SDN Framework for Securing IoT Networks

Prabhakar Krishnan(✉), Jisha S. Najeem, and Krishnashree Achuthan

Center for Cybersecurity Systems and Networks, Amrita University, Amritapuri, India
kprabhakar@am.amrita.edu

Abstract. Internet of Things (IoT) paradigm is the interconnection of machines, intelligent devices and location aware analytics platforms that collectively enable us to have smart world around us. As the billions of already connected devices and newly added devices grow this network, IoT pose the most complex operational and information technology challenges to the way networks are designed and operated. With the emerging technologies like SDN, SD-WAN, NFV, IXP evolving into standards, researchers are proposing new communication platforms to deliver secure and scalable networks for Internet of Things (IoT). In this paper, we discuss major security challenges in IoT networks and present the notion of security architecture for IoT based on programmable and virtualization technologies SDN/NFV, explain the architectural choices and its applications for IoT. We review prior works in this area and discuss our future work to solve security and privacy challenges of heterogeneous systems and networks in IoT.

Keywords: Internet-of-Things (IoT) · Software-Defined-Networking · SDN
Network security · Network-Function-Virtualization

1 Introduction

The proliferation of IoT based smart devices, estimated to become a 20 billion interconnected network by 2020, and brings with it several challenges and hard problems with regard to security and privacy of devices, users and the data consumed by applications. With the kind of ubiquity within society predicted, it will create the need for flexible sophisticated methods of integrating these large farms of embedded devices within overall network architectures. This integration will potentially be dynamic, as users and devices roam in and out of the wireless networks, within their context or zones (e.g. fleet with vehicular network sensors that access context aware applications and services from local networks).

IoT architecture can be visualized in 3 tiers; on the top tier are usually well-protected devices, secured servers, personal computers, laptops and smart phones with sophisticated firewall software hardware. The middle tier typically consists of devices of less complex smart appliances and devices such as refrigerators, lights, cameras, televisions, digital screens and luxury HiFi devices. The bottom tier comprises of devices from consumer electronics, mechatronics and lifestyle gadgets such as smart locks, digital doors, perimeter safety and surveillance machines, air-conditioners and wearable, medical implants, geo-sensory equipment, vehicular network devices and so on.

N. Kumar and A. Thakre (Eds.): UBICNET 2017, LNICST 218, pp. 116–129, 2018.
https://doi.org/10.1007/978-3-319-73423-1_11

None of these three tiers of devices may pose a threat independently restricted to that autonomous homogeneous network. But when we interconnect the devices from across the tiers, the resulting architecture will consist of heterogeneous devices, integrating disparate technologies and communication protocols, Application Program Interfaces (APIs) etc. And this heterogeneous interconnected IoT architecture may pose serious risks and challenges for Quality of Service (QoS), security and privacy. So far we don't have a single one-size solution to address all these challenges.

Thus, ensuring the trust and security of the configuration, topology and integration of all heterogeneous devices into large networks are some major operational challenges. Experimental exploitation of current generation of smart devices or things have demonstrated that breaching and tampering is possible and also established the need for handling IoT devices network security (Fig. 1).

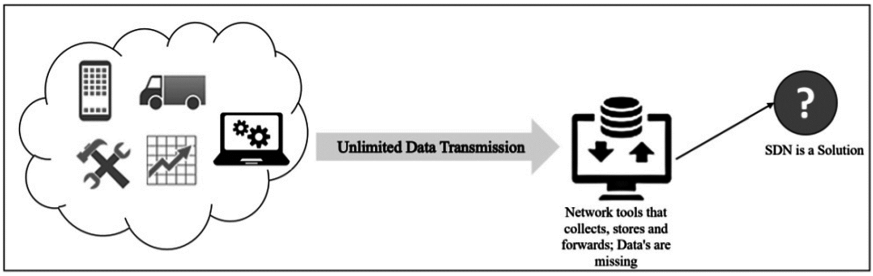

Fig. 1. SDN being a solution

The modern SDN paradigm has initiated a fundamental redesign of how network traffic management, routing control logic, forwarding and network orchestration are architected. The design should also provision flexible, agile device management policies. This design philosophy is implemented through the separation of control logic or brain from the packet forwarding functions. In other words, SDN consists of one centralized control plane that is connected over a standard communication channel to distributed physical data or switching plane.

Some key criteria for evaluating the SDN in IoT network include:

- Ability to securely connect and manage hundreds or even thousands of heterogeneous IoT devices.
- Low latency security monitoring overhead to deliver real-time awareness and operations.
- Scale-out elastic architecture to scale and dynamically load balance/shift workloads, and
- Programmability for enforcing custom policies and applications.

In this paper, we discuss the effectiveness of approaches to design new secured network architecture based on SDN, advanced network virtualization functionalities and clusters.

This paper is organized as follows: Sect. 1 introduces the emerging technologies for the interconnected IoT networks, applications and sets the context for incorporating SDN in IoT architecture, Sect. 2 provides an overview of the threat landscape in IoT and current approaches to IoT Security Sect. 3 gives an overview of the security threats and risks to IoT network, Sect. 4 explains the feasibility and efficacy of SDN architecture in the context of IoT networks and discusses related works in SDN IoT integration. Section 5 articulates some key challenges for this SDN/IoT domain. Section 6 proposes our SDN framework for securing the IoT networks, architecture, design choices and case studies. Section 7 presents our experience from initial experimentation and evaluation, Sect. 8 provides a general outlook of our future work and concludes the paper.

2 Approaches to IoT Security

To implement dependable security architecture for an IoT network, both system characteristics and data centric parameters must be considered. The security framework should combine them to achieve the desired level of privacy, security, risk level, interoperability, recoverability and trustworthiness. Vendor com- munities, business applications, government norms and regulations may drive these factors. Security in IoT network must be implemented at various levels: the manufacturing vendors and supply chain, hardware ASIC or SoC, Operating systems, systems software and application software, middle-box appliances, networking hubs, routers, and switches.

The target IoT environment may have several constraints, For example:

- Real-time infrastructures cannot be brought down for security updates and patching.
- Low-latency, proprietary protocols limit the ability to deploy antivirus and anti-malware software.
- Embedded processors have limited processing power and memory to execute security software.
- IoT devices have a small form factor, limited connectivity and are designed for very low power consumption.
- Attacks toward wireless network infrastructure can cause the unavailabity of network component and data loss.
- Many IoT devices are physically accessible to the attacker.

Despite all these threats, two key areas of IoT security that have not received much attention are:

1. Software integrity: Ensuring the authenticity and integrity of the software on the device. By allowing software that digitally signed, whitelisted and certified by trusted entity, to run and access data.
2. Device authentication: Authentication of the end devices before they can transmit or receive information. Authentication of devices and data is a key success factor for the Internet of Things. A single compromised node can be turned into a malicious one that brings down whole systems or causes disasters with cars, planes, drones, the grid etc.

The known shortcomings of knowledge-based authentication approaches like passwords and PINs must be augmented with standard solutions like Public Key Infrastructure (PKI) in conjunction with new technologies like Physical Unclonable Functions (PUFs). These provide measures to strengthen IoT security from a self-enforced identity perspective. Using a block chain to store data that has been secured with PUF derived keys and attributes provides an immutable assurance that data has not been tampered with, in addition to providing traceability and transparent auditing capabilities.

The majority of proposed security solutions use cryptographic algorithms that normally require high amount of resources. Considering that most IoT devices are associated with low energy and computing resources capabilities, such solutions cannot be implemented to IoT devices with an application of traditional cryptographic mechanisms.

3 Integrating SDN into IoT Networks

This section briefly introduces the area of software defined networking (SDN) and discusses its applicability to both acting as a gateway for IoT devices and as a security controller mechanism.

3.1 Background About SDN

SDN is an open network architecture proposed in recent years to address some of the key shortcomings of traditional networks. The proponents of SDN argued that the control logic of the network and network functions are two separate concepts, and should therefore be separated in different layers. To this end, SDN hence introduced the concepts of control plane and data plane: The centralized control plane (controller) manages the network logic, control traffic engineering functions from the data plane (switches) that just take care of forwarding the packets between the networks. So, the SDN can be considered as a physically distributed switching framework with a logically

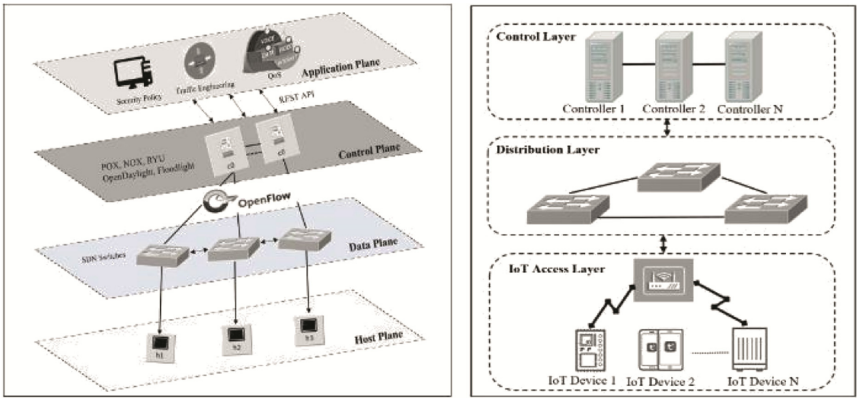

Fig. 2. (a) SDN architecture (b) SDN - IoT integrated architecture

centralized control. SDN is designed for provisioning highly dynamic orchestration and quality of service/security policies (Fig. 2).

3.2 Significant Benefits of SDN-IoT

IoT network environment is a large interconnection of multiple smaller local or adhoc networks or wireless or industrial control networks. The orchestration and visibility of end-to-end traffic and devices is essential for establishing QoS and Security policies. To accurately visualize the operating environment, automated programmed mechanisms are needed and that be provided by SDN. It can validate the addition or deletion or modification of devices/configurations in the monitored network and it can program the policies at various points of the network at run time to react differently depending upon the behavioural characteristics of the devices.

The key feature of the SDN is the dynamic provisioning at run time. The capability can be extended for security monitoring of the network. The SDN applications and elements can be programmed for anomaly detection and diversion of suspicious attack traffic to sandbox or honeypot deception framework for further analysis.

For modern IoT applications which encompass multiple interconnected networks or micro-networks in the Cloud, we can incorporate SDN elements to create a suite of semantic monitoring, fine-grained security analytics, defense mechanisms, software defined perimeter, firewalls at different vantage points or locality or layer boundaries of network.

4 Related Works in SDN-IoT

In this section, we present an overview of the related works that have been proposed in the context of SDN-IoT. Flow based security monitoring mechanism [1] has explained the numerous attacks and mitigation approach. Their infrastructure consists of statistics manager that collects data in real-time from log cache and analyses the flows and mitigation actions such as blacklisting are taken based on the various characteristics of flow.

Fog Computing [2] sets another security feature for IoT devices using SDN. The architecture comprises of gateway edge nodes and servers. Edge gateways and servers are connected via high-speed interfaces that can be either wired or wireless such as 3G, LTE etc. Gateways have their own unique role for master mode that controls the virtual path of gateway function located in slave nodes. Using ClickOS, a virtualized software middle box can concurrently run on a commodity server.

One of the most common and significant security threats deeply researched is that of distributed denial of service (DDoS) attacks. Numerous projects are currently seeking to use SDN based security systems as means of mitigating such attacks. Choi [3] suggested a new framework that discovers generation of new traffic thereby performing DDoS mitigation by limiting the amount of traffic generated for each application. Another technique was to identify malicious flows by developing an anomaly detection technique [4] based on the history of the networks stored in the log cache and then comparing with the real-time traffic generated. Significant effort is also made in wireless

network security enhancement by applying SDN in wireless/adhoc networks. An SDN based enterprise solution Odin [5] has built a virtualized multi-layered network architecture that uses abstraction of access points. Another open source project OpenRoads [6], decouples the data plane layer and the network layer providing dynamic control over the network management. In addition to the ongoing SDN-based security research projects, there are a small number of commercially developed security applications that are designed to integrate with SDN controllers in IoT networks.

5 Challenges in SDN and IoT

SDN and IoT integration provides a convincing approach to simplify network management and security control, but SDN has inherent design vulnerabilities that pose serious threats to the integrated IoT network and applications [7]. In the SDN architecture, (a) the switches that maintain the flow tables and its capacity (b) communication channel speed, reliability and bandwidth are the critical points for SDN operation. The following issues could lead to critical point failures:

1. The SDN switches/data-plane evaluates incoming packets, matches with Rule table or Flow tables, which are stored in switch fabric (TCAM) memory, having nite capacity, can be attacked.
2. SDN switches out there in the open, may be compromised and recruited in to the botnets, leading to massive DDoS attack campaign.
3. Control-Data Plane link is vulnerable and if saturated, it may lead to total network breakdown. Network level new flow attacks, DoS attacks such as TCP-SYN/DNS/ICMP Amplification and flooding are common in recent times. Hence the placement of controller and protection of communication channel between controller and switches are critical aspects for security availability of the SDN-based IoT applications.

Though the notion of SDN in the context of IoT applications, is still at an early stage, research is gaining momentum to secure the SDN stack, tackling all the above mentioned critical points of attack and IoT communities are investigating the hybrid SDN/IoT architecture for design choices and implementation trade-offs.

6 SDN Based IoT Network Architecture: Our Proposal

In this section, we discuss our SDN-IoT integration architecture, with two design choices, varying in terms of implementation and modifications to the standard SDN or IoT components. We walk through the building blocks of the architecture following both design choices. The core functions of security analytics, access control, policy decisions and enforcement are implemented in the SDN layer and the data from IoT layer is selectively forwarded through this core framework. By acting on contextual information exposed by the IoT applications sensory network, the gap between the IoT and IT networks are filled by the SDN (Fig. 3).

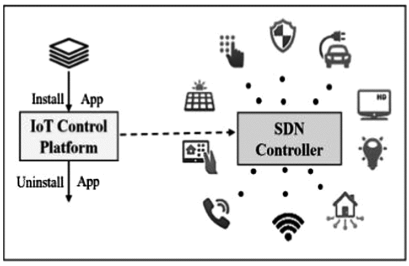

Fig. 3. SDN - IoT control architecture

A systematic design approach to monitor large-scale IoT networks with the SDN gateway and controller, allows for a holistic view of the network and removes the need for additional dedicated hardware. The two design choices for SDN and IoT integration are:

1. Loosely coupled Integration: A flexible flow based monitoring and security mechanisms implemented at SDN control plane as applications. The IoT layer has no modifications and a new layer for SDN is added at the Edge.
2. Tightly coupled Integration: Hybrid Gateway Switch (IoT gateway and SDN Switching) and a security controller, implemented as a sandwich layer between Edge and IoT network. This requires modifications to both SDN IoT layer.

6.1 Design Choice 1 - Loosely Coupled Integration

This design is implemented as a defensive mechanism, attaching the SDN stack to the IoT layer (Gateway) at the Edge security processing. In this framework, we will have SDN applications that monitor the flows and configuration, generate blacklists and whitelists, in the IoT network and analyse packet streams for spatial, temporal and volumetric correlations in their behaviours, protocol violations, and attack signatures (Fig. 4).

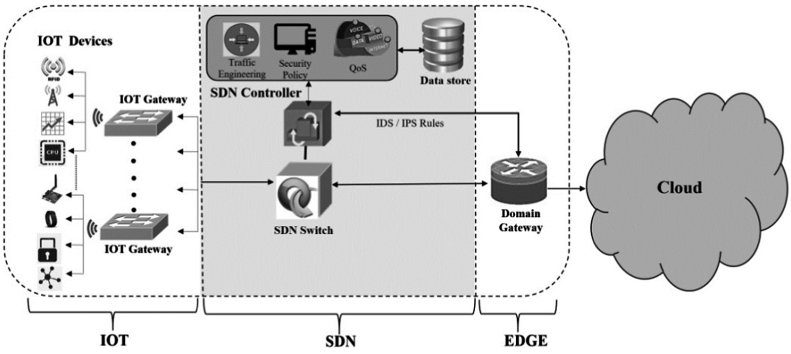

Fig. 4. Loosely coupled SDN-IoT integration

Security threats can be from External or Internal network. The Outside attackers or botnets can target the IoT Gateway or sensor devices or Services, Vulnerable apps installed in the devices in the internal network e.g. Home WiFi-Router/Mode, Webcams. The common indicators of such attacks are: (a) Login access or scanning traffic from the public network, to key IoT gateway or sensor devices, (b) The malicious usage of the IoT devices/apps is beyond their declared functionality.

6.2 Design Choice 2 - Tightly Coupled Integration

This architecture is based on an extending the SDN stack to interface with the IoT stack, specifically the IoT gateway functionalities and protocols. This architecture is built by inheriting the major modules of the SDN Openflow switch stack and by adding new functionalities in the packet processing workflow. This approach is similar to a middle-box device running a modified network operating system that combines IoT gateway and SDN data plane functionalities. This device has extensible architecture and

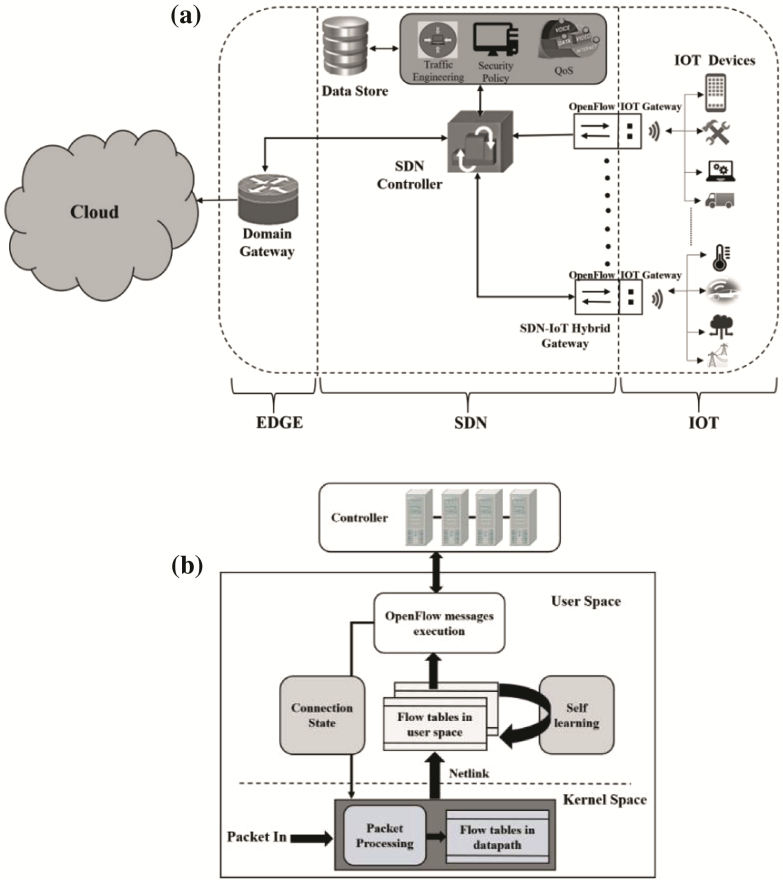

Fig. 5. (a) Tightly coupled SDN-IoT integration, (b) flow analysis in SDN-IoT hybrid device

dynamically loadable modules to support several protocols defined for IoT networks OpenFlow message processing is in charge of receiving and forwarding SDN OF protocol messages in the kernel directly.

In the Fig. 5b, SDN-IoT Hybrid device, Connection-state module performs the connection state tracking, synchronization. Self-Learning module performs flow table analysis and detect anomalies in the network traffic and track down the end points. Initialized with predetermined signatures and correlation rules for well-known attacks, this device has a learning module that learns the features/attributes set, this improving accuracy and granularity of detection. It also supports custom applications and associated libraries for IoT security and monitoring.

6.3 Global Cloud Command and Control

This conceptual global management network encompasses multiple local domains and a central command and control systems are hosted in the cloud. The domain level controllers (i.e. Fog) are interconnected with secure communication (SSL/TLS) channels. It runs a suite of business specific applications to manage enforce end-to-end security policies, traffic QoS, and big data analytics for the IoT network. In a federated architecture, a domain gateway controller negotiates with the global/other domain controller to determine for further processing (Fig. 6).

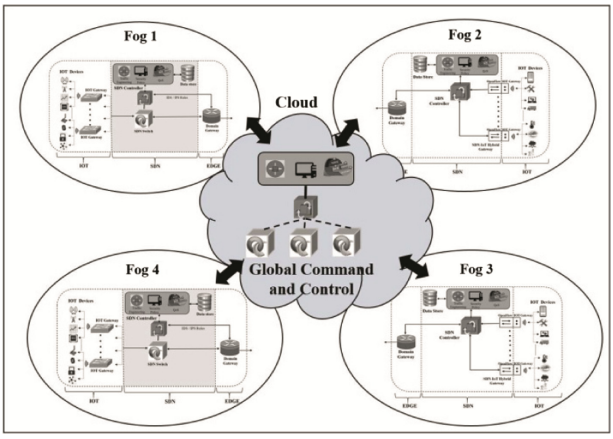

Fig. 6. SDN integrated IoT application

7 Initial Experimentation and Evaluation

In order to design the SDN based detection and mitigation more practical and dependable, we have to face the following challenges:

1. Traditional monitoring mechanisms based on IP entropy and TCP protocol proportion, Blacklisting, signatures are not effective with sophisticated arbitrary packet injection in network and botnets attack flows performing like a normal burst of traffic.
2. The cost of monitoring should be minimal and limited by the link bandwidth, speed and the real time requirements of the applications in target network.
3. The attack detection process should be followed up by mitigation strategies. Once the attacks are detected, it should be mitigated quickly by generating alerts, notifications and defensive rules communicated to SDN controller so that the actionable Rules are installed into the switching plane.

Hence to address these challenges, we have defined some key evaluation criteria in our SDN security controller, especially dealing with DoS attacks: 1. Packet handling rate/response to new connections or new flows and 2. Packet matching efficiency 3. monitoring cost for the new-flow attack. Our implementation strategy included fine-grained monitoring and defense mechanism that has lesser overhead in-terms of: new added modules code foot-print, instruction size in fast path for benign/normal traffic, memory usage for meta-data, extended flow-tables for dynamic security analytics, control protocol overhead and other costs. It can differentiate the DoS flow attack from the normal flow burst which ensures the minimal delay for normal packets to flow through our SDN framework and at the same time the attack/suspicious malware packets are detected at high accuracy and diverted to the self-learning anomaly detection module.

7.1 Experimental Network Topology

We have established a reconfigurable testbed to implement our design choices.

The IoT end-to-end architecture as depicted in this figure is divided in two parts (Fig. 7):

1. *Internal Network*
   - Edge domain gateway, a Linux firewall appliance running SNORT/IDS
   - SDN stack: modified RYU Controller and security/attack detection applications, modified vSwitch (OVS) switching software.
   - IoT stack: gateway running ContikiOS, supporting about 6 network protocols both Wi RF and modified middleware protocol stack, SDN Open flow enabled.
   - IoT Sensor network: 4–6 physical sensors/motes, 2 workstations running a virtual simulation of IoT sensors, running all WiFi/RF protocols
   - IoT Attack: This is a software simulator tool that generates attack traffic, fuzzing protocols and jamming
   - General Internal attacks: We use a set of machines that runs the widely used exploit kits and attack tools.
2. *External Network*
   - We setup legitimate hosts, and users and applications using transport protocols (TCP/IP, MQTT) to gain access into our test IoT network.
   - Attacking hosts users, botnet applications, who gain access through covert channels in TCP/IP, generate DoS attacks and targeted attack to test IoT network to infiltrate malware or steal data.

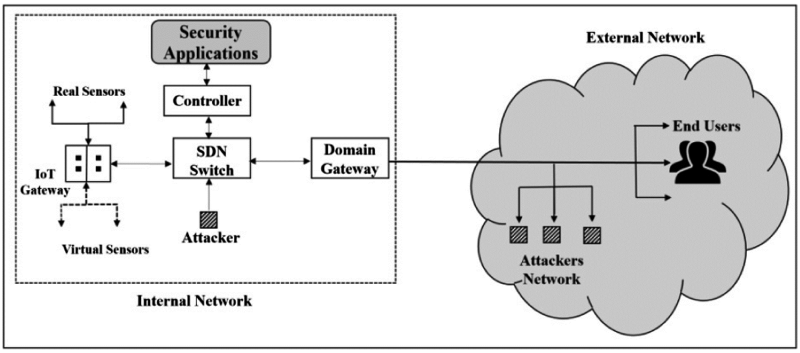

Fig. 7. Experimental network topology

7.2 Case Study: DDoS HTTP BOTNET ATTACK

Attack: A distributed DoS attack is usually mounted by a botnet, which uses a fleet of its victim machines who have legitimate IP address (no spoofing, hence difficult to detect and track and do not exhibit explicit indicators or statistical anomalies).

Detection: Taking the article [8] as reference, we improved upon their work in two aspects: 1. eliminated the need for the HTTP server to inform the SDN application about the botnet. Essentially the botnet detection logic is implemented based on traffic flow analysis at the SDN switch itself. 2. Optimised the detection processing overhead by employing better algorithm implementation approaches. The removal of the back-channel communication overhead between the HTTP server and SDN application (DBA) itself saved us major cycles. We conducted similar experiments and demonstrated that our mechanism has better performance, more portable and with no modifications to the target server environment.

Experiment:
SDN Defence policy: HTTP DoS blocking application runs in the SDN stack.

1. If the number of new connections/rate of new connection attempts exceed a threshold (in this case, 350 connections and 1 connection/second), then it's concluded that a botnet is active from external network. Drop the connections and packets to that destination address of the HTTP server D.
2. Send redirect message with a new destination address D encoded in the HTTP Response, it's assumed that the botnet are not programmed to decode the redirection scheme.
3. The legitimate clients will re-establish new connection to the D (address it is able to decode from the redirect response botnets are expected to continue attacking the original victim address D and are dropped.
4. Any new connections established to HTTP server at D, will be processed through the same detection logic.

Results: The Fig. 8(a) shows the botnet connections reaching the threshold of 350 connections, at which point all connections to destination D are dropped. And at the same time, the SDN application sends a HTTP response with redirection to new HTTP server D'. The genuine HTTP clients then establish new connections to D' which is shown in Fig. 8(b). There is an outage of few seconds (less than 3 s) for the genuine HTTP traffic and it's in acceptable as it's in new connection establishment phase. As we can see from the graphs that the overhead for packet processing by the SDN application at the SDN gateway switch is optimal (less than 3 s) and using dynamic flow rule learning entropy analysis, we can make this botnet detection mechanism responsive and practical for deployment in production IoT network.

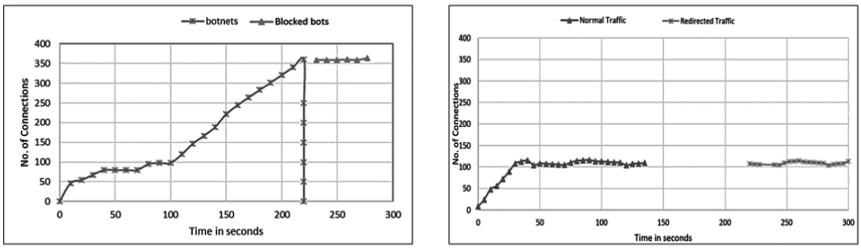

Fig. 8. (a) Botnet attack mitigation dynamics (b) genuine HTTP connections

7.3 Case Study: DoS Flooding Attack

Our experimental consisted of simple setup with IoT Gateway acting as a target of the attack connected behind an SDN gateway in the internal network.

Attack: Attacker nodes are simulated by running LOIC & hPing DoS attack tool from a group of nodes from the external network.

Detection: The DoS detection mechanism running as an SDN application on controller platform, executes a statistical function and analyses each flow based on threshold rates and based on the result, it installs new actionable rules on the data plane SDN switches - to forward or drop packets to the internal network. We based our experiment and compared with the work of [1] and we also ran the Cbench with identical setup, we demonstrated the efficacy of our defense mechanism in terms of implementation approach processing overhead in the SDN gateway.

Experiment: As the IoT link bandwidth is typically constrained by power and speed, we configured a peak link bandwidth of 2.5 Mbps. The genuine TCP traffic is run at 2 Mbps and after a while we ran attack traffic saturating the link at 2.5 Mbps.

Results: Figure 9a shows that at 7 s, the attack traffic kicked in to saturate the link and the genuine traffic was disrupted. But the DoS detection mechanism intercepted those attack traffic and in less than 3 s the bandwidth is recovered for the genuine TCP traffic. The DoS Flooding traffic is dropped at the SDN data plane itself without impacting the

SDN stack. Figure 9b shows the SDN controller performance in terms of number of flow installations per second. About 4.2 average flow installations per second on our DoS attack detecting switch compared to an average 7 flow installations per second on the standard L2 learning switch. So our work has clearly improved the agility of the DoS detection mechanism with SDN, compared to the prior work [1].

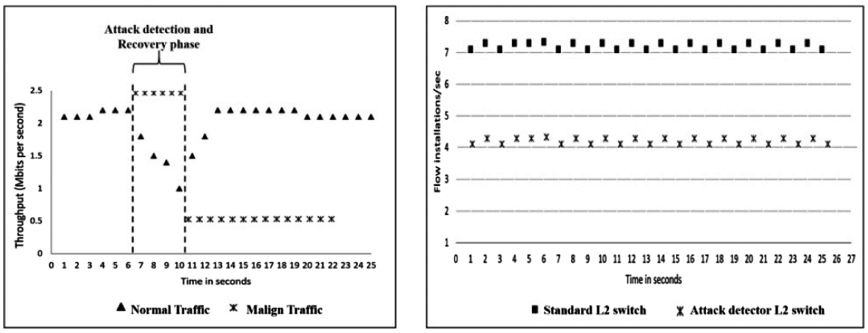

Fig. 9. (a) Link Saturation (b) controller performance

8 Conclusion

In this paper, we have discussed the potential of SDN and its capabilities such as traffic engineering and monitoring dynamic policy enforcement, access control at run time and mobility of devices. We conducted extensive simulations and the results confirm that the SDN-based-IoT applications can detect and mitigate the DoS attacks systematically. We developed reference applications for security policy and access control, in our IoT testbed using Openflow/REST interfaces and the results are proving the feasibility and efficacy of SDN in IoT networks.

Hence we make a strong case for SDN that privacy, trust and security policies can be efficiently enforced in IoT networks. This paper has provided an overview of challenges in IoT security, emphasized the need for flexible and dynamic methods of IoT network security, integration of SDN in IoT network. Our future work will expand these initial experiments to real Industrial IoT networks, fine tune and improve our design choices and position us to develop more efficient implementation to realize a SDN security framework for IoT applications. We believe that, our work has provided a practical proof and direction for applying SDN and other software-defined architectures to tackle extreme proliferation of IoT devices and deploy secure IoT networks for smart applications.

References

1. Bull, P., Austin, R., Popov, E., Sharma, M., Watson, R.: Flow based security for IoT devices using an SDN gateway. In: 2016 IEEE 4th International Conference on Future Internet of Things and Cloud (FiCloud), pp. 157–163. IEEE (2016)

2. Lee, W., Nam, K., Roh, H.G., Kim, S.H.: A gateway based fog computing architecture for wireless sensors and actuator networks. In: 2016 18th International Conference on Advanced Communication Technology (ICACT), pp. 210–213. IEEE (2016)
3. Choi, Y.: Implementation of content-oriented networking architecture (CONA): a focus on DDoS countermeasure. In: Proceedings of 1st European NetF-PGA Developers Workshop (2010)
4. Zhang, Y.: An adaptive flow counting method for anomaly detection in SDN. In: Proceedings of the Ninth ACM Conference on Emerging Networking Experiments and Technologies, pp. 25–30. ACM (2013)
5. Ezefibe, C.A., Shayan, Y.R.: Towards virtualisation and secured software defined networking for wireless and cellular networks. In: 2016 IEEE Canadian Conference on Electrical and Computer Engineering (CCECE), pp. 1–5. IEEE (2016)
6. Lin, H., Sun, L., Fan, Y., Guo, S.: Apply embedded openflow MPLS technology on wireless openflow–openRoads. In: 2012 2nd International Conference on Consumer Electronics, Communications and Networks (CECNet), pp. 916–919. IEEE (2012)
7. Flauzac, O., Gonzalez, C., Nolot, F.: Developing a distributed software defined networking testbed for IoT. Procedia Comput. Sci. **83**, 680–684 (2016)
8. Lim, S., Ha, J., Kim, H., Kim, Y., Yang, S.: A SDN-oriented DDoS blocking scheme for botnet-based attacks. In: 2014 Sixth International Conference on Ubiquitous and Future Networks (ICUFN), pp. 63–68. IEEE (2014)
9. Dinesh, M.K., Bhakthavatchalu, R.: Storage memory/NVM based executable memory interface IP for advanced IoT applications. In: 2016 International Conference on Recent Trends in Information Technology (ICRTIT), pp. 1–9. IEEE (2016)
10. Tortonesi, M., Michaelis, J., Morelli, A., Suri, N., Baker, M.A.: SPF: an SDN-based middleware solution to mitigate the IoT information explosion. In: 2016 IEEE Symposium on Computers and Communication (ISCC), pp. 435–442. IEEE (2016)
11. Vandana, C.: Security improvement in IoT based on Software Defined Networking (SDN). Int. J. Eng. Technol. Res. (IJSETR) **5**(1), 291–295 (2016)
12. Xu, T., Gao, D., Dong, P., Zhang, H., Foh, C.H., Chao, H.C.: Defending against new-flow attack in SDN-based internet of things. IEEE Access **5**, 3431–3443 (2017)