MASQUERADE DETECTION USING IMPROVED SEMI GLOBAL POSITIONING ALGORITHM

Chitra S Nair[1], Lavanya S[2], Chaithra R[3], Ayesha Siddiqua E W[4]

[1]*Assistant Professor, Department of CSE, TJIT, Karnataka, India*
[2]*Perceiving B.E, Department of CSE, TJIT, Karnataka, India*
[3]*Perceiving B.E, Department of CSE, TJIT, Karnataka, India*
[4]*Perceiving B.E, Department of CSE, TJIT, Karnataka, India*

Abstract

A masquerade attack is an attack that uses a fake identity, such as a network identity to gain unauthorized access to personal computer information. The semi-global positioning algorithm (SGA) is one of the most dynamic ways to find these type of Sybil(fake user) attacks, but it has not stretch the target and executions as expected by multi user systems. However, the problem of false positive and false negative rate have not been reduced to as expected. To improve both the effectiveness and the performances of the SGA method, we are proposing Improved SGA, ISGA approach. ISGA improves the marking systems. ISGA even allows a little change in the user command. It as well makes suitable changes in the user behavior by upgrading the signature of the user according to the current behavior. After describing about ISGA, we represent the experimental results, the result shows that the ISGA gets the hit ratio of 90 percent and above with a low false positive rate. It increases the hit ratio by about 21.9 percent. Hence, ISGA results in increasing hit ratio.

Keywords: *Masquerade Detection, Series Positioning, Security, Intrusion Detection, Attacks*

----------***----------

1. INTRODUCTION

A masquerader attacker gains access to the account of the legitimate user by stealing the users credentials or by violating the authentication service of the user. The attack can be one either by someone within the organization or by an outsider if the organization is connected to a public network. It is difficult to detect an insider masquerader at its initiation because the attacker appears to be a normal user with valid authority and privileges. An outsider tries to gain the access to the account of a legal user either by stealing the victims account ID and password or by exploiting a legal users laziness and trust, for example if a legal user leaves the terminal or session open and logged in , a co-worker may act as a masquerade attacker. Initially masquerade detection constructs profile for every user by collecting the information of the user such as time of login, session details, place of use, time utilized by CPU, commands issued, ID of the user and user IP address.

The detection of an imposture attacker depends upon the user signature, a series of commands that are collected from the legal user. This signature will be compared to the users current session. A series of commands that are produced by the legitimate user should match well with template in the users signature.

Semi Global Positioning Algorithm have been used for detecting masquerade attacks and it is the most efficient algorithm till now. But the problem with Semi Global Positioning is determining the best marking system. This paper introduces Improved Semi Global Positioning (ISGA), which enhances both accuracy and the computational performance of SGA. ISGA computes the best marking scheme for Semi Global Positioning. The main idea used in ISGA is to consider the best positioning of the session in process series to the recorded series of the active user. After the mismatch areas are found, we mark them as abnormal and several abnormal areas are a strong indication of a imposture attack.

2. RELATED WORK

We outline some of the detection methods for imposture detection. “The uniqueness approach” [1] considers that if the inputs that are entered by the user are not visible in the data set then the user is classified as the imposture. While the performance obtained from this approach is very poor.

“Naïve bayes one-step markov approach” [2] is dependent upon the “single step transition” from one command to proceeding command. It provides two conversion matrices for every user that is upbringing data set and examining data set. When the difference between these two matrices are noticed the alarm triggers. But this approach didn’t provide satisfactory false alarm rate. The drawback is that this approach does not update the user profile according to user. The main idea that has been used in “Compression Approach” [1] is to compress the new and the old data that has been entered by the same user to some ratio, while the compressed ratio of the masquerading attacker will be different.

Szymanski and Zhang [3] have proposed a technique for recursive data mining that collects frequently used template in the series of commands entered by the user and these

commands will be encoded into a new symbols and the series will be re-written with the new coding. Then by using support vector machine (SVM) [4] masqueraders can be classified. While this approach demands for the combing of user data which may not be easy to be implemented in the real world.

There are many "Series Positioning Algorithms" such as global positioning, local positioning and semi global positioning (SGA), While semi global positioning is the most accurate and efficient algorithm.SGA gives low false positive rate and high hit ratio. The advantage of this approach is that this approach can be used in any heterogeneous environment with different operating systems.

Coull and Syzmanski [5] made changes in semi global positioning to avoid the problem of false positive rate that is found in the previous approaches. Two marking systems were introduced such as "command grouping" and "binary marking system". These marking systems are used to fix the positioning score and the gap insertion penalties. The binary marking scheme compares the current signature series with new behaviors and user lexicon of the current user. This scheme provides a threshold value for every user to make sure that the signature series and the lexicon of the valid user remains protected from the masquerade attacks. This threshold value is calculated through the snapshots of the user signature.

3. ISGA APPROACH

To overcome the drawbacks that we are facing in the present system, we are proposing a new system which is called ISGA. It is completely based on Enhanced SGA [5]. The main idea that we will be using in ISGA is to compare the users present huddle series with the preceding series of the same user, and label the miss positioning areas as abnormal. We will be signaled with the masquerade attack if the percentage of the anomalous area is greater than that of the users threshold value.

ISGA even tolerates small changes in the user seriess. The command that will be entered by the user will be converted into user series and it will be decomposed into three phases. The phases are configuration phase, detection phase, update phase. Fig 1 shows the phases and modules that we are implementing.

In the configuration phase, positioning parameters will be calculated for each user, that will be used in both detecting phase and the updating phase. In detecting phase the users current session series will be aligned to the signature series. Two approaches are used to improve the computational performance of detection phase. Mainly "Top-Matching Based overlapping" and "Parallelized approach". If new template are encountered in the user current signature and the lexicon list then those template are updated to system parameters in the update phase.

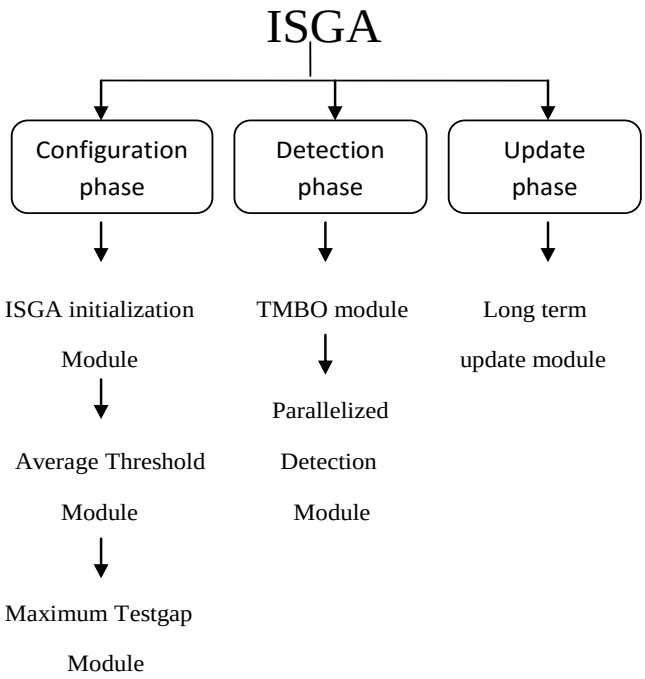

Fig 1: ISGA modules and phases

3.1 The Configuration Phase

The few parameters that will be used in ISGA are computed in this phase and they are as follows:

1. Preferred gap penalties: The signature gap penalties are paired with preferred gap penalties when a space is added into the test series and the signature series. As every user behaves in a distinct manner, ISGA identifies the distinct behavior by computing two distinct penalties governing each other.

2. Mismatch Score: ISGA calculates the mismatched score through the alteration system.

3. Average preferred threshold: The computation of distinct Threshold value for each individual user is carried out in ISGA. The threshold is computed as per to the variations in the user actions.

$$threshold = \frac{signature\ \ gap\ penalty}{no.of\ users} \quad (1)$$

4. Maximum test gap: highest number of spaces that can be put into the user test series is equal to that of the length of the series that is evaluated. A distinct boundaries is computed for every user which eventually gets updated in the update phase.

3.1.1 ISGA Initialization Module

For the configuration phase, this module provides an autonomous set of test and signature series. The user signature will be split into non-overlapped blocks of length p, which will be used as test seriess of a particular user. The seriess that are generated represent all the combinations of the user's seriess and these seriess are used by every modules in the phase of configuration that are used to compute user positioning boundaries.

3.1.2 Average Threshold Module

Average threshold is computed for every user which can be further made use in the phase of detection and which can be upgraded in the updation phase. If the positioning score is greater than the threshold in the detection phase, then he is not an masquerade attack. A masquerade attack is when the positioning score is less than the threshold. This is computed by considering all the user data. This module makes use of same test series and the signature series which are computed in the initialization module. In the detection phase, the two marking systems are compared with the output of the module.

The preferred positioning path and the number of test spaces to be put are computed by applying "Trace Backward Transition Algorithm" (TBTA). This method is a dynamic programming method that can select the preferred positioning path by tracing back through the preferred scores are evaluated by SGA. This algorithm is implemented to build a backward transaction matrix. TBTA is used to take out the final positioning path. A transition matrix is generated to align the test series to the user series. Equation 3 and 4 is used to compute positioning score and max score positioning.

$$alignment\ score = \frac{mismatc\ h\ penalty}{no.\ of\ users} \quad (2)$$

$$max - score\ alignment = \frac{matc\ h\ score}{no\ .of\ users} \quad (3)$$

Table 1: Shows how TBTA works. Let us consider two commands such as ATCG and TCG, we will align +1 if there is match, -1 if mismatch occurs and -2 in case of gap insertion.

3.1.3 Maximum Test Gap module

The signature series is decomposed into overlapped subseries of length 2n because, if the subseries of length n are matched with the signature series of length 2n then the maximum of n spaces can be inserted into test series of all the users. After tracing the SGA algorithm it is noticed that even if the length of test series is long enough, the number of spaces is all most half of length of series.

3.2 Detection Phase

In the detection phase we make use of two modules, namely "Top Match Based Overlapping Module" and "Parallelized Detection Module". We should calculate positioning parameters and marking systems of test and signature field. Equation 4, 5 and 6 shows ISGA matrices.

$$false\ positive = \left[\frac{\left(\frac{mismatc\ \ h\ penalty}{match\ score}\right)}{total\ \ no\ .of\ \ users}\right] \times 100 \quad (4)$$

$$false\ negative = \left[\frac{\left(\frac{matc\ \ h\ penalty}{mismatc\ \ \ h\ score}\right)}{total\ \ no\ .of\ \ users}\right] \times 100 \quad (5)$$

Table 1: Semi-Global Positioning Algorithm

G	-8	-10 -5 -7 -5	-7 -2 -4 -2	-4 1 1 -4
C	-6	-8 -3 -6 -3	-5 0 0 -4	-2 -2 -3 -6
T	-4	-6 -1 -1 -3	-3 -2 -2 -5	-4 -4 -4 -7
A	-2	-4 -1 -1 -4	-3 -3 -3 -6	-5 -5 -5 -8
Gap	0	-2	-4	-6
	Gap	T	C	G

3.2.1 Parallelized Detection Module

To match the different signature subseries with the user commands in the active period we are using detection algorithm, subsequently TMBO decomposes the user commands into certain overlapped consequences.

To match subseries we are executing the threads in parallel. If "detection_update_threshold" is equal to the estimated user matching value, then the thread displays a message as no attacker found. Else if "detection_update_threshold" value is less than the user matching value, then the thread displays the message as the attacker is detected. Fig 3 shows the working of the parallelized detection module.

3.2.2 Top Matching-Based Overlapping Module

This module makes use of "marking systems" and "Highest factor of test spaces" of each user to position current session template to set of overlapped subseries of the user signature. TMBO selects the subseries with the highest match that is used during the matching of user subseries and the test series.

The working of TMBO depends on two values (1) number of intermediate positioning for detection process and (2) the effect of TMBO on false alarm rates and hit ratio.

TMBO is evaluated in three steps. In first step TMBO considers the length of overlapped consequences. In second step TMBO shows the matching between every test

subseries and user subseries. In third step TMBO selects the best user series that are matched with test parameters of the other user. To identify the attackers, the last step distinguishes between the extreme values of positioning with the "detection-n-update-threshold".

3.3 Update phase

This phase is mandatory when the user is not masquerader. Since any ID gets updated automatically to the behavior of the user, the update takes place by two modules. Modules are inline update module and long term update module. The primary functions of inline update module are (1) searching areas in user signature subseries to be updated with new user action pattern. (2) Updating the lexicon of user with adding new commands.

3.3.1 Long Term Update Module

From the result of "inline update module", the long term updation module updates the user commands into the user system. To execute this module we are using three approaches, namely periodic, idle time, threshold. Corresponding to the properties and essential needs of the regular computer approach is selected.

If the rate of periodicity is stable, then the periodic approach gets executes in the translation step. When the computer is idle, the idle time approach gets executed to decreases too many commands in the system.

Since a number of test commands has been added in the middle of user commands, the threshold approach executed the translation step and reaches the threshold which is different for each user and updates constantly. This approach is extremely efficient as it executes the module when the user commands have been varied.

Figure 3 shows the activity diagram of the Improved Semi Global Positioning Approach.

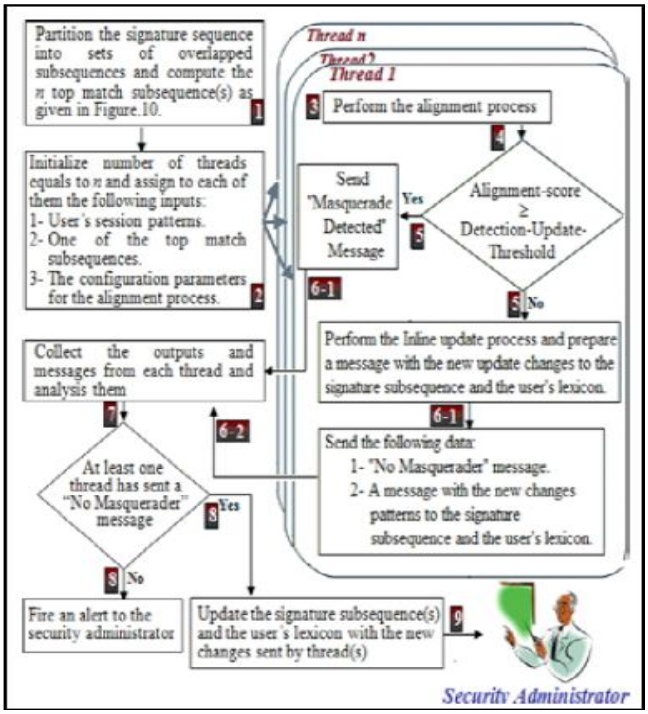

Fig 2: The processes of parallelized detection module[8]

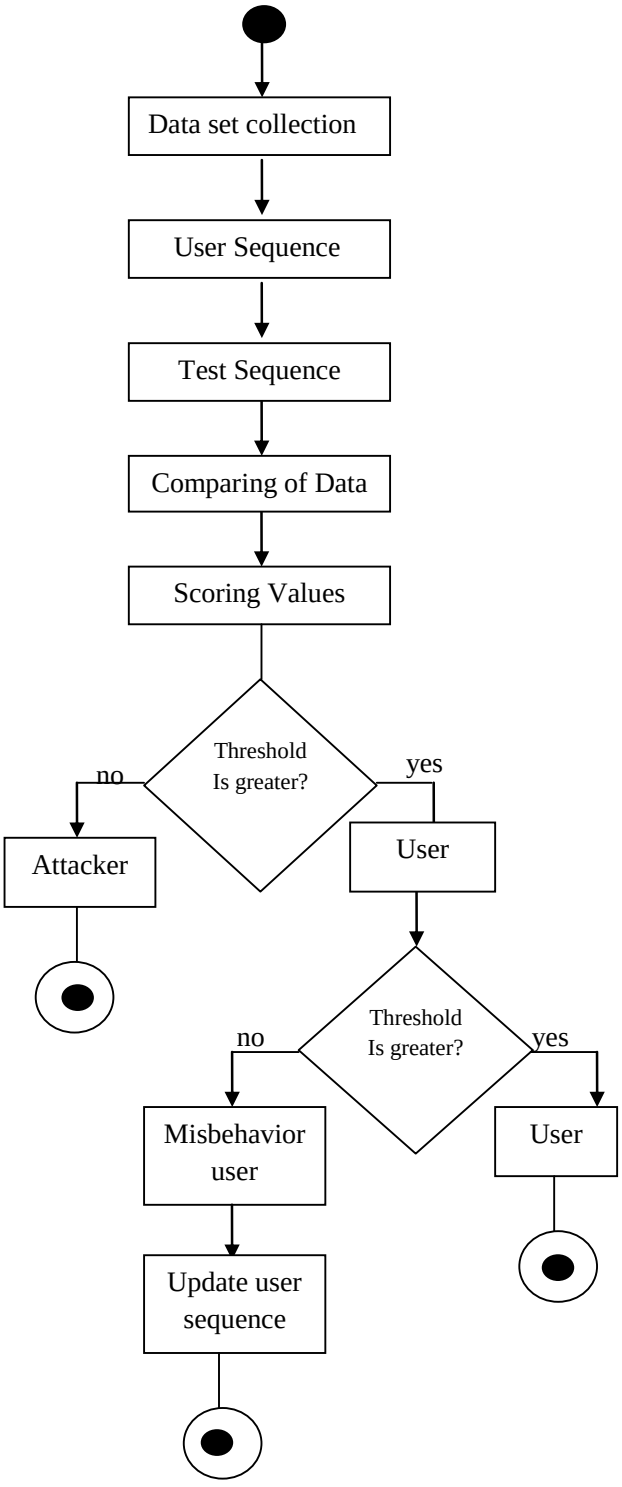

Fig 3: Activity diagram for ISGA approach

4. CONCLUSIONS

Masquerading means the attack that has been done intentionally to steal the credentials of an authenticated user. It is one of the critical attacks. SGA algorithm is used to detect these types of attacks and it is the most efficient algorithm till now. But SGA has low false positive rate and performance is not satisfactory. So to overcome the problems that we are facing in SGA we have ISGA algorithm which is an enhanced version of SGA. It provides

different parameters to different user to maintain consistency. It even provides marking systems and even tolerates small mutations in the user behavior. All these features strongly degrade false positive rates and increase hit ratio. The performance of ISGA is better when compared with SGA.

REFERENCES

[1]. M. Schonlau, W. DuMouchel, W. Ju, A. F. Karr, M. huess, and Y. Vardi, "Computer intrusion: Detecting masquerades",2001.
[2]. W. Dumouchel "Computer intrusion detection based on Bayes Factors for comparing command transition probabilities". Technical report 91, National Institute of Statistical Sciences.
[3]. B.Szymanski and Y.Zhang, "Recursive data mining for masquerade detection and author identification", in Proc.IEEE 5th Syst,2004.
[4]. B. Christoper, "A tutorial on support vector machine for pattern recognition", Data Mining Knowl,1998.
[5]. S.E.Coulla and B.K. Szymanski, "Sequence alignment for masquerade detection," J. Comput. Statist. Data Anal,2008.
[6]. K. Wang and S. J. Stolfo, "One Class training for masquerade detection," in proc.IEEE 3rd conf,2003.
[7]. S. K. Dash, K. S. Reddy and A. K. Pujari, "Episodic based masquerade detection," in Proc.1st Int. Conf. Inf.
[8]. Hisham A. Kholidy, Fabrizio Baiardi, and Salim Hariri, Member, IEEE Computer Society "Data Driven Semi Global Alignment Approach", 2015.
[9]. A. Sharma and K K Paliwal, "Detecting masquerades using a combination of Na€ıve Bayes and weighted RBF approach," J. Comput. Virology, 2007.
[10]. A. S. Sodiya, O. Folorunso, S. A. Onashoga, and P. O. Ogundeyi, "An improved semi-global alignment algorithm for masquerade detection," Int. J. Netw. Security, May 2011.