Security Orchestrator (SECOR): is responsible for managing, configuring, and providing notification information related to security policies applied in HAN. Upon receiving the result of a prediction from the AI Workflow Orchestrator, the decision engine checks the applicable security policies to defeat or mitigate the detected threat and then notifies the controller so that the change in the flow table is performed to block or limit access to the device on which the threat was detected. The user can also change the security policies, being able to choose to either block the infected device immediately or be notified and make the blocking decision later.

- AI Workflow Orchestrator (AIWO): in charge of receiving prediction requests and generating results for the Security Orchestrator, being also responsible for managing and administering services available for predictions, categorized as Data Preparation Services, Anomaly Detection Services, and Services of Anomaly Classification.

Figure 4 exemplifies the structure of the AIWO, which allows for the definition of multiple workflows using a set of available models. SSPs and users are set free to create and define workflows to meet specific HAN needs with this structure. For example, researchers can have workflows that identify anomalies in specific scenarios, such as in IoT devices, and ignore traditional home devices, such as notebooks and personal computers.

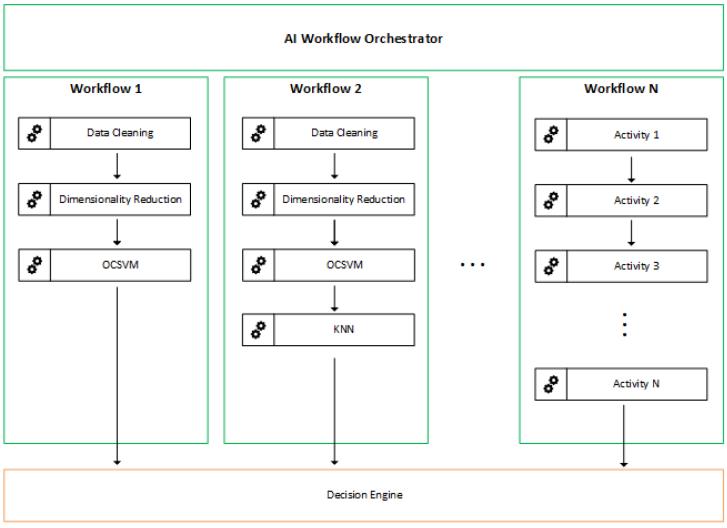

Figure 4. Workflow Orchestrator is responsible for managing several workflows.

It is essential to highlight that researchers can have several workflows in operation; however, a prediction request does not have to go through all workflows. AIWO allows a configuration in which flows from certain devices are routed to

specific workflows. In this way, researchers can have models that identify anomalies at a lower level of granularity.

3.2. Controller

The SDN controller is installed in a computing device responsible for receiving the HSU instructions and relaying them to the home switch (HS), where all devices in the home must be connected using wired or wireless connections. The approach used in this work was to deploy the HSU and the controller on the same device. The main advantage of running the HSU with the controller was maintaining a simple deployment scenario in a smart home, avoiding using another device to execute the HSU instructions. Performance is another aspect that motivated this decision, as the exchange of messages between the controller and the HSU is faster when they are on the same device.

Using an SDN controller in the scope of a residential environment could bring several advantages such as greater agility, more programmability, centralized data control, simplified operations, and better management of network resources. The HSU, for example, can send a blocking rule to the controller when it identifies a threat. In this way, the HSU takes advantage of the dynamism provided by the use of the controller to actively act on the network to mitigate threats that could harm users' privacy.

Despite its benefits, adding an SDN controller in a residential scenario might not be a simple task for a user. However, for the scenario involving smart homes, a small device with low cost and good computing power can meet this need. The ISP or SSP can also provide an SDN-based device when offering such a solution for home users. The potential complexity for end users in dealing with an SDN controller is eliminated by adopting the FamilyGuard architecture, which performs all the management (communication) of the controller and provides APIs for developing web pages or mobile applications to control their devices and information.

3.3. Network Flow Generator (NFG)

The NGF component collects traffic that traverses a given network, or network segment, to generate [33] network flows. According to RFC 7011, a flow can be defined as a set of IP packets passing an observation point in the network during a specific time interval. Network flows have been used in several network applications ranging from troubleshooting connectivity issues to planning future bandwidth allocation. Here, researchers use network flows to identify and mitigate security issues.

Monitoring network flows can provide insights into how a network operates, its overall utilization, application usage, potential bottlenecks, anomalies that can signal security threats, etc. Several different standards and formats are used in monitoring network flows, including NetFlow [34], sFlow [35], and Internet Protocol Flow Information Export (IPFIX) [36].

The adoption of a solution to generate network flows is encouraged by the benefits of detecting anomalous traffic and other threats to network security. The information in the IP packet header provides the basis for generating network flows. The amount of data processed by the flow-based intrusion detection system is less, as it contains summary information. Another factor contributing to the decision to use network flows for analysis is the number of network applications that use end-to-end encryption. Since flow-based inspection only works with statistical features extracted from the packet header, this approach raises fewer privacy concerns than packet-based inspection because user information is protected from any intermediate scans.

Despite the benefits, flow-based intrusion detection also has some limitations. For example, the network flows represent a snapshot of summarized network traffic at a specific time. Therefore, it might be more difficult to distinguish some attack types [37].

3.4. Central Security Assistant (CSA)

The CSA aims to provide services that collaborate with the performance of the activities by the HSUs. The CSA can be provided by *Internet Service Providers* (ISPs) or by service providers interested in providing an adequate structure for the management of security in residential environments.

The flexibility in CSA positioning allows the architecture to be employed in the smart home context and in different IoT environments such as smart grid, healthcare, and others. However, it is necessary to evaluate, among other things, the number of devices deployed on the network and the traffic generated by them to define the hardware resources needed to meet the environment. One of the benefits of installing CSA on an ISP or cloud is the ability to sell security services to customers as models for detecting threats.

Figure 5 presents the data flow performed by the CSA, in which the notification process receives notifications provided by the HSU and saves them as notifications in the notification store. The CSA also runs the statistics process, which is responsible for creating metrics based on notifications received by the HSU. In addition to notifications, the CSA maintains the anomaly detection models used by HSUs.

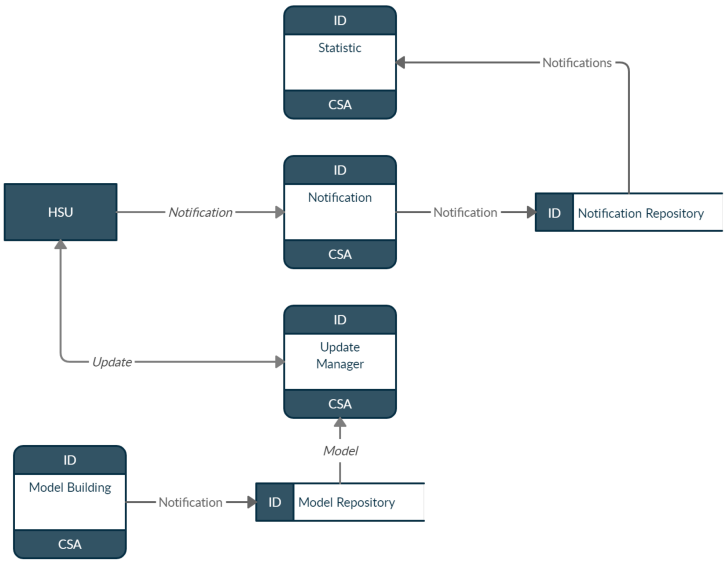

Figure 5. Communication between CSA components.

Thus, the model building process is responsible for building these models and saving them in the repository. The update manager process can perform queries to check the models for updates and send messages notifying the HSU if there are any updates.

The architecture's conceptual design aims to detect anomalies, allowing for the management of heterogeneous IoT devices deployed in smart home environments, and also focuses on the analysis of activities generated in the network. Therefore, a multi-layer structure was adopted that allows for the independent development of the components.

3.5. FamilyGuard Layers

FamilyGuard is organized into four layers: Device layer, Network layer, Detection layer, and Management layer, depicted in **Figure 6**. The HSU and SDN controller operate at the Network and Detection layers. The Management layer hosts (i) the CSA, which helps different HSUs to perform their tasks, and (ii) applications, which help to control and configure local services that exist in residential environments. The structures of each layer are described below:

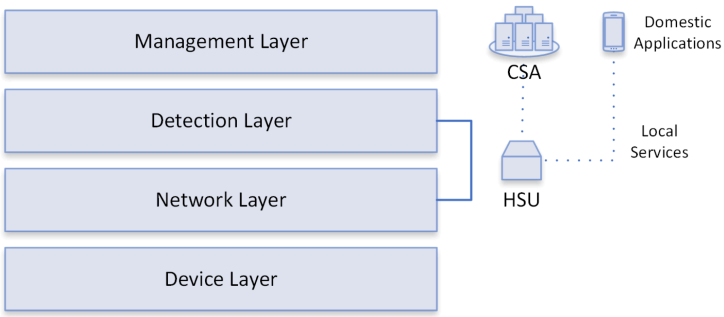

Figure 6. FamilyGuard layers and their relationship with HSU and CSA.

- Device Layer: represents all devices that can communicate in home environments, including laptops, smartphones, and smart devices such as sensors (temperature and presence) and actuators (light switches). There are several smart devices on the market, created by different manufacturers; therefore, residential environments are heterogeneous and complex for risk and threat management.

- Network Layer: has the ability to handle multiple protocols and receive/transmit data through the Devices layer, so that data packets are transferred over the data link, such as Wi-Fi, Ethernet, Wireless Sensor Network (WSN), and Machine- to-Machine (M2M).

- Detection Layer: performs anomaly detection (primary function) through well-defined services, from network traffic reception to notification, for layer management, by classifying a given flow as anomalous.
- Management Layer: is responsible for monitoring and controlling the settings of the residential environment through CSA and home control apps; the CSA collaborates so that HSUs can perform their functions through services that are essential for the functioning of the environment.

References

1. Dacier, M.C.; Konig, H.; Cwalinski, R.; Kargl, F.; Dietrich, S. Security Challenges and Opportunities of Software-Defined Networking. IEEE Secur. Privacy 2017, 15, 96–100.
2. Kolias, C.; Kambourakis, G.; Stavrou, A.; Voas, J. DDoS in the IoT: Mirai and other botnets. Computer 2017, 50, 80–84.
3. Roman, R.; Lopez, J.; Mambo, M. Mobile edge computing, Fog et al.: A survey and analysis of security threats and chal lenges. Future Gen. Comput. Syst. 2018, 78, 680–698.
4. Conti, M.; Dehghantanha, A.; Franke, K.; Watson, S. Internet of Things security and forensics: Challenges and opportu nities. Future Gen. Comput. Syst. 2018, 78, 544–546.
5. Kalkan, K.; Zeadally, S. Securing Internet of Things (IoT) with Software Defined Networking (SDN). IEEE Commun. Ma g. 2017, 56, 186–192.
6. Zhou, W.; Zhang, Y.; Liu, P. The Effect of IoT New Features on Security and Privacy: New Threats, Existing Solutions, a nd Challenges Yet to Be Solved. IEEE Internet Things J. 2018, 6, 1606–1616.
7. Augusto-Gonzalez, J.; Collen, A.; Evangelatos, S.; Anagnostopoulos, M.; Spathoulas, G.; Giannoutakis, K.M.; Votis, K.; Tzovaras, D.; Genge, B.; Gelenbe, E.; et al. From internet of threats to internet of things: A cyber security architecture f or smart homes. In Proceedings of the IEEE International Workshop on Computer Aided Modeling and Design of Com munication Links and Networks, CAMAD, Limassol, Cyprus, 11–13 September 2019; Institute of Electrical and Electron ics Engineers Inc.: Limassol, Cyprus, 2019; Volume 2019.
8. Sharma, P.K.; Park, J.H.; Jeong, Y.S.; Park, J.H. SHSec: SDN based Secure Smart Home Network Architecture for Inte rnet of Things. Mobile Netw. Appl. 2019, 24, 913–924.
9. Alves, A.R.; Moura, H.D.; Borges, J.R.; Mota, V.F.; Cantelli, L.H.; Macedo, D.F.; Vieira, M.A. HomeNetRescue: An SDN service for troubleshooting home networks. In Proceedings of the IEEE/IFIP Network Operations and Management Sy mposium: Cognitive Management in a Cyber World, NOMS 2018, Taipei, Taiwan, 23–27 April 2018; Institute of Electric al and Electronics Engineers Inc.: Taipei, Taiwan, 2018; pp. 1–7.
10. Ammi, M.; Alarabi, S.; Benkhelifa, E. Customized blockchain-based architecture for secure smart home for lightweight I oT. Inf. Process. Manag. 2021, 58, 102482.
11. Mascarenhas, C.; Prasad, R.; Borges, P.; Syed, S.F. Project Urban Patrol: Building an Attack Resilient Smart Home Arc hitecture. In Proceedings of the 2021 International Conference on Nascent Technologies in Engineering, ICNET 2021-P roceedings, NaviMumbai, India, 15–16 January 2021.
12. Ameer, S.; Benson, J.; Sandhu, R. The EGRBAC Model for Smart Home IoT. In Proceedings of the 2020 IEEE 21st Int ernational Conference on Information Reuse and Integration for Data Science, IRI 2020, Las Vegas, NV, USA, 11–13 A ugust 2020; pp. 457–462.
13. Kim, J.T.S. Analyses of Open Security Issues for Smart Home and Sensor Network Based on Internet of Things. IoT Ap pl. Comput. 2022, 179–196.
14. Lee, C.; Zappaterra, L.; Choi, K.; Choi, H.A. Securing smart home: Technologies, security challenges, and security requ irements. In Proceedings of the 2014 IEEE Conference on Communications and Network Security, San Francisco, CA, USA, 29–31 October 2014; pp. 67–72.
15. Gram-Hanssen, K.; Darby, S.J. “Home is where the smart is”? Evaluating smart home research and approaches agains t the concept of home. Energy Res. Soc. Sci. 2018, 37, 94–101.
16. Darby, S.J. Smart technology in the home: Time for more clarity. Build. Res. Inf. 2017, 46, 140–147.
17. Das, S.K.; Cook, D.J. Designing Smart Environments: A Paradigm Based on Learning and Prediction. In Lecture Notes in Computer Science (including subseries Lecture Notes in Artificial Intelligence and Lecture Notes in Bioinformatics); S pringer: Berlin/Heidelberg, Germany, 2005; Volume 3776 LNCS, pp. 80–90.
18. Xia, F.; Yang, L.T.; Wang, L.; Vinel, A. Internet of things. Int. J. Commun. Syst. 2012, 25, 1101–1102.

19. International Telecommunication Union, Telecommunication Standardization Sector (ITU-T), O. ITU-T Y.4000/Y.2060. A vailable online: https://www.itu.int/rec/T-REC-Y.2060-201206-I (accessed on 1 February 2022).
20. Almusaylim, Z.A.; Zaman, N. A review on smart home present state and challenges: Linked to context-awareness inter net of things (IoT). Wirel. Netw. 2018, 25, 3193–3204.
21. Chan, M.; Estève, D.; Escriba, C.; Campo, E. A review of smart homes—Present state and future challenges. Comput. Methods Programs Biomed. 2008, 91, 55–81.
22. Lin, H.; Bergmann, N. IoT Privacy and Security Challenges for Smart Home Environments. Information 2016, 7, 44.
23. Bugeja, J.; Jacobsson, A.; Davidsson, P. On Privacy and Security Challenges in Smart Connected Homes. In Proceedi ngs of the 2016 European Intelligence and Security Informatics Conference (EISIC), Uppsala, Sweden, 17–19 August 2016; pp. 172–175.
24. Chakrabarty, S.; Engels, D.W. A secure IoT architecture for Smart Cities. In Proceedings of the 2016 13th IEEE Annual Consumer Communications & Networking Conference (CCNC), Las Vegas, NV, USA, 9–12 January 2016; pp. 812–81 3.
25. Cahn, A.; Hoyos, J.; Hulse, M.; Keller, E. Software-defined energy communication networks: From substation automatio n to future smart grids. In Proceedings of the 2013 IEEE International Conference on Smart Grid Communications (Sm artGridComm), Vancouver, BC, Canada, 21–24 October 2013; pp. 558–563.
26. Sivaraman, V.; Gharakheili, H.H.; Vishwanath, A.; Boreli, R.; Mehani, O. Network-level security and privacy control for s mart-home IoT devices. In Proceedings of the 2015 IEEE 11th International Conference on Wireless and Mobile Comp uting, Networking and Communications (WiMob), Abu Dhabi, United Arab Emirates, 19–21 October 2015; pp. 163–167.
27. Correa Chica, J.C.; Imbachi, J.C.; Botero Vega, J.F. Security in SDN: A comprehensive survey. J. Netw. Comput. Appl. 2020, 159, 102595.
28. Sezer, S.; Scott-Hayward, S.; Chouhan, P.K.; Fraser, B.; Lake, D.; Finnegan, J.; Viljoen, N.; Miller, M.; Rao, N. Are we r eady for SDN? Implementation challenges for software-defined networks. IEEE Commun. Mag. 2013, 51, 36–43.
29. Rawat, D.B.; Reddy, S.R. Software defined networking architecture, security and energy efficiency: A survey. IEEE Co mmun. Surv. Tutor. 2017, 19, 325–346.
30. McKeown, N.; Anderson, T.; Balakrishnan, H.; Parulkar, G.; Peterson, L.; Rexford, J.; Shenker, S.; Turner, J. OpenFlow: Enabling Innovation in Campus Networks. SIGCOMM Comput. Commun. Rev. 2008, 38, 69–74.
31. Open Networking Foundation. ONF Overview. 2017. Available online: https://opennetworking.org/software-defined-stan dards/overview/ (accessed on 1 February 2022).
32. Ali, W.; Dustgeer, G.; Awais, M.; Shah, M.A. IoT based smart home: Security challenges, security requirements and sol utions. In Proceedings of the ICAC 2017-2017 23rd IEEE International Conference on Automation and Computing: Add ressing Global Challenges through Automation and Computing, Huddersfield, UK, 7–8 September 2017; Institute of Ele ctrical and Electronics Engineers Inc.: Huddersfield, UK, 2017.
33. Aouini, Z.; Pekar, A. NFStream: A flexible network data analysis framework. Comput. Netw. 2022, 204, 108719.
34. Claise, B. Cisco Systems NetFlow Services Export Version 9; RFC 3954; Internet Engineering Task Force (IETF), 200 4.
35. Giotis, K.; Argyropoulos, C.; Androulidakis, G.; Kalogeras, D.; Maglaris, V. Combining OpenFlow and sFlow for an effect ive and scalable anomaly detection and mitigation mechanism on SDN environments. Comput. Netw. 2014, 62, 122–13 6.
36. Hofstede, R.; Čeleda, P.; Trammell, B.; Drago, I.; Sadre, R.; Sperotto, A.; Pras, A. Flow monitoring explained: From pac ket capture to data analysis with netflow and ipfix. IEEE Commun. Surv. Tutor. 2014, 16, 2037–2064.
37. Umer, M.F.; Sher, M.; Bi, Y. Flow-based intrusion detection: Techniques and challenges. Comput. Secur. 2017, 70, 238 –254.