The ORT project
ORT 프로젝트

OpenChain KWG April 2024

Thomas Steenbergen

Photo by Markus Winkler on Unsplash

OSS
Review Toolkit

Thomas Steenbergen 👋🏻

I help organizations manage open source in a strategic, safe and efficient manner that meets their business needs

저는 조직이 오픈소스를 전략적이고 안전하며 효율적인 방식으로
전략적이고 안전하며 효율적인 방식으로 오픈소스를 관리하도록 돕습니다

Former Head of OSPO at EPAM & HERE. Currently #OpenToWork

EPAM & HERE 의 전 오픈 소스 책임자. 현재 일자리를 찾고 있습니다

Maintainer / Contributor to:

유지 관리자 / 기여자:

OSS Review Toolkit

// TODO

SPDX

OPENCHAIN

See also / 참조하세요: github.com/tsteenbe

What are you working on? 어떤 작업을 하고있나요??

Working on answering / 답변 작업 중:

How can we do open source at **scale and speed**
whilst staying **safe**, **respect licenses**,
enable upstream **sustainability**,
and make **life easier** for developers?

어떻게 하면 안전을 유지하면서 규모와 속도에 맞게 오픈 소스를 수행하고, 라이선스를 존중하며, 업스트림 지속 가능성을 지원하고, 개발자의 삶을 더 쉽게 만들 수 있을까요?

See also / 참조하세요: github.com/tsteenbe

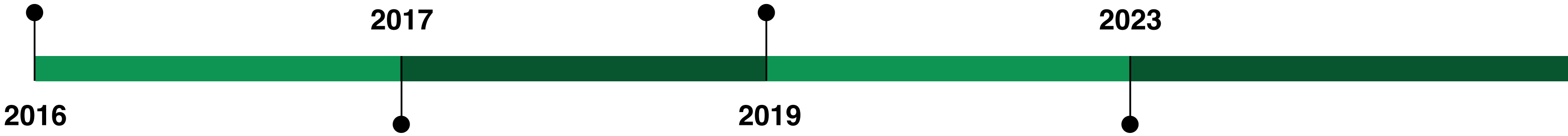
What is ORT? ORT란 무엇인가요?

The OSS Review Toolkit (ORT) is a **FOSS policy automation and orchestration toolkit** which you can use to manage your (open source) software dependencies in a strategic, safe and efficient manner.

ORT는 전략적이고 안전하며 효율적인 방식으로 (오픈 소스) 소프트웨어 종속성을 관리하는 데 사용할 수 있는 FOSS 정책 자동화 및 오케스트레이션 툴킷입니다

See also / 참조하세요: https://oss-review-toolkit.org

What is ORT history? ORT 히스토리란 무엇인가요?



2016

ORT development started

ORT is born within the Open Source Office of HERE Technologies - a mapping and location data company

ORT 개발 시작

ORT는 지도 및 위치 데이터 회사인 HERE Technologies의 오픈 소스 사무실에서 탄생했습니다

2017

Code open sourced on GitHub

HERE publishes ORT as a company open source project to enable collaboration with its suppliers and customers

GitHub에 공개된 코드

HERE는 공급업체 및 고객과의 협업을 가능하게 하기 위해 ORT를 회사 오픈 소스 프로젝트로 공개했습니다

2019

ORT become a Linux Foundation project

ORT became part of ACT (automatecompliance.org)

ORT가 Linux Foundation 프로젝트가 됨

ORT가 ACT(automatecompliance.org)의 일부가 되었습니다

2023

First ORT Community Days

Maintainers and users meet in Berlin to meet, learn and collaborate

첫 번째 "ORT Community Days" 이벤트

메인테이너와 사용자들이 베를린에서 만나서 배우고 협력했습니다

Why was ORT created? ORT는 왜 만들었나요?

- Commercial solutions are expensive, and no single one met our shared technical, licensing & contractual requirements
  상용 솔루션은 비용이 많이 들고, 당사의 공통 기술, 라이선스 및 계약 요구 사항을 충족하는 솔루션이 없습니다

  Key issues with commercial solutions:
  - Number of supported build tools
  - Proof for each found license
  - Unable to code the desired FOSS policy using available data

  상용 솔루션의 주요 문제
  - 지원되는 빌드 도구 수
  - 각 라이선스에 대한 증명이 누락됨
  - 사용 가능한 데이터를 사용하여 원하는 FOSS 정책을 코딩할 수 없음

- Our ideal automation solution was to glue together multiple good enough open source tools and supplement as needed with commercial tools to fill the gaps
  우리의 이상적인 자동화 솔루션은 충분히 좋은 여러 오픈소스 도구를 하나로 묶어 오픈 소스 도구를 결합하고 필요에 따라 상용 도구로 보완하여 부족한 부분을 채우는 것이었습니다

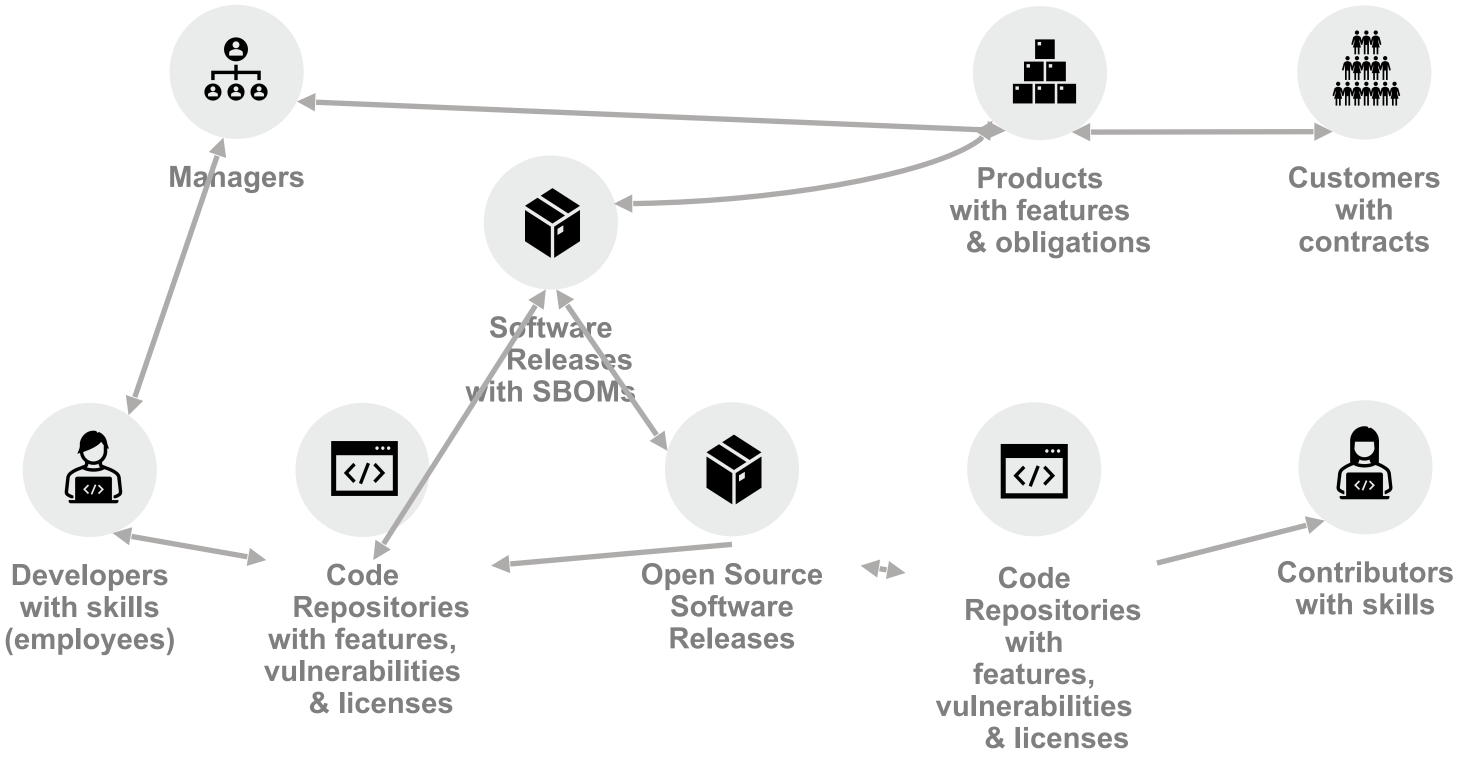
- Open source solutions are the only way forward to resolve in the long-term the licensing, security and sustainability issues within the open source community
  오픈 소스 솔루션은 오픈 소스 커뮤니티의 라이선스, 보안 및 지속 가능성 문제를 장기적으로 해결할 수 있는 유일한 방법입니다

Automation is key to scaling an underfunded OSPO

Focusing on automating recurring inquiries, we discovered similar data is needed.

Engineering	Which devs have skills X and/or know Product Y? -> Code repositories -> People -> Skills & Products? Who can help with open source project Z? -> Code repositories -> uses Z as dependency -> People? How to level up engineering (open source) standards? -> Projects -> Code Repositories -> Best Practices?
Security	Is project X safe to use? -> Project + Dependencies -> SBOM -> Vulnerabilities? What is the impact of known vulnerability X in Y? Dependency Y in SBOMs -> Products? -> Customers?
Licensing	Are licenses / contracts respected? -> Project -> Dependencies -> Source Code -> Licensing? OK to use open source project Z? -> Project Z -> Source Code -> Licensing & Contracts?
Open Sourcing	OK to use open source project Z? -> Project Z -> Source Code & Dependencies -> Licensing & Security? Who is using/contributing open source project Z? -> Projects -> Dependencies -> People How to educate best practices? -> Project -> Code Repository -> README.md / CONTRIBUTING.md?
Business Continuity	Is open source project Z healthy? -> Project Z -> Source Code + Community -> Metrics? Which open source projects are essential? -> Products -> Projects -> Dependencies -> Open Source?
InnerSource	How to educate best practices? -> Project -> Code Repository -> README.md / CONTRIBUTING.md? Where are the active projects and people? -> Code Repositories -> Projects & People?

Translating inquiries into a graph…



Managers

Products with features & obligations

Customers with contracts

Software Releases with SBOMs

Developers with skills (employees)

Code Repositories with features, vulnerabilities & licenses

Open Source Software Releases

Code Repositories with features, vulnerabilities & licenses

Contributors with skills

Key features? 주요 기능은?

You can use ORT to / 다음과 같은 용도로 사용할 수 있습니다:

- **Generate CycloneDX, SPDX SBOMs**, or custom FOSS attribution docs
  CycloneDX, SPDX SBOM 또는 사용자 정의 FOSS 어트리뷰션 문서를 생성합니다
- **Automate your FOSS policy** using risk-based Policy as Code to do licensing, security vulnerability, InnerSource and engineering standards checks
  위험 기반 Policy as Code를 사용하여 라이선스, 보안 취약성, 내부 소스 및 엔지니어링 표준 검사를 수행하여 FOSS 정책을 자동화합니다
- **Create a source code archive** for your software project and its dependencies to comply with certain licenses or for business continuity
  특정 라이선스를 준수하거나 비즈니스 연속성을 위해 소프트웨어 프로젝트와 그 종속성에 대한 소스 코드 아카이브를 만듭니다
- **Correct package metadata or license findings** - fix up issues within your organization leveraging InnerSource or with the help of the FOSS community
  패키지 메타데이터 또는 라이선스 결과 수정 - InnerSource 혹은 FOSS 커뮤니티의 도움을 활용해 조직 내 오픈소스 패키지 문제를 해결합니다.

Supported languages? 지원 언어?

Our focus is on supporting library and application development
라이브러리 및 애플리케이션 개발 지원에 중점을 두고 있습니다

C/C++	Bazel*, Conan	Objective-C / Swift	Carthage*, CocoaPods*, SwiftPM
Flutter	Pub	PHP	Composer
Go	GoMod	Python	Pip, Pipenv, Poetry
Haskell	Stack	Ruby	Bundler*
Java	Gradle, Maven	Rust	Cargo
JavaScript	Bower, NPM*, PNPM*, Yarn (1, 2+)	Scala	SBT
.Net	DotNet*, NuGet*	Others	Package-List YAML or SPDX documents

*supported with limitations / 제한적으로 지원됨

Supported CI/CD? 지원 CI/CD?

GitHub Actions

github.com/oss-review-toolkit/ort-ci-github-action

GitLab

github.com/oss-review-toolkit/ort-ci-gitlab

Jenkins

github.com/oss-review-toolkit/ort/tree/main/integrations/jenkins

TEKTON

github.com/oss-review-toolkit/ort/tree/main/integrations/tekton

Azure and BitBucket support under development / Azure 및 BitBucket 지원을 위한 개발 중

Why choose ORT? 왜 ORT를 선택해야 할까요?

- **Enables different risk/effort trade-offs to minimize end user compliance efforts whilst maintaining acceptable compliance levels**
  허용 가능 위험/투입 노력에 대한 절충을 통해 최종 사용자의 규정 준수 노력을 최소화하는 동시에 허용 가능한 Compliance 수준을 유지할 수 있습니다.

- **Enables your supply chain to check their software with your ORT's policy as code**
  정책을 공급업체와 공유하여 공급업체가 귀하의 정책에 따라 소프트웨어를 확인할 수 있도록 하여 공급망에 대한 신뢰도를 높일 수 있습니다.

- **Capable of producing high quality software bill of materials**
  고품질 SBoM 생성 가능

- **Created by the Open Source/OSPO community for the community**
  커뮤니티를 위해 오픈 소스/OSPO 커뮤니티에서 만든 Software

- **Highly modular and extensible solution**, adaptable to your (organization's) needs
  고도로 모듈화되고 확장 가능한 솔루션으로 조직의 요구 사항에 맞게 조정 가능

When should you not choose ORT?
언제 ORT를 선택하지 않아야 하나요?

- **Best effort and speed is more important than accuracy**
  정확성보다 속도와 적절한 수준의 비용이 더 중요합니다

- **Insufficient resources available to build solution with ORT**
  ORT로 솔루션을 구축하는 데 사용할 수 있는 리소스가 부족합니다

- **Not comfortable with how things are done within the ORT project**
  ORT 프로젝트 내에서 작업이 수행되는 방식이 불편합니다

- **You need a solution to scan Docker or operating systems**
  Docker 또는 운영 체제 스캔을 위한 솔루션이 필요합니다

Who is using ORT? 누가 ORT를 사용하고 있나요?

ORT has a vibrant community of people/organizations (mostly automotive OSPOs) using and contributing. ORT users also participate in SPDX, OpenChain, OpenSSF and the TODO group.
ORT에는 사용하고 기여하는 활발한 사람/조직(주로 자동차 OSPO) 커뮤니티가 있습니다. ORT 사용자들은 SPDX, OpenChain, OpenSSF 및 TODO 그룹에도 참여합니다.

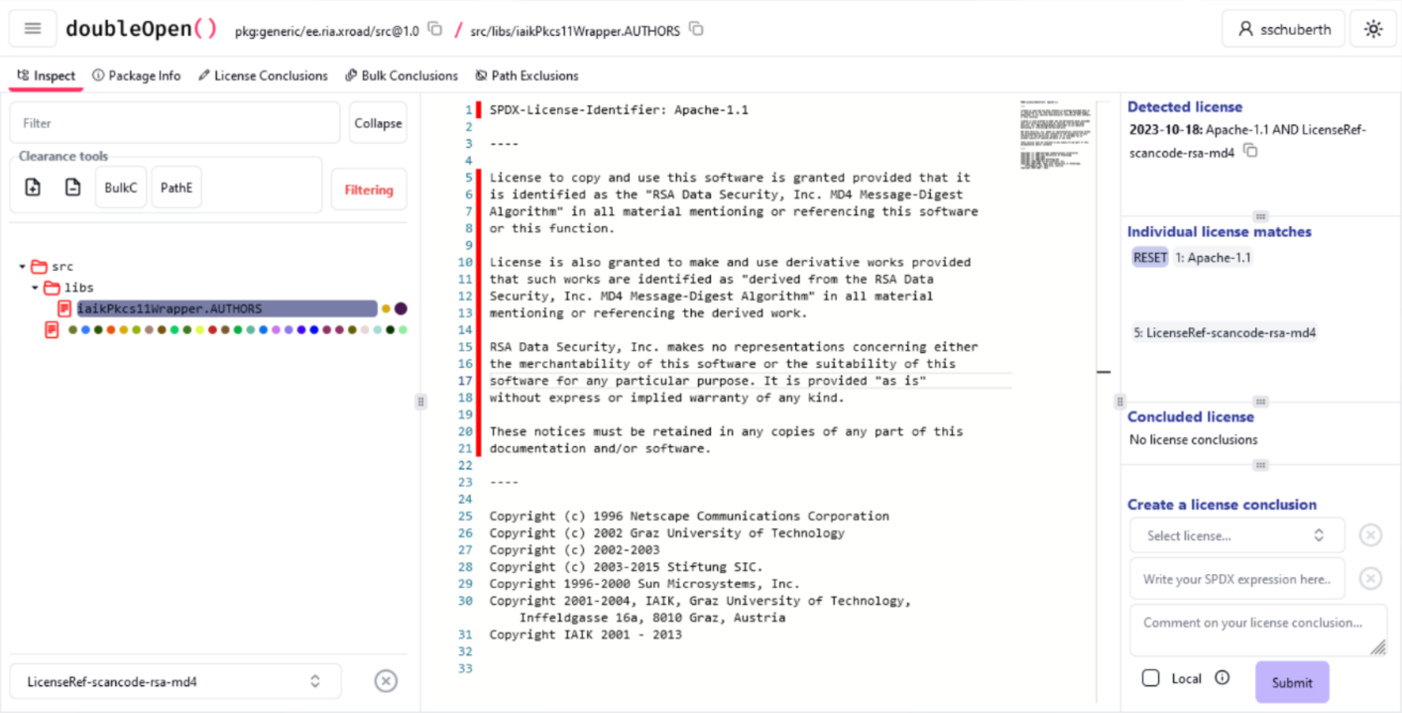
ORT is a battle-test solution is used in production within multiple large organizations from startup to large enterprise and hundreds of thousands ORT scans have been over the last 7 years.
ORT는 스타트업부터 대기업까지 여러 대규모 조직의 프로덕션에서 사용되는 검증된 솔루션이며 지난 7년 동안 수십만 개의 ORT 스캔이 이루어졌습니다.

Known adopters of ORT include Alliander, Bosch, Cariad, Deutsche Telekom, EPAM Systems, HERE, Porsche and Volkswagen.
ORT를 채택한 것으로 알려진 업체로는 Alliander, Bosch, Cariad, Deutsche Telekom, EPAM Systems, HERE, Porsche 및 Volkswagen이 있습니다.

See also / 참조하세요: github.com/oss-review-toolkit/ort/blob/main/ADOPTERS.md

Who is using ORT? 누가 ORT를 사용하고 있나요?

FOSS projects leveraging ORT: Double Open, Eclipse Apoapsis, Hermine and Oppossum
ORT를 활용하는 FOSS 프로젝트: 더블 오픈, 이클립스 아포앱시스, 헤르민, 오포섬



```
SPDX-License-Identifier: Apache-1.1

----

License to copy and use this software is granted provided that it
is identified as the "RSA Data Security, Inc. MD4 Message-Digest
Algorithm" in all material mentioning or referencing this software
or this function.

License is also granted to make and use derivative works provided
that such works are identified as "derived from the RSA Data
Security, Inc. MD4 Message-Digest Algorithm" in all material
mentioning or referencing the derived work.

RSA Data Security, Inc. makes no representations concerning either
the merchantability of this software or the suitability of this
software for any particular purpose. It is provided "as is"
without express or implied warranty of any kind.

These notices must be retained in any copies of any part of this
documentation and/or software.

----

Copyright (c) 1996 Netscape Communications Corporation
Copyright (c) 2002 Graz University of Technology
Copyright (c) 2002-2003
Copyright (c) 2003-2015 Stiftung SIC.
Copyright 1996-2000 Sun Microsystems, Inc.
Copyright 2001-2004, IAIK, Graz University of Technology,
    Inffeldgasse 16a, 8010 Graz, Austria
Copyright IAIK 2001 - 2013
```

LicenseRef-scancode-rsa-md4

Detected license

2023-10-18: Apache-1.1 AND LicenseRef-scancode-rsa-md4

Individual license matches

RESET 1: Apache-1.1

5: LicenseRef-scancode-rsa-md4

Concluded license

No license conclusions

Create a license conclusion

Select license...

Write your SPDX expression here..

Comment on your license conclusion...

☐ Local

Submit

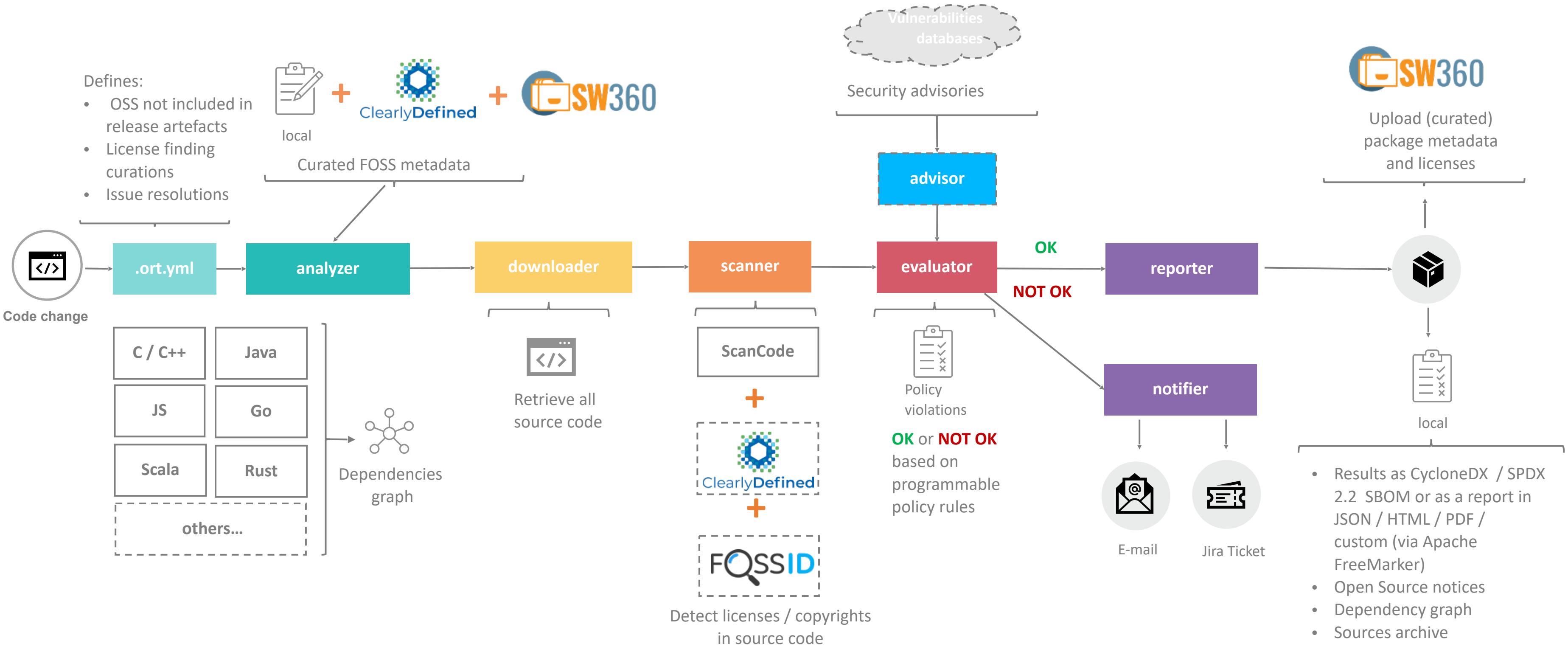
Building an open source management solution with ORT

ORT로 오픈소스 관리 솔루션 구축

Photo by Louise Viallesoubranne on Unsplash

Scan

Combine tools and develop the policy rules to create scalable processes that works for your organization
도구를 결합하고 정책 규칙을 개발하여 조직에 적합한 확장 가능한 프로세스를 만드세요.



Defines:

- OSS not included in release artefacts
- License finding curations
- Issue resolutions

local + ClearlyDefined + SW360

Curated FOSS metadata

Vulnerabilities databases

Security advisories

advisor

SW360

Upload (curated) package metadata and licenses

Code change → .ort.yml → analyzer → downloader → scanner → evaluator

OK → reporter

NOT OK → notifier

C / C++	Java
JS	Go
Scala	Rust
others...	

Dependencies graph

Retrieve all source code

ScanCode + ClearlyDefined + FOSSID

Detect licenses / copyrights in source code

Policy violations

OK or NOT OK based on programmable policy rules

E-mail

Jira Ticket

local

- Results as CycloneDX / SPDX 2.2 SBOM or as a report in JSON / HTML / PDF / custom (via Apache FreeMarker)
- Open Source notices
- Dependency graph
- Sources archive

OK /**NOT OK** = Code? + Licenses? + Product? + Security? + InnerSource? + Contracts? + Community? + ...

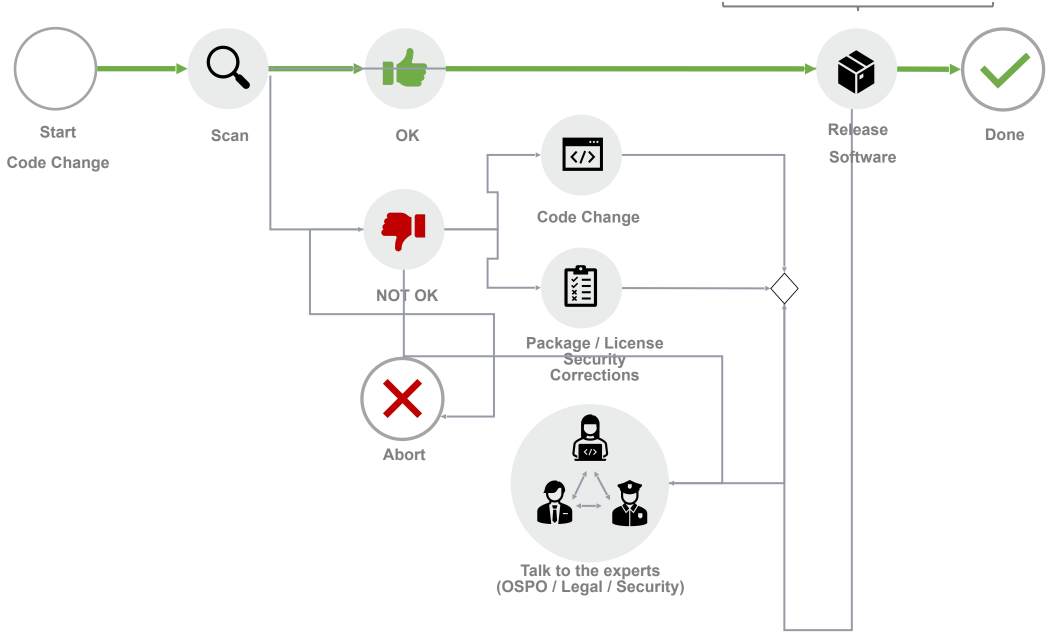
FOSS Review Process integrated in GitHub/GitLab/Jenkins CI (continuous)

GitHub/GitLab/Jenkins CI에 통합된 FOSS 리뷰 프로세스 (연속)

➡ Optimal Path

Compliance Artifacts

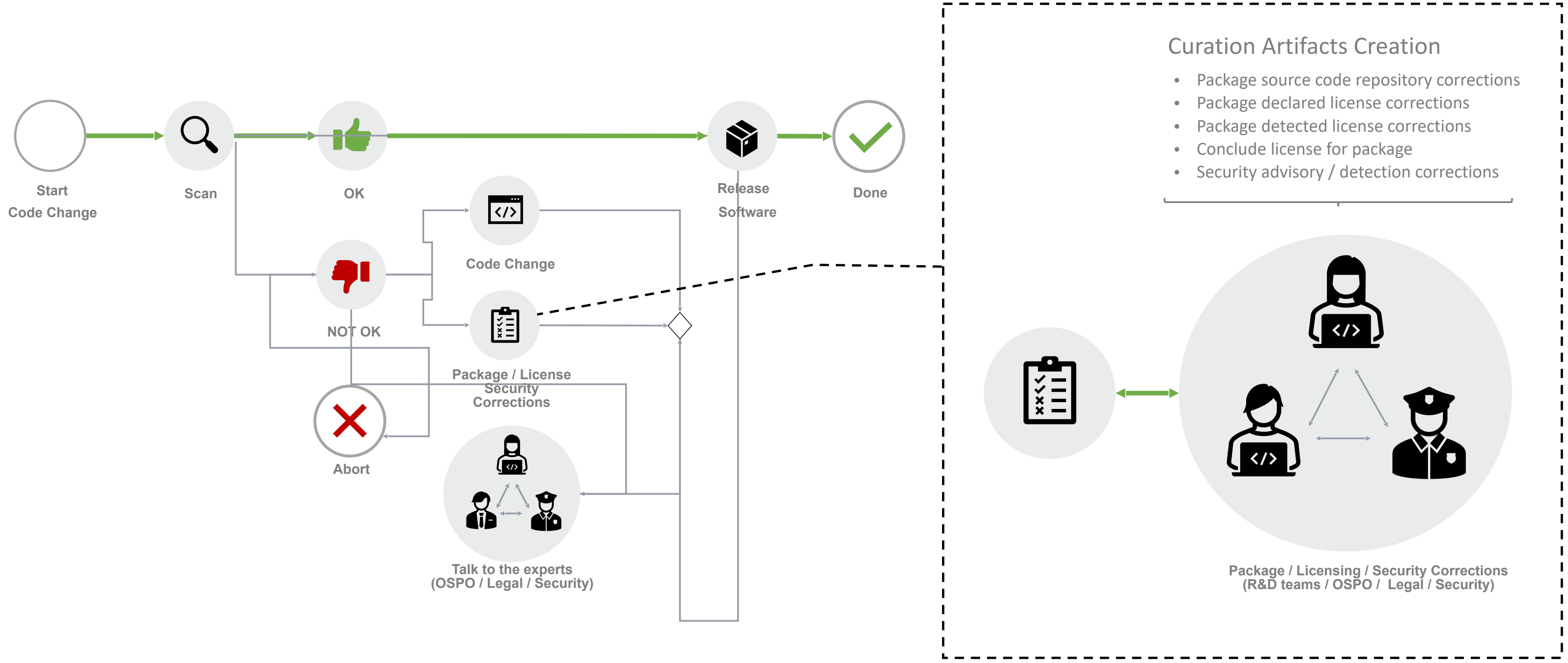
- SBOMs (CycloneDX / SPDX))
- Open Source notices
- Sources archive



REFERENCE PROCESS

Scaling FOSS Data Correction Process via InnerSource

InnerSource를 통한 FOSS 데이터 수정 프로세스 용량 확장



Curation Artifacts Creation

- Package source code repository corrections
- Package declared license corrections
- Package detected license corrections
- Conclude license for package
- Security advisory / detection corrections

REFERENCE PROCESS

Different ORT workflows? 다른 ORT 워크플로?

Organization / 단체	Workflow/ 워크플로
HERE	Code Change/Release => ORT GitLab job runs in R&D team repository makes API call to ORT Central GitLab pipeline maintained by OSPO => ORT Docker image runs => ORT results uploaded to Artifactory => R&D's ORT GitLab job gets copy of results plus if run is for merge request, comment is added with links to results 코드 변경/출시 => ORT GitLab job은 R&D 팀 코드 Repository에서 실행됩니다 => R&D 팀 Repository에서 ORT GitLab job이 실행되어 OSPO에서 유지 관리하는 ORT Central GitLab 파이프라인에 대한 API 호출이 이루어집니다 => ORT Docker 이미지 실행 => Artifactory에 업로드된 ORT 결과 => R&D의 ORT 'GitLab job'은 결과 사본을 얻고 병합 요청을 위한 실행인 경우 결과에 대한 링크와 함께 주석이 추가됩니다 See also / 참조하세요: tinyurl.com/here-2021-how-we-do-compliance
Bosch	ORT GitHub action runs in R&D team repository or R&D team uses Self-Service UI which makes REST API call to Eclipse Apoapsis instance => Orchestrator manages Worker job executing ORT in Kubernetes => ORT results made available to R&D team via Self-Service UI, API or email ORT GitHub action을 R&D 팀 Repository에서 실행하거나 Eclipse Apoapsis 인스턴스에 대한 REST API 호출을 만드는 셀프 서비스 UI를 사용합니다 => 오케스트레이터는 Kubernetes에서 ORT를 실행하는 작업자 작업을 관리합니다 => SelfService UI, API 또는 이메일을 통해 R&D 팀에 ORT 결과 제공됨 See also / 참조하세요: tinyurl.com/fosdem-2024-apoapsis

Different ORT workflows? 다른 ORT 워크플로?

Organization / 단체	**Workflow/ 워크플로**
Zeiss	Code Change / Release => CMake generates package list YAML file and executes ORTH to generate analyzer result => ORT executed by CMake => ORT results included in build artifacts 코드 변경/출시 => CMake는 패키지 목록 YAML 파일을 생성하고 ORTH를 실행하여 분석기 결과를 생성합니다 => CMake는 ORT 스캐너, 평가자 및 보고자를 실행합니다 => 빌드 아티팩트에 ORT 결과가 포함됨 See also / 참조하세요: tinyurl.com/ort-cd-2024-epam-zeiss
"Unnamed"	Code Change/Release => ORT Analyzer runs in R&D team's own CI/CD of R&D => uploads afterwards the result to central Jenkins instance => ORT Docker image runs => ORT results uploaded to SW360 and DependencyTrack 코드 변경/출시 => ORT 분석기는 R&D 팀 자체 CI/CD에서 실행됩니다. => 이후 중앙 Jenkins 인스턴스에 결과를 업로드합니다. => ORT Docker 이미지 실행 => SW360 및 DependencyTrack에 업로드된 ORT 결과

What's coming soon?

- **Rewrite of project documentation** to be more clear and concise
  프로젝트 문서를 보다 명확하고 간결하게 다시 작성할것입니다.

- **Improvements to copyright scanning and findings corrections**
  저작권 스캔 및 스캔 결과 보완 기능을 개선할것입니다.

- **Expanding ORT config repository contain example policies and FOSS finding corrections**
  예제 정책 및 FOSS 찾기 수정 사항이 포함되는 ORT config 리포지토리를 확장할 것입니다.니다.

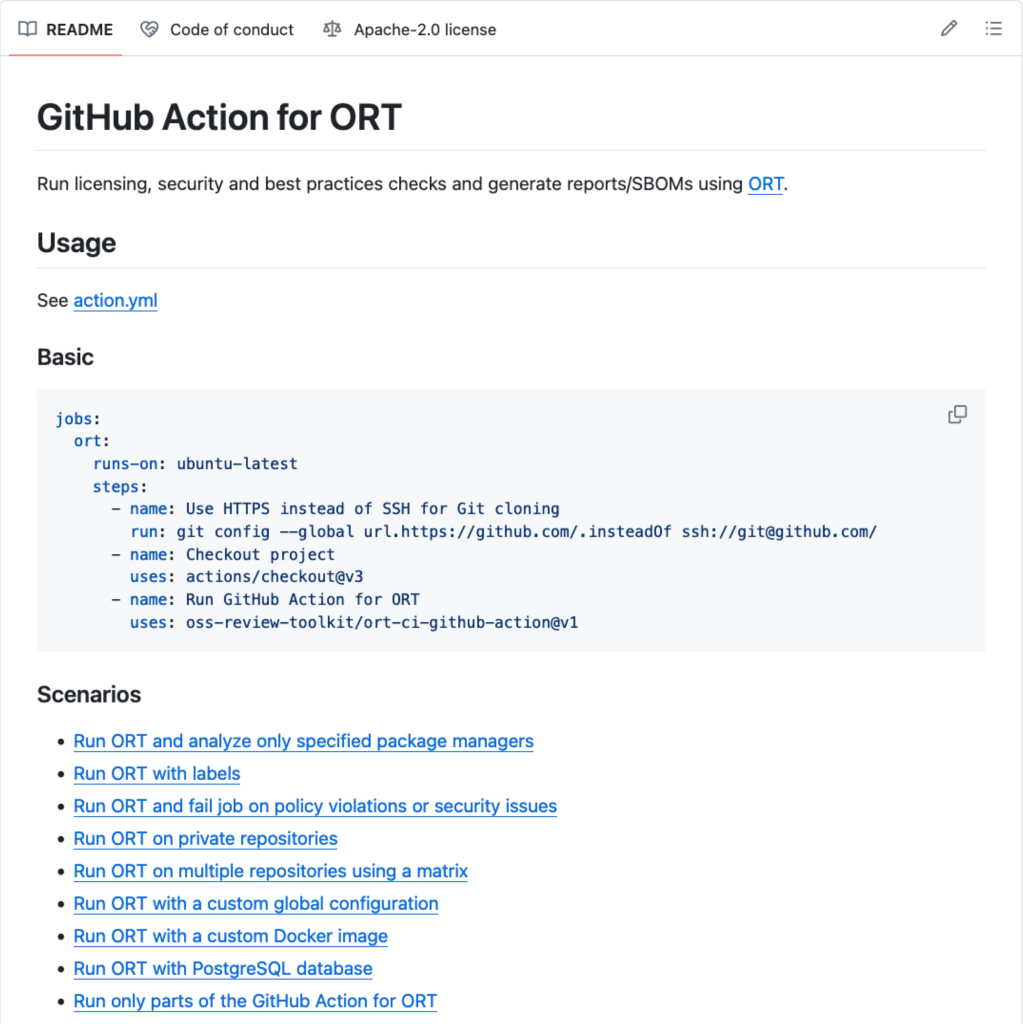
See also / 참조하세요: github.com/oss-review-toolkit/ort-config

OSS
Review Toolkit

Demo / 도구 데모

Photo by John Schnobrich on Unsplash

Available at / 다음에서 이용 가능:
github.com/oss-review-toolkit/ort-ci-github-action



GitHub Action for ORT

Run licensing, security and best practices checks and generate reports/SBOMs using ORT.

Usage

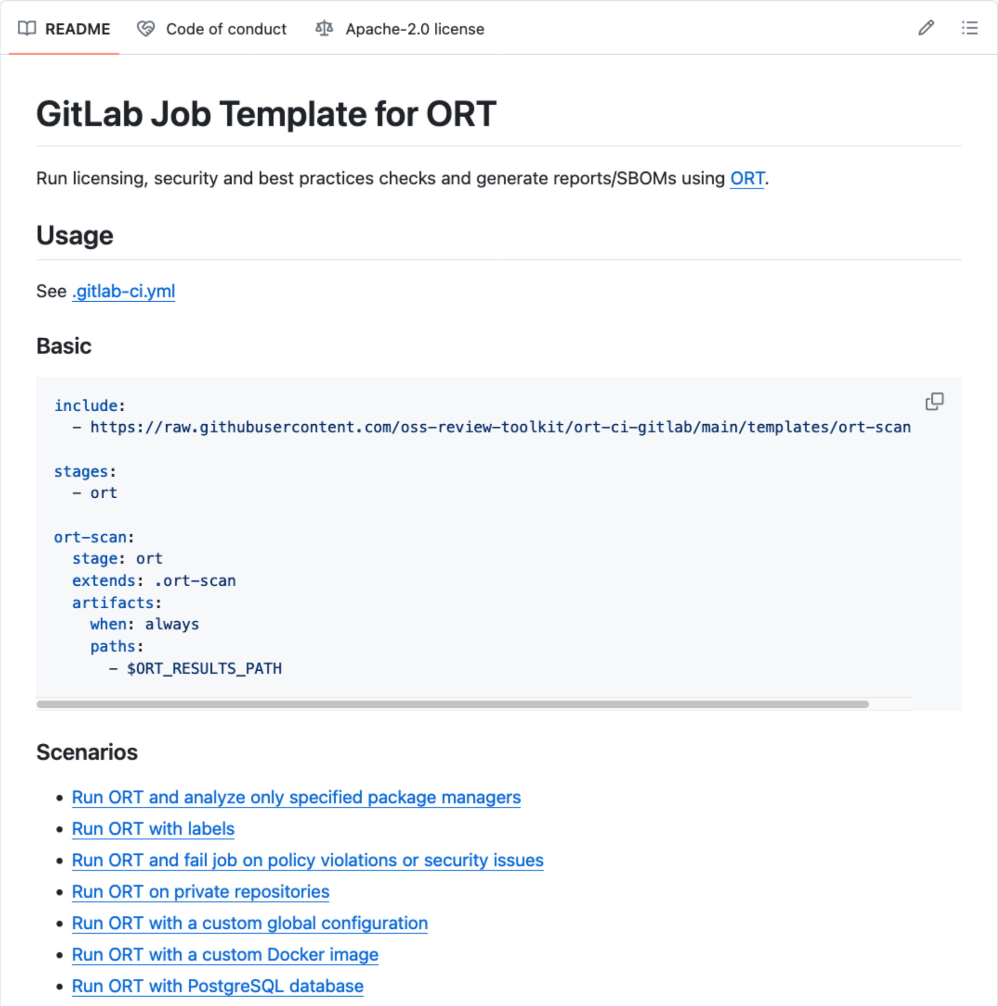
See action.yml

Basic

```
jobs:
  ort:
    runs-on: ubuntu-latest
    steps:
      - name: Use HTTPS instead of SSH for Git cloning
        run: git config --global url.https://github.com/.insteadOf ssh://git@github.com/
      - name: Checkout project
        uses: actions/checkout@v3
      - name: Run GitHub Action for ORT
        uses: oss-review-toolkit/ort-ci-github-action@v1
```

Scenarios

- Run ORT and analyze only specified package managers
- Run ORT with labels
- Run ORT and fail job on policy violations or security issues
- Run ORT on private repositories
- Run ORT on multiple repositories using a matrix
- Run ORT with a custom global configuration
- Run ORT with a custom Docker image
- Run ORT with PostgreSQL database
- Run only parts of the GitHub Action for ORT



GitLab Job Template for ORT

Run licensing, security and best practices checks and generate reports/SBOMs using ORT.

Usage

See .gitlab-ci.yml

Basic

```
include:
  - https://raw.githubusercontent.com/oss-review-toolkit/ort-ci-gitlab/main/templates/ort-scan

stages:
  - ort

ort-scan:
  stage: ort
  extends: .ort-scan
  artifacts:
    when: always
    paths:
      - $ORT_RESULTS_PATH
```

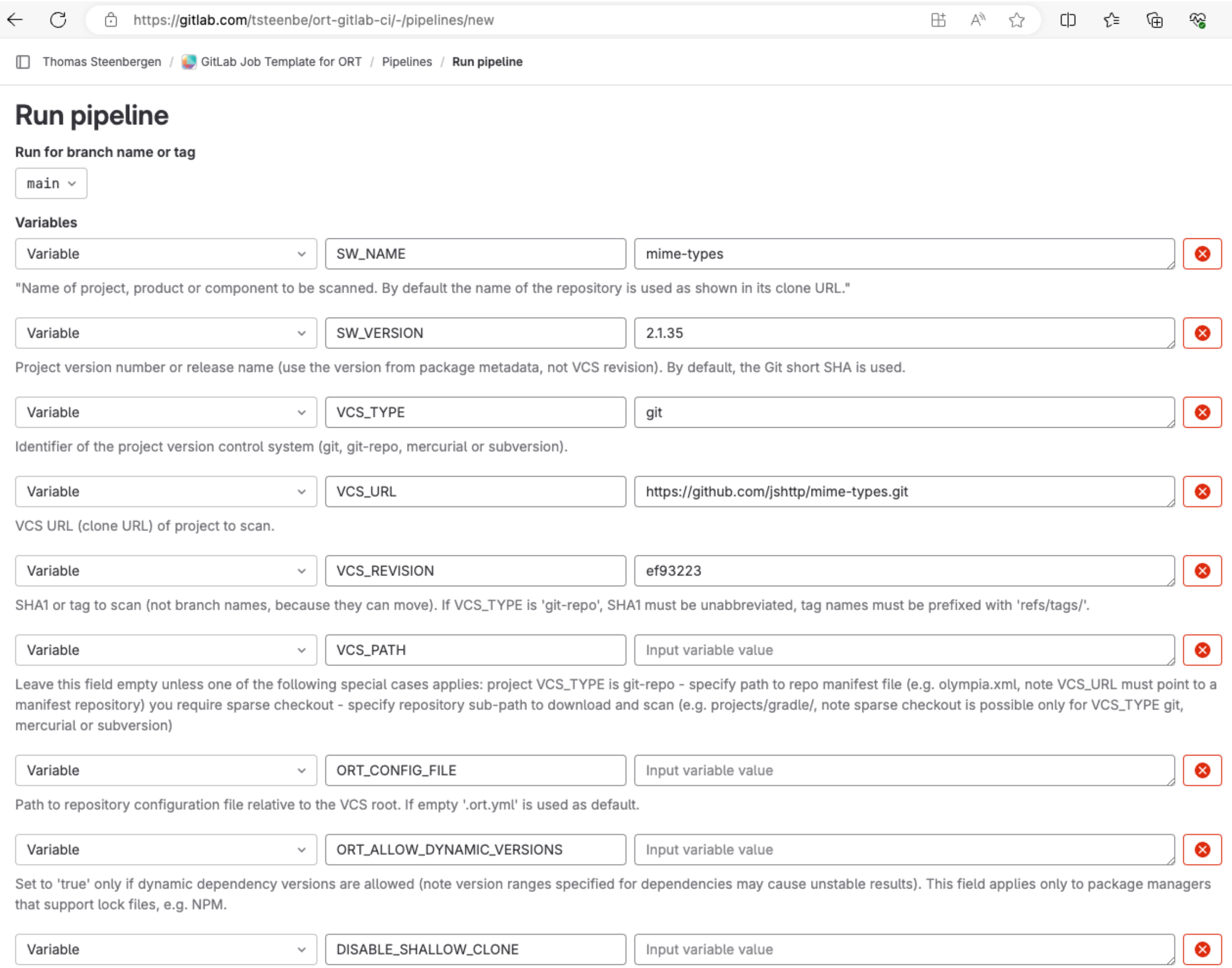
Scenarios

- Run ORT and analyze only specified package managers
- Run ORT with labels
- Run ORT and fail job on policy violations or security issues
- Run ORT on private repositories
- Run ORT with a custom global configuration
- Run ORT with a custom Docker image
- Run ORT with PostgreSQL database

Available at / 다음에서 이용 가능:
github.com/oss-review-toolkit/ort-ci-gitlab

Scan any public code repository

공개 코드 저장소를 스캔하세요


https://gitlab.com/tsteenbe/ort-gitlab-ci/-/pipelines/new
Thomas Steenbergen / GitLab Job Template for ORT / Pipelines / Run pipeline

Run pipeline

Run for branch name or tag

main

Variables

Variable | SW_NAME | mime-types

"Name of project, product or component to be scanned. By default the name of the repository is used as shown in its clone URL."

Variable | SW_VERSION | 2.1.35

Project version number or release name (use the version from package metadata, not VCS revision). By default, the Git short SHA is used.

Variable | VCS_TYPE | git

Identifier of the project version control system (git, git-repo, mercurial or subversion).

Variable | VCS_URL | https://github.com/jshttp/mime-types.git

VCS URL (clone URL) of project to scan.

Variable | VCS_REVISION | ef93223

SHA1 or tag to scan (not branch names, because they can move). If VCS_TYPE is 'git-repo', SHA1 must be unabbreviated, tag names must be prefixed with 'refs/tags/'.

Variable | VCS_PATH | Input variable value

Leave this field empty unless one of the following special cases applies: project VCS_TYPE is git-repo - specify path to repo manifest file (e.g. olympia.xml, note VCS_URL must point to a manifest repository) you require sparse checkout - specify repository sub-path to download and scan (e.g. projects/gradle/, note sparse checkout is possible only for VCS_TYPE git, mercurial or subversion)

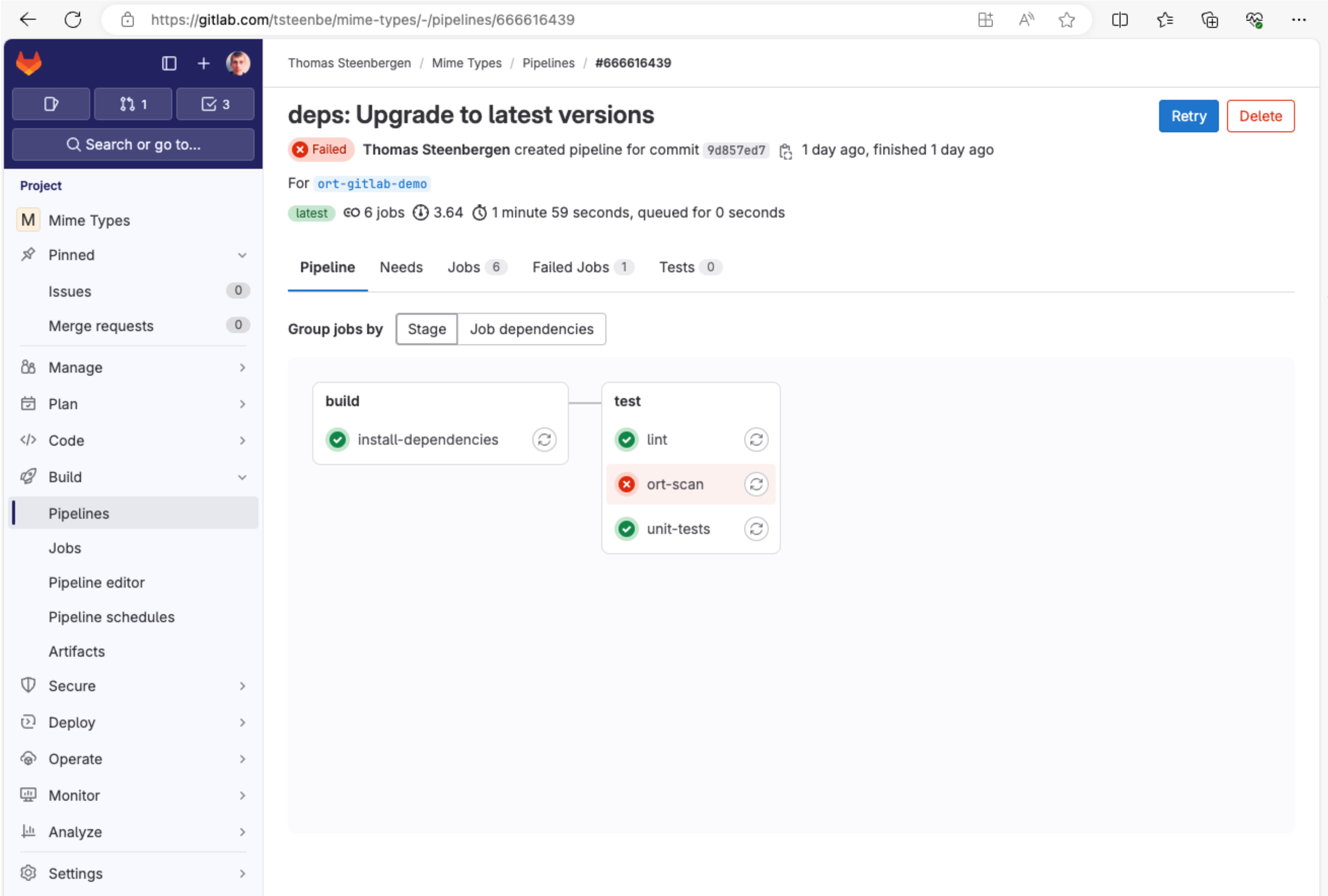
Variable | ORT_CONFIG_FILE | Input variable value

Path to repository configuration file relative to the VCS root. If empty '.ort.yml' is used as default.

Variable | ORT_ALLOW_DYNAMIC_VERSIONS | Input variable value

Set to 'true' only if dynamic dependency versions are allowed (note version ranges specified for dependencies may cause unstable results). This field applies only to package managers that support lock files, e.g. NPM.

Variable | DISABLE_SHALLOW_CLONE | Input variable value



deps: Upgrade to latest versions

Failed Thomas Steenbergen created pipeline for commit `9d857ed7` 1 day ago, finished 1 day ago

For `ort-gitlab-demo`

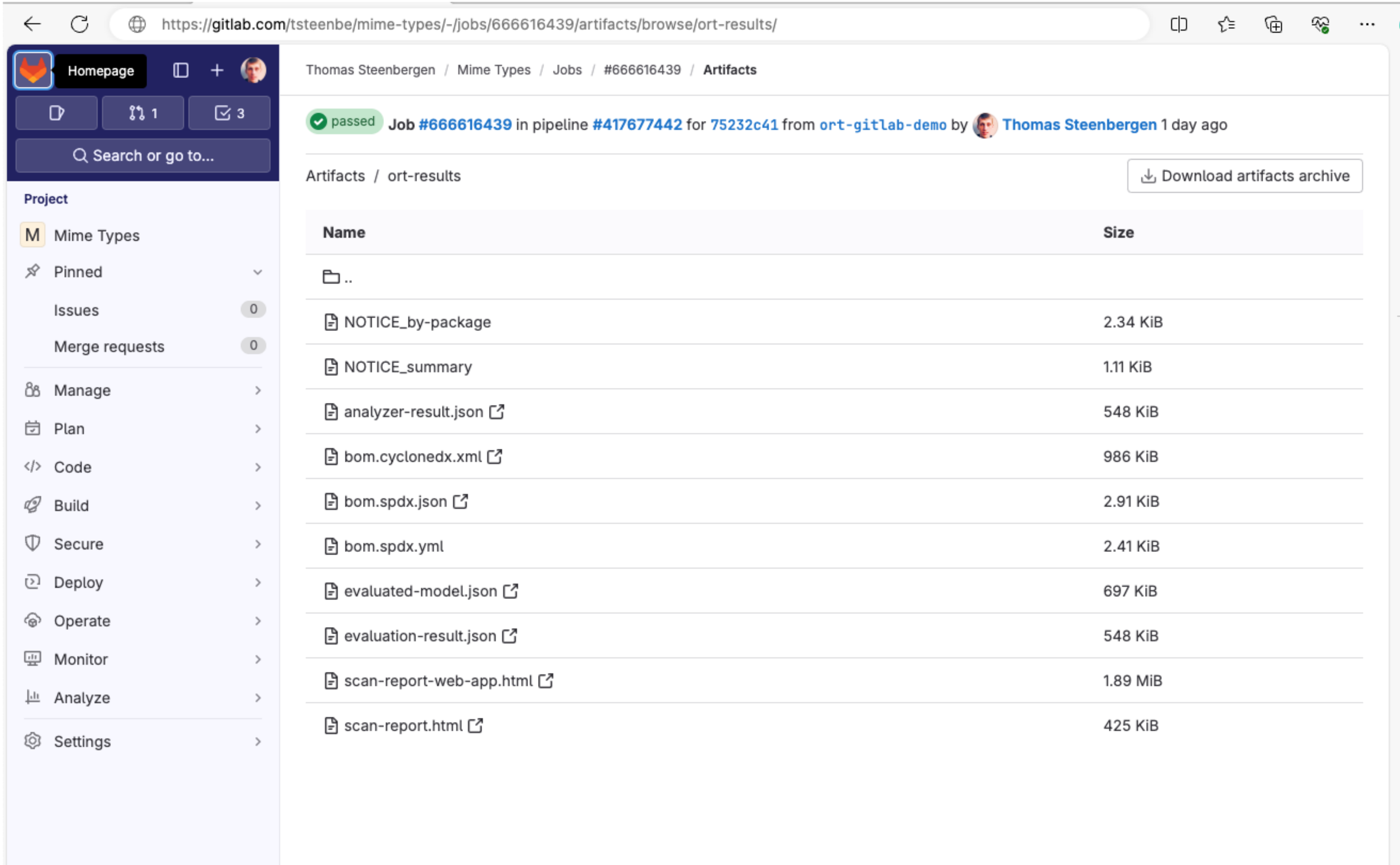
latest 6 jobs 3.64 1 minute 59 seconds, queued for 0 seconds

… or run ORT during code build

또는 코드 빌드 중에 ORT를 실행하세요

ORT results as CI/CD build artifacts

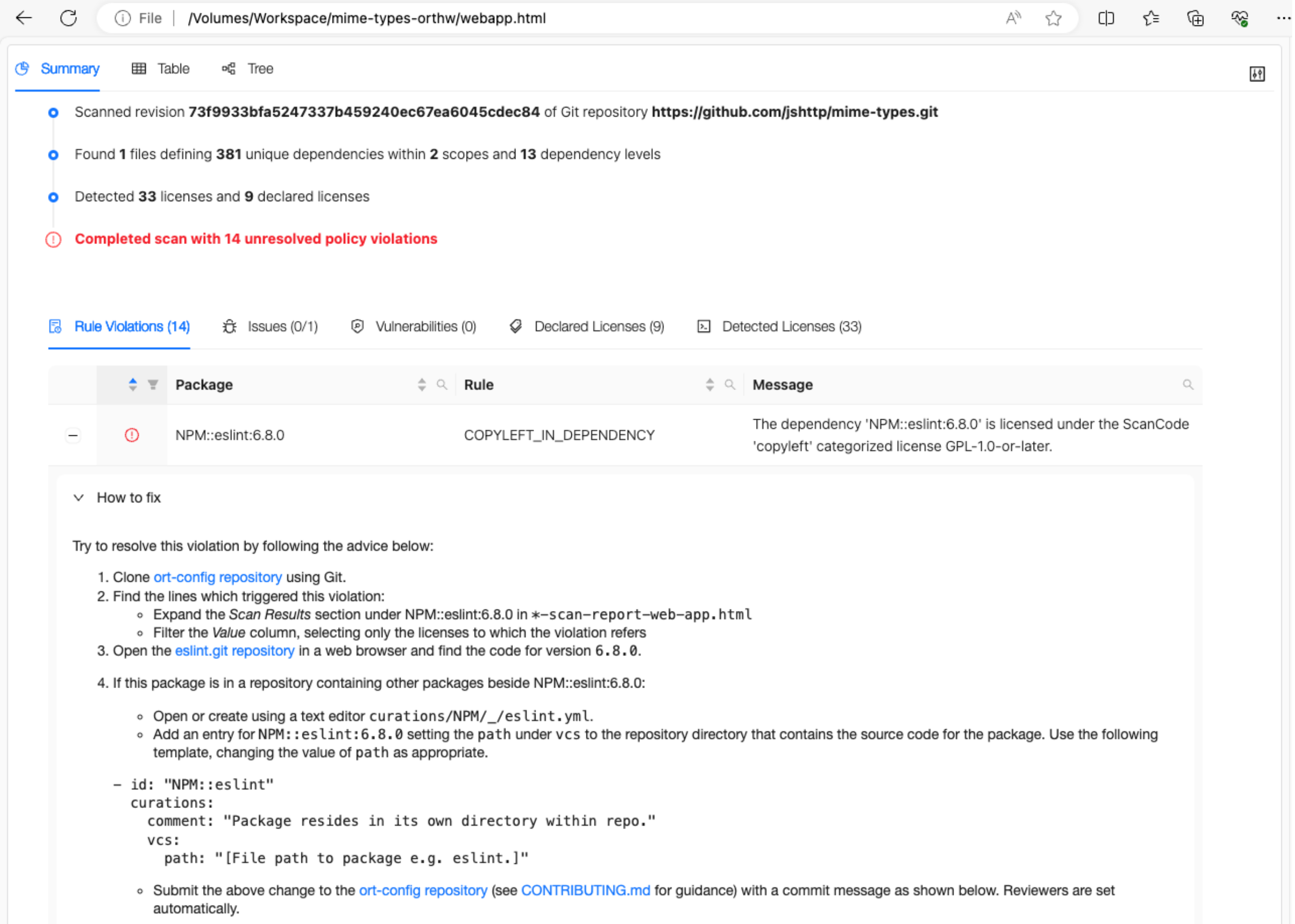
CI/CD 빌드 아티팩트로서의 ORT 결과


https://gitlab.com/tsteenbe/mime-types/-/jobs/666616439/artifacts/browse/ort-results/
Homepage
1
3
Search or go to...
Project
Mime Types
Pinned
Issues 0
Merge requests 0
Manage
Plan
Code
Build
Secure
Deploy
Operate
Monitor
Analyze
Settings
Thomas Steenbergen / Mime Types / Jobs / #666616439 / Artifacts
Download artifacts archive

passed Job #666616439 in pipeline #417677442 for 75232c41 from ort-gitlab-demo by Thomas Steenbergen 1 day ago

Artifacts / ort-results

Name	Size
..	
NOTICE_by-package	2.34 KiB
NOTICE_summary	1.11 KiB
analyzer-result.json	548 KiB
bom.cyclonedx.xml	986 KiB
bom.spdx.json	2.91 KiB
bom.spdx.yml	2.41 KiB
evaluated-model.json	697 KiB
evaluation-result.json	548 KiB
scan-report-web-app.html	1.89 MiB
scan-report.html	425 KiB



Scanned revision **73f9933bfa5247337b459240ec67ea6045cdec84** of Git repository **https://github.com/jshttp/mime-types.git**

Found **1** files defining **381** unique dependencies within **2** scopes and **13** dependency levels

Detected **33** licenses and **9** declared licenses

Completed scan with 14 unresolved policy violations

Package	Rule	Message
NPM::eslint:6.8.0	COPYLEFT_IN_DEPENDENCY	The dependency 'NPM::eslint:6.8.0' is licensed under the ScanCode 'copyleft' categorized license GPL-1.0-or-later.

How to fix

Try to resolve this violation by following the advice below:

1. Clone ort-config repository using Git.
2. Find the lines which triggered this violation:
   - Expand the *Scan Results* section under NPM::eslint:6.8.0 in `*-scan-report-web-app.html`
   - Filter the *Value* column, selecting only the licenses to which the violation refers
3. Open the eslint.git repository in a web browser and find the code for version `6.8.0`.
4. If this package is in a repository containing other packages beside NPM::eslint:6.8.0:
   - Open or create using a text editor `curations/NPM/_/eslint.yml`.
   - Add an entry for `NPM::eslint:6.8.0` setting the `path` under `vcs` to the repository directory that contains the source code for the package. Use the following template, changing the value of `path` as appropriate.

```
- id: "NPM::eslint"
  curations:
    comment: "Package resides in its own directory within repo."
    vcs:
      path: "[File path to package e.g. eslint.]"
```

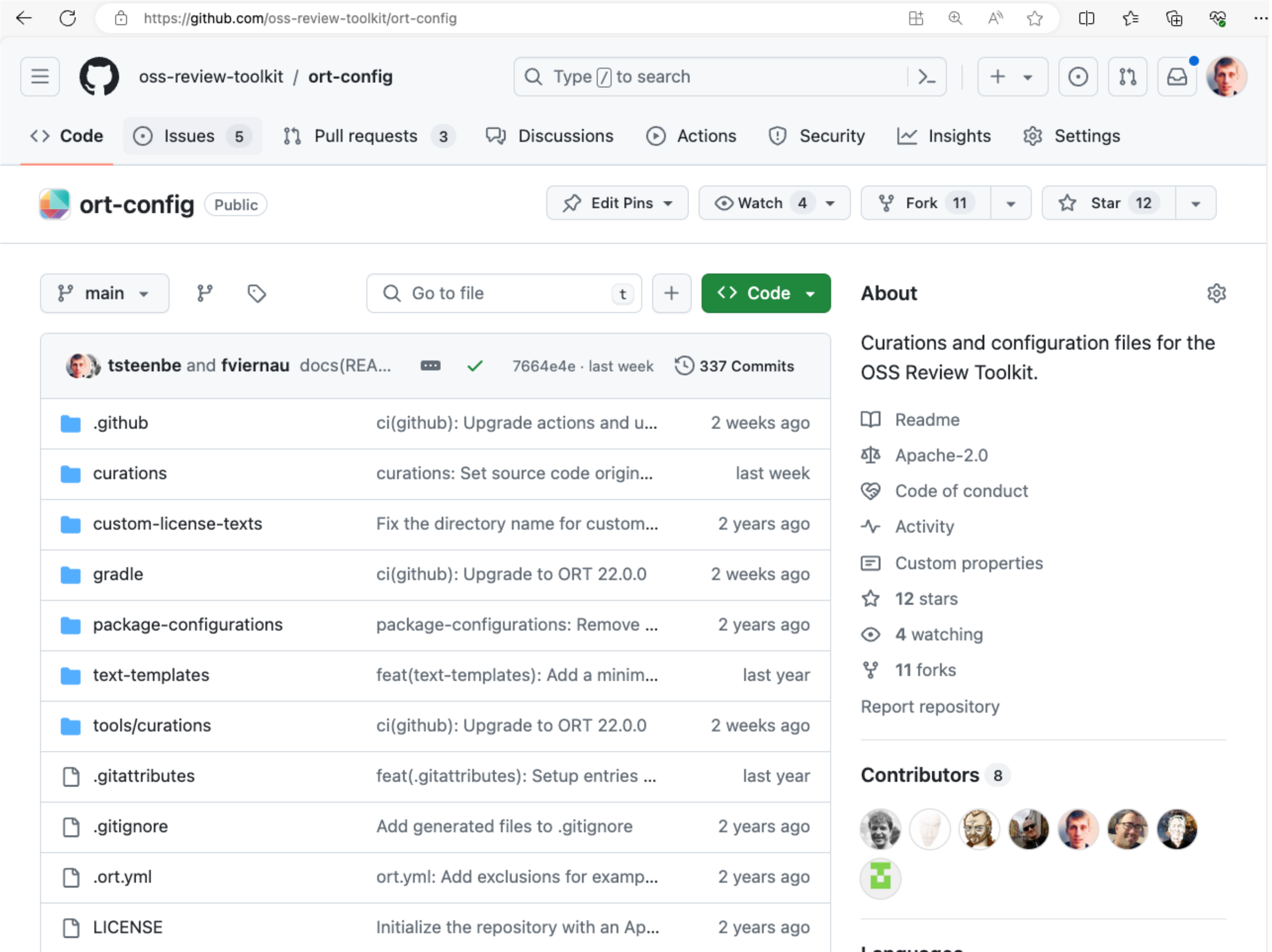
   - Submit the above change to the ort-config repository (see CONTRIBUTING.md for guidance) with a commit message as shown below. Reviewers are set automatically.

Enable R&D teams to try fix policy violations by themselves

R&D 팀이 스스로 정책 위반 문제를 해결할 수 있도록 지원

Share open source data corrections and policies across supply chains to increase trust

공급망 전반에 걸쳐 오픈 소스 데이터 수정 및 정책을 공유하여 신뢰도를 높입니다.



ort-config Public

tsteenbe and fviernau docs(REA... 7664e4e · last week 337 Commits

.github	ci(github): Upgrade actions and u...	2 weeks ago
curations	curations: Set source code origin...	last week
custom-license-texts	Fix the directory name for custom...	2 years ago
gradle	ci(github): Upgrade to ORT 22.0.0	2 weeks ago
package-configurations	package-configurations: Remove ...	2 years ago
text-templates	feat(text-templates): Add a minim...	last year
tools/curations	ci(github): Upgrade to ORT 22.0.0	2 weeks ago
.gitattributes	feat(.gitattributes): Setup entries ...	last year
.gitignore	Add generated files to .gitignore	2 years ago
.ort.yml	ort.yml: Add exclusions for examp...	2 years ago
LICENSE	Initialize the repository with an Ap...	2 years ago

About

Curations and configuration files for the OSS Review Toolkit.

- Readme
- Apache-2.0
- Code of conduct
- Activity
- Custom properties
- 12 stars
- 4 watching
- 11 forks

Report repository

Contributors 8

Questions? 질문이 있으신가요?

Interested in collaborating?
협업에 관심이 있으신가요?

Join the ORT Slack channel to chat with project maintainers and users
ORT Slack 채널에 참여하여 프로젝트 관리자 및 사용자와 채팅하세요
http://slack.oss-review-toolkit.org

Photo by Cytonn Photography on Unsplash

Photo by Matt Jones on Unsplash

들어주셔서 감사합니다

Interested to learn more? Email me
더 자세히 알고 싶으십니까? 나에게 이메일을 보내세요

Thomas Steenbergen
opensource@steenbe.nl
github.com/tsteenbe

OSS Review Toolkit
oss-review-toolkit.org