SEMI-AUTOMATED CYBER THREAT INTELLIGENCE (ACT)

Workshop – FIRST Conference 2018

Martin Eian and Fredrik Borg
Goal

To collect and organize our knowledge of threats to make it useful
Data and Information

Legend
- Sample Hash
- Domain
- Filename
- Path
- Campaign
- IP Address
Knowledge

MATCH p=(n:Domain)-->(o:IP)<--(m:Domain) RETURN p
Semi-Automated...

- Analysis
- Enrichment
- Information Sharing
- Countermeasures
Semi-Automated Cyber Threat Intelligence (ACT)

The main objective of the research project is to develop a platform for cyber threat intelligence to uncover cyberattacks, cyber espionage and sabotage.

The project will result in new methods for data enrichment and data analysis to enable identification of threat agents, their motives, resources and attack methodologies.

In addition, the project will develop new methods, work processes and mechanisms for the generation and distribution of threat intelligence and countermeasures, to stop ongoing and prevent future attacks.
Data Model

• Objects
  - Global
  - Example: IP address

• Facts
  - Connected to a single object or multiple objects
  - Immutable
  - Timestamped
  - Owner
  - Role-based and explicit access control
  - Backed by evidence and comments
## Models, Taxonomies and Vocabularies

- **MITRE ATT&CK**  
  - [https://attack.mitre.org](https://attack.mitre.org)
- **MITRE PRE-ATT&CK**  
  - [https://attack.mitre.org/pre-attack/](https://attack.mitre.org/pre-attack/)
- **MISP galaxy**  
  - [https://github.com/MISP/misp-galaxy](https://github.com/MISP/misp-galaxy)
- **STIX 2.0 vocabularies**  
  - [https://oasis-open.github.io/cti-documentation/](https://oasis-open.github.io/cti-documentation/)
- **Ryan Stillions’ DML model**  
Current OSINT Sources

• APTNotes
  - https://github.com/aptnotes/data

• APT & CyberCriminal Campaign Collection
  - https://github.com/CyberMonitor/APT_CyberCriminal_Campaign_Collections

• RSS Feeds
  - Infosec blogs

• mnemonic PassiveDNS
  - https://passivedns.mnemonic.no/

• VirusTotal
THE ACT PLATFORM
Platform Architecture Version 0.01
Platform Architecture – Core technologies

- Cassandra
- elasticsearch
- Apache Nifi
- Apache TinkerPop
- OPEN NLP
Platform Architecture – Workflow orchestration

- Originally developed by NSA
- Open sourced and transferred to the Apache Foundation in 2014
- Manage flows of data supporting a large number of inputs and outputs:
  - HTTP, FTP, SCP, Kafka, Elasticsearch, JMS, Syslog, MongoDB, Hadoop, Cassandra, SMTP, POP3, etc
Platform Architecture – Graph database

• Looked into existing graph databases, but they lacked proper fine granular permissions (and many of them had commercial licenses that could not be used in the research project)

• Apache Tinkerpop implemented on top of Cassandra/Elasticsearch

• Graph queries opens up a range of possibilities that is not possible on a flat data structure
ACT Core

Backend

REST API

GUI

Apache TinkerPop

Cassandra

elasticsearch

mnemonic
## API - Swagger

<table>
<thead>
<tr>
<th>Method</th>
<th>Path</th>
<th>Operation</th>
</tr>
</thead>
<tbody>
<tr>
<td>POST</td>
<td><code>/v1/fact</code></td>
<td>Create a new Fact</td>
</tr>
<tr>
<td>GET</td>
<td><code>/v1/fact/user/fact/access</code></td>
<td>Retrieve a Fact's ACL</td>
</tr>
<tr>
<td>POST</td>
<td><code>/v1/fact/user/fact/access/subject</code></td>
<td>Grant a Subject access to a Fact</td>
</tr>
<tr>
<td>GET</td>
<td><code>/v1/fact/user/fact/comments</code></td>
<td>Retrieve a Fact's comments</td>
</tr>
<tr>
<td>POST</td>
<td><code>/v1/fact/user/fact/comments</code></td>
<td>Add a comment to a Fact</td>
</tr>
<tr>
<td>POST</td>
<td><code>/v1/fact/user/fact/retract</code></td>
<td>Retract an existing Fact</td>
</tr>
<tr>
<td>GET</td>
<td><code>/v1/fact/id</code></td>
<td>Retrieve a Fact by its UUID</td>
</tr>
<tr>
<td>POST</td>
<td><code>/v1/factType</code></td>
<td>Create a new FactType</td>
</tr>
<tr>
<td>GET</td>
<td><code>/v1/factType</code></td>
<td>List available FactTypes</td>
</tr>
<tr>
<td>PUT</td>
<td><code>/v1/factType/uuid/id</code></td>
<td>Update an existing FactType</td>
</tr>
<tr>
<td>GET</td>
<td><code>/v1/factType/uuid/id</code></td>
<td>Retrieve a FactType by its UUID</td>
</tr>
<tr>
<td>GET</td>
<td><code>/v1/object/type/fact</code></td>
<td>Retrieve an Object by its type and value</td>
</tr>
<tr>
<td>POST</td>
<td><code>/v1/object/type/fact/facts</code></td>
<td>Retrieve Facts bound to a specific Fact</td>
</tr>
<tr>
<td>POST</td>
<td><code>/v1/object/type/fact/traverse</code></td>
<td>Traverse the Object/Fact graph starting at an Object identified by its type and value</td>
</tr>
<tr>
<td>POST</td>
<td><code>/v1/object/search</code></td>
<td>Search for Objects</td>
</tr>
<tr>
<td>POST</td>
<td><code>/v1/object/traverse</code></td>
<td>Traverse the Object/Fact graph after performing an Object search</td>
</tr>
<tr>
<td>GET</td>
<td><code>/v1/object/uuid/id</code></td>
<td>Retrieve an Object by its UUID</td>
</tr>
<tr>
<td>POST</td>
<td><code>/v1/object/uuid/id/facts</code></td>
<td>Retrieve Facts bound to a specific Object</td>
</tr>
<tr>
<td>POST</td>
<td><code>/v1/object/uuid/id/traverse</code></td>
<td>Traverse the Object/Fact graph starting at an Object identified by its UUID</td>
</tr>
<tr>
<td>GET</td>
<td><code>/v1/objectType</code></td>
<td>List available ObjectType</td>
</tr>
<tr>
<td>POST</td>
<td><code>/v1/objectType</code></td>
<td>Create a new ObjectType</td>
</tr>
</tbody>
</table>
API – Python library (act-api on pypi)

---

**Project description**

**python-act**

python-act is a library used to connect to the ACT platform.

The platform has a REST api, and the goal of this library is to expose all functionality in the API.

**Objects and Facts**

The act platform is built on two basic types, the object and fact.

Objects are universal elements that can be referenced uniquely by its value. An example of an object can be an IP address.

Facts are assertions or observations that ties objects together. A fact may or may not have a value describing further the fact.

Facts can be linked on or more objects. Below, the seenIn fact is linked to both an ipv4 object and report object, but the hasTitle fact is only linked to a report.

<table>
<thead>
<tr>
<th>Object type</th>
<th>Object value</th>
<th>Fact type</th>
<th>Fact value</th>
<th>Object type</th>
<th>Object value</th>
</tr>
</thead>
<tbody>
<tr>
<td>ipv4</td>
<td>127.0.0.1</td>
<td>seenIn</td>
<td>report</td>
<td>report</td>
<td>cbc80bb5c0c0f8944bf73...</td>
</tr>
<tr>
<td>report</td>
<td>cbc80bb5c0c0f8944bf73...</td>
<td>hasTitle</td>
<td>Threat Intel Summary</td>
<td>n/a</td>
<td>n/a</td>
</tr>
</tbody>
</table>
• Implemented
  - Core platform
  - API
  - GUI
  - Workflow orchestration
  - Graph queries

• Github project
  - https://github.com/mnemonic-no/act-platform
  - License: ISC (BSD compatible)

• Python API wrapper
  - https://pypi.org/project/act-api/
WORKSHOP - INTRODUCTION
Before We Start
Introduction 1
Introduction 1 – Right Click / Left Click
Introduction 1 – History, Layouts and Filtering
Introduction 1 – Fact Types
Introduction 2

Try the following object queries and explore the graph:

- threatActor: Sofacy
- technique: Credential Dumping
- tool: foosace
- hash: da2a657dc69d7320f2ffc87013f257ad
Task 1

Try the following object query:

- ipv4: 40.112.210[.]240

What is the role of this IP address? Find any related Threat Actor(s).
Introduction 3 – Threat Actor Aliases
Task 2

Try the following object queries in sequence:

• ipv4: 85.25.100[.]104 – expand reports
• ipv4: 74.201.40[.]28
• ipv4: 74.201.40[.]32

What are the roles of these IP addresses? Find any related Threat Actors.
Task 3: Find the Report

https://blog.talosintelligence.com/2018/05/VPNFilter.html
Introduction 4 – Create/Retract Fact

Fact
- Fact type: threatActor:Alias
- Fact value: -

Objects
- Object Type: threatActor
  - Object value: Lazarus Group
  - Direction: BiDirectional
- Object Type: threatActor
  - Object value: Silent Chollima
  - Direction: BiDirectional

Options
- Access mode: Public
- Comment: Added by Martin Eliar

CANCEL  SUBMIT
Bonus Task:

Investigate the domain name rannd[.]org.
WORKSHOP – GRAPH QUERIES

With Great Power Comes Great Responsibility
Graph Query 1
Graph Query 2 – Show Edges
Graph Query 3 – 2 hops
Graph Query 4 – Filter Edges (Facts)
Graph Query 5 – Filter Nodes (Objects)
Graph Query 6 – Warp Speed
Task 4: Find the Report

The fqdn fsw.adobeus[.]com is seen in one report. A sinkhole IPv4 address is also seen in the same report. What is the title of the other report mentioning that sinkhole IPv4 address?

Hint: Fact Type ‘seenIn’
Task 4 Solution
g.outE('seenIn').otherV().inE('seenIn').otherV().hasLabel('ipv4') .where(outE('isSinkhole')).outE('seenIn').otherV().path().unfold()
Graph Query 7 – Unique Tool Usage
EXERCISES
Exercises

There are two Threat Actors known to use certutil.exe. Which other tool do they have in common?

Which Threat Actor is associated with the domain name www.eye-watch[.]in?

How many DNSRecord facts are connected to the IP address 8.8.8.8?

How many Threat Actors are known to originate (sourceGeography) from France (location)?

How many of the Threat Actors known to originate from Russia use the tool psexec?
Exercises - Answers

There are two Threat Actors known to use certutil.exe. Which other tool do they have in common? **mimikatz**

Which Threat Actor is associated with the domain name www.eye-watch[.]in? **Lazarus Group**

How many DNSRecord facts are connected to the IP address 8.8.8.8? **18741**

How many Threat Actors are known to originate (sourceGeography) from France? **1**

How many of the Threat Actors known to originate from Russia use the tool psexec? **3**
FURTHER WORK
New Information Sources

• Security alerts
• Incidents
• Reputation lists
• Malware analysis systems
• WHOIS
• MISP feeds
• STIX feeds
• ...

Graph Analytics

- Post. doc. @ UiO
- Post. doc. @ NTNU
Information Sharing

- Mechanism for sharing schema
- Format (STIX?)
- Trust models
Trust and Confidence

- Trust (source)
- Confidence (fact)
- Subjective Logic (quantify uncertainty)
GUI Improvements

• Context menu
  - Pre-defined graph queries
  - Download report
  - ...

• Timelines
• Share workspace
• Prune graph