UNIVERSITY OF HOUSTON

Detecting Intruders in a Host Computer Using a Behavioral Graph Approach

Stephen Huang
University of Houston

Zechun Cao
Texas A&M University-San Antonio

TEXAS A&M UNIVERSITY SAN ANTONIO

Introduction

Intruders inside a computer system can cause damages such as data breaches, embezzlement, and disruption of operations. Because of their malicious intent, adversaries exhibit different cyber behaviors than normal users. This study aims to develop a behavioral approach for Host Intrusion Detection System (HIDS) to prevent data breaches. To achieve this aim, we propose to use a graph to model a computer user's behavior and use the behavioral difference to identify intruders from normal users. We validated our hypothesis with an existing user activity dataset by adopting an anomaly detection approach. Our hypothesis was validated by the experiment based on one-class Support Vector Machine model trained on only normal users' data.

> Can we detect intruders by their behavior?
> How do we model the user behavior?

Hypothesis

Adversaries exhibit different cyber behaviors than normal users, which can be used for detection.

Methods

Using file access logs of users, we aim to demonstrate the following:

- model the user behavior using a graph,
- extract attributes from the graphs for machine learning,
- set up an anomaly detection based on the attributes,
- using a one-class Support Vector Machine as the detection algorithm
- identify intruders from normal users.

Dataset

We use an existing dataset called Windows-Users and -Intruder Simulation Logs (WUIL) with 76 normal Windows users in 9 different roles. The normal user data were collected, but the intruder data were simulated. The dataset uses three file access methods: manual search, using a search tool, and using a predefined script. The log contains date/time stamps, depth in the directory tree, and the complete path from the root for each file access.

Feature Extraction

We extracted several features from the graphs and tested them individually on their ability to classify the users correctly. The top five features listed below are combined for the final experiment.

- Number of Vertices (NV) of the graph.
- Graph Connectivity (GC): The highest degree of all vertices divided by the number of vertices (NV). The degree of a vertex is the sum of its in-degree and out-degree.
- Longest Segment with Degree-2 Nodes (LSD2): The length of the longest segment in a trace of degree-2 nodes.
- Average Length of Shortest Path (ALSP): The average length of the shortest paths between all pairs of vertices.
- Longest Duration Maximal Clique (LDMC): Each vertex is associated with a duration attribute, the amount of time a user stays on the file. LDMC is the largest sum of durations from all the nodes of a maximal clique.

Behavior Discrepancy

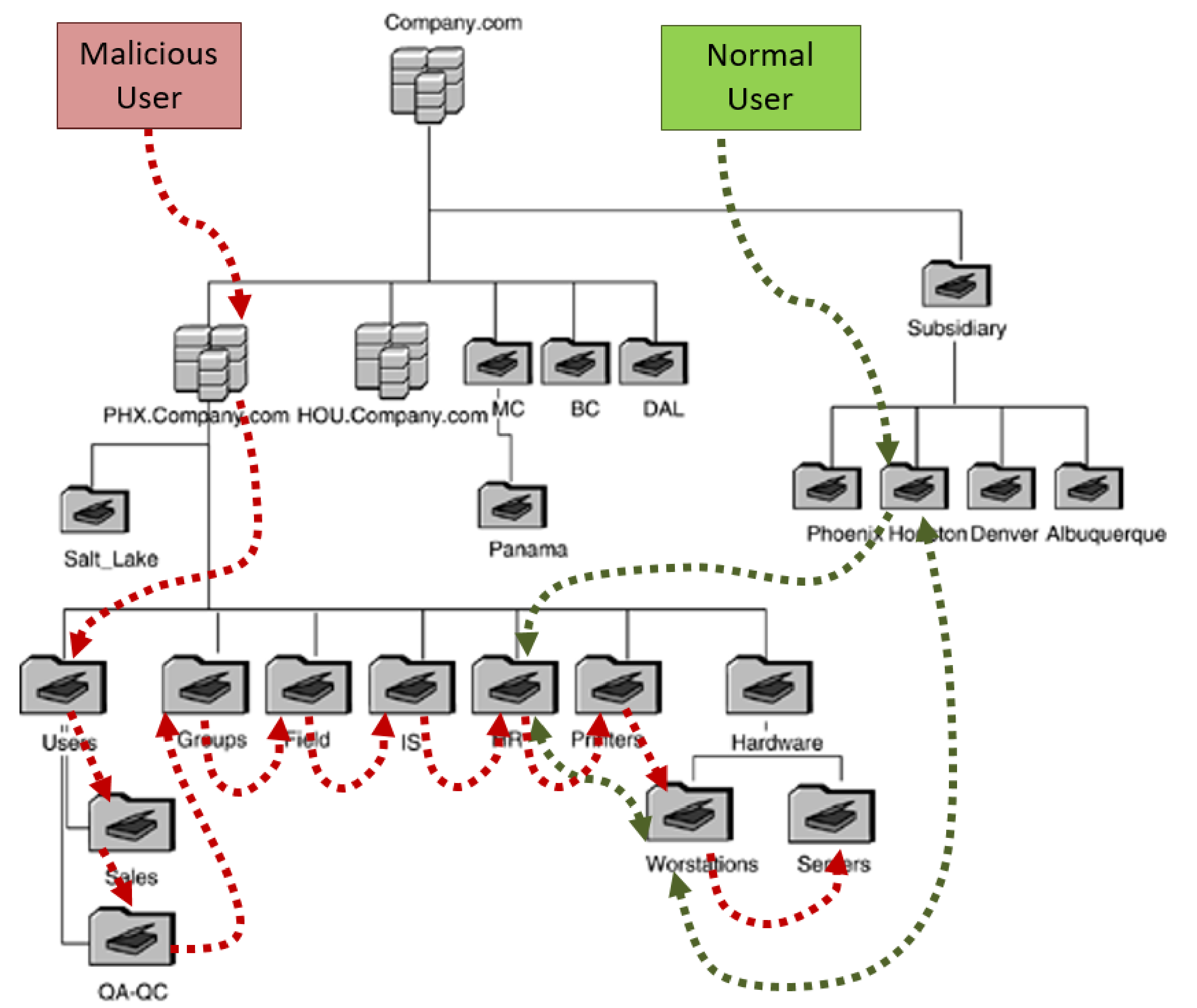

Figure 1. As depicted, an adversary's file-accessing behavior may differ from a normal user's.

Graph Model

Definition 1 (File Access Log) Given an integer $n \in \mathbb{N}$, a file access **log** $L = (f_1, f_2, ..., f_n)$ is a list where f_i $(1 \leq i \leq n)$ is the file identifier in the file system.

Definition 2 (Trace) Given a file access log $L = (f_1, f_2, ..., f_n)$, we define an equivalence relation $f_i \equiv f_{i+1}$ $(1 \leq i < n)$, if f_i's file identifier is identical to f_{i+1}'s file identifier. A **trace** is defined as $T = (r_1, r_2, ..., r_m)$, where m≤n, and r_i $(1 \leq i \leq m)$is the first element of each equivalence class.

Definition 3 (Graph) Given a trace $T = (r_1, r_2, ..., r_m)$, we define its associated **graph** as a directed **graph** $G_T = (V, E)$, where $V = \{r_i | r_i \in T\}$ is a set of vertices, and $E = \{(u, v) | (u, v) = (r_i, r_{i+1}), \forall i, 1 \leq i \leq m-1, r_i \in T\}$ is a set of edges.

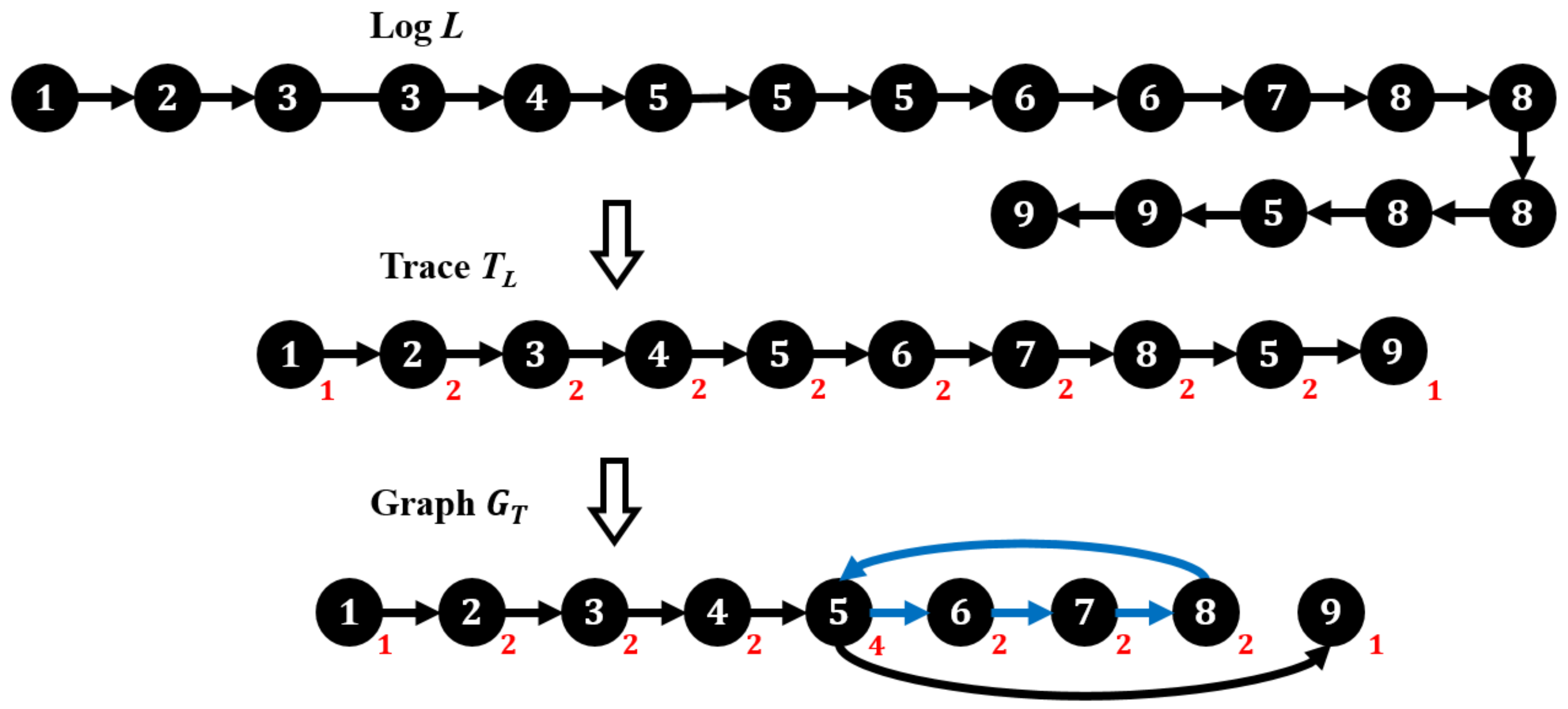

Figure 2. In this example, most trace elements have a degree (in red) of 2, with a few exceptions. There is only one cycle(in blue) in the graph.

Machine Learning

The dataset we used contains simulated behaviors of intruders, which we are unsure if they are similar to intruders. This study uses unsupervised machine learning methods on the normal user data during the training phase, i.e., one-class machine learning methods. Thus, we eliminated the concern about simulating adversary behavior. However, the simulated attack data were used in the testing part to derive the accuracy. Two unsupervised machine learning methods were used: **Support Vector Machine (SVM)** with three different parameter settings and **Isolation Forest**. We used a 10-fold cross-validation technique for the four experiments to guarantee that the proportions of data from attackers and typical users are equal in each testing split.

Testing Accuracy

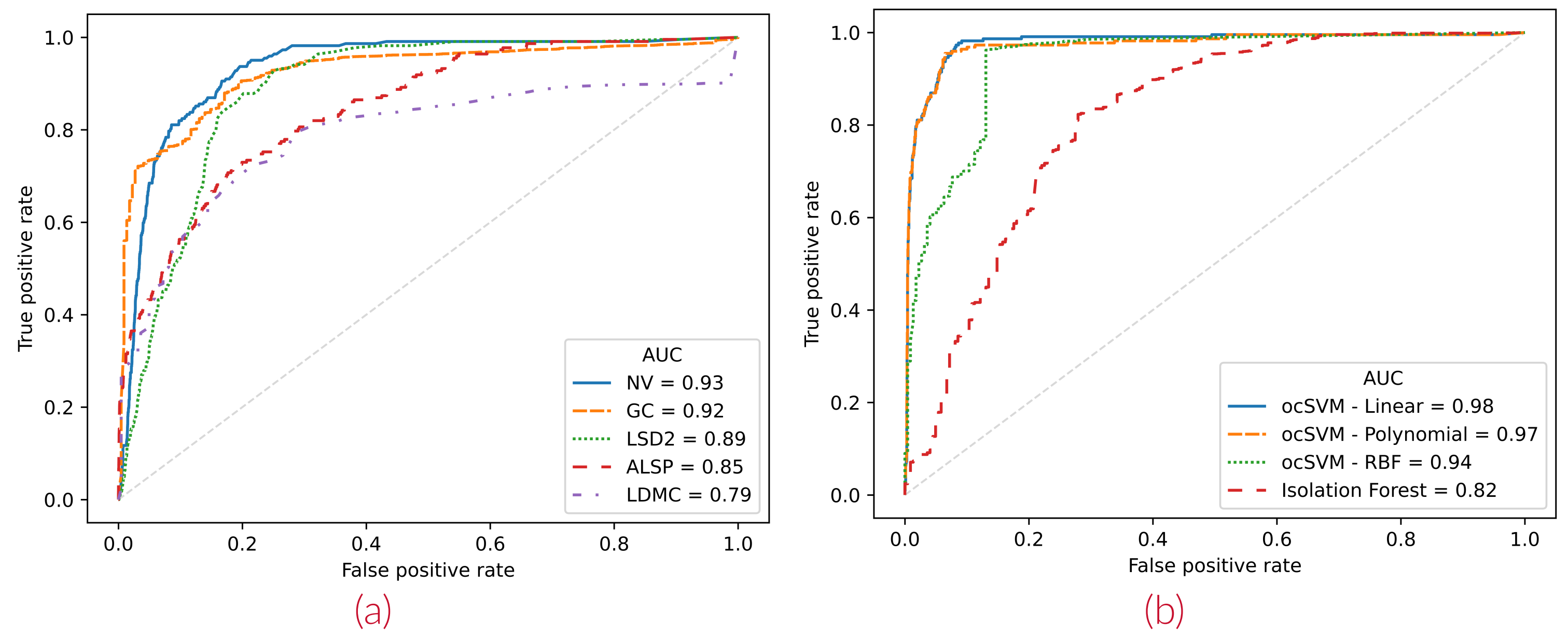

Figure 3. Performance evaluation of individual feature and anomaly detection models with combined features preprocessed. (a) A comparison of ROC curves of the individual features from the graphs. (b) ROC curves with the top five features combined using isolation forest and one-class SVM with linear, polynomial, and RBF kernels.

Conclusions

This paper proposed a behavioral graph model to describe user behaviors in a host computer system. Several features were extracted from the graphs as input to machine learning algorithms. We tested several algorithms on a dataset consisting of attack and normal users. One-class SVM algorithm with linear kernel shows high intrusion detection performance — 87% TPR at 5% FPR, with 0.98 AUC value of its ROC curve. The study demonstrated that the graph model is capable of capturing user behaviors. Other logs, such as system-call logs, may also be used in addition to the file-access logs to obtain more accurate results. For this study, we limited the files to what each user owns, which are unique to each user. It would be interesting to see if the result will be better if we include user behavior on system files if we can find a suitable dataset.

Acknowledgement

This work was supported in part by the National Science Foundation (1950297), the US Department of Education (P200A210119), and the National Security Agency (H98230-22-1-0323). The authors would like to thank the students for preparing this poster.

Reference

This poster is based on a paper presented at a conference: Zechun Cao and Shou-Hsuan Stephen Huang "Host-Based Intrusion Detection: A Behavioral Approach Using Graph Model" 18^{th} International Conference on Information Assurance and Security, December 13-15, 2022 (Vol. 647, pp. 1337–1346). Springer Nature Switzerland. https://doi.org/10.1007/978-3-031-27409-1_122.