GET THE LOW HANGING FRUITS! FINDING SECURITY VULNERABILITIES IN IOS APPS

OWASP APPSEC DAY NZ 2022

BY SVEN SCHLEIER

Thank You to Our Sponsors and Hosts!

Without them, OWASP New Zealand Day couldn't happen

MY NAME IS SVEN!

- Technical Director for WithSecure Consulting
- Living in ☀️ Singapore
- Seasoned Penetration Tester / Security Architect / Application Security Evangelist
- One of the project leaders for:
  - OWASP Mobile Security Testing Guide (MSTG) and
  - OWASP Mobile AppSec Verification Standard (MASVS)

https://www.linkedin.com/in/sven-schleier/

http://bsddaemon.org/

@bsd_daemon

AGENDA FOR TODAY

- Scan for secrets in your code base
- SAST (Static Application Security Testing)
- Software Composition Analysis (SCA)

OWASP APPSEC DAY NZ 2022 - FIND VULNERABILITIES IN IOS APPS

SECRET SCANNING

WHAT ARE SECRETS?

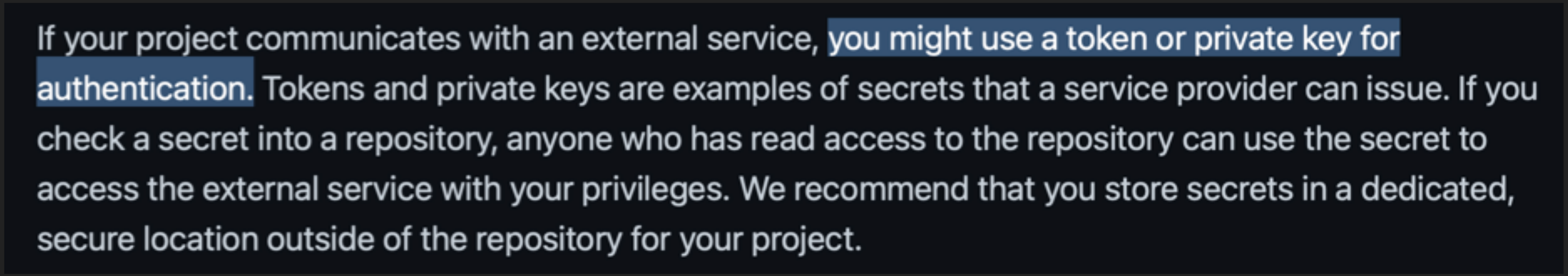
- Encryption Keys
  - Symmetric Encryption Key
  - Private Key (Asymmetric Encryption)
- Certificates (mTLS)
- API Tokens
- Credentials (username and password)

EXAMPLES OF THREATS IN THE CONTEXT OF A MOBILE APP

Abuse Case	Prerequisits	Likelihood	Severity	Risk
As a malicious user, I can decompile/dissasemble the app to identify secrets in the production build.	None	2	2	Medium
The pipeline can use secrets that are not encrypted.	None	2	2	Medium
As a developer, I can commit secrets into the repository.	User interaction	2	3	High
As a developer, I can have secrets on my workstation.	None	2	2	Medium
A 3rd Party SDK is having hardcoded secrets.	None	1	2	Low

THESE ARE JUST EXAMPLES, YOU NEED TO RE-EVALUATE EVERY RISK SPECIFIC TO YOUR APP!

SECRET SCANNING



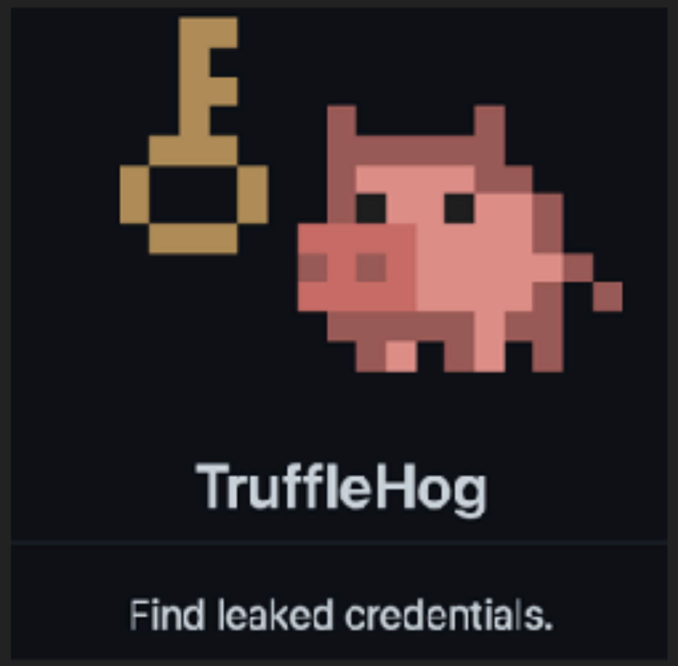
If your project communicates with an external service, you might use a token or private key for authentication. Tokens and private keys are examples of secrets that a service provider can issue. If you check a secret into a repository, anyone who has read access to the repository can use the secret to access the external service with your privileges. We recommend that you store secrets in a dedicated, secure location outside of the repository for your project.

https://docs.github.com/en/code-security/secret-scanning/about-secret-scanning

SECRET SCANNING

Scanning for secrets is a common attack vector against repositories (including scanning the git commit history!):

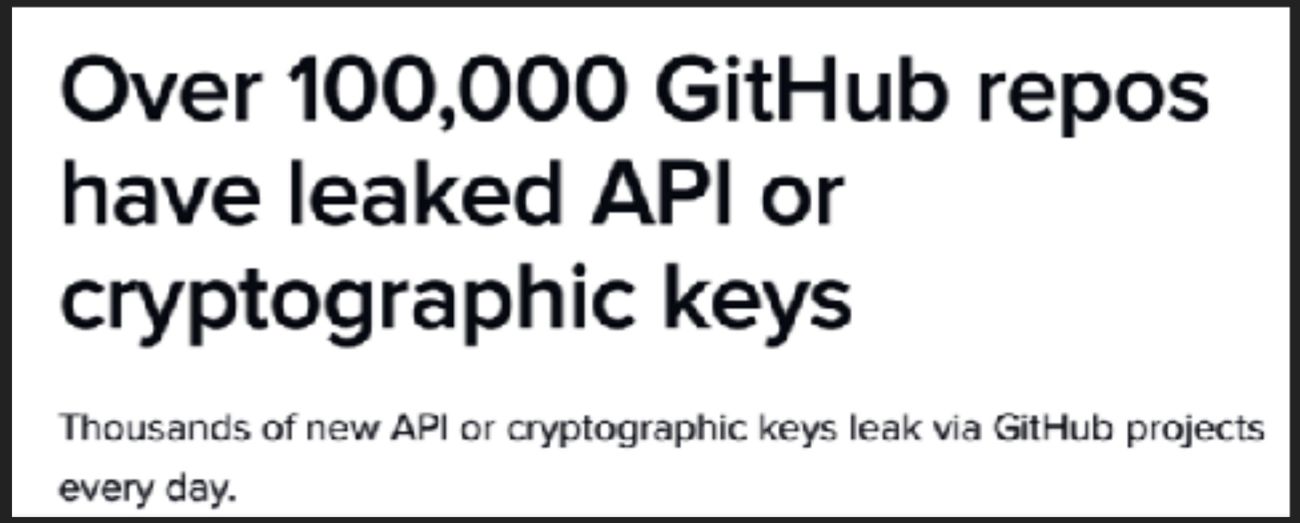
- Public repos are being scanned regularly
- Private repos are being scanned after a breach through an attacker



TruffleHog

Find leaked credentials.

https://github.com/trufflesecurity/trufflehog

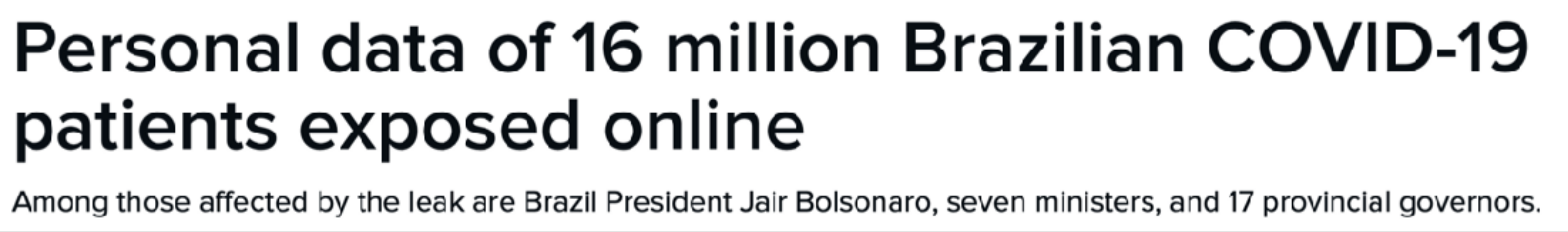


Over 100,000 GitHub repos have leaked API or cryptographic keys

Thousands of new API or cryptographic keys leak via GitHub projects every day.

https://www.zdnet.com/article/over-100000-github-repos-have-leaked-api-or-cryptographic-keys/

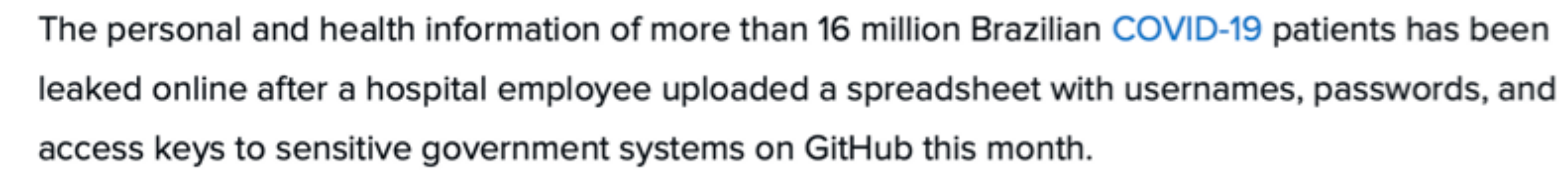
OUCH...



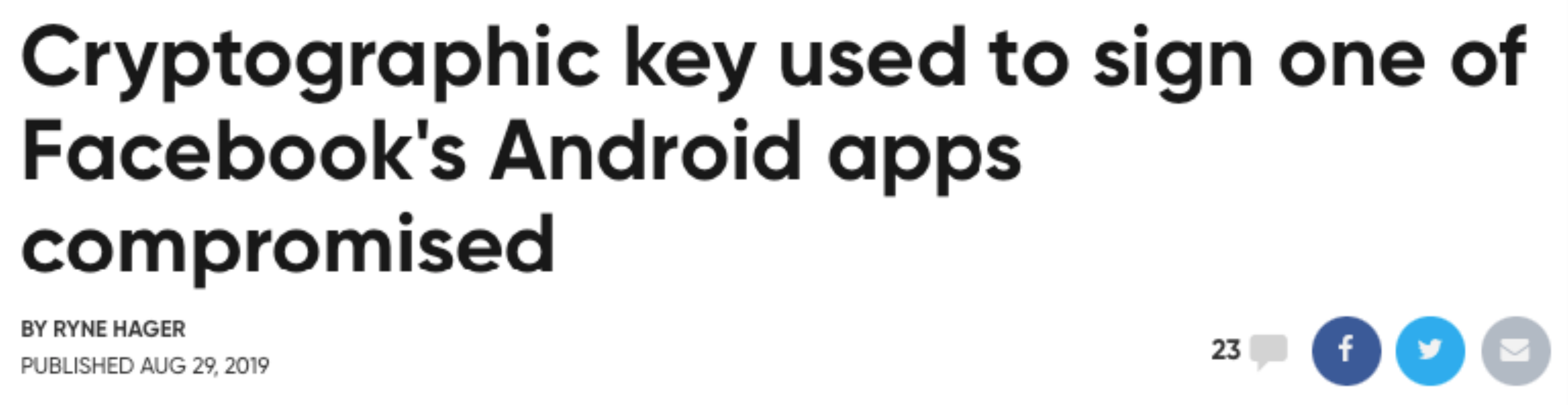
Personal data of 16 million Brazilian COVID-19 patients exposed online

Among those affected by the leak are Brazil President Jair Bolsonaro, seven ministers, and 17 provincial governors.

https://www.zdnet.com/article/personal-data-of-16-million-brazilian-covid-19-patients-exposed-online/



The personal and health information of more than 16 million Brazilian COVID-19 patients has been leaked online after a hospital employee uploaded a spreadsheet with usernames, passwords, and access keys to sensitive government systems on GitHub this month.

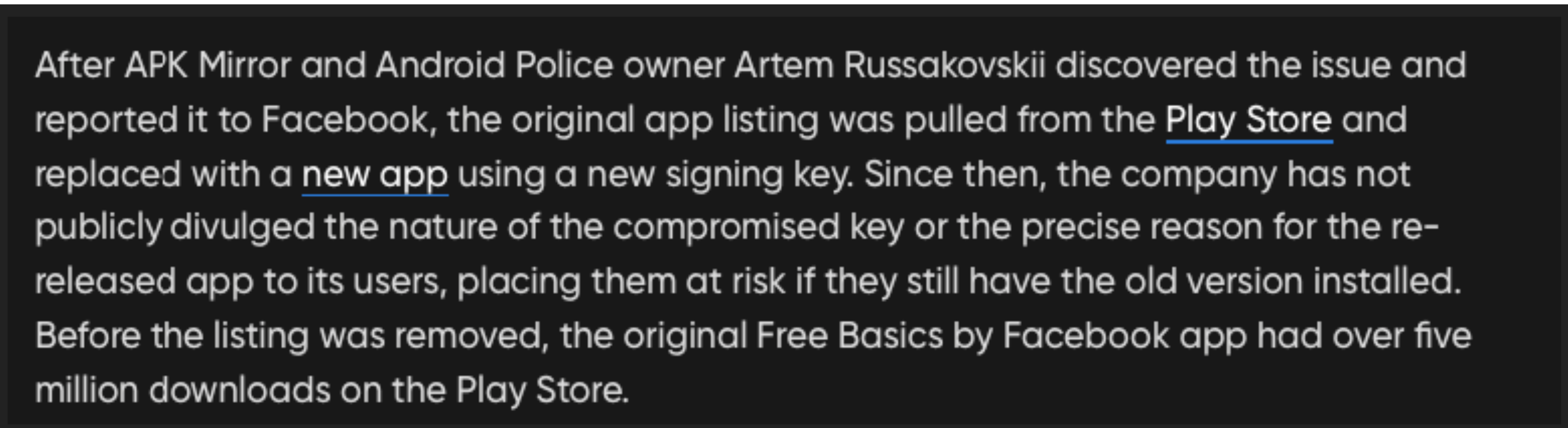


Cryptographic key used to sign one of Facebook's Android apps compromised

BY RYNE HAGER

PUBLISHED AUG 29, 2019

https://www.androidpolice.com/2019/08/29/cryptographic-key-used-to-sign-one-of-facebooks-android-apps-compromised



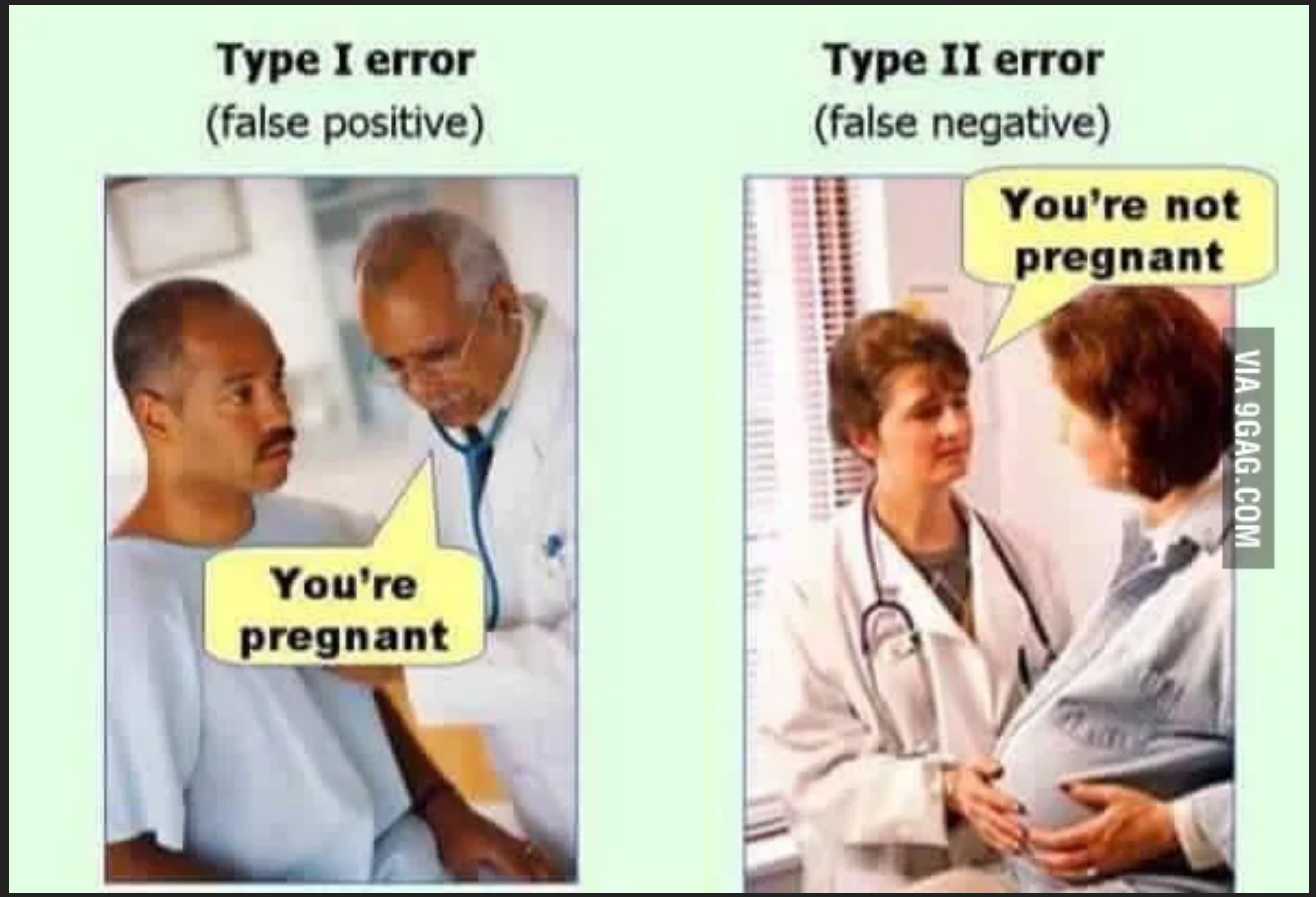
After APK Mirror and Android Police owner Artem Russakovskii discovered the issue and reported it to Facebook, the original app listing was pulled from the Play Store and replaced with a new app using a new signing key. Since then, the company has not publicly divulged the nature of the compromised key or the precise reason for the re-released app to its users, placing them at risk if they still have the old version installed. Before the listing was removed, the original Free Basics by Facebook app had over five million downloads on the Play Store.

SECRET SCANNING

- Usually will flag out some false positives
- Needs tweaking and customisation to your environment
- Should happen throughout the SDLC, e.g.:
  - Pre-commit hook
  - During build in the pipeline

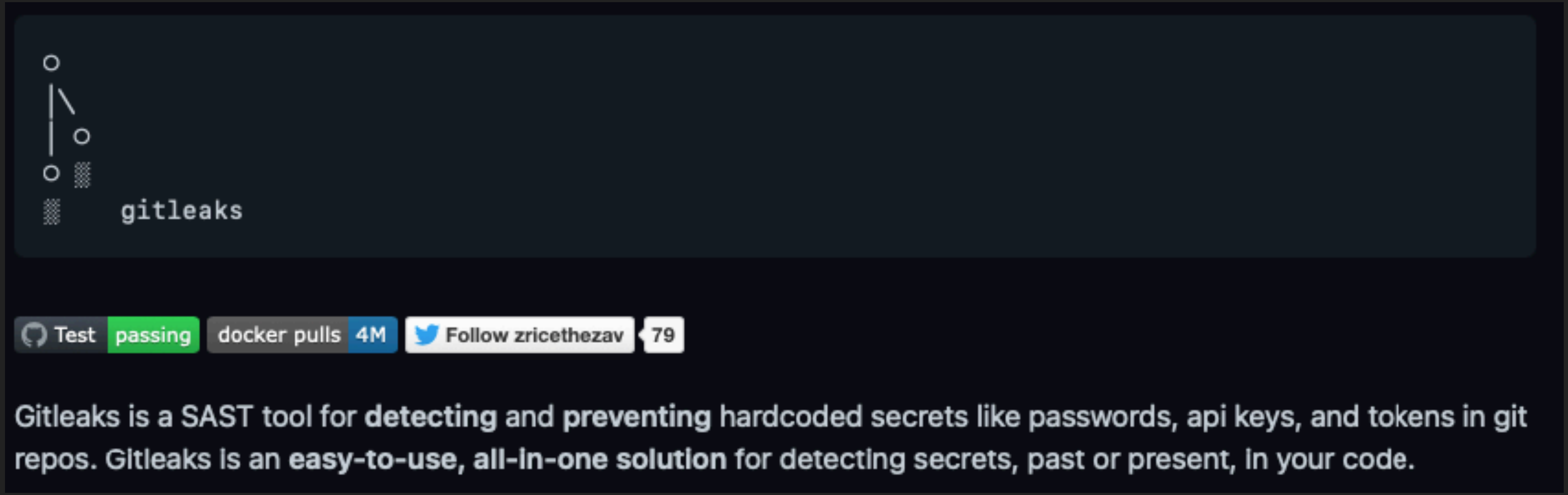
DEFINITIONS OF TERMS

- **False Positives** occur when a scanning tool incorrectly flags a security vulnerability. False positives describe the situation where a test case fails with an identified vulnerability, but there is no bug and the functionality is working correctly.
- **False Negatives** occur when a scanning tool doesn't flag out a security vulnerability. False negatives describe the situation where a test case is cleared without identifying a vulnerability, even though that there is one.



GITLEAKS

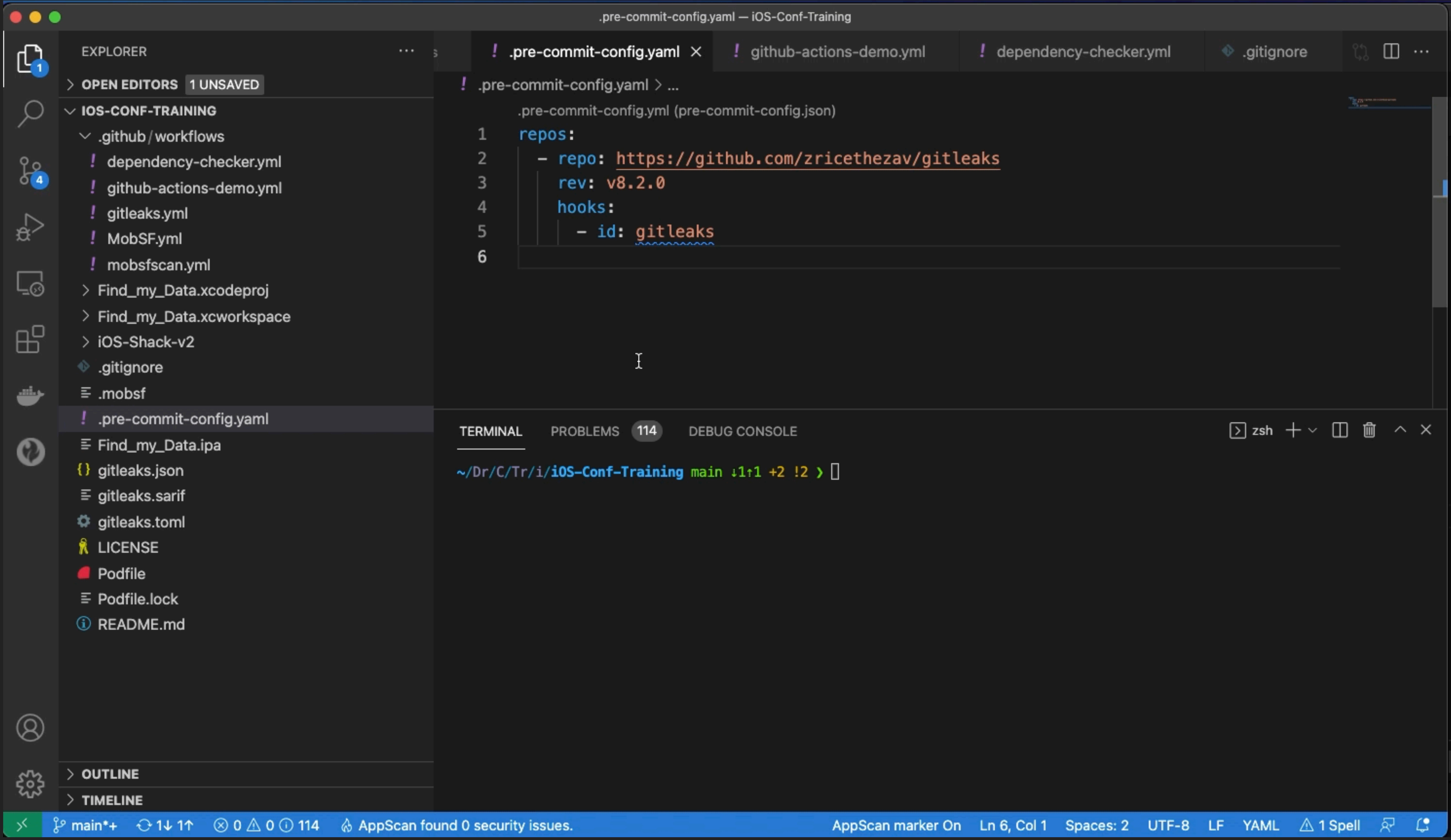
- Many secret scanning tools are out there, we will focus today on **gitleaks** (https://github.com/zricethezav/gitleaks)
- Scan during pre-commit and in the CI/CD pipeline



```
○
|\
| ○
○ ░
░    gitleaks
```

Gitleaks is a SAST tool for **detecting** and **preventing** hardcoded secrets like passwords, api keys, and tokens in git repos. Gitleaks is an **easy-to-use, all-in-one solution** for detecting secrets, past or present, in your code.

FAIL COMMIT



```yaml
repos:
  - repo: https://github.com/zricethezav/gitleaks
    rev: v8.2.0
    hooks:
      - id: gitleaks

```

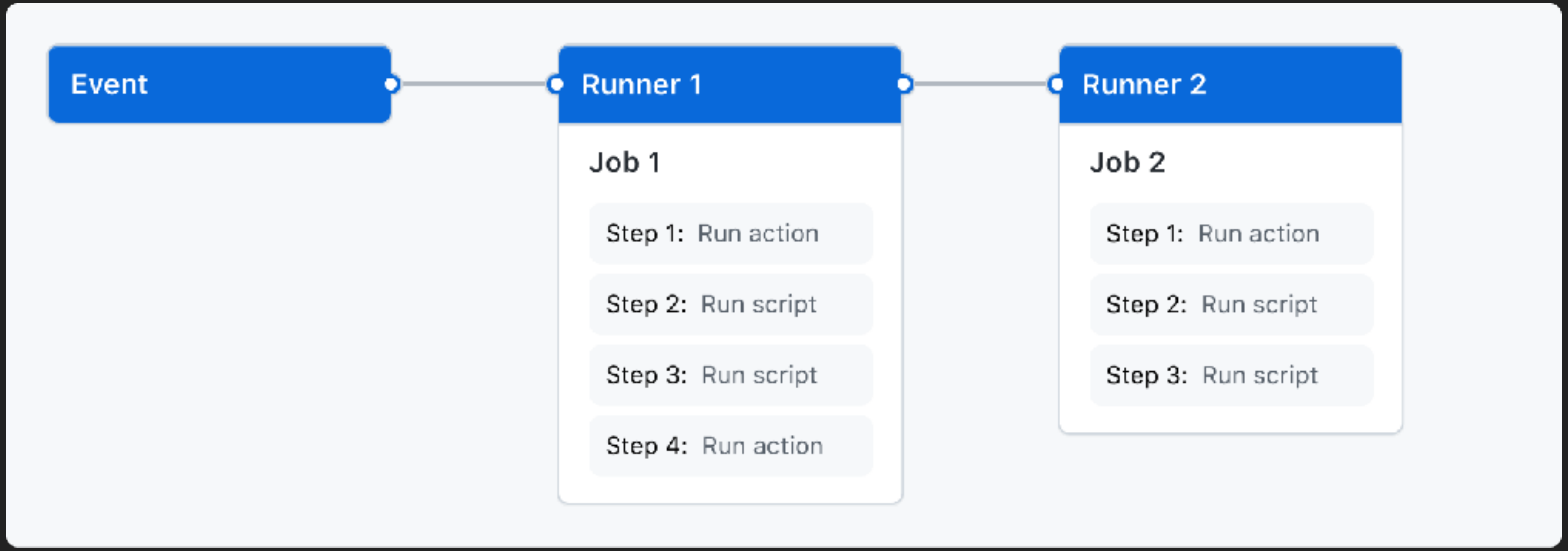
```
~/Dr/C/Tr/i/iOS-Conf-Training main ↓1↑1 +2 !2 ❯ 
```

AppScan found 0 security issues.

GITHUB ACTION

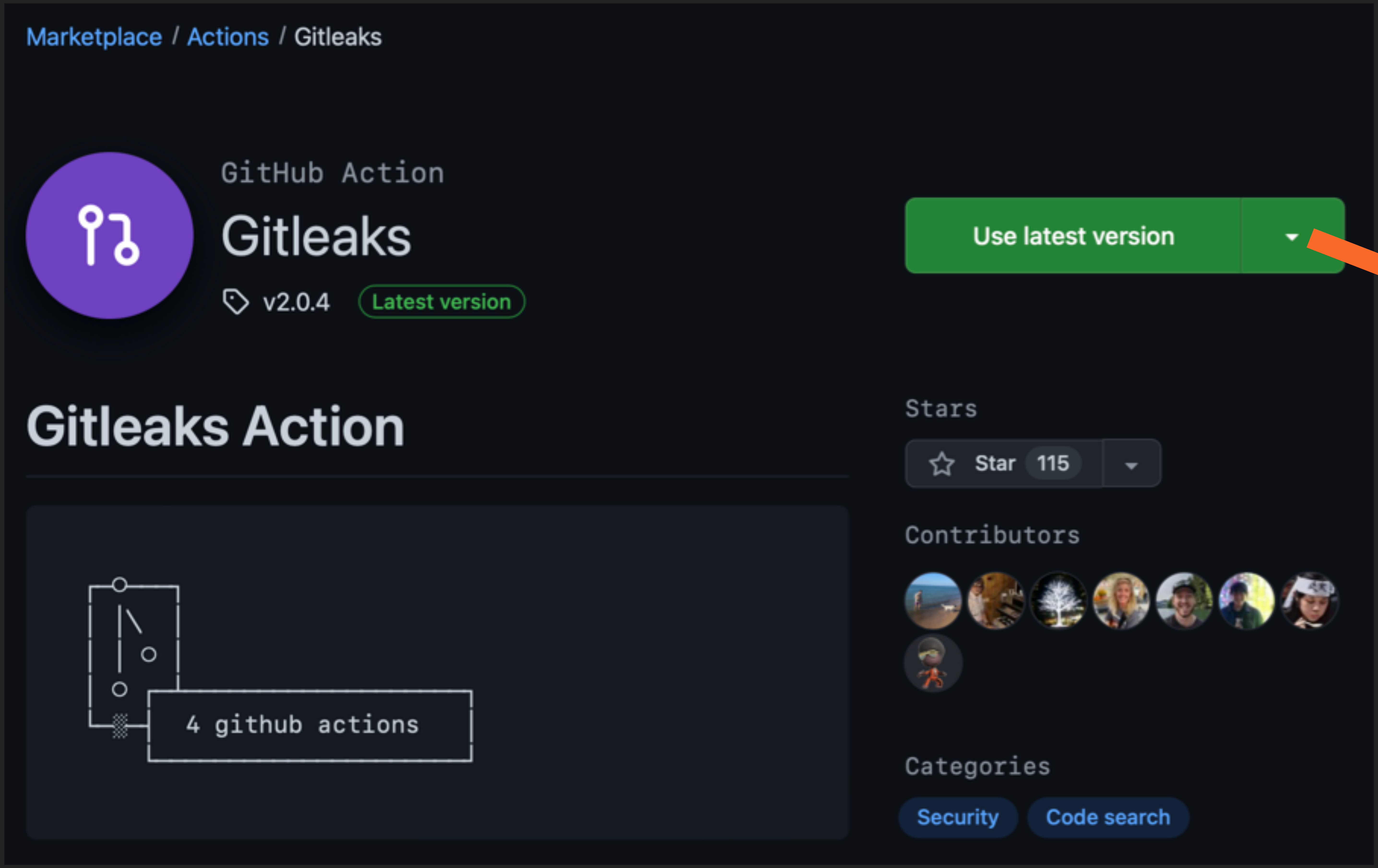
GitHub Actions is a continuous integration and continuous delivery (CI/CD) platform that allows you to automate your build, test, and deployment pipeline.

You can create workflows that build and test every pull request to your repository, or deploy merged pull requests to production.

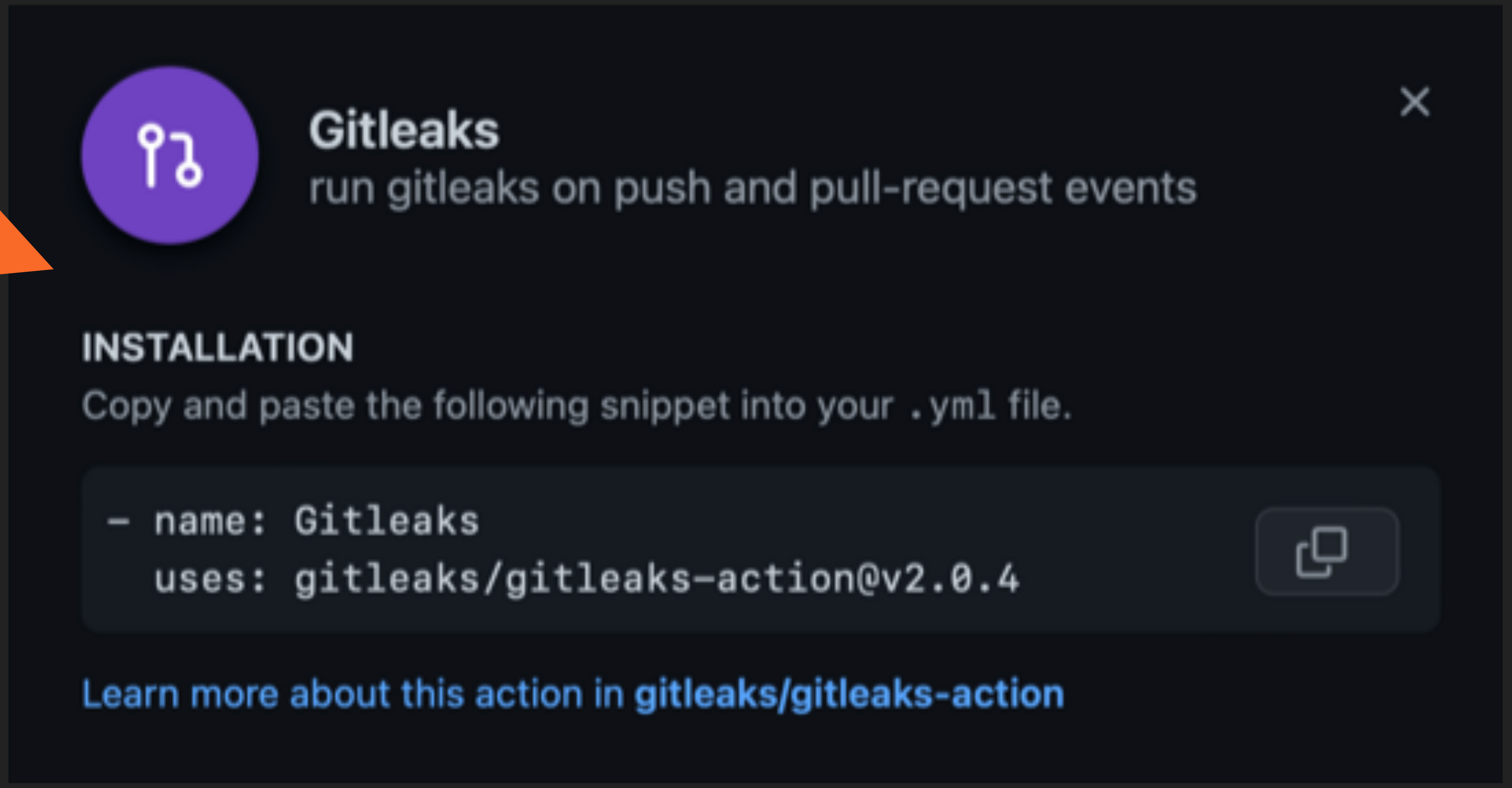


https://docs.github.com/en/actions/learn-github-actions/understanding-github-actions

GITHUB ACTION – GITLEAKS



Gitleaks Action



run gitleaks on push and pull-request events

INSTALLATION

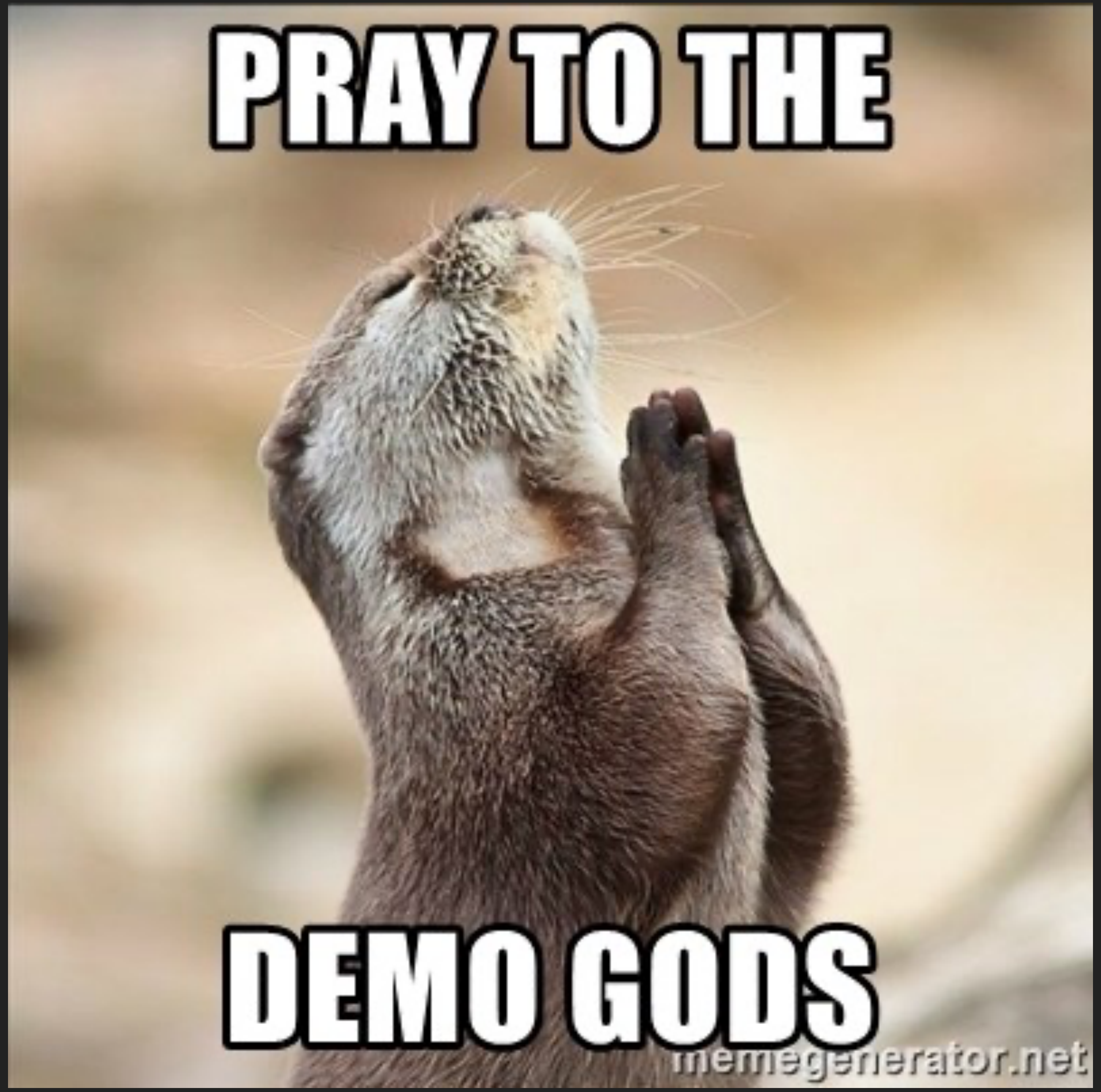
Copy and paste the following snippet into your .yml file.

```
- name: Gitleaks
  uses: gitleaks/gitleaks-action@v2.0.4
```

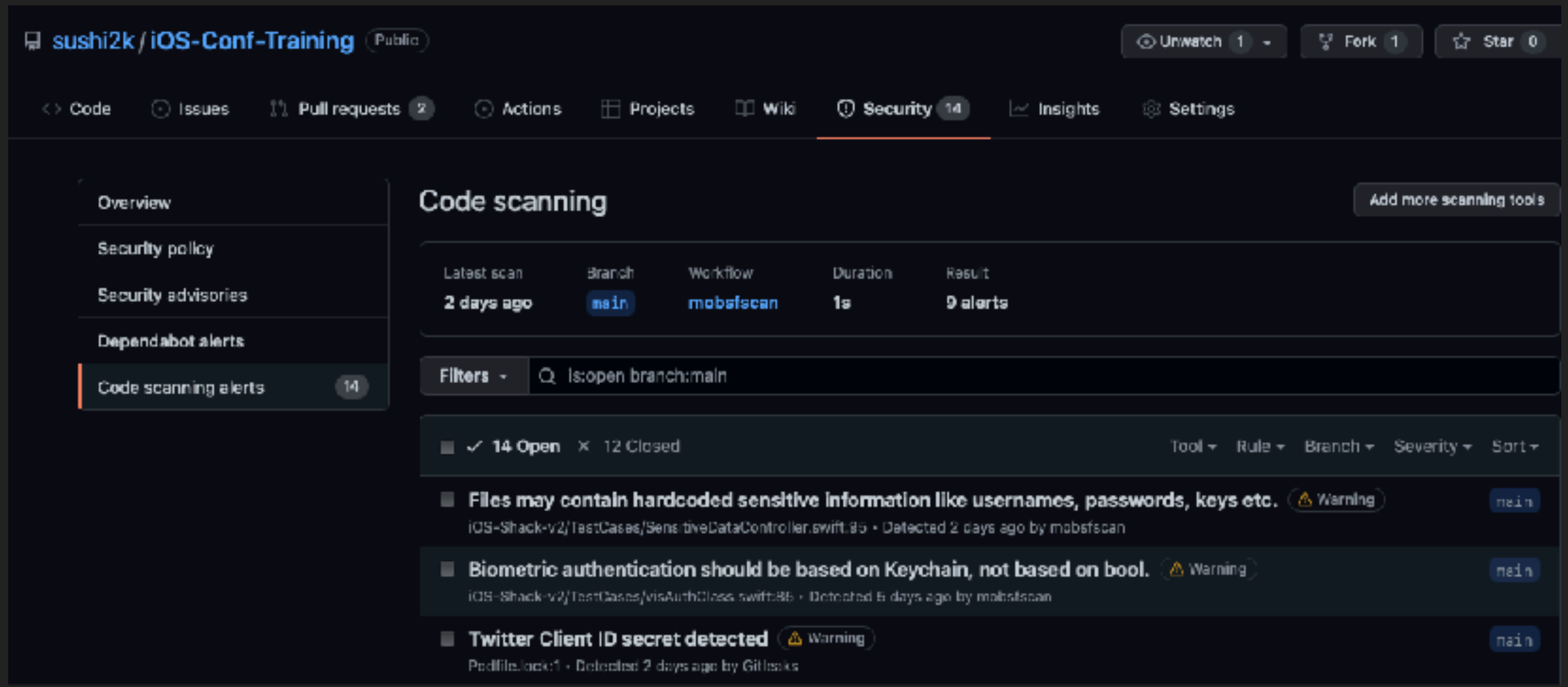
Learn more about this action in gitleaks/gitleaks-action

https://github.com/marketplace/actions/gitleaks

DEMO - SECRET SCANNING WITH GITHUB ACTIONS



WHAT IS SARIF?



Code scanning

Latest scan: 2 days ago | Branch: main | Workflow: mobsfscan | Duration: 1s | Result: 9 alerts

is:open branch:main

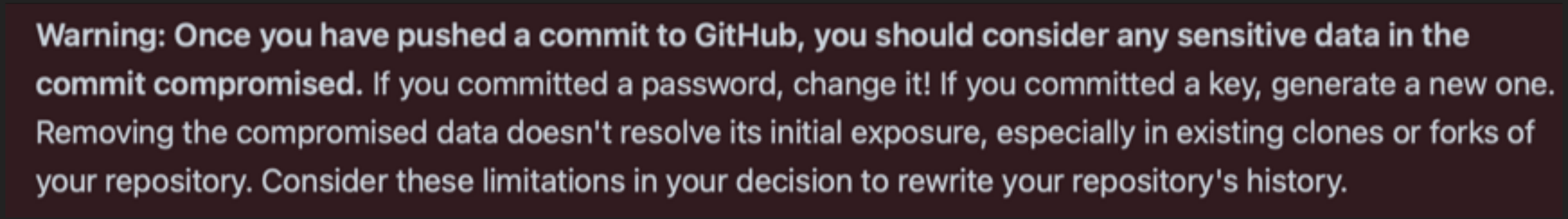
✓ 14 Open × 12 Closed

- Files may contain hardcoded sensitive information like usernames, passwords, keys etc. ⚠ Warning — main
  iOS-Shack-v2/TestCases/SensitiveDataController.swift:85 • Detected 2 days ago by mobsfscan
- Biometric authentication should be based on Keychain, not based on bool. ⚠ Warning — main
  iOS-Shack-v2/TestCases/visAuthClass.swift:86 • Detected 5 days ago by mobsfscan
- Twitter Client ID secret detected ⚠ Warning — main
  Podfile.lock:1 • Detected 2 days ago by Gitleaks

- Static Analysis Results Interchange Format (SARIF)
- Is an output file format and defined as JSON schema
- Is trying to harmonize security tool output and is becoming more popular in various security scanning tools.
- Supported by Github Security / Code Scanning alerts
- Introduction: https://github.com/microsoft/sarif-tutorials

WHAT TO DO IF SENSITIVE INFORMATION WAS IDENTIFIED OR COMMITTED?

- Don’t panic, when you scan the first time most likely the project was living with committed secrets in the repo already for some time and you just finally identified them.



Warning: Once you have pushed a commit to GitHub, you should consider any sensitive data in the commit compromised. If you committed a password, change it! If you committed a key, generate a new one. Removing the compromised data doesn't resolve its initial exposure, especially in existing clones or forks of your repository. Consider these limitations in your decision to rewrite your repository's history.

- First check who is using this secret and notify them that its compromised and that the secret is about to be revoked. This is to avoid disruption to any users and services.

WHAT TO DO IF SENSITIVE INFORMATION WAS IDENTIFIED OR COMMITTED?

- Generate the new secret according to its best practices
- Use a secure mechanism to store the secret and make it available in the CI/CD pipeline:
  - Github Secrets (for Github actions) in Environments
    - https://docs.github.com/en/actions/deployment/targeting-different-environments/using-environments-for-deployment#environment-secrets
    - https://docs.github.com/en/actions/security-guides/encrypted-secrets
  - Hashicorp Vault, AWS Secrets Manager, $cloudprovider
- Revoke the key/reset the password and make the leaked secret unusable.

OWASP APPSEC DAY NZ 2022 - FIND VULNERABILITIES IN IOS APPS

STATIC APPLICATION SECURITY TESTING (SAST)

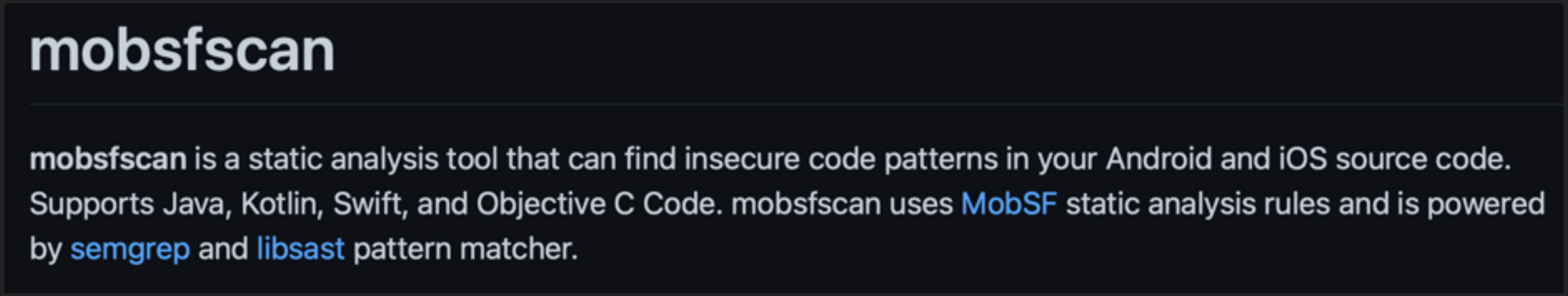
TESTING TECHNIQUES

Linting	Static Testing	SCA	Dynamic Testing
Continuous inspection of code quality to perform automatic reviews with static analysis of code to detect bugs or code smells.	Static Application Security Testing (SAST) involves examining an application's components without executing them, by analyzing the source code only.	Software Composition Analysis (SCA) is scanning the open-source libraries used in your software for known security vulnerabilities.	Dynamic Application Security Testing (DAST) involves examining the app during runtime (Ideally done during QA). This can be done automatically by using a tool or manually.
SonarQube	Semgrep, MobSF	dependency-checker	NowSecure

See the Vulnerability Scanning Tools from OWASP for various open source and enterprise tools:

https://owasp.org/www-community/Vulnerability_Scanning_Tools

MOBSFSCAN



mobsfscan

mobsfscan is a static analysis tool that can find insecure code patterns in your Android and iOS source code. Supports Java, Kotlin, Swift, and Objective C Code. mobsfscan uses MobSF static analysis rules and is powered by semgrep and libsast pattern matcher.

Installation: $ pip install mobsfscan

Can be used as CLI, but also in Github Actions: https://github.com/MobSF/mobsfscan

Export as SARIF possible, so can be integrated into Github Code Scanning

DEMO – SCANNING WITH MOBSFSCAN

OWASP APPSEC DAY NZ 2022 - FIND VULNERABILITIES IN IOS APPS

SOFTWARE COMPOSITION ANALYSIS (SCA)

SCA - SOFTWARE COMPOSITION ANALYSIS

- “Scan 3rd party libraries for security vulnerabilities”
- Majority of software consists of libraries, mobile apps are no exceptio
- Libraries ensure consistent approach and not “reinventing the wheel”
- For security critical operations like encryption libraries should be used!
- Also libraries can have security vulnerabilities that need to be detected
- Massive amount of 3rd party libraries out there

SCA - SOFTWARE COMPOSITION ANALYSIS

- OWASP Dependency Check (Tool to scan libraries in software projects) -
  - https://jeremylong.github.io/DependencyCheck/dependency-check-cli/index.html
- OWASP dependency-check-cli is a command line tool and can be installed via “brew”
- Can detect publicly disclosed vulnerabilities in project libraries.
- The tool will generate a report listing the dependency and it’s risk

SCA - SOFTWARE COMPOSITION ANALYSIS

DEPENDENCY-CHECK
CLI

- 3 widely used package managers for iOS:
  - Swift Package Manager (experimental support by dependency check)
  - Carthage (not supported by dependency check)
  - CocoaPods (experimental support by dependency check)

DEMO – SCANNING WITH DEPENDENCY CHECK

Do we get to mess with him now?
No. We wait until the worst possible moment.
CI/CD Demo
Rosdbob 2018

FIXING VULNERABLE LIBRARIES

Several options:

- There are no false positives in SCA! Even if you are not calling the vulnerable function in a library it need to be fixed.
- Upgrade the library to the "nearest" version that is free of vulnerabilities, e.g. vulnerable version is 2.4.5 and fixed versions are 2.4.6 and 3.0.1. Major version version upgrades might have many changes in the usage of the API and might need refactoring of the code.
- In case there is no fix available for the library:
  - Remove the library (if not needed)
  - Change to another library
  - Accept the risk and security team will track it in their risk register

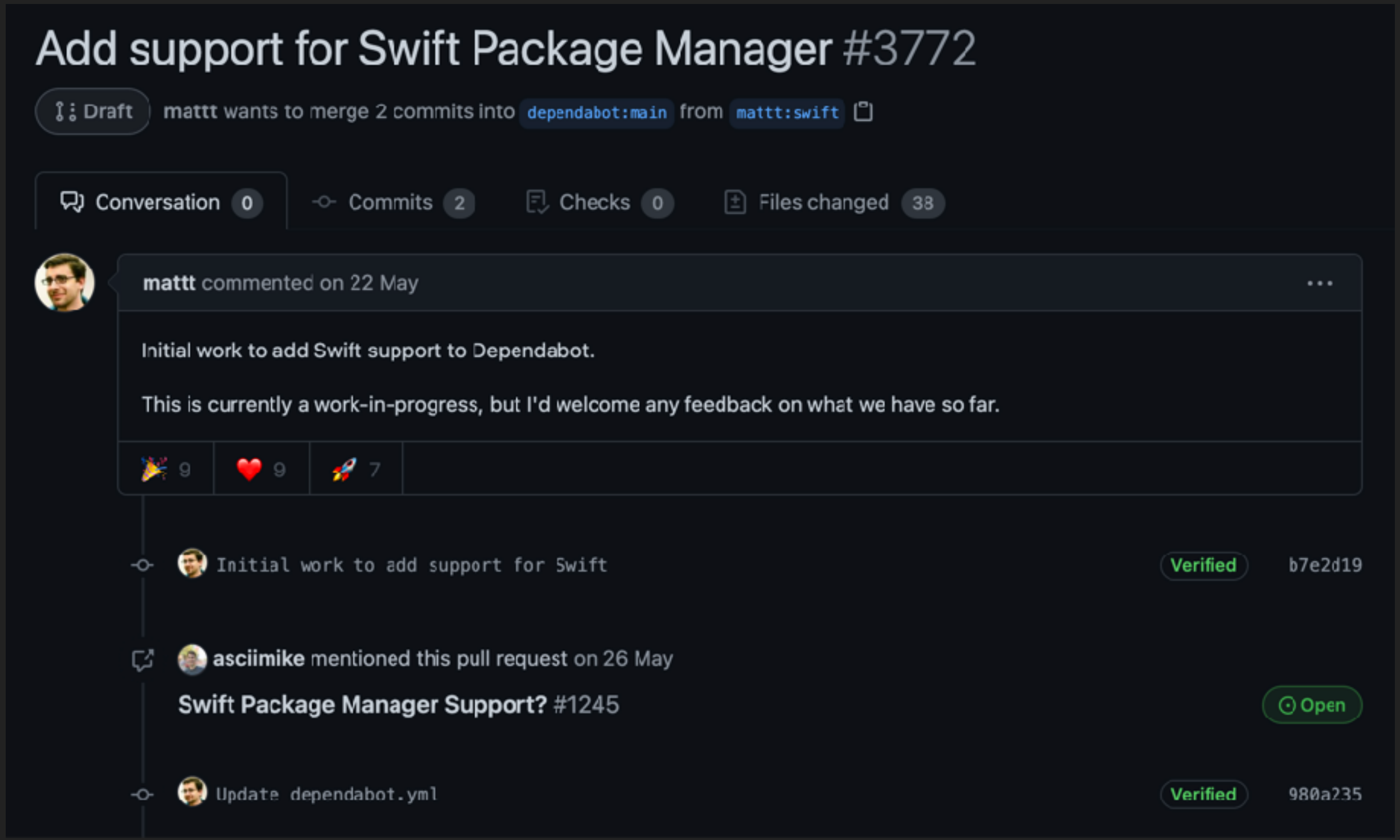
FIXING VULNERABLE LIBRARIES

How do you know which library is free of vulnerabilities? => Manual research or enterprise tools.

Commercial products (sorted alphabetically) can detect outdated libraries and offer an overview of library versions that are free of known vulnerabilities:

- Blackduck
- Snyk
- Sonatype Nexus IQ
  - Cocoapods
  - Swift Package Manager
- $vendor

DEPENDABOT IN GITHUB – WIP FOR SWIFT PACKAGE MANAGER



Add support for Swift Package Manager #3772

Draft mattt wants to merge 2 commits into dependabot:main from mattt:swift

mattt commented on 22 May

Initial work to add Swift support to Dependabot.

This is currently a work-in-progress, but I'd welcome any feedback on what we have so far.

Initial work to add support for Swift — Verified — b7e2d19

asciimike mentioned this pull request on 26 May

Swift Package Manager Support? #1245 — Open

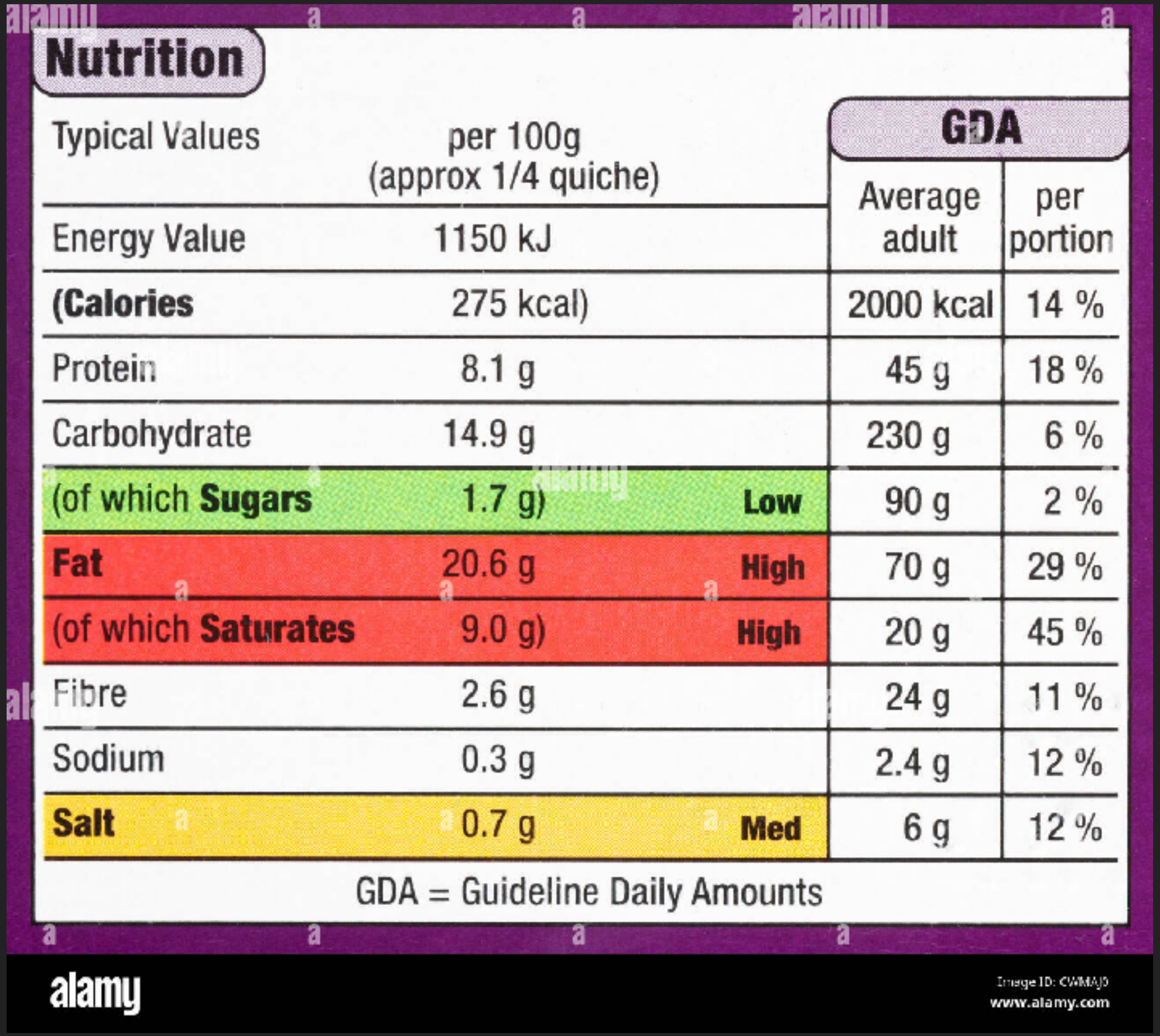
Update dependabot.yml — Verified — 980a235

- https://github.com/dependabot/dependabot-core/pull/3772

BESIDES SCANNING FOR KNOWN VULNERABILITIES - SBOM

Software Bill of Material (SBOM)

- List of components used to build software
- SBOM describes the components in detail
- Single Source of Truth and your inventory for libraries (https://dependencytrack.org)
- Identify affected software that uses vulnerable libraries



Nutrition

Typical Values	per 100g (approx 1/4 quiche)		GDA	
			Average adult	per portion
Energy Value	1150 kJ			
(Calories	275 kcal)		2000 kcal	14 %
Protein	8.1 g		45 g	18 %
Carbohydrate	14.9 g		230 g	6 %
(of which **Sugars**	1.7 g)	**Low**	90 g	2 %
Fat	20.6 g	**High**	70 g	29 %
(of which **Saturates**	9.0 g)	**High**	20 g	45 %
Fibre	2.6 g		24 g	11 %
Sodium	0.3 g		2.4 g	12 %
Salt	0.7 g	**Med**	6 g	12 %

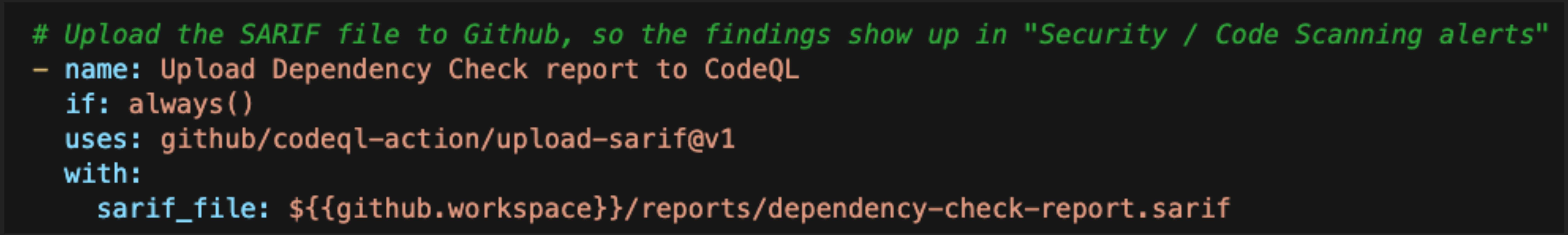
GDA = Guideline Daily Amounts

OWASP APPSEC DAY NZ 2022 - FIND VULNERABILITIES IN IOS APPS

SUMMARY

TIME TO WRAP-UP

- The Github Actions will now be executed every time we are doing a commit or pull request and will add vulnerabilities to the Github “Code Scanning Alerts”.
- SARIF (Static Analysis Results Interchange Format) is a standarized output file format, that is trying to harmonize security scanning tool output. The SARIF standard is used to streamline how static analysis tools share their results and is integrated into Github.
- For example, this is the relevant snippet from the dependency-check.yml file to upload the vulnerabilities:



```
# Upload the SARIF file to Github, so the findings show up in "Security / Code Scanning alerts"
- name: Upload Dependency Check report to CodeQL
  if: always()
  uses: github/codeql-action/upload-sarif@v1
  with:
    sarif_file: ${{github.workspace}}/reports/dependency-check-report.sarif
```

- Software Composition Analysis, Static Scanning of source code and secret scanning should become an integral part of your development workflow to avoid having "low hanging fruits" in your repo.

TIME TO WRAP-UP

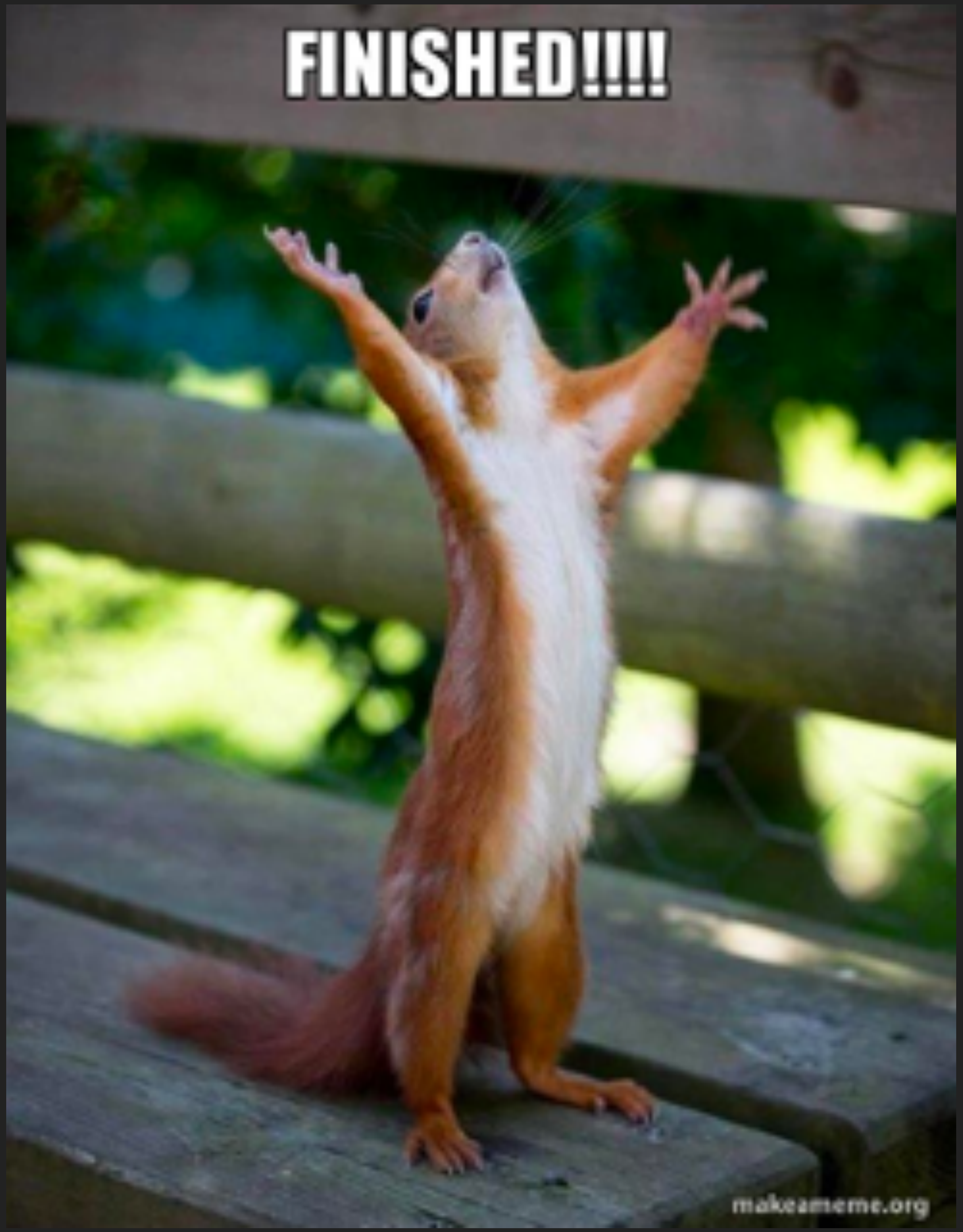
- Tools like gitleaks, dependency-check or mobsfscan can be integrated into your CI/CD pipeline to detect secrets and vulnerabilities early on.
- There are many other open source and enterprise tool that have similar capabilities. Investigate which ones cove your tech stack, but keep in mind that you need to:
  - have a mechanism for de-duplication of findings,
  - be able to flag out false positives permanently,
  - be able to customise the scanning rules and patterns and
  - can manage the findings (which is as important as detecting them, for example throug Github Code Scanning).

SWAG!!!

FRIDA
Catmosphere
CAT CAFE
ARDUINO
APPSECDAY AUSTRALIA
H Ac K
DARK ARMY
OWASP
NEW ZEALAND DAY 2022
Sven
Sven Schleier
Speaker / Trainer
WithSecure
AUT
QUANTUM
CyberCX
DATACOM
snyk
auth0
Checkmarx
kordia
LATERAL SECURITY
MICRO FOCUS
MUST TRY!!
Full of MSTG
移動安全
OPEN SOURCE
Free download
NEW ZEALAND DAY

WE ARE DONE!

Any questions?



Thank you for your time today and I hope you can use your new skills soon and spread your new knowledge!

https://www.linkedin.com/in/sven-schleier/

@bsd_daemon

http://bsddaemon.org/