Secure File Sharing in Darknet

Amita A. Dwivedy
PG School GTU (ITSNS), Department of Computer Engineering, Gujarat Technological University, Ahmedabad

Abstract

Darknet is a collection of networks and technologies used to share digital content. It is not a separate physical network but an application and protocol layer riding on existing networks [1]. A private, distributed Peer-to-Peer file sharing network, where connections are made only between trusted peers using non-standard protocols and ports is darken. It is also called F2F (Friend to Friend) network. File sharing in darknet is anonymous, and therefore users can communicate with least fear of third party interference which enhances the privacy feature in network communication. Along with user privacy it also promotes illegal activity such as distribution of copyrighted materials or distribution of abusive content. Due to the property of anonymity it is difficult to track the source of such file distribution. To resolve the issue of distribution of illegal content along with preserving the privacy of the user in peer to peer network a private networks is proposed in which the data will be provided through server to its peer client or the client can share data or file that are made available by the server in the network. In this way, server will be responsible for each and every file sharing within the network i.e. server is only allowed to inject file in the network and clients cannot inject file in the network. Therefore, despite of anonymous file sharing, the data transferred will be monitored by the server.

1. Introduction

Darknet is defined as a collection of networks or technologies used by private groups of people communicating or sharing digital content protecting from outside party's supervision of the transactions [1]. Darknet is not a separate physical network but it is an application and protocol layer riding on existing networks [1]. (Peter Biddle, 2002) defines darknet in their paper The Darknet and the Future of Content Distribution as an idea based on three assumptions [1]:

1. Any widely distributed object will be available to a fraction of users in a form that permits copying.
2. Users will copy objects if it is possible and interesting to do so.
3. Users are connected by high-bandwidth channels.

To operate effectively, the darknet has a small number of technological and infrastructure requirements, which are similar to those of legal content distribution networks. These infrastructure requirements are [1]:

1. Input, facilities for injecting new objects into the darknet.
2. Transmission, a distribution network that carries copies of objects to users.
3. Output, ubiquitous rendering devices, which allow users to consume objects.
4. Database, a search mechanism to enable users to find objects.
5. Storage, e.g. a caching mechanism that allows the darknet to retain objects for extended periods of time.

Features of Darknet (private P2P content sharing)

Desired features of Darknet are as follows:

1. Authenticity: Unauthorized peers are not allowed to make changes to the content shared.
2. Scalability: There is no limitation of the size of the system
3. Reliability: The malfunction on any given node will not affect the whole system.
4. Performance: It is defined in terms of time duration between the request and the final receipt of desired content, searching, uploading and browsing for content.
5. System stability: The topology of a P2P system changes continuously as peer join and leave despite of such changes the system remains stable and continue to provide services at all times.
6. Privacy: Privacy could be ensured by either anonymity or trust.

- Anonymity: Anonymity ensures that the identity of user is not disclosed.
- Trust: Assured reliance on the character, ability, strength, or truth of someone or something.

A peer to peer (P2P) computer network is type of network in which each computer in this network can act as a client or server. Client means placing a request i.e. client is a running application programs on a local site that requests service from a running application program on a remote site. Server means a program that can provide services to others program. New measures imposed by governments, Internet service providers and other third parties which threaten the state of privacy are also opening new avenues to protecting it. The darknet is a rising contender against these new measures and will preserve the default right to privacy of Internet users.

Although, darknet file sharing system provides privacy to the users, it also promotes illegal activity such as distribution of copyrighted materials or distribution of abusive content. Due to the property anonymity it is difficult or rather impossible to track the source of such file distribution.

To resolve the issue of distribution of illegal content along with preserving the privacy of the user in peer to peer network a private networks is proposed in which the data will be provided through server to its peer client or the client can share data or file that are made available by the server in the network. Once the data is released in the network, the clients can share data amongst themselves. In this way, server will be responsible for each and every file sharing within the network i.e. server is only allowed to inject file in the network and clients cannot inject file in the network. Therefore, despite of anonymous file sharing, the data transferred will be monitored by the server.

2. Analysis of Peer to Peer Network

P2P system is defined as the sharing of computer resources and services by direct exchange between systems [15]. P2P can be categorized into two groups classified by the type of model: pure P2P, and hybrid P2P.

Pure P2P model: does not have a central server.

Hybrid P2P model: employs a central server to obtain meta-information such as the identity of the peer on which the information is stored or to verify security credentials. In a hybrid model, peers always contact a central server before they directly contact other peers.

Client - Server Architecture

Client-server is a traditional way of file sharing over internet. For example website, in which, a client sends request to the web server and the server responds with the requested information or the file. Figure 6 shows the message exchange process in client-server architecture. Client-server architecture is illustrated in figure 7. Figure shows that there is a scalability issue in client-server model, i.e. with the increase in the number of client work load on the server also increases; hence system becomes slow which results in degradation of system performance.

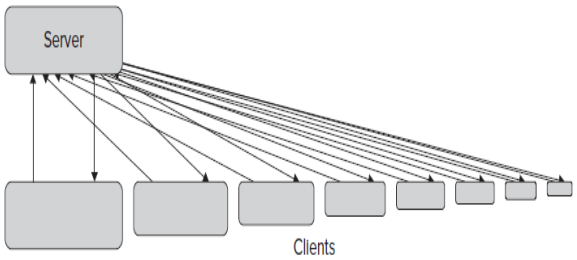

Fig. 1 Client-Server Interaction

P2P Architecture

The alternative to client-server approach is P2P technology. In P2P architecture, file is directly sent to few clients in the network, the remaining other clients can then download the file from those clients. Each time the client download the file, the file is replicated in the network i.e. the number of resources increases within the network. The process of file sharing is made even faster by dividing the file into chunks and distributing these chunks among clients in the network, other clients can download these chunks simultaneously by connecting to different clients parallel.

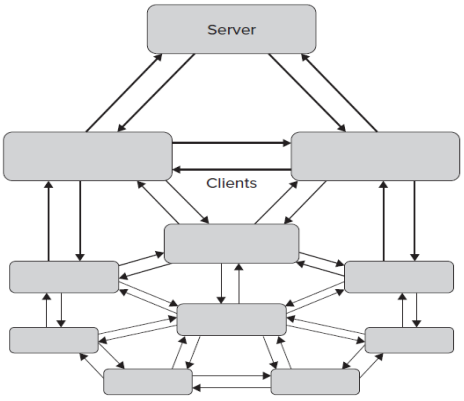

Fig. 2 Peer-to-Peer Architecture

2.1 P2P Networks Topologies

In P2P network, file transfers between peers are done directly between the peer sharing the file and the

peer requesting for it. P2P file sharing networks can be classified into four basic categories: the centralized, decentralized, hierarchical and ring systems. Complex system is made by combining the existing system, called hybrid system.

2.1.1 Centralized

Centralized topology is based on the traditional client/server model in which a centralized server manages the files and user databases of multiple peers that log onto it. The client contacts the server to inform it of its current IP address and names of all the files that it is willing to share. The information collected from the peers will then be used by the server to create a centralized database dynamically, that maps file names to sets of IP addresses.

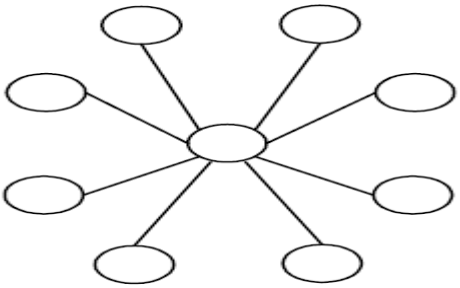

Fig. 3 Centralized Topology

2.1.2 Decentralized

In decentralized architecture there is no centralized server. All the peers are connected with each other forming an unstructured network topology. To join the network, a peer must first contact a bootstrapping node that is always online. Bootstrapping node gives the joining peer the IP address of one or more existing peers. Each peer, however, will only have information about its neighbouring peer. Gnutella is an example of decentralized topology.

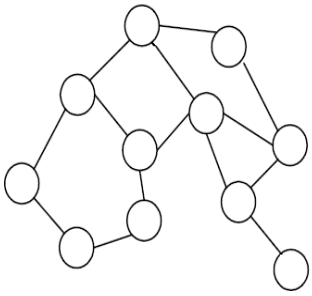

Fig. 4 Decentralized Topology

2.1.3 Hierarchical

This topology is suitable for systems that require a form of governance that involves delegation of rights or authority [16]. An Example of a system that makes use of the hierarchical topology is the Certification Authorities (CAs) that certify the validity of an entity on the Internet [16]. The root CA can actually delegate some of its authoritative rights to companies that subscribe to it, so that those companies can, in turn grant certificates to those that reside underneath it [16].

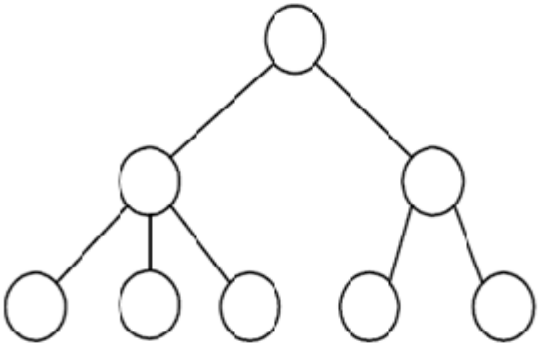

Fig 5 Hierarchical Topology

2.1.4 Ring

It consists of a group of machines which act as a distributed server and they are arranged in the form of a ring. All machines in such system work together and provide better load balancing and higher availability. This topology is used when all the machines are close to each other on the network, for example network owned by an organization; where anonymity is not an issue.

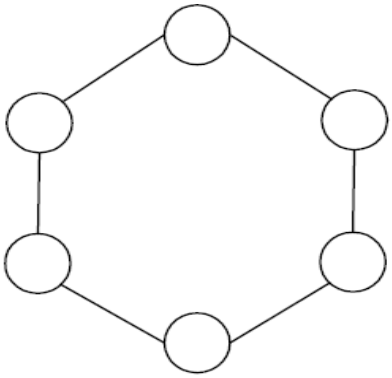

Fig 6 Ring Topology

3. Proposed System

The main title (on the first page) should begin 1-3/8 inches (3.49 cm) from the top edge of the page, centered, and in Times 14-point, boldface type. Capitalize the first letter of nouns, pronouns, verbs, adjectives, and adverbs; do not capitalize articles, coordinate conjunctions, or prepositions (unless the title begins with such a word). Leave two 12-point blank lines after the title.

3.1 System Overview

The system is mainly divided into different modules and the task of file sharing is evenly divided into these modules. If a user wants to join the network it has to register itself into the network. After registration the user should login using the user id and password to join the network. The whole task of registration and login is managed by the authentication server. After successful login the tracker server will respond with the list of available file in the network. The client will select the file from the list which he wants to download and send it to the tracker server, now the tracker server will respond with the list of live nodes or peers having the particular file. Now, after getting the node information the client will make the connection with any of the node given in the list and request for the file and the process of file transfer begins. The file will be divided into smaller chunks and these chunks of files will be delivered to the client. The client can make connection with more than one node simultaneously and download the file. File server will inject a new file into the network, in this process the file will be placed in the shared directory of the file server in the network. Database is used to store client details, file details, node details etc.

The identity of the user involved file sharing in the network is hidden i.e. the identity of user is not revealed either inside or outside the system network.

3.2 System Design

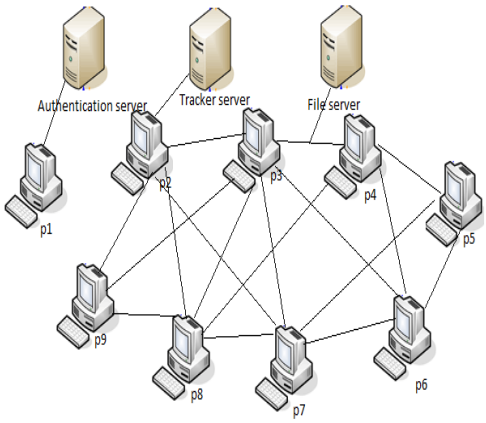

Fig 7 Proposed System Design

As shown in the above figure, peer p1 is interacting with the authentication server for login process so that it can join the swarm and get access into the network. Other peers from are connected to each other in the network. Peers are also connected to the tracker server to get the updated list of files and active nodes in the network at regular time interval. The file server is connected to the network for injecting the new file into the network. File server do not interact with the peers directly.

3.3 System Architecture

The system architecture consists of 5 modules. The modules include:

1) Server Module – Authentication server is responsible for authenticating the client. Tracker server is responsible for providing the information about the live peers involved in the file sharing. File server is responsible for injecting the new file into the network.
2) Client Module – responsible for connecting with the tracker server and retrieving the information about the live nodes, connecting with the other client and uploading and downloading the file.
3) Database Module – responsible for storing and retrieving information about nodes and files available in the system (IP Addresses, port numbers, user IDs, File name, Metadata information etc.).
4) File Search Module – responsible for storing and retrieving information about shared files in the system (filename, host sharing file, local file information for file transfers, etc.
5) File Transfer Module – responsible for both serving requested files stored at a local node and initiating requests to remote nodes for files.

3.4 Server Module

Server module is divided into three sub module depending on the task that they have to perform.

3.4.1 Authentication Server

It performs the initial step of identifying the valid user and allowing him to join the network. The task of this server is to manage the user information, to permit login access to user and to register new user into the network. All user information detail is handled by the authentication server.

3.4.2 Tracker Server

It helps the user to join the swarm by providing him the list of files available into the network and the list of nodes involved in the transfer of particular file. It reduces the overhead of searching for the file in the network.

3.4.3 File Server

File server inserts the file into the system. It is the initial source of file, once the file is inserted into the network the client downloading the file can become the source of the same file for other clients.

3.5 Client Module

Client will login in order to join the network with the help of authentication server. Once the login is made successfully user communicates with the tracker server for file list as well as active node list for a selected file. Client makes connection with the active node and downloads the required file. Client can also share the file with the other requesting user/client.

3.6 Database Module

3.6.1 Peer/Node database

The node database stores all information necessary to join the network and create connections between users. This includes the IP address for each node, as well as the port number that a given node uses to accept new connection requests. Each node also has a user ID, a string representing a user name and password. The database also stores activity data of each and every node in the network.

The node database provides following functionality:

- Adding a node to the database.
- Deleting a node from the system.
- Changing the state of nodes in the database when a user logs out.
- Retrieving the updated list of all nodes in the system.

The node database communicates directly with tracker server and authentication server. Communication with the authentication server is done for peer/user login and registration process. The tracker server interacts directly with the node database to retrieve the list of active nodes in the network involved.

3.6.2 File database

The file database stores all the information necessary to allow individual nodes to transfer files between them. This includes the file name, the user id of the node that has the file, the size of the file, hash value of the file to verify the integrity of the file. The file database provides following functionality:

- Adding a file to the database.
- Removing a file from the database.
- Retrieving a list of all files available in the network.
- Retrieving a list of all nodes serving a specified file.

The file database communicates directly with tracker server and file server. The tracker server interacts directly with the file database to retrieve the list of active nodes in the network. File server will insert the file details when ever new file is injected into the network.

3.7 File Search Module

Whenever a client wants to search for the file, he sends the file list request to the tracker server. The tracker server will respond with the updated list of files available in the network. The user can select the file from the list to download.

3.8. File Transfer Module

- Client can download a large file in parallel from several servers or several other clients simultaneously.
- User chooses a file from the list of files made available by the tracker server to the user and requests for the list of all active nodes for that file from file database.
- Tracker server will respond with the list of active nodes as well as the list of the file pieces and its details i.e. piece id, length, hash value etc required to identify the piece.
- The download task of the large file is divided into smaller sub-tasks (i.e. downloading chunks at a time).
- Client then attempts to establish simultaneous connections to number of nodes from the list of nodes passed to it.
- Once a number of connections are established, the client can download the chunks of file simultaneously from different connections.
- The details of chunks of files downloaded are maintained at user side.
- Downloading the file pieces simultaneously from different connections/nodes allows the file transfer process to distribute the load well amongst the nodes and enhances the performance by utilizing the resources to its maximum and faster downloading of the large files.

4. Conclusion

Recently, privacy has become scarce on public domain networks such as the Internet. Threats such as filtering, traffic shaping and large scale pervasive censorship schemes have left many users without refuge for securing their data as it is transmitted over these networks. While it is foolish to forecast the future as the privacy and security landscape is ever changing, the current trends in online privacy indicate that darknets will play a role as a "last bastion" for users who feel their right to privacy is threatened.

In this research, design and flow of processes for the proposed peer to peer system for secure and efficient file sharing is developed. In this system, the functionality of client/server file sharing system and peer to peer file sharing system is combined to build up a file sharing system which will be fast and secure, maintaining the privacy of the user.

5. References

[1] Biddle Peter, England Paul, Peinado Marcus and Willman Bryan , "The Darknet and the Future of Content Distribution", ACM Workshop on Digital Rights Management. Washington, D.C.: Microsoft Corporation. 18 Nov 2002.
[2] Conor Mc Manamon and Fredrick Mtenzi, "Defending Privacy: The Development and Deployment of a Darknet", Dublin Institute of Technology, Ireland , IEEE 2010.
[3] Chao Zhang, Prithula Dhungel, Di Wu, Zhengye Liu and Keith W. Ross, "BitTorrent Darknets", Polytechnic Institute of NYU, Brooklyn, NY Sun Yat-Sen University, Guangzhou, China, IEEE INFOCOM 2010.
[4] Öznur Altintas and Niclas Axelsson, "Combining Bittorrent with Darknets for P2P privacy".
[5] Symon Aked, "An Investigation Into Darknets And The Content Available Via Anonymous Peer-To-Peer File Sharing", School of Computer and Security Science, Edith Cowan University, Perth Western Australia 2011.
[6] M. Eric Johnson, Dan McGuire and Nicholas D. Willey, "The Evolution of the Peer-to-Peer File Sharing Industry and the Security Risks for Users", Center for Digital Strategies, Tuck School of Business, Dartmouth College, Hanover NH 03755, Proceedings of the 41st Hawaii International Conference on System Sciences – 2008 IEEE.
[7] Megan Hoogenboom , "The darknet as the new Internet A thesis about how a darknet like Freenet is much like the beginning of the public Internet and how it is involved in the future of the net", 1st june 2011.
[8] Johan Andersson & Gabriel Ledung , "Darknet file sharing application of a private peer-to-peer distributed file system concept", Uppsala university,Department of Informatics and Media, Degree Project, 20 june 2010.
[9] Ian Clarke, Oskar Sandberg, Matthew Toseland, Vilhelm Verendel, "Private Communication Through a Network of Trusted Connections: The Dark Freenet".
[10] Shruti Dube, "Peer-to-Peer file sharing accross private network using proxy server", Department of computer science and engineering, Indian Institute of Technology, Kanpur, May 2008.
[11] Jem E. Berkes , "Decentralized Peer-to-Peer Network Architecture: Gnutella and Freenet", University of Manitoba Winnipeg, Manitoba, Canada, April 9, 2003.
[12] Xiaowei Chen, Xiaowen Chu, Jiangchuan Liu , "Unveiling Popularity of BitTorrent Darknets", IEEE Globecom 2010.
[13] Stefanie Roos and Thorsten Strufe , "A Contribution to Analyzing and Enhancing Darknet Routing", IEEE INFOCOM 2013.
[14] Anjali Malekar and Neelabh Sao, "Comparative Analysis of File Transfer Algorithms to Reduce Average Download Time in Peer to Peer Networks", International Journal of Computer Applications Technology and Research Volume 2–Issue 3, 374 - 377, 2013.
[15] Kan, G., (2001), *Gnutella*, Peer-to-Peer: Harnessing the Power of Disruptive Technologies, A. Oram (ed.), O'Reilly Press, USA.
[16] Choon Hoong Ding, Sarana Nutanong, and Rajkumar Buyya, ―Peer-to-Peer Networks for Content Sharing‖, Grid Computing and Distributed Systems Laboratory, Department of Computer Science and Software Engineering, The University of Melbourne, Australia.