Neural Computing and Applications (2020) 32:16119–16133
https://doi.org/10.1007/s00521-020-04772-3

S.I. : APPLYING ARTIFICIAL INTELLIGENCE TO THE INTERNET OF THINGS

AI4SAFE-IoT: an AI-powered secure architecture for edge layer of Internet of things

Hamed HaddadPajouh[1,2] · Raouf Khayami[1] · Ali Dehghantanha[2] · Kim-Kwang Raymond Choo[3] · Reza M. Parizi[4]

Received: 16 November 2019 / Accepted: 3 February 2020 / Published online: 25 February 2020

Abstract
With the increasing use of the Internet of things (IoT) in diverse domains, security concerns and IoT threats are constantly rising. The computational and memory limitations of IoT devices have resulted in emerging vulnerabilities in most IoT-run environments. Due to the low processing ability, IoT devices are often not capable of running complex defensive mechanisms. Lack of an architecture for a safer IoT environment is referred to as the most important barrier in developing a secure IoT system. In this paper, we propose a secure architecture for IoT edge layer infrastructure, called AI4SAFE-IoT. This architecture is built upon AI-powered security modules at the edge layer for protecting IoT infrastructure. Cyber threat attribution, intelligent web application firewall, cyber threat hunting, and cyber threat intelligence are the main modules proposed in our architecture. The proposed modules detect, attribute, and further identify the stage of an attack life cycle based on the Cyber Kill Chain model. In the proposed architecture, we define each security module and show its functionality against different threats in real-world applications. Moreover, due to the integration of AI security modules in a different layer of AI4SAFE-IoT, each threat in the edge layer will be handled by its corresponding security module delivered by a service. We compared the proposed architecture with the existing models and discussed our architecture independence of the underlying IoT layer and its comparatively low overhead according to delivering security as service for the edge layer of IoT architecture instead of embed implementation. Overall, we evaluated our proposed architecture based on the IoT service management score. The proposed architecture obtained 84.7 out of 100 which is the highest score among peer IoT edge layer security architectures.

Keywords Internet of things · IoT · Service-oriented architecture · Secure architecture · Artificial intelligence · Fog computing · Edge layer

1 Introduction

Internet of things (IoT) is playing an increasingly important role in human–computer environments. In IoT terminology, "things" refer to devices that are communicating with one another or the network gateways [1]. The things can be sensors, smart devices, automobiles, home appliances, or any other Internet-enabled objects [2]. The number of such connected objects in each IoT environment is dramatically increasing. However, the fast adoption of IoT in different domains has led to a broad range of attack surface and increasing security concerns [3–5]. So far different general architectures [6–9] have been defined for IoT environments

✉ Raouf Khayami
Khayami@sutech.ac.ir

Hamed HaddadPajouh
hp@sutech.ac.ir; hhaddadp@uoguelph.ca

Ali Dehghantanha
adehghan@uoguelph.ca

Kim-Kwang Raymond Choo
raymond.choo@utsa.edu

Reza M. Parizi
rparizi1@kennesaw.edu

[1] Shiraz University of Technology, Shiraz, Iran

[2] University of Guelph, Guelph, ON, Canada

[3] University of Texas at San Antonio, San Antonio, TX, USA

[4] Kennesaw State University, Marietta, GA, USA

operations. The proposed architectures are defined by a layered-based structure, and each layer has specific functionality and includes different standards and protocols. These layers mainly include: application layer, network layer, and edge layer (a.k.a perceptional layer) [10].

Application layer Although there is no specific comprehensive standard exists for the IoT application layer, this layer can be structured in a flexible way upon the characteristics of the services to which it offers. For example, the applications of the IoT in smart cities and home [11, 12], smart grids [13, 14], health care [15], or intelligent transportation like autonomous vehicles [16–20]. The application layer usually acts as a middleware [21], a communication protocol, and cloud computing for services support; thus, security concerns would be different based on the hosting environment and the nature of the applications. As can be seen in Fig. 1, the application layer has different components in which the operation of each depends on deliverable services in the application layer [22]. For instance, in health care for medical records retrieval special application programming interface (API) or special applications as binary in client and server side are needed [23, 24].

Network layer This layer accounts for transferring information among layers and gives full access to the edge layer via different protocols and standards like IEEE802.1× [25], global positioning system (GPS), and near-field communication (NFC). As can be seen in Fig. 2, this layer also includes cloud computing as a back-end infrastructure, mobile devices, and the Internet [26, 27].

Edge layer Edge layer or device level plays a vital role in the IoT environment from an end-user perspective [28]. In this layer, end-users interact with IoT devices like sensors, smart meters, or IoT edge layer data center implemented on a gateway which coordinates tasks.

The cloud-edge devices control and transmit data to back-end cloud infrastructure. Each device in this layer has a connection to other layers of IoT architecture according to its functionality. For example, a gateway (coordinator) device at the bottom of the architecture runs its applications on its operating system (OS) and creates an appropriate channel for transmitting data through network layer and edge computing functionalities for delivering the defined tasks. The edge layer devices have a limited computational capability for doing their defined tasks like information validation and verification and nodes data aggregation processes. Also, these devices have their own OS [29], such as RIOT [30] and Tiny OS [31], with the ability to host edge-level applications that install on such OSs.

From a security perspective, each layer that has relation to edge devices can be targeted by the attackers to exploit vulnerabilities that exist in the protocols and standards [32]. In the application layer, most of the threats occur in HTTP protocol which includes API calling between end-user and cloud infrastructure. In the network layer, denial of service (DoS) and Bluejacking are the most common threats which occur in protocols and standards of the IoT environment [33]. In the cloud computing components, IoT users face a wide range of security threats and vulnerabilities such as misconfiguration, identity management, data access control, infrastructure security, and data privacy [34]. In the edge layer, due to the predictable location of the connected devices in the IoT infrastructures and the limitation in computational resources, the user faces a myriad of threats. The threats include node sabotaging, node failure, node disconnection, offline information gathering, false node message corruption, exhaustion, Sybil, jamming, tampering, and collisions [33].

According to the secure list report, in Q4 2018, more than one million infections from 70 servers infiltrate cloud-edge gateways devices (edge layer gateways) that include

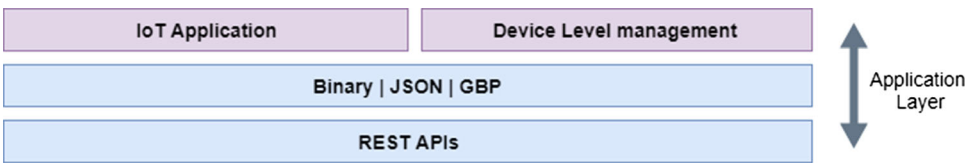

Fig. 1 Application layer components in IoT architecture

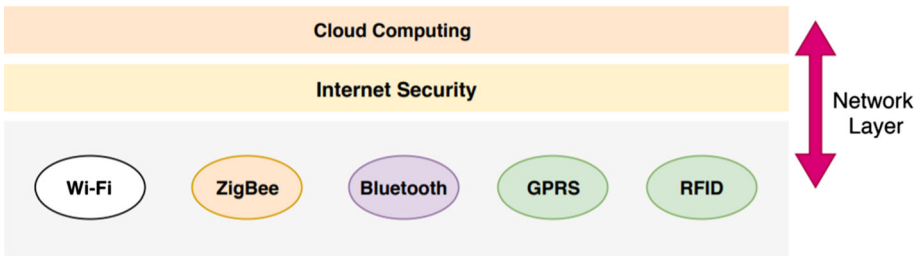

Fig. 2 Network layer components in IoT architectures

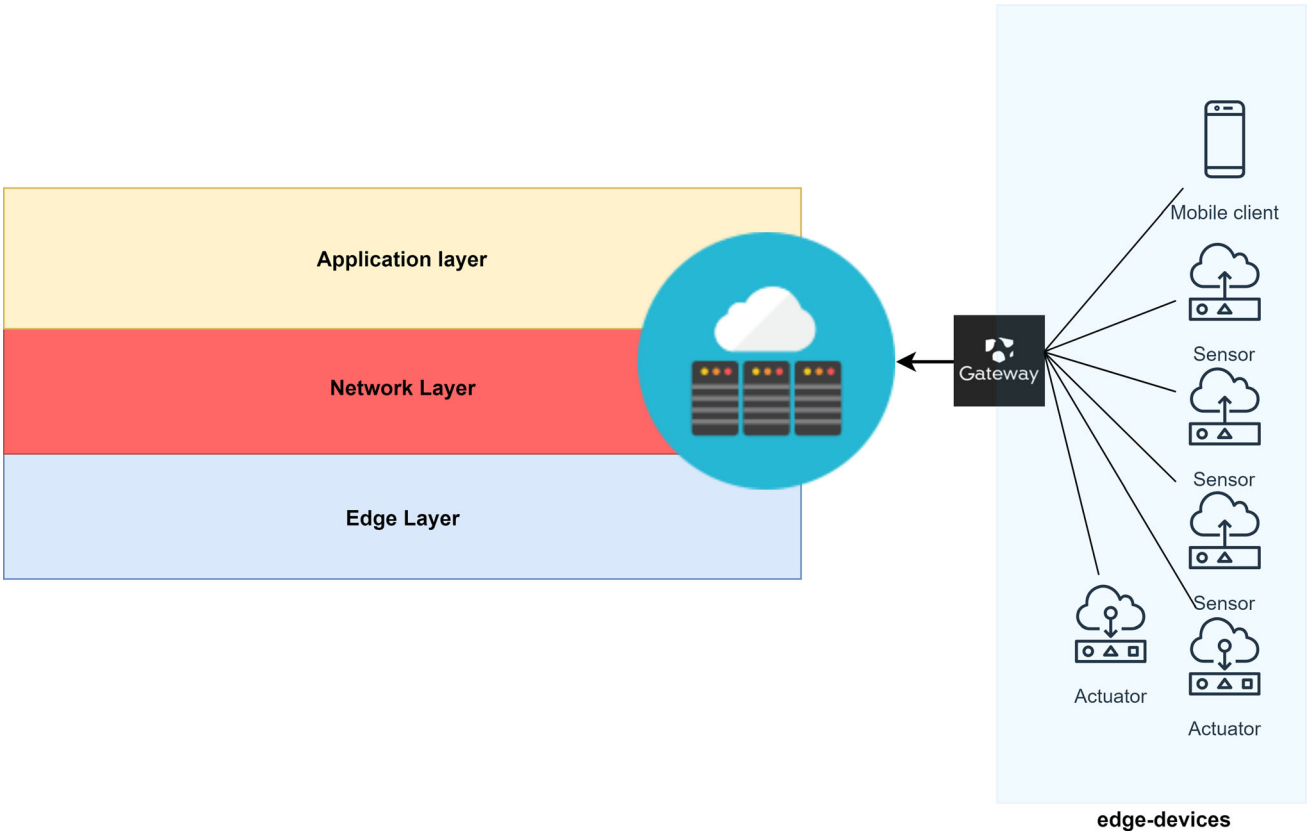

Fig. 3 Edge (device) layer components in IoT architecture

32- and 64-bit ARM, x86, x86_64, MIPS, MIPSEL, and PowerPC processors architecture [35]. Attackers target this segment of the IoT environment more than the other parts. As shown in Fig. 3, devices usually are connected to one or more gateways named "cloud-edge" devices. Cloud-edge devices are also facing cyber threats frequently such as misconfiguration, hacking, signal lost, distributed denial of service, war dialing, protocol tunneling, man-in-the-middle attack, interruption interception, and modification fabrication [9]. Since there is a wide range of threats exist in each layer of IoT architecture that has interaction with edge-devices, each layer needs a dynamic secure mechanism to tackle these threats. In the edge layer of IoT infrastructure due to the limitation in computational resources of nodes and public accessibility, more vulnerabilities are existing in comparison with other layers. Therefore, having a robust and secure architecture for this layer would help to tackle the existing threats and mitigate attacks propagation in other layers. Since the number of threats is increasing, heuristic methods will not be sufficient anymore. As a result, the current structures need a robust mechanism to face unseen threats as well as well-known threats in IoT environments.

AI-based mechanisms due to their capabilities in learning from environments are suitable to tackle unseen threats. Up to this time, several AI-based methods are proposed for cyber threat hunting in the edge layer of the IoT environment and obtained promising results in dealing with unseen threats [33, 36]. The efforts show that having a security architecture based on AI engines can help to detect, locate and attribute the existing threats that occur in the edge layer. By having a defensive mechanism in the edge layer, we can mitigate the propagation of attacks or even prevent their infiltration into other layers.

The main contributions of this paper include:

- Propose a secure architecture for the edge layer of IoT environments based on multiple AI-powered components.
- Design AI-engines applied in defense components based on the service-oriented architecture for securing the edge layer of IoT environments.
- Evaluate the proposed IoT secure architecture based on a representational evaluation metric named IoT service management score.

The rest of this paper is structured as follows. Section 2 reviews the literature related to architectures in IoT environments. In Sect. 3, we define our proposed secure architecture for the edge layer of the IoT environment. Section 4 includes a comparison of our security architecture and other peer architectures for the edge layer of the

IoT environment. Finally, we will discuss our architecture advantage and disadvantage and provide concluding remarks in Sect. 5.

2 Related work

With increasing the usage of IoT devices in different applications, new vulnerabilities and threats have emerged in the edge layer of IoT environments. To deal with the existing threats, a wide range of the models and frameworks for securing the edge layer of the IoT architecture have been proposed [36–38]. Most of the proposed architectures are only able to address specific security challenges between edge layer devices and one of the working layer of IoT architecture, following a heuristic method.

Moosavi et al. [39] proposed a safe IoT architecture for edge layer of healthcare environment based on DTLS protocol which is a basic form of IPSec solution for IoT. Their architecture benefits from curve digital signature algorithm and the elliptic curve Diffie–Hellman for key exchange to provide data integrity. Their model can address the authentication of edge layer devices in the healthcare application layer of the IoT environment. As middleware architecture of IoT, Tiburski et al. [64] defined a service-oriented architecture (SOA) for smart transportation applications. The proposed SOA provides data privacy and confidentiality between edge layer devices and the application layer of IoT architecture. Although this architecture provides a service-based security architecture, its working domain confined to the application layer. Vučinić et al. [41] proposed object security architecture for IoT end-to-end devices in the edge layer of the smart grid domain. The architecture protects edge layer nodes from replay attacks [42] which occur during the interaction between an edge layer device and the network layer of the IoT architecture, by providing end-to-end encryption, and eavesdropping during the communication. Giuliano et al. [23] proposed a secure conceptual framework for the business application of the IoT with two different architectures. The first secure architecture is three layered for the application layer of IoT, and the second one is a five-layered autonomic architecture for addressing traditional security issues in IoT systems. For the edge layer security, several architectures have been proposed which rely on a specific installed security mechanism on the device rather than providing a higher aspect of architecture. Pinto et al. [43] proposed an architecture called IIoTEED for ARM-based devices in the edge layer of the IoT environment. Their architecture provides a trust zone for devices to communicate which other. This architecture is highly device dependable which causes overhead for a wide range of edge-level devices and for implementations need external hardware resources for providing a trust zone for edge layer devices. In addition, this architecture is not able to address end-to-end security for working devices. Jang et al. proposed SeCRet [44] a secure architecture that established a secure channel between device connection. This architecture also provides isolation to maintain the integrity of data transmission by a session key. This architecture is resource demanded and not suitable for the resource-constrained environment. Dai et al. [45] proposed a secure execution environment for cloud-edge devices. Their architecture takes advantage of virtualization and trusted computing for providing a secure environment for operating system edge layer devices. This architecture performance depends on the hardware power of edge layer devices and able to address just a few specific threats. Guen et al. [63] proposed a secure architecture named TrustShadow for ARM-based processor devices of the IoT environment. Their architecture benefits from TrustZone technology, and it partitions computational resources in secure Zone.

3 Proposed architecture

In this section, we define the concept of our proposed secure architecture in a holistic and detail view for the edge layer of the IoT environment. The main architecture is built upon a three-layer structure covered by a pervasive security layer for the edge layer. The reasons why a three-layer structure was adopted are twofold: (a) the three-layer structure's architecture has been highly considered safer than other existing architecture (i.e., a four-layer structure) in recent works; (b) since there is no specific standard for IoT architecture, there is not a union security structure for each architectural layer; therefore, we used a three-layer architecture as our base structure. In the proposed security layer, we define different security modules to provide a resilient and robust security mechanism into the edge-level of the IoT environment. Figure 4 shows the conceptual view of the proposed architecture with the different security modules for each layer of the IoT architecture to communicate with edge layer devices. The security modules and their inner workings are explained in detail in the following subsections.

3.1 Application layer security

In the application layer, edge devices usually deal with IoT devices applications via web services. We propose two different secure modules for maintaining a secure interaction between devices and service provider as follows:

CoAP-DTLS CoAP [46] functions as a sort of HTTP for constrained devices, enabling such component level

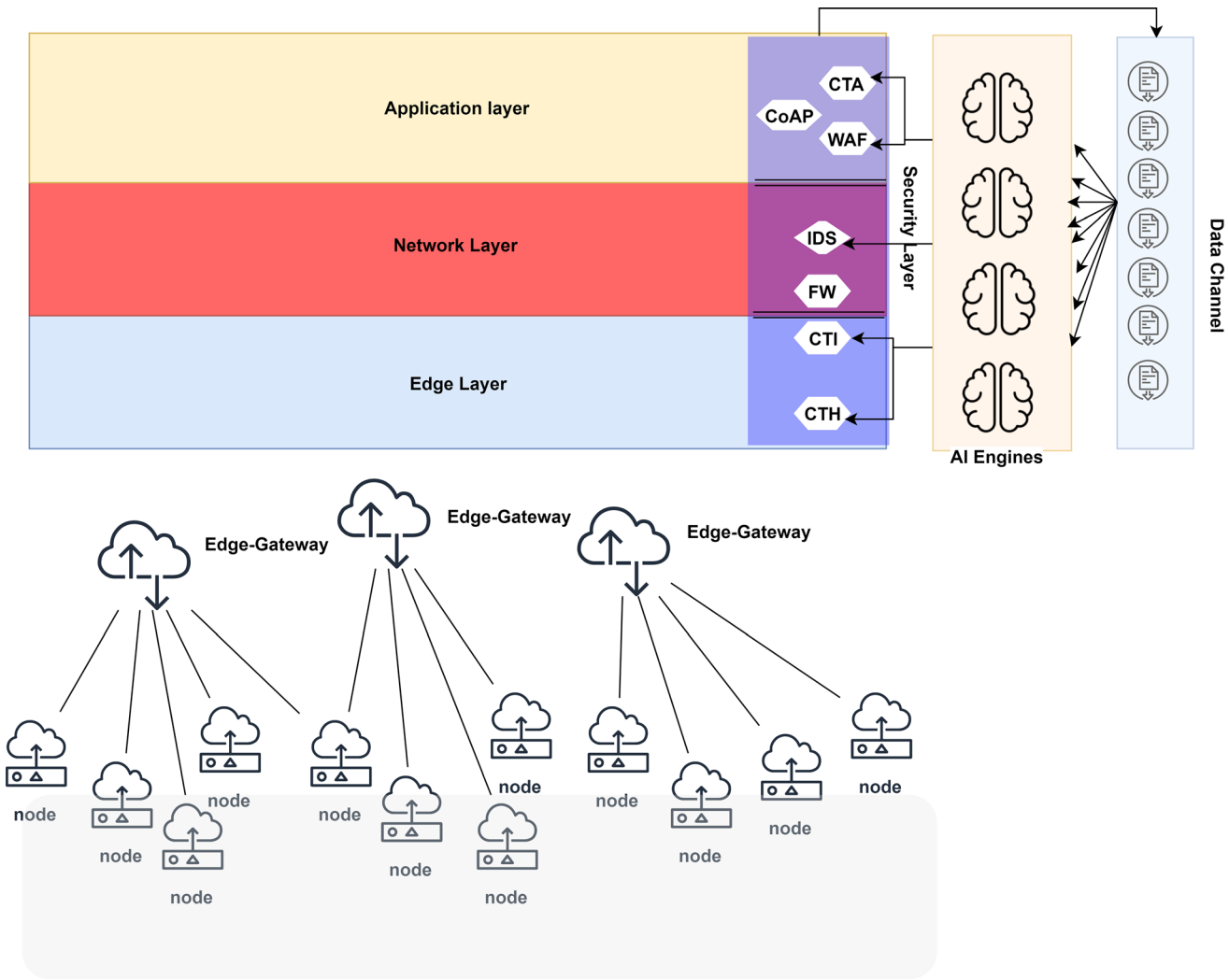

Fig. 4 AI4SAFE-IoT architecture model

equipment as sensors or actuators to communicate on the IoT, being controlled and passing along their data as part of a system. The protocol is designed for reliability in low bandwidth and high congestion through its low power draw and low network overhead. CoAP can continue to work where TCP-based protocols such as message queuing telemetry transport (MQTT) fail to complete a handshake [47]. CoAP is a plaintext protocol like HTTP but it operates under UDP TLS and can be used by default. The encryption is most commonly accomplished using datagram transport layer security (DTLS). Algorithm 1 shows the CoAP procedure for establishing a secure connection to maintain confidentiality and integrity between two hosts (edge-devices) in the application layer.

Intelligent Web Application Firewall (IWAF) Most of the edge clients of IoT environments are served by HTTP protocol through REST API. For detecting anomalies in the given services' request through HTTP protocol, an intelligent WAF [48] operates as Algorithm 2 in the security architecture. This mechanism preserves edge layer devices application from attackers by defining a set of rules which are optimized by the AI engine for the edge devices web applications.

Cyber threat Attribution (CTA): One of the main challenges in the application layer is dealing with the source of threats and choosing the best action corresponds to a threat. Knowing tactics, techniques, and procedures (TTPs) of the attacker in this layer can help to find the source of attacks and make optimum decisions based on the character of the attacker campaign. CTA module in the application layer has the responsibility of attributing the malicious activity that resides on edge layer devices by a profile matching engine to its original malicious actor and recommends the optimum decision (course of the action) against the threat/attack—see Algorithm 3.

Algorithm 1 CoAP-DTLS procedure for securing between hosts communication

```
function CoAP-DTLS(host_A, host_B)
    host_A → sendHello(host_B);
    host_B→verifyHelo(host_A);
    host_A→generateCookie();
    host_B→serverHello(hostA):keyShare,Certificate;
    host_B→certificateRequest(host_A);
    host_A→sendCertificate(host_B);
    host_A→certificateVerify(certificate_B);
    host_A→finishHanshake();
    hostB→ finishHanshake();
    if certificate_A && certificate_B then
        host_A→sendRequest(host_B, payload);
        host_B→sendRespond(host_A, payload);
    end if
end function
```

3.2 Network layer security

In the network layer, most of the threats are included in TCP/IP stack protocol. However, in an industrial environment like Smart [49], Operational Technology (OT) protocols like Mudbus are also work in the network layer. Therefore, for having a security mechanism in this layer similar to typical TCP/IP protocols, following modules are provided in AI4SAFE-IoT:

Network-based Firewall (FW) Since network firewalls are primitive defense mechanisms in network-based structure, at the network layer of the IoT architecture providing a rule-based mechanism for blocking suspicious traffic is essential. Most of the edge devices in the network layer of IoT use the same TCP/IP protocol; thus in AI4SAFE architecture, we adopted the same defensive mechanism from TCP/IP protocol to block malicious requests in the network layer.

Intrusion Detection System (IDS) Network-based IDS [50] in terms of active defense mechanisms is widely used in the IoT environment [51–53]. Thus, this module also included in the security layer of the proposed architecture. As can be seen in Table 1, the IDS mechanism be able to protect the IoT environment against most of the threats that occur in the network layer of edge layer clients. In the proposed architecture, the IDS module benefits from a trained engine under normal and malicious traffics profile. The trained AI engine can use the existing profiles from current TCP/IP protocols or current traffic flows in the network layer of edge layer devices for training purpose. In addition, the edge layer devices may face connectivity issues during their operations due to security challenges in their network layer. As shown in Table 1, the common critical network security challenges in IoT devices include denial of service threats which are difficult to address by primitive security modules like rule-based firewalls or even signature-based IDSs [51]. Contrary, by intelligence modules like AI-powered threat hunting and threat attribution, most challenging threats can be mitigated. With that in mind, the proposed AI-powered modules in the network and edge layer of the proposed architecture build profiles from devices' behavior and will be able to detect, prevent and attribute malicious behaviors.

Algorithm 2 AI-Powered Web Application Firewall procedure for blocking unauthorized request

```
function IWAF(request, sender, receiver)
    features := getFeature(request);
    predict := evaluate(trainedModel,features);
    if predict == malicious then
        block(request);
        sendToBlacklist(sender);
    else
        if authorize(sender) && authorize(receiver) then
            receiver→sendRequest(sender);
            sender→sendRespond(receiver);
        end if
    end if
end function
```

Algorithm 3 AI-Powered Cyber Threat Attribution procedure for finding threat intent in the application layer of the IoT environment.

```
1: function CTA(observable, node)
2:     features := getFeature(observable);
3:     predict := evaluate(trainedModel,features);
4:     if predict == malicious then
5:         isolate(node);
6:         sendToBlacklist(node);
7:         actor := attribute(predict);
8:     end if
       return getCourseofActions(actor);
9: end function
```

3.3 Edge layer security

Although there are different standards exist for (edge layer) end-point devices, most of the threats are targeting edge gateways due to higher damages that they can cause to IoT environment. For example, Mirai attack in 2016 [54, 55] caused a huge downtime in a wide range of IoT infrastructure. In the edge layer, each end-point device needs to send its captured data to the IoT cloud back-end structure. There are three different ways for establishing a connection among end-point devices to the IoT back-end structure: (a) direct connection to cloud gateways, (b) direct connection through field gateways, and (c) indirection connection via virtual private network connection [56].

Cyber Threat Hunting (CTH) In AI4SAFE architecture, we focus on AI-powered security modules for IoT gateways due to their critical role in the environment. Regardless of the type of connection among devices and their coordinators/gateways, the proposed modules will act based on the working agent installed on gateways/devices. Algorithm 4 shows the threat hunting module for edge-level gateway devices in the AI4SAFE architecture. In CTH module, threats can emerge in different aspects. For example, a malware threat or suspicious network traffic (i.e., a pcap file) will be represented as a vector of features in sequential or discrete manners which in turn will be preprocessed. In other words, CTH assigns a label as an anomaly or normal for suspicious behavior. Anomalies need further analysis for finding indicators of compromise (IoCs) on the current set of nodes. If any evidence found on each device, two actions should be done. First, the evidence will be passed to the CTA module as observable to find out what course of action it must apply. The next action will be to call the cyber threat intelligence (CTI) module for determining the stage of an attack that compromised the node.

Cyber Threat Intelligence (CTI) Each threat has a life cycle according to Cyber Kill Chain (CKC) threat modeling taxonomy [57–60]. Therefore, determining the current threat stage after the detection of the IoC in the compromised node helps to find out an optimum decision on the current stage of threat. The CTI module in the proposed architecture acts as a complementary component to designate the campaign behind the attack and gives a bright insight into the type and severity of the threat which is occurrent in the IoT environment. Algorithm 5 illustrates the procedure of the CTI module in the AI4SAFE.

With regard to the limitation in computational resources in IoT devices and memory resources in edge layer gateways, designing defensive mechanisms in a lightweight mode is inevitable. There are two approaches that exist to implement a security module in edge layer of IoT. The first approach takes service from the application layer of architecture, which means implementing the AI engine on top of the architecture. In the second approach, the AI engine of the module is installed embedded on the gateways as one of the gateway functionalities.

Table 1 Network layer attack type and impact on IoT environments

Attack type	Impacts on IoT network layer	Comments on the detection mechanism
Sinkhole	Large traffic flows through attacker node	Parent fail-over would detect the attack
Denial of service (DoS)	Make resources unavailable to Intended user	Detecting the source of the false node would mitigate the attack
Rank	Packet delay, delivery ratio, and generation of unoptimized path and loop	Detecting the suspicious delivery time would detect the attack
Local repair	Control overhead disrupt routing and traffic flow	Monitoring the traffic flow for anomalies would detect the attack

One of the main components of the AI4SAFE architecture which has high interaction is its AI engines repository. In this component regardless of the application of IoT structure, plenty of trained models exist. Each model has been trained with one type of data. For example, if the edge layer faces malicious binary executable data type, the gateway CTH, module engine must connect to the trained models with an opcode, bytecode, and system calls. We separated this section from the security module to emphasize on the usability of the engines for different layers of the architecture. For operating independently of engines, we proposed a data channel component as well. This component will be responsible for capturing, normalizing and transforming precept data from the environment and feed the engines for training and testing tasks.

dramatically. This module would act as a traditional preprocessing engine in the traditional machine learning task but would work with different layers of data and making them ready for feeding the nominated engine. To elaborate on the working mechanism of this channel, we provide an overview of the functionality data channel module. Let's assume a suspicious behavior, which is malicious, raises the alarm in the edge layer of an IoT environment. In order to ensure about certain nature of this behavior, CTH, CTI, and CTA modules will be involved to make an appropriated decision about it. For ensuring final decision about the behavior, each AI engine needs a certain type of data to perform its classification tasks. In the meanwhile the threat behavior passes to the data channel, based on the availability of the existing views, include static, dynamic, network and cloud-lever interactions. From there, the corresponding feature set will be generated and will be ready for feeding each layer AI engine. Figure 5 shows the data channel mechanism and its interaction with the AI engine module.

Algorithm 4 AI-Powered Cyber Threat Hunting procedure for finding threat indication in end-point devices of edge layer.

```
function CTH( <a set of Node>, threat, trainedModel )
    for node in nodes do
        features := getFeature(threat, node)
        predict := evaluate(trainedModel,features)
        if predict == anomalies then
            if existIoCin(node) then
                isolate(node);
            end if
        end if
    end for
return CTA(IoC→obeservable);
end function
```

Algorithm 5 AI-Powered Cyber Threat Intelligence procedure for finding the stage of attack in the edge layer of the IoT environment.

```
function CTI(<a set of Node>, threat, trainedModel )
    for node in nodes do
        features := getFeature(threat, node)
        predict := evaluate(trainedModel,features)
        if predict == anomalies then
            if existIoCin(node) then
                isolate(node);
                findcampaign(IoC);
                detemindStageofThreat(threat);
            end if
        end if
    end for
return CTA(IoC→obeservable);
end function
```

3.4 Data channel

In the AI4SAFE architecture, we provide a data channel for processing data collected in the different layers of the architecture for feeding the AI engines. This will reduce decision making procedure on a specific threat

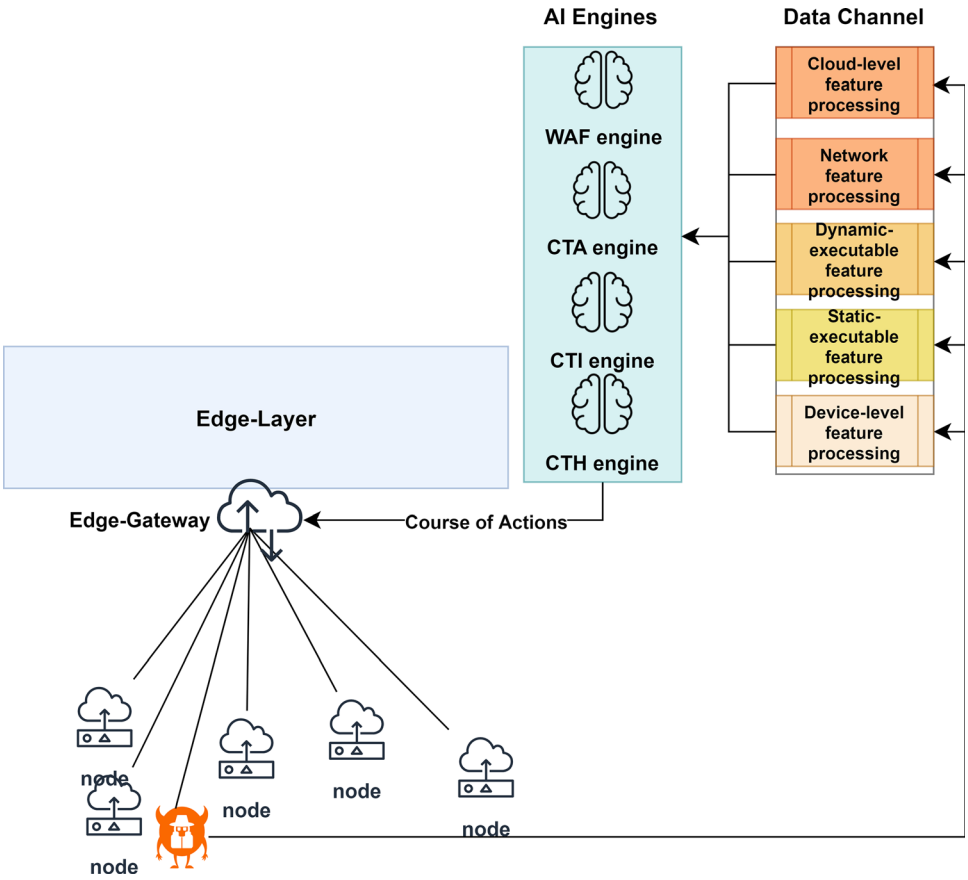

Fig. 5 AI4SAFE-IoT data channel operational mechanism with correspondent AI engines

3.5 Scalability and interoperability

The AI4SAFE architecture is designed within the framework of service-oriented architecture (SOA). As can be seen in Fig. 6, each SOA architecture must handle the business activity and then provide proper protocol for handling business procedures [62]. Each procedure managed orchestration module [61]. Hence, AI4SAFE

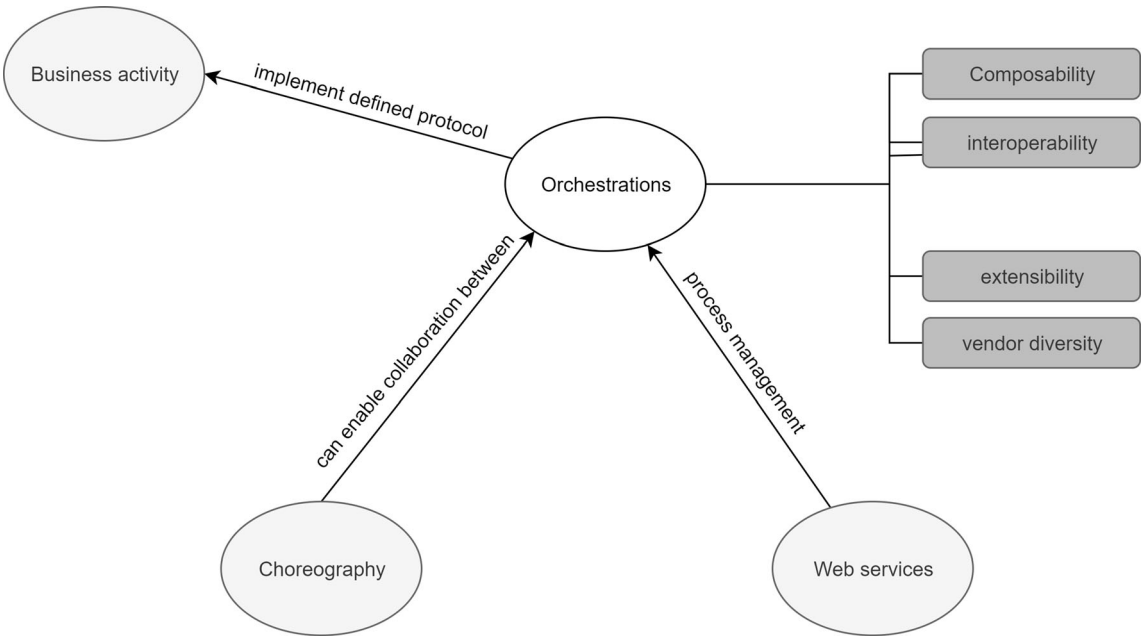

Fig. 6 SOA functionality and service orchestration requirements [61]

architecture can be mapped to the same structure. As we demonstrated in the overall architecture model in Fig. 4, the proposed architecture includes several security modules (services) for keeping the IoT environment safe. Each proposed module preserves each IoT layer from potential intrusions. For example, in the edge layer of AI4SAFE, we proposed three different security modules named CTA, CTH, and CTI; all these modules have a specific agent which can be called by other modules (Algorithms 3 and 4). The AI engines repository (serving as the orchestration module in our work) module acts as a mediator to send appropriate service to requested modules. With that said, we believe that the proposed architecture meets all the SOA requirements in terms of scalability and interoperability.

4 Results and discussion

Most of the edge-level secure architectures mainly rely on end-point devices structure and underpinning hardware of such devices that may well cause high overhead on the resource consumption. For instance in IIoTEED [43], a trust zone-based architecture for isolating edge-level devices beside dual boot for edge-level devices for ensuring integrity is included. SeCRet [44] architecture just uses secure communication between devices based on the session key to sign messages which could cause overhead. TEE [45] and Trust Shadow [63] use secure boot and encryption in edge devices of the IoT environment which could also cause computational overhead for devices due to the installing security function on each device. Although these architectures have demonstrated the ability to secure IoT devices to some degree, they are mainly not feasible to implement on resource-constraint devices due to their computational overhead and hardware dependency. Tiburski et al. [64] architecture is designed for resource-constrained devices based on trust mechanism with embedded methodology, but it cannot deal with new types of (unknown) attacks. Another shortcoming of recent secure architectures is that they are designed for certain types of the existing threats and cannot be easily updated for new threats or even for a new version of a current threat that they were initially designed for it.

AI4SAFE-IoT is an AI-powered architecture that enables easy implementation on cloud-edge devices (gateway) instead of on edge-level devices. This unique feature gives a high privilege to its enabling components to act as active defense mechanisms to prevent threat propagation in the other layers. Since the proposed architecture uses AI engines to detect, locate and attribute threats in a different layer of IoT environment that interacts with edge layer devices, it will be able to tackle unseen threats based on matching their profile of behaviors with each trained engine. The proposed architecture also has lower overhead on the resource consumption of edge devices compared with embedded defensive architecture. In addition, it is defined by a set of security modules that have interoperability functionality in each layer of IoT architecture that interacts with edge devices.

We conducted a particular comparison between the proposed architecture and the three peer secure architectures (discussed earlier in this section) based on service management concept for IoT presented in [65]. The service management of IoT consists of six dimensions named service type, architectural organization, middleware, run time management, security, and application for authorizing different IoT architecture capabilities. Figure 7 illustrates all service management components for IoT.

As it can be seen in Tables 2, 3, 4, 5, 6, and 7, we evaluated each architecture based on each service management aspect that presented in [65]. We reviewed each proposed secure architecture from each IoT service management aspect's features. To this end, we assigned 1 to each feature if it exists in each proposed secure architecture otherwise we assigned 0. Afterward, we defined a score, presented in Eq. 1, for each secure architecture to asses their universality in order to deliver timely service.

$$\text{Service Mgmt. Score} = \frac{1}{n}\sum_{i=1}^{n}\sum_{j=1}^{m} W_i \times f_j \tag{1}$$

where i belongs to each IoT service management aspect and defines as $i = \{a_1, \ldots, a_n\}$, and j belongs to each aspect's feature and it can define in different size. W_i indicates each aspect weight in order to scale up all scores in the same direction. At last, we scaled all scores in [1, 100] interval for comparing each IoT secure architecture service management comprehensiveness. The proposed metric can be used to evaluate all the IoT security architectures that include the basic service delivery requirement in each layer like AI4SAFE. Therefore, if security requirements delivered as a service in the IoT environment, we could measure the service management score to evaluate the scalability of the proposed architecture.

Since the application category in service management is biased to each environment, we skipped this feature for evaluating our architecture to avoid unfairness and obtaining unbiased results. Moreover, AI4SAFE is a security architecture that can be applied in every IoT applications. Therefore, we did not include the application category to asses service management score of the proposed architecture. The application category can be used to evaluate the degree of functionality of the architecture in a specific IoT environment.

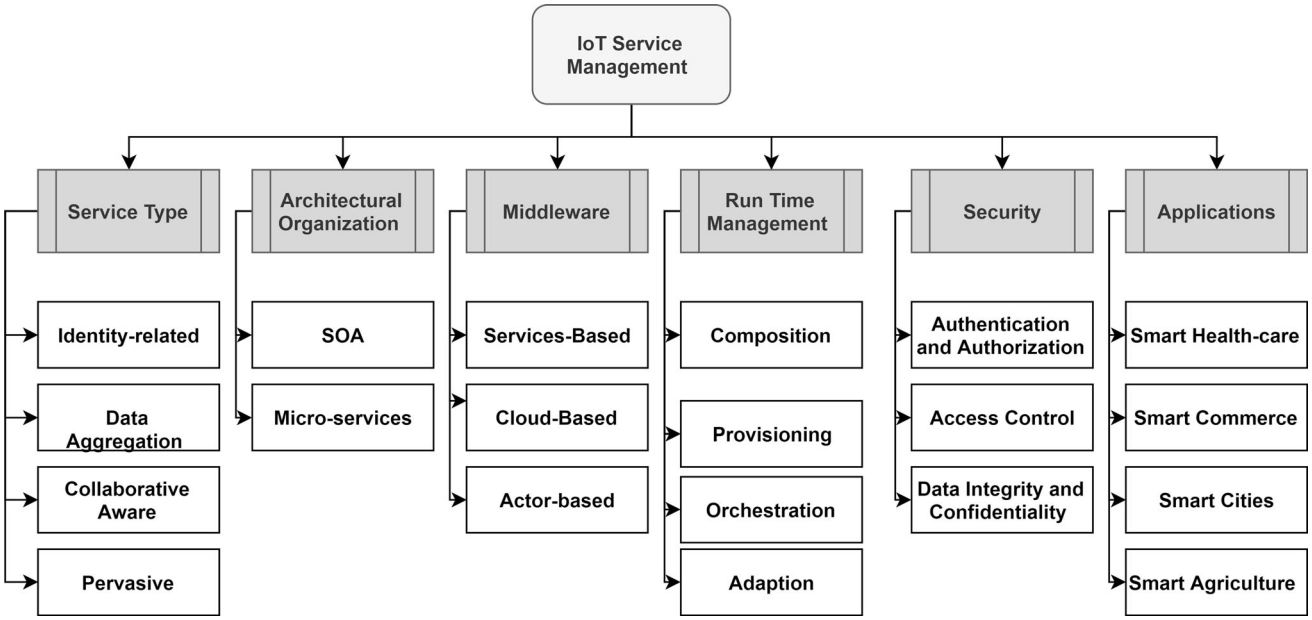

Fig. 7 IoT service management taxonomy [65]

Table 2 Comparison analysis of AI4SAFE-IoT architecture based on service-type aspect

Architecture	Identity related	Data aggregation	Collaborative aware	Pervasive
IIoTEED [43]	1	0	1	1
SeCReT [44]	1	0	1	1
TEE [45]	1	0	1	1
AI4SAFE	1	1	1	1

Table 3 Comparison analysis of AI4SAFE-ioT architecture based on architectural organization aspect

Architecture	SOA	Micro-services
IIoTEED [43]	0	1
SeCReT [44]	0	1
TEE [45]	0	1
AI4SAFE	1	1

Table 4 Comparison analysis of AI4SAFE-IoT architecture based on middleware aspect

Architecture	Service based	Cloud based	Actor based
IIoTEED [43]	0	1	1
SeCReT [44]	0	0	1
TEE [45]	1	1	0
AI4SAFE	1	1	0

Figure 8 shows a comparative analysis of each IoT edge layer security architecture's service management score. As it can be seen from the figure, AI4SAFE-IoT obtained the highest score among the peer architectures due to its modularity with regard to the implementation of multi-purpose AI engines.

4.1 Healthcare environment and AI4SAFE: a case study

In a particular analysis to demonstrate the functionality of the proposed architecture, we chose an IoT application in the healthcare environment. We adopted one of the latest models for IoT which was proposed by Woo et al. [66] as a reliable personal healthcare system. In their proposed system, they defined three main components named management server, MN gateway and end-user (medical staff) devices for delivering healthcare service based on M2M protocols. All three main components can be mapped into AI4SAFE architecture and the security modules that applied in each layer. Figure 9 demonstrates the IoT structure proposed by Woo et al. [66] which is integrated with AI4SAFE security modules. As it shows in the overview of proposed healthcare system architecture, we defined each module in the IoT base architecture as follows:

Table 5 Comparison analysis of AI4SAFE-ioT architecture based on run time management aspect

Architecture	Composition	Provisioning	Orchestration	Adaption
IIoTEED [43]	0	1	0	1
SeCReT [44]	0	1	0	0
TEE [45]	1	1	0	1
AI4SAFE	1	1	1	0

Table 6 Comparison analysis of AI4SAFE-ioT architecture based on security aspect

Architecture	Authentication and authorization	Access control	Data integrity and confidentiality
IIoTEED [43]	0	1	1
SeCReT [44]	1	1	1
TEE [45]	1	1	1
AI4SAFE	1	1	1

Table 7 Comparison analysis of AI4SAFE-IoT architecture based on application aspect

Architecture	Health care	Smart commerce	Smart cities
IIoTEED [43]	0	1	1
SeCReT [44]	0	1	1
TEE [45]	1	1	0
AI4SAFE	0	1	1

Application layer The proposed healthcare system architecture provides a management server for handling requests from edge layer device and end-user request, respectively. This component in health care can be protected by AI4SAFE intelligence module as well as the CoAP protocol.

Network layer We have an M2M server that facilitates and connect edge layer devices to the management server. This component operates as a cloud-gateway in the base architecture. As we demonstrated in the AI4SAFE, we can apply IDS, network-based firewalls, and cyber threat hunting modules.

Edge layer Patients sensor and PDAs are the major edge layer devices in healthcare environments that they are mostly connected to the main application or a cloud-end server by cloud gateway or MN server as we have in the healthcare environment. Therefore, these devices need to be protected individually, while they are connected to an orchestration module. AI4SAFE intelligence security module can address this challenge by monitoring edge layer device's behavior as well as their connectivity.

Although there is a wide range of models and architectures for different IoT environments, as demonstrated by this case study, most of which can be mapped to a three-layer architecture supported by the AI4SAFE.

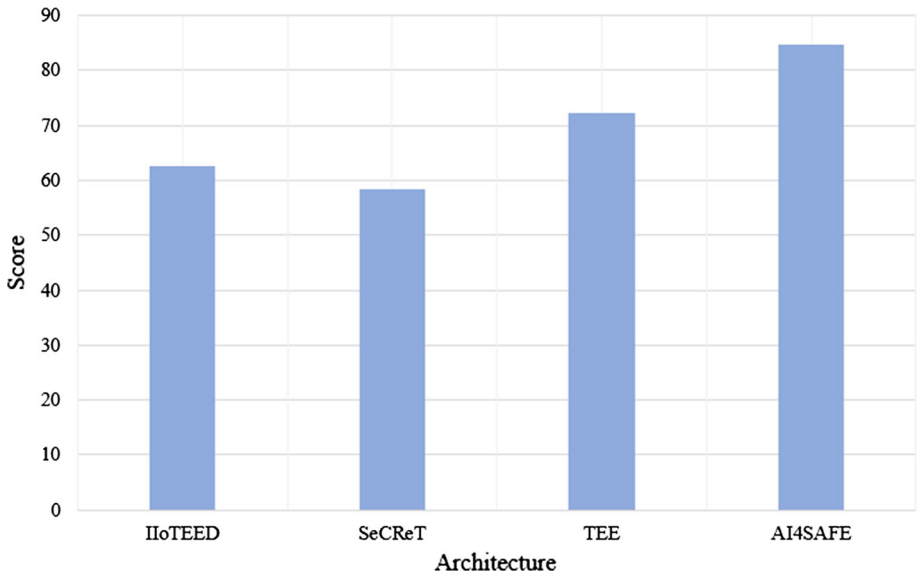

Fig. 8 Comparative analysis of IoT service management score on different IoT edge layer architecture

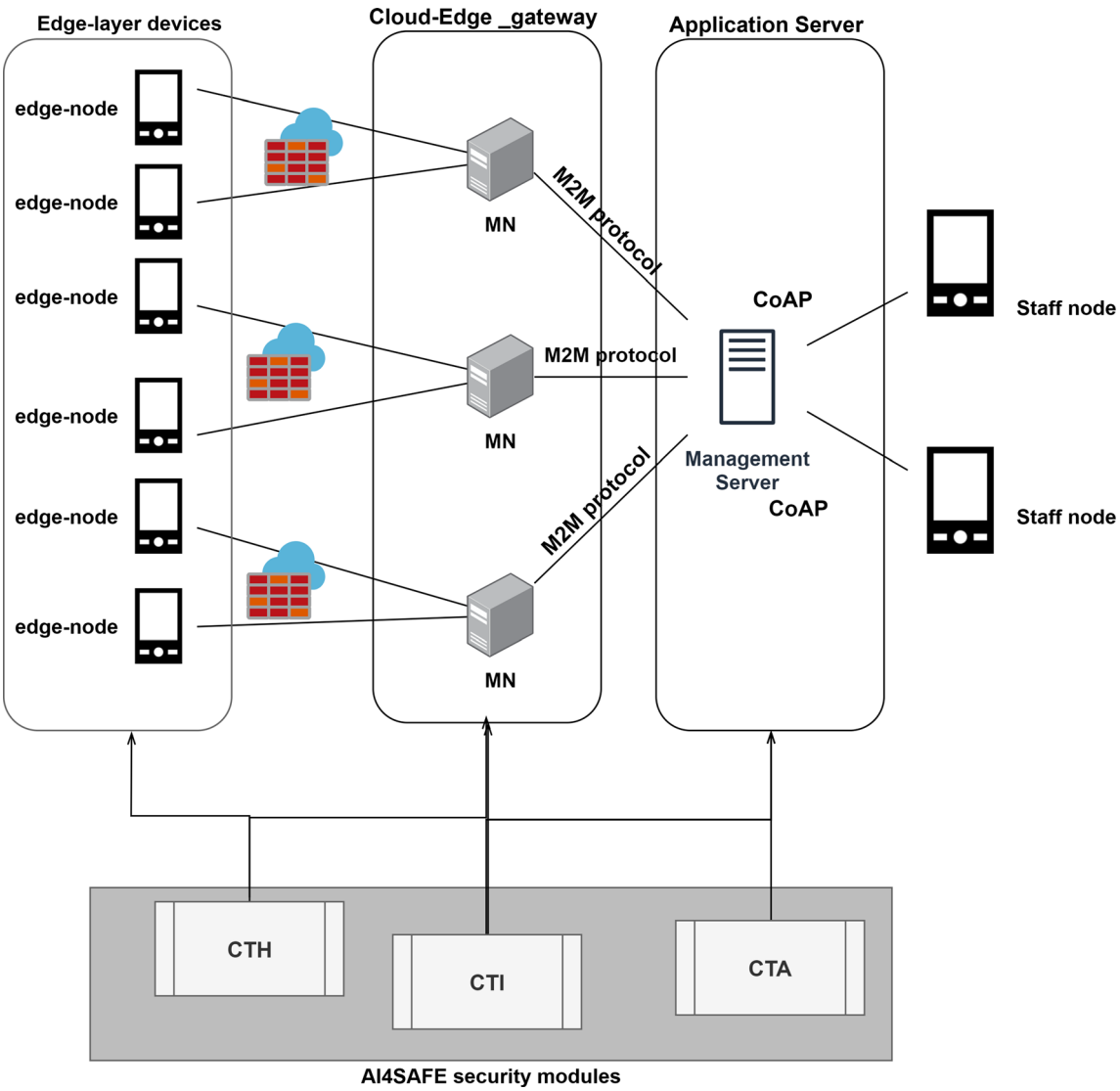

Fig. 9 Healthcare security architecture: AI4SAFE case study overview

5 Conclusion

With the rapid growth in the Internet of things infrastructure and emerging new applications based on its functional architecture, security challenges become more and more prominent. Several security models and architectures for the edge layer of IoT have been proposed in the literature. Although the existing architectures are able to address some security challenges in the IoT environment, most of such architectures suffer from the two main shortcomings. First, they virtually focus on a specific threat in the edge layer of IoT architecture without the ability to deal with new threats or even new versions of current threats that they were initially designed for it. Second, their underlying mechanisms are device dependable with the need for external hardware for implementing the suggested secure architectures in most cases. In this paper, we proposed an AI-powered architecture for IoT environments, named AI4SAFE-IoT. In AI4SAFE-IoT, we defined a three-layer architecture with security modules based on AI engines for each layer of IoT which has interaction with the edge layer devices. In addition, we presented AI engines as a separated layer along a data channel that collects every existing view from a threat and transforms each view as an appropriate feature set for feeding AI engines. One of the main advantages of AI4SAFE-IoT architecture is its independence of operation from end-point device resources by delivering a pervasive security mechanism based on AI through service.

Compliance with ethical standards

Conflict of interest The authors declare that they have no conflict of interest.

References

1. Zhu Q, Wang R, Chen Q, Liu Y, Qin W (2010) Iot gateway: bridgingwireless sensor networks into Internet of Things. In: 2010 IEEE/IFIP international conference on embedded and ubiquitous computing. IEEE, pp 347–352
2. Chiang M, Zhang T (2016) Fog and IoT: an overview of research opportunities. IEEE Internet Things J 3(6):854–864
3. Conti M, Dehghantanha A, Franke K, Watson S (2018) Internet of Things security and forensics: challenges and opportunities. Elsevier, Amsterdam
4. Sakhnini J, Karimipour H, Dehghantanha A, Parizi RM, Srivastava G (2019) Security aspects of Internet of Things aided smart grids: a bibliometric survey. Internet Things. https://doi.org/10.1016/j.iot.2019.100111
5. Darabian H, Dehghantanha A, Hashemi S, Taheri M, Azmoodeh A, Homayoun S, Choo K-KR, Parizi RM (2020) A multiview learning method for malware threat hunting: windows, IoT and android as case studies. World Wide Web. https://doi.org/10.1007/s11280-019-00755-0
6. Domingo MC (2012) An overview of the Internet of Things for people with disabilities. J Netw Comput Appl 35(2):584–596
7. Jia X, Feng Q, Fan T, Lei Q (2012) RFID technology and its applications in Internet of Things (IoT). In: 2012 2nd International conference on consumer electronics, communications and networks (CECNet). IEEE, pp 1282–1285
8. Wang L, Da Xu L, Bi Z, Xu Y (2013) Data cleaning for RFID and WSN integration. IEEE Trans Ind Inf 10(1):408–418
9. Liu CH, Yang B, Liu T (2014) Efficient naming, addressing and profile services in Internet-of-Things sensory environments. Ad Hoc Netw 18:85–101
10. HaddadPajouh H, Dehghantanha A, Parizi RM, Aledhari M, Karimipour H (2019) A survey on Internet of Things security: requirements, challenges, and solutions. Internet Things. https://doi.org/10.1016/j.iot.2019.100129
11. Gaur A, Scotney B, Parr G, McClean S (2015) Smart city architecture and its applications based on IoT. Procedia Comput Sci 52:1089–1094
12. Binti N, Kamaludeen A, Lee SP, Parizi RM (2019) Guideline-based approach for IoT home application development. In: 2019 International conference on Internet of Things (iThings) and IEEE green computing and communications (GreenCom) and IEEE cyber, physical and social computing (CPSCom) and IEEE smart data (SmartData), pp 929–936
13. Yun M, Yuxin B (2010) Research on the architecture and key technology of Internet of Things (IoT) applied on smart grid. In: 2010 International conference on advances in energy engineering. IEEE, pp 69–72
14. Behera TM, Mohapatra SK, Samal UC, Khan MS, Daneshmand M, Gandomi AH (2019) Residual energy-based cluster-head selection in wsns for IoT application. IEEE Internet Things J 6:5132–5139
15. Catarinucci L, De Donno D, Mainetti L, Palano L, Patrono L, Stefanizzi ML, Tarricone L (2015) An IoT-aware architecture for smart healthcare systems. IEEE Internet Things J 2(6):515–526
16. He W, Yan G, Da Xu L (2014) Developing vehicular data cloud services in the IoT environment. IEEE Trans Ind Inf 10(2):1587–1595
17. Behera TM, Mohapatra SK, Samal UC, Khan MS, Daneshmand M, Gandomi AH (2020) I-sep: an improved routing protocol for heterogeneous WSN for IoT-based environmental monitoring. IEEE Internet Things J 7(1):710–717
18. Paranjothi A, Tanik U, Wang Y, Khan MS (2019) Hybrid-vehfog: a robust approach for reliable dissemination of critical messages in connected vehicles. Trans Emerg Telecommun Technol 30(6):e3595 (e3595 ETT-18-0175.R3)
19. Gardašević G, Veletić M, Maletić N, Vasiljević D, Radusinović I, Tomović S, Radonjić M (2017) The IoT architectural framework, design issues and application domains. Wirel Pers Commun 92(1):127–148
20. Karanki SS, Khan MS (2017) SMMV: secure multimedia delivery in vehicles using roadside infrastructure. Veh Commun 7:40–50
21. Ngu AH, Gutierrez M, Metsis V, Nepal S, Sheng QZ (2016) IoT middleware: a survey on issues and enabling technologies. IEEE Internet Things J 4(1):1–20
22. Kelly SDT, Suryadevara NK, Mukhopadhyay SC (2013) Towards the implementation of IoT for environmental condition monitoring in homes. IEEE Sens J 13(10):3846–3853
23. Giuliano R, Mazzenga F, Neri A, Vegni AM (2016) Security access protocols in IoT capillary networks. IEEE Internet Things J 4(3):645–657
24. Zhang Y, Raychadhuri D, Ravindran R, Wang G (2013) ICN based architecture for IoT. IRTF Contribution
25. Chang K-H (2014) Bluetooth: a viable solution for IoT?[Industry perspectives]. IEEE Wirel Commun 21(6):6–7
26. Santos J, Rodrigues JJ, Silva BM, Casal J, Saleem K, Denisov V (2016) An IoT-based mobile gateway for intelligent personal assistants on mobile health environments. J Netw Comput Appl 71:194–204
27. Behera TM, Khan MS, Mohapatra SK, Samail UC, Bhuiyan MZA (2019) Energy-efficient routing for greenhouse monitoring using heterogeneous sensor networks. In: 2019 International conference on Internet of Things (iThings) and IEEE green computing and communications (GreenCom) and IEEE cyber, physical and social computing (CPSCom) and IEEE smart data (SmartData), pp 953–958
28. Ren J, Guo H, Xu C, Zhang Y (2017) Serving at the edge: a scalable IoT architecture based on transparent computing. IEEE Netw 31(5):96–105
29. HaddadPajouh H, Dehghantanha A, Khayami R, Choo K-KR (2018) A deep Recurrent Neural Network based approach for Internet of Things malware threat hunting. Future Gener Comput Syst 85:88–96
30. Baccelli E, Gündoğan C, Hahm O, Kietzmann P, Lenders MS, Petersen H, Schleiser K, Schmidt TC, Wählisch M (2018) RIOT: an open source operating system for low-end embedded devices in the IoT. IEEE Internet Things J 5(6):4428–4440
31. Levis P et al (2005) TinyOS: an operating system for sensor networks. In: Weber W, Rabaey JM, Aarts E (eds) Ambient intelligence. Springer, Berlin, Heidelberg, pp 115–148
32. Dovom EM, Azmoodeh A, Dehghantanha A, Newton DE, Parizi RM, Karimipour H (2019) Fuzzy pattern tree for edge malware detection and categorization in IoT. J Syst Architect 97:1–7
33. Alaba FA, Othman M, Hashem IAT, Alotaibi F (2017) Internet of Things security: a survey. J Netw Comput Appl 88:10–28
34. Jing Q, Vasilakos AV, Wan J, Lu J, Qiu D (2014) Security of the Internet of Things: perspectives and challenges. Wireless Netw 20(8):2481–2501
35. Kupreev O, Kupreev O, Badovskaya E, Gutnikov A. DDoS attacks in Q4 2018, Securelist english. [Online]. https://securelist.com/ddos-attacks-in-q4-2018/89565/. Accessed 21 Dec 2019
36. Patel P, Ali MI, Sheth A (2017) On using the intelligent edge for IoT analytics. IEEE Intell Syst 32(5):64–69
37. Li H, Ota K, Dong M (2018) Learning IoT in edge: deep learning for the Internet of Things with edge computing. IEEE Netw 32(1):96–101

38. Yazdinejad A, Parizi RM, Dehghantanha A, Zhang Q, Choo KR (2020) An energy-efficient SDN controller architecture for IoT networks with blockchain-based security. IEEE Trans Serv Comput. https://doi.org/10.1109/TSC.2020.2966970
39. Moosavi SR, Gia TN, Rahmani A-M, Nigussie E, Virtanen S, Isoaho J, Tenhunen H (2015) SEA: a secure and efficient authentication and authorization architecture for IoT-based healthcare using smart gateways. Procedia Comput Sci 52:452–459
40. Leusse PD, Dimitrakos T (2010) SOA-Based security governance middleware. In: 2010 fourth international conference on emerging securityinformation, systems and technologies
41. Vučinić M, Tourancheau B, Rousseau F, Duda A, Damon L, Guizzetti R (2015) OSCAR: object security architecture for the Internet of Things. Ad Hoc Netw 32:3–16
42. Nawir M, Amir A, Yaakob N, Lynn OB (2016) Internet of Things (IoT): taxonomy of security attacks. In: 2016 3rd International conference on electronic design (ICED). IEEE, pp 321–326
43. Pinto S, Gomes T, Pereira J, Cabral J, Tavares A (2017) IIoTEED: an enhanced, trusted execution environment for industrial IoT edge devices. IEEE Internet Comput 21(1):40–47
44. Jang J, Kong S, Kim M, Kim D, Kang BB (2015) SeCReT: secure channel between rich execution environment and trusted executionenvironment. In: Proceedings 2015 network and distributed system security symposium
45. Dai W, Jin H, Zou D, Xu S, Zheng W, Shi L, Yang LT (2015) TEE: a virtual DRTM based execution environment for secure cloud-end computing. Future Gener Comput Syst 49:47–57
46. Bormann C, Castellani AP, Shelby Z (2012) Coap: an application protocol for billions of tiny internet nodes. IEEE Internet Comput 2:62–67
47. Hunkeler U, Truong HL, Stanford-Clark A (2008) MQTT-S A publish/subscribe protocol for wireless sensor networks. In: 2008 3rd International conference on communication systems software and middleware and workshops (COMSWARE'08). IEEE, pp 791–798
48. Ito M, Iyatomi H (2018) Web application firewall using character-level convolutional neural network. In: 2018 IEEE 14th International colloquium on signal processing and its applications (CSPA). IEEE, pp 103–106
49. Bekara C (2014) Security issues and challenges for the IoT-based smart grid. Procedia Comput Sci 34:532–537
50. Pajouh HH, Dastghaibyfard G, Hashemi S (2017) Two-tier network anomaly detection model: a machine learning approach. J Intell Inf Syst 48(1):61–74
51. Pajouh HH, Javidan R, Khayami R, Dehghantanha A, Choo KR (2016) A two-layer dimension reduction and two-tier classification model for anomaly-based intrusion detection in IoT backbone networks. IEEE Trans Emerg Top Comput. 7(2):314–323
52. Azmoodeh A, Dehghantanha A, Conti M et al (2018) Detecting crypto-ransomware in IoT networks based on energy consumption footprint. J Ambient Intell Hum Comput 9:1141–1152
53. Al-Garadi MA, Mohamed A, Al-Ali A, Du X, Guizani M (2018) A survey of machine and deep learning methods for Internet of Things (IoT) security. arXiv:1807.11023
54. Antonakakis M, April T, Bailey M, Bernhard M, Bursztein E, Cochran J, Durumeric Z, Halderman JA, Invernizzi L, Kallitsis M (2017) Understanding the mirai botnet. In: 26th {USENIX} security symposium ({USENIX} Security 17), pp 1093–1110
55. Kolias C, Kambourakis G, Stavrou A, Voas J (2017) DDoS in the IoT: Mirai and other botnets. Computer 50(7):80–84
56. Yaqoob I, Ahmed E, Hashem IAT, Ahmed AIA, Gani A, Imran M, Guizani M (2017) Internet of Things architecture: recent advances, taxonomy, requirements, and open challenges. IEEE Wirel Commun 24(3):10–16
57. HosseiniNejad R, HaddadPajouh H, Dehghantanha A, Parizi RM (2019) A cyber kill chain based analysis of remote access trojans. In: Dehghantanha A, Choo KK (eds) Handbook of big data and iot security. Springer, Cham
58. Taylor PJ, Dargahi T, Dehghantanha A (2019) Analysis of APT actors targeting iot and big data systems: shell_crew, nettraveler, projectsauron, copykittens, volatile cedar and transparent tribe as a case study. In: Handbook of big data and iot security, pp 257–272
59. Mwiki H, Dargahi T, Dehghantanha A, Choo KKR (2019) Analysis and triage of advanced hacking groups targeting western countries criticalnational infrastructure: APT28, RED October, and Regin. In: Gritzalis D, Theocharidou M, Stergiopoulos G (eds) Critical infrastructure security and resilience. Advanced sciences and technologies for security applications. Springer, Cham
60. Bahrami PN, Dehghantanha A, Dargahi T, Parizi RM, Choo KK, Javadi HH (2019) Cyber kill chain-based taxonomy of advanced persistent threat actors: analogy of tactics. techniques, and procedures. J Inf Process Syst 15(4):865–889
61. Erl T (2005) Service-oriented architecture (SOA): concepts, technology, and design
62. Bhuyan P, Ray A, Mohapatra DP (2015) A service-oriented architecture (SOA) framework component for verification of choreography. In: JainL, Behera H, Mandal J, Mohapatra D (eds) Computational intelligence in data mining. Smart Innovation, Systems and Technologies, vol 3. Springer, New Delhi
63. Guan L, Liu P, Xing X, Ge X, Zhang S, Yu M, Jaeger T (2017) Trustshadow: secure execution of unmodified applications with arm trustzone. In: Proceedings of the 15th annual international conference on mobile systems, applications, and services. ACM, pp 488–501
64. Tiburski RT, Moratelli CR, Johann SF, Neves MV, de Matos E, Amaral LA, Hessel F (2019) Lightweight security architecture based on embedded virtualization and trust mechanisms for IoT edge devices. IEEE Commun Mag 57(2):67–73
65. Ahmed AIA, Gani A, Hamid SHA, Abdelmaboud A, Syed HJ, Habeeb Mohamed RAA, Ali I (2019) Service management for IoT: requirements, taxonomy, recent advances and open research challenges. IEEE Access 7:155472–155488
66. Woo MW, Lee J, Park K (2018) A reliable IoT system for personal healthcare devices. Future Gener Comput Syst 78:626–640