EDITOR: Rajiv Ranjan, raj.ranjan@ncl.ac.uk

DEPARTMENT: BLUE SKIES RESEARCH

Securing SDN Controllers in Internet of Things Environments

Rajiv Ranjan, *Newcastle University, UK*

The Blue Skies Research column is intended to identify the most important cutting-edge research for the edge-cloud continuum. The complexity around such research is huge as it sits at the intersection of multiple, interdependent disciplines involving complex systems and technologies. Example research areas include Internet of Things, Big Data Analytics, Cloud Computing, and Edge Computing. We will cover one specific research topic in each issue as it relates to the theme of the issue.

In this issue, we cover research work around securing Software Defined Network (SDN) controllers in IoT Environments, jointly developed by Duaa AlQattan, Omer Rana, Graham Morgan, Khaled Alwasel, Ayman Noon and myself. Specifically, it focuses on the new attack surface that is formed as a result of the deployment of SDN-IoT. The integration of SDN with the architecture of the IoT network has resulted in the network being made susceptible to new vulnerabilities, which may be used to launch attacks on the SDN controller. The vulnerabilities that may be exploited by controller attacks are described in depth, along with the challenges and potential research direction that arise when attempting to protect controllers against such attacks.

IoT systems generally have three infrastructure layers: (i) network of sensors and actuators that monitor and react to their environment; (ii) edge devices that receive data from these sensors or support actuation; (iii) a cloud-based data center connected over a multi-hop network, if data processing cannot be supported by edge devices or other cases, such as when access to legacy data or systems, better protection from DDoS, a single origin database, and so on are required. Understanding how data exchange across these three layers can be coordinated while ensuring cyber-resilience remains a challenge, and the use of software-managed network controllers to support these layers could benefit infrastructure operators and applications that make use of these layers.

SDN controllers employ software-based technologies to dynamically construct and control network components, supporting network stability and security by offering automated decision-making, routing optimization, network policies implementation and multipath secure transmission. As IoT ecosystems rely on processing and transmission of tasks, SDN may be used to simplify and accelerate task execution by reconfiguring interaction between the three layers identified above. SDN-IoT integration improves network performance and administration. Despite the benefits of SDN, SDN-IoT increases the potential attack surface and exposes new vulnerabilities and security breaches for the controller and the systems it manages. As an SDN controller is the main component that regulates and controls network and traffic in SDN-IoT systems, it can be vulnerable to attacks that might cause system damage and risk human life and safety, especially in safety-critical systems. This article focuses on threats to SDN controllers in SDN-IoT environments, identifying the types of attacks that can be launched against SDN controllers within an IoT environment.

SDN AND IOT INTEGRATION

IoT devices can use sensor data to control a physical environment in real time. As IoT devices have limited processing capacity, the generated data may be sent to an edge or cloud system for processing. A car in an Internet of Vehicles (IoV) network (a sub-paradigm emerging from IoT), for instance, can use sensors to interact with other vehicles and roadside units (RSUs) to prevent accidents. The combined use of vehicle sensors and RSUs can also be used to forward vehicle failures or emergency data to a cloud system, allowing the vehicle manufacturer (or a third party responsible for the maintenance of the vehicle) to monitor car

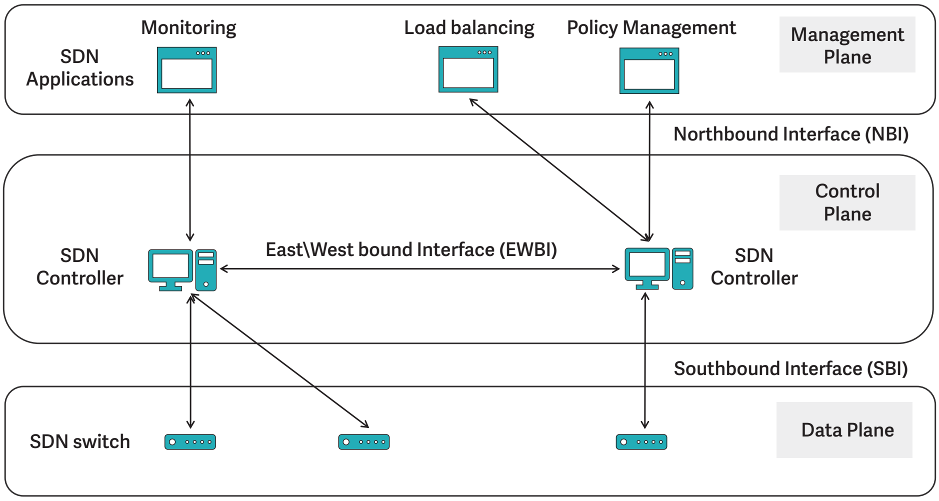

FIGURE 1. SDN controller interaction with Northbound and Southbound interfaces.

production and improve the manufacturing process for future vehicles. An SDN controller is increasingly being used in this scenario as it decouples network control tasks from tasks that involve data processing, to manage IoT network performance easily and automatically in real time, hence improving IoT network performance.

Next, we define the three layers of SDN-IoT architecture (see Figure 1) in more detail: *The perception and actuation layer* comprise sensors and actuators in large-scale distributed IoT systems. A sensor may cause an event that demands a real-time actuator response. Sensor data is sent to more powerful devices at *the collection layer* (that is, edge gateway). At edge devices, data is processed, analyzed, and stored, along with IoT actuator commands. Sensor networks link to gateways and edge nodes via ZigBee, BLE, low-power Wi-Fi, NFC, and RFID. Some data may need additional processing; thus, it's sent to cloud data centers through WAN in a distribution layer using network protocols including 4G/5G, LTE, and long-range wide area network (LoRaWAN). Any distribution-layer communication must meet IoT system performance requirements, such as latency, bandwidth, throughput, density and mobility. The requirement management is the responsibility of the SDN controller. It is responsible for managing the forwarding devices through the Southbound interface (SBI). Developers may enhance the controller's services via programmability through the Northbound interface (NBI; see Figure 1). The NBI connects controller services. Different SDN controllers could also be deployed to form distributed controllers. These controllers communicate through the East/West-bound interface (EWBI). The distribution layer sends data to *the processing layer*. Data centers store and analyze huge quantities of data using thousands of virtual computers or non-virtualized hardware devices ranging from servers to supercomputers. The next section will describe the attack surface, which consists of many entry points that an attack could exploit in the SDN-IoT architecture.

ATTACK SURFACE OF SDN-IOT ARCHITECTURE

The attack surface of an SDN-IoT system can consist of multiple vulnerable entry points that an attacker could exploit. Vulnerabilities may be introduced at design time or may occur during system operation. An attacker can use one or several vulnerable points to attack the controller[1], requiring a defence mechanism to be cognisant of these entry points. The attack entry points in SDN-IoT architecture include: IoT devices, data forwarding devices, storage and computing servers, virtual machines and container frameworks, communication protocols, the controller operations software, and any humans involved at any of these points who may be subject to phishing or other social engineering or may create configuration errors. At each entry point, an attacker can utilize several threats to attack the controller. These

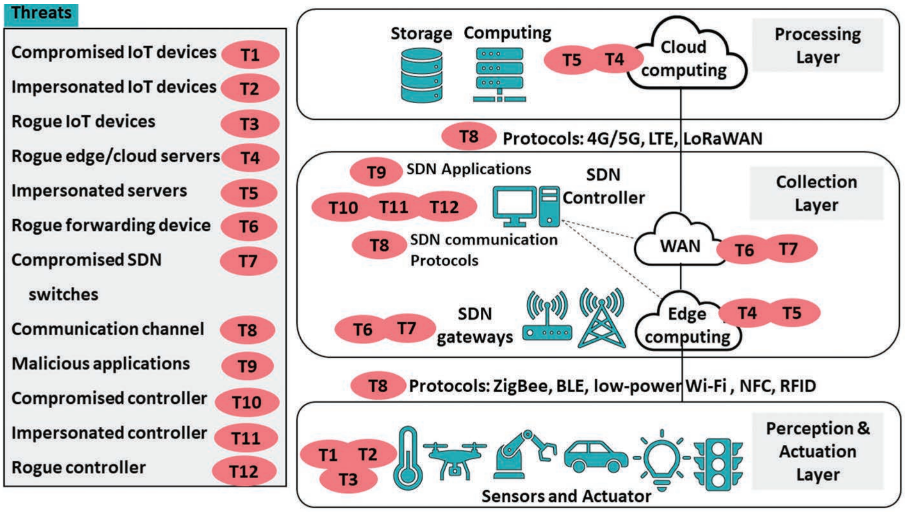
Threats
Compromised IoT devices T1
Impersonated IoT devices T2
Rogue IoT devices T3
Rogue edge/cloud servers T4
Impersonated servers T5
Rogue forwarding device T6
Compromised SDN switches T7
Communication channel T8
Malicious applications T9
Compromised controller T10
Impersonated controller T11
Rogue controller T12
Storage
Computing
T5 T4
Cloud computing
Processing Layer
T8 Protocols: 4G/5G, LTE, LoRaWAN
T9 SDN Applications
SDN Controller
T10 T11 T12
Collection Layer
T8 SDN communication Protocols
WAN
T6 T7
T6 T7
SDN gateways
Edge computing
T4 T5
T8 Protocols: ZigBee, BLE, low-power Wi-Fi, NFC, RFID
T1 T2 T3
Sensors and Actuator
Perception & Actuation Layer

FIGURE 2. Threats towards attacking the SDN controller.

can involve utilizing compromised, rogue or impersonated devices. These are described in Figure 2 and include compromised IoT devices (T1); impersonated IoT devices (T2); rogue IoT devices (T3); rogue edge/cloud servers (T4); impersonated edge/cloud servers (T5); rogue forwarding device (T6), compromised SDN switches (T7); communication channel (T8); malicious applications (T9); compromised controller (T10); impersonated controller (T11); rogue controller (T12). Details of these threats in different layers are explained hereafter (see Figure 2).

Threats in perception and actuation layer

Every IoT device can become part of a potential attack surface. These IoT devices are the greatest threat to SDN-IoT. Despite having different software and hardware environments, IoT devices are quite similar in their architecture and operation, although their data formats can vary significantly. All devices should have secure software, provide support for access management, data storage and data transfer. Vulnerabilities in any of these operations can render IoT devices vulnerable to attack.

The Open Web Application Security Project (OWASP)[2] presents a list of attacks and related vulnerabilities for IoT devices. Common weaknesses include

- disclosure of sensitive data stored in device memory such as unencrypted usernames, passwords, third-party credentials and encryption keys;
- misuse of physical device interfaces to extract device firmware, expose device ID/serial number, and the ability to reset a device to an insecure state;
- insecure device firmware can also lead to sensitive data exposure. Similarly, firmware version display and last update date can be used by an attacker to determine the current version of the firmware and potential vulnerabilities with that version;
- poorly implemented device network services such as firmware loaded over an insecure channel (no TLS/SSL), message integrity check or support for credential management; and
- lack of authentication and authorization between devices, device to edge/cloud servers, and device to web application.

Attackers may use these vulnerabilities to spread malware and corrupt IoT devices (T1) by downloading malware to IoT devices or enabling these devices to be enrolled in IoT botnets. Vulnerabilities in IoT devices may also be used to get private data and device identification, leading them to performing destructive activities such as spoofing controllers with impersonated devices (T2). Due to the lack of authentication and authorization, rogue IoT devices may pose a threat to SDN controllers (T3).

Threats in the collection layer

Threats in the collection layer include gateways and network devices, communication protocols, and controllers.

Gateways and network devices threats

Gateways and network devices (switches and routers) that link datacenters and IoT devices can include a combination of physical and virtual devices that send packets between nodes depending on a controller's decision, using a full-stack communications protocol, that gives access to the forwarding plane of an SDN switch or router over the network, such as OpenFlow.

OpenFlow switches have packet flow tables which may leak information without authentication or encryption. Servers host programmable switches (virtual switches) like Open vSwitch, with vulnerabilities in servers also exposing the virtual switches they host. Lack of authentication and authorization in an SDN controller makes SDN switches vulnerable. By exploiting this vulnerability, rogue network devices may be connected to a controller (T6).

To interact with a controller, an SDN switch must use a secure transport protocol like TLS, a protocol not available (or used) on some switches. An attacker may also use TLS flaws like faulty OpenFlow handshake authorization. SDN switches are vulnerable due to outdated and insecure operating systems with default administrator username and password, which can easily compromise the switch. Compromised SDN switches (T7) may attack the comptroller and the whole system.

Communication protocols threats

Vulnerabilities in communication protocols used by an edge gateway or WAN network devices can be used as entry points to attack an SDN controller. Some protocols lack connection authorization and authentication and lack of connection encryption or checking frame integrity. By exploiting these vulnerabilities, an attacker may be able to eavesdrop on network traffic (T8), leading to access to sensitive information. In addition, malicious packets may be inserted into the communication connections to modify controller behavior. Other protocols do not support out-of-band control, which implies that control data is sent on the same connection as primary data. Consequently, the connection is more vulnerable to be congested especially with insufficient bandwidth, this makes the controller disconnected with legitimate devices.

Controller threats

A vulnerable controller can be an attack entry point to compromise the controller itself or other controllers. Vulnerabilities of the SDN controller are associated with the

- controller operating system (OS);
- core functions supported on the controller, including its data storage and interfaces with other applications or data plane;
- network applications used to communicate with controller services; and
- communication interfaces with network applications (Northbound Interface) and forwarding devices (Southbound Interface).

An SDN controller may use a general-purpose OS, making it vulnerable owing to host OS vulnerabilities.[3] Some controller designs include weaknesses, such as absence of authentication and authorization in both interfaces, and unsecured communication with other SDN planes (application and data plane).[4]

Threats to the controller may emerge from applications (application plane) that operate over the Northbound Interface (NBI).[5] Network and third-party applications may access the controller's basic functionalities and storage to set rules, operate the SDN switch (data plane), or alter network state. Malicious applications (T9) may misuse these rights and attack the controller due to a lack of verification and authentication.[6] Most NBI and SBI protocols lack authentication and permission, making the controller vulnerable to malicious NB and SB communication.[7] If NB or SB communication channels are not encrypted, an attacker may eavesdrop on the information and utilize it for another attack (such as the Man In The Middle attack).[6]

These vulnerabilities can further compromise a controller (T10) or impersonation of a controller (T11), enabling an attacker to control the associated IoT network. The security of an SDN-IoT system can also be compromised by a rogue SDN controller (T12) that tries to change the settings of network components and communicate with other controllers.

Threats in processing layer

Significant vulnerabilities related to processing layer (cloud data centers) include the following:

- Storage servers (such as Amazon S3, Amazon Elastic File System)—which are used to host and manage IoT data on a long-term basis—could be leaked and eavesdropped by cyber attackers.
- Application user interfaces (APIs) that are intended to streamline computing processes.
- Virtualization infrastructure, which enables the deployment of several co-hosted services (owned and managed by different, unrelated application providers).

OWASP2 has also provided a list of vulnerabilities associated with the processing layer of IoT infrastructure. These vulnerabilities are

- insecure storage servers with lack of access restrictions and security group misconfiguration;
- insecure APIs that have weak authentication and weak access controls, lack of authentication/ authorization between IoT Devices to cloud servers and edge computing gateways;
- insecure administrative interface of edge gateways and cloud servers with no two-factor authentication and weak passwords.

The open nature of edge gateways contributes to adding more vulnerabilities where an attacker can develop their gateway devices and use them to eavesdrop on the network communication. In addition, improper access management of virtual machines (VM) and lightweight containers (such as Docker) on edge gateways and/or cloud servers might lead to exploiting the VMs and/or containers by attackers to execute malicious programs.

If these virtualized cloud servers, for instance, are left insecure, attackers can violate and control an entire data center by privilege escalation or by deploying rogue servers (T4). In this case, attackers can direct data to the compromised cloud server. In addition, attackers can tamper with all communication data exchanged from remote IoT devices via the Internet. The attacker can also leak the servers' identities (T5) to imitate the behavior of the server to spoof the network with false traffic. These behaviors can jeopardize the SDN controller that orchestrates the traffic in the distribution layer and make the controller unavailable by performing a DoS attack. While edge gateways are part of the distribution layer in Figure 2, they are vulnerable to Threats 4 and 5 like cloud servers.

Many attacks against the SDN controller may be launched due to the vulnerabilities discussed above from any layer of SDN-IoT architecture.

ATTACK TYPES AGAINST THE CONTROLLER OF SDN-IOT ARCHITECTURE

Attackers could target the SDN controller in the SDN-IoT architecture by seizing the vulnerabilities and the threats discussed in the previous sections. The attacker could use many mechanisms. These mechanisms are classified into types of attacks. Figure 3 illustrates the relationship between attack types (to be discussed next) and related threats (mentioned in the above sections).

Spoofing attack

By impersonating one of the legitimate network devices (T2,5,10), attackers can create malicious devices.[3] Following that, the attacker can request a connection between the malicious devices, the unsuspecting controller, and possible other legitimate network devices. Once the connection is established, the controller will start to receive the spoofed packet via a Packet-In messaging interface. This can lead to the controller being spoofed into believing that the impacted device has changed its location. As this happens, the controller will need to update the identification information of the impacted device to the false (or impersonated) identification information mimicked by the concerned device. In this case, if the controller or other devices try to communicate with an impacted device, the attacker will be able to receive the messages.[8] Consequently, there is a risk of an information disclosure attack since the malicious devices can freely exchange information with the controller as well as other legitimate networking devices. In addition, it has the potential to lead to a Denial-of-Service (DoS) attack due to malicious devices' ability to receive messages on behalf of the legitimate spoofed devices in the SDN-IoT environments.

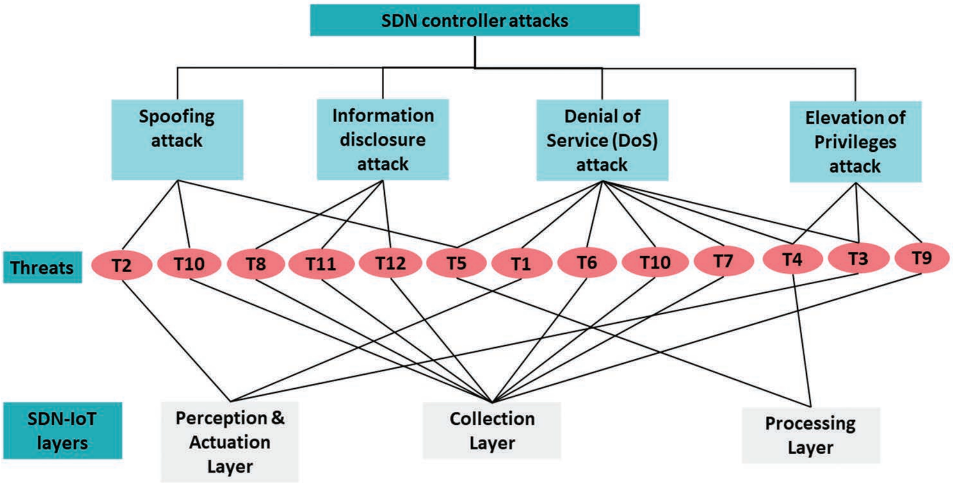

FIGURE 3. Taxonomy of Attack Types against the controller of SDN-IoT architecture.

Information disclosure attack

Information disclosure attacks specifically aim to expose the critical information flowing through the SDN-IoT environments. The centralized organization of the controllers makes them vulnerable to the network information attack (T9). As noted by recent research,[9] backup flow tables, configuration data, and network topology in the controller are all threatened by an information disclosure attack.[9]

Decoupling the controller plane from the forwarding plane, the core premise of the SDN paradigm, means sensitive messages (such as flow entries, statistics, and log information) could travel (see Figure 1) through the Northbound Interface channel to the applications, through the Southbound Interface channel to the forwarding devices and through the East/West Interface channel to another controller. Such multi-directional network control message flow leads to the risk of message disclosure and sensitive information leakage at various Interfaces (SBI, NBI, EWBI) due to insecure channels (T8) and unprotected messages[9] as well as communicating with a rogue controller (T12) or an impersonated controller (T11). Device spoofing attacks can be conducted to falsely receive the messages, which were intended for other legitimate networking devices. In this case, the attacker can conduct an information disclosure attack by acting as a Man in the Middle (MITM) and eavesdropping on sensitive information.

Denial of Service (DoS) attack

The centralization of the controller makes the entire network more vulnerable to Denial of Service (DoS) attacks compared to the traditional network. DoS attacks can target the controller, in the SDN-IoT environments, by making botnets send unwanted volumes of packets to compromised or rogue devices (T1,3,4,5,6,7,10) and/or the controller itself. Consequently, the controller becomes unreachable and/or becomes slow to respond to legitimate packets. As SDN switches are required to request a new rule from the controller for every new data flow, an attacker can send many new malicious packets to the switch, which in turn can overwhelm the controller with many unnecessary requests. A DoS attack could launch due to the vulnerabilities in the communication protocol (T8). An attack can achieve this by injecting packets continuously into the communication link at a high rate and sending them to the controller. Processing a large number of messages may cause the controller to be unavailable due to resource saturation and/or network saturation.

Elevation of Privileges attack

In an Elevation of Privileges attack (T9), the attacker utilizes intelligent tools to seek the possibility of accessing the controller's privileged information (such as the flow table). This attack targets authorization and access control modules in the controller to increase the attacker's access privileges. An Elevation of Privileges attack could be injected locally by an anonymous user or remotely by malicious applications and/or devices (T3,4). Once the attacker's privilege has been elevated, it can execute the controller level commands that jeopardise the functioning of the entire SDN-IoT network.

CONTROLLER PROTECTION FROM ATTACK IN SDN-IOT: CHALLENGES AND RESEARCH DIRECTIONS

To protect the controller from attacks in SDN-IoT, several challenges and research directions may be explored. We outline some research challenges and potential research directions below.

Attack Monitoring

Determining deviation from expected behavior (stable state vs attacked state) in any environment requires consideration of the interactions of actors, that in turn manifest in overall system behavior. In evolving systems such as IoT environments, simply predetermining rule-based behavior for determining attacks would be insufficient. This is because a system of evolved functionality may ascertain a context of behavior that may not be predetermined (such as new devices, updated protocols, and additional governance due to internationalised laws). Therefore, the ability to determine attacks through monitoring the large amount of data, internally generated through such aspects as message passing between devices, is probably the only practical manner with which to detect attacks in an evolving system. However, given the vast scale of these SDN-IoT environments, research is required to provide level of detail (LOD) strategies to determine just what is, and what is not, of a particular concern in attack detection. This lack of LOD consideration in the literature highlights this as a major opportunity in attack monitoring for IoT scale systems. As noted by ourselves[10] and others, existing available approaches in the cloud-edge-IoT continuum are not capable of quantifying and monitoring security attacks in the context of SDN-IoT environments.

Attack Detection

Handling large volumes of data for attack monitoring while focusing down on attack issues is in itself inexplicably linked to actual attack detection. Once LOD considerations are determined with the aid of an algorithm that has some ability to identify a heightened risk that something is an attack, then another algorithm must determine if an attack is occurring. This stepped approach is key for achieving scale while ensuring sufficient coverage and accuracy are maintained. To achieve this, the literature commonly employs AI techniques, focusing on the patterns of activity betrayed by the data itself. However, AI techniques do suffer from an inability to specifically indicate just why something is an attack, usually only identifying a pattern of behavior that resembled an attack previously. The issue of learned attack behavior not only must evolve itself but provide provenance of and evidence of an attack to ensure a guaranteed exposition of law breaking may be exhibited in a legal sense. As the current literature lacks usable SDN-IoT attack data sets, a formidable research challenge is how to produce such usable attack data sets which can be used to train data hungry AI techniques (such as Deep Neural Networks). To solve this challenge, the research community should investigate high fidelity simulations (https://rajivranjan.net/iotsim/iotsim-release/) and an integrative testbed (https://urbanobservatory.ac.uk/) that is scalable and configurable to facilitate cybersecurity research into SDN-IoT environments.

Attack Diagnosis

Assuming an attack can be identified as well as evidence- and provenance-secured, then there is a requirement to investigate system weaknesses in the context of how the attack was allowed to occur. The diagnosis of attack should provide an automated forensic approach for highlighting shortcomings in design and implementation, not to mention usage possibilities. This is a challenging, time consuming act and research is still far from a totally automated solution. In fact, this is probably the most manual-intensive aspect of the whole cybersecurity lifecycle in the context of general IoT environments including SDN-IoT. In essence, once an attack has occurred and has been detected, it then requires human intervention and many hours, if not months, of work process to diagnose issues, presenting the research community with its most difficult challenge. The automated response would allow greater freedom in handling attacks when they occur. This is possible but presents the most significant challenge for the general case.

Attack Mitigation

Attack mitigation may occur after a forensic analysis has brought forward a full diagnosis of an attack or during an attack itself. During an attack, one is reliant on rather blunt approaches of isolation, shutdown, or managed degradation. However, the research effort is toward self-healing activity or misdirection for the attacker (to localize the damaging effect). Research that provides more focused automated responses for mitigation than present will provide robustness that enables systems to continue in every greater degree unhindered from attacks. These focused AI-led initiatives to achieve attack mitigation and healing are key to ensuring automated continuation of service. In an IoT infrastructure this is problematic as many real-time services (such as transport) rely on life-critical systems and degradation, even cessation of service, may be favored if human life is at stake. Alternatively, the cessation of a service may not be possible for the same reasons (for example, self-driving vehicles in freeways). Attack mitigation in such circumstances is possibly the one most important research challenge in the area of IoT. Hence, the research community needs to investigate automate attack mitigation based on emerging reinforcement learning techniques (such as Deep Q-Learning). One of the challenges in undertaking attack mitigation actions in SDN-IoT environments is the prohibitively large reconfiguration search space. To resolve this issue, one can harness the integrated simulator platforms (https://rajivranjan.net/iotsim/iotsim-release/) for training of reinforcement learning agents to take best mitigation actions based on policies derived using multicriteria decision-making techniques.

CONCLUSION

With the increasing use of IoT devices within home, work and industry/factory environments, managing and controlling data stored on these devices is challenging. Combining the programmability of SDN controllers with IoT systems provides significant benefits, such as the ability to remotely manage, configure, and control devices using a controller interface. However, SDN-IoT controllers also lead to an increased attack surface, with a number of additional vulnerabilities. In this article, we describe the attack surface introduced by the adoption of an SDN-IoT environment. We describe how the development of the IoT network architecture, as well as its integration with SDN, has introduced new vulnerabilities to the network, which may be exploited to launch attacks on the SDN controller. A categorization of controller attacks that can use such vulnerabilities is outlined, along with challenges and research directions for protecting controllers from attacks.

REFERENCES

1. W. Stallings et al., Computer security: principles and practice. Pearson Upper Saddle River, 2012, vol. 2.
2. OWASP,"Owasp internet of things project -owasp." [Online], https://wiki.owasp.org/index.php/OWASP InternetofThingsProjecttab = IoTAttackSurfaceAreas.
3. Y. Tseng, F. Naït-Abdesselam, and A. Khokhar, "A comprehensive 3-dimensional security analysis of a controller in software-defined networking," *Security and Privacy*, vol. 1, p. e21, 3 2018. [Online]. Available: http://doi.wiley.com/10.1002/spy2.21
4. M. Iqbal et al., "Security issues in software defined networking (SDN): Risks, challenges and potential solutions," *IJACSA International Journal of Advanced Computer Science and Applications*, vol. 10, no. 10, pp. 298–303, 2019.
5. K. Nisar et al., "A survey on the architecture, application, and security of software defined networking: Challenges and open issues," *Internet of Things*, vol. 12, p. 100289, 12 2020.
6. Y. Maleh, Y. Qasmaoui, and K. El Gholami, A comprehensive survey on SDN security: threats, mitigations, and future directions. *J Reliable Intell Environ (2022)*. https://doi.org/10.1007/s40860-022-00171-8
7. C. C. Chica, J. C. Imbachi, and J. F. B. Vega, "Security in sdn: A comprehensive survey," *Journal of Network and Computer Applications*, vol. 159, p. 102595, 6 2020
8. T. Han et al., "A comprehensive survey of security threats and their mitigation techniques for next-generation sdn controllers," Concurrency and Computation: Practice and Experience, vol. 32, p. e5300, 8 2020. [Online], https://onlinelibrary.wiley.com/doi/abs/10.1002/cpe.5300
9. O. N. Foundation, "Threat analysis for the sdn architecture," 2016. [Online], www.opennetworking.org.
10. R. Ranjan et al., "The Next Grand Challenges: Integrating the Internet of Things and Data Science," Volume 5, Issue 3, Pages 12-26, May/Jun. 2018, *IEEE Cloud Computing*.