Continuous Pattern Detection on Streaming Data: Present and Future

SUTANAY CHOUDHURY

JOINT WORK WITH: KHUSHBU AGARWAL, SHERMAN BEUS, DANIEL DOHNALEK, GEORGE CHIN

Workshop on “Graphs and Security”, Microsoft BlueHat, Redmond WA, 2018
Introduction and Outline

Part 1: What we have now (StreamWorks)
- Stream based reasoning – problem definition and algorithms
- Cyber use cases for streaming analytics
- Demonstration

Part 2: The future for Autonomous Cyber systems
- Extending stream-based reasoning towards autonomous operation
- Conceptualizing tasks for benchmarking
Asking a Different Question: Tell me when …

- Continuous monitoring of streaming data
  - Instead of traditional “find me all things that happened in past 24 hours” – move to a “tell me when X happens” paradigm
  - Example: Standard feature from a stock brokerage – “Tell me when MSFT goes to $150 and more than 50 million shares were traded”
  - Cyber equivalent: “Tell me when a chain of 3 logins are detected with increasing privileges?”

Malware?

Malware installed on developer’s workstation

DB Exploit?

Project DB compromised by lateral movement

IP Theft?

Backdoor opens to CEO’s machine when he accesses Project DB
Specific Problem Definitions

- Find the pattern from event or alert stream
  - Who is involved as source (outside) and target (inside)?
  - What type of interaction is involved?
  - What type of kill-chain behavior is represented by the pattern?

- Given (source, target and kill-chain behavior), project possible instances of the kill-chain
  - Use the Knowledge Graph to generate high probability pathways with explanations

- Interfacing with the Human Expert
  - Summarize contextual patterns to the human expert
  - Learn patterns of reviewed alerts
What are property graphs, and how are they different from other representations?

Social Networks

Multi-Relational RDF Graphs

Property Graphs
Example Graph Models for Cyber Data
Query and database are both represented as graphs
- Report embedding of query pattern instances in a dynamic graph
- **Pattern Queries in Action**
  - “Tell me when a chain of 3 logins are detected with increasing privileges?”
StreamWorks Demonstration – Botnet Control and Command
StreamWorks Demonstration – Exfiltration
StreamWorks Demonstration – Watering Hole

Watering Hole

- HOST
- BAIT
- CONTROLLER

Time Window: 00:00:00

HOSTS: IP...
BAIT: IP...
CONTROLLER: IP...

MSG 1: HTTP
MSG 2: HTTP
MSG 3: HTTP
The StreamWorks Architecture

Assume selectivity order
\[ \text{sel}(A) > \text{sel}(B) > \text{sel}(C) > \text{sel}(D) \]

Search for a Subgraph corresponding to B or D only where their sibling (A or C) is matched.
Selectivity. We compute the selectivity of all *primitives* by counting their frequencies

Frequency counting is expensive beyond 2-edge subgraphs

Demo: StreamWorks

- **Visual Querying:** Real users should not need to learn a new query language to use the system.
Use in 3 simple steps – No need to learn SQL/SPARQL

- We provide templates to users to specify questions and auto-generate queries
- **Step 1:** Choose a query
Step 2: Specify constraints, choose properties to specialize the queries and hit Go!

This notifies the Apache Spark cluster to register the query and start pattern matching on the stream.
Step 3: When matches are found in the stream, results are sent back to the web server for visualization.

Showing a botnet match. Color gradient used to indicate confidence.
Finding the Needle in a Haystack

- Embedded multiple embeddings of exfiltration into a large-scale dataset
Exfiltration
Demo
Explaining Top Matches

Finding a pattern is the first step. The context is more important.

- Given an alert on host Dev1, provide explanations such as "Dev1 frequent connects to service DB2. CE0’s machine frequently connects to DB2"
Introducing the Knowledge Graph
Building Knowledge Graph for Cyber

A Knowledge Graph that learns and stores behavioral summary about entities of interest

- **Role mining**: grouping systems based on their functional roles
- **Rules**: Relational constraint using properties of entities
- **Event prediction models**: recurrent neural network based approaches to score an event

Demo: Explaining Top Matches
The Future: Towards a Passive AI for Cyber

D1. Learn and generate workflows such as one with these two queries

Siri: Good morning, John. I’ve detected a privilege escalation exploit on System A.

John: Show me all systems communicating with system A in the last 24 hours.

Siri: Here you go.

John: Have there been any other recent report of similar exploits?

Siri: Yes. US-CERT reports two incidents at facility X in the last 2 hours. Both incidents involved an attack from location Y.

John: Show me a list of users of system A with admin privileges who have logged in over the last 24 hours.

Siri: Here you go. Also, one of these users, Alice also logged on to System X multiple times in last 24 hours.

John: Disable all accounts for user Alice.

Siri: Disabled all accounts for user Alice. Here is a list of services that may be impacted. Sending a text to Alice.

John: Thank you, Siri.

Siri: Always nice working with you, John.

D2. Auto-generate multiple queries and execute them

D3. Filter query results and summarize important findings

D4. Evaluate potential side effect of actions
How do we evaluate AI systems operating in a streaming environment?

- Develop a benchmark of tasks and dataset

- **Proposing 5 “Cyber IQ” levels representing progressive levels of complexity**

  - **IQ level 1: Find Anomalous Events of Interest**
    - Can perform basic tasks such as detecting unusual events (VPN logins from multiple locations for same user)

  - **IQ level 2: Learn from human experts and build its Knowledge Graph**
    - Observe data and generate questions for the human expert
      - Example: Which systems/services have more strategic importance?
      - What does action X accomplish? Which systems does this affect?
      - Don’t ask, but observe. Try to observe the impact/success of action X in steering the system back to normalcy.
Passive AI (level 3)

▶ IQ level 3: Work with human experts

- Start sending recommendations “In past, we saw case X on days A, B and C, and you took actions P and Q. Q had higher success rate.”

- Explanations: “You took actions P and Q in situations like X. However, this time we are also seeing strong presence of features M and N that were not present in X before.”
Autopilot (Level 4 and 5)

► Autopilot (Level 4)
- Develop a conservative "autopilot mode"
- Generate reports providing reasoning for every action chosen
- It has to be better than shutting down everything in the event of an anomaly, but smartly using tools such as throttling traffic to certain VMs or isolating them

► Meta-System: A system to audit the primary
- Understand when human and machines underperform
- Compare votes of the human expert and that of the machine
- Inject purposeful disruptions to test robustness against unseen problems
A tale of two graphs

- **A fast moving graph** – Dynamic graph updated in near real-time and maintained as sliding window in time
- **A slowly changing graph** – The Knowledge graph reflects what we learn about behavioral patterns in the data

Applications: Build complex systems with the graphs as their memory

- **Today**: Continuously maintain the memory, search and serve top-k queries with summarization
- **Future**: Build complex systems using graphs algorithms and machine-learning techniques as tools that evolve the memory and operate on it
Motivating Example

Let’s take a very simple action (John, an employee working on a DOE project launches a job on a cluster) and try to categorize that as normal or unusual.

Classifying someone as an Insider Threat builds on multiple indicators. Categorization such as above is a basic building block in that process.
What do we know about John?

Red facts are summaries obtained from streaming data
Blue facts are static information
Let’s expand by adding data from HR database and project publications.
Tell me more about his Projects
What do we know about the specific network activity?