Network Security Situation Assessment Based on Text SimHash in Big Data Environment

Pengwen Lin and Yonghong Chen
(Corresponding author: Yonghong Chen)

College of Computer Science, Technology, Huaqiao University
No.668 Jimei Avenue, Xiamen, Fujian 361021, China
(Email: djandcyh@163.com)

(Received Feb. 1, 2018; Revised and Accepted June 21, 2018; First Online Dec. 10, 2018)

Abstract

The existing methods of network security situation assessment have high complexity and not effectively in the big data environment. This paper proposes an assessment model based on SimHash in the big data environment. First, a large-scale network is divided into multiple modules. Then get secure data of the internal nodes of modules. Based on the SimHash algorithm, in turn quantifies the node security situation, module security situation, network security situation. Finally, the experiment is designed to verify the model. Results show that the model can effectively adapt to a large-scale network and have high accuracy.

Keywords: Big Data; Complex Network; Network Security Situation Assessment; Text SimHash Algorithm

1 Introduction

With the continuous development and popularization of network technology, the amount of data is growing at an unimaginable speed in recent years. More and more people use the term “big data” to describe and define the generation of massive data during the information explosion era. Today, both industry and academia have generated a great deal of interest in the field of big data. The value inherent in the big data has become the driving force behind the storage and processing of big data [19]. Many studies, including [26] and [17], have made cloud storage widely used, which provides the basis for big data analysis. [14] pointed out that because of the concept of data processing changes in the big data environment, many scholars have already begun to study the big data analysis technology deeply. For instance, [24] elaborates on the techniques of big data analysis and the challenges it faces.

On the other hand, the Internet has become a critical infrastructure and Internet security has a direct bearing on the fundamental interests of the public [11]. Today the scale of the network is getting bigger and bigger, the topological structure and environment of the network are more and more complicated, cybersecurity incidents have risen dramatically, and the issues of cybersecurity have become increasingly prominent [1]. In order to address these challenges, Intrusion Detection System, Firewall, security-audit and other security protection and management system have been widely used. However, these products all consider network security from a single aspect. The lack of a synergistic mechanism between each of these products can only be used by themselves and form isolated islands of information.

Network security situation assessment has been proposed by many scholars under such a background. Strengthening the assessment of the security situation of information systems is a necessary management measure to protect the core information infrastructure [8]. Network security situation assessment means that the security-related elements are perceived and acquired from the perspective of time and space through technological means, and the network security status is judged through the integrated analysis of data information [12, 23].

Endsley in [7] put forward and defines the concept of situation awareness for the first time in 1988. Endsley believed that situation awareness was the perception of the elements of the environment within a volume of time and space, the comprehension of their meaning, and the projection of their status into the near future. However, this concept was mainly applied to the aviation field. Bass [2] first introduced the concept of situational awareness into the field of network security in 1999 and proposed an evaluation framework based on intrusion detection system, but it did not implement it. Gorodetsky *et al.* [9] proposed an evaluation method of network security situation based on asynchronous data flow, which used multi-agent anomaly detection network data flow to analyze various security events to get the security situation. But usually, data flow cannot represent all the basic security information in networks. XiuZhen-Chen *et al.* [4] proposed a quantitative hierarchical threat evaluation model for network security. The evaluation policy used in the model is from bottom to top and from local to the whole. The threat metrics of services, hosts, and local networks are calculated by weighting the importance of services and hosts based on attack frequency, severity and network bandwidth consumption, and then evaluates the status of the security threat. [25] proposed a situational awareness model, which analyzes the current situation in the network environment and generates corresponding measures. The situation is influenced 00by the measures and then a new trend is formed. [13] references the mechanism of body temperature change caused by biological immune system imbalance, analyzes antibody concentrations change caused by the change process of various types of detectors in computer immune system, and proposes a quantitative risk evaluation model for network security based on body temperature. [27] classifies the security events in the network to identify attackers, then casually correlates each attack scenario, identifies the corresponding track and phase of an attack, and finally establishes the situation quantitative criteria, combining the attack phase and its threat index evaluation of the cybersecurity situation. However, all of the models mentioned above are difficult to adapt to the situation of a large quantity of data and fast generating of data in the big data environment. Secure data in [29] includes intrusion detection log, firewall log, virus log, network scanning, illegal external links, and running state of equipment and real-time alarm. Then combined with the PSR method, the fuzzy logic model and the entropy weight method in an empirical study for feasible urban public security evaluation modeling. It gives us a good reference value. However, it does not give a comprehensive measure of the value of the situation. [5] used SIEM as input data, and based on CVSS and attack models, the technologies used including a set of integrated security metrics to conduct risk assessment. [6] considered the uncertainty of the assessment data and translates it into an objective weight through uncertainty measures. Then, using D-S evidence theory and pignistic (from the Latin pignus, a bet) probability transformation, a consensus decision about the degree of network security risk is obtained.

The above methods provide a feasible solution for researching network security situation assessment. In the meanwhile, there are some common defects. For instance, the existing methods are hard to adapt to the big data environment because of the high complexity of evaluation models and algorithms, which leads to the deviation of the quantitative results of the network security situation, and the feedback is not timely enough. To address these problems, this paper presents a network security situation evaluation model based on the SimHash algorithm to adapt to the big data environment. First, the method of complex networks is used to divide a large-scale network. Fusion of multi-source heterogeneous data on each node in the local network to obtain the secure data, and then use SimHash to assess the security situation of nodes quickly and efficiently. Finally, integrated by the weights of nodes and modules to quantitative the status of network security.

The limitation of the model in this paper is that in a large-scale network, the topology is dynamically changing. However, the division of modules in our model is completed before the assessment of the network security situation and does not change in real time following the change of the topology.

The rest of this paper is arranged as follows: In Section 2 we introduce the framework of our assessment model. Section 3 gives the key algorithms and related theoretical basis. Section 4 presents the experimental results and discussion. The paper is concluded in Section 5.

2 Network Security Situation Assessment Model

A large-scale network usually contains a great number of hosts, network devices, and various detection systems, and these detection systems monitor the network from different perspectives and generate logs and alerts. Traditional network security situation assessment usually uses only a single alert or log detected, and the single data source also directly leads to the deviation of the assessment result from the actual situation. And the method of evaluation often adopted a relatively complicated algorithm, which directly affected the timeliness of the assessment, and delays the best time for the network administrator to take measures. Aiming at these problems, the network security situation assessment model based on the SimHash algorithm in the big data environment is put forward.

Firstly terms used in the assessment model are explained.

Topology (T). It's graph structure which used to represent the information about nodes and their connection in a large-scale network environment.

Service (S). It refers to the services provided by the node to determine the weight of the node.

Log (L). It contains information such as system log, security log, application log, and alert log generated during network operation. The information of every log can be characterized by a sextuple $(id_l, time_l, type, info_l, id_{st}, id_{dt})$, where (id_l) is the unique identification of log, $time_l$ is the time when log generates, $type$ is the type of log, $info_l$ is the description of log, id_{st} is the identification of the node which generates log, and id_{dt} is the identification of the node which is the target of the security event.

Vulnerability (V). It refers to the vulnerability of the node and determines the success probability of an attack when it occurs. Every vulnerability information could be characterized by a quadruple $(id_v, time_v, pro_v, impact_v, info_v)$, where id_v is the unique identification of vulnerability, $time_v$ is the time when vulnerability scans, pro_v is the probability of successful exploitation, and $info_v$ is the description of the vulnerability.

Attack (A). It represents the attack on the node. The information of every attack can be characterized by a sextuple $(id_a, time_a, st, dt, info_a, id_v)$, where id_a is the unique identification of attack, $time_a$ is the time when attack occurs, st and dt represented the source and destination of attack respectively, $info_a$ is the description of attack, and id_v is the identification of vulnerability which used by attack.

Node situation awareness (NSA). It's the value of the security situation of the node and consists of topology, vulnerability and attack and denoted by $NSA = (T, V, A)$.

Module situation awareness (MSA). It's the value of the security situation of the module and consists of NSA and the weight of the nodes in the module and denoted by $MSA = (NSA, \omega_{node})$.

Network situation awareness (SA). It's the value of the security situation of a large-scale network and consists of MSA and the weight of the modules and denoted by $SA = (MSA, \omega_{module})$.

Then the framework of the model is shown in Figure 1.

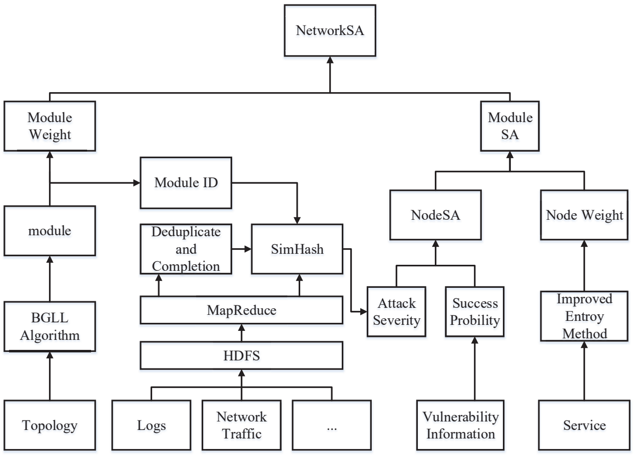


Figure 1: Network security situation assessment model

The calculation steps is shown in Table 1.

3 Network Security Situation Assessment Based on Big Data Analytics and SimHash

In this section, first, we divide the network into multiple modules and preprocess the data on the Hadoop platform. Then we introduce SimHash algorithm and use it to evaluate the security of the nodes. Finally, we determine the weight of the nodes and calculate the security situation of module and network.

3.1 Data Preprocessing

Due to the data obtained from various data sources such as logs and network traffic, their format is different, their generation is fast and the data contains dirty data. All of these leads to data exchange and sharing cannot be performed with each other efficiently. At the same time, [21] pointed out today's sophisticated network-attacks that occur across multiple dimensions and stages, traditional platforms will have no chance to defend a network.

To address these problems, the idea of the module is brought from the complex network into network security situation assessment.

The complex network generally consists of a mass of nodes and the connections between nodes are seriously complex. A complex network is widely used in various scientific fields to model and analyzes complex systems. Many networks have a community structure. Community structure is there are many associations in the network, and the connection among these associations is relatively sparse and the connection within the associations is relatively dense. Community discovery is using information contained in the topological structure from the complex network to resolve its modular community structure.

Community module index Q [20] is usually used to characterize the strength of community characteristics. Defined as in Equation (1):

$$Q = \frac{1}{2m}\sum_{ij}\left(A_{ij} - \frac{k_i k_j}{2m}\right)\delta(C_i, C_j). \tag{1}$$

Where k_i and k_j are the degrees of nodes, A_{ij} are the weight of the edge between node i and node j, C_i is the community of node i, m is the total number of network edges. $\delta(C_i, C_j) = 1$ when $C_i = C_j$, otherwise is 0. The value of Q is between 0 and 1, generally $Q = 0.3$ as the lower bound of the social structure of the network.

In this paper, we use BGLL algorithm [3] to classify a large-scale network. The algorithm uses the positive or negative of ΔQ to determine whether the ith node should join the module the jth node belongs to. ΔQ is defined in Equation (2):

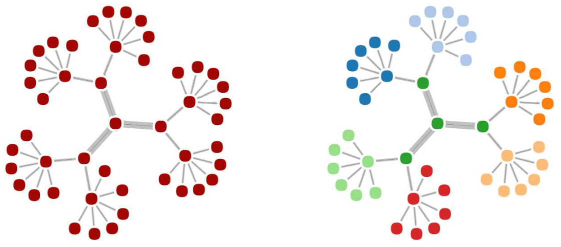
$$\Delta Q = \left[\frac{\sum_{in} + 2k_{i,in}}{2m} - \left(\frac{\sum_{tot} + k_i}{2m}\right)^2\right] - \left[\frac{\sum_{in}}{2m} - \left(\frac{\sum_{tot}}{2m}\right)^2 - \left(\frac{k_i}{2m}\right)^2\right]. \tag{2}$$

Where $\sum_{in}$ is the sum of the weights of all the edges inside the community; $\sum_{tot}$ is the sum of the weights of all the edges associated with the nodes inside the community; $k_{i,in}$ is the sum of weight of all edges connected to the community C. An example of dividing the network is shown in Figure 2.

Next, multi-source heterogeneous data, which is generating by the nodes inside each partitioned module is integrated. The purpose of data integration is to organize data in various independent systems into a whole

Table 1: Assessment steps

Step 1	Modules and its weight are obtained by using algorithms for detecting community structure in a complex network to divide a large-scale network.
Step 2	Collecting network security situation elements and then upload to the distributed file system to storage.
Step 3	The type of attack and the number of the attack in a period is obtained by preprocessing elements of network security situation.
Step 4	Inside the module, scan the vulnerabilities of nodes and calculate the success probability for each type of attack.
Step 5	For each node in the module, an algorithm based on SimHash is used to calculate the severity of the attack by attack type and a number of attacks.
Step 6	According to the severity of the attacks and the probability of successful attacks, node security situation is calculated.
Step 7	The weights of the nodes are calculated by using the services provided by the nodes, and then module security situation is obtained according to the security situation of the nodes.
Step 8	Using the module weight, combined with the security situation of modules, get the network security situation.



(a) predetect

(b) detect

Figure 2: An example of dividing

according to certain rules by some technical means, so that other systems or users can access data effectively. First, collect multi-source heterogeneous data, and then upload them to the distributed file system for storage. Because of the correlation between the data generated by various detection systems, on the one hand, there is a large amount of redundant information in these data and cannot be directly used in the network security situation evaluation. On the other hand, the detection system also has omissions and false positives, and the data of multiple detection systems will be merged to complement each other. On the basis of the distributed file system, it is possible to unify the format of multi-source heterogeneous data, excluding a lot of noise data which are not related to network security situation assessment, and merge duplicate attribute data. Finally, these pre-processed data are stored in the database as the data that can be used directly by the network security situation assessment.

3.2 Node Security Situation Assessment Based on Simhash

3.2.1 SimHash

SimHash algorithm [10] is an efficient algorithm uses to find similar texts. It avoids the complicated way of comparing texts with each other, which greatly improves the efficiency compared with algorithms such as cosine similarity, Euclidean distance, Jaccard similarity coefficient.

SimHash algorithm is a type of dimension reduction method essentially, which maps high-dimensional vectors into smaller-sized signatures to represent the features of the original vectors. The main character is the Hamming distance between two signatures is positively correlated with cosine similarity between the corresponding feature vectors. [18] and [22] improve SimHash algorithm and apply the improved algorithm to different fields. This brings great inspiration to this research.

The SimHash algorithm is described in (Algorithm 1).

```
Algorithm 1 SimHash
Require: Text $T$, the length of hash $b$.
Ensure: Array $W[\ldots]$.
Begin
$F(t) \leftarrow$ feature vector in $T$
$W \leftarrow$ array of b zeros
for $i \in F(t)$ do
    $\phi_i \leftarrow \text{TraditionalHash}(i)$
    for $j = 1$ to $b$ do
        if $\phi_{ij} = 1$ then
            $W[j] \leftarrow W[j] + \omega_i$
        else
            $W[j] \leftarrow W[j] - \omega_i$
        end if
    end for
end for
for $j = 1$ to $b$ do
    if $W[j] \geq 0$ then
        $W[j] \leftarrow 1$
    else
        $W[j] \leftarrow 0$
    end if
end for
return $W$
End
```

3.2.2 Text Processing

Before using the SimHash algorithm for node security situation assessment, we need to construct the text as the input of the algorithm.

Firstly, a text is randomly generated, and the words that make up the text are not repeated with each other, and the number of words that make up the text is related to the total number of attacks during that time.

Then assign different words for different types of attacks. These words do not overlap with the words in the text.

Finally, extract the attack information within a period of time, and calculate the attack number for different types of attack. If there are n types of attacks, n copies of the original text are generated. For each type of attack, according to the number of attacks, replace some words in the copy with the assigned words.

According to the above description, if there are several attacks over a period of time, several texts which modified will eventually be obtained. Utilizing the SimHash algorithm to get several hash values corresponding to these texts which modified, and compare the Hamming distance between these hash values and the hash value generated by the original text.

Hamming distance is the number of different bits between the hash values of two b-bits that can be used to estimate the similarity between two vectors. The greater the Hamming distance, the less similarity between the two vectors is. This feature can be used to quantify the severity of a certain attack on a node over a period of time.

For the existing vulnerabilities, different types of attacks have different successful probability. If there are numerous attacks over a period of time, we will get several Hamming distances. We need to combine these Hamming distances with the corresponding attack success probability, and then reduce the result.

3.2.3 Assessment Algorithm Based on SimHash

Traditionally, SimHash algorithm is used for web page deduplication and document similarity detection. Due to the huge amount of data volume is generated by various security devices in a large-scale network, an efficient network security situation evaluation algorithm is urgently needed to enable network administrators to quickly understand the current security status of the network. However, most existing evaluation algorithms have a disadvantage in the computation time because of its complexity. So it's difficult to apply to a large-scale network environment.

In order to solve the problems, we introduce SimHash algorithm to the network security situation awareness. Based on SimHash, we propose our node security situation assessment algorithm. First, we use text processing which described above to generate pre-attack text and post-attack text. And then use these texts to quantify the severity of the attack and finally quantify the security situation of the node. The algorithm is shown in (Algorithm 2).

Algorithm 2 Node security situation assessment algorithm

```
Require: Attack information $A$, vulnerability information $V$.
Ensure: Node security situation $NSA$.
Begin
$b \leftarrow$ the length of hash
$d \leftarrow 0$
$T_1 \leftarrow$ randomly generate $n$ words and each word isn't repeating
$F_1(t) \leftarrow$ feature vector on $T_1$
$H_1 \leftarrow \text{SimHash}(T_1, b)$
for $a \in A$ and $i = 0$ to $n$ do
    $word \leftarrow$ randomly generate a word
    $T_2 \leftarrow \text{replace}(T_1, i, word)$
end for
$F_2(t) \leftarrow$ feature vector on $T_2$
$H_2 \leftarrow \text{SimHash}(T_2, b)$
for $i = 1$ to $b$ do
    if $H_1[i] = H_2[i]$ then
        $d \leftarrow d + 1$
    end if
end for
for $a \in A$ do
    if $id_v$ in $V$ then
        $NSA \leftarrow NSA + d \cdot impact_v \cdot pro_v$
    else
        $NSA \leftarrow NSA + 0$
    end if
end for
return $NSA$
End
```

Based on vulnerability information about a node, the success probability of the attack is obtained and the security situation of the node is calculated using Equation (3).

$$NSA = \sum_{i=1}^{n} (svy_i \times p_i). \tag{3}$$

Where svy_i is the severity of the ith attack, a node may be attacked by many types of attack, p_i is the success probability of the attack based on vulnerability information.

3.3 Determine the Weight of Node

The improved entropy method [16] is used to determine the node weight. In information theory, entropy reflects the degree of disorders of information and is a measure of uncertainty. The smaller the entropy of an index, the smaller the uncertainty and the greater the amount of information carried, the greater the impact of this index on the comprehensive evaluation. The main steps of calculation are as follows:

Step 1: Depending to the service status of the host node, constructs a judgment matrix **R**:

$$\mathbf{R} = \begin{pmatrix} r_{11} & r_{12} & \dots & r_{1n} \\ r_{21} & r_{22} & \dots & r_{2n} \\ \vdots & \vdots & \ddots & \vdots \\ r_{m1} & r_{m2} & \dots & r_{mn} \end{pmatrix} \quad (4)$$

Where r_{ij} is the quantized value of the j service for the i nodes, 0 if it contains. n is the total number of nodes in the network and m is the total number of services in the network.

Step 2: Using traditional concept of entropy to calculate the entropy of the jth services(H_j):

$$H_j = -\Big(\sum_{i=1}^{n} f_{ij} \ln f_{ij}\Big)/\ln n \quad (5)$$

$$f_{ij} = r_{ij}/\sum_{i=1}^{n} r_{ij} \quad (6)$$

Where f_{ij} is the proportion of the ith nodes under the jth service in the service. Obviously, if $f_{ij} = 0$ that $\ln f_{ij}$ is meaningless, so the calculation of f_{ij} is modified to be:

$$f_{ij} = (1 + r_{ij})/\sum_{i=1}^{n}(1 + r_{ij}). \quad (7)$$

Step 3: Calculating the difference coefficient for the jth service g_j:

$$g_j = (1 - H_j)/(m - E_c) \quad (8)$$

$$E_c = \sum_{j=1}^{m} H_j; 0 \le g_i \le 1, \sum_{j=1}^{m} g_i = 1 \quad (9)$$

For the jth service, the smaller the entropy, the greater the difference coefficient, the greater the impact on the node is.

Step 4: Calculating the objective weight (w_i) of each node in the network:

$$w_i = \sum_{j=1}^{m}(g_j/\sum_{j=1}^{m} g_j) \cdot f_{ij} \quad (10)$$

3.4 Calculate the Value of Network Security Situation

The previous section calculates the node's security situation NSA and the weight of the node. Then use Equation (11) to quantify network security situation of the module:

$$MSA = \sum_{j=0}^{m}(NSA \times \omega_{node}). \quad (11)$$

Where ω_{node} is the weight of the node.

Finally, use Equation (12) to quantify the network security situation:

$$SA = \sum_{i=0}^{n}(MSA \times \omega_{module}). \quad (12)$$

Where ω_{module} is the weight of the module.

4 Experiment and Analysis

To verify the applicability of the proposed model, we selected the 2000 DARPA assessment dataset [15] provided by MIT Lincoln Lab datasets as experimental data. This dataset provides two attack scenarios LLDOS1.0 and LLDOS2.0.2 and contains network traffic and host audit logs, which can be used as a data source for the proposed model. In this paper, we will conduct an assessment of network security situation against these attack scenarios.

4.1 Experiment Environment

Due to there are many network nodes involved in the dataset, it is not clear enough if the complete network topology is drawn. Therefore, the network topology contains the key nodes only.

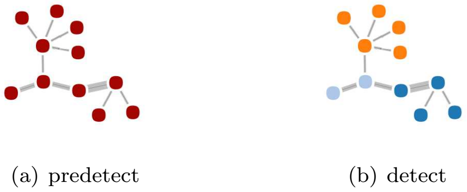

(a) predetect (b) detect

Figure 3: Network division

As shown in Figure 3, the figure on the left shows the network topology that contains the key nodes. We use the BGLL algorithm to divide the network, and the figure on the right shows the result. As you can see from the figure on the right, we divide the network into three modules, which are consistent with the dataset in which all the nodes are distributed in three regions: inside, outside, and in the DMZ. Besides these key nodes, we still need to consider the weight of those nodes that are not attacked in the network security situation assessment process.

Based on the information provided in the dataset, vulnerability information and the probability of success is shown in Table 2.

The service information of the key nodes in the network is shown in Table 3.

4.2 Node Security Situation Assessment Based on SimHash

Algorithm 2 is used to evaluate the security situation of the node. First, we write detection rules of the IDS to

Table 2: Network host vulnerability information

Vulnerability information	Mill	Locke	Pascal	Hume	Robin	af.mil	pro	impact
ICMP incorrectly configured	√	√	√	√	√	×	1.0	0.1
SunRPC incorrectly configured	√	√	√	×	×	×	0.8	0.2
Sadmind buffer overflow	√	√	√	×	×	×	0.8	0.8
RCP incorrectly configured	√	√	√	×	×	×	1.0	0.2
HINFO query incorrectly configured	√	×	×	×	×	×	0.8	0.6
SYN Flood	×	×	×	×	×	√	0.7	1.0

Table 3: Network hosting service information

Service information	Mill	Locke	Pascal	Hume	Robin	af.mil
HTTP	×	√	√	×	√	√
FTP	√	√	√	×	×	×
TELNET	√	√	√	×	×	×
DNS	√	×	×	×	×	×
SMTP	×	×	×	√	×	×
POP3	×	×	×	√	×	×

analyze the network traffic and get the alerts. Then upload these alerts and system logs to the distributed file system (HDFS) to storage. Because of the intrusion detection system alerts, some are found in the logs. Then we design MapReduce program to analyze the data on the HDFS to exclude duplicate Data, and finally get the attack information A, according to Table 2 to construct vulnerability information V. Taking A and V as the input data of the Algorithm 2, the security situation values of each node are generated. In order to ensure that the chart clearly, we only plot the security situation of the three nodes and shows in Figure 4.

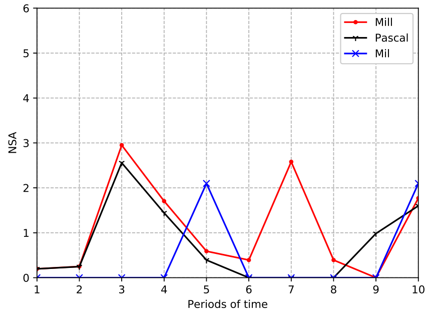


Figure 4: Node security situation

It can be seen from Figure 4 that in the periods of 1, 2 and 6, the nodes are scanned and the impact of scanning on the nodes is minimal, so we also calculated the NSA to be lower. In the period of 3, Mill and Pascal suffered a buffer overflow attack, so the value of NSA we calculated is higher. In the periods of 3 and 10, the attacker controls Mill and Pascal to initiate a DDoS attack on Mil node. Therefore, the NSA of the three nodes in both periods is greater. It can be seen that the algorithm we use to evaluate the node security situation is accurate, which can reflect the severity of each node being attacked.

4.3 Module Security Situation Assessment

We use the improved entropy method, according to the services provided by each node, get the weight of each node in the module, and the security situation of the module is calculated by Equation (11). There are 3 modules in our experiment: INSIDE, OUTSIDE, and DMZ. The result of the security situation of INSIDE module is shown in Figure 5.

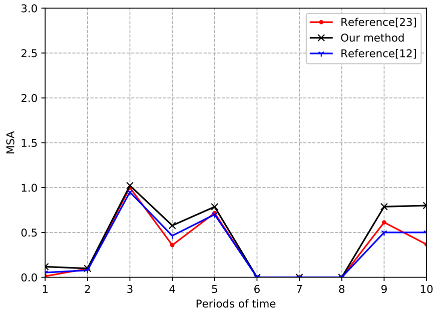


Figure 5: Module security situation assessment

There is no concept of module in [16] and [28]. Therefore, the results of these two methods are obtained by calculating the sum of the security situation of nodes which in INSIDE module. As can be seen from Figure 5, the trends of three methods are consistent. However, in some key stages, our method gets a higher value of the situation. For example, in the periods of 5 and 10, the attacker will have Pascal nodes. The attacker performs a DDoS attack on the Mil node through Pascal and mill node, and Pascal node is in the INSIDE module. Therefore, we hold the view that the value of the INSIDE module is higher in these two periods, but in [28] it is lower in the period of 10 than before.

4.4 Network Security Situation Assessment

In the base of MSA, the security situation of the whole network is calculated using Equation (12), where the weight of the module is obtained by dividing the whole network using the BGLL algorithm. The network security situation obtained is shown in Figure 6, where the larger the SA is, the more insecure the network is.

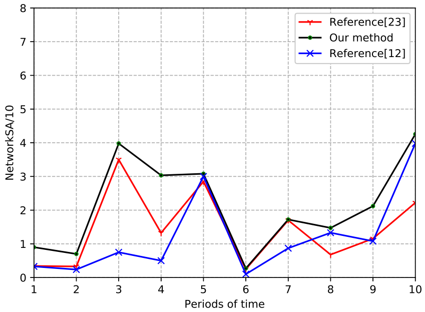


Figure 6: Network security situation assessment

It can be seen from the Figure 6 that our model can better reflect the security state of the network than [16] and [28]. As in the periods of 5 and 10, the network was attacked by a large number of distributed denial-of-service attacks, so our assessment results were relatively high for both periods. In the periods of 3 and 8, key nodes are compromised and the root access is taken by the attacker. At this point, the subsequent series of attacks are all based on root access, so both of these evaluations higher. In the end, network administrators can decide whether to take action or not based on the network security situation.

4.5 Performance Analysis

On the storage, we only need to store the hash values corresponding to the initial text and the modified text respectively. This saves a lot of space compared to storing network traffic and logs directly. And we can store these hashes in HDFS in the big data environment. Using the LZO compression algorithm to compress the data, which can save the disk space occupied by the data further and speed up the data transmission in the disk and the network, so as to improve the processing speed of the system. LZO compression algorithm allows us to split the compressed algorithm allows us to split the compressed file processing, file segmentation in the big data processing is very important. It will affect the number of parallel execution of the job, thus affecting the efficiency of the implementation of the job. Table 4 shows the comparison of several compression algorithms.

In the big data environment, the traditional situation assessment algorithm is more complex. When the data of the network node increases sharply, network status cannot be feedback to the network administrator timely and effectively. To test the efficiency of our algorithm, we randomly generate two different types of attacks in every period, and the number of each type also generate randomly. The length of hash value which is calculated by the SimHash algorithm is 64-bits. The result is shown in Figure 7. It can be seen that the time complexity of our algorithm is closed to $\mathcal{O}(n^2)$. Through the theoretical analysis of the algorithm, it can be seen that although there are several loops, only one is a two-layer loop, so the time complexity is $\mathcal{O}(n^2)$ is correct. The algorithm calculates the security situation of four million nodes takes only about 70 seconds. Moreover, this is just a node's computing power, we can dynamically increase the number of computing nodes if it is needed in the big data environment.

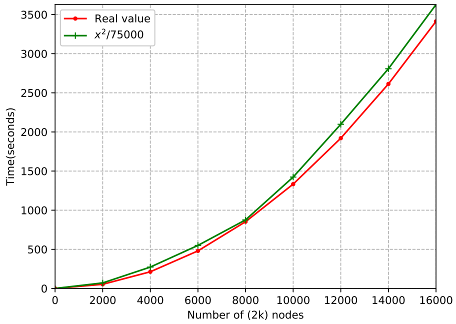


Figure 7: Calculating speed

Our model can be applied to a large-scale network, because using BGLL algorithm divides a large-scale network into multiple modules, as long as there is a topology, the topology can be determined when the network is generated. Alerts, a variety of logs and other network security-related data can be collected and uploaded to HDFS to storage and analysis as they are generated, so this process is generally done in parallel with data generation.

5 Conclusions

This paper analyzes and compares the existing evaluation methods of network security situation. To address the problem that these methods are difficult to adopt in a large-scale network environment, this paper proposes a network security situation assessment model based on text SimHash algorithm in the big data environment. The model divides a large-scale network into multiple modules by using the method of dividing the network structure of complex networks, and then analyzes the nodes in each module and quantifies the node security situation, the module security situation, and the network security situation gradually. Administrators know the status of network security at any time. And the experimental analysis

Table 4: Compression algorithm comparison

Compression algorithm	Initial file size	Compressed file size	Compression speed	Decompression speed	Separability
LZO	8.0GB	2.0GB	148.95MB/s	234.06MB/s	Yes
GZIP	8.0GB	1.3GB	33.99MB/s	113.78MB/s	No
BZIP2	8.0GB	1.06GB	6.13MB/s	24.5MB/s	Yes

verifies the applicability and characteristics of the evaluation model we proposed.

In the future, we will improve a large-scale network security situation assessment model and the quantitative assessment method, and on this basis, make a prediction of a large-scale network security situation. And designing a multidimensional visualization system to help administrators seize the status of network security more accurately.

Acknowledgments

This study was supported by the National Natural Science Foundation of China (No.61370007), Program for New Century Excellent Talents of Fujian Provincial (No. 2014FJ-NCET-ZR06), and the Postgraduate Scientific Research Innovation Ability Training Plan Funding Projects of Huaqiao University (1611314012).

References

[1] A. A. Al-khatib, W. A. Hammood, "Mobile malware and defending systems: Comparison study," *International Journal of Electronics and Information Engineering*, vol. 6, no. 2, pp. 116–123, 2017.

[2] T. Bass, "Intrusion detection systems and multisensor data fusion," *Communications of the ACM*, vol. 43, no. 4, pp. 99–105, 2000.

[3] V. D. Blondel, J. L. Guillaume, R. Lambiotte, and E. Lefebvre, "Fast unfolding of communities in large networks," *Journal of Statistical Mechanics: Theory and Experiment*, vol. 2008, no. 10, pp. P10008, 2008.

[4] X. Z. Chen, Q. H. Zheng, X. H. Guan, and C. G. Lin, "Quantitative hierarchical threat evaluation model for network security," *Journal of Software*, vol. 17, no. 4, pp. 885–897, 2006.

[5] E. Doynikova and I. Kotenko, "Cvss-based probabilistic risk assessment for cyber situational awareness and countermeasure selection," in *25th Euromicro International Conference on Parallel, Distributed and Network-based Processing (PDP'17)*, pp. 346–353, 2017.

[6] Y. Duan, Y. Cai, Z. Wang, and X Deng, "A novel network security risk assessment approach by combining subjective and objective weights under uncertainty," *Applied Sciences*, vol. 8, no. 3, pp. 428, 2018.

[7] M. R. Endsley, "Design and evaluation for situation awareness enhancement," *Proceedings of the Human Factors Society Annual Meeting*, vol. 32, no. 2, pp. 97–101, 1988.

[8] U. Franke and J. Brynielsson, "Cyber situational awareness–a systematic review of the literature," *Computers & Security*, vol. 46, pp. 18–31, 2014.

[9] V. Gorodetsky, O. Karsaev, and V. Samoilov, "Online update of situation assessment based on asynchronous data streams," in *Proceedings of the Knowledge Based Intelligent Information and Engineering Systems*, pp. 1136–1142, 2004.

[10] M. Henzinger, "Finding near-duplicate web pages," *Proceedings of the 29th annual International ACM SIGIR Conference on Research and Development in Information Retrieval (SIGIR'06)*, pp. 284-291, 2006.

[11] M. S. Hwang, C. T. Li, J. J. Shen, Y. P. Chu, "Challenges in e-government and security of information", *Information & Security: An International Journal*, vol. 15, no. 1, pp. 9–20, 2004.

[12] S. Islam, H. Ali, A. Habib, N. Nobi, M. Alam, and D. Hossain, "Threat minimization by design and deployment of secured networking model," *International Journal of Electronics and Information Engineering*, vol. 8, no. 2, pp. 135–144, 2018.

[13] Y. P. Jiang, C. C. Cao, X. Mei, and H. Guo, "A quantitative risk evaluation model for network security based on body temperature," *Journal of Computer Networks and Communications*, vol. 2016, pp. 3, 2016.

[14] S. J. Walker, "Big data: A revolution that will transform how we live, work, and think," *International Journal of Advertising*, vol. 33, no. 1, pp. 181–183, 2014.

[15] MIT Lincoln Lab, *2000 Darpa Intrusion Detection Scenario Specific Datasets*, 2000. (`https://www.ll.mit.edu/ideval/data/2000data.html`)

[16] C. Li and X. L. Shen, "Network security situation awareness model based on multi-period assessment," *Applied Mechanics and Materials*, vol. 411-414, pp. 613–618, 2013.

[17] C. W. Liu, W. F. Hsien, C. C. Yang, and M. S. Hwang, "A survey of public auditing for shared data storage with user revocation in cloud computing," *International Journal Network Security*, vol. 18, no. 4, pp. 650–666, 2016.

[18] Jie Liu, Ting Jin, Kejia Pan, Yi Yang, Yan Wu, and Xin Wang, "An improved knn text classification algorithm based on simhash," in *IEEE 16th Interna-*

tional Conference on Cognitive Informatics & Cognitive Computing (ICCI'17), pp. 92–95, 2017.

[19] L. Liu, Z. Cao, C. Mao, "A note on one outsourcing scheme for big data access control in cloud," *International Journal of Electronics and Information Engineering*, vol. 9, no. 1, pp. 29–35, 2018.

[20] M. E. J. Newman and M. Girvan, "Finding and evaluating community structure in networks," *Physical Review E*, vol. 69, no. 2, 2004.

[21] E. U. Opara and O. A. Soluade, "Straddling the next cyber frontier: The empirical analysis on network security, exploits, and vulnerabilities," *International Journal of Electronics and Information Engineering*, vol. 3, no. 1, pp. 10–18, 2015.

[22] Y. Qiao, X. Yun, and Y. Zhang, "Fast reused function retrieval method based on simhash and inverted index," in *IEEE Trustcom/BigDataSE/ISPA*, pp. 937–944, 2016.

[23] A. Tayal, N. Mishra and S. Sharma, "Active monitoring & postmortem forensic analysis of network threats: A survey," *International Journal of Electronics and Information Engineering*, vol. 6, no. 1, pp. 49–59, 2017.

[24] K. Vassakis, E. Petrakis, and I. Kopanakis, "Big data analytics: Applications, prospects and challenges," in *Mobile Big Data*, pp. 3–20, 2018.

[25] J. Webb, A. Ahmad, S. B. Maynard, and G. Shanks, "A situation awareness model for information security risk management," *Computers & security*, vol. 44, pp. 1–15, 2014.

[26] J. Wu, L. Ping, X. Ge, Y. Wang, and J. Fu, "Cloud storage as the infrastructure of cloud computing," in *International Conference on Intelligent Computing and Cognitive Informatics (ICICCI'10)*, pp. 380–383, 2010.

[27] H. P. Yang, H. Qiu, and K. Wang, "Network security situation evaluation method for multi-step attack," *Journal on Communications*, vol. 38, no. 1, pp. 187–198, 2017.

[28] W. Yong, L. Yifeng, and F. Dengguo, "A network security situational awareness model based on information fusion," *Journal of Computer Research and Development*, vol. 3, pp. 000, 2009.

[29] Qingyuan Zhou and Jianjian Luo, "The study on evaluation method of urban network security in the big data era," *Intelligent Automation & Soft Computing*, pp. 1–6, 2017.

Biography

Pengwen Lin graduate student. His main research direction is network security situation awareness.

Yonghong Chen prefessor. He mainly engaged in computer network and information security research, including Internet of things and security, cloud computing and security, intrusion detection, digital watermarking, big data security.