Comparison of BP and SVM on SLA based Masquerader Detection in Cloud

Tej Bahadur Shahi,
Tribhuvan University Central Department of Computer Science and Information Technology
Kirtipur – Kathmandu, Nepal

Dadhi Ram Ghimire
Tribhuvan University Central Department of Computer Science and Information Technology
Kirtipur – Kathmandu, Nepal

ABSTRACT
Cloud computing is a new dynamics in the IT sector which imitates the pay per use modality of many commodity items. Organizations can use the new offerings in the IT field without significant capital investment. However, there is scrutiny in adoption of cloud computing in organizations as there are many economic, security and business risk associated with it. SLA between cloud service provider and cloud service user is a key in maintaining trust in the cloud services and achieve a common goal. Most SLAs focus on cloud computing performance while other issues don't get much attention. This study is oriented to build a masquerade detection system in cloud computing, based on the proposed SLA.

The new SLA contains additional security constraints than that found in traditional SLA such as length of temporal sequence, weight of each activities and the threshold weight of the temporal sequence. The performance analysis includes comparison of Back Propagation algorithm with Support Vector Machine (SVM). The detection rate and false alarm rate is observed and found that it can detect masqueraders well from the small set of training data with small false alarm rate.

Keywords
Support Vector Machine; Back Propagation; Masquerader Detection; Cloud Computing; Service Level Agreement.

1. INTRODUCTION
Cloud computing is a comparatively new paradigm following the shift from mainframe to client–server in the early 1980s. In this paradigm, details are abstracted from the users, who no longer have need for expertise in, or control over, the technology infrastructure "in the cloud" that supports them. Cloud computing describes a new supplement, consumption, and delivery model for IT services based on the Internet, and it typically involves over-the-Internet provision of dynamically scalable and often virtualized resources[1]. It is a byproduct and consequence of the ease-of-access to remote computing sites provided by the Internet. It is both a platform and type of application. Cloud Service Providers dynamically allocates configures servers as per need of a Customer [2]. Servers in the cloud are abstract to the users they may be physical machines or virtual machines. Cloud also includes other resources such as storage area networks (SANs), network equipment, firewall and other security equipments. Cloud Service Providers use mammoth data centers and powerful servers to host Web applications and Web services. Any person with a reasonable internet speed and a web browser can easily access the cloud computing services. It is Internet-based computing, whereby shared resources, software, and information are provided to computers and other devices on demand, like the telephone network. This frequently takes the form of web-based tools or applications that users can access and use through a web browser as if it was a program installed locally on their own computer. Cloud computing often appears as single points of access for all consumers' computing needs [3]. Commercial offerings are generally expected to meet quality of service (QoS) requirements of customers, and typically include SLAs (Service Level Agreement). SLA is agreement between cloud service provider and customer containing various terms and conditions of the cloud service that should be followed by both the parties. Main advantage of SLA is to gain common understanding of various issues including service levels and responsibilities of provider and customer. The stated issues and service levels in SLA depend on negotiation between provider and customer. Cloud computing is venerable to many security risks like Privileged User Access, Authentication, Data Security, Legal Issues and among others that need to be considered appropriately. Customers can trust cloud for their computing need if the Cloud service provider have the long-term viability. Due to these challenges cloud customers therefore need to institute mechanisms to measure and improve security of their information assets operating in the cloud. A masquerader is an intruder trying to impersonate a legitimate user. A masquerade attack occurs when an illegitimate user tries to impersonate a legitimate user; therefore, the masquerade user gets the privileges from the legitimate user account. The task of detecting masquerade users is not easy since the masquerade user has yield the name and password of a valid user (probably an administrator). However, detecting illegitimate users could be done if information about the behavior of the impersonate user is taken as a characteristic pattern which is valid only for this user. Early and effective intrusion detection is a critical factor in securing a cloud system.

2. PROBLEM DEFINITION
A user logs into cloud service account from a device that may or may not have been registered in the SLA agreement between that cloud user and the CSP (Cloud Service Provider) .The user's logging time, MAC address is recorded along with temporal sequence activities in a log file.

Let, S = (a1,a2,a3,a4,a5,a6,a7,a8……an) be the sequence of activities of a user during the user session. This user session consists of many temporal sessions of length *k* as shown in figure 2.1. If there are *n* activities in a session then there will be (*n*-k + 1) temporal sessions. Cloud service provider has assigned a dummy weight on each of the activities/events that is agreed on the SLA agreement. Such as:

- Deleting Contacts = 8
- Changing Passwords = 9
- Checking Mails = 5
- Transferring Data =10

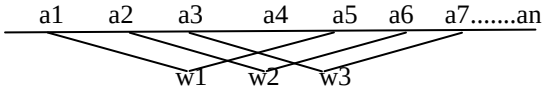

Figure 1: User activity graph

The SLA also includes the threshold weight i.e. maximum tolerable weight of a temporal sequence of length *k*, where threshold weight and *k* can vary. Temporal weights of temporal sequences are computed in succession one after another and compared against the threshold weight defined in SLA agreement. Based on the weight calculated in succession, user's logging time and MAC address of the user's logging device; the session is classified as normal or suspicious.

3. LITERATURE REVIEW

Cloud Computing is a new paradigm. Many works have been done in this field for intrusion and Masquerader detection. Hisham A.K and Fabrizio Baiardi in [4] created a Cloud Intrusion Detection Dataset (CIDD) including both knowledge and behavior based audit data using log analyzer and correlation system. Xijun Cheng and Juanjuan Chen in [5] modeled user interests in cloud for masquerade detection based on the how a user interacts with a computer system. This work is an attempt to detect masquerader on the cloud based on the SLA agreement which is probably first in this field. There has been a lot of research works that dealt the Masquerader Detection in computer and computer network and various methods are proposed.

3.1 Information Theoretic based Approaches

A simple compression based approach was first proposed by Schonlau et al. in [6], called the Compression method. Behavioral data from many users is used on the premise that data from the same user compresses quickly than the combined data from different users. The number of additional bytes needed to compress a testing block when appended to the training data at the end determined the score of that block. Lempel-Ziv algorithm [7] based method was applied for the UNIX command compression. A grammar inference algorithm called Minimum Descriptor Length (MDL) compress [8] was proposed by Evans et al. in 2007. This algorithm is based on the MDL principles from the Kolmogorov Complexity theory and Algorithmic Information Theory. This algorithm is used to model the activity of the legitimate user and the resulting grammar from the algorithm is used to detect masqueraders. This algorithm is said to have good detection rate but the actual data is scarce despite the best efforts to find them.

3.2 Time based Inductive machine

TIM (Time –based Inductive machine was originally developed as a general purpose tool with potential applications in many domains. TIM discovers temporal pattern [13] from observations of a temporal process, where the patterns represents highly repetitive activities [7] can be used for prediction with high accuracy.

3.3 Text Mining-based Approaches

A text mining approach based on the sequitur algorithm was proposed by M. Latendresse [14] in 2005. Sequitur algorithm is the method for inferring compositional hierarchical structures of a string by forming a context-free grammar [10] which detects repetition and factors it out of the string by forming rules of the grammar. A Sequitur grammar is generated for each user to find out the repetitive sequences of commands the user used and their associated frequencies. Legitimate user's profile is compared with the corresponding testing block recursively. This method achieved higher detection rate.

3.4 Hidden Markov Model based Approaches

A Hidden Markov Model (HMM) is a statistical model where the modeled system is assumed to be a Markov process i.e. the process is stateless and the next state of the process depends only on the current state. User Profiles are constructed [16] using the Hidden Markov Model, where the approach is different than the profile construction method. The legitimacy of the testing block is decided by the score obtained from every agent by means of a voting mechanism and a threshold value.

3.5 Naive Bayesian based Approaches

The Naive Bayesian is a probabilistic classifier; one of the prominent text classifiers due to its simplicity and inherent robustness to noise and their fast learning curve. Naive Bayesian approach is used with the "bag of words" model [11]. This particular model makes a profile of document based on number of times a word is used in the document i.e. word frequency, but it ignores the sequence information. In this model the profile of a document is updated online using Naive Bayesian approach [11]. This approach performed very well and achieved one of the best detection results.

3.6 SVM-based Approaches

Support Vector Machines are based on the concept of decision planes that define decision boundaries. A decision plane is one that separates between a set of objects having different class memberships. It classifies data by constructing a hyperplane in the n-dimensional feature space of training inputs; this hyperplane maximizes the margin between them [12]. One-class training for masquerader detection in [13] introduced this approach for masquerader detection.

4. IMPLEMENTATION AND RESULTS

In this work we are proposing a SLA with additional constraints than a traditional SLA. These additional constraints are assignment of dummy weight for each activity/event in the cloud service, length of the temporal sequence *k*, threshold weight of the temporal activities, device likely to be used to access cloud service and time frame within which a customer is likely use his cloud service account.

4.1 Assignment of Dummy Weight

There are so many activities that a user can do in a cloud service. In this work we are working with only 16 activities and these activities are categorized into two categories, normal and critical. Each activity in each category is associated with a dummy weight. We have assigned weight 1 to 8 to the activities in the normal category and 9 to 16 to the activities in the critical category.

4.2 Length of Temporal Sequence (k)

Selecting the length of the temporal sequence is crucial in classifying the user session and to create a user profile. A user session may have more than one temporal sequence. As per our 20 best knowledge, we are first to apply temporal sequence length *k* in a masquerade detection approach. We have chosen $k = 5$; as the minimum length of the temporal sequence and we increased the length of temporal sequence to 6 and 7.

4.3 Selection of Threshold weight of Temporal Sequence

Suppose (a1, a2, a3 ….an) is a user session with n activities in the session. There are $(n-k+1)$ temporal sequences in a *n* activity session. Choosing the right threshold weight for each tem-

poral sequence is crucial for classifying a user session.
Let (a1, a2, a3 ….ak) is a temporal sequence and total weight of temporal sequence be W_{tm} is calcluated with equation 4.1.

We have chosen threshold weight, W_{th} as the product of *k* and weight of first activity belonging to the critical category. This threshold weight covers all the *k* permutations involving the critical category.

$w_{tm} = \sum a_i$……(equation 4.1)

Comparison between W_{tm} and W_{th} along with other constraints determines whether a session is normal or suspicious.

$W_{th} = k \times C_{1st}$……(equation 4.2)

Where, k is temporal sequence and C1st is the weight of first activity of the critical category.

4.4 Device and Time Frame

In this SLA, cloud costumers specify at least two devices, from which they are most likely to log into their cloud service account. MAC address of these devices is recorded in the SLA agreement. We use 1 to simulate a user accessing from the device specified in SLA and 0 for those who are not. Similarly cloud costumers together with CSP agree upon on a time frame, within which frame he/she is most likely to use the cloud service. We use 1 to simulate a user accessing cloud service within the time frame specified in SLA and 0 for those who are not.

4.5 Training and Testing Data

We have used MSNBC data set to create user profile in this work. MSNBC dataset contains 989818 user sessions, ranging from single activity to 345 activities per session. The dataset is filtered by removing very short and very long sessions. The resultant dataset is then divided to make user profile of 50 users having 1000 session per user. Sample of MSNBC dataset is given below.

1
1 21 2
3 2 2 4 2 2 2 3 3
5
1
6 7 7 7 6 6 8 8 8 8
6 9 4 4 4 10 3 10 5 10 4 4 4

4.6 Back Propagation Training and Testing Data

Each user profile containing 1000 session is taken and MAC address (0 and 1) and time (0 and 1) is added randomly. The resulting sessions with MAC address and time factor is then fed to the classification algorithm to label the session, 1 for normal session and 0 for suspicious session. The data are then normalized using min-max normalization [14] to keep values between 0 and 1. The sample of training data is shown below.

1 0 0.75 0.45 0.75 0.45 0.45 0.75 0.45 0.75 0.45 0.75 0.45 0.75 0.8 0.8 0.8 0.75 0.75 0
1 1 0.15 0.2 0.15 0.8 0.8 0.8 0.8 0.8 0.8 0.8 0.8 0.8 0.8 0.8 0.8 0.15 0.15 0
0 1 0.3 0.3 0.6 0.6 0.6 0.6 0.2 0.2 0.2 0.2 0.6 0.7 0.7 0.6 0.45 0.45 0.45 0
0 0 0.5 0.5 0.5 0.5 0.5 0.75 0.75 0.75 0.75 0.5 0.5 0.5 0.5 0.5 0.5 0.5 0.5 0
1 0 0.75 0.75 0.75 0.8 0.8 0.8 0.8 0.8 0.8 0.8 0.8 0.8 0.8 0.8 0.8 0.8 0

4.7 SVM Training and Testing Data

Each user profile containing 1000 session is taken and MAC address (0 and 1) and time (0 and 1) is added randomly. The resulting sessions with MAC address and time factor is then fed to the classification algorithm to label the session, 1 for normal and -1 for suspicious session. A feature vector is constructed using the resulting session where there are altogether 18 features. Feature 1 to 16 represents the activities whose value is the number of occurrence of particular activity in a session. Whereas feature 17 and 18 has binary values representing the true or false condition of MAC address and time frame. The training data sample is shown below.

-1 13:9 14:6 17:1 18:0

-1 2:4 7:5 13:6 17:1 18:1

1 2:3 4:5 7:4 8:8 17:0 18:1

-1 7:1 13:12 14:3 17:0 18:0

-1 1:2 5:2 6:7 15:4 17:1 18:0

1 1:7 2:4 4:1 6:1 11:1 16:1 17:1 18:1

4.8 Classification Algorithm based on SLA Agreement

Input:

***k**, no of temporal activities wth, threshold weight*

***n**, no of events\activities done by the user done user during the user session.*

***add**, MAC Address of the user's login device*

***sadd** , MAC address of allocated device of particular user in SLA*

***ltime** ,users logging time*

***stime**, time frame of user in SLA*

***Output**: classification of the cloud login session into two classes*

Begin Procedure

//compute the temporal weights

***Repeat** for j=0; j<=n-k+1 with single increment of j*

w=0;

***Repeat** for i=0 to i<=k-1*

w = w + a[j+i]; //events with their respective weight

***End** for.*

// compare time, address and weight to classify session

If (add = = sadd && ltime = = within (stime) &&

*w<= w_{th}) , classify as **normal session***

else if(add = = sadd && ltime = = within (stime)

*&& w >= w_{th}) ,classify as **suspicious session***

else if(add != sadd && ltime = = within (stime)

*&& w <= w_{th}) classify as **normal session***

else if(add != sadd && ltime = = within(stime)

*&& w >= w_{th}) classify as **suspicious session***

else if(add = = sadd && ltime != within(stime)

*&& w >= w_{th}) classify as **suspicious session***

else if(add = = sadd && ltime != within(stime)

*&& w <= w_{th}) classify as **suspicious session**.*

***End** for.*
End

4.9 Tools used

PHP

In this work we have used PHP to implement different algorithms. The two algorithms SVM and Neural Network are implemented. Back-propagation method is used to train the neural network.

4.10 SVM Light Tool

In this work SVM Light [15] tool is used to implement SVM. SVM Light is a C program by Thorsten Joachim's that implements a support vector machine. SVM Light is used for the problem of pattern recognition, for the problem of classification, for the problem of regression and for the problem of ranking the function. It includes two efficient estimation methods for both error rate and precision/recall. One is svm_learn for training and other is svm_classification.

5. RESULTS

The detection rate and False alarm rate for different 20 user with the value of k=5 is shown in table 1. Similar experiment were done for the value of k=6 and k=7 and the aggregate result is shown in table 2.

Table 1: Detection rate and False alarm rate of each user (*k*=5)

User	Back Propagation		SVM	
	Detection rate (%)	False Alarm Rate (%)	Detection rate (%)	False Alarm Rate (%)
User1	80	0	89	8
User2	83	0	92	1
User3	87	1	89	2
User4	90	0	89	3
User5	87	2	87	1
User6	90	7	87	2
User7	90	6	88	2
User8	92	6	89	2
User9	83	5	81	4
User10	82	0	88	5
User11	85	5	92	1
User12	83	0	90	2
User13	86	3	82	2
User14	84	8	91	3
User15	84	0	91	4
User16	89	0	88	2
User17	83	0	89	3
User18	84	0	85	4
User19	84	0	87	0
User20	84	4	82	5

Table 2: Detection rate and False alarm rate of BP and SVM

K	Back Propagation		SVM	
	Detection Rate (%)	False Positive Rate (%)	Detection Rate (%)	False Positive Rate (%)
5	86.04	2.02	87.8	2.8
6	87.34	0.9	88.96	2.83
7	89.42	2.28	90.06	2.94

6. CONCLUSION

With Cloud Computing becoming a popular term on the Information Technology (IT) market, security and accountability has become important issues to highlight. In our research we focused on security risks with Cloud Computing and the associated services. The main concern of this work was to assess the risk of cloud computing and to implement a new methodology in detecting masqueraders involving SLA agreement between cloud computing users and CSPs. This work also proposed a new SLA by mutual consent of CSP and cloud costumers, which will be helpful in detecting masquerades, insider attack. We also compared the efficiency of Back Propagation Algorithm with SVM; we found out that SVM has a better detection rate with a higher false alarm rate compared to the Back Propagation Algorithm.

7. ACKNOWLEDGEMENTS

Authors would like to thank Dr. Shashidhar Ram Joshi, Professor of Computer Science at institute of Engineering, Pulchwok and Prof. Dr. Tanka Nath Dhamala for their supervision during the completion of this work.

7. REFERENCES

[1] "The NIST definition of Cloud," 2011(NIST Special Publication 800-145) available at http://csrc.nist.gov/publications/PubsSPs.html#800-145.

[2] R. Kaur, "Cloud computing," international journal of computer science and technology, vol. 2,2011,PP 373-381.

[3] G. Christos A. Yfoulis, "Honoring SLAs on cloud computing services: a control perspective," 2009, available at http://www.techrepublic.com/resource-library/whitepapers/honoring-slas-on-cloud-computing-services-a-control-perspective/#

[4] F. Baiardi. H. A. Kholidy, "CIDD: A Cloud Intrusion Detection Dataset For Cloud Computing and Masquerede Attacks," *IEEE:NInth International Conference Of Information Technology-New Generations*, 16-18 April 2012, PP 397-402 doi:10.1109/ITNG.2012.97.

[5] X. Cheng,J. Chen, "Modeling User Interests Based on Cloud Model for Masquerade Detection," *International Conference on Computational Intelligence and Software Engineering* , Wuhan, 11-13 Dec. 2009 ,PP 1-4, 2009 doi:10.1109/CISE.2009.5366294.

[6] M. Schonalu, Dumouchel, W.,Karr, "Computer Intrusion:Detectiong masqueraders," Statistical Science, 2001,Vol. 17 PP. 1-17.

[7] T. A. Welch, "A Technique for high performance data compression" IEEE Computer Vol. 17 Issue 6, 1984 PP. 8-19, doi: 10.1109/MC.1984.1659158.

[8] S. E. Evans, E. Markham, S. Impson, J. Laczo, "Mdlcompress for intrusion detection: Signature inference and masquerade attack.," *Military Communications Conference,* 2007.

[9] A. R. Pankesh Patel, Amit Sheth, "Service Level Agreement in Cloud Computing," 2010.

[10] I. H. W. Craig G. Nevill-Manning, "Identifying Hierarchical Structure in Sequences:A linear-time algorithm," Journal of Artificial Intelligence Research Vol. 7 Issue 1, July 1997 PP 67-82.

[11] K. Nigam, A. McCallum, "A Comparison of Event Models for Naive Bayes Text Classication," In *AAAI-98 Workshop on Learning for Text Categorization*, 1998.

[12] C. J. C. Burges, "A tutorial on support vector machines for pattern recognition.," Data Mining and Knowledge Discovery Vol. 2, Issue 2 1998, pp 121-167 doi: 10.1023/A:1009715923555.

[13] K. Wang, S. Stolfo, "One-Class Training for Masquerade Detection," In: *ICDM Workshop on Data Mining for Computer Security (DMSEC)*, 2003

[14] M. Latendresse, "Masquerade Detection via Customized Grammars," in: Second International Conference, DIMVA 2005, Vienna, Austria, July 7-8, 2005, PP. 141-159 doi 10.1007/11506881_9

[15] Thorsten Joachims, "Learning to Classify Text Using Support Vector Machines" Dissertation, Kluwer, 2002.

[16] Maximiliano Bertacchini, Pablo I. Fierens, " A Survey of Masquerader Detection Approaches"