ONTOLOGY-DRIVEN DATA SEMANTICS DISCOVERY FOR CYBER-SECURITY

Author: Sarah Kushner Advisor: Marcello Balduccini, PhD Institution: Drexel University

DREXEL UNIVERSITY College of Computing & Informatics

Abstract

We present a software architecture for data semantics discovery, capable of extracting semantically-rich content from human-readable files without prior specification of the format. Human-readable files come in a massive variety of formats. The architecture, based on work at the intersection of knowledge representation and machine learning, includes machine learning modules for automatic file format identification, tokenization, and entity identification.

The process is driven by an ontology, a formal hierarchy of interrelationships between domain-specific concepts and their properties. The ontology also provides a layer of abstraction for querying the extracted data.

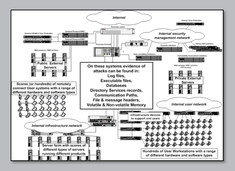
This architecture can be applied in a variety of domains. However, we focus on cyber-forensics applications, aiming to allow the parsing of log files, for which there are no readily-available parsing and analysis tools. We also aim to aggregate and query data from multiple, diverse systems across large networks.

The key contributions of our work are: the development of an architecture that constitutes a substantial step toward solving a highly-practical open problem, the creation of one of the first comprehensive ontologies of cyber assets, and the demonstration of a non-trivial combination of declarative knowledge specification and machine learning.

Use Case

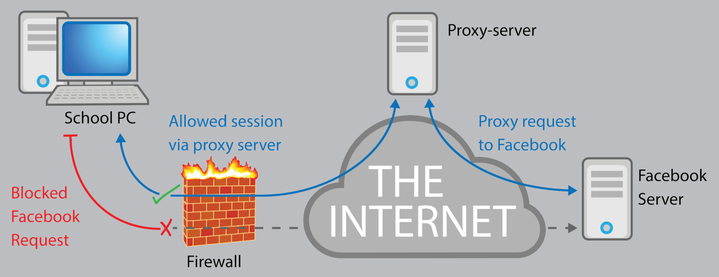
Cyber attack on a very large network:
→A malicious e-mail is received somewhere on the network.
→The recipient of the e-mail opens the attachment, unaware that it is a virus.
→The virus establishes a DNS (Domain Name Server) tunnel towards a server with the domain name "cyberattacks.com"

Security analyst:
→Are there indications that this attack may have occurred on my network?

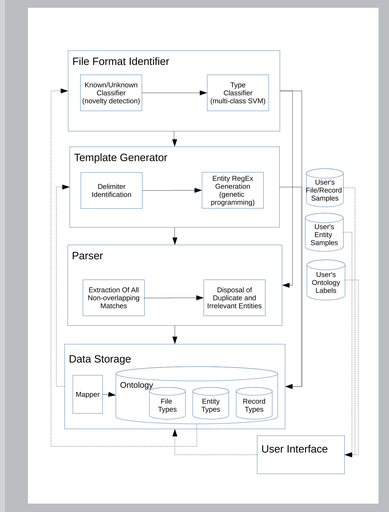


In a complicated network, threats and breaches can be hard to detect.

A simple network security example shows how a firewall can be bypassed.



System Architecture



Input:
→Ontology linking file types, record types, entity types.
→Concepts organized at various levels of abstraction

→Samples: (attached to ontology concepts)
→Files types of interest.
→Record types of interest.
→Entity types of interest.

Queries are
→High-level, spanning over all available knowledge
→Resolved using the ontology and reasoning

File Types

1) Determining if file type is known:
→one-class SVM to recognize as "known"

2) Identifying the file type:
→several "traditional" pairwise SVM classifiers to create a multi-class classifier
→labels are class names from the ontology

Features:
→N-grams of space-delimited tokens (3/4-grams)

Parsing

```
client 192.168.157.1#5544: query: maliciousserver.com IN AXFR +T (192.168.157.129)
client 192.168.157.5: invalid request
client 192.168.157.3#5544: query: microsoft.com IN AXFR +T (192.168.158.12)
```

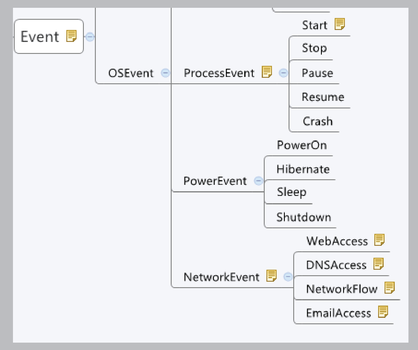
1) Delimiter identification
→variety of different line formats
→heuristic similarity measure used to group similar lines
→delimiters for each line group

Metrics: # occurrences of each punctuation character;
characters between occurrences.
Heuristics: minimum standard deviation in per-line count
largest standard deviation in characters between occurrences.

2) Regular Expression Generation
→genetic programming to narrow down regular expressions based on their accuracy

192.168.157.3 ⟶ \d*\.\d*\.\d*\.\d*

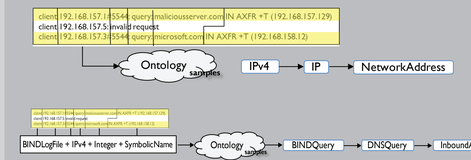
Cyber Assets Ontology

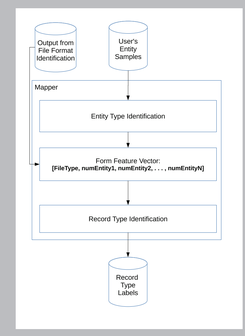


An ontology is basically a hierarchy that shows the relationships between different elements.

For this project, we created the first ontology of cyber-security related assets. Any identified record or entity types will be a label from the ontology.

Record and Entity Types





Querying

```
SELECT R1, R2, R3 WHERE
R1 is a mailRecord,
  R1.contains (emailAddress, .net),
  R1.contains (DateTime D1),
R2 is a dnsQueryRecord,
  R2.contains (DateTime D2),
  D2 > D1,
  R2.contains (domainName, *cyberattacks.com),
  R2.contains (networkAddress, victimPC),
R3 is a dnsQueryRecord,
  R3.contains (DateTime D3),
  D3 > D2,
  R3.contains (domainName, *cyberattacks.com),
  R3.contains (networkAddress, victimPC)
```

The query is independent of application, format used and location where info is stored.

BIND? MS DNS server?

Same server?
Is **victimPC** IPv4?
IPv6? MAC?

Results

We have had rather successful performance of identification algorithms.

	Precision	Recall	F-Measure
File Format Identification	0.9791	0.9808	0.9799
Record Type Identification	0.835	0.8438	0.8394
Entity Type Identification	0.8279	0.7819	0.8042

Table 1. Supervised learning performance

Acknowledgements

Thank you to Marcello Balduccini, PhD and Jacquelin Speck, MS for the guidance and ideas throughout the process.