Perpetrate cyber-attacks using IoT devices as attack vector: the ESP8266 use case

Ivan Vaccari, Sara Narteni, Maurizio Mongelli, Maurizio Aiello and
Enrico Cambiaso

Consiglio Nazionale delle Ricerche (CNR-IEIIT), Genoa, Italy

Abstract

Security of the Internet of Things is a crucial topic, due to the criticality of the networks and the sensitivity of exchanged data. In this paper, we evaluate the adoption of IoT devices to execute cyber-threats by using a specific Wi-Fi module called ESP8266. This module may implement custom user applications, but it could also be adopted for malicious purposes, as to perpetrate cyber-attacks. In particular, we implemented a social engineering attack to steal sensitive information and a slow denial of service attack to saturate the resources of a web service based on an Apache2 server. Obtained results report that the ESP8266 module is able to perform both attacks successfully. Hence, we demonstrate that even a simple and cheap module is able to execute critical cyber-attacks.

Keywords

Internet of Things, cyber-attack, ESP8266, Wi-Fi, cyber-security

1. Introduction

Internet of Things (IoT) is a consolidated technology developed in recent years for Information Technology (IT) and Operational Technology (OT) infrastructures [1]. The IoT phenomenon has been labelled by experts as “the next Industrial Revolution” [2]: nowadays, around 30 billions of IoT devices are connected over the Internet and such number is expected to increase in the next years to nearly 75 billion of IoT devices in late 2025. Thanks to IoT, simple objects gain the ability to store, elaborate and communicate sensitive information among themselves or with external systems.

In recent years, IoT has gained much attraction from researchers and industries from around the world due to the potential benefits that this technology could bring. Given the characteristics brought by IoT networks and systems, such devices are widely adopted in different scenarios such as home automation, Industry 4.0, critical infrastructures (hospital, banks, power grid). For instance, IoT environments may be adopted to control temperature and humidity remotely, turn on/off light bulbs, or to monitor and control patients’ health parameters (in this case, we talk of Internet of Medical Things (IoMT) [3]). Also, fields like robotics, wireless sensor networks and embedded systems are adopting such technology.

ITASEC21: Italian Conference on Cybersecurity, April 7-9, 2021, Online

ivan.vaccari@ieiit.cnr.it (I. Vaccari); sara.narteni@ieiit.cnr.it (S. Narteni); maurizio.mongelli@ieiit.cnr.it (M. Mongelli); maurizio.aiello@ieiit.cnr.it (M. Aiello); enrico.cambiaso@ieiit.cnr.it (E. Cambiaso)
0000-0001-7721-5737 (I. Vaccari); 0000-0002-0579-647X (S. Narteni); 0000-0001-6201-6225 (M. Mongelli); 0000-0001-5008-225X (M. Aiello); 0000-0002-6932-1975 (E. Cambiaso)

Given the nature of the different applications and contexts, several options and infrastructures are available to implement IoT networks composed by different devices and communication protocols. For instance, a possible solution is to adopt well-known standards such as the IEEE 802.11 Wi-Fi protocol, making use of the full TCP/IP stack. This approach would provide integration of IoT devices on existing networks and with existing applications, since it is based on a consolidated and widely adopted standard protocol. Nevertheless, such solution also inherits security vulnerabilities and limits, since every Wi-Fi IoT device would be exposed to already existing well-known Wi-Fi threats, such as denial of service [4], de-authentication [5], replay [6, 7] or brute force attacks [8]. On the other side, alternative IoT protocols provide features such as low power consumption capabilities provided by the ZigBee protocol [9], or wide range support offered by 6LowPan [10], at the cost of adopting a new and less mature network protocol. Another approach followed to implement IoT networks is based on existing protocols such as Constrained Application Protocol (CoAP) [11], on top of UDP, or Message Queue Telemetry Transport (MQTT) [12], on top of TCP.

Addressing cyber-security aspects of IoT devices is particularly important and crucial, since IoT nodes are often demanded to accomplish critical activities, or because of the nature of the devices, often able to collect and manage sensitive information (e.g. vital parameters of patients in healthcare contexts, or citizens' location data in smart cities environments). In addition, the limited capabilities of IoT nodes is often in contrast with the deployment of proper (cyber) protection, as the implementation of security measures introduces an additional computation layer on the sensor node. Given the spread of IoT devices, it is also necessary to evaluate the adoption of these devices as attack vectors, since with simple devices it is possible to carry out very important cyber-attacks. An example of cyber attacks executed by IoT devices is Mirai [13], considered nowadays one of the most predominant DDos IoT botnets in recent times. Mirai infected more than 4,000 IoT devices per hour and it is currently estimated to have just over half a million active infected IoT devices in the world. The Mirai botnet generated around 1.1Tbps of DDos attack traffic, by exploiting more than 148,000 IoT devices like CCTV cameras, DVRs and home routers [14, 15]. This lead to a DoS affecting huge portions of the Internet, including Amazon, Twitter, the Guardian, Netflix, Reddit, Github and CNN. Based on these considerations, it is important to consider possible cyber-threats executed by IoT devices.

By following such possibility, in our paper, we investigate cyber-attacks executed by IoT devices communicating through the Wi-Fi protocol. Particularly, we designed, implemented and perpetrated two cyber-attacks by using the ESP8266 Wi-Fi module: a slow denial of service (slow dos) attack [16, 17] and a social engineering [18] attack. If we consider the execution of such threat from low-power devices like the ESP8266 module, social engineering attacks could be executed to steal sensitive information, while the capturing device is physically hidden in order to make it difficult to identify. In particular, the reduced size of the ESP8266 module (similar to a coin), coupled with the possibility to power it by batteries, make it possible to hide the ESP8266 effectively, while running a social engineering attack. Instead, regarding the Slow DoS Attack, the possibility to successfully lead a DoS on an Internet service, by using a low cost and low power device opens to new possibilities that cyber-criminals could exploit. To the best of our knowledge, no previous works consider the adoption of the ESP8266 module to execute social engineering and Slow DoS Attacks.

The remaining of the paper is organized as follows: Section 2 reports the related work on the topic. Instead, Section 3 reports the description of the ESP8266 module, while Section 4 describes the implemented threats and the obtained results. Finally, Section 5 concludes the paper and reports further work on the topic.

2. Related work

IoT networks can be implemented by using different devices and protocols, depending on the application needs. By focusing on Wi-Fi networks, the ESP8266 module is widely used for different purposes and in different contexts. For instance, regarding sport applications, [19, 20] embed the ESP8266 module on wearable devices in order to measure the high-performance and low-injury real-time high jump during sports activities. Concerning healthcare, [21, 22] theorize a heart rate monitoring infrastructure to combine pulse sensors with ESP8266 devices to remotely control patient's health parameters. Instead [23] develops a remote control prosthetic hand controlled by an Android application and making use of an ESP8266 to simulate the hand gesture selected on the app. [24] implements an IoT application to monitor the closure of the eyelid to avoid a drowsiness problems. In this case, the ESP8266 is attached to the eyeglass is order to remotely monitor eyes parameters. The ESP8266 module is also adopted in industrial applications to monitor and control energy systems [25, 26, 27, 28], to automatic implement air condition systems [29], or to prevent landslide [30]. Referring to home automation contexts, [31, 32, 33, 34, 35] implement an ESP8266-based system using the MQTT protocol to monitor different physical parameters retrieved by home environments. Instead, [36] develops a smart irrigation system controlling and monitoring irrigation, embedding both an ESP8266 module and an Arduino micro-controller.

Mentioned works focus on the adoption of the ESP8266 module for different contexts. The key point is related to the modularity of the module under analysis. Given the widespread possibility of applications, the device could also be used to perform cyber-attacks against networks or infrastructures. For this reason, in the proposed paper, the main focus is on the adoption of the ESP8266 as source of cyber-attacks, in particular we focused on captive portal and slow denial of service attack. Captive portal is social engineering attack [37] aimed to steal sensitive information of victims [38, 39], for example login credentials, banks information or credit cards. Our work exploit this cyber-attack by using the ESP8266 module as attack vector. Regarding slow denial of service attacks, such threat belongs to the category of Slow DoS Attacks, making use of minimum attack bandwidth and resources to target a network service executing a denial of service [40, 41, 42]. In the IoT context, implementation of this attack against MQTT and ZigBee is investigated [43, 16, 17] but the source of the attack are usually micro-controllers with discrete power computing capabilities (in particular, Raspberry Pi 3). Constrained module such as the ESP8266 module are not investigated.

The proposed work focuses on the implementation of cyber-attacks by using the ESP8266 module as attack vector. Due to the critical computation hardware of the module, we can state that applications that require strong computation are complex to develop and implement on the ESP8266 module, in virtue of its limited capabilities. Hence, we propose and validate two cyber-attacks, developed in the module, able to steal sensitive information (social engineering attack) and to saturate services on the network (slow denial of service attack). To the best of our knowledge, works focused on adoption on ESP8266 module as attack vector are limited. Mentioned works highlight applications based on the ESP8266 module. Nevertheless, they do not focus on the use of such device to carry out cyber-attacks. In addition, the proposed work should be considered relevant in the network protocols security topic, since it highlights the possibility of using simple devices to perform critical cyber-attacks.

3. The ESP8266 Module

The ESP8266 module is a low-price micro-controller, developed by Espressif, providing Wi-Fi connectivity based on a full TCP/IP stack to an embedded micro-controller. The module is composed by a L106 32-bit RISC microprocessor with 32 KiB instruction RAM, 32 KiB instruction cache RAM, 80 KiB user-data RAM and 16 KiB ETS system-data RAM. Moreover, this module could be connected to different sensors through 16 GPIO pins and UART on dedicated pins. Different software development kit (SDK) are available to program custom applications by users. Being programmable, this device can be used for a large number of applications, such as environment control [44], health parameters [45] or industrial processes [46]. In addition, the size of the device and the price are extremely reduced. There are several variants of the original ESP8266 developed by Espressif, although the basic computational characteristics are often unchanged [47].

Being programmable, the module can host applications of different nature according to the needs of the users. Based on this concept, users can develop any application and in this case they can also develop cyber-attacks. The advantage of using a simple module such as the ESP8266 module is related to the use of IoT sensors distributed throughout the territory without requiring hardware with high computational capabilities. Furthermore, these devices are often used in critical infrastructures where the information exchanged are very sensitive [48]. A cyber-attack on these infrastructures can lead to serious damage such as the malfunction of entire systems or the steal of sensitive information [49]. In this work, two attacks against these scenarios are reported and described. The first attack is related to the theft of sensitive information through the use of a social engineering attack. The ESP8266 module has been programmed to create a fake free Wi-Fi network that requires credentials from a well-known social network to authenticate possible victims with the aim of stealing login credentials. The second attack, on the other hand, is a known attack in the context of cyber-security called denial of service attack: in this scenario, the ESP8266 module is able to make a service in the network unreachable by legitimate requests through the use of a specific attack, called slow denial of service, that requires limited computational capabilities to run. In these scenarios, the real innovation is related to the use of the ESP8266 module to execute the cyber-attacks as, as previously described, the module has limited computational capabilities.

4. Cyber-attacks executed by the ESP8266 module and obtained results

In this section, we report a detailed description of the attacks developed and the obtained results. Initially, the proposed attacks are described in detail. Then, the testbed used to validate the attacks is presented. Finally, the obtained results are reported.

4.1. Using ESP8266 as vector attack to execute Slow Dos Attack

During our research, we focused on the implementation of a denial of service attack into the ESP8266 module, aimed to make a network service unavailable to its intended users. As conventional (e.g. flooding) denial of service require a large amount of resources to the attacker [40], we focused on Slow DoS Attacks [42, 50], characterized by low attack requirements. Slow DoS Attacks (SDA) are designed to execute a DoS attack by adopting a low-rate approach, making use of minimum attack bandwidth and resources to target a network service. Our aim is to demonstrate the possibility to adopt simple (not powerful) and cheap IoT modules to carry out potentially relevant cyber-attacks against conventional network/Internet services.

In the proposed scenario, the victim is represented by an Apache2 web server, able to handle by default 150 connections simultaneously. Therefore, it is assumed that the dos state is achieved by the attacker when 150 connections are established/seized on the victim. In order to evaluate the attack and the obtained results, connections on the server are monitored by considering an attack of 10 minutes.

In detail, the ESP8266 module has the goal of instantiating all connections available on the server with the aim of avoiding legitimate connections. The attack flow is very simple: the ESP8266 module initializes all available connections on the network with the aim of reaching 150 connections and keeping these connections alive as long as possible to avoid legitimate requests. Initially, the ESP8266 module established a communication with the server by exploiting the 3-way handshake available in the TCP communication for each connection binded on different source ports. Then, by exploiting the KeepAlive parameter, it tried to maintain the connection alive. In order to implement this cyber-attack, the Arduino IDE is adopted. The code is written in C, for this reason it is fast and efficient even if inside the device, it takes up a lot of the space available for the code. The proposed version of the attack is not fully optimized indeed, as reported in the results, the ESP8266 lost connections during the attack. As scope of future work, the optimization of the cyber-attack will be investigated in order to maintain connections alive for more time and to target other web servers.

Obtained results are shown in Figure 1. Particularly, just after a few seconds from the beginning of the attack, the ESP8266 module is able to lead the dos on the victim. Nevertheless, unlike for conventional attacking nodes [50], the dos is not continuously maintained over time, because of the computational limits of the module. Nevertheless, as connections are re-established and the attack influence [50] is kept high. Such aspect is especially measured after around 8 minutes from the beginning of the attack, where most of the established connections were closed by the server. Nevertheless, as shown, the IoT module is able to re-establish connections as soon as a closure is detected, hence maintaining an high attack influence [50] on the server.

Although the dos is not maintained over time, obtained results prove how low-cost IoT devices can be adopted to perpetrate cyber-attacks against network services. Such threats target networks of different nature, also including critical environments such as healthcare [51] or medical IoT devices [52]. In such sensitive contexts, it is important to consider that a lack in protection may compromise patients' health conditions, hence, potentially compromise not

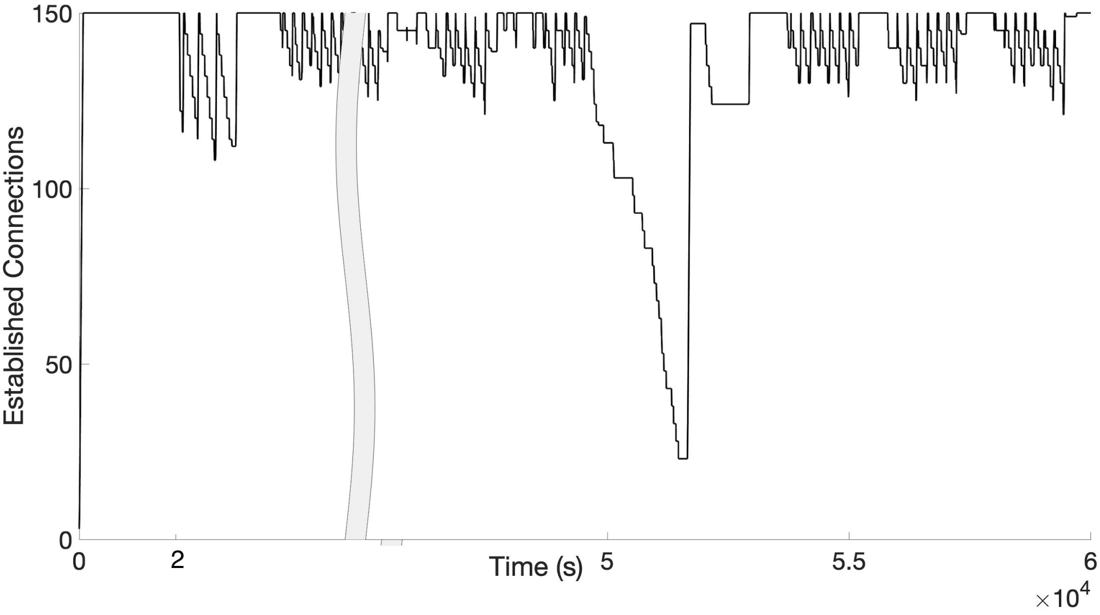

Figure 1: Results of a Slow DoS Attack perpetrated by a single ESP8266 module against an Apache2 web server

only the functioning of the underlying system, but also expose potential threats to human lives. In virtue of this, regarding IoT and network security topics, by following the concept behind the Mirai attack [53], it is important to consider attacks against single network services perpetrated by coordinated IoT nodes.

4.2. Captive portal attack by exploiting the ESP module

The second cyber-attack developed using the ESP8266 module is a social engineering attack, implemented through a fake captive portal. Normally a captive portal is an Access Point or a gateway that allows the connection to the Internet after a login procedure through a web page. In the context of cyber-security, however, the attack creates a malicious Access Point with the aim of stealing access credentials from victims who connect to the fake Wi-Fi network created by the attacker: for this reason, a malicious captive portal is considered a social engineering attack. Social engineering represents a set of techniques used by cyber-criminals to lure unsuspecting users to send them their confidential data, infect their computers with malware or open links to infected sites. This type of attack turns out to be very dangerous since, usually, the aim of malicious users is to recover sensitive user information.

In the proposed scenario, the attack was developed using the NodeMCU ROM within the ESP8266 module. The attack was fully developed during the research activities relating to the ESP8266 module and does not use particular libraries as the activities performed, i.e. the creation of Wi-Fi networks and the release of web pages, are developed on the ESP8266 modules. Although scientifically it does not impact the security sector as these attacks are known, the innovation point is related to adoption of this simple sensor to implement the attack. In fact, despite the small size and limited performance, it is possible to implement a Captive portal attack with the ESP module. In detail, the project developed has as its aim the theft of the victim's Facebook credentials. At the first start of the malicious device, a network with service set identifier (SSID) called *ConfigureAP* is created, which can only be accessed via password in order to limit access to the malicious user only: from here the attacker can decide the name of the network that will be displayed from other users and that it will actually be the fake Captive Portal. Once the SSID name has been decided, the device is automatically restarted and a free wireless network will be created with the chosen SSID. Once the network is created, the device is ready to retrieve the victims' Facebook credentials. In particular, the victim who connects to the network will get redirected to a page where to continue, and therefore authenticate on the network and browse freely, it is necessary to enter their Facebook credentials, as often happens in free networks (for example free Wi-Fi city networks). Figure 2 reports the sequence diagram.

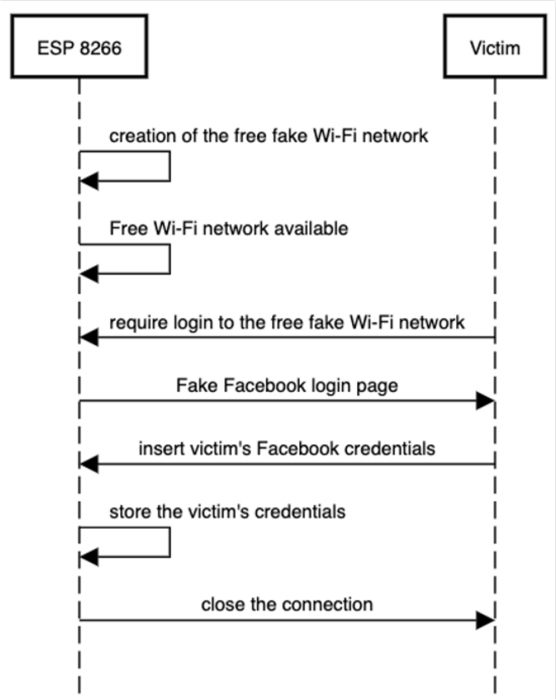
ESP 8266
Victim
creation of the free fake Wi-Fi network
Free Wi-Fi network available
require login to the free fake Wi-Fi network
Fake Facebook login page
insert victim's Facebook credentials
store the victim's credentials
close the connection

Figure 2: Sequence diagram of the captive portal attack

At the end of the insertion and sending of the credentials, the user will not get any reply message regarding the authentication so he will be led to interpret it as an error in the login procedure or a network malfunction, in reality his credentials will be saved on a file called *credentials.txt* inside the device. Here the Fake Captive Portal attack was successful. Once the data has been collected, the attacker will be able to retrieve the victim's login credentials. The challenge of this attack is related to the file used to store the credentials as if it became too large, the ESP8266 module would saturate the available memory (as previously mentioned very limited) and this could lead to a malfunction of the attack. Given the sensitivity of the information gathered by this attack, it could not be tested in a public setting. The operation of the attack was tested with up to 10 credentials stored on the file using fake credentials. Further tests will be carried out in relation to the simulation of victims by entering fictitious credentials to verify the behavior of the module as the file size increases. A possible solution could be file compression which however requires computational capacity added to the module while the captive portal is running. Furthermore, the credentials could be sent to an external server, but in this case a connection and additional computational capacity is required.

A strong point of this device is its small size and the power supply through simple AA batteries, therefore it is even possible to hide it in a public place, and then recover it at a later time. The device was built using the ESP8266 module, two AA batteries, two switches which serve respectively as on/off and as a reset of the default credentials, all connected by jumper cables, soldered to each other.

As previously anticipated the innovation of this attack turns out to be the implementation on a simple and accessible device such as the ESP8266 module. With a simple device from a few euros it is therefore possible to retrieve sensitive and private information of users, thus affecting their privacy and information security.

5. Conclusions and future works

In this work, we presented two cyber-attacks executed by the ESP8266 module, an IoT device adopted to implement custom applications through the Wi-Fi communication protocol. As discussed in the paper, through a not powerful and cheap device, it is possible to implement potentially critical cyber-attacks. In the context of this paper, a social engineering attack and a slow denial of service attack. In the Slow DoS Attack, the ESP8266 executes the attack against an Apache2 web server, to inhibit legitimate communications with the server. Instead, in the social engineering scenario, the ESP8266 module is used to create a fake free Wi-Fi network, with the aim to steal social network credentials required to the connected users in order to provide them connectivity. Although such scenarios are different between themselves, the proposed work highlights the possibility to exploit low-cost and popular IoT devices for different malicious purposes. For instance, in the next years, it is not excluded that new attacks using the same approach of Mirai will target IoT devices, or that an underground network of attacking nodes will be built, maybe also exploiting the characteristic of blockchain technologies. Therefore, by focusing on providing adequate protection to IoT networks and devices is a necessary step in order to make such technology a pervasive, but secure, technology.

As future work, we will evaluate other cyber-attacks executed directly by the ESP8266 module. Moreover, we will validate these threats in a critical scenario to demonstrate the efficient of the threats. Another interesting work will be focused on the resources and energy required by the module to execute cyber-attacks, by comparing different attacks of different nature. Also, the proposed cyber-attacks will be optimized in order to improve efficient and quality. Regarding the Slow DoS Attack, the tool will be optimized to target other possible web servers while the captive portal will be validated by simulating a large number of credentials to monitor resources and space on the ESP8266 module. Moreover, a possible extension of the work may focus on the adoption of the ESP8266 module to execute distributed attacks, coordinated, e.g., by decentralised infrastructures sharing attack parameters, for example, to perpetrate distributed denial of server (DDoS) attacks.

Acknowledgement

This work was supported by the following research project: the Integrated Framework for Predictive and Collaborative Security of Financial Infrastructures (FINSEC) project, which has received funding from the European Union's Horizon 2020 Research and Innovation Programme under Grant Agreement No. 786727.

References

[1] M. Mikusz, S. Houben, N. Davies, K. Moessner, M. Langheinrich, Raising awareness of iot sensor deployments (2018).

[2] C. MacGillivray, V. Turner, D. Lund, Worldwide Internet of Things (IoT) 2013–2020 Forecast: Billions of Things, Trillions of Dollars, Gartnet Market Analysis (2013).

[3] F. Alsubaei, A. Abuhussein, S. Shiva, Security and privacy in the internet of medical things: taxonomy and risk assessment, in: 2017 IEEE 42nd Conference on Local Computer Networks Workshops (LCN Workshops), IEEE, 2017, pp. 112–120.

[4] L. Xiao, X. Wan, X. Lu, Y. Zhang, D. Wu, Iot security techniques based on machine learning: How do iot devices use ai to enhance security?, IEEE Signal Processing Magazine 35 (2018) 41–49.

[5] A. Sun, W. Gong, R. Shea, J. Liu, A castle of glass: Leaky iot appliances in modern smart homes, IEEE Wireless Communications 25 (2018) 32–37.

[6] M. Nawir, A. Amir, N. Yaakob, O. B. Lynn, Internet of things (iot): Taxonomy of security attacks, in: 2016 3rd International Conference on Electronic Design (ICED), IEEE, 2016, pp. 321–326.

[7] I. Vaccari, M. Aiello, F. Pastorino, E. Cambiaso, Protecting the esp8266 module from replay attacks, in: 2020 International Conference on Communications, Computing, Cybersecurity, and Informatics (CCCI), IEEE, 2020, pp. 1–6.

[8] H. Berghel, J. Uecker, Wifi attack vectors, Communications of the ACM 48 (2005) 21–28.

[9] P. Kinney, et al., Zigbee technology: Wireless control that simply works, in: Communications design conference, volume 2, 2003, pp. 1–7.

[10] G. Mulligan, The 6lowpan architecture, in: Proceedings of the 4th workshop on Embedded networked sensors, ACM, 2007, pp. 78–82.

[11] Z. Shelby, K. Hartke, C. Bormann, The constrained application protocol (coap) (2014).

[12] U. Hunkeler, H. L. Truong, A. Stanford-Clark, 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, 2008, pp. 791–798.

[13] M. Antonakakis, T. April, M. Bailey, M. Bernhard, E. Bursztein, J. Cochran, Z. Durumeric, J. A. Halderman, L. Invernizzi, M. Kallitsis, et al., Understanding the mirai botnet, in: 26th {USENIX} Security Symposium ({USENIX} Security 17), 2017, pp. 1093–1110.

[14] B. Krebs, Source code for iot botnet ‘mirai’released, KrebsonSecurity.(Oct. 2016). Retrieved Feb 23 (2016) 2017.

[15] B. Herzberg, D. Bekerman, I. Zeifman, Breaking down mirai: An iot ddos botnet analysis, Incapsula Blog, Bots and DDoS, Security (2016).

[16] I. Vaccari, M. Aiello, E. Cambiaso, Slowite, a novel denial of service attack affecting mqtt, Sensors 20 (2020) 2932.

[17] I. Vaccari, M. Aiello, E. Cambiaso, Slowtt: A slow denial of service against iot networks, Information 11 (2020) 452.

[18] F. Salahdine, N. Kaabouch, Social engineering attacks: a survey, Future Internet 11 (2019) 89.

[19] M. F. Roslan, A. Ahmad, A. Amira, Real-time high jump wearable device with esp8266 for high-performance and low-injury, International Journal of Integrated Engineering 10 (2018).

[20] S. Hiremath, G. Yang, K. Mankodiya, Wearable internet of things: Concept, architectural components and promises for person-centered healthcare, in: 2014 4th International Conference on Wireless Mobile Communication and Healthcare-Transforming Healthcare Through Innovations in Mobile and Wireless Technologies (MOBIHEALTH), IEEE, 2014, pp. 304–307.

[21] A. Škraba, A. Koložvari, D. Kofjač, R. Stojanović, V. Stanovov, E. Semenkin, Prototype of group heart rate monitoring with nodemcu esp8266, in: 2017 6th Mediterranean Conference on Embedded Computing (MECO), IEEE, 2017, pp. 1–4.

[22] A. Škraba, A. Koložvari, D. Kofjač, R. Stojanović, V. Stanovov, E. Semenkin, Streaming pulse data to the cloud with bluetooth le or nodemcu esp8266, in: 2016 5th Mediterranean Conference on Embedded Computing (MECO), IEEE, 2016, pp. 428–431.

[23] S. S. Pakalapati, G. G. Chary, A. K. Yadaw, S. Kumar, H. K. Phulawariya, R. Kumar, A prosthetic hand control interface using esp8266 wi-fi module and android application, in: 2017 International Conference on Innovations in Information, Embedded and Communication Systems (ICIIECS), IEEE, 2017, pp. 1–3.

[24] D. Artanto, M. P. Sulistyanto, I. D. Pranowo, E. E. Pramesta, Drowsiness detection system based on eye-closure using a low-cost emg and esp8266, in: 2017 2nd International conferences on Information Technology, Information Systems and Electrical Engineering (ICITISEE), IEEE, 2017, pp. 235–238.

[25] P. Srivastava, M. Bajaj, A. S. Rana, Iot based controlling of hybrid energy system using esp8266, in: 2018 IEEMA Engineer Infinite Conference (eTechNxT), IEEE, 2018, pp. 1–5.

[26] Q. M. Ashraf, M. I. M. Yusoff, A. A. Azman, N. M. Nor, N. A. A. Fuzi, M. S. Saharedan, N. A. Omar, Energy monitoring prototype for internet of things: Preliminary results, in: 2015 IEEE 2nd World Forum on Internet of Things (WF-IoT), IEEE, 2015, pp. 1–5.

[27] S. Karthikeyan, P. Bhuvaneswari, Iot based real-time residential energy meter monitoring system, in: 2017 Trends in Industrial Measurement and Automation (TIMA), IEEE, 2017, pp. 1–5.

[28] S. Thakare, A. Shriyan, V. Thale, P. Yasarp, K. Unni, Implementation of an energy monitoring and control device based on iot, in: 2016 IEEE Annual India Conference (INDICON), IEEE, 2016, pp. 1–6.

[29] L. K. P. Saputra, Y. Lukito, Implementation of air conditioning control system using rest

protocol based on nodemcu esp8266, in: 2017 International Conference on Smart Cities, Automation & Intelligent Computing Systems (ICON-SONICS), IEEE, 2017, pp. 126–130.

[30] S. Biansoongnern, B. Plungkang, S. Susuk, Development of low cost vibration sensor network for early warning system of landslides, Energy Procedia 89 (2016) 417–420.

[31] R. K. Kodali, S. Soratkal, Mqtt based home automation system using esp8266, in: 2016 IEEE Region 10 Humanitarian Technology Conference (R10-HTC), IEEE, 2016, pp. 1–5.

[32] A. Bhatt, J. Patoliya, Cost effective digitization of home appliances for home automation with low-power wifi devices, in: 2016 2nd International Conference on Advances in Electrical, Electronics, Information, Communication and Bio-Informatics (AEEICB), IEEE, 2016, pp. 643–648.

[33] J. Prabaharan, A. Swamy, A. Sharma, K. N. Bharath, P. R. Mundra, K. J. Mohammed, Wireless home automation and security system using mqtt protocol, in: 2017 2nd IEEE International Conference on Recent Trends in Electronics, Information & Communication Technology (RTEICT), IEEE, 2017, pp. 2043–2045.

[34] I. Froiz-Míguez, T. Fernández-Caramés, P. Fraga-Lamas, L. Castedo, Design, implementation and practical evaluation of an iot home automation system for fog computing applications based on mqtt and zigbee-wifi sensor nodes, Sensors 18 (2018) 2660.

[35] R. K. Kodali, K. S. Mahesh, Low cost ambient monitoring using esp8266, in: 2016 2nd International Conference on Contemporary Computing and Informatics (IC3I), IEEE, 2016, pp. 779–782.

[36] P. Singh, S. Saikia, Arduino-based smart irrigation using water flow sensor, soil moisture sensor, temperature sensor and esp8266 wifi module, in: 2016 IEEE Region 10 Humanitarian Technology Conference (R10-HTC), IEEE, 2016, pp. 1–4.

[37] K. Krombholz, H. Hobel, M. Huber, E. Weippl, Advanced social engineering attacks, Journal of Information Security and applications 22 (2015) 113–122.

[38] R. S. Tahina, G. Pathak, Network security: Captive portal, International Journal of Wireless Network Security 6 (2020) 7–9.

[39] S. Ali, T. Osman, M. Mannan, A. Youssef, On privacy risks of public wifi captive portals, in: Data Privacy Management, Cryptocurrencies and Blockchain Technology, Springer, 2019, pp. 80–98.

[40] E. Cambiaso, G. Papaleo, G. Chiola, M. Aiello, Slow dos attacks: definition and categorisation, International Journal of Trust Management in Computing and Communications 1 (2013) 300–319.

[41] E. Cambiaso, G. Papaleo, G. Chiola, M. Aiello, Designing and modeling the slow next dos attack, in: Computational Intelligence in Security for Information Systems Conference, Springer, 2015, pp. 249–259.

[42] E. Cambiaso, G. Papaleo, M. Aiello, Taxonomy of slow dos attacks to web applications, in: International Conference on Security in Computer Networks and Distributed Systems, Springer, 2012, pp. 195–204.

[43] I. Vaccari, E. Cambiaso, M. Aiello, Remotely exploiting at command attacks on zigbee networks, Security and Communication Networks 2017 (2017).

[44] A. Karumbaya, G. Satheesh, Iot empowered real time environment monitoring system, International Journal of Computer Applications 129 (2015) 30–32.

[45] G. Marques, R. Pitarma, Monitoring health factors in indoor living environments using

internet of things, in: World Conference on Information Systems and Technologies, Springer, 2017, pp. 785–794.

[46] K. S. Shinde, P. H. Bhagat, Industrial process monitoring using lot, in: 2017 International Conference on I-SMAC (IoT in Social, Mobile, Analytics and Cloud)(I-SMAC), IEEE, 2017, pp. 38–42.

[47] N. Kolban, Kolban's Book on ESP8266, an introductory book on ESP8266 (2015).

[48] T. Pornchaiyasutthi, T. Anusas-amornkul, A model for victim-rescuer communications under collapsed structures using node mcu esp8266, in: Proceedings of the 2019 2nd International Conference on Electronics, Communications and Control Engineering, ACM, 2019, pp. 34–38.

[49] I. Perekalskiy, S. Kokin, Development of a smart electricity meter for households based on existing infrastructure., in: IOP Conference Series: Earth and Environmental Science, volume 510, IOP Publishing, 2020, p. 022006.

[50] E. Cambiaso, G. Papaleo, M. Aiello, Slowcomm: Design, development and performance evaluation of a new slow dos attack, Journal of Information Security and Applications 35 (2017) 23–31.

[51] S. A. Butt, J. L. Diaz-Martinez, T. Jamal, A. Ali, E. De-La-Hoz-Franco, M. Shoaib, Iot smart health security threats, in: 2019 19th International Conference on Computational Science and Its Applications (ICCSA), IEEE, 2019, pp. 26–31.

[52] P. Kamble, A. Gawade, Digitalization of healthcare with iot and cryptographic encryption against dos attacks, in: 2019 International Conference on contemporary Computing and Informatics (IC3I), IEEE, 2019, pp. 69–73.

[53] C. Kolias, G. Kambourakis, A. Stavrou, J. Voas, Ddos in the iot: Mirai and other botnets, Computer 50 (2017) 80–84.