Python Code Quality & Security Implementation Guide

Overview

This document provides a comprehensive plan for implementing code quality, typing, and security checks for Python projects using GitHub Actions, from local development through CI/CD deployment.

Local Development (Pre-commit)

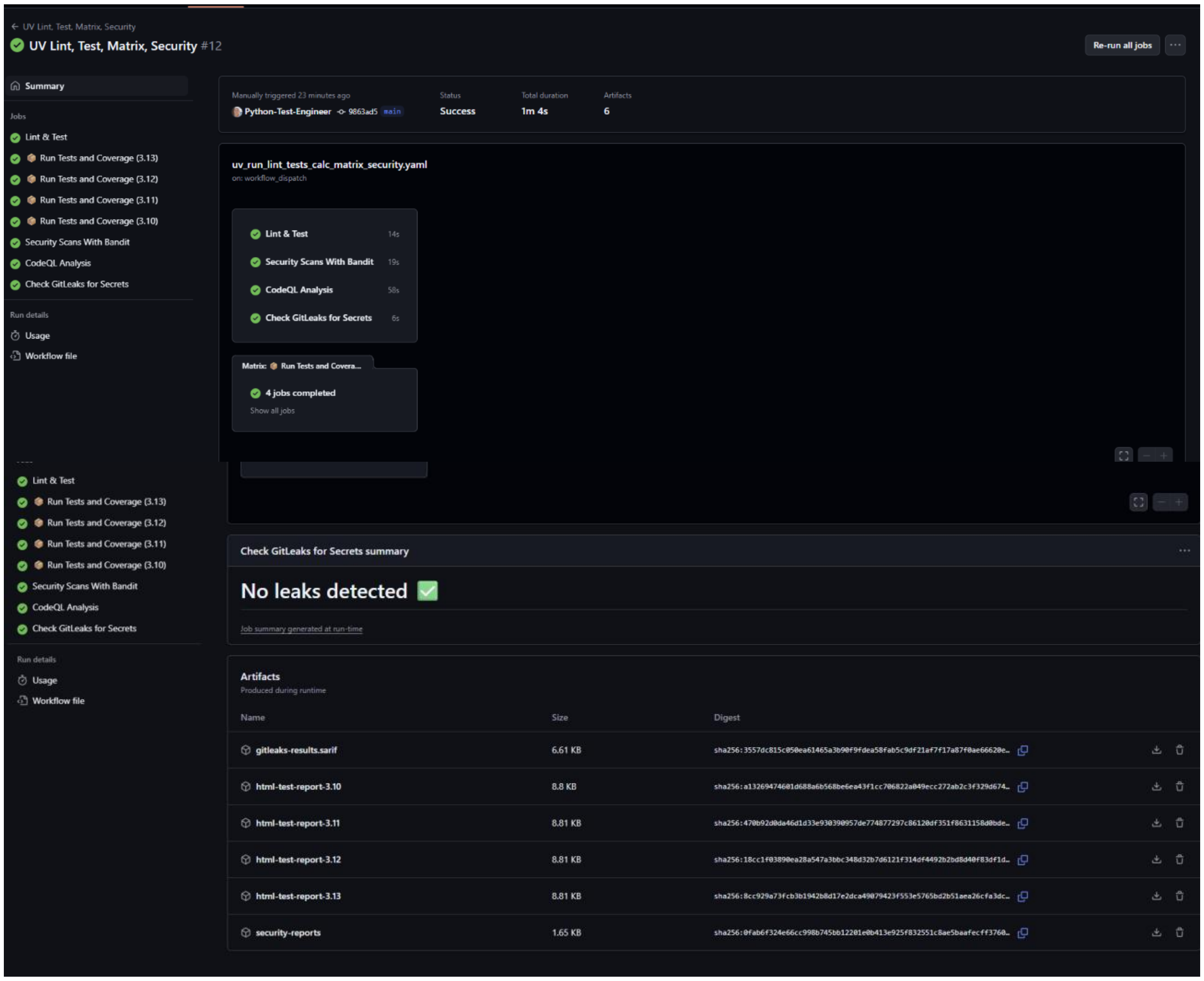
Pre-commit Hook Configuration

- Precommit hooks built in
  - id: trailing-whitespace
  - id: end-of-file-fixer
  - id: check-yaml
  - id: check-added-large-files
  - id: check-merge-conflict
  - id: debug-statements
- id: pyright Pyright is a powerful, fast, and feature-packed static type checker explicitly designed for Python. It helps to ensure code quality, catch errors early, and boost productivity through static type checking.
- id: ruff-check linter args: [–fix]
- id: ruff-format
- id: conventional-pre-commit ensures commit message is standardised
  - stages: [commit-msg]
- id: python-script LEAK DETECTION CUSTOM
  - **enables easy dev change if needed.**
  - name: Python Script
  - entry: python custom.py
  - language: python
  - stages: [pre-commit]
  - *We can add other checks here if needed*
- id: bandit

Conventional Commit Messages

- This provides structure around what has been committed and can be used for versioning.

CI/CD Pipeline (GitHub Actions)



← UV Lint, Test, Matrix, Security

UV Lint, Test, Matrix, Security #12

Re-run all jobs

Summary

Jobs

- Lint & Test
- Run Tests and Coverage (3.13)
- Run Tests and Coverage (3.12)
- Run Tests and Coverage (3.11)
- Run Tests and Coverage (3.10)
- Security Scans With Bandit
- CodeQL Analysis
- Check GitLeaks for Secrets

Run details

- Usage
- Workflow file

Manually triggered 23 minutes ago	Status	Total duration	Artifacts
Python-Test-Engineer -○- 9863ad5 main	Success	1m 4s	6

uv_run_lint_tests_calc_matrix_security.yaml
on: workflow_dispatch

Lint & Test	14s
Security Scans With Bandit	19s
CodeQL Analysis	58s
Check GitLeaks for Secrets	6s

Matrix: Run Tests and Covera...
4 jobs completed
Show all jobs

- Lint & Test
- Run Tests and Coverage (3.13)
- Run Tests and Coverage (3.12)
- Run Tests and Coverage (3.11)
- Run Tests and Coverage (3.10)
- Security Scans With Bandit
- CodeQL Analysis
- Check GitLeaks for Secrets

Run details

- Usage
- Workflow file

Check GitLeaks for Secrets summary

No leaks detected ✅

Job summary generated at run-time

Artifacts
Produced during runtime

Name	Size	Digest
gitleaks-results.sarif	6.61 KB	sha256:3557dc815c050ea61465a3b90f9fdea58fab5c9df21af7f17a87f0ae66620e...
html-test-report-3.10	8.8 KB	sha256:a13269474601d688a6b568be6ea43f1cc706822a049ecc272ab2c3f329d674...
html-test-report-3.11	8.81 KB	sha256:470b92d0da46d1d33e930390957de774877297c86120df351f8631158d0bde...
html-test-report-3.12	8.81 KB	sha256:18cc1f03890ea28a547a3bbc348d32b7d6121f314df4492b2bd8d40f83df1d...
html-test-report-3.13	8.81 KB	sha256:8cc929a73fcb3b1942b8d17e2dca49079423f553e5765bd2b51aea26cfa3dc...
security-reports	1.65 KB	sha256:0fab6f324e66cc998b745bb12201e0b413e925f832551c8ae5baafecff3760...

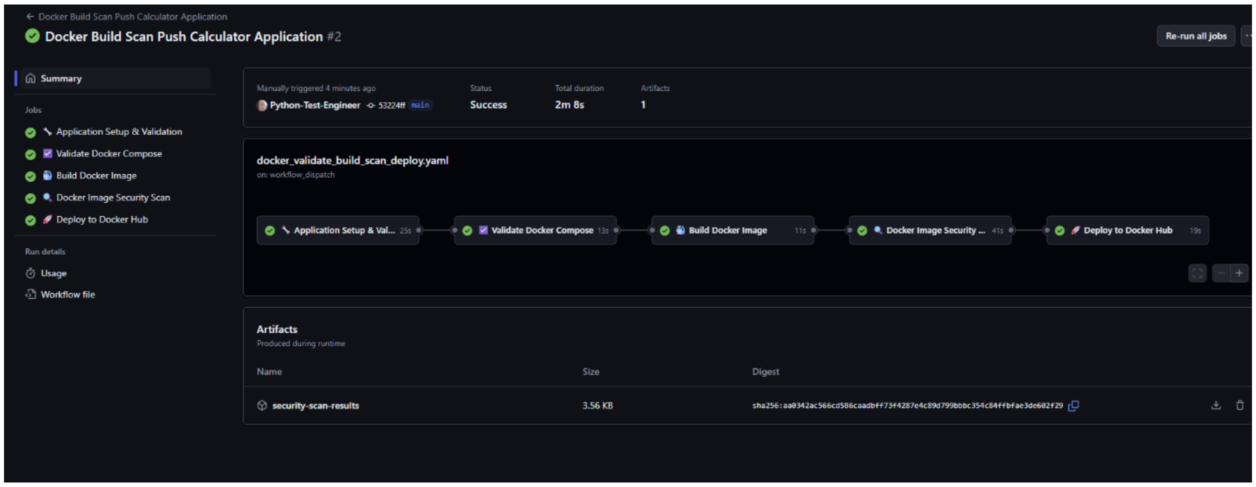
Build Matrix Strategy

- Set up matrix testing across Python versions (3.9, 3.10, 3.11, 3.12).
- Test across Ubuntu only.
- Install dependencies with caching for faster build times.
- Download Pytest-HTML reports.

Code and Security Quality Checks

- Run Ruff, MyPy/Pyright, Unit tests/Coverage, CodeQL, Bandit, Safety,Pip-audit in CI
- Scan for secrets and credentials in commit history and codebase
- Check for dependency vulnerabilities with GitHub's dependency review TO DO

Docker



← Docker Build Scan Push Calculator Application

Docker Build Scan Push Calculator Application #2

Jobs

- Application Setup & Validation
- Validate Docker Compose
- Build Docker Image
- Docker Image Security Scan
- Deploy to Docker Hub

Run details

Manually triggered 4 minutes ago	Status	Total duration	Artifacts
Python-Test-Engineer 53224ff main	Success	2m 8s	1

docker_validate_build_scan_deploy.yaml
on: workflow_dispatch

Application Setup & Val... 25s → Validate Docker Compose 13s → Build Docker Image 11s → Docker Image Security ... 41s → Deploy to Docker Hub 19s

Artifacts
Produced during runtime

Name	Size	Digest
security-scan-results	3.56 KB	sha256:aa0342ac566cd586caadbff73f4287e4c89d799bbbc354c84ffbfae3de602f29

We use Trivy and Docker Scout, inbuilt in GitHub Actions, to scan Docker images for vulnerabilities and secrets, with a downloadable report.

We can run matric Python version for the Dockerfile using arguments: –build-arg PYTHON_VERSION=3.9 etc 02_CI/DockerfileMultiple and 02/cicd_pipeline.yaml.