

Monitoring the Bittorrent Monitors: A bird's eye view

Georgos Siganos, Josep M. Pujol, and Pablo Rodriguez

Telefonica Research
{georgos,jmps,pablorr}@tid.es

Abstract. Detecting clients with deviant behavior in the Bittorrent network is a challenging task that has not received the deserved attention. Typically, this question is seen as not 'politically' correct, since it is associated with the controversial issue of detecting copyright protection agencies. However, deviant behavior detection and associated blacklists might prove crucial for the well being of Bittorrent. We find that there are other deviant entities in Bittorrent besides monitors. Our goal is to provide some initial heuristics that can be used to automatically detect deviant clients. We analyze for 45 days the top 600 torrents of Pirate Bay. We show that the empirical observation of Bittorrent clients can be used to detect deviant behavior, and consequently, it is possible to automatically build dynamic blacklists.

1 Introduction

P2P file-sharing networks have brought a revolution to the way we communicate and exchange files on the Internet. The ease with which new applications and technologies can be deployed have captured the interest of the Internet community. P2P systems like Bittorrent are widely used for file transfers for they are more efficient than classical client/server architectures. The subject of distribution, however, is sometimes copyright protected material. This fact has outraged copyright holders, who try to stop or hinder the distribution of their material over P2P networks.

The tension between users and copyright protection companies has triggered an arms race, in which both sides play a hide and seek game. This arms race has lead to the current situation in which Bittorrent network is populated by different kinds of clients. Standard clients are those whose intension is to find and share content, whereas other clients are focused in monitoring and/or preventing the activity of the standard clients. We aim to develop a methodology to automatically detect and classify the different clients that populate Bittorrent networks.

We should stress here that wide coverage accurate real-time blacklist creation is not exclusively intended to prevent detection from legit copyright protection agencies. We are not in any way condoning unlawful sharing of copyrighted material, but rather showing that Bittorrent ecosystems have more entities besides

seeders, leechers and monitoring clients. We show that Bittorrent swarms contain a high density of clients belonging to Botnets, which seem to indicate that a potential escalation of the arms race cannot be completely overwritten.

The research community has already start discussing ways by which Botnets can be used to attack Bittorrent [4]. Additionally, large scale Internet attacks have been reported against P2P systems, and in some cases, allegedly by established anti P2P companies [5][6]. Accurate and up-to-date methods to discriminate between *standard* and *deviant* clients will play a key role on sustaining the success of Bittorrent networks.

Paper contributions: In this paper, we provide some initial heuristics that can be used to automatically detect deviant clients.

- We analyze for 45 days the top 600 torrents of Pirate Bay. The dataset is composed of hourly snapshots that amount for over 1.8 Terabytes of data, which contain 37 million unique IPs.
- This data captures the actual empirical behavior of Bittorrent clients, both in one hour snapshots and across time.
- Access to the aggregated behavior data of Bittorrent allow us to develop heuristics that discriminate clients by their behavior.
- We show that is possible to automatically build dynamic blacklists that capture those clients that deviate from the typical behavior. This is achieved by using a fraction of the IP space when compared to the state-of-the-art blacklists.
- Additionally, we show that a significant number of deviant clients are Botnet controlled.

2 Related Work

The presence of copyright enforcement agencies monitoring peer-to-peer users is neither new nor exclusive to Bittorrent. Understanding how copyright protection agencies monitor Bittorrent has been studied by Piatek et al [7]. Their focus, however, is to characterize the methods used by enforcement agencies to detect copyright infringement. Their study provides empirical evidence that enforcement agencies rely mostly on indirect detection methods. They received more than three hundred DMCA takedown notices without uploading or downloading any file whatsoever. These DMCA takedown notices are issued by agencies that veil for the interest of copyright holders. However, in their case the *cease or face the consequence* legal threads are unfounded since they do not take part in the exchange of the copyrighted content. The authors also show that besides false positives, the indirect detection is also weak against malicious users who can easily frame virtually any network endpoint, even their own network printer.

Banerjee et al [2] observe that between 12% and 17% of the peers in Gnutella are blacklisted. Consequently, the likelihood of interacting with peers that are susceptible to monitor is 100% unless countermeasures are in place; if peers actively avoid interaction with the top-5 most prevalent IP ranges of the blacklist, the risk of detection is reduced to 1%. They also report that most of the

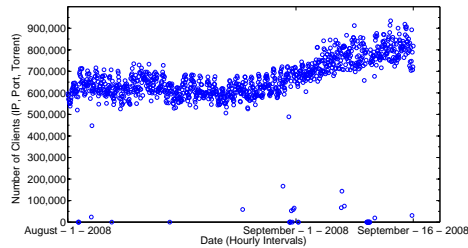


Fig. 1. Temporal evolution of the Number of clients we monitor per hour. Each dot corresponds to number of clients captured in one hour snapshot

encounters with blacklisted peers correspond to those peers run by copyright enforcement agencies. However, the risk reduction provided by blacklist is only effective when coverage is perfect, i.e no monitoring client goes unnoticed.

3 Measurement Details

Our analysis is based on the data collected by the Apollo project [8]. In that project, we develop an instrumented bittorrent client that can connect and exchange control messages with hundreds of thousands of other bittorrent clients within a small time frame (half to one hour).

Data collected: We monitor the 600 most popular torrents of Pirate Bay [3]. We monitor these 600 top torrents once per hour for a duration of 45 days, from August 1st, 2008 to September 16th 2008. During these 45 days period, we collected over 37 million unique IPs and 3/4 of a billion hourly instances of clients. Figure 3 depicts the number of clients analyzed on an hourly base. The number of clients per hour ranges between 500 and 900 thousand. The figure also shows the existence of seasonal effects in the Bittorrent network; the transition from August to September provides approximately 200K additional clients.

Blacklists: As a guideline to detect suspicious clients we use public available lists (blacklists) that are used to prevent establishing connections to monitoring clients. These lists are typically used in conjunction with programs like the Peer-Guardian. A repository of available blacklists can be found at [1]. We use the list provided by the Bluetack company called Level-1 which provides IP ranges for companies that either are involved in trying to stop file-sharing (e.g. MediaDefender), or produce copyrighted material (e.g. Time Warner), or are government related (e.g DoD). The list has 222,639 ranges entries and in total it covers 796,128,149 IPs, that is more than 47 /8.

Botnets: A plague of the current Internet are the millions of computers that are exploited and compromised. These computers (often called Zombies) are used to run applications that are controlled by the owner of the Botnet. We want to analyze if deviant bittorrent behavior can be partly attributed to clients that are part of a Botnet. To this end, we query the Spamhaus database and check whether an IP address is part of a botnet or not. We use two lists from

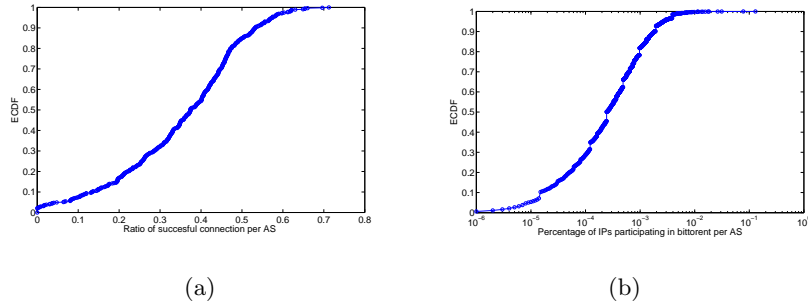


Fig. 2. (a) ECDF of the Ratio of successful connections per AS. Zero means we could not connect to any client within that AS, while one means we connected to all (b) ECDF of the Percentage of IPs of an AS that participate in bittorrent.

Spamhaus. The SBL consists of IP addresses that have been verified as spam sources. On the other hand, the XBL consists of IP addresses that are known to be exploited. If an IP appears in either list, we classify it as being part of a Botnet.

4 Deviant Client Discovery

This section deals with the detection of deviant clients from the Bittorrent data previously described. We analyze the data at the level of Autonomous Systems (ASes) and at the level of prefixes and we reveal a wide variety of abnormal behavior. Further exploration of these anomalies allows us to propose a simple set of heuristics to sort out those clients that do not behave as the *standard* Bittorrent clients. These heuristics, combined with a data collection system such as Apollo [8], can lead to the automatic creation of dynamic blacklists for Bittorrent networks.

4.1 AS-level Deviant Behavior

The first analysis is performed at the Autonomous System (AS) level in order to look for agencies big enough to have their own AS number. We focus our analysis in three different aspects and propose a heuristic to identify those ASes that are unlikely composed of *standard* Bittorrent clients. Note that the explanation is based on the data snapshot collected between 20h-21h CEST on August 1st. The observed anomalies, however, are consistent across time and the proposed heuristics do not change for different snapshots.

1. **Blocking connection:** Figure 2(a) shows the empirical cumulative distribution of the Ratio of successful connections per AS. We see that there are ASes which do not accept any connection, which is odd to say the least. The

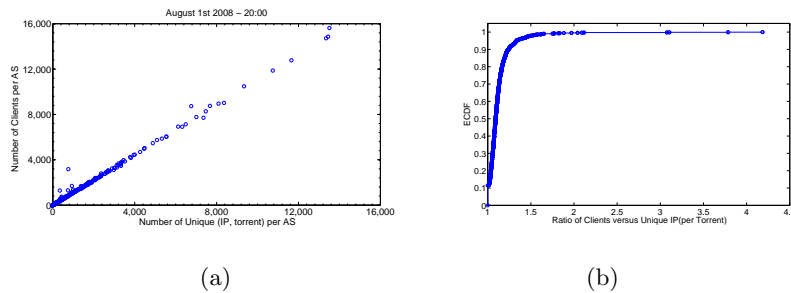


Fig. 3. (a) Number of clients versus number of IPs per AS. Ratio of over 1 means either the presence of DHCP or multiple clients per machine.(b) ECDF of the Ratio of clients versus IPs for ASes with over 100 clients.

first heuristic rule is that ASes that do not accept incoming connections are deviant provided they have at least 100 clients. The exception is set in order to avoid the statistical effect due to small number of clients. This rule is typically met by 10 to 15 ASes. Using the Routing Registries the ASes can be classified into four categories: wireless, universities, ISPs with network wide firewalls, and the rest. The latter being ASes that belong to companies suspicious of carrying out monitoring activities. For instance, the last category has two ASes: AS11662 and AS33970. The first corresponds to MediaDefender, a well known company that does P2P monitoring. The second AS number corresponds to a hosting provider used by an unidentified monitoring P2P company.

2. **Percentage of active bittorrent users:** We expect that an AS used for monitoring will have a large fraction of its advertised IPs devoted to run P2P clients. Figure 2(b) shows the ECDF of the ratio between the AS's IPs that participate in the Bittorrent network and the number of advertised IPs. We find that AS11662 has 13% of its IPs in the Bittorrent network. The next AS has less than 1% of their IPs active. Note that using this rule we do not find AS33970, since a hosting provider has many customers that do not take part in Bittorrent.
3. **Multiple clients per IP:** Figure 3(a) depicts the number of clients in an AS versus the number of distinct (IP,Torrent) tuples. The plot clearly shows that some ASes deviate from the typical behavior. Especially, those ASes that have a small number of distinct IPs, typically less than 1,000. This can mean that either the AS is using heavily DHCP and has a small number of IPs, or that we have multiple clients running in every machine. To further explore this relation we plot, in figure 3(b), the ECDF of the ratio between the number of distinct tuples and clients for those ASes over 100 clients. As expected this plot illustrates that the most common ratio is the expected one, one client per (IP,Torrent) tuple. There exist cases in which the same IP is running a surprisingly amount of clients. We set the rule by which any

AS with a ratio higher than 2 to be suspicious. For this case we identify two ASes: the already discussed AS33970, and AS29131 that like AS33970 is a provider suspicious of hosting a monitoring P2P agency.

The heuristics can be tested against the data from a single hour snapshot. Any AS that meets at least one of these heuristic rules will have all its clients classified as deviant. In the case of the examined snapshot, we identify three ASes that meet at least one of the rules: AS11662, AS33970, AS29131.

By looking at the combination of the resources that these ASes use and the torrents they participate in, it is obvious that they are doing some form of monitoring. For example, for the snapshot we examine, they use 257, 87 and 214 unique IPs respectively to monitor 2-3 movies each¹. Regarding the accuracy of the Level-1 blacklist, for AS11662 and AS29131 the blacklist covers all their IPs. On the other hand, for AS33970, it only covers 55 IPs out of the 87 used. Finally, we should note two interesting observations. First, surprisingly AS33970 and AS29131 were basically monitoring the same Torrents, with the addition of two Torrents for AS29131. Second, we could not detect any monitoring activity for AS11662 and AS29131 in September. In the case of AS11662 (MediaDefender), one possible reason might be the discovery that the company had financial problems, and that the revenue from anti-piracy services had declined considerably. It is not clear, however, whether they actually stopped monitoring activities or they just found a better way to hide.

4.2 Network-level Analysis (IP Prefixes)

After analyzing deviant behavior at the AS level, we turn into the prefix level. Suspicious clients are expected to appear more regularly than clients of standard Bittorrent users. The typical DSL customer should not be visible for more than a few hours per day. Additionally, we also expect colluded deviant clients to have a similar IP range as IP are allocated sequentially. Note that in the case of ISPs the same IP within a week can be assigned to different users.

Unlike the analysis on the AS level which requires only an hour snapshot, we need more than one snapshot for the prefix analysis as we check for temporal presence of clients. We chose a period of one week. Analogously to the AS level we choose the first available data range for the discussion, which in this case is the first week of August. Nonetheless, the findings and the proposed heuristics rules do not depend on this particular weekly data and are consistent across time. For the first week of August we collected 7,775,444 unique IPs.

Figure 4(a) displays the empirical cumulative distribution of the number of times (in hours) that an IP appears over the week interval. The majority of IPs are visible for less than 24 hour per week. We label all IPs whose presence in the network last longer than 50 hours as possible candidates. We then proceed to analyze the distance between the IP's of the candidates. Since we expect deviant clients to be grouped together by IP, either sequentially or within some

¹ Some movies have multiple torrents. For example for the movie "Wanted", they monitor 4 different Torrents

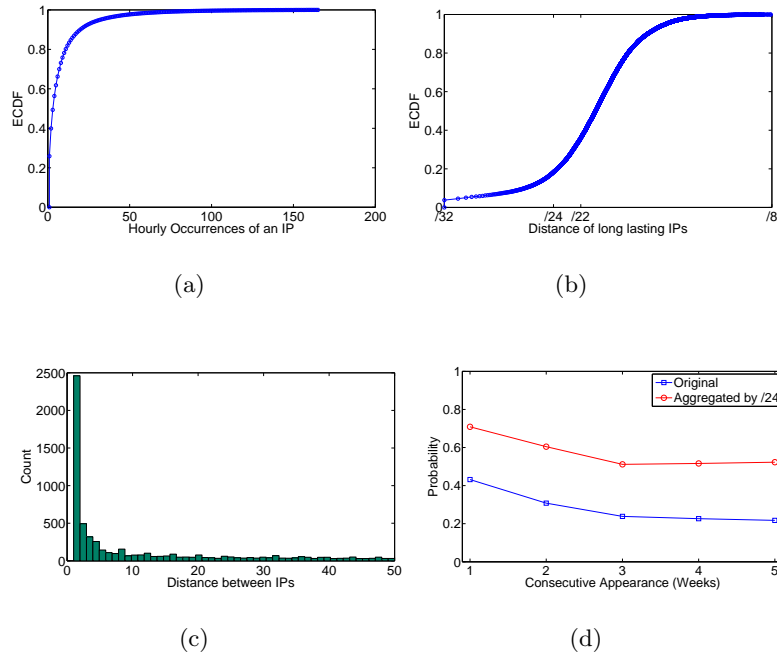


Fig. 4. ECDF of number of unique occurrences of IPs per hour. Interval August 1 - August 7 (Max 168) (b) ECDF of distance between IPs that appear at least in 50 of the hour instances. The distance is computed by sorting the IPs and find the distance between adjacent IPs. (c) Histogram of the distance between IPs (d) Probability of continuous appearance of an IP range in weeks

small distance, we calculate the distance between candidates. Figure 4(b) shows that majority of the candidates' IPs are not correlated by distance, yet some correlation exists. We group those IPs that have a distance of 6 or less to form an IP-range. IP-ranges are further grouped if they have a distance less to 100. Applying these heuristics rule to the first week of August yields 575 IP ranges and 6,317 IPs that classified as deviant.

Stability of ranges: Since we are using weekly data to find the candidates depending on their presence in the Bittorrent network we need to check for the stability of the IP ranges over the one week period. Stable IP ranges are necessary to for the aggregation process, if IP ranges are volatile it implies that the heuristic is aggregating IP by pure chance. To evaluate the stability we compute the probability of continuous appearance of an IP range throughout the size weeks of the experiment. Given an IP range that appears in week i , we calculate what is the probability that it also appears in week $i + 1$. Provided that the IP range appeared in week $i + 1$, we compute the probability that it appears again in week $i + 2$ and so on. Figure 4(d) shows the probability on the

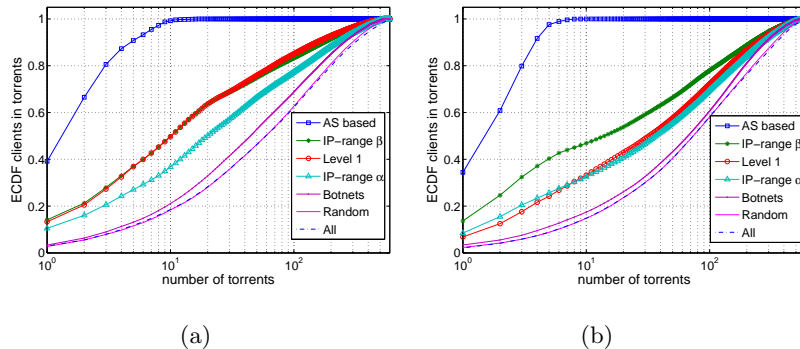


Fig. 5. (a) August 7th behavior of different blacklists (b) September 7th behavior of different blacklists

original ranges in addition to the original ranges aggregated in /24s. We observe that over 20% of the original ranges are long-lasting and active during the full length of our experiment. 43% of the original ranges last for at least two weeks. When aggregation takes place at the /24 level, we observe that 50% of the /24 IP ranges are active throughout our experiment and that 70% of them appear at least for 2 weeks.

5 Evaluation of blacklist IPs

The absence of ground truth makes evaluation of blacklist a difficult problem to tackle. Accuracy of blacklists could be assessed by the coverage that different blacklists offer against receiving copyright infringement reports. However, as Piatek shows [7], DMCA takedown notices can be issued by indirect detection alone. In this case, blacklist are ineffective as they only prevent direct interaction².

We propose an alternative way to evaluate blacklist by observing the behavior of suspicious clients against the whole population³. Since most clients participating in swarms are people, we do expect that the torrents downloaded are chosen

² Note, that DMCA takedown notices are only legal threats. In order to provide proof of copyright infringement, exchange of data by direct contact is necessary, which can be avoided by blacklists.

³ We must warn the reader about a potential pitfall of this method when the criteria to partition the space of clients is a function of popularity itself. In the degenerate case, one must choose all those legitimate clients who are downloading a given movie and nothing else. This would give a popularity distribution far off the predicted by the aggregated behavior of all clients. The criteria that we use to build the partitions do not fall into this pitfall as they do not depend of the distribution itself and consequently the aggregate clients' behavior is not an artifact.

according to people’s interests, and therefore, follow a pareto-like distribution. Figure 5, shows the cumulative distribution of clients by torrent for a sample of one hour for two different days: August 7th in Fig. 5(a) and September 7th in Fig 5(b). Each line corresponds to different partitions of the 667K clients in August and 887K clients in September.

Let us focus on the September data-set, since it shows more BitTorrent activity. Note that different hourly instances show quantitative variations. The qualitative results, however, are consistent across all the samples analyzed. Taking all the 887K clients into consideration, *all* partition, yields the expected pareto distribution. The top-10 torrents account for 15% of the clients whereas the top-100 torrents account for 57% of clients. The remaining 500 torrents are downloaded by only 43% of clients in the BitTorrent network. The long-tail phenomenon is quite strong and shows that despite most clients are focused on *block-busters* torrents, the interest of the whole population of clients spawns over a wide range of torrents.

In order to show that the skewed distribution of clients per torrent is consistent, we pick at random sets of 10K out of the 887K clients. The figure shows that the behavior of about 1% of the clients, *Random* partition, is undistinguishable from the whole population behavior depicted in the *All* partition.

The next partition to consider is based on our analysis on the the AS level (Section 4.1). This method detects 4,339 suspicious clients, 80% of which are focused only in the top-3 torrents. Devoting so much clients in so few torrents completely ignoring the rest denotes that the clients do not follow the general trend of interest in torrents, and therefore, they are doing something else other than downloading content. We show that this behavior, yet in a smaller scale, is also observed in the other sets. Both the state-of-the-art Level-1 blacklist and our blacklist build at the level of Prefix (Section 4.2) display the same *unnatural* concentration of clients in the most popular torrents at the expenses of the long-tail. Level-1 blacklist detects 7,037 suspicious clients whereas our IP-range- α black list detects 18,073. The intersection of both blacklists gives a shared 3,394 clients. The small overlap between blacklists cannot be attributed exclusively to false positives; both blacklist have a similar distribution of clients per torrent. This finding shows that both blacklists explore different areas of the *hidden* space of deviant peers in the BitTorrent network. Different methods finding deviant clients with small overlap might indicate that the space is much larger than initially expected.

Exploring in detail individual deviant clients, we see that many are monitoring agents acting in behalf on copyright enforcement agencies. However, there are many other that are do not seem to take part in monitoring but they are detected because they are parts of a Botnet. Approximately 39% of the deviant peers are reported as botnets by Spamhaus. We also test 10K IPs from the *all* set chosen at random to see if the ratio is maintained outside the suspicious client: the test yield that 2.5% of all IP’s taking part in BitTorrent were botnets according to Spamhaus. Because of the high density of botnets detected by our prefix-based blacklist, we partition the *IP-range α* depending on whether the IP is found in Spamhaus or not. This yields the Botnet-free partition *IP-range*

β and the only Botnet partition *botnet*, which are the two remaining partitions depicted in Figure 5.

Interestingly, the distribution of torrents by clients of *botnet* does behave very similarly to the *standard* BitTorrent clients. There is no bias in the way Botnet clients select their torrents, and consequently, monitoring is unlikely. Eliminating the botnets has the benefit of clearing out the unbiased clients leaving a more accurate blacklist of monitoring clients. The *IP-range* β partition gives 11,036 suspicious clients, 47% of which target the top-10 movies. This is 3 times more than the expected distribution.

6 Conclusions

We evaluate the potential of using the global view of the Bittorrent network to detect deviant behavior in clients. We show that this is feasible by detecting well known monitoring clients and by discovering a large number of known Botnets, using a fraction of the IP space when compared to the state-of-the-art blacklists. We believe that the need to differentiate between a real client and a deviant client is only going to become larger in the future. Our methodology just scratches the surface of the problem. We provide some initial heuristics that can be used to automatically detect deviant clients. The goal of our work is to be able to automatically generate on the fly dynamic blacklists.

References

1. iblocklists. <http://iblocklist.com/lists.php>.
2. Anirban Banerjee, Michalis Faloutsos, and Laxmi Bhuyan. The p2p war: Someone is monitoring your activities. *Comput. Netw.*, 52(6):1272–1280, 2008.
3. The Pirate Bay. Top 100 torrents per category. <http://thepiratebay.org/top>.
4. Karim El Defrawy, Minas Gjoka, and Athina Markopoulou. Bittorrent: misusing bittorrent to launch ddos attacks. In *SRUTI'07: Proceedings of the 3rd USENIX workshop on Steps to reducing unwanted traffic on the internet*, pages 1–6, Berkeley, CA, USA, 2007. USENIX Association.
5. P Dhungel, BS Di Wu, and KW Ross. A measurement study of attacks on bittorrent leechers. 2008.
6. J. Liang, N. Naoumov, and K.W. Ross. Efficient blacklisting and pollution-level estimation in p2p file-sharing systems. 2005.
7. 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 *Proc. of 3rd USENIX Workshop on Hot Topics in Security (HotSec 08)*, 2008.
8. Georgos Siganos and Pablo Rodriguez. Network transparency from the edge: what can p2p users reveal? *under preparation*. <http://www.cs.ucr.edu/~siganos/papers/apollo.pdf>.