VOL 4 (2020) NO 2

e-ISSN : 2549-9904
ISSN : 2549-9610

INTERNATIONAL JOURNAL ON INFORMATICS VISUALIZATION

Security Improvement Mechanisms in Software-Defined Internet of Things

Hussaindad Saadat[#], Razieh AllamehZadeh[#], Sayed Akbar Mostafavi

[#] *Department of Computer Engineering, Yazd University, Yazd, Iran*

E-mail: a.mostafavi@yazd.ac.ir, Hussaindad.saadat@stu.yazd.ac.ir, Allamehzadeh@stu.yazd.ac.ir

***Abstract*— The IoT contains millions of heterogeneous smart objects that are connected together through the Internet platform. These heterogeneous smart objects deal with different protocols, technologies and resources, therefore each of them requires diverse security services in heterogeneous environments. Therefore, providing security services in heterogeneous environments is a daunting task for network providers that cannot be guaranteed through the traditional network architecture. Wide distribution and openness of IoT smart objects makes them very vulnerable to attacks and it can be easily targeted by cyber-attacks. Software-Defined Networking (SDN) is a new paradigm that separates the control plane from data plane t a global network view by centralized controller. Integrating the software-defined network with the Internet of Things can provide better access control and security mechanisms. Software-defined networking provides better control and management possibilities to manage and secure Internet of Things in a good manner. In this paper, we discuss about IoT architecture, security challenges in IoT, SDN architecture, security challenges in each layers of the SDN and software-defined IoT. In addition, we provide solutions to security problems in IoT through software-defined networking approach.**

***Keywords*— Software-Defined Networking, Internet-Of-Things, Security Mechanisms, Network Architecture**

I. Introduction

In Internet of Things (IoT) is a large number of smart devices that are connected together through the Internet platform. Openness and wide distribution of various smart devices which are located at different places cause big problems for network managers and providers to secure IoT network. IoT smart devices with different applications use diverse protocols where each protocol follow different access mechanisms a security measures. However unified security mechanism has not been implemented in IoT to guarantee security services in IoT networks[1].

Traditional security approaches such as intrusion detection systems and firewall are implemented at the border of the network to prevent external attacks. However IoT network encompass a large border which makes access control more difficult [2]. Therefore, traditional networking solutions cannot provide acceptable level of security through state-of-the-art mechanisms.

Software-defined networking as a new emerging technology which separates control plane from data plane to provide a global view of the network through logically centralized controllers [3]. With implementation of the SDN, we can manage and control network easily in a good manner by software-defined controller[4]. The SDN controller controls and manages all connected switches to the controller through OpenFlow channels. All instructions are generated by SDN controller and are sent to the OpenFlow-enabled switches through OpenFlow channels. According to generated rules by controller, switches will be employed as security, the Open Flow channel uses cryptographic and authentication mechanisms such as TLS. But this security services is not enough to protect OpenFlow channel against the security attacks. For example, an intruder can compromise TLS link through client certificate. In addition, due to limited resources of IoT devices, the common security techniques and protocols are not applicable in these networks. Therefore, cryptographic and authentication techniques cannot fully protect the IoT network. IoT architecture is comprised of a layering structure including perception layer, network layer, service layer and application layer. The Security vulnerabilities and challenges may impose the IoT network at risk in different layers.

In this paper, we provide a network-centric review of the current research activities for security of software-defined IoT. This paper is structured as follows. We first present the basic concepts for IoT and SDN including their architecture and main features. In section 3, the main IoT security challenges are presented and the state-of-the-art solutions are reviewed. , SDN architecture, IoT security framework based on SDN, software-defined security services for IoT and at the end we showed some research criteria in IoT networks based on SDN.

II. Background: IoT and SDN

A good definition for IoT will be such as: a world where smart devices connect through the Internet platform for exchange of information. All smart devices have its own identity for connection and have its own resources for consuming, for instance, sensors are connected to actuators rather information exchange between each other. These smart objects implement at different places and provide different services to human life[5].

Fig. 1 IoT Applications**[6]**

Today's Internet platform plays a key role in interconnecting of these smart devices through various communication technologies such as wired, wireless, adhoc etc. Each technology has its own protocol for connecting of heterogeneous smart devices. IoT structure is broadly sectioned into three layers but it can be four and five layers[7]. In this study, we consider just three layer architecture[8].

A. *IoT characteristics*

There are three important characteristics are needed to focus and prepare a great security techniques and mechanisms to prevent security vulnerability[9].

1) *Heterogeneity*

In the IoT, heterogeneity is the difference of numerous devices which are located at different places and have a lots of differentiation such as hardware performance (storage and central processing unit computation), protocols, platforms, policies, kind of usage etc. The biggest problem of the heterogeneity is absence of general security service.

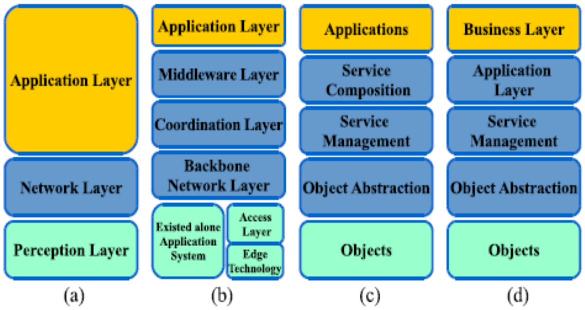

Fig. 2 IoT layers**[10]**

Heterogeneity is a weakness in IoT because interoperability is unreachable thus causes Extra cost about performance and money to interpret each other. Without preparing security services, creating security policies and updates are very complicated. We can use a variety of technologies to solve these problems, for instance those technologies are: (Meta data registry or MDR, middleware); however, it is not a complete way for solving all these challenges. One of the ways which works a little better between users and providers that are used based on standards. Some organizations should make standards and all other companies should obey instructions from standard organization. After that, most of the protocols and hardware characteristics will be the same therefore, this will be better for users and providers.

2) *Resource constraint*

Performance in IoT depends on the resources (e.g. sensors, actuators etc.)[11]. The legacy of security services are TLS, but for transport layer security using AES (advanced encryption standard) which cannot implement to the IoT devices directly. Therefore, these services or algorithms have to be designed lightweight to increase efficiency of CPU, storage and battery capacity of the IoT devices. Furthermore, scalability should also be considered until when changes happen in the network, all these architectures and resources should be responsible to unknown changes, Because of the low bandwidth and use of different devices multicast is preferred to better performance than unicast. Multicast is a little more flexible which confront to changes. CoAP (constrained application protocol) is another protocol which support multicast in RFC 7252.

3) *Dynamic Environments*

According to the mobility and weakness of the network connections, IoT has a dynamic network topology which trust on network connections without that there is no any other way[12]. For example in a smart city we can say there is a lot of requests which answer network. Therefore, flexibility and scalability are more important and crucial requirements in IoT communication protocols. According to the Cisco expectancy in 2020 IoT network will have 50billion devices, therefore, flexibility and scalability are so crucial requirements[13].

B. *Security requirements in IoT*

According to the complexity and heterogeneity of smart objects in IoT, authentication and authorization techniques are not implementable. In addition, according to the resource constraint stats in IoT operators can't use from complex security mechanism[14]. Some security challenges are described below:

1) *Object identification*

Object identification is a challenge in IoT network. Domain Name Systems which are used for translating name to IP and IP to name are vulnerable to attack, such as DNS cache poisoning attack, man in the middle attack. To overcome this challenge, IETF RFC 4033 implemented Domain Name Service Security Extension (DNSSEC) which is the new version of the DNS with a more security capabilities, but still it's not implemented because of high communication overhead.

2) Privacy and integrity

After data sensed from heterogeneous smart object, IoT requires to be collected and anonymized. Furthermore, sensed should be able to encrypt and decrypt in a good manner. Resource constraint smart objects aren't able to do such complex cryptographic operation thus aren't able to achieve privacy and integrity in IoT network.

3) Authentication and authorization

Public key (a pre-shared key which use for authentication and other goals in cryptographic ecosystem) exchange cryptosystem cannot work in IoT ecosystem. Thus, key management is so hard in IoT. Be without a global certification authority CA in the IoT is the main cause of the delaying. Furthermore, cryptographic algorithms are naturally heavy thus, require massive storage thus in the resource constraint smart objects it won't work.

C. Software-defined networking

Software-defined network is a new architecture which decupled data plane from control plane which can enable network control part to become programmable, because of abstracted from applications and network services[15]. Since SDN emerged in 2011, it was worked with OpenFlow protocol (a protocol which connect data plane to control plane in a remote purpose through a secure channel. Through this, protocol SDN controller controls all switches which are located at the data plane). After 2011 most of the companies started moving toward SDN and OpenFlow protocol. Google is a good instance which used from SDN and OpenFlow protocol to connect its data centers.

SDN was introduced in 2009, and originated from a lab job at Stanford University after SDN became a commercial. SDN which was first developed at Stanford University using HP switches and with the firmware upgrade, they were able to run an OpenFlow test network[16].

By using virtual layers, virtual switches, central controllers, communication standards and high level application interface attempt to control and management work on the switches and routers in the higher layers in software based. In a simple sentence, SDN reduces hardware dependency and enhance the network's software and intelligence capabilities[16][17].

1) SDN architecture

SDN architecture focused on four main points[18]:

- Separate the control layer from the data layer
- Centralized control and comprehensive view of the network.
- Having a functional relationship between the control and the data layer.
- Programmable capabilities with outside programs from network.

Through logically centralizing the network control plane and recommending programmability, SDN empowers security automation and run-time deployment of security procedures and policies. Network security systems leveraging from SDN can reply/respond to network abnormality and bad traffic conditions at run-time. To explain the functionality of the SDN architecture, the three main functional layers or SDN planes are showed in figure (3) and are formed of:

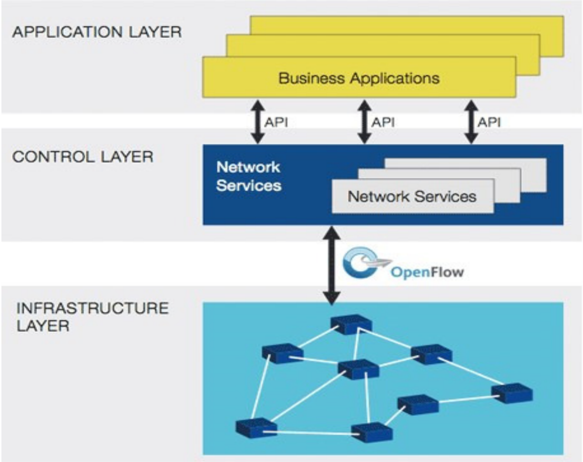

Figure 2: Software-defined Network Architecture**[19]**

2) Application plane

It consists different number of functionalities; like security services, network management and policy implementation. Also Application layer in SDN architecture consist of numerous network applications such as security virtualization etc. that communicate with controllers to apply abstract view of the network for internal decision-making processes. These applications communicate with controller through northbound application programming interface[20] [21].

3) Control plane

It is a logically centralized control infrastructure that runs the NOS, prepare hardware abstractions to SDN applications and maintain global view of the network. Also control plane, describe the functional components of SDN controller and its relation to other controllers and other administrative domains. SDN control layer consist of one or more SDN controllers to provide control functionality by address network behavior through OpenFlow protocol.

4) Data plane

It is the union of forwarding elements used to forward traffic flows based on instructions which control plane generated. Also data layer, is the undermost layer between SDN architecture layers. This layer consists of Network devices, such as switches, routers, access points etc. these devices are accessible through OpenFlow protocol by SDN controllers for managing network devices

Network security techniques can be applicable as applications in the application plane/ application layer. These applications or techniques achieve the network state/condition or resource information from the network control layer through the north bound interface. At the same through the control plane, security applications can collect samples of packets. After all process and security analysis, security system or security applications can redirect the traffic according to security policies and its level, through the control plane using south bound API. Unlike traditional network, SDN implemented security rules and policies as software modules rather than embedding them in the hardware, thus, allowing run time employment of security rules, policies and procedures[22].

Nonetheless, SDN has its own limitations, issues and challenges in terms of supportability, security and scalability. Between all of the challenges and issues, security is a major part which we should consider in a good manner. Until a centralized controller is responsible to our requests which manage the entire network, the security of the entire network depends on the controllers. If the central controller is compromised, all the entire network will be affected. In addition, a security slip on the communication path between control plane and data plane can pose a greater security threat in the entire network. At the other side, SDN enables applications to communicate with the control plane to have access with network resources, manipulate the network behavior, and implement new functionalities. Therefore by protecting SDN network from malicious programs is extremely difficult and overwhelming. Furthermore, network security is crucial for advancing technology and gaining user's satisfaction thus, SDN security is important and creating a secure SDN is very difficult[23].

III. Security Issues and Challenges in SDN and IoT

Suo et al. [24] studied about IoT and described security issues in IoT. Authors denoted about security features, requirements and security architecture in IoT. The researcher, described some issues in the layers of perception, network, support and application. All issues which these authors denoted consist of privacy, authentication, DDoS and encryption which according to the authors focus, can be so important and should be considered at the different layers.

Qiang et al. [25] researched on existing network security, based on that, a new security method for IoT was prepared and highlighted the problems in processing large amount of IoT data and ensuring reliability and security in this part and also mentioned about the need to solve security issues to avoid security risk on the application of IoT. Furthermore, authors described different security issues for instance wireless security, transmission security, privacy protection, and information security.

Jing et al. [26] considered security difficulties in each layer of IoT. Authors analyzed and compared between security issues in IoT and traditional networks. Furthermore, open security issues of IoT were analyzed in a deep manner. Also, authors discussed about cross layer heterogeneous integration and security issues very well. Researchers denoted about security issues of RFID technology, RSN technology and WSN technology were discussed and comparable solutions were offered. At the end security architecture for IoT system defined.

Zhao et al. [27] studied about three layer system structures and addressed numerous security issues of IoT, also recommended the solutions for security problems in each layer. In addition, demonstrated some general attacks in perception layer such as node capture, malicious data, timing attack, denial of service, reply attack and routing threats. Also, for solution and to prevent such from attacks implemented cryptographic algorithms and key management techniques. Furthermore, for resolving compatibility and cluster security problems, they used from WPK1, PK1 from key agreement mechanism. In addition, researchers demonstrated about general security problems in the application layer such as data access permission, identity authentication, data privacy and software vulnerabilities.

Kraijak et al. [28] discussed about architecture, security issues and protocols in IoT and explained about usable protocols, security and privacy issues in IoT applications. By usage of the Arduino device, implemented IoT system. Furthermore, future IoT is shown and trended in a good manner. In addition they focused on five layers of the IoT such as perception layer, network layer, middleware layer, application layer, and business layer. And also described about each layer functions in IoT system.

Matharu et al. [29] considered several challenges in IoT such as robustness in connectivity, interoperability, standardization, identity management, safety and security of objects, data confidentiality and encryption. In addition, authors described the common layer architecture in IoT. Furthermore, researchers discussed, analyzed and determined security issues in the four layers of the IoT architecture. At the end recommended strategies for solving security issues in IoT.

Said et al. [30] considered the research challenges and open problems at the IoT criteria. Furthermore, they introduced concept of IoT database and recommended IoT data base architecture also discussed about six layers IoT data base models such as IoT layer, data collection layer, data warehousing layer, event processing layer, data mining service layer and application layer. In addition they described the functions of each layers and said about future vision of the IoT. However, IoT has different layer according to the IoT architecture but the researchers focused on three layers architecture and five layers architecture in IoT. Also different challenges and open Problems in IoT were discussed.

Atamli et al. [31] described about three major entities and IoT features such as bad manufacturer, malicious and external adversary that pose risk to the privacy and security in IoT. They also discussed about security concerns for each IoT device such as sensors, actuators, RFID tags and network NFC. In addition, different security attacks were analyzed in IoT. The requirement to build a new security framework for IoT was proposed and security properties was emphasized such as tamper resistant, protected storage and access control, data exchange, identification, authentication. The availability is required to ensure confidentiality and integrity of the system, which privacy properties prevent revealing information about users and devices.

Granjal et al. [32] surveyed existing protocols in IoT and described different security issues in IoT. Existing protocols were analyzed to present security communication between IoT devices. Numerous existing protocols investigated to enable security in physical (PHY), Medium Access Control (MAC) layers low energy communications, network layer, routing and application layer with CoAP. Possible methods to offer noble security mechanisms were prepared based on security requirements.

A. SDN security challenges and issues

Security has been a terrific task in communication networks according to the underlying network complexities and property security solutions that are complex to manage the weak concept of identity in IP networks. At the same, the

Internet architecture that defines procedures for usage of the underlying infrastructure[33]. Authors considered different ways for security services and security challenges in SDN. In this part, about security in SDN will be discussed and described some methods which authors proposed about security services and challenges in SDN.

Separation of different planes and integrating the control plane operations to a centralized system such as OpenFlow controller can be a base to future networks; nonetheless all of these innovations open new security challenges and new security problems. For instance, communication channels between different planes can be a good place for hackers and communication channels provide a good opportunity for hackers which test their attacks. Due to the existence of centralized controller, the control plane can be a crucial and a significant point for attackers. Thus, the control plane is more attractive to security attacks and especially to DoS and DDoS attacks because of its visible nature. The SDN controller can become a single point of failure and thus single point vulnerability can affect the whole network. Network resource visibility is important in SDN[12]. Thus, these resources must not be visible to all or unconcerned applications. When we want to use from SDN technologies, we must consider the challenges that lie ahead. Therefore, in this section we discussed about security challenges[34] [35]. From a basic point of view, security vulnerabilities in SDNs are focused on these three areas:

- Applications
- Control plane
- Data plane

Therefore the security challenges have been described at different layers of the SDN.

1) Security challenges in application plane

SDN has two basics characteristics, these two characteristics will present challenges and strategies. First software-based network control capability, second centralized network intelligence through central controllers[36]. Before, most of the network functions can be executed as SDN applications, if malicious programs don't stop sooner, it will bring many problems to the network. Because of this, in this section application of plane security challenges have been considered.

Until there is not a comprehensive standard to facilitate open APIs for application to control network services, operations, functions, through the control plane, applications can cause a lot of damages to the network resources, services and functions[36]. Whereas, OpenFlow enables deploying flow based on security detection algorithms in the form of security applications, there are no constraining OpenFlow security applications[37]. The variety of vendor and third party applications developed in particular independent development conditions using numerous programming models and paradigms could create interoperability constraints and security policy crash. We will point to some of these challenges:

Authentication and Authorization

Authentication and identifying identity in today's software and finding solutions is today's major software challenges. Application in OpenFlow inherit the privileges for access to network resources, without proper security mechanisms for protecting network resources from malicious activities[38]. Therefore, authentication of the increasing number of applications in programmable networks with centralized control network architecture is a crucial security challenge.

Diego and et al. [36] considered threat vectors to describe security vulnerabilities in SDN. Authors, denoted that there are no constraining mechanisms to establish trusted relationship between applications and controllers in SDN. Therefore, malicious programs can introduce bugs into the SDN. In addition, numerous techniques exist to certify network devices in a network. But there are no mechanisms to certify network applications. A centralized system to certify SDN applications is needed.

Accountability and Access control

Hence applications deploy most of the services in the SDN, better access control and accountability mechanisms are required to ensure the security of a network. The following example refers to access control and accountability in SDN.

Hartman and et al. [39] identify three categories of applications that is capable of influencing the network security in SDN. a) Network sensitive applications that need specific network characteristics such as path characteristics, traffic flow cost etc. b) Applications that prepare services for the network such as firewall, intrusion detection, access control etc. c) packaged network services that consist applications from a and b categories. For instance in [40] stating the applications in SDN can be either SDN-aware or unaware of SDNs. SDN- aware applications are capable of directly communicating with SDN controllers at the confront SDN-unaware applications communicate indirectly with application data grams in specific format.

2) Security challenges in control plane

In the SDN architecture, the control plane is the crucial and centralized decision making entity. Therefore, the SDN controller is more attentive to attack because if it carries SDN controller, it can assume overall network management. Here are some of the challenges will be faced:

Threats from applications

The deployed applications on top of the control plane can present serious security threats to the control plane. Commonly, the controller security is a challenge from the view of controller capability to authenticate applications and authorize resource used by applications with suitable isolation, auditing and tracking[40]. We need to separate applications to meet different application security requirements, before access to network, information and resources are provided. Different application has its own functional requirements from the underlying controller and data path must qualify different security requirements. For instance, participatory networking discussed in[4] which enable users and their applications to participate in network configuration. These kinds of users and applications should check before access to the network. In terms of privileges, these user applications have less privileges than vendor applications. Hence, a customized security mechanism for different types of application is needed in the north bound API of the controller.

Threats due to Scalability

Most of the complexity pushed towards controller whereby decisions are taken in a logically centralized manner[41]. If there is a need for the controller to make the rules, the controller comes the bottleneck in the network. The researchers in [42] described that nowadays, SDN controllers implementations are not capable to respond to large number of new flows when using from OpenFlow in high speed networks such links with 10Gbps. Also described in [43] with the presence of network scalability, SDN has weaker security function than the traditional network, thus, as the number of controllers increases, security becomes more difficult. Therefore, controller scalability makes it a favorite choice for DoS (Denial of Service) and DDoS (Distributed Denial of service) attacks.

DoS attacks

DoS and DDoS attacks occur more frequently on the controllers and hence these attacks can also cause stains especially the single central controller that is more compromise-able against such attacks. DoS and DDoS attacks focus on resources and prevent the provision of services to users. A DoS attack considered by authors in [44] to a network scanning tool is developed that can identify a particular new flow, there is a difference in flow response time for new flow and existing flows. The scanner achieve the time values with the help of header field. Increasing the number of flows in the data-path will make the switches bombard flow setup requests on the controller and therefore, controller will break in long time. Also a DoS attack on the SDN controller denoted in [45] which an attacker periodically sends IP packets with random headers to become SDN controller in non-responsive state.

3) Security challenges in Data Plane

The switches have tables in the network and the controller is responsible for embedding the rules in the tables. All of these flow rules can be installed before a user sends packets to the network(proactive rule installation method) or it can be when the first packet from a new user sends(reactive installation method). Unless the rules are set for switches, it is impossible to decide as for such the rules must be given to switches to perform their operation. So, where does it turn out to be those rules are real with SDN controller generated? These rules may be malicious and continuously will be compromised network. It can be a security challenge in SDN and another challenge is buffer size in SDN switches and number of flow entries a switch can maintain. This could be weak against attacks that fill the flow rules table.

Different networks have their particular advantages and disadvantages, as mentioned in SDN security, the SDN also has its strengths and weakness. Referring to the advantages of the SDN, SDN can solve resource management problems in IoT due to the following reasons. Firstly, separation of data plane and control plane by differentiating the services between them. Also separation of the data plane and control plane support to abstract low level network functionalities. Besides separation, single point of view will help to control and manage network functionalities in a good manner. However, this failure could be solved using replication format [46-49].

One of the advantages of the SDN is programmability which let us to provide dynamic and fast creation of new network services. Furthermore, OpenFlow protocol is an open source protocol which is the key element for SDN architecture [50-51]. OpenFlow protocol let the controllers in the SDN, to determine flow paths in a network of OpenFlow enabled switches, because of this reason OpenFlow provide easy traffic management through data plane from control plane. In SDN architecture, SDN controllers control all switches through OpenFlow channels. Controller in SDN architecture is the generator of the rules in the network. Most of the action done by generated rules which controller generated. OpenFlow enabled switches Communicate with controller through OpenFlow channel. Thus, security and reliability of the OpenFlow channels which connect data plane and control plane are so important for operation, configuration and management in SDN architecture. Through OpenFlow protocol, a SDN controller can bring changes at the data plane such as update, delete, add, subtract, flow entries (consist of reactive and proactive). For reactive: when a packet/flow arrived at the OpenFlow enabled switch and will check the flow table, if matched it will forward, otherwise send to the controller for making a decision for such packets/flow. Controller will generate rule and send to the OpenFlow enabled switch through OpenFlow channel. It is possible that an attacker uses from this opportunity and send numerous packets which does not exist in the flow table, because of table miss these packet send to the controller upon which in a long time will become down or do not work so correctly [52]. For prevention from such attack proactive method is better. Proactively flow entry method is the action which before start flow transmission, manager add some default paths until prevent from denial of service attack on controller; at the first time when a user send its packets, it will check in the flow table in OpenFlow enabled switch, some paths defined thus, new entry flow don’t have table miss therefore, forward to the next hop until fine the best path for such flows. It will prevent from sending numerous of unmatched new flows.

In software-defined IoT, controller requires to receive state information from heterogeneous network. In IoT network most of the communications are time sensitive and real time exchanging information thus, there should be provision until reduce collection overhead. In IoT most of the focused topics are Delay, packet loss, jitter and throughput [53].

SDN controller cannot manage all IoT networks but it is possible to monitor all IoT networks, such as incoming and outgoing packets which are exchanging in the IoT network. SDN controller is capable to efficiently overcome security attack at the both side (inside and outside) [54-56]. A lot of researchers worked on IoT security by implementing IPS(Intrusion prevention system), firewalls and IDS(Intrusion Detection System) through SDN controller[57]. OpenFlow enabled switches are the crucial element which all security rules should install on them. However it is not implemented until now[58].

centralized controller will be used in IoT network in which much researcher worked on this[59]. The most important challenge is the occurrence of DDoS attack on centralized controller. Furthermore, it is possible for user attack the controller inside, if this attack will be succeeded then single point of failure will occur [60-61].

For solving DDoS attack and single point of failure proposed multi SDN controller. Multi controller will be more fault tolerance and trustable. For instance, when a SDN controller failed, another one will be responsible for all requests. When there is multi controller, performance will decrease. Each controller can control and able to have partial network view. But it will have overhead, because multi controller should exchange information between each other.

IV. Conclusion

The Millions of heterogeneous smart devices connected together through Internet platform and created IoT. These smart devices have their characteristics in the network and protocols. These smart objects connected together are connected through various communication technologies. According to the usage of different technologies and protocols, different security services required for secure heterogeneous smart objects. Providing security services for these smart objects are so crucial and also very hard. Traditional network cannot secure such network and technology in a good manner. The main problem in IoT was openness and wide distribution thus it is difficult to provide security services. Software-defined network is a new architecture which decoupled control plane from data plane and have a global network view by centralized controller. Software-defined network architecture can monitor all incoming packets and outgoing packets very well, thus, have a global view from IoT network. Furthermore, SDN architecture can implement IPS, firewalls and IDS for secure IoT network. In this study general feature of IoT, IoT architecture, security challenges in IoT, software-defined network, software-defined network architecture, security challenges in each layers of the software-defined network and IoT based software-defined network are discussed. Furthermore solutions to security problems through software-defined networking is clearly identified**.**

References

[1] A. C. Sarma and J. Girão, “Identities in the future internet of things,” Wirel. Pers. Commun., vol. 49, no. 3, pp. 353–363, 2009.

[2] S. Rizvi, A. Kurtz, J. Pfeffer, and M. Rizvi, “Securing the Internet of Things (IoT): A Security Taxonomy for IoT,” Proc. - 17th IEEE Int. Conf. Trust. Secur. Priv. Comput. Commun. 12th IEEE Int. Conf. Big Data Sci. Eng. Trust. 2018, no. August, pp. 163–168, 2018.

[3] M. Liyanage, A. Gurtov, and M. Ylianttila, “SoftwareDefined Mobile Networks (SDMN): Beyond LTE Network Architecture,” Softw. Defin. Mob. Networks Concepts Challenges, pp. 1–390, 2015.

[4] A. D. Ferguson, A. Guha, C. Liang, R. Fonseca, and S. Krishnamurthi, “Participatory networking,” ACM SIGCOMM Comput. Commun. Rev., vol. 43, no. 4, pp. 327–338, 2013.

[5] H. Elhammouti, E. Sabir, M. Benjillali, L. Echabbi, and H. Tembine, “Self-Organized Connected Objects: Rethinking QoS Provisioning for IoT Services,” IEEE Commun. Mag., vol. 55, no. 9, pp. 41–47, 2017.

[6] A. Kingatua, “Top 10 IoT Applications,” Electronics, Information & Communications Technology. [Online]. Available: https://electronicsandict.com/top-10-iot-applications/.

[7] L. Da Xu, W. He, and S. Li, “Internet of things in industries: A survey,” IEEE Trans. Ind. Informatics, vol. 10, no. 4, pp. 2233–2243, 2014.

[8] I. Yaqoob et al., “Internet of Things Architecture: Recent Advances, Taxonomy, Requirements, and Open Challenges,” IEEE Wirel. Commun., vol. 24, no. 3, pp. 10–16, 2017.

[9] A. Haroon, M. Ali, Y. Asim, W. Naeem, M. Kamran, and Q. Javaid, “Constraints in the IoT: The World in 2020 and Beyond,” Int. J. Adv. Comput. Sci. Appl., vol. 7, no. 11, 2016.

[10] V. Schmidt, “Impact Analysis of the Internet of Things on the Value Chain in Manufacturing Industries,” no. July, 2016.

[11] M. M. J. Krishnamurthy, “Constrained Device,” Science Direct, 2016. [Online]. Available: https://www.sciencedirect.com/topics/computer-science/constrained-device.

[12] B. Cheng, M. Wang, S. Zhao, Z. Zhai, D. Zhu, and J. Chen, “Situation-Aware Dynamic Service Coordination in an IoT Environment,” IEEE/ACM Trans. Netw., vol. 25, no. 4, pp. 2082–2095, 2017.

[13] B. R. Srinivasan, “Internet of Things in Smart Cities,” 2014.

[14] M. Gloukhovtsev, “Iot Security : Challenges , Solutions & Future Prospects,” 2018.

[15] S. Mostafavi and M.A. Dawlatnazar and F. Paydar, "Edge Computing for IoT: Challenges and Solutions", Journal of Communications Technology, Electronics and Computer Science, Vol. 25, pp. 5-8, 2019.

[16] F. Chahlaoui, M. Raiss El-Fenni, and H. Dahmouni, “Performance analysis of load balancing mechanisms in SDN networks,” ACM Int. Conf. Proceeding Ser., vol. Part F1481, 2019.

[17] B. Darabinejad, “An Introduction to Software-Defined Networking,” Int. J. Intell. Inf. Syst., vol. 3, no. 6, p. 71, 2014.

[18] O. Akpovi A., E. Seun, A. A. O., and O. F. Y., “Introduction to Software Defined Networks (SDN),” Int. J. Appl. Inf. Syst., vol. 11, no. 7, pp. 10–14, 2016.

[19] W. Braun and M. Menth, “Software-Defined Networking Using OpenFlow: Protocols, Applications and Architectural Design Choices,” Futur. Internet, vol. 6, no. 2, pp. 302–336, 2014.

[20] D. L. Love, “Wireless Evolution, Software Defined Networks and Network Function Virtualization: Enablers of IoT,” Stanford University. [Online]. Available: https://mse238blog.stanford.edu/2017/07/dllove/software-defined-networks-and-network-function-virtualization-precursors-to-5g/.

[21] ONF, “SDN Architecture (TR-521),” ONF White Pap., no. 1.1, 2016.

[22] ONF, “OpenFlow Switch Specification 1.4.0,” Current, vol. 0, pp. 1–3205, 2013.

[23] O. Oladunjoye, “SOftware Defined Networking– The Emerging Paradigm To Computer Networking,” p. 38, 2017.

[24] K. Raghunath and P. Krishnan, “Towards A Secure SDN Architecture,” 2018 9th Int. Conf. Comput. Commun. Netw. Technol. ICCCNT 2018, no. July 2018, 2018.

[25] H. Suo, J. Wan, C. Zou, and J. Liu, “Security in the internet of things: A review,” Proc. - 2012 Int. Conf. Comput. Sci. Electron. Eng. ICCSEE 2012, vol. 3, pp. 648–651, 2012.

[26] X. F. Wang, “Research on security issues of the internet of things,” Adv. Mater. Res., vol. 989–994, no. 6, pp. 4261–4264, 2014.

[27] Q. Jing, A. V. Vasilakos, J. Wan, J. Lu, and D. Qiu, “Security of the Internet of Things: perspectives and challenges,” Wirel. Networks, vol. 20, no. 8, pp. 2481–2501, 2014.

[28] K. Zhao and L. Ge, “A survey on the internet of things security,” Proc. - 9th Int. Conf. Comput. Intell. Secur. CIS 2013, pp. 663–667, 2013.

[29] S. Kraijak and P. Tuwanut, “A survey on internet of things architecture, protocols, possible applications, security, privacy, real-world implementation and future trends,” Int. Conf. Commun. Technol. Proceedings, ICCT, vol. 2016-Febru, pp. 26–31, 2016.

[30] G. S. Matharu, P. Upadhyay, and L. Chaudhary, “The Internet of Things: Challenges & security issues,” Proc. - 2014 Int. Conf. Emerg. Technol. ICET 2014, pp. 54–59, 2014.

[31] O. Said and M. Masud, “Towards internet of things: Survey and future vision,” Int. J. Comput. Networks, vol. 5, no. 1, pp. 1–17, 2013.

[32] A. W. Atamli and A. Martin, “Threat-based security analysis for the internet of things,” Proc. - 2014 Int. Work. Secur. Internet Things, SIoT 2014, pp. 35–43, 2014.

[33] J. Granjal, E. Monteiro, and J. Sa Silva, “Security for the internet of things: A survey of existing protocols and open research issues,” IEEE Commun. Surv. Tutorials, vol. 17, no. 3, pp. 1294–1312, 2015.

[34] D. Clark et al., “New Arch : Future Generation Internet Architecture,” Security, vol. 0553, no. August, 2004.

[35] A. S. Mustafa, D. Mkpanam, and A. Abdullahi, “Security in Software Defined Networks (SDN): Challenges and Research Opportunities for Nigeria.,” Int. J. Comput. Appl. Technol. Res., vol. 7, no. 8, pp. 297–300, 2018.

[36] P. Joshi, “Software-Defined-Networks-Security-An-Analysis-of-Issues-and-Solutions.docx,” Int. J. Sci. Eng. Res., vol. 6, no. 5, pp. 1270–1275, 2015.

[37] D. Kreutz, F. M. V. Ramos, and P. Verissimo, “Towards secure and dependable software-defined networks,” HotSDN 2013 - Proc. 2013 ACM SIGCOMM Work. Hot Top. Softw. Defin. Netw., pp. 55–60, 2013.

[38] S. Shin, P. Porras, V. Yegneswaran, and G. Gu, “A Framework For Integrating Security Services into Software-Defined Networks,” Proc. 2013, vol. 1, no. 1, pp. 11–12, 2013.

[39] S. Mostafavi and M. Dehghan, "Decentralized Adaptive Helper Selection in Multi-channel P2P Streaming Systems," IEEE 34th International Conference on Distributed Computing Systems Workshops (ICDCSW), Madrid, 2014, pp. 1-7

[40] S. Mostafavi and V. Hakami, "A new rank-order clustering algorithm for prolonging the lifetime of wireless sensor networks", International Journal of Communication Systems, 2019.

[41] S. Mostafavi and W. Shafik, "Fog Computing Architectures, Security and Privacy", Journal of Communications Technology, Electronics and Computer Science, Vol. 26, pp. 1-9, 2019.

[42] X. Wen, Y. Chen, C. Hu, C. Shi, and Y. Wang, “Towards a secure controller platform for OpenFlow applications,” HotSDN 2013 - Proc. 2013 ACM SIGCOMM Work. Hot Top. Softw. Defin. Netw., pp. 171–172, 2013.

[43] M. W. and D. Z. S. Hartman, “Software driven networks problem statement,” Network Working Group Internet-Draft. [Online]. Available: https://tools.ietf.org/html/drafthartman-%0Asdnsec-requirements-00.

[44] V. G. H. Xie, T. Tsou, D. Lopez, H. Yin, “Use cases for ALTO with software defined networks,” Working Draft, IETF Secretariat, Internet-Draft, 2012. [Online]. Available: https://tools.ietf.org/%0Ahtml/draft-xie-alto-sdn-use-cases-01.

[45] S. Mostafavi, M. Dehghan, "Game-theoretic Bandwidth Procurement Mechanisms in Live P2P Streaming Systems", Multimedia Tools and Applications, vol. 75, no. 14, pp. 8545-8568, 2016.

[46] S. Mostafavi, M. Dehghan, "Game-theoretic Auction Design for Bandwidth Sharing in Helper-assisted P2P Streaming", International Journal of Communication Systems, vol. 29, no. 6, pp. 1057-1072, 2016.

[47] M. Sanaei and S. Mostafavi, "Multimedia Delivery Techniques over Software-Defined Networks: A Survey," 5th International Conference on Web Research (ICWR), 2019, pp. 105-110.

[48] S. Mostafavi, W. Shafik, “Fog Computing Architectures, Privacy and Security Solutions”, Journal of Communications Technology, Electronics and Computer Science, Vol. 24, pp. 1-14.

[49] J. Naous, D. Erickson, G. A. Covington, G. Appenzeller, and N. McKeown, “Implementing an OpenFlow switch on the NetFPGA platform,” Proc. 4th ACM/IEEE Symp. Archit. Netw. Commun. Syst. ANCS ’08, pp. 1–9, 2008.

[50] M. Jarschel, S. Oechsner, D. Schlosser, R. Pries, S. Goll, and P. Tran-Gia, “Modeling and performance evaluation of an OpenFlow architecture,” Proc. 2011 23rd Int. Teletraffic Congr. ITC 2011, pp. 1–7, 2011.

[51] M. Douglass, “Endometrial tumors in abdominal scars,” J. Am. Med. Assoc., vol. 90, no. 23, pp. 1853–1856, 1928.

[52] S. Shin and G. Gu, “Attacking software-defined networks,” p. 165, 2013.

[53] P. Fonseca, R. Bennesby, E. Mota, and A. Passito, “A replication component for resilient OpenFlow-based networking,” Proc. 2012 IEEE Netw. Oper. Manag. Symp. NOMS 2012, pp. 933–939, 2012.

[54] H. Hu, W. Han, G. J. Ahn, and Z. Zhao, “FLOWGUARD: Building robust firewalls for software-defined networks,” HotSDN 2014 - Proc. ACM SIGCOMM 2014 Work. Hot Top. Softw. Defin. Netw., pp. 97–102, 2014.

[55] S. Scott-Hayward, G. O’Callaghan, and S. Sezer, “SDN security: A survey,” SDN4FNS 2013 - 2013 Work. Softw. Defin. Networks Futur. Networks Serv., 2013.

[56] S. Son, S. Shin, V. Yegneswaran, P. Porras, and G. Gu, “Model checking invariant security properties in OpenFlow,” IEEE Int. Conf. Commun., pp. 1974–1979, 2013.

[57] S. Mostafavi, M. Dehghan, "A Stochastic Approximation Resource Allocation Approach for HD Live Streaming", Telecommunication Systems, vol. 64, no. 1, pp.

[58] S. Mostafavi, M. Dehghan, "Optimal visual sensor placement for coverage based on target location profile", Ad Hoc Networks, vol. 9, no. 4, pp. 528-541, 2011.

[59] B. A. A. Nunes, M. Mendonca, X. N. Nguyen, K. Obraczka, and T. Turletti, “A survey of software-defined networking: Past, present, and future of programmable networks,” IEEE Commun. Surv. Tutorials, vol. 16, no. 3, pp. 1617–1634, 2014.

[60] R. Skowyra, S. Bahargam, and A. Bestavros, “Software-Defined IDS for securing embedded mobile devices,” 2013 IEEE High Perform. Extrem. Comput. Conf. HPEC 2013, 2013.

[61] N. McKeown et al., “OpenFlow: enabling innovation in campus networks,” ACM SIGCOMM Comput. Commun. Rev., vol. 38, no. 2, p. 69, 2008.

[62] S. Mostafavi, V. Hakami, "A new rank-order clustering algorithm for prolonging the lifetime of wireless sensor networks", International Journal of Communication Systems, vol. 33, https://doi.org/10.1002/dac.4313, 2020.

[63] S. Mostafavi, V. Hakami, "A stochastic approximation approach for foresighted task scheduling in cloud computing", Wireless Personal Communications, https://doi.org/10.1007/s11277-020-07398-9, 2020.

[64] S. Mostafavi, V. Hakami, F. Paydar, "Performance Evaluation of Software-Defined Networking Controllers: A Comparative Study", Journal of Computer and Knowledge Engineering, 2020.

[65] S. Mostafavi, V. Hakami, F. Paydar, "A QoS-Assured and Mobility-Aware Routing Protocol for MANETS", JOIV: International Journal on Informatics Visualization, vol. 4, no. 1, pp. 1-9, 2020.