Detecting Anomalies in User Profiles in the Cloud via Machine Learning

Clarissa Wong[1,2], Trishita Tiwari[2], Aditya Narayan[2], Prof. Ayse K. Coskun[2]

Algonquin Regional High School, 79 Bartlett Street, Northborough, MA 01532[1]

Electrical and Computer Engineering Department, Boston University, 8 St. Mary's Street, Boston, MA 02215[2]

Abstract

Cloud computing provides a network of remote servers hosted on the Internet and it allows users to run their applications and store their information on this network, in contrast to using local personal computers for computing and storage (Figure 1). As cloud computing is becoming increasingly popular for almost all domains of computing (e-commerce, healthcare, mobile, finance, science, and others), the vulnerability of its users to cyber attacks rises. In one type of attack, an attacker might take advantage of a cloud user's resources to run illegal or computationally-intensive applications, inducing cost increases for the compromised user. This project proposes a potential predictive solution by detecting such attacks using statistical and machine learning techniques. Typically, users exhibit distinct patterns in terms of their resource usage; thus, a break in the user's resource utilization trends could be indicative of a cyber attack. We implement algorithms such as k-Nearest Neighbors or min-max based and percentile-based thresholding to construct a model based on the user's patterns. We then test our method with two sets of anomalous data, which were simulated and collected from virtual machines on the Massachusetts Open Cloud. In the first data set, the attacker continuously runs a computationally-intensive application in the background, while in the second case, the attacker runs the same application intermittently. Our most accurate predictive analytics approach was percentile-based thresholding, which was able to accurately identify over 96% of the anomalies. Such approaches could be applied to an actual cloud context in order to detect user anomalies and improve security by alerting on abnormal user activities.

Monitoring Data in the Massachusetts Open Cloud

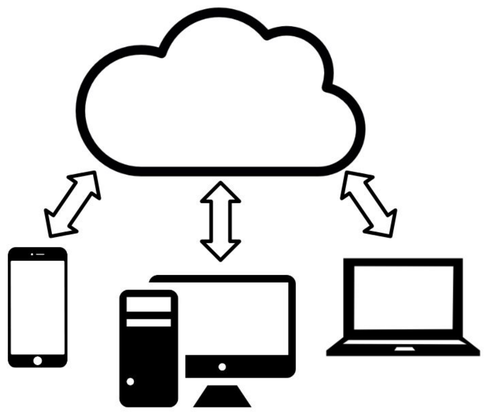

Figure 1. Cloud computing diagram

- The Massachusetts Open Cloud (MOC) is a public cloud platform that is a collaboration between government, universities, and industry and is primarily operated by Boston University. [1]
- MOC's monitoring infrastructure can track various user metrics, such as virtual CPU (vCPU) usage, network traffic, number of virtual machines (VMs), etc.
- In this project, the data analyzed was collected from a VM running on the MOC using custom scripts and reflects a synthetic user's vCPU usage.

References

[1] Massachusetts Open Cloud - An Open Cloud Exchange Public Cloud. (n.d.). Retrieved August 1, 2017, from https://massopen.cloud/

[2] Samuel, A. L. (2000). Some studies in machine learning using the game of checkers. *IBM Journal of research and development*, *44*(1.2), 206-226.

[3] Altman, N. (1992). An Introduction to Kernel and Nearest-Neighbor Nonparametric Regression. *The American Statistician, 46*(3), 175-185. doi:10.2307/2685209

Methodology

- Users typically display distinct patterns in terms of their resource usage.
- In the example graph below, the user exhibits high CPU usage during the day when he is running applications and low CPU usage at night when the VM is idle (Figure 2).
- Because some of such patterns are predictable, we can apply statistical and machine learning techniques to detect anomalous behavior.

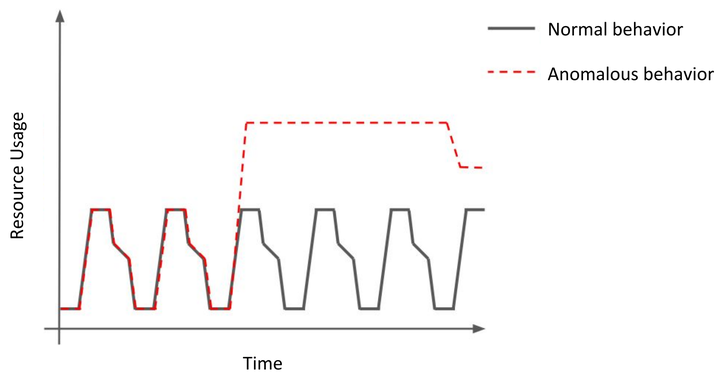

Figure 2. The user exhibits a distinct trend in resource usage, as indicated by the grey line. When an attacker runs an application on the user's resources, the dotted red line shows a sudden break in the trend that indicates an attack.

- We implement the following techniques with Python libraries to construct a model based on the user's patterns. We then test our methods with two sets of anomalous data.
  - In min-max thresholding, we set the threshold as 1.95 times of the maximum normal data point. In our experiment, any number that exceeds this threshold is labeled as anomalous.
  - Percentage-based thresholding is a statistical technique that sets a percentile threshold used to classify numbers in a data set. In our experiment, the threshold is the value of the 95th percentile of the normal data.
  - Machine learning allows a user to teach a computer by inputting training data with examples of normal and anomalous data. When the user inputs testing data, the computer can then apply this knowledge to predict instances of normal or anomalous data [2] (Figure 3). We use a type of machine learning algorithm called the k-Nearest Neighbors (kNN) algorithm, which is a method of classifying data points based on their location in the context of other data points [3] (Figure 4).

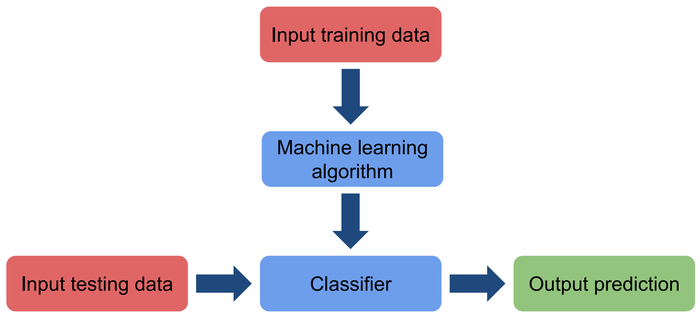

Figure 3. Machine learning diagram

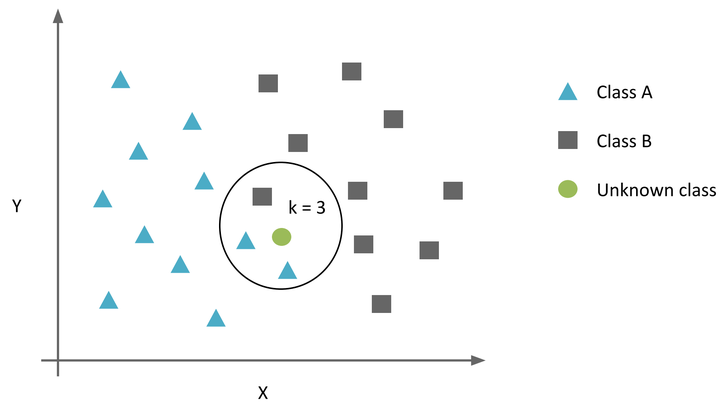

Figure 4. The uncategorized point belongs to Class A or Class B. The kNN algorithm uses Euclidean distance to calculate which points are closest to the uncategorized point. If k = 3, the three closest points are shown to be two Class A points and one Class B point. Thus, the uncategorized point is in Class A.

Acknowledgements

I would like to thank Prof. Ayse Coskun and PEAC Lab Research Group members for offering insightful input throughout this project and guiding me through my research experience. I would also like to thank Boston University's Department of Electrical and Computer Engineering and the Boston University RISE Internship Program for this opportunity to pursue research in a laboratory setting.

Anomalous Test Cases and Results

- We tested the statistical and machine learning techniques on two anomalous test cases (Figure 5 and Figure 6).
- Examples of the user's normal behavior trend are shown in areas of the graph *not highlighted* in red.

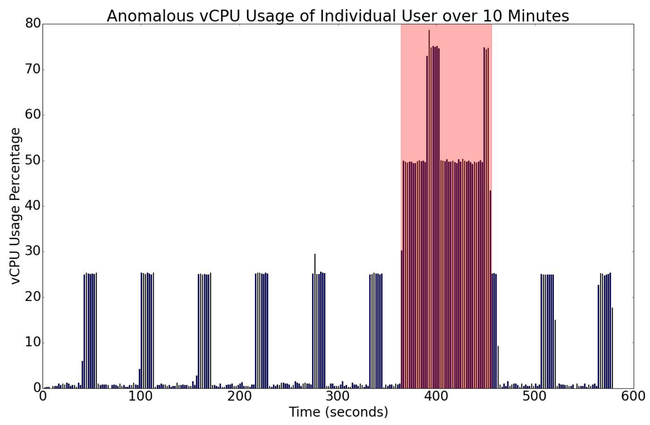

Figure 5. In the first anomalous data set, the sudden spike in vCPU usage is indicative of an attacker running a computationally-intensive application continuously. The data points classified as anomalous are highlighted in red.

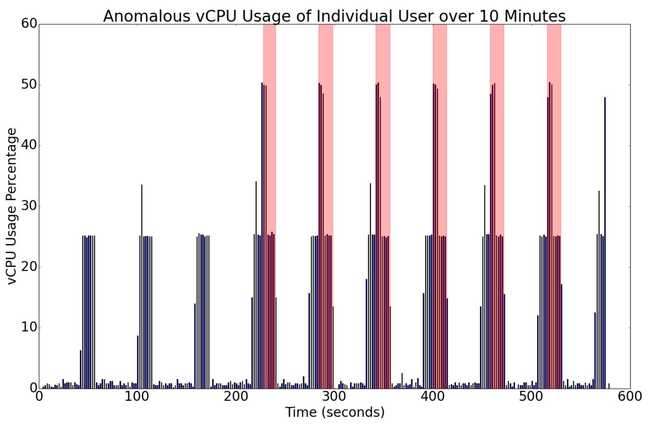

Figure 6. In the second anomalous data set, an attacker ran a computationally-intensive application intermittently. The data points classified as anomalous are highlighted in red.

Figure 7. Average results of both anomalous data sets

Method	False Positives	False Negatives	Accuracy of Detection
Min-max thresholding	13.67%	0.69%	85.64%
Percentile-based thresholding	3.29%	0%	96.71%
kNN algorithm	0.56%	8.43%	91.01%

Key Takeaways

- As cloud computing is becoming increasingly popular in private and public settings, the vulnerability of its users to cyber attacks rises.
- We use statistical and machine learning techniques (min-max thresholding, percentile-based thresholding, kNN algorithm) to understand trends in a user's normal resource usage behavior; we then use these models to detect any user anomalies that could indicate an attack.
- In the three methods we use for identifying anomalous data, percentile-based thresholding yielded the highest accuracy (96.71%), followed by the kNN algorithm (91.01%) and min-max thresholding (85.64%).
- Although these approaches were unable to flag all anomalous data points, they were able to detect all anomalous "phases" in both data sets. In a real cloud context, these methods could be used to alert the user or system administrators on abnormal activity.