UC San Diego

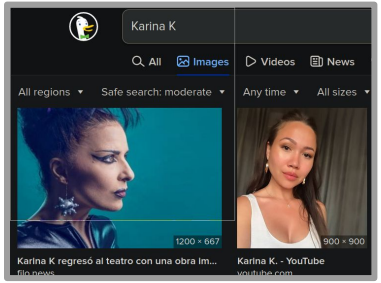
Uncommon Insights for Common Vulnerabilities

•••

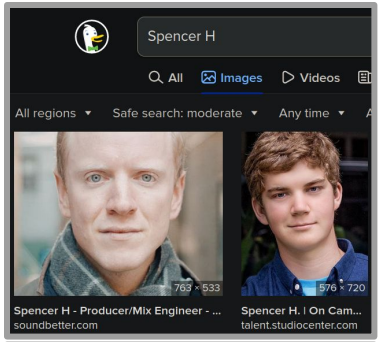
Team #3 - Karina, Spencer, Chris

Introductions

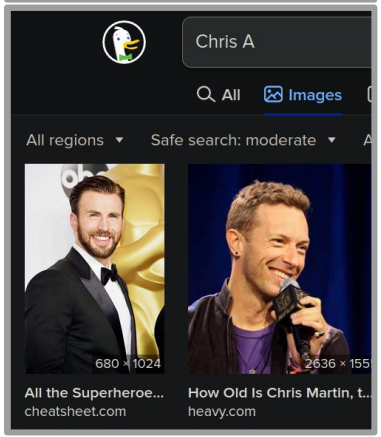
Karina Kanjaria



Spencer Hutton

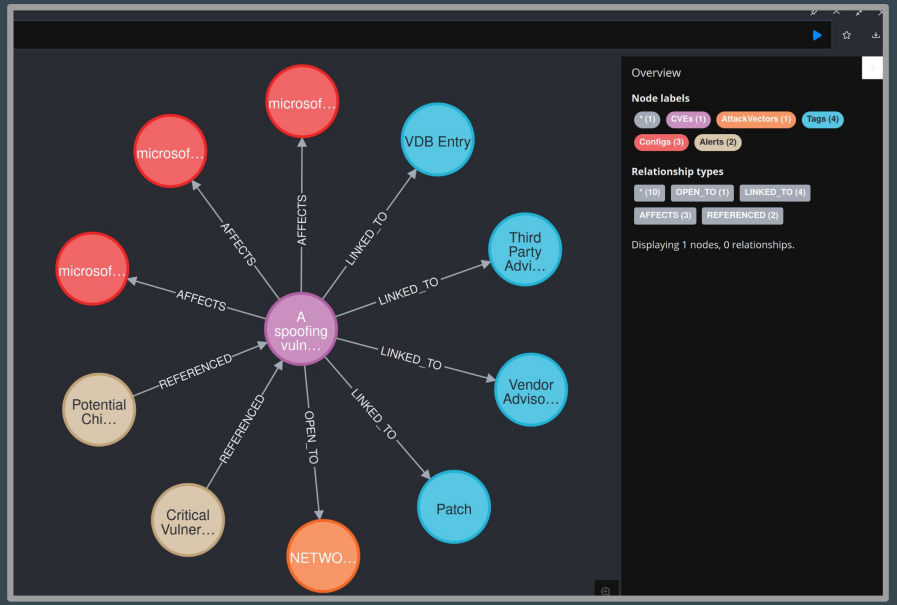


Chris Armstrong



Overview

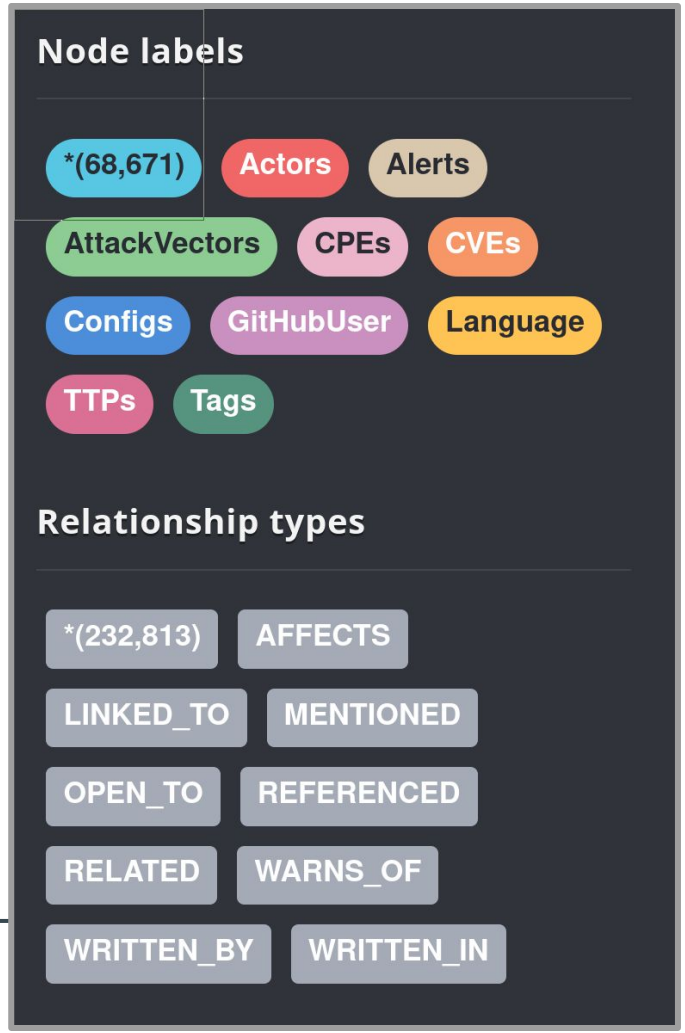
1. Introductions
2. Data Sources
3. Processing the Data
4. Building the Graph
5. Exploring the Graph
6. Demo: Graph + Queries
7. Results and Insights



Data Sources

1. Common Vulnerability Database
2. CISA Alerts
3. Github Repositories

C



Data Sources - NIST CVD

National Institute of Standards and Technology (NIST)

1. Common Vulnerability Database (CVD) lists **Common Vulnerabilities and Exposures (CVEs)**
2. National Vulnerability Database Data Feeds with U.S. government repository of standards based vulnerability management
3. Details of issues that are often leveraged by hackers to exploit vulnerabilities
4. 20 years worth of structured RAW JSON (2002 - 2022)

https://nvd.nist.gov/vuln/data-feeds#JSON_FEED

Data Sources - CISA Alerts

Cybersecurity and Infrastructure Security Agency (CISA)

1. Warnings and notifications of important **security issues affecting the United States**
2. Nearly 300 alerts from 2008 to 2022
3. Unstructured text similar to news articles.
    a. References to CVEs
    b. Related entities
    c. Affected systems
    d. Techniques, Tactics, Procedures

S

Alert (AA22-335A)

#StopRansomware: Cuba Ransomware

Original release date: December 01, 2022 | Last revised: December 05, 2022

Print Tweet Send Share

Summary

Note: This joint Cybersecurity Advisory (CSA) is part of an ongoing #StopRans
network defenders that detail various ransomware variants and ransomware
advisories include recently and historically observed tactics, techniques, and
compromise (IOCs) to help organizations protect against ransomware. Visit s
#StopRansomware advisories and to learn more about other ransomware th

More Alerts

Actions to take today to mitigate cyber threats from ransomware:

- Prioritize remediating known exploited vulnerabilities.
- Train users to recognize and report phishing attempts.
- Enable and enforce phishing-resistant multifactor authentication.

, with ransoms demanded and paid on the

ble link between Cuba ransomware actors,

likelihood and impact of Cuba ransomware

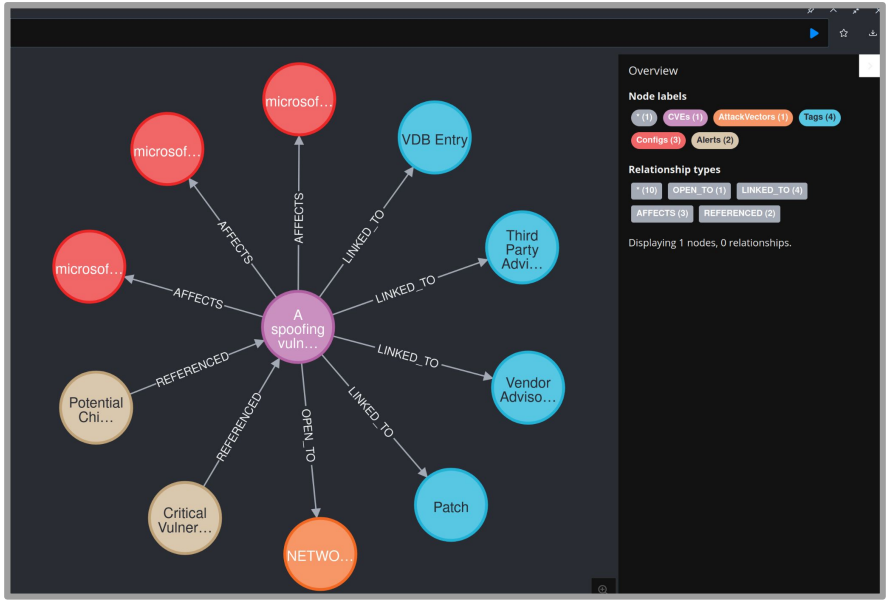
Cuba ransomware actors continuing to target
c Health, Critical Manufacturing, and

Data Sources - GitHub (aka GitHub)

1. GitHub links related to NIST CVEs
2. Underlying repository links for direct GitHub page links
3. Structured data from APIs to link **collaborators and programming languages** used

K

Processing the Data



C

Processing the Data - Common Vulnerabilities and Exposures

Download raw JSON files

Parse them for the core CVE data using JSON Path

- ID
- Severity
- URLs & Tags
- Description
- Configurations

```
nvdcve-1.1-2002.json  nvdcve-1.1-2009.json  nvdcve-1.1-2016.json
nvdcve-1.1-2003.json  nvdcve-1.1-2010.json  nvdcve-1.1-2017.json
nvdcve-1.1-2004.json  nvdcve-1.1-2011.json  nvdcve-1.1-2018.json
nvdcve-1.1-2005.json  nvdcve-1.1-2012.json  nvdcve-1.1-2019.json
nvdcve-1.1-2006.json  nvdcve-1.1-2013.json  nvdcve-1.1-2020.json
nvdcve-1.1-2007.json  nvdcve-1.1-2014.json  nvdcve-1.1-2021.json
nvdcve-1.1-2008.json  nvdcve-1.1-2015.json  nvdcve-1.1-2022.json
❯ du -d1 -h
985M    .
```

Processing the Data - CISA Alerts

1. Web scraping to pull CISA alerts using requests and beautifulsoup
2. Find CVE references using regular expressions
3. Identify entities using SpaCy
4. Separate entities into types and cluster using Dedupe
5. Weights based on entity frequency
6. Link Product-type entities to Configs using string matching

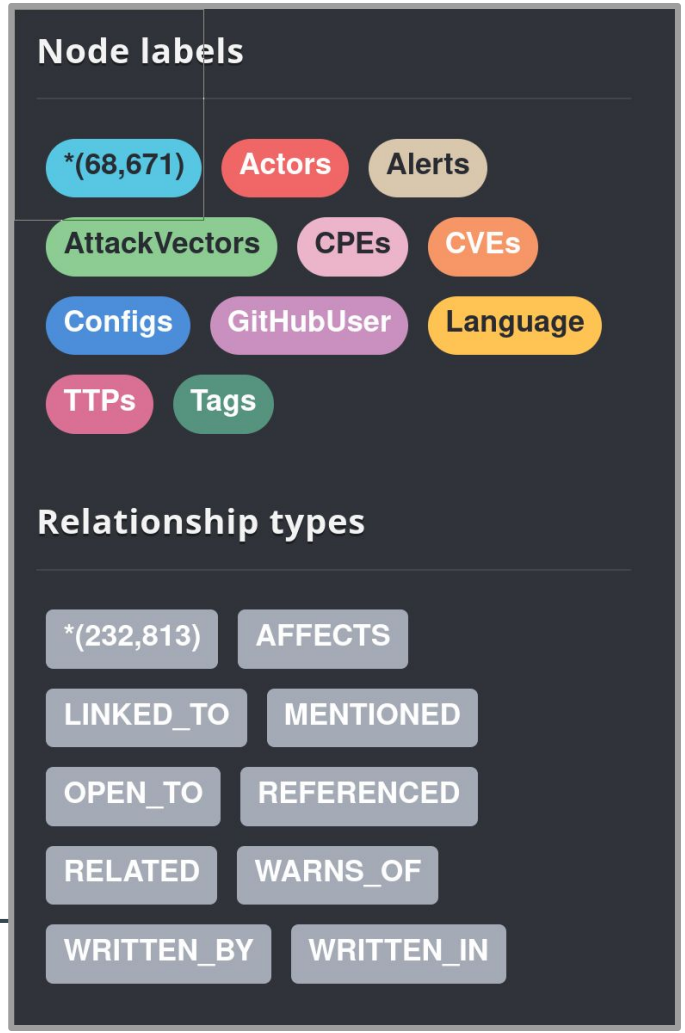
S

Processing the Data - GitHub

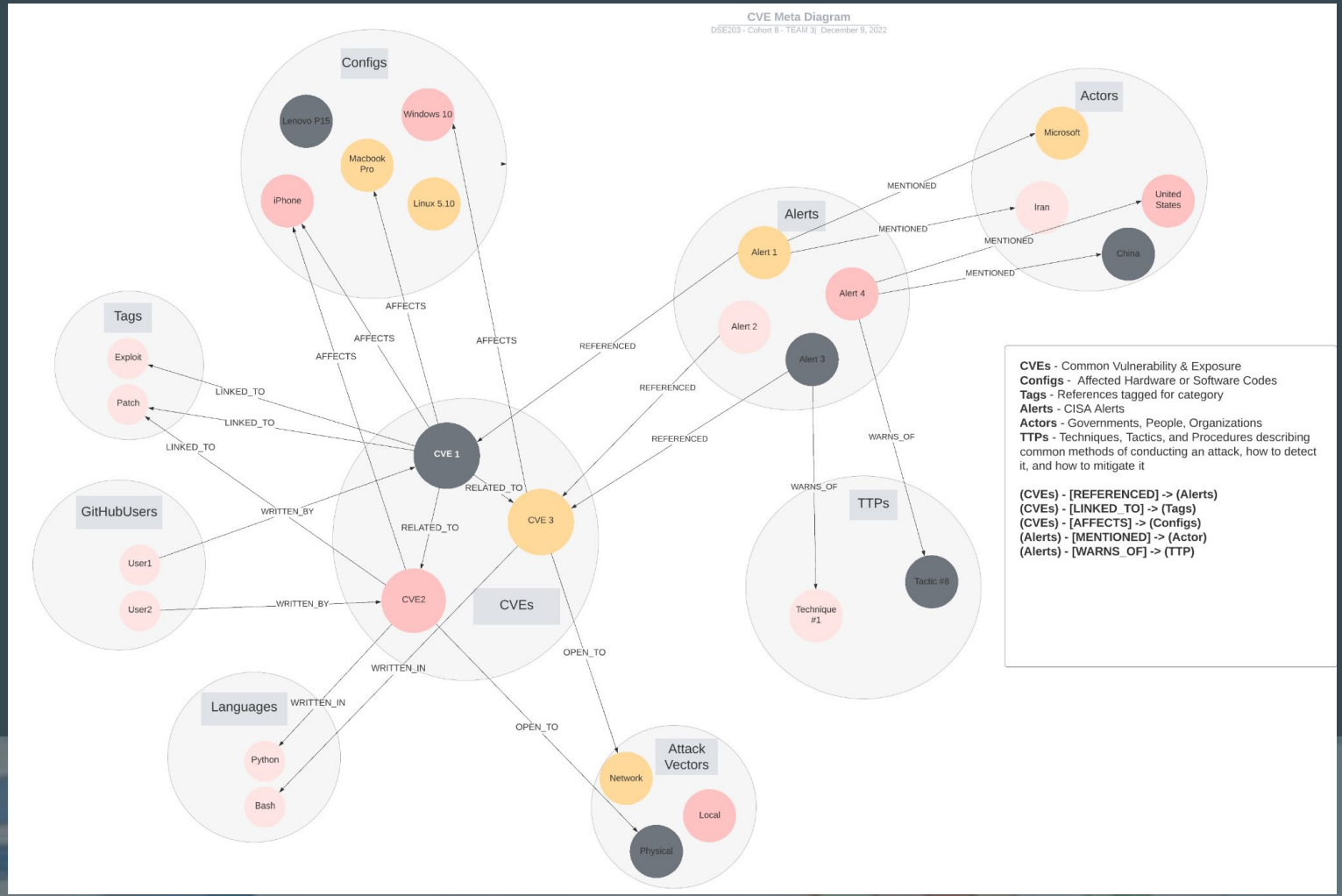
1. Filter for GitHub references in CVE data
2. URL split to find repository urls
3. Leverage GitHub APIs for repositories
    a. Stay within API rate limits...
4. Link vulnerabilities to list of contributors
5. Link vulnerabilities to list of programming languages

K

Building the Graph

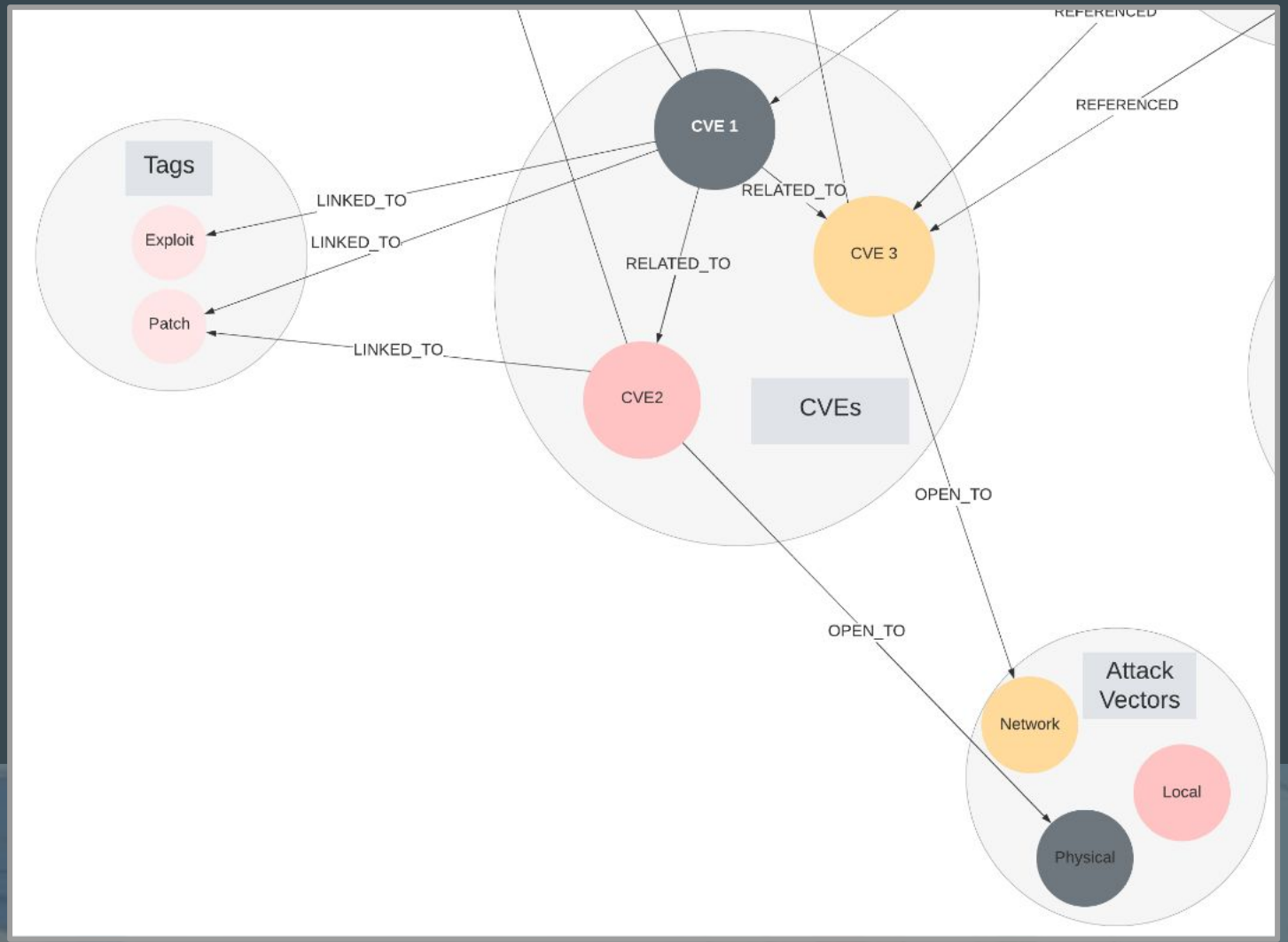


Meta Graph Diagram

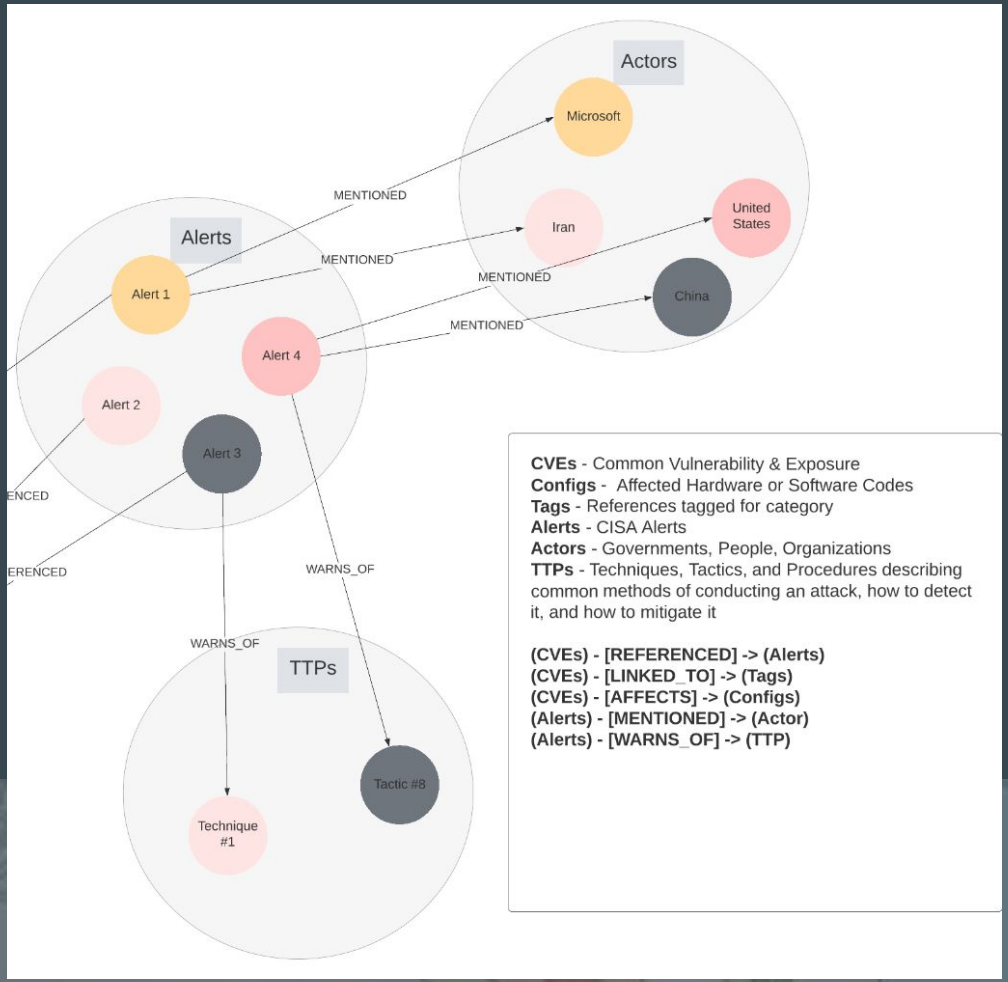


C

CVE Focus

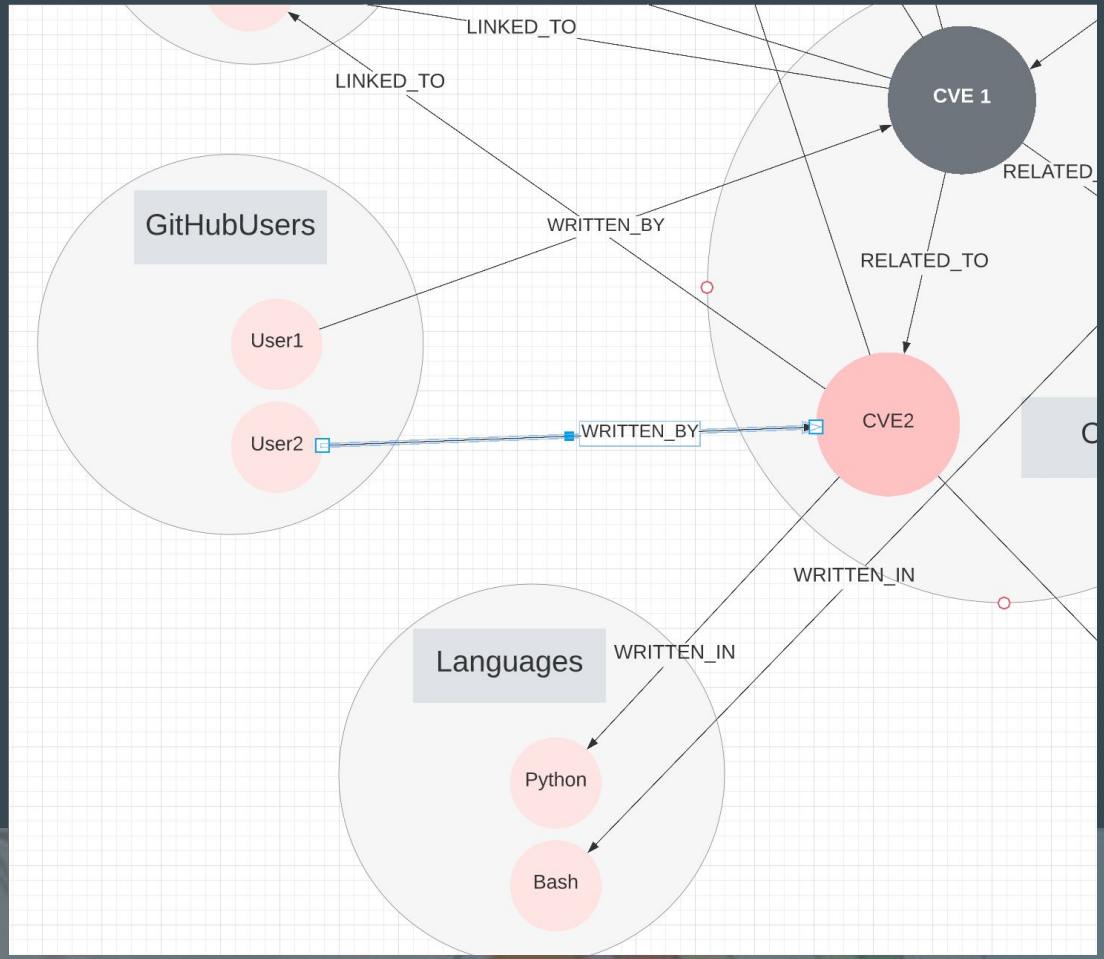

REFERENCED
REFERENCED
CVE 1
Tags
LINKED_TO
Exploit
LINKED_TO
RELATED_TO
CVE 3
RELATED_TO
Patch
LINKED_TO
CVE2
CVEs
OPEN_TO
OPEN_TO
Attack Vectors
Network
Local
Physical

Alerts Focus



S

GitHub Focus



K

Building the Graph - #1 CVE Data

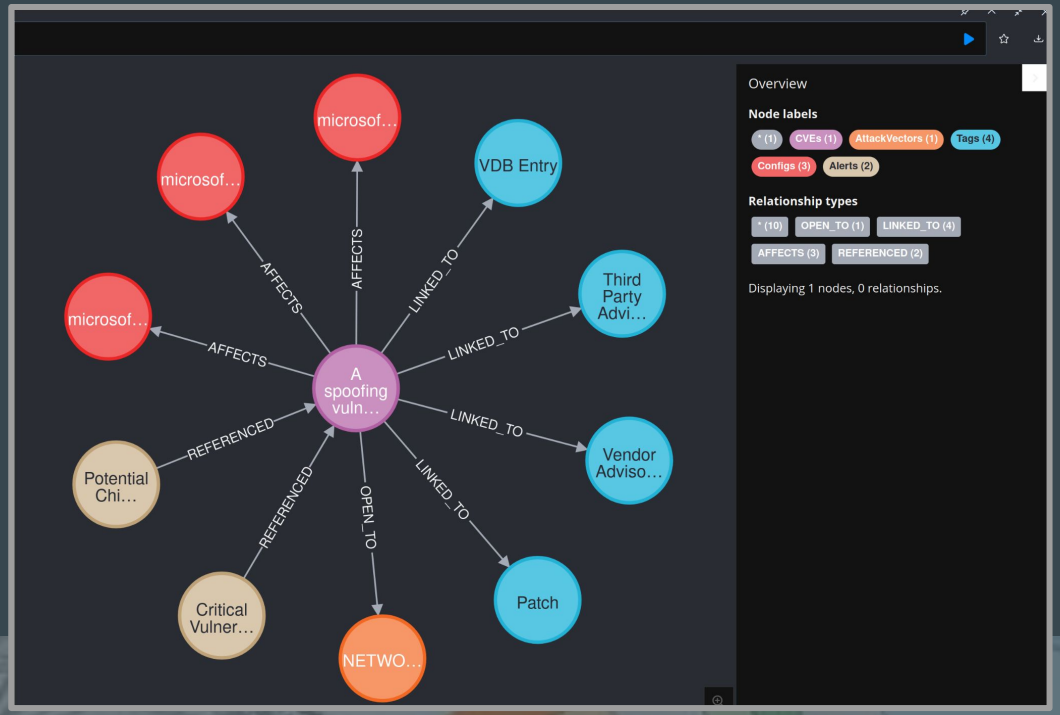
JSON CVE data ->

Nodes:

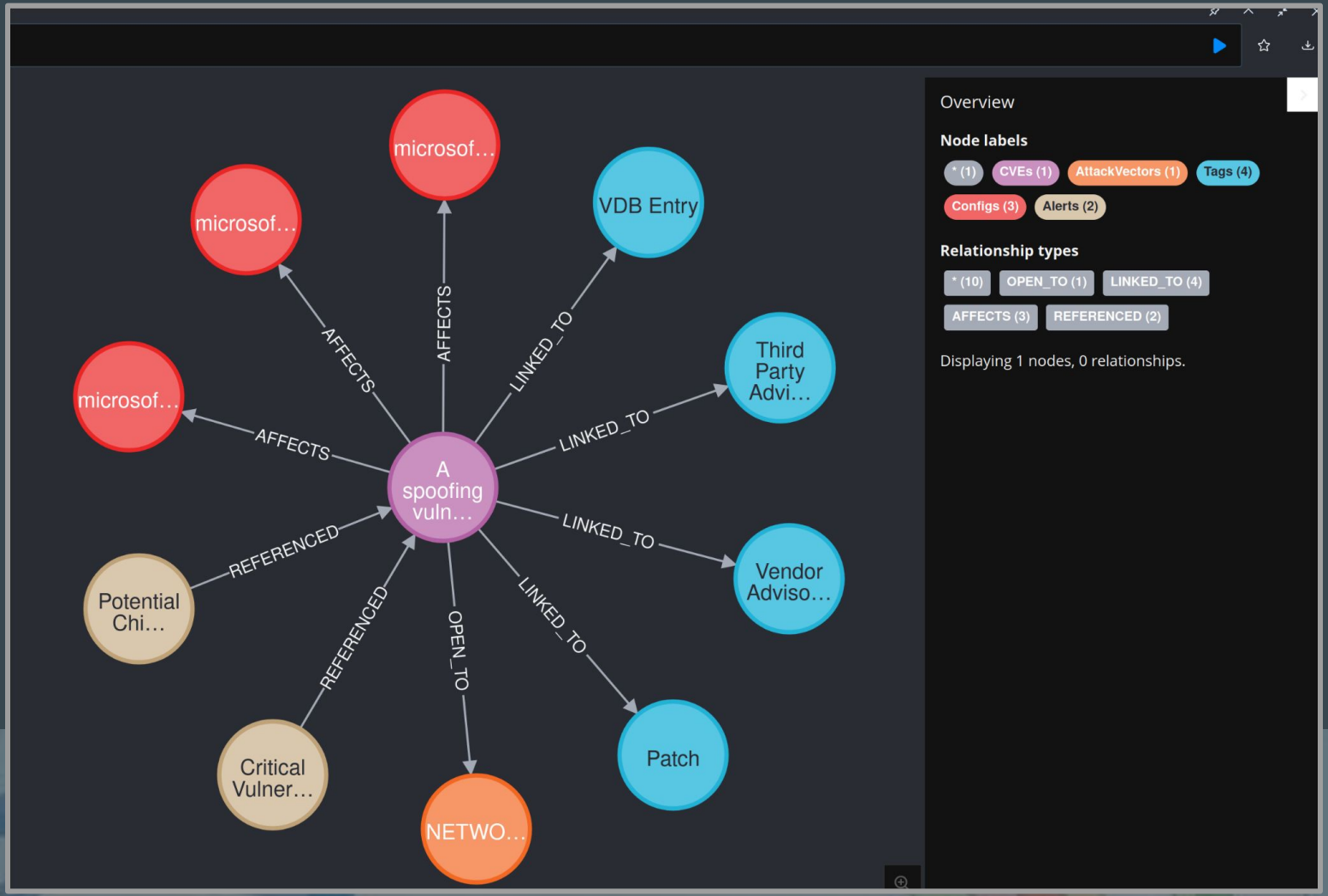
- CVEs
- Tags
- Configs
- Attack Vectors

Relationships:

AFFECTS, LINKED_TO, REFERENCED, OPEN_TO



C


microsof…
microsof…
VDB Entry
microsof…
Third Party Advi…
microsof…
A spoofing vuln…
Potential Chi…
Vendor Adviso…
Critical Vulner…
NETWO…
Patch
AFFECTS
AFFECTS
LINKED_TO
AFFECTS
LINKED_TO
REFERENCED
LINKED_TO
REFERENCED
OPEN_TO
LINKED_TO
Overview
Node labels
* (1)
CVEs (1)
AttackVectors (1)
Tags (4)
Configs (3)
Alerts (2)
Relationship types
* (10)
OPEN_TO (1)
LINKED_TO (4)
AFFECTS (3)
REFERENCED (2)
Displaying 1 nodes, 0 relationships.

Building the Graph - #2 CISA Alerts

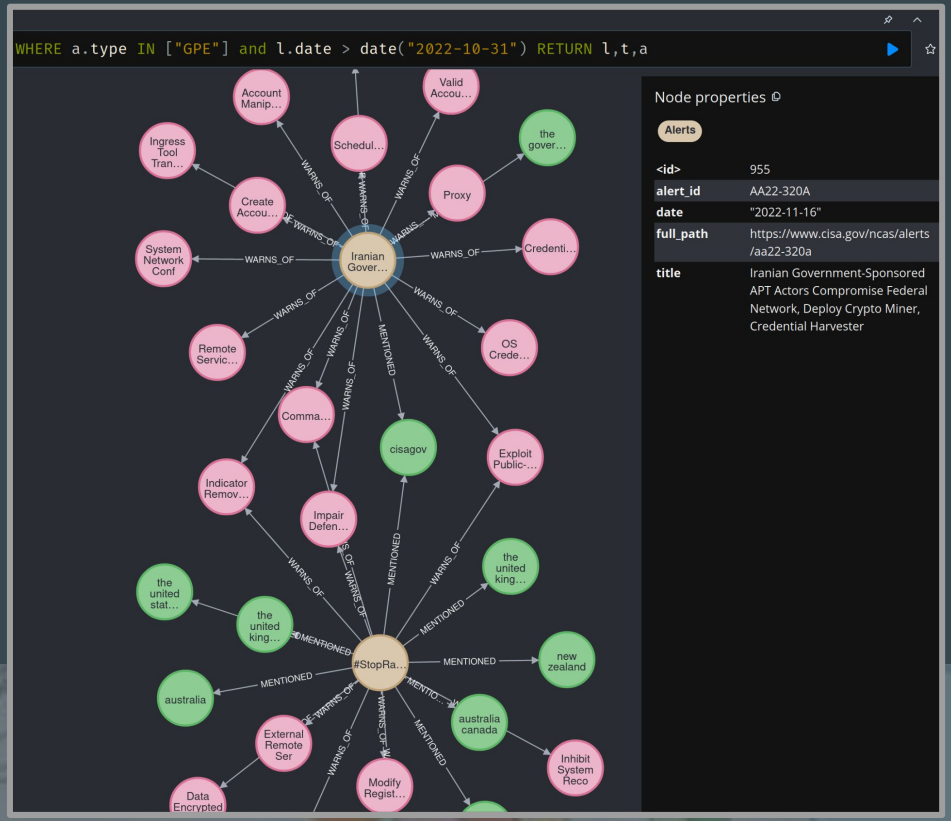
Cybersecurity and Infrastructure Security Agency **Alerts**

Nodes:

- Alerts
- Techniques, Tactics, Procedures
- Actors

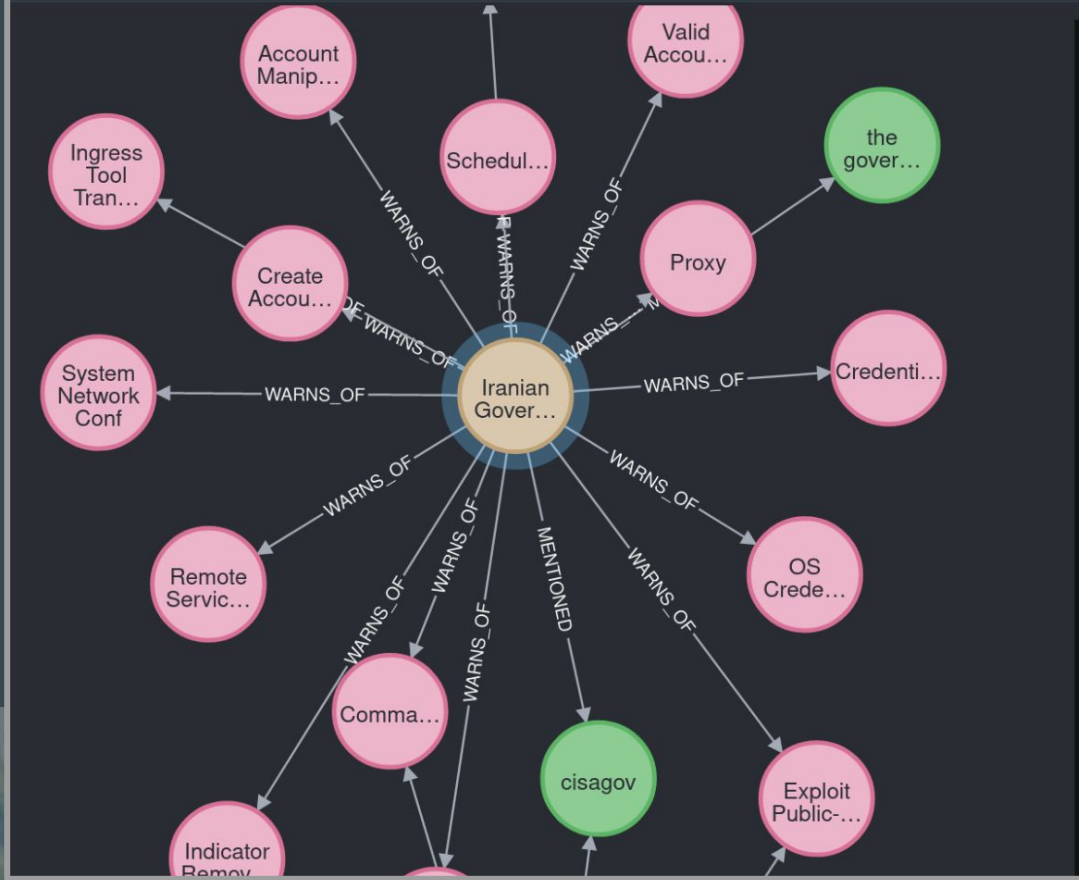
Relationships:

WARNS_OF, MENTIONED



S

```
e IN ["GPE"] and l.date > date("2022-10-31") RETURN l,t,a
```



Node properties

Alerts

<id>	955
alert_id	AA22-320A
date	"2022-11-16"
full_path	https://www.cisa.gov/ncas/alerts/aa22-320a
title	Iranian Government-Sponsored APT Actors Compromise Federal Network, Deploy Crypto Miner, Credential Harvester

Building the Graph - #3 GitHub Data

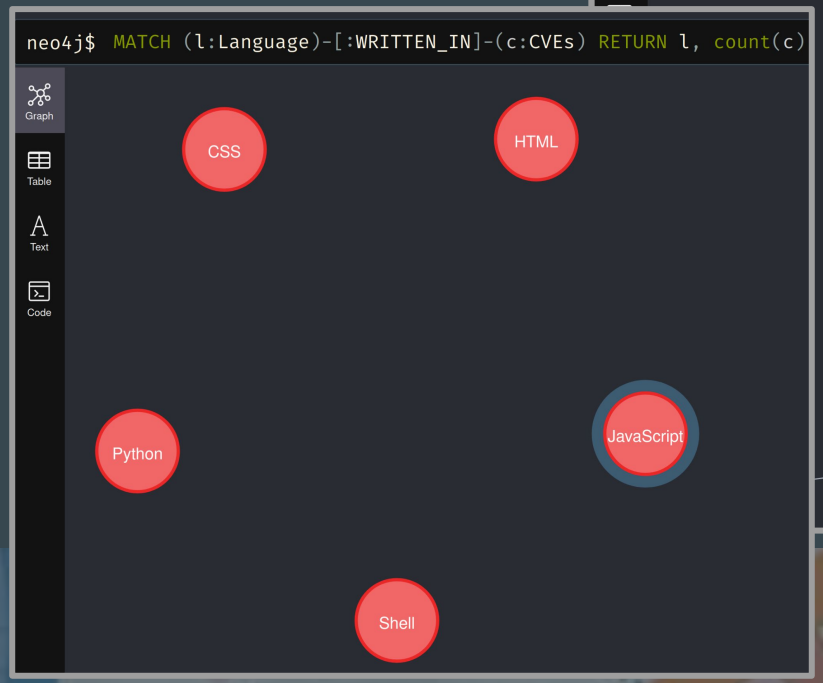
Nodes:

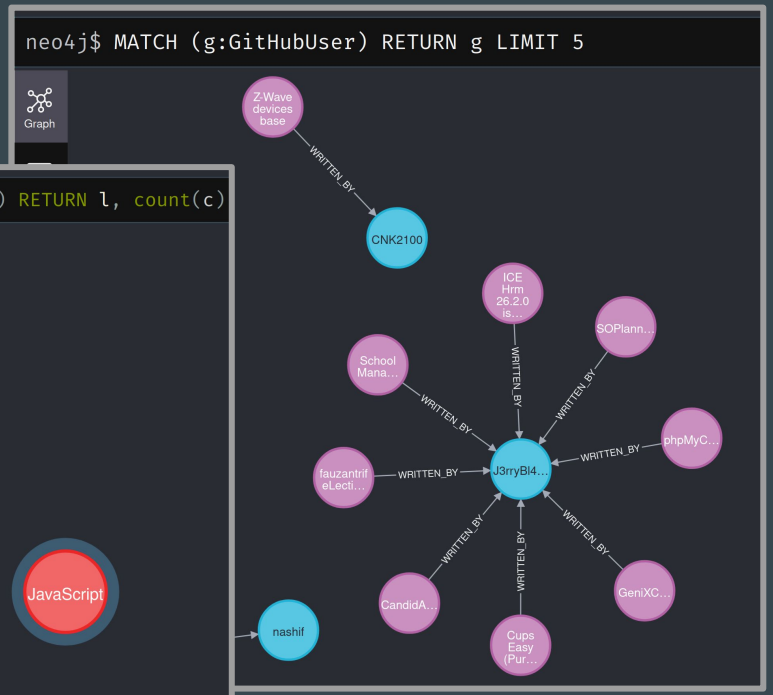
- User ID
- Language

Relationships:

WRITTEN_IN /

WRITTEN_BY

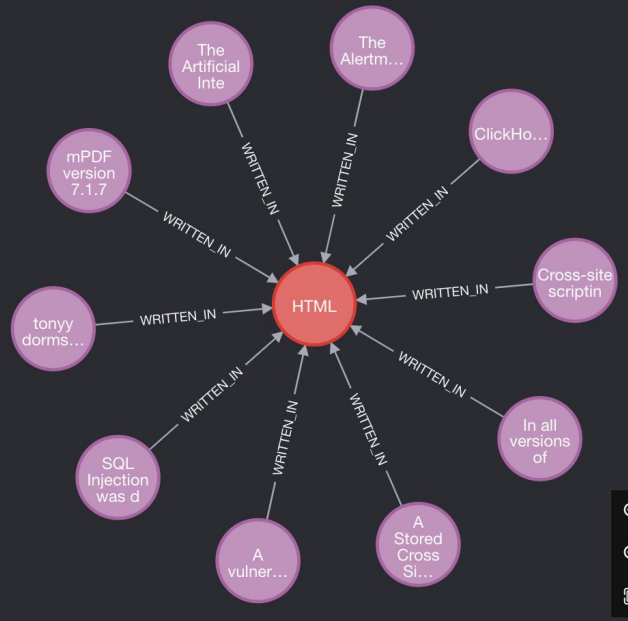




K

```
neo4j$ MATCH (g:GitHubUser) RETURN g LIMIT 5
```

```
MATCH (c:CVEs)-[:WRITTEN_IN]→(lan:Language) RETURN lan, c LIMIT 10
```



Overview

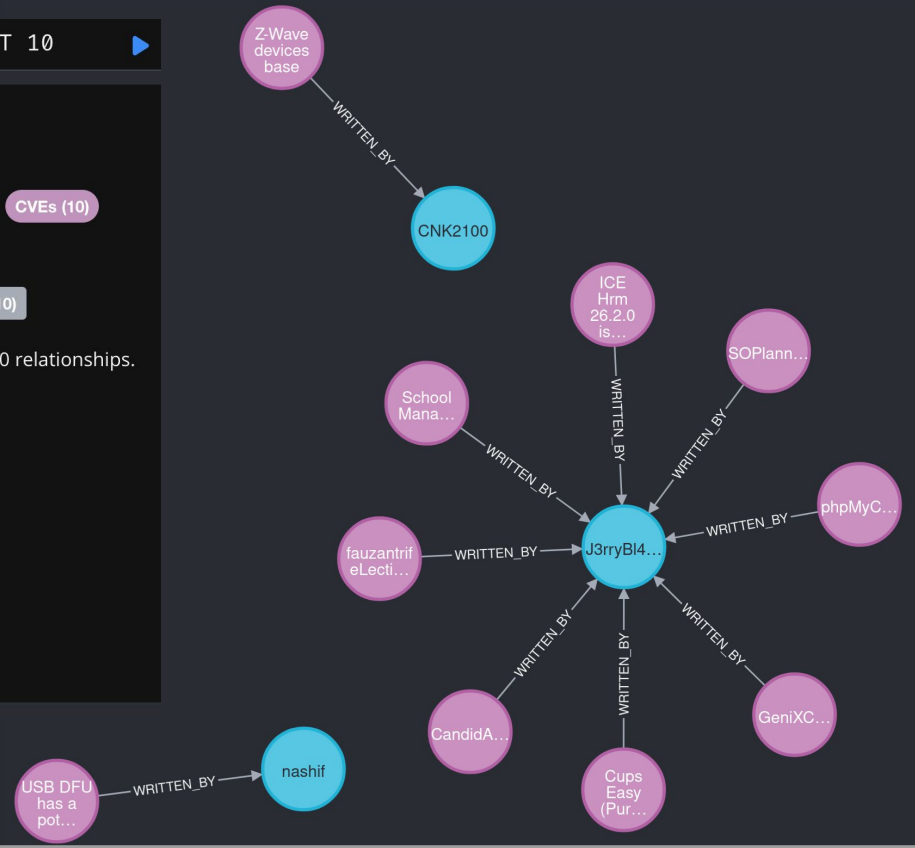
Node labels

* (11) | Language (1) | CVEs (10)

Relationship types

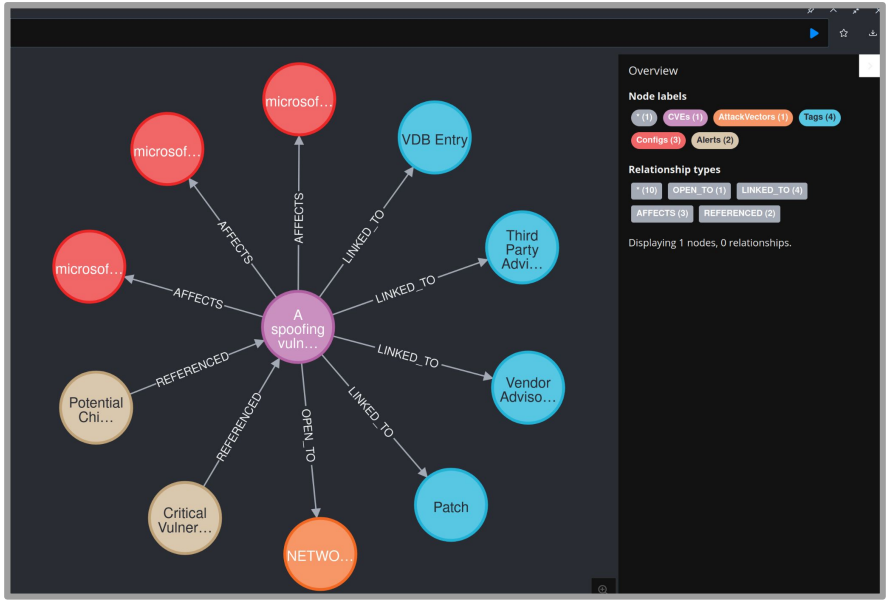
* (10) | WRITTEN_IN (10)

Displaying 11 nodes, 10 relationships.



Exploring the Graph

(WHAT IS IN THIS WEB)



C

Processes for Exploring the Graph

1. Tags for vulnerabilities
2. Number of vulnerabilities per alert
3. Time between vulnerability publish date and an alert being issued
4. How are entities mentioned in alerts related to vulnerabilities?

How Vulnerabilities Are Tagged

```
neo4j$ MATCH (c:CVEs)-[]-(t:Tags) WITH COUNT(c) AS CVEs,t RETURN CVEs,t.tag O…
```

Table | Text | Code

	CVEs	t.tag
1	26661	"Third Party Advisory"
2	17000	"Exploit"
3	12310	"Patch"
4	6861	"Issue Tracking"
5	5440	"Vendor Advisory"

K

How Vulnerabilities Are Tagged

```
neo4j$ MATCH (c:CVEs)-[]-(t:Tags) WITH COUNT(c)
```

Table | Text | Code

	CVEs	t.tag
1	26661	"Third Party Advisory"
2	17000	"Exploit"

Vulnerabilities vs Alerts

```
MATCH (c:CVEs)-[]-(a:Alerts) WITH COUNT(c) as CVEs,a RETURN avg(CVEs)
```

avg(CVEs)
4.546511627906976

C

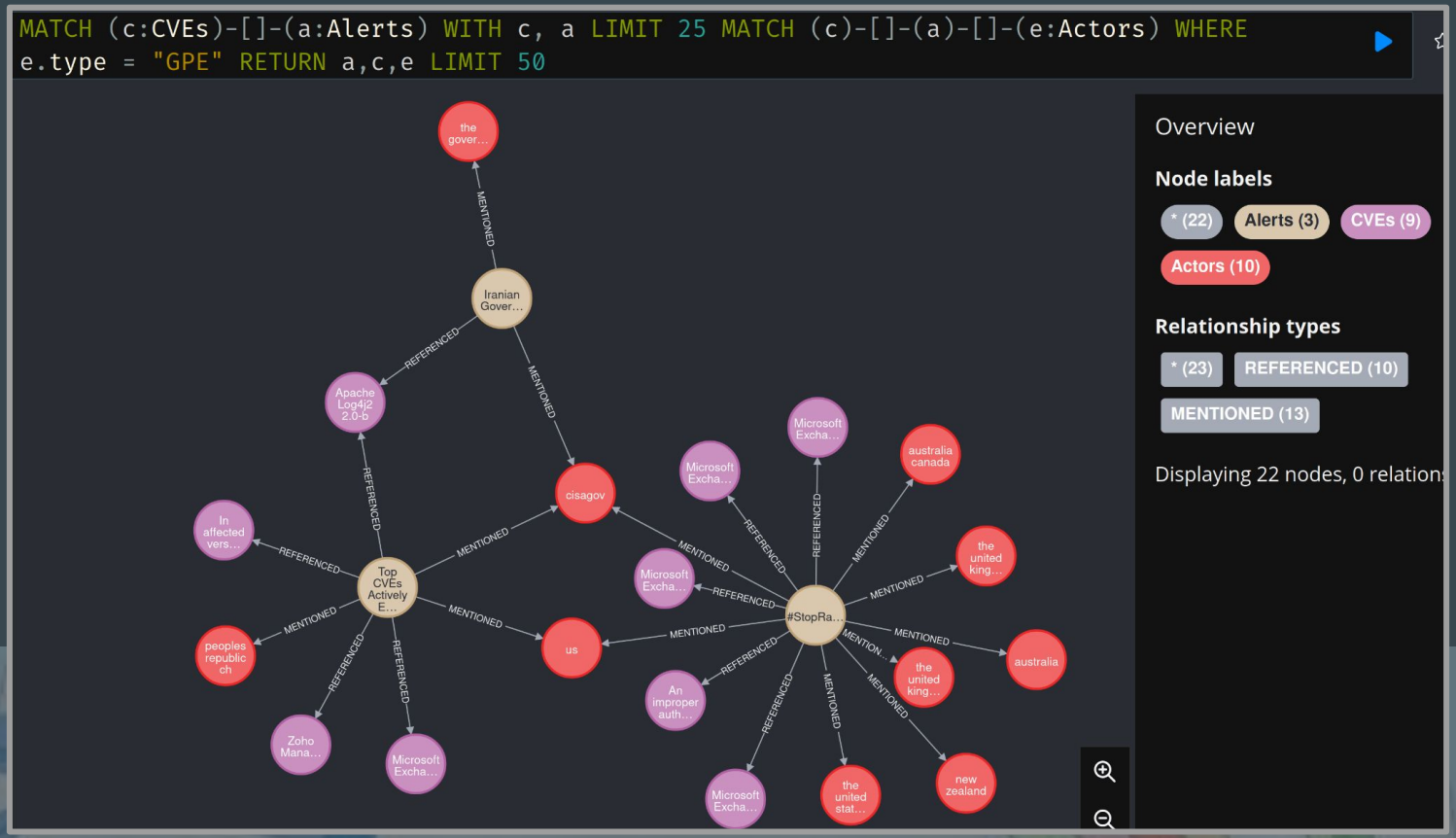
Time between CVE -> Alert

```
1  // Average time difference between when a CVE is published
2  // and when the Alert is issued
3  MATCH (c:CVEs)-[:REFERENCED]-(a:Alerts)
4  RETURN avg(duration.between(a.date, c.published)) AS Incubation
```

Table

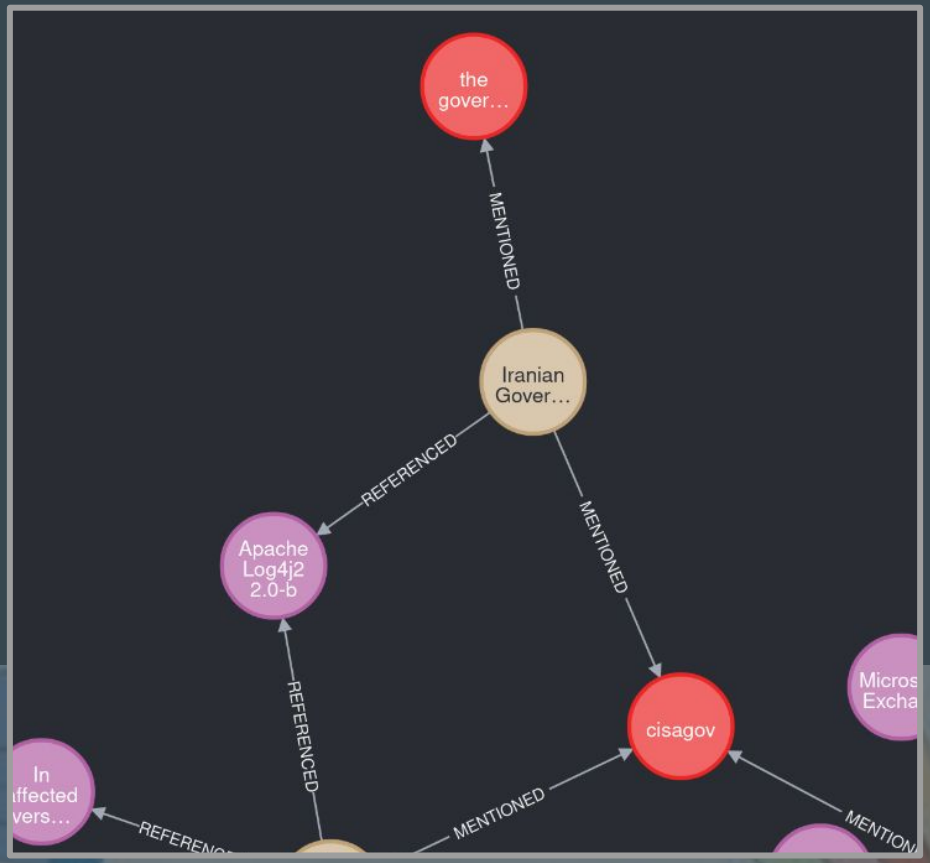
	Incubation
1	P-1Y-4M-6DT-22H-22M-30.03S

Actors and Related Vulnerabilities



S

Focus on - Alerts, Actors, and Related Vulnerabilities



Demo

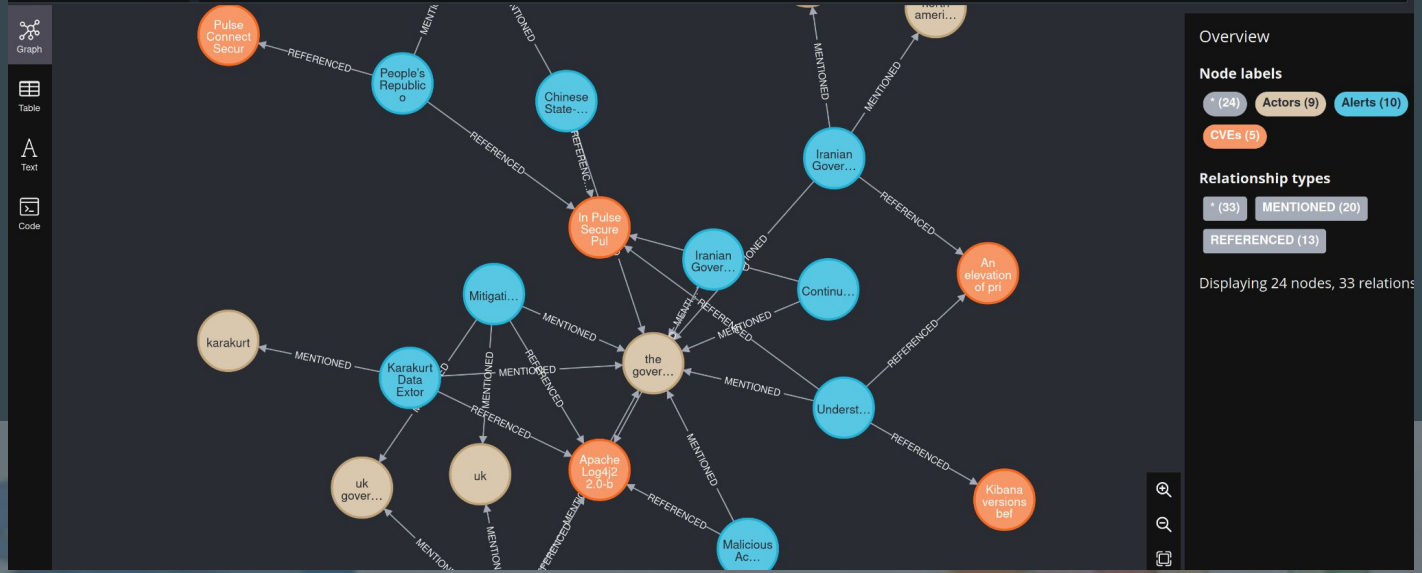
C

The Queries

- What geopolitical entities are related to the most severe vulnerabilities?
- What attack vectors do Russians use the most?
- Contributor centrality
- Community detection
- Which languages are the most popular amongst actors?

Query- Foreign Entities Related to the Most Severe Vulnerabilities

```
MATCH (ac:Actors)←[m:MENTIONED]-(at:Alerts)-[r:REFERENCED]→(c:CVEs)-[:OPEN_TO]-(v:AttackVectors)
WHERE ac.type = "GPE"
    AND NOT ac.best_label CONTAINS "united" // Removing US
    AND NOT ac.best_label CONTAINS "cisa"// Removing US
    AND NOT ac.best_label CONTAINS "us" // Removing US
    AND NOT ac.best_label CONTAINS "cyber" // Removing US
    AND NOT ac.best_label CONTAINS "lan" // Removing US
    AND NOT at.title CONTAINS "Top" // Removing Aggregated Alerts
    AND c.score ⩾ 10 // PERFECT SCORE
RETURN ac,at,c
```



```
MATCH (ac:Actors)<-[m:MENTIONED]-(at:Alerts)-[r:REFERENCED]->(c:CVEs)-[:OPEN_TO]-(v:AttackVectors)
WHERE ac.type = "GPE"
    AND NOT ac.best_label CONTAINS "united" // Removing US
    AND NOT ac.best_label CONTAINS "cisa"// Removing US
    AND NOT ac.best_label CONTAINS "us" // Removing US
    AND NOT ac.best_label CONTAINS "cyber" // Removing US
    AND NOT ac.best_label CONTAINS "lan" // Removing US
    AND NOT ac.best_label CONTAINS "north america" // Removing US
    AND NOT at.title CONTAINS "Top" // Removing Aggregated Alerts
    AND c.score >= 10 // PERFECT SCORE
RETURN ac.best_label, count(m) AS popularity ORDER BY popularity desc LIMIT 5
```

Table

Text

Code

	ac.best_label	popularity
1	"the government albania"	11
2	"peoples republic china"	3
3	"uk government"	2
4	"uk"	2
5	"canada"	1

Query- Russian Attack Vectors (NO Entity Recognition)

Shows only 73 NETWORK type attack vector related CVEs

```
match (a:Alerts) -[:REFERENCED]→(c:CVEs)-[:AFFECTS]→(n:Configs), (c)-[
targets,v.attack_vector
```

targets	v.attack_vector
73	"NETWORK"
3	"LOCAL"

Query - Improved - Russian Attack Vectors (With NER)

```
MATCH (e:Actors)←[:MENTIONED]-(a:Alerts) -[:REFERENCI
(c)-[:OPEN_TO]-(v:AttackVectors) WHERE e.best_label C(
targets,v.attack_vector
```

targets	v.attack_vector
57	"LOCAL"
936	"NETWORK"
10	"ADJACENT_NETWORK"

Query- Popular Languages

```
1 MATCH (a:Actors)←[:MENTIONED]-(l:Alerts)-[r:REFERENCED]→(c:CVEs)-[:WRITTEN_IN]→(lan:Language)
2 WHERE NOT lan.language CONTAINS "message"
3 RETURN lan.language, COUNT(a) as nums
4 ORDER BY nums DESC
```

Table | Text | Code

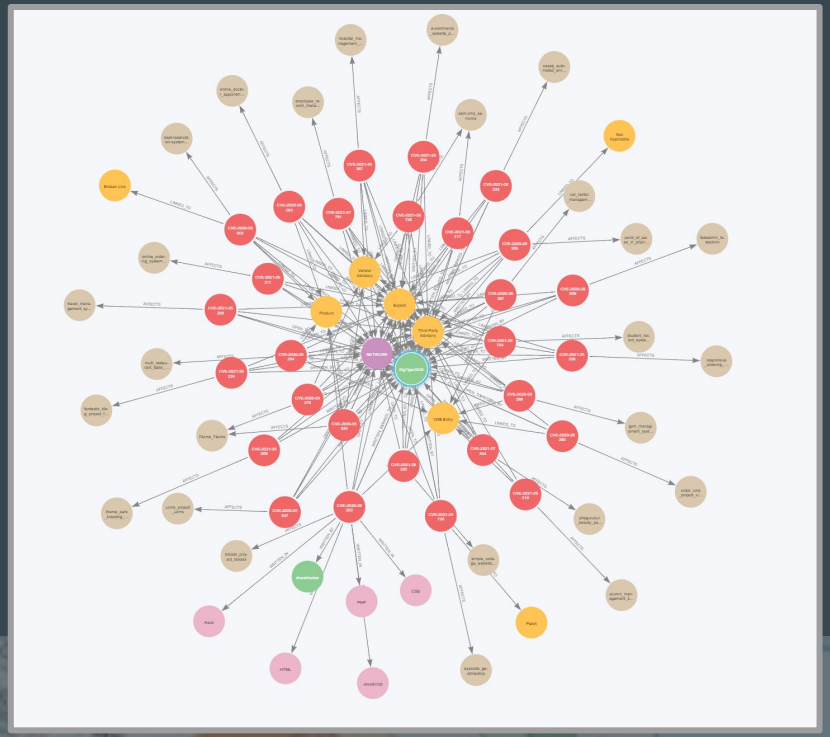
	lan.language	nums
1	"Python"	1744
2	"C"	1147
3	"Shell"	944
4	"Ruby"	597
5	"Batchfile"	550
6	"Smalltalk"	550

K

Query- Contributor Centrality

Use GDS centrality pagerank to rank GitHub contributors

Top user subgraph shows contributions on a large number of vulnerabilities related to a range of other nodes



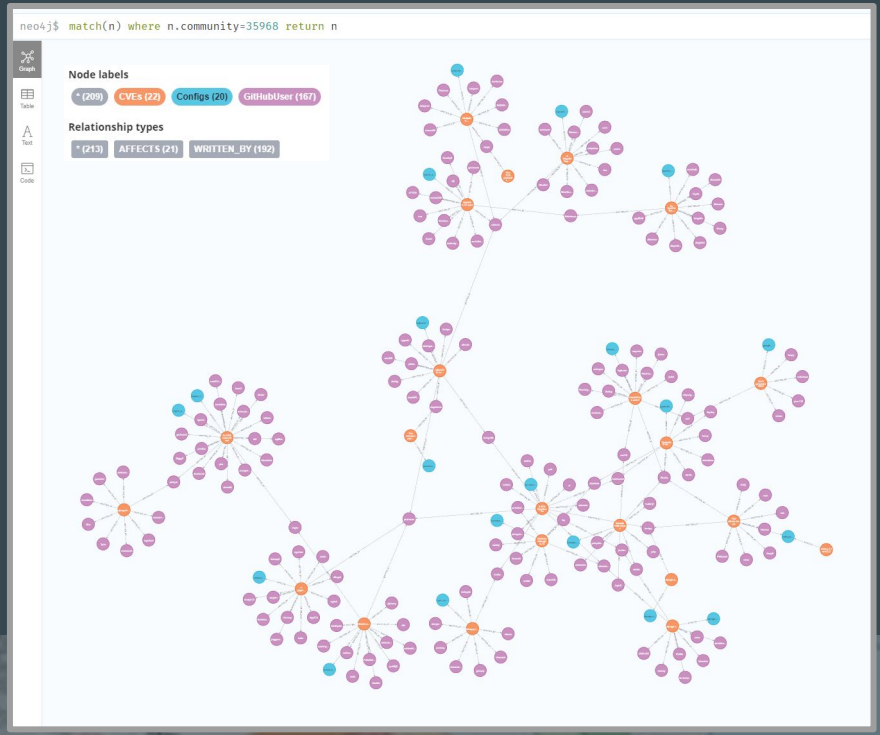
S

Query- Community Detection

Use GDS Louvain algorithm to detect communities within the graph

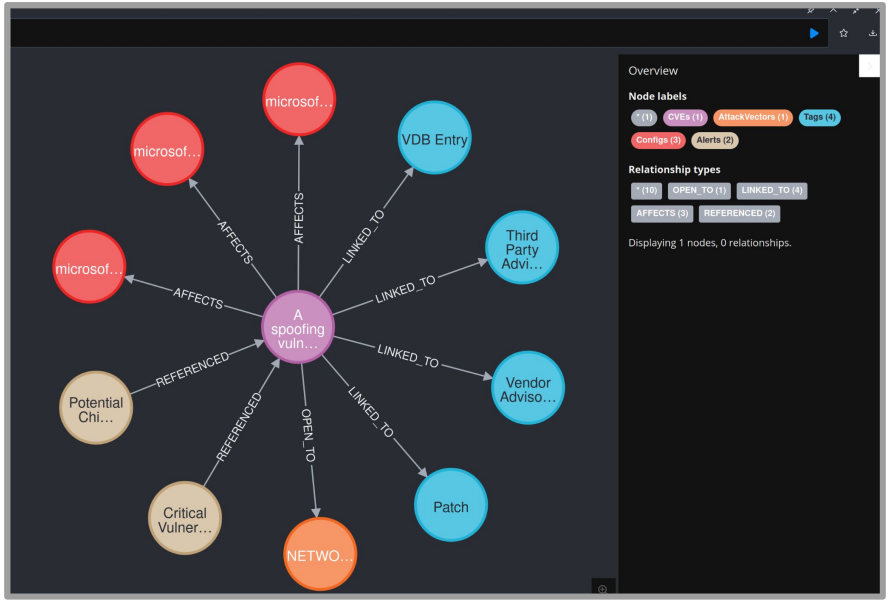
Write community property back to nodes

Community subgraph shows network of vulnerabilities with a large number of contributors and shared contributors



Results and Insights

Discoveries

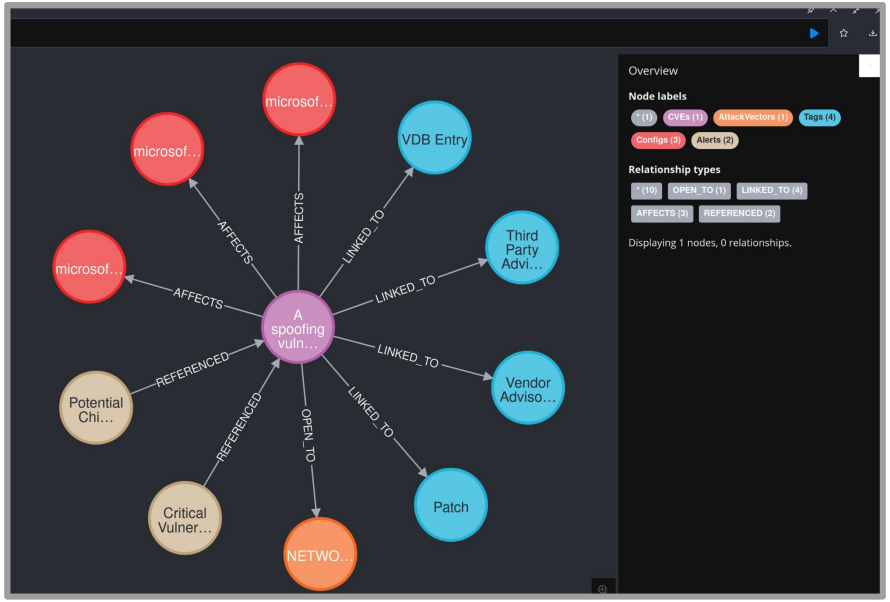


C

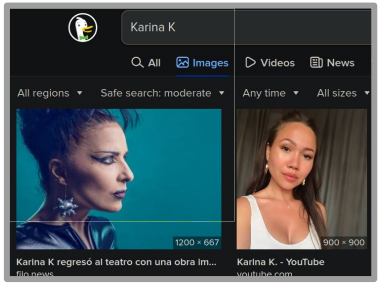
Results and Insights

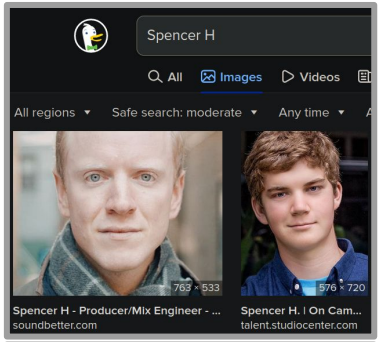
- 1+ yrs between vulnerability publicly known and alert issued
- Albania was the most mentioned entity related to severe vulnerabilities
- Programming languages connected to most vulnerabilities had slightly different trends than those connected to vulnerabilities with alerts and actors
    - Scripting languages are more popular overall while actors prefer procedural languages

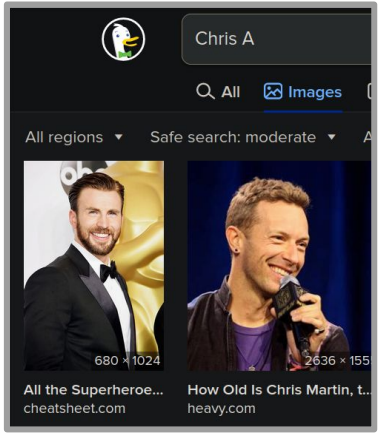
Questions?


Overview
Node labels
* (1)
CVEs (1)
AttackVectors (1)
Tags (4)
Configs (3)
Alerts (2)
Relationship types
* (10)
OPEN_TO (1)
LINKED_TO (4)
AFFECTS (3)
REFERENCED (2)
Displaying 1 nodes, 0 relationships.
microsof…
microsof…
microsof…
VDB Entry
Third Party Advi…
A spoofing vuln…
Vendor Adviso…
Patch
NETWO…
Critical Vulner…
Potential Chi…
AFFECTS
AFFECTS
AFFECTS
LINKED_TO
LINKED_TO
LINKED_TO
LINKED_TO
OPEN_TO
REFERENCED
REFERENCED

Thank you!


Karina K
All
Images
Videos
News
All regions
Safe search: moderate
Any time
All sizes
1200 × 667
900 × 900
Karina K regresó al teatro con una obra im...
filo.news
Karina K. - YouTube
youtube.com


Spencer H
All
Images
Videos
All regions
Safe search: moderate
Any time
763 × 533
576 × 720
Spencer H - Producer/Mix Engineer - ...
soundbetter.com
Spencer H. | On Cam...
talent.studiocenter.com


Chris A
All
Images
All regions
Safe search: moderate
680 × 1024
2636 × 155
All the Superheroe...
cheatsheet.com
How Old Is Chris Martin, t...
heavy.com