

Attack Graph Generation and Threat Evaluation in Network Situation Awareness (NSA)

Patil Shalaka, Ahir Minakshi, Kale Dattatraya

Abstract -A Network is a collection of many devices, where each node is said to be wired or wireless connection between them. And now a day's most of the threat comes to the network either from outside or from a sort of situation which occurs internally due to many reasons. So the Intrusions or threat which occurs due to these situations are generally more damageable than the normal ones. This paper is introducing a technique to analyze various types of alerts & also generating attack graph for such alerts by using two algorithm i.e. correlation of isolated alerts to alert-pair, attack graph generation. And after analyzing the threat we are also performing evaluation technique to determine the seriousness of the threat and remove it. In this paper our vital focus is on alert analysis. In the current intrusion detection systems, it produces large volumes of alerts. These overwhelming alerts make it challenging to recognize and manage them. Therefore, we have to condense the amount of the alerts and external useful information from them. However, the NSA requires the alert analysis techniques to suggest high-level information such as how serious of attacks are and how dangerous of devices are and which attacks or devices require administrator to pay attention to. To deal with this problem we put forward a time and space based alert analysis technique which can correlate related alerts without background knowledge and offer attack graph to help the administrator understand the attack steps visibly and efficiently. And a threat evaluation is given to find the most hazardous attack, which further saves administrator's time and energy in handing out large amount alerts.

Index Terms— DARPA IDS Evaluation Dataset, Intrusion Detection, NIDS, Snort.

I. INTRODUCTION

With the ever rising use of the Internet by all types of companies, security threats such as attacks on enterprise infrastructure by hackers, the leakage of personal data and the infection of secret business information caused by e-mail based viruses, have become major issues in the security literature over the last few decades. Security systems such as IDS (Intrusion Detection Systems) and Firewalls have been developed to detect and protect these systems in wired and wireless networks.

However, along with the changes that have occurred in the patterns of attack as well as the increasing use of variant methods,

Manuscript Received on February 2013.

Patil Shalaka, B.E. student of Vishwabharti Academy's College of Engg., Department of Computer Engg Ahmednagar.State-Maharashtra, India.

Ahir Minakshi, B.E.student of Vishwabharti Academy's College of Engg., Department of Computer Engg Ahmednagar.State-Maharashtra, India.

Kale Dattatraya , B.E. student of Vishwabharti Academy's College of Engg., Department of Computer Engg Ahmednagar. State-Maharashtra, India.

attacks are becoming more common using diverse and mixed techniques, rather than being limited to a single attack technique, or making use of multiple exploits. For example, the Nimda virus, which first appeared in 2001, is a mass-mailing worm that uses many methods to spread itself.

This worm sends itself out by email, searches for open network shares, attempts to copy itself to unpatched or previously vulnerable Microsoft IIS web servers, and is a virus that infects both local files and files on remote network shares. Unfortunately, there are presently no information protections systems that can recognize these mixed attacks, such as those involving the Nimda and Agobot worm, and sound an alert. present information protection systems only detect and warn against individual intrusions, and are unable to provide a collective and synthesized alert message. Therefore, it is complicated to detect and react effectively against variant exploits. It not only requires a great deal of time to analyze and determine the practical vulnerabilities from the huge amounts of composed data in order to detect the potential intrusions and protect the system, but it also requires a great deal of manpower, who are trained in such security issues. In this paper, we essentially focus on alert analysis. It is well-known that current Intrusion Detection Systems generate large volumes of alerts. These overwhelming alerts make it challenging to understand and handle them. Therefore, we have to decrease the amount of the alerts and extract useful information from them. Researchers projected many approaches to analysis alerts. But most of them are designed to reduce false positives and false negatives DSs which is not the purpose of NSA system. NSA system aims to get awareness of the network, it has to offer intuitionistic information to administrators, such as how serious of an attack is or how dangerous of a device is, rather than directly suggest alerts to administrators. Actually, it is tough for administrators to conclude how serious of the attack is via checking the alerts manually. We propose our approach to automatically correlate the alerts to generate simple attack graphs based on time and space restriction. In addition, we give an attack evaluation technique. We, first propose our alert analysis method to correlate related alerts and suggest simple attack graph. Then, we give an evaluation function for possible threats. Via these proposed methods, administrators can understand the network situation and find out how serious of an attack without checking individual alerts or evaluation values. Here NSA just wants to know where, when and how serious of an attack is, so we simply need a small subset of alert fields. Using short alert message also saves time and storage space.

II. LITERATURE SURVEY

Numerous researchers have proposed several modules which are used to analyze the attacks (alerts). Let's see some workings of these Review Stage: Endsley defined SA as "the perception of the elements in the environment within a volume of time and space; the comprehension of their meaning and the project of their condition in the near future". Episodic analysis, which can span protocols and sessions, provides a higher-level understanding of a network's structure along with behavior. This capability can be used to identify network events that cannot be perceived within packet analysis or session analysis only. For example, it is nearly impossible to detect the rebooting of a Windows machine in a single packet or session, but it can be seen through the recognition of a chain of specific packets (e.g. three gratuitous ARPs). Event recognition lends a temporal context to the network data that can be extremely useful to an administrator wishing to recognize a network beyond simply how it's put together, and into how it behaves temporally. Episodic analysis also supports larger, time-domain descriptions of common network characteristics such as packet flows, amount of traffic sent, etc. This analysis permits characterization of server machines by connection rates as well as network links by traffic rates.

IIISYSTEM MODEL

The aim of this project is to develop a network security situation awareness system which will combine and analyze security alert events together from security situation sensors and generate the network security situation by extracting the frequent patterns and sequential patterns from the dataset of network security situation based upon Knowledge Discovery method and transform these patterns to the correlation rules of network security situation and finally to automatically generate the network security situation graph. We offer our approach to automatically correlate the alerts to generate simple attack graphs based on time and space restriction. In addition, we provide an attack evaluation method. We propose our alert analysis method to correlate related alerts and offer simple attack graph. Then, we offer an evaluation function for possible threats (either from

attacks or on devices). Via these proposed methods, administrators can understand the network situation and learn how serious of an attack without checking individual alerts. Here NSA just wants to know where, when and how serious of an attack is, so we only require a small subset of alert fields. Using short alert message, it also saves time and storage space. First, it is important to understand the definition of alert.

Alert: An alert *a* is a seven tuple (aid; srcip; dstip; srcport; dstport; type; time): **aid** is an AUTO INCREMENT integer generated by Database. It is used to identify each alert.

srcip represents the source IP address. The operation Srcip(*a*) Means get the source IP address of the alert *a*. **dstip** represents the destination IP address, the corresponding operation is Dstip(*a*).

srcport represents the source port, the corresponding operation is Srcport(*a*)

dstport represents the destination port, the corresponding operation is Dstport(*a*).

Type represents the alert's type; it is a short string which gives a simple description of the attack, the corresponding operation is Type(*a*).

time represents time of the alert generate, the corresponding operation is Time(*a*).

Algorithm 1 shows the method of correlating two isolated alerts to an alert-pair.

Algorithm 1: Correlation of Isolated Alerts to AlertPair:

Input: individual hyper-alerts a_1, a_2, \dots, a_n

Output: set of alert-pairs (ai,aj).denoted APs. Let TW be the time-window which is set by administrator

Let HyperAlert be the hyper-alert table.

Let AlertPairs be the alert-pairs table.

1. for all the hyper-alerts in HyperAlert do
2. if Srcip(ai) = Srcip(aj) and Dstip(ai) = Dstip(aj) and Time(ai) < Time(aj) and Time(aj) - Time(ai) < TW Then 3.
- put (ai,aj) into Alert-Pairs. 4. if Dstip(ai) = Srcip(aj) and Time(ai) < Time(aj) and Time(aj) - Time(ai) < TW Then 5.
- Put (ai,aj) into Alert-Pairs.

Then, we correlate these alert-pairs to an attack graph as Algorithm 2

Algorithm 2 : Attack Graph Generation:

Input: set of alert-pair (ai,aj) - APs.

Output: attack graph G(N,E) Put every hyper-alert ai of APs into node set N; Put every alert-pair (ai,aj) of APs into edge set E;

1. for every edge(ni,nj) do
2. if there is a indirect path ni,.....,nk,....,nj Then 3. Remove the edge (ni, nj) from edge set E 4. Return G(N,E) System Architecture of the project is shown in the below picture.

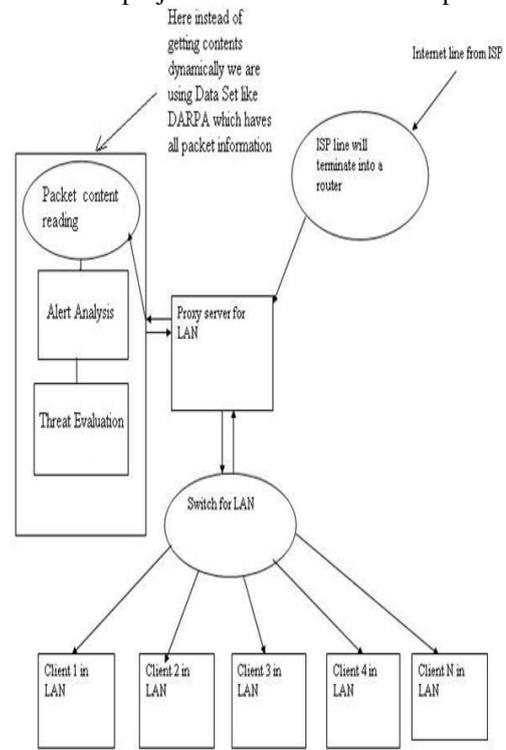


Fig.1.Network diagram

IV. PROPOSED WORK

Here we use the following modules for implementing our system:

1. Reading Packet dataset

Step 0: Start

Step 1: Get the Excel Dataset

Step 2: Read all the rows and column and put in array. Step 3:

Then allocate all individual packet information in a queue.

Step 4: Stop

2. Creating Alert Object

Step 0: Start

Step 1: Get the data packet queue

Step 2: Read srcip, dstip, srcport, dstport, type, time from the queue for each packet information. Step 3: Create a alert object

Step 4: Stop

3. Time and Space Restriction Analysis (TSRA)

Step 0: Start

Step 1: Get all alerts

Step 2: For all alert which are critical and hyper check the condition if(sip(ai)=sip(aj) and dip(ai)=dip(aj) and time(aj)<time(ai) and time(aj)-time(ai)<TW) if yes then goto step 6 else goto step 2.

Step 5: Put (ai,aj) in alert pair.

Step 6: Stop

4. Attack Graph Generation

Step 0: Start

Step 1: Get the set of alert pair.

Step 2: Create a node set N which consists of ai of each alert pair.

Step 3: Create an edge set E which consists of (ai,aj) of all alert pair.

Step 4: For each edge(ni,nj) check the condition if a indirect path ni,nk,nj then delete (ni,nj) from edge set E and return graph G(N,E).

Step 5: Stop

5. Creation of Alert Device Evaluation Matrix

Step 0: Start

Step 1: Get Number of devices and alerts.

Step 2: For each alerts in rows create a set E such that E

(ai,dj).

Step 3: Store it in a hash set matrix.

Step 4: Stop.

6. Calculation of UNIT THREAT EVALUATION (UTE)

Step 0: Start

Step 1: Get a particular device and its alert set E from the matrix of module 5.

Step 2: Get alert level l(a) and device level l(d).

Step 3: Calculate $Ead = 10l(a) - 1 * 10l(d) - 1$. Where a refers to the alert generated by IDS for corresponding attack; d represents the device which is attacked; l(a) and l(d) represent the levels of the alert and the device.

Step 4: Stop

7. Calculation of ATTACK THREAT EVALUATION (ATE)

Step 1: Start

Step 2: Get all the UTE of all the devices.

Step 3: For each UTE Calculate, $Ea = Ea + Eadi$.

Step 4: Stop

8. Calculation of DEVICE THREAT EVALUATION (DTE)

Step 0: Start

Step 1: Get all the UTE of all the devices.

Step 2 : For each UTE calculate, $Ed = Ed + Eajd$

Step 3 : Stop.

9. Calculation of NETWORK THREAT EVALUATION

Step 0: Start

Step 1: Get all the UTE of all the devices.

Step 2: For each UTE calculate, $EN = EN + Eaj OR$

$EN = EN + Edi$

Step 3: Stop.

V. SYSTEM DESIGN FRAMEWORK OF NSSA

The framework for network security situation awareness is based on knowledge discovery and consists of two parts, the modeling of network security situation and generation of network security situation. The modeling of network security situation is to build the formal model adapted for time(ai)<time(aj) and time(aj)-time(ai)<TW) if Yes

then goto step 3 else goto step 2.

Step 3: Put (ai,aj) in alert pair.

Step 4: Check the condition if(dip(ai)=sip(aj) and theme assuring of network security situation based upon the D-S Evidence Theory, and support the general process of the fusion plus correlation analysis of different types of alert events from security situation sensors. The generation of network security situation mainly consists of following steps:

A. The modeling of network security situation

The primary objective of the modeling of network security situation is to create the standardized data model suited for the measuring of network security situation, and support the general process of the simplification, filtering and fusion of alert events from security situation sensors. The data sources used for the modeling of network security situation are different types of security alert events collected from heterogeneous situation sensors distributed in the supervised network. The method of the modeling of network security situation is composed of several phases. During the initial phase of preprocessing, all the received security events are changed to the standard format that can be understood by data process module through the specification of the alert events. The alert events might be from different sensors, and have distinct formats, such as the events of IDS, the records of firewall, the log file of host system, and the information from net flow, etc. The intention of specification is to transform all the event attributes of each sensor to a uniform format. In our framework, we offer different preprocessing modules for the corresponding sensors, and transform the information from specific sensor to the attribute values of the information model defined in this. Based on the information model, each primitive event is preprocessed and transformed to the standard format, and each attribute field is set to the appropriate value.

B. Generation of network security situation

There are two network security situation data sources on hand for knowledge discovery: one is the set of security alert events generated from attack simulations; the other is the set of historical security alert events. The purpose of knowledge discovery in our framework is to find out and extract the knowledge from these set of alert events, which is necessary for the correlation of security situation.

Due to the complexities of alert events generated from various types of security situation sensors, the process is hardly to be performed fully by manual work. In this paper, we propose a knowledge discovery based technique, which provides the means of extracting the security situation correlation rules through the pattern mining, analysis and learning from the set of security alert events, and finally generate the network security situation graph. This process is separated into the following steps:-

Simplification and Filtering of Security Alert Events:

We found that there exists large numbers of meaningless frequent patterns in the set of primitive alert events from security situation sensors by investigative the experiment data, and these frequent patterns mostly relate to the problems of system configuration or harmless access. If the process of knowledge discovery is directly performed on such set of primitive intrusion events, it is inevitable to generate various types of meaningless knowledge. Therefore, it is necessary to establish the method of alert event filtering in the foundation of D-S evidence theory, which executes the statistical analysis based on the confidence level of alert events. Firstly, the distributions of various types of security events are statistically analyzed via automatic tools; secondly, the meaningless events are deleted by evaluating the importance of all type of alert events based upon the rules of simplification and filtering, which uses D-S evidence theory as the foundation of event processing. **Knowledge Discovery from the Set of Security Alert Events:** The frequent pattern in addition to sequential pattern discovery algorithm are adopted to get the security situation knowledge from the set of security alert events.

VI. TECHNICAL SPECIFICATION

A. Advantages

-It reduces the false alarm rates generated by the Intrusion Detection Systems.

-It offers intuitionistic information to administrators, such as how serious of an attack is rather than directly offer alerts to the administrators.

-It provides dynamic generation of Network Security Situation Graph.

-Network Security Situation Graph is updated in accordance of system settings and notifies to the administrator.

B. Disadvantage

-Protecting the Intrusion Detection Sensors from attackers is a challenge; if the IDS sensors are attacked it will bring down the whole system.

C. Applications

-NSSA system can be deployed in to the networks to provide situation awareness.

-It can be integrated with different network security equipments and converts alerts generated from them into high level graphic plus provide evaluation to attacks and devices.

VII. CONCLUSION

In this paper, we developed techniques to extract useful information of network situation from alerts, while most of existing NSA systems just focuses on net flow data. We proposed an approach to automatically correlate the alerts to

generate a simple attack graph based upon time and space restriction. The graph helps to administrator to understand the attack steps easily. This approach can discover new alert relations and it does not depend on background knowledge. At last, we tested our method on DARPA 2000 Dataset. The simulations showed that with the proposed methods NSA system can efficiently analyze large amount alerts and save administrators' time and energy also.

ACKNOWLEDGMENT

My sincere thanks go to Vishwabharti Academy's College of Engineering for providing me a strong platform to develop my skill and capabilities. I would like to thanks to my friends & family for their constant support and motivation for me. We would like to thank Prof. D.M.Deshpande for there support. I am extremely grateful to F. A. Author for giving me an opportunity for presenting this paper. Last but not least, I would like to thank all those who directly or indirectly assist me in presenting the paper.

REFERENCES

1. Fang Lan, Wang Chunlei, and MaGuoqing , "A Framework for Network Security Situation Awareness Based on Knowledge Discovery" 2010 2nd International Conference on Computer Engineering and Technology 2010 IEEE.
2. Juan Wang,Feng-li Zhang,Jing Jin,Wei Chen, "Alert Analysis and Threat Evaluation in Network Situation Awareness" 2010 IEEE.
3. Cyril Onwubiko , "Functional Requirements of Situational Awareness in Computer Network Security" 2009 IEEE.
4. Liu Mixi, Yu Dongmei and Zhang Qiuyu et al., "Network Security Situation Assessment Based on Data Fusion, " 2008 Workshop on Knowledge Discovery and Data Mining, 2008.
5. Wang Huiqiang, Lai Jibao, and Ying Liang, "Network Security Situation Awareness Based on Heterogeneous Multi-Sensor Data Fusion and Neural Network, " Second International Multisymposium on Computer and Computational Sciences, 2007 IEEE.
6. Mr. Marc Grégoire , "Visualisation for Network Situational Awareness in Computer Network Defence" (2005). In Visualisation and the Common Operational Picture (pp. 20-1 - 20-6). Meeting Proceedings RTO MP-IST-043, Paper 20. Neuilly-sur-Seine, France: RTO. Available from: <http://www.rto.nato.int/abstracts.asp>.
7. Yu Dong and Frincke, D., "Alert Confidence Fusion in Intrusion Detection Systems with Extended Dempster- Shafer Theory, " 43rd ACM Southeast Conference, March 18-20, 2005.
8. J Hall, J Pei, Y Yin. Mining frequent patterns without candidate generation. 2000 ACM. SIGMOD int'ICConf on Management of Data (SIGMOD'00), DallaS, TX, 2000.
9. Bass, T., "Intrusion Detection Systems and Multisensor Data Fusion, Communications of the ACM, Vol. 43, No. 4, April 2000.
10. Jia Han, Micheline Kamber., "Data Mining concepts and techniques", second edition 2006, Elsevier Inc.

AUTHOR BIOGRAPHY

Patil Shalaka B.E. Computer, student in VACOE Ahmednagar. State-Maharashtra, India.

Ahir Minakshi B.E. Computer, student in VACOE Ahmednagar. State-Maharashtra, India.

Kale Dattatraya B.E. Computer, student in VACOE Ahmednagar.State-Maharashtra,India