Three new challenges on security data generation

Pierre-François Gimenez
PIRAT team

SecGen plenary meeting
March 6th, 2024

CentraleSupélec

Context

Context

- Intrusion detection systems (IDS) seek to detect attacks in IT environment
- We lack good quality datasets, mostly due to privacy and confidentiality issues
- Synthetic generation of data with machine learning is a way to create new datasets

Some new challenges

In this presentation, I'll propose three new research questions on security data generation

CentraleSupélec

Generation for concept drift evaluation

Context

- Concept drift describes a change of distribution over time
- In network security, it could mean a new equipment, a change of network topology, etc.
- Such concept drift typically affect negatively anomaly detectors, because the legitimate behavior evolve
- Robustness to concept drift is rarely evaluated, in part because there is no suitable dataset

Goal

Design a method to generate network traffic that change over time

CentraleSupélec

Proposed approach

Extension of current work

- We could extend Joscha and Adrien's work to condition generation with some network topology
- A possibility is to map existing pattern to roles (e.g., Web server traffic, user laptop)
- That way, we could add or remove machines with specific roles (e.g., add a new Web server)

Difficulties

The evaluation methodology of such generator is not straightforward

Who?

We got funding for a PhD on this topic, starting in October 2024 (do you know any good candidate?)

CentraleSupélec

Causal learning

Context

- To react automatically to attack, it can be useful to understand the causal dependencies between flows: if A causes B, I can avoid B by preventing A from happening
- Such causal learning could also be very useful for explainable IDS

A live testbed

- Joscha and Adrien's method works on fixed datasets, so it's difficult to extract causality
- Our team is working with a testbed (called SOCBED)
- We could use this environment to verify causality hypotheses (akin to the "do-operator")

Goal

Design a method to actively identify causal relationships between network events

CentraleSupélec

Proposed approaches

Proposed approach 1

- Start from the patterns identified by Joscha and Adrien's method and verify whether the correlations are causality
- This would act as a "second stage", and could potentially improve (reject or complete) previously found patterns

Proposed approach 2

- Use reinforcement learning to learn the dependencies between protocols
- Could help identifying patterns with longer temporal dependencies

Difficulties

- So far we worked on metadata, but to elicit an answer we need to send actual packets
- A way to circumvent this issue could be not to add data, but only drop communications

CentraleSupélec

System log generation

Context

- System log are a useful source of data to detect attacks
- Their basic structure is fixed, details differ
- They are written by many different applications with various syntax

```
march 04 09:13:06 optimistic-owl systemd[2724]: dunst.service: Main
process exited, code=dumped, status=5/TRAP
january 12 15:26:11 optimistic-owl acvpnagent[458374]: Loading preferences
for pf from profile P-anyconnect-Inria.xml
```

Goal

Design a method to generate synthetically system logs to evaluate system IDS

CentraleSupélec

Proposed approach

Step 1: mine patterns from vector representation

- Parsers can transform well-known log formats into triplets `(source, action, object)`
- We could mine patterns from such triplets with a method similar to what has been proposed by Joscha and Adrien
- From these patterns, we could afterward generate such triplets

Step 2: generate logs

To transform triplets into actual logs

- for well-known logs: we can code it directly
- for non-documented logs: I propose to use LLMs

And beyond: generate system logs and network data jointly

CentraleSupélec

Conclusion

Three research questions we could explore together

Challenge 1: generation for concept drift evaluation

Condition generation by network architecture to generate evolving datasets and evaluate IDS's robustness to concept drift

Challenge 2: causal learning

Leverage network test bed for active learning of causal relationships between events

Challenge 3: system logs generation

Generate system logs to evaluate system IDS as well, possibly thanks to LLMs