

# Introduction to **DOS**

**D**ouble **O**pen **S**erver for the



OSS  
**Review Toolkit**

`doubleOpen()` – We **DO** compliance

Sebastian Schuberth, CEO of Double Open, ORT Community Days 2024

# Brief History of Double Open

- Started in 2019 as a project by HH Partners
  - Automate OSS compliance checks
  - Increase scanning performance
  - UI driven clearance workflow
  - Reuse of results
- Active in the OpenChain community
- Successful piloting phase with selected customers
  - E.g. NIIS (Petteri Kivimäki)
- Double Open Oy founded as a startup company in 2023

# Double Open Server (DOS)

- Client
  - [Scanner plugin](#) for ORT
    - `PackageScannerWrapper` for a remote scanner
    - A bit similar to FOSSID, but uses `ScanCode` underneath
  - [Package configuration provider plugin](#) for ORT
- Server
  - Back-end
    - Scanner agent / worker (TypeScript)
    - Package configuration provider (TypeScript)
  - Front-end
    - Clearance UI (TypeScript / React / Next.js / shadcn)

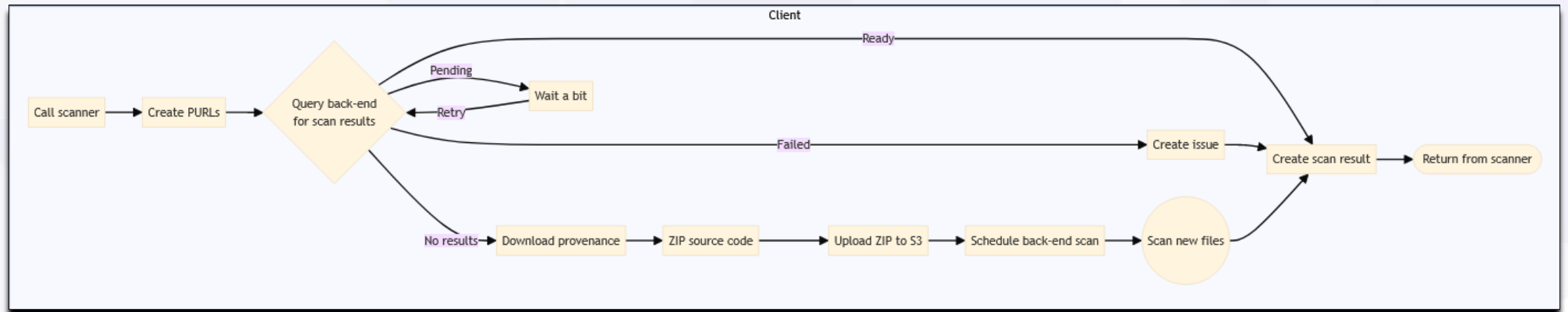
# DOS vs. ORT Server (Eclipse Apoapsis)

- DOS is a companion server to ORT
  - Needs ORT already set up and running
  - Provides APIs that integrate with ORT
  - Focuses on improving an ORT pipeline run end-to-end
- ORT server operates ORT itself
  - Replaces the CLI to run ORT
  - Scales ORT across multiple parallel pipeline runs

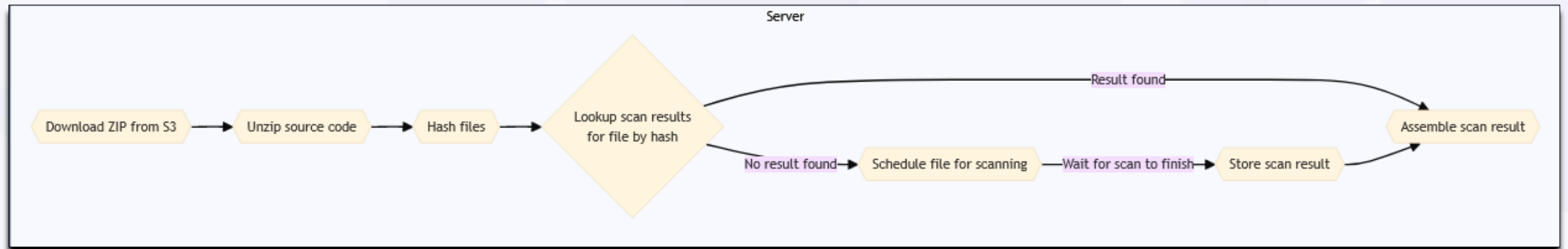
# DOS Scanner

- Inherent “scan storage”
  - Does not implement `ScanStorageReader` **or** `ScanStorageWriter`
- Reuse of scan results per file
  - Identified by SHA256
  - No integration with `FileListResolver` (yet)
- Blocking implementation

# DOS Scanner (client-side workflow)

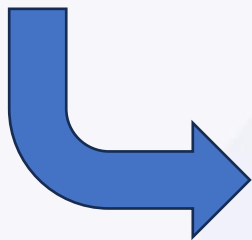
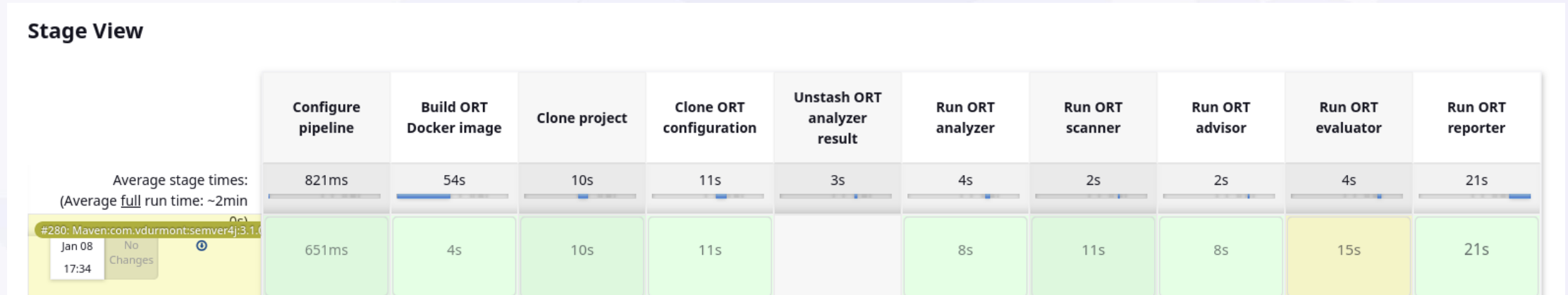


# DOS Scanner (server-side workflow)



# Clearance UI (pilot)

## Jenkins pipeline integration:



Last Successful Artifacts

adviser-result.yml	1.61 MB	view
analyzer-result.yml	15.51 KB	view
evaluation-result.yml	1.61 MB	view
AsciiDoc_defect_report.pdf	54.63 KB	view
AsciiDoc_disclosure_document.pdf	98.15 KB	view
AsciiDoc_vulnerability_report.pdf	67.33 KB	view
bom.cyclonedx.xml	7.89 KB	view
bom.spdx.yml	14.57 KB	view
NOTICE_DEFAULT	3.10 KB	view
scan-report-web-app.html	2.05 MB	view
scan-report.html	49.01 KB	view
scan-result.yml	1.61 MB	view

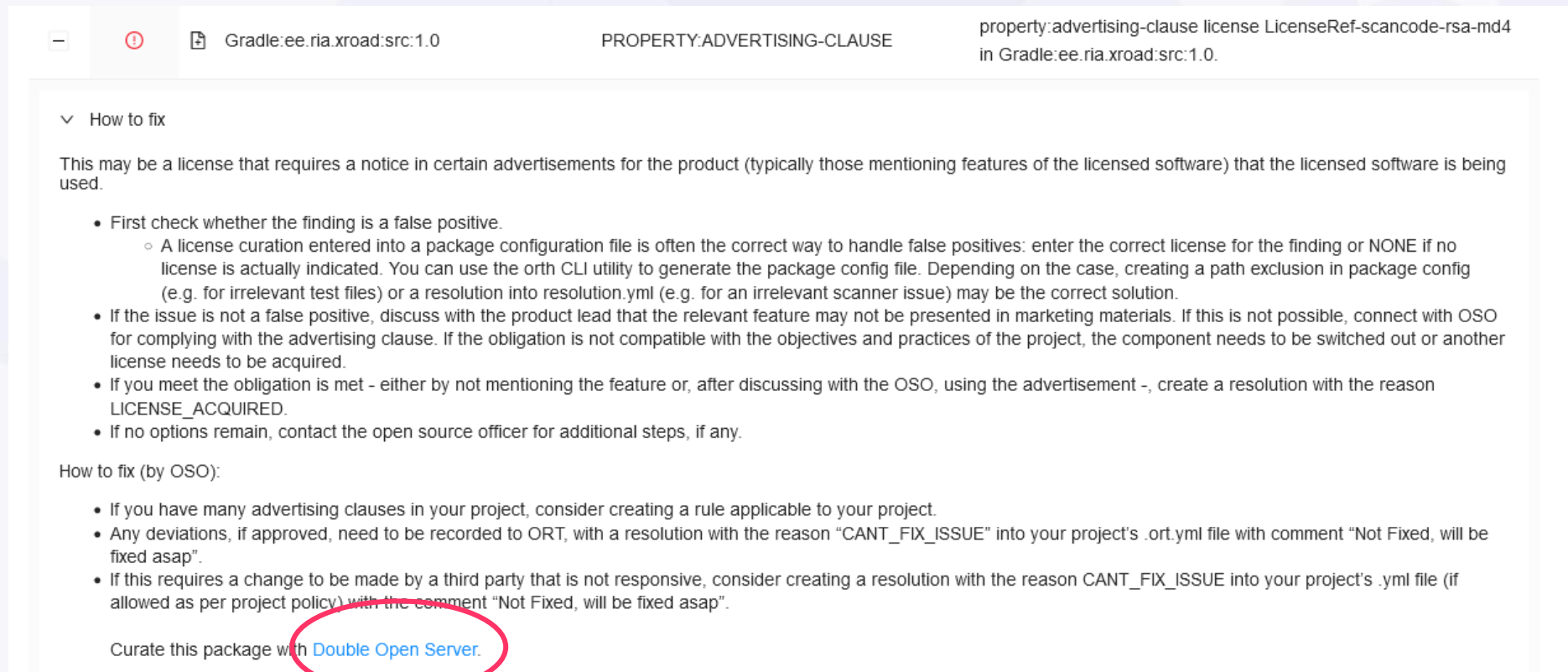


**OSS Review Toolkit**  
Almost ready to display scan report...  
[Progress bar with green checkmark]



# Clearance UI (pilot)

## Web-app report integration:



Gradle:ee.ria.xroad:src:1.0      PROPERTY:ADVERTISING-CLAUSE      property:advertising-clause license LicenseRef-scancode-rsa-md4 in Gradle:ee.ria.xroad:src:1.0.

▼ How to fix

This may be a license that requires a notice in certain advertisements for the product (typically those mentioning features of the licensed software) that the licensed software is being used.

- First check whether the finding is a false positive.
  - A license curation entered into a package configuration file is often the correct way to handle false positives: enter the correct license for the finding or NONE if no license is actually indicated. You can use the orth CLI utility to generate the package config file. Depending on the case, creating a path exclusion in package config (e.g. for irrelevant test files) or a resolution into resolution.yml (e.g. for an irrelevant scanner issue) may be the correct solution.
- If the issue is not a false positive, discuss with the product lead that the relevant feature may not be presented in marketing materials. If this is not possible, connect with OSO for complying with the advertising clause. If the obligation is not compatible with the objectives and practices of the project, the component needs to be switched out or another license needs to be acquired.
- If you meet the obligation is met - either by not mentioning the feature or, after discussing with the OSO, using the advertisement -, create a resolution with the reason LICENSE\_ACQUIRED.
- If no options remain, contact the open source officer for additional steps, if any.

How to fix (by OSO):

- If you have many advertising clauses in your project, consider creating a rule applicable to your project.
- Any deviations, if approved, need to be recorded to ORT, with a resolution with the reason "CANT\_FIX\_ISSUE" into your project's .ort.yml file with comment "Not Fixed, will be fixed asap".
- If this requires a change to be made by a third party that is not responsive, consider creating a resolution with the reason CANT\_FIX\_ISSUE into your project's .yml file (if allowed as per project policy) with the comment "Not Fixed, will be fixed asap".

Curate this package with [Double Open Server](#).

# Clearance UI (pilot)

The screenshot displays the 'doubleOpen()' interface for license clearance. The top navigation bar shows the package path: pkg:generic/ee.ria.xroad/src@1.0 / src/libs/iaikPkcs11Wrapper.AUTHORS. The main content area is divided into three panels:

- Left Panel:** Contains a 'Filter' input, 'Collapse' button, and 'Clearance tools' (BulkC, PathE, Filtering). A file tree shows the selected file 'iaikPkcs11Wrapper.AUTHORS' under 'src/libs'.
- Center Panel:** Displays the license text from the file, with line numbers 1 through 33. A red vertical bar highlights the license text starting from line 5.
- Right Panel:** Shows the results of the license scan:
  - Detected license:** 2023-10-18: Apache-1.1 AND LicenseRef-scancode-rsa-md4
  - Individual license matches:** RESET 1: Apache-1.1
  - Concluded license:** No license conclusions
  - Create a license conclusion:** Includes a 'Select license...' dropdown, a text input for 'Write your SPDX expression here.', and a 'Submit' button.

# Next Steps

- Migrate from Jenkins to [ORT Server](#)
  - [Keycloak](#)-based authentication and role management
  - Holistic web front-end UI
    - User and project management
    - Incorporate features from web-app report
    - Dashboards for technical people and management

# Questions & Answers

Thank you!  
Any questions?

Reach out to us

<https://www.doubleopen.org/>

Join the community

<https://oss-review-toolkit.org/>