Time Based Graph Mining to Detect Insider Threats

Niraj Rajbhandari , Advisor: Dr. William Eberle
Department of Computer Science, Tennessee Tech University

ABSTRACT

- Insider threats such as sabotage, theft, espionage, fraud and competitive advantage are accomplished by abusing access to the organization's network, system or data, theft of materials and mishandling of physical devices and negatively affects the confidentiality, integrity or availability of the organization's information system.
- We try to identify anomalous insider activity which can be malicious in the email communication of the organization.
- We use graph mining approach that incorporates the time element of the email communication to identify these anomalous instances.

RESEARCH OBJECTIVE

- The aim of this work is to mine the graph that represents email communication in an organization to identify suspicious activities in the communication.

INSIDER THREATS

- Threats from authorized users:
  - Compromise the network
  - Deliberate malicious exploitation or destruction of data
- Difficult to distinguish from normal behavior
  - Hampered the organization's business activities
- Different strategies available to tackle
  - Access control frameworks
  - Anomaly detection strategies
  - Expert-informed suspicious behavior classifiers

DATASET

- Publically available Enron Corpus Dataset [2]
- 600,000 emails from 149 employees
- Used SQL dump version of the dataset [3]
- 4 Data Tables: ***employeelist***, ***message***, ***recipientinfo***, and ***referenceinfo***
- Do not use ***referenceinfo*** table

DATA PREPROCESSING

- Added a fifth table, ***link_forwarded_message*** which links the forwarded message to its original message.
- Divided the timestamp of email communication into **date** and **time** component.
- Further divided **date** component into two groups: *weekday,* and *weekend*
- Further divided **time** component into 6 buckets with 4 hours in each bucket: *early-morning, morning, afternoon, evening, night,* and *late-night*

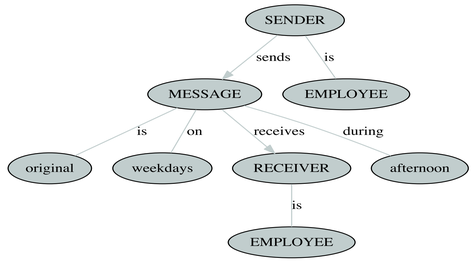

Figure 1: Graph representation of email communication

EXPERIMENTAL SETUP

- Convert the parsed graph into a graph stream
- Mine the stream to identify the suspicious activities
  - Uses window-based approach
  - Creates a ***"scaled-down"*** sample
  - Identify anomalies in the sampled graph for each window
  - Sample Size: 16% of original graph
  - For different components: ***Graph Parser***, ***Graph Stream Generator, Graph Sampler,*** and ***Graph Based Anomaly Detection Tool***[1]

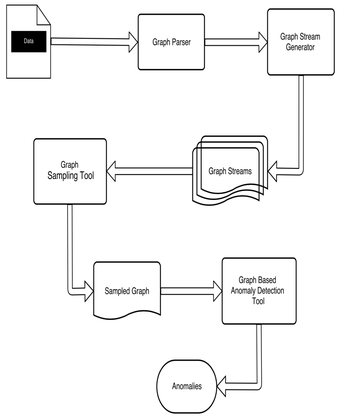

Figure 2: Experimental Setup Architecture

Algorithm 2 Intelligent-PIES(Sample Size n, Stream S)

Input: Sample Size n, Graph Stream S
Output: Sampled Graph $G_s \leftarrow (V_s, E_s)$

```
V_s ← ∅, E_s ← ∅
V_count ← 0
t ← 1
while graph is streaming do
    {u,v} ← e_t
    if |V_s| < n then
        if u ∉ V_s then V_s ← V_s ∪ {u}
        if v ∉ V_s then V_s ← V_s ∪ {v}
        E_s ← E_s ∪ {e_t}
    else
        P_t ← V_count
        Draw r from Continuous Uniform {0,1}
        significantEdge ← Is e_t significant?
        if r ≤ p_t and significantEdge then
            v_i ← ∅
            v_j ← ∅
            while v_i is ∅ do
                draw i from discrete Uniform [1,|V_s|]
                if V_s[i] is not important then
                    v_i ← V_s[i]
                end if
            end while
            while v_j is ∅ do
                draw j from discrete Uniform [1,|V_s|]
                if V_s[j] is not important then
                    v_j ← V_s[j]
                end if
            end while
            if u ∈ V_s then V_s ← V_s ∪ v_i, drop node v_i with all its incident edges
            if v ∈ V_s then V_s ← V_s ∪ v_j, drop node v_j with all its incident edges
        end if
        if u ∈ V_s and v ∈ V_s then E_s ← E_s ∪ e_t
    end if
    V_count ← V_count + 2
    t ← t + 1
end while
```

Figure 3: Intelligent-PIES

EXPERIMENTAL RESULTS

- Able to detect 3 different anomalous instances
- Identified suspicious activity of ***"vice-president"*** of the company sending email to the ***"in-house lawyer"*** at an unusual time, i.e. late at night (around 4 in the morning).
- One instance identified was False Positive.
  - Just a normal communication with ***"in-house lawyer"*** during regular office hour is flagged as anomaly

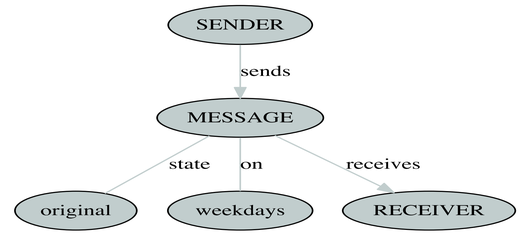

Figure 4: Normative Pattern from Sampled Graph

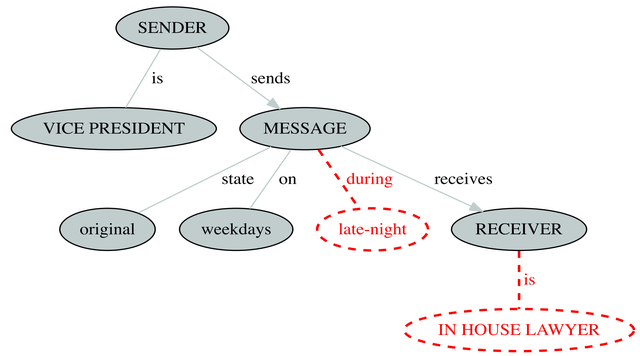

Figure 5:Suspicious activity with respect to the time element

CONCLUSION

- Able to identify suspicious activity with respect to the time element of email communication
- Able to identify anomalous instances from a very small sample of the original graph
- Smaller processing time to detect anomalies
  - Because of smaller sampled graph to process.

FUTURE WORKS

- Integrate the sentiment of the email content into the graph
- Identify suspicious activity from the email content
- Perform experiments on different sample size to identify the tradeoff between processing time and loss in accuracy

REFERENCES

1.Eberle, William and Holder, Lawrence. "Discovering structural anomalies in graph-based data." In "Data Mining Workshops, 2007. ICDM Workshops 2007. Seventh IEEE International Conference on," pages 393–398. IEEE, 2007.

2. https://www.cs.cmu.edu/~enron/

3.http://www.ahschulz.de/enron-email-data

4.MONDAL, S. AND BOURS, P. 2016. Combining keystroke and mouse dynamics for continuous user authentication and identification. Identity, Security and Behavior Analysis (ISBA), 2016 IEEE International Conference on 1-8. .