BitTorrent Monitoring and Statistics

Research Project Proposal

Hidde van der Heide
hidde.vanderheide@os3.nl

August 15, 2012

UNIVERSITEIT VAN AMSTERDAM

Master System and Network Engineering

1 Abstract

We developed a monitoring framework able to monitor the BitTorrent protocol from a clients perspective and an analysis method to interpret the results. We focused on the number of peers in a particular swarm and if we can reliably measure the distribution of peers over ISPs. These results are used to measure the effect of blockades of sites forced on the ISPs.

The monitoring daemon is Python code that interfaces with the libtorrent BitTorrent library. We modified its default behavior where possible to maximize the number of peers that could be found in the swarm. The peers, torrents and their relations are stored in a relational database for analysis. The analyzer is build on top of the Django web-framework that visualizes different aspects of the protocol.

The analyzed results were used to validate previous research done by Brein and the UvA. We also tried to validate the claim that blocking torrent sharing sites has a significant impact on the number of peers in the swarm for the affected ISPs. We found that it is impossible to make reliable claims without sophisticated monitoring and a deep understanding of the BitTorrent protocol.

Contents

2 Introduction

In January 2012 Brein, a dutch anti-piracy organization, won a preliminary criminal case against two dutch Internet Service Providers (ISPs) forcing them to block access to the Pirate Bay (TPB). The Pirate Bay is a website that shares torrent files used to share files with the BitTorrent protocol. According to Brein TPB was mainly used to share copyright infringing material which is considered a criminal offense in The Netherlands.

Brein collected the evidence on the popularity and the distribution of peers over the Dutch ISPs itself. The evidence was however not independently validated and the methods used not backed by sufficient research. During the Easter weekend of 2012 the System and Network Engineering research group from the University of Amsterdam (UvA) did similar experiments to validate the Brein research[1]. They used a more professional method to gather the data and redid the statistical analysis. They found a few flaws in the methods used by Brein which could be statistically corrected. The measurements done by the UvA researchers were done after TPB blockade by the two ISPs. After comparing the UvA research with the Brein research, no significant shift in peer distributions were found. This validates the hypothesis that blocking torrent sharing sites does not significantly affect the sharing of copyrighted content using the BitTorrent protocol

During this project we developed a method and the necessary tools to perform professional analysis of the BitTorrent protocol. The focus is on the peers participating in the protocol and their origin. With this information we can verify the claims made by Brein and have it gives us insight in the state and trends of the protocol. We emphasize we will not be able to get a complete picture of all the connected peers. The number of swarms to monitor are to great and the number of monitoring probes it would take are not available to us. We strive to get a correct sample of the BitTorrent protocol. We also use the terms swarm and torrents interchangeably. Torrents can however mean a swarm of peers or the actual torrent file with meta-data of the data that is shared in the swarm.

2.1 BitTorrent

The BitTorrent protocol is a file sharing protocol developed by Bram Cohen in April 2001. Unlike most protocols, it does not follow the client-server model but works on a peer-to-peer basis. A swarm is created by peers (nodes) with interest in the same file. The data is broken up in pieces which can separately be downloaded from different peers. A peer that has downloaded a piece makes this available to the swarm. Once a peer has finished downloading a file, it does not have to (and should not) stop making its data available to the swarm, called "seeding". Only with enough seeding peers an efficient "collaboration" of peers is possible.

Sharing big chunks of data in this way will become much more efficient since the download rate is not limited to the maximum upload bandwidth of the server. The first few peers will be limited to the maximum upload bandwidth of the content provider, but once a complete piece has been downloaded a peer will make it immediately available to the swarm. A peer that is downloading from the swarm is called a leecher. Eventually peers can download pieces from different sources in parallel with a much greater aggregated upload bandwidth (although shared across other leechers).

The basic information about the swarm and the file that is shared (meta-data) is stored in a torrent file. This file is generated by the initiator of the swarm and often uploaded to websites dedicated to making the torrents available to the public. A new peer can download this torrent file and use the meta-data to kick-start the joining of the swarm and start sharing the data. Each torrent is assigned a unique 160 bit SHA1 info hash. This hash can be used to request the meta-data from the swarm if the torrent file is not available. In this case the peer needs another way of receiving its initial peers from which it can get additional peers and

possibly the required meta-data.

2.2 Gathering Peers

One of the major challenges in any peer-to-peer protocol is the process of learning the members of the swarm and keeping this information up-to-date. BitTorrent has three ways that are often used in parallel: (1) Trackers, (2) distributed trackers and (3) peer exchange.

2.2.1 Trackers

Trackers are the oldest and most straight-forward method of learning peers. The communication with the tracker does follow the basic client server model and in its basic form is a regular HTTP service. The information (URL) needed to contact the tracker is found in the torrent file. Extensions to the original protocol are proposed (and already in use today) to store more than one tracker in a torrent file[2] or to gossip new or leaving trackers between peers[3]. Since the trackers are the only aspect of the BitTorrent protocol which is not distributed, they are the easiest target for both technical as legal attacks[4][5].

2.2.2 Distributed Hash Table (Distributed Tracker)

The BitTorrent community recognized the problem described in the previous section and developed a distributed way of sharing the data provided by trackers. The Distributed Tracker Protocol[6] is based on the Kademlia[7] distributed hash table (DHT). A DHT can be visualized as a ring built from a key-space of n-bits. Each node and each file has its own key and a corresponding place in the ring. The distance between nodes an files is based on a XOR metric and files close to a node become the responsibility of that node. Each node has a routing table which is built using the initial information of a tracker or a DHT node which can be found using the torrent meta-data, a magnet link (described in section 2.3) or DHT router nodes (static nodes specifically for the initial DHT kick-start).

2.2.3 Peer Exchange

To reduce the load on both trackers as the DHT, an extension to the BitTorrent protocol has been developed; Peer Exchange (PEX). Since different peers in the swarm will receive different lists of peers from the tracker and find different peers in the DHT, exchanging peer-lists between peers adds efficiency to the protocol. A peer will periodically send a list of peers it has seen joining or leaving the swarm to all its currently connected peers. The BitTorrent community agreed that peers that are using PEX will contact the tracker and DHT with a much larger interval[8].

2.3 Magnet Links

The current preferred method for sharing meta-data is by the use of magnet links[9]. Most torrent sharing sites actively promote the use of magnet links because it lowers the load on their servers and provides an easier way of adding torrents for the user. A magnet link is a URI which is automatically picked up by modern BitTorrent clients. In the URI the most vital meta-data is encoded including but not limited to: The info hash, the trackers, initial DHT routers and the torrent name. Once the user clicks on the magnet

link the BitTorrent client is started and uses the meta-data from the URI to get its initial set of peers and queries the peers for additional meta-data, including: The number of pieces, the checksum, the files, etc.

2.4 Previous Research

A good introduction to BitTorrent monitoring can be found in the research by Siganos et. al.[10]. The available research done on monitoring BitTorrent was more limited than expected. There is more information on how to detect monitoring which can be used to create blacklists, etc. A good example is the experiments by Piatek et. al.[11]. They connected to swarms which shared illegal copies of popular movies and collected and analyzed the DMCA complaints. With a (rarely implemented) BitTorrent extension header they could spoof the source IP and received complaints for printers which of course never participated in the swarm. They proved that copyright enforcement groups only check the IP addresses received from trackers and do not check the peer itself by connecting and requesting the data available.

The challenges faced in monitoring BitTorrent are summarized in the work of Hughes et. al.[12]. They propose a system of three classifications: (1) Network-level traces, (2) passive application-level traces and (3) active application-level traces. The research done by the UvA researchers and BREIN falls in the passive application-level category. They joined the swarms and saved the IP addresses the BitTorrent process provided them. Since they used only the messages exchanged between the swarm and their "normal" BitTorrent client they could gather only a small portion of the complete swarm. This is also called the small world problem as discusses by Hughes et. al. We will however take a more active approach. The goal is to gather as much participating IP addresses as we can and therefore we will keep asking the tracker or DHT for more peers. This is however a more risky approach since this will stand out compared to other "normal" peers and could get our probes blacklisted.

3 Method

Our method is built on two parts. First we gather the data with our monitoring solution. This continuous process is designed to gather as much peers as possible and store their information in the database. We also developed an analyzer and visualizer that gives real-time insight in the status of the monitor, the data it has gathered and ultimately the BitTorrent protocol.

The second part of our method is our statistical analysis. The goal is to validate the previous research by the UvA researchers and Brein.

3.1 Monitoring Solution

The monitoring tool has to gather data from a lot of resources and for future applications should be able to participate in all aspects of the BitTorrent infrastructure. We decided to use a very feature-rich library[13], used by a number of popular BitTorrent clients. The number of supported features is the main advantage of using this library. It also makes hiding the monitoring tool in the swarm a lot easier since it will look like and behave like most other clients. A major disadvantage is that the library has been used for a purpose it was not designed for and was not the best tool for some of the jobs it has to do. Given the number of features that had to be implemented and the limited time, it was the most efficient solution however.

3.1.1 Monitor (Probe)

The monitor is designed to participate in a number of swarms like any other client. The major difference that could be noticed is that it queries for new peers as often as is allowed within the BitTorrent standards and it never shares any pieces. The latter difference is possibly the most noticeable and might have an impact on the number of peers we can gather.

The torrents we monitored are ten torrents which are selected to be similar as the torrents monitored in the UvA and Brein research; Popular movies with Dutch subtitles (Appendix A). This is done so our results can be compared with the research from Brein and its research can be validated, as can their claims on the effect of the blocking of torrent sharing sites.

The monitor process is a continuous process that participates in the swarms it is instructed to monitor. It periodically checks a table in the database for new meta-data and starts a new thread if new data is found. The thread will contact a tracker or DHT router and request the initial set of peers. In the case only an info hash is available it will first gather the meta-data from the DHT or one of the trackers from the OpenBitTorrent project[14]. It will then configure the process not to share any of the data but keeps the connections to the trackers and clients open. Periodically it will force a re-announce to DHT and the trackers to force the exchange of a new peer list. The peers it finds are stored in a database.

Since the list of connected peers constantly changes we have to keep track of disconnected peers and take appropriate action. A disconnected host does not necessarily mean it stopped sharing data. It could have dropped the connection to free up resources for other (better) peers or it could have stopped sharing for a particular torrent and continues sharing in different swarms. A peer is therefore never deleted from the database. While a peer is connected to the monitor, the monitor will update a "last seen" field for the peer. Once the monitor notices the peer is not active in any of the swarms it monitors, the peer is marked inactive. These two parameters can be used in the analysis to compensate for the fact that it is very hard to keep connections open to all peers in the swarm.

3.1.2 Database

The database, analyzer and visualizer are closely tied together. In a typical situation they reside on the same host. The database used is the relational database MySQL. Any relational database could have been chosen however.

The torrents and peers are stored in separate tables. The info hash is the unique look-up key for the torrents and the IP address is the unique key for the peers. A third table provides the many-to-many relationships (one torrent has many connected peers and one peer can participate in multiple torrents). Because keeping track of which peers and torrents are active is very difficult (as described in the previous section) we never delete this data. The monitor will update a "last seen" field and a garbage collector will mark them inactive if the number of references in the relation table drops to zero.

The per ISP statistics are stored in a separate table per ISP indicated by their Autonomous System (AS) number. With a regular interval (currently 10 minutes) a new measurement is added to the table with its current number of peers in all the swarms. This information is used by the visualizer to graph trends and the distribution of peers over the ISPs.

3.2 Analyzer and Visualizer

The analysis of the measurements is done in the analyzer. The analyzer is a set of methods that queries the database, performs calculations on the data and transforms it into a format that is understood by the visualizer. The analyzer is closely tied to the visualizer since all analysis methods are called from the visualizer. The analysis that are performed range from simple counting the number of active peers to more complex statistical analysis.

Currently the visualizer shows graphs of the currently active peers by country and by ISP in the Netherlands. We focus on the Netherlands since we will use our measurements to validate the research done by Brein, which also focused only on torrents popular in the Netherlands and peers coming from large Dutch ISPs. The geolocation of the IP addresses was done with the MaxMind Lite GeoIP and ASnum databases[15][16].

The only real-time statistical analysis that is done is the standard deviation. The standard deviation is used to calculate the average of a set of measurements (active peers) and the deviation of measurements from the average. A relatively small deviation indicates that a large portion of the measurements are close to the average. This in turn indicates the correctness of the results. The standard deviation is calculated on a per ISP and per torrent basis.

3.3 Comparing Statistics

With the results gathered from our monitoring solution we should be able to make a good comparison with previous research done by Brein and the UvA. To compare several statistical methods were considered. The field we are focusing on is the analysis of variances, or ANOVA. The F-test has been suggested and we will test its applicability on this research. At the very least we should generate box-plots of our measurement which enables us to make a similar comparison as the UvA researches.

The box-plots visualize the number of peers per ISP compared to the total number of peers from the Netherlands. If the blockades are effective, we should see a drop in the share of peers from the affected ISP.

The UvA research have carried out new measurements days before the forced blockade of TPB by the other major Dutch ISPs (KPN, UPC, Tele2, T-Mobile and Telfort). If the blockade of TPB had a noticeable effect we should see their shares decrease in the box-plots. If the ANOVA fails this should be noticeable enough in the plots.

3.4 Challenges

The focus of the monitoring is the origin of peers in different swarms. To get a good sample of the distribution of peers per country and per ISP, our monitoring solution must gather as much IP addresses from the different sources as possible. None of the sources can be seen as the authoritative source. Trackers are subject to blocking by ISPs and governments[17][18]. The information in DHT can easily be spoofed, is not activated by all BitTorrent clients and could be throttled by ISPs[19]. Peer exchange is not implemented or activated in an even larger number of clients[20]. We decided to monitor as much resources as possible to get the best sample possible.

3.4.1 Monitor Development

Since we chose to monitor all available peer resources, a lot of time was spent on developing the tools needed to monitor these resources. Since we do not have the time and resources to start from scratch, a BitTorrent library[13] was chosen as a starting point. Most of the functionality we need is available through the library. The library is however developed for a BitTorrent client and therefore has to be made to work like a monitor which is not the ideal solution.

We also have to handle large amounts of data, which should be easy to query for analysis. A good solution for this problem exists in the form of a database.

3.4.2 Copyright Infringement

BitTorrent, as most data sharing services on the Internet, is mostly used for sharing copyrighted content. A look on one of the torrent sharing sites shows that the most active torrents are the ones sharing the newest movies, music, etc. Our monitor solution should be able to monitor swarms sharing copyrighted content without actually sharing the content itself. In some countries sharing copyrighted content is only considered a criminal offense if the data is actually uploaded by the user or his computer. In this situation it would suffice to limit the monitor just to downloading.

Since we do not want to limit the monitor to just these countries we will also prevent the monitor from downloading the content. It is also not practical to store the unused data for a large number of swarms.

The effect of not sharing the data has to be looked into since this could be identified by the swarm as suspicious behavior. Another effect could be that connections to our monitor are dropped at a higher rate by the peers because we are not sharing any data. This in turn has a negative impact on the Peer Exchange method.

3.4.3 Detection of the Monitor

Each torrent forces a re-announce to the tracker and DHT for new peers as often as is allowed by the different sources. Since the monitor tries to comply with the specifications we expect its behavior will not be noticed. The BitTorrent community is however increasingly aware of the monitoring done by copyright holders and anti-piracy groups. A number of blacklists are available for clients[21]. A client using the blacklist will refuse connections from hosts on the blacklist making it harder to determine the data available on the peer. A peer that has joined a swarm but refuses to request or provide any data is however already suspicious. This could be noticed by both peers and trackers.

During development the monitor solution has not been detected (as far as known). It is however something that should be monitored. One of the more popular blacklist providers is i-blocklist[21]. They provide a number of lists including the ”level 1” and ”bad peers” list. These lists include IP addresses of known anti-piracy organizations, governments and reported bad peers. Its not clear how these lists are established and the validity can be questioned. These lists are however the primary source for most BitTorrent users and we should avoid getting on these lists. The best way currently known is to comply to the standards and monitor the lists for the presence of our IP addresses.

3.4.4 Analysis

The analysis of the data will be limited to validate the measurement done by Brein and the UvA researchers. With the data that is gathered a lot more can be learned but due to the limited time it is not considered.

4 Results

After developing the BitTorrent monitoring we can collect and analyze the results. Some of the results are automatically analyzed in real-time through the visualizer. The analysis of the peer sources and the validation of the previous research by Brein and the UvA are done by hand.

4.1 Peer Sources

One of the first questions we needed to answer was the effect of not sharing data with the swarm on the number of peers we could gather. Since we do not announce the availability of pieces of the data and do not request any pieces, we are very uninteresting for our peers. We immediately noticed that all BitTorrent connections were very short-lived.

Peer source	Without sharing data	With sharing data
Tracker	2655 (7.6%)	4836 (15.9%)
DHT	11125 (31.9%)	10803 (35.5%)
PEX	5976 (17.2%)	4809 (15.8%)
Incoming	11008 (31.6%)	2694 (8.9%)
Tracker + DHT	745 (2.1%)	3984 (13.1%)
Tracker + PEX	112 (0.3%)	462 (1.5%)
Tracker + Incoming	175 (0.5%)	15 (<0.1%)
DHT + PEX	441 (1,3%)	1902 (6.2%)
DHT + Incoming	911 (2.6%)	108 (0.4%)
PEX + Incoming	672 (1.9%)	21 (0.1%)
Tracker + DHT + Incoming	128 (0.4%)	12 (<0.1%)
Tracker + PEX + Incoming	103 (0.3%)	12 (<0.1%)
DHT + PEX + Incoming	412 (1.2%)	9 (<0.1%)
Tracker + DHT + PEX	233 (0.7%)	779 (2.6%)
Tracker + DHT + PEX + Incoming	127 (0.4%)	3 (<0.1%)

Table 1: Sources of found peers

The results of a comparison test can be found in table 1. It shows the comparison of two measurements. The first measurement was done without sharing data, the second was with sharing copyrighted content. The numbers are the number of peers found through the single or multiple source. The percentage indicates the number of peers it added to the list compared to all found peers.

A noticeable effect can be seen on the number of peers learned through the tracker. Sharing data does seem to have a negative effect on the number of peers learned through incoming connections. It should however be noted that the tests were run on the same day but with several hours in between. Due to time limitations the tests were not redone on a more comparable time of day.

4.2 Real-Time Analysis

Our BitTorrent visualizer gives real-time insight in the status of the swarms it is monitoring through a number of graphs. Most graphs are made on a per torrent, per (Dutch) ISP basis and for all of them combined. To give the user insight in the geographical origin of peers a heat map is generated. For the ten torrents (appendix A) we monitor it can clearly be seen in figure 6 they are the most popular in the Netherlands.

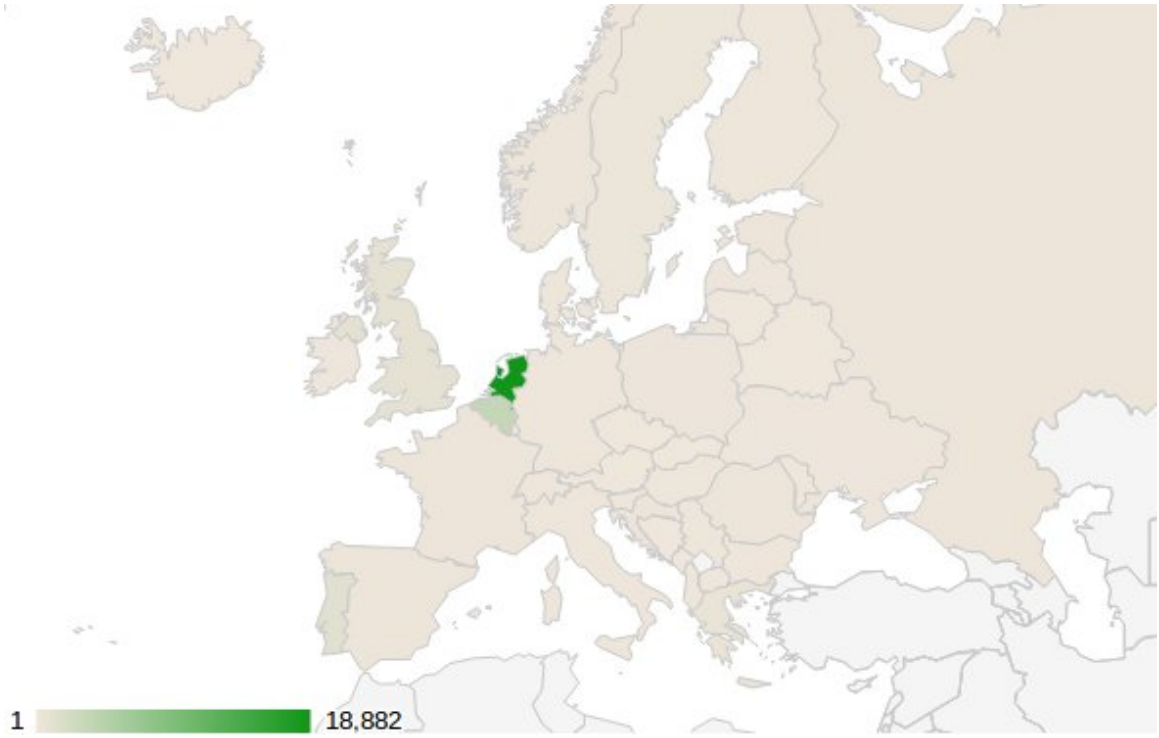


Figure 1: Example of distribution per country map

A large number of standard deviations are calculated since they are a quick and easy way to validate the measurements. For comparison of for example the origin of peers by ISP, the results are graphed in this way next to each-other. It allows the user to visually compare the results and at the same time see the deviation of results which has information on the validity. An example can be seen in figure 2. Due to the limitations of the graphing library a workaround has been made to point out the average in the candlestick graph. The candles show the first standard deviation and the sticks the maximum values measured.

To give insight into the trends of the BitTorrent swarms we also graph the measurements in line graphs. Irregularities in the healthiness of the BitTorrent infrastructure or changes in popularity's of individual torrents are easily seen. An example can be found in figure 3.

4.3 Validation Of Previous Research

The validation of previous research was much harder than anticipated at the beginning of the project. This was mostly due to the very different methods of monitoring. The results seemed very similar but this was hard to prove in a scientific way.

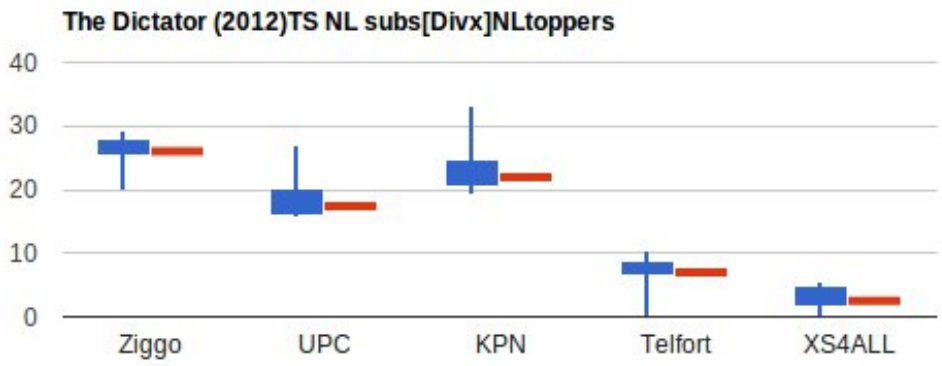


Figure 2: Example of torrent standard deviation graph

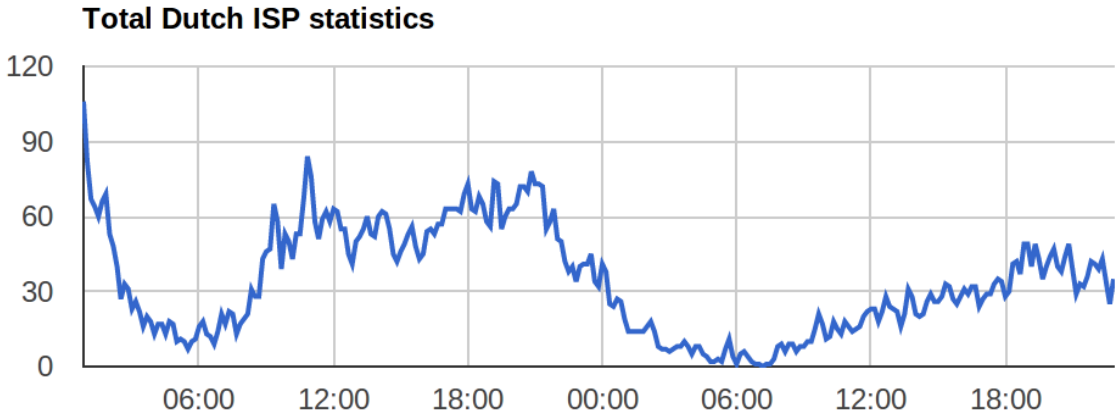


Figure 3: Example of torrent trend graph

Another challenge is the day-night pattern that can be seen in figure 3. This figure shows the analysis of a 24 hour measurement. During the 24 hour period the number of peers per ISP was recorded every 10 minutes. Each measurement point was compared to the total number of peers. The box-plot shows the minimum, maximum and the fist standard deviation of these calculated values per ISP. It clearly shows that during the day the swarm is twice as big as during the night. This also explains the large deviations that can be seen in figure 4. This can be expected since we focus on the Netherlands in our analysis. At night most people shut down their PCs disconnecting them from the swarm. We expect this effect would be less noticeable if we would focus on internationally oriented torrents.

Figure 5 shows the same data but here we devided the day in two 12 hours periods from 10am to 10pm and from 10pm to 10am. This already shows the large difference between day and night and less deviation in the measurements. The fact that we measure a much smaller share of peers from the Netherlands at night does mean that the torrents monitored do attract international attension. Possibly people from the Netherlands living abroad but more likely English speaking people that do not mind the dutch subtitles.

The day-night effect also makes it hard to compare the results with the previous results since they do not state the time the measurements were taken. Also the duration of the test is unknown for the measurements done by BREIN.

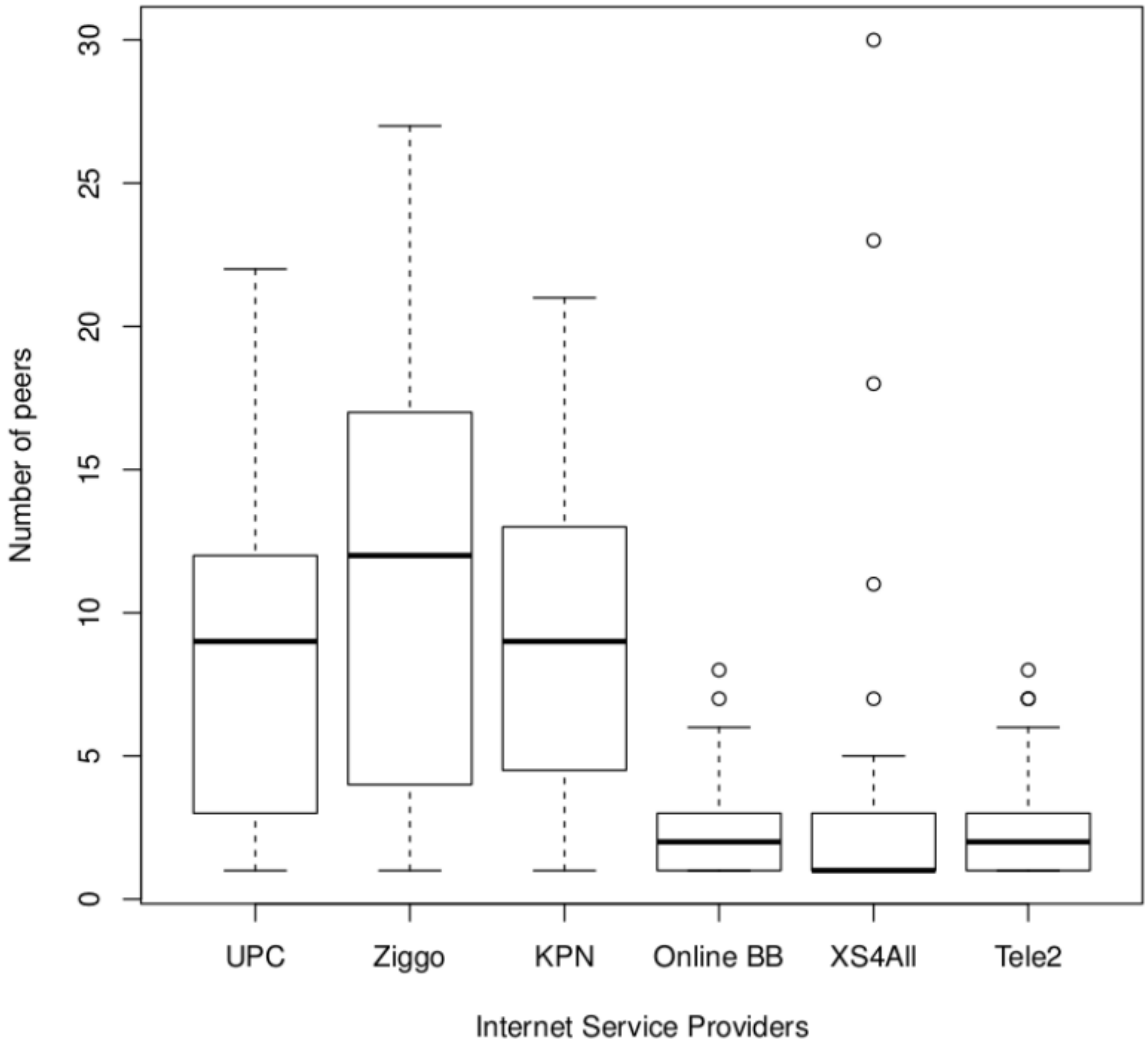


Figure 4: Box-plot of peers per ISP compared to total in percentages

4.3.1 Analysis of Variance

We experimented with several ways to do analysis of variance (ANOVA). We did however not find a reliable way to test our findings with previous research. This is mainly due to the very different methods of measuring. Also some of the context (time of day and duration) is missing from the previous research.

Initially the F-test was proposed to test the null hypothesis that two variances are the same. This works perfectly well on two measurements done by our own monitor but does not give a reliable outcome for comparison of a sample with our monitor and a sample from previous research. We believe this is due to the very different methods of measuring. During our research on the F-test and ANOVA we found that the F-test has been rejected by most of the statistical analysts in favor of others, for example the Bartlett's test.

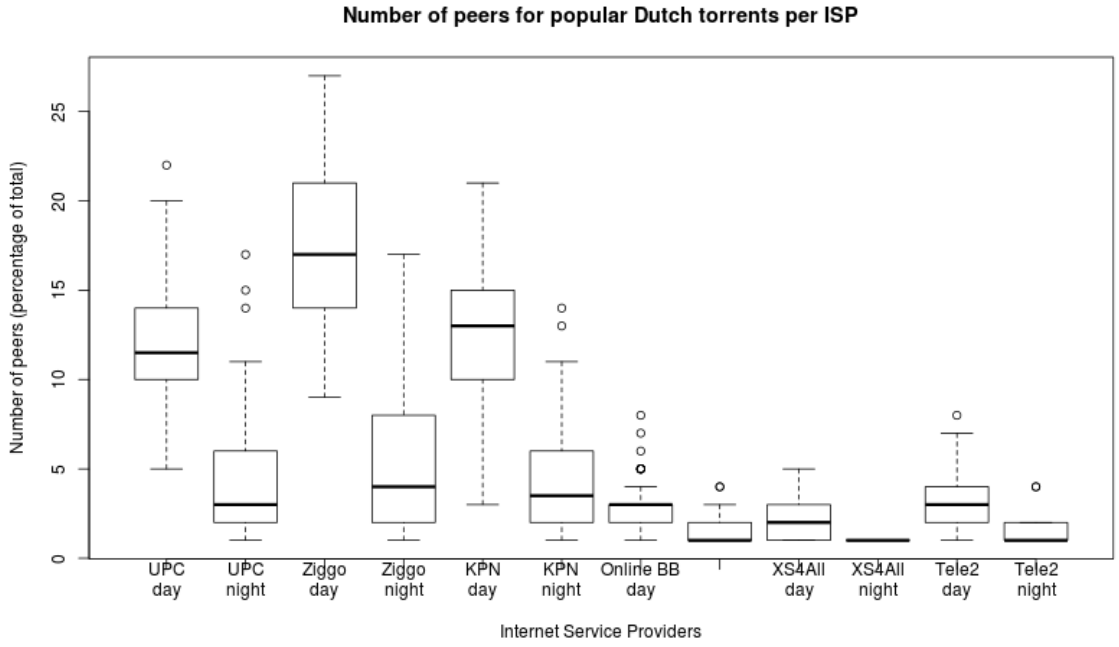


Figure 5: Box-plot of peers per ISP compared to total in percentages, day and night comparison

5 Conclusions

The main conclusion that should be drawn from this research is that the BitTorrent protocol is far to complex to be measured with just a few simple tools. Much more research and a deep understanding of the protocol is required to make any claim on the impact of the proposed measures by Brein and other copyright protectors. We were unable to scientifically prove that our findings are comparable to the previous research and therefore can not make any claims on the effect of the blocking of the pirate bay. The only reliable way to test this effect is with both measurements from our monitoring tool, possibly after being improved on the areas described below.

One of the areas that could use more attentions is the effect of not sharing the data on the number of peers that could be gathered. Due to time limitations we could do only a small number of measurements. The preliminary conclusions we can draw from the results are that the effect is minor. The biggest change we see is the number of peers learned through the tracker. This was expected since it can easily see our lack of participation in the swarm for which we are "punished" with a less complete peer list. Our main source of peers is however the DHT in all tests which is positive. This is easily explained since the DHT is not closely tied to the BitTorrent protocol. It is used to share the meta-data but is in no way dependent on it. Also the number of peers learned through PEX without sharing data is surprising. We are not participating in any exchange of data and the connections to our peers are very short-lived but still we get a good portion of peers through this method. We can conclude that we get a comparable sample of peers without sharing data as we would if we would share data.

We did not look into how the number of peers we gathered compared to the total size of the swarm as reported by the tracker or estimated from DHT with DHT scrape[22] or Bloom Filters. Based on this could be estimated or tested how much probes would be needed to gather close to all peers.

When we join a new swarm we measure a relatively large number of peers compared with measurement done a few hours after we joint the swarm. The algorithm running on most BitTorrent trackers distributes a larger list of peers to hosts joining a swarm to speed up the search for initial peers suitable for sharing data. This tells us that swarms are larger than we can currently measure. More research is needed on if it is possible to trick the tracker in sending the larger list for a longer period or if we can make this effect less relevant by improving the other peer exchange methods.

More research can be done in the possible analysis of the data collected by the monitor. The gathered data can easily be extended to make other analysis of BitTorrent possible. This could help researches measure the effect of new techniques or the effect of problems in the BitTorrent infrastructure. A good example is the analysis we did on the source of learned peers. This shows that we received the most information, by far, from the DHT. Clients should be save to rely heavily on DHT for their meta-data, making trackers obsolete. We would like to make this information publicly available, possibly through an API, giving everyone the chance to do their own measurements without heaving to worry about the measurement infrastructure. This does require a lot of work still.

Other problems that need more work are the distribution of measurements over a (possibly large) number of probes and the storage of the large amounts of gathered data. Getting a reasonable query performance after gathering close to 40.000 peers proved to be a challenge. When monitoring more than the 10 torrents we focused on, this will be a challenge.

The monitoring solution we developed is a good step on the way of a scientifically proven solution validated by the community. We described the ideas and design decisions in this paper and analyzed some aspects of their effectiveness. Hopefully this will inspire the community to validate our claims and extend our research.

Since we have only taken the first steps we feel we cannot claim our results are scientifically valid until more research is done. We find it therefore remarkable that these claims are made by Brein and to a lesser extend by the UvA researchers. Even more remarkable is the fact that the results from the Brein research was used in court and formed a key part in the decision to block the Pirate Bay.

Hopefully this research can be used to validate the effect of a new blockade in the future. Although we do not want to take a stance against Brein we do notice the dangers of these forced blockades on the free Internet. We feel that these measures are not the solution to the problem and that the problem is not well understood by the people that have the authority to act on it.

6 References

[1] Jeroen van der Ham, Hendrik Rook, Cosmin Dumitru, Ralph Koning, Niels Sijm, and Cees de Laat. Review en herhaling brein steekproeven 7-9 april 2012 (dutch). 2012.

[2] John Hoffman. Multitracker metadata extension. `http://bittorrent.org/beps/bep_0012.html`, 2008.

[3] Arvid Norberg. Tracker exchange extension. `http://bittorrent.org/beps/bep_0028.html`, 2009.

[4] `http://torrentfreak.com/russias-largest-tracker-under-huge-ddos-attack-120208/`.

[5] `http://torrentfreak.com/bittorrent-tracker-loses-lengthy-legal-battle-100211/`.

[6] Andrew Loewenstern. Dht protocol. `http://bittorrent.org/beps/bep_0005.html`, 2008.

[7] Petar Maymounkov and David Mazires. Kademlia: A peer-to-peer information system based on the xor metric. In Peter Druschel, Frans Kaashoek, and Antony Rowstron, editors, *Peer-to-Peer Systems*, volume 2429, pages 53–65. Springer Berlin / Heidelberg, 2002.

[8] `http://wiki.vuze.com/w/Peer_Exchange`.

[9] Greg Hazel and Arvid Norberg. Dht protocol. `http://bittorrent.org/beps/bep_0005.html`, 2008.

[10] Georgos Siganos, Josep Pujol, and Pablo Rodriguez. Monitoring the bittorrent monitors: A birds eye view. In Sue Moon, Renata Teixeira, and Steve Uhlig, editors, *Passive and Active Network Measurement*, volume 5448, pages 175–184. Springer Berlin / Heidelberg, 2009.

[11] Michael Piatek, Tadayoshi Kohno, and Arvind Krishnamurthy. Challenges and directions for monitoring p2p file sharing networks-or: why my printer received a dmca takedown notice. In *Proceedings of the 3rd conference on Hot topics in security*, pages 12:1–12:7. USENIX Association, 2008.

[12] Kevin Lee Danny Hughes, James Walkerdine. Monitoring challenges and approaches for p2p file-sharing systems. In *Internet Surveillance and Protection, 2006. ICISP '06. International Conference*, page 18, 2006.

[13] Arvid Norberg. Libtorrent c++ bittorrent implementation. `http://www.rasterbar.com/products/libtorrent/index.html`.

[14] `http://www.openbittorrent.com`.

[15] `http://www.maxmind.com/app/geolite`.

[16] `http://www.maxmind.com/app/asnum`.

[17] `http://torrentfreak.com/demonoid-bittorrent-tracker-blocked-for-russians-090818/`.

[18] `http://wiki.vuze.com/w/Bad_ISPs`.

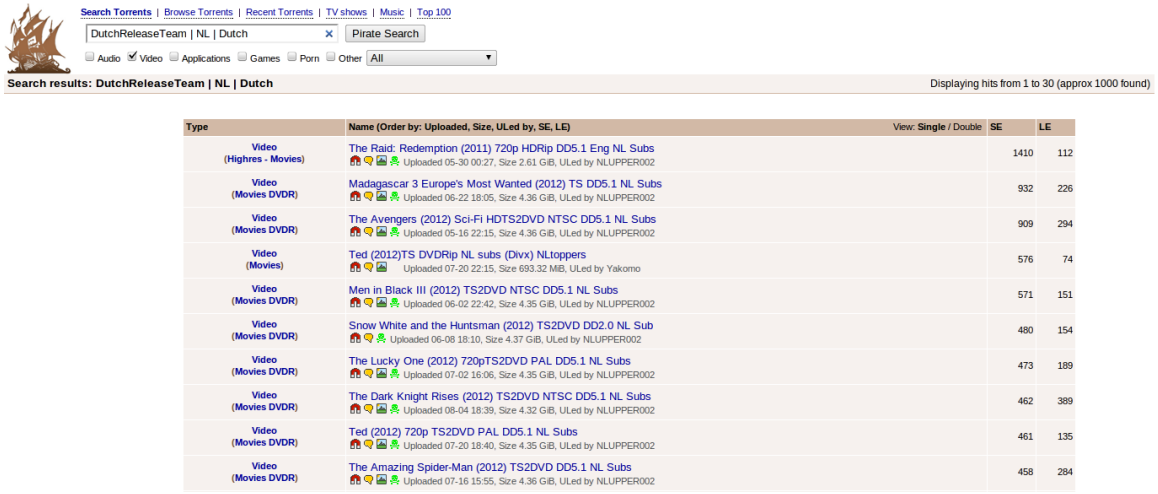
[19] Jop van der Lelie Tarik El Yassem. Assessing the authenticity of distributed hash table entries in peer-to-peer networks. 2012.

[20] `http://en.wikipedia.org/wiki/Peer_exchange`.

[21] `http://www.iblocklist.com`.

[22] The 8472. Dht scrapes. `http://bittorrent.org/beps/bep_0033.html`, 2010.

A Appendix A - Monitored Torrents



Search Torrents | Browse Torrents | Recent Torrents | TV shows | Music | Top 100

DutchReleaseTeam | NL | Dutch — Pirate Search

Audio Video Applications Games Porn Other All

Search results: DutchReleaseTeam | NL | Dutch — Displaying hits from 1 to 30 (approx 1000 found)

Type	Name (Order by: Uploaded, Size, ULed by, SE, LE) — View: Single / Double	SE	LE
Video (Highres - Movies)	The Raid: Redemption (2011) 720p HDRip DD5.1 Eng NL Subs Uploaded 05-30 00:27, Size 2.61 GiB, ULed by NLUPPER002	1410	112
Video (Movies DVDR)	Madagascar 3 Europe's Most Wanted (2012) TS DD5.1 NL Subs Uploaded 06-22 18:05, Size 4.36 GiB, ULed by NLUPPER002	932	226
Video (Movies DVDR)	The Avengers (2012) Sci-Fi HDTS2DVD NTSC DD5.1 NL Subs Uploaded 05-16 22:15, Size 4.36 GiB, ULed by NLUPPER002	909	294
Video (Movies)	Ted (2012)TS DVDRip NL subs (Divx) NLtoppers Uploaded 07-20 22:15, Size 693.32 MiB, ULed by Yakomo	576	74
Video (Movies DVDR)	Men in Black III (2012) TS2DVD NTSC DD5.1 NL Subs Uploaded 06-02 22:42, Size 4.35 GiB, ULed by NLUPPER002	571	151
Video (Movies DVDR)	Snow White and the Huntsman (2012) TS2DVD DD2.0 NL Sub Uploaded 06-08 18:10, Size 4.37 GiB, ULed by NLUPPER002	480	154
Video (Movies DVDR)	The Lucky One (2012) 720pTS2DVD PAL DD5.1 NL Subs Uploaded 07-02 16:06, Size 4.35 GiB, ULed by NLUPPER002	473	189
Video (Movies DVDR)	The Dark Knight Rises (2012) TS2DVD NTSC DD5.1 NL Subs Uploaded 08-04 18:39, Size 4.32 GiB, ULed by NLUPPER002	462	389
Video (Movies DVDR)	Ted (2012) 720p TS2DVD PAL DD5.1 NL Subs Uploaded 07-20 18:40, Size 4.35 GiB, ULed by NLUPPER002	461	135
Video (Movies DVDR)	The Amazing Spider-Man (2012) TS2DVD DD5.1 NL Subs Uploaded 07-16 15:55, Size 4.36 GiB, ULed by NLUPPER002	458	284

Figure 6: Example of distribution per country map