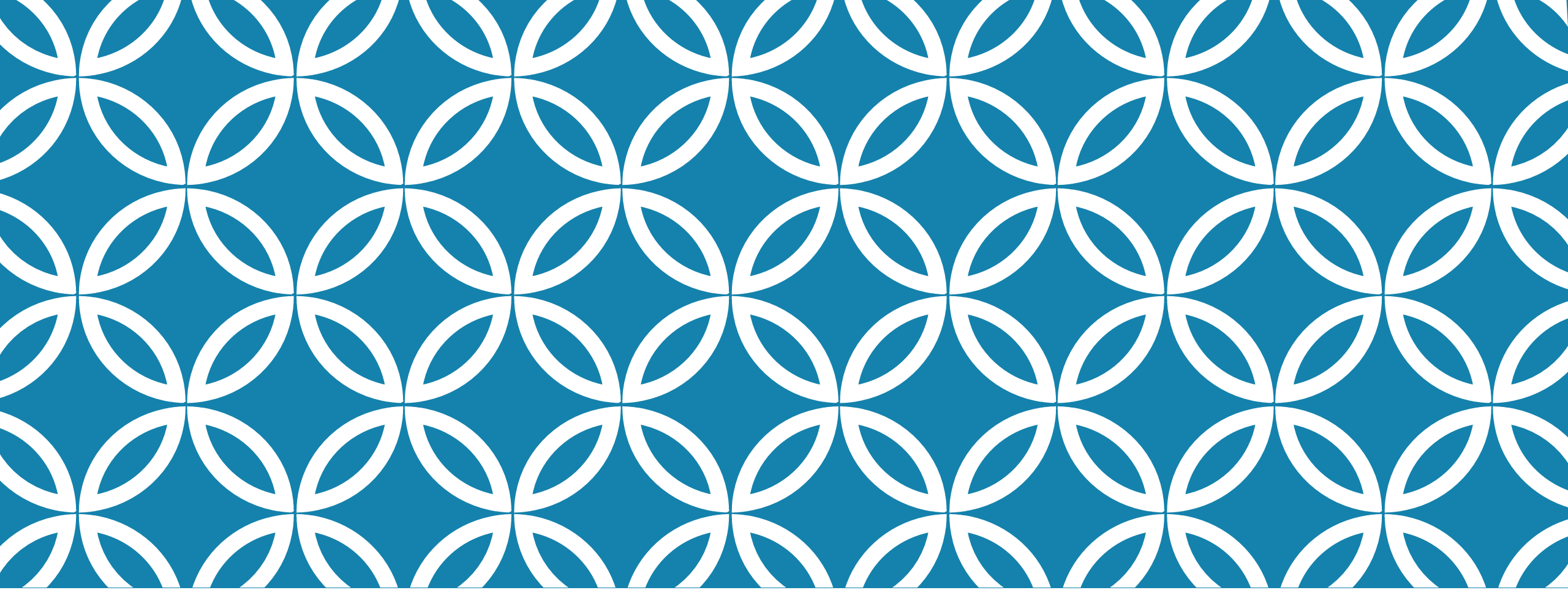


OPEN SOURCE IN FSI

Jeff Luszcz
@JeffLuszcz
jluszcz@peak6.com

A LITTLE ABOUT ME: JEFF LUSZCZ

Director Open Source @ PEAK6

Founded Palamida in 2004

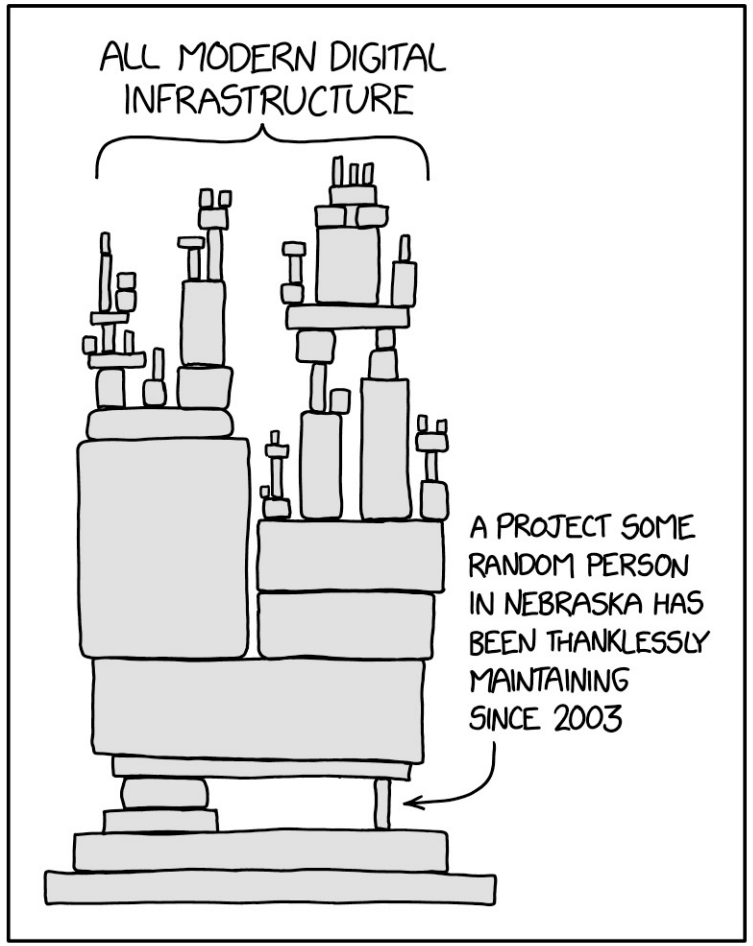
One of the first Scanning tools to manage FOSS

Designed compliance audit programs covering everything from basic compliance, OSS security, M&A due diligence, and open source project hygiene

TODAY'S AGENDA

- Open Source usage today
- The Software Supply Chain
- OSS Vulnerability Management
- Open Source License Compliance
- OSS Scanning
- Best Practices

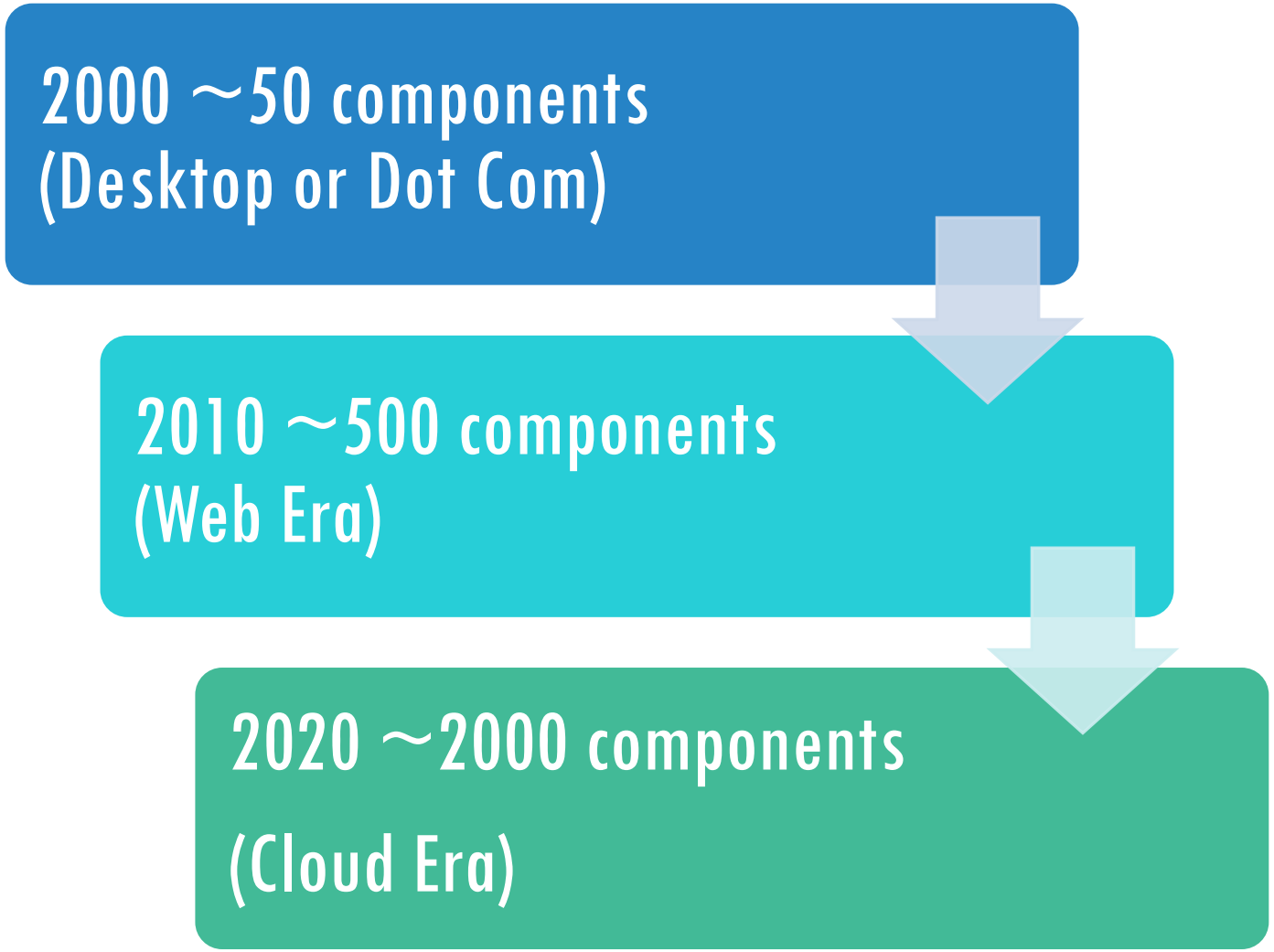
OPEN SOURCE: WHAT'S GOING ON BEHIND THE SCENES?



- Your applications are likely >90% Open Source
- > 90% of the the software libraries your organization uses are unknown and untracked
- Many are outdated and contain security vulnerabilities
- If you don't know what you are using, you can't comply with your legal license obligations
- If you don't what you are using, you can't support those projects with money, contributions, credit and other support
- The Software Supply Chain is long and insecure

https://xkcd.com/2347/

4



HOW HAS OSS USE CHANGED OVER THE YEARS?

THERE ARE MANY WAYS TO ACQUIRE OSS

Using a repository manager like Maven, NPM, pip, etc...

Direct download of source from web / Github

Loaded from a Content Delivery Network (CDN)

Bundled with other projects (OSS and Commercial!)

As part of your infrastructure (OS, DB, etc...)

From a vendor / supplier

A magic install script

Copied from a Pastebin / Gist

Cut and Paste of snippets

THE SOFTWARE SUPPLY CHAIN

The Software Supply Chain is similar to the physical product supply chain

Often contains hundreds or thousands of individual organizations

Mixture of OSS, Commerical and "free"

You may have no access or contact with many of your suppliers

You may not even know who they are!

BILLS OF MATERIALS

A Bill of Materials (BOM) is a record of the third party software we depend on

Consists of the Component name, Version, License and other information

SUPPLY CHAIN ATTACKS

Initial focus after the Heartbleed (OpenSSL) and Equifax (Struts 2) attacks

SolarWinds attack has put Federal and Industry focus on Supply Chain management

Federal requirements for Bills of Materials have now been published:

https://www.whitehouse.gov/briefing-room/presidential-actions/2021/05/12/executive-order-on-improving-the-nations-cybersecurity/

8

OSS SECURITY: WHAT IS A CVE?

All software bugs, some are well known and even have names and webpages!

The CVE list is a list of public software vulnerabilities (OSS and Commercial)

https://cve.mitre.org

Each defect is given a number CVE-2021-0001 (label-year-id)

MANY other security defects don't get this level of visibility. They live in the project defect tracker, are not named, and are hard to identify.

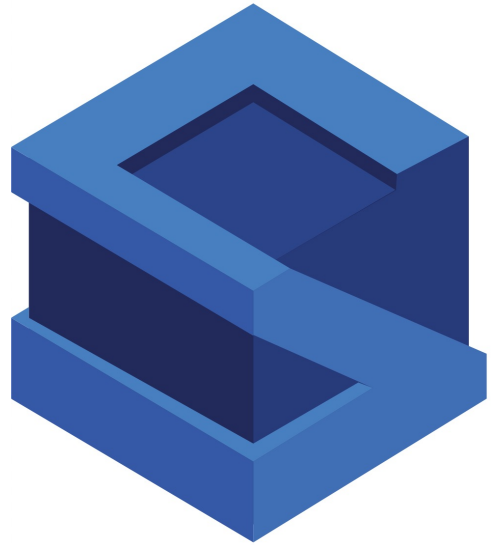
VULNERABILITIES, CVES AND PATCHING

Heartbleed
CVE-2014-0160

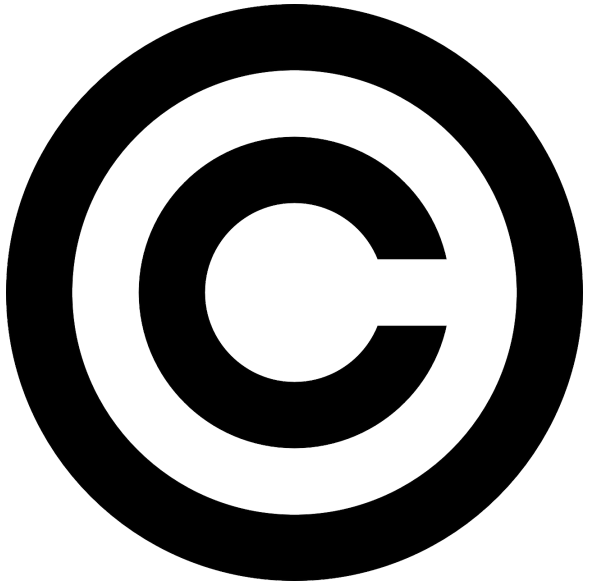
Affects OpenSSL

Struts
CVE-2017-5638

Affects Apache Struts, led to the massive Equifax breach ($500 million and counting)



OPEN SOURCE LICENSE COMPLIANCE



Copyright law allows authors to control their work (software).

You need explicit permission to use someone else's work

An author gives others permission using a license

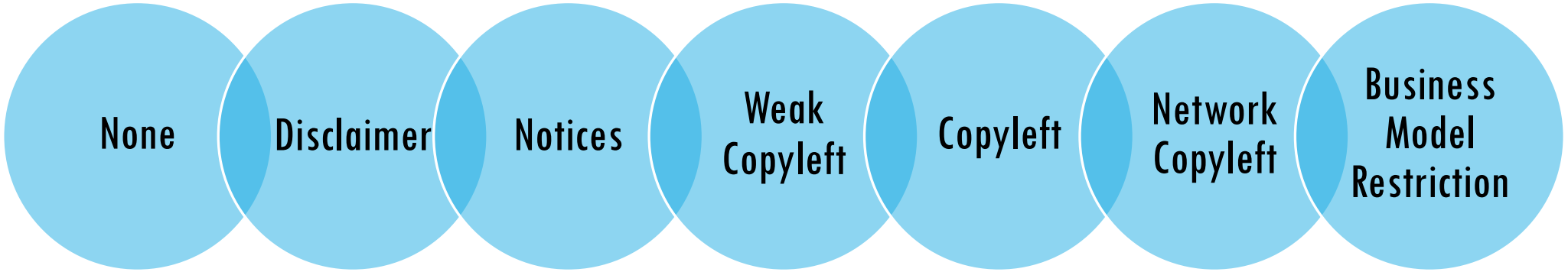
A Commercial license typically gives permission for money

An Open Source License gives permission as long as certain obligations are fulfilled

A license is a legal agreement which may be difficult to understand....

So we re-use COMMON open source licenses to make software re-use easier!

THERE IS A SPECTRUM OF OBLIGATIONS



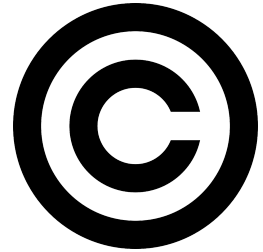
A license may require <u>one</u> or more obligations

Some obligations are easier to comply with than others

WHAT IS A LICENSE OBLIGATION?

Obligation	AKA	Description
Pay Money	Commercial	Pay money to use
Share Source Code	Copyleft / Viral	Bundle or share source code if used
Share Credit	Attribution / Notices	Requires copyright or notice to be shown in About Box / Documentation / Webpage / Source Code
Share Patents	Patent Grant	Provide free use of patents required to use software
No Patent Lawsuits	Patent Retaliation Clause	Removes patent rights if user sues for patent infringement
Restriction on Use	Prevent use by certain industries / companies / governments / military	Prevent use by military, nuclear power plant, aviation, companies, countries, business partners
Vanity License Obligation	Requires some non-traditional action	Buy me a beer if this helps you, Do no evil, Get vaccinated
"Don't Sue Me"	Disclaimer / AS-IS	The user can't sue the OSS author

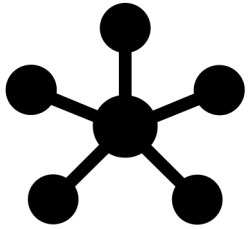
TWO (ORIGINAL) STYLES OF OSS LICENSES



"Permissive" sometimes called Attribution or Notice licenses

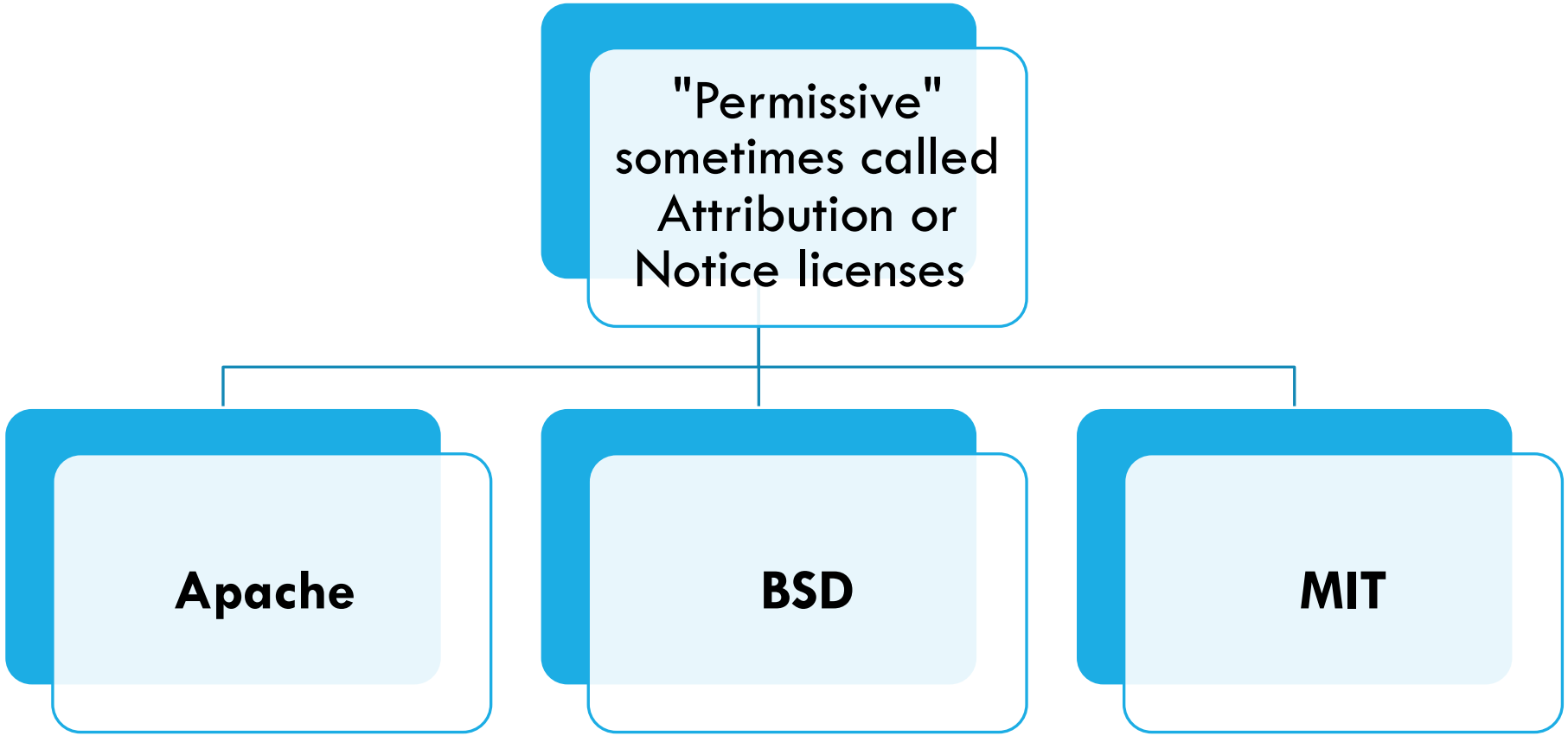
Requires preserving or supplying copyright notices and and/or license text

Copyleft (sometimes called Reciprocal or Viral) Licenses



Requires supplying some or all of the source code of a program under certain conditions

"PERMISSIVE" LICENSE EXAMPLES



NOTICES

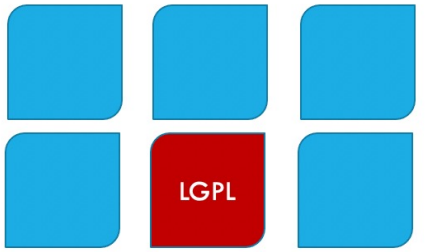
Many open source license requires copyright statements and/or license text to be preserved and passed along to the end user.

These notices are often found in

- About Box
- Legal Info menu
- Documentation

- An OSS management program needs to be able to produce these automatically

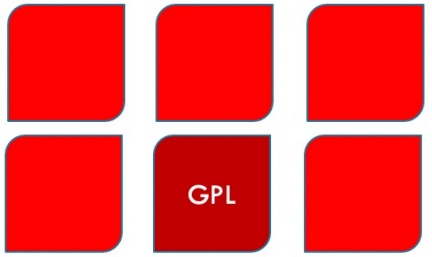
COPYLEFT / RECIPROCAL / VIRAL LICENSES



Copyleft (sometimes called Reciprocal or Viral) Licenses

Lesser General Public License (LGPL)

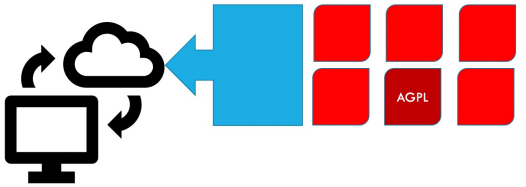
Requires supplying source all code from LGPL module if distributing a program using a LGPL module



General Public License (GPL)

Requires supplying source for all linked code if distributing a program

Affero General Public License (AGPL)



Requires supplying source code if using a modified network application under the AGPL

POST AGPL NON-OSS LICENSES: COMMONS CLAUSE, SERVER SIDE PUBLIC LICENSE

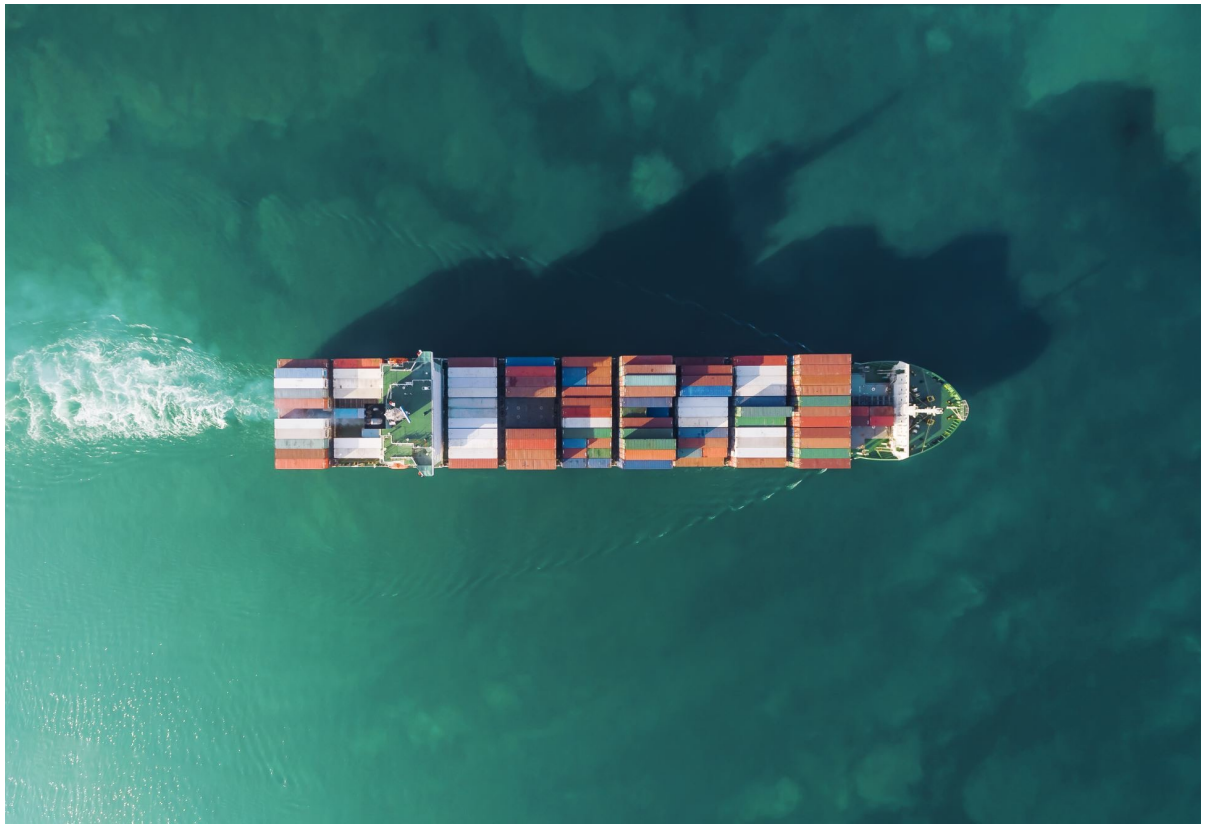
The AGPL attempted to close what was perceived as a loophole for OSS license obligations for Cloud applications

Some companies are building applications / databases and seeing others make money off of selling access or hosting to those same applications

The Commons Clause, Server Side Public License and other similar licenses put restrictions on certain business cases such as hosting builds of the original software

These are not OSS licenses, but often mentioned in similar contexts

Often seen around Open Core projects!



DISTRIBUTION / WHEN DO I NEED TO CARE?

Many open source licenses ONLY come into effect when software is distributed

This might be as a downloaded application, App, Container or on a Device

DISTRIBUTION USE CASES

Products (or modules of products) can be used and distributed in many ways:

- Internal Use
- Binary/ EXE delivered to end user
- Container based
- Mobile applications
- Self-hosted Software as a Service (SaaS)
- SaaS Pushed to "The Cloud!" (AWS, Azure, Google Cloud Platform)
- Javascript files downloaded to local web browser as part of SaaS app
- "Private" cloud version for Marquee customer

Distribution models affect OSS License obligations!

OPEN SOURCE LICENSE POLICIES

Not all licenses are appropriate for your use case

Open Source License Policies are how you can define what licenses are acceptable for your organization or product.

Often based on distribution model

It is important to make a clear license policy and have all developers understand what is expected.

Need to be updated periodically

It is VERY expensive to rip out unacceptable code at a later point!

Licenses

https://opensource.google/docs/thirdparty/licenses/

HOW TO BECOME COMPLIANT

Build a team of OSS Experts (Through an OSPO, OSS Working Group, etc…)

Education (e.g. Linux foundation IP and licensing Courses)

Create an initial Bill of Materials (BOM – pronounced like bomb)

Remediate Security and Licensing Problems

Become Openchain conformant

BEST PRACTICES: EDUCATION

Software developers lack training regarding licensing and security

OSS Policies are missing, neglected or impossible to find

Legal can be scared to look for problems

Cost to fix goes up with every layer built upon a mistake

Discovering problems at "Sales time" become red alerts and can destroy roadmaps and deals

No excuse not to Have EVERYONE get a basic training, good free training exists

https://training.linuxfoundation.org/training/open-source-licensing-basics-for-software-developers/

OSS IN MERGERS AND ACQUISITIONS

If you are buying or selling a company it is very common to perform OSS Due Diligence using a third party expert

This typically involves

- Sell side providing "Disclosures" of the OSS they depend on
- Sell side providing access to source code to the independent third party
- Buy side may respond with a list of requested Remediations
- Buy side may require financial hold backs due to IP risk

Time frame for this is typically 2 weeks for first report, a few more weeks for remediation

RELEASING SOMETHING UNDER AN OSS LICENSE

Choose a license that works for your use case

Remove commercial code (as necessary)

Remove “secrets”, passwords, machine names, etc….

Review use and license of multimedia, images, fonts, sounds, etc..

Review OSS usage and compliance with selected license

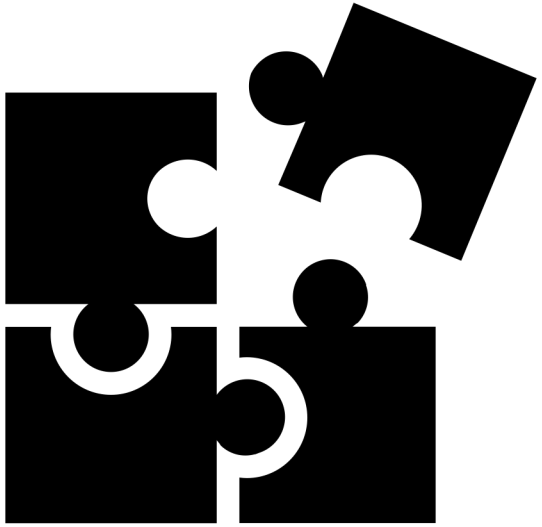
Review of Source Code Snippets may be warranted!

Remediate OSS as necessary, sometimes this means changing YOUR license

Generate License Notices

Decide on a Contributor Licensing Agreement, Developer Certificate of Origin and/or Code of Conduct, etc…

SOFTWARE COMPOSITION ANALYSIS (SCA) OSS SCANNING TOOLS



Automates discovery of OSS components, esp. those brought in by repository manager tools like Maven or NPM

Allows license policy to be set, enforced and modified

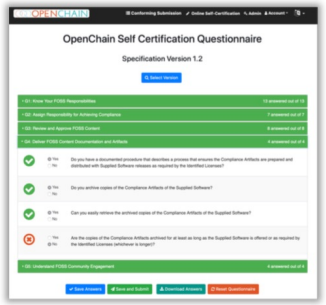
Allows vulnerability policy to be set, enforced and modified

Allows easy creation of up to date Bill of Materials (BOM) reports

Allows for alerting on security or license policy problems

OPENCHAIN: A BLUEPRINT FOR MANAGEMENT

OPENCHAIN



The OpenChain Project defines the key requirements for a quality open source compliance program.

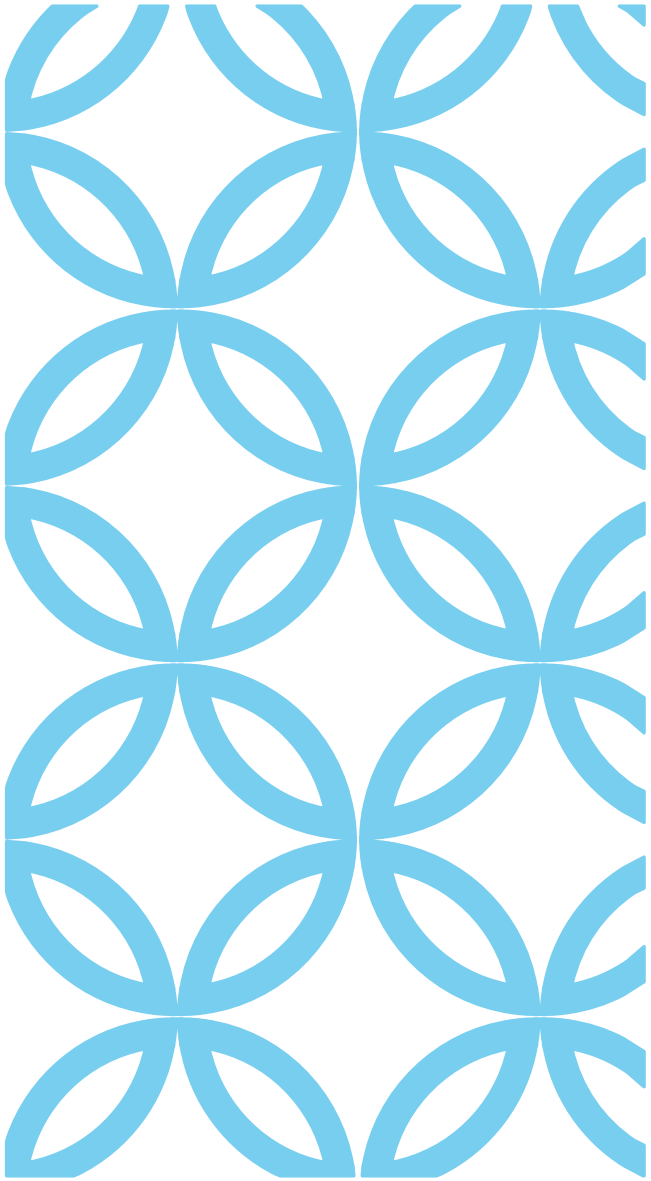
It is a simple, effective standard for organizations of all sizes in all markets. It is openly developed and freely available to all.

The OpenChain Project is supported by extensive reference material, free online self-certification and a vibrant community.

Learn more: www.openchainproject.org

COMPLIANCE BEST PRACTICES

- ❑ Use a Software Composition Analysis (SCA) scan tool or tools to build your BOM
- ❑ Automatically generate License reports and NOTICES files
- ❑ Create Source bundles (e.g. tarballs) of copyleft licensed code (GPL, LGPL, etc..)
- ❑ Track Commercial libraries and dependencies, keep track of payments / EULAs
- ❑ Track webservices
- ❑ Track changes to OSS source files, mark them appropriately
- ❑ Check patent issues esp. when dealing with codecs, audio and video
- ❑ Review Vulnerability Reports / CVEs
- ❑ Run SAST/DAST
- ❑ You keep this current!



QUESTIONS AND THANKS!

Jeff Luszcz
@JeffLuszcz
jluszcz@peak6.com