Evaluating Online Payment Transaction Reliability using Rules Set Technique and Graph Model

Trung Le[1], Ba Quy Tran[2], Hanh Dang Thi My[3], Thanh Hung Ngo[4]

[1] *GSR, Information System Lab., University of Information Technology – VNU-HCM – HCM – Vietnam*
[2] *Engineering – HelloPay Company – HCM – Vietnam*
[3] *Lecturer, Faculty of Mathematics and Informatics – People's Police University – HCM – Vietnam*
[4] *PhD, Information System Lab., University of Information Technology – VNU-HCM – HCM – Vietnam*

Abstract - ***Online Transaction using Credit/Debit Card is a payment method which is very popular recently. However, transactions on the Internet are not really safe. It is very easy to steal information from those transactions. Banks and insurance companies have consumed both time and cost to trace back transaction and resolve fraud. This paper research the overview of current situation in attack and protection methods about online fraud card in Vietnam, then proposes an expert system model to detect fraud, in which rules are based on the analysis and detection of unusual values in online payment transactions. The model is applied and experimented on graph database.***

Key Words: *Bank Fraud, Online Transaction, Fraud Detection System.*

1. INTRODUCTION

With the strong impact of the scientific and technological progress, especially in the field of internet, in recent years the pace of development of online marketing, e-commerce and online payment methods has become extremely popular. But the recent tech-crimes on card also tend to increase, leading to the loss of billions of dollars per year worldwide, and affect the reputation and brand of business organizations and banks. Among financial frauds on the internet, credit card fraud is the act exists longstanding, widespread, and the most dangerous fraud due to widespread uses and its convenience. The risk of card activity is becoming more diverse and complex as stealing information, making fake cards, card trapping, money trapping, reversing transactions....

With fraudulent transactions increasing in number and sophistication, counterfeits resemble legitimate transactions more, models and simple inspection techniques are currently not effective and intelligent enough to detect unusual transactions. Therefore, the construction of the fraudulent warning system need to be sophisticated enough to ensure that card transactions are done safely is essential. This is the practical problem attracting many researchers and security companies.

Detecting fraud is seen as a classification problem, in which credit card transactions will be categorized into 2 types: legitimate or fraudulent. Although detection of fraud is a problem posed very early, but currently only a few researches have been published. Part of the difficulty is mainly due to the limitation of the actual data to facilitate researchers to experiment because banks are not willing to provide data of customer transactions. The existing experimental data are very scarce for security purpose. However, some experiments have successfully applied of data mining techniques different [1, 2] as outlier detection, self-organizing maps, neural networks, Bayesian classifier, support vector machine, Artificial immune system, Genetic Algorithm K-nearest neighbor, and hidden Markov model.

The application of graph database for detecting frauds is a trend of some recent research [3, 4, 5]. With the characteristics of the graph consists of vertices and relations, graph database has many advantages in performing compared with relational database. Therefore, we can easily detect fraudulent sites with high accuracy and has the ability to detect subtle suspected fraud during real time, which the relational database very difficult to implement effectively.

This paper will propose a list of logging from online transaction history and how to store those fields in the form of a graph database. This paper will also propose fraud detection rules model and fraud detection algorithm based on defined rules. The rules are defined based on the

experience of experts and can easily be updated and edited on time.

2. ONLINE TRANSACTION GRAPH MODEL

Through the study, research group conducted a survey to investigate online transaction flows [9, 10, 11, 12], collected and analyzed samples of the transaction history (**Fig-1**) provided by some co-operate online payment companies. Refer advices from many experts, we have synthesized information and propose models of online transactions including necessary components and attributes to construct rules, which is presented in details in following sections.

2.1 Graph database

A graph is a representation of a set of objects where some pairs of objects are connected by links. The interconnected objects are represented by mathematical abstractions called vertices or nodes, and the links that connect some pairs of vertices are called edges. Graph Database [6] is the database management system using graph structure to organize data storage, developed with many advantages over traditional relation database system.

According to [7], the concept of a graph database is often organized by 4 components, such as:

(Nodes, Relationship, Properties, Labels)

In which:

- *Nodes*: A set of vertices in a graph, representing entities
- *Relationships*: A set of relationships in the graph
- *Properties*: A set of attributes of the vertices in the graph
- *Labels*: A set of labels using to group the vertices for each specific object.

2.2 Online transaction model

Based on the structure of the graph database platform, we propose the model of credit card transactions, including the following components:

(Card, Transaction, Attr, Relations, Rules)

In particular, the components are defined as following:

Card: A collection of vertices representing payment cards, labeled as Card.

Payment cards are the most important element in the transaction, all transactions online require payment cards information. Structurally, a payment card is represented by one node on the graph, including basic attributes like:

- *User Info*: Information of card owner (name, address, company, ...)
- *Card Number*: Card identification number. At the security level, this property should be encoded.
- *Card Type:* Card Type, which will have value such as: visa, MasterCard ...
- *Current Amount*: The current money amount left in card.

Transaction: A collection of vertices representing transactions, labeled as Transaction.

Each vertex represents the information of one online transaction using payment cards. When trading online, the purchaser must provide the card information for online payment systems (e.g. pay gate websites), in addition, other important information will be analyzed and stored by the online payment systems. Thus, a transaction will contain a lot of relevant information. One transaction vertex of the graph will contain these following properties:

- *Raw data*: unprocessed transactions data.
- *Status*: The status of the transaction, including the values: safety, suspicion, fraud.
- *Amount*: The amount of money in the transaction.
- *Transaction Time*: The time of the transaction.
- *Merchant ID*: The ID number of merchant.
- *Order ID*: The ID number of current order.
- *Address*: Delivery address.

Additionally, other transaction attributes will be represented by Attr vertices on the graph related to the current transaction.

UPDATE	TIME	AMOUNT	REFNUM	ADDRESS_NAME/STREET	CITY	MERCHANT-ID	ACQINSTID	Status
20140106	173626	505,100	-----	---- ----- ---- ---- ---		---- ----- ---- ---- ---	-----	
20140304	222427	500,000	-----	---- ----- ---- ---- ---	TP HCM	---- ----- ---- ---- ---	-----	
20140304	222144	500,000	-----	---- ----- ---- ---- ---	TP HCM	---- ----- ---- ---- ---	-----	
20140318	122002	200,000	-----	---- ----- ---- ---- ---	HO CHI MINH	---- ----- ---- ---- ---	-----	
20140324	111331	500,000	-----	---- ----- ---- ---- ---	HA NOI	---- ----- ---- ---- ---	-----	Fraud
20140324	111352	200,000	-----	---- ----- ---- ---- ---	HA NOI	---- ----- ---- ---- ---	-----	
20140324	111437	100,000	-----	---- ----- ---- ---- ---	HA NOI	---- ----- ---- ---- ---	-----	Fraud
20140324	111412	150,000	-----	---- ----- ---- ---- ---	HA NOI	---- ----- ---- ---- ---	-----	
20140321	172044	1,000,000	-----	---- ----- ---- ---- ---	HA NOI	---- ----- ---- ---- ---	-----	Fraud

Fig -1. Online transactions log

Attr: a collection of vertices labeled as Attribute, represents attributes of the transaction.
Special attributes of the transaction are represented by a vertex for representing relationship among the transactions, for example, the IP address of a transaction may coincide with the IP address of other transactions, as then, a vertex labeled Attribute will have relations with

many transactions. Relations between vertices labeled Attribute and transactions will be utilized to detect frauds. A vertex is a pair of attributes including:

- *Key*: Name of the attribute.
- *Value*: Value of the attribute.

In which, each Attr vertex A contain a pair (key, value) must satisfy:

$$\exists! A(key, value) \in Attr$$

Relations: A collection of relations "Relationship", represents the relationship among the vertex of the graph. Within the subject, we consider types of relations in the graph such as: *belongs_to, purchased_by, located_in, assigned_by...*

Rules: A set of fraud detection rules, will be described in details in *section 2.3*.

The represent of Online transaction model in graph database includes Card, Transaction, Attr and its relation is demonstrated as **Fig-2**.

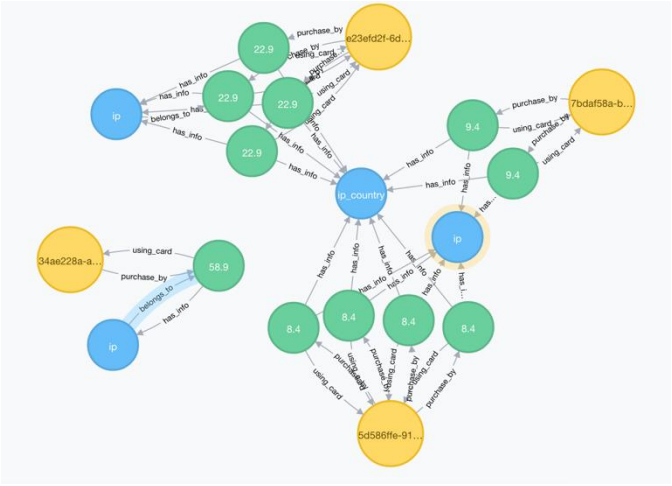

Fig -2. Online transactions represent in graph, 'yellow' nodes are labeled as Card, 'green' nodes are transactions and 'blue' nodes are marked as Attr.

2.3 Model of Fraud Detection Rules

The use of graph databases provides a great advantage in the detection of fraud, by considering the unusual relationship between vertices in the graph, we can detect how many reliable values and how many suspicious values the current transaction has. The decision of abnormal values in a transaction is represented by fraud detection rules - or rules. These rules can be managed and defined by experts. A fraud detection rule is represented by the following components:

(func, trust_value)

In which:

- *func - rule function*: The content of logic code used for detecting fraud based on one or some properties of incoming transaction, is defined by the programming language with Cypher query language [8]. The result of the function is *true* or *false*, corresponding with incoming transaction are suspicious or safety.
- *trust_value - Reliability*: A trust value of the rule defined by expert. Satisfied:

$$0 \leq trust\_value < 1.$$

2.4 Reliability of a transaction to a Rule

If we call t as a transaction, r as a rule, rule result $c = func_r(t)$, we have the reliability of the transaction t to the rule r, called $r(t)$, is determined by:

$$r(t) = \begin{cases} 1, & c = false \\ trusted\_value, & c = true \end{cases}$$

Accordingly, if the transaction t does not satisfy the rule r (not in the case of suspicion r), then t has the reliability $r(t) = 1$. Otherwise, t will have the reliability $r(t) = trusted\_value$.

2.5 Reliability of a transaction to Rules set

Based on the model of credit card transactions, let a transaction $t \in Transaction$, a set of $Rules = \{r_i \mid 1 \leq i \leq |Rules|\}$, we have a subset $R \subset Rules$ in which $\forall r \in R, func_r(t) = \text{true}$. The reliability α of the transaction t in $Rules$, as determined by the formula:

$$\alpha = \sqrt[|R|]{\prod_{i=1}^{|R|} r_i(t)}$$

3. CONSTRUCTING RULES AND FRAUD DETECTION ALGORITHM

Based on the models described and advices from experts in banking, we have built a trial fraud detection rule set and designing detecting card frauds algorithm, along with basic treatments to solve the main problem set for the system. Checking and detecting a fraudulent transaction means checking the reliability of incoming transaction

with each rules in Rules set. Call $t \in Transaction$ as a transaction need to detect, we build a Fraud Detection Rule set and propose detect algorithm as following:

3.1 Constructing Fraud Detection Rule set

Refer advices from many experts, we describe some rule as following:

- R1: There are a large number of transactions was made by one payment card at a moment of time.
- R2: Transaction has the order's address in different country with registered address of the card.
- R3: IP address of the incoming transaction also conduct many other transactions using different payment cards.
- R4: payment card is being used at many IP addresses from different countries.

Fraud Detection Rules are set up via system programming language combine with Cypher query language, based on vertices and relations among the vertices on the graph database, along with basic condition functions in the programming language which is using.

3.2 Reliability calculation algorithm

With built in graph database models and rules set, transaction fraud detection depends on reliability calculation algorithm is described by pseudo code as follows:

Step 1: *Initialization*

```
multi_trusted_val = 1;
match_rule_counter = 0;
reliability  = 0;
```

Step 2: *calculate the multiplication trusted_value of matched rules*

```
Foreach r in Rules
    Begin
        c = r.Func(t);
        if (c == true)
        Begin
            multi_trusted_val *=  r.trusted_value;
            match_rule_counter += 1;
        End
    End
```

Step 3: *calculate reliability*

$$\text{reliability} = \sqrt[\text{match\_rule\_counter}]{\text{multi\_trusted\_val}}$$

3.3 Determining the transaction's status

Reliability α will determine the status of a transaction, as mentioned in *section 2*. Thereby, the transaction status belongs to the thresholds of the reliability α. Within a scope of this paper, we define the thresholds of fraud status as follows:

- $\delta \le \alpha \le 1$: Transaction status is safety, with δ called safety threshold. Those transactions will be processed for payment.
- $\varepsilon \le \alpha \le \delta$: Transaction status is doubt, with ε called doubt threshold. Those transactions will be confirmed before making payment.
- $0 \le \alpha \le \varepsilon$: Transaction status is fraud and these transactions will be refused to pay.

4. IMPLEMENTATION & TESTING

4.1 Implementation

Card fraud detection system is designed to build as an API service and written by programming language Ruby facilitate the definition of the rules, used graph database Neo4j [9]. Schematic design of the system is as following:

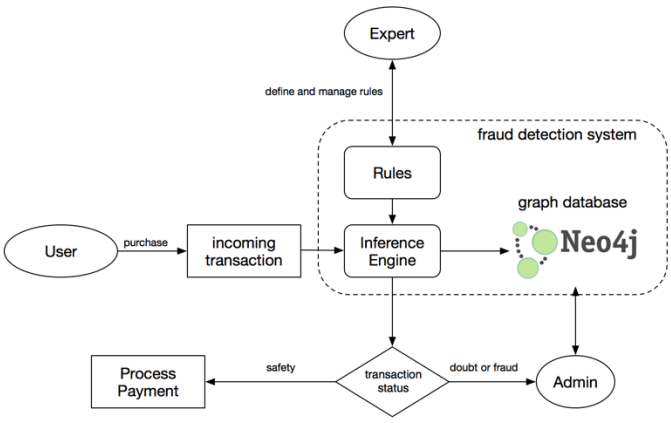

Fig -3. Schematic design of the system.

Fraud detection system is a central part of the card transaction processing system. Experts will define and manage rules in the Rules set. When user purchases something online using credit card, transaction will be processed by the Inference Engine. According to the Rules

set, Inference Engine will assign status for transaction. If transaction status is safety, it will be processed for payment. If transaction status is doubt or fraud, it will be reported to admin to confirm or reject. Finally, the transaction will be saved in graph database.

4.2 Testing dataset

Due to the security and privacy of banking accounts and transactions, there are no real transaction dataset can be found. Our testing dataset includes over 52,000 transactions provided by a private pay gate company. However, because of privacy condition, some fields are missing or encoded. Thus, each transaction in experiment dataset include following attributes:

- *Encrypted email*: User email (encrypted) used for making a payment, which is treated as a key-unique value. Each email can be marked as a card node in system.
- *Encrypted First/Last name*: User name of a transaction.
- *IP address*: the current IP address of user who is making the payment
- *Order ID*: Identification number of current order is in the transaction
- *Session*: a random string represents the current session of user.
- *Amount*: the amount of money in transaction.

In order to have the full experiment of described rule set. There are some field needs to be simulated, such as: Card current amount, transaction date…

4.3 Testing

On testing, the measurement and the accuracy of the card fraud detection algorithm is calculated by turns based on the formula:

$$Precision = \frac{T_{correct}}{E(T)}$$

$$Recall_{safety} = \frac{T_{safety}}{R_{safety}(T)} \quad ; \quad Recall_{fraud} = \frac{T_{fraud}}{R_{fraud}(T)}$$

Where:

- $T_{correct}$: The set of transactions in which each transaction has been assigned status exactly.
- T_{safety}: The set of transactions detected correctly as safety status.
- T_{fraud}: The set of transaction detected correctly as fraud and doubt status.
- E(T): The set of all transaction.
- R_{safety}(T): The set of transactions have ‘safety’ status.
- R_{fraud}(T): The set of transactions have ‘fraud’ or ‘doubt’ status.

Based on counting after testing on the set of transactions include 52.000 sample of transactions (in which there are 50,370 transactions marked as safety, 1,630 transactions marked as doubt/fraud), with safety threshold $\delta = 0.8$, doubt threshold $\varepsilon = 0.6$, the precision and recall of the fraud detection system are: $Precision = 76.61\%$; $Recall_{safety} = 76.5\%$, $Recall_{fraud} = 81.16\%$. The average of processing time is about 400 milliseconds per transaction.

According to the result, we found that the system basic functions which detect transaction fraud worked effectively and fast, so our system can be deployed for real-time processing.

5. CONCLUSION

Through this paper, we have described the solution to develop a fraud detection system in online payments using credit cards. The designs based on the model include vertices and the relationship between the vertices in a graph database as well as the design and engineering solutions to detect fraud based on fraud detection rules model put into unusual vertices of the graph, the system shows the flexibility, ease of upgrade and edit the information. Besides, the query is processed fast, fraud detection system is easily deployed to handle transactions in real-time. However, the current rules set still need to be populated and upgraded with more details in order to increase the accuracy in future works.

ACKNOWLEDGEMENT

This research is funded by Vietnam National University HoChiMinh City (VNU-HCM) under grant number TX2015-26-02.

REFERENCES

[1] L. Delamaire, H. Abdou, J. Pointon: Credit card fraud and detection techniques: a review. Banks and Bank Systems, Volume 4, Issue 2 (2009) 57-68.

[2] K. Tripathi, A. Pavaskar: Survey on Credit Card Fraud Detection Methods, International Journal of Emerging Technology and Advanced Engineering, ISSN 2250-2459, Volume 2, Issue 11, November 2012.

[3] Leman Akoglu: Anomaly, event, and fraud detection in large network datasets, Proceeding WSDM '13 Proceedings of the sixth ACM international conference on Web search and data mining 2013, 773-774.

[4] L. Akoglu, C Faloutsos: Graph-based Irregularity and Fraud Detection. IEEE International Conference on Data Mining, Brussels Dec 2012.
[5] Chau, D.H., Faloutsos, C., Tong, H., Hong, J.I. Gallagher, B. and Eliassi-Rad, T. GRAPHITE: A visual query system for large graphs. In Proc. ICDM 2008, 963-966.
[6] R. Angles, C. Gutierrez, Survey of graph database models, Journal ACM Computing Surveys (CSUR) Surveys Homepage Archive Volume 40 Issue 1, February 2008 Article No. 1
[7] J. Miller: Graph Database Applications and Concepts with Neo4j, Proceedings of the Southern Association for Information Systems Conference, Atlanta, GA, USA March 23rd-24th, 2013.
[8] F. Holzschuher, R. Peinl, Performance of graph query languages: comparison of cypher, gremlin and native access in Neo4j, Proceeding EDBT '13 Proceedings of the Joint EDBT/ICDT 2013 Workshops 2013, 195-204.
[9] Stripe Co. – fraud document, https://stripe.com/docs/fraud.
[10] Paypal.com – Pay flow Payment Gateway, https://www.paypal.com/au/cgi-bin/webscr?cmd=_payflow-gateway-overview-outside
[11] Threatmetrix - Cyber security, https://www.threatmetrix.com/cyber-security/
[12] EMV vs. Magnetic Stripe Transaction Flow, https://www.vantiv.com/vantage-point/safer-payments/emv-vs-magnetic-stripe