SecGen kick-off meeting

21/06/2023

Hélène Orsini

CentraleSupélec UMR IRISA Inria

Introduction

Auto-ML Pipeline for Network Attack Incident Detection and Classification

Focus

- Machine Learning driven Network Traffic Flow based Intrusion Detection System (IDS)

Challenges

- Data
- Label
- Scalability
- Interpretable ML pipelines
- Concept drift

Challenges

1. **Data labeling (C1)**: most of the time no label
2. **Amount of data (C2)**: Reduce labeling effort with cluster
3. **Adaptability (C3)**: Follow up behavior change (Concept-drift)

Model evolution

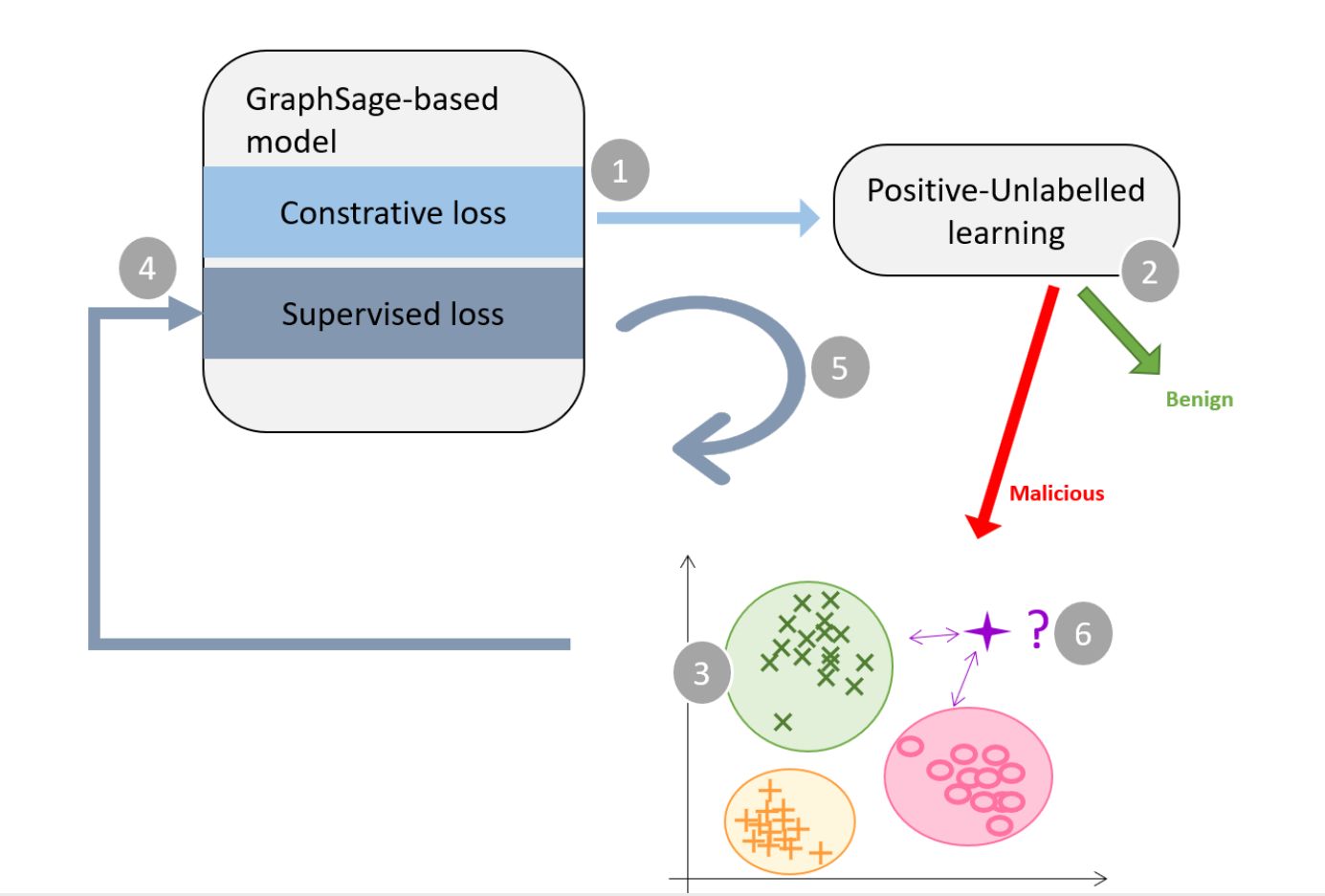


Figure: Pipeline

Training

1st round: 1, 2, 3

2nd round: 4, 5

Testing

6

Preparation of the data

In: Communication traffic in binetflow format

Preparation of the data: feature vectors (X), Links vector (L)

- **Encoded feature** For each communication: get encoded feature vector
- **Link** Link all the communications from the same source ip
- **Link** A communication is the n past communications

Cluster the communications from X

- Label some benign data
- For each point i in a cluster, record its nearest neighbors ($\mathscr{P}_i$), others are not neighbor ($\mathscr{N}_i$)

Out: Dataset of communications with X and L, some benign label, and $\mathscr{P}_i/\mathscr{N}_i$

1st Round - Step 1 - Embedding - C1 & C2

In: Dataset of communications with X, L, and $\mathscr{P}_i/\mathscr{N}_i$

Contrastive learning

Machine learning technique used to learn the general features of a dataset without labels by teaching the model which data points are **similar or different**.

M : **GraphSage model**, X_i: all the embeddings, $\mathscr{P}/\mathscr{N}$: positive/negative pairs of netflow data

$$L = \arg\min_{M} \min_{X_i, X_j \in \mathscr{P}_i} \|M(X_i) - M(X_j)\|^2 + \min_{X_i, X_k \in \mathscr{N}_i} (C - \|M(X_i) - M(X_k)\|^2) \quad (1)$$

Out: Embeddings

1st Round - Step 2 - Detector - C1 & C2

In: Embeddings, some benign labels

Pu learning

Train a classifier to distinguish between positive and negative.

Learning phase: **Positive and Unlabelled** (*only some of the positive examples in the training data are labeled and none of the negative examples are*)

Out: Semi-supervised detector

1st Round - Step 3 - Classifier - C1 & C2

In: Malicious traffic out of the semi-supervised detector

Cluster – > Classify

classification, gather same botnet behavior in the same clusters

Get more labels

Identify some malicious label

Out: Classifier, some malicious label

2nd round - Step 4 - Embedding

In: Previous trained embedder + some labels from malicious clusters

Retrained GraphSage model with some label from cluster:

$$L = L_{contrastive\_learning} + L_{supervised\_learning} \tag{2}$$

Out: New embedding with label enforcement

2nd round - Step 5 - Detector

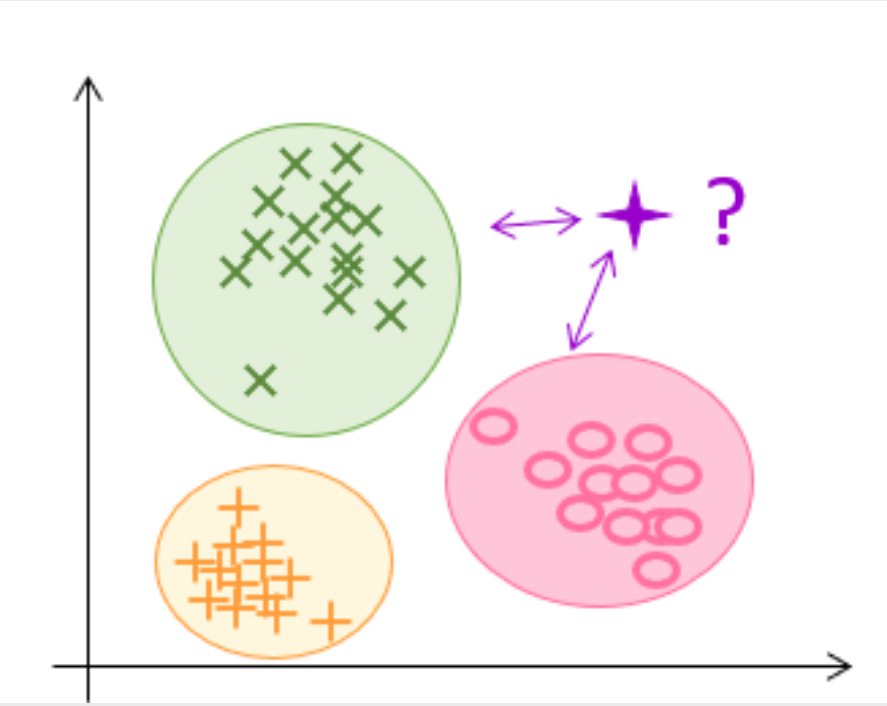
In: New Embeddings

Update PU-learning classifier with the new embeddings

Out: Semi-supervised updated detector

Testing phase - Step 6 - C3

In: Malicious traffic from the detector



Concept Drift

Too far from other clusters ?

New point ?

Out: Updated Mutli-class Classifier

Experience

Ongoing - 1st round

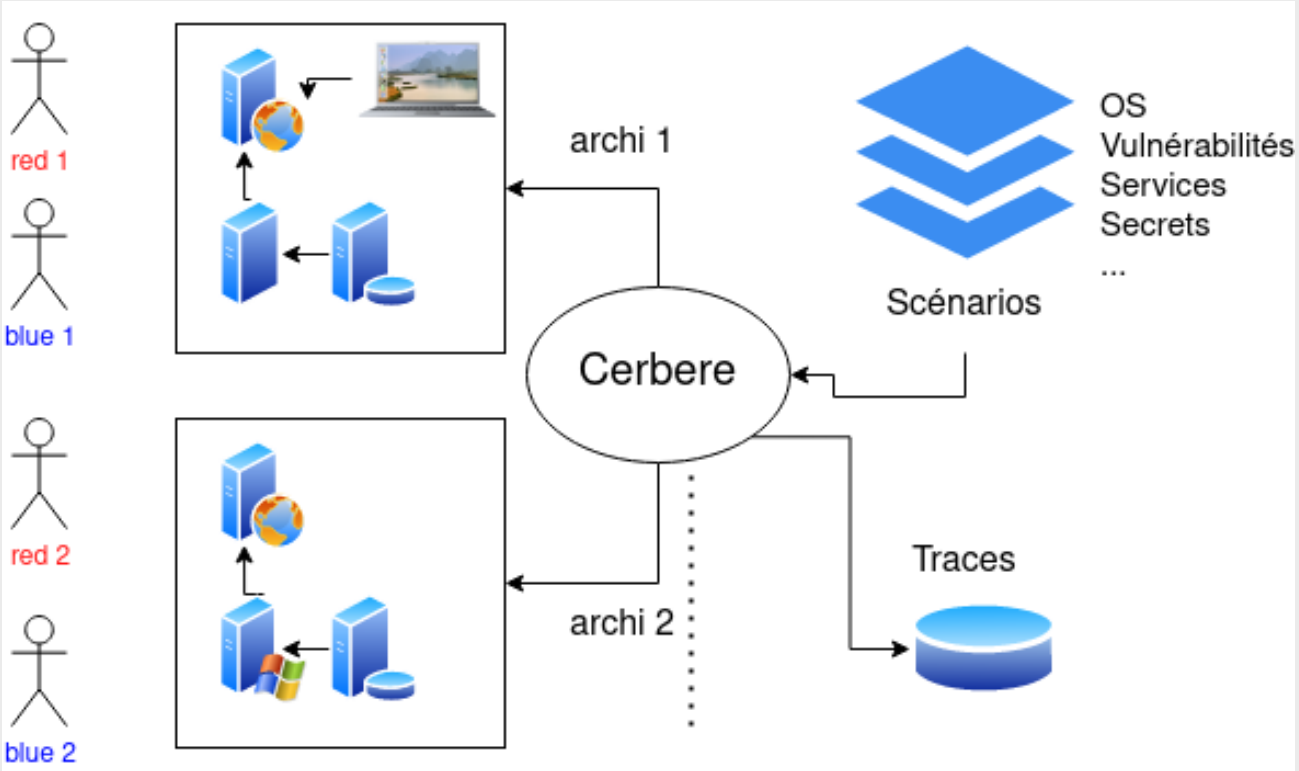
- Preparation: implemented
- Step 1: implemented
  (play with parameters: loss_type, numbers of neibg, feature dim, lr, dropout)
- Step 2: implemented and tested (Acc : 96%, F1-score : 96%)
- Step 3: implemented

Next step - 2nd round

CERBERE

Un projet entre plusieurs doctorants

- Automatically **deploy** multiple scenario **variations**
- Ensure the **existence** of **exploitable** attack paths
- Collect/Investigate attack/behavior **traces**



Honeypot - Internship

OBJECTIF : collect a Honeypot dataset from the Hoplab platform

1. **State of the art** on honeypot and its use in IT security.
   Output: description of the Honeynet: how many machines? how to make it credible (false network life, false system life)? Survey/restoration hygiene?
   Time: record the drift –> at least one month
2. Check recent vulnerabilities exploited by botnet
3. **Design and set up** honeypots to attract attackers.
4. Configure honeypot to **record** attack data and malicious behavior.
5. **Analyze** the data collected.

Next Steps / Planning

- *Short term (after August)*: one paper (ACNS, CSF, ASIACSS, ...)
- *Short term (Summer)*: Play with the dataset from CERBERE, Hoplab Honeypot
- *Long term*: Add the challenge of explainability or AutoML
- *Long term*: Prepare thesis

Conclusion

- Finish all the pipeline this summer to publish
- Continue investigation
- Write

Conclusion

Thank you for your attention

Questions ?

References