Available online at www.elixirpublishers.com (Elixir International Journal)

Computer Science and Engineering

Elixir Comp. Sci. & Engg. 58 (2013) 15043-15047

ISSN: 2229-712X

An Overview on Different Techniques used in Intrusion Detection

Ashish Kumar[1,*], Deepesh Rawat[2] and Ankit Aswal[2]

[1]Department CSE BTKIT, Dwarahat, Uttarakhand, India.

[2]Department of ECE BTKIT, Dwarahat, Uttarakhand, India.

ARTICLE INFO

Article history:

Received: 11 March 2013;

Received in revised form:

15 May 2013;

Accepted: 21 May 2013;

Keywords

Intrusion,

Network,

Accuracy,

Performance.

ABSTRACT

In today's modern and digital world countless milestones have been achieved by the human. Technology has completely enveloped us in some way or the other. Hence if there is not complete dependency but most it is on the use of technology. For example, communicating with someone using a device is technology. Today, there are endless organizations that are maneuvering in recent advancements in technology. Among these the one which concerns us is to secure information or data on the network. Network security is the most functional requirement of any system. So the question arises that how to secure the data on the network from the intruder. There are innumerable intrusion detection techniques to detect the intrusion in the system. This paper deals with some handful of the techniques with which we can secure our data on the network.

Introduction

Intrusions are the activities that contravene with the security protocols of the system network in information system. Any operation that follows in compromising the attainability, integrity, or clandestinity of information is termed as intrusion. The process used to identify intrusions in a network is called intrusion detection. Intrusion detection is being studied for many years and rests on the fact that the behavior of an intruder will be certainly disparate from that of a standard user and that many unauthorized operations will be recognizable. Hence for several accounts intrusion detection is mandatory for the entire security system. Originally, several orthodox systems and applications were developed that lacked security in system. In several other approaches, systems and applications were materialized in a manner so as to work in a different environment. Paramount focus of intrusion detection is in the domain of security of computer systems and networks.

The intrusion detection had been defined as one of the six substantial elements by Halme and Bauer [5] in their classification of anti-intrusion techniques.

The initial three components which they identified are deterrence, pre-emption, and prevention, which are fundamentally based on passive measures which depreciate the probability of a conclusive attack on a system network. These elements ascertain the protocol related concerns of network information security and those elements which can be assimilated into a system with less effort. The last three elements, which rely on more active measures, are deflection, detection, and countermeasures. These are devised to assure the protection of the vital elements of a network. Out of the six elements the most essential is the proper detection of an intrusion in a network. Denning and Neumann [14] recommended the requirement for efficient intrusion detection mechanisms as a constituent of security mechanism for computer systems.

For employing intrusion detection they analyzed four justifications within a secure computing framework:

1. There are numerous existing systems with security malfunction which allow intruders to attack, but cannot recognize and remove as a result of several technical and economic reasons.
2. It is very complex to replace existing systems with security malfunctions by more secure systems due to economic and application considerations.
3. Perfect secure systems are probably unrealizable.
4. Even extremely secure systems are vulnerable to misuse by authentic users.

To recognize intrusion and security threats a wider knowledge of network attacks is mandatory. These network attacks can be hypothesized into a five step approach [4]:

1. Reconnaissance: The intruder compiles high degree information about the network system.
2. Scanning: By employing the information collected in the last step, the intruder identifies feasible chance of attacks in the system and hence collects complete information regarding the network system.
3. Gaining Access: Network system consists of two methods to achieve access relying on the authorization of the user. A legitimate user exploits the loop holes in the operating system whereas an illegitimate user makes uses the network to connect to the system.
4. Maintaining Access: The intruder after accessing the system tries to excerpt information out of the system and also tries to control the network.
5. Covering Tracks: The intruder accords system logs and other appropriate information to control the system completely and to assure that there remains no evidence of contravention in the security system.

Sommer exemplifies the following tasks that can be accomplished by IDS [15]:

- Prevent a damage dynamically that detected intrusions could cause.

Tele:

E-mail addresses: kumarash1986@gmail.com

- Mitigate a damage dynamically that detected intrusions could cause.
- To find new attack patterns.
- To find an activity that could be a precursor of a more serious attack.
- To identify an attack perpetrator.

IDS must follow certain requirements in order to accomplish its tasks. On another hand, these requirements may be considered as IDS efficiency evaluation criteria. A systematic overview of these requirements is given as follows [16]:

- **Accuracy:** A legitimate activity must not be identified by IDS in a system environment as an aberration or a misuse.
- **Performance:** IDS performance must be commendable enough so as to identify real-time intrusion detection.
- **Completeness:** IDS should not fail to identify an intrusion. But we should admit the fact that fulfilling this requirement is rather difficult since to possess a thorough knowledge about past, present, and future intrusions is almost impractical.
- **Fault tolerance:** IDS must be able to resist intrusions and their consequences.

The following figure 1 has shown the taxonomy of intrusion detection system.

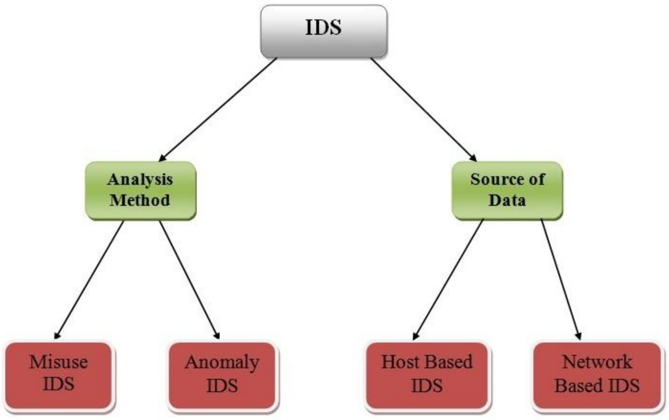


Figure 1: Taxonomy of IDS

2. Analysis method used to identify intrusion

2.1 Anomaly Intrusion Detection System

Techniques based on Anomaly Intrusion Detection System follow an approach which rely on models, or profiles of the usual behavior of users [1][7], applications [3][24] and network traffic [6] [8] [9] and is parallel to misuse detection. Those approaches which deviate from the established models are depicted as intrusions.

Modeling of Anomaly IDS is done by analyzing the behavior of the system over a span of time to create activity profiles which characterize standard use of the system. For identifying intrusions the Anomaly IDS enumerates the analogy of the traffic present in the system with the profiles. Every user of a computer system has certain functionality within the system and is capable of performing some tasks.

Any divergence from the above model is considered as anomalous. Anomaly detection systems are trained on enormous amounts of system audit data involving different intelligent techniques like machine learning, rules generation, neural networks, etc in order gain adequate knowledge regarding the user behavior.

Anomaly detection systems consist of a collection of models for evaluating various characteristics of an event. These models acknowledge an anomaly count or a probability value which reverberate the ‘normality’ of this event in accordance with their current profiles. However, the system is bestowed with the task of assembling the various model outputs into a single and exclusive result. But the complication rests on the fact that this assembling is difficult to implement especially when the outputs of individual models mismatch considerably. Recognition of new intruders by the system is the biggest pro of this model.

This model is conned by the following summarized facts based on [2]:

a. Non availability of a defined method or a model to choose the threshold value against which the profile is correlated.

b. Continuous comparison and updating of the profiles make it computationally expensive.

c. The model must furnish a provision of revising and updating due to time varying user behaviors.

2.2 Misuse detection

Misuse detection functions using priori prepared patterns, known as signatures, of known intruders and uses pattern matching on audit information to identify intrusions. Most of the organizations make use of Misuse IDS design methodology in developing anti-virus solutions. The system design is based on the signature of all-known intruders. Rules and signatures describe anomalous and risky activity. Simple creation of intruder signature databases, swift and easier implementation of IDS and nominal utilization of the system resources is the prime advantage of this system [15]. The main weakness of this system is the use of standard and established rule based techniques since these techniques depend greatly on the audit results. This one-to-one conformity between rules and audit records calls for the reason for the system to be inflexible.

While anomaly detection generally makes use of threshold supervising to denote when a certain authorized metric has been attained, misuse detection techniques frequently makes use of a rule-based method. When these rules are applied to misuse detection they become platform for network intruders [15]. The intrusion detection process detects a potential intrusion as soon as user's actions are found to be uniform with the authorized rules. The application of extensive rules plays a very crucial role in the use of expert systems for intrusion detection. Similar to anomaly detection techniques, potential performance degradation greatly affects misuse detection systems resulting solely from a dependency on audit trails for input. But reduced audit record can mitigate the above disadvantage since for known intrusions it sets present misuse detection systems to act superior than anomaly detection systems and also improve the system performance. This means patterns of known intrusions are identified more accurately by misuse detection systems with lesser false alarms being generated at the same time. This better performance results from the fact that misuse detection systems take advantage of clear knowledge of the intrusions. The major drawback of misuse detection is that it fails to identify novel or unknown intrusions. Hence the computer systems using misuse detection systems usually are at the risk of being comprised without identifying the intrusions. Besides these, misuse detection also requires to understand the nature of the intrusions due to the need of having the explicit representation of the same [15].

3. Intrusion Detection Methods:

3.1 State Transition Analysis:

State Transition analysis was created recently by Santa Barbara, a credible software group at University of California [10]. Generally, it finds its application in representing a chain of operations that an intruder executes to attack the system. Intrusions possess following two properties [17].

An intruder accesses a target system in some way or the other. Intruder gains some authorities resulting from the intrusion that were not possessed by it before. This method represents the intrusion in the form of state. An intruder uses the primary state to initiate the penetration process or it can be said that the initial state is identified prior to intrusion by the system. A state is achieved when the intruder succeeds in the system which is known as compromising state. This works in the accomplishment of the penetration process. After the initial state and prior to compromising state numerous intermediate states are allocated and numerous transactions are being done which are termed as state transactions.

Apart from the state, there are some actions known as signature actions which lead to some of these. Signature action signifies a nominal set of operations required to execute the intrusion. In the absence of these operations execution of penetration process fails or is unsuccessful.

Finally, graphical representations of state transition and signature make state transitions diagrams. A worthy characteristic of this approach is that a threat scenario is being represented using this diagram in a very easy to understand visual form. This is a good feature of the approach [17].

A demonstration of this approach is given by the following example. Suppose that accounts of some students are maintained in an office application server at University of Delhi. This method lets the remote users to access the server and execute some instructions without any authorization. This may lead in giving an intruder a system command prompt under super-user privileges. This intrusion constitutes the steps as follows [17]:

- Connecting to TCP port 255.
- Executing a "FormSet" command. This brings the server to the factory settings and erases the super-user password just after the next time the system rebooted.
- System reboot: This may be accomplished by some different intrusions, SYN flood intrusion being an example.
- Connecting TCP port 255 again.
- Executing "Z" command. This provides a system login prompt and calls for a super-user password. As it was extricated in the second step, it can be deleted and the intruder gets a system command prompt.

The following state transition diagram represents the above intrusion scenario:

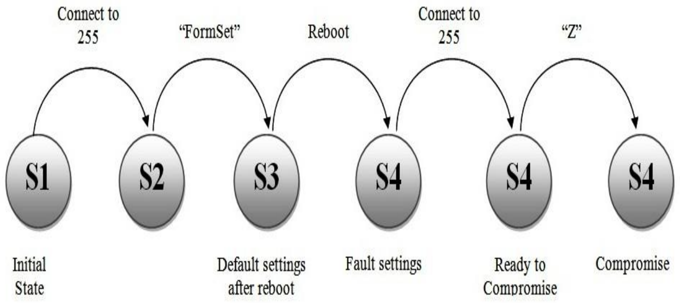


Figure 2: Example of State Transition Analysis

State has been applied for misuse detection in UNIX system distributed system and networks USTAT is the first prototype of state, which aimed at misuse detection in UNIX system. Later USTAT was proposed to process audit data which is collected on various UNIX hosts. The resulting system is called NSTAT. Another system named Net STAT resulted from a later approach of STAT to network based misuse detection [20].

3.2 Rule Based expert system:

This system is the most comprehensively used approach to misuse detection. The most extraordinary aspect of this approach is that it finds its use in both anomaly and misuse detection. These systems separates declarative knowledge related to intrusion from an inference engine performing reasoning regarding the fact base. Hence three important elements can be distinguished [17]:

- *Facts base* consisting facts on system states.
- *Rules base* consisting rules that represent intrusions scenarios.
- *Inference (deduction) eng*ine that builds reasoning on the basis of facts and rules for identifying an intrusion.

Inference engine explores the facts space for those that correspond to what is expected by the rule. The rule is actuated and its consequent is dismissed as soon as any match is found. P-BEST, a rule-based misuse detection expert system toolset, finds its application in numerous intrusion detection environments [17].

Expert systems are also applied in anomaly detection. This approach intends a bit of knowledge of usual user conduct and anomalies contained in it. This is what truly the primary difference in employing the rule-based expert systems for anomaly and misuse detection. The rules are instituted using different techniques in the first case. In the latter case, the rules are provided to the system in priori. There are various methods which are used to acquire rules that describe the user behavior. *Data Mining* is one of the following known methods [11]. This approach excerpts explanatory models from gigantic reserves of data. Basically it uses three most substantial algorithms which are discussed below [17]:

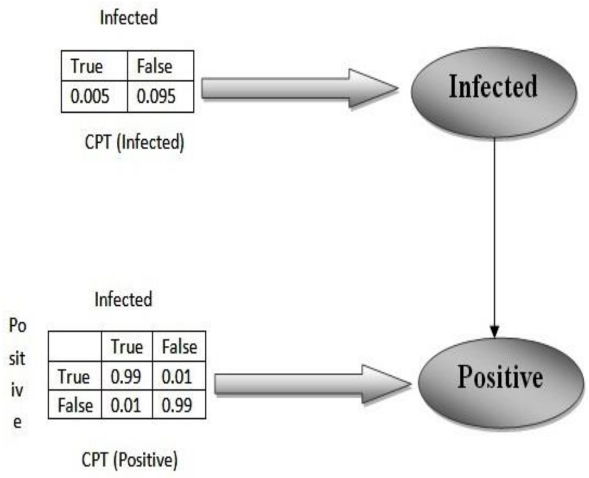
- *Classification:* mapping data elements into some preordained categories.
- *Link analysis:* determining a correlation among various elements in the audit information.
- *Sequence analysis:* modeling of sequential patterns – audit events that occur successively.

3.3 Bayesian Network:

A Bayesian network is a graphical representation of the function of joint probability distribution over a group of elements. The network structure is represented by a Directed Acyclic Graph (DAG) where every individual node corresponds to a random variable and every individual edge signifies a dependent correlation among the connected variables. Each variable (node) in a Bayesian Network is correlated to a Conditional Probability Table (CPT), which computes the conditional probabilities for the given variable and gives every possible combination of its parents' magnitudes [12]. Hence, in a Bayesian Network, the DAG grabs causal analogies among random variables, and the CPT estimates these analogies. A Bayesian Network has been used as our inference model since in a Bayesian Network nodes can represent individual events in an intrusion and edges can be modeled to represent the causal relations between events. A Bayesian Network model is efficient in determining the causal relationships and anticipating the consequences of an intervention in the problem domain from an existing dataset. Therefore a Bayesian Network is an ideal representation model for assembling prior knowledge with new data and interpreting posterior knowledge [18].

A Bayesian Network is used in modeling a complex domain [19, 13]. It is a directed acyclic graph (DAG) where each node, representing a discrete random variable of interest, consists of the states of the random variable and a conditional probability table (CPT) which consists of the probabilities of the node being in a particular state given the states of its parent. The direction of causality between the corresponding variables is indicated in a Bayesian network by the parent-child correlation between the nodes. That is, the variable which the child node represents causally depends on the ones that its parents represent [19].

Let us consider the following example. Here a farmer has a bottle of milk. It can be either clean or infected. He can perform a test which can conclude whether the milk is infected or not (i.e., the result of the test will be either positive or negative) with a high probability. Two random Boolean variables, infected and positive, can represent the situation. When the milk is actually infected the variable "infected" is true otherwise it is false. When the test claims that the milk is infected the variable "positive" is true and is false when the result of the test is negative. Note that there is a possibility that the milk is clean when the test is concluded with a positive result and vice versa.



True	False
0.005	0.095

	True	False
True	0.99	0.01
False	0.01	0.99

Figure 3: Bayesian Network and CPTs

Figure 3 shows a possible Bayesian network that models the above mentioned situation. Two nodes in the network represent the two random variables. It is assumed that the farmer is aware of the CPT for the "positive" variable, that is, the probabilities of the positive result provided that the milk is infected or clean. He also knows the CPT for the variable "infected", which represents the probability of the bottle containing infected milk. Causal relationship between the respective variables is indicated by the arrow directed from the infected to the positive node. In this case, it is expected that the outcome of the test depends on the initial state of the milk (infected or clean). Other variables do not influence the variables without parents directly [19].

3.4 Colored Petri Automata:

Kumar and Spafford (1994) and Kumar (1995) examined the misuse detection as a pattern-matching method. What they advised was an abstract hierarchy for classifying intrusion signatures (i.e., attack patterns) that was based on the structural correlations among the events that constitute the signature. Such a hierarchy has high-level events which are defined on the basis of low-level audit trail events and are used to instantiate the hierarchy into a concrete one. The prime advantage of this technique remains in clarifying the complexity of identifying the signatures in every step of the hierarchy. In addition to this it also identifies the requirements that patterns must meet in all categories of the classification in representing the full range of regular intrusions that happen (i.e., the specification of context, activities, and invariants in intrusion schemes). Kumar and Spafford adopted colored Petri nets, known as *colored Petri automata* (CPA), to signify attack signatures, with guards to signify signature perspective and nodes to represent system states. A CPA signifies the system states transition along routes that converge to intruded states. A CPA is also allied with pre- and post circumstances that must be fulfilled prior and later the match, as well as invariants that must be fulfilled while the pattern is being matched. A prototype misuse detection system also implemented CPA called Intrusion Detection In Our Time (IDIOT).

4. Conclusion

Intrusion detection continues to exist as an active research field. Even after two decades of research, the intrusion detection community is still facing various difficult problems. To identify unknown patterns of intrusions without the generation of too many false alerts still remains an unresolved task, although recently, various results have concluded that there is a feasible resolution to above problem. Another difficulty which follows is evaluating and standardization of IDSs which, once resolved, may help organizational decision makers and end users in bestowing the effective supervision. The execution and the usability of IDSs will be enhanced by renovating attack scenarios from intrusion warnings and integration of IDSs. These problems are being addressed actively by many researchers and practitioners. Intrusion detection is expected to become a practical and effective solution in securing information systems.

References:

[1] D. Denning, An Intrusion Detection Model, IEEE Transactions on Software Engineering, 13(2):222–232, Feb. 1987.

[2] Aurobindo Sundaram , An Introduction to Intrusion Detection, Crossroads, Volume 2, Issue 4, Pages: 3 – 7, 1996

[3] S. Forrest, A Sense of Self for UNIX Processes, In Proceedings of the IEEE Symposium on Security and Privacy, pages 120–128, Oakland, CA, May 1996.

[4] Khaled Labib, Computer security and intrusion detection, Crossroads, Volume 11, Issue 1, August 2004 Pages: 2 – 2.

[5] Halme, L.R. & Bauer, R.K., AINT Misbehaving: A Taxonomy of Anti-Intrusion Techniques, Proceedings of the 18th National Information Systems Security Conference, Baltimore, MD.

[6] C. Kruegel, T. Toth, and E. Kirda, Service Specific Anomaly Detection for Network Intrusion Detection, In Symposium on Applied Computing (SAC), ACM Scientific Press, March 2002.

[7] H. S. Javitz and A. Valdes. The SRI IDES Statistical Anomaly Detector. In Proceedings of the IEEE Symposium on Security and Privacy, May 1991.

[8] P. Porras and P. Neumann. EMERALD: Event Monitoring Enabling Responses to Anomalous Live Disturbances. In Proceedings of the 1997 National Information Systems Security Conference, October 1997.

[9] P. A. Porras and A. Valdes. Live traffic analysis of TCP/IP gateways. In Proceedings of the 1998 ISOC Symposium on Network and Distributed System Security (NDSS'98), San Diego, CA, 1998.

[10] K. Ilgun, R. Kemmerer, P. Porras, "State transition analysis: a rule-based intrusion detection approach", IEEE Transactions on Software Engineering, Vol. 21, No. 3, March, 1995.

[11] W. Lee, S. Stolfo, "Data mining approaches for intrusion detection", In Proceedings of the 7th USENIX Security Symposium (SECURITY-98), January, 1998

[12] F. Jensen, Bayesian Networks and Decision Graphs, Springer, New York, USA, 2001.

[13] J. Pearl. Probabilistic Reasoning in Intelligent Systems: Networks of Plausible Inference. Morgan Kaufmann, 1997.

[14] Neumann, P.G, Audit Trail Analysis and Usage Collection and Processing. Technical Report Project 5910, SRI International.

[15] P. Sommer, "Intrusion detection systems as evidence", Computer Networks, No. 31, 1999

[16] H. Debar, M. Dacier, A. Wespi, "Towards a taxonomy of intrusion-detection systems",Computer Networks, No. 31, 1999.

[17] C. Warrender, S. Forrest, and B. Pearlmutter. Detecting intrusions using system calls: Alternative data models. In IEEE Symposium on Security and Privacy, pages 133–145, 1999.

[18] Network Intrusion Detection Based on Bayesian Networks,Alma Cemerlic, Li Yang, Joseph M. Kizza, Department of Computer Science and Engineering, University of Tennessee at Chattanooga, Chattanooga, TN 37403.

[19] Bayesian Event Classification for Intrusion Detection, Christopher Kruegel, Darren Mutz, William Robertson, Fredrik Valeur, Reliable Software Group, University of California, Santa Barbara.

[20] An Overview of Intrusion Detection Systems, Sriram Sundar Rajan, Vijaya Krishna Cherukuri, Masters in Software Engineering, Malardalen University.

[21] http://www.forum-intrusion.com/archive/Intrusion 22/08/2003.