Using Identity Credential Usage Logs to Detect Anomalous Service Accesses

Daisuke Mashima  
Georgia Institute of Technology  
266 Ferst Drive  
Atlanta, GA  
mashima@cc.gatech.edu

Mustaque Ahamad  
Georgia Institute of Technology  
266 Ferst Drive  
Atlanta, GA  
mustaq@cc.gatech.edu

ABSTRACT

For e-commerce companies providing online services, fraudulent access resulting from theft of identity credentials is a serious concern. Such online service providers deploy a variety of defenses and invest significant time and effort to the analysis of a large amount of log data to detect malicious activities and their impact. To reduce this burden, we explore the effectiveness of an anomaly detection based approach that relies on identity credential usage log records. More specifically, we use an anomaly-based metric to score the risk of each identity credential usage, e.g., a login request. Scores are determined based on categorical attribute values extracted from log records, such as timestamps. We utilize actual log data of login attempts to a university portal to evaluate the effectiveness of this approach. Our approach can work in conjunction with intrusion or fraud detection systems. It is also possible that stronger authentication can be required only when the risk score is high, which can help balance security and usability demands.

Categories and Subject Descriptors

K.6.5 [Security and Protection]: Unauthorized Access

General Terms

Algorithms, Experimentation, Measurement, Security

Keywords

Identity Theft and Misuse Detection, Risk Scoring, Online Access Log Monitoring

1. INTRODUCTION

The theft and misuse of online identity are major concerns for online service providers. According to Federal Trade Commission’s report released in February, 2009 [1], identity theft is ranked in the first place with more than 300,000 complaints, and the number shows a significant increase from the previous year. Such identity theft and misuse not only victimize consumers but they also negatively impact companies and financial institutions [2]. Thus, the increase in the number of identity theft cases requires that online service providers pay more attention to detect and prevent accesses by malicious sources to minimize financial loss and also to protect legitimate customers.

To counter such threats, many companies providing online services record information about the usage of users’ online identity credentials to detect malicious access. Then, if suspicious identity usage records are identified, they are further investigated. Currently, in many companies, including some that we work with, monitoring and investigation of identity usage logs requires significant human effort. Thus, a great amount of time and money need to be invested to handle the enormous amount of log data. In addition, analysis of log data is not usually done in real-time basis, which implies that there is a window of vulnerability. To improve this situation, an automated mechanism to monitor identity credential usage is desired. This can help identify suspicious requests that need greater scrutiny by human specialists.

In this paper, based on analysis of login attempt log records of actual online services, we present an anomaly-based risk scoring mechanism for identity credential usage records. The scores are computed based on deviation from the corresponding user’s normal usage pattern. The output of our system is just a numeric score, which can be utilized in various ways afterwards, indicating how suspicious each identity usage is, and additional description for the resulting score is also included in order to support investigation. Such scores can be used as input to fraud and intrusion detection systems or can be used as part of user authentication systems to decide when additional factors for authentication are necessary. Furthermore, our scoring scheme relies on generic categorical data, so it can be widely applicable regardless of types of services. In addition, since our scheme can be used to monitor real-time usage of identity credentials, it can shorten the window of vulnerability in the event identity credentials are lost or stolen.

This paper is organized as follows. In Section 2, we will explain the identity usage log data sets used in our research and discuss key design principles for our system. Then, in Section 3 we describe the mechanisms used to manage user profiles and calculate scores based on them. We will present the results of experiments and evaluation in Section 4. Related work is discussed in Section 5. In Section 6, we will conclude the paper with future work.
2. APPROACH

To develop a risk scoring scheme that can be used with identity credential usage, we first looked into two real data sets. One is a set of log records of login requests to BuzzPort, Georgia Tech's campus portal site, via which students and faculty can access a number of online systems, such as a course registration system, web mail, and so on [3]. Each record in this data set contains only a login timestamp and anonymized user ID. The other is a set of records of login attempts to a portal of an e-commerce company. Each record in this set consists of user ID, login timestamp, IP addresses, and other information that comes from user registration records, such as an organization name. As can be seen, available information is quite limited compared to what other fraud detection schemes expect. This is due to the fact that, at the time of identity credential usage, for instance a login request with some authentication credential, often only limited information is available. Since our goal is to help identify potentially malicious accesses to online services even in this sort of setting, a practical approach must rely on only such available information.

One of the common characteristics of these data sets is that neither of these data set records were labeled as legitimate or malicious, i.e. we did not know the ground truth about malicious activities. This implies that we can not use mechanisms that rely on such knowledge or distributional assumption, such as discriminant analysis. In reality, it is rare that we are given a sufficient amount of real attack records, so we believe this is a practical limitation that any widely applicable scheme must deal with.

In intrusion detection domain, there are two general types of techniques: signature-based misuse detection and anomaly detection. In our context, signature-based schemes are not expected to work well. Unlike intrusion detection systems, the login request patterns of users are highly diverse, thus it is difficult to create attack signatures that can cover many users. In this sense, establishing scoring criteria for each user, just as anomaly detection schemes do by utilizing each user’s normal profile, is reasonable.

A number of anomaly detection schemes utilizing a sequence of events to profile normal behavior are proposed in intrusion and fraud detection areas. For instance, Warrender [4] proposed a way to detect intrusions based on the sequence of events (system calls). However, such mechanisms have limited applicability in our context. Specifically, when monitoring just login attempts to a system, as is the case of our data sets, there is only a single type of event which occurs at lower frequency. Although intervals between consecutive events and sequence of the intervals could be used for profiling, based on our preliminary analysis of the data sets, we found that a human user's access pattern almost never follows exactly the same pattern. Moreover, access frequency varies significantly among users, so determining the reasonable granularity of interval is difficult. Although duration of events could be a possible parameter for profiling, some types of credential usage, for example a login attempt or usage of a credit card number, do not have any duration.

Furthermore, owing to the same reason, mechanisms relying on transition probabilities among events, such as one discussed in [5], will not be effective either.

As explored in [5], utilizing various attributes of an event is useful. The authors of [5] used the CGI parameters or length of requests to detect anomalous HTTP accesses. In our context, each parameter in an identity usage log record, such as timestamp or a requester’s IP address, can be considered as attributes of an event. Then, as proposed in [6], we can build a normal profile by keeping track of the occurrence frequency of each attribute value and then determine risk scores based on the relative frequency of an attribute value that an observed identity usage has. Such a scheme does not suffer from the problem mentioned earlier and can cover data sets that are investigated by us.

Based on these observations, we propose an anomaly scoring scheme that can cover online identity credential usage. Assuming that a normal usage pattern shows some specific characteristics in terms of the frequency distribution over appropriately chosen categorical attributes, we use such frequency information for scoring. For instance, when a user accesses the system (e.g., logs into it) on specific days of week, using “day of week” as an attribute to define a user’s profile is reasonable. This will result in 7 categories corresponding to each day of week. The details will be explained in Section 3.2. We consider only categorical parameters as input to the scoring algorithm. The primary reason for this is that many attributes can be represented as categorical values. For example, day of week and hour of day derived from a login timestamp and source IP address are all categorical data. In addition, other types of data, such as ordinal measures, can be converted into categorical measures by appropriately defining bins. Thus, we believe this approach does not limit the applicability of our scheme. In addition, when a user’s normal usage pattern shows more peculiar and distinctive traits, scores derived from such a profile should be considered more significant, compared to a case in which a user’s identity usage does not exhibit any specific pattern.

Our scoring scheme includes a weighting scheme to ensure this, which is discussed in Section 3.3.

3. ANOMALY-BASED RISK SCORING

3.1 User Profile Management

In our scoring scheme, a user profile is built as a frequency distribution over categories of an attribute. Every time identity credential usage is observed, the frequency counts in corresponding categories are incremented by 1. For instance, if we use hours of day as categories, a user profile consists of 24 categories (bins). If identity usage is observed at 1:30am, for example, the frequency count of the second category, which indicates 1-2am, is incremented. Multiple profiles can be defined for each user, such as hour-of-day profile and day-of-week profile. In this case, both profiles are updated.

However, when considering the long-term operation, such a profiling scheme could have a problem. For example, it is not unlikely that the user’s identity usage pattern changes gradually over time. In such a case, the frequency distribution would approach a flat shape, which is less distinctive. To prevent that effect, we can apply data aging, which makes older observations less effective and recent ones more influential. Such data aging can be implemented by periodically, e.g., once a month, multiplying some decay factor with all the frequency counts in profiles. For instance, if the decay factor is 0.5, the frequency counts in user profiles get halved once the data aging is run. After that, the counts are again incremented by 1 as described above when identity credentials are used. Thereby, the influence of recent observations is more significant than older ones.
3.2 Base Score Computation

A base score, which is the primary part of our anomaly-based risk score, is computed based on the relative frequency of a category corresponding to an observed attribute value. For instance, in the example using hours of day as categories mentioned in Section 3.1, the relative frequency of the second category is used for the score computation. Relative frequency of each category can be computed as follows where \( C \) is the set of all categories in a profile.

\[
\text{RelFreq}_c = \frac{\text{FreqCount}_c}{\sum_{c \in C} \text{FreqCount}_c} \tag{1}
\]

After data aging is applied as explained in Section 3.1, the frequency counts might not be integer values, but it does not matter how to compute relative frequency. By using relative frequency, we can compare the frequency distribution as a discrete probability distribution, which is also an important property in the weight calculation shown in Section 3.3.

From this relative frequency, a base score for an observed identity usage can be computed using the following equation.

\[
\text{BaseScore} = -\log_2(\text{RelFreq}_c^A) \tag{2}
\]

where \( c \) is the index of the category that the observed attribute value is classified into. \( A \) is an amplifying factor and can be equal to 1. If \( A \) is set to be greater than 1, since \( \text{RelFreq}_c \) is always smaller than or equal to 1, the resulting base scores become larger. In addition, as discussed in [6], in case the relative frequency is very small or equal to 0, the resulting score can be very large. In order to place an upper bound of base scores, a lower bound value of \( \text{RelFreq}_c^A \) needs to be enforced. For instance, if it is set to be \( 2^{-20} \), base scores are bounded up to 20. In this way, identity usage records of categories that are infrequently observed in the past are scored high while records of frequently observed categories are scored lower.

3.3 Score Weight based on Distinctiveness of User Profile

To quantify the distinctiveness of a user’s identity usage pattern in terms of a certain profile, we introduce a weight based on the distance between a user’s profile, which is a frequency distribution over the pre-defined categories, and a uniform distribution over the same categories. For instance, if the number of categories comprising the user profile is 10, the probability of each category in the uniform distribution used for the comparison is 0.1. In this way, if the identity credential usage is frequent in some categories and is scarce in others, i.e. the shape of the frequency distribution is far from flat, the weight becomes larger. On the other hand, if the pattern is nearly random, the frequency distribution of the profile is close to uniform. Thereby the resulting weight is small.

There are multiple ways to compute such distance. The first one is to use a distance measurement that can be used to compare the shape of discrete probability distributions (histograms), such as Bhattacharyya Distance and other distance measures discussed in [7]. Bhattacharyya Distance (\( BD \)) can be computed as follows. Where \( p \) is a profile distribution, \( u \) is a uniform distribution over the same categories, and \( C \) represents the set of all categories,

\[
BD_{pu} = -\ln(\sum_{c \in C} \sqrt{p(c)u(c)}) \tag{3}
\]

This distance is 0 when the two distributions are identical. On the other hand, when \( \sum_{c \in C} \sqrt{p(c)u(c)} \) is very small, the resulting distance becomes very large.

Another option is using a \( \chi^2 \) statistics used for Goodness-of-Fit testing. This can be computed as follows.

\[
\chi^2 = \sum_{c \in C} \frac{(p(c) - u(c))^2}{u(c)} \tag{4}
\]

This statistics also has the property we need. In other words, when the two distributions are identical, the value is 0. When the shapes of the distributions differ significantly, the resulting value becomes large. Therefore, we can also use this for the weight calculation. Although there are other alternative ways, we do not enumerate all the possibilities here. Any metric that measures the distance of the shapes of two discrete probability distributions can be used to compute weights.

In addition, we can consider transformation of such distance measurements instead of using them directly. For example, as can be seen, a raw value of Bhattacharyya Distance can be infinite, but in practice, it is desirable to set an upper bound on the scores. In such a case, we can transform the distance by using a simple transformation as follows. \( D \) represents a raw distance measurement and \( B \) indicates the upper-bound value.

\[
D_{bounded} = B(1 - e^{-D}) \tag{5}
\]

The resulting value is bounded in \([0, B]\). Another possible transformation just multiplies a raw weight by a constant factor when we want to increase the impact of the distinctiveness of user profiles.

3.4 Score Aggregation

As the final step of the score computation, we need to aggregate scores. In case we are using only one user profile, i.e. only one attribute, this step is just multiplying the base score by the corresponding weight.

However, we can have multiple profiles for each user. For instance, we can build an hour-of-day profile, day-of-week profile, and week-of-month profile for each user. Then, three base scores and three weights are computed based on the three profiles. In such a case, having a single overall score is often useful. We call a product of a base score and the corresponding weight a subscore. We can come up with several ways to compute the final score based on subscores. The first one is calculating the weighted sum of subscores. In case we prefer to prioritize some subscores over others, this is a reasonable option. Alternatively, we can pick the maximum value of subscores. The best way to aggregate scores varies depending on contexts, which will be explored in our future work.

4. EVALUATION

We implemented an anomaly scoring module based on the algorithm described in Section 3 by using Java and PostgreSQL database. In this section, we briefly discuss some preliminary results.

4.1 Data set and Experiment Setting

We used BuzzPort data set explained in Section 2 for evaluation, considering a login attempt using a password as online identity usage. We used 3 profiles per user, hour-of-day
profile (24 categories), day-of-week (7 categories), and week-of-month profile (5 categories).

In addition, we set the upper bound of base scores to be 20 and the upper bound of weights to be 5. Weights are bounded by using $D_{\text{bounded}}$ with $B = 5$ in (5). Thus, the product of a base score and weight is in $[0, 100)$ range. We also note that $A$ in (2) is set to be 2 and that the maximum of three subscores computed based on three profiles is chosen as the final output.

4.2 Results of Experiments

First, we pick one user among over 20,000 users found in the log data. Figure 1 shows the trend of the computed scores from the beginning. In the figure, a larger Record ID implies a more recent record of this user. As expected, though the scores are somewhat unstable at the beginning, the score values become stable as the user profile is sufficiently trained. In case of the user under consideration, approximately 75 records, which corresponds to accesses for about 1 month, were necessary for training.

Second, we chose 5 users with different access frequency and analyzed the accuracy in terms of false positive. To evaluate the false positive rate, we specified score thresholds, above which the corresponding accesses were considered anomalous. In our experiment, a tentative threshold was manually chosen for each user based on the distribution of scores in the training data set. Mechanisms to automatically calculate and update such thresholds will be explored in our future work. The straightforward way to determine such thresholds would be picking a certain percentile of each user’s empirical distribution of risk scores within a certain time window in the recent past. For each user, we first built profiles based on the log data excluding the access records in the last two months. Then, by using the access records of the last two months, which we assume do not include any malicious activities, we counted the number of records whose scores exceeded the pre-determined thresholds. The results of the experiments are summarized in Table 1.

By using the same threshold values of these users, we evaluated the detection rate (true positive rate), which is another aspect of accuracy. In order to evaluate the detection rate, we created a data set that consisted of 50 records randomly chosen among other users’ access records in the last two months. After we built profiles of each user in the same way as mentioned above, scoring was executed against this data set. Assuming that the records in the test data set are anomalous accesses, we counted how many of them were scored higher than the thresholds. The results are shown in Table 1. Because the records in the test data set are not actual attack attempts crafted by intelligent adversaries, this experiment itself is not enough to fully evaluate the ability of our scoring mechanism. However, we can view this result as one baseline for the effectiveness of our scheme.

We conducted the same experiments over the same users with data aging. We set the decay factor to be 0.5. The results are shown in Figure 2 and Table 2.

So far we have examined only a small amount of data samples, but considering that we only used the access timestamps, the accuracy is reasonably good. According to the FP Rate and TP Rate in Table 1, the results are comparable to other fraud detection schemes evaluated in [8]. This implies that the computed risk scores enough represent the suspiciousness of observed access attempts and that the accuracy and usefulness of our scheme.

Regarding the effect of data aging, according to the results in Table 2, we can see that the data aging widens the score range in general, which means that the weights representing the distinctiveness of profiles remain large. In addition, data aging seems to be effective in case the access frequency is relatively high while it worsened the result when the access frequency was very low. This implies that the data aging can negatively impact the profile building process in case the number of accesses is small, so we need to take this into consideration when tuning the system.

We applied the same scoring scheme on another data set available from an e-commerce company, and our preliminary experiments showed similar trends. Extensive analysis of this data set will be included in our future work.

4.3 Processing and Storage Costs

We briefly mention the overheads associated with our scheme. These measurements were done when our prototype was run on a Linux machine with Intel Core 2 Duo E6600 and 3GB RAM. On average, our system using three profiles, which are hour-of-day profile, day-of-week profile, and week-of-month profile, takes 5 milliseconds to process an access record. Therefore, it is fast enough to be used in a real-time manner as well as batch processing modes.

Regarding the storage-space requirement, our prototype with three profiles for each user requires approximately 1.4 KB per user, which is small enough to accommodate a large number of users. Since a profile needs to be created for each attribute used for profiling, increasing the number of attributes will make the required storage space larger. However, even in case we use a larger number of attributes, the required space would be smaller than NIDES, which is a versatile real-time intrusion detection scheme and uses about 100 KB per profile [9].

5. RELATED WORK

Various fraud detection schemes have been explored in credit card, telecommunication, and network intrusion detection areas [10][11][12]. While most of these schemes are highly context specific, our scheme can be used in a variety of settings. Although we focused on attributes derived from login records in this paper, any categorical parameters can be directly used in our scoring scheme. Moreover, as discussed earlier, non-categorical parameters also can be converted into categorical data by defining bins.

In addition, we envision that risk scores computed by our system can be used as an additional parameter for other detection mechanisms to enhance their effectiveness. Integration into a user authentication mechanism is also promis-
Figure 1: Trend of Anomaly Risk Scores (without data aging)

Figure 2: Trend of Anomaly Risk Scores (with data aging)
ing. Based on the risk scores, a system could flexibly change the strength of its authentication process. For example, by requiring additional credentials only when a score is high, impact on the usability will be minimized. Our system can also be used in a setting in which credential usage is monitored by an external party where available information for monitoring is often limited [13][14].

Our frequency based scheme is similar to one proposed by Krugel et al. [6]. However, in their scheme, the concept of weights as proposed by us is not used. If the number of categories is large, for example 100, and the distribution of weights as proposed by us is not used. If the number of categories is large, for example 100, and the distribution of weights is not used, the score in that case should not be high since the user’s identity usage is almost equally likely in all categories. In our scheme, by virtue of weights, scores in such a case result in small values, which is helpful to reduce false alarms.

NIDES [9] is a highly versatile and efficient intrusion detection system that can work in real-time. Our work has similarity with NIDES in the following perspectives. First, both schemes utilize only the information about user’s normal access patterns characterized by categorical parameters. Secondly, both can process each access attempt in a real-time manner. Thirdly, in both scheme, user profiles, which are used for scoring or anomaly detection, are established and updated without requiring human intervention. The common disadvantage is that the lack of capability to utilize multivariate statistical inferences, which we will explore in our future work. Theoretically, NIDES under appropriate parameter tuning and configuration is expected to work even in a context that our scheme is targeting. However, NIDES expects reasonably consistent and frequent activities. More specifically, it uses the last $N$ events to build a short-term profile. When occurrence of target events is infrequent, for example less than once a month, it would be difficult to establish a meaningful short-term profile since it is likely that the user’s behavior pattern changes while building short-term profile. Because our system uses each single event for scoring, it does not suffer from this problem.

There exist other types of intrusion/fraud detection mechanisms. Wang’s scheme [15][16] utilized the frequency distribution of byte values in the payload of packets to detect anomalous accesses or worm activities. Even though this technique can be applied to each single access attempts, their focus is detecting deviation in byte-pattern level. In our context, adversaries are expected to follow the correct protocol, and thereby exhibit normal byte-value distribution, to manipulate systems with compromised identity credentials. Thus, their scheme does not seem to work good in our context. Using behavior graphs of attacks or malwares to detect intrusion is also proposed [17][18]. Their schemes, like Warrender’s scheme [4] discussed in Section 2, utilize the sequence and transition among multiple events. Thus, it is not necessarily suitable to our goal. But, needless to say, it should be highly effective to use other intrusion/anomaly detection schemes, including ones mentioned above, in addition to our mechanism, which only relies on generic identity usage log records, in case they are available.

6. CONCLUSIONS AND FUTURE WORK

In this paper, we presented a real-time scoring scheme to compute a risk score for login requests that present identity credentials to access online services. Risk scores are calculated based on the degree of deviation from a user’s normal identity usage pattern characterized by categorical attributes extracted from identity usage log records. Our scheme also includes a weighting mechanism that reflects the distinctiveness of user’s identity usage pattern. We implemented the prototype of our algorithm and evaluated its performance with data from a real online service to assess its effectiveness and viability.

Our preliminary experiments shown in this paper only utilized timestamps of login attempts to a campus portal site. Although experiments showed reasonably good accuracy, it might be argued that evasion of such anomaly scoring could be easy. Actually, it might be easy for adversaries to mimic user’s behavior when only a single feature, access timestamp in this paper, is used for profiling. However, we believe that

<table>
<thead>
<tr>
<th>User</th>
<th>Access Freq. [per Month]</th>
<th>Tentative Threshold</th>
<th>FP Rate</th>
<th>TP Rate</th>
</tr>
</thead>
<tbody>
<tr>
<td>A</td>
<td>4</td>
<td>40</td>
<td>16.6 %</td>
<td>96.0 %</td>
</tr>
<tr>
<td>B</td>
<td>25</td>
<td>20</td>
<td>0.0 %</td>
<td>90.0 %</td>
</tr>
<tr>
<td>C</td>
<td>62</td>
<td>30</td>
<td>0.0 %</td>
<td>90.0 %</td>
</tr>
<tr>
<td>D</td>
<td>172</td>
<td>20</td>
<td>18.6 %</td>
<td>100.0 %</td>
</tr>
<tr>
<td>E</td>
<td>331</td>
<td>17</td>
<td>11.0 %</td>
<td>26.0 %</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>User</th>
<th>Access Freq. [per Month]</th>
<th>Tentative Threshold</th>
<th>FP Rate</th>
<th>TP Rate</th>
</tr>
</thead>
<tbody>
<tr>
<td>A</td>
<td>4</td>
<td>40</td>
<td>60.0 %</td>
<td>90.0 %</td>
</tr>
<tr>
<td>B</td>
<td>25</td>
<td>30</td>
<td>0.0 %</td>
<td>90.0 %</td>
</tr>
<tr>
<td>C</td>
<td>62</td>
<td>40</td>
<td>6.0 %</td>
<td>90.0 %</td>
</tr>
<tr>
<td>D</td>
<td>172</td>
<td>30</td>
<td>3.0 %</td>
<td>100.0 %</td>
</tr>
<tr>
<td>E</td>
<td>331</td>
<td>17</td>
<td>12.0 %</td>
<td>36.0 %</td>
</tr>
</tbody>
</table>
adding more parameters will improve accuracy and robustness and prevent attacks from adversaries. For instance, we can add estimated geographic location, such as country names etc. Location information can be derived from IP geolocation schemes such as [19]. While accesses recorded in our BuzzPort [3] data set are mostly initiated in similar geographic locations, access attempts contained in another data set provided by an e-commerce company are made from a variety of locations all over the world. Incorporating location information in the scoring scheme is expected to be useful especially in such settings. Organization information can be additionally obtained from WHOIS services, such as [20], which is another possible data source for profiling. We plan to integrate them in our future work. Utilizing correlation among such parameters might be effective. Another component of the future work is exploring the optimum configuration, such as the best choice of distance measurement, the most suitable score aggregation method, and optimum parameter settings, via detailed experiments. Additionally, we plan to combine our risk scoring mechanism with other security mechanism, such as fraud detection systems and user authentication systems, to prove the feasibility and effectiveness of such integration. We will also perform more detailed evaluation using multiple data sets and comparison with other existing schemes in the future.

7. ACKNOWLEDGMENT

We thank Jerry Perullo and Chuck Adkins at IntercontinentalExchange Inc. for valuable discussions. This material is based in part on work supported by IntercontinentalExchange Inc. This research was also supported in part by the National Science Foundation (under Grant CNS-CT-0716252).

8. REFERENCES