User behavior analysis based on Identity management systems' log data

Borut Rožac[1], Radovan Sernec[1], Andrej Košir[2], Andrej Kos[2]

[1]*Telekom Slovenije d.d, Cigaletova 15, 1000Ljuljana*
[2] *Fakulteta za elektrotehniko, Tržaška c. 35, 1000 Ljubljana*
E-pošta: borut.rozac@telekom.si

Abstract

Identity Management Systems (IdM systems) are repositories where users' security credentials are kept and managed. One of many tasks performed by IdM system is an authentication and authorization of users while they are using applications. In case where IdM systems are organized as central repository, which means that authentications and authorizations for different applications are carried out on one central IdM system. This case is common for enterprises with more than 50 employees where various aspects of users' behavior can be analyzed. By applying machine learning method SVM (Support Vector Machines) on IdM system log data with application on searching employees that are acting enthusiastically within working day, we showed that analysis of IdM systems log data can be successfully applied for analyzing employees behavior within an enterprise.

1 Introduction

An identity management system refers to an information system that can be used for identity management within or across enterprise's environment. Identity management includes managing digital identities, their authentication, authorization, roles and permissions within environment of their operation. Digital identity is a set of attributes that uniquely describes a subject that can be a person, computer, phone, tablet device, printer, server, group of users, etc.. Within enterprise's environment subjects, in this paper meant as employees and their devices, can access to a different information systems, for instance to the internal web page, e-mail system, CRM (Customer Relation Management) system, VPN (Virtual Private Network) service and other internal applications. Many of these applications require from users to claim their identity usually carried out thru login form where users pass their username and password. For a purpose of increasing security and usability, authentications and authorizations for different applications within an enterprise are carried out thru central IdM system.

Identity management systems like other information systems produce records about their operation, commonly known as logs. Records related to a security and ordered chronologically are known as audit trails or audit logs, in this paper named as logs. Such logs are also produced during various financial transactions, health transactions, or in communications systems. Audit logs are normally stored and processed for different purposes for instance, troubleshooting, security analysis, criminal prosecution and justice, and in research area where various machine learning methods are applied on these data for different purposes. In [1] an example of analyzing traces of OS systems calls and network traffic by machine learning methods is described in order to detect intrusion attacks on computer systems. In [3] an example of credit card fraud detection by three different machine learning methods SVM (Support Vector Machines), logistic regression and random forests is described. In [4] a web users behavior profiling is being described based on web server log data with SVM method. The goal of this paper is to analyze employees' behavior by applying machine learning methods on IdM systems audit logs combined with logs from other information systems within enterprise.

This paper is organized as follows. In second chapter two machine learning methods briefly describe which were applied for searching user behavior on data similar in nature to IdM systems' log data, followed by description of problem of searching enthusiastic employees within the enterprise based on IdM system log data analysis. Chapter is concluded with description of data model in which log data from three different information systems within the enterprise were combined, IdM system, DHCP server and access-control system. In third chapter we will describe test setup and analyze results of the test.

2 Materials and methods

2.1 Experiment preparation and datasets

A collection and analysis of IdM systems' log data has been mainly performed for the purposes of auditing security accesses of users within an enterprise, we decided to demonstrate more flexible application of searching enthusiastic employees within the enterprise. Analysis was performed on larger enterprise's IdM system, where data was previously anonimized with one-way hash function HMAC-SHA1, data on which experiments were performed for this paper were presented in a way that neither names of applications, places or any identifiable data could be associated with real names.

Based on nature of work in the company and for the purpose of the experiment we defined two cases where employees can act as enthusiastic employees. In the first case enthusiastic employees are employees who work on the field, for instance, sales people or maintenance people. After their return to a company they use some specific internal application or set of applications where log-in with username and password is required. In second case enthusiastic employees are those who attend meetings on different locations within the company, and after returning from the meeting they act similarly as employees in first case, or for instance they are accessing to the specific application from the meeting, which can be seen as event where employee's computer was leased an IP address from a VPN IP subnet. Events where employees are leaving the company for a meeting on different location within a company or are heading to meet a customer can be observed as events on access-control system log data that occurs for instance at entrance doors, doors within a building, or at parking lot ramp where employees place their cards on card reader.

To prepare data for classification with machine learning methods, we manually labeled data for enthusiastic employees based on graphical representation of event sequences in a trellis graph for a period of one day. Trellis graph is a graph where the nodes are ordered in vertical slices and each node at each time is connected to at least one node at an earlier and at least node at a later time. Depiction of trellis graph is in Figure 2 where nodes are presented with arrows and are meant as occurrence of an event. Trellis graph includes IdM systems' events triggered as results of employees log in events to the specific applications, events triggered by employees' devices, events from DHCP servers for employee's devices, and events from access-control system. For a specific day for each user a graph of consecutive events is created. Such graph can be presented as a vector of events, where vector length varies depends on number of events, or by a vector of fixed length where length is determined by number of all possible events. Each possible event in such vector represents a feature, thus we construct a feature vector where we mark a presence of an event with "1" and absence with "0". Such representation of data is known as Bernoulli multivariate model, contrast to this model is multinomial model where events are represented with frequencies of their occurrences. Similar models were used for text and SPAM classification [2], and for detecting intrusive behavior of computer applications [1]. We have to note that every distinct event is marked as index in the feature vector and no transitional properties of the employees' behavior is included in that model.

By constructing such model we first manually labeled 237 different samples of employees, whose working habits, defined by our criteria, show that they work more enthusiastically, remaining employees who are working normally were chosen automatically with R-language scripts where for each employee, activity for three randomly chosen working days were collected. When choice of ordinary employees was performed, we excluded enthusiastic employees and dates on which they were labeled. Together we gathered 3416 samples of employees events on three different information systems, 237 were samples of enthusiastic employees, remaining 3179 samples were of ordinary employees. In machine learning language enthusiastic employees means positive labeled data, ordinary employees means negative labeled data. Features were combined from distinct events of three different data sources where each feature means an occurrence of one of possible events within a period of one day. Events on IdM system can be triggered by employees or their workstations when performing authentication to a specific service. There were 38 distinct events for a DHCP server, where event type was a DHCP lease to employees' workstations at different locations, where locations are meant as different buildings where employees are working. Access control events are triggered when employee enters at specific part of building, which can be an entrance doors at different buildings, automatic doors within building or entrances to a premises across country where communication equipment is held.

Table 1: Features of feature vector where distinct event type means new feature

Event type	Number
IdM System Employees	60
IdM System Workstations	60
DHCP events	38
Access control events	143
All features	301

2.2 Machine learning methods in log analysis

In last two decades many machine learning techniques have been applied in the area of information systems security, where with the help of machine learning techniques intrusions and misuse of those systems were detected. This field is also known as Anomaly-based intrusion detection, where monitored activities of those systems are classified as normal or anomalous. Analysis is usually performed on application servers' log data, OS system- call log data and network traffic data.

In [1] a benchmark of several methods was performed where authors proved that by reducing dimension of system call log data with PCA (Principal Component Analysis) method before applying machine learning methods on these data can significantly reduce training time without penalty on classification accuracy. Authors have demonstrated that classification performance, based on thresholds with distance based methods (Euclidian distance, Cosine distance, Signal/Noise Ratio distance) where dimensionality reduction with PCA was applied, was equally good as with HMM (Hidden Marko Models) and ECM (Eigen Co-occurrence Matrix) method while training times were significantly shorter. It must be noted that datasets

were OS logs of program executions, where each program is executed by several processes, each process produces a trace o system calls. For a PCA method traces of system calls were presented with vector of length *m*, where *m* is number of distinct system calls or features. Authors performed tests with reduced number of features where instead of 168 features in original data they used from only 1 to 41 principal components.

One of the more contemporary machine learning method is SVM (Support Vector Machines). SVM is a binary classification method, also known as maximum margin linear classifier, which tries to find optimal decision boundary with maximum margin between linearly separable data. In case that data is not linearly separable, SVM maps that data in higher dimensional feature space, with the help of kernel functions, where previous non linearly separable data becomes linearly separable [5], [6]. Classification performance of SVM can be tuned by changing cost parameter C, where greater values of parameter mean smaller training error. SVM have fund its use in many areas where machine learning methods are applied for solving certain problems from image processing, bioinformatics, text classification, time series analysis and credit card fraud detection [3].

In this paper we will perform a classification with SVM method, with and without dimensionality reduction with PCA.

3 Experiments and results

The objective of our experiment was to check how class imbalance affects performance of SVM classifier with and without PCA transformation of data. All tests were executed with 10-cross fold validation scheme, where 70% of data were used for training the model, remaining 30% for checking how well model classifies previously unseen data.

3.1 Experiment setup

For testing class imbalance we build models with different ratios of enthusiastic and ordinary employees where number of enthusiastic employees was always 237, while number of ordinary employees was ranging from 200 to 3000 in steps of 200 samples. Totally 16 different tests were performed with different ratios of enthusiastic and normal employees. At each test run, counts of true/false positive and true/false negative predictions were taken, from which precision, accuracy, recall and F-measure metrics can be calculated. To check which the best value of cost parameter C for a SVM, tests were performed for six different values of C as described by test procedure Figure 1. Test procedure with PCA included data transformation which was performed separately. By applying PCA we reduced dimensionality of data from 301 features to only two features where 52% of data variance vas retained. It seems that a lot of variance was lost, but according to visualized transformed data, classes with different labels are well separated, and SVM method easily finds a plane that perfectly separates data with different labels Figure 3.

```
C = [0.1,0.5,1,10,50,100];
for i = 1:length(C)
   numOfOrdEmployees = [100,200,400,...,2800,3000];
   for j = 1:length(numOfOrdEmployees)
      data = merge(OrdinaryEmployees[j], EnthusiasticEmployees);
      trainData = 0.7*data;
      testData = 0.3*data;
      model = SVMTrain(trainData(X),trainData(Y),C);
      prediction = SVMtest(testData(X));
      performance = (prediction == testData(Y));
   endfor
endfor
```

Figure 1: Test procedure description.

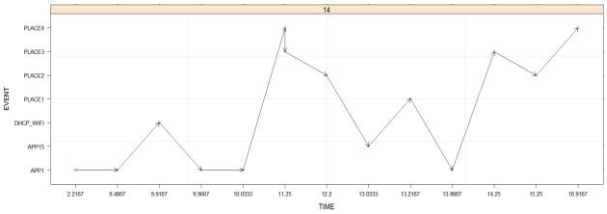

Figure 2: An example of trellis net.

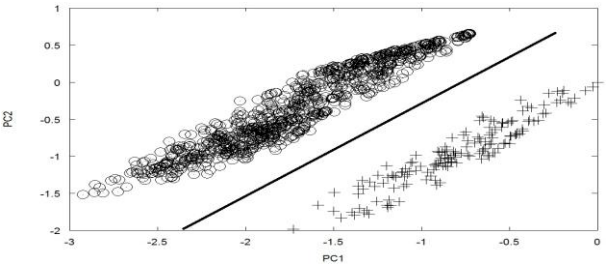

Figure 3: Visualization of data transformed with PCA with two principal components and SVM decision boundary.

3.2 Results

Test results for SVM method are very satisfying in terms of accuracy for test performed on original data sets and data sets with reduced dimension. As can be observed from Table 2 and Table 3, SVM method classified almost all test samples correctly, for the case where dimensionality reduction was performed prior to classification, all samples were correctly classified. For the case without dimensionality reduction at test run with 100, 200, 600 and 2200 negative samples only 1, 3, 1 and 1 sample were misclassified respectively Table 2. F-measure that combines both precision and recall in one single row number has values 1, except in cases where misclassified samples occur where F-measure has values 0.993, 0.981, 0.993 and 0.992 when number of negative samples was 100, 200, 600 and 2200 respectively. We observed that times required to train the models were significantly shorter at data where dimensionality reduction with PCA was applied. An average training time for SVM cost parameter value C=10 and 2000 negative samples was 2.2 seconds on reduced data, while average time at original data and

same number of negative samples and cost parameter was 8.3 seconds. In case of much higher number of training samples, data transformed with PCA might significantly reduce training time and by that computer resources. By using Bernoulli multivariate data model, classification for application of searching enthusiastic employees within a company can still be performed at high accuracy, even at the fact that no transitional or frequency properties of events were included in this data model. Data model on which we performed our tests requires a manual labeling, which in case where inspection of trellis diagrams is included might be a cumbersome process. For labeling 237 samples of enthusiastic employees three working days were spent, but in our occasion quick solution with simple data model proved to work quite well.

Table 2: Test results for different number of negative samples at C=10

Negatives	TN	FN	FP	TP	Precision	Recall	F-Measure	Accuracy
100	37	1	0	62	1	0.984	0.992	0.99
200	49	1	2	78	0.975	0.987	0.981	0.977
400	117	0	0	73	1	1	1	1
600	178	1	0	71	1	0.986	0.993	0.996
800	238	0	0	72	1	1	1	1
1000	300	0	0	70	1	1	1	1
1200	366	0	0	64	1	1	1	1
1400	417	0	0	73	1	1	1	1
1600	486	0	0	64	1	1	1	1
1800	533	0	0	77	1	1	1	1
2000	588	0	0	82	1	1	1	1
2200	665	0	1	64	0.985	1	0.992	0.999
2400	711	0	0	79	1	1	1	1
2600	783	0	0	67	1	1	1	1
2800	840	0	0	70	1	1	1	1
3000	897	0	0	73	1	1	1	1

Table 3: Test results for different number of negative samples on data with reduced dimension (dimensionality of subspace *k* =2) at C=10

Negatives	TN	FN	FP	TP	Precision	Recall	F-Measure	Accuracy
100	41	0	0	59	1	1	1	1
200	65	0	0	65	1	1	1	1
400	117	0	0	73	1	1	1	1
600	183	0	0	67	1	1	1	1
800	237	0	0	73	1	1	1	1
1000	296	0	0	74	1	1	1	1
1200	349	0	0	81	1	1	1	1
1400	424	0	0	66	1	1	1	1
1600	471	0	0	79	1	1	1	1
1800	538	0	0	72	1	1	1	1
2000	599	0	0	71	1	1	1	1
2200	652	0	0	78	1	1	1	1
2400	736	0	0	54	1	1	1	1
2600	776	0	0	74	1	1	1	1
2800	842	0	0	68	1	1	1	1
3000	918	0	0	52	1	1	1	1

4 Conclusion

In this paper we presented an example of users' behavior analysis based on machine learning method SVM, on IdM system log data, combined with log data from DHCP servers and access control systems log data. Use of SVM method on Bernoulli multivariate data model proved that SVM method can correctly classify data for an application of searching enthusiastic employees based on the predefined criteria. Dimensionality reduction enabled faster training and better performance on test data. For better understanding how well IdM system log data can be used for user behavior analysis, similar test must be performed on different data models, for instance on multinomial data model or on data where transitional properties of events are included. Finally different machine learning methods should be applied on those models, and different aspects of user behavior should be tested.

5 References

[1] Wei Wang, Xiaohong Guan, and Xiangliang Zhang. 2008. Processing of massive audit data streams for real-time anomaly intrusion detection. Comput. Commun. January 2008, p.p.58-72

[2] McCallum, Andrew, and Kamal Nigam. "A Comparison of Event Models for Naive Bayes Text Classification." Dimension Contemporary German Arts And Letters 752.1 (1998) : 41–48.

[3] Siddhartha Bhattacharyya, Sanjeev Jha, Kurian Tharakunnel, J. Christopher Westland, Data mining for credit card fraud: A comparative study, Decision Support Systems, Volume 50, Issue 3, February 2011, Pages 602-613

[4] Yinghui (Catherine) Yang, Web user behavioral profiling for user identification, Decision Support Systems, Volume 49, Issue 3, June 2010, Pages 261-271

[5] Lutz H. Hamel, Knowledge Discovery with Support Vector Machines, Wiley & Sons, 2009

[6] Andrew Moore, Support Vector Machines, content available at: http://www.autonlab.org/tutorials/svm.html

6 Acknowledgements

Operation was partly financed by the European Union, European Social Fund, Contract number: P-MR-08/43