The Anonymity (or lack thereof) of Nodes in a Peer to Peer Network

Matt Dowell [June 2016]

Peer to Peer networks might indemnify content holders, but they do not hide node information.

The concept of “Peer to Peer” sharing came about in 1999 with the music file sharing service Napster. Shawn and John Fanning founded the company along with Sean Parker to make it easy for people to share music files over the internet. Napster utilized a central server to index and categorize MP3 files. The Fannings and especially, Sean Parker went on to fame and fortune, but Napster was easily shut down (and sold) after costly court battles.

Napster’s concept was copied and expanded by several other, more general, file sharing applications including Gnutella, BearShare, Kazaa. The “second generation”[1] of file sharing utilities became more distributed, not relying on a central server but still relying on certain utilitarian servers that initiated the sharing. Some of the other weaknesses included proprietary clients, and questionable content and security.

The BitTorrent protocol was published in 2001 and soon after there were many client applications available for use on all operating systems. The initial architecture did rely on utilitarian “tracker” servers that initiated the process, but anyone could create their own public tracker servers as the specification was open and simple.

[1] http://www.symantec.com/connect/articles/identifying-p2p-users-using-traffic-analysis

"*BitTorrent is a peer-to-peer protocol, which means that the computers in a BitTorrent "swarm" (a group of computers downloading and uploading the same torrent) transfer data between each other without the need for a central server.*"[2]

When you want to download a file, you load a special file called a "torrent" file that is a pointer to the file you wish to download. The torrent file contains an address that points to a "tracker" server. Your client connects to the tracker server and it knows the IP addresses of a group of peers you should become part of called the "swarm". Your client then connects to other peer nodes in the swarm and then it asks for parts of the file you wish to download from each peer. This is an important legal distinction, the tracker server holds a map of IP addresses only, not any file data. Also, each peer might only contain parts of a file you wish to download, not the whole thing. Legally most important of all, is high-profile websites like The Pirate Bay are just search engines for Torrent files, and hold not copyrighted material at all.

To publish (and make known) a tracker server in the distributed Bittorrent network you have to have a special server that runs the required software[3] then you will have list your new tracker server address within a torrent file. There are websites that host torrents that will list your new server as a tracker for you as well.

If a company that created or distributed copyrighted content found their content was being shared on a peer-to-peer network, one way to legally fight the offenders would be to create a custom tracking server to act as a "honeypot". When a client connects to their tracking service, it requests a certain file, and hands over their public IP address. Sometimes this address is the public facing address of a larger NAT enterprise, but mostly it would be the IP address of a home broadband connection.

The peer-to-peer community has not been standing still. The glaring weakness in the Bittorrent protocol is (was) the tracker server. The tracker servers are a single point, easily

[2] http://www.howtogeek.com/141257/htg-explains-how-does-bittorrent-work/
[3] http://www.review-ninja.com/2009/01/build-your-own-torrent-tracker.html

created and used to acquire a list of IP addresses searching for your content. How can a swarm be created without an initial server pointing you to your peers? The answer is a distributed hash table.

A distributed hash table is exactly what it sounds like, a hashed key/value mapping that is distributed across multiple nodes. A typical architecture would have the nodes logically arranged in a circle. When a node wants to join the circle, an algorithm finds the best point on the circle and informs the neighbors of the new node. The new node takes some of the key/value pairs from its neighbors and is now responsible for managing that portion of the table (content). Likewise, when a node leaves the circle, its neighbors take over the responsibility for its key/value pairings.

To make the DHT efficient, each node has a routing table to other nodes that contain key/value pairs it might not. If it gets a request for a key that it might not have, it can route the request to the nearest owner of that pairing.

The Distributed Hash Table concept might remove the need for tracker servers, but it does not hide nodes in a network. For Kanye West to get a list of IP addresses that are sharing, and/or looking for ***Life of Pablo***, all he has to do is become a node in a sharing swarm. His specialized software could act like a node and keep track of the IP addresses of the other nodes.

The “overlay network” or logical ring and routing information in a P2P network is what contains the node information and when compromised, will provide the address of the destination for requests. There are variations of P2P architectures that try to eliminate that weakness by making each sender/receiver node a “forwarder” only so the request for data is forwarded to “middle men” nodes and forwarded along, so in theory it is more difficult to determine where the actual request for ***Life of Pablo*** originated from.[4]

[4] http://www.lix.polytechnique.fr/~tomc/P2P/index.html

Content companies have been compiling lists of IP address blocks along with “offending” actions (requesting a copy of copyrighted material) and filing large legal claims with requests to the ISP’s for the identification of the home users. There are several downsides to this tact. It is very expensive to file a swarm of legal claims against small offenders, and most importantly, the court has repeatedly set a legal precedent stating an IP address alone does not identify an individual.

““[A] *single IP address usually supports multiple computer devices – which unlike traditional phones can be operated simultaneously by different individuals.*”[5]

Content companies have shifted their fight from legal to convenience. They have fostered agreements with the large ISP’s that business-wise, is a win-win. In 2015 Sandvine concluded that Bittorrent accounts for “6.3% of total traffic in North America, and 8.5% in Latin America”[6] So reducing that number helps ISP’s reduce load on their networks, and content providers purportedly will see less pirating of their products. Instead of filing claims in courts, they file claims with the ISP’s. The ISP’s have more efficient ways of identifying the IP and can, without legal ramifications, just cancel or downgrade the service contract with the offending parties. The press has dubbed the new policy, the “six strikes plan”[7]. Time will tell whenther ISP’s will feel the need to act as the police force for content companies.

In summary, a client-server architecture was easily defeated in techo-legal battles because there was a single point of contention (failure) and removing the server made all the clients useless. The true peer-to-peer network came about in response to the weaknesses of the client-server architecture, but technically one peer cannot connect to another without sharing their IP address. The courts have concluded that an IP address does not identify an individual if pointed to a home network, but it does identify the owner of the service.

[5] http://blog.ericgoldman.org/archives/2012/05/new_york_judge.htm

[6] https://www.sandvine.com/pr/2015/5/28/sandvine-in-the-americas-netflix-google-facebook-the-internet.html

[7]

References

Dissecting Darknets: Measurement and Performance Analysis
Xiaowen Chu, Xiaowei Chen, Adele Lu Jia, Johan A. Pouwelse, Dick H. J. Epema
April 2014 ACM Transactions on Internet Technology (TOIT): Volume 13 Issue 3, May 2014
http://dl.acm.org.mtrproxy.mnpals.net/ft_gateway.cfm?id=2611527&ftid=1471058&dwn=1&#URLTOKEN#

The Frog-Boiling Attack: Limitations of Secure Network Coordinate Systems
Eric Chan-Tin, Victor Heorhiadi, Nicholas Hopper, Yongdae Kim
October 2011 ACM Transactions on Information and System Security (TISSEC): Volume 14 Issue 3, November 2011
http://dl.acm.org.mtrproxy.mnpals.net/citation.cfm?id=2043627&CFID=772594625&CFTOKEN=20873865

Napster Then and Now
http://iml.jou.ufl.edu/projects/spring01/burkhalter/napster%20history.html

Bittorent Protocol Definition
http://www.bittorrent.org/beps/bep_0003.html

Identifying P2P Users through Traffic Analysis
http://www.symantec.com/connect/articles/identifying-p2p-users-using-traffic-analysis

Sandvine: In The Americas, Netflix + Google + Facebook = The Internet?
https://www.sandvine.com/pr/2015/5/28/sandvine-in-the-americas-netflix-google-facebook-the-internet.html

New York Judge *Slams* Bittorrent Copyright Plaintiffs
http://blog.ericgoldman.org/archives/2012/05/new_york_judge.htm

A Survey of Anonymous Peer to Peer File Sharing
http://www.lix.polytechnique.fr/~tomc/P2P/index.html

Identifying P2P Users Using Traffic Analysis
http://www.symantec.com/connect/articles/identifying-p2p-users-using-traffic-analysis

HGT Explains: How Does Bittorrent Work
http://www.howtogeek.com/141257/htg-explains-how-does-bittorrent-work/

Build Your Own Torrent Tracker
http://www.review-ninja.com/2009/01/build-your-own-torrent-tracker.html