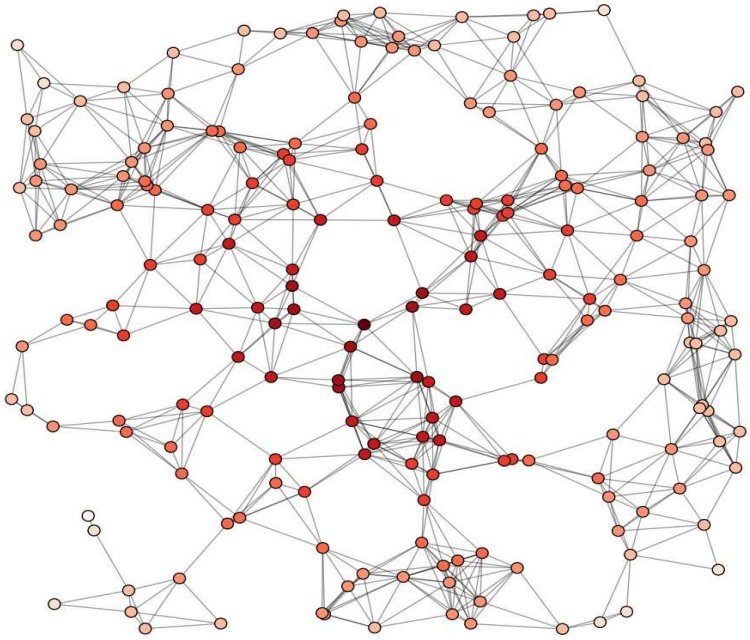
INTERNET COMMERCE SECURITY LABORATORY

Investigation into the extent of infringing content on BitTorrent networks



Robert Layton and Paul Watters

April, 2010.

Executive Summary

BitTorrent is a useful and technically advanced protocol for sharing files over a network without posing a large impact on a single server or point of contact. BitTorrent uses a distributed network, where a file is downloaded from other peers on the network, hence the term peer to peer (P2P). While there are many legitimate uses for BitTorrent, it has been used regularly to download and distributed copyright infringing material, such as movies, music and software. This report aims to take an objective look at the extent of the illegal use of BitTorrent, in order to facilitate increased awareness of the usage of BitTorrent today. To do this, the network itself is analysed to determine what files are being shared, how much each file is being shared, what is the percentage of shared files that are copyright infringing and what types of files are listed in these categories.

This report contains the following information. Chapter 1 contains information on the BitTorrent protocol relevant to the experimental design and motivations behind this research. Chapter 2 describes the research questions and methodology that was used for the experiments. Chapter 3 contains the results from those experiments. Finally, Chapter 4 discusses the implications of those results, outlines known issues with the methodology and some directions on how to overcome these issues. Chapters 1 and 2 are provided for background information to understand how the results are obtained and can be skipped for those wanting only further details on the results obtained.

Through our investigations, we found that 43.3% of BitTorrent torrents are movies, 29.1% are TV shows and 16.5% are music. Using our sample of trackers we discovered that a total of 117 million current seeds are available across more than one million torrents, based on the number of seeders available for the files. The top two files were being seeded more than one million times each and the third more than 500,000 times. In summary, our results indicate that 89% of all torrents from our sample are confirmed to be infringing copyright, both by the number of files and total number of current seeders. Of the torrents in the top three categories (Movies, Music and TV shows), there were no legal torrents in the sample. Further research into the distribution of infringing pornographic content would be required to determine a complete overall figure.

Acknowledgements

This research was conducted at the Internet Commerce Security Laboratory which is funded by the State Government of Victoria, IBM, Westpac, the Australian Federal Police and the University of Ballarat. Support for this research was provided by Village Roadshow. More information can be found at `http://www.icsl.com.au` or by contacting the authors.

Contents

Chapter 1

The Bittorrent protocol

The BitTorrent protocol (hereafter just BitTorrent) is a peer to peer file sharing protocol, allowing files to be shared to a large number of users without a drastic impact on a single server. BitTorrent is especially popular for sharing very large files, often hundreds of megabytes or larger over one's Internet connection. It was developed in 2001 by Bram Cohen and is now maintained by his company BitTorrent, Inc. BitTorrent outlines a method for users (peers) to download a single file (or a collection of files), from other peers who have that file. Tracking which peers have which files is managed by a server called a 'tracker', which is the first contact point for a new peer to join the network.

While BitTorrent, Inc. has its own BitTorrent program (confusingly, also called BitTorrent), there are numerous other clients, including open-source variants, commercial variants and web-based clients. Other clients exist, such as the version used by Blizzard which uses BitTorrent to distribute updates to their popular online role playing game World of Warcraft. This highlights one of the positive sides to BitTorrent: in using BitTorrent, Blizzard reduce the cost of updating many different clients. These updates would also cause a large strain on Blizzard's servers, as all updates usually occur within a short timeframe, meaning that there would be many updates occurring in a very short period of time.

While there are many legal uses for BitTorrent, the ability of BitTorrent to allow users to download large files has made if very popular for downloading illegal movies, music and television shows. This illegal use is popular on other peer to peer protocols as well, however BitTorrent has risen quickly to be considered the most used protocol for sharing large files. It is known that BitTorrent is widely used, estimated to account for 43% of all Internet traffic[1], even in times of an increasing use of streaming high definition video through sites like YouTube.

BitTorrent works by distributed the bulk of the downloading load from the server to the other peers in a network. As an example, a company wanting

[1] URL: `http://torrentfreak.com/bittorrent-still-king-of-p2p-traffic-090218/`

to share a 100mb file to 100,000 users can either make the file available on a server or through BitTorrent. If they store it on their server, then each user downloads the file individually by creating a new connection to the single server and downloading a new copy of the file. Ignoring ISP side optimisations such as caching, over 10 terabytes will be downloaded for that single file. To contrast, if the company makes a torrent file available, then each user will download this torrent file, usually only a few kilobytes. The users then share this file between themselves and the company, often resulting in a much lower amount of data needing to be served by the company.

The BitTorrent protocol has three main aspects. The first is the torrent file, the second are the trackers and the third is the network of peers. The torrent file is a bencoded file (see subsection 1.2) which describes which trackers the user should contact to be informed of which peers they can download pieces of the file from.

The rest of this chapter is devoted to outlining the details of the BitTorrent protocol that are necessary for understanding the methodology used in the experiments. The details are not, however, needed for understanding the results, as the shortcomings and expected error of the results are outlined in the results chapter itself.

1.1 Brief Overview

A client is a person wanting to download a file using BitTorrent. In order to do this, the search for the file to find a torrent file which contains information on how to download their file. As an example, they may use a BitTorrent search engine to find a torrent file for a movie. The torrent file itself does not contain the movie itself, only information on how to download the movie. The torrent file contains information on which file it contains information for, how many pieces are needing to be downloaded to make up the file and the SHA-1 hash[2] of each piece so that the client can ensure that the pieces they have downloaded are valid.

As well as that, a torrent file contains information on which *tracker* to use. The tracker is a server that maintains a list of all clients that have connected to the tracker, and remembers which files the clients have to download, and which pieces of each file they have. When a new client connects wanting a movie, the client calculates the info hash (see section 1.4) of the torrent and uses this as an identifier of the movie. The tracker searches its database for this identifier and returns all of the other clients that have the same file that is being requested. The tracker returns the IP address of each of these clients, called 'peers', to the new client. At this stage, the new client contacts each of these peers and requests to download a piece of the movie from them.

[2]SHA-1 is a hashing algorithm which creates an almost random string of characters from an input. The result, the hash, is the same for the same input, but usually very different if the input changes only slightly. The aim of a hashing algorithm is to create a one-way function; it should be impossible to reconstruct the input from a hash.

As the BitTorrent protocol distributes the load of downloading from one server to many peers popular files on BitTorrent, i.e. those with many peers, download much faster for a new client than the traditional server based method of downloading. The major downside is that it is necessary that every piece is available for downloading for any single client to download the entire file. To overcome this, clients who have finished downloading are often asked to stay attached to the tracker and make their completed download, with all pieces, available for new clients. These people, who have completed their download but keep the file available, are called 'seeders' or 'seeds'. Users who only download files are often negitively called 'leechers', although this term is also used neutrally to mean any user who is currently downloading a file, even if they are simultaniously uploading data as well to other users. In this report, we use the neutral meaning of this word.

1.2 Bencoded Files

Bencode is a simple format which is an abbreviation of the term 'binary encoding'. It is a method of storing simple data formats including numbers, strings, lists and dictionaries in a file. The bencoded format has the following data types:

Integers Integers (whole numbers) are represented by the encoding `i`X`e`, where X is replaced with the number being encoded. As an example, the number 42 is encoded as `i42e`.

Strings Strings (sequences of characters) are encoded using the format `<length>:<contents>`, where `<length>` is replaced by the length of the string and `<contents>` is replaced by the string itself. The string 'cat' is encoded as `3:cat`.

List Lists of values can be given using the formation `l<contents>e`, where contents is replaced by the individual items in the list, without separators between items. A list of the two numbers 12 and 32, along with the word 'dog' is encoded as `li12ei32e3:doge`, which can be read by inserting spaces between the items: `l i12e i32e 3:dog e`.

Dictionary Dictionaries map a 'key' to a 'value', the same way an English dictionary maps a word (key) to its definition (value). They are encoded using the format `d<contents>e`, where contents is a series of key/value pairs without separators. A dictionary with the first four numbers by name mapped to their numerical values is `d3:onei1e3:twoi2e5:threei3e4:fouri4ee`, read as `d 3:one i1e 3:two i2e 5:three i3e 4:four i4e e`.

1.3 Torrent files

A torrent file is a bencoded file containing a dictionary with the information necessary to download the files that the torrent file points to. The make-up of a

torrent file is slightly different depending on whether the files being shared are a single file only or a combination of many files.

For a single file, the keys and values in a dictionary are:

announce URL of tracker

info A dictionary containing:

- **name** the filename where the file is to be saved
- **piece length** number of bytes per piece, commonly $2^{18} bytes = 256 KB$
- **pieces** the concatenation of the SHA-1 hashes of each piece. Its length is in multiples of 160-bits, being the concatenation of the SHA-1 hashes (which are 160 bits each) of each piece of the file.

length the size of the file in bytes

If the torrent represents multiple files, such as a music album with each song as a different file, the torrent file contains information on each file. The torrent file then has the following keys:

announce URL of tracker

info A dictionary containing:

- **name** the directory name where the files are to be saved
- **piece length** number of bytes per piece, commonly $2^{18} bytes = 256 KB$
- **pieces** the concatenation of the SHA-1 hashes of each piece. Its length is in multiples of 160-bits, being the concatenation of the SHA-1 hashes (which are 160 bits each) of each piece of the entire set of files.

files a list of dictionaries, each dictionary corresponds to a different file, and each contains

- **path** a list of strings corresponding to the subdirectory name, the last is the actual file name
- **length** the size of the file in bytes

There are a number of extensions to this standard schema, however these are optional. One commonly used extension is the ability to have a list containing the URLs of a number of different trackers. On opening a torrent with such a list, the client connects to all trackers, which usually increases the speed of downloading. Other extensions include peer based information gathering such as DHT and PEX, which are explained in the conclusions chapter.

1.4 Info hash

An important derivative of a torrent file is its 'info hash'. This is the SHA-1 hash of the value in the torrent file corresponding to the key `info`. This is used as a shortcut in recognising the torrent in a large list, as it is possible that two torrents have the same name, but it has been proven to be highly unlikely that two different torrents would have the same info hash, even if the file that they point to is almost the same file[3].

The info hash is outputted as a 'binary-encoded' string, which often contains many non-printable characters and is also difficult to use in searches on the Internet. One method that is used to overcome this is to use a 'hex-encoding' version, which encodes the hash using a hexadecimal base, resulting in a string containing numbers and the letters A through F. Another method that is used for Internet based searches is a 'URL-quoted' string, which encodes the info hash using encoding that is commonly used for web based URLs. These types of encodings make it possible to use the info hash value of a torrent to search for and use BitTorrent search engines and trackers. The info hash is the single piece of information used by trackers to determine which torrents each client is interested in. Trackers rarely contain filename information from the torrents themselves.

1.5 Usage

From an end-user's point of view, using BitTorrent is often an easy process. First, the user must find a torrent file, which can usually be obtained by searching the Internet using a general search engine such as Google, or a specialised torrent-based search engine like The Pirate Bay. Once the torrent file has been obtained, the user opens this torrent in their BitTorrent client program, which starts the downloading process. After a while, the download is complete and the user has the file.

In the background, when a client has a new torrent to download, the client reads the bencoded information in the torrent to find which tracker or trackers the clients needs to ask for peers. The client will contact all of the trackers listed in the torrent file, and will be returned information on other peers in the network that are currently sharing this file. The client then contacts these other peers and begins downloading pieces of the shared file from them. Once a piece has been downloaded, the client checks that the SHA-1 hash of the downloaded piece matches the SHA-1 hash contained in the torrent file for that piece. If they are a match, the piece is kept for the downloaded file. If they are different, the piece is discarded and downloaded again, sometimes from a different peer. This process prohibits a 'bad user' from uploading corrupt pieces of the file to other users.

[3] If the file is exactly the same, the info hashes should be the same. If just one byte is different, the info hash is completely different.

During the download process, the client will contact the tracker again to update it on the download progress of the files and also to see if there are new peers that can be contacted. This update process usually happens once every 15 minutes or half an hour and is defined by the tracker. If a client is constantly contacting the tracker, they may be blocked from future contact, as this could be a form of attack. Often, trackers will include other metrics, such as ensuring that clients are also helping other peers to download pieces that the client has already downloaded. Trackers can have a lot of control in these situations, but what control they exert is dependent on the managers of the trackers themselves.

Further to this, there are also public and private trackers. Public trackers allow anyone to contact them and start downloading files (although some limit whether a new torrent can be uploaded to the tracker). Private trackers require login credentials and there is often a limit on who can obtain login credentials for these trackers.

1.6 Scrapes

A scrape is a method of obtaining data about a file from a tracker. In normal usage, a client will scrape a tracker to obtain information about a single torrent that the client is downloaded. The client can use this information to determine whether it is worth checking the server for new peers or determining if the tracker is active or not. A query to scrape involves the info hash of the torrent and returns the number of complete downloads, called seeds. Also returned are the number of people with an incomplete download that are still downloading (leechers) and the number of times the file has been downloaded. There are other values that could be returned, the official BitTorrent specification lists the name as an optional key value, although this is usually discarded to save bandwidth for the tracker.

Additionally to a scrape for an individual file, many trackers allow multiple scrape requests to occur for a number of files in a single request. Also available on some servers is a full tracker scrape, giving the details of all files hosted by the tracker. This full scrape is an important piece of information in determining the extent of copyright infringing files that are being shared through BitTorrent. The next chapter describes the methodology we created that uses this full server scrape to answer questions about this problem.

Chapter 2

Investigation

2.1 Research Questions

This research is motivated by the need for objective answers to the following research questions, needed to define the size and scope of copyright infringing activity using the BitTorrent protocol.

1. How many files are shared using BitTorrent and what are the categories of shared files?
2. At a given point in time, how much sharing of files is actually occurring using BitTorrent?
3. For each shared file, how many times has it been shared in total?
4. Overall, what is the number and percentage of shared files which are infringing, both by number of files and total downloads?

Obtaining an exact value for any of these questions would be impossible due to the scope and distributed nature of BitTorrent. There are thousands of BitTorrent trackers available, as well as other technologies such as Distributed Hash Tables and Peer Exchange which prohibit a complete study being performed. Instead we aimed to calculate approximate answers to each question with a very low rate of error. To do this, we sampled a large and disperse range of BitTorrent trackers and present an overview of each tracker. The methodology that was used for this is described in the following section and the results from running these experiments is given in chapter 3.

2.2 Methodology of experiments

BitTorrent is still a predominately server-based system through the use of trackers, as described in Chapter 1. To answer the research questions posed in section 2.1 a methodology was developed based on scraping servers and recording the

results of these scrapes. This provides an objective look at the usage of BitTorrent through the trackers, rather than relying on using a sampling of torrents. In order to gain reliable results, a representative sample of available trackers was needed and the procedure for sampling trackers is given in subsection 2.2.1. Once a sample of trackers was created, they were scraped according to the procedure defined in subsection 2.2.2. Once data about the available torrents has been scraped, the filenames were determined using the procedure developed in subsection 2.2.3, the results were categorised using the procedure in subsection 2.2.4 and the determination of the number of infringing files was performed using the procedure in subsection 2.2.5.

2.2.1 Sampling of Trackers

To perform the experiments needed, it was important to use the most popular trackers in use. For that reason, to sample the trackers, the trackers used were obtained from actual torrents being used. The torrents were the most popular torrents listed on the website torrentz (`http://www.torrentz.com/`) as of April 21st, 2010. All trackers listed for the top 10 most popular torrents were used.

2.2.2 Scraping Trackers

Once a list of trackers has been created, they were scraped for their peer information. This scrape is downloaded in a similar method that a normal HTTP download occurs. If the download is interrupted, the scrape is not tried again in that scraping iteration. An interrupted download is still useful, as it contains valid scrape information up until the end of the downloaded portion. As an example, if a scrape is interrupted after downloading 80% of the file, there is still 80% of the scrape information available.

When parsing the scrape data, the consistency of the file is not verified to ensure that information can be gathered from interrupted downloads. Rather any valid data for each file is collected and saved into a database. The collected information from this procedure is the following for each file that the tracker has listed:

Info Hash : The info hash as described in subsection 1.4.

Downloaded : The number of times the file has been completely downloaded.

Complete : The number of seeders on the network at the given time.

Incomplete : The number of leechers on the network at the given time.

It is our understanding from observing the data that the downloaded number is usually faked and is therefore not considered in our results. As an example, many of the trackers we retrieved data from indicated that all files had been downloaded 10 times, even when the number of current seeders was in the thousands. Obviously this is impossible (as a seeder is someone who has completely downloaded the file), so we consider this data faked. For some trackers, this

number varies and could likely indicate the true value. However for the purposes of maintaining integrity in the results, we do not use this value for any listed tracker. Instead, we use the 'complete' count as our number of downloads, as a file must be fully downloaded by a user for that user to be listed as a seeder.

As the procedure that calculates the info hash is a one-way function, we cannot recreate the filename from the scrape data alone. In order to find out the filenames that correlate with the info hashes scraped in this step, the procedure in the next subsection is used.

2.2.3 Determining Filenames

With the scrape information from each of the trackers, a large list of info hashes is available. There is no algorithmic way to determine the filename from its info hash making it impossible to do without an external data source.

To determine the filename, we use a combination of a BitTorrent search engine and Google. The procedure starts by searching the BitTorrent search engine for the info hash that has been hex encoded. If the BitTorrent search engine has the torrent that generated this info hash, it will return the torrent, including the name of the files that are in it. We parse the search results to extract only the filename and store the resulting filename in the database. If this procedure fails, we perform a Google search for the hex encoded info hash. If there are results from Google, we search them in order of appearance. If the title of the search result (the title of the corresponding webpage found by Google) includes the hex hash, it is ignored, as many websites repeat this value in their title, giving a non-result. If the hex hash is not in the title, we use the title as our filename result. This filename is likely to be 'dirty', as the title of the search result is likely to contain other information such as the name of the website linked to by the Google search. A full parsing of the returned results is a problem for automatic parsing and is not considered in this methodology.

To determine the accuracy of these filename results, the results were validated by performing a reverse lookup. The top 50 seeded torrents with filenames were used and a random sample of 50 torrents from the full set of named torrents was also used. For each of these 100 torrents, the original torrent file was searched for using the given info hash. The torrent file was downloaded and the info hash re-calculated to verify that the torrent is correct. This sampling method was chosen to ensure that both the most seeded torrents were considered, as well as to ensure that there were no biases between the top torrents compared to the rest of the named torrents.

After the above procedure, many of the info hashes are correlated with filename. In order to answer the proposed questions, we must still determine the category a file should be listed as and also determine if the file is infringing. In the next subsection, we outline the procedure for determining the category of a file.

2.2.4 Categorising

Determining the category of a file, once we have its filename, is easier for some files than others. Most movies are of the form:

`<movie-title> (<year>) <release-group>`

in which fields can be separated by spaces, periods or other characters. This format changes a little bit as well between release groups and sometimes is a different format altogether. Another common pattern is:

`S<season-number>E<episode-number>`

which is used for TV shows to indicate which episode is available. An example of this would be `The.Simpsons.S10E04`, to indicate the forth episode of the tenth season of The Simpsons TV show.

To automatically categorise, we use a simple rule based system. A list of patterns, in the form of regular expressions, are listed along with the category they correspond to. The full list of all rules used is given in appendix A. The rules are listed in the author's view from the most accurate to the least accurate. To categorise a rule, each rule in order is applied to the file. Once a rule is triggered, which happens when the filename contains the pattern given by the regular expression, the file is given the category from the rule and the procedure stops.

To validate the results, the top 500 torrents by seeders and a random sample of 500 torrents is taken and these categorisations are manually verified. Further to this, the percentage of torrents that are classified, the coverage, is calculated to give an overall value on the percentage of all torrents that are categorised correctly.

This procedure was applied to all filenames that were retrieved using the methodology explained in the previous section. To determine whether a file is infringing, we use the sampling method described in the next section.

2.2.5 Determining Infringing Copies

Once the torrents have been assigned filenames, we must then determine what percentage of torrents are infringing copyright. To do this, we sample 1000 torrents with filenames and determine, for each file, whether they are infringing copyright. If we are unsure, we error on the side of caution and guess that it is legal.

2.3 Torrent Monitoring Program

At the ICSL, we have developed a system to perform this methodology. The system currently scrapes trackers, creates database entries for each torrent and download information. We can then gather statistics needed for answering these and other research questions relating to BitTorrent use. The system is able to be extended to meet new challenges and was developed to be able to do this easily for future research.

Chapter 3

Technical Results

3.1 Trackers and Scrape Collection

To create this list of trackers, the top 10 torrents on the website Torrentz[1] were downloaded and the trackers that were listed for each of the torrents were collected. It was found that most torrents used similar trackers and despite each torrent having at least 10 trackers associated with it, there were only 23 unique trackers. In no specific order, the trackers used for this research are:

- `http://inferno.demonoid.com:3395/announce`
- `http://tracker.packy.se:2710/announce`
- `http://tracker.mightynova.com/announce`
- `http://tracker.torrentbay.to:6969/announce`
- `http://p2p.lineage2.com.cn:6969/announce`
- `http://sombarato.org:6969/announce`
- `http://tracker.openbittorrent.com:80/announce`
- `http://kubanmedia.org:2710/announce`
- `http://bt1.the9.com:6969/announce`
- `http://bt.rghost.net/announce`
- `http://tracker.bitreactor.to:2710/announce`
- `http://tracker.mightynova.com:4315/announce`
- `http://free.btr.kz:8888/announce`

[1] `http://www.torrentz.com`

- http://tracker.torrent.to:2710/announce
- http://tracker.irc.su:80/announce
- http://www.desidhamal.com:6883/announce
- http://tracker.prq.to/announce
- http://linuxoid.in:4443/announce
- http://tracker.ilibr.org:6969/announce
- http://tracker.ilibr.org:80/announce
- http://tracker.hkreporter.com:6999/announce
- http://idowns.org:6969/announce
- http://tracker.desi6.com:7979/3paz2gybgymgo6aub7e9d3u15784ubfx/announce

Of the trackers listed 19 scrapes were recovered, of which 2 were unintelligible. Some of these scrapes were partial scrapes only, with only some information being retrieved. The following trackers did not allow a full server scrape to occur:

- http://tracker.mightynova.com/announce
- http://sombarato.org:6969/announce
- http://tracker.bitreactor.to:2710/announce
- http://tracker.mightynova.com:4315/announce

While it could be seen as a sign of illegal activity, there is a legitimate reason for not allowing a full server scrape, as a full server scrape encompasses a large amount of bandwidth. A smaller tracker may wish to minimise their bandwidth usage by disabling this feature. For this reason, we will no longer discuss these servers in the report. Two trackers returned invalid scrapes, which we were unable to parse to gain any information at all.

3.2 Filename Determination

The above method for gaining information about different torrents resulted in a very large number of results. There were a total of 1,046,713 different[2] sets of torrent information were retrieved from the above servers. To determine every single one of these torrents would of been time prohibitive, resulting in a descending sampling method being used. Based on observation about the downloading distribution, the authors hypothesised that just 20% of torrents were responsible for 80% of all downloads. To test this, we ranked all of the torrents

[2]Based on different info hashes

by their available seeders information and determined that we had underestimated this imbalance. For the results we collected, just 4.0% of torrents, a total of 15367, were responsible for 80% of seeders. Further to this, just 9.9% of torrents, just 38365, were responsible for 90% of seeders. This new information drastically reduces the number of times the naming procedure must be run to achieve a high accuracy for our future predictions. For this reason, all results are sampled at a descending sampling based on the number of times the file was currently being seeded.

For the filename determination, each torrent was retrieved from our database in order of highest current seeds. The filenames were determined for the torrents in descending order by number of current seeds reported. For our results, there were 151,268 attempts to determine the filename, of which 121,684 succeeded and resulted in a filename being given. This gives the filename determination script an 80.4% chance of returning a filename. The torrents which were assigned filenames, account for 99.36% of all seeders listed. There were no failed filename determinations in the top 50 most seeded torrents, with the first occurring in rank 68 and a total of 6 in the top 100. In the top 1000, there were 119 failed filename determination attempts. A validation on the top 50 torrents and a random 50 torrents was performed using the methodology given in subsection 2.2.3 which resulted in all torrents being correctly named, where a name was given.

3.3 Categorising

The categorisation was performed using a set of manually derived rules, found in appendix A. Categorisation was performed on the top 15,367 torrents, to account for 80% of all current seed population. Of these torrents, 10,741 were categorised, giving a coverage of 69.9%.

After applying the categorisation, the categories were manually verified for two samples, the top 500 torrents and a random sample of 500 torrents. The classification accuracy achieved was 98.8%, with only 6 entries given incorrect categorisations. The percentages of files in each category are given in table 3.1.

The incorrect entries, along with the rule that caused their categorisations, are given in table 3.2. Of the six errors listed, 4 are caused by the rule 'XVid' which normally indicates that the torrent is a movie with a usually high accuracy. The AVS Video Editor is a commercial program which needs a 'crack' to run without paying, and this procedure is normally used for pirated video games. The High Stakes Poker file indicates that the Season/Episode rule should be above the XVid rule, as this file was marked as a movie, despite having the season episode information in the filename. Finally, the last line is for the movie 'Zack and Miri Make a Porno', which is classified as porn because it has the word Porn in its title. Apart from the errors listed above, all of these rules are usually accurate. A more detailed rule based system could account for this small number of errors, however the system used currently has a 98.8% accuracy and 69.9% coverage. The high precision indicates that this approach works but the

Category	Count	Perc%
Anime	9	0.1%
Book	22	0.2%
Child Porn	4	>0.0%
Documentary	7	0.1%
Game	477	4.4%
Hentai	4	>0.0%
Movie	4651	43.3%
Music	1775	16.5%
Pictures	18	0.2%
Porn	400	3.7%
Software	252	2.3%
TV Show	3122	29.1%
Total	10741	100%

Table 3.1: Percentage of files categorised with each category.

Filename	Classification	Category	Rule
`Alicia Keys - As I Am [2007][CD+SkidVid_XviD+Cov] 192Kbps`	Movie	Music (Video)	'XVid'
`katharine mcphee-had it all -dvdrip-x264-2009-mv4u-(0001).mkv`	Movie	Music (Video)	'XVid'
`Lily Allen- It's Not Me It's You [2009][CD+SkidVid_XviD+Cov]`	Movie	Music (Video)	'XVid'
`AVS Video Editor v4.2.1.166 + Crack SETUP [ResourceRG Apps by Lop`	Game	Software	'Crack'
`High.Stakes.Poker.s06e03 .PDTV.XviD-TH.avi`	Movie	TV Show	'XVid'
`Zack And Miri Make A Porno . DVDrip(CanusRG-pill)`	Porn	Movie	'Porn'

Table 3.2: A sample of the incorrect categorisations.

coverage needs to be extended in further work in this area.

Investigating the 30.1% of torrents that are not classified, most of them are simple filenames which could not be automatically categorised without a context aware search. As an example, the torrent file `'Blackadder complete'` was not automatically categorised. Without being aware that Blackadder is a TV series and that `complete` contextually refers to the torrent containing all released episodes of the show, it would not be possible to create a pattern for this type of file. Such a context aware search could potentially be done using a Internet based search or a given list of known movies, TV shows and music artists.

For the uncategorised files, a sample of 100 files were manually classified. Of those files 55 were movies, 26 were music, 10 games, 5 were software, 1 was TV and 3 were unknown. This is a slightly different distribution from the categorised filenames, indicating that there are categories which are more easy to create rules for than others. As an example, the rule $s\backslash d+\backslash W*e\backslash d+$ is very good at classifying individual episodes of TV shows as this is a very common rule: S03E06 indicates the sixth episode of the third season of a TV show, which is commonly used. This is one of the causes of the low number of unrecognised TV show torrents. Movies are more difficult as there is no universal way to indicate that a torrent is a movie and the problem is even worse for software. Often these torrents have just the name of the file and sometimes the year it was released. Without contextually aware information, its hard to tell whether `Indiana Jones` refers to the movie, the video game or even a scanned copy of one of the books.

3.4 Infringement

To determine the percentage of infringing and non-infringing files from the torrents listed, a sample of 1000 random torrents that have been assigned filenames were selected. This sample comes from the most active seeded files which were the ones given filenames. These were manually checked to determine if they were infringing or legally allowed to be distributed.

In formal terms, the classification task is to identify cases which are copyright infringing (True Positives) as opposed to cases which are legitimate (True Negatives).

Of the 1,000 torrents in the sample, we found that:

- 3 cases were confirmed as being non-infringing (True Negatives), or 0.3%
- 890 cases were confirmed as being infringing (True Positives), or 89.0%
- 16 cases were ambiguous; we could not determine if they were infringing or not
- 91 cases related to pornographic torrents. Cases in this category were not classified further, as the provenance of many files was unclear. For exam-

ple, many files were tagged as amateur (suggesting no copyright infringement) but further inspection revealed that they were in fact infringing.

In summary, we can say that 89% of torrents in the sample were definitely copyright infringing, and that 0.3% of torrents were definitely not copyright infringing. Of the torrents in the top three categories (Movies, Music and TV shows), there were no legal torrents in the sample.

Overally, 89.3% of torrents were given a definitive legality. Of those 89.3% of torrents, 99.66% are infringing. If we assume that all of the 16 cases of ambiguous legality are not infringing, we arrive at an overall figure of 97.9% infringing content shared over BitTorrent networks. Further research into the disambiguation of pornographic torrent metadata is required to verify if the pornographic material has a similar distribution of infringing material. The overall range for the proportion of our sample being infringing is therefore between 89%, if none of the pornographic material is infringing, to 98.1% if all of the pornographic material is copyright infringing.

We note that these results may vary from sample to sample, and from time to time, and the ICSL is in the process of continuously updating its database and tools. This includes building more robust tools for classifying pornographic content as infringing or not.

Chapter 4

Conclusions

From the technical results given in the previous chapter, we are now able to derive informative answers to the research questions originally posed in section 2.1. Each of these questions are now answered in the following sections.

4.1 Information on Shared Files

How many files are shared using BitTorrent and what are the categories of shared files?

From the BitTorrent trackers we obtained, we can show that at least a million different files are being shared on BitTorrent for the 17 BitTorrent trackers we monitored that returned usable scrape results. This number is expected to increase at a lower rate with more trackers included. It would be impossible to determine a complete value, as there are a large number of BitTorrent trackers and some are private.

For categories, our automatic categoriser is able to categorise torrents at 98.8% precision with 69.9% recall. This means that it is able to categorise 69.9% of torrents, and when a category is given it is 98.8% accurate. From this procedure we arrive at the following percentage of torrents:

Anime 0.1%

Book 0.2%

Child Porn >0.0%

Documentary 0.1%

Game 4.4%

Hentai >0.0%

Movie 43.3%

Filename	Current Seeds
The Incredible Hulk[2008]DvDrip-aXXo97065494792.4447	1112628
Indiana Jones And The Kingdom Of The Crystal Skull[2008]-aXXo	1029695
College[2008]DvDrip-aXXo339166021846.017	509576
Sherlock Holmes (2009) DVDSCR XviD-MAX	479655
Avatar (2009) PROPER TS XviD-MAX889790305026.795	332665
Meet Dave[2008]DvDrip-aXXo	311894
Lady GaGa - The Fame Monster 2CDRip 2009 [Cov+2CD][Bubanee]	308117
The Andromeda Strain[2008]DvDrip-aXXo	284221
Shutter Island (2010) R5 DVDRip XviD-MAX851029283088.936	282628
2012 (2009) R5 DVDRip XviD-MAX883775626338.402	277043

Table 4.1: Maximum avaialble seeder counts for the top 10 most seeded torrents from our sample.

Music 16.5%

Pictures 0.2%

Porn 3.7%

Software 2.3%

TV Shows 29.1%

4.2 Current Extent of File Sharing

At a given point in time, how much sharing of files is actually occurring using BitTorrent?

For the trackers that we scraped, we recorded a minimum of 117,420,061 current seeds. This value is calculated by determining the highest available seeder count for each torrent from any tracker that was scraped. This was needed as many users will use more than one tracker, which would register more than one peer complete event for the same torrent. This value removes this duplication, leaving the lower bound on the number.

4.3 Sharing Information for Each File

For each shared file, how many times has it been shared in total?

Through our investigations, we scraped information for more than one million torrents. The top 10 most seeded torrents are listed in table 4.1. Of note is that all of the top 10 are infringing torrents; this pattern is mostly the same for the top 100 as well. The top 100 most seeded torrents are given in appendix B, and the full list can be obtained through the authors.

4.4 Extent of Infringing Files

Overall, what is the number and percentage of shared files which are infringing, both by number of files and total downloads?

Through our investigations we discovered that 97.9% of non-pornographic files were infringing copyright. There is also a clear trend that more popular torrents are infringing. There is only one legal torrent in the top 100 listed in appendix B, an open source program VLC player which uses BitTorrent as a distribution method.

Information on more than one million torrents were collected through our investigation, however there is a clear skew towards the most seeded torrents. Just 4.0% of torrents, a total of 15367, were responsible for 80% of the current seed population and 9.9% of torrents, just 38365, were responsible for 90%. Despite this, we gave names to more then 120,000 of the top 150,000 most seeded torrents, accounting for 99.36% of all seeders. This means that our 97.9% infringing figure is applicable to both the overall percentage of infringing files *and* total seeders. Further to this finding, there were no legal torrents in the sample for the top three categories (Movies, Music and TV shows).

4.5 Future Research Paths

There are multiple avenues for future research in this area. The most pressing is the switch that is already underway by large BitTorrent sites such as The Pirate Bay in moving away from the tracker based model. These methods are Distributed Hash Tables (DHT) and Peer Exchange (PEX). The use of these technologies is a response to the lawsuits that have been brought about against the operators of larger BitTorrent trackers in order to allow them to continue to help users share files, without the litigation problems that running a tracker encompasses. Developing an extension to the methodology given in this paper to allow for determining the above statistics in a DHT enabled environment compose a large challenge but not impossible due to the nature of peers to freely make this information available.

Another avenue is the automatic labelling of files, both by category and their legality. The video sharing website YouTube uses a content management system called ContentID [1] to determine if a new video uploaded is a recognised infringing copy of a copyrighted file. This system could be extended toward the automatic categorisation of downloaded files, but suffers from the problem that the files must be downloaded in order to check them. This would be a huge use of resources to download the millions of files being shared. Instead, it may be possible to download only necessary portions of the file and determine if it is infringing from a few small pieces. Such an extension to ContentID would overcome this problem.

The method we used in these investigations achieves a 98.8% precision and 69.9% recall based on a simple ruleset. Through extended research a better sys-

[1] `http://www.youtube.com/t/content_management`

tem that achieves a higher recall, while maintaining the already high precision, could be developed. This would provide a smaller range for estimates in the overall infringement percentage.

Appendix A

Categorisation Rules

The following rules were derived from expert knowledge and are applied in order. The first term before the space is a regular expression[1], which is fired if there is a match for this regular expression anywhere in the filename. The rest of the line is the category that is applied if the rule is fired, which is applied without case sensistivity (`I` is the same character as `i` for these rules). The first firing rule being applied, and if no rules apply to the filename, then no categorisation is given.

```
# high accuracy rules
pc game Game
preteen Child Porn
underage Child Porn
XXX Porn
porn Porn
hdtv TV
dvdrip Movie
dbrip Movie
pc-dvd Game
documentary Documentary
pcdvd Game
soundtrack Music
fucks Porn
aXXo Movie
xvid Movie
discography Music
CDRIP Music
# S03E45
s\d+\W*e\d+ TV
orgy Porn
```

[1] A regular expression is a string searching method that can search for very complex patterns, such as addresses, within a large body of text.

```
bondage Porn
HDTV TV
nintendo Game
xbox Game
playstation Game
# a famous TV decoding group
xvid-lol TV
pdf Book
microsoft Software
photoshop Software
wii Game
psp Game
music video Music
hardcore Porn
wallpapers Pictures

# medium accuracy rules
sex Porn
milf Porn
hentai Hentai
Anime Anime
erotic Porn
horny Porn
season TV
bangbros Porn
nude Porn
\brape\b Porn
windows Software

# lower accuracy rules
pc Game
cd Music
song Music
# Blowjob, etc. Low accuracy
job\b Porn
\btour\b Music
xvid Movie
mp3 Music
crack Game
adult Porn
amateur Porn
\bhits\b Music
site rip Porn
lolita Child Porn
```

Appendix B

Top 100 Most Seeded Torrents

The following tables contain the top 100 most seeded torrents from our investigations. Some of the filenames have been slightly altered for formatting reasons, however no information has been changed within them (the exact list is available from the authors). For the few entries that could not be assigned a filename, the phrase `<unknown>` appears instead.

Filename	Current Seeders
The Incredible Hulk[2008]DvDrip-aXXo97065494792.4447	1112628
Indiana Jones And The Kingdom Of The Crystal Skull[2008]-aXXo	1029695
College[2008]DvDrip-aXXo339166021846.017	509576
Sherlock Holmes (2009) DVDSCR XviD-MAX	479655
Avatar (2009) PROPER TS XviD-MAX889790305026.795	332665
Meet Dave[2008]DvDrip-aXXo	311894
Lady GaGa - The Fame Monster 2CDRip 2009 [Cov+2CD][Bubanee]	308117
The Andromeda Strain[2008]DvDrip-aXXo	284221
Shutter Island (2010) R5 DVDRip XviD-MAX851029283088.936	282628
2012 (2009) R5 DVDRip XviD-MAX883775626338.402	277043
Nirvana - Discography9843381055.49025	263315
The Men Who Stare at Goats (2009) R5 DVDRip XviD-MAXSPEED	254286
LimeWire PRO 4.18.8.1	238072
From Paris with Love (2010) R5 DVDRip XviD-MAX489229147326.545	234916
The Book Of Eli 2010 TELESYNC H264 AAC-SecretMyth (Kingdom-Relea	217049
Legion (2010) R5 DVDRip XviD-MAX559343324207.015	212854
Queen - Discography525461962058.291	212619
Zombieland (2009) R5 DVDRip XviD-MAX	211317
Next Avengers-Heroes Of Tomorrow[2008]DvDrip-aXXo	199894
Ninja Assassin (2009) DVDRip XviD-MAX375149626046.111	195660
District 9 (2009) DVDRip XviD-MAX	193568
Alicia Keys - The Element Of Freedom (Deluxe) CDRip 2009 [Cov+CD][Bubanee].rar	189797
Daybreakers (2009) DVDSCR XviD-MAX297214896957.329	185235
Law Abiding Citizen (2009) DVDRip XviD-MAX575548696431.369	179429
Gorillaz - Plastic Beach [2010-MP3-Cov][Bubanee]	174760
The.Lovely.Bones.2009.DVDSCR.XviD-Lynks-PrisM	171729
The.Princess.And.The.Frog.DVDRSCREENER.XviD-MENTiON.avi	171638
The Blind Side.2009.DvdScr.Xvid 1337x-Noir	162618
The.Twilight.Saga.New.Moon.2009.DVDRip.XviD-NeDiVx761950398497.179	162040
Boy A[2007]DvDrip	160925
Michael Jackson - Black Or White (1991) [DVDRip-SyNtEr] [Subs] [	155427
Michael Jackson This Is It (2009) DVDRip XviD-MAX	153963
The.Book.of.Eli.2010.TS.XviD-IMAGiNE.avi	152952
The.Pacific.Pt.I.HDTV.XviD-SYS.avi	151327
Ahead Nero v7 5 9 0 Multilingual Incl Keymaker-EMBRACE	143525
The Hurt Locker DVD eng 2008 xivid [switch]	143077
Surrogates (2009) R5 DVDRip XviD-MAX	141448
House MD Season 3	136795
Michael Jackson - Thriller [DVD-Rip] [Subs] [AVI] [POP-USA] [AC3	134944
Bigfish Games - Mystery Case Files - Ravenhearst + Crack	134748
Metallica - Discography - Mega Collection	134621
Michael Jackson - Remember The Time [1992] [DVD-Rip-SyNtEr] [AC3	133459
Old.Dogs.DVDRip.XviD-DiAMOND.avi	133150
Percy Jackson and the Olympians (2010) R5 DVDRip XviD-MAX455829186426.8	132043
House MD Season 2644074056414.786	130852
WALT DISNEYS [ALICE IN WONDERLAND][DVDRIP][ENG]-kidzcorner	129513
House MD Season 1470432445622.182	127860
The.Children.Of.Huang.Shi[2008]DvDrip-aXXo	127281
Pussycat Dolls - Doll Domination (Deluxe Edition) (2008) and bonus disc s-srg mrsidhq	126380
Weeds - Season 4 HDTV	125411

Table B.1: Top 50 most seeded torrents from our investigations.

Filename	Current Seeders
Planet 51 (2009) DVDRip XviD-MAXSPEED	123636
Michael Jackson - Smooth Criminal [1988] [DVD-Rip-SyNtEr] [Subs]	123449
Michael Jackson - Bad [1987] [DVD-Rip-SyNtEr] [Subs] [AVI] [POP-	122108
Sherlock Holmes DVDSCR AC3 - IMAGiNE[ExtraTorrent]	121383
Michael Jackson - Billie Jean [SyNtEr] [AC3] [DVDRip]	118125
Eminem Presents - The Re-Up	117936
Up In The Air (2009) DVDRip XviD-MAX136331498949.959	117902
Michael Jackson - Earth Song (1995) [DVDRip-SyNtEr] [Subs] [AC3]	116190
Michael Jackson - Scream (1995) [DVDRip-SyNtEr] [Subs] [AC3]	113245
Worms Armageddon - FULL ISO	113105
VLC Media Player 0 9 2 NEW RELEASE!!! (September 15th 2008) Legal	112743
Windows Traktor DJ Studio v3	112677
Weezer album discography	112281
The Red Hot Chilli Peppers - Discography782914009744.44	111730
Michael Jackson - Heal The World [SyNtEr]	111628
Couples Retreat.2009.DvdRip.Xvid (1337x)-Noir no rar	111420
<unknown>	110686
Michael Jackson - The Way You Make Me Feel [1987] [Subs] [AVI] [	109817
The Twilight Saga New Moon 2009 HORROR TS-Scr DivX nEHAL	109446
<unknown>	109018
The Boondock Saints II All Saints Day (2009) DVDRip XviD-MAX	108778
Sim City 4 Deluxe Edition [ISO] - By Bobjba	107129
Funny People (2009) DVDRip XviD-MAX	104449
<unknown>	102440
Rihanna - Rated R CDRip 2009 [Cov+CD][Bubanee]	102304
The Hurt Locker (2008) DVDRip XviD-MAX	100945
Music Johnny Cash - Live at Montreux - 1994 - TV recording	99451
Tooth.Fairy.R5.LINE.XviD-MENTiON.avi	99052
The.Fourth.Kind.2009.DVDSCR.XviD-SilentNinja270618989809.115	98634
How.I.Met.Your.Mother.S05E12.HDTV.XviD-NoTV.avi	96122
Elvis Presley Discography by Nogueira neto By Mega Seeders JP	95932
Avatar 2009 DVDScr H264 AAC-SecretMyth (Kingdom-Release)	94781
GLADIATOR[2000]DvDrip-GHZ	94568
Inglourious Basterds (2009) DVDRip XviD-MAX959480928538.983	93268
Air58437913118.6827	92039
Pink Floyd - Dark Side of the Moon - Live at Earls Court	90494
Bob Dylan DVD Compilation 7 Letterman Grammy and more	90082
Transformers 2 Revenge Of The Fallen DVDRip XviD-MAX	89935
Young Jeezy - The Inspiration	89020
No.Direction.Home.Bob.Dylan.2of2.XviD.AC3	88061
No Direction Home Bob Dylan 1of2 XviD AC3	87972
<unknown>	85899
<unknown>	85803
Requiem For A Dream [DVD-Rip ENG]	85186
Breaking Benjamin - Discography - 2007 !!!	83482
Ultimate Avengers[Double Feature][2006]DvDrip-aXXo	83275
<unknown>	83230
Avatar TS XviD-IMAGiNE(No Rars)	82977
WALT DISNEYS PINNOCHIO[DVDRIP][ENG]-kidzcorner	82959
The.Fantastic.Mr.Fox.DVDSCR.XviD-DONEDEAL.avi	82881

Table B.2: The 51st to 100th most seeded torrents from our investigations.