

White Paper

# Peer-To-Peer (P2P): Understanding the Insatiable File Sharing Technology

© 2007 Allot Communications Ltd. Allot Communications, NetEnforcer and the Allot logo are registered trademarks of Allot Communications. NetXplorer is trademark of Allot Communications. All other brand or product names are trademarks of their respective holders. All information in this document is subject to change without notice. Allot Communications Ltd., and/or its affiliates (collectively "Allot Communications") assume no responsibility for any errors that appear in this document.

## Abstract

Peer-to-Peer: it's amazing how one term can elicit such extreme reactions, depending on whom you're speaking to. Many consider P2P to be the content distribution medium for the future, outstripping every other communication and distribution protocol. For individuals, it's opened up a whole new world of communal file sharing, from music to videos to computer programs – mostly free and accessible through hundreds of P2P applications, available to anyone with a computer and Internet access. For music and film industry moguls, until recently it was considered the enemy, with declarations of illegality and lawsuits against "offenders". However, trends in these industries today are towards developing legal content downloading solutions. For enterprises, it's a nightmare, as more and more employees snatch a significant amount of bandwidth for recreational P2P use, at the expense of business-related applications. And for Internet service providers (ISPs), it's an epidemic, comprising 60-80% of their traffic day and night.

*"In most broadband networks, traffic from a minority of subscribers using P2P file sharing can cause poor performance of applications that are enjoyed by the majority of subscribers, such as web browsing."*

**James Crawshaw, Research Analyst, Light Reading Insider, August 2006**

## About P2P

### P2P: The Burden to Networks

P2P file-sharing applications, such as eDonkey (today also known as eMule) or BitTorrent, enable users to share content directly with each other. Increasingly, subscribers are using P2P applications to distribute multimedia content, such as audio and video files, which are extremely large and require significant network bandwidth. Unlike applications using the traditional client-server model where a well-known source provides content "downstreams" to requesting clients, P2P applications create heavy upstream traffic because all users' computers accept data requests. Heavy upstream traffic places a significant burden on asymmetric networks, such as digital subscriber line, or DSL, and cable networks, that were originally designed to handle only heavy downstream traffic.

Additionally, P2P applications are frequently left unattended for long periods of time while files upload and/or download, resulting in increased congestion during hours of peak network usage. According to an August 2005 article from GigaOM.com, a news and weblog for hi-tech consumers and professionals, on average 80% of upstream broadband capacity is consumed by P2P traffic. As a result, users of P2P applications dilute bandwidth for all users on the network, despite the fact that these users do not pay incremental amounts for increased bandwidth consumption. At the same time, P2P applications, which are not particularly sensitive to network delays, result in a diminished performance for latency-sensitive applications, such as VoIP, Internet video and online video gaming.

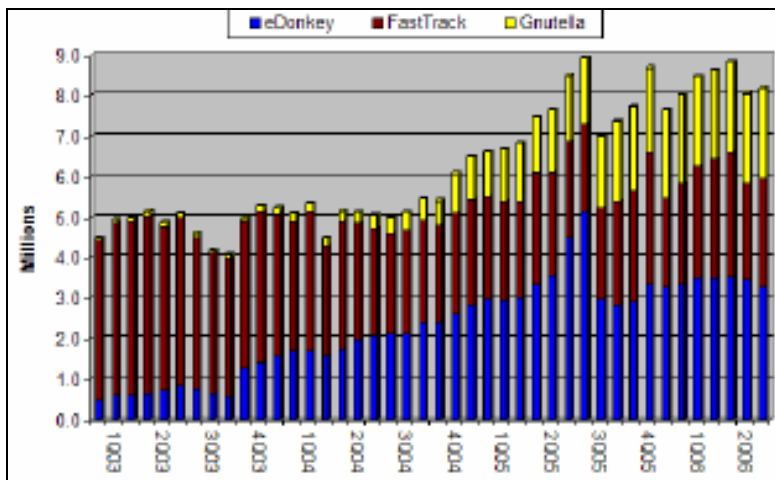


Figure 1: Increase of P2P users between January 2003 and June 2006 for 3 of the dozens of P2P services available today (source: Light Reading Insider, August 2006).

### P2P: More Than Just File Sharing

For many people, P2P relates to illegal file sharing. However, P2P is a lot more, because it really refers to a distributed system or network architecture. It does not have to be specifically for file sharing, and has other uses, such as for computers.

The significance of P2P is that there are basically many nodes. Each node is symmetric in function - meaning it can either create, receive or transmit and be part of another network. Furthermore, each node can take advantage of shared resources i.e., other peers with some computational power or storage or bandwidth. Finally, each node operates in a dynamic environment - meaning that the network tends to fluctuate and change over time.

### P2P Applications: A Paradigm Shift

P2P also brings a paradigm shift because it is a different concept to the controlled world of computers and networks, where everything is coordinated, centralized, and there is no incentive for users to be active participants. P2P is about the shift from coordination to cooperation; the shift from centralization to decentralization; and the shift from control to incentives.

In the P2P world, coordination is abandoned. P2P is about cooperating - about many nodes cooperating to achieve a common goal, such as copying or sharing a file. Consequently, P2P is the opposite of the centralized network approach, where there is some kind of central server trying to control the bandwidth and the applications that each and every client is permitted to use. With P2P, there is no "true" server focused on eliminating bottlenecks. However, some P2P protocols are far from really reaching this cooperation goal.

The recent news of the cooperation between BitTorrent and Time Warner concerning the issue of video streaming is a good example. The result of the problem faced by many of the video server services and applications, where everybody is trying to download the latest movie or TV series chapter at the same time, P2P offers a way to bypass this bottleneck. However, this is not true of every P2P application.

Finally, on the shift from control to incentives, many P2P applications are incentive driven. Such applications run on what is called “tit-for-tat”, which basically means that the more bandwidth you give for uploading, the more bandwidth you will receive for downloading from others. This incentive approach is rapidly replacing the early days of P2P applications, when it was possible to be what is commonly known as a “leech” i.e., only receive data, only take information from the network and then shut down or disconnect the computer from the P2P network – since such an approach will only be provided with a very low bandwidth.

*"50-65% of all download traffic is P2P. 75-90% of upload traffic is P2P. In 2004, one CacheLogic server registered 3m IP addresses in 30 days. In 2006, one CacheLogic server registered 3m in 8 days. Downloads comprised 61.4% video, 11.3% audio and 27.2% games/software. Average shared file size was 1 gigabyte."*  
 Source: Citigroup Global Markets, Equity Research, September 2006

### Recent P2P History

The current moves associated with P2P go back to Napster in the late 1990’s, which led to the proliferation of Gnutella in 2000, Kazaa and eDonkey in 2001, Gnutella-2 and BitTorrent in 2002 and Skype in 2003. Additionally, there are networks being used by many other applications, such as Chord, Pastry and Tapestry in 2001, Viceroy, P-Grid and Kademia in 2002 and Koorde, SkipGraph and SkipNet in 2003.

All of these implement a range of P2P models, which creates some confusion. For example, Napster was a centralized P2P application, eDonkey is some kind of a hybrid, and Skype is an unstructured type of network. And there are others.

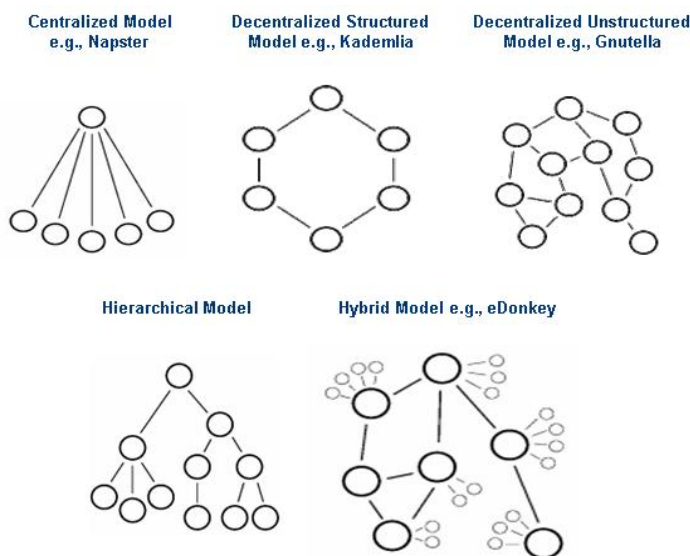


Figure 2: Examples of P2P models

## Typical P2P Models

### Centralized Model

The most typical example of a centralized P2P model was the first incarnation of Napster, which revolved around a central server. In this model, the central server does not contain content, but serves as an index service. Each participant in the original Napster network would log in and post the list of content that they are willing to share e.g., a list of music. Clients then looking for specific music would log in and search the index for another client prepared to share that music. If the required content was found, the requester would be provided with the appropriate IP address, enabling direct communication without the server in the middle. However, the server in the middle was a big issue for legal authorities, and was the main reason that led to the shutting down of the original Napster service. Today's Napster service is completely different – not in architecture, but in the way that payment is made for content.

### Decentralized, Unstructured Model

The best example of this model for P2P is the search flooding method performed by Gnutella, which basically removes the need for a central server. Unlike the Napster model, there is no way for legal authorities to actually shut it down. In this P2P model, participants connected to the Gnutella network search for a specific song or file by requesting the required data from up to seven different nodes (depending on the application) connected to them. If none of these nodes have the data, they each forward the request to additional groups of seven nodes each, snowballing the request forward until a positive source is located. Once the required data has been located, there is no longer any need for the complicated network, and a direct connection is established between the requester computer and the provider computer. This focus on a decentralized search mechanism means that it is never possible to pull the plug.

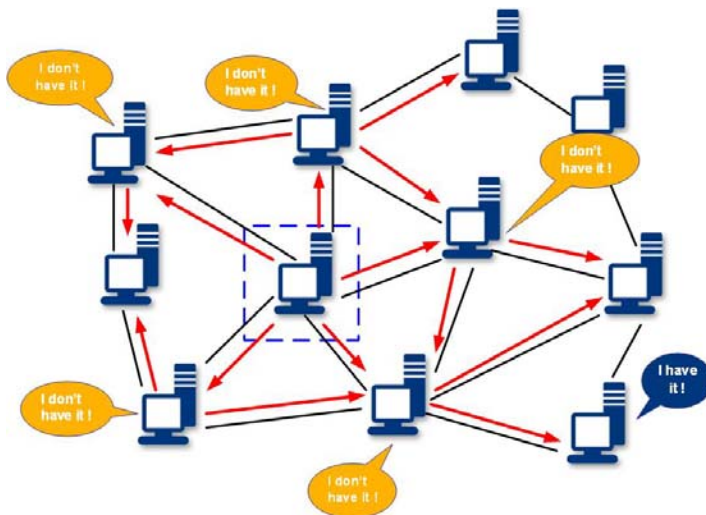


Figure 3: Search flooding in a decentralized, unstructured model – constant forwarding of the request until the data is located.

## BitTorrent Model

BitTorrent, which has become popular in the last two years, is a company, a software application, and a file distribution protocol. On one hand, this makes life easy – there is only one body to address. However, on the other hand, BitTorrent has about 40 “self-made” clients, meaning that there are 40 different forms of the BitTorrent application, some of which are proprietary and even encrypted.

BitTorrent takes the mission of file sharing seriously and is designed to effectively distribute large amounts of data far and wide. This explains its popularity, with some claiming that it reaches up to 35% of total Internet traffic worldwide. This figure may be exaggerated – but the trend and significance is clear.

BitTorrent virtual networks are based on a “seed” or “seeder”, which is a computer that has a complete copy of the requested file. Groups of computers trying to simultaneously download or upload this file are known as a “swarm”, and they communicate different pieces of the total file between themselves. Transactions among the swarm are managed by a “tracker”, which is actually the server, and the BitTorrent protocol is responsible for offloading some of the file tracking work to the tracker. This tracker is directed by a “.torrent” file, which serves as a pointer file directing the tracker to the exact location where the file can be found, making it easy for the tracker to manage the activities of the swarm.

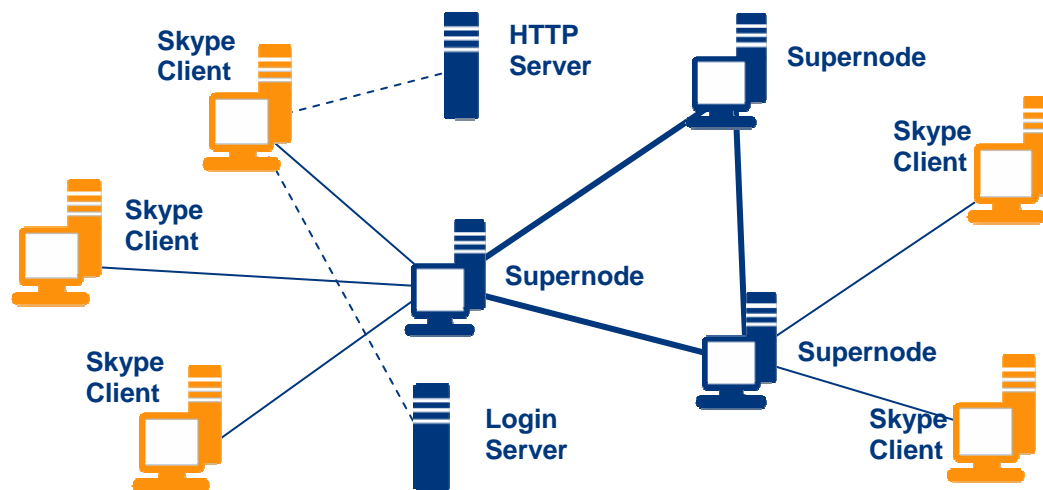
BitTorrent falls into the category of incentive-driven P2P applications - the more bandwidth you give for uploading, the more bandwidth you will receive for downloading from others. This tit-for-tat approach seeks to overcome the problem of “leeches” - BitTorrent terminology for people (computers) that are only prepared to grab information, without sharing it.

BitTorrent enables the downloading of different pieces of the target file simultaneous from multiple computers, and the uploading of the downloaded pieces in parallel. This is achieved by splitting the whole file located on the seeder into multiple pieces, ready for the swarm to start downloading different pieces. Very quickly, this reduces the load from the seeder, eliminates the central server (since the moment that any piece is on the virtual BitTorrent network, it can be shared with all others on the network) and opens many connections on each client.

## Skype Model

Skype is not a regular file sharing P2P application, but a VoIP application using a pure P2P networking method for connectivity. Known as a voice over IP (VoIP) network, it competes against classical server-based VoIP services and networks. However, Skype is not a typical VoIP network, since it has over 100 million subscribers, with 8 million concurrent users online in December 2006. Skype is a proprietary, non-standard, closed source implementation considered by many as a security hazard. This is because it embeds a lot of functions which literally hibernate on networks, waiting to be roused by some central server to perform other activities besides just communicating voice. Finally, Skype is a bandwidth leech, using someone else's infrastructure.

Skype networks consist of a Skype client, a Skype supernode, login servers and HTTP servers.



*Figure 4: Typical Skype network - the Skype supernode is typically a computer running a Skype client with a lot of bandwidth, or at least access to a lot of bandwidth. Skype has about 20,000 supernodes, all of which know about each other, thereby providing complete control of the network. The HTTP and Login servers are centralized entities in the Skype network, ensuring the update of client applications and the authentication and uniqueness of login names, respectively.*

The Skype login process is very simple. The client contacts the HTTP server to check for updates and the supernode list is refreshed by contacting the default hard-coded supernode. Connection is then made with another supernode for exchange of information about online nodes, after which the username and password are verified with the Login server. Finally, another supernode tests whether the client can act as a supernode, identifying various data such as whether the client is behind a firewall or has access to a lot of bandwidth, in which case the client unassumingly becomes a supernode. Up to Skype version 3.0, the list of supernodes was a clear textual list very easily open to hacking. However, since then, the list is encrypted.

### Encrypted P2P

Protocol encryption (PE), message stream encryption (MSE), and protocol header encryption (PHE) are typical examples of encryption and obfuscation methods employed by some P2P file sharing clients, such as BitTorrent. They attempt to make traffic harder to identify, and therefore harder to throttle, by third parties, including ISPs.

Most of these encryption methods are comparatively new (from 2005 onwards), and are constantly being forced to face a range of sophisticated countermeasures being employed by many ISPs, such as pattern/timing analysis or categorizing ports based on side-channel data to detect P2P traffic, which enables the throttling of even encrypted applications.

## The Downside to P2P

For enterprises and service providers, the Internet has become an essential way to do business. P2P is a big headache to them, because it is insatiable, congesting previously smooth traffic, slowing down downloads and uploads, increasing traffic volume asymmetrically, and raising operating costs by 30% and more. The trend seems to point to commercial P2P file sharing schemes, which will only increase the current high rate of downloads and uploads over networks. And increased use on already congested traffic will only exacerbate the situation.

The main P2P-based issues affecting enterprises and ISPs alike are:

**Congested Internet Links:** The P2P bandwidth bottleneck occurs at the ISPs main Internet pipe, clogging the main ISP links for all customers – enterprises and subscribers alike – and degrading the service experience for nearly 80% of non-P2P users. Everyone suffers, with individual computers and LAN and WAN environments facing slower response times.

**Volume of Traffic:** Before P2P, traffic patterns did not change dramatically - when users stopped using their computer, the traffic also ceased. With P2P, links are never idle. P2P users often queue several large files for downloading and then leave the computer or start another task. The P2P application runs in the background, downloading the files and using as much bandwidth as the network can provide – without any limitations.

**Always On Line:** Broadband technologies and their flat-rate charging model have provided users with continuous connectivity, although it was assumed that users would be connecting for only a few hours a day. P2P applications, however, changed this model. They are the perfect way to squeeze all the juice out of broadband, using this “always on bandwidth” to download and upload all day – and all night. Broadband connections can carry heavy traffic at high-data rates at a fraction of the cost, therefore reducing ISP profitability.

**Upstream vs. Downstream Ratio:** Networks were designed to be asymmetric bandwidth, supporting more bandwidth for downloads. This was not problematic, since the upload traffic was significantly lower than download traffic, comprising client/server protocols like HTTP requests – which usually contained small text requests initiated by clients – and TCP Acknowledgements (ACKs). P2P, however, uses both download and upload directions, reducing overall performance drastically. This phenomenon is particularly noticeable on broadband ISP networks, which unintentionally provide fast upload bandwidth to “freeloaders” - external, non-subscribers uploading files free from the ISP paying subscribers.

**Network Security Risks:** Some P2P applications can breach network security and corporate policy. No matter how administrators secure data, in the end, anybody can walk up to a desktop computer and observe P2P processes, disrupt them, or hack into the computer and compromise it. Users can traverse any firewall or proxy and pass sensitive information from their company’s network using encrypted, hard-to-identify P2P applications such as Winny in Japan. In 2006, Winny caused a serious security leak of military and customer-sensitive information in Japan, creating an incident of such

magnitude that the Prime Minister of Japan was called to a special meeting in parliament on the subject. Furthermore, other decentralized P2P applications carry extra “baggage,” such as viruses, worms, adware, spyware and other “malware” installed by third-party P2P applications. Since there is no way to ensure that downloaded files are worm or virus-free, any business network is exposed to the leakage of confidential information. P2P networks may also be more vulnerable to hackers who take over one peer and use it to launch further attacks.

## How to Detect P2P in the Network

The key to identifying P2P is network visibility, which enables determination which P2P applications are running on the network, which P2P applications are eating up expensive resources and which users are using excessive bandwidth and congesting traffic. Availability of this information enables taking steps to utilize existing bandwidth to maximize traffic efficiency, limit or block P2P, and divert newly-available bandwidth for other applications and subscribers.

### **Deep Packet Inspection (DPI): The Only Way to Reliably Classify P2P**

Since most P2P file-sharing applications use random port numbers or abuse well-known ports to gain access, identifying such traffic by ports is insufficient. All packets must be inspected at the application layer: the payload section of the transport protocol (e.g., TCP) must be searched for specific patterns indicating the application type. Furthermore, more than one pattern is often required for matching a unique application signature. Only sophisticated DPI can properly identify applications via the TCP payload.

DPI is the foremost technology for identifying and authenticating protocols and applications (IP flows or sessions in general) conveyed by IP, examining Layers 4-7. This is particularly important when positioning DPI and DPI devices among other categories of devices in the industry. Switches and routers are essentially located at Layer 2 and 3, typically looking at the source, destination address and port of packets, plus other easily-accessible information such as the V-LAN or Type of Service fields. Such equipment provides a response on where packets should be sent.

Conversely, DPI devices located at Layer 4 and even higher at Layer 7 first address the question of what the packet really is. Given the complexity of the latest P2P applications, and all the mechanisms they use to obfuscate themselves, the real role of DPI is to determine whether the packet is what it seems to be, and if not, what it is in reality.

## DPI Methods of Signature Analysis

DPI uses several possible methods of analysis used to identify and classify traffic. These range from analysis by port, by string match, by numerical properties, by behavior and heuristics.

**Analysis by Port** is probably the easiest and most well known form of signature analysis. The reasoning is the simple fact that many applications use either default ports or some chosen ports in a specific manner. A good example is POP3 used for an email application. The incoming POP3 typically uses port 110, and if it is secure, it will use port 995. The outgoing SMTP is port 25. However, since it is very easy to detect application activity by port, this is in fact a weakness, particularly because many current applications disguise themselves as other applications. The most notorious example is the Port 80 syndrome, where many applications camouflage as pure HTTP traffic. Since most P2P file-sharing applications use random port numbers or abuse well-known ports to gain access, identifying such traffic by ports is insufficient.

**Analysis by String Match** involves the search for a sequence of textual characters or numeric values within the contents of the packet. Furthermore, string matches may consist of several strings distributed within a packet or several packets. For example, many applications still declare their names within the protocol itself, as in Kazaa, where the string “Kazaa” can be found in the User-Agent field with a typical HTTP GET request. From this example, it is possible to understand the importance of DPI for correct classification. If analysis is performed by port analysis alone, then port 80 may indicate HTTP traffic and the GET request will further corroborate this observation. However, since the User-Agent field information is missing, this analysis will result in inaccurate classification i.e., HTTP and not Kazaa.

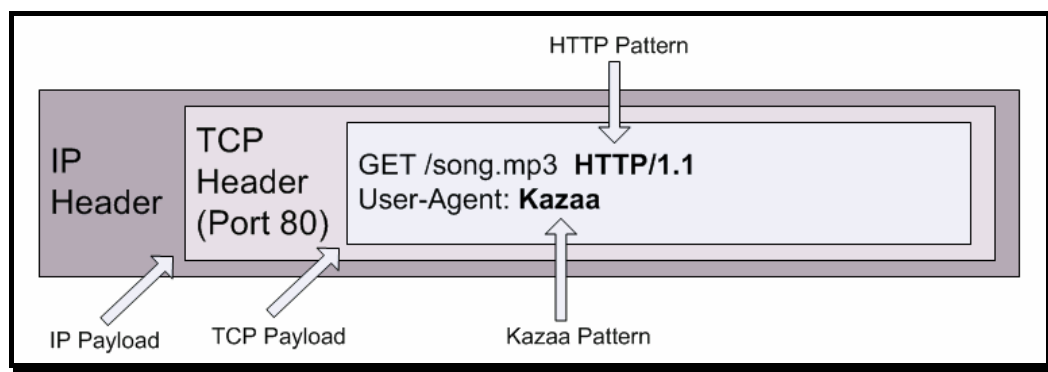


Figure 5: Kazaa string match analysis

**Analysis by Numerical Properties** involves the investigation of arithmetic and numerical characteristics within a packet, and of a packet or several packets. Some examples of properties analyzed include payload length, the number of packets sent in response to a specific transaction, and the numerical offset of some fixed string (or byte) value within a packet.

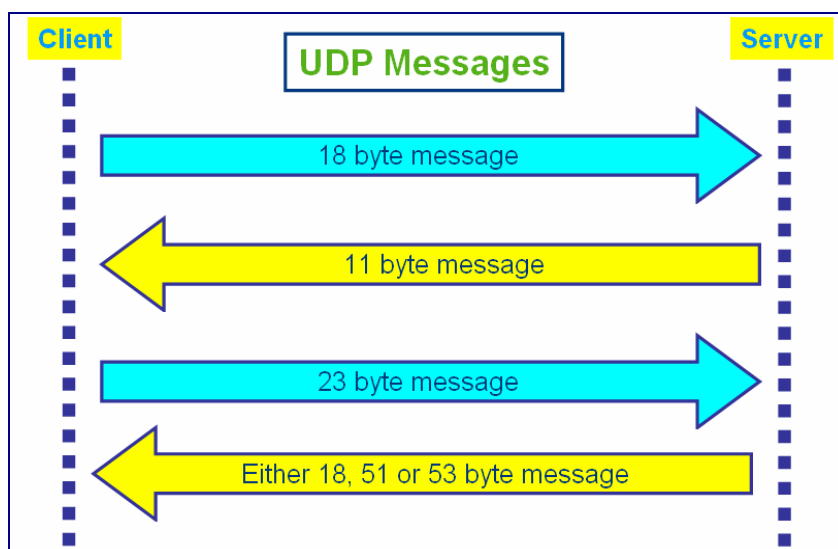


Figure 6: Skype (versions prior to 2.0) numerical properties analysis

For example, consider the process for establishing a TCP connection using some UDP transactions in Skype (versions prior to 2.0). The Client sends an 18 byte message, expecting in return an 11 byte response. This is followed by the sending of a 23 byte message, expecting a response which is 18, 51 or 53 bytes.

**Behavioral Analysis** refers to the way a protocol acts and operates. **Heuristic Analysis** typically boils down to the extraction of statistical parameters of examined packet transactions. Often, behavioral and heuristic analyses are combined to provide improved assessment capabilities.

For example, actions leading to other actions can clearly indicate a behavioral pattern which can be traced, as in the case where an active UDP connection eventually transforms into a TCP connection (using the same IP and port settings).

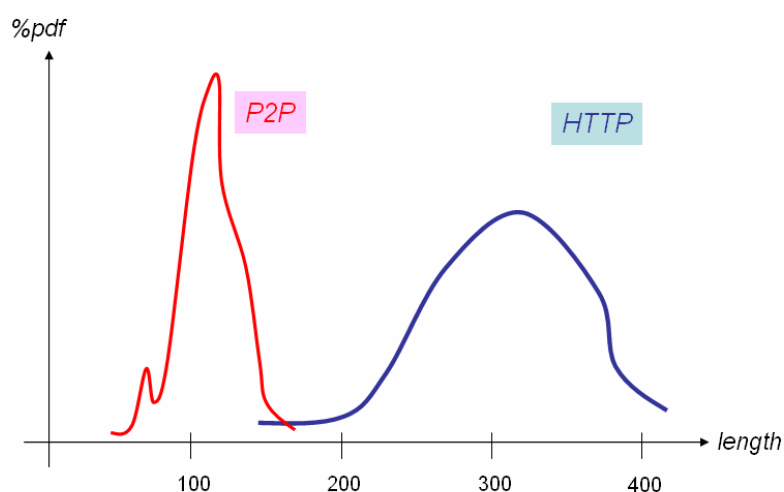


Figure 7: HTTP vs. P2P

Another example of behavior and heuristic analysis is shown in Figure 7, which compares HTTP and a typical P2P file sharing application. If the packet length histogram (PDF) alone is examined while ignoring the file download or upload transaction itself (which tends to use large packet lengths), it becomes apparent that while pure HTTP packets tend to concentrate around a few hundred bytes in length, P2P control layer information tends to use shorter packet lengths. In this way, by examining some short-term statistics, it is possible to conclude whether a port 80 connection carries pure HTTP traffic or other P2P-related traffic.

## Handling P2P

Enterprises and ISPs have attempted various solutions to solve the P2P problem, including adding bandwidth, banning of P2P, limiting (but not banning) P2P, restricting uploads only, penalizing P2P abusers, and applying DOCSIS to control P2P (for cable operators).

**Adding Bandwidth:** The addition of bandwidth is the most obvious and simple solution to P2P, since it temporarily and very briefly eases congestion. However, this will only last for the time it takes the P2P application to “recognize” that more bandwidth is available. Furthermore, increasing bandwidth is a never-ending expense, because it only provides P2P applications with more bandwidth to grab. Enterprise networks face the same dilemma if they try to reduce congestion by upgrading their Internet links.

**Banning P2P:** Naturally, the total banning of P2P use reduces congestion to normal levels. Additionally, in enterprises it will increase productivity among employees currently taking up time and bandwidth for recreational surfing. However, ISPs banning the use of P2P will lose subscribers, particularly since many of their users subscribed just to be able to share unlimited numbers of files.

**Limiting (Not Banning) P2P:** Using an application control and subscriber management solution that provides Layer 7 identification, enterprises and ISPs can accurately detect P2P and then limit it. Control can be accomplished in many ways, including throttling P2P or assigning low priority to P2P when links are congested. Furthermore, ensuring fairness among subscribers or users with the same service level will provide a better quality of experience (QoE).

**Restricting Uploads Only:** For ISPs, restriction of uploads only is critical, since they are particularly concerned about non-subscribers who upload from subscribers, yet pay nothing for these file transfers. ISPs obviously have no interest in upgrading the upstream link for supporting these non-subscribers, and that is why they could choose to restrict upload traffic.

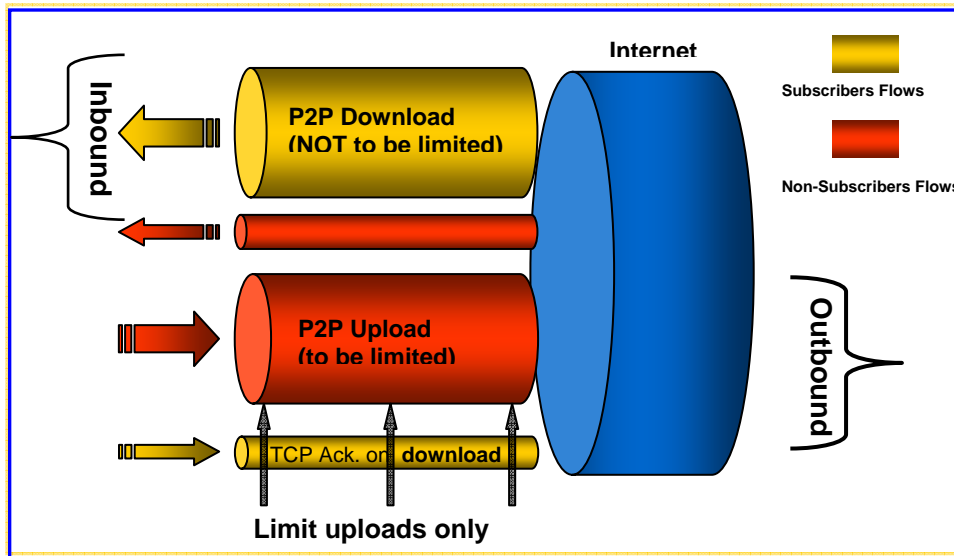


Figure 8: Dedicated traffic management devices can prevent non-subscribers or non-authorized users from taking up precious upload bandwidth

Such restriction of uploads does not affect download traffic. Using a traffic management product that is “smart enough” to limit uploads without affecting downloads, everyone gains. Subscribers can download at will, and ISPs can significantly save on their own backbone provider costs.

**Penalizing P2P Abusers:** ISPs could charge different rates for subscribers who are heavy P2P users. This places the decision to use alternative bandwidth levels in the hands of customers, who can choose how much P2P downloading and uploading is worth to them. However, this can also lead to an en masse desertion of subscribers to an ISP providing unlimited P2P access, causing loss of a substantial installed base.

**Applying DOCSIS to Control P2P (for Cable Operators):** Data-over-Cable Service Interface Specifications (DOCSIS), which define technical specifications for equipment at both subscriber locations and cable operator headends, enable high speed Internet access and include bandwidth control capabilities. However, the European DOCSIS uses a fixed packet length, which is not helpful to track the typically short packet traffic patterns of P2P. Even though some cable equipment can set a “primitive” maximum limit for all subscriber traffic, the maximum limit is per total traffic of each modem. Consequently, the subscriber’s business-traffic will also be speed-limited. Additionally, DOCSIS remains ineffective against elusive port-hopping P2P, because it cannot classify Layer 7 traffic.

## Living with P2P

P2P is not going to disappear because of disgruntled musicians, content providers and filmmakers, or network administrators, ISPs and carriers frustrated by bandwidth-hungry applications. If anything, more and more P2P applications will be appearing, together with stepped-up use of commercialized P2P file sharing.

All this activity means more bandwidth use and having to learn to live with P2P. The first step toward managing and controlling P2P applications is the implementation of a DPI-based management solution, which can:

- Auto-detect network status at a glance
- Spot peaks, bursts and bottlenecks immediately
- Correctly identify P2P applications running on the network using Layer 7 visibility
- Support recognition of new P2P applications as they are developed
- Discover P2P users and abusers
- Enforce policies to limit P2P and divert idle bandwidth to other, more essential applications
- Monitor policies and adjust quickly to meet changing network conditions
- Help save money by utilizing existing bandwidth to maximize traffic efficiency
- Create reports for use by 3rd-party billing and operation support systems



---

<b>Americas</b>	7664 Golden Triangle Drive, Eden Prairie, MN 55344 USA Tel: (952) 944-3100; Toll Free: (877) 255-6826 Fax: (952) 944-3555
<b>EMEA</b>	22 Hanagar Street, Industrial Zone B, Hod Hasharon, 45240 Israel Tel: 972 (9) 761-9200 Fax: 972 (9) 744-3626
<b>Europe</b>	NCI-Les Centres d'Affaires Village d'Entreprises, 'Green Side' 400 Avenue Roumanille, BP309 06906 Sophia Antipolis, Cedex France Tel: 33 (0) 4-93-001167, Fax: 33 (0) 4-93-001165
<b>Asia Pacific</b>	6 Ubi Road 1, Wintech Centre 6-12, Singapore 408726 Tel: 65 6841-3020 Fax: 65 6747-9137
<b>Japan</b>	Puri-zaido Ochanomizu 301, Kanda Surugadai 4-2-3, Chiyoda-ku, Tokyo 101-0062 Tel: 81 (3) 5297 7668 Fax: 81 (3) 5297 7669; www.allot.jp

---

**w w w . a l l o t . c o m      i n f o @ a l l o t . c o m**

---

© Allot Communications, 2007. All rights reserved. Allot Communications and the Allot logo are registered trademarks of Allot Communications. All other brand or product names are trademarks of their respective holders.

---