

OneSwarm

Chryptology Assignment
2009-05-18
Björn Lindelöf
Erik Bohlin
Kristoffer Hellstrand

Table of Contents

Introduction	3
Background	3
BitTorrent traffic	3
OneSwarm	4
Encryption	4
Privacy	5
In practice	5
Conclusion	6
References	7

Introduction

We chose to present this subject because it's a topical issue that you can read about in today's newspaper, because this is one of the first times we have come in contact with substantial encryption in real life and that we find this area very interesting. We will try to explain the technique of this new application OneSwarm and how it could protect ourselves from having our Internet habits monitored by any third party.

Background

To explain what OneSwarm really has to offer we need to explain the flaws in today's Internet traffic and present what OneSwarm could do to solve these problems. We will do so by introducing today's most used Internet application and see how it handles privacy and comparing it to the OneSwarm application. Basically one third of Internet communication consists of peer-to-peer communication (2002) and the most popular application for network transmission is through some BitTorrent clients.

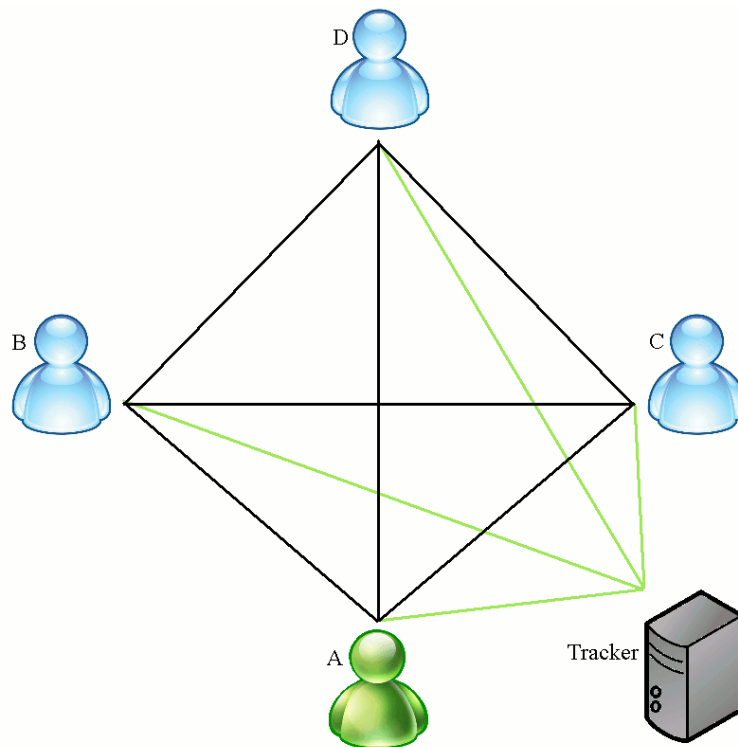
BitTorrent traffic

Network peer-to-peer communication is mainly distributed by servers connecting users to each other to exchange desired files. These servers have been called trackers and store information about where to find users with the desired files. Peers may simultaneously upload files to a single user within a BitTorrent network increasing the file source availability and decreasing the download time, this is one of the reasons BitTorrent traffic has become so popular. These networks are open to anyone and display all the activities and connectivity information about its users and those connected to the network. This means that all communications within these networks are completely visible for third parties.

Trackerless networks also exist, then the tracker assignment is carried out by a DHT (Distributed Hash Table) which exists within every peer. These kinds of networks also display all information about the users and the traffic.

By using these kinds of networks users are exposing themselves to countless monitoring opportunities which may intrude on their personal integrity.

Since it has recently become more common to have Internet connections under surveillance due to file sharing issues, the current encryption is not enough to keep personal communication private since the current encryption is nonexistent.



Model over a BitTorrent network where all peers are connected to each other, as well as the tracker who keeps track of which peers got which files.

OneSwarm

OneSwarm is a new BitTorrent based application for distributing data over the internet. As already known, the BitTorrent protocol have some weaknesses regarding security concerns, and OneSwarm was developed as an effort to fix these problems.

OneSwarm is what is called a F2F (friend-to-friend) protocol. Which means that users only can connect to users with whom they're friends with. This way a third party can't connect to a network in which it has no friends, and thus cannot see the traffic inside the network.

This is however not the only measure taken to ensure better privacy of the transfers. All data is encrypted with the RSA encryption algorithm, so that sniffing a connection won't give away the content of the traffic either.

Encryption

Encryption is used in several steps in the OneSwarm solution. First of all OneSwarm uses a DHT in which all users IP-addresses and port numbers are saved. To ensure that only friends can find a user, the peer inserts its IP and port once for each friend and encrypts this data with the friends public key. This way only friends of the user can decrypt the information and connect to the peer.

OneSwarm is encoded in Java and uses 1024 bit SSLv3 encryption with client certificates which is provided within the SSL-library in Java. Each users cryptographic key is determined upon software installation and then serves as a persistent identification for the user. This key is a 1024 bit public/private RSA key pair. Upon connection, the transmission consists of XML-compressed file lists that are encrypted with each users public key. Within this messages attributes describing the name, size, date shared, and other meta-data that only should be visible to the receiver is encrypted. A user can create multiple cryptographic relationships with multiple friends with each friend having

a individual public key so only that specific friend with that specific key can send or receive the intended file.

Privacy

OneSwarm handles privacy in more ways than just encryption. Encryption makes sure that third parties can't read the transferred data, but it's still possible to see who's connected to who. Also, in BitTorrent, it's possible to see what search queries are made, and by whom. This can be a bit sensitive when it comes to political views and such. OneSwarm handles this in a way such that only friends can see what another user search for and download. When a search is done in OneSwarm, a query is sent to all friends in the network. These then forwards this search to all their friends and so on. When a download then starts, data is sent back the same way as the search query came, and thus peers only talk to their friends, even if the source isn't a friend.

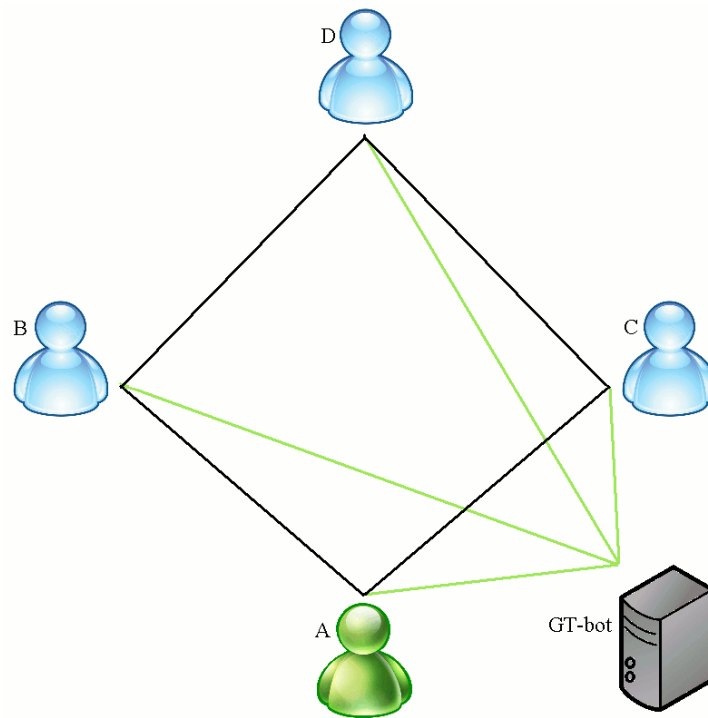
In practice

Each user is given a unique identity in form of a cryptographic key, this is also the public key in the RSA key pair. The users connectivity information, such as IP-address and port number, is associated with its identity. The key is a long-term identity that is generated on installation of the application. Since the connectivity information is short-term, OneSwarm uses a DHT to link long-term identity to short-term so that friends easily will find the correct connection information.

To find and add friends, OneSwarm uses Google Talk. A user submits its request to add a friend to a bot in Google Talk which adds this request to a SHA-1 hash table. In order for two peers to become friends, the other peer has to submit a request of its own and then the bot see that the two peers what to become friends and handles the exchange of public keys, names and connectivity information so they'll be able to communicate.

Even though peers can only see their friends, this doesn't limit downloads to friends only. When a user makes a search, the search is sent to friends. These then forward the search to their friends, whom also forwards the search again, and so on. This means that a user can download data from almost anyone connected to OneSwarm, as long as the other user is a part of the same friends network. To ensure privacy, no information about the search origin is forwarded.

When the transfer of the requested data starts, it's sent backwards the same way as the search query came. Since the data source doesn't know who requested the file from the beginning, it can't send it straight there. This because each node makes its own search and doesn't forward any information of the search origin.



Model over a OneSwarm network where all peers are connected to friends, but A and D can still share data between B and C without knowing each others identities. All peers are connected to a Google Talk bot for key exchange when needed.

Conclusion

It's obvious that it's not the encryption that's the most interesting part of OneSwarm, since it uses the standard 1024 bit SSLv3 encryption and RSA. It's more interesting how they handle privacy and hiding the sender/receiver information from hosts in the network.

By simply encrypting the data sent between two hosts, third parties can still monitor who talks to who, and that might not be desirable. OneSwarm takes care of this problem by addressing the data to a friend, the friend from which the request came, who then forwards the data again towards the request origin. This way no one in the chain of forwarding hosts can know who the requesting host is, since there is no way to know what happens after forwarding. The next host can either forward it again, or keep it, being the origin of the request.

OneSwarm is a good alternative for those who really want to hide their identity exchanging data between friends. However, there are of course negative aspects of this new application. When adding friends you need a third party, Google Talk. Which means that users have to register and set up a Google Talk account, and we're sure not everyone is prepared to do so.

Also, we're not certain how to find enough friends to reach the same speeds as with BitTorrent. Are there communities for OneSwarm users where you can find friends? Or are there lists of users available on the internet? Answering these questions could be an interesting extension for this report.

References

“Friend-to-friend data sharing with OneSwarm”

Tomas Isdal, Michael Piatek, Arvind Krishnamurthy and Thomas Anderson, Washington University, 2009-02

http://oneswarm.cs.washington.edu/f2f_tr.pdf

OneSwarm Homepage

<http://oneswarm.cs.washington.edu>

“OneSwarm friend-to-friend P2P likely to irk Big Content, ISPs”

John Timmer, Ars Technica, 2009-02-25

<http://arstechnica.com/software/news/2009/02/oneswarm-friend-to-friend-p2p-likely-to-irk-big-content-isps.ars>