Advances in Computer Science Research, volume 89
2019 International Conference on Wireless Communication, Network and Multimedia Engineering (WCNME 2019)

A Method of Network Behavior Recognition and Attack Scenario Reconstruction for Attack Kill Chain

Jiawei Du[1], Xing Zhang[1], Guowei Suo[1], Ronghua Guo[1] and Gang Lu[1]
[1]Chinese Luoyang Electronic Equipment Center, Luoyang 471003, China

***Abstract*—Currently, most successful network attacks are aimed at a particular target, composed of several attacks at different stages, and the attack is always carried out in a certain sequence, which coincides with the attack kill chain proposed by the US Army. Aiming at the typical multi-step attack type of attack kill chain, this paper proposes a hierarchical association analysis method for attack events based on directive database. By building a series of knowledge bases and using automatic means, the attack directives of key steps in the attack kill chain are reconstructed and generated, which can improve the accuracy of identifying and analyzing multi-step attack behaviors like attack kill chain.**

Keywords—attack kill chain; association analysis; knowledge database; directive reconstruction; behavior recognition

I. Introduction of Attack Kill Chain

A. Classification of Traditional Attacks

Traditionally, attacks refer to all attempts to destroy the integrity, confidentiality and availability of resources in an attempt to bypass computer security controls. Usually, the success of attack depends on the vulnerability of computer system and the effectiveness of security countermeasures. According to whether the attack steps can be divided, the attacks can be divided into simple attacks and multi-step attacks[1]. The former refers to an attack with an independent and indivisible attack purpose, usually consisting of a single attack step or an independent repetitive attack step; the latter refers to a series of attacks formed in a specific space-time by combining simple attacks according to a certain logical relationship, so as to achieve the purpose of attack that cannot be achieved by simple attacks alone. Multi-step attacks usually include multiple attack steps with strong temporal and logical relationships, and the purpose of these attacks is the same. Attack kill chain is a typical multi-step attack.

B. Generation and Definition of Attack Kill Chain

Militarily, the Advanced Research Projects Agency of the United States Department of Defense put forward the concept of kill chain in 2001, which was originally used in K (kill) in military C5KISR system. In cyberspace security, Lockheed Martin first proposed the concept of cyberspace attack kill chain in 2011. At present, the top security companies in the United States are studying it, and have obtained a large number of products and results. In May 2014, the United States announced the prosecution of five Chinese officers on the grounds of so-called cyber-theft, which was based on the report provided by MANDIANT Company based on the analysis of cyber kill chain. In 2013, the United States Federal Government NIST SP 800-150 "Draft Guidelines for Information Sharing of Cyber Threats" proposed to describe cyber security incidents with the kill chain. In 2014, the United States introduced the "Cyber Security Intelligence Sharing Act". So far, the attack kill chain has been formally included in American law.

The so-called attack kill chain refers to a series of continuous stages and activities that an attacker needs to succeed in an attack. These stages and activities form a chain of end-to-end action flow which is linked by "Reconaissance→Weaponization→Delivery→Exploitation→Installation→Command and control →Continuous attack"[2]. Fig. 1 is a seven-step model of network attack kill chain.

FIGURE I. ATTACK KILL CHAIN SKETCH MAP

II. Hierarchical Association Analysis and Recognition Processing of Attack Events

A. Advantages and Disadvantages of Attribute Similarity-based Attack Behavior Association Analysis

It is found that the same kind of attack often contains some similar attributes, so an attack behavior association analysis method based on attribute similarity degree appears. This method studies a large number of attacks, and finds that as long as any two alarm messages have the same attributes, and the similarity of these attributes is within a certain threshold, the two alarms may be related in the same attack [3]. Usually, the following steps are taken to judge: first, to define the similarity function of the same attributes of the alarm; second, to calculate the similarity of the alarm attributes according to the

attribute similarity function; third, to define the expected value of the attribute similarity and calculate the similarity of the alarm; fourth, to analyze the correlation of the alarm, when the new alarm appears, to compare the similarity of the common attributes of the existing alarm, if the similarity satisfies a certain value, the two alerts may belong to the same kind of attack.

The advantage of this method is that it reduces the redundancy of alarms and improves the efficiency of data processing of alarm information; moreover, this method does not need to know the relationship between alarms beforehand, and can automatically correlate and compare alarms through key attribute values [4]. The disadvantage of this method is that it is difficult to understand the attack itself, and the parameters such as similarity criterion, similarity threshold and weight coefficient are difficult to determine.

B. Construction of Hierarchical Attack Directive Database

At present, more and more attackers adopt multi-step attack. Analyzing and identifying multi-step attack can use hierarchical attack directive database. It is based on the sequence of attacks that have occurred. It is manually written and stored in an XML file in a tree structure. These files contain multiple attack directives. Each attack directive is composed of a rule tree. The nodes of the tree are the rules that match the attack steps in the attack directive. Association rules are formulated based on an attack directive composed of different attacks. Hierarchical association rules can represent a complete attack directive. They describe detailed alert information. They have the characteristics of simple format, clear structure and strong expansibility. They are suitable for behavior analysis and identification of typical multi-step attacks such as attack kill chain. There are many elements involved in the attack directive database, as shown in Figure 2.

directive= (id, name, priority, rule)	rule= (type, event, reliability, level)
event= (facility_id, facility_sid, ip_from, ip_to, port_from, port_to, protocol)	

FIGURE II. ATTACK DIRECTIVE DATABASE FACTOR COMPOSITION

Directive is a complex attack consisting of a series of single attacks. It is represented by a quaternion as directive= (id, name, priority, rule). It involves the identification of attack directive, the name of attack directive, the priority of attack directive and the rules that constitute the attack directive. Each layer of rules includes the type of security devices, events, attack efficiency probability and the rules that constitute the attack directive. In the matching process, only if the upper rule matches successfully, can the lower rule be matched.

Event refers to the basic event triggered by the attack, and it is also the main attack evidence submitted by most security devices to administrators. Each event has many attributes. The general set of attributes is event=(facility_id, facility_sid, ip_from, ip_to, port_from, port_to, protocol), which involves the type of event, the index number of security device, the IP address of source host and target host, the source port number or target port number of a specific level, and the protocol used in event generation.

C. Hierarchical Association Analysis of Attack Events Based on Directive Database

Hierarchical attack directive construction is the premise of attack event correlation analysis. When an attack event arrives, the association engine responsible for attack event association refers to the Hierarchical attack directive database to correlate the event. Matching tree is used to match the tree rules of attack event and directive database, as shown in Figure 3.

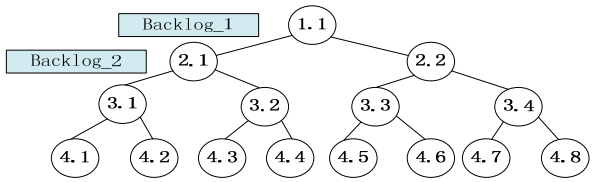

FIGURE III. MATCHING TREE

Each node in the graph represents a matching rule, and each matching rule contains several subsequent sub-matching rules. Association matching process is a hierarchical process, such as matching rule 1.1. After successful matching, two secondary matching rules 2.1 and 2.2 need to be further validated. Backlog-n represents a temporary state in the process. It stores a pointer to a rule node in the rule tree, also known as a priority matching queue. When an attack event is received, each backlog in the priority matching queue will be checked in turn to determine whether the rules pointed to are matched successfully. If the matching is successful, a new backlog will be created one by one according to the lower sub-rules of the rules pointed to, and inserted into the priority matching queue to match the lower rules. Each Backlog has different priority according to its policy configuration. The higher the priority is, the more likely it will attack the corresponding association matching analysis process.

III. RECONSTRUCTION OF NETWORK ATTACK SCENARIO FOR ATTACK KILLING CHAIN

Attack kill chain is realized by a series of attack steps and stages, and the purpose of attack is achieved through multiple attacks. These attacks do not exist in isolation and often have causal relationship. Therefore, the important step of scene reconstruction of attack kill chain is to establish causal knowledge database.

A. Establishment of Causal Knowledge Database

1) Causes and consequences of attacks

The so-called cause event is the prerequisite for the implementation of the attack, and the consequence event is all the possible impact of the success of the attack. Take an example of attack kill chain. Firstly, we lock the target for a host with Unix system, exploit a vulnerability in the verification process of a program in Unix system, develop and launch weaponized code for release, obtain root privilege through remote login, then install and run daemon and master control program on the penetrated host, and finally control it. The main control side attacks the target host by denial of

service. That is to say, the precondition set of the alarm is {Unix system, Sadmind service, root privilege, daemon and master program are installed on the infiltrated host}. The possible consequences of this attack are: by remote login to the host, the host can launch a denial-of-service attack on the target host, resulting in a large consumption of resources of the target host, leading to suspension or even downtime.

In order to facilitate computer recognition, we use predicates (words to describe the state, characteristics, nature of a person or behavior or the relationship between a person and a behavior[5]). In this paper, Alarm_Causes and Alarm_Effects are used to represent the cause set and the consequence set of the attack. The three preconditions for the above attack are: Alarm_Causes=IP_Unix.(target_ip, Unix) AND service (target_ip, sadmind)AND GetRoot Access (tarset_ip); consequence set Alarm_Effects = Alarm_DDoS (Target_ip) and iP_Service(Target_ip, Telnet).

By analyzing the preconditions and possible consequences of multiple attacks, the multi-step attacks belonging to the same attack process are correlated in proper order, and the attack scenarios are reconstructed, which can reveal the relationship between the alarm events, draw the attack route taken by the attacker, and identify the unknown attack behavior.

2) Design of causal knowledge database, the core of attack scenario database reconstruction

Usually, the events captured by security equipment have some relationship. For example, X event is the precondition of Y event, Y event is the precondition of Z event, Z event is the possible consequence of X event, and the mutual restriction of X, Y and Z is the cause-consequence relationship [6]. According to this relationship, we can correlate isolated alarm events, then describe the distributed attack process of multi-step attack, and finally reconstruct the attack scenario in the form of step diagram.

Because the core of attack scenario reconstruction is causal knowledge database, it defines the cause premise and possible consequences of each attack. First, we define some basic elements.

Definition 1: Hyper Alarm Type for short H, using triple H=(Alarm_Attributes, Alarm_Causes, Alarm_Effects). Among them, the set of alarm attributes, the set of reasons for alarm occurrence and the set of possible consequences after alarm occurrence are described.

Definition 2: Hyper Alarm is abbreviated as H. It fills in the specific attribute value of the alarm according to the given H type. Each attribute in H corresponds to the start time of occurrence.

Definition 3: For H, C (H) is used to denote the set of predicates in the cause set and E (H) is used to denote the set of predicates in the consequence set. Correspondingly, for h, C (h) and E (h) are used to denote C (H) and E (H) respectively.

Definition 4: For any two hyper-alarm instances hm and hn, if there are Alarm_Causes∈C(hn), Alarm_Effects∈E(hm) and satisfying hmBegin_Time<hnEnd_Time, the attack step hm is considered as the precondition for the attack step hn, and hn is the subsequent attack step of hm, which is recorded as hm→hn.

Definition 5: Each hyper-alarm class contains a set of causes and consequences expressed by predicates. For two hyper-alarm class instances X and Y, the presence of c∈C(Y) and c∈E(X) indicates that the attack represented by X is to prepare for the attack represented by B, which is recorded as X→Y. When several instances of hyper-alarm satisfy the above association conditions, the relationship between X and Y can be stored in the Alarm Sequence list sequence.

The definition of the above attack examples in causal knowledge database is as follows:

```
<HyperAlarmType
Name=”Killchain”Attack_Type=”DDos”>
<Alarm_Attributes>
<Attribute1Name=”Alarm_Source_Address”AtrType=”var
char(15)”></ Attribute1>
<Attribute2
Name=”Alarm_Source_Service_port”AtrType=”int”></
Attribute2>
……
</Alarm_Attributes>
<Alarm_Causes>
<Alarm_Cause1 Name=”SadmindPort”
<Arg Name=”Alarm_Desc_Address”></Arg>
<Arg Name=”Alarm_Desc_Service_port”></Arg>
<Arg Name=”Alarm_Time_Interval.Begin_Time”></Arg>
</Alarm_Cause1>
……
</Alarm_Causes>
<Alarm_Effects>
<Alarm_Effect1 Name=”GetRootAccess”
<Arg Name=”Alarm_Desc_Address”></Arg>
<Arg Name=”Alarm_Time_Interval.End_Time”></Arg>
</Alarm_Effect1>
```

B. Realization of Attack Scenario Reconstruction

Attack scenario reconstruction is to reconstruct the attacker's attack process by re-analyzing and re-organizing the alarm information and finally gathering the intrusion steps belonging to the same attack. Because of the relationship among the attack steps, the process used in the attack scenario construction is basically the same, so an automatic method can be adopted to deal with it [7]. According to Definition 5, by associating the causes and consequences of hyper-alarm classes, all possible attack sequences consisting of hyper-alarm classes can be obtained, stored in a linked list, and each attack sequence has an identification number as the identification number.

We can construct attack scenarios according to the following ideas; we can use alarm names to detect the existence of the type of hyper-alarm in the causal knowledge database for new alarm events, discard the event if it does not exist, instantiate the hyper-alarm if it exists, and put the real IP and

Port information in the alarm into the hyper-alarm predicate; then we can correlate the causes and consequences of the hyper-alarm to find out the order of the hyper-alarm. Whether there is a hyper-alarm association condition among detected alarm instances, if there is, it is stored in the alarm sequence list.

In order to analyze the attack situation in the network for a period of time, we can set a time range for the construction of attack scenario. When the time limit is set, the basic attack sequence of constructing attack scenario graph is almost formed. The attack scenario graph is reconstructed according to the causality of each step in the sequence, and the scenario's logo is reported for attack response processing. Thus, the process steps of attack scenario construction are the same, but the composition of each attack scenario is different, so the process customization method can be used to build attack scenario, as shown in Figure 4.

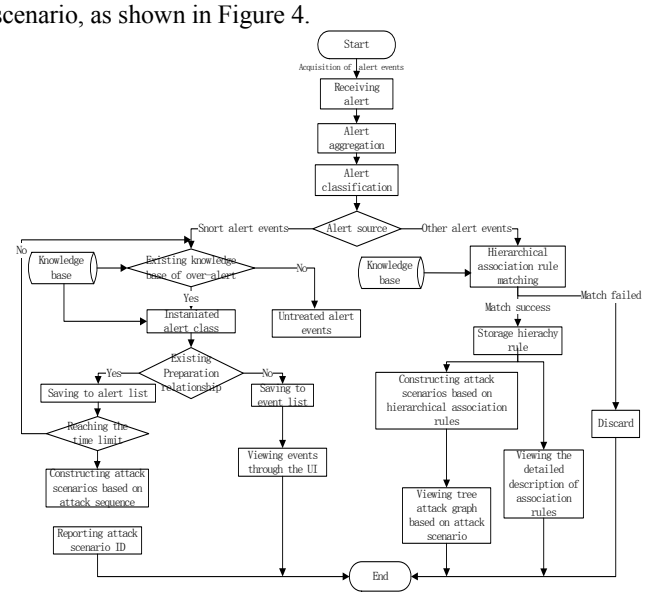

FIGURE IV. FLOW CHART FOR AUTOMATIC ATTACK SCENARIO CONSTRUCTION

As can be seen from the above figure, there are two ways to deal with alarm events according to the source of events. One is the alarm event generated by IDS devices such as Snort, and the other is the alarm generated by other security devices. Because these two kinds of alarms are very different in themselves, the processing flow is also different. Firstly, according to the different sources of alarm events, we need to classify the alarm sources after aggregating them. For non-Snort alarm events, if there is a hierarchical relationship between events, the hierarchical attack scenario can be detected by matching the association rules, and the scenario number can be stored in the sequence table. According to the scenario number, the hierarchical attack scenario can be reconstructed into a tree attack graph, which can be displayed to security administrators in a more intuitive way. If it fails to match successfully, it is necessary to calculate whether the attack meets the risk condition of alarm, and if it does, it will produce alarm, and if it fails to achieve no alarm.

Through a series of processing, the data stored in the attack sequence table is shown in Table 1, which records some alarm sequence pairs with causal-consequence relationship. The first column is the serial number, the second column is the attack name as the cause in the attack sequence, and the third column is the attack name of the consequence in the attack sequence. Each row in the table forms an attack sequence.

TABLE I. ATTACK SEQUENCE TABLE

Inde	Cause	Consequence
1	ATTACK-RESPONSES 403 Forbidden923	ATTACK-RESPONSES directory listing35
2	SNMP request udp4537	SNMP public access udp4726
3	ATTACK-RESPONSES 403 Forbidden55	ATTACK-REPONSE directory listing5
4	MS-SQL version overflow attempt4474	RSERVCICES rsh root4478
5	SNMP request udp4491	SNMP public access udp4492
6	FTP_Syst1533	“HTTP_Shells”1508
7	RPC portmapsadmind request UDP50	RPC sadmind UDP NETMGT_PROC_SERVICE CLIENT_...
8	RPC sadmind query with root credentials attempt UDP37	ATTACK-RESPONSES directory listing57
9	RPC sadmind UDP NETMGT_PROC_SERVICE CLIENT_...	ATTACK-RESPONSES directory listing57
10	RPC sadmind UDP NETMGT_PROC_SERVICE CLIENT_...	ATTACK-RESPONSES directory listing71

IV. CONCLUSION

Aiming at the network directed multi-step attack-attack killing chain, this paper analyses the advantages and disadvantages of the traditional attributes similarity-based attack behavior correlation analysis method, introduces a hierarchical attack event correlation analysis method based on directive database, introduces the principle and model of attack scenario construction in detail, and designs and customizes the knowledge database with an example. Processing, and finally generate the attack sequence list needed to build the attack scenario. This method clearly shows the relationship between related multi-step attack events, and can provide effective support for security administrators to distinguish network attack behavior and respond to security protection.

REFERENCES

[1] Ioan-CosminMIHAI, Stefan PRUNA, Ionut-Daniel BARBU. CyberKill Chain Analysis[EB/OL].

http://ijisc.com/articles/2014-02-04.pdf,2015-02-14.

[2] Schwartz, Matthew. Beyond Firewalls and IPS: Monitoring Network Behavior[EB/OL].

http://esj.com/articles/2006/02/07/beyond-firewalls-and-ips- monitoring-network-behavior.aspx,2015-02-14.

[3] Ahmed Youssef, Ahmed Emam. network intrusion detection using data mining and network behaviour analysis[EB/OL].

http://airccse.org/journal/jcsit/1211csit07. pdf, 2015-02-14.

[4] Fredrik Valeur, Giovatmi Vigna and Christo Pher Kruegel. A comprehensive approach to intrusion detection alert correlation. IEEE Trans. Dependable and Secure Computing. 2004: 146-169.

[5] Salles-Loustau G, et al. Characterizing attackers and attacks: An empirical study[C]//Dependable Computing(PRDC), 2011 IEEE 17th Pacific Rim International Symposium on. IEEE, 2011:174-183.

[6] Wen Hui, Xu Kaiyong. Research on Network Security Event Association Analysis and Active Response Mechanism [J]. Computer application, 2010, 27(4): 60-63.

[7] Wang Hang, Gao Qiang, et al. Computer Application of Network Vulnerability Evaluation Based on Attack Graph and Security Measurement [J]. Computer Engineering.2010, 36(3):128-130.