IAAL: What Peer-to-Peer Developers Need to Know about Copyright Law

Fred von Lohman
Electronic Frontier Foundation

Presented by Eylul Dogruel

Roadmap

- Concepts and Terminology
- Important P2P cases
- Possible defenses available to developers
- 10 things you can do to reduce the risk of copyright infringement liability.

Copyright Basics

- Every novel, song, program, or other form of expression that can be captured in a tangible medium like paper, tape, hard drive is considered copyrighted.
- The owner holds the right to reproduce, distribute and publicly perform the work.
- Transmitting the file is considered distribution or potentially public performance of the work.

Definitions

- Direct Infringement:
  - e.g. End users who share copyrighted files without permission from the copyright holders are “direct infringers”
- Secondary Infringement
  - Inducement
  - Contributory infringement
  - vicarious liability

Inducement

- It means encouraging the illegal activity
  - Advertising a product for infringing uses
    - Promotion efforts to attract infringing customers also counts.
  - Instructions on how to infringe
    - Customer support records are also included in this
- Intent
  - How the company makes money or if the software could be potentially modified
  - Internal communication within the company also considered evidence

Contributory

- Knowingly contributing
  - e.g. The infringer knew about activity or should have known
  - Providing the site and facilities for the infringement
- General Purpose Tools
  - Is the item capable of noninfringing uses?
  - Discussion about proportion of infringing uses (discussed)

Vicarious Liability

- Right and ability to supervise
  - Being able to block a user is considered enough ability.
- Direct financial benefit from the infringement
  - e.g. Benefiting from attracting more users.

Napster Case – Contributory Infringement

- Napster had knowledge of the infringement
  - internal emails
  - the list of 12000 infringing files, provided by the RIAA
  - Downloading habits of top executives
- Appearance of well known song titles in promotional screenshots
- Napster failed to block access or remove the material

Napster Case – Vicarious Infringement

- Napster has the ability to control the users by being able to block them
- Infringing activity was a draw for customers.

Aimster

- Failed to introduce any legal uses.
- Activities that demonstrated clear knowledge of infringing activities
  - Tutorials that encouraged uses to download popular copyrighted music
  - Network traffic encryption: willful blindness
  - Proportion of infringing use mentioned during the talks.

Grokster Case

- Kazaa Morpheus and Grokster, Decentralized Software
- Inducing infringement
  - Advertisement aimed at attracting Napster users
  - Newsletters that contained links to documents discussing illegal use
  - Customer support messages responding to users trying to locate copyrighted media
  - Internal communications
  - Failure to implement filtering
  - Business model.

Potential Defenses

- No Direct Infringer: all my users are innocent
  - Even if the file sharing falls under fair use, hard to prove that is the case for every single user
  - Might work for a very specialized network
- Capable of substantial noninfringing use
  - Unclear if this defense holds against vicarious liability.
  - Ongoing service and community based models does not fit.

Potential Defenses

- Safe Harbors
  - Mainly established to protect ISPs
  - Online service provider must
    - Have a policy of terminating accounts of subscribers who are repeat infringers
    - Accommodate and not interfere with “standard technical measures”
    - Designate a copyright agent
    - Not know about the infringement, or should be aware from the facts
    - Not receive any financial benefits
  - Seek legal advice early

Potential Defenses

- The DMCA ban on circumvention technologies
  - Cracking the encryption on a copyrighted file
  - Building a software to crack the encryption to a file and distribute it
  - Spoofing authentication handshakes.

Tips

- Do not make and store copies in your own system
  - Even in RAM
- Do not promote, encourage or foster infringing uses
  - Check websites, promotional materials, links to third party documents,
  - instructional materials, customer support
  - Any effort to attract infringing users is also considered fostering infringement

Tips

- Either total control or no control
  - Block access when receiving complaints or chose an architecture that doesn't allow monitoring at all
  - Simple right to block users is considered enough control
  - Draw of users considered benefit for any commercial software
  - Redesign argument, so far not accepted
- Stand alone software better than ongoing service
  - Ongoing service fits the control/benefit paradigm

Tips

- Substantial noninfringing use
  - Intend your product for something other than copyrighted file sharing
- Disaggregate functions
  - Some services like search can be legally protected on its own
- Don't make money from infringing activities
- Give up the EULA
  - You still keep the rights to your software

Tips

- Beware of direct customer support
  - User communities can support themselves
- Be open source
  - You cannot be asked to implement a feature that can be removed
  - Limits financial benefit

Pros and Cons

Pros:

Well written and clearly defining the limitations of the legal knowledge given.

The tips are especially useful.

Cons:

Fair use is also a disputed aspect of the law, that can potentially affect P2P. Not much information on it.

Betamax's defense includes time-shifting broadcasting, why is this different than streaming media? (not necessarily the fault of the paper)

?

Scooped, Again

Jonathan Ledlie, Jeff Shneidman,
Margo Seltzer, John Huth
Harvard University

Presented by Eylul Dogruel

Grid

- A Type of Parallel and Distributed system that enables the sharing, selection and aggregation of resources distributed across multiple administrative domains.
  - Large scale resource sharing
  - High performance orientation
- Goals:
  - Self-configuring, self-tuning and self-healing
  - Utilize the shared storage and cycles and the middle and edges of internet
  - Focus on using computational power

P2p

- Takes advantage of resources at the end of internet
- Focus on decentralization, instability, fault tolerance, Anonymity, Redundant storage
- Instances includes
  - File sharing
  - Distributed computation
  - Anonymity

Why the communities stayed separate?

- The technical problems in Grid system are different than those in P2P system
  - Handling data is becoming in a problem in grid
  - P2P is moving toward desktop synchronization
- Difference in architectures requires different solutions
- Grid projects do not have the flexibility to try new algorithms

Shared Technical Problems

- Formation
  - Grid can benefit from the active formation of P2P
- Utilization
  - Search
  - Resource Management and Optimization
  - Scheduling
  - Load Balancing
  - Limitations of current P2P structure

Shared Technical Problems

- Coping with Failure
  - Grid needs less lossy file handling that can handle larger files
  - Redundancy might also help with correctness
  - Authentication, availability and authorization
- Maintenance
  - No standardization in P2P although efforts are there
- Disjoint Problems
  - Like anonymity

Pros and Cons

- Pros:
  - Good summary of what both fields and where they overlap
  - The point about having a P2P development that uses common tools rather than starting from scratch is a good point
- Comments
  - The grid community can build on top of an existing P2P solution, or can come up with with an API that P2P developers can implement.

?