Big Data at the Service of SaaS Security Project

October 18, 2014

Evgeniy Pats - sjeka@t2.technion.ac.il

Michael Pekel - mipmip@t2.technion.ac.il

under the guidance of Yonatan Most - yonatan@adallom.com

Technion - Israel Institute of Technology and Science
# Contents

I  Project overview 3

II  An Overview and analysis of SaaS security 4

   1 SaaS overview 4
   2 Attack vectors 4
   3 Security mitigations 6

III  An overview of Machine learning 7

   4 Overview 7
   5 Anomaly Detection 7
      5.1 An overview of three algorithms for anomaly detection 8
         5.1.1 Discrete probability distribution 8
         5.1.2 Markov Chain 8

IV  Implementation and Results 10

   6 Overview 10
   7 Discrete probability distribution 11
   8 Markov chain 22
      8.1 Analysis 25
      8.2 Improvement 25
   9 Markov Chain with memory 26

V  Summary 27

VI  Bibliography 28
Part I

Project overview

SaaS technology becomes more and more popular these days and new security measures must be taken to protect this service. The following key security elements should be carefully considered as an integral part of the SaaS application development and deployment process:

- SaaS deployment model
- Data security
- Network security
- Regulatory compliance
- Data segregation
- Availability Backup
- Identity management and sign-on process

In this paper we will give an overview and analysis of different attack vectors and security measures used to protect SaaS applications.

Our key contribution will be the demonstration of three algorithms which detect and alert anomalies of user’s activity in a SaaS application. Each of the following algorithms will be analysed and tested for it’s effectiveness and false-positive/false-negative rate.
Part II

An Overview and analysis of SaaS security

1  SaaS overview

SaaS is a software delivery method that provides access to software and its functions remotely as a Web-based service. Software as a Service allows organizations to access business functionality. Also, because the software is hosted remotely, users don’t need to invest in additional hardware. Software as a Service removes the need for organizations to handle the installation, set-up and often daily upkeep and maintenance but most advantage is the security of its services maintained by professionals, and keeping all the softwares on the server patched and up to date. SaaS is actually anything from gmail to office 365, salesforce and many more.

2  Attack vectors

SaaS adoption is the safest route for enterprise workloads, but adopting SaaS requires a new approach to security that accounts for the limitations of existing technologies in protecting this new vector. In the context of on-premise applications, attackers needed to perform multi-stage attack chain:

1. Break-in: Phishing and remote exploits to gain access
2. Latch-on: Malware and back doors installed to establish a foothold
3. Expand: Reconnaissance and lateral movement to increase access and maintain a presence
4. Gather: Acquisition and aggregation of confidential data
5. Exfiltrate: Data exfiltration to external networks

For on-premise applications, attackers follow a 5-Stage attack chain

Traditional network security methods have implemented "defense-in-depth" by deploying detection and mitigation products on each of the stages as mentioned in the picture.
However, in the case of SaaS a new security measure need to be added for two main reasons:

1. SaaS can be accessed from anyplace in the world that means it can be accessed by an employee from an “unmanaged device” - phone, home computer, internet cafe. What this means is that the attacker is already bypassed the break-in stage and Latch-on easily and now using key-logger, Session injection and redirection (like Zeus or Dyre) the credentials of the SaaS application are stolen and can be accessed by the attacker.

2. Even if the enterprise enforce a very strict policy (which is not so common because it kills one of the biggest advantages of SaaS) as accessing the SaaS only from internal network and only from managed devices, it still deserves a new security measures, because lack of visibility, to protect from malicious users and to protect against APTs (Advanced Persistent Threat) that were able to infect the internal network of the enterprise.
3 Security mitigations

The solution to overcome these new techniques is to install a proxy device in-between the desired network and the application to monitor the data flow.

System diagram

The analysis module can range from new policies enforced by IT which are not available as part of the SaaS to more sophisticated heuristic engine, behavioural analysis and machine learning algorithm. In this paper we will present three ML algorithms which detect and alert user anomaly.
Part III

An overview of Machine learning

4 Overview

We define machine learning as a set of methods that can automatically detect patterns in data, and then use the uncovered patterns to predict future data, or to perform other kinds of decision making under uncertainty. Machine learning is usually divided into two main types:

The predictive or supervised learning approach, the goal is to learn a mapping from inputs $\chi$ to outputs $\gamma$, given a labeled set of input-output pairs $T = \{\chi_i, \gamma_i\}$, $\text{where } 1 \leq i \leq N$. Here $T$ is called the training set, and $N$ is the number of training examples.

In the simplest setting, each training input $\chi_i$ is a $D$-dimensional vector of numbers, representing some features, attributes or covariates. In general, however, $\chi_i$ could be a complex structured object, such as an image, a sentence, an email message, a time series, etc.

Similarly the form of the output or response variable can in principle be anything, but most methods assume that $\gamma_i$ is a categorical or nominal variable from some finite set, $\gamma_i \in \{1, ..., C\}$, or that $\gamma_i$ is a real-valued scalar. When $\gamma_i$ is categorical, the problem is known as classification or pattern recognition, and when $\gamma_i$ is real-valued, the problem is known as regression. Another variant, known as ordinal regression, occurs where label space $\gamma$ has some natural ordering, such as grades 0-100.

The second main type of machine learning is the descriptive or unsupervised learning approach. Here we are only given inputs, $D = \chi_i$, $\text{where } 1 \leq i \leq N$ and the goal is to find “interesting patterns” in the data. This is a much less well-defined problem, since we are not told what kinds of patterns to look for, and there is no obvious error metric to use (unlike supervised learning, where we can compare our prediction of $\gamma$ for a given $\chi$ to the observed value).

For example, classification of $\chi$ input vectors to $\gamma$ sets without the information of which vector in which set - those are known as clustering algorithms.

5 Anomaly Detection

Like unsupervised problem though some aspects are similar to supervised learning.

In this problem we are given a dataset $x(1), x(2), \ldots, x(m)$ and new example, $x(\text{test})$, and we want to know whether this new example is normal/anomalous.

This problem has few variants, first variant is when the the given dataset contains anomalous vectors but we don’t know which of them (because the anomalous samples are small part of the whole dataset). The second variant is when we know which of the samples are anomalous. We will be dealing with the first variant.
5.1 An overview of three algorithms for anomaly detection

5.1.1 Discrete probability distribution

Given a training set of \( m \) samples \( x^{(1)}, x^{(2)}, \ldots, x^{(m)} \) when each sample \( x^{(i)} \in \mathbb{R}^\ell \), in our case we decided to go with a Bernoulli distribution, which is more appropriate for the case when the response is binary \( \{0,1\} \).

We define \( \text{count}(x_i) \) as the number of times \( x \) occurred in feature \( i \) in all \( m \) samples. The observed probability mass function of each feature \( x_i \), where \( 1 \leq i \leq n \) is

\[
 f(x_i) = \frac{\text{count}(x_i)}{m}.
\]

We define a “model” \( p(x) \) that tells us the probability of the sample vector \( x \) is not anomalous. We also use a threshold \( \epsilon \) as a dividing line, so we can say which examples are anomalous and which are not.

The computed function is:

\[
 p(x) = f(x_1) \cdot f(x_2) \cdot \ldots \cdot f(x_\ell),
\]

this assumes the features are independent, otherwise we get more false positives results and not accurate regression, and to decide whether we got anomalous or not we use \( \epsilon \) as follows, if \( p(x) \leq \epsilon \) then \( x \) is anomalous and otherwise not.

5.1.2 Markov Chain

A Markov chain is a sequence of random variables with the Markov property, namely that, given the present state, the future and past states are independent.

Formally,

\[
 \Pr(X_{n+1} = x | X_1 = x_1, X_2 = x_2, \ldots, X_n = x_n) = \Pr(X_{n+1} = x | X_n = x_n)
\]

The possible values of \( X_i \) from a countable set \( S \) called the state space of the chain.

Markov chain often described by a transition matrix \( M \) from state \( n \) to \( n + 1 \), i.e

\[
 M[i][j] = \Pr(X_{n+1} = i | X_n = j)
\]

- **Time-homogeneous Markov chains**

Let \( p_{ij} \), denote the probability that the system is in a state \( j \) at time \( t + 1 \) and given the system is in state \( t \) at the time \( t \).

We are dealing with a finite number of states (same as number of events) then a stationary Markov-Chain can be defined by a transition probability matrix:

\[
 P = \begin{pmatrix}
 p_{11} & p_{12} & p_{13} & \ldots & p_{1n} \\
 p_{21} & p_{22} & p_{23} & \ldots & p_{2n} \\
 \vdots & \vdots & \vdots & \ddots & \vdots \\
 p_{n1} & p_{n2} & p_{n3} & \ldots & p_{nn}
\end{pmatrix}
\]

and an initial probability distribution: \( Q = [q_1 \ q_2 \ \ldots \ q_n] \) where \( q_i \) is the probability that the system is in state \( i \) at time 0, and \( \sum_{j=1}^{n} p_{ij} = 1 \).

The probability that a sequence of states \( X_1, X_2, \ldots, X_T \) where \( T \) means we inspect each time a sequence window of \( T \) events in a row, when they occurs in the context of the stationary Markov-Chain, and can computed as follows:
\[
P(X_1, X_2, \ldots, X_T) = q_{x_1} \cdot \prod_{t=2}^{T} P_{X_{t-1}, X_t}
\]

the initial probability distribution of a stationary Markov chain can be learned from the observations of the system state in the past. So the initial probability distribution as follows:

\[
p_{ij} = \frac{N_{ij}}{N_i},
\]

\[
q_{xi} = \frac{N_i}{N}
\]

where:

- \( N_{ij} \) - is the number of observation pairs \( X_t \) and \( X_{t+1} \) with \( X_t \) in state \( i \) and \( X_{t+1} \) in state \( j \);
- \( N_i \) - is the number of observation pairs \( X_t \) and \( X_{t+1} \) with \( X_t \) in state \( i \) and \( X_{t+1} \) in any one of the states \( 1, 2, \ldots, T \);
- \( N_{i} \) - is the number of \( X_t \)'s in state \( i \);
- \( N \) - is the total number of observations;

In order to determine whether we got anomalous or not, we use \( \epsilon \) as follows, if \( P(X_1, X_2, \ldots, X_T) \leq \epsilon \) then \( X_1, X_2, \ldots, X_T \) is an anomalous session and otherwise not.

- **Markov chain of order \( m \)**

A Markov chain of order \( m \) (or a Markov chain with memory \( m \)), where \( m \) is finite, is a process satisfying:

\[
Pr(X_{n+1} = x_{n+1} | X_n = x_n, X_{n-1} = x_{n-1}, \ldots, X_1 = x_1) =
\]

\[
= Pr(X_{n+1} = x_{n+1} | X_n = x_n, X_{n-1} = x_{n-1}, \ldots, X_{n-m+1} = x_{n-m+1})
\]

Here the process is a discrete-time Markov chain of order \( m \) and the transition probability to a state at the next time is a sum of functions, each depending on the next state and one of the \( m \) previous states. An additive Markov chain of order \( m \) is a sequence of random variables \( X_1, X_2, \ldots, X_m \) possessing the following property:

the probability that a random variable \( X_n \) has a certain value \( x_n \) under the condition that the values of all previous variables are fixed, depends on the values of \( m \) previous variables only, and the influence of previous variables on a generated one is additive.

In other words, the future state depends on the past \( m \) states (expansion of the previous model, all other things same as we explained above).
Part IV

Implementation and Results

6 Overview

We implemented algorithms [5.1.1],[5.1.2],[5.1.3] in python 2.7 on Ubuntu 12.3.11.0
libraries used: numpy 1.7.1, scipy 0.12.0, matplotlib 1.2.1.

The data we are working with is sanitized statistics provided by Adallom (in Json format) for different applications. For example user event has the next json structure:

```
{
    'username': 'josh_a',
    'timestamp': 1391811809738,
    'country': 'Israel',
    'IP': 93.170.24.61,
    'app': 'OFFICE_365',
    'OS': 'WINDOWS_7',
    'event': 'MAIL_VIEW',
    'browser': 'CHROME'
}
```

In the next following sections we will demonstrate only events generated by office_365 app.

\[ \text{os} \in \{\text{windows, android, osx, windows 8, ios 7, osx 10.8, windows xp, windows 7, android 4.0.x, android 4.2, ...}\} \]

\[ \text{...JVM(PlatformMicroEdition), ios}\]

\[ \text{browser} \in \{\text{Outlook Client, Chrome, Safari, Apple Mail, Mobile Safari, Firefox, Mozilla, IE, ...}\} \]

\[ \text{...Chrome Mobile, UC Browser}\]

\[ \text{event} \in \text{o365} - \text{outlook} - \text{message} - \text{view},\text{event} - \text{o365} - \text{user} - \text{login},\text{event} - \text{o365} - \text{outlook} - \text{send},\ldots\]

\[ \text{...event} - \text{o365} - \text{user} - \text{login} - \text{failure},\text{event} - \text{o365} - \text{outlook} - \text{folder} - \text{new},\text{event} - \text{o365} - \text{outlook} - \text{conversation} - \text{view},\text{event} - \text{o365} - \text{user} - \text{logout},\ldots\]

\[ \text{...event} - \text{o365} - \text{sp} - \text{site} - \text{view},\text{event} - \text{o365} - \text{sp} - \text{document} - \text{upload},\text{event} - \text{o365} - \text{sp} - \text{form} - \text{view},\text{event} - \text{o365} - \text{sp} - \text{document} - \text{inbrowser} - \text{view},\ldots\]

\[ \text{...event} - \text{o365} - \text{excel} - \text{document} - \text{view},\text{event} - \text{o365} - \text{outlook} - \text{conversation} - \text{move},\text{event} - \text{o365} - \text{outlook} - \text{message} - \text{delete},\ldots\]

\[ \text{...event} - \text{o365} - \text{outlook} - \text{message} - \text{draft},\text{event} - \text{o365} - \text{outlook} - \text{conversation} - \text{delete},\text{event} - \text{o365} - \text{onenote} - \text{document} - \text{edit},\ldots\]

\[ \text{...event} - \text{o365} - \text{sp} - \text{document} - \text{preview},\text{event} - \text{o365} - \text{excel} - \text{preview},\text{event} - \text{o365} - \text{sp} - \text{document} - \text{delete},\text{event} - \text{o365} - \text{outlook} - \text{folder} - \text{delete},\ldots\]

\[ \text{...event} - \text{o365} - \text{outlook} - \text{folder} - \text{rename},\text{event} - \text{o365} - \text{outlook} - \text{folder} - \text{move},\text{event} - \text{o365} - \text{sp} - \text{folder} - \text{add},\text{event} - \text{o365} - \text{sp} - \text{document} - \text{download},\ldots\]

\[ \text{...event} - \text{o365} - \text{excel} - \text{document} - \text{edit},\text{event} - \text{o365} - \text{sp} - \text{document} - \text{create},\text{event} - \text{o365} - \text{sp} - \text{folder} - \text{view},\text{event} - \text{o365} - \text{outlook} - \text{attachment} - \text{download},\ldots\]

\[ \text{...event} - \text{o365} - \text{sp} - \text{list} - \text{view},\text{event} - \text{o365} - \text{sp} - \text{document} - \text{share},\text{event} - \text{o365} - \text{word} - \text{document} - \text{view},\text{event} - \text{o365} - \text{outlook} - \text{folder} - \text{view}\]
7 Discrete probability distribution

We executed the algorithm on couple of users, which we had enough data recorded for them, and found abnormal events.

The feature vector was \([\text{country}, \text{os} \cdot \text{browser}, \text{os} \cdot \text{event}]\) because there is a strong correlation between os and event, os and browser. Moreover we have a finite number of training examples with which to train a classifier, therefore for a fixed number of training examples, increasing the number of features typically increases classification accuracy to a point but as the number of features continue to increase, classification accuracy will eventually decrease because we are then undersampled relative to the large number of features.

By making a feature which is a cartesian product of both features will eliminate false positives because having too many correlated features relative to the training sample size will render the classifier unusable in the original feature space (the covariance matrix of the sample data becomes singular).

Let's take a look at specific users which used office 365.

We got 63 anomalies, when they can be divided into true positives and false positives.

![Top 10 anomalies chart](image)

**Figure1:** Histogram of Top10 anomaly events
Sorted top anomaly events for user 25

```
[{'p': 2.2836554736501145e-11, 'vector': ['country_1', (u'Android 4.0.x Ice Cream Sandwich', u'Chrome Mobile')],
 (u'Android 4.0.x Ice Cream Sandwich', u'EVENT_O365_USER_LOGIN'))},
{'p': 8.7132639363246841e-10, 'vector': ['country_1', (u'Windows 7', u'Chrome')],
(u'Windows 7', u'EVENT_O365_OUTLOOK_MESSAGE_VIEW')},
{'p': 2.6145529252303507e-09, 'vector': ['country_1', (u'Windows 7', u'Chrome')],
(u'Windows 7', u'EVENT_O365_USER_LOGOUT')},
{'p': 2.6145529252303507e-09, 'vector': ['country_1', (u'Windows 7', u'Chrome')],
(u'Windows 7', u'EVENT_O365_USER_LOGOUT')},
{'p': 3.4864531891411785e-09, 'vector': ['country_1', (u'Windows 7', u'Chrome')],
(u'Windows 7', u'EVENT_O365_OUTLOOK_MESSAGE_DRAFT')},
{'p': 3.4864531891411785e-09, 'vector': ['country_1', (u'Windows 7', u'Chrome')],
(u'Windows 7', u'EVENT_O365_OUTLOOK_MESSAGE_DRAFT')},
{'p': 3.4864531891411789e-09, 'vector': ['country_1', (u'Windows 7', u'Chrome')],
(u'Windows 7', u'EVENT_O365_OUTLOOK_SEND')},
{'p': 3.4864531891411789e-09, 'vector': ['country_1', (u'Windows 7', u'Chrome')],
(u'Windows 7', u'EVENT_O365_OUTLOOK_SEND')},
{'p': 5.2308280282575374e-09, 'vector': ['country_1', (u'Windows 7', u'Chrome')],
(u'Windows 7', u'EVENT_O365_USER_LOGIN')},
{'p': 5.2308280282575374e-09, 'vector': ['country_1', (u'Windows 7', u'Chrome')],
(u'Windows 7', u'EVENT_O365_USER_LOGIN')},
{'p': 7.3482759101877195e-09, 'vector': ['country_1', (u'Windows 7', u'Mozilla')],
(u'Windows 7', u'EVENT_O365_USER_LOGOUT')},
{'p': 9.79877658418176e-09, 'vector': ['country_1', (u'Windows 7', u'Mozilla')],
(u'Windows 7', u'EVENT_O365_OUTLOOK_SEND')},
{'p': 9.79877658418176e-09, 'vector': ['country_1', (u'Windows 7', u'Mozilla')],
(u'Windows 7', u'EVENT_O365_OUTLOOK_SEND')},
{'p': 1.3964204860430936e-08, 'vector': ['country_1', (u'Windows 7', u'Chrome')],
(u'Windows 7', u'EVENT_O365_OUTLOOK_FOLDER_VIEW')},
{'p': 1.3964204860430936e-08, 'vector': ['country_1', (u'Windows 7', u'Chrome')],
(u'Windows 7', u'EVENT_O365_OUTLOOK_FOLDER_VIEW')},
{'p': 1.3964204860430936e-08, 'vector': ['country_1', (u'Windows 7', u'Chrome')],
(u'Windows 7', u'EVENT_O365_OUTLOOK_FOLDER_VIEW')},
{'p': 1.3964204860430936e-08, 'vector': ['country_1', (u'Windows 7', u'Chrome')],
(u'Windows 7', u'EVENT_O365_OUTLOOK_FOLDER_VIEW')},
{'p': 1.3964204860430936e-08, 'vector': ['country_1', (u'Windows 7', u'Chrome')],
(u'Windows 7', u'EVENT_O365_OUTLOOK_FOLDER_VIEW')},
{'p': 1.3964204860430936e-08, 'vector': ['country_1', (u'Windows 7', u'Chrome')],
(u'Windows 7', u'EVENT_O365_OUTLOOK_FOLDER_VIEW')},
{'p': 1.3964204860430936e-08, 'vector': ['country_1', (u'Windows 7', u'Chrome')],
(u'Windows 7', u'EVENT_O365_OUTLOOK_FOLDER_VIEW')},
{'p': 1.3964204860430936e-08, 'vector': ['country_1', (u'Windows 7', u'Chrome')],
(u'Windows 7', u'EVENT_O365_OUTLOOK_FOLDER_VIEW')},
```
More anomalies..

```json
[{'p': 1.4701392050418379e-08, 'vector': ['country_1', (u'Windows 7', u'Mozilla'), (u'Windows 7', u'EVENT_O365_USER_LOGIN')]},
{'p': 1.4701392050418379e-08, 'vector': ['country_1', (u'Windows 7', u'Mozilla'), (u'Windows 7', u'EVENT_O365_USER_LOGIN')]},
{'p': 1.4701392050418379e-08, 'vector': ['country_1', (u'Windows 7', u'Mozilla'), (u'Windows 7', u'EVENT_O365_USER_LOGIN')]},
{'p': 1.4701392050418379e-08, 'vector': ['country_1', (u'Windows 7', u'Mozilla'), (u'Windows 7', u'EVENT_O365_USER_LOGIN')]},
{'p': 1.801136893561057e-08, 'vector': ['country_1', (u'Windows', u'Chrome'), (u'Windows', u'EVENT_O365_USER_LOGOUT')]},
{'p': 1.801136893561057e-08, 'vector': ['country_1', (u'Windows', u'Chrome'), (u'Windows', u'EVENT_O365_USER_LOGOUT')]},
{'p': 2.4088605302131049e-08, 'vector': ['country_2', (u'Windows 8', u'Chrome'), (u'Windows 8', u'EVENT_O365_OUTLOOK_ATTACHMENT_DOWNLOAD')]},

Sorted anomaly events for user 3

```json
[{'p': 6.2286156290422956e-12, 'vector': ['country_2', (u'Windows 8', u'Chrome'), (u'Windows 8', u'EVENT_O365_USER_LOGIN')]},
{'p': 2.7201203355536764e-11, 'vector': ['country_2', (u'Windows 8', u'Chrome'), (u'Windows 8', u'EVENT_O365_OUTLOOK_FOLDER_VIEW')]},
{'p': 1.6672156778264361e-10, 'vector': ['country_2', (u'Windows 8', u'Chrome'), (u'Windows 8', u'EVENT_O365_OUTLOOK_CONVERSATION_VIEW')]},
{'p': 1.6672156778264361e-10, 'vector': ['country_2', (u'Windows 8', u'Chrome'), (u'Windows 8', u'EVENT_O365_OUTLOOK_CONVERSATION_VIEW')]},
{'p': 1.6672156778264361e-10, 'vector': ['country_2', (u'Windows 8', u'Chrome'), (u'Windows 8', u'EVENT_O365_OUTLOOK_CONVERSATION_VIEW')]},
{'p': 1.6672156778264361e-10, 'vector': ['country_2', (u'Windows 8', u'Chrome'), (u'Windows 8', u'EVENT_O365_OUTLOOK_CONVERSATION_VIEW')]},
{'p': 1.6672156778264361e-10, 'vector': ['country_2', (u'Windows 8', u'Chrome'), (u'Windows 8', u'EVENT_O365_OUTLOOK_CONVERSATION_VIEW')]},
```
- True positives

11 events from different country.

<table>
<thead>
<tr>
<th>Anomaly event with p=1.66721567783e-10</th>
</tr>
</thead>
<tbody>
<tr>
<td>['country_2', (u'Windows 8', u'Chrome'), (u'Windows 8', u'EVENT_O365_OUTLOOK_CONVERSATION_VIEW')]</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Anomaly event with p=1.66721567783e-10</th>
</tr>
</thead>
<tbody>
<tr>
<td>['country_2', (u'Windows 8', u'Chrome'), (u'Windows 8', u'EVENT_O365_OUTLOOK_CONVERSATION_VIEW')]</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Anomaly event with p=1.66721567783e-10</th>
</tr>
</thead>
<tbody>
<tr>
<td>['country_2', (u'Windows 8', u'Chrome'), (u'Windows 8', u'EVENT_O365_OUTLOOK_CONVERSATION_VIEW')]</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Anomaly event with p=1.66721567783e-10</th>
</tr>
</thead>
<tbody>
<tr>
<td>['country_2', (u'Windows 8', u'Chrome'), (u'Windows 8', u'EVENT_O365_OUTLOOK_CONVERSATION_VIEW')]</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Anomaly event with p=1.66721567783e-10</th>
</tr>
</thead>
<tbody>
<tr>
<td>['country_2', (u'Windows 8', u'Chrome'), (u'Windows 8', u'EVENT_O365_OUTLOOK_CONVERSATION_VIEW')]</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Anomaly event with p=1.66721567783e-10</th>
</tr>
</thead>
<tbody>
<tr>
<td>['country_2', (u'Windows 8', u'Chrome'), (u'Windows 8', u'EVENT_O365_OUTLOOK_CONVERSATION_VIEW')]</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Anomaly event with p=1.66721567783e-10</th>
</tr>
</thead>
<tbody>
<tr>
<td>['country_2', (u'Windows 8', u'Chrome'), (u'Windows 8', u'EVENT_O365_USER_LOGIN')]</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Anomaly event with p=1.66721567783e-10</th>
</tr>
</thead>
<tbody>
<tr>
<td>['country_2', (u'Windows 8', u'Chrome'), (u'Windows 8', u'EVENT_O365_OUTLOOK_FOLDER_VIEW')]</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Anomaly event with p=1.66721567783e-10</th>
</tr>
</thead>
<tbody>
<tr>
<td>['country_2', (u'Windows 8', u'Chrome'), (u'Windows 8', u'EVENT_O365_OUTLOOK_CONVERSATION_VIEW')]</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Anomaly event with p=6.22861562904e-12</th>
</tr>
</thead>
<tbody>
<tr>
<td>['country_2', (u'Windows 8', u'Chrome'), (u'Windows 8', u'EVENT_O365_OUTLOOK_FOLDER_VIEW')]</td>
</tr>
</tbody>
</table>
anomaly event with p=1.66721567783e-10
['country_2', (u'Windows 8', u'Chrome'), (u'Windows 8', u'EVENT_O365_OUTLOOK_CONVERSATION_VIEW')]
anomaly event with p=1.66721567783e-10
['country_2', (u'Windows 8', u'Chrome'), (u'Windows 8', u'EVENT_O365_OUTLOOK_CONVERSATION_VIEW')]
anomaly event with p=1.66721567783e-10
['country_2', (u'Windows 8', u'Chrome'), (u'Windows 8', u'EVENT_O365_OUTLOOK_CONVERSATION_VIEW')]
anomaly event with p=1.66721567783e-10
['country_2', (u'Windows 8', u'Chrome'), (u'Windows 8', u'EVENT_O365_OUTLOOK_CONVERSATION_VIEW')]
anomaly event with p=1.66721567783e-10
['country_2', (u'Windows 8', u'Chrome'), (u'Windows 8', u'EVENT_O365_OUTLOOK_CONVERSATION_VIEW')]
anomaly event with p=6.22861562904e-12
['country_2', (u'Windows 8', u'Chrome'), (u'Windows 8', u'EVENT_O365_USER_LOGIN')]
anomaly event with p=1.66721567783e-10
['country_2', (u'Windows 8', u'Chrome'), (u'Windows 8', u'EVENT_O365_OUTLOOK_CONVERSATION_VIEW')]
anomaly event with p=1.66721567783e-10
['country_2', (u'Windows 8', u'Chrome'), (u'Windows 8', u'EVENT_O365_OUTLOOK_CONVERSATION_VIEW')]
anomaly event with p=1.66721567783e-10
['country_2', (u'Windows 8', u'Chrome'), (u'Windows 8', u'EVENT_O365_OUTLOOK_CONVERSATION_VIEW')]
anomaly event with p=1.66721567783e-10
['country_2', (u'Windows 8', u'Chrome'), (u'Windows 8', u'EVENT_O365_OUTLOOK_FOLDER_VIEW')]

4 login events from android device:

anomaly event with p=2.10874606925e-10
['country_1', (u'Android 4.0.x Ice Cream Sandwich', u'Chrome Mobile'), (u'Android 4.0.x Ice Cream Sandwich', u'EVENT_O365_USER_LOGIN')]
anomaly event with p=1.89852303608e-09
['country_1', (u'Android 4.2 Jelly Bean', u'Chrome Mobile'), (u'Android 4.2 Jelly Bean', u'EVENT_O365_USER_LOGIN')]
anomaly event with p=1.89852303608e-09
['country_1', (u'Android 4.2 Jelly Bean', u'Chrome Mobile'), (u'Android 4.2 Jelly Bean', u'EVENT_O365_USER_LOGIN')]
anomaly event with p=2.72012033555e-11
['country_2', (u'Windows 8', u'Chrome'), (u'Windows 8', u'EVENT_O365_OUTLOOK_FOLDER_VIEW')]
After manual analysis we can say these are really strange events because those were the only events from android devices.

**6 delete events**

- anomaly event with p=6.29493416518e-07
  - ['country_1', (u'Windows', u'Chrome'), (u'Windows', u'EVENT_O365_OUTLOOK_FOLDER_DELETE')]
- anomaly event with p=6.29493416518e-07
  - ['country_1', (u'Windows', u'Chrome'), (u'Windows', u'EVENT_O365_OUTLOOK_FOLDER_DELETE')]
- anomaly event with p=6.29493416518e-07
  - ['country_1', (u'Windows', u'Chrome'), (u'Windows', u'EVENT_O365_OUTLOOK_FOLDER_DELETE')]
- anomaly event with p=6.29493416518e-07
  - ['country_1', (u'Windows', u'Chrome'), (u'Windows', u'EVENT_O365_OUTLOOK_FOLDER_DELETE')]
- anomaly event with p=6.29493416518e-07
  - ['country_1', (u'Windows', u'Chrome'), (u'Windows', u'EVENT_O365_OUTLOOK_FOLDER_DELETE')]
- anomaly event with p=6.29493416518e-07
  - ['country_1', (u'Windows', u'Chrome'), (u'Windows', u'EVENT_O365_OUTLOOK_FOLDER_DELETE')]
- anomaly event with p=7.73878163032e-07
  - ['country_1', (u'Windows8', u'Chrome'), (u'Windows', u'EVENT_O365_SP_FORM_VIEW')]

**7 other rare events:**

- anomaly event with p=3.14665671517e-07
  - ['country_1', (u'Windows', u'Chrome'), (u'Windows', u'EVENT_O365_SP_FORM_VIEW')]
- anomaly event with p=1.04870556733e-07
  - ['country_1', (u'Windows', u'Chrome'), (u'Windows', u'EVENT_O365_SP_DOCUMENT_INBROWSER_VIEW')]
- anomaly event with p=1.04870556733e-07
  - ['country_1', (u'Windows', u'Chrome'), (u'Windows', u'EVENT_O365_SP_DOCUMENT_INBROWSER_VIEW')]
- anomaly event with p=1.04870556733e-07
  - ['country_1', (u'Windows', u'Chrome'), (u'Windows', u'EVENT_O365_SP_DOCUMENT_INBROWSER_VIEW')]
- anomaly event with p=1.04870556733e-07
  - ['country_1', (u'Windows', u'Chrome'), (u'Windows', u'EVENT_O365_SP_DOCUMENT_INBROWSER_VIEW')]
- anomaly event with p=1.04870556733e-07
  - ['country_1', (u'Windows', u'Chrome'), (u'Windows', u'EVENT_O365_SP_DOCUMENT_INBROWSER_VIEW')]
- anomaly event with p=1.04870556733e-07
  - ['country_1', (u'Windows', u'Chrome'), (u'Windows', u'EVENT_O365_SP_DOCUMENT_INBROWSER_VIEW')]
anomaly event with p=3.14665671517e-07
['country_1', (u'Windows', u'Chrome'), (u'Windows', u'EVENT_O365_SP_FORM_VIEW')]
anomaly event with p=1.04870556733e-07
['country_1', (u'Windows', u'Chrome'), (u'Windows',
 u'EVENT_O365_SP_DOCUMENT_INBROWSER_VIEW')]
anomaly event with p=1.04870556733e-07
['country_1', (u'Windows', u'Chrome'), (u'Windows',
 u'EVENT_O365_SP_DOCUMENT_INBROWSER_VIEW')]
anomaly event with p=1.04870556733e-07
['country_1', (u'Windows', u'Chrome'), (u'Windows',
 u'EVENT_O365_SP_FOLDER_VIEW')]
anomaly event with p=3.14665671517e-07
['country_1', (u'Windows', u'Chrome'), (u'Windows',
 u'EVENT_O365_SP_FOLDER_VIEW')]
anomaly event with p=3.14665671517e-07
['country_1', (u'Windows', u'Chrome'), (u'Windows',
 u'EVENT_O365_SP_FOLDER_VIEW')]
anomaly event with p=3.14665671517e-07
['country_1', (u'Windows', u'Chrome'), (u'Windows',
 u'EVENT_O365_SP_FOLDER_VIEW')]
anomaly event with p=3.14665671517e-07
['country_1', (u'Windows', u'Chrome'), (u'Windows',
 u'EVENT_O365_SP_FOLDER_VIEW')]
anomaly event with p=3.14665671517e-07
['country_1', (u'Windows', u'Chrome'), (u'Windows',
 u'EVENT_O365_SP_FOLDER_VIEW')]

- False positives

6 logout events:

anomaly event with p=9.4483360255e-07
['country_1', (u'Windows', u'Chrome'), (u'Windows', u'EVENT_O365_USER_LOGOUT')]
anomaly event with p=9.4483360255e-07
['country_1', (u'Windows', u'Mozilla'), (u'Windows', u'EVENT_O365_USER_LOGOUT')]
anomaly event with p=9.4483360255e-07
['country_1', (u'Windows', u'Chrome'), (u'Windows', u'EVENT_O365_USER_LOGOUT')]
anomaly event with p=9.4483360255e-07
['country_1', (u'Windows', u'Chrome'), (u'Windows', u'EVENT_O365_USER_LOGOUT')]
anomaly event with p=9.4483360255e-07
['country_1', (u'Windows', u'Chrome'), (u'Windows', u'EVENT_O365_USER_LOGOUT')]
anomaly event with p=9.4483360255e-07
['country_1', (u'Windows', u'Chrome'), (u'Windows', u'EVENT_O365_USER_LOGOUT')]
anomaly event with p=9.4483360255e-07
['country_1', (u'Windows', u'Chrome'), (u'Windows', u'EVENT_O365_USER_LOGOUT')]

Those are false positives because after manual analysis we saw the user almost never perform an active logout. A simple solution to overcome those false positives is to ignore logout events.
11 events of conversation/folder view from iOS:

anomaly event with p=2.09088682288e-08
['country_1', (u'iOS', u'Mobile Safari'), (u'iOS', u'EVENT_O365_OUTLOOK_CONVERSATION_VIEW')]
anomaly event with p=2.09088682288e-08
['country_1', (u'iOS', u'Mobile Safari'), (u'iOS', u'EVENT_O365_OUTLOOK_CONVERSATION_VIEW')]
anomaly event with p=2.09088682288e-08
['country_1', (u'iOS', u'Mobile Safari'), (u'iOS', u'EVENT_O365_OUTLOOK_CONVERSATION_VIEW')]
anomaly event with p=4.64362404327e-09
['country_1', (u'iOS', u'Mobile Safari'), (u'iOS', u'EVENT_O365_OUTLOOK_FOLDER_VIEW')]
anomaly event with p=2.09088682288e-08
['country_1', (u'iOS', u'Mobile Safari'), (u'iOS', u'EVENT_O365_OUTLOOK_CONVERSATION_VIEW')]
anomaly event with p=2.09088682288e-08
['country_1', (u'iOS', u'Mobile Safari'), (u'iOS', u'EVENT_O365_OUTLOOK_CONVERSATION_VIEW')]
anomaly event with p=2.09088682288e-08
['country_1', (u'iOS', u'Mobile Safari'), (u'iOS', u'EVENT_O365_OUTLOOK_CONVERSATION_VIEW')]
anomaly event with p=2.09088682288e-08
['country_1', (u'iOS', u'Mobile Safari'), (u'iOS', u'EVENT_O365_OUTLOOK_CONVERSATION_VIEW')]
anomaly event with p=4.64362404327e-09
['country_1', (u'iOS', u'Mobile Safari'), (u'iOS', u'EVENT_O365_OUTLOOK_FOLDER_VIEW')]

After manual analysis we can see that those are all the events from iOS device.
It doesn’t seem to us like a bad events, a simple solution would be to make a privilege rule for these kind of events, this should be applied after verifying that the user really use iOS device.
8 Markov chain

We run this algorithm on events of couple of users in a “session”.

In our first try we defined “session” as - All the events that happened from a specific os, specific browser, specific app.

We found 6 anomalies,

Markov chain top 3 anomaly events for user 25

```python
[{
    'IP': '107',
    'OS': 'Windows',
    'app': 11161,
    'browser': 'Outlook Client',
    'country': 2,
    'event': 'EVENT_O365_USER_LOGIN',
    'failed_username': None,
    'timestamp': '2014-02-16 20:40:28.373000',
    'username': 25
},
{
    'IP': '107',
    'OS': 'Windows',
    'app': 11161,
    'browser': 'Outlook Client',
    'country': 2,
    'event': 'EVENT_O365_USER_LOGIN',
    'failed_username': None,
    'timestamp': '2014-02-16 20:40:28.373000',
    'username': 25
}]
```

Figure 2: The distribution of $p$ on all the events
In this test we will be looking at: username=5, app=office 365, user_agent = Outlook Client, total number of events = 6697.

- False positives

4 sequence events of send-login:

```python
anomaly events with p=0.00318640467339
[{'u'OS': 'Windows', 'u'app': 11161, 'u'browser': 'u'Outlook Client', 'u'event': 'u'EVENT_O365_OUTLOOK_SEND'},
 {'u'OS': 'Windows', 'u'app': 11161, 'u'browser': 'u'Outlook Client', 'u'event': 'u'EVENT_O365_USER_LOGIN'},
 {'u'OS': 'Windows', 'u'app': 11161, 'u'browser': 'u'Outlook Client', 'u'event': 'u'EVENT_O365_OUTLOOK_SEND'},
 {'u'OS': 'Windows', 'u'app': 11161, 'u'browser': 'u'Outlook Client', 'u'event': 'u'EVENT_O365_USER_LOGIN'}]
```

```python
anomaly events with p=0.00318640467339
[{'u'OS': 'Windows', 'u'app': 11161, 'u'browser': 'u'Outlook Client', 'u'event': 'u'EVENT_O365_OUTLOOK_SEND'},
 {'u'OS': 'Windows', 'u'app': 11161, 'u'browser': 'u'Outlook Client', 'u'event': 'u'EVENT_O365_USER_LOGIN'},
 {'u'OS': 'Windows', 'u'app': 11161, 'u'browser': 'u'Outlook Client', 'u'event': 'u'EVENT_O365_OUTLOOK_SEND'},
 {'u'OS': 'Windows', 'u'app': 11161, 'u'browser': 'u'Outlook Client', 'u'event': 'u'EVENT_O365_USER_LOGIN'}]
```

```python
anomaly events with p=0.00318640467339
[{'u'OS': 'Windows', 'u'app': 11161, 'u'browser': 'u'Outlook Client', 'u'event': 'u'EVENT_O365_OUTLOOK_SEND'},
 {'u'OS': 'Windows', 'u'app': 11161, 'u'browser': 'u'Outlook Client', 'u'event': 'u'EVENT_O365_USER_LOGIN'},
 {'u'OS': 'Windows', 'u'app': 11161, 'u'browser': 'u'Outlook Client', 'u'event': 'u'EVENT_O365_OUTLOOK_SEND'},
 {'u'OS': 'Windows', 'u'app': 11161, 'u'browser': 'u'Outlook Client', 'u'event': 'u'EVENT_O365_USER_LOGIN'}]
```

```python
anomaly events with p=0.00318640467339
[{'u'OS': 'Windows', 'u'app': 11161, 'u'browser': 'u'Outlook Client', 'u'event': 'u'EVENT_O365_OUTLOOK_SEND'},
 {'u'OS': 'Windows', 'u'app': 11161, 'u'browser': 'u'Outlook Client', 'u'event': 'u'EVENT_O365_USER_LOGIN'},
 {'u'OS': 'Windows', 'u'app': 11161, 'u'browser': 'u'Outlook Client', 'u'event': 'u'EVENT_O365_OUTLOOK_SEND'},
 {'u'OS': 'Windows', 'u'app': 11161, 'u'browser': 'u'Outlook Client', 'u'event': 'u'EVENT_O365_USER_LOGIN'}]
```
• True positives

2 sequence events of send-login:

anomaly events with p=0.00318640467339

[[{ u'OS': u'Windows', u'app': 11161, u'browser': u'Outlook Client', u'event': u'EVENT_O365_OUTLOOK_SEND',
   u'timestamp': u'2014-05-01 22:15:54.067000', u'username': 5},
  { u'OS': u'Windows', u'app': 11161, u'browser': u'Outlook Client', u'event': u'EVENT_O365_USER_LOGIN',
   u'timestamp': u'2014-05-01 22:15:55.010000', u'username': 5}]]

anomaly events with p=0.00318640467339

[[{ u'OS': u'Windows', u'app': 11161, u'browser': u'Outlook Client', u'event': u'EVENT_O365_OUTLOOK_SEND',
   u'timestamp': u'2014-05-01 22:50:32.921000', u'username': 5},
  { u'OS': u'Windows', u'app': 11161, u'browser': u'Outlook Client', u'event': u'EVENT_O365_USER_LOGIN',
   u'timestamp': u'2014-05-01 22:50:33.646000', u'username': 5}]]
8.1 Analysis

We got six anomalies where all of them are the sequence of send event and then login, which is indeed a strange sequence of events that can only happen when a session expires. Four of the anomalies are false positives because the session probably expired before the login event occurred after a couple of hours. Here we can demonstrate you some interesting results:

![Figure 3: Anomaly distribution between events in a session consists of 10 events](image1)

![Figure 4: Markov-Chain anomaly distribution between events](image2)

8.2 Improvement

To eliminate these false positives we can define our session to be more precise, “session” - will be the same as before, but with addition element that two consecutive events are in same session only if the interval is less than 30 minutes.
9 Markov Chain with memory

We executed the same algorithm with memory of 2 (second order) and got the same sequence of events, but when we ran the same algorithm with memory of 3, got the same sequence of events with 2 more false positives:

```
[{
    'OS': 'Windows',
    'app': 11161,
    'browser': 'Outlook Client',
    'event': 'EVENT_O365_OUTLOOK_SEND',
    'timestamp': '2014-02-13 03:19:20.206000',
    'username': 5
},
{
    'OS': 'Windows',
    'app': 11161,
    'browser': 'Outlook Client',
    'event': 'EVENT_O365_OUTLOOK_MESSAGE_VIEW',
    'timestamp': '2014-02-13 03:19:24.143000',
    'username': 5
},
{
    'OS': 'Windows',
    'app': 11161,
    'browser': 'Outlook Client',
    'event': 'EVENT_O365_OUTLOOK_SEND',
    'timestamp': '2014-02-13 03:19:56.131000',
    'username': 5
},
{
    'OS': 'Windows',
    'app': 11161,
    'browser': 'Outlook Client',
    'event': 'EVENT_O365_OUTLOOK_SEND',
    'timestamp': '2014-02-13 03:20:13.277000',
    'username': 5
}]
```

```
[{
    'OS': 'Windows',
    'app': 11161,
    'browser': 'Outlook Client',
    'event': 'EVENT_O365_OUTLOOK_SEND',
    'username': 5
},
{
    'OS': 'Windows',
    'app': 11161,
    'browser': 'Outlook Client',
    'event': 'EVENT_O365_OUTLOOK_MESSAGE_VIEW',
    'username': 5
},
{
    'OS': 'Windows',
    'app': 11161,
    'browser': 'Outlook Client',
    'event': 'EVENT_O365_OUTLOOK_SEND',
    'timestamp': '2014-01-21 21:00:34.722000',
    'username': 5
},
{
    'OS': 'Windows',
    'app': 11161,
    'browser': 'Outlook Client',
    'event': 'EVENT_O365_OUTLOOK_SEND',
    'timestamp': '2014-01-21 21:00:35.780000',
    'username': 5
}]
```

When we used more memory we got worse results, which make sense, because in a session the next event depends strongly on the last-event/one before, but not more of that, because of the common use of web browsing and the number of parameters we need to estimate grows exponentially with higher order (the higher the order, the less reliable we can expect our parameter estimates to be).
Part V

Summary

In this paper we were representing 3 algorithms which we implemented on audited SaaS data. We were able to find anomalies using those algorithms and to improve them using rules and different features, although this approach does not provide hermetic security, but rather provides leads for security personnel to research; we present to the security personnel a sorted list of anomalous events to investigate, and they provide feedback to exclude certain combinations of features (logout events, iOS usage by certain users, etc.). As always in security it’s a cat-and-mouse game, those algorithms will prevent from current malwares that are unaware of those heuristics to be detected, when malwares will progress and will be aware of such systems, it will be harder to detect them and require more algorithms to be ran on the audited data.
Part VI

Bibliography

1. A SaaS security proposal - Adallom

2. A Markov Chain Model of Temporal Behavior for Anomaly Detection - Nong Ye

3. Coursera - machine learning, Andrew NG


5. Machine Learning - A Probabilistic Perspective, Kevin P. Murphy