Double Open

An automated open source compliance pipeline for Yocto built on SPDX

Mikko Murto

- Lawyer* at HH Partners, Attorneys-at-Law (https://www.hhpartners.fi/en/)
- Developer at Double Open project (https://www.doubleopen.org/)

- mikko.murto@hhpartners.fi
- https://github.com/mmurto

Background

- Part of Double Open project
  - Tooling
  - Competence Development
- Tooling project focuses on automating license compliance for Yocto builds (https://www.yoctoproject.org/)
- Currently working on a proof of concept with a company using Yocto
  - Support for other build systems possible

Goals

- Use SPDX Document as the data format throughout
- Analyze a software project
- Create a detailed (SPDX) bill of materials of the project
- Get license and copyright information from Fossology
- Evaluate the project against a license policy
- Create a notice file for the project
- Create a workflow to support this as a part of CI
- Store relevant data for vulnerability management

```
doubleopen_cli 0.1.0
HH Partners
Analyze software projects for their bill of materials, get license and copyright data for them,
evaluate license compliance and build notice files

USAGE:
    doubleopen_cli <SUBCOMMAND>

FLAGS:
    -h, --help       Prints help information
    -V, --version    Prints version information

SUBCOMMANDS:
    analyze      Analyze a Yocto project and save the bill of materials as an SPDX document
    evaluate     Evaluate the license compliance of an SPDX file with a provided policy
    fossology    Interact with Fossology
    help         Prints this message or the help of the given subcommand(s)
    notice       Create a notice file from an SPDX Document
```

https://www.doubleopen.org/

Analyzer

- Analyze a software project for its *Bill of Materials*
- Yocto as the target for the proof of concept
- Determine the packages that are packaged with the image
- Determine the source files that are used to build the packages
- Store data in an **SPDX document**

```
doubleopen_cli-analyze 0.1.0
HH Partners
Analyze a Yocto project and save the bill of materials as an SPDX document

USAGE:
    doubleopen_cli analyze --manifest <manifest> --build <build> --output <output>

FLAGS:
    -h, --help       Prints help information
    -V, --version    Prints version information

OPTIONS:
    -b, --build <build>          Build directory of the Yocto build. Default location at `build/`
    -m, --manifest <manifest>    Manifest file of the Yocto build. Default location at
                                 `build/tmp/deploy/images/<arch>/<image>.manifest`
    -o, --output <output>        Path to output the SPDX document to
```

https://www.doubleopen.org/

Fossology

- Upload source of packages used in the Yocto build to Fossology for license and copyright scanning
- Query the information from Fossology and store it in an SPDX document
- Retrieves data for individual files using Fossology's REST API

```
doubleopen_cli-fossology 0.1.0
HH Partners
Interact with Fossology

USAGE:
    doubleopen_cli fossology --uri <uri> --token <token> <SUBCOMMAND>

FLAGS:
    -h, --help       Prints help information
    -V, --version    Prints version information

OPTIONS:
    -t, --token <token>    Access token for the Fossology instance
    -u, --uri <uri>        URL of the Fossology instance to use. Example:
                           `http://localhost/repo/api/v1`

SUBCOMMANDS:
    help      Prints this message or the help of the given subcommand(s)
    query     Populate an SPDX file with license and copyritght information from Fossology
    upload    Upload the source for a Yocto build to Fossology
```

```
doubleopen_cli-fossology-upload
Upload the source for a Yocto build to Fossology

USAGE:
    doubleopen_cli fossology --uri <uri> --token <token> upload --manifest <manifest> --build <build> --folder <folder>

FLAGS:
    -h, --help       Prints help information
    -V, --version    Prints version information

OPTIONS:
    -b, --build <build>          Build directory of the Yocto build. Default location at `build/`
    -f, --folder <folder>        ID of the folder in Fossology to upload the source to
    -m, --manifest <manifest>    Manifest file of the Yocto build. Default location at
                                 `build/tmp/deploy/images/<arch>/<image>.manifest`
```

https://www.doubleopen.org/

```
doubleopen_cli-fossology-query
Populate an SPDX file with license and copyritght information from Fossology

USAGE:
    doubleopen_cli fossology --uri <uri> --token <token> query --input <input> --output <output>

FLAGS:
    -h, --help       Prints help information
    -V, --version    Prints version information

OPTIONS:
    -i, --input <input>      Path to the input SPDX
    -o, --output <output>    Path to output the populated SPDX document to
```

https://www.doubleopen.org/

Policy Engine

- Evaluates license compliance of the software described in an SPDX document against a license policy
- License policy is a combination of a global policy and a repository specific policy
- Produces a report on the compliance check

```
doubleopen_cli-evaluate 0.1.0
HH Partners
Evaluate the license compliance of an SPDX file with a provided policy

USAGE:
    doubleopen_cli evaluate [OPTIONS] --spdx <spdx> --context <context>

FLAGS:
    -h, --help       Prints help information
    -V, --version    Prints version information

OPTIONS:
    -c, --context <context>         The context from the policies to use
    -p, --policies <policies>...    List of policies to use in the evaluation. Latter policies take
                                    precedence over earlier ones
    -s, --spdx <spdx>               Path to the input SPDX
```

https://www.doubleopen.org/

Notice Generator

- Generate notice file from the data in the SPDX Document
- Templating with Handlebars (https://handlebarsjs.com/)

```
doubleopen_cli-notice 0.1.0
HH Partners
Create a notice file from an SPDX Document

USAGE:
    doubleopen_cli notice --input <input> --output <output> --template <template>

FLAGS:
    -h, --help       Prints help information
    -V, --version    Prints version information

OPTIONS:
    -i, --input <input>          Path to the input SPDX
    -o, --output <output>        Path to output the notice file to
    -t, --template <template>    Path to a template for the notice
```

https://www.doubleopen.org/

Project status

- In active development, the proof of concept is expected to be functional during the spring
- A lot of the functionality works, the pipeline from start to finish is not yet functional
- Still iterating on some details, e.g. the policy specification
- We’re looking for a second partner to implement the pipeline with as a proof of concept to gain feedback on the tooling
- https://github.com/doubleopen-project/doubleopen-cli

Thank you

mikko.murto@hhpartners.fi