Computer Communications 122 (2018) 1–8

Contents lists available at ScienceDirect

Computer Communications

journal homepage: www.elsevier.com/locate/comcom

OpCloudSec: Open cloud software defined wireless network security for the Internet of Things

Pradip Kumar Sharma, Saurabh Singh, Jong Hyuk Park*

Department of Computer Science and Engineering, Seoul National University of Science and Technology (SeoulTech), Seoul 01811, Republic of Korea

ARTICLE INFO

Keywords:
Internet-of-Things
Software Defined Networking
Security
Deep learning

ABSTRACT

Cutting-edge cloud frameworks will require a paradigm shift in regards to how they are built and managed. Traditional management and control platforms face significant challenges in terms of security, reliability, and flexibility that these cutting-edge frameworks must deal with. On the other hand, Distributed Denial of Service (DDoS) attacks have become a weapon of choice for cyber-terrorists, cyber-extortionists, and hackers. Recently, the simplicity of programmability in Software-Defined Networking (SDN) makes it a good platform for the implementation of various initiatives that includes decentralized network management, dynamic topology changes, and application deployment in a multi-tenant data center environment. Motivated by the capabilities of SDN, we are proposing a mitigation architecture for security attacks that incorporates a highly programmable monitoring network so as to make it possible to identify attacks. It has a flexible control structure to quickly define the reaction of attacks and particular side, and we show how SDN can be used as a key application in the cloud IoT. We evaluated the performance of our proposed architecture and compared it with the existing models to obtain various performance measures. The results of our evaluation show that our OpCloudSec architecture model can efficiently and effectively meet the security challenges created by the new network paradigm.

1. Introduction

Recently, cloud computing has emerged as one of the most influential paradigms due to its essential features, such as rapid elasticity, the pooling of resources, on-demand self-service, and the provision of measured service for industry and academic research. Lower spending and capital expenditures, increased operational efficiency, adaptability, flexibility, and scalability are benefits of cloud computing. While the overall benefits of the cloud computing paradigm are exciting for academic researchers and IT industries, cloud computing security issues are becoming serious barriers that without being adequately addressed will limit the applications and use of cloud computing in the future. Network security is considered one of the major security concerns in cloud computing and poses the same threat as privacy disclosure and data security [1,2]. A particular concern is the security of the Internet of Things (IoT), as it includes all objects or devices, such as medical devices, home and industry sensors, cars, and even nuclear reactors with network capabilities. All of which can pose risks to human life [3].

Recently, as a new paradigm for networking and a transformative approach for network design and implementation, SDN has aroused great interest [4]. Due the provision of programmable network interfaces for deployments to multi-tenant data centers, SDN uses an ideal platform that requires dynamism and flexibility. This is especially true in an Infrastructure-as-a-Service (IaaS) cloud where tenants seeking financial and technological flexibility, are managed Virtual Machines (VMs). In SDN, network state and intelligence are logically centralized and the underlying network infrastructure is abstract from applications by decoupling the data and control plans [5]. Due to the controller's potential bottlenecks, the decoupling of control and data plans in SDN leads to scalability problems. To provide networking-as-a-service and to take control of the network infrastructure in cloud computing environments, the SDN based cloud is a new type of cloud [6].

In the development of cloud computing, security has been seen as the main barrier [7]. To overcome attacks in cloud computing environments, SDN features offer new opportunities [4]. Availability is one of the vital security requirements for cloud computing in order to make it possible to offer on-demand services of different levels [7]. In the network and cloud infrastructure, DDoS attacks have been a real threat [8]. There could be various purposes behind DDoS attacks being launched, such as political gains, financial gains, disruption, and potentially causing massive disruptions to the cloud infrastructure. With illegitimate traffic, DDoS attacks overwhelm network devices, network links, and servers and can disrupt networks and services. All of this leads to a complete denial of service or degradation of service and

* Corresponding author.
E-mail addresses: pradip@seoultech.ac.kr (P.K. Sharma), singh1989@seoultech.ac.kr (S. Singh), jhpark1@seoultech.ac.kr (J.H. Park).

https://doi.org/10.1016/j.comcom.2018.03.008
Received 3 April 2017; Received in revised form 8 September 2017; Accepted 11 March 2018
Available online 12 March 2018

results in huge losses. The expansion of dependency on the Internet and data centers has hampered this issue. In SDN, DDoS exists and an attacker can leverage SDN's features to initiate DDoS attacks against SDN applications, data plans, and control plan layers [9].

In this paper, we present the integration of SDN and cloud computing, which are applied to the IoT environment for efficient and better security management. First, we studied the impact of the integration of SDN and cloud computing on DDoS and other attack defense mechanisms. We found that when designed correctly, SDN can be used to address the safety concerns raised by cloud computing and defending against attacks, which can be made more effective and efficient in the era of SDN and cloud computing. Based on our research,we are proposing a new architecture for reducing attacks using the software defined abbreviated network, OpCloudSec, to demonstrate and prove our conclusions.

The rest of this paper is comprised of the following sections: In Section 2, the challenges and security issues in IoT on cloud and security attack defense impact on cloud computing are discussed. OpCloudSec architecture is proposed in Section 3, and Section 4 provides a description of our results. Lastly, we present our conclusions in Section 5.

2. Existing research and analysis

2.1. Challenges and security issues in IoT for cloud computing

Developers need to prevent physical and network attacks by both internal and external malicious attackers that are accomplished through a design that consists of both device security infrastructure and architecture [10]. As the number of connected devices in the IoT environment increases, as shown in Fig. 1, the data privacy is still a major issue. Thus, the secure end-to-end security architecture should be addressed to data protection and the key through all the IoT devices [11].

2.1.1. Data leakage protection and monitoring issues

In both public and private clouds, data that has been stored in IaaS infrastructure needs to be carefully monitored [2]. The convergence of IoT in the cloud brings up some issues in regards to the protection of data such as homeowners' concerns about data breaches of sensitive and personal information. The loss of privacy and trust and the direct effect on the Service Level Agreement (SLA) are cloud users' primary concerns. The leakage of data affects web applications and then attackers can take advantage of configured permissions in cloud implementations [12].

2.1.2. Logical network segmentation

To reduce the latency and damage of unpredictable disasters and provide availability with good performance the cloud infrastructure spans multiple geographical sites. IoT technology utilizes the virtualization concept of IaaS in the cloud which is vulnerable to DDoS, MitM, and IP Spoofing [13]. A network administrator must choose the best and most secure connection system that monitors the traffic packet being sent between the IoT device and cloud infrastructure.

2.1.3. Protecting and managing devices

In the context of IoT devices, each device boots and runs some codes when powering up. It is crucial that the device only does what it is programmed to do and that it cannot be programmed to behave maliciously. A protection device against attacks requires code signing to ensure that all codes are allowed to run and that there is run-time protection against malicious attacks that overwrite the code after loading [14].

2.1.4. Protecting network traffic communication

The IoT paradigm is rapidly gaining momentum in wireless communication. The idea behind this concept is the pervasive and ubiquitous deployment of wireless devices, such as sensors, actuators, and Radio Frequency Identification (RFID). The challenges that need to be addressed deal with the identification of devices, providing security, the mobility of wireless devices, and ensuring the quality of secure communication between devices [15]. In the convergence of the IoT and cloud, there is the lack of a standard framework for an IoT protocol stack for incompatible products and applications. The protocol communication must have a security feature, a high-efficiency feature, and localization measurement capabilities [16].

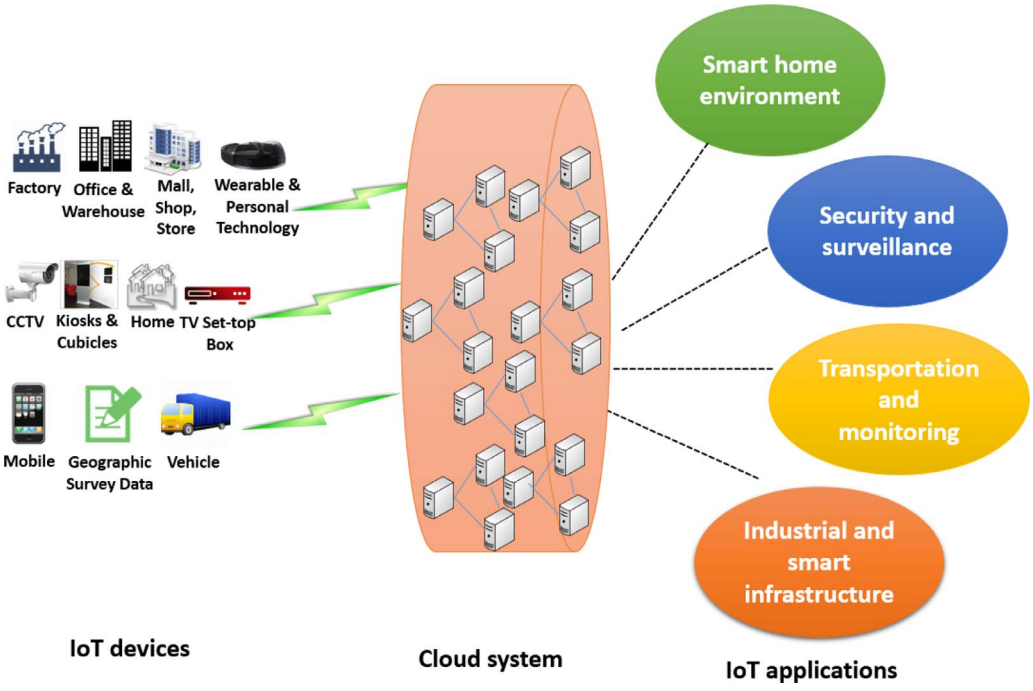

Fig. 1. IoT environment in cloud.

2.1.5. Infrastructure hardening issue

VMs and VM templates are harder to clean in the primary system when an image is being created and want to take advantage of this technology and update it with least service of security updates. IaaS includes the unified infrastructure resource stack from facilities to their hardware platforms. The requirement is to improve data encryption, OS capabilities, and network management between the IOT and cloud infrastructure [17].

2.2. Security attack defense impact on cloud computing

2.2.1. DoS attack

In a DoS attack, an adversary gains control of a tenant's VM and makes another's web server unavailable. In the context of cloud computing, DoS attacks and their characterization changes, which affects the victim server. In 2015, a report by VeriSign Defense Security Intelligence Services (Tara Seals 2015) [18] pointed out that most of the targets for DDoS attacks are cloud and SaaS [19]. A SYN Flood DoS attack exploits the flaws in a TCP three-way handshake procedure in which the attacker sends the SYN flood series from the spoofed IP address so that the server will assign all the needed resources and wait for an ACK packet from the client that will never arrive [20]. In UDP, the Flood attacker sends a lot of UDP data packets to a random port of the target system using zombies. There are many types of DoS attacks that remain in the cloud network system [21].

2.2.2. DNS reflection

A DNS Reflection attack takes advantage of the UDP source address, the availability of open resolvers, and the asymmetry of DNS requests and responses. It is a type of amplification attack on the internet. Basically, in this attack, the adversary sends a set of DNS queries to multiple DNS servers with a spoofed IP address to open resolvers. In a cloud system, it has its own DNS server to respond to the DNS queries from tenants and there should not be any response from the internet to the cloud by the DNS server. Therefore, any inbound DNS response may indicate a potential DNS reflection attack [19].

2.2.3. SDN defense mechanism against network attacks

Although SDN looks like a promising technology for future networking innovations, it is still facing security issues that need to be solved. Regardless, the solution should be scalable, efficient, and secure when defending or protecting the controller and establishing trust between devices, or a robust policy framework should be created to check and balance the SDN controller that has had a task assigned to it [22–24].

Some of the ongoing security projects to secure the SDN as Rosemary employ several strategies to build reliable and robust high-performance technologies. It focuses small and basic streams in network application can cause crushing of the control plan and operate network functionality.

AVANT-GUARD: SDN is associated with the OpenFlow protocol, which is susceptible to control plane saturation attacks because it has the issue of limited network monitoring that exploits bottlenecks that occur between the control plane and data plane [25].

FRESCO encounters the difficulties of securing the SDN because it requires complex logical implementation. As such, FRESCO provides an efficient, effective, and secure programming framework to implement various security detection techniques [26].

2.3. Existing research

Yanbing et al. [27] introduced a multi-layered-based software defined security architecture to allow business software vendors and security developers to work only in their area of expertise without worrying about the design and implementation of the development of business logic programs or security structures. By using an interface for network security functions, Kim et al. [28] presented a framework for creating a stand-alone network attack protection system capable of providing rapid responses to new threats. To provide transforming network policies into flow entries, packet data scan detection, including mandatory network policy enforcement and a range of security services, Zaalouk et al. [29] proposed a comprehensive SDN based security architecture. Flauzac et al. [30] presented an idea about how to build an SDN based network architecture with distributed controllers to ensure the security of the entire network to prevent attacks. Pisharody et al. [31] proposed a framework for distributed cloud environments based on SDN to monitor and maintain a conflict-free environment. There are only a limited number of solutions, and infrastructure-based SDNs are still being adopted and developed, while security issues are being explored and discovered.

Recently, to detect anomalies in traditional Intrusion Detection Systems (IDSs), machine learning techniques, like self-organizing maps, bayesian networks, fuzzy logic, and artificial neural networks principles and concepts, are widely used. Both in wireless and wired networks, these machine learning techniques have been considerably successful. Similarly, in the SDN platform, these techniques have been effectively applied in DDoS attack detection [8].

3. OpCloudSec design

3.1. Design overview

At present, virtualized infrastructures present a couple of challenges to traditional hardware-based networks. First, VMs need to move between the physical hypervisor and different data centers or physical locations. As VMs are relocated, traffic flows often change dynamically and virtual servers are added. Furthermore, the topologies change in real-time with them, and network configurations also need to be changed. Second, network management is detached from the supervision of virtual servers and sometimes is not even controlled by the same administrator. If the ultimate goal is to consolidate all cloud services under a centralized, single, easy-to-control then the two biggest risks are inadequate standards and vendor lock-in.

Although IoT is comprised of compiled multiple networks supporting applications with diverse requirements, such as reliable delivery with minimum delay, and heterogeneity of the device application is much harder to IoT.

Based on our analysis, we determined that we needed to integrate the defense against security attacks in the cloud by using SDN for IoT. We were able to improve server utilization, optimize bandwidth usage, closer integration with security, configuration management, storage, and much more by using SDN. To productively deal with these challenges in the new network surroundings, we had to achieve the objectives listed below.

- First, the system must be efficient, and the design should be able to protect services in both public and private clouds. It also needs to have the capacity to adjust to the system topology changes and pacify DDoS attacks effectively.
- Second, the approach should engage small overloads, and the computation and communication overhead should be limited to a small amount.
- Finally, it should reduce deployment costs. The solution should require modest deployment costs, as the existing protocol change for cloud providers and enterprise services and additional hardware.

To meet these challenges, our idea was to match the network traffic of the enterprise with the primary network by the slice. At that point we then let the cloud provider send the slice to its possessor. Like stage virtualization or equipment, a slice includes the system streams identified with the undertaking and is darker than other slices. The organizations shutting off between slices guarantee that a slice is obvious to

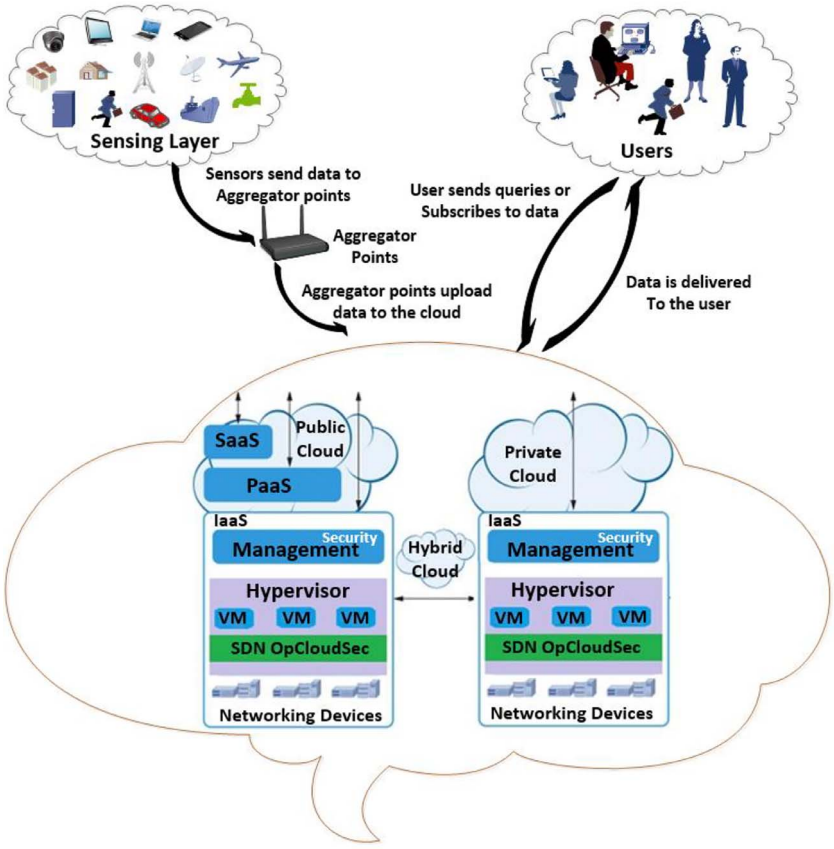

Fig. 2. OpCloudSec architecture.

its having a place party as it were. In this way, strategies performed on the slice are perfectly clear to other cloud clients.

To cut down on the overhead costs of communications, a competent attack detection and prevention approach that requires as meager data as potential must be selected. The existing detection of security attack algorithms could work if it is not dependent on specific hardware. It should also be a reference to the detection of the detection or the signature based on the anomaly detection based or a fused system may be used herein. To substantiate these claims, we are proposing a new architecture called OpCloudSec as shown in Fig. 2. The two most important core factors of OpCloudSec are its ability to connect to and manage in order to provide administrators and end users a single interface to operate a virtual infrastructure. By extending support to SDN controllers, OpCloudSec allows for networking with the same flexibility and vendor-independence currently offer virtual server environments. OpCloudSec enables service providers to administrate their network components and switches from the same interface that is utilized to manage Virtual Machines (VMs) and storage. This allows service providers to progress to the ultimate goal for their distributed virtual data centers, to have a single central interface for all physical and virtual components, and for those systems to can reconfigure themselves dynamically and not require significant administrator intervention.

3.2. Workflow of OpCloudSec

OpCloudSec is an attack detection and reaction system based on an anomaly. We contend that attack detection based on a signature could also work for an attack detection system, but it is not efficient. In fact, even in current work on DDoS security for today's Internet, the reaction choices are extremely basic and deficient in light of the fact that the post-taking care of method of reasoning requires switches working all things considered distributedly. It oversees and actualizes these capacities are blunder inclined and tedious. That is why in SDN, the responsibility for generating a transition package signing of a switch or a middlebox to a remote control program and we can apply this complicated logic, as isolate different types of packages to different places. Now, we assume that we have a detection and prevention of attack algorithms implemented as shown in Fig. 3.

In OpCloudSec, the cloud provider first finds out that the packet belongs to whenever a new packet arrives at the switch, and the cloud provider must inform the related network operating system about the slice. After receiving notification checks the operating system of the network owner of the slice in regards to if the packet belongs to flow or nonexistent. Otherwise, it builds a new flow record and asks for the detection pattern of the update or the new static flow. Otherwise, it updates flow statistics. If the query result shows an attack, OpCloudSec issues an alert and sends the alert and packet information for the level

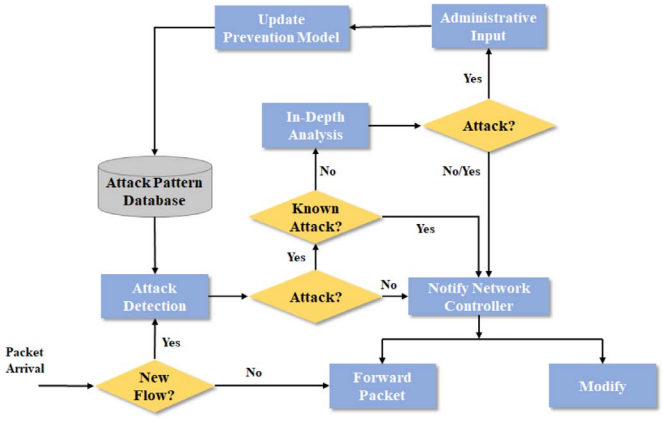

Fig. 3. OpCloudSec workflow.

of detail of analysis. If the query result is a pattern, the packet is forwarded to its designated destination. Sometimes, if the packet has a place with another kind of attack, then afterward the identification display cannot discover the sort of an attack packet. In this case, the request packets for analysis. When OpCloudSec receives an alert for an in-depth analysis, and deep packet reanalyze and try to check with the existing rules. If the rule corresponds to the packet before its intended destination, and if the rule does not match then it drops the packet and makes some contribution to the management when needed and customize their own defense systems and to update their own prevention model and store it in the basic attack pattern data to detect pattern similar attack.

3.3. Attack prevention modeling

For creating an attack prevention model, we used the Deep Belief Network (DBN) to analyze the newly captured network traffic. As compared to conventional machine learning, there are several hidden levels of functionality in deep learning. All the features are discovered and grouped automatically at different levels to produce outputs. Based on the features discovered in the previous level, each level represents abstract features. Due to heterogeneous patterns and highly unstructured data in multiple domains, we used a deep learning algorithm that outperforms other solutions in several domains. As we know that data in Cloud based IoT network are heterogeneous. Due to its ability to extrapolate new features from a limited set of training data, deep learning also has advantages over other forms of classical machine learning algorithms. In addition, for deployment in a cloud infrastructure, the layered and thin structure of sequential deep neural network models makes it the best fit and help to facilitate real-time anomaly detection.

As shown in Fig. 4(a), in the DBN model, the weights of all the hidden layers are denoted by W_i. The objective of our proposed model is to diminish the total cost. Each hidden layer is linked to the next hidden layer by using linear combinations of outputs, and feeds the filtered output generated by the activating function of the rectified linear unit to the next layer. The function of activating the rectified linear unit defines all the negative values in the matrix I to 0, and rests all the values held constant. Where I is a matrix from a convolved image. The activation function of the rectified linear unit defined as

$$f(I) = Max\ (0,\ I) \tag{1}$$

Each meta-feature set is signified as the feature vector f_m to represent the probability of meta-features generated from each data-packet. f_m is calculated as:

$$f_m(k) = i_m(k)\ XOR\ i_m(k-1) \tag{2}$$

where, i_m represents the data-vector.

We set the training data set as: $S = (f_m^1, l^1),\ (f_m^2, l^2),\ \ldots,\ (f_m^S, l^S)$, where l^j is the tag information for each network transaction data packet. We assigned a cost function at each layer to diminish the total cost. The cost function can be defined as:

$$Cost(W, f_m,\ l) = \frac{1}{2}\ [g_w(f_m) - l]^2 \tag{3}$$

Where, for each meta-feature f_m, $g_w(f_m)$ is the hypothesis function. Fig. 4(b) shows the modeling work flow for attack prevention.

3.4. End-to-end OpCloudSec solutions

The IoT is a case of occurrences in which people or objects are given unique identifiers and the potential to transfer data over a network without necessitating human-to-computer or human-to-human interaction. The IoT has developed from the convergence of micro-electro-mechanical systems, wireless technologies, and the Internet.

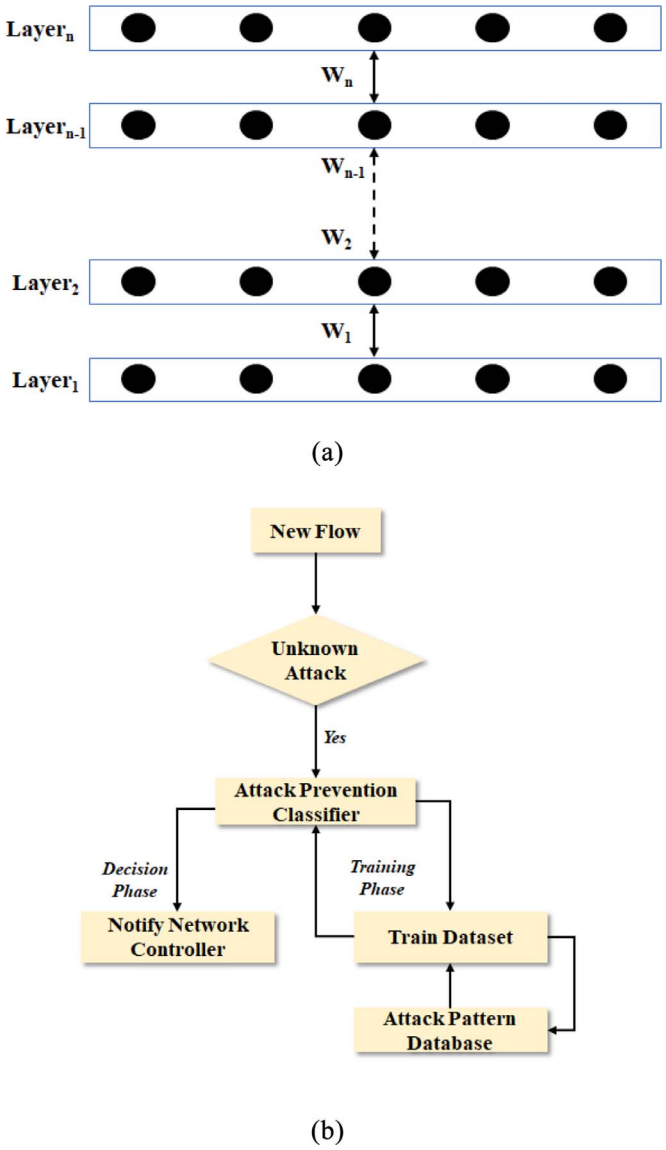

Fig. 4. Attack prevention modeling: (a) deep belief network model; (b) attack prevention modeling workflow.

3.4.1. User layer

The IoT User (people/system) is an individual or an automated system that makes use of one or more end user applications to achieve some goal. The IoT User is one of the primary beneficiaries of IoT solutions.

End User Application is a domain-specific or device-specific application. The IoT user may use end user applications that run on smartphones, tablets, PCs, or specialized IoT devices, such as control panels.

3.4.2. Sensing layer

Thw Sensor/Actuator is a component that senses or measures certain characteristics of the real world and converts them into digital representations. An actuator is a constituent that accepts a digital command to act on a physical entity in some way. In our solutions, the sensors received data from the real world and sent it to the aggregator points.

3.4.3. Aggregator point

This provides the services needed to allow data to flow safely from the internet into the cloud provider and the enterprise. In our proposed method, this component must be able to handle and transform high

volumes of messages and quickly upload data to the cloud service provider.

3.4.3. Flow for the general insurance scenario for the IoT

Sensors and actuators are positioned in the home or organization and are attached to the aggregator point to upload data to the cloud service. For example, the possessor logs into the insurance mobile app and authorizes the insurance service to access his or her cloud service and device data. The mobile app sends the approval token and the insurance company identifier to the cloud service. This info is used to map the devices, user, and insurance policy within the cloud service. The insurance service obtains the insurance id/authorization/device details from the insurance mobile app and processes this in various nodes (application logic, device registry, and data store). The devices are recorded with the device registry, and data mapping is created in the application logic module. The insurance service app connects to the device maker (peer) cloud using the approval token and data requests. To pull data at configured intervals, the application is setup. For use in analysis, the application can be configured to access other data sources apart from device data, such as weather data. Data from devices and other sources are frequently updated and sent for analysis to find out if a potential risk threshold has been exceeded. To determine if there is potential damage to the home or organization, the data is analyzed. If there is a problem, then once it is figured out, using the analysis from notifications is sent to the Possessor and the insurance company. The Possessor can then take an activity to respond to the notification and find out if damage has occurred and the insurance company can initiate a claim process. If damage has occurred, then the insurance diligence litigate of claims management is started out. The insurance company's business processes can be carried out in the cloud service, their enterprise app, or their mobile app. This is dependent on how and where the insurance company decides to perform the business logic. OpCloudSec makes this type of solution easier to implement and maintain and provide more security. As demand increases, more resources must be acquired.

4. OpCloudSec assessment

In this section, we evaluate the proposed OpCloudSec architecture in different scenarios. We discuss the experiments conducted on it and the results obtained from our experiments using real-world network traffic. We start with describing how the model is implemented by expanding an existing SDN simulator.

4.1. Experimental setup

To study the behavior and evaluate the performance of the SDN framework, several SDN simulators can be used. In our simulation, we used the Mininet simulator because of its simplicity and utility. To build a private cloud, we used three Linux servers, which were run on an Ubuntu 16.04 64-bit operating system. We also used Floodlight to construct the system controller. To quantify the cost of communication and the efficiency of OpCloudSec in the public cloud, we used the AWS EC2. Here, we considered the AWS EC2 public cloud as a core model.

We used the Python language to implement and test the proposed intrusion detection system. We used the Keras library to create a deep-learning model because of its ability to quickly and easily process large volumes of data.

4.2. Assessment dataset

We used the dataset UNB ISCX [32] to assess the attack discovery execution of our recognition module. The set of labeled data network traffic DDoS attacks, which implies that we have the ground truth traffic. The entire information set was then partitioned into equivalent shares. The first and last segments were utilized as test information and preparation information, and the rest of the parts were utilized to redesign the model overhaul handle.

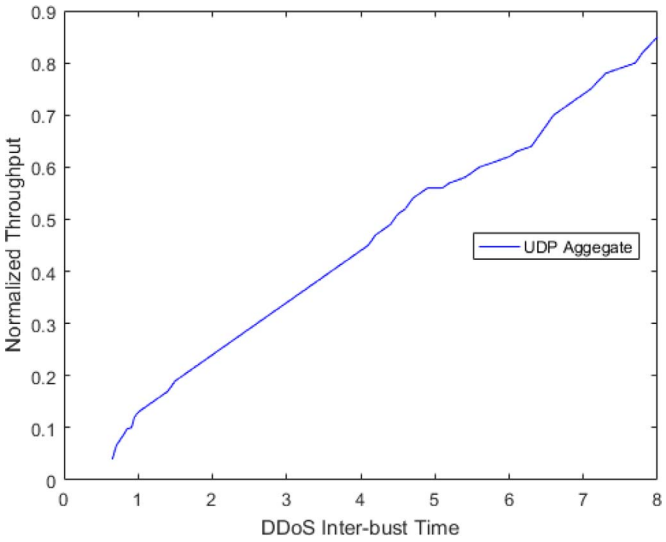

Fig. 5. Throughput vs DDoS Inter-bust time.

4.3. Defense effects

To evaluate the defense effect of our proposed model, we performed throughput, bandwidth, and used space evaluation during saturation attacks. We installed some clients to dispatch a UDP flooding attack to the SDN switches. We built a simple test that generated a DDoS attack and showed its effect on the throughput, bandwidth, and the storage host's node.

Fig. 5 shows the normalized throughput of our proposed model with respect to various DDoS inter-bust times. The results showed that the throughput of the proposed model increase as the inter-bust time is increase. To evaluate the effect of bandwidth during the saturation attack, we started dispatching flooding attacks through setup clients. Fig. 6 shows the effect in bandwidth of our proposed model during saturation attacks. We noticed that the bandwidth starts at 1.5 Gbps at attack rate 0 Packets per Sec (PPS), and that with increase of attack rates, the bandwidth decreased rapidly without our proposed model. In the case of a system that had our proposed model implemented in it, the bandwidth of remains was almost unchanged during saturation attacks.

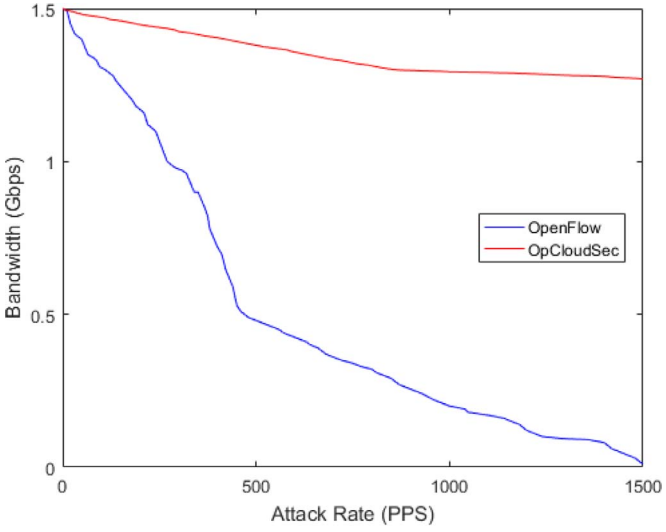

Fig. 6. Effect in bandwidth of during saturation attack.

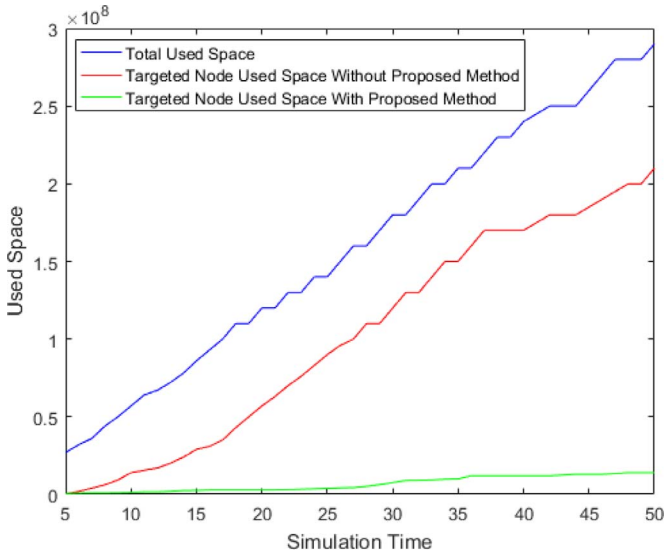

Fig. 7. Consumption of storage during saturation attack.

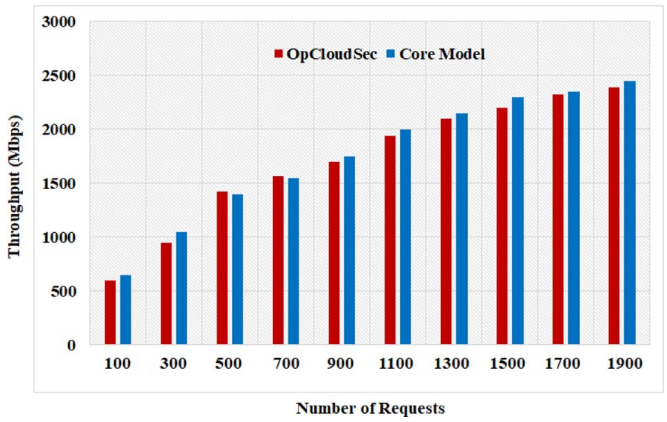

Fig. 8. Throughput vs number of requests.

We also measured the consumption of storage during a saturation attack, as shown in Fig. 7. As shown in Fig. 7, there is a slightly variation in the storage consumed in the case of a target node without a proposed model, which clearly indicates that the majority of requests are sent to the targeted node. Whereas, in the case of using a targeted node with our proposed model, the system worked perfectly and the storage consumed was reasonable in comparison to the total amount of storage used.

To assess the performance overhead, we also evaluated the variation in throughput using core cloud infrastructure and our proposed model with respect to the number of requests. Fig. 8 presents the throughput variation with respect to the number of requests. The results in Fig. 8 show that the proposed model provided higher throughput as compared to the core model.

4.4. Accuracy

4.4.1. Metrics

We evaluated the performance of our proposed Attack Prevention Model (APM) in terms of precision, recall, and F-measure parameters. A confusion matrix was used to compute the True Positive (TP), False Positive (FP), True Negative (TN), and False Negative (FN) parameters. The evaluation metrics are defined as follows:

Precision signifies that the number of predicted intrusions are real intrusions. It is defined as:

Fig. 9. Comparison of results of attack prevention model: (a) Precision; (b) Recall; (c) F-measure.

$$Precision = \frac{TP}{TP + FP} \tag{4}$$

Recall signifies the percentage of predicted intrusions upon all intrusions presented. It is defined as:

$$Recall = \frac{TP}{TP + FN} \tag{5}$$

F-measure gives the measure of the accuracy considering both the precision and recall. It is defined as:

$$F - measure = \frac{2 \times Precision \times Recall}{Precision + Recall} \tag{6}$$

Table 1
Summary of the results of our proposed method compared to other methods of measuring performance parameters.

Method	Learning rate	Evaluation metrics		
		Precision	Recall	F-measure
SOM	0.5	73.1	60.2	66.02
NB		70.2	59.1	64.17
SVM		70	58.6	63.79
Our APM		**79.1**	**72.3**	**75.54**
SOM	0.1	76.2	65.6	70.50
NB		77.2	67.1	71.79
SVM		75.1	61.2	67.44
Our APM		**83.2**	**79.3**	**81.20**
SOM	0.05	83.5	73.2	78.01
NB		84	72.1	77.59
SVM		83.9	71	76.91
Our APM		**89.4**	**81.5**	**85.26**
SOM	0.01	88.1	78.6	83.07
NB		86.5	77	81.47
SVM		87.1	77.3	81.90
Our APM		**92.1**	**85.1**	**88.46**
SOM	0.005	91.9	83.2	87.33
NB		90.5	82.5	86.31
SVM		89.5	81.2	85.14
Our APM		**94.6**	**91.3**	**92.92**
SOM	0.001	90.7	81.2	85.68
NB		89.1	80.5	84.58
SVM		88.1	79.8	83.74
Our APM		**93.1**	**89.8**	**91.42**

4.4.2. Performance

We trained our proposed attack prevention model (APM) and repeated the experiments 10 times to reduce the uncertainty of the datasets. We also trained our dataset using a Support Vector Machine (SVM), Self-organizing Map (SOM), and Naïve Bayes (NB) classifiers [8,33] to compare our proposed method with other state-of-the-art methods. We tried to optimize the model by varying the value of the learning rate, which leads to optimal classification results. As shown in Fig. 9(a)–(c), the proposed model demonstrates significantly higher precision, recall, and F-measure rate with respect to varying learning rates than other methods. We used the learning rate in the range of (0.5, 0.1, 0.05, 0.01, 0.005, 0.001). With a decreased rate in learning, the accuracy rate increased. As shown in Fig. 9, we obtained better accuracy rate at the learning rate of 0.005. Table 1 summarizes the results of our proposed method compared to other methods for measuring performance parameters. The end results of our evaluations empirically demonstrate that our method has a better accuracy rate.

5. Conclusion

With the advances in cloud and networking technologies, the increased reliance on cybernetic physical systems has highlighted the need to protect the cloud infrastructure and network from DoS attacks. However, the detection and mitigation of DDoS attacks remains an unfinished task.

In this paper, we proposed the architecture for OpCloudSec, which supports our finding and provides a solid basis for understanding the IoT in cloud systems. It also meets the many challenges that exist in a systematic and logical manner. Network control based on OpCloudSec and monitoring mechanisms enable organizations to control and reconfigure their defense mechanisms effectively in the cloud without influencing other cloud clients. To assess the performance of our proposed model, we performed a simulation study using real network traces. The outcomes showed that when it comes to dealing with new challenges, our OpCloudSec proposal was successful. The detection algorithm is quick enough for performing deduction line packages and accomplishes a high identification rate.

Acknowledgments

This work was supported by the National Research Foundation of Korea (NRF) grant funded by the Korea government (MSIP) (No. 2016R1A2B4011069).

References

[1] G. Somani, et al., DDoS attacks in cloud computing: issues, taxonomy, and future directions, Comput. Commun. (2017).
[2] S. Singh, Y.S. Jeong, J.H. Park, A survey on cloud computing security: Issues, threats, and solutions, J. Netw. Comput. Appl. 75 (2016) 200–222.
[3] E. Cavalcante, et al., On the interplay of internet of things and cloud computing: a systematic mapping study, Comput. Commun. 89 (2016) 17–33.
[4] W. Xia, et al., A survey on software-defined networking, IEEE Commun. Surv. Tutorials 17 (1) (2015) 27–51.
[5] S. Sezer, Are we ready for SDN? Implementation challenges for software-defined networks, IEEE Commun. Mag. 51 (7) (2013) 36–43.
[6] S. Azodolmolky, P. Wieder, R. Yahyapour, SDN-based cloud computing networking, Proc. IEEE ICTON, 2013, pp. 1–4.
[7] Z. Xiao, Y. Xiao, Security and privacy in cloud computing, IEEE Commun. Surv. Tutorials. 5 (2) (2013) 843–859.
[8] N.Z. Bawany, J.A. Shamsi, K. Salah, DDoS attack detection and mitigation using sdn: methods, practices, and solutions, Arab. J. Sci. Eng. 42 (2) (2017) 425–441.
[9] Arbor Networks Inc., accessed online 04 Sep. http://www.arbornetworks.com, (2017) accessed online 04 Sep..
[10] R. Roman, J. Zhou, J. Lopez, On the features and challenges of security and privacy in distributed internet of things, Comput. Netw. 57 (10) (2013) 2266–2279.
[11] J. Kar, M.R. Mishra, Mitigating threats and security metrics in cloud computing, J. Inf. Process. Syst. 12 (2) (2016) 1–8.
[12] D. Chen, H. Zhao, Data security and privacy protection issues in cloud computing, Computer Science and Electronics Engineering (ICCSEE), 2012 International Conference on IEEE, 1 2012, pp. 647–651.
[13] E.J. Lee, C.H. Kim, I.Y. Jung, An intelligent green service in internet of things, JoC 5 (3) (2014) 4–8.
[14] C. Kang, F. Abbas, H. Oh, Protection scheme for IoT devices using introspection, Network of the Future (NOF), 2015 6th International Conference on the IEEE, 2015, pp. 1–5.
[15] H. Kim, N. Feamster, Improving network management with software defined networking, IEEE Commun. Mag. 51 (2) (2013) 114–119.
[16] J. Liu, et al., Device-to-device communication for mobile multimedia in emerging 5G networks, ACM Trans. Multimedia Comput. Commun. Appl. (TOMM) 12 (2016) 1–20.
[17] V. Vidhya, A review of DOS attacks in cloud computing, IOSR J. Comput. Eng. (IOSRJCE) 16 (5) (2014) 32–35.
[18] Q1 2015 DDoS Attacks Spike, Targeting Cloud, Tara Seals, http://www.infosecurity-magazine.com/news/q1-2015-ddos-attacks-spike/, (2015) accessed online 04 Sep. 2017.
[19] F. Motavaselalhagh, et al., Knowledge-based adaptable scheduler for SaaS providers in cloud computing, Hum. Centric Comput. Inf. Sci. 5 (1) (2015) 16–34.
[20] accessed online 04 Sep. http://wikisites.cityu.edu.hk/sites/netcomp/articles/Pages/HardeningStepstoSecureCloudComputingEnvironment.aspx, (2017) accessed online 04 Sep..
[21] R. Miao, et al., The dark menace: characterizing network-based attacks in the cloud, Proceedings of the 2015 ACM Conference on Internet Measurement Conference, ACM, 2015, pp. 169–182.
[22] Y. Sung, et al., FS-OpenSecurity: a taxonomic modeling of security threats in SDN for future sustainable computing, Sustainability 8 (9) (2016) 919–945.
[23] B. Wang, DDoS attack protection in the era of cloud computing and software-defined networking, Comput. Netw. 81 (2015) 308–319.
[24] J. Gubbi, et al., Internet of Things (IoT): a vision, architectural elements, and future directions, Future Gener. Comput. Syst. 29 (7) (2013) 1645–1660.
[25] S. Shin, Avant-guard: scalable and vigilant switch flow management in software-defined networks, Proceedings of the 2013 ACM SIGSAC Conference on Computer & Communications Security, ACM, 2013, pp. 413–424.
[26] B. Lantz, B. Heller, N. McKeown, A network in a laptop: rapid prototyping for software-defined networks, Proceedings of the 9th ACM SIGCOMM Workshop on Hot Topics in Networks, ACM, 2010, pp. 1–6.
[27] L. Yanbing, et al., SDSA: a framework of a software-defined security architecture, China Commun. 13 (2016) 178–188.
[28] J. Kim, et al., SDN-based security services using interface to network security functions, Information and Communication Technology Convergence (ICTC), 2015 International Conference on, 2015, pp. 526–529.
[29] Z. Hu, et al., A comprehensive security architecture for SDN, Intelligence in Next Generation Networks (ICIN), 2015 18th International Conference on, 2015, pp. 30–37.
[30] O. Flauzac, et al., SDN based architecture for IoT and improvement of the security, Advanced Information Networking and Applications Workshops (WAINA), 2015 IEEE 29th International Conference on, IEEE, 2015, pp. 688–693.
[31] S. Pisharody, et al., Brew: a security policy analysis framework for distributed SDN-based cloud environments, IEEE Trans. Dependable Secure Comput. (2017).
[32] UNB ISCX Datasets, accessed online 04 Sep. http://www.unb.ca/cic/research/datasets/index.html, (2017) accessed online 04 Sep..
[33] R. Braga, E. Mota, A. Passito, Lightweight DDoS flooding attack detection using NOX/OpenFlow, LCN '10 Proceedings of the 2010 IEEE 35th Conference on Local Computer Networks, IEEE, 2010, pp. 408–415.