The Arms Race in P2P

KEVIN BAUER,
DIRK GRUNWALD,
and
DOUGLAS SICKER
University of Colorado

Peer-to-peer (P2P) networks have recently grown in popularity for a variety of applications such as content distribution, streaming multimedia and voice-over IP. P2P networks are built around a decentralized architecture to distribute data in a manner that offers high availability of content, inherent fault-tolerance and efficiency. While P2P networks offer several important advantages over traditional client/server architectures, experience has shown that these networks are file sharing copyright protected content, which presents significant problems for network management and copyright enforcement. P2P networks utilize a large amount of bandwidth, complicating network management for broadband Internet service providers (ISPs), particularly during times of peak network utilization. In addition, the illegal dissemination of copyright protected media is an obvious problem for the respective copyright holders that may result in a loss of revenue. As a consequence, there is ample incentive for both broadband ISPs and copyright holders to work to stop the proliferation of file sharing within P2P networks.

Our primary goal in this paper is to understand the current techniques for distributing and hiding copyright protected content within P2P networks. We focus our discussion primarily on BitTorrent, since it is currently the most popular P2P protocol for file sharing. We observe that an arms race has already begun between file traders and copyright holders in which the file traders have started to develop techniques for hiding their involvement in the transfer of a copyright protected media file. In response, the investigative tactics used by copyright holders are evolving to match these changing strategies. We provide a survey of the current tactics used by file traders to hide their involvement in illegal file transfers and speculate about future strategies that may emerge on both sides of the arms race.

In this paper we provide an introduction to BitTorrent, the most common P2P protocol for file sharing in use today. Next, we describe the most common techniques that copyright holders have employed to track the distribution of their copyright protected content. These strategies often include investigative tactics, locating individual users and issuing DMCA take-down letters, or even pursuing more serious legal actions against suspected file sharers. We then discuss the past tactics used by broadband ISPs to throttle BitTorrent traffic. In response to the copyright holder's desire to protect their content, there is now significant incentive for P2P users to shed their network identities and seek a certain degree of anonymity. In addition, to avoid traffic shaping, P2P users have incentive to try to hide the nature of their traffic by using encryption. We consider the current tactics used to evade ISP traffic shaping practices and copyright enforcement, describe the most common methods for achieving anonymity online, and present evidence from a prior study by the authors that P2P users are using BitTorrent anonymously. We next briefly outline a variety of proposals to incorporate anonymity mechanisms into P2P networks and speculate about the future tactics that may be employed to distribute copyright protected content. To conclude, we examine some of the policy implications of this arms race and consider what impact they may have on copyright and broadband policy.

Authors' Address: University of Colorado, 430 UCB, Boulder, CO 80309-0430 USA
E-mail: {bauerk,grunwald,sicker}@colorado.edu

37th Research Conference on Communication, Information and Internet Policy, September 2009.

Contents

1. INTRODUCTION

Peer-to-peer (P2P) networks have become increasingly popular for a variety of applications including streaming multimedia, voice-over-IP telephony, and file sharing. The P2P content distribution model offers several advantages over traditional client/server architectures including data replication, fault tolerance, and increased efficiency and performance. However, experience has shown that P2P file sharing networks are often used to distribute copyright protected media files illegally [48]. P2P networks are an ideal mechanism to facilitate this type of file sharing because the data is replicated throughout the network, rather than being hosted on a single centralized server that can be easily located and shut down.

There are several policy issues surrounding the proliferation of P2P networks. We focus our discussion on two classes of problems. First, P2P networks ostensibly distribute copyright protected files illegally. Consequently, to mitigate the lost revenue resulting from piracy, the respective copyright holders have a strong desire to find and stop the users who are responsible. The second problem presented by P2P networks is one of network management. P2P networks – and the popular BitTorrent protocol in particular – are notoriously greedy with regard to bandwidth consumption. Excessive P2P traffic can cause significant network management problems for ISPs who aim to provide a consistent degree of quality of service for their customers. As a result of these problems, both the copyright holders and the ISPs have a mutual interest in mitigating P2P usage.

In this paper, we seek to understand the methods used by the copyright holders and ISPs to identify P2P usage and discourage it. We restrict our discussion to BitTorrent – the most widely used P2P protocol currently available for file sharing – however, our discussion generalizes to most P2P file sharing networks. We first summarize the past and present large-scale BitTorrent monitoring practices conducted by investigative organizations acting on behalf of various copyright stakeholders. We next examine how ISPs from around the world have attempted to mitigate BitTorrent usage on their networks by using protocol identification and traffic shaping techniques.

Having described the current climate of BitTorrent monitoring and traffic throttling, we identify the common counter-measures that BitTorrent users have applied to avoid identification by copyright investigators and to defeat traffic shaping. We observe that an “arms race” of tactics is emerging with each side, the P2P users and the copyright holders/ISPs, developing more sophisticated techniques to defeat the other side’s current strategy. We also speculate about future tactics that may emerge on both sides of this arms race. Finally, having discussed the nature of the P2P arms race, we examine but a few of the many policy implications and consider the impact the arms race may have on future broadband and copyright policy.

2. THE PEER-TO-PEER CONTENT DISSEMINATION MODEL

The peer-to-peer (P2P) data dissemination model has become widely popular for a variety of applications including streaming multimedia [55], voice-over-IP (VoIP) [50] and file sharing [7; 15; 28]. This is largely a consequence of the many benefits that P2P architectures offer in comparison to the traditional client/server communication model. In this section, we enumerate the significant differences between the traditional client/server communication model and P2P from a general perspective.

2.1 Client/Server Architecture

Protocols built on the client/server model consist of a single centralized server that hosts content and several clients that consume the content. This approach is simple and convenient. However, as the number of clients increases, the server may become overwhelmed by the many requests and, thus, client/server architectures do not scale well. Furthermore, if the server experiences a failure, its content may be lost and the clients cannot complete their requests. Therefore, the client/server architecture does not tolerate server failures. Finally, the content on a centralized server is easy to find and remove, if the content is deemed to be of an inappropriate nature (for example, copyright infringing content). Therefore, client/server architectures are often easy to censor.[1]

2.2 P2P Architecture

To address the limitations of the client/server model, protocols built on the P2P model have become popular recently. In this model, participants, called *peers*, act as both a client and a server. Peers not only consume data, but also bear partial responsibility for hosting the content. However, as a consequence of this dual role, the content is decentralized and often replicated over a set of peers, providing inherent tolerance to failures and a high degree of data availability. Furthermore, content may be difficult to locate and remove, since it may be replicated across a large portion of the network. This feature has contributed to P2P protocols becoming popular for file sharing applications, particularly involving copyright protected content.

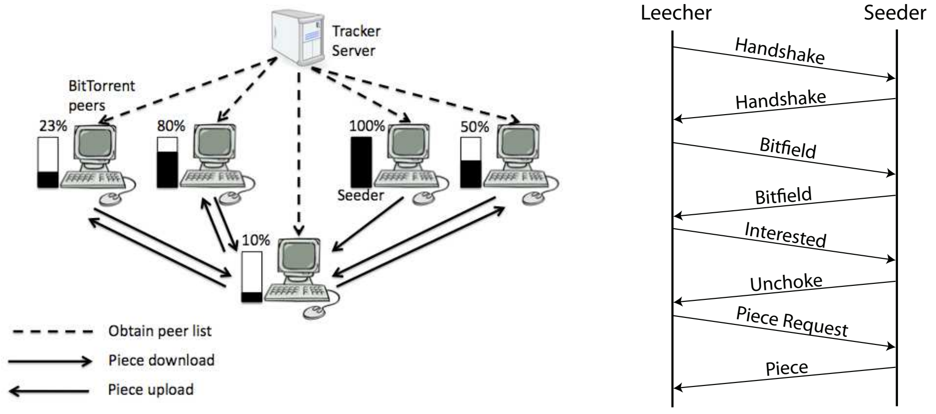
3. BITTORRENT BACKGROUND

BitTorrent is the most popular P2P protocol for file sharing applications [28]. Sharing a file with BitTorrent is very simple and proceeds as follows:

(1) The file is broken into several fixed-sized *pieces*. A cryptographic hash is computed for each piece to ensure integrity as pieces are shared. This aims to prevent data corruption by malicious peers who distribute invalid pieces. Pieces are further sub-divided into fixed-sized *blocks* which are typically 16 KB in size.

(2) To advertise a file's availability for download with BitTorrent, a metadata file is created and published on the web. This metadata file contains a unique identifier called an `info_hash` that is derived from the semantic description of the file, the cryptographic hashes of each piece, a link to a *tracker server* that helps to organize the peers, the number of pieces, and the piece size. Optionally, other information may be included. The trackers are operated by organizations such as The Pirate Bay [54].

[1] Websites such as Google, Amazon, and Akamai go to elaborate lengths to overcome the various limitations of the client/server communication model by applying what are essentially "peer-to-peer-like" techniques for balancing the traffic load and selecting an optimal server for a particular client request.



(a) A file transfer with BitTorrent

(b) Message exchange to initiate a block transfer

Fig. 1. BitTorrent overview

(3) Once the metadata file has been obtained, a new peer wishing to start sharing the file queries the tracker server to obtain a list of other peers who are currently sharing the file. Since the peer list may be arbitrarily large, the tracker typically replies with a fixed number of randomly selected peers. This also helps to balance the traffic load over the participating peers.

(4) The peer requests specific parts of the file, in a non-sequential manner, addressed by piece number and offset.

The tracker functionality may be implemented as a centralized server, a distributed hash table, or a gossip-based peer discovery mechanism and uses a standard HTTP interface. By querying the tracker, a peer implicitly registers itself with the tracker's peer list and may subsequently receive requests from other peers for particular parts of the file that the peer has obtained. In BitTorrent's vernacular, peers who possess a full copy of the file being shared are called *seeders* and peer who are still downloading are called *leechers*. The general protocol behavior is shown in Figure 1(a).

Once a peer has completed a download, they may continue to participate in the protocol to help other peers download, or they may leave. BitTorrent does attempt to mitigate selfish peer behavior by incorporating a "tit-for-tat" mechanism into the piece request process. In essence, it attempts to promote fairness and reciprocity in the piece sharing.

3.1 Protocol Messages

To initiate a piece transfer, the following sequence of request-reply messages are exchanged (shown in Figure 1(b)):

(1) A leecher sends a handshake message to another peer (who may be another leecher or a seeder). The handshake contains the `info_hash` value that identifies

the file being shared and a unique peer_id field that identifies the peer and provides information about the peer's BitTorrent client.[2]

(2) After the handshake, peers exchange bitfields. A bitfield is a bit-string data structure that concisely encodes precisely which pieces of the file that the peer has obtained. With this knowledge, the requesting peer can ask for specific parts of the file that the other peer possesses.

(3) If the requesting peer desires to obtain any of the pieces that the responding peer has advertised, the peer sends an "interested" message.

(4) If the responding peer is able to fulfill the request – *i.e.,* if they have sufficient available bandwidth or download slots to satisfy the request – the responding peer sends an "unchoke" message.

(5) Upon receipt of an unchoke message, the requesting peer is free to begin downloading. The requesting peer sends a request message containing the piece number, offset, and length of the data they wish to obtain.

(6) The responding peer sends the data requested.

Using the basic request-reply protocol outlined above, peers may issue requests for arbitrary parts of the file from several other peers in parallel. To minimize download time, the protocol behaves aggressively with regard to requesting pieces. Unless the BitTorrent client applies an artificial rate limiting mechanism, the protocol is designed to transfer data as fast as possible, and consequently often consumes as much bandwidth as available. BitTorrent's aggressive nature is often described as *swarming* behavior.

3.2 Two Problems That BitTorrent Presents

BitTorrent has achieved great popularity for its ability to quickly and easy share files. However, there are many significant challenges that it presents for network operators and copyright enforcement agencies.

The first challenge is one of network management and is a direct consequence of BitTorrent's aggressive network behavior. Peers sharing content utilize a significant amount of bandwidth, in both the down and up-load directions. Broadband Internet service providers (ISPs) have an obligation to their customers to provide reasonable quality of service, even during times of peak utilization. Excessive BitTorrent usage on these networks complicates effective network management. Consequently, broadband ISPs from around the world have adopted various policies of throttling or explicitly blocking BitTorrent traffic [10].

The second challenge that BitTorrent poses is one of copyright enforcement. Experience has shown that BitTorrent's decentralized nature has made it an attractive vehicle for users wishing to share copyright protected media content such as popular music, movies, and television programs. In the client/server computing model, copyright holders or law enforcement agencies may easily identify infringing content and request the hosting server to remove the content, or face a penalty defined by the local legal system. However, with a decentralized P2P protocol like BitTorrent, it is challenging for copyright holders or law enforcement agencies to contact

[2] Some BitTorrent clients implement additional features beyond the basic specification, so knowing another peer's client implementation enables peers to utilize implementation-specific features.

each individual peer and request that the content be removed, since there could be millions of peers distributed throughout several legal jurisdictions. Definitively identifying peers who participate in illegal file sharing is a significant challenge.

Despite the challenges that BitTorrent presents for network management and copyright enforcement, extensive efforts have been made by ISPs and copyright enforcement agencies alike to curtail the proliferation of BitTorrent usage. To this end both groups have engaged in large-scale network monitoring practices aimed at identifying BitTorrent traffic on ISP networks or identifying BitTorrent users. As a result of the attempts made to mitigate this type of file sharing, file sharers have begun to utilize tools to hide the nature of their traffic using data confidentiality techniques and even hide their identities using anonymity tools. In the remainder of this paper, we explore the past techniques used by ISPs and copyright enforcement agencies to detect and mitigate BitTorrent usage. We then describe how file sharers' tactics have evolved to combat their tactics.

4. P2P INVESTIGATIONS

BitTorrent has recently become a target of large-scale network monitoring practices aimed at either 1) ISPs throttling the bandwidth consumption that is associated with BitTorrent traffic, or 2) copyright protection entities identifying individual users who actively share content. In this section, we summarize the techniques used by each respective party.

4.1 Broadband ISP Tactics

Many ISPs from around the world have recently adopted policies aimed at actively identifying BitTorrent traffic flowing through their networks and explicitly blocking or throttling it. A common approach has been to deploy in-network traffic analysis tools such as Sandvine [20] to identify BitTorrent traffic by its recognizable application-layer protocol header and specifically target TCP flows identified as BitTorrent with forged TCP reset packets.[3] This causes the TCP connection to abruptly close. This tactic has been criticized by network neutrality proponents and consumer advocates in part because of the lack of transparency and disclosure regarding these practices. Furthermore, in August 2008 the United States Federal Communications Commission (FCC) ruled that Comcast had inappropriately interfered with its customers' ability to access content and use applications freely by not treating all types of Internet traffic equally [24].[4] Comcast is not alone in this practice, as this network management strategy has been common in most regions of the world [10]. In response, researchers have developed a variety of techniques and tools to detect packet forgery and other abnormal traffic manipulation by ISPs [35; 53; 56].

Given the nature of network management that actively identifies and throttles BitTorrent traffic, BitTorrent users have incentive to hide the protocol. In Sec-

[3] Incidentally, similar techniques have been used by the Chinese government's so-called "Great Firewall of China" to filter its citizens' Internet access [30].

[4] In response to the FCC's ruling, protocol-agnostic traffic management strategies such as periodic bandwidth caps and tiered pricing models have been proposed.

tion 5.1, we describe the counter-measures that have been applied to combat traffic shaping.

4.2 Copyright Holder Tactics

Given the reality that P2P protocols are often used to share copyright protected content, representatives of the copyright holders such as the Motion Picture Association of America (MPAA) and the Recording Industry Association of America (RIAA) have demonstrated significant interest in identifying the parties who host content illegally and request that the content be removed. However, due to the decentralized nature of BitTorrent, the task of identifying every peer involved in sharing a file in question is a challenging task for a variety of reasons. First, it is necessary to obtain a list of every peer – identified by IP address – who has participated in the file sharing. Second, it is necessary to correctly identify the actual person behind that IP address.

Despite these challenges, organizations such as the MPAA and RIAA have hired investigative entities including BayTSP [27], Media Defender [41], and Safenet [46] to identify users participating in file sharing of copyright protected content. A recent study explored how these investigations are conducted and how users are identified [43]. This study found that these investigators typically rely on passive methods including querying tracker lists for a peer list. Once the peer list has been obtained, an ICMP (ping) message is sent to each IP address listed by the tracker to ensure that it exists and is active. After pruning the peer list to contain only those peer IP addresses that respond to ping messages, the investigators have often distributed Digital Millennium Copyright Act (DMCA) [5] take-down letters to network operators, asking for them to be forwarded to the users responsible. In some instances, identification in a tracker's peer list has been sufficient evidence to initiate legal proceedings against alleged copyright violators.[5]

However, as is noted in the study, the passive methods used to collect evidence can be wildly inaccurate. BitTorrent's tracker protocol uses a standard HTTP interface where the peer's IP address can be provided as an HTTP parameter to the tracker URI. Consequently, it is easy to register arbitrary IP addresses with the tracker server. This can be regarded as a simple pollution attack against the trackers. The ease with which trackers can be polluted by fake peers is poignantly demonstrated when the authors of the study registered networked printers and wireless access points – devices that clearly lack the ability to participate in the BitTorrent protocol – to tracker lists and subsequently received DMCA take-down notices for their suspected participation.

Other sources of false identification are possible beyond active tracker list pollution. For example, consider the case where a user obtains a new DHCP lease for

[5] Here are the details of two of the cases that have recently gone to trial: In the case of *Capitol v. Thomas*, a jury concluded that the defendant had willfully infringed upon the copyrights of 24 songs by sharing them on the Kazaa network. The plaintiff was awarded statutory damages totaling $1.92 million, or $80,000 per song [23]. In *Sony BMG Music Entertainment et al. v. Tenenbaum*, the defendant was ordered to pay $675,000 in damages for downloading 30 songs, or $22,500 per song [14]. While the defendants in these cases did not use BitTorrent to share the files in question (they used the Kazaa file sharing network), similar investigative techniques akin to tracker list monitoring were employed.

an IP address whose previous user had participated in file sharing that was under surveillance. This new user may be accused of participating in the file sharing since their IP address could still be listed by the tracker at the time when it is queried by the investigator. Despite the real potential for false identification, recent attention has been given to developing mechanisms to enable BitTorrent users to shed their network identities and participate anonymously, without fear of surveillance or legal consequence.

In December 2008, the music industry announced that it planned to discontinue its past practice of pursuing legal action against alleged file sharers. Instead, the RIAA plans to partner with ISPs to help inform users of suspected copyright infringement [18]. It is also possible that future strategies for identifying file sharers may include deploying in-network hardware to inspect traffic and identify illicit content at an ISP gateway [21]. By inspecting the BitTorrent header, it is possible to identify content by its `info_hash` header field. Specialized hardware could examine BitTorrent traffic for a set of illicit or copyright-protected `info_hash` values at line-rate within an ISP. This approach may provide better precision than tracker list queries. However, it would require cooperation between ISPs and copyright investigation agents. Furthermore, this simple content identification technique could be easily circumvented by countermeasures such as BitTorrent encryption. In the next section, we extend the "arms race" metaphor and discuss how such countermeasures are currently being used by file sharers to avoid protocol and content identification.

5. RESPONSE TO ANTI-P2P CAMPAIGNS

Given the recent climate of large-scale network surveillance of P2P networks by ISPs and copyright enforcement organizations, users who wish to share copyright protected content now have significant incentive to hide the nature of their traffic, or even attempt to conceal their own identities. In this section, we describe the primary methods applied by P2P users to evade the tactics employed by ISPs and copyright infringement investigators.

5.1 Concealing BitTorrent from ISPs

In an effort to provide confidentiality for file transfers and combat traffic shaping, BitTorrent has been extended to support application-layer security. BitTorrent's *Message Stream Encryption* (MSE) allows peers to establish encrypted connections to each other [42]. The protocol can obfuscate BitTorrent's application-layer header to help prevent traffic shaping by ISPs and may optionally encrypt the data payload to increase the user's privacy from eavesdroppers. To establish a secure connection between a pair of peers (p_1, p_2) using MSE, the protocol proceeds as follows:

(1) To establish a shared symmetric session key, p_1 initiates a Diffie-Hellman key establishment protocol [32] with p_2. To mitigate man-in-the-middle attacks (MiTM)[6] during the key establishment, the torrent's `info_hash` value is used as a pre-shared "secret key" to provide a low degree of mutual authentication. This "secret key" ensures that both parties are sharing the same file.

[6] For instance, the user's ISP could be regarded as a man-in-the-middle.

(2) The peers p_1 and p_2 decide whether to encrypt the protocol header only, or also to encrypt the payload.
(3) Once p_1 and p_2 have agreed on a session key, the protocol proceeds as normal, except that all messages are RC4 encrypted using the session key (RC4 is chosen over AES for its speed).

With MSE, the BitTorrent header (and optionally the payload) appears to be random to a third-party eavesdropper who does not possess the secret session key. Consequently, this primitive security mechanism effectively defeats the simple pattern matching detection methods used by Sandvine and other protocol analyzers to identify and subsequently shape BitTorrent traffic.

5.1.1 *Limitations.* While MSE may be sufficient to defeat the past and current protocol identification techniques, it provides a very weak form of security. Message stream encryption's primary design goal is to provide protocol obfuscation, *not* to enable security in terms of strong authentication, integrity, and confidentiality as is provided by strong security protocols such as SSL/TLS [31]. As a consequence, there are several simple ways that an adversary can defeat the protocol. MSE's vulnerabilities include the following:

—*The key establishment is not (really) authenticated.* While the protocol uses the `info_hash` value as a pre-shared key, this does not ensure that the other party in the key establishment is really who they say they are. The `info_hash` value is stored in the torrent's metadata file and is publicly available to anyone. Knowing the `info_hash` proves that the peer is sharing the correct file. However, it is possible for a third-party such as the ISP to observe the `info_hash` value from the metadata and use it to perform a MiTM attack during the key establishment phase. Given this vulnerability, a third-party can observe all data exchanged between the peers. However, as a possible countermeasure in the arms race, the file sharers could obtain the torrent metadata through an alternate secure channel or at a time significantly prior to the start of their download.

—*Message integrity is not ensured.* If the ISP is able to identify the encrypted BitTorrent traffic, it is possible to alter packets in flight. MSE provides no way to detect manipulation of data in flight. Consequently, it may be possible for an ISP to introduce noise in the form of random errors in the payload, which could cause each retrieved piece to fail the integrity check used by BitTorrent to detect piece pollution.

—*Vulnerable to traffic analysis using packet sizes and timing.* Despite even the best security practices, it is often possible to use side channel information such as packet sizes, counts, and timing information to identify the underlying application-layer protocol, and even the content, in some cases. Examples of the types of information that may be inferred by a third-party eavesdropper from observed secure connections include videos watched [47], passwords typed [51], web pages viewed [39; 52], languages and phrases spoken [57; 59], and applications run [58]. Statistical or machine learning-based techniques may be applied to uniquely identify BitTorrent traffic despite even the best known data confidentiality practices.

In summary, BitTorrent's current encryption protocol is *reactionary* and tailored *only* to defeating past and present protocol detection and traffic shaping methods (*i.e.,* Sandvine) and provides very weak security in the traditional sense. We next discuss the current methods that are used to evade the investigations conducted by copyright enforcement agents.

5.2 Evading Copyright Authorities

Recently, BitTorrent users who share copyright protected content have been targeted with DMCA take-down letters and even experienced legal action initiated by the respective copyright holders. In Section 4.2, we outlined the investigations that have been conducted to identify these file sharers. Given that climate, P2P users have increased incentive to shed their network identities and be anonymous. In this section, we describe the current techniques that may be used to achieve this kind of anonymity.

Tor is the most popular anonymizing network in use today [33]. Tor provides low-latency anonymity for TCP-based applications by allowing users to source-route their traffic through a virtual circuit of Tor routers using a layered encryption scheme based on onion routing [36]. With Tor, only the first Tor router (called the entry guard) knows the user's true network identity, and only the last (or exit) Tor router knows the destination of the user's traffic. Consequently, from the destination server's perspective, it appears that the traffic originated at the exit Tor router. In order to determine both the client and destination identities, it is necessary for the first and last Tor routers to collude and apply timing analysis [25]. Furthermore, if session-layer or application-layer security mechanisms (such as SSL/TLS) are not used, the exit Tor router can observe the contents of the communications, which may also reveal user identities [40].

Tor is the best practical tool for achieving anonymity and is used widely for a variety of purposes including censorship resistance and the promotion of various freedoms within the jurisdictions of repressive political regimes. A recent study (by the authors of this paper) aiming to understand how Tor is used in practice found that BitTorrent is among the most popular protocols used with Tor [40]. Based on observations and analysis of a sample of traffic over four days (in December 2007) leaving a popular Tor exit router, the authors found that over 40% of the Tor router's bandwidth was consumed by BitTorrent traffic. This was second only to HTTP traffic. Tor effectively defeats the P2P investigations based on capturing tracker lists since the tracker list will contain the Tor exit router that was used by the BitTorrent user, keeping the user's true IP address hidden. While BitTorrent comprises a large portion of Tor's bandwidth, the study found that just over 3% of the TCP connections observed exiting Tor were identified as BitTorrent. This highlights BitTorrent's bandwidth-greedy behavior.

While Tor ensures a high degree of privacy by providing a strong form of anonymity to its users, it has recently been suggested that the significant amount of P2P traffic forwarded through Tor is negatively impacting Tor's ability to provide a low latency anonymous transport service for interactive traffic (such as web traffic) [34; 44]. Furthermore, the strength of the anonymity provided by Tor is sufficient to protect cyber dissidents and bloggers residing within repressive jurisdictions; however, it may be unnecessarily strong and have too much performance overhead to be practical for P2P users. Consequently, a variety of anonymity tools have recently been developed specifically to anonymize P2P traffic, potentially reducing the load on Tor and providing a sufficient degree of anonymity to evade copyright investigators and frustrate traffic shaping. In the next section, we summarize these tools.

6. EMERGING STRATEGIES TO HIDE P2P

Beyond application-layer security mechanisms and heavy-duty anonymity tools such as Tor, a variety of other techniques are beginning to emerge that aim to either 1) defeat protocol analysis and ISP throttling practices, or 2) increase file sharers' privacy and provide protection against copyright investigations. In this section, we outline the next iteration of the arms race: We describe several of the emerging tools and tactics that are becoming available to enable file sharers' to evade traffic shaping and P2P investigations.

6.1 Virtual Private Network-based Anonymizers

The Pirate Bay, a popular BitTorrent tracker operator, offers a virtual private network (VPN)-based anonymity service for BitTorrent traffic called iPredator [37]. BTGuard is another VPN-based service for anonymizing BitTorrent traffic [3]. The services are simple: Registered users pay a small monthly fee to forward their BitTorrent traffic through an encrypted VPN tunnel before communicating with the destination. The VPN's encryption provides protocol obfuscation, since the entire application-layer header and payload is encrypted. This mitigates the potential for traffic shaping. In addition, the traffic appears to have originated at the VPN server, thus the user's network identity is not advertised by the trackers. The services claim to keep no access logs and traffic is anonymous from the perspective of the trackers.

While this type of VPN solution would defeat the past and current protocol throttling and copyright investigation tactics, the VPN provides a weak form of anonymity. The single VPN server tunneling user traffic knows both the client's and destination's network identity, thus there is no anonymity from the VPN's perspective. Given this fact, VPN-based anonymizers may be easy targets for legal attacks where the operators are legally coerced to log the traffic that is forwarded. With stronger decentralized anonymity tools such as Tor, this type of attack is significantly more difficult.

6.2 BitBlender: Crowds-style Anonymity for BitTorrent

Another way to provide anonymity for BitTorrent was proposed by the authors of this paper [26]. BitBlender achieves a degree of anonymity for BitTorrent users by forwarding traffic through one or more *relay peers* that merely act as proxies on behalf of the normal peers who are sharing a particular file. The fundamental idea behind BitBlender was first proposed by Crowds, a distributed anonymizing network for HTTP traffic where source anonymity is achieved by forwarding traffic through proxies that either deliver the message to the destination server, or forward the message to another proxy with a certain probability [45]. The relay peers are advertised by a special directory server.

BitBlender achieves the condition of *plausibility deniability* for peers listed by the trackers. Given only the tracker list, an investigator cannot be sure about which peers are real file sharers, and which peers are acting as relays. Additional traffic analysis techniques are necessary to determine the set of real, active peers [60].

BitBlender's relay peers introduce an intriguing question of legality and policy. Do peers that act as proxies have any legal liability for the content that they forward? Put another way, if a relay peer forwards part of a copyright protected media file, can the relay peer be held legally responsible for facilitating the transfer of such content? We provide additional discussion of this issue in Section 7.1.

6.3 SwarmScreen

A recent study has observed that despite various attempts to promote privacy in BitTorrent using encryption, a phenomenon emerges in which users form natural communities based on their download interests [29]. For example, using information about which peers established connections with which other peers, the study shows that it is possible to classify users by their download interests by observing their connections. A "guilt-by-association" attack is possible where the download interests of a community of users can be accurately classified by observing a small fraction of peers (or a single peer).

To address the loss of privacy that knowledge of peer communities causes, SwarmScreen is proposed [29]. The intuition behind this system is that peers can achieve plausible deniability with regard to their community membership by establishing random connections to other peers sharing a variety of types of content that are difficult to distinguish from real connections. The fraction of random connections added to a peer's normal behavior is called "SwarmScreen Protection Factor." Higher SPF values offer more privacy than lower values – *i.e.,* it is more difficult for an observer to ascertain the peer's true file sharing behavior.

Like BitBlender, SwarmScreen also presents new questions to consider regarding legality. For instance, peers that transfer parts of files chosen at random may unknowingly share part of an illicit file. Can peers who obey the SwarmScreen protocol be held responsible if they share illicit or copyright protected content *unknowingly* or *without explicit intent to redistribute*?

6.4 OneSwarm: A Friend-to-Friend Network Built on Social Links

Another approach to improving privacy within P2P networks is to restrict the set of peers with whom data can be shared. In BitTorrent, a peer may share data with any other peer who requests data. As a consequence of this openness, P2P users are vulnerable to monitoring by copyright enforcement organizations. OneSwarm proposes a *Friend-to-Friend* (F2F) network in which users take advantage of their social connections to define their sharing relationships [38]. For example, a peer can choose to share files with their friends, family, or co-workers. By giving users the ability to control their sharing behavior with fine granularity, users achieve greater privacy by reducing the threat of monitoring. This type of system poses questions regarding fair use of content which we discuss in detail in Section 7.3.

6.5 BitTorrent Over UDP Transport

Another radical method to avoid past and present traffic shaping practices is to re-design the BitTorrent protocol to run on top of the connectionless, best-effort delivery UDP transport protocol [17]. A BitTorrent variant that is built on top of

UDP is ostensibly not vulnerable to forged TCP RST traffic throttling methods. However, it is still possible to throttle UDP traffic in a protocol-agnostic manner.

6.6 Hiding the Trackers and the Torrent Metadata

BitTorrent's vulnerability to large-scale monitoring is a consequence of its reliance on well-known and publicly accessible tracker servers. Furthermore, tracker operators such as The Pirate Bay have recently faced severe legal consequences for hosting the necessary infrastructure to enable the distribution of copyright protected content. Future strategies for hiding trackers may include using location hidden services provided by Tor [33]. One such service has already emerged called Hidden Tracker that hosts BitTorrent trackers as Tor hidden services [11]. The legal action against The Pirate Bay is discussed in Section 7.2 in detail.

BitTorrent also requires that the torrent metadata be published through an out-of-band mechanism, such as the web. However, this metadata can be easily identified and removed. Consequently, it is foreseeable to expect techniques to obfuscate the torrent metadata using encryption. Hidim [12] is a tool that converts a torrent metadata file into a steganographically encoded image that can be posted to a blog or social networking page. To avoid automated identification by copyright enforcement organizations, P2P users could use such techniques to hide the existence of the torrent metadata.

6.7 Turning the Tables: Blacklisting the BitTorrent Monitors

Another way that BitTorrent users may find shelter from BitTorrent monitoring agencies is to develop methods for detecting the monitors and blacklisting them. Siganos *et al.* analyzed the top 600 torrents from The Pirate Bay for 45 days and developed a set of heuristics for identifying and dynamically blacklisting peers that exhibit anomalous behavior, or peers that may be monitoring rather than participating in the file sharing [49]. Examples of these heuristics include peers that do not accept connections, percentage of active peers within a single autonomous system (AS), multiple clients on a single IP address, and frequency of IP address occurrence across a large number of diverse torrents. However, and perhaps somewhat ironically, the investigators themselves could use anonymization tools such as Tor to blend in with other peers who are using such tools.

While these heuristics can be applied to generate dynamic blacklists to prevent peers from communicating with suspicious or anomalous peers, it does not prevent their network identities from being advertised by the trackers. However, these blacklists could be more useful if the monitoring agencies assumed a more *active* strategy for identifying real peer participation in illegal file sharing. An active strategy where the monitors communicate with peers to confirm their participation could be one way to reduce the potential for the false positives that are possible with the past and current passive peer identification methods.

7. THE P2P ARMS RACE AND ITS POLICY IMPLICATIONS

Having presented an overview of BitTorrent and some of the challenges it poses for network management/copyright enforcement and a survey of the current and emerging tactics used by file sharers to hide, we next discuss a sampling of the policy issues that have surfaced in this discussion. This discussion is certainly not exhaustive, however we hope to elucidate a few interesting policy issues and encourage additional dialogue with regard to future policy decisions.

7.1 Potential Legal Issues Involving Anonymizing Networks

The legality of participating in an anonymizing network such as BitBlender (see Section 6.2) ultimately boils down to questions regarding the legality of operating an open relay. While operating a relay peer does not necessarily enable or encourage others to commit copyright infringement, many legal systems have some notion of *indirect liability.* More specifically, such liabilities can be dichotomized into two general legal classifications: *vicarious liability* and *contributory liability.*

Vicarious liability may occur when a third-party has the means by which to influence or control the actions of another party. Some users may perform certain actions behind the shroud of anonymity that they would not do if their identities were to be revealed. Consequently, one could argue that anonymizing networks may not only enable misbehavior, but they may encourage it. It is unclear whether the operators of anonymizing network infrastructure (such as Tor routers, BitBlender relay peers, etc.) could be held vicariously liable for the potentially illegal actions of the system's users.

Another form of legal liability – contributory liability – may occur when a third-party has encouraged or otherwise has knowledge of illegal actions and fails to appropriately prevent the illegal behavior. One example of how an entity may be found to be contributorily liable is if an ISP receives a notice that they are hosting some copyright infringing content. Compliance with the DMCA requires that the ISP attempt to remove the infringing content. There is, however, a provision in the DMCA that protects ISPs from liability for content that they may *transport or cache* [1].[7] It is unclear whether anonymizing network operators could be liable for transporting illicit content – there is little legal precedent.[8] One could argue that the anonymizing network is effectively an ISP. Thus, it is possible that anonymizing network operators may also find protection under this provision.

7.2 Tracker Hosting and the Recent Pirate Bay Decision in Sweden

The Pirate Bay is among the largest BitTorrent tracker hosting services available, claiming to track tens of millions of unique peers [54]. Located in Sweden, The Pirate Bay operates many of the servers that are responsible for advertising the network addresses of peers participating in BitTorrent file transfers. The Pirate Bay's operators were recently targeted with both civil and criminal proceedings brought by the Swedish government and a variety of media companies including Sony, Warner Bros, and Columbia Pictures. The principal charge was that The Pirate Bay was helping to make copyright protected content available by hosting the necessary infrastructure to support illegal file transfers with BitTorrent [4].

In April 2009, The Pirate Bay's four operators were convicted in a Swedish district court of contributory copyright infringement. The operators were sentenced to one year in jail and ordered to pay $3.6 million in damages and fines [19].

[7] This provision was reaffirmed when Google's caching policy was challenged [8].

[8] However, the court ruling against the Grokster P2P file sharing network indicates that a service is likely to be found liable if it is marketed as a tool for committing copyright infringement [16].

This verdict represents perhaps the first legal precedent regarding the legality of operating BitTorrent tracker servers. One could argue – as the defendants did – that hosting tracker servers merely provides an information service and does not engage in the transfer of any copyright protected content. However, as made clear by the trial's outcome, the operators were found guilty of essentially being accessories to copyright infringement. Of course, copyright laws vary across different legal jurisdictions. To avoid prosecution, it may be possible for a tracker hosting service to move to another legal jurisdiction where the local laws are more favorable to their activities. In addition, a more radical tactic to avoid legal action could be to operate trackers as location hidden services (as previously described in Section 6.6).

7.3 Fair Use and P2P/F2F File Sharing Networks

To mitigate privacy concerns that may arise from the climate of P2P monitoring by copyright enforcement agents or ISPs, small-scale Friend-to-Friend (F2F) networks have been proposed. In Section 6.4, we outlined the design of OneSwarm, an F2F network that leverages a user's social communities to define the extent and scope of users' file sharing. Such a system, however, brings up questions of fair use of copyright protected content. Software such as iTunes allows users to sample content before purchasing by playing a short streaming sample. However, such software does not download the entire content.

In the P2P file sharing case of *BMG Music v. Gonzalez* in 2005, the defendant claimed that she was merely sampling potential content for later purchase by downloading it via the Kazaa file sharing network as may be allowed by section 17 U.S.C. 107. However, since the defendant downloaded the content while other sampling means were available (*i.e.,* iTunes), the fair use exemption did not apply [2]. F2F networks could present similar arguments with regard to fair use, but this case indicates that simply downloading content – whether from a small group of friends or a large P2P file sharing network – may not fall under fair use.

7.4 P2P File Sharing and the "Three Strike Law"

In France, a controversial "three strike" law for file sharing has been proposed. The law essentially works by informing users that they are suspected of participating in file sharing of copyright protected content. After issuing two such notices within a certain time period, a third notice results in the user's Internet service being discontinued [9]. The reasoning behind allowing three strikes is to warn the user and to ensure that the user is not wrongly implicated, for instance, if the user has an open wireless access point that is used by someone else to download copyright protected content. Similar laws have also been proposed in South Korea [22], Ireland [6], and Italy [13]. As previously discussed in this paper, there are a variety of very real possibilities for false positive content and protocol identification. Furthermore, encryption and anonymity tools are becoming more popular for hiding traffic characteristics and user identities. Consequently, these three strike laws and variants may face significant enforcement challenges.

8. CONCLUSION

We explored the past and present tactics employed by ISPs and copyright enforcement organizations to mitigate the proliferation of file sharing with BitTorrent. These tactics include protocol analysis and throttling by ISPs and large-scale tracker monitoring campaigns conducted by investigative agents acting on behalf of copyright stakeholders aimed at identifying file sharers and sometimes even prosecuting them. In response to these tactics, we observe that BitTorrent users have attempted to hide the nature of their traffic and employed anonymity tools to help protect their network identities to evade copyright investigators. In presenting the details of this arms race, our hope is to promote further dialogue with regard to the legal and policy issues that have surfaced.

REFERENCES

[1] 17 United States Code Section 512. http://www4.law.cornell.edu/uscode/17/512.html.

[2] BMG Music v. Gonzalez. http://www.llrx.com/features/bmgvgonzalez.htm.

[3] BTGuard. http://btguard.com.

[4] Court nears decision on file-sharing web site. http://online.wsj.com/article/SB123957384486211849.html.

[5] The Digital Millennium Copyright Act of 1998. http://www.copyright.gov/legislation/dmca.pdf.

[6] Eircom Irma briefing note March 2009. http://www.scribd.com/doc/13630351/Eircom-Irma-Briefing-Note-March-2009.

[7] eMule. http://www.emule-project.net.

[8] Field v. Google, Inc., 412 F. Supp 2d. 1106 (D. Nev. 2006).

[9] French pass 'three strikes' file-sharing law. http://www.theregister.co.uk/2009/04/03/french_three_strikes.

[10] Glasnost: Results from tests for BitTorrent traffic blocking. http://broadband.mpi-sws.org/transparency/results.

[11] The hidden tracker. http://twitter.com/HiddenTracker.

[12] Hidim. http://www.hid.im.

[13] Italian politicians look to push through French-style 3 strikes law. http://www.techdirt.com/articles/20090121/0834043477.shtml.

[14] Joel Tenenbaum joins the I-Owe-the-RIAA-My-Firstborn club. http://latimesblogs.latimes.com/technology/2009/08/joel-tenenbaum-riaa.html.

[15] Limewire. http://www.limewire.com.

[16] MGM Studios Inc. v. Grokster, Ltd., 545 U.S. 913 (United States Supreme Court 2005).

[17] New UDP μTorrent takes aim at throttling. http://www.dslreports.com/shownews/99366.

[18] No ISP filtering under new RIAA copyright strategy. http://www.wired.com/threatlevel/2008/12/no-isp-filterin.

[19] The pirate bay guilty; jail for file-sharing foursome. http://www.wired.com/threatlevel/2009/04/pirateverdict.

[20] Sandvine incorporated: Intelligent broadband network management. http://www.sandvine.com.

[21] Sniffing out illicit BitTorrent files. http://www.technologyreview.com/computing/22107.

[22] Upload a song, lose your Internet connection. http://www.koreatimes.co.kr/www/news/tech/2009/04/133_42594.html.

[23] Virgin Records America, Inc. v. Thomas. http://news.justia.com/cases/featured/minnesota/mndce/0:2006cv01497/82850.

[24] Commission orders Comcast to end discriminatory network management practices. Federal Communications Commission Press Release, August 2008.

[25] Bauer, K., McCoy, D., Grunwald, D., Kohno, T., and Sicker, D. Low-resource routing attacks against Tor. In *Proceedings of the Workshop on Privacy in the Electronic Society (WPES 2007)* (October 2007).

[26] Bauer, K., McCoy, D., Grunwald, D., and Sicker, D. BitBlender: Light-weight anonymity for BitTorrent. In *Proceedings of the Workshop on Applications of Private and Anonymous Communications (AlPACa 2008)* (September 2008), ACM.

[27] BayTSP. `http://www.baytsp.com`.

[28] BitTorrent protocol specification. `http://wiki.theory.org/BitTorrentSpecification`.

[29] Choffnes, D. R., Duch, J., Malmgren, D., Guierma, R., Bustamante, F. E., and Amaral, L. SwarmScreen: Privacy through plausible deniability in P2P systems. Tech. rep., March 2009.

[30] Clayton, R., Murdoch, S. J., and Watson, R. N. M. Ignoring the great firewall of China. In *Privacy Enhancing Technologies* (2006), pp. 20–35.

[31] Dierks, T. RFC 4346: The Transport Layer Security (TLS) Protocol Version 1.1, April 2006.

[32] Diffie, W., and Hellman, M. E. New directions in cryptography. *IEEE Transactions on Information Theory IT-22*, 6 (1976), 644–654.

[33] Dingledine, R., Mathewson, N., and Syverson, P. Tor: The second-generation onion router. In *Proceedings of the 13th USENIX Security Symposium* (August 2004).

[34] Dingledine, R., and Murdoch, S. Performance improvements on Tor or, why Tor is slow and what we're going to do about it. `http://www.torproject.org/press/presskit/2009-03-11-performance.pdf`, March 2009.

[35] Dischinger, M., Mislove, A., Haeberlen, A., and Gummadi, K. P. Detecting BitTorrent blocking. In *IMC '08: Proceedings of the 8th ACM SIGCOMM conference on Internet measurement* (October 2008), ACM.

[36] Goldschlag, D. M., Reed, M. G., and Syverson, P. F. Hiding routing information. In *Proceedings of Information Hiding: First International Workshop* (May 1996), R. Anderson, Ed., Springer-Verlag, LNCS 1174, pp. 137–150.

[37] Ipredator. `http://ipredator.se`.

[38] Isdal, T., Piatek, M., Krishnamurthy, A., and Anderson, T. Friend-to-friend data sharing with oneswarm. Tech. rep., February 2009.

[39] Liberatore, M., and Levine, B. N. Inferring the source of encrypted HTTP connections. In *CCS '06: Proceedings of the 13th ACM conference on Computer and communications security* (2006), ACM.

[40] McCoy, D., Bauer, K., Grunwald, D., Kohno, T., and Sicker, D. Shining light in dark places: Understanding the Tor network. In *Proceedings of the 8th Privacy Enhancing Technologies Symposium* (July 2008).

[41] Media defender. `http://www.mediadefender.com`.

[42] Message stream encryption. `http://www.azureuswiki.com/index.php/Message_Stream_Encryption`.

[43] Piatek, M., Kohno, T., and Krishnamurthy, A. Challenges and directions for monitoring P2P file sharing networks – or – Why my printer received a DMCA takedown notice. In *3rd USENIX Workshop on Hot Topics in Security (HotSec)* (July 2008).

[44] Reardon, J., and Goldberg, I. Improving Tor using a TCP-over-DTLS tunnel. In *Proceedings of the 18th USENIX Security Symposium* (August 2009).

[45] Reiter, M., and Rubin, A. Crowds: Anonymity for web transactions. *ACM Transactions on Information and System Security 1*, 1 (June 1998).

[46] Safenet Inc: The foundation for information security. `http://www.safenet-inc.com`.

[47] Saponas, T. S., Lester, J., Hartung, C., Agarwal, S., and Kohno, T. Devices that tell on you: Privacy trends in consumer ubiquitous computing. In *Proc. 16th USENIX Security Symposium* (2007).

[48] Schulze, H., and Mochalski, K. Ipoque Internet study 2008/2009. `http://www.ipoque.com/resources/internet-studies`.

[49] Siganos, G., Pujol, J. M., and Rodriguez, P. Monitoring the BitTorrent monitors: A bird's eye view. In *PAM* (2009), S. B. Moon, R. Teixeira, and S. Uhlig, Eds., vol. 5448 of *Lecture Notes in Computer Science*, Springer, pp. 175–184.

[50] Skype - Peer-to-peer VoIP application. `http://www.skype.com`.

[51] Song, D. X., Wagner, D., and Tian, X. Timing analysis of keystrokes and timing attacks on SSH. In *10th USENIX Security Symposium* (2001).

[52] Sun, Q., Simon, D. R., Wang, Y.-M., Russell, W., Padmanabhan, V. N., and Qiu, L. Statistical identification of encrypted web browsing traffic. In *IEEE Symposium on Security and Privacy* (2002).

[53] Switzerland network testing tool. `http://www.eff.org/testyourisp/switzerland`.

[54] The Pirate Bay. `http://thepiratebay.org`.

[55] Tran, D. A., Hua, K. A., and Do, T. T. A peer-to-peer architecture for media streaming. *Journal of Selected Areas in Communications* (January 2004).

[56] Weaver, N., Sommer, R., and Paxson, V. Detecting forged TCP reset packets. In *Proceedings of NDSS* (February 2009).

[57] Wright, C., Ballard, L., Monrose, F., and Masson, G. Language identification of encrypted VoIP traffic: Alejandra y Roberto or Alice and Bob? In *Proceedings of the 16th USENIX Security Symposium* (2007).

[58] Wright, C., Monrose, F., and Masson, G. On inferring application protocol behaviors in encrypted network traffic. *Journal of Machine Learning Research* (2006).

[59] Wright, C. V., Ballard, L., Coull, S. E., Monrose, F., and Masson, G. M. Spot me if you can: Uncovering spoken phrases in encrypted VoIP conversations.

[60] Wright, M. K., Adler, M., Levine, B. N., and Shields, C. The predecessor attack: An analysis of a threat to anonymous communications systems. *ACM Trans. Inf. Syst. Secur. 7*, 4 (2004), 489–522.