Copyright extraction – tools, challenges, myths

Armijn Hemel, MSc
Tjaldur Software Governance Solutions
armijn@tjaldur.nl

December 4/5, 2014

About Armijn

- using Open Source software since 1994
- MSc Computer Science from Utrecht University (The Netherlands)
- core team `gpl-violations.org` from 2005 - May 2012
- Tjaldur Software Governance Solutions since May 2011
- creator of the Binary Analysis Tool for compliance engineering of binary files

Today's topic: copyright extraction

Some licenses require proper attribution of copyrights. This has not been a priority until now but in recent times it has been given more thought, both by copyright holders doing enforcement, as well as companies (re)distributing software.

Problems and solutions are not always clear so this talk might be a bit chaotic. I will make some assumptions which are not necessarily true. This is still "work in progress" and I might be completely wrong.

If any lawyer wants to jump in, please do as I will need you!

GPLv2 section 1

1. You may copy and distribute verbatim copies of the Program's source code as you receive it, in any medium, provided that you conspicuously and appropriately publish on each copy an appropriate copyright notice and disclaimer of warranty; [...]

GPLv2 section 2

2. You may modify your copy or copies of the Program or any portion of it, thus forming a work based on the Program, and copy and distribute such modifications or work under the terms of Section 1 above, [...]

What does this mean?

Some questions:

- Is it enough to mention who wrote parts of the code?
- How verbose does an attribution need to be?
- Do filenames have to be included?
- Do years need to be included (possibly because copyright law might have changed in the meantime and the file predates the change?)

Let's look at an example and briefly discuss.

Example copyright statement from a file from the Linux kernel

```
/* tunnel4.c: Generic IP tunnel transformer.
 *
 * Copyright (C) 2003 David S. Miller (davem@redhat.com)
 */
```

Just four lines of text...and many questions

Tooling for copyright extraction

I use a few open source tools for copyright extraction:

- Ninka
- FOSSology
- homebrew script

and still have to do a lot by hand.

Ninka

Ninka is a *license* scanner, but it has a nice feature useful for copyright scanning: it can dump the header of the file containing comments.

Most of the time the copyright statements will be in the header of the file. Ninka does not search for any copyright statements in the header, I *only* use it to dump the header.

FOSSology

FOSSology is not just for licenses, it also contains a *copyright* scanner, which searches for copyright statements. The scanner catches too much on one hand, and misses copyright notices that are really really difficult to find in an automated way.

Let's not even *start* about trying to automatically find non-English copyright statements!

Example of where FOSSology fails

```
* Substantial contributions to this work comes from:
*
*    David S. Miller, <davem@davemloft.net>
*    Stephen Hemminger <shemminger@osdl.org>
*    Paul E. McKenney <paulmck@us.ibm.com>
*    Patrick McHardy <kaber@trash.net>
```

This file from the Linux kernel (`net/ipv4/fib_trie.c`) has many authors, but FOSSology only recognizes the ones at the start of the file (not listed here) that have a much clearer copyright statement, not the ones later in the header (listed here).

Homebrew script

Simple Python script that walks a source code tree and for each file:

1. dump header with Ninka
2. run FOSSology copyright extraction
3. check if all FOSSology results are in the header

Then I have to assemble/verify/cleanup results by hand.

Current version is customer specific and expects a fully configured build tree of their software. It will be cleaned up and released under the Apache 2 license.

It is far from perfect, and the whole process is incredibly frustrating and very time consuming.

Fast forward to the 21st century

Development has become much more distributed than ever before because of systems like Git.

These systems record a lot of history: if you really want you can track all changes of every line of code for most of the Linux kernel. Many people no longer bother to record their authorship statements in files as a result!

Can this (authorship) history replace copyright notices in the file? From a technical point of view it makes complete sense.

Let's discuss for a few minutes!

GPLv2 section 2 (reprise)

[...]

provided that you also meet all of these conditions:

a) You must cause the modified files to carry prominent notices stating that you changed the files and the date of any change.

[...]

Questions for the 21st century

- Can Git history replace the header information, even if GPLv2 says otherwise? What if Git is the only source of authorship information?
- How do we get the copyright information from the authorship information?
- Do I need to include every copyright holder of the file (possibly obtained from Git history) even if their code is no longer there because it has been completely replaced over time?
- How should I represent information from Git to make it an "appropriate copyright notice" (note: Git records authorship)?
- Can a copyright holder still demand that all the information is added to the header when I redistribute the code?

Let's come back to these questions in a minute.

My reality as a compliance engineer

Often I do not get a Git tree: I get raw dumps of build trees with no Git metadata. I would first have to map the files back to information in Git. If there are changes to the file this is difficult.

For now I have to use my "stone age" methods until I have a way to easily, efficiently and cheaply determine the origin of (possibly modified) files.

My ideal situation: I upload a file to a service and I get the whole list of known Git hashes that were used for the file.

Now let's go back to the questions.

Questions for the 21st century

- Can Git history replace the header information, even if GPLv2 says otherwise? What if Git is the only source of authorship information?
- Do I need to include every author of the file (obtained from Git history) even if their code is no longer there because it has been completely replaced over time?
- How should I represent information from Git to make it an "appropriate copyright notice"?
- Can a copyright holder still demand that all the information is added to the header when I redistribute the code?
- YOUR QUESTION HERE

The floor is open!

Contact

Any questions? Feel free to contact me!

- armijn@tjaldur.nl
- http://www.tjaldur.nl/
- Binary Analysis Tool: http://www.binaryanalysis.org/