Methods of Machine Learning in System Abnormal Behavior Detection

Pavel A. Savenkov^(✉) and Alexey N. Ivutin^(✉)

Department of Computer Technology, Tula State University, Tula, Russia
pavel@savenkov.net, alexey.ivutin@gmail.com

Abstract. The aim of the research is to develop mathematical and program support for detecting abnormal behavior of users. It will be based on analysis of their behavioral biometric characteristics. One of the major problems in UEBA/DSS intelligent systems is obtaining useful information from a large amount of unstructured, inconsistent data. Management decision-making should be based on real data collected from the analysed feature. However, based on the information received, it is rather difficult to make any management decision, as the data are heterogeneous and their volumes are extremely large. Application of machine learning methods in implementation of mobile UEBA/DSS system is proposed. This will make it possible to achieve a data analysis high quality and find complex dependencies in it. A list of the most significant factors submitted to the input of the analysing methods was formed during the research.

Keywords: DLP · UBA · Big data · Data science · Software · Information system · Machine learning · Unstructured data · Dynamic model · Neural network · K-nearest neighbors algorithm

1 Introduction

Over the past few years there has been a sustained increase in interest in data security challenges in enterprise information systems. Many experts in the field of information security (IS) show a trend of increased number of internal incursions compared to external ones. Concerns about the issue are reinforced by the fact that companies usually focus on protecting against external threats, while analysts point out that more than half of intrusion and computer security breaches are caused by their own employees or others with legitimate access to the information system. Theft and sale of confidential information, dissemination of information with limited asses are only a small part of IS-incidents directly related to internal threats [1].

Thus, information security internal threats are caused by harmful actions of users (insiders) who have legitimate access to the corporate network. This type of attack is usually distinguished from attacks that result from compromising the company employees' accounts, where the intruder (hacker) gets access to corporate IT resources using stolen accounts.

In the case of an internal attack, the insider usually acts maliciously and most likely knows that he is violating his company's security policy. However, when internal threats

are classifying, a group of threats is sorted out. There are threats committed without malicious intent (random), by negligence or due to technical ignorance.

The sources of internal threats can be referred to different categories of users who have or have had access to the corporate network. The group of potentially malicious users of the corporate network is difficult to identify, and it can be much wider than it may seem at first glance [2].

In addition, the amount of data that can be the goal of internal attacks is constantly increasing at a high speed. Financial reports, customer or employee data, product technical documentation, and etc. can be examples of such vulnerable data. Similar data may be located in different locations on the corporate network at the same time, as they are required for processing by different departments/employees, they are stored on corporate mail servers, backed up and etc.

Data breach is one of the most dangerous internal threats for modern companies. The number and complexity of internal attacks continues to grow. In 2015 it was registered an increase of about 64% more attacks than in 2014. According to research provided by the Ponemon Institute, which was supported by IBM, the company suffered an average loss of \$4 million per incident in 2016, while the average value of a lost or stolen document was estimated at \$158. The data were based on an analysis of 383 companies in 12 countries.

1.1 Typical Data Breach Stages

As show modern researches, from the moment the user decides to steal the data to directly the data forwarding, it takes from weeks to months to prepare the breach and this time is spent on breach preparation stage. Therefore, more experts now agree that data breaches need to be identified even before the stage of data transfer beyond the company's information technologies sphere [3].

We will describe the typical stages of data loss more detailed. (see Fig. 1). A legitimate employee becomes an insider beginning at some crucial point, for example at the moment after social media/email communication with one of company's competitors (the "Start of internal invasion" phase). After that, the employee-insider enters the research phase ("Research phase"), where he attempts to find and access information, in which he is interested, herewith using his current rights or trying to expand them in legitimate ways. At this stage, cases in which an insider under various excuses asks his colleagues to give access to their rights to reach a certain category of information often appear there, and Edward Snowden's actions are usually used as an example of such behaviour. It will also be relevant to note here the importance of the users' authenticating task, i.e. determining that the user is the one on behalf of whom he or she has authorized. The insider's "Research phase" may continue for weeks and months, but as time goes on, he tends to find a way to gain access to the data which he is interested in.

After getting access to the desired information, the "Data hiding" phase begins. At this stage, the main goal of the insider is to test the existing information security systems of the company and find the optimal way to safely exfiltrate the received information. No attempt had been made to transfer data beyond the information perimeter of the organization before this stage, so traditional Data Loss Prevention (DLP) protection did not work. To achieve the goal of "Data hiding" stage it will be suitable for the insider to use

Fig. 1. Scheme of typical data breach

any actions which can be justified with carelessness (negligence) or ignorance (technical incompetence) in case of their disclosure, i.e. to reduce to unintentional violation. Insiders often use fairly simple techniques, such as creating “fictive” data which are similar in content structure to data planned for exfiltration, but that is not confidential at the same time. The insider will repeat such attempts to transfer data with a certain frequency until he can determine the method of transmission, at which the IS systems do not work properly. Then, having access to the relevant confidential information and choosing the method of its theft, the insider goes to the final stage of leakage - “Data exfiltration”.

1.2 Abnormal User Behavior

It follows from the description of the data breach scenario that in most cases the actual theft of information is preceded by abnormal (though possibly permitted) behavior of the user, i.e. the user even before the theft of information begins to take actions that are not typical for his previous activity both according to the set of performed operations and to the content of the processed information. Also, the very stage of data breach preparation during which abnormal behavior of the user is observed usually takes quite a long time, up to several months. Therefore the direction of user behavior analysis for detecting anomalies has been actively developed over the past few years [4].

1.3 Purpose of Internal Invasions

The purpose of internal intrusions is usually to gain access to textual information (financial reports, contracts, technical documentation, e-mail, etc.). Therefore, the key moment is to detect abnormal behavior of users during working process with data. Abnormal behavior may indicate that the user is not the one on behalf of whom he authorized (user authentication task), or the user is interested in corporate documents that are not related to his current work activity. This is a sign of potential information leakage (early detection of information theft attempts task) [5].

1.4 UBA as a Stand-Alone Class of Information Security Systems

At present, an independent class of information security systems has been formed, based on machine learning methods which are used to identify signs of unusual user behavior. Gartner designates this system class as UBA (User and Entity Behavior Analytics). UBA systems, unlike DLP, monitor a wide range of user actions and make decisions which are not based on expert security policies, but choose right direction on the basis of historical data on legitimate user performance. These systems detect early signs of breach, so their main purpose is not to block user actions, but to provide analytical data to the IS service describing why the detected actions are abnormal to a particular user. As defined in the Gartner report, UBA systems construct and apply user behaviors (profiles) models based on machine learning methods to identify signs of abnormal behavior [6].

1.5 Early Signs Detection of Abnormal Behavior

It is relevant to develop the direction of users abnormal behavior early signs detection based on machine learning methods to solve the following problems of information security:

1. The task of early attempts to steal information detection is the procedure of detecting the facts of abnormal or suspicious behavior of insiders, who may precede or be directly part of the organization attempting to steal information.
2. The user authentication task is validity evaluation procedure. It examines if the user working with the protected computer system is really the person on whose behalf he authorized.

When analyzing a user's behavioral image, a large amount of real data is collected. However, it is quite difficult to make any decision on their basis, as the data are heterogeneous and the number of parameters for analysis is extremely large.

To solve the problem of anomaly search by means of users behavioral biometric characteristics analysis, it is proposed to use methods of machine learning and intelligent data processing [7].

1.6 Factors of Users Abnormal Behavior

To solve the problem of detecting anomalies in user behavior, we will determine the factors of users abnormal behavior. These factors are shown in Table 1.

Table 1. Factors for abnormal user behavior

Data source	User behavioural sign
Used applications	Changing of the activity time in different applications
Visited sites	Changing of the visited sites list; Visiting sites of another user group
Camera, images	Finding prohibited fragments in the image (photos of documents with signatures, stamps)
Calls, dictophone records	Finding prohibited key phrases, or phrases belonging to another user group
Typed and received text	Phrases which are prohibited to input
GPS coordinates	Change of location

The following scenario is common practice in the field of experimental studies in the internal threats detection sphere.

1. All actual behavioral data which was collected is considered to be legitimate.
2. Data modeling pre-specified internal threats are added to the collected behavioral data.
3. The users daily activity is analyzed and the task of binary classification is considered: it is necessary to determine days with abnormal users activity, which corresponds to specified threats.

2 Anomaly Search Methods Determination

For the implemented UBA system with DSS functionality, based on the analysis of behavioral biometric characteristics of the enterprise personnel due to the large volume of input analyzed data it is proposed to use machine learning methods and intelligent data processing. Such actions will reduce the number of resulting parameters.

Collecting input data is implemented through a mobile application installed on the mobile device with OC Android of a certain employee of the enterprise. For behavioral analysis, it is proposed to use the following methods for data analysis:

- The k method of the nearest neighbors:
 - (a) the used applications;
 - (b) GPS coordinates (employee's travel history);
 - (c) the visited sites;
 - (d) the typed text;
 - (e) the received text.
- neural networks:
 - (a) – calls, dictophone records;
 - (b) – camera, image.

The software indicates certain deviations of the user's behavioural characteristics, suggests to perform a number of actions to the administrator. In some cases the system administrator decides to block the user.

Neural networks are used to analyze data such as recorded calls, recorded sound from a dictaphone and photos. In order to find deviations the network preliminary training takes place.

To find deviations from the user's reference profile in such data as Employee Movement History (GPS), typed text, the resulting text the k method of the nearest neighbors is used. You can reduce the data analysis load as well as bring down the number of iterations in training applying this method. During training this method only stores training data. Classification is performed when new untagged data is obtained at the input of the method. In this case, the data received from the user is checked and it starts process of finding if it belongs to a specific users' group or user.

User characteristics are compared by searching Euclidean distance to all records from the obtained sample [6].

$$d(p, q) = \sqrt{(p_1 - q_1)^2 + (p_2 - q_2)^2 + (p_3 - q_3)^2} \quad (1)$$

Then k records are selected, for which the Euclidean distance from the current record to the new one will be minimal.

The sum of the class weights (distances) is calculated by the formula:

$$Z_{classN} = \sum w(x_{(i)}) \quad (2)$$

Z_{classN} – is sum of class weights from a new point to N class;

$w(x_{(i)})$ – is weight of i-th object of class N falling into the area of nearest k objects.

Then the sum of the inverse squares of the distances between the records of this class and the new one is counted for each user. The class is given to the new record for which the sum of the inverse squares is the largest. In Fig. 2, it is shown an example of assigning a new object to one of two existing classes with $k = 6$.

Fig. 2. Representing the k-nearest neighbors method in a two-dimensional factor space

In this case knn method uses such parameters as:

- $d(p, q)$ is the distance between points;
- p_n is the coordinate of the p point along the axis n ;
- q_n is the coordinate of the q point along the axis n .

If the user or group ID assigned by the method k of the analyzed record corresponds to the user or group ID obtained by the initial authorization in the system, it is considered that the obtained characteristics correspond to the reference performances and there were found no deviations from these ones.

In case the ID obtained by the initial authorization in the system does not correspond to the ID assigned by the method k of the newly generated record, it is considered that the obtained characteristics differ from the reference performances or belong to another user or group of users.

Based on the selected data, the software offers to implement a number of actions indicating certain deviations of the user from the reference profile. Thereby the administrator should be informed about this range of actions. In some cases the system administrator can make the decision to block the user.

3 Software Implementation of Experimental Client-Server Software Sample

3.1 Basic Structure of the Client-Server Software Exploratory Prototype

To collect behavioral biometric user characteristics collecting software agents are used, they are installed directly on mobile devices of users (data sources) and transmit the collected information to a single repository for its subsequent processing.

Behavioral information processing during the user's working process with data consists of three stages:

1. Collecting of user behavioral data. Software agents should implement the collecting and intermediate local memory storage of behavioral information in order to optimize the load on the data network or in case if there is no connection with the single storage.
2. Sending of the collected behavioral information to the server. Transfer of behavioral data from different mobile devices of users to a single repository.
3. Reception of behavioral information from monitoring agents and storing it in a single centralized repository.

In Fig. 3 the basic diagram of connections in the system is presented.

A mobile agent application is installed on each mobile device connected to the system. After the application is installed on the employee's mobile device, the system administrator gives the user the list of parameters collected from his device for its identification. The set of analysed parameters that will be collected on the device and analysed on the server differs, it depends on the user/group of users. The system administrator generates the list of analyzed parameters and user groups.

Fig. 3. Scheme of a system communications

The mobile application starts when the Mobile Device begins its work as a foreground service. The mobile device requests a list of commands from the server once in N minutes. The instruction fetch period dynamically varies, it depends on the level of user activity. After selecting the list of commands which are available for realization on the employee's device, they are further processed, information is received and data are sent to the Main Server. The commands (Event Server) have different order of operations priorities. Commands have different statuses such as running once and cycle operation with a timer. After sending the data, the central Main Server receives this information, processes it, then analyzes this one with the Data Analysis server and writes the source and result parameters to the Data Base. If an error occurs, this command is realized again on the mobile device.

The Admin Console panel is directly connected to the master server and has certain capabilities such as:

- Ability to manage a group or user-defined Mobile Control Server;
- Adding new Event Server commands to the run;
- Generation of reports.

Direct access to the server provides continuous access to management in case of DDOS attack bypassing DDOS attack filtering by the "DDOS Protection" module.

Based on the selected data the software offers to realize a number of actions to the administrator therewith indicating certain deviations of the user's behavior from the reference profile. In some cases the system administrator decides to block the user.

3.2 Monitoring Agent Architecture

The monitoring agent is implemented in the high-level programming language C# with the help of using the mobile application development framework “XAM-ARIN.ANDROID.” A monitoring agent distribution to collect behavioral biometric information is an “apk” format file designed to be installed on an employee’s mobile device. The distribution contains all components and libraries which are necessary for employee monitoring and includes such modules as:

- Data Collection Module;
- Local Database;
- TCP/IP Exchanger.

The Data Collection Module is a module that collects and pre-processes information from an employee’s mobile device before making a record in a local database.

Local Database is a module that temporarily stores data to unload the network channel when information is transmitted to the main server. Also it stores data if there is no connection to the server.

TCP/IP Examiner is a module which is responsible for the client server exchange and power saving during the process of data sharing over the network. Figure 4 shows the basic structure of the mobile agent.

Fig. 4. Basic mobile agent structure

3.3 Main Server and Its Child Units Operation Scheme

The “Main Server” is a central server that receives data from client devices with installed mobile agent applications. “Main Server” can be installed on the Windows operating

system. The “Main Server” is implemented on the basis of Web API ASP.NET technology in C # language. The “Main Server” is connected with all major system modules such as:

- Event Server;
- Mobile Control Server;
- Data Analysis Server;
- Database;
- Admin Console.

4 Conclusion

During the study, a list of the most significant factors submitted to the input of the analysing methods was formed. With the increase of the signs number, the number of objects that must be in the learning sample to cover all kinds of situations has exponentially increased. Number of input parameters reducing helped to lower the amount of learning sample for the knn method. The diagram of user identification correctness is shown on Fig. 5.

Fig. 5. Correctness of identification diagram

User data was correctly identified in 93% of cases and incorrect in 7% of cases on average. Due to the decrease in the number of output data and the increase in the correctness of user identification by their characteristics, the average time to obtain useful data in the management decision support system was reduced.

Moreover, the basic architecture of the server software complex client was built and it provides high stability in data processing.

Early anomaly detection will allow the system administrator to make balanced management decisions, reduce operational costs, and increase enterprise competitiveness.

Due to the application of various algorithms and methods of data analysis and machine learning in mobile UBA it has been possible to improve the informativity of the resulting data. Early anomaly detection will allow the system administrator to make balanced management decisions, reduce operational costs, and increase enterprise competitiveness [8].

Acknowledgments. The reported study was funded by RFBR, project number 19-37-90111.

References

1. Szaniawska, K., Lehmann, M.: The human factor in managing the security of information. In: Ahram, T., Karwowski, W. (eds.) AHFE 2019. Advances in Intelligent Systems and Computing, vol. 960, pp. 38–47. Springer, Cham (2019). https://doi.org/10.1007/978-3-030-20488-4_4
2. Magklaras, G.B., Furnell, S.M.: A preliminary model of end user sophistication for insider threat prediction in IT systems. *Comput. Secur.* **24**(5), pp. 371–380 (2005)
3. Hu, Q., Tang, B., Lin, D.: Anomalous user activity detection in enterprise multi-source logs. In: 2017 IEEE International Conference on Data Mining Workshops (ICDMW), pp. 797–803. IEEE (2017)
4. Ryan, J., Lin, M.J., Miiikkulainen, R.: Intrusion detection with neural networks. In: Advances in Neural Information Processing Systems, pp. 943–949 (1998)
5. McGough, A.S., et al.: Insider threats: identifying anomalous human behaviour in heterogeneous systems using beneficial intelligent software (ben-ware). In: Proceedings of the 7th ACM CCS International Workshop on Managing Insider Security Threats, pp. 1–12 (2015)
6. Kambhampaty, K.K.: Detecting Insider and Masquerade Attacks by Identifying Malicious User Behavior and Evaluating Trust in Cloud Computing and IoT Devices: дис. – North Dakota State University (2019)
7. Savenkov, P.A., Ivutin, A.N.: Methods and algorithms of data and machine learning usage in management decision making support systems. In: 2019 8th Mediterranean Conference on Embedded Computing (MECO), pp. 1–4. IEEE (2019)
8. Ivutin, A.N., Savenkov, P.A., Veselova, A.V.: Neural network for analysis of additional authentication behavioral biometric characteristics. In: 2018 7th Mediterranean Conference on Embedded Computing (MECO), pp. 1–3. IEEE (2018)