Controlling access to BitTorrent content from mobile devices

Péter Ekler, Zsolt Pszota
Department of Automation and Applied Informatics
Budapest University of Technology and Economics
tyrial@sch.bme.hu, neel@sch.bme.hu

Abstract. BitTorrent technology is one of the most popular peer-to-peer solution. The technology ensures fast download speed without large servers. The download procedure needs special torrent files, which contains information about the requested content and the tracker address. The main tasks of the trackers are to control the traffic and help the participant to find each other. The main goal of the protocol is not to share illegal content, it is because how it works. In order to stop the illegal traffic, we have to stop only the tracker, which is responsible for the copyrighted files. The BitTorrent technology based on a standard, but it has several extensions. One important question about the protocol is the defense of the shared content. How can we share something through BitTorrent if we do not want to make it available for everybody, only for a set of people? This function would be very useful, if we want to share private content to our friends, or if a company wants to share large files between the employees. Peer-to-peer file sharing technology has a play to role in the mobile world too. We took part of developing the SymTorrent application, which is a complete BitTorrent solution for mobile devices. Defending the shared content is also very important on mobile devices. We worked on a solution, how we can ensure that the shared content can be downloaded only by those, who we want to share it. We introduce some ideas and solutions about the topic, and a concrete method, with which the download can begin only after an authentication. We focus on SymTorrent application and solution for mobile devices, but we use the existing experiences in the PC world.

Keywords: BitTorrent; Mobile devices; Security; Shared content; Protection

1 Introduction

P2P (peer-to-peer) networks are becoming more and more popular. They are useful for many purposes, for example sharing files, like audio, video or anything in digital format, besides that real-time data, such as telephony, and TV traffic is also passed using P2P technology. [1] Generally people think that P2P networks are for transferring illegal content and copyrighted software. There are several web sites where people can download illegal content, however they are not so convicted than the existing P2P solutions, hence we think that the negative opinion about P2P technology is a popular error. P2P technology has several advantages, thus it has a definite role to play in the future of the Internet.

[2] A pure peer-to-peer network has equal nodes, every peer can be "client" or "server", it depends on the current function and role of the node. In this case the control of the traffic is quite difficult, for example it is hard to find the origin of an illegal content, and if we want to stop the sharing of this data, we have to find every peer which have already downloaded it.

There are hybrid peer-to-peer implementations, where there are some accentuated nodes. These nodes can be used to control and observe the traffic, thus we have more possibilities to embarrass the illegal file sharing.

BitTorrent is a hybrid P2P solution, which allows to download large content quickly. If we make a large file available on a web site and people start to download this file, then after several requests the server will be overloaded because of the huge traffic. But in BitTorrent it works on a different way, if people start to download for example a large file, then after a while the others can download the pieces (small section of the file) from each other, so the traffic and requests will be balanced.

[3] For better understanding, following we will introduce shortly the basics of the BitTorrent technology. In BitTorrent there are some special peers, called trackers, which are responsible for providing information to the peers, about where they can download the requested data from. The first step of the download is to get the torrent file, which contains information about the shared data, which we want to download. Figure 1 illustrates the content of a torrent file.

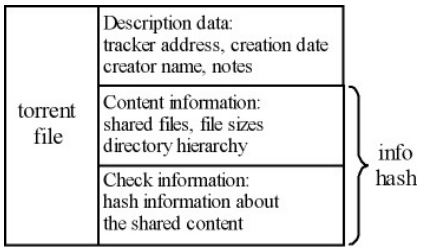

Figure 1: Content of a torrent file

The "info hash" of the torrent file represents the content of the file, whereby during the protocol a 20 byte hash will be generated. This hash value is used for identification during the protocol. After we managed to get the torrent file, we load it into a client application, which reads the content of the file and generates the info hash. Then the client application connects to the tracker, which address was defined in the torrent file, and sends - among other things - the hash value to it. If the tracker recognizes this torrent file via the info hash, then it sends back a list of peer-addresses, from where the client can start to download the pieces of the file(s). While we are downloading pieces, other peers can also ask pieces from us, which means that during the download, we also upload pieces to other peers. This is one of the tricky part of the protocol, which makes the download faster for everybody.

If we want to share files via the BitTorrent technology, then it is a little bit more complicated than the downloading part was. First we have to create a

torrent file of what we want to share, after it we have to run a tracker on our computer, or we have to upload the torrent file to a tracker and ask it to share this torrent. Besides that we also have to seed this torrent, which means, that we have to ensure, that other peers can start download it from us. If we do not seed the torrent file any more, and no one has managed to download the whole content, than other peers will not be able to download the data, even if they have the torrent file, because there are no available peers, which has the required content. Figure 2 demonstrates the BitTorrent technology.

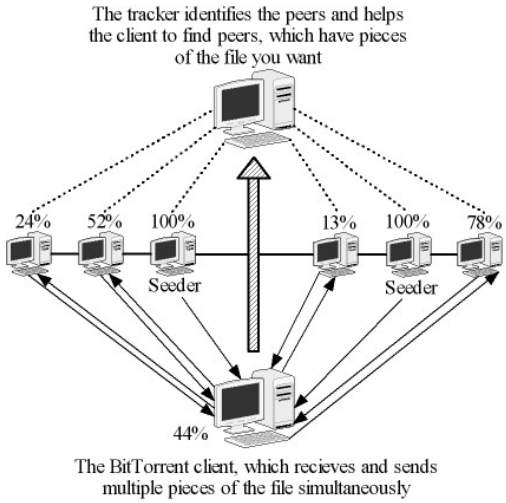

Figure 2: BitTorrent technology

This is the basics of the BitTorrent technology. There are some extensions of the technology but they are not part of the standard. The technology was not designed for illegal file sharing, it is because how it works. Illegal file sharing can be avoid, we have to find only the tracker(s), which is responsible for the suspected torrent files. The other thing in the protocol is the defense of the shared content. There are some use-cases when we want to share large files via BitTorrent technology, but we do not want to share it for everybody but only a few people. There are no standards for this case. If somebody gets the torrent file, then the download procedure can be started without any authorization. There are existing solutions on the PC, which we will introduce shortly in this paper.

There is an interesting implementation of the BitTorrent technology for mobile devices, called SymTorrent. This is a complete BitTorrent client and tracker for Symbian OS. We took part in developing of the application, but currently the system does not offer any solution for defending the shared content. In this paper first we want to show why defending the shared content is important, after it we introduce our solution with which the shared content can be downloaded only by those, who we want share the files. We focus on mobile devices and on the SymTorrent application.

2 Motivation and Problem statement

The SymTorrent application contains a tracker part, thus it is not only able to download contents, but upload and share files from the mobile devices too. SymTorrent is a good solution for people to share contents fast and easily on mobile devices. The new mobile devices have quite large memory, and some type has a built-in hard drive too. Besides that people store more and more data and files on their mobile phones, hence file sharing solutions have a definite role to play in the mobile world.

Defending the shared content on mobile devices is important in several use-cases:

- Private content for family members
- Business content for company workers and partners
- Valuable files, which we want to share only between people, who paid for the service

In the following we show the background of the topic of content defense, after it we introduce our solution for defending shared content in mobile environment.

3 Backgrounds and Related Work

[1] The BitTorrent technology has several advantages comparing to other P2P file sharing solutions. The main goal of the technology is not to share copyrighted content, thus it can not be convicted. There are several properties of the technology which proves this statement. The first one is - we have mentioned it before - that there are trackers in the network. These trackers are for controlling the traffic, if there are any shared illegal content on the network, then we have to shut down only the tracker to stop the traffic. The tracker is easy to be localized because its address is included in the torrent file. If there is no available tracker for the torrent, then the client can not get the address of other peers and they can not start downloading the pieces. ([4] There is an extension of the BitTorrent protocol, called DHT - Distributed Hash Table, which makes possible for the peers to send other peer addresses to each other, hence there is no need for the tracker, but the DHT is not the part of the standard BitTorrent protocol.)

The second thing is that searching for content is not the part of the BitTorrent protocol, which is another difference comparing to other P2P solutions.

[5] Another advantage of the protocol is that the risk of viruses and other infections is very low. It is because of how the protocol works. In BitTorrent there is no need for shared folders like in other solutions. Every downloaded piece has a hash value in the torrent file, so after the download, the piece can be checked if the content was modified or not.

Because of the mentioned advantages, we found the BitTorrent technology the right choice for mobile environment, it is a well designed, fast and reliable

protocol. We successfully managed to implement the earlier mentioned SymTorrent application with tracker part, but the important question of defending the shared content remained. In the PC world there are some existing solutions. The most prevalent solution is that you can not download directly the torrent file from a website, first you have to log in on the page, and only after the authentication the torrent will be downloadable. In this case the torrent is available only for the registered users, but if you downloaded the torrent, and somehow other people get this file, they can download the content too. A better solution would be, that the download procedure can be started directly after an authentication, so when you add the torrent to a client application and start the download you have to enter a password, or an ID first.

Following we introduce deeply the security questions of BitTorrent and show our solution about the mentioned shared content defending problem in the point of SymTorrent.

4 Contributions

There are several different aspects of the security of a file sharing solutions. During this chapter we examine them and we show our solution for protection of the shared content. In the BitTorrent protocol we have to talk about two different types of communication. One is between the tracker and the client (tracker communication), and another is between the clients (peer communication). The tracker communication does not contain any shared content, only parts of the communication controlling, but it is necessary for the download process. The peer communication contains shared content data. It works on a distributed way, it is not trivial to collect all the pieces and put them together, hence stole a complete file from the traffic is quite difficult.

Protection of the communication
It is possible to protect communications over Internet against monitoring attacks. This is a low level weakness. Currently the BitTorrent solution does not allow real protection. For the tracker communication we are allowed to use HTTPS protocol which gives us protection, but currently the peer communication supports only simple plain text TCP protocol. Using some kind of security system for the peer communication part could solve the protection problem, but in this case it should be implemented in all BitTorrent clients.

For the tracker part the HTTPS is a good choice if the protection is necessary, but we do not have any opportunities to change the peer protocol using the current standard.

Protection of the tracker connections
It is possible to build in some protection into the tracker. Then it allows only a special set of the users to connect. Others get an error message, so they are not able to start the downloading process. It is a good centralized solution using the standard protocol, only one thing needs to be solved, an extended authenti-

cation has to be built in the tracker. There are no standard solutions for that, but as it is not the part of the BitTorrent system, many solutions can be used in this case. For example a built-in user authentication attached to the tracker is a good solution for this problem. Then the users should log in the server and then they get access for the tracker for a specified time interval. This should be implemented separately from the client, because we should not expect this extension from all clients.

Protection of the peer connections
It may be possible to protect the connections between the peers. This way the clients should accept connections and requests from only a set of safe peers, so the unauthorized clients will not be able to join to the protected file sharing system. The problem is that the clients need to administrate the allowed and unwanted peers. For that the protocol needs a large extension, which we can not expect from all existing client applications.

which we can not expect from all existing client applications. In this case the undesirable clients would be able to connect the tracker or gain the peer addresses in a different way, but they would not be able to get any piece of the content, because the other peers would refuse their requests.

Protection of the torrent file and the tracker connections
An advanced solution is when we protect both the torrent files and the tracker communication together. In this case we need an authentication before getting a torrent file. So during this process, the client address can be stored. If we can share this information with the tracker, it will be able to check if the connecting client uses the same IP address that was used to get the torrent file. The tracker should accept only those connections which have matching addresses to the downloaders. The problem of this solution is that the client should use the same IP address during the whole downloading process, but in a few cases it is not possible. For example if the user has an Internet provider, that share out dynamic IP addresses for the subscribers, then this user may have unexpected refuses during the downloading process. In the case of mobile devices is the same, they do not have own public IP addresses, thus this solution is not the best in mobile environment. Whereas this solution does not need any changes in the protocol, we only need some modifications at the tracker side.

Protection of the torrent file
At first the protection of only the torrent file sounds slight to be a real protection, but in several cases it is just enough. By protecting only the torrent file, we can make it harder for the unwanted clients to get the requested content. In this case we do not have anything to do with the protocol, the only thing needs to be solved to create an effective protection for the torrent file. As it is not the part of the BitTorrent standard, we can use any way to do it. The easiest and probably the most popular solution is to create a web portal with a built-in authentication system, so the users can download the torrent files only after they entered the correct username and password.

This solution works this way, but we should consider of some extensions. After the authentication the users lose there anonymity, but as they download the same torrent file, they are anonymous again during the downloading process. In some cases we might need to identify the different users.

What is needed to be protected?
The protection against the external attackers, who try to get the content by monitoring methods, is quite difficult. We need to change the standard, because our current opportunities are limited.

But there is another issue which does not include any low level method. By protecting the BitTorrent system in any way, we can not forbid to the users to redistribute the downloaded content. We can only make it harder to get the requested content. If one of the subscribers wants to forward the downloaded files to someone else we can not stop that. The only way to defend the shared files, if we build the protection into the content, but this is not the part of the BitTorrent system.

The difference between protecting the torrent and the connections is not so much. In both ways it is possible to send the content to an unwanted third party. But it is much more difficult and less resource efficient to protect the connections, so the torrent-protection seems to be the better choice.

Extensions for the torrent file protection
There is an existing solution to extend the torrent file protection. The problem, that we have no chance to identify the different users during the downloading process. If we had an opportunity to add a little difference to the tracker communications we could isolate the user activities.

The problem is that the torrent files are identified by their info hash, which includes the most of the torrent file, but not the whole. As we discussed earlier, the content of the torrent file can be separated into three different parts and the info hash contains the content and the check information, but not the description data. Thus if we want to make some changes in the torrent file, it has to be in this description area.

The next thing we need to think about, that after this torrent is downloaded from the server and this extra information is added, we expect the tracker to identify this client during the tracker communication. Unfortunately according to the standard, the client does not send back much information from the torrent file to the tracker. Only thing in the request that is based on the torrent file content is the info hash, but it should be not modified. If we look at it more carefully, we can find that there is one more thing used from the torrent file, the address of the tracker called announce URL. If we encode the identification into this network address, then it does not change the info hash, but it is sent back to the tracker so the user will be identified.

To do not change the info hash is quite important, because this binary value is used for identify the torrent file in many cases in the BitTorrent protocol.

There is an option to add some extra information to the torrent file as a comment. We might be able to use it for our identification process, but the

problem is, that in this case the client is expected to send back some comment related information to the tracker and this expect changes for all existing clients.

There is one more question in this identification case. We have to specify where and how this id should be generated. We suggest that the id should be created at the server side, so the users can not change their identity every time they want. This id should be any random data, for example some kind of hash of the username, but the same value should be used every time for the same user.

The way of the distribution of the torrent file is not the part of the protocol. The simplest way is to create a web server attached to the tracker, and include the authentication and the id generation into the web page.

Our extensions for this solution
The problem with this built-in id version is that if someone unexpected gains this torrent file, he or she will be able to act like the legal user. So we should add another protection into the system. We can previously share passwords for each user and then we should request this password when the user starts the downloading process. In our SymTorrent client application for mobile devices it is possible to build in an additional authentication. The software recognizes that it is a special torrent file which need password, then the program asks the password from the user and attaches it automatically to the end of the announce URL. This way the torrent protocol is not corrupted. The communication goes in the common way, but an additional password is encoded in the tracker address, so the user got more protection. We have to solve the problem, if the user wants to use another BitTorrent client, which does not have any built-in authentication support. So we can create a tool application which opens a torrent file, and helps us to add the password to the end of the tracker URL. After using this tool we get a standard torrent file which includes the authentication password, so it will be accepted by the tracker.

5 Case Study

We analyzed the problem of BitTorrent content security and we found several existing solutions that work on PC. We studied all of them and our result is that the protection of the torrent file is enough in the most of the cases. So we suggest protecting the downloading of the torrent file using authentication in the web server. If we add an identification value into the announce URL, it does not change the info hash, and it is automatically sent to the tracker encoded in the address. So the protocol does not need any changes. This solution is already used in some tracker and it works fine with currently available PC clients and our mobile client too.

As an extension we added a password into the announce URL. This is a previously shared value which gives an additional protection for the user, thus the shared content can be downloaded only by those, who know the right password.

6 Conclusions and Future Work

BitTorrent technology is one of the most reliable peer-to-peer file sharing solution. P2P technology should not convict, because of the illegal traffic, because there are at least as much illegal content on the world wide web too. In several cases the BitTorrent technology could be very useful, for example to share the latest linux distributions and so on. The technology has several extensions but they are not standard yet. An other interesting question about the technology is the defense of the shared content. It could be useful, for example if we want to share something, but we want to make it available only for our friends.

We have studied the P2P solutions for mobile devices earlier. We took part of developing the SymTorrent application, which is a complete BitTorrent based P2P solution for mobile devices. The topic of protection of the shared content is also important on mobile devices. In this paper we have talked about the existing solutions in the PC world, we showed the advantages and disadvantages. After it, we have introduced our solutions for mobile devices, with which we were able to reach, that the shared content can be downloaded only by those, who we want to share the content. Our solution is based on the method, that directly before the download, we have to enter the right password, to begin the procedure. This password is known for the tracker, thus it can answer the request only if the client sends the right one at announce (when the client connects to the tracker).

This way we managed to defend the shared content. The technology has several existing extension. One of the most popular is the DHT, which is not a standard yet. In DHT peers can get other peer addresses from each other and they do not depend from the tracker. In this case we have to find other content defense solution, which could be an interesting research area.

We should also mention that our solution works only if we do some modification on the SymTorrent client side, where users can enter their password before download. But if we want to use the password-protected torrent with a PC client, than it will not work, because the existing clients do not have the function to send a user defined password to the tracker at announce time. One of our future plan is to find a way to cooperate with the existing PC clients.

Summary it can be said that with our solution we are able to protect the shared content, and we managed to reach that goal, without modifying the BitTorrent protocol, it was enough to modify the existing SymTorrent application only.

Acknowledgments

The authors would like to express their thanks to Tihamér Levendovszky and Hassan Charaf for their support as a scientific advisor. This work has been supported by the Department of Automation and Applied Informatics, BUTE.

References

[1] J. Stewart, "Bittorrent and the legitimate use of p2p." http://www.joestewart.org/p2p.html, February 2004.

[2] "Description of the peer-to-peer technology." http://en.wikipedia.org/wiki/Peer-to-peer, May 2007.

[3] "Bittorrent protocol specification v1.0." http://wiki.theory.org/BitTorrentSpecification, May 2007.

[4] "Description of the bittorrent technology." http://en.wikipedia.org/wiki/BitTorrent, May 2007.

[5] "About bittorrent and related security concerns." http://www.broadbandinfo.com/got-high-speed/peer-to-peer/bit-torrent-security.html, May 2007.