

Open code in Closed Environments

Challenges of non-open use of open code

Michael Hirsch, Ph.D.

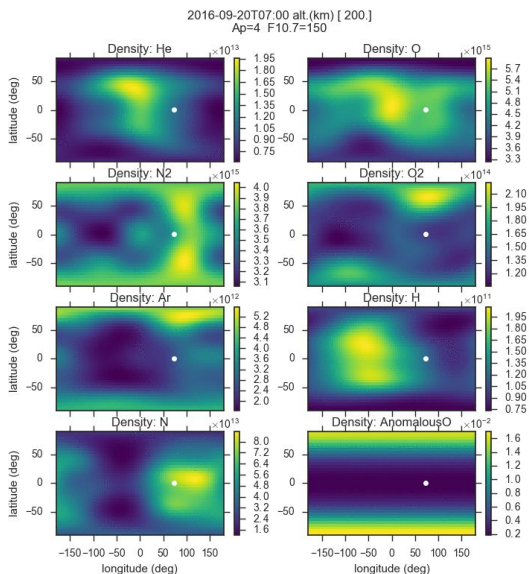
17 Jan 2018

This presentation represents the individual views of the authors.

Thanks to David F. Crouse for discussion and insights

Background (1)

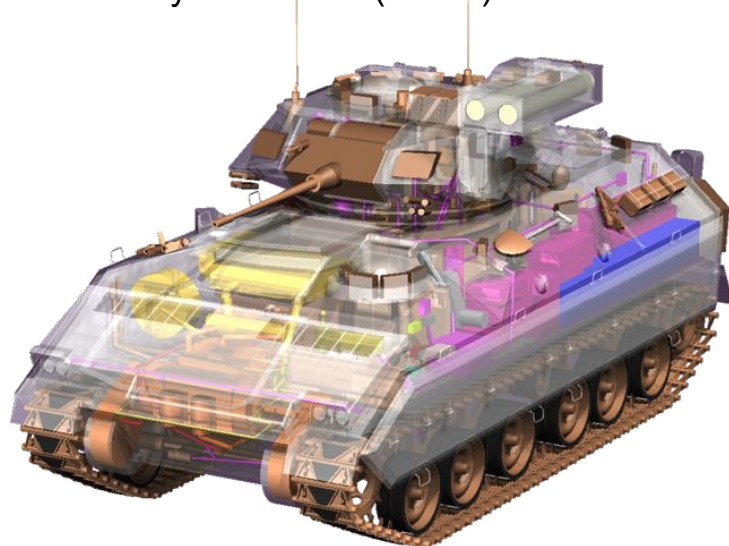
- Some spaces are closed by definition
- Generic code brings opportunity for “sharing back”



NRL MSIS (public domain):
Fortran 77

Open source community:
Python and Matlab
[https://github.com/scivision/
msise00](https://github.com/scivision/msise00)

Army BRL-CAD (LGPL)



Background (2)

- Over 680 government agencies identified on Github
<https://government.github.com/community/>
- 288 of these are US governmental entities
- 15 of these are US Military and Intelligence

I added 4 to the list recently, so there are probably more not listed.

Background (3)

US Federal Policy includes (<https://sourcecode.cio.gov/>):

- US CIO memorandum M-16-21 (August 2016)

“Each year, the Federal Government spends more than \$6 billion on software through more than 42,000 transactions”

“This policy also establishes a pilot program that requires agencies...to release at least **20 percent** of new custom-developed code as Open Source Software (OSS) for **three years**, and collect additional data...to gauge the performance of this pilot”

IP protection concepts

- **Strict copyleft licenses (e.g. GPL) Copyright + Contractual IP Protection**
 - GPL requires source code of your program be provided to its users
 - Example: Send private binary + code to an affiliated department. Program exists in closed spaces only with all source available
- **Binary + copyleft library usages (LGPL)**
 - Private binary distribution + LGPL library
 - Example: Send contractor *.exe, they have dynamically linked LGPL library. If you include LGPL library, must send library source code too.
- **Cloud/SaaS (Affero GPL)**
 - Affero GPL cloud app requires users have source code
 - Example: private cloud app

Closed/open code intermingling

GPL Taint

- Corporations vs. GPLv3 and Affero GPL
- Affero GPL: fights cloud GPL workaround
- Mix of licenses used (GPL, BSD/MIT among most popular)
- Dual licensing allows developer to make money from private use

Example: Ghostscript (1989)

- Originally used self-made GPL-like license to avoid commercial sale
- Switched to Affero GPL (2013)
- Artifex v. Hancom (2016 -> [ongoing](#))

Indirect open code benefits

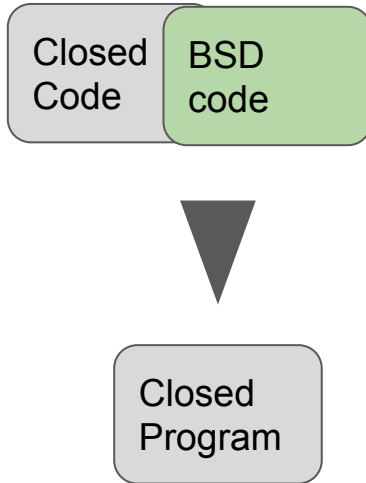
- Recruitment
- Reputation
- Retention

Avoid

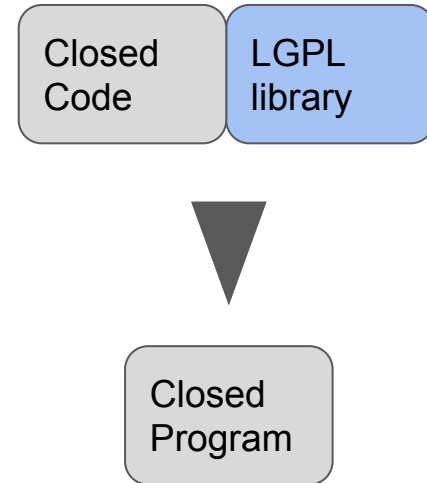
- Inventing your own license
- Not using any license
- Using “clever” uncommon license

Working Example Product: Hardware + API

Detection algorithm used (BSD)



Tracking algorithm used (LGPL)



Closed environment limitations

- Closed code, GPL library used, comfortable sharing source with all who legally have binary access
 - Can be a problem to share all source
 - May reveal capabilities/parameters that are more restricted IP
- Alternatives
 - Closed binary + LGPL library
 - Cloud

Just giving out binaries obfuscates but doesn't fully protect IP

- Stifles innovation
- Hidden indirect costs (recruitment, reputation, retention)

Further Notes

- Dynamic vs. Static linking of LGPL library
<http://www.gnu.org/licenses/gpl-faq.en.html#LGPLStaticVsDynamic>
- US Federal non-public code sharing @ DI2E <https://www.di2e.net/>
- US DoD open source guidance
<http://www.doncio.navy.mil/contentview.aspx?id=312>
- US DoD open source FAQ
<http://dodcio.defense.gov/Open-Source-Software-FAQ/>