Free and Open Source Software

Covering OSS Compliance with Software Tools

Contributed by Dr. Catharina Maracke
Dr. Michael C. Jaeger
Software Compliance Academy

Introduction

- Why we would need tools?
- First demand and process, then the tool
- A tool cannot provide decisions
- Only data for decisions
- Many cases where expert knowledge is required

“A fool with a tool is still a fool” (from the hardware world)

About Tools

Tools can be good for ...

- … generating reports
- … analyzing data
- … managing policies

Where is this required?

Snippet Scan

License Scanner

Product Report

Component Scanner

Disclosure Document

Binary Scan

Code Scanner

Compliance Workflow

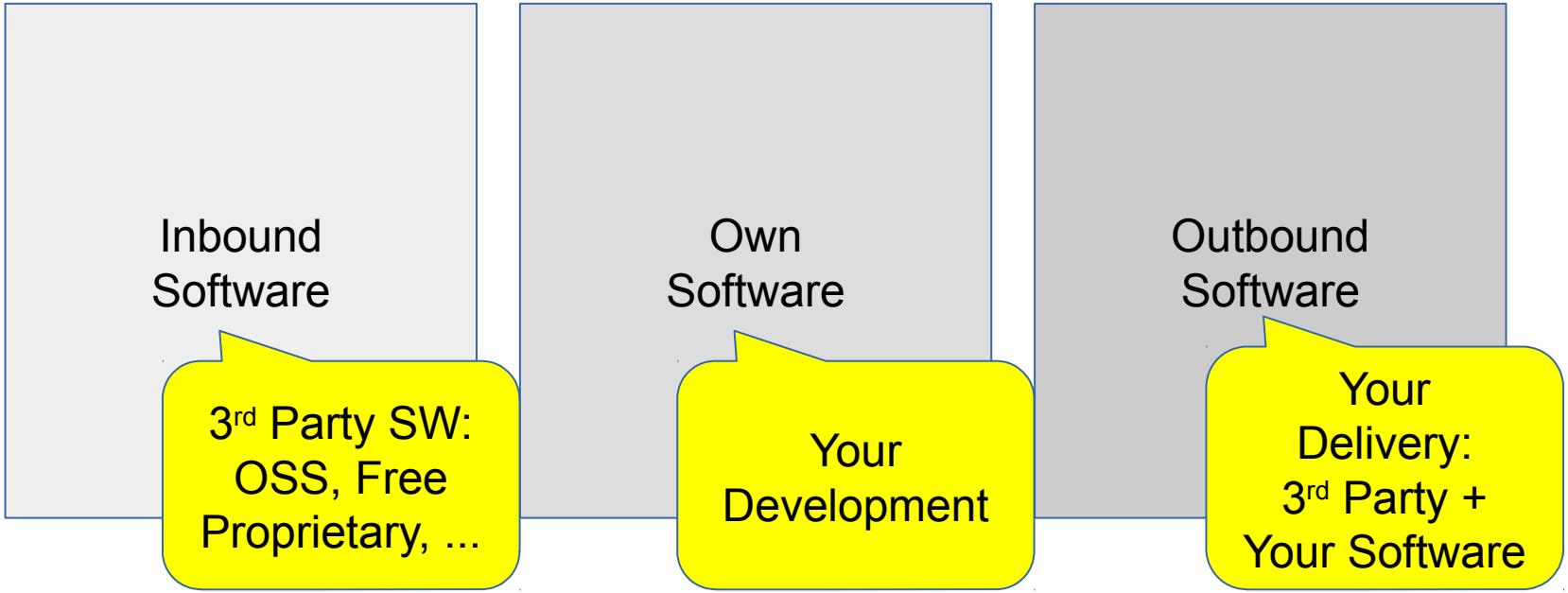
License Analysis

Compliance Workflow

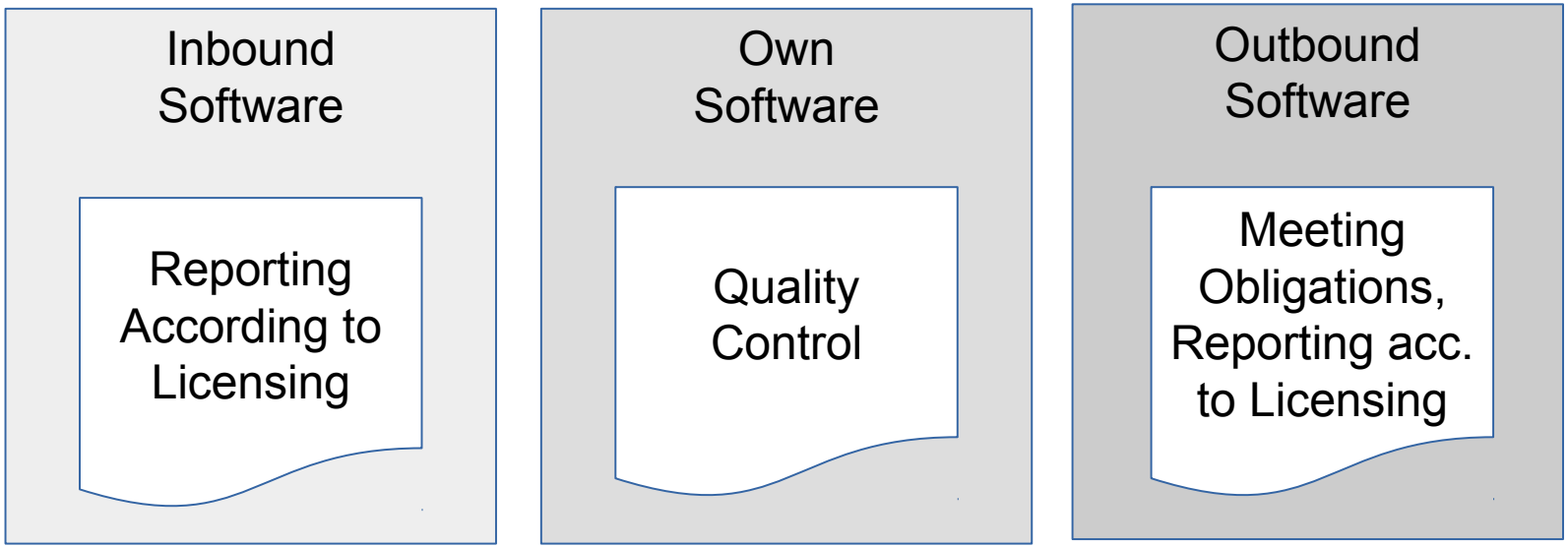
Software Situation

Inbound Software	Own Software	Outbound Software

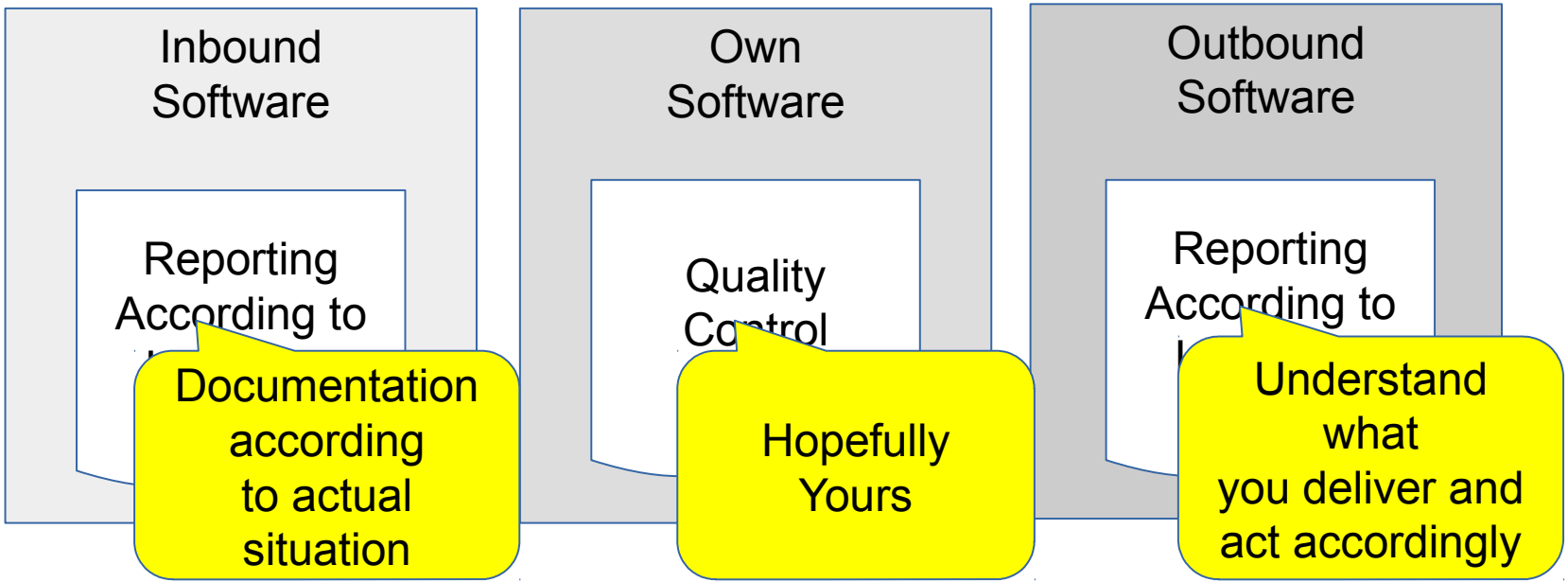
Software Situation – What it Means



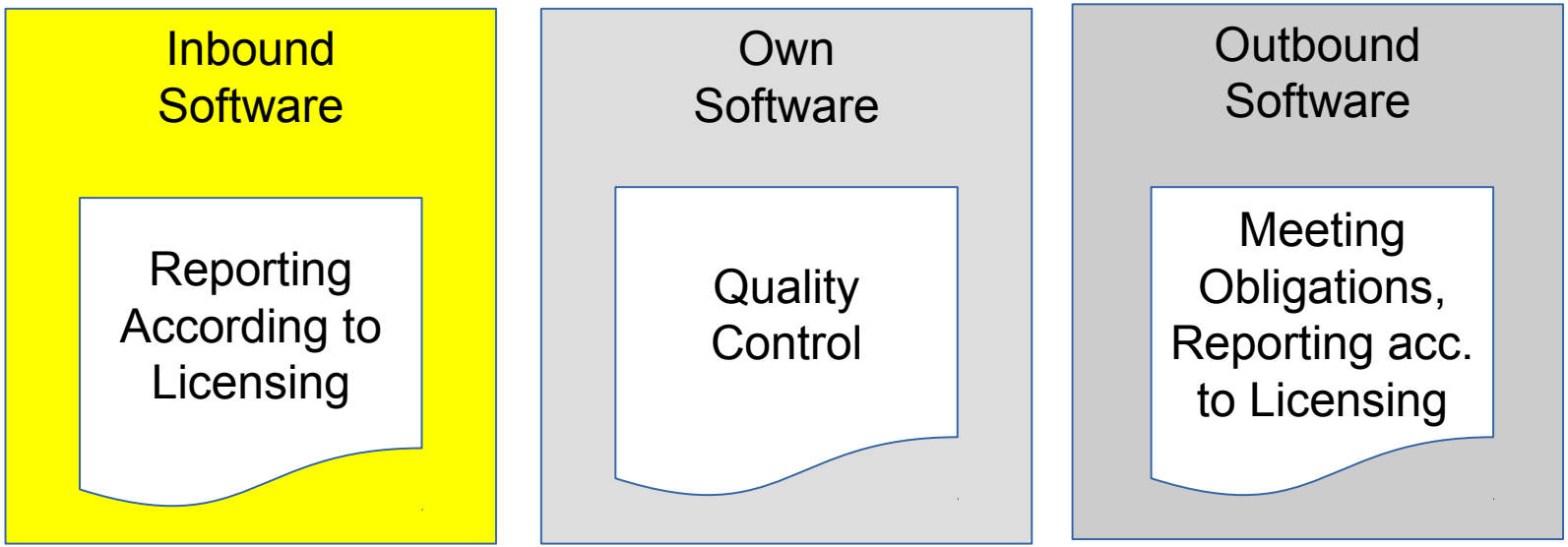
OSS License Compliance from 10k Feet



Again What this Means



Part I: Analysing Inbound



Understanding Inbound

- Determining which software is used (commercial + OSS actually)
  - Because commercial software can contain OSS as well!
- OSS components involved and their involved licensing
- Identifying licenses
- Identifying authorships and copyrights
- Determining any further points from licensing obligations

How to Understand What is Inbound

- Depends on the software technology used
- Modern software projects use dependency management
  - Declaration of imports, dependencies, used libraries, etc.
  - Defined dependencies can be extracted
  - In some cases for OSS, used component source code can be extracted
- However, involved software can be also in form of binaries
  - Origin and contents of binaries must be determined
- “Manual dependencies”: commercial software added

Identifying Licensing within Inbound Software: Easy Cases

- License, copying or notice document provided along with software
- At infrastructure, home page or project pages
  - e.g. Github or Sourceforge metadata
- Project definition file
  - e.g. in Java pom.xml
- Already provided license info
  - e.g debian-copyright or SPDX documentation

Identifying Licenses whithin Inbound Software: The Problem (1)

- License proliferation
  - About 350 „main“ licenses exist
  - A lot more out there
  - Existing licenses come at new versions
- Licenses in different languages (the French CeCILL)
- License obligations must be understood
- Commercial licenses such as an EULA lack standardization

Identifying Licenses within Inbound Software: The Problem (2)

- OSS = reuse
  - OSS components are not (always) homogeneous
  - If OSS exists, pull it from elsewhere
  - Code from many sources, different licensing
- Main license does not apply to all contents
  - If project does not enforce common licensing for all contributions
  - CLA: contributor license agreements

Identifying Licenses: The Fun (1)

Identifying license statements is not straightforward ...

* See README and **LICENS**E files in bz/ directory
* for more information
* about bzip2 library code.
*/

This file is part of Jam - see jam.**c** for **Copyright** information.

---

* See **LICENS**E.qla2xxx for **copyright** and **licens**ing details.

/* **Licens**ing details are in the COPYING
file accompanying popt source **distribut**ions, available from
ftp://ftp.rpm.org/pub/rpm/dist. */

Copyright (c) Insight Software Consortium. All rights reserved.
See ITK**Copyright**.txt or http://www.itk.org/HTML/**Copyright**.htm for details.

---

* See wps_upnp.**c** for more details on **licens**ing and code history.

Identifying Licenses: The Fun (2)

… or just very difficult statements

```
* Copyright (c) 1998-1999 Some Company, Inc. All Rights Reserved.
*
* This software is the confidential and proprietary information of Some
* Company, Inc. ("Confidential Information").  You shall not
* disclose such Confidential Information and shall use it only in
* accordance with the terms of the license agreement you entered into
* with Some Company.
*
* Some Company  MAKES NO REPRESENTATIONS
* OR WARRANTIES ABOUT THE SUITABILITY OF THE
* SOFTWARE, EITHER EXPRESS OR IMPLIED,
* INCLUDING BUT NOT LIMITED TO THE ….
```

Identifying Copyright

Some licenses ask for copyright notice or author listing

- Resulting obligation of providing these
- Generally, there is software for these problems
- Challenge: wrongly expressed copyright statements

Identifying Copyright: Fun (again)

Identifying copyright statements is not less fun:

```
Copyright by many contributors; see http://babel.eclipse.org/
---
 *  Original Code <s>Copyright (C) 1994, Jeff Hostetler, Spyglass, Inc.</s>
 *  Portions of Content-MD5 code <s>Copyright (C) 1993, 1994 by Carnegie Mellon
 *     University</s> (see Copyright below).
 *  Portions of Content-MD5 code <s>Copyright (C) 1991 Bell Communications
 *     Research, Inc. (Bellcore</s>) (see Copyright below).
 *  Portions extracted from mpack, John G. Myers - jgm+@cmu.edu
 *  Content-MD5 Code <s>contributed by Martin Hamilton (martin@net.lut.ac.uk)</s>
```

Identifying Licenses: Binaries

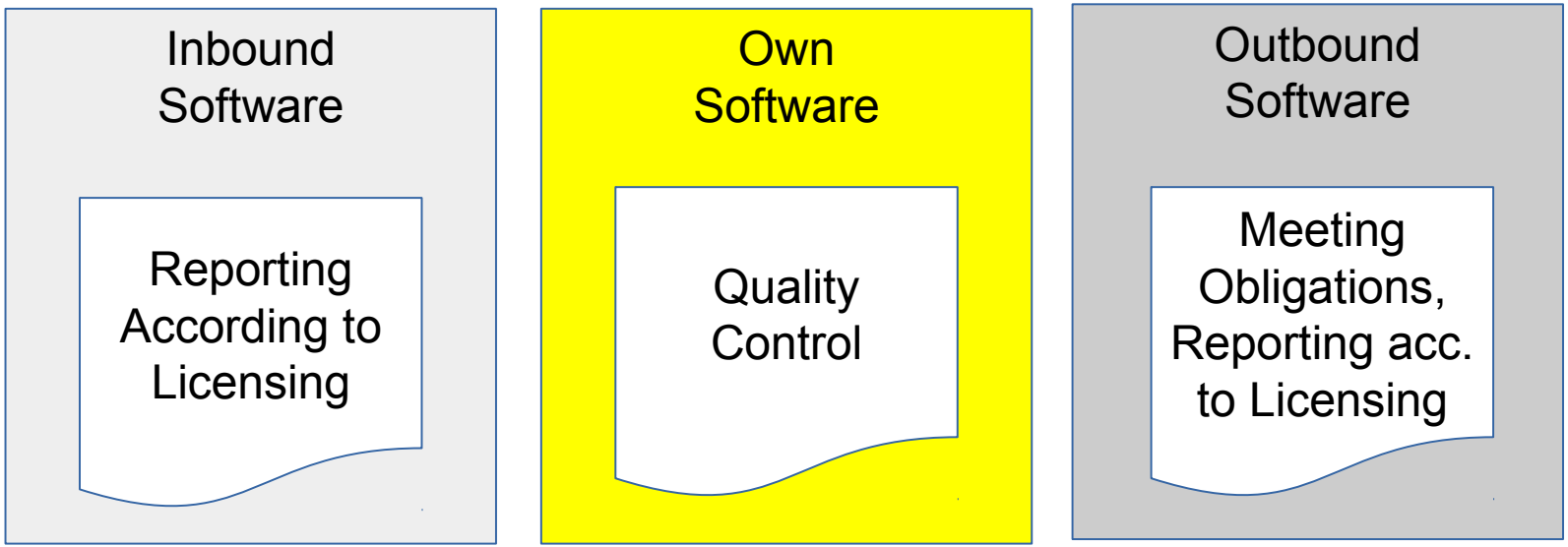
Binaries are compiled applications, libraries, software that can be used

- Binary = code translated from programming language to executable code by processor → information encoded
- Binaries can be part of an OSS component distribution
- Binaries can include OSS

How to understand what is contained in a binary?

- Main problem 1: different binary technologies
- Main problem 2: small variations, new binary

Part II: Your Own Software



What is the Issue with Your Software?

Sometimes, genuinely written software is expected
but “copy & paste” solution can be very near

- Open source projects are publicly available
- But also other files are valuable: scripts, icons, images, css files
- and code copied from Web sites for best practices and snippets

Copy paste of source code from the Internet in your code can be done:

- Respecting the author’s interests required: licensing, copyright
- Generally, reuse is good - opposed to reinventing the wheel

Code Scanning

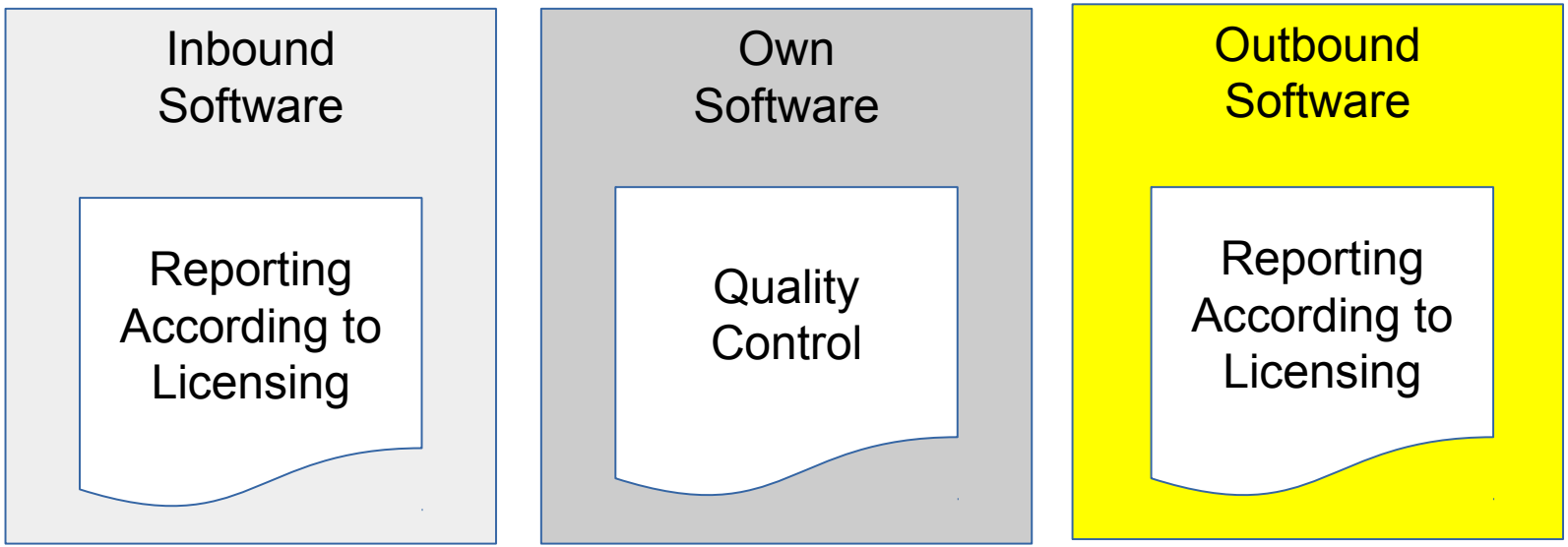
Good education and engineering codex can be solution

- Plain “copy & paste” of source code is bad practice anyway today
- Duplicated code reduces maintainability
- Engineers like clean dependency management

For all other cases

- Scanning tools for source code based on comparing text portions
- Using a database of already published source code (by other party)
- What is in Internet, tutorial code from vendors, Github
- Licensing: scan for licensing statements again

Part III: Outbound Software



Case 1: Distribution of OSS (1)

Distributing OSS as part of product or project

- E.g. requires notice file
  - Listing all licenses, listing copyright notice
  - … as a basic and common license obligation
- E.g. written offer to provide the OSS code

Builds upon knowledge on

- Which OSS components are in (here comes the BOM!)
- Which licenses in there, copyright notices

Case 2: Quality Management

Project or product documentation can require, e.g.

- All tests passed
- But as well: all licenses checked?
  - For their obligations, for their compatibility
- Or: All OSS required material ready for distribution

Requires (as well)

- Which OSS components are in
- Which licenses in there, copyright notices

Case 3: Ensuring Distribution Rights

Some licenses are not compatible

- That is life, for example GPL <-> EPL incompatibility
- *Distribution based on GPL works and EPL works: maybe a problem*

Some license statements are ambiguous

- For example „Licensed under BSD”
- *Requires legal decision how did you decide this statement*

Besides Delivering, Internal Work

Some license statements need documentation

- For example: „for license conditions, see web site”
- *Web site needs to be archived*

Some licenses are just not compatible with the business case

- E.g. Start up implements medical analysis algorithm after years of research, danger of being copied by market leaders
- *License obligations need to be compatible with business goals*

Excursus: Not OSS only, all 3rd Parties

Also with commercial software, appropriate licensing must be ensured:

- Does contract cover rights for intended commercial use?
- Where is the contract by the way?

Ensuring distribution obligations is required, for example:

- Documentation of distribution
- Time- / volume-limited licensing
- Logo printed on box necessary

BOM Documentation (1)

„Bill of Material” such as SPDX:

- It is a general question what is in the delivery
- Understand the nature of the delivery (How much OSS?)
- Understand potential issues (IP)
- How else to ensure license compliance?
- Basics of supply chain issues actually apply also to software

BOM Documentation (2)

Bill of Material is general obligation, for example at:

- USA: Cyber Supply Chain Management and Transparency Act of 2014
- Germany: KRITIS: BSI-Kritisverordnung [1]
  - Obliged to report service disturbances
  - Obliged to implement information security
  - Requires knowledge about BOM

[1] https://www.bmi.bund.de/SharedDocs/pressemitteilungen/DE/2017/06/nis-richtlinie.html

Your Own Software as OSS (1)

Yes, it is true: sometimes software developers
want to publish their work

- Excursus: Motivation 3.0 [2]
- How to publish? - A process topic
- But documentation is required (besides the publication)
  - What are the involved licenses
  - What is the own license
  - Are formal aspects met?

[2] https://www.youtube.com/watch?v=u6XAPnuFjJc

Your Own Software as OSS (2)

Analysis here has the goal to

- Confirm involved OSS licensing, business compatible?
- Identify dependencies and binaries
- Checking if all the source code is of our origin?

General quality points (including, but not limited to):

- Do all files have headers? (disclaimers for config files)
- Do all files have copyright and authorship statements
- Is the documentation of the licensing appropriate?

Summary

Summary of Tool Support

Tools are there, but requirements and purpose require understanding

- First comes the definition of what is needed and then the tool
- Tools are there for analysis, reporting and management

Different tools serve different purposes

- Requires integration of different functions
- Integration poses classic IT problems
- Interfaces must be understood to avoid manual effort

Questions?

office@scompliance.com