Knowledge Graphs for Cybersecurity Reasoning

Alice Cheatum, Brandon Richards, Nicklas Cahill, Carter Kitelinger, Michael Watkins, Micah Gwin

Introduction

With cybersecurity threats constantly emerging, up-to-date knowledge is required while working in the cybersecurity field. Cybersecurity researchers, incident responders, and system administrators need to be able to efficiently query information about a specific software, malware, threat, etc., as well as new and emerging ones. Sorting through relevant security news articles can be challenging and time consuming.

Objectives:

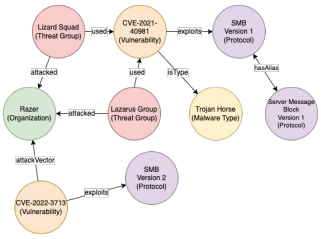
- Efficient querying of cybersecurity entities and their relations
- Assist cybersecurity researchers and incident responders in contextualizing latest threats
- Streamline decision-making for cybersecurity risk management

Overview

The solution to this problem is a system that can collect articles, parse and store that information, then generate a knowledge graph. The graph can then be efficiently queried to streamline the research and decision-making process for cybersecurity professionals.

Knowledge Graph Overview:

- Represents a network of
- real-world entities and relationships between them
- Consists of 3 main components: Nodes, Edges, and Labels
- Used by search engines to complete search queries and give expanded data



Nodes in the graph should be entities of interest in the cybersecurity domain. For example, "Organization", "Threat Group", and "Vulnerability". The process of extracting these entities is called Named-Entity Recognition (NER). The edges in the graph should be the relations between these entities. The process of extracting these relations is called Relationship Extraction (RE). Together these tasks are called Information Extraction (IE).

Methodology

Methodology used to delegate work and manage progress: Agile.

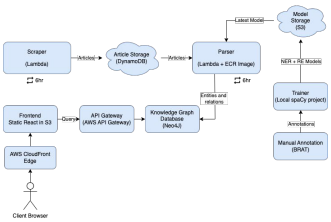
- All work was divided into sprints and stories.
- The team used a Kanban board to assign group members cards to work on.

Architecture design:

- Functionality of the project is divided between modules (Parser, Scraper, etc).
- Modules communicate by passing data through a variety of inputs and outputs (files, databases, json api request).

Implementation

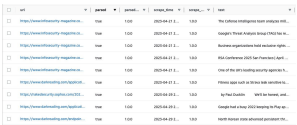
- The training component of this project is a spaCy project that is configured to train an NER and RE model from manually annotated articles.These models are then uploaded to S3 to be used when parsing articles.
- The scraper is a Python module responsible for scraping from the source list and storing articles in database. Web scraping is done the scrapy library. The scraper is built and deployed to AWS Lambda where it runs on a schedule to provide up-to-date articles.
- The parser is a Python module responsible for using a pre-trained model to parse entities and relations from article text. It then stores the entities and relations as nodes and edges in a knowledge graph. The spaCy library provides a framework for performing natural language processing on text using our model. An image is then built using Docker and uploaded to AWS ECR to later execute.
- The knowledge graph is stored in a Neo4J database running on an AWS EC2 instance. It is able to be queried using HTTPS and JSON using AWS API Gateway.
- The frontend is a static webapp written in React and TypeScript. It includes a visual query builder, queries the Neo4J Knowledge Graph, and displays the results. The frontend is stored in AWS S3 and distributed by AWS CloudFront.



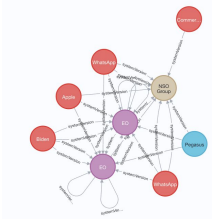
Results

- Create NER and RE training pipeline using spaCy.

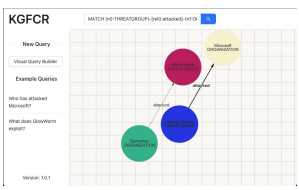
- Scrape articles from source list on schedule and store in NoSQL database.



- Build multi-threaded modular parser that reads input from article storage, parses article text using trained models, and stores output in Neo4J knowledge graph database.



- Deploy React frontend with query builder which queries Neo4J and displays nodes and edges.



Conclusion

- News on cybersecurity can often be hard to find and understand.
- Extracting entities and relations from article text can be done using Named-Entity Recognition and Relation Extraction.
- Condensing the information into a knowledge graph formats cybersecurity topics in a coherent overview of new and emerging threats.
- Allows users to better comprehend large amounts of cybersecurity information.