



Goran Cvijanović
Senior Software Architect
Reversing Labs

Detecting GitHub Repository Vulnerabilities with Graph Databases

Inspiration

Due to the meteoric rise of supply chain attacks, especially through the NPM and PyPI package repositories, it seems that more scrutiny and mitigations should be added

Why we wouldn't be able to visualize our software structure as graph of dependencies and analyze which components are exposed to CVE security issues

Dependencies to other repositories, libraries and projects increase complexity of the problem, but can be easily represented and visualized using graph technology

Public Security Information

Common Vulnerability Exposure



The mission of the CVE® Program is to identify, define, and catalog publicly disclosed cybersecurity vulnerabilities
Currently, there are 173,267+ CVE Records accessible

National Vulnerability Database



The NVD performs analysis on CVEs that have been published

Common Platform Enumeration – CPE

Common Vulnerability Scoring System - CVSS



[CVE List▼](#)[CNAs▼](#)[WGs▼](#)[Board▼](#)[About▼](#)[News & Blog▼](#)[Search CVE List](#)[Downloads](#)[Data Feeds](#)[Update a CVE Record](#)[Request CVE IDs](#)TOTAL CVE Records: **173267****NOTICE: Transition to the all-new CVE website at WWW.CVE.ORG is underway and will last up to one year. ([details](#))****NOTICE: Changes coming to [CVE Record Format JSON](#) and [CVE List Content Downloads](#) in 2022.**[HOME](#) > [CVE](#) > CVE-2021-45046

CVE-ID

CVE-2021-45046[Learn more at National Vulnerability Database \(NVD\)](#)

• CVSS Severity Rating • Fix Information • Vulnerable Software Versions • SCAP Mappings • CPE Information

Description

It was found that the fix to address CVE-2021-44228 in Apache Log4j 2.15.0 was incomplete in certain non-default configurations. This could allow attackers with control over Thread Context Map (MDC) input data, either a Context Lookup (for example, `$(ctx:loginId)`) or a Thread Context Map pattern (`%X`, `%mdc`, or `%MDC`) to craft malicious input data using a JNDI Lookup pattern resulting in an information leak and remote code execution in certain environments. Log4j 2.16.0 (Java 8) and 2.12.2 (Java 7) fix this issue by removing support for message lookup patterns and disabling JNDI functionality by default.

References

Note: [References](#) are provided for the convenience of the reader to help distinguish between vulnerabilities. The list is not intended to be complete.

- CERT-VN:VU#930724
- URL:<https://www.kb.cert.org/vuls/id/930724>
- CISCO:20211210 Vulnerabilities in Apache Log4j Library Affecting Cisco Products: December 2021
- URL:<https://tools.cisco.com/security/center/content/CiscoSecurityAdvisory/cisco-sa-apache-log4j-qRuKNEbd>
- CONFIRM:<https://cert-portal.siemens.com/productcert/pdf/ssa-397453.pdf>
- URL:<https://cert-portal.siemens.com/productcert/pdf/ssa-397453.pdf>
- CONFIRM:<https://cert-portal.siemens.com/productcert/pdf/ssa-479842.pdf>
- URL:<https://cert-portal.siemens.com/productcert/pdf/ssa-479842.pdf>
- CONFIRM:<https://cert-portal.siemens.com/productcert/pdf/ssa-661247.pdf>
- URL:<https://cert-portal.siemens.com/productcert/pdf/ssa-661247.pdf>
- CONFIRM:<https://cert-portal.siemens.com/productcert/pdf/ssa-714170.pdf>
- URL:<https://cert-portal.siemens.com/productcert/pdf/ssa-714170.pdf>
- CONFIRM:<https://logging.apache.org/log4j/2.x/security.html>

🔒 CVE-2021-45046 Detail

Current Description

It was found that the fix to address CVE-2021-44228 in Apache Log4j 2.15.0 was incomplete in certain non-default configurations. This could allow attackers with control over Thread Context Map (MDC) input data when the logging configuration uses a non-default Pattern Layout with either a Context Lookup (for example, `$$${ctx:loginId}`) or a Thread Context Map pattern (`%X`, `%mdc`, or `%MDC`) to craft malicious input data using a JNDI Lookup pattern resulting in an information leak and remote code execution in some environments and local code execution in all environments. Log4j 2.16.0 (Java 8) and 2.12.2 (Java 7) fix this issue by removing support for message lookup patterns and disabling JNDI functionality by default.

[+View Analysis Description](#)

Severity

CVSS Version 3.x

CVSS Version 2.0

CVSS 3.x Severity and Metrics:



NIST: NVD

Base Score: **9.0 CRITICAL**

Vector: CVSS:3.1/AV:N/AC:H/PR:N/UI:N/S:C/C:H/I:H/A:H

NVD Analysts use publicly available information to associate vector strings and CVSS scores. We also display any CVSS information provided within the CVE List from the CNA.

QUICK INFO

CVE Dictionary Entry:

[CVE-2021-45046](#)

NVD Published Date:

12/14/2021

NVD Last Modified:

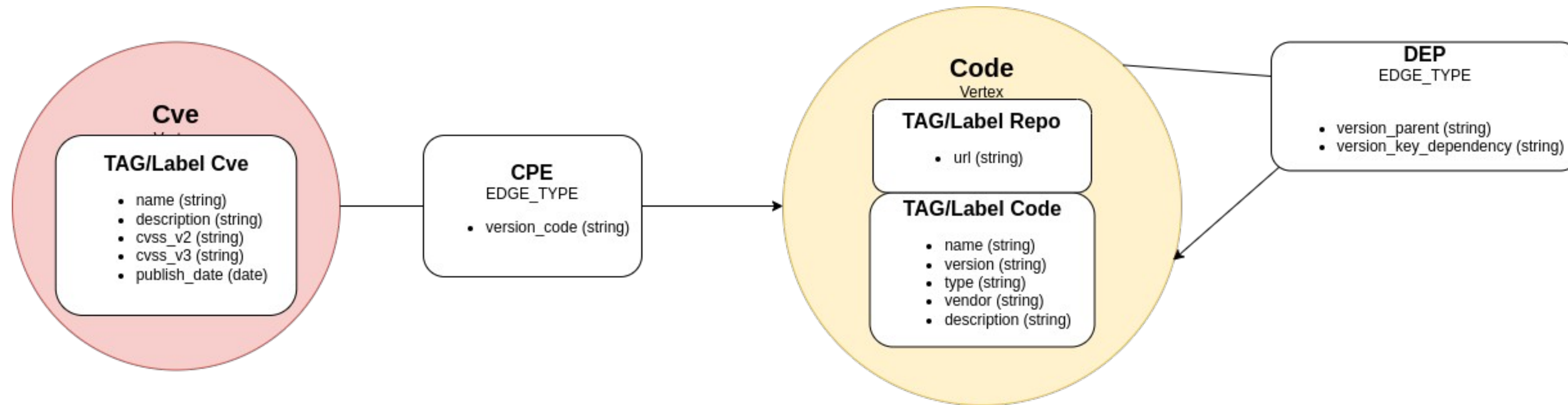
02/18/2022

Source:

Apache Software Foundation

Using Graph Models

- **Easy**
to model data
- **Easy**
to maintain and synchronize data
- **Powerful & Fast**
to handle large dataset



Implementation

- **Python & Javascript code**

Custom python code for web application service
Javascript visualization library

- **Application server**

Nginx web server

- **Graph Database**

Memgraph

<https://secure.codegraph.space>

<https://github.com/mirac7/codegraph>

install MemGraph database with docker

```
docker pull memgraph/memgraph-platform
```

```
docker run -it -d --name memgraph -p 7687:7687 -p 3000:3000 \
-e MEMGRAPH="--bolt-port=7687" \
-v mg_lib:/var/lib/memgraph \
-v mg_log:/var/log/memgraph \
-v mg_etc:/etc/memgraph \
memgraph/memgraph-platform
```

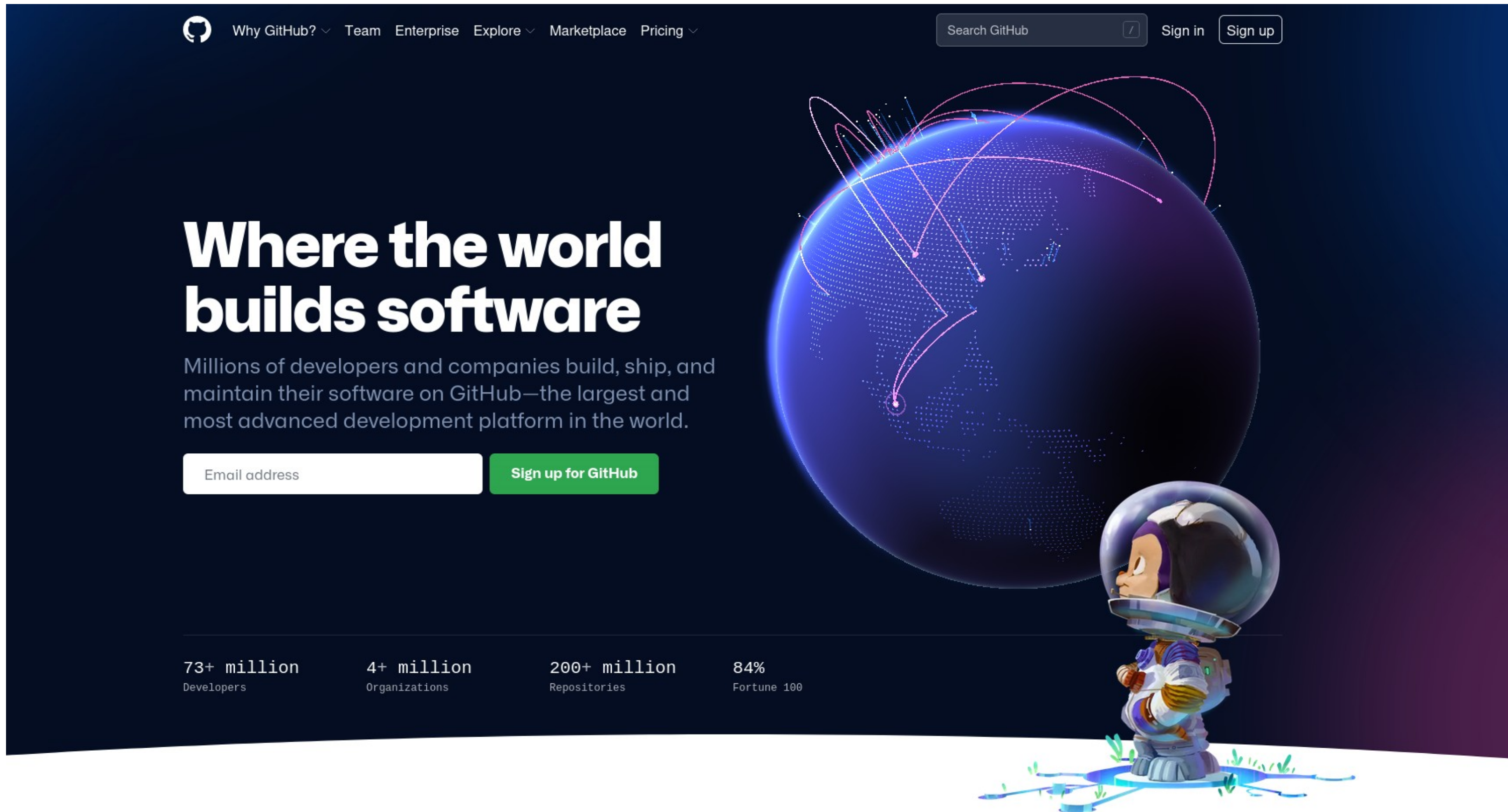
connect to MemGraph using MemgraphLab console

```
http://localhost:3000/lab/overview
```

execute in Query console screen

```
CREATE INDEX ON :Cve(name);
CREATE INDEX ON :Code(name);
```

Why this is important ?



The image shows the GitHub homepage hero section. It features a dark blue background with a large, glowing blue globe on the right side. The globe is covered in a grid of dots and has several red and blue orbital lines around it. In the bottom right corner, there is a small, stylized astronaut character in a white and blue suit, standing on a small patch of green grass. The astronaut is looking up at the globe. On the left side, there is a large white heading "Where the world builds software". Below it, there is a paragraph of text: "Millions of developers and companies build, ship, and maintain their software on GitHub—the largest and most advanced development platform in the world." Below the text, there is a white input field labeled "Email address" and a green button labeled "Sign up for GitHub". At the top of the page, there is a navigation bar with the GitHub logo, links for "Why GitHub?", "Team", "Enterprise", "Explore", "Marketplace", and "Pricing". There is also a search bar labeled "Search GitHub" and buttons for "Sign in" and "Sign up". At the bottom of the page, there is a statistics section with four items: "73+ million Developers", "4+ million Organizations", "200+ million Repositories", and "84% Fortune 100".

Why GitHub? Team Enterprise Explore Marketplace Pricing

Search GitHub Sign in Sign up

Where the world builds software

Millions of developers and companies build, ship, and maintain their software on GitHub—the largest and most advanced development platform in the world.

Email address Sign up for GitHub

73+ million Developers 4+ million Organizations 200+ million Repositories 84% Fortune 100

Why this is important ?

GitHub Security

Trusted by millions of developers

We protect and defend the most trustworthy platform for developers everywhere to create and build software.

Explore security at GitHub Universe

Contact Sales

It is Your Code !

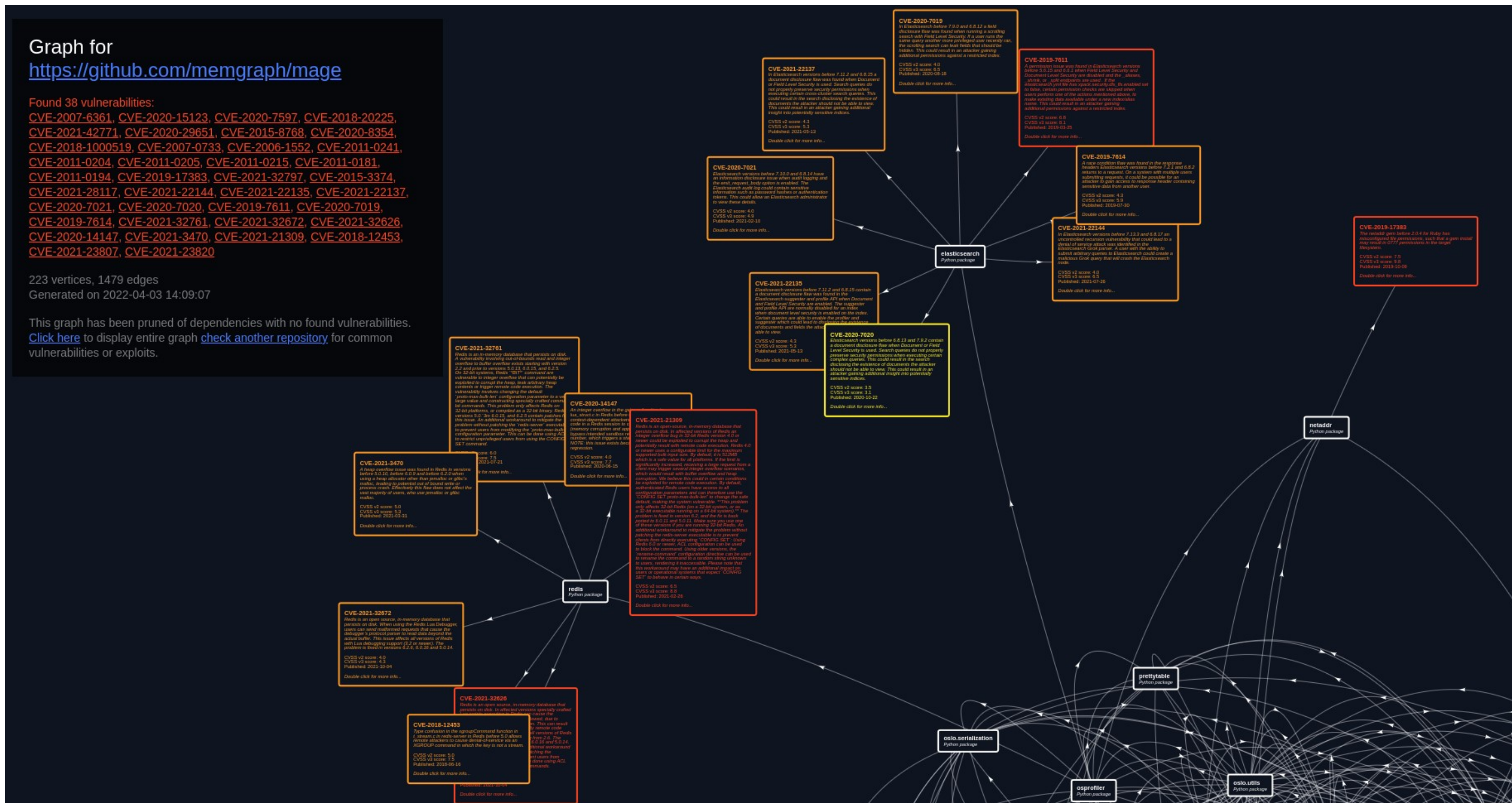
Let's make whole GitHub more secure

Graph for
<https://github.com/memgraph/mage>

Found 38 vulnerabilities:
[CVE-2007-6361](#), [CVE-2020-15123](#), [CVE-2020-7597](#), [CVE-2018-20225](#),
[CVE-2021-42771](#), [CVE-2020-29651](#), [CVE-2015-8768](#), [CVE-2020-8354](#),
[CVE-2018-1000519](#), [CVE-2007-0733](#), [CVE-2006-1552](#), [CVE-2011-0241](#),
[CVE-2011-0204](#), [CVE-2011-0205](#), [CVE-2011-0215](#), [CVE-2011-0181](#),
[CVE-2011-0194](#), [CVE-2019-17383](#), [CVE-2021-32797](#), [CVE-2015-3374](#),
[CVE-2021-28117](#), [CVE-2021-22144](#), [CVE-2021-22135](#), [CVE-2021-22137](#),
[CVE-2020-7021](#), [CVE-2020-7020](#), [CVE-2019-7611](#), [CVE-2020-7019](#),
[CVE-2019-7614](#), [CVE-2021-32761](#), [CVE-2021-32672](#), [CVE-2021-32626](#),
[CVE-2020-14147](#), [CVE-2021-3470](#), [CVE-2021-21309](#), [CVE-2018-12453](#),
[CVE-2021-23807](#), [CVE-2021-23820](#)

223 vertices, 1479 edges
 Generated on 2022-04-03 14:09:07

This graph has been pruned of dependencies with no found vulnerabilities.
[Click here](#) to display entire graph [check another repository](#) for common vulnerabilities or exploits.



Data – CVE Nodes

// Example data in Cypher syntax

// Cve Nodes

```
MERGE (cve:Cve { name: "CVE-2021-28377" })
ON CREATE SET cve.description="ChronoForums 2.0.11 allows an Directory Traversal to read arbitrary files.",
cve.cvss_v2_score="5.0", cve.cvss_v3_score="5.3",
cve.publish_date="2022-01-12";
```

```
MERGE (cve:Cve { name: "CVE-2022-0159" })
ON CREATE SET cve.description="orchardcore is vulnerable to Improper Neutralization of Input During Web Page Generation ('Cross-site Scripting')",
cve.cvss_v2_score="3.5", cve.cvss_v3_score="5.4",
cve.publish_date="2022-01-12";
```

```
MERGE (cve:Cve { name: "CVE-2021-4104" })
ON CREATE SET cve.description="JMSAppender in Log4j 1.2 is vulnerable to deserialization of untrusted data when the attacker has write access to the Log4j
configuration.
The attacker can provide TopicBindingName and TopicConnectionFactoryBindingName configurations causing JMSAppender to perform JNDI requests that result
in remote code execution in a similar fashion to CVE-2021-44228. Note this issue only affects Log4j 1.2 when specifically configured to use JMSAppender, which is
not the default.
Apache Log4j 1.2 reached end of life in August 2015. Users should upgrade to Log4j 2 as it addresses numerous other issues from the previous versions.",
cve.cvss_v2_score="6.0", cve.cvss_v3_score="7.5",
cve.publish_date="2021-12-14";
```

Data – Code Nodes and Edges to Vulnerabilities

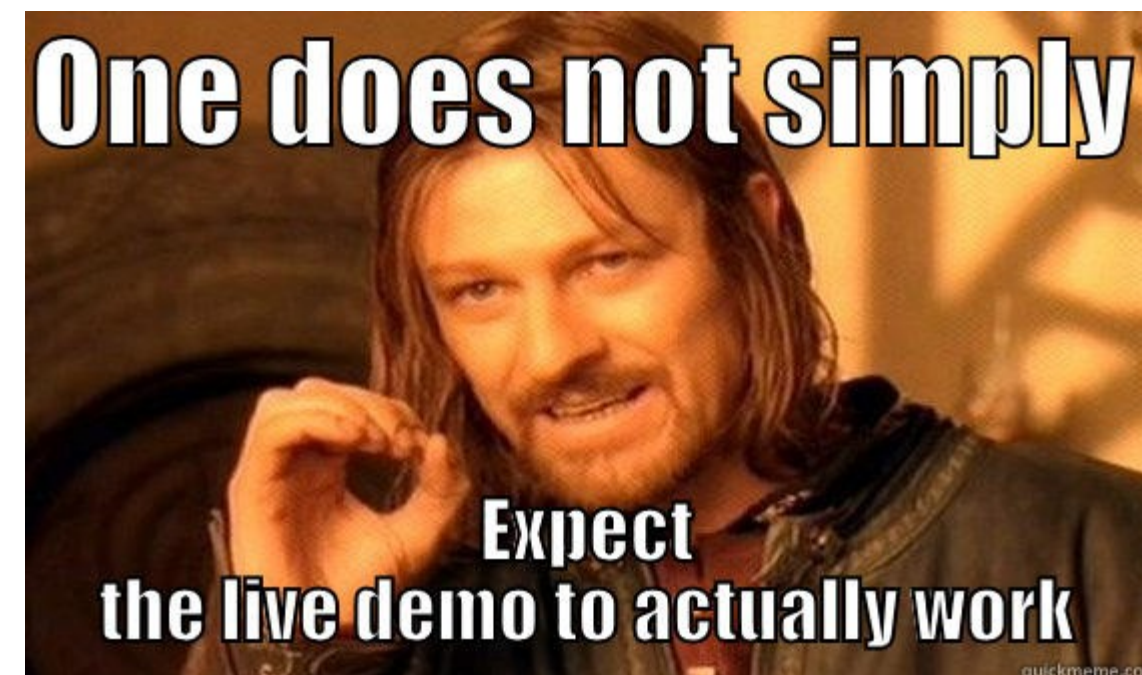
```
// Code Nodes and CPE Edges
```

```
MERGE (code:Code:Repo { name: "easy_forms_for_mailchimp" });
MATCH (cve:Cve {name: "CVE-2021-24985" }), (code:Code { name: "easy_forms_for_mailchimp" }) MERGE (cve)-[:CPE { version_code: "<:6.8.6" }]->(code);
MATCH (cve:Cve {name: "CVE-2019-15318" }), (code:Code { name: "easy_forms_for_mailchimp" }) MERGE (cve)-[:CPE { version_code: "<:6.5.3" }]->(code);

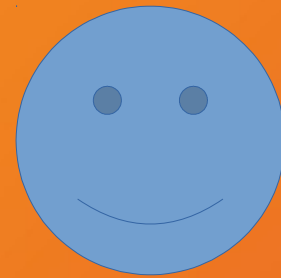
MERGE (code:Code:Repo { name: "log4sh" });
MATCH (cve:Cve {name: "CVE-2005-1915" }), (code:Code { name: "log4sh" }) MERGE (cve)-[:CPE { version_code: "==:1.2.5" }]->(code);
MATCH (cve:Cve {name: "CVE-2005-1915" }), (code:Code { name: "log4sh" }) MERGE (cve)-[:CPE { version_code: "==:1.2.3" }]->(code);
MATCH (cve:Cve {name: "CVE-2005-1915" }), (code:Code { name: "log4sh" }) MERGE (cve)-[:CPE { version_code: "==:1.2.4" }]->(code);

MERGE (code:Code:Repo { name: "dolphin_browser" });
MATCH (cve:Cve {name: "CVE-2010-1730" }), (code:Code { name: "dolphin_browser" }) MERGE (cve)-[:CPE { version_code: "==:2.5.0" }]->(code);

MERGE (code:Code:Repo { name: "zipwrangler" });
MATCH (cve:Cve {name: "CVE-2010-1685" }), (code:Code { name: "zipwrangler" }) MERGE (cve)-[:CPE { version_code: "==:1.20" }]->(code);
```



Thank you



Graph for
<https://github.com/memgraph/gqlalchemy>

No vulnerabilities found!

1 vertices, 0 edges

Generated on 2022-04-03 13:50:51

[Click here](#) to prune dependencies with no found vulnerabilities or [check another repository](#) for common vulnerabilities or exploits.

gqlalchemy
Repository

