Behavioral approach for intrusion detection Application field: "'E-Health"'

Taha Ait Tchakoucht || Mostafa Ezziyyani
Laboratory IBDD-TC
Computer Science Department
University of Abdelmalek Essâadi
Faculty of Sciences and Techniques, Tangier
taha.ait@gmail.com || ezziyyani@gmail.com

Abstract

In this article, we present a new method for intrusion detection based on behavioral approach.it helps demonstrate the effectiveness of data-mining techniques using the k-means algorithm. This approach involves learning user behavior with respect to the system and to build user profils.A new user behavior is considered abnormal when it deviates from its profil.When detected,this anomaly can prove to be an intrusion. Our experimental results will be applied to the hospital information system (HIS)

1 Introduction

Any violation of the security policy of an information system is considered as an intrusion.[Als13, BG, Ca02]. An intrusion detection system (IDS) corresponds to a set of tools, made to identify any activity that contradicts the security policy of an information system, and thus could affect the efficiency and continuity of service. In general, there are two major families of intrusion detection ; Anomaly detection and Misuse detection. **Anomaly detection :** This approach is based on the behavior of the user and / or application, it is called user profile or behavior of an application. It was proposed by Anderson in 1980 and taken over by Denning [E.D86] in 1987. Anderson proposed to describe the user profile by a set of relevant measures modeling its behavior to detect subsequent deviation from the usual behavior already learned.This approach then seeks to answer the question :"Is the current behavior of the user and / or application coherent with its past behavior ?" (For more details on the various tools of the behavioral approach, see [Ja93, DN85, Da92, Sa91, Sma88, VL89]).

Misuse Detection : This approach seeks to find known attacks in the audit file. It therefore requires prior knowledge of well-defined attacks. This approach then seeks to answer the question : "Does the current behavior of the user and / or application contain an known attack ? ".In this case,a construction of a database of attacks or attack scenarios is necessary. (see [HCM+92, KS94, PK92a, Ilg93a, Me98] for more details). An intrusion detection hybrid which combines both techniques in the same time can be added as a third approach of the IDS.(see [And80, PK92b, Ilg93b, SAA04, SAA03, NM07, Jol02a, Jol02b] for more details)

Often, these two approaches are based on Data-mining techniques, discussed in the next section

2 Data-mining Techniques

This is a set of techniques for the extraction of motifs from large data sets, combining the of statistical and machine learning methods with database management.Those techniques involve learning association rules, cluster analysis, classification and regression. Applications include clients data mining to determine the segments that most likely respond to an offer , the mining of human resources data to identify the characteristics of

employees who are more successful, or analysis of the market basket for modeling the purchasing behavior of clients.Working summary, the Data-mining allows :

- Classification and prediction
- Estimation
- The research of association and the research of sequence
- Cluster analysis
- Regression

These tasks are performed through :

- Neural Networks
- Bayesian networks
- Decision trees
- Genetic Algorithms
- K-means Algorithme
- KNN Algorithm,...

- **Presentation of K-means :** The K-means algorithm is an algorithm for finding the classes in the data. This is an "non-hierarchical" algorithm :The classes it builds never maintain hierarchical relationships : a class is never included in another class. This is a widely used algorithm.

3 Related work

In this section we will present a series of works in the same context.

- **Eigen Profiles for intrusion detection :** [BG],proposes an intrusion detection method belonging to behavioral family, based on principal component analysis (PCA). This method is divided into two principal parts.The first is a learning step in which we are auditing the different behaviors of users in the network, and then we calculate the distribution of behaviors (which is to compute the eigenvectors of all user profiles, and extract the features profiles) .Then we determine for each user class the reference feature profile and the threshold.The second part(detection)consists of projecting a user profile, newly audited, on the space of eigenprofiles(eigenvectors of all user profiles) and comparing its feature profile with those who are known to find a possible deviation from his former behavior.
- **Detection of anomalies from user profiles generated from system logs :** [CMC11] proposes a method that aims to model an anomaly detection system using users profiles generated from system log files.This approach consists,of collecting data using a VB script running daily.Then only events recorded as direct results of user actions, such as authentication, logging off, starting and closing applications, will be used for more investigations.for more processing (preparation of user profiles and alert generation), data is stored in a relationnal database.Then users profiles are generated to identify the usual behaviors of all users.When an abnormal event occurs, an alert is generated,which is going to be checked to ensure Whether it is a real threat or not.
- **An intrusion-detection model :** [E.D86] introduces a real time intrusion detection model based on an expert system.This model aims to detect violations of the internal security of a company ; infiltration attempts by external users, or penetrations and internal users abuses .This method offers different types of events to be considered for users profiles construction such as :
  - Activities during user session (authentication,signing out) :for example LoginFrequency that calculate daily,the user's authentification frequency.
  - Programs executed by users :for example ExecutionFrequency which is the number of times a program is executed during a certain period of time.
  - Files accessible by users : For example ReadFrequency that defines the access frequency to a file during a period of time.

  In the following sections,we present a new intrusion detection method that relies on k-means algorithm.

4 Stages of the approach

According to [E.D86], the exploitation of the vulnerability of a system involves abnormal use of the system ; therefore, the violation of security can be detected from the abnormal behavior of the system.Thus,the user profile can be identified through a set of information such as :

- the number of incorrect passwords entered during a time interval.

-The type of used applications or editors

-Directories and files accessed by the user

-the number of visited web pages

-the number of open files during a period of time,...

We have proposed a set of variables (counters) evaluating daily activities associated with a user session.

- ***LoginFlow*** :calculates the number of times a user is connected to the system.
- ***LoginFails*** :countes the number of incorrect passwords.
- ***SessionDuration*** :measures the time elapsed during a session.
- ***SessionCPU*** :CPU time during a session.
- ***FormatCounter*** :Represents the frequency of use of a data format (extension) knowing that the data are grouped according to the extension(.doc, .txt.csv ...). It is the sum of the number of reads and writes to each file in a given format. An anomaly may indicate the existence of a Masquerade
- ***AccessFails*** :Specifies the number of attempts to manipulate (read, write, create and delete) unauthorized data (violation of permissions and privileges).A anomaly may represent an attempted penetration by a legitimate user
- ***DataVolume*** :Amount of requested data (application + files) daily.
- ***QuotaOverloadFails*** :epresents the number of failed systems due to exceeding the quota on the space dedicated to the user
- ***NbrWebpage*** :Number of web pages visited during a day.

The following figure shows the different stages of our system :

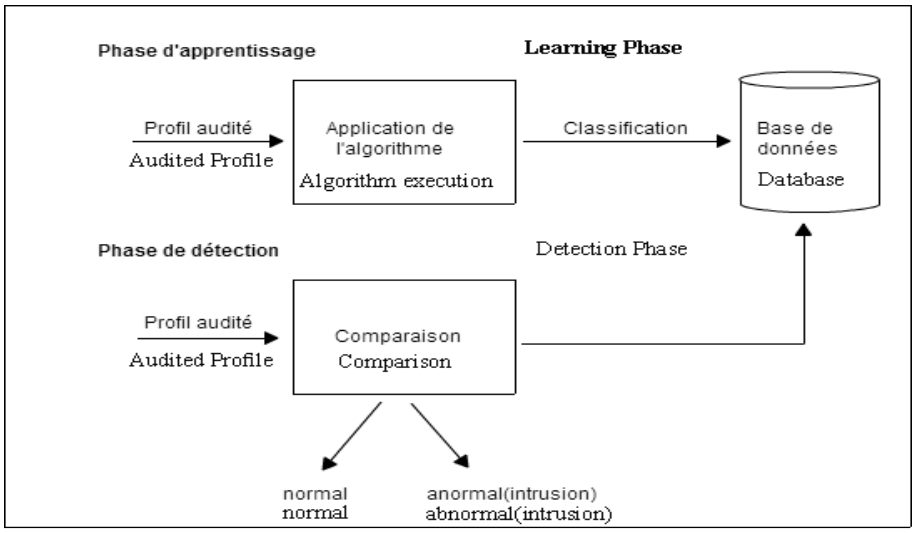

Fig.1 General scheme of the system

- **Audit the different user profiles :** each profile is represented as a vector of the measures cited above.

If we denote *lambda* the profile corresponding to a behavior of a given user, we have :

$$\lambda = \begin{pmatrix} x_1 \\ x_2 \\ . \\ . \\ . \\ x_n \end{pmatrix}$$

avec $x_i, i = 1, ..., n$ are the representative measures of the user profile

- **Classify users according to their behavior :** We are using for the purpose the K-means algorithm.The choice of this method is due to its efficiency and speed especially when dealing with large data files, as opposed to hierarchical methods. The goal is to divide the observations into K partitions where each observation belongs to the partition with the nearest mean. The following notation is used :
  - Les $\lambda_i \in \mathbb{R}^{\mathbb{P}}$,$i = 1, ..., n$ are the points to be separated.
  - Les z_i^k are indicator variables associated with λ_i so that $z_i^k = 1$ if λ_i belongs to cluster k, $z_i^k = 0$ if not. z is the matrix z_i^k.
  - μ is the vector of $\mu_k \in \mathbb{R}^{\mathbb{P}}$, where μ_k is the center of cluster k. We define the distortion measure$J(\mu, z)$, with :

$$J(\mu, z) = \sum_{i=1}^{n} \sum_{k=1}^{n} z_i^k ||\lambda_i - \mu_k||^2$$

  the algorithme aims to minimize $J(\mu, z)$, it is in the form of an alternating minimization algorithm
  - Etape 1 : "'choose the vector μ"'
  - Etape 2 : J is minimized compared to $z : z_i^k = 1$ for k $\in arg\ min\ ||\lambda_i - \mu_k||$, ie we associate to λ_i the nearest center μ_k .
  - J is minimized compared to $\mu : \mu_k = \frac{\sum_i z_i^k \lambda_i}{\sum_i z_i^k}$ The minimization step with respect to z returns to spread the λ_i according to Voronoi cells whose centers are the μ_k. In the minimization step μ, μ_k is obtained canceling the k-th coordinate of the gradient in Jμ.
    Users will therefore be classified according to their behaviors.We note the user and the class to which it corresponds in a database.
  - **Intrusion detection :** After learning of profiles and the constitution of classes, comes the detection phase.A profile of a user will be audited, if a new user, it must be rejected.If not (ie a User used in learning),his behavior will be compared to his past behavior .If there is a deviation, then we have an anomaly (intrusion) .The similarity is calculated through a Euclidean distance as it is the most commonly used because the easiest to calculate.

5 Experimental results

According to [PG00], a HIS explicitly refers to the internal information system in a health organization. The establishments are typically :

- Hospitals
- clinics

The players involved in this type of system are :

- Patients
- Health professionals (Doctors, Nurses ...)
- Administrative staff

Our approach consists of three user profiles :

– -Patient
– -Doctor
– -Administrator

-To test our method we will use a simulation that lasts a month of work.The variables involved are **SessionCPU** in hours(h) et **DataVolume**(integer)

-Users are divided into two groups : Familiar group (Patient + Doctor) : We will use 22 profiles of each class (2 classes) for Learning. The 16 other profiles will be used to test the generalization ability.

Unfamiliar group(Administrator) :To test the ability to reject unknown users (Detection).
The following table shows the profiles generated :index "1" represents "Patient" class,"2" represents "Doctor" and "3" represents "Administrator".

	$S.CPU_1$	$S.CPU_2$	$S.CPU_3$	$DataVolume_1$	$DataVolume_2$	$DataVolume_3$
jour1	4.8	2.3	2	9	5	8
jour2	4.9	2.5	3	7	5	10
jour3	5	3	4.5	9	4	7
jour4	4.7	2.8	2.5	10	3	9
jour5	4.5	2	4.6	6	6	6
jour6	4.8	2.5	2.6	11	5	8
jour7	4.7	3.1	3	8	3	7
jour8	4.9	2.4	2.6	10	4	9
jour9	5.2	2.6	2.1	10	3	7
jour10	5	3	2.7	11	5	4
jour11	4.7	2.5	2.8	9	4	9
jour12	4.8	2.3	2.4	8	4	8
jour13	5	2.4	2.3	9	5	7
jour14	5.1	2.2	2.5	10	5	7
jour15	5.4	3	2.7	7	5	8
jour16	6	3.1	3.3	8	4	8
jour17	4.6	3.2	2.5	10	4	8
jour18	4.2	3.3	3	11	6	9
jour19	5	3.6	2.2	10	5	7
jour20	5	3.7	2.9	6	4	6
jour21	4	2.5	2.7	7	4	7
jour22	4.3	2.8	2.6	8	3	8
jour23	4.2	2.7	3.1	10	4	7
jour24	4.7	2.4	2.5	11	6	9
jour25	4.6	2.3	2.5	9	7	8
jour26	4.5	2.4	2.5	8	6	9
jour27	4.7	2.5	2.4	10	5	8
jour28	5.3	2.4	3.1	7	4	7
jour29	5.4	2.2	2.5	6	1	9
jour30	5.5	2.7	2.7	7	4	6

Tab.1 Generated Profiles

The following graph shows the familiar group (used for learning).

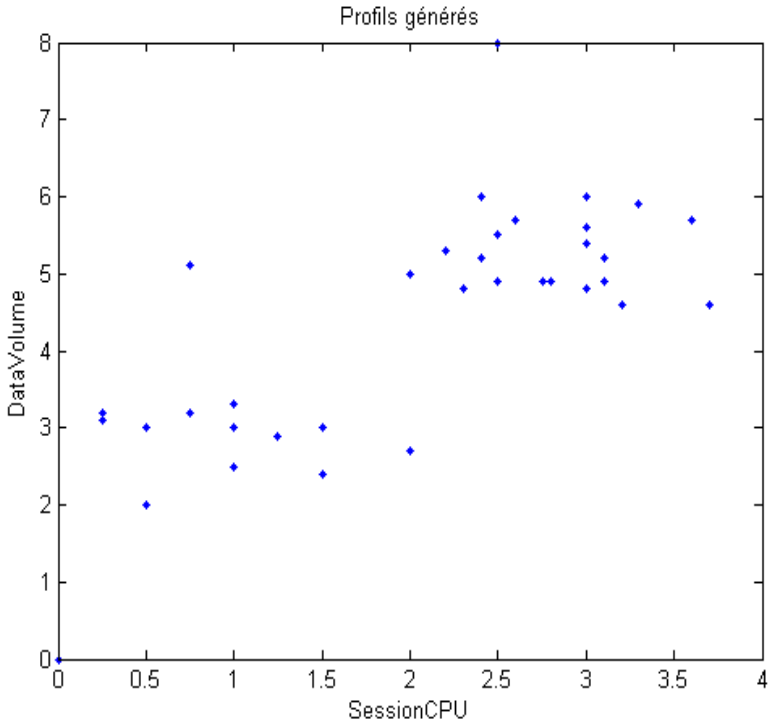

Fig.2 Familiar Group

After classification done using MATLAB, we get the following graph :

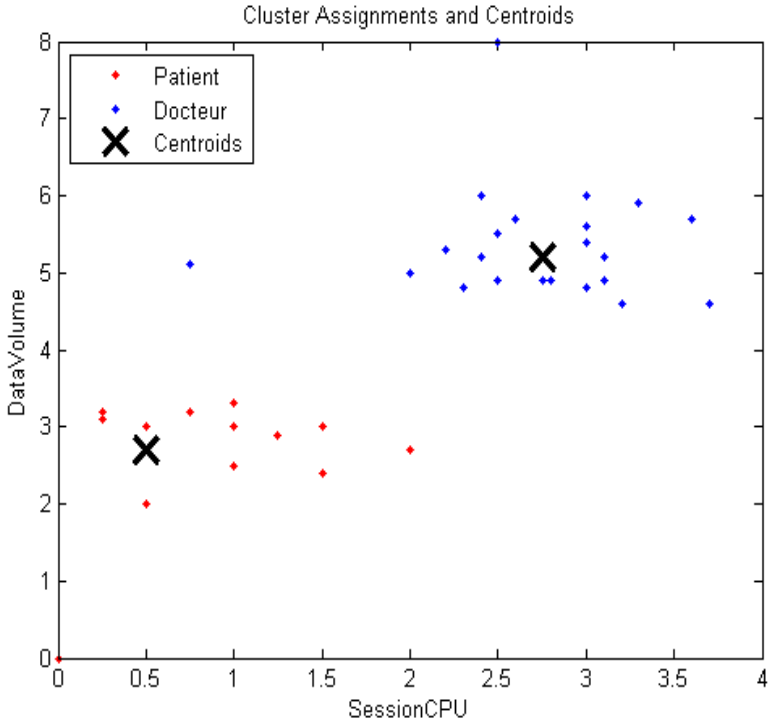

Fig.3 Classification

We considered the following rates to present our results :

- **Identification with success** Rate of profiles successfully identified,
- **False Negative :** The percentage of profiles that are not successfully identified nor rejected,
- **Rejection with success :** Defines the rate of unfamiliar profiles that are rejected,
- **False positive :** The rate of rejection in familiar group.

The threshold for the Patient is 1.82.For The Doctor is 0.6. The following table summarizes the results obtained (L) :Learned group of the familiar group. (UnL) :Unlearned group of the familiar group.

Rates(%)	Familiar Group	Unfamiliar Group
Identification with success	44/44(100%)(L)+ 12/16(75%)(UnL)	-
False Negatives	0%(L)+0%(UnL)	0%
rejection with success	-	100%
False positives	0(0%)(L) + 4/16(25%)(UnL)	-

Tab.2 System performance

The table shows relatively high performance system. Besides, It was able to identify all the profiles of learned group(in familiar group) with 25 % of unlearned group(familiar group) rejected,something due to the change of user behavior during the period of experiment.The system have rejected successfully the whole unfamiliar group.

Conclusion and outlook

In this article we have modeled a new intrusion detection system based on the K-means algrorithm.The method gives results quite satisfactory.However two constraints arise :

- When the user behavior changes significantly over time, it produces an increase of false positive rate of the familiar group(ie increase in the rejection rate of familiar group).
- The method may not work properly when the data is huge (huge number of users and / or long duration of audit time)

In a future work, we will try to address these constraints while introducing the concept of Big Data that addresses the problem of huge amounts and multiple varieties of data .

Références

[Als13] Wafa' Alsharafat. Applying artificial neural network and extended classifier system for network intrusion detection. *The International Arab Journal of Information Technology,Vol.10, No.3*, May 2013.

[And80] J. P. Anderson. Computer security threat monitoring and surveillance. *Technical report, James. P. Anderson Co., Fort Washington, Pennsylvania*, 1980.

[BG] Yacine Bouzida and Sylvain Gombault. Profils propres pour la detection d'intrusions. *Actes du symposium SSTIC03*.

[Ca02] F. Cuppens and al. Recognizing malicious intention in an intrusion detection process. *Second International Conference on Hybrid Intelligent Systems*, December 2002.

[CMC11] Malcolm Corney, George Mohay, and Andrew Clark. Detection of anomalies from user profiles generated from system logs. *AISC*, 2011.

[Da92] H. Debar and al. A neural network component for an intrusion detection system. *Proc. 1992 IEEE Computer Society Symp. On research in security and Privacy, Oakland, CA*, May 1992.

[DN85] D. E Denning and P. G. Neumann. Requirements and model for ides, a real time intrusion detection expert system. *Technical Report , Computer science Laboratory, SRI International, Menlo Park, CA*, 1985.

[E.D86] Dorothy E.Denning. An intrusion-detection model. *IEEE*, 1986.

[HCM+92] N. HabraB., Le Charlier, A. Mounji, I. Mathieu, and Asax. Software architecture and rule based language for universal audit trail analysis. *Proc. 2nd Symp. on Research in Computer Security (ESORICS), Toulouse, Berlin, Lecture Notes in Computer Science, vol. 648, Springer, Berlin*, November 1992.

[Ilg93a] K. Ilgun. Ustat, a real time intrusion detection system for unix. *Proc. IEEE Symp. On Research on Security and Privacy*, May 1993.

[Ilg93b] K. Ilgun. Ustat, a real time intrusion detection system for unix. *Proc. IEEE Symp. On Research on Security and Privacy, Oakland, CA*, May 1993.

[Ja93] R. Jagannathan and al. System design document : Next generation intrusion detection expert system (nides). *Technical Report A007/A0014, SRI International, Ravenswood Avenue, Menlo Park, CA 94025*, March 1993.

[Jol02a] I. T. Jolliffe. Principal component analysis. *2nd Edition, New York : Springer Verlag*, 2002.

[Jol02b] I. T. Jolliffe. Principal component analysis. *2nd Edition, New York : Springer Verlag*, 2002.

[KS94] S. Kumar and E. Spafford. A pattern matching model for misuse intrusion detection. *Proc. 17th National Computer security Conf.*, October 1994.

[Me98] L. Me. Gassata, a genetic algorithm as an alternative tool for security audit trail analysis. *RAID, the first international workshop on Recent Advances in Intrusion Detection*, October 1998.

[NM07] Toosi N. and Kahani M. A new approach to intmsion detection based on an evolutionary soft computing model using neuro-fuzzy classifiers. *Computer Communications, vol. 30, no. 10, pp. 2201-2212*, 2007.

[PG00] PonÃ§on and GÃ©rard. Le management du systÃ¨me d'information hospitalier : la fin de la dictature technologique. *(ISBN 2-85952-786-9), p.25*, 2000.

[PK92a] P. Porras and R. Kemmerer. penetration state analysis, a rule based intrusion detection approach. *Proc. 8th Annual Computer Security Applications Conf.*, November 1992.

[PK92b] P. Porras and R. Kemmerer. penetration state analysis, a rule based intrusion detection approach. *Proc. 8th Annual Computer Security Applications Conf.*, November 1992.

[Sa91] S.R Snapp and al. Dids (distributed intrusion detection system), motivation, architecture, and early prototype. *Proc. 14th National Computer Security Conf., Washington, DC*, October 1991.

[SAA03] Mukkamala S., Sung A., and Abraham A. Identifying significant features for network forensic analysis using artificial intelligent techniques. *International Journal of Digital Evidence, vol. 1, no. 4, pp. 1-17*, 2003.

[SAA04] Mukkamala S., Sung A., and Abraham A. Modeling intrusion detection system using linear genetic programming approaches. *Proceedings of 17"' International Conference on Industrial and Engineering Applications of Artificial Intelligence and Expert Systems, vol. 3029, pp.633-642*, 2004.

[Sma88] S. Smaha. Haystack : an intrusion detection system. *4th Aerospace Computer Security Applications Conf.*, October 1988.

[VL89] H. S. Vaccaro and G. E. Liepins. Detection of anomalous computer session activity. *Proc. IEEE symp. On Research in Security and Privacy*, 1989.