



# OSS Lifecycle Management with SPDX

December 2022

Keiya Nobuta, Fujitsu



# Who am I?

- I'm an embedded engineer at Fujitsu.
  - My main job is...
    - Maintain in-house Linux distribution
    - Embedded software development support, especially device drivers for Linux/Zephyr.
- Other Activities:
  - OpenChain Japan workgroup member
  - AGL Instrument Cluster Expert Group member



Japan workgroup

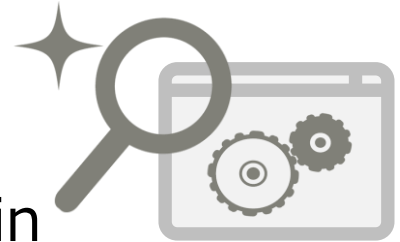


- SBOM Overview
- SPDX generation and Vulnerabilities detection by Yocto
- SBOM and Vulnerabilities management with SW360

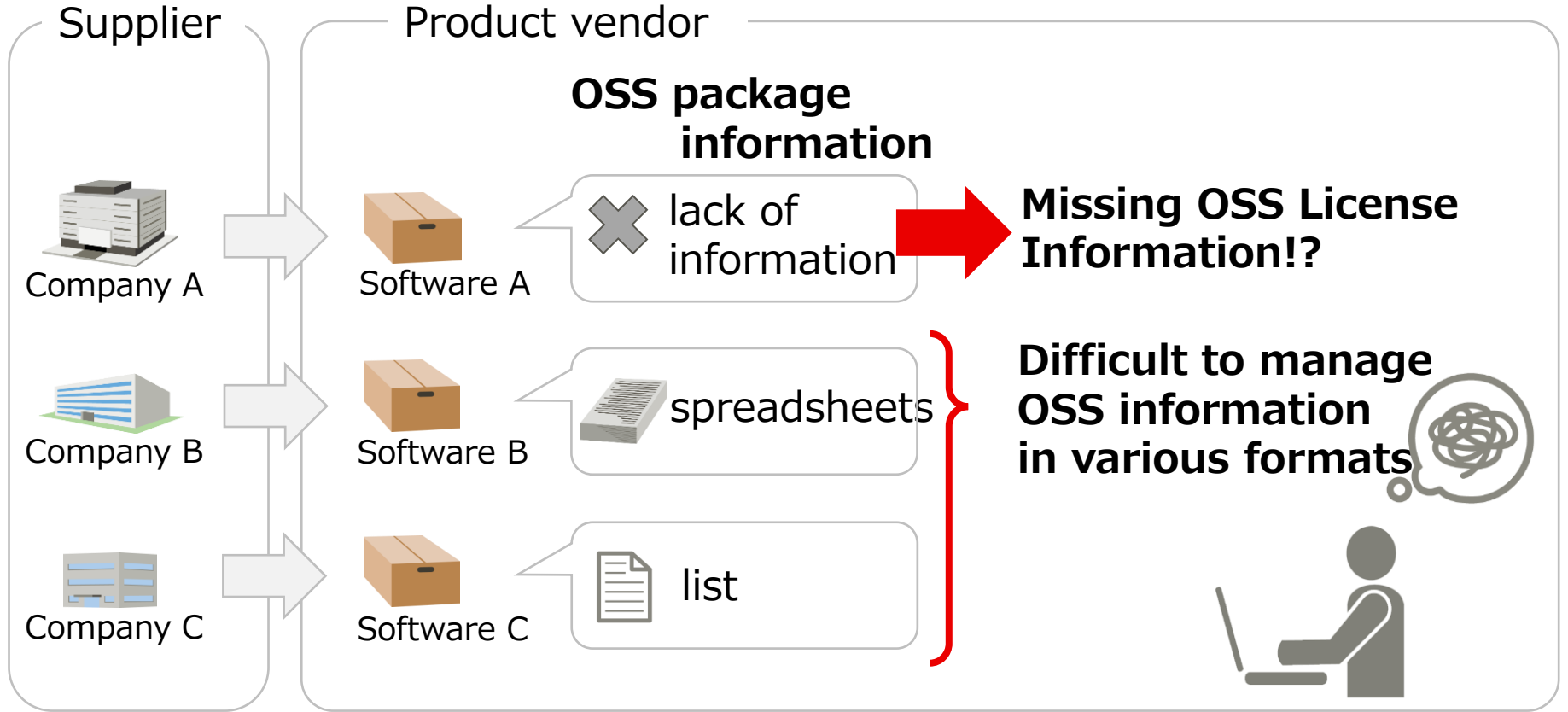
# SBOM Overview

# What is SBOM ?

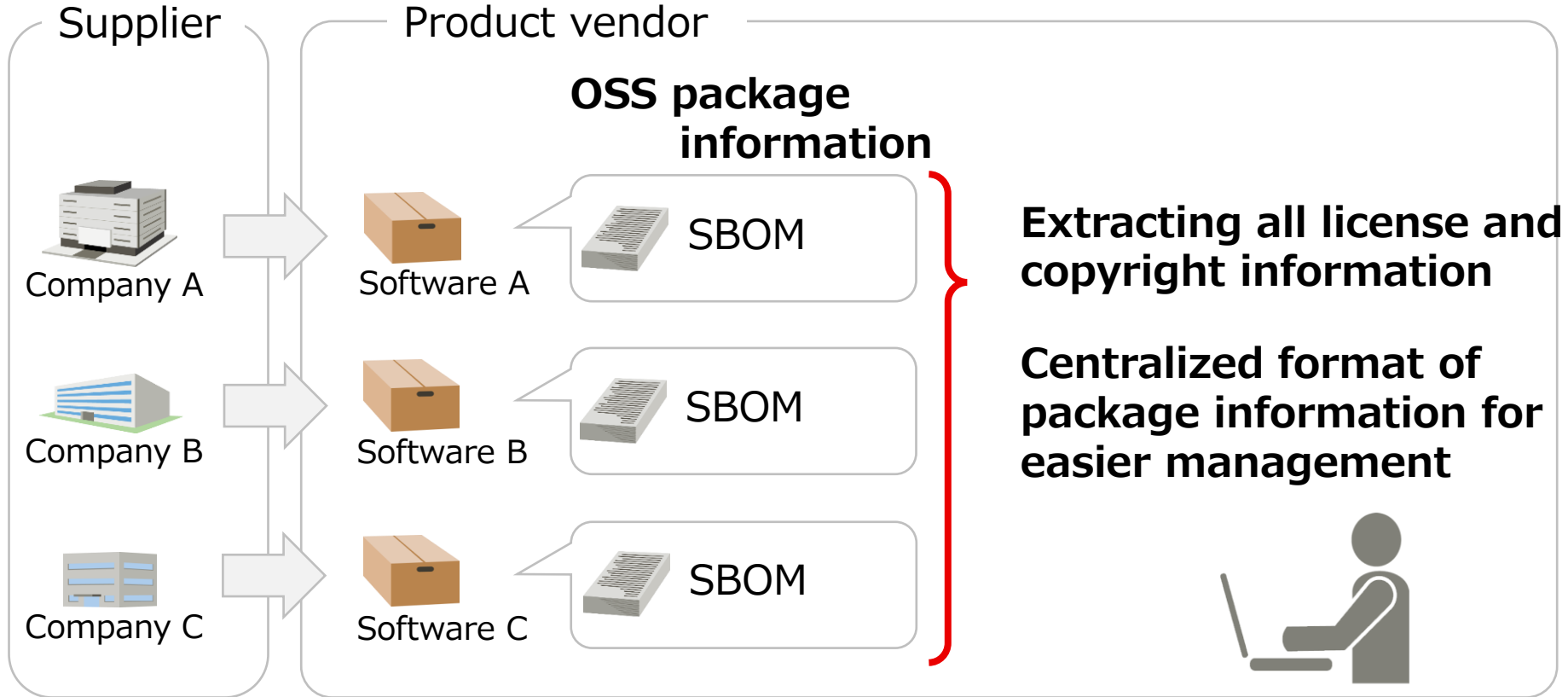
- SBOM : **S**oftware **B**ill **o**f **M**aterials
- Complete list of software parts included in the package
- Information contained in the SBOM
  - Package Name, Version, Author / Supplier,
  - File info & hash, Relationship, etc.
- Can be used to protect the software supply chain



# Why using standardized SBOM?



# Why using standardized SBOM?

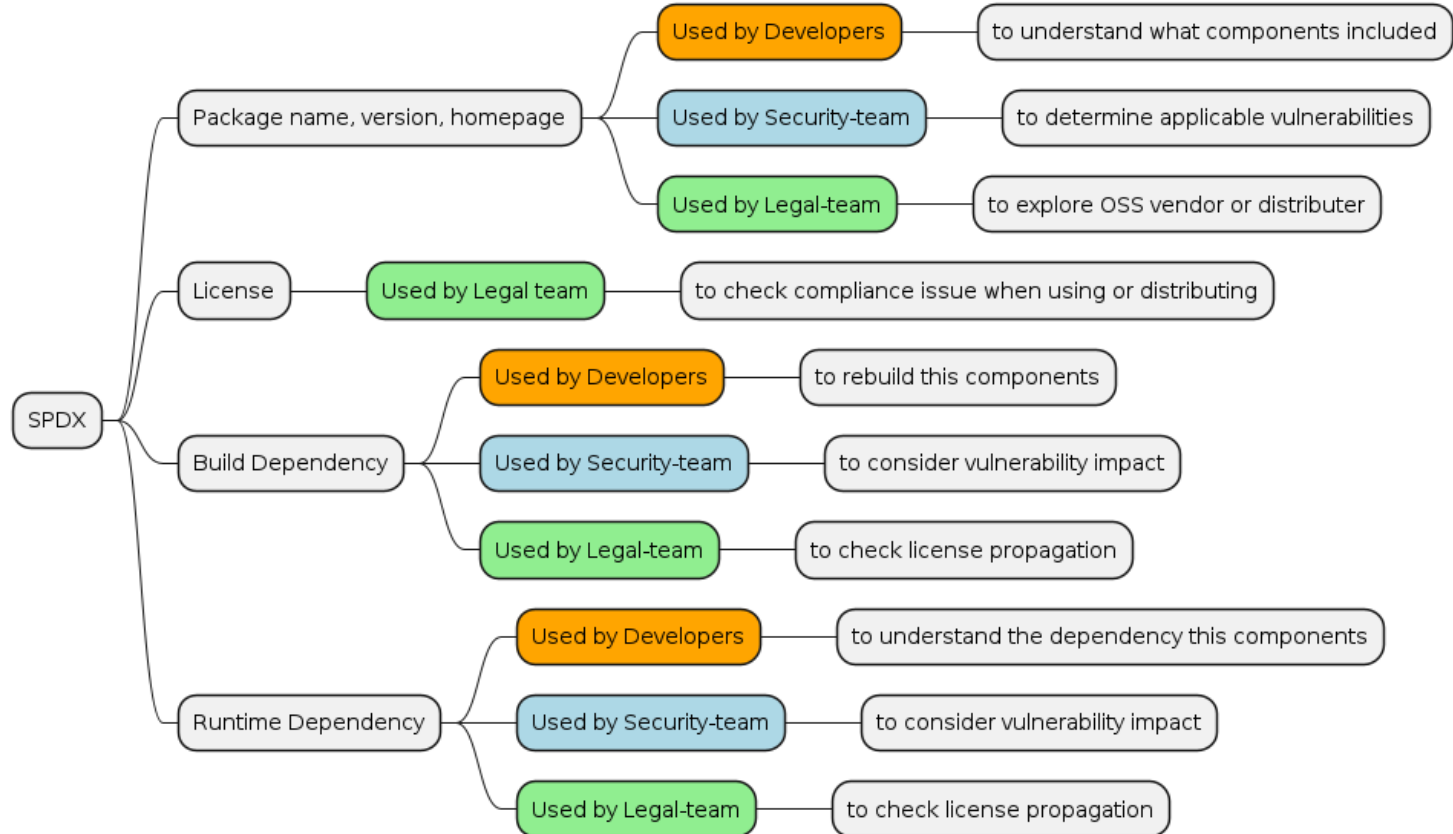


- **SPDX: Software Package Data Exchange**
  - ISO/IEC 5962:2021, Linux Foundation
  - A standard language for communicating software-related components, licenses, copyrights, and security information in multiple file formats.
  - Can be expressed in tag/value (.spdx), JSON, YAML, RDF/XML, and spreadsheets formats.



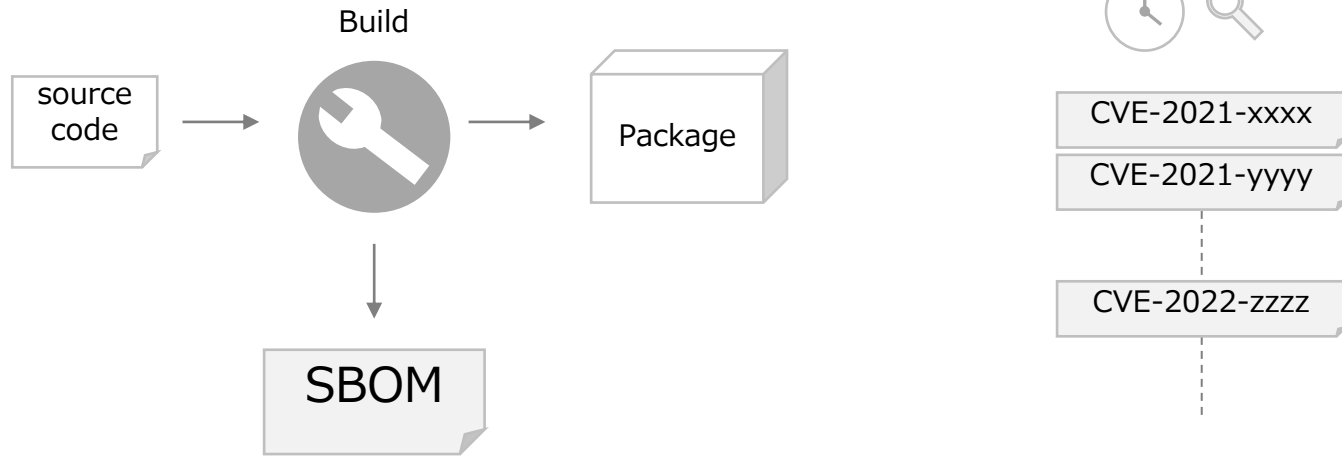


# Use-Case Example of SPDX fields

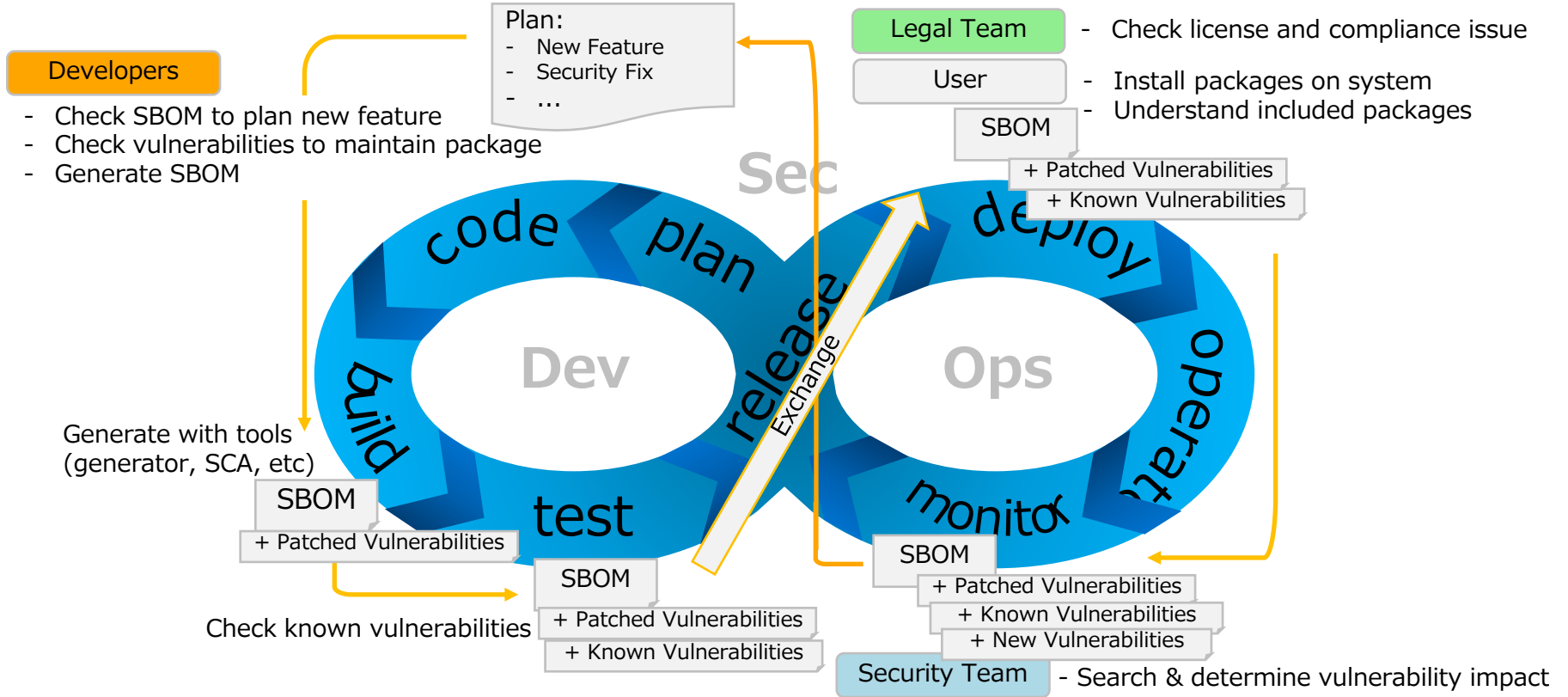


# SBOM and Vulnerabilities

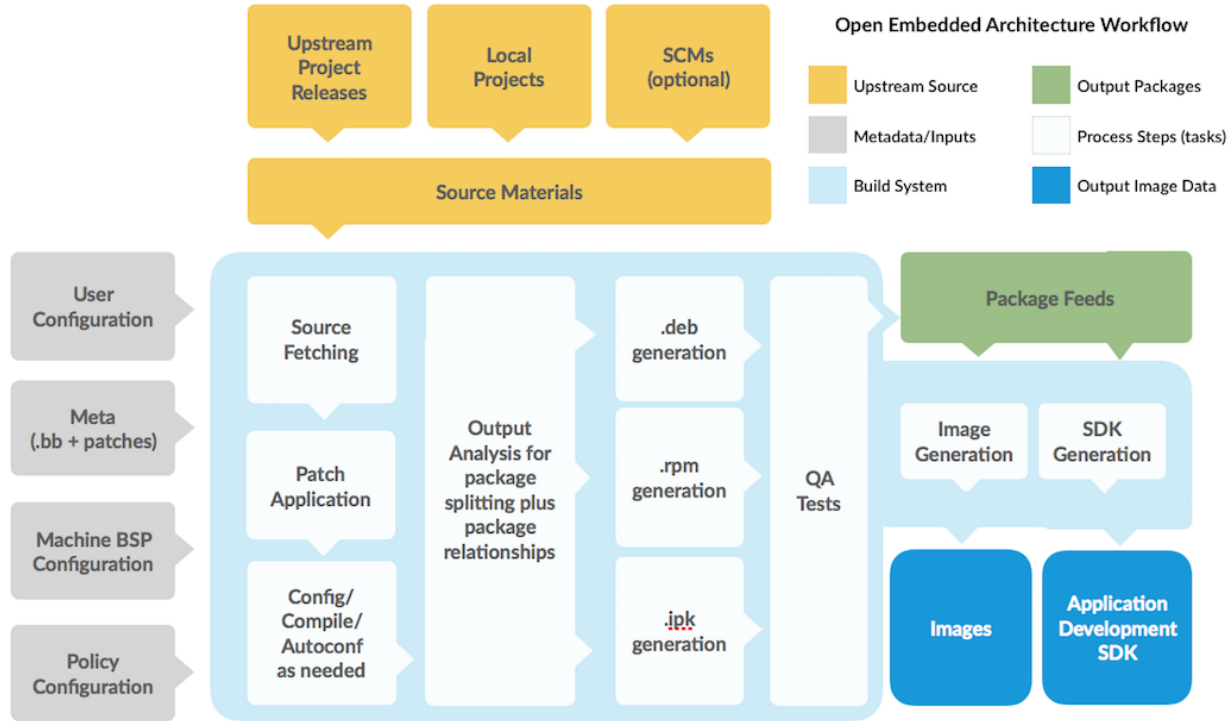
SBOM can be determined at Build-Time,  
but Vulnerabilities appear randomly.



# SBOM in Software Lifecycle



# SPDX generation and Vulnerabilities detection by Yocto



<https://www.yoctoproject.org/software-overview/>



cve



images



licenses



rpm



sdk



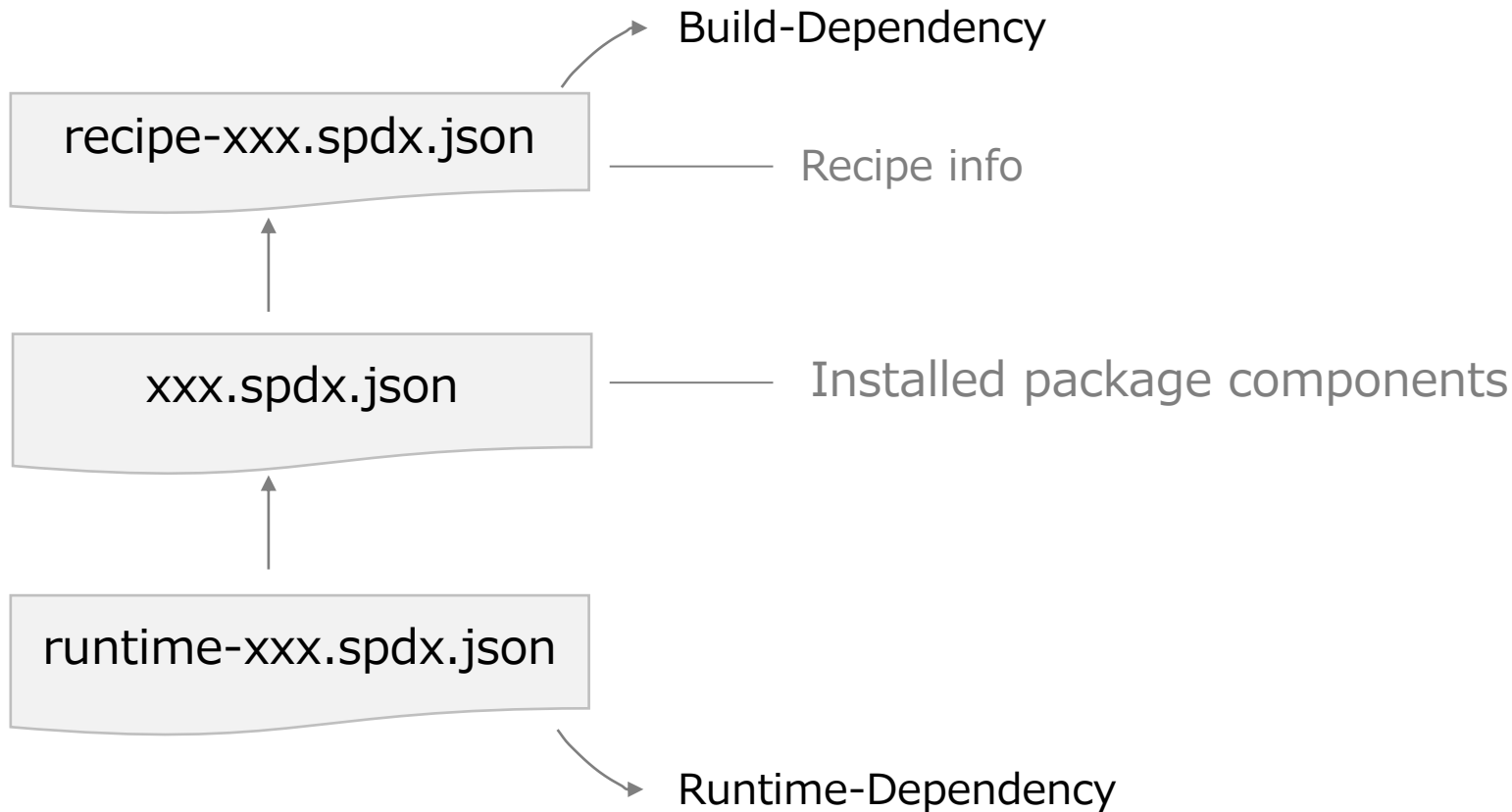
spdx

## Output from yocto builds:

<build>/tmp/deploy/

- **cve/** -- Vulnerabilities generated by cve-check
- images/ -- bootloader, kernel & initramfs, rootfs and metadata
- licenses/ -- Copyright & License text files per-package
- rpm/ -- Packages (rpm/deb/ipk)
- sdk/ -- Toolchain for cross-building
- **spdx/** -- **SPDX files**

- meta-spdxscanner
  - Generate a SPDX file by calling FOSSology or ScanCode Toolkit (Not work alone)
  
- create-spdx
  - What's New in Yocto 3.4 (honister)
  - Operate independently, easy to use
  - Only add the following settings to conf/local.conf  
INHERIT += "create-spdx"





# create-spdx output for busybox

```
▼ spdxDocument: "http://spdx.org/spdxdoc/recipe-busybox-1dd695ee-b779-5d09-9314-a0c15f2a4031"
▶ files: [-]
▼ hasExtractedLicensingInfos:
  ▼ 0:
    ▼ extractedText:
      "\n-----\n\nThis program, \"bzip2\", the associated library \"libbzip2\", and all\ndocumen
      1996-2006 Julian R Seward. All\nrights reserved.\n\nRedistribution and use in source and binary forms, with or without\nmodification, are permitted provided t
      conditions\nare met:\n\n1. Redistributions of source code must retain the above copyright\n notice, this list of conditions and the following disclaimer.\n\n
      software must not be misrepresented; you must \n not claim that you wrote the original software. If you use this \n software in a product, an acknowledgme
      documentation would be appreciated but is not required.\n\n3. Altered source versions must be plainly marked as such, and must\n not be misrepresented as bei
      software.\n\n4. The name of the author may not be used to endorse or promote \n products derived from this software without specific prior written \n permi
      PROVIDED BY THE AUTHOR ``AS IS'' AND ANY EXPRESS\nOR IMPLIED WARRANTIES, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED\nWARRANTIES OF MERCHANTABILITY AND FITNESS
      PURPOSE\nARE DISCLAIMED. IN NO EVENT SHALL THE AUTHOR BE LIABLE FOR ANY\nDIRECT, INDIRECT, INCIDENTAL, SPECIAL, EXEMPLARY, OR CONSEQUENTIAL\nDAMAGES (INCLUDIN
      PROCUREMENT OF SUBSTITUTE\ngOODS OR SERVICES; LOSS OF USE, DATA, OR PROFITS; OR BUSINESS\nINTERRUPTION) HOWEVER CAUSED AND ON ANY THEORY OF LIABILITY,\n\nWHETHER
      LIABILITY, OR TORT (INCLUDING\nNEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY OUT OF THE USE OF THIS\nSOFTWARE, EVEN IF ADVISED OF THE POSSIBILITY OF SUCH DAMAGE.
      Cambridge, UK.\n\njseward@bzip.org\n\nbzip2/libbzip2 version 1.0.4 of 20 December 2006\n\n-----\n\n
      licenseId: "LicenseRef-bzip2-1.0.4"
      name: "bzip2-1.0.4"
      name: "busybox"
▼ packages:
  ▼ 0:
    SPDXID: "SPDXRef-Package-busybox"
    copyrightText: "NOASSERTION"
    downloadLocation: "NOASSERTION"
    ▶ hasFiles: [-]
    licenseConcluded: "NOASSERTION"
    licenseDeclared: "GPL-2.0-only AND LicenseRef-bzip2-1.0.4"
    ▶ licenseInfoFromFiles: [-]
      name: "busybox"
    ▶ packageVerificationCode: {...}
      supplier: "Organization: OpenEmbedded ()"
      versionInfo: "1.35.0"
```

- Validation tools fail with warnings on Yocto Project's output  
<https://lists.openembedded.org/g/openembedded-core/message/173723>

- Additional settings in conf/local.conf

```
INHERIT += "cve-check"
```

- Location of output results

```
<<build dir>>/tmp/deploy/cve
```

- Run Build

```
$ bitbake core-image-minimal
```

# cve-check output for busybox (json)

```
version: "1"
package:
  0:
    name: "busybox"
    layer: "meta"
    version: "1.35.0"
    products:
      0:
        product: "busybox"
        cvesInRecord: "Yes"
    issue:
      0:
        id: "CVE-2006-1058"
        summary: "BusyBox 1.1.1 does not use a salt when generating passwords, which makes it easier for local users to guess passwords from a stolen password file using techniques such as rainbow tables."
        scorev2: "2.1"
        scorev3: "0.0"
        vector: "LOCAL"
        status: "Patched"
        link: "https://nvd.nist.gov/vuln/detail/CVE-2006-1058"
      1:
        id:
        summary:
        scorev2:
        scorev3:
        vector:
        status: "Patched"
        link: "https://nvd.nist.gov/vuln/detail/CVE-2006-5050"
      2:
        id: "CVE-2011-2716"
```

- "Unpatched" marked if this CVE affects this package
- "Patched" marked if cve-check determines that this CVE does not affect this package from range of cpe version, patches for this CVE included in recipe, etc.
- "Ignored" marked if cve-check found this CVE in Ignore-list (CVE\_CHECK\_IGNORE)

# SBOM and Vulnerability Management with SW360

Home Projects Components Licenses ECC Vulnerabilities Requests Search Admin Pre

Home

MY PROJECTS

Project name	Description	Approved Releases
There are no projects found with your selection.		

Showing 0 to 0 of 0 entries previous next

MY COMPONENTS

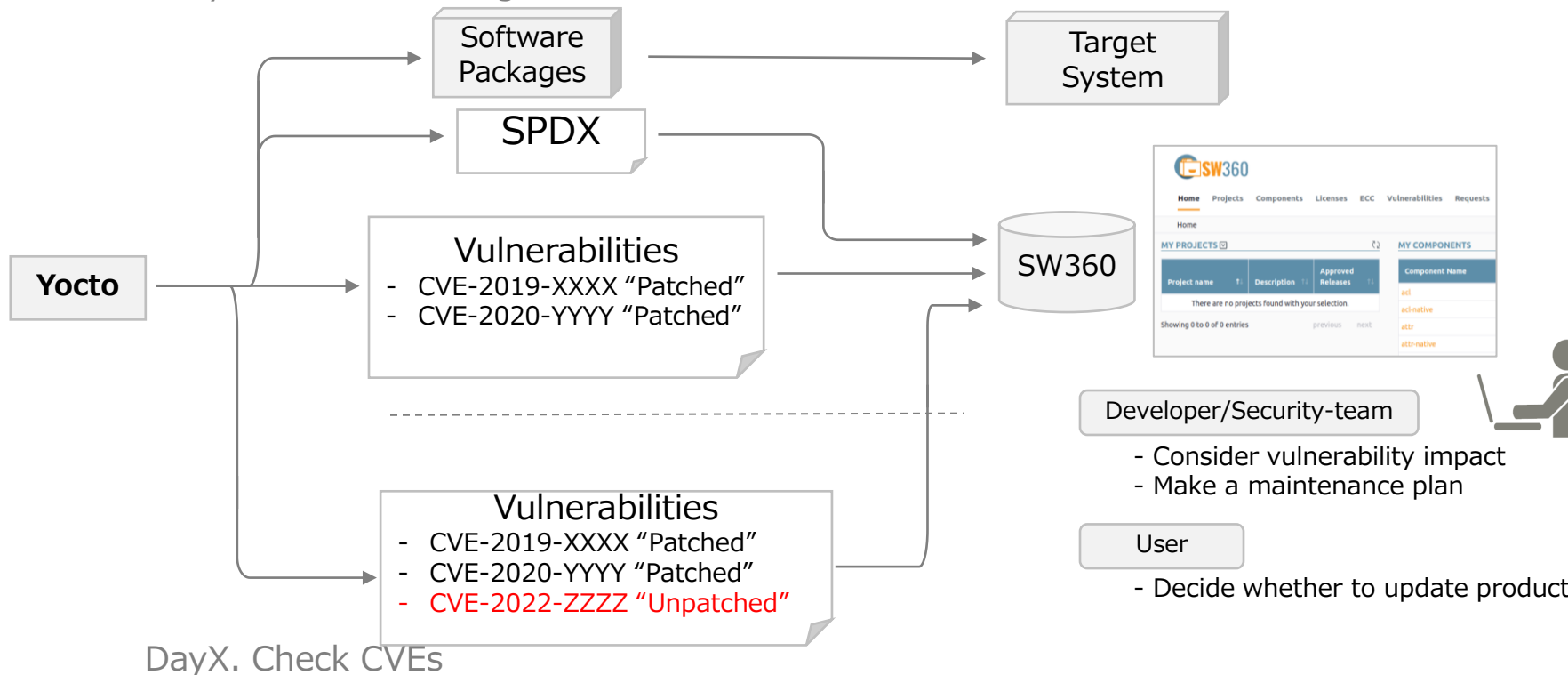
Component Name	Description
acl	ACL allows you to pr
acl-native	ACL allows you to pr
attr	Implement the abili
attr-native	Implement the abili
autoconf-archive	a collection of freely
autoconf-archive-native	a collection of freely
autoconf-native	Autoconf is an exten
automake-native	Automake is a tool fc
base-files	The base-files packag
base-passwd	The master copies of

Showing 1 to 10 of 218 entries previous 1 2 3 5 6 ... 22 nt

- SW360 is software component management tool
- Feature:
  - Software component management
  - License management
  - Vulnerability viewing
  - etc.
- HomePage: <https://www.eclipse.org/sw360/>
- License: EPL-2.0

# PoC Overview

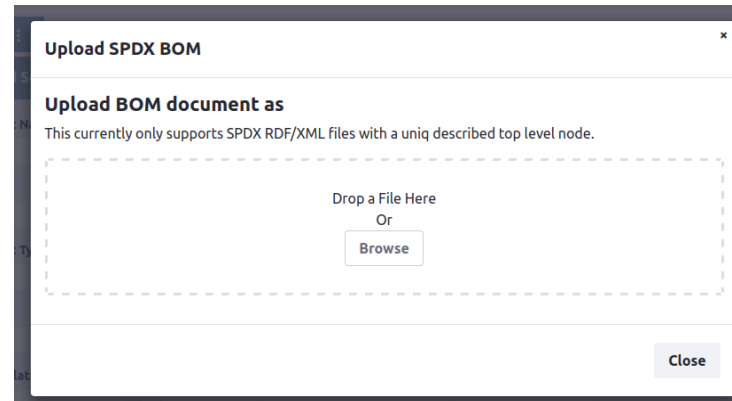
Day1. Release Packages



# PoC: Import SPDX to SW360

- 2 Ways to upload your SPDX.

- Web UI
- REST-API (\*need to change little)



```
$ curl 'http://localhost:8080/resource/api/components' -X POST \
-H 'Content-Type: application/hal+json' \
-H 'Authorization: Token z46P1IwS9JlxdkiDbv7v' \
-d "$(cat ${spdx} | jq '{
  name: .packages[0].name,
  description: .packages[0].description,
  componentType: "OSS"
}')"
```

SW360's field      SPDX's field

## Step:


- Create Vulnerabilities
- Create release-vulnerability relations

REST-API example

```
for cve_json in `ls *_cve.json`; do
  for i in $(seq 0 $(cat ${cve_json} | jq '.package[0].issue | length')); do
    curl 'http://localhost:8080/resource/api/vulnerabilities' -X POST \
      -H 'Content-Type: application/hal+json' \
      -H 'Authorization: Token z46P1IwS9JlxdkiDbv7v' \
      -d "$(cat ${cve_json} | jq "{
        externalId: .package[0].issue[${i}].id,
        title: .package[0].issue[${i}].id,
        description: .package[0].issue[${i}].summary,
        cvss: .package[0].issue[${i}].scorev3,
        references: .package[0].issue[${i}].link
      }")"
  done
done
```

CVE ID,  
Summary,  
Score,  
link to <https://nvd.nist.gov/>



Search...

[Home](#) [Projects](#) [Components](#) [Licenses](#) [ECC](#) **[Vulnerabilities](#)** [Requests](#) [Search](#) [Admin](#) [Preferences](#)

Vulnerabilities

**Quick Filter**

Show latest 200 ▾

Show 10 entries

**Advanced Filter**

**CVE ID**

**Vulnerable Configuration**

**Filter**

External Id	Title	Weighting	Publish Date
CVE-2018-1121	CVE-2018-1121	5.9	
CVE-2022-44034	CVE-2022-44034	6.4	
CVE-2022-44033	CVE-2022-44033	6.4	
CVE-2022-44032	CVE-2022-44032	6.4	
CVE-2022-43945	CVE-2022-43945	7.5	
CVE-2022-43750	CVE-2022-43750	7.8	
CVE-2022-42722	CVE-2022-42722	5.5	
CVE-2022-42721	CVE-2022-42721	5.5	

**Vulnerable configurations:**

**CVE-2020-15523 IS PRESENT IN THE FOLLOWING RELEASES**

Release
python3 (3.11.0)
python3-native (3.11.0)

## ● Summary

- SBOM helps solve software lifecycle issues such as compliance issues, security issues, and more.
- It is important to manage and operate vulnerability information together with SBOM.

## ● Future work

- Tool support is essential for effective utilization of SBOM, and tools should be selected according to the use case.  
I will make Yocto's DevSecOps PoC with OSS Review Toolkit (ORT).

