International Journal of Engineering & Technology, 7 (4.5) (2018) 24-27

International Journal of Engineering & Technology

Website: www.sciencepubco.com/index.php/IJET

Research paper

Anomaly Detection for RBAC Systems in UNIX/LINUX Environment with User Behavior

Shashidhar V [1]*, Venkatesulu D [2]

[1] *Asst Professor CSE Dept Presidency University Bengaluru, Ph D Research Scholar CSE Dept VSFTR Deemed to be University*
[2] *Professor & Head CSE Dept VSFTR Deemed to be University*
**Corresponding author E-mail: shashidhar.virupaksha@gmail.com*

Abstract

UNIX and LINUX are popular operating systems because of their free open source. By using RBAC model permissions are allocated based on user roles in UNIX/ LINUX environment. The RBAC model manages the privileges of the user within a system or application. However it is possible for a user to misuse his privileges that are assigned. Misuse of privileges cannot be identified by RBAC as it works based on rules. This paper suggests an anomaly detection technique to detect misuse of privileges. It keeps track of the user behavior and identifies anomalies in user behavior which helps in misuse of privileges.

Keywords: Anomaly Detection, RBAC, User Behavior

1. Introduction

UNIX OS has become popular because of its stability, performance and increasing. It also provides a platform for internet and intranet services for TCP/IP[1] connection.

However, the problem with UNIX operating system in enterprises is with access control which is not suitable to modern techniques. Because the access control is dependent on discretionary control [2] as well as it is not suitable to large enterprises because there is no technique to provide centrally access privilege for a user.

For these two problems RBAC is proposed. RBAC means Role Based Access Control. RBAC is the most widely used technique for security mechanisms. The main idea in RBAC model is to separate the distribution of access permission to specific system roles and the assignment of roles to users, this simplifies the access management tasks [3].

RBAC is based on role of users, who are login into the system and whether the user is accessing the system according their role or not. And also whether the login user is performing their actions or other role actions. And this can be done by finding the behavior of that particular user.

Issues like policy verification [4],[5],[6] ,[28] a problem of critical importance to system administrators have a major part in research work on RBAC mostly. Structure of access control frame works is affected by the rules which are specified in seminal work of [7].

RBAC systems presently being used for administrative purposes have similar features. They provide features that allow manipulation of RBAC policies. This is done by reassignment of users to roles and then modification of permissions for the roles [8].

The verification of policies for providing access control is specified as an security issue. Security question is explained in terms of an analysis of reachability. For example, the reachability of an user role in RBAC systems deal with the construction of the problem of determination of the target user in consideration of the sequential policy states. This kind of task is challenging, because it creates subtle and undesirable effects due to the nature of language state changes that are procedural and whose detection is hard to predict since a tool which can do analysis is not available.

2. Related Works

Hoffman [9] proposed a method which combines RBAC and DTE. This method supports the least privilege feature, an extra abstraction layer is supplied for Type Enforcement (TE). This is a necessary feature for an operating system to be secure. Hoffman's implementation approach deals with a granularity level which is coarse and does not arrive at the operation of events and there is no explicit differentiation shown between normal and privileged access mode of user. Multiple authorization framework types in health care systems is explained by Chandramouli [10].

In RBAC NFS server model, a new approach or process to NFS. This program uses a separate database which contains roles as well as permission and does not require to access files for UNIX permissions. Object permissions for read and write are done from the database. Limitations in relation to role or permissions for an object are also waived off. This solution is provided at the Linkopings Universitet in Sweden [11]. They have degraded performance for all NFS calls.

Security for Linux kernel at the level of operating system has been investigated in Gresecurity. A method for implementation of mechanisms for purpose of detection, containment and preventation model through the maintenance of different features on for stable kernels was proposed[12]. It provided mechanisms to guard against privilege escalation, memory corruption and malicious code execution. It implemented an advanced auditing system.

Copyright © 2018 Authors. This is an open access article distributed under the Creative Commons Attribution License, which permits unrestricted use, distribution, and reproduction in any medium, provided the original work is properly cited.

Further investigation for Hosting companies in order to harden web servers especially adopting Grsecurity system and providing services to locally logged users has been done in [13].

Definition of the GTRBAC model, which included notions of temporal events and constraints was proposed in [14] by Joshi et al. It also included specification of temporal contexts by introducing points in time and duration.
Alur and others [15],[16] proposed TIOA. Several researchers have proposed test generation for real time systems as in [17],[18],[19],[20].
To test generation for RBAC policies Masood et al. [21][22] have investigated a state-based approach.
Mallouli et al. [23] used integrating OrBAC (Organizational Based Access Control) rules in order to propose a model-based approach to testing access control policies.
Present RBAC Model working only on the principle of rules cannot detect such actions since RBAC model just ensures what he user can access, it cannot ensure that he is accessing them properly. More over it is difficult to create rules for each and everything because the rules would be practically a large set and the system would get complex.
Apart from this the rules would vary from situations from situation depending on the user and the role.
In this paper, these problems are handled by considering the user behavior. Behavior of user is constructed from the previous behavior. Construction of behavior is predicting or estimating the behavior of user.
Behavior of user is predicted from sequence of events or actions performed by the user per a login session. In UNIX environment sequence of actions are represented by the sequence of commands typed by the user in the login session. Based on the previous behavior of the user as well as users with the same role, malicious activity is identified.
To detect malicious user behavior is generated using trie structure [27].

3. Procedure

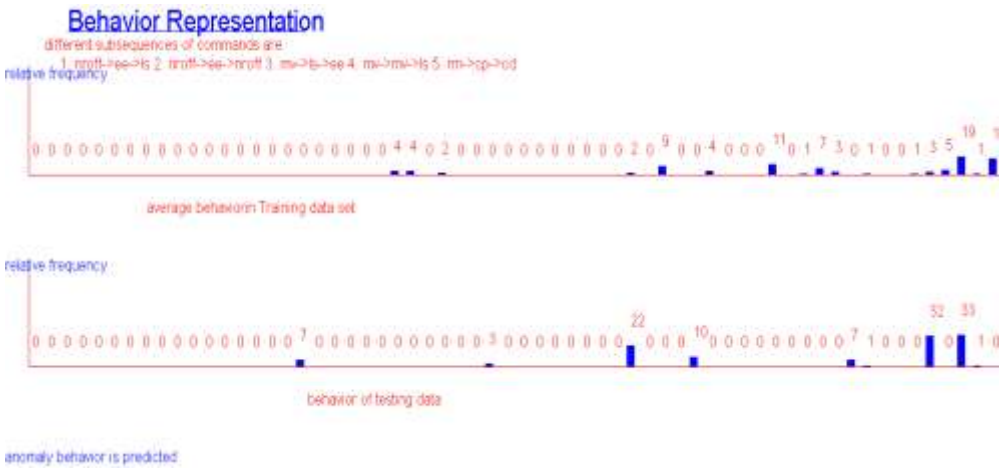
In proposed approach, behavior is generated from sequence of commands typed by the user. Initially, sequence of commands is stored in tree structure in order represent the sequence of commands in efficient and effective manner. From the generated trie structure behavior is constructed. User Profile represents the probability of particular sub sequence of commands typed by the user. Cosine distance algorithm is used to measure the similarity between profiles. Average behavior of user is predicted and next we have to find out the similarity between current behavior and average behavior. In addition to this similarity, similarity which is measured between current behavior and past behavior is added to strengthen the final output or result to predict the malicious behavior. After the construction of the trie , user profile is characterized by subsequences in trie and by traversing the trie, relevance of profile is calculated using frequency-based methods. The evaluation of the relevance of a subsequence is calculated by the support or relative frequency. In this scenario, the support [25] of a subsequence is measured as the ratio of the count of subsequence in the trie and the total count of subsequences at same level.
This type of profile is generated for each and every login session data. All these profiles are maintained. Whenever new profile is occur to classify then it is compare with already stored profiles and then finds out the support for that using Cosine distance algorithm.

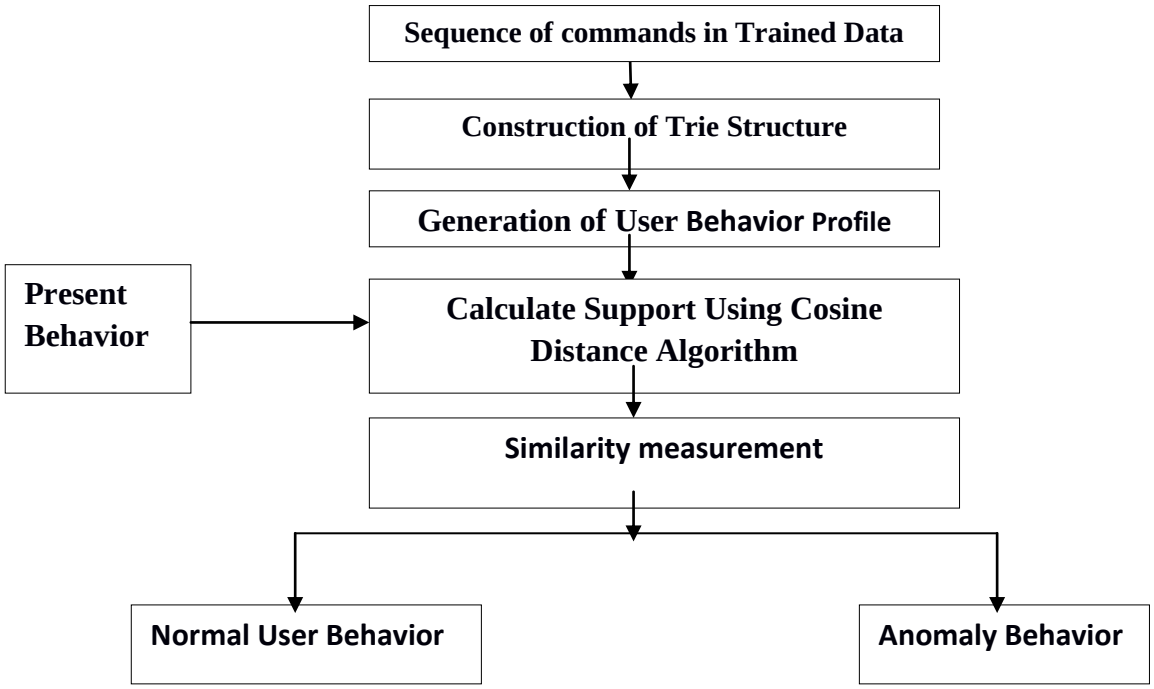
4. Result

In this paper, the data set used has been collected by Greenberg [26] in UNIX environment. The four types of users in this data set are Non Programmers, Experienced Programmers, Novice Programmers, and Computer Scientists. Novice programmers have little knowledge or no knowledge about programming or UNIX environment working nature. They mainly use the basic system functions or services.

Experienced users have some knowledge about programming and UNIX environment. They use advanced system facilities for coding, word processing etc. Computer scientists are computer engineers and professors. The nature of this type of users is easily predictable. The tasks performed are like research investigations, maintaining database etc. Non programmers Faculty or other Environment Designers are classified to this type of users.

The algorithm compares the current behavior with the normal behavior based on the cosine distance algorithm. The average behavior and the normal behavior are represented in the form of a graph. This helps to identify anomaly behavior which is predicted.





5. Conclusion

UNIX/ LINUX environments use RBAC in order to provide security. RBAC model manages the privileges of user within in a system or application and provides security by allowing access based on the rules generated. It is possible for a user misuse his privileges that are assigned. Such incidents cannot be identified by RBAC as it works based on rules. Also it is not possible to make rules because the systems would become very complex apart from that identification of such situations is difficult. In this paper an malicious behaviour detection technique is proposed based on user behavior to identify such misuse of privileges. Sequence of UNIX commands is used to represent Computer user behavior. From the sequence, is transformed relevant subsequences distribution is obtained which helps to find out the profile defining the behavior. A user profile is evolving and changing and may not be fixed, than evolving method, therefore an evolving method that ensures updated profiles is proposed. This Classifier has evolving nature in order to identify the malicious behavior in more accurate manner. Cosine distance algorithm is used to find the similarity between profiles. Anomaly is thus detected can be detected if there is a big deviation in the behavior of a user. The algorithm is implemented in Java and run on command line interface data set collected by Greenberg in UNIX environment is used. Thus it is able to predict misuse of privileges which cannot be done by RBAC systems.

Acknowledgement

I would like to thank Bhuvaneswari A, Nagalakshmi K for their help in coding, getting the results and in writing the paper.

References

[1] V. Cerf and R. Kahn. A Protocol for Packet Network Interconnection. IEEE Transactions on Communications, 22:637648, May 1974.
[2] D. Denning. Cryptography and Data Security. Addison-Wesley Publishing Company, January 1983.
[3] Gran: model checking grsecurity RBAC policies Michele Bugliesi Stefano Calzavara Riccardo Focardi Marco Squarcina DAIS, Università Ca' Foscari Venezia, Italy.
[4] A. Sasturkar, P. Yang, S. D. Stoller, and C. R. Ramakrishnan, "Policy analysis for administrative role based access control," in *CSFW*. IEEE Computer Society, 2006, pp. 124–138.
[5] A. Armando and S. Ranise, "Automated symbolic analysis of arbac-policies," in *STM*, ser. Lecture Notes in Computer Science, J. Cuéllar, J. Lopez, G. Barthe, and A. Pretschner, Eds., vol. 6710. Springer, 2010, pp. 17–34.
[6] M. I. Gofman, R. Luo, A. C. Solomon, Y. Zhang, P. Yang, and S. D. Stoller, "Rbac-pat: A policy analysis tool for role based access control," in *TACAS*, ser. Lecture Notes in Computer Science, S. Kowalewski and A. Philippou, Eds., vol. 5505. Springer, 2009, pp. 46–49.
[7] M. A. Harrison, W. L. Ruzzo, and J. D. Ullman, "Protection in operating systems, " *Commun. ACM*, vol. 19, no. 8, pp. 461–471, 1976.
[8] R. S. Sandhu, V. Bhamidipati, and Q. Munawer, "The arbac97 model for role-based administration of roles," *ACM Trans. Inf.Syst. Secur.*, vol. 2, no. 1, pp. 105–135, 1999.
[9] Hoffman J., Implementing RBAC on a type enforced system, In Proceedings of 13th Annual Computer Security Applications Conference, 1997, pp. 158~163.
[10] Chandramouli R. A Framework for Multiple Authorization Types in a Healthcare Application System. In Proceedings of the 17th Annual Computer Security Application Conference, 2001, pp. 137~148.
[11] M. Gustafsson, B. Deligny, and N. Shahmehri. Using NFS to Implement Role-Based Access Control. In 6th Workshops on Enabling Technologies: Infrastructure for Collaborative Enterprises; Second Internation Workshop on Enterprise Security, MIT, Cambridge, USA, June 18-20 1997.
[12] B. Spengler, "Increasing performance and granularity in role based access control systems," 2004. [Online]. ble :http://grsecurity.net/researchpaper.pdf "Sponsor page of grsecurity." [Online]. Available:http://grsecurity.net/sponsors.php
[13] R. Alur and D.L. Dill, "A Theory of Timed Automata," Theoretical Computer Science, vol. 126, no. 2, pp. 183-235, 1994.
[14] R. Alur and D.L. Dill, "A Theory of Timed Automata," TheoreticalComputer Science, vol. 126, no. 2, pp. 183-235, 1994.
[15] D. Kalles and T. Morris, "Efficient Incremental Induction of Decision Trees," Machine Learning,vol. 24, no. 3, pp. 231 242, 1996.
[16] P.E. Utgoff, "Id5: An Incremental Id3,"Proc. Int'l Conf. Machine Learning,pp. 107 120, 1988.
[17] G.A.Carpenter, S. Grossberg, N. Markuzon, J.H.Reynolds, and D.B. Rosen, "Fuzzy Artmap: A Neural Network Architecture for Incremental Supervised Learning of Analog Multidimensional Maps," IEEE Trans. Neural Networks, vol. 3, no. 5, pp. 698 13, Sept.1992.

[18] G.A. Kaminka, M. Fidanboylu, A. Chang, and M.M. Veloso,"Learning the Sequential Coordinated Behavior of Teams from Observations,"Proc. RoboCup Symp.,pp. 111 125, 2002.
[19] P. Angelov and D. Filev, "An Approach to Online Identification of Takagi Sugeno Fuzzy Models," IEEE Trans. Systems, Man, andCybernetics, Part B,vol. 34, no. 1, pp. 484 498, Feb. 2004.
[20] A. Masood, R. Bhatti, A. Ghafoor, A. Mathur. "Scalable and Effective Test Generation for Role-based Access Control Systems," IEEE Trans. on Software Engineering, vol. 35, no. 5, pp. 654-668, 2009.
[21] A. Masood, A. Ghafoor, A., Mathur. "Conformance Testing of Temporal Role-based Access Control Systems," IEEE Trans. On Dependable and Secure Computing, vol. 7, no. 2, pp. 144-158, 2010.
[22] W. Mallouli, J.M. Orset, A. Cavalli, N. Cuppens, F.A. Cuppens, "A Formal Approach for Testing Security Rules," In Proc. of SACMAT'07, pp.127-132, 2007.
[23] J.A. Iglesias, A. Ledezma, and A. Sanchis, "A Comparing Method of Two Team Behaviours in the Simulation Coach Competition, " Proc. Int'l Conf. Modeling Decisions for Artificial Intelligence (MDAI), pp. 117 128, 2006.
[24] R. Agrawal and R. Srikant, "Mining Sequential Patterns," Proc.Int'l Conf. Data Eng.,pp. 3 14, 1995.
[25] S. Greenberg, "Using Unix: Collected Traces of 168 Users, "master's thesis, Dept. of Computer Science, Univ. of Calgary, Alberta, Canada, 1988.
[26] Jose Antonio Iglesias, Plamen Angelov "Creating Evolving user behavior profile Automatically", 2012.
[27] Jun Luo and Hongjun Wang and Xun Gong and Tianrui Li, A Novel Role-based Access Control Model in Cloud Environments, International Journal of Computational Intelligence Systems Vol 9,Iss 1 Pag 1-9,2016 Taylor and Fransis