

Integrating DevSecOps in CI/CD Pipeline

Currently refining the Continuous Integration and Continuous Delivery (CI/CD) pipeline for simple Python app, with a strong emphasis on embedding security measures throughout the entire development lifecycle. This involves applying **DevSecOps** principles by integrating automated security checks directly into the build and testing phases.

1. Create a directory named my-devsecops and include files inside it :

```
## Struktur :  
  
my-devsecops/  
├── .github/  
│   └── workflows/  
│       └── ci-cd-devsecops.yml      # GitHub Actions pipeline untuk CI/CD + security scan  
├── app/  
│   └── main.py                     # Source code aplikasi utama  
├── Dockerfile                     # Instruksi build Docker image  
├── requirements.txt                # Daftar dependency Python (jika ada)  
├── .gitleaks.toml                  # Konfigurasi rule untuk Gitleaks  
├── trivy-ignore.yml                 # Daftar CVE yang di-ignore oleh Trivy  
└── README.md                       # Dokumentasi project
```

main.py

```
from flask import Flask  
  
app = Flask(__name__)  
  
@app.route('/')  
def home():  
    return "Hello from DevSecOps-secured Flask App on port 5100!"  
  
if __name__ == '__main__':  
    app.run(host='0.0.0.0', port=5100)
```

trivy-ignore.yml

```
ignoreUnfixed: true  
severity:  
  - CRITICAL  
  - HIGH
```

ci-cd-devsecops.yml

```
name: DevSecOps CI/CD Pipeline

on:
  push:
    branches: [ "main" ]
  pull_request:
    branches: [ "main" ]

jobs:
  devsecops-pipeline:
    runs-on: ubuntu-latest

    steps:
      - name: Checkout repository
        uses: actions/checkout@v4

      - name: Run Gitleaks
        uses: gitleaks/gitleaks-action@v2
        with:
          config-path: .gitleaks.toml
          fail: true

      - name: Run Trivy FS scan
        uses: aquasecurity/trivy-action@master
        with:
          scan-type: fs
          ignore-unfixed: true
          severity: CRITICAL,HIGH
          exit-code: 1

      - name: Build Docker image
        run: |
          docker build -t my-devsecops:latest .

      - name: Run Trivy on Docker image
        uses: aquasecurity/trivy-action@master
        with:
          image-ref: my-devsecops:latest
          format: table
          exit-code: 0
          ignore-unfixed: true
          severity: CRITICAL,HIGH

      - name: DockerHub Login
        run: echo "${{ secrets.DOCKER_PASSWORD }}" | docker login -u "${{ secrets.DOCKER_USERNAME }}" --password-stdin

      - name: Push Docker image to DockerHub
        run: |
          docker tag my-devsecops:latest ${{ secrets.DOCKER_USERNAME }}/my-devsecops:latest
          docker push ${{ secrets.DOCKER_USERNAME }}/my-devsecops:latest
```

Dockerfile

```
FROM python:3.11-slim

WORKDIR /app

COPY requirements.txt .
RUN pip install --no-cache-dir -r requirements.txt

COPY app/ .

CMD ["python", "main.py"]

(base) dynoaryawana@Dynos-MacBook-Pro my-devsecops %
```

.gitleaks.toml

```
[allowlist]
description = "Allow known test patterns"
regexes = ['dummy-secret']
```

requirements.txt

```
flask==2.3.3
```

Running Flask

← → ↺ 127.0.0.1:5100



Hello from DevSecOps-secured Flask App on port 5100!

Installing Trivy and Gitleaks on mac with :

- brew install gitleaks
- brew install aquasecurity/trivy/trivy

Creating Image

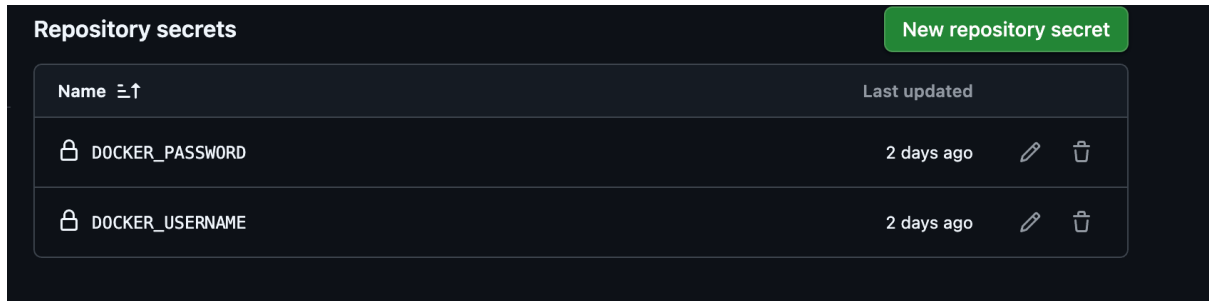
(base) dynamyawall@Dynos-MacBook-Pro my-devsecops % docker build -t my-devsecops .

```
[+] Building 11.3s (11/11) FINISHED
=> [internal] load build definition from Dockerfile
=> => transferring dockerfile: 220B
=> [internal] load metadata for docker.io/library/python:3.11-slim
=> [auth] library/python:pull token for registry-1.docker.io
=> [internal] load .dockerignore
=> => transferring context: 2B
=> [1/5] FROM docker.io/library/python:3.11-slim@sha256:139020233cc412ef4c813500efc17569dc8b28dd88b109b764f8977e30
=> => resolve docker.io/library/python:3.11-slim@sha256:139020233cc412ef4c813500efc17569dc8b28dd88b109b764f8977e30
=> => sha256:08296bc8060181c1317303f5013a081f9cd078913707486330c4222c4f0f89 16.14MB / 16.14MB
=> => sha256:6e88b460b0856dca77c5ce5fc92d0fd2b23bde9fcf47c0b39ac5949c05 3.34MB / 3.34MB
=> => sha256:37259e7330667afd74c3386d3ed869f06bd9b7714370c78e3065f4e28607cc02 28.08MB / 28.08MB
=> => extracting sha256:37259e7330667afd74c3386d3ed869f06bd9b7714370c78e3065f4e28607cc02
=> => extracting sha256:6e88b460b0856dca77c5ce5fc92d0fd2b23bde9fcf47c0b39ac5949c05
=> => extracting sha256:08296bc8060181c1317303f5013a081f9cd078913707486330c4222c4f0f89
=> => extracting sha256:f22d47757e2c8bdf36bee053bcc2c5a040225436c69f1dafd67209044
=> [internal] load build context
=> => transferring context: 446B
=> [2/5] WORKDIR /app
=> [3/5] COPY requirements.txt
=> [4/5] RUN pip install --no-cache-dir -r requirements.txt
=> [5/5] COPY app/
=> => exporting to image
=> => exporting layers
=> => exporting manifest sha256:59184129fc995307a93114496c1fa848c9480633fcf324b24c0309a58e99
=> => exporting config sha256:56b0c1f42911f60b9c28bdc21c04486fd39c196b07b153b04f4e52e9706505
=> => exporting attestation manifest sha256:9a54cdfc3d5b1eddb7578ac3d073f1759da86e060a4174ac5d7fbd19c12
=> => exporting manifest list sha256:db0182a7084168f823681f6a329c99440330851d5d45050e17d979a38e97a008
=> => naming to docker.io/library/my-devsecops:latest
=> => unpacking to docker.io/library/my-devsecops:latest
View build details: docker-desktop://dashboard/build/desktop:linux/desktop:linux/1a1ezfhfk7eba0833011a90
(base) dynamyawall@Dynos-MacBook-Pro my-devsecops %
```

Running container

```
docker run -p 5100:5100 my-devsecops
```

Enter Docker username and Password into github :



Test trivy (scan the current directory)

```
(base) dynoaryawana@Dynos-MacBook-Pro my-devsecops % trivy fs .
2025-07-01T20:05:52+03:00      INFO    [vuln] Need to update DB
2025-07-01T20:05:52+03:00      INFO    [vuln] Downloading vulnerability DB...
2025-07-01T20:05:52+03:00      INFO    [vuln] Downloading artifact...      repo="mirror.gcr.io/aquasec/trivy-db:2"
66.11 MiB / 66.11 MiB [-----] 100.00% 7.46 MiB p/s 9.1s
2025-07-01T20:06:03+03:00      INFO    [vuln] Artifact successfully downloaded      repo="mirror.gcr.io/aquasec/trivy-db:2"
2025-07-01T20:06:03+03:00      INFO    [vuln] Vulnerability scanning is enabled
2025-07-01T20:06:03+03:00      INFO    [secret] Secret scanning is enabled
2025-07-01T20:06:03+03:00      INFO    [secret] If your scanning is slow, please try '--scanners vuln' to disable secret scanning
2025-07-01T20:06:03+03:00      INFO    [secret] Please see also https://trivy.dev/v0.64/docs/scanner/secret#recommendation for faster
er secret detection
2025-07-01T20:06:03+03:00      INFO    Number of language-specific files      num=1
2025-07-01T20:06:03+03:00      INFO    [pip] Detecting vulnerabilities...

Report Summary

```

Target	Type	Vulnerabilities	Secrets
requirements.txt	pip	0	-

```

Legend:
- '-': Not scanned
- '0': Clean (no security findings detected)

(base) dynoaryawana@Dynos-MacBook-Pro my-devsecops %
```

Test trivy (scan image)

```
(base) dynoaryawana@Dynos-MacBook-Pro my-devsecops % trivy image my-devsecops
2025-07-01T20:06:56+03:00      INFO    [vuln] Vulnerability scanning is enabled
2025-07-01T20:06:56+03:00      INFO    [secret] Secret scanning is enabled
2025-07-01T20:06:56+03:00      INFO    [secret] If your scanning is slow, please try '--scanners vuln' to disable secret scanning
2025-07-01T20:06:56+03:00      INFO    [secret] Please see also https://trivy.dev/v0.64/docs/scanner/secret#recommendation for faster secret detection
2025-07-01T20:06:57+03:00      INFO    [python] licenses acquired from one or more METADATA files may be subject to additional terms. Use '--debug' flag to see all affected packages.
2025-07-01T20:06:58+03:00      INFO    Detected OS: family=debian version=12.11
2025-07-01T20:06:58+03:00      INFO    [debian] Detecting vulnerabilities...      os.version="12" pkg_num=105
2025-07-01T20:06:58+03:00      INFO    Number of language-specific files      num=1
2025-07-01T20:06:58+03:00      INFO    [python-pkg] Detecting vulnerabilities...
2025-07-01T20:06:58+03:00      WARN    Using severities from other vendors for some vulnerabilities. Read https://trivy.dev/v0.64/docs/scanner/vulnerability#severity-selection for details.
2025-07-01T20:06:58+03:00      INFO    Table result includes only package filenames. Use '--format json' option to get the full path to the package file.

Report Summary

```

Target	Type	Vulnerabilities	Secrets
my-devsecops (debian 12.11)	debian	101	-
usr/local/lib/python3.11/site-packages/MarkupSafe-3.0.2.dist-info/METADATA	python-pkg	0	-
usr/local/lib/python3.11/site-packages/blinker-1.9.0.dist-info/METADATA	python-pkg	0	-
usr/local/lib/python3.11/site-packages/click-8.2.1.dist-info/METADATA	python-pkg	0	-
usr/local/lib/python3.11/site-packages/flask-2.3.3.dist-info/METADATA	python-pkg	0	-
usr/local/lib/python3.11/site-packages/itsdangerous-2.2.0.dist-info/METADATA	python-pkg	0	-
usr/local/lib/python3.11/site-packages/jinja2-3.1.6.dist-info/METADATA	python-pkg	0	-
usr/local/lib/python3.11/site-packages/pip-24.0.dist-info/METADATA	python-pkg	0	-
usr/local/lib/python3.11/site-packages/setuptools-65.5.1.dist-info/METADATA	python-pkg	2	-
usr/local/lib/python3.11/site-packages/werkzeug-3.1.3.dist-info/METADATA	python-pkg	0	-
usr/local/lib/python3.11/site-packages/wheel-0.45.1.dist-info/METADATA	python-pkg	0	-

```

Legend:
- '-': Not scanned
- '0': Clean (no security findings detected)


my-devsecops (debian 12.11)
Total: 101 (UNKNOWN: 1, LOW: 74, MEDIUM: 18, HIGH: 7, CRITICAL: 1)

Library      Vulnerability      Severity      Status      Installed Version      Fixed Version      Title

```

Scan with gitleaks :

```
(base) dynoaryawana@Dynos-MacBook-Pro my-devsecops % gitleaks detect --source .
```



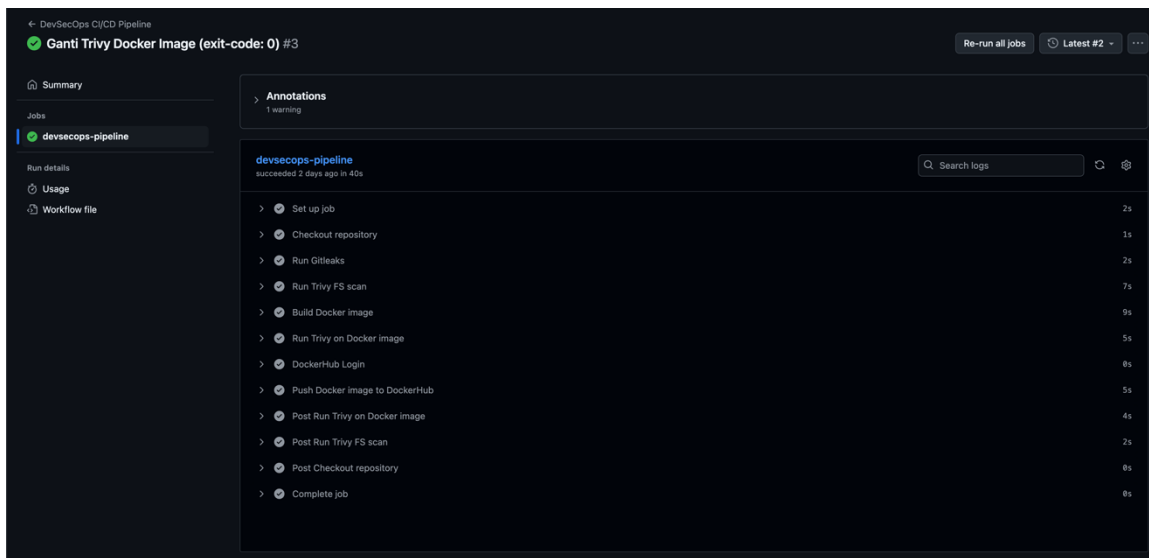
```
6:21PM INF 3 commits scanned.
6:21PM INF scanned ~1836 bytes (1.84 KB) in 41.1ms
6:21PM INF no leaks found
(base) dynoaryawana@Dynos-MacBook-Pro my-devsecops %
```

Push into github with :

```
git init
git remote add origin https://github.com/Dyno07/my-devsecops.git
git branch -M main

git add .
git commit -m
git push -u origin main
```

CI/CD Github



The screenshot shows the GitHub Actions interface for a workflow named 'devsecops-pipeline'. The workflow is in a 'Completed' state, having succeeded 2 days ago in 40s. The left sidebar shows the 'Summary' tab selected. The main area displays a list of jobs with their durations:

Job	Duration
Set up job	2s
Checkout repository	1s
Run Gitleaks	2s
Run Trivy FS scan	7s
Build Docker image	9s
Run Trivy on Docker image	5s
DockerHub Login	8s
Push Docker image to DockerHub	5s
Post Run Trivy on Docker image	4s
Post Run Trivy FS scan	2s
Post Checkout repository	8s
Complete job	8s

THANK YOU