Security Against Distributed Denial of Service (DDoS) using Software-Defined Networking (SDN)

Mr. R.B. Sarooraj , Kshitiz Lall, S.Krishna Kumar, Shivam Patel, Ashwani Kumar

Abstract— **Aspiration to provide and analyze a behavior-rule specification-based technique for intrusion detection of medical devices embedded in a medical cyber physical system (MCPS) in which the patient's safety is of the utmost importance. We propose a methodology to transform behavior rules to a state machine, so that a device that is being monitored for its behavior can easily be checked against the transformed state machine for deviation from its behavior specification. Using vital sign monitor medical devices as an example; we demonstrate that our intrusion detection technique can effectively trade false positives off for a high detection probability to cope with more sophisticated and hidden attackers to support ultra safe and secure MCPS applications. Moreover, through a comparative analysis, we demonstrate that our behavior-rule specification-based IDS technique outperforms two existing anomaly-based techniques for detecting abnormal patient behaviours in pervasive healthcare applications.**

Index Terms- DDoS, SDN.

I. Introduction

DDoS stands for Distributed Denial of Service attack. It is a form of attack where a lot of zombie computers (infected computers that are under the control of the attacker) are used to either directly or indirectly to flood the targeted server(s) – victim, with a huge amount of information and choke it in order to prevent legitimate users from accessing them (mostly web servers that host websites). In most cases, the owners of the zombie computers may not know that they are being utilized by attackers. In some cases, there is only a periodic flooding of web servers with huge traffic in order to degrade the service, instead of taking it down completely. Software-defined networking is not a technology, but an architecture that provides support for virtual machine mobility independent of the physical network. Over the last year, the hottest topics in networking have been software-defined networking (SDN) and Network Virtualization (NV) have been one of the hottest topics in networking. There is, however, considerable confusion amongst enterprise IT organizations relative to these topics. There are many sources of that confusion, including the sheer number of vendors who have solutions that solve different problems using different solution architectures and technologies, all of whom claim to be offering SDN and/or NV solutions.

A. *Medical Cyber-physical system*

The most prominent characteristic of a medical cyber physical system (MCPS) is its feedback loop that acts on the physical environment. In other words, the physical environment Provides data to the MCPS sensors whose data feed the MCPS control algorithms that drive the actuators which change the physical environment. MCPSs are often characterized by sophisticated patient treatment algorithms interacting with the physical environment including the patient. In this paper, we are concerned with intrusion detection mechanisms for detecting compromised sensors or actuators embedded in an MCPS for supporting safe and secure MCPS applications upon which patients and healthcare personnel can depend with high confidence. Intrusion detection system (IDS) design for cyber physical systems (CPSs) has attracted considerable attention because of the dire consequence of CPS failure. However, IDS techniques for MCPSs are still in its infancy with very little work reported. Intrusion detection techniques in general can be classified into four types: signature, anomaly, trust, and specification-based techniques. In this paper, we consider specification rather than signature-based detection to deal with unknown attacker patterns. We consider specification rather than anomaly based techniques to avoid using resource-constrained sensors or actuators in an MCPS for profiling anomaly patterns (e.g., through learning) and to avoid high false positives.

B. *monitoring of attacks in ddos*

DDoS attacks come in many different forms, from Smurfs to Teardrops, to Pings of Death. Below are details about the types of attacks and amplification methods found on the map.

a) Session Monitoring
b) Analysis of Dataset
c) Static Model Building Algorithm
d) Attack Detection

a) Session Monitoring

In this Session Monitoring each user session into a different container; however, this was a design decision. For instance, we can assign a new session per each new IP address of the client. In this implementation, sessions were recycled based on events or when sessions time out. This will able to use the same session tracking mechanisms as implemented by the Apache server because lightweight virtualization containers do not impose high memory and storage overhead. Thus, we could maintain a large number of parallel-running Apache instances similar to the Apache threads that the server would

Mr. R.B. Sarooraj, Asst. Prof. **.** Computer Science and Engineering, SRM University, Chennai, India
Kshitiz Lall, Computer Science and Engineering, SRM University, Chennai, India
S.Krishna Kumar, Computer Science and Engineering, SRM University, Chennai, India
Shivam Patel, Computer Science and Engineering, SRM University, Chennai, India
Ashwani Kumar, Computer Science and Engineering, SRM University, Chennai, India

maintain in the scenario without session containers. If a session timed out, the Apache instance was terminated along with its container. In our prototype implementation, we used a 60-minute timeout due to resource constraints of our test server.

b) *Analysis of Dataset*

Based on the web server's application logic, different inputs would cause different database queries. For example, to post a comment to a blog article, the web server would first query the database to see the existing comments.

If the user's comment differs from previous comments, then the web server would automatically generate a set of new queries to insert the new post into the back-end database. Otherwise, the web server would reject the input in order to prevent duplicated comments from being posted (i.e., no corresponding SQL query would be issued). In such cases, even assigning the same parameter values would cause different set of queries, depending on the previous state of the website.

Likewise, this nondeterministic mapping case (i.e., one-to-many mapping) happens even after we normalize all parameter values to extract the structures of the web requests and queries. Since the mapping can appear differently in different cases, it becomes difficult to identify all of the one-to-many mapping patterns for each web request. Moreover, when different operations occasionally overlap at their possible query set, it becomes even harder for us to extract the one-to-many mapping for each operation by comparing matched requests and queries across the sessions.

c) *Static Model Building Algorithm*

An algorithm that takes the input of training data set and builds the mapping model for websites. For each unique HTTP request and database query, the algorithm assigns a hash table entry, the key of the entry is the request or query itself, and the value of the hash entry is AR(Admin Request) for the request or AQ (Admin Query) for the query, respectively. The algorithm generates the mapping model by considering all mapping patterns that would happen in websites.

c) *Attack Detection*

The attacker visits the website as a normal user aiming to compromise the web server process or exploit vulnerabilities to bypass authentication. At that point, the attacker issues a set of privileged (e.g., admin-level) DB queries to retrieve sensitive information. We log and process both legitimate web requests and database queries in the session traffic, but there are no mappings among them. Double Guard separates the traffic by sessions.

If it is a user session, then the requests and queries should all belong to normal users and match structurally. Using the mapping model that we created during the training phase, Double Guard can capture the unmatched cases. We establish the mappings between HTTP requests and database queries, clearly defining which requests should trigger which queries. For an SQL injection attack to be successful, it must change the structure (or the semantics) of the query, which our approach can readily detect.

First of all, according to our mapping model, DB queries will not have any matching web requests during this type of attack. On the other hand, as this traffic will not go through any containers, it will be captured as it appears to differ from the legitimate traffic that goes through the containers. Double Guard is designed to mitigate DDoS attacks. These attacks can occur in the server architecture without the back-end database.

II. Existing Approach

Since Yahoo, Amazon and other well-known web sites were subjected to DDoS attacks in 2000, researchers have presented many methods to mitigate DDoS attacks. The defence mechanisms against network/transport-level DDoS flooding attacks can be classified into four categories based on the deployment location

a) Source-based mechanisms
b) Network-based mechanisms
c) Destination-based mechanisms
d) Hybrid-based mechanisms

a) Source-based mechanisms

Source-based mechanisms are deployed near the sources of the attack to prevent network customers from generating DDoS flooding attacks. Some examples of source-based mechanisms including ress/egress filtering, which filters packets with spoofed IP addresses at the source's edge routers based on the valid IP address range internal to the network, and Source Address Validity Enforcement (SAVE) Protocol. SAVE protocol enables routers to update the information of expected source IP addresses on each link and block any IP packet with an unexpected source IP address.

b) Network-based mechanisms

These mechanisms are deployed inside networks and mainly on the routers of the Autonomous Systems (ASs). Generally there are two groups of DDoS attack detection techniques. The first group is called DDoS-attack-specific detection. The second group is called anomaly-based detection (e.g., route-based packet filtering). DDoS-attack-specific detection is an attack detection method based on the special features of different DDoS attacks. For example, using SYN cookie technique can resist SYN flood attacks. Ina SYN flood,a victim server receives spoofed SYN request sata high packet rate that contain fake source IP addresses. The SYN flood overwhelms the victim server by depleting its system resources (connection table memory) normally used to store and process these incoming packets, resulting in performance degradation or a complete server shutdown. SYN cookies are particular choices of initial TCP sequence numbers by TCP servers. They allow a server to avoid dropping connections when the SYN queue fills up. Anomaly-based detection models the behavior of normal traffic, and then reports any anomalies

c) Destination-based mechanisms

In the destination-based defence mechanisms, detection and response is mostly done at the destination of the attack (i.e., victim). Some examples of destination-based mechanisms include input debugging, probabilistic packet marking, and hash-based IP trace back. Input debugging is a link testing mechanism, in which the trace back process starts from the router closest to the victim and iteratively tests its upstream links until it can be determined which link is used to carry the attacker's traffic. In probabilistic packet marking, routers in the path to the victim probabilistically mark packets (i.e., add routers' identification to each packet) so that the victim can identify the path of attack traffic and distinguish it from legitimate traffic after the detection. In hash-based IP trace back, routers in the path to the victim keep a hash record of every packet passing through the router using Bloom Filter, which is a hash structure to reduce the memory requirement to store packet records.

d) Hybrid-based mechanisms

Hybrid Defence mechanisms are deployed at (or their components are distributed over) multiple locations such as source, destination or intermediate networks and there is usually cooperation among the deployment points (e.g., Active Internet Traffic Filtering (AITF), which enables a receiver to contact misbehaving sources and ask them to stop sending it traffic).Existing work only considered specification based state machines for intrusion detection of communication protocol misbehaving patterns. Untreated in the literature, in this paper we also investigate the impact of attacker behaviors on the effectiveness of MCPS intrusion detection. The authors use an anomaly-based approach while we use a specification based approach. They provide numerical results that measure internal validity (the effectiveness of the data mining implementation) but do not provide externally valid metrics like Receiver Operating Characteristic (ROC) which can reveal the tradeoff between the detection rates versus the false positive probability.

System Architecture

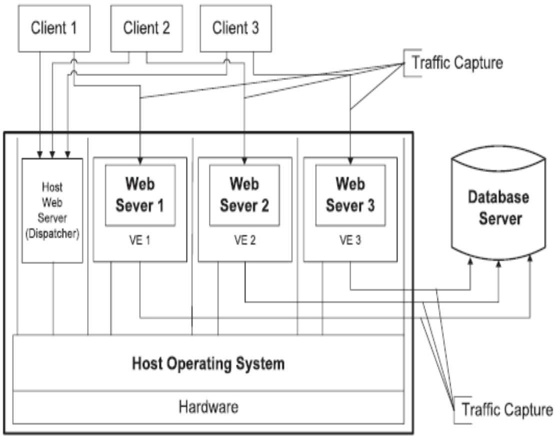

fig1. System Architecture of Server and Client

III. FUCTIONAL MODULES

Virtualization is the creation of a virtual (rather than actual) version of something, such as a hardware platform, operating system (OS), storage device, or network resources. While a physical computer in the classical sense is clearly a complete and actual machine, both *subjectively* (from the user's point of view) and *objectively* (from the hardware system administrator's point of view), a virtual machine is *subjectively* a complete machine (or very close), but *objectively* merely a set of files and running programs on an actual, physical machine (which the user need not necessarily be aware of). Virtualization can be viewed as part of an overall trend in enterprise IT that includes autonomic computing, a scenario in which the IT environment will be able to manage itself based on perceived activity, and utility computing, in which computer processing power is seen as a utility that clients can pay for only as needed. The usual goal of virtualization is to centralize administrative tasks while improving scalability and overall hardware-resource utilization. With virtualization, several operating systems can be run in parallel on a single central processing unit (CPU). This parallelism tends to reduce overhead costs and differs from multitasking, which involves running several programs on the same OS.

IV. DDOS ATTACK DEFENCE MECHANISMS USING SDN

SDN brings us new chances to defeat DDoS attacks in cloud computing environments.

1) Separation of the control plane from the data plane: SDN decouples the data plane and control plane and thus enables to establish easily large scale attack and defence experiments. High configurability of SDN offers clear separation among virtual networks permitting experimentation on a real environment. Progressive deployment of new ideas can be performed through a seamless transition from an experimental phase to an operational phase. Moreover it enables innovation and evolution by providing a programmable network platform to implement, experiment, and deploy new ideas, new applications. The feature of SDN brings great convenience in putting forward new thoughts and methods of DDoS attacks mitigation.

2) A logical centralized controller and view of the network: The controller has network-wide knowledge of the system and global views to build consistent security police and to monitor or analyses traffic patterns for potential security threats. Centralized control of SDN permits dynamically quarantine of compromised hosts and authentication of legitimate hosts based on information obtained through requesting end hosts, requesting a Remote Authentication Dial In User Service (RADIUS) server for users' authentication information and system scanning during registration.

3) Programmability of the network by external applications: The programmability of SDN supports a process of harvesting intelligence from existing Intrusion Detection Systems (IDSs) and Intrusion Prevention Systems (IPSs). More intelligent algorithms can be flexibly used based on different DDoS attacks.

4) Software-based traffic analysis: Software-based traffic analysis greatly enables innovation, as it is possible to improve the capabilities of a switch using any software-based technique. Traffic analysis can be performed in real time using machine learning algorithms, databases and any other software tool. Traffic of interest can be explicitly directed to IPSs for Deep Packet Inspection (DPI).

5) Dynamic updating of forwarding rules and flow abstraction: Dynamic updating of forwarding rules helps promptly respond to DDoS attacks. Based on the analysis, new or updated security policy can be propagated across the network in the form of flow rules. If attacks are detected, SDN can install packet forwarding rules to switching devices

to block the attack traffic from entering and propagating in a network.

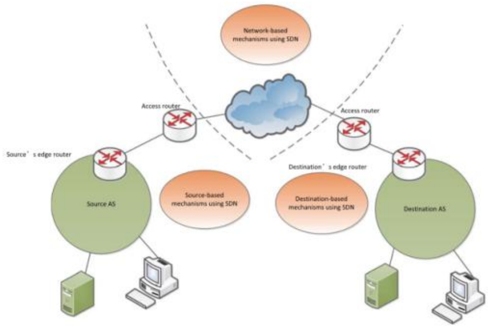
Network-based mechanisms using SDN
Access router
Access router
Source's edge router
Destination's edge router
Source AS
Source-based mechanisms using SDN
Destination-based mechanisms using SDN
Destination AS

fig2. A classification of the defence mechanisms against DDoS attacks using SDN.

V. Flow Diagram of the Entire Process

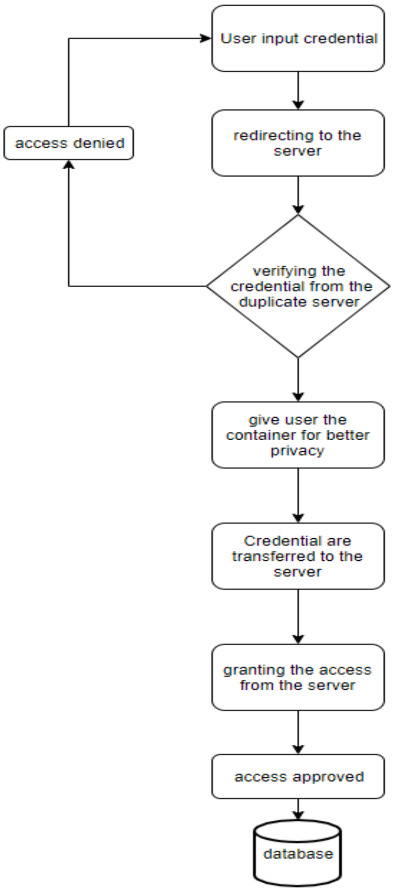
User input credential
redirecting to the server
access denied
verifying the credential from the duplicate server
give user the container for better privacy
Credential are transferred to the server
granting the access from the server
access approved
database

fig3. Flow diagram of MCPS

VI. Conclusion

In this paper, we first discussed the reasons why DDoS attacks are growing in cloud computing environments. Then we summarized the difficulty in defeating DDoS attacks in cloud computing environments. We provided a comprehensive survey on some of the works that have already been done to defend DDoS attacks using SDN. In summary, SDN brings a fascinating dilemma: a promising tool to defeat DDoS attacks in cloud computing environments, versus a vulnerable target to DDoS attacks. It is in favor of the community to study how to make full use of SDN's advantages to defeat DDoS attacks and how to prevent SDN itself becoming a victim of DDoS attacks in cloud computing environments

Acknowledgment

I would like to express my deepest gratitude to my guide, Mr. R. B. Sarooraj for his valuable guidance, consistent encouragement, personal caring, timely help and providing us with an excellent atmosphere for doing project. All through the work, inspite of his busy schedule, he has extended cheerful and cordial support to us for completing this research work.

References

[1] G. Pallis, "Cloud computing: The new frontier of Internet computing," IEEE Internet Comput., vol. 14, no. 5, pp. 70–73, Sep. 2010.

[2] T. Taleb, "Toward carrier cloud: Potential, challenges, and solutions," IEEE Wireless Commun., vol. 21, no. 3, pp. 80–91, Jun. 2014.

[3] F. R. Yu and V. C. M. Leung, Advances in Mobile Cloud Computing Systems. New York, NY, USA: CRC Press, 2015.

[4] Y.-D. Lin, D. Pitt, D. Hausheer, E. Johnson, and Y.-B. Lin, "Software defined networking: Standardization for cloud computing's second wave," Computer, vol. 47, no. 11, pp. 19–21, Nov. 2014.

[5] R.-I. Chang and C.-C. Chuang, "A service-oriented cloud computing network management architecture for wireless sensor networks," AdHoc Sens. Wireless Netw., vol. 22, no. 1/2, pp. 65–90, 2014.

[6] Z. Yin, F. R. Yu, S. Bu, and Z. Han, "Joint cloud and wireless networks operations in mobile cloud computing environments with telecom operator cloud," IEEE Trans. Wireless Commun., vol. 14, no. 7, pp. 4020–4033, Jul. 2015.

[7] Y. Cai, F. R. Yu, and S. Bu, "Dynamic operations of cloud radio access networks (C-RAN) for mobile cloud computing systems," IEEE Trans. Veh. Tech., accepted for publication, DOI: 10.1109/TVT.2015.2411739.

[8] F. Hu, Q. Hao, and K. Bao, "A survey on software-defined network (SDN)and openflow: Fromconcept toimplementation," IEEECommun. Surveys Tuts., vol. 16, no. 4, pp. 2181–2206, 4th Quart. 2014.