International Journal of Electrical and Computer Engineering (IJECE)
Vol. 3, No. 4, August 2013, pp. 461~466
ISSN: 2088-8708

Alert Correlation through a Multi Components Architecture

Saeid Dadkhah, M. R. Khalili Shoja, Hassan Taheri
Department of Electrical Engineering, Amirkabir University of Technology

Article Info

Article history:

Received Mar 5, 2013
Revised May 27, 2013
Accepted Jul 1, 2013

Keyword:

Intrusion detection system
Alert correlation
Alert reduction
Attack scenario

ABSTRACT

Alert correlation is a process that analyzes the raw alerts produced by one or more intrusion detection systems, reduces nonrelevant ones, groups together alerts based on similarity and causality relationships between them and finally makes aconcise and meaningful view of occurring or attempted intrusions. Unfortunately, most correlation approaches use just a few components that aim only specific correlation issues and so cause reduction in correlation rate. This paper uses a general correlation model that has already been presented in [9] and is consisted of a comprehensive set of components. Then some changes are applied in the component that is related to multi-step attack scenario to detect them better and so to improve semantic level of alerts. The results of experiments with DARPA 2000 data set obviously show the effectiveness of the proposed approach.

Corresponding Author:

Saeid Dadkhah,
Department of Electrical Engineering, Amirkabir University of Technology, Tehran, Iran
Email: saeiddadkhah@aut.ac.ir

1. INTRODUCTION

With the burst of the public Internet and e-commerce, private computers and computer networks, if not safe enough, are increasingly defenseless to damaging attacks. Hackers, viruses, malicious employees and even human error all represent clear and present dangers to networks. And all computer users, from the most casual Internet surfers to large enterprises, could be affected by network security violations. However, security violations can often be prevented. Some of the steps organizations can take to protect networks from threats and ensure that the data traveling across the networks is safe, are: anti-virus packages, security policies, access control, encryption, network scanning, firewalls and intrusion detection systems (IDSs).

Organizations continue to use firewalls as their central gateway to prevent unauthorized users from entering their networks. However, network security is in many ways like physical security in that no one technology provides all needs—rather, a layered defense offers the best results. Organizations are increasingly paying attention to additional security technologies to face risk and vulnerability that firewalls alone cannot address. An IDS analyzes packet data flows within a computer or network, searching for unauthorized activity, such as attacks by hackers, and enabling users to reply to security violations before systems are compromised. When unauthorized activity is detected, the IDS can send alerts to a management console with details of the activity and can often ask other systems, such as routers, to limit the unauthorized connections.

Still current IDSs techniques are not good enough, as they suffer from some limitations [5], [6]:

- IDSs generate a huge number of alerts [1].
- Analyzing thousands of alerts each day is impossible, especially if most of them are false positives (events wrongly classified as attacks) [8].
- IDSs may not detect some attacks [1].

To conquer these limitations and increase the effectiveness of IDSs, alert correlation has been recommended. Correlation analyzes the raw alerts, reduces nonrelevant ones and groups together alerts based

on similarity and causality relationships between them. Currently, there are five techniques for alert correlation [1], [2]:

- Similarity based approaches,
- Predefined attack scenarios based approaches,
- Multi-stage approaches,
- Multiple information sources,
- Filter based approaches.

This paper uses the correlation model that has been presented in [9]. Then a different technique is used in the *multistep* component to improve the efficiency of this component. As you can see in Figure1 this model is a collection of components.

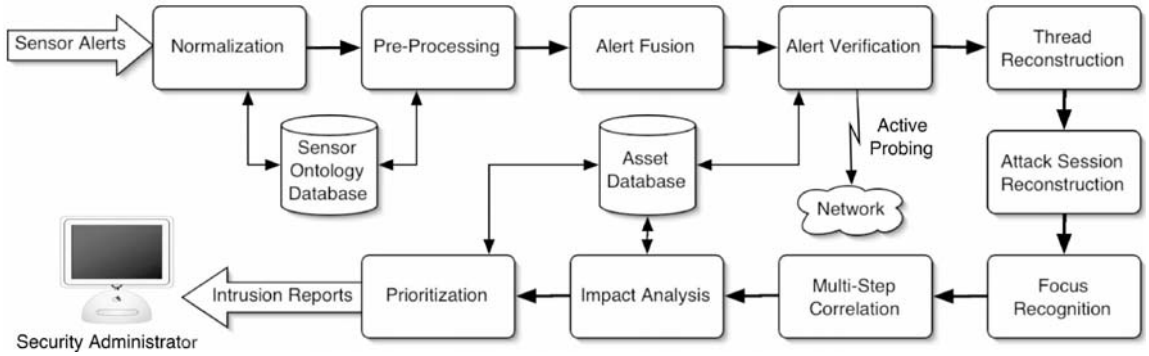

Figure 1. Graphical representation of alert correlation process [9].

The first two tasks in this model make alerts ready for the correlation process. The *Normalization* component, normalize received alert from different IDSs into a standardized format that is understood by other components. In the correlation process some alert attributes are important and must not be null. So *Preprocessing* component adds meaningful values to all required alert attributes (such as timestamp, source and target of the attack).

The next four components correlate duplicate and close alerts and tag nonrelevant ones. The task of *Fusion* component is merging alerts which denote the independent detection of the same attack instance by various intrusion detection systems. The *Verification* is helpful for identifying false positive alerts. It takes each alert and checks the result of the attack that corresponds to this alert. If the attack is not successful, it properly tags the alert to exclude it from correlation process. In the *Thread Reconstruction* component, a series of alerts that belong to attacks launched by a single attacker against a single target is identified and merged together. The *Attack Session Reconstruction* component links network-based alerts with host-based alerts that refer to the same attack.

The *Focus Recognition* component identifies the hosts that are either the source or the target of a significant number of attacks. This is useful for identifying Denial of Service (DoS) and port scanning attacks. The task of *Multistep* component is identifying attack scenarios. These attacks are composed of a series of single attacks that can happen sequentially in the network.

The *Impact Analysis* component checks the effect of an attack on the appropriate function of the network. Finally the *Prioritization* component gives a suitable priority to each alert, based on security policies.

The rest of this paper is organized as follows. In the next section alert correlation techniques are reviewed. In Section 3, our proposed alert correlation approach and results of our experiments are discussed. Finally, conclusion is presented in Section 4.

2. ALERT CORRELATION TECHNIQUES

To overcome IDSs limitation that was mentioned previously, several alert correlation techniques have been proposed [5], [6]:

2.1. Approaches Based on Similarity between Alert Attributes

These approaches [26], [16], [13], [21] correlate alerts based on the similarity between alert attributes. Each alert is a set of attributes such as source and destination IP address, source and destination port number, start-time and end-time. A function measures the similarity between alerts and the output of this function decides if alerts will be correlated. All similarity based approaches are effective for clustering and merging similar alerts into groups because similar alerts in a group may correspond to the same attack [5], [6]. However, most of them are not able to making high-level alerts.

2.2. Approaches Based on Predefined Attack Scenarios

These approaches [17], [27] correlate alerts based on predefined attack scenarios. These attack scenarios can either be specified by the users or learned from training data sets. Most alert correlation approaches in this category are effective in detecting some known attacks. However, they fail to detect new attacks. Furthermore, an explicit attack scenario database can be difficult to build [4].

2.3. Approaches Based on Prerequisites and Consequences of Attacks

These approaches [10], [11], [19], [20] (also called Multi-stage) correlate alerts based on relationship of earlier and later alerts. In these approaches, each attack is modeled by its prerequisites (the condition of a successful attack) and consequences (the result of a successful attack).They build attack scenarios through matching the consequences of earlier attacks with the prerequisites of later attacks [18], [22]. First order logic or some attack modeling languages (such as LAMBDA [20]) is used to model attacks. These approaches have the ability to detect new attack. However, building library of attack steps is expensive and time-consuming as there are a large number of attack types [4].

2.4. Approaches Based on Multiple Information Sources

These approaches [12], [23], [24] correlate alerts in a network with several complementary security systems e.g. anti-virus packages, security policies, access control, encryption, network scanning, firewalls and IDSs. Generally various systems have different ability. So a layered defense offers the best results.

Better protection with several heterogeneous security system also has some troubles. As we mentioned previously, an IDS can generate lots of alerts per day so several security systems can complicate this situation, and security manager will be overwhelmed in floods of alerts. In addition, security systems make alerts in different format. So alert correlation between several security systems is very challenging [5], [6].

2.5. Approaches Based on Filtering Algorithms

Filter based approaches have been presented to remove the need for a complex attack scenario database and to reduce nonrelevant alerts. Porras et al. [24] proposed a mission-impact-based multistep approach where a filtering algorithm is used in the alert processing steps.

The filtering algorithms are system specific. Therefore, they are expensive to use in comparison to the general method. Also detection accuracy of alert correlation depends on elaborate description of patterns in the filtering algorithm. So, there is a trade-off between the expressiveness of the filtering algorithm and the equivalent computational complexity.

3. PROPOSED APPROACHAND RESULTS

Most correlation approaches use just a few components that aim only specific correlation issues and so cause reduction in correlation rate. This paper uses correlation model in [9]. This model uses STATL [25] language to define attack scenarios in the *Multistep* component. As we mentioned earlier, predefined attack scenario based approaches fail to detect new attacks, but multi-stage approaches does not have this defect. So we use the approach that has been presented in [20] (LAMBDA language) to model attacks based on prerequisites and consequences.

In LAMBDA an attack is specified with five fields:

- Attack Precondition: logical conditions to be satisfied for a successful attack.
- Attack Postcondition: logical conditions that specify the result of the successful attack.
- Attack scenario: the scenario of the attack that intruder performs. This field gives us the benefit of predefined attack scenarios based approaches.
- Detection scenario: the combination of events that correspond to this attack.

– Verification scenario: a combination of actions to be done to check if the attack succeeds. As we use correlation model in [9], and there is a *Verification* component that does this task, we do not need this field.

After specifying an attack base in LAMBDA, the offline correlation process analyzes these attack descriptions to automatically generate a set of correlation rules. Correlation rules can be devided into two types: direct correlation and indirect correlation. The online correlation process then applies to these correlation rules on the alerts generated by the IDSs to detect more multistep attack scenarios [20].

To evaluate the efficiency of our approach in building attack scenarios we performed an experiment using the "LLDOS 1.0 Inside" from DARPA 2000 data set [28].

The five phases of the multistep attack scenario in LLDOS 1.0 are [28]:

– IPsweep from a remote site to determine which hosts are live.
– Probe of live IP's to determine which of the hosts selected in previous phase are running the Sadmind service.
– Breakins via performing the sadmind Remote-to-Root exploit several times.
– Installation of a DDoS software in the three compromised hosts.
– Launching the DDoS attack against the final victim.

DARPA 2000 data set is the raw event streams of the network and does not contain any IDS alerts. So we need to replay the related packets on a network interface card and set the IDS to generate alerts. We used Colasoft Packet Player software for repalying packets, and Snort [7], [14], [15] as an IDS.

Figure2 shows the correlation graph of this multistep attack that has been detected by our method in the *Multipstep* component. However, the method that has been used in [9] cannot detect this attack scenario completely.

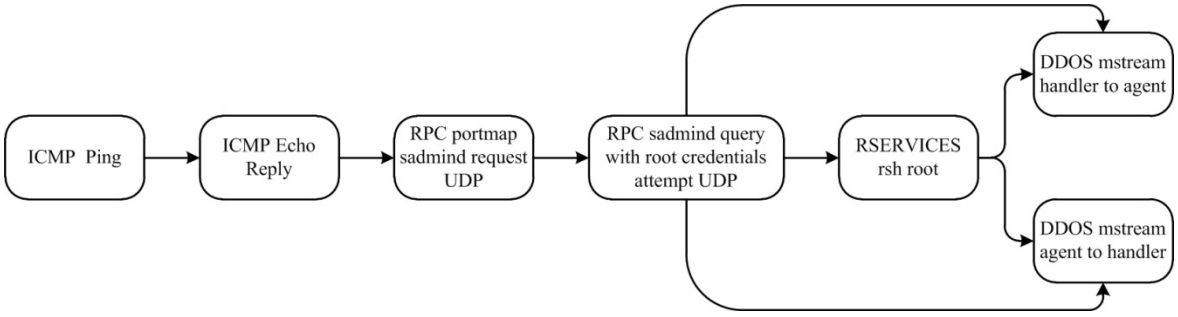

Figure 2. Correlation graph of the multistep attack.

Snort generated "*ICMP Ping*" and "*ICMP Echo Reply*" for the first phase of the attack. "*RPC portmapsadmind request UDP*" is generated when an attempt is made through a portmap GETPORT request to discover the port where the Remote Procedure Call (RPC) sadmind is listening and belongs to the second phase."*RPC sadmind query with root credentials attempt UDP*" is generated in phase three when an attempt is made to exploit a known vulnerability associated with the Remote Procedure Call (RPC) sadmind. "*RSERVICES rsh root*" means attacker may have gained superuser access to the host and belongs to phase four. "*DDOS mstream agent to handler*" and "*DDOS mstream handler to agent*" indicate a host may have been compromised and mstream may have been installed and belong to phase four and five. Snort did not detect the DDOS attack against the final victim.

Snort generated 720 alerts for this data set. After correlating alerts, the result obtained for the data set is as follows:

– Normalization: this component passes all alerts through; therefore, it does not reduce the number of the alerts received as input.
– Preprocessing: based on the task of this component, it does not reduce the number of alerts.
– Alert Fusion: the analyzed data set was produced by a single sensor; therefore, no fusion was possible.
– Alert Verification: the alert verification process needs the protected resources and a detailed model of the installed network services to be available for real-time verification. Unfortunately, this information is not available for data set that we analyzed.

- Thread Reconstruction: this component can significantly reduce alert when a very large number of alerts have been generated as the result of a single attack. Thread Reconstruction reduced alerts from 720 to 255, so reduction rate is 64.58 percent.
- Attack Session Reconstruction: this component requires either real-time access to the systems being protected or very detailed auditing information in order to map network-based alerts to host-based alerts. For the analyzed data set, this information was not provided.
- Focus Recognition: this component is effective in reducing alerts of our data set. The reduction is especially high when DDoS or large-scale scanning attempts exist in the data set. Focus Recognition reduced alert from 255 to154, so reduction rate is 39.60 percent.
  Multistep Correlation: while Multistep attack component may not reduce alerts as much as other components, it often provides a high improvement in the abstraction level of the generated meta-alerts. This component reduced alert from 154 to 141, thus reduction rate is 8.44 percent.
- Impact Analyzer: impact analysis needs a exact modeling of the relationships between resources in a network and needs the continuous monitoring of the health of those resources. So, DARPA 2000 data set cannot be used to evaluate the effectiveness of this component.
- Prioritization: alert prioritizing needs information about the network and nature of the attack to identify if one alert or meta-alert should be considered more important than another. So, we need an exact description of how to prioritize alerts which is not available for the data set.

Table 1 shows the impact of proposed alert correlation method, briefly.

Table 1. Impact of proposed alert correlation method

	Input Alerts	Output Alerts	Reduction
1.Normalization	720	720	0.00%
2.Preprocessing	720	720	0.00%
3.Alert Fusion	720	720	0.00%
4.Alert Verification	720	720	0.00%
5.Thread Reconstruction	720	255	64.58%
6.Attack Session Reconstruction	255	255	0.00%
7.Focus Recognition	255	154	39.60%
8.Multistep Correlation	154	141	8.44%
9.Impact Analyzer	141	141	0.00%
10.Prioritization	141	141	0.00%
Total	720	141	80.41%

4. CONCLUSION

In this paper, we discussed the importance of alert correlation process in reducing alerts and making a high-level state report of the protected network. Then alert correlation techniques have been reviewed. Some techniques like predefined known scenario approach can detect known attacks. Some like prerequisite and consequence approach can detect new attacks but needs a library of single attacks. For alert correlation, we used the model that has been proposed in [9] and tried to improve multistep attack component. Actually we used an approach based on prerequisite and consequence. As the results show, this change led to detect multistep attack scenario completely and so has a better correlation rate. Our future work will focus on other architectures and making a more complete library of attacks to detect more multistep attack scenarios.

REFERENCES

[1] HT Elshoush and IM Osman. "Alert correlation in collaborative intelligent intrusion detection systems – A survey". *Applied Soft Computing Journal*. 2011, 4349–4365.
[2] SH Ahmadinejad, S Jalili, M Abadi. "A hybrid model for correlating alerts of known and unknown attack scenarios and updating attack graphs". *Computer Networks*. Volume 55, Issue 9, 23 June 2011, 2221–2240.
[3] SX Wu, W Banzhaf. "The use of computational intelligence in intrusion detection systems: a review". *Applied Soft Computing Journal*. 2010; 10: 1–35.
[4] CV Zhou, C Leckie, S Karunasekera. "A survey of coordinated attacks and collaborative intrusion detection". *Computers and Security*. 2010, pp 124-140.
[5] RD Pietro, LV Mancini. "Intrusion detection systems". in: S. Jajodia (Series editor), Handbook of Advances in Information Security, Springer, 2008, ISBN 978-0-387-77265-3, e-ISBN: 978-0-387-77266-0.

[6] D Xu, P Ning. "Correlation analysis of intrusion alerts". in: R. Di Pietro, L.V. Mancini (Eds.), Intrusion Detection Systems, Advances in Information Security. vol. 38, Springer, 2008, pp. 65–92, ISBN 978-0-387-77265-3.
[7] Brian Caswell, Jay Beale, Andrew R Baker. "Snort IDS and IPS Toolkit". Open Source Security Series, Syngress. 2007, ISBN: 978-1-59749-099-3.
[8] R Perdisci, G Giacinto, F Roli. "Alarm clustering for intrusion detection systems in computer networks". *Engineering Applications of Artificial Intelligence*. 2006; 19: 429–438.
[9] F Valeur, G Vigna, C Kruegel, RA Kemmerer. "A comprehensive approach to intrusion detection alert correlation". *IEEE Transactions on Dependable and Secure Computing*. 2004; 1(3).
[10] P Ning, Y Cui, DS Reeves, D Xu. "Tools and techniques for analyzing intrusion alerts". *ACM Transactions on Information and System Security*. 2004; 7(2): 273–318.
[11] P Ning, D Xu. "Hypothesizing and reasoning about attacks missed by intrusion detection systems". *ACM Transactions on Information and System Security*. 2004; 7(4): 591–627.
[12] D Xu, P Ning. "*Alert correlation through triggering events and common resources*". in: Proceedings of the 20th Annual Computer Security Applications Conference (ACSAC 04). December 2004.
[13] S Noel, E Robertson, S Jajodia. "*Correlating intrusion events and building attack scenarios through attack graph distances"*. in: 20th Annual Computer Security Applications Conference (ACSAC'04). 2004, pp. 350–359.
[14] Charlie Scott, Paul Wolfe, and Bert Hayes. "Snort for Dummies". Willey Publishing. 2004, ISBN: 0-7645-6835-3.
[15] Rafeeq Ur Rehman. "Intrusion Detection Systems with Snort". Prentice Hall PTR. 2003, ISBN: 0-13-140733-3.
[16] X Qin and W Lee. "*Statistical causality analysis of infosec alert data"*. Proc: 6th International Symposium on Recent Advances in Intrusion Detection (RAID 2003), Pittsburgh, PA, September 2003.
[17] B Morin, H Debar. "*Correlation of intrusion symptoms: an application of chronicles"*. in: G. Vigna, E. Jonsson, C. Krgel (Eds.), Proceedings of the 6th International Symposium on Recent Advances in Intrusion Detection (RAID2003), Lecture Notes in Computer Science. vol. 2820, Springer, September 2003, pp. 94–112.
[18] P Ning, Y Cui, DS Reeves, D Xu. "*Towards Automating Intrusion Alert Analysis"*. in: 2003 Workshop on Statistical and Machine Learning Techniques in Computer Intrusion Detection. September 2003.
[19] P Ning, Y Cui, DS Reeves. "*Constructing attack scenarios through correlation of intrusion alerts"*. in: Proceedings of the 9th ACM Conference on Computer and Communications Security, Washington, D.C., November 2002, pp. 245–254.
[20] F Cuppens, A Miege. "*Alert correlation in a cooperative intrusion detection framework"*. in: Proceedings of the 2002 IEEE Symposium on Security and Privacy. IEEE Computer Society, Berkeley, California, USA, May 2002, p. 202.
[21] S Staniford, JA Hoagland and JM McAlerney. "Practical automated detection of stealthy portscans". *Journal of Computer Security*. 2002; 10(1/2): 105–136.
[22] P Ning, DS Reeves, Y Cui. "An Intrusion Alert Correlator based on Prerequisites of Intrusions". Technical Report TR-2002-01, North Carolina State University, Department of Computer Science. 2002.
[23] B Morin, L Me, H Debar, M Ducasse. "*M2D2: a formal data model for IDS alert correlation"*. in: Proceedings of the 5th International Symposium on Recent Advances in Intrusion Detection (RAID 2002). 2002, pp. 115–137.
[24] PA Porras, MW Fong, A Valdes. "*A mission-impact-based approach to INFOSEC alarm correlation"*. in: Proceedings of the 5th International Symposium on Recent Advances in Intrusion Detection (RAID 2002). 2002, pp. 95–114.
[25] ST Eckmann, G Vigna, and RA Kemmerer. "STATL: An Attack Language for State-Based Intrusion Detection". *J. Computer Security*. 2002; 10(1-2): 71-104,.
[26] A Valdés, and K Skinner. "*Probabilistic Alert Correlation"*. Proc: 4th International Symposium on Recent Advances in Intrusion detection (RAID), Springer Verlang, California, USA. 2001, pp. 54-68.
[27] H Debar, A Wespi. "*Aggregation and correlation of intrusion-detection alerts"*. in: Proceedings of the 4th International Symposium on Recent Advances in Intrusion detection (RAID), Springer Verlag, California, USA, 2001, pp. 85–103.
[28] DARPA Intrusion Detection Data Sets, available at http://www.ll.mit.edu/mission/communications/cyber/CSTcorpora/ideval/data/