SAS® GLOBAL FORUM 2020 #SASGF

Paper 4521-2020

Common Points of Compromise Detection Using Network Analysis

Prathaban Mookiah, Hamoon Azizsoltani, Ian Holmes, and Tom O' Connell
SAS Institute Inc.

ABSTRACT

Common points of compromise (POC) are entities such as merchants and websites that suffer from a security breach that results in the compromise of a multitude of cards, online credentials, and so on. The form of such breaches can range from sophisticated security attacks on large merchants that are well publicized to an opportunistic staff member harvesting details on a regular basis. Such forms of fraud continue to thrive, and therefore POC detection continues to be a key tool in combating various forms of banking fraud such as card fraud and online banking fraud. In this paper, we introduce a process that is designed to identify POCs by combining techniques from network analysis and machine learning, engineered completely within the SAS® ecosystem.

INTRODUCTION

In this paper, we describe a process that is designed to identify POCs by combining techniques from network analysis and machine learning to proactively monitor for potential POCs. The task of identifying POCs has been formulated as a semi-supervised machine learning problem whose goal is to identify anomalies that are consistent with the behavior of POCs. We chose a semi-supervised approach so that the process can be more generally applied, because a list of identified POCs might not be available at every institution, depending on the fraud channel.

The process was engineered entirely within the SAS ecosystem by using some of the most versatile and powerful procedures available in SAS® Visual Data Mining and Machine Learning for machine learning and network analytics.

TRADITIONAL APPROACH

We briefly describe a simplified version of the traditional method for identifying POCs using a card fraud problem because it provides a clear way to understand and appreciate the need for POC detection. The objective, in this case, is to identify potential merchants that were the source of compromise in the recent past. Error! Reference source not found. illustrates the method.

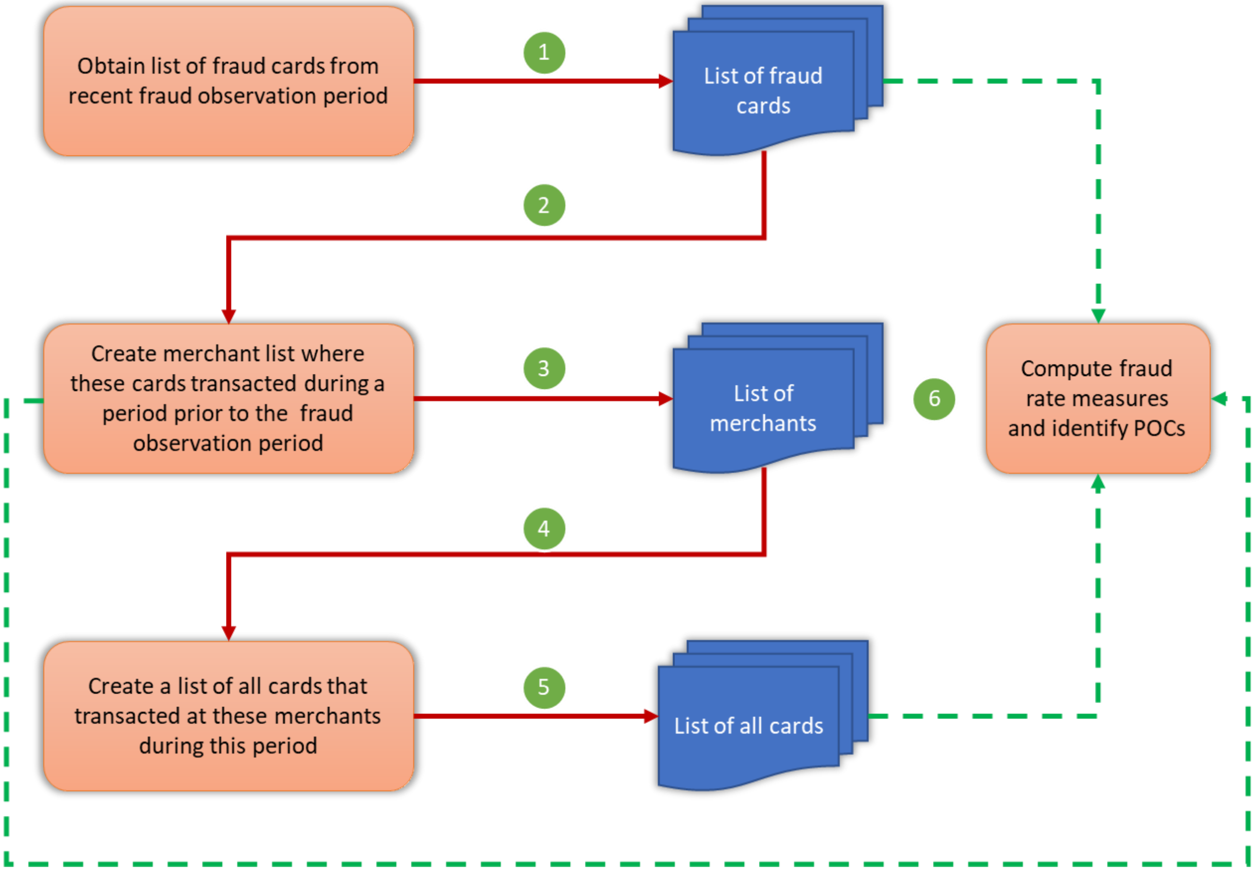
Obtain list of fraud cards from recent fraud observation period
1
List of fraud cards
2
Create merchant list where these cards transacted during a period prior to the fraud observation period
3
List of merchants
6
Compute fraud rate measures and identify POCs
4
Create a list of all cards that transacted at these merchants during this period
5
List of all cards

Figure 1: Traditional Approach to Detect Common Points of Compromise

As shown in Figure 1, this approach relies on known frauds to trace back to the POCs, which makes it a reactive and supervised process. Detection may also occur too late, depending on the lag in fraud reporting and lack of fraud reporting in certain situations.

Moreover, this method does not rely on any interconnections that exist between the entities. Network analytics allows us to introduce various behavioral and risk metrics that are associated with the relationship between different entities present in the data, which is particularly powerful for payment fraud problems. The traditional approach works only when a direct relationship is observed between the monitored entity and the compromised entity.

PROCESS DETAILS

Our process is based on the premise that in certain types of compromises, when an entity is under a state of compromise, there will be anomalous behavior and relationship-related signals that can be indicative of the compromise. We rely on behavioral analytics and network analytics to capture these signals in order to actively alert on potential compromises.

Figure 2 illustrates the key steps of the process.

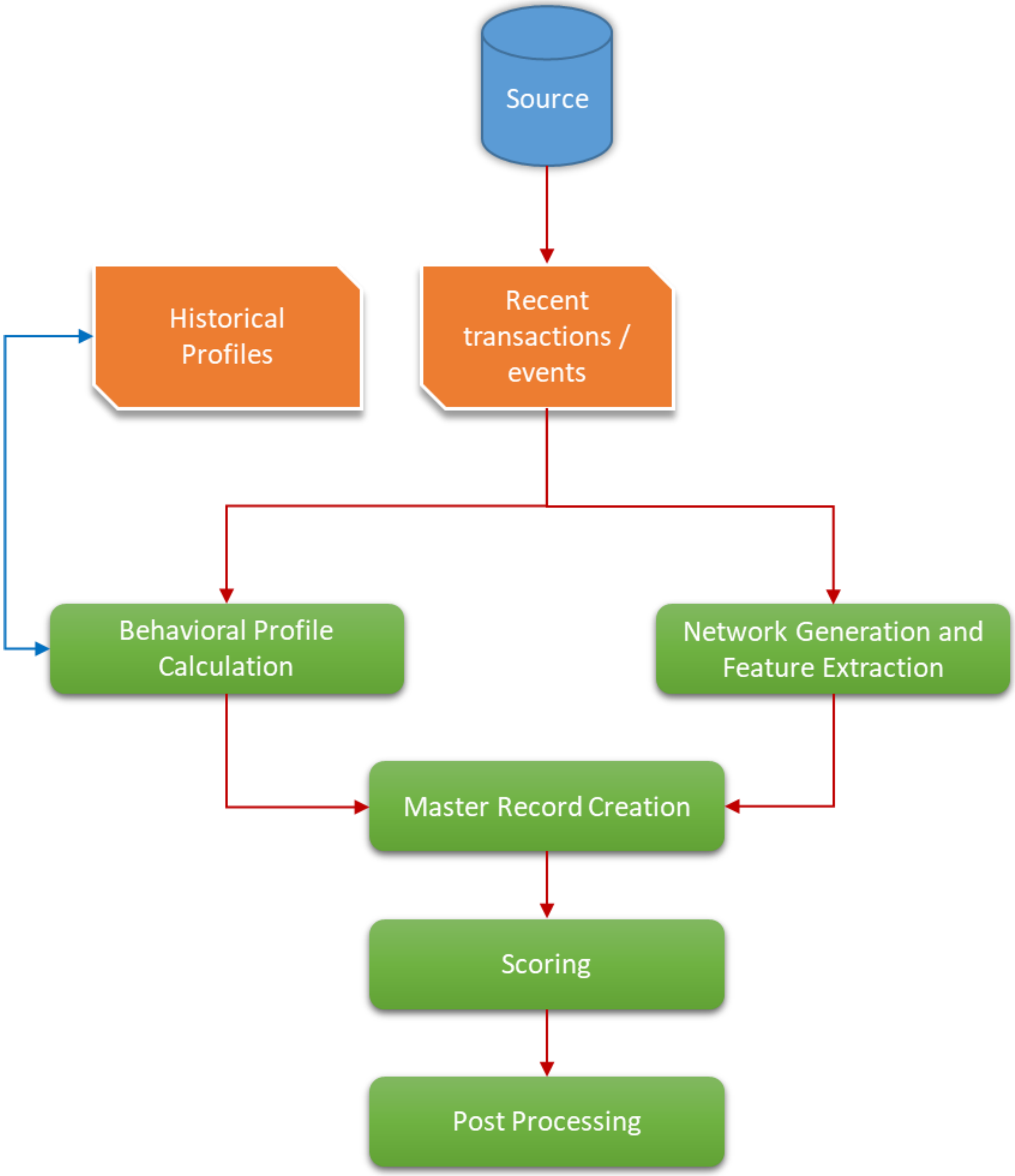
Source
Historical Profiles
Recent transactions / events
Behavioral Profile Calculation
Network Generation and Feature Extraction
Master Record Creation
Scoring
Post Processing

Figure 2: Process Flow

The process is designed to be executed at regular intervals. During each run, either a subset or all instances of the entity type being monitored is analyzed and rank ordered in terms of a risk score that indicates the probability that the entity is in a state of compromise. Additional business strategies can be layered on this ordered list to select the actual POCs on which to focus for investigation and further action. The following sections provide additional details on each step of the process.

DATA EXTRACTION

The objective of this process is to proactively monitor entities to determine whether they are in a state of compromise. Therefore, the most recent events (which typically are events that occurred since the last time the process was run) are extracted from the appropriate data source.

NETWORK GENERATION AND FEATURE EXTRACTION

The most recent events are used to generate entity networks that describe the various interconnections that exist between the events. We extensively use the NETWORK procedure that is available in SAS Visual Data Mining and Machine Learning for this purpose. We consider these types of information:

- Network level information: number of disjoint networks, number of cliques and communities within the network, the attributes of the networks
- Entity level information: centrality measures, articulation points

Currently, we limit the process to consider only the networks that are manifested within the most recent events, mainly due to computational performance reasons. However, it is relatively straightforward to consider a larger subset of prior events (or all prior events) to form comprehensive networks that can conceivably improve the analytic performance of the process (albeit with more computational overhead).

GENERATE BEHAVIORAL PROFILE

In this step, the events are collapsed into an entity level profile that contains various behavioral summaries about each entity. In addition to the current behavioral profile, a historical profile is also maintained for each entity. At the end of each run, this historical profile is updated based on the current profile and persisted for use when the process is run next time. Depending on the fraud channel, an exhaustive set of features can be derived from the behavioral profile. Examples of classes of features include:

- Number of events and numbers of different types of events
- Quantity-based statistics at various chronological bins
- Various indicator variables (such as "had incoming payment" or "associated with foreign bank")
- Various measured deviations of the listed metrics compared to prior time periods

CREATE MASTER RECORD

This step simply combines the various features from the networks and behavioral profiles to form a master record for each instance of an entity type. This record serves as the input to the final step, where each entity is scored. If necessary, any additional third-party data pertaining to the entities may be applied here to enrich the final dataset.

SCORING

The final step in the process is scoring each entity in order to produce a risk score that is indicative of the risk of it being compromised. The exact mechanism used to produce the score is interchangeable, as long as the chosen mechanism consumes the master record from the prior step and produces a risk score (or a measure that indicates risk).

We currently utilize a semi-supervised machine learning model to produce an anomaly score. We use a semi-supervised approach because it produces a generalized out-of-the-box model that works across a wide range of use cases without having to rely on a set list of identified POCs that might not be available at every institution (depending on the fraud channel).
Other alternatives to produce the risk assessment include supervised models, scorecard-based approaches, or a set of empirical business rules.

POST - PROCESSING

The final post-processing step involves using rules written by business users to further refine the score. Because the score produced in the previous step is typical of most fraud prevention departments, these additional rules incorporate other factors and actions into the risk assessment.

MODELING APPROACH

Our modeling approach is based on a semi-supervised technique that is composed of a series of unsupervised steps followed by a final supervised validation step. These are the key steps:

1. unsupervised non-linear dimension reduction step
2. unsupervised cluster generation
3. cluster identification that relies on previously labeled POCs

This semi-supervised approach provides a framework to generate well-generalized models that can easily be fine-tuned for individual use cases.

DIMENSIONALITY REDUCTION

For multiple reasons, the master record dataset from the previous step has too many dimensions to be suitable for effective model development in machine learning. Therefore, we begin with an unsupervised feature reduction step. We rely on the well-known autoencoder approach for dimensionality reduction. This approach maps the high dimensional input variables into a low dimensional feature space.

Feature reduction provides several benefits that are specific to our problem:

1. Most clustering algorithms work more accurately in a reduced feature space.
2. Inference in a reduced feature space is more robust to the input noise, so the results are more generalizable.
3. Dimensionality reduction can provide much better visualization into the data for analysis and investigation.

The reduced dimensional space now provides a good basis to generate clusters.

CLUSTER GENERATION

Clustering is a well-known class of unsupervised techniques for classification problems. In our approach, we apply a two-step process to generate clusters. In the first step, we estimate the optimum number of clusters in the reduced dimensional feature space. In the next step, we use a standard clustering approach to form the actual clusters. When properly designed, these clusters represent the different behavioral classes among the entity type being monitored, with one or more clusters representing the actual POCs.

CLUSTER IDENTIFICATION

We use a supervised approach to identify the clusters that capture the actual POCs. A small number of labeled POCs are used to distinguish the POC clusters from the other clusters.

These steps are performed iteratively until we arrive at a final set of clusters that can classify the POCs with the desired level of accuracy. Standard model development practices were observed in order to achieve a good balance between accuracy and generalization.

FINAL MODEL

The final model consists of these two components as shown in Figure 3.

1. The encoder: In production, each master record is passed through the encoder to obtain a reduced dimensional representation.
2. Centroid for the cluster that represents POCs: In the simplest method, the distance from the reduced dimensional representation to this cluster is computed and transformed into a risk score that is based on the distance. However more complex methods can also be utilized to compute scores based on other business and analytical considerations.

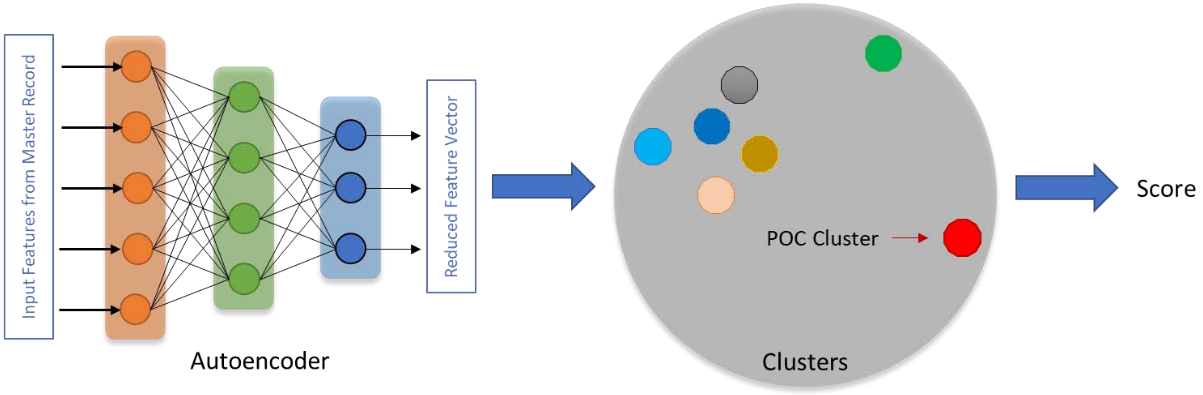

Figure 3: Final Model Structure

PROCESS VALIDATION METHODOLOGY

We validated the results of the designed process on a relatively large credit card transaction dataset that spanned six months and contained about 18 – 20M transactions per day. In order to create a reference set of POCs, we applied the traditional approach to generate a set of suspected POCs. We used the last month as the fraud observation period and the preceding five months to identify POCs. This resulted in approximately 1,500 merchants, based on these filtering criteria:

- fraud rate > 1%
- minimum number of cards transacted during the five-month period > 100
- maximum number of cards transacted during the five-month period < 100,000

These conditions were selected because they are close to typical thresholds that are used in practice.

Our process was validated by comparing the overlap between the list and the top-scoring merchants. Though an arbitrarily high level of overlap can be achieved by tuning the model, care should be taken not to bias the model in order to yield a high degree of accuracy against the test dataset, which can result in an overfitted model.

However, in practice, it is imperative that the model and the process should be custom tuned based on a test dataset that is specific to the fraud channel and the organization. This tuning is needed to achieve a good balance between optimal performance and generalization.

CONCLUSION

This paper provided an overview of a process designed to proactively detect common points of compromise in various fraud channels. The process combines network analytics, behavioral analytics, and unsupervised machine learning to look for abnormal behavior in monitored entities to detect points of compromise.

REFERENCES

SAS Institute Inc. 2019. *SAS Visual Data Mining and Machine Learning 8.5: The NETWORK Procedure*. Cary, NC: SAS Institute Inc. Available https://go.documentation.sas.com/?docsetId=casmlnetwork&docsetTarget=titlepage.htm&docsetVersion=8.5&locale=en

SAS Institute Inc. 2019. *SAS Visual Data Mining and Machine Learning 8.5: Procedures*. Cary, NC: SAS Institute Inc. Available https://go.documentation.sas.com/?docsetId=casml&docsetTarget=titlepage.htm&docsetVersion=8.5&locale=en

Ian Holmes, Global Security Intelligence Practice, SAS Institute Inc How to uncover common point of purchase. https://www.sas.com/en_us/insights/articles/risk-fraud/common-point-of-purchase.html

CONTACT INFORMATION

Your comments and questions are valued and encouraged. Contact the author at:

Prathaban Mookiah
SAS Institute Inc.
prathaban.mookiah@sas.com