2018 17th IEEE International Conference On Trust, Security And Privacy In Computing And Communications/ 12th IEEE International Conference On Big Data Science And Engineering

An Implementation of Hierarchical Intrusion Detection Systems Using Snort and Federated Databases

Sarvjot Kaur Kang[1], Dale Lindskog [2], Hamman Samuel[3]
Information System Security Management, Concordia University of Edmonton, Canada
[1]skang1@student.concordia.ab.ca, [2]dale.lindskog@concordia.ab.ca, [3]hamman.samuel@concordia.ab.ca

Abstract – **This research presents a pragmatic implementation of a hierarchical distributed intrusion detection system. Several hierarchical distributed intrusion detection architectures have been proposed for use in various network topologies. However, to our knowledge, practical implementations of these solutions have not been explored. This study proposes to implement such an architecture using a combination of Snort and MySQL databases. Intrusion Detection Systems may act as defensive mechanisms, since they monitor network activities in order to detect malicious actions performed by intruders, and then initiate the appropriate countermeasures. This research work also shows the root node is at the top of the IDS hierarchy and receives aggregated/consolidated intrusion detection information from the entire network by using federated databases. Intrusion detection occurs at the cluster head of each cluster by gathering data from cluster members to have faster detection. Ultimately, this is an initial step towards evaluation of hierarchical intrusion detection approaches.**

Keywords–Intrusion detection systems, Snort, MySQL, Federated storage engines

I. INTRODUCTION

An Intrusion Detection System (IDS) is a hardware and software application which monitors various networks and systems for intrusions, network packets, root kit analysis, and/or system logs. IDS also keep checking on the network and system for various malicious events which can intrude and crumble the functioning system.

This approach can also work for mobile ad hoc network topologies, so it is important to understand the basics of mobile ad hoc networks, A mobile ad hoc network (MANET) is a collection of autonomous nodes that form a dynamic, purpose-specific, multi-hop radio network in a decentralized fashion. An effective way to identify when an attack occurs in a MANET is the deployment of an Intrusion Detection System.

An IDS can be divided in two main parts: (a) the architecture, which exemplifies the operational structure of the IDS; and (b) the detection engine, which is the mechanism used to detect malicious behavior. Moreover, the existing IDS architectures for MANETs fall under three basic categories: [1] (a) stand-alone, (b) cooperative and (c) hierarchical. A detailed description about architectures is provided in a second section of the document. This research document mainly focuses on the implementation of the hierarchical architecture.

Chadli et.al. [1] investigated several existing intrusion detection approaches to learn how they have implemented their intrusion detection. Moreover, they proposed a hierarchical intrusion detection system based on a synthesis of the existing approaches. The limitation of this paper they didn't focus on implementing their proposed architecture on a simulator.

In this research document for a multi-layered infrastructure, Hierarchical IDS (HIDS) architecture has been implemented, which is organized into levels, with a root node present at the top of hierarchy. The network is subdivided into groups called clusters. Each cluster has its cluster head which acts as a control point and has more responsibility than other nodes for providing communication to other cluster heads and zones. In this architecture, local detection is carried out by a cluster, whereas global detection is carried out by cluster heads and inter-zone nodes.

We used a federated storage engine to implement the HIDS hierarchy. A federation layer in an application that unifies underlying systems, or adapts them to look as a single unified system. [2]. Hence, for this implementation, we have used the federated engine of the MySQL database management system to store the output generated by Snort. The MySQL federated storage engine for the MySQL is a storage engine which allows a user to create a table that is a local representation of a foreign table.

A federated table is a virtual table which points to a table in MySQL database instance on a different server. We can access the data in the table on any node. The main objective of this research is to implement a HIDS by using Snort and MySQL. First of all, various software applications are needed to install to accomplish the goal of research. Another challenging aspect of the research is to use federated storage engines to convert the IDS structure from distributed to hierarchical.

II. REVIEW OF RELATED RESEARCH

There are several proposed IDS architectures that depend upon the underlying network architecture. It has been found that many researchers [3] [4] [5] [6] have either modified the existing ideas or proposed new IDS architectures. But their implementation and practical work is still pending.

Bedi et al. [7] summarized the brief descriptions of IDS architecture, IDS Techniques, types of attacks detected and data gathering techniques, followed by the author's comments on strength, weaknesses and limitations of each technique.

2324-9013/18/31.00 ©2018 IEEE
DOI 10.1109/TrustCom/BigDataSE.2018.00216

Moreover, this paper summarized the most prominent IDS Architectures for MANETs published in the last five years which shows that local detection has low detection rates but works well in some cases with mobility factor. Hierarchical architecture is complex in building clusters and reelection procedure due to high mobility in node [7].

To improve the efficiency of IDS operations, researchers have proposed different IDS architectures as per requirements of the deployment. [1] The existing IDS architectures fall under three basic categories discussed in the following sections.

A. Stand-Alone Architecture

In stand-alone IDS architecture, the IDS agent is installed on each node and runs independently to detect intrusions locally. But these types of IDS cannot resolve attacks on a wider range because nodes do not involve themselves in a cooperative detection.

B. Cooperative & Distributed Architecture

In this type, an IDS agent is installed on each node of the network as that of stand-alone IDS. But here, the IDS agent's responsibilities involve collecting evidence locally, then making decisions and sharing the audit data with the neighboring detection systems to detect attacks on a wider range. Zhang et al. [8] proposed this type of architecture by considering the salient features of the mobile ad hoc network. They also examined the vulnerabilities of wireless networks and argued that there should be intrusion detection in the security architecture for mobile computing environments. They developed such an architecture and evaluated a key mechanism in this architecture. But this architecture and various other Cooperative IDS architectures, such as the Friend assisted IDS architecture, the Cooperative IDS architecture based on social network analysis, etc., have limitations, studied and analyzed by Chadli et al [1]. They argued that, for the entire set of architectures of this type that they studied, the ratio of the false positives being comparatively high and detection is negatively affected by high node mobility.

C. Hierarchical IDS Architecture

This is an advanced version of the distributed and cooperative IDS architecture, proposed for multi-layered network infrastructure where the network is divided into clusters and each cluster has a cluster head. Clustering means virtual partitioning of the network and arrangement of the nodes into clusters [1].

Hierarchical IDS (HIDS) have a few advantages over traditional IDS. Firstly, most of hierarchical systems attempts to detect attacks faster (either by employing multiple layers of detection, or by employing one cluster-head to monitor large portions of a network, or by monitoring the elected cluster heads). Secondly, some of them focus on the fair distribution of the processing workload among nodes (either by considering nodes battery power, or by rotating cluster-heads). Thirdly, a few of them try to eliminate the imposed processing and communication overhead (by employing a detection mechanism based on voting or by selecting cluster-heads with the objective of "last longer").

Various research studies have covered the HIDS concept and variations. Albers et al. [9] examined the different intrusion detection techniques and point out the reasons why they usually cannot be used in an ad hoc context. Moreover, they went through the requirements of an intrusion detection system for ad hoc networks, and defined an adapted architecture for an intrusion detection system.

Kaur et al. [10] studied intrusion response in mobile ad hoc networks and theorized improvements to some existing dynamic and hierarchical IDS architecture for MANETs. This work aimed to theoretically enhance the hierarchical IDS architecture to form an underlying base IDS for an imagined IRS. The root node acts as an attack information database Finally, they enhanced dynamic intrusion detection hierarchy architecture where they argue forms a better basis for an IRS. The major limitation of this paper, it was not implemented to prove their theorized improvements. We implement such HIDS architecture with the help of Snort and MySQL databases.

Bent et al. [11] described a new type of database architecture which is defined as a dynamic distributed federated database (DDFD) using biologically inspired principles of network growth, combined with graph theoretic methods. This paper also explains the development and maintenance DDFD as well as utilization of DDFD to perform distributed semantic search over a network of search engines. Bent et al. used federated databases in distributed systems and demonstrated the use case for federated databases.

Pahlevanzadeh et al. [12] implemented an efficient distributed hierarchical IDS, using mobile agent over a MANET that has connection to the internet by AODV+ and study the effect of the proposed IDS in case of detection and CPU usage. Moreover, it was the first IDS framework designed for AODV+ routing protocols. They also demonstrated some improvements for ideal and accurate IDS based on mobile agent, clustering and distributed hierarchical IDS for wireless mobile ad hoc network.

III. OVERVIEW OF RELATED TECHNOLOGIES

This section providers an overview of various technologies needed to build the hierarchical system, such as the Snort NIDS (Network Intrusion Detection System) and other software.

A. Virtual Internetworking Environment

A virtual environment is the emulation of a computer system. With the help of Virtual Machines (VMs), we can emulate multiple machines and networks on the same physical computing hardware. We created multiple machines to emulate a network using virtual machines, known as virtual internetworking environment. In this paper, we used QEMU, which executes guest code directly on the host hardware, can emulate machines across hardware types with dynamic translation, and supports auto-resizing virtual disks.

B. Ubuntu Operating System

Ubuntu is an open-source Linux based operating system. For our implementation, we used version 16.04 Xenical Xerus. Ubuntu 16.04 is built on the 4.4 series of Linux kernels and the installed software needs to be compatible with this version of the operating system.

C. Snort NIDS

Snort is an open-source, free and lightweight NIDS software for Linux and Windows to detect emerging threats. Snort can detect intrusions on a computer network by reading pre-defined rules during startup and building internal data structures or chains to apply these rules to captured data [2]. Snort can be operated in three modes: sniffer mode, to view packages going through the network; logger mode, to record all packets on the network for later analysis; and intrusion detection mode, to detect the attacks over computer networks.

D. Barnyard Interpreter

Barnyard2 is an open source interpreter for Snort's "Unified2" binary output files. Barnyard2 reads the binary files from Snort, and then resends the data to a database backend. It is also aware when the database can accept connections again and will start sending the alerts again.

E. MySQL Database Server

Oracle's MySQL is a leading open source database management system. It is a multi-user, multithreaded database management system. MySQL database is available on most operating system platforms. Since Barnyard saves alerts to our MySQL database, we need to create appropriate database structure to complete the implementation.

IV. DESCRIPTION OF THE PROPOSED RESEARCH

This research proposal aims to implement a hierarchical distributed intrusion detection. This implementation work could be done by using a combination of Snort and MySQL databases.

A. Research Methodology

To achieve the research goal, and to explore and emphasize the idea of implementing a hierarchical distributed IDS, a virtual internetworking environment was firstly created.

One experimental internetwork topology was configured, shown in Figure 1. This experimental internetwork is constructed to understand the inter-networking scenario i.e. a hierarchical organizational structure.

In our topology, 19 nodes (machines) were used to construct our scenario, which has applications in Mobile Ad Hoc Networks (MANETs).

Next, Snort NIDS was installed on each node. This implementation also required a set up for an appropriate database tables, users, and passwords at each node because Snort was configured to store its logs into a database.

Consequently, in the next step, communication between Snort and MySQL was set up. Since Snort generated alerts in binary format, Barnyard was needed to read the alerts from Snort and then store them in the MySQL database tables. Hence, we installed Barnyard2 on all the nodes and it acted as an interpreter for Snort binary output files. At this point, the system can read and capture data from this basic implementation of a distributed intrusion detection system. Thus, this topology is organized into distributed multi-leveled hierarchies.

As the research goal is an implementation of hierarchical structure, transformation from distributed IDS to hierarchical IDS was carried out next.

To explore this research work, one more distributed intrusion detection system has been created as we described in above paragraph. Similarly, we created 4 distributed systems. Now, the primary objective of this research document is hierarchy structure, for this purpose we connected 2 distributed systems by using federated databases. Hence, on all the databases federated engines have been set up and we had aggregated data at the root node so successful implementation of HIDS is achieved. A detailed diagram of inter-networking topology is shown in Figure 1.

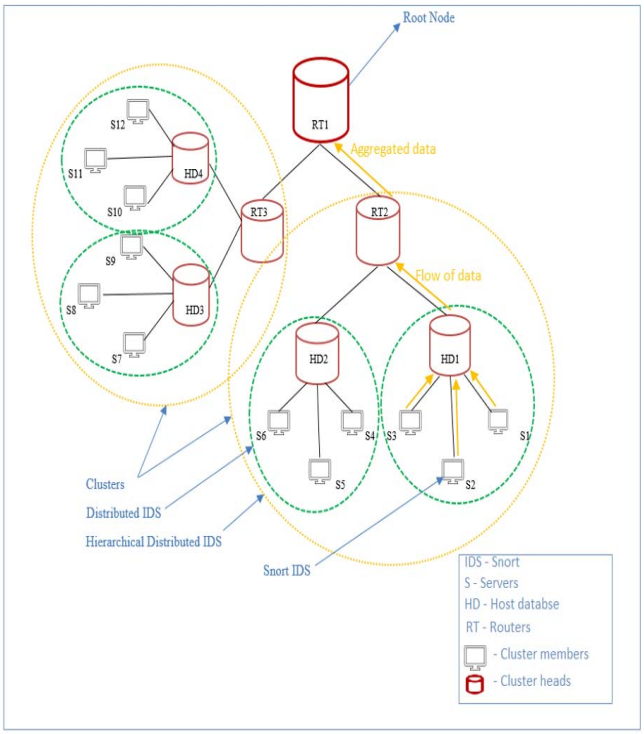

Figure 1. Hierarchical Distributed Intrusion Detection System

B. Establishing Virtual Environment

We constructed a hierarchical network topology consisting of 19 nodes from Server1 to Router1(S1-Rt1). Due to the hierarchical structure of the topology, the nodes are divided into clusters, in used topology we have 3 clusters as router1, router2 and router3(RT1, RT2, RT3) made clusters with their member nodes. All clusters consist of multiple distributed intrusion detection systems as host database1, host database2, host database3, host database4 mentioned as HD1, HD2, HD3, HD4 making distributed systems with their member nodes. Each cluster consists of a cluster head and cluster members. All the cluster nodes are interconnected with each other via cluster heads. Snort as IDS has been installed on all nodes server1-server12 given as S1-S12.

The node sitting at the top of the hierarchy is designated as 'root node' which is RT1 that also works as a server database. Initially, all distributed IDS create the first level of hierarchies by forming autonomous clusters. Afterwards the cluster-heads of the previously formed clusters are selected to participate in the next level of hierarchies.

C. *Configuring Servers* (*S1 – S12*)

Firstly, Snort was installed on all of the servers from S1 to S12 for the purpose of generating alerts. Nodes in the distributed IDS were configured identically. Snort was configured to run in NIDS mode. Initially, each level of distributed IDS consists of several clusters in which specific nodes act as Cluster heads gathering local audit data from its Cluster members as HD1 is cluster head and collected data from S1, S2, andS3. Similarly, all cluster-heads do their job. For testing Snort, some experimental rules were created, one example is given below.

```
alert icmp any any -> $HOME_NET any
(msg:"ICMP test detected"; GID:1;
sid:10000001;rev:001;classtype:icmpe
vent;)
```

Consequently, Snort was able to generate an alert when it saw an ICMP ping, and save a copy of this information in a log file. The results are shown in Figure 2.

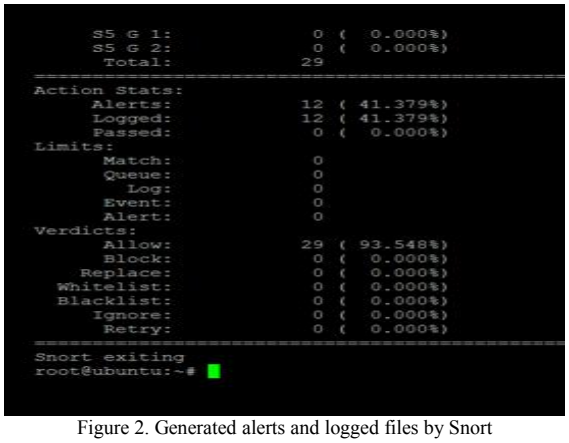

Figure 2. Generated alerts and logged files by Snort

D. *Configuring Databases* (*HD1 – HD4*)

HD1-HD4 are the host databases which will help to show the data generated by server machines. Firstly, Barnyard2 was set up on all hosts containing databases. Barnyard2 could then read the binary event log files generated by Snort asynchronously and insert them to our MySQL database. We checked the MySQL database to see if Barnyard2 properly inserted the events and alerts by using an ICMP request and reply packets via S1 pinging Hd1. As shown in Figure 2, the database had 742 events written to the database.

E. *Configuring routers* (*RT1 – RT3*)

RT1 is the root of our topology, which was configured as a remote database and RT2 and RT3 were set up as local servers, similarly as we did for HD1, HD2, HD3 and HD4. Next, to create a local cluster and simple hierarchy, 2 MySQL databases were connected to one single MySQL database as HD1 and HD2 is connected to RT2.

F. *Federated Storage Engine Setup*

Finally, a Federated Storage Engine (FSE) was set up to allow connecting to a remote server and synchronizing a table from a local server (example of federated databases is given in figure 3), [2] thereby linking to the data on the remote server RT1. To accomplish this goal, the local table was replicated on the remote server and all other machines set up with a database.

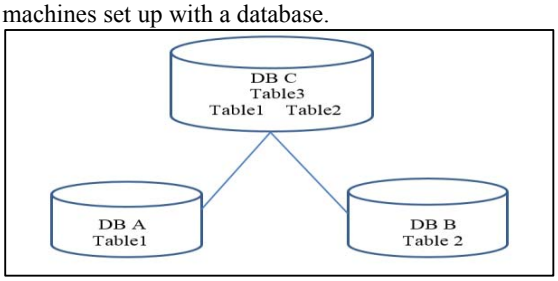

Figure 3: Federated Databases

RT1 has its own database called “snort” on remote server and RT2, RT3 has their own databases called “snort1” on local servers. On Rt2 and RT3 we enabled federated engines. On snort1 we created table called “data” by using same syntax as we had in database snort. Now, the table “data” carried the information which is generated by snort rules. After setting up MySQL connection and federated engines between remote and local server we had aggregated data on root node-RT1 carried from RT2 and RT3.

Similarly, all clusters received data from their member nodes. RT2 and RT3 making clusters with host databases so they are receiving information from their member nodes as RT1 did. Snort supports the MySQL database to get an output. Hence, the root node is at the top of the IDS hierarchy and receives aggregated/consolidated intrusion detection information from the entire network, and therefore research has completed the most important part.

V. RESEARCH OUTCOMES

In this research, a hierarchical IDS demonstrated and implemented by using Snort rules and MySQL federated databases. This research work shows the root node is at the top of the IDS hierarchy and receives aggregated/consolidated intrusion detection information from the entire network as shown in Figure 5, and therefore this research completed the most important part.

We can see the total number of detected data from the entire network is 742 collected from all databases, where federated databases are used to join all databases.

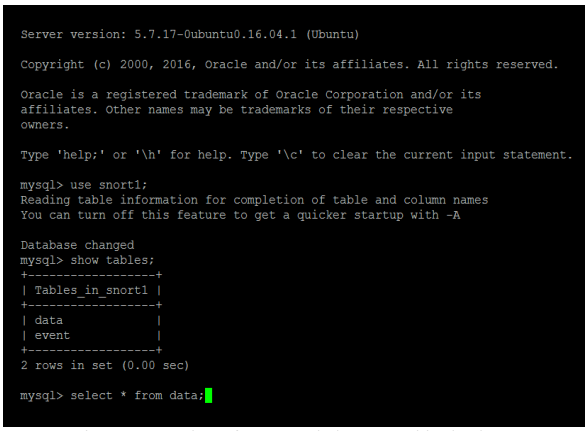

Figure 4. Number of generated alerts saved in database

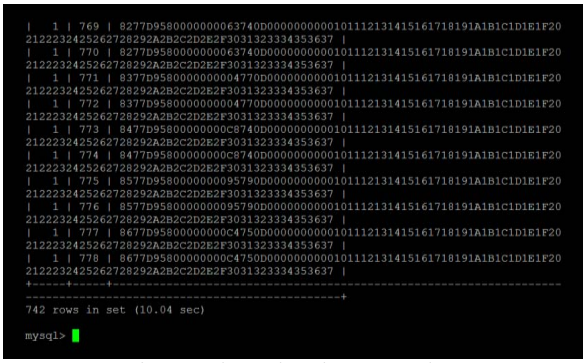

Figure 5. Observed results at root node

Moreover, regarding strengths of hierarchical IDS, we can deduce that nodes with the highest battery power are elected as to serve as cluster-heads. Secondly, multiple layers of detection could provide faster detection rates. Thus, we chose hierarchical IDS because in general, while studying distributive and corporative IDS, we found some major weaknesses. For instance, in the entire set of the studied architectures, the ratio of false positives and detection is negatively affected by high nodes' mobility. Also, the majority of them are vulnerable to attacks (i.e., man in the middle, blackmail, etc.).

But on the other side hierarchical IDS can monitor and capture live packet traffic on the network. By referring some network intrusion detection system, we found NIDS has less overhead and it presents better detection because it is broad in scope and it can detect the attacks from outside as well. So, this implemented distributed hierarchical IDS model consists of three layers, cluster member layer and cluster head layer.

VI. CONCLUSION AND FUTURE WORK

This research contributes to the understanding of intrusion detection system in virtual inter-networking environments and shows the implementation of a hierarchical distributed intrusion detection architecture that makes the use of Snort technology and MySQL federated databases to make transition from distributed IDS to hierarchical IDS.

As in this document, the implemented topology is static and pre-configured. This work can be extended by converting this static topology to dynamic topology as well. In this way, this implementation could be useful not only for static environment, but it will also work for MANET's topologies. Moreover, in this architecture, the set up for root node is done at the top of the IDS hierarchy and receives aggregated/consolidated intrusion detection information from the entire network. So, this implemented research idea can be used in organizations where they deal with large amount of data. In this implementation, intrusion detection occurs at the cluster head of each cluster by aggregating data from cluster members to have faster detection.

VII. REFERENCES

[1] S. Chadli, M. Emharraf, M. Saber and A. Ziyyat, "Combination of hierarchical and cooperative models," *In Proceedings of Tenth International Conference on Signal-Image Technology & Internet-Based Systems*, 2014.

[2] A. P. Sheth and J. A. Larson, "Federated Database Systems for Managing Distributed, Heterogeneous, and Autonomous Databases'," *ACM Computing Surveys,* vol. 22, no. 3, Sept. 1990.

[3] I. Butun, Riah and R. Shankar, "An Intrusion Detection System Based on Multi-Level Clustering for Hierarchical Wireless Sensor Networks", *Sensors*, vol. 15, no. 11, pp. 28960-28978, Nov. 2015.

[4] M. Hadi, M. K. Entin, A. Pratiarso and J. C. Ellysabeth, "Intrusion Detection System Based On Snort Using Hierarchical Clustering," *International Seminar on Scientific Issues and Trends* (ISSIT), 2011.

[5] Md. S. Islam, R. H. Khan and D. M. Bappy, "A Hierarchical Intrusion Detection System in Wireless Sensor Networks," *International Journal of Computer Science and Network Security* (IJCSNS), vol. 10, no. 8, pp. 21-26, Aug. 2010.

[6] S. R. Snapp and J. Brentano, "DIDS (Distributed Intrusion Detection System) − Motivation, Architecture and An Early Prototype," in *Computer Security Laboratory Division Of Computer Science*, California.

[7] S. S. Bedi and D. Singh, "A State of an Art Survey of Intrusion Detection System in Mobile Ad-hoc Network," *International Journal of Computer Applications (0975 – 8887),* vol. 82, p. 6, Nov. 2013.

[8] Y. Zhang, W. Lee and Y. A. Huang, "Intrusion Detection Techniques for Mobile Wireless Networks," *Wireless Networks*, vol. 9, no. 5, pp. 545-556, Aug. 2003.

[9] P. Albers, O. Camp, J-M. Percher and B. Jouga, "Security in Ad Hoc Networks:a General Intrusion Detection Architecture Enhancing Trust Based Approaches, " *In Proceedings of the First International Workshop on Wireless Information Systems*, 2002.

[10] M. Kaur, D. Lindskog and P. Zavarsky, "Integrating Intrusion Response Functionality into the MANET Specific Dynamic Intrusion Detection Hierarchy Architecture,". MSc Thesis. Concordia University of Edmonton, 2016.

[11] G. Bent and P. Dantressangle, "A Dynamic Distributed Federated Database," *IBM, Engineering Technology Services.*

[12] B. Pahlevanzadeh and A. Samsudin, "Distributed Hierarchical IDS for MANET over AODV+," in *Proceedings of the 2007 IEEE International Confbrence on Telecommunications and Malaysia*, Penang, Malaysia, May 2007.