ISSN (Print): 2476-8316
ISSN (Online): 2635-3490
Dutse Journal of Pure and Applied Sciences (DUJOPAS), *Vol. 6 No. 2 June 2020*

Demonstrating the Technical Complexities on Transport Layer of Peer to Peer Networks Investigation

Ahmad Musa
Department of Software Engineering,
Faculty of Computer Science and Information Technology,
Bayero University Kano

Email: asmusa.se@buk.edu.ng

Abstract
The size of recorded and transmitted computer resources is rapidly growing, and new challenges are born every day as part of the progression of highly distributed systems. The safe collection and processing of digital evidence have proven to be a complicated business, even in the year's past. A peer to peer (P2P) network is such a distributed system that is dependent on a diverse number of nodes to create and maintain the network. The nodes exchange digital content of various kinds of files such as audio, video, and etcetera while also exchanging CPU, data storage, and other computer resources to maintain the network. The network can be used for many functions, including distributed computing and communication, but it is of best use for sharing large amounts of files and data. However, security and privacy are the most significant concern of the network and addressing it has been a challenge to law enforcement and researchers. While research is still ongoing into how to secure these types of networks, understanding the avenues of preliminary investigations is an essential step that this study has provided. We propose a methodology that monitors and capture P2P network log traffic for analysing. We used Wireshark as a new approach of packet sniffing, capture, and analysis while also documenting the technical challenges faced during the investigation.

Keywords: Wireshark, Investigation, uTorrent, Network, BitTorrent

INTRODUCTION
Today, sharing any digital content through simplified file-sharing systems is effortless, making it accessible to obtain copyrighted material and pirated copies of commercial applications. Peer to Peer (P2P) networks such as BitTorrent uses a decentralized structure that allows users to share files with millions of other peers worldwide. P2P networks such as BitTorrent, Skype, Gnutella, and LimeWire enable anyone to connect with those networks and download any media material, as quickly as downloading a file (Kim et al., 2010). As the usage of file-sharing services keeps growing exponentially through the years, the risks for users kept increasing dramatically at the same time.

**Author for Correspondence*

BitTorrent is a peer to peer application for distributing files that uses metadata files known as torrents. The metadata instructs a BitTorrent client to facilitate connection to remote computers using the client for immediate file download.The client identifies content using aUniform Resource Locator (URL) that is designed to integrate persistently with other clients (Teing *et al.*, 2017). Its superiority over plain Hypertext Transfer Protocol (HTTP) is that multiple secure simultaneous downloads of the same resource are possible (Awan et al., 2006). The downloaders upload chunks of packets to each other, making it feasible for the file source to support vast numbers of downloaders with only an economic increase in its load (Bauer et al., 2010).Today, BitTorrent is the most common protocol to share large digital materials despite any limitation imposed by copyright laws. Through BitTorrent, it is easy to download every type of file: songs, movies, TV shows, games, and software. Unfortunately, the ease and freedom of downloading desired content could pose severe risks for unaware users ranging from malware infection to copyright violations (Venckauskas *et al.*, 2015). Criminals often embed the torrent client with malicious code to gain access to one's computer or destroy it.

Before starting with the analysis of the content that could be downloaded by users, let's introduce the BitTorrent client used for this research. Everyone who wants to download content from the BitTorrent infrastructure has to use a BitTorrent client, the most popular one being uTorrent(uTorrent, 2018; Gao & Zhai, 2016). The uTorrent software client is the most popular BitTorrent peer to peer network application worldwide. When it comes to monitoring BitTorrent traffic, we need to understand how the uTorrent client operates. It is not like the popular traditional download, where downloading is done from a single source or IP address. Rather, pieces of chunks are downloaded from other clients (peers), while the management of the process is being looked after by trackers, also known as Distributed Hash Tables(DHTs) (Leong et al., 2010). Every download started comes with an associated INFO-HASH value, which is anessential to any BitTorrent investigation (Liberatore et al., 2010).

The most reliable form of monitoring investigation is capturing network log traffic as "packets don't lie". According to a Das & Tuna (2017) study, network packet tracing and analysis of network cameras with Wireshark was used to show how partial access to personal information and location can be obtained. The study has validated the use of Wireshark to retrieve information that can be traced to a BitTorrent client. Another study on a preliminary torrent forensics experiment was performed based on a network monitoring module that collects .torrent files and a detection module that analyzes a given resource and decides if it was observed in the network (Alhazmi *et al.*, 2017). Wireshark is a module that is capable of monitoring, collecting, detecting, and analyzing a given resource in a captured network. Our study is distinct from the torrent forensics study in that our given resource is known and is being investigated after capture while the former research is still a hypothesis. Our novelty is based on the usability and reliability of Wireshark because packets don't lie and a backed literature of (Das & Tuna, 2017; Alhazmi *et al.*, 2017).

Wireshark is an extremely powerful tool that can be used to debug networks, examine security problems, inspect network protocols, monitor and capture network traffic (Wireshark, 2016).The network analysis tool captures packets in real-time and display them in human-readable format. We used Wireshark to capture and analyse packets of anuTorrent client in real-time. Wireshark allows the capture of all incoming traffic to and

from the client that passes through it. However, the accuracy of the information that can be recovered is variable. It depends on the methodology used for the packet analysis and the network administrator's third-party software. Hence, there is a need to measure the accuracy and reliability of information revealed through packet analysis. In this study, the usability, reliability, and limitations of Wireshark for network analysis are investigated. The limitations are the technical challenges that can ruin the integrity of data as discussed in the evidence validation section. An illustration of the strategic advantage Wireshark can provide in network monitoring is displayed in fig 1. below:

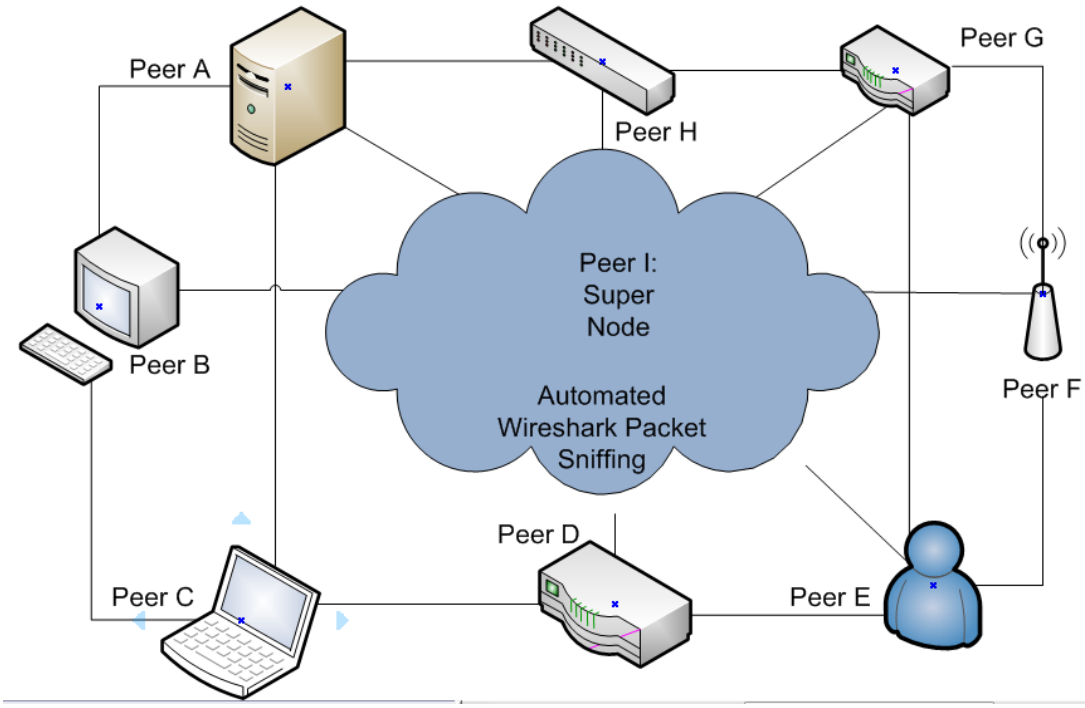


Fig 1: Wireshark Security Architecture (Source: Musa et al., 2019)

MATERIALS

The preliminary stages of this research involve evidence collection using Wireshark on an uTorrent client, specified in the following steps:

Hardware Specification

A Laptop PC running Windows 10 Home 64-bit Operating System, with Installed memory (RAM) of 8.00 GB, 500 GB Hard Disk Drive and Processor details: Intel(R) Core (TM) i5-3230M CPU @ 2.60GHz.

Software Specification

The main software used in the experiment are uTorrent, version 1.8.7 and Wireshark, version 2.6.6.

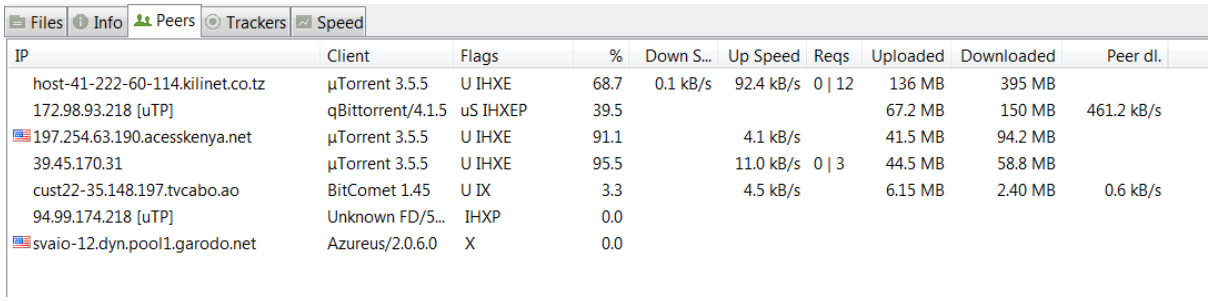
METHODOLOGY

We ensured that the hardware used above (laptop) was relatively new and free of jargon applications that could ruin the result of our experiments. Subsequently, we downloaded and installed the latest versions Wireshark and uTorrent client to our laptop. We then searched for a BitTorrent tracker as described in the introduction section of this study so as to search for a current on demand file for download on the uTorrent client while Wireshark

was running. To achieve that, we made sure that our laptop has no running application on the background. We then started Wireshark capture before introducing the URL tracker to uTorrent for download. As fig 2. Shows, Wireshark was displaying all the peers involved on the network in real-time while downloading at the beginning of the experiment. The whole download session of the file was captured with Wireshark for analysis.

RESULTS AND DISCUSSIONS

As indicated in our experimental set up, we are only interested in verifying the main hypotheses that Wireshark can be used in digital forensic investigations of peer to peer networks. First, we checked that we are able to monitor and generate report while downloading of a file and capture is ongoing as demonstrated in fig. 2 below.


Files | Info | Peers | Trackers | Speed

IP	Client	Flags	%	Down S...	Up Speed	Reqs	Uploaded	Downloaded	Peer dl.
host-41-222-60-114.kilinet.co.tz	µTorrent 3.5.5	U IHXE	68.7	0.1 kB/s	92.4 kB/s	0 | 12	136 MB	395 MB	
172.98.93.218 [uTP]	qBittorrent/4.1.5	uS IHXEP	39.5				67.2 MB	150 MB	461.2 kB/s
197.254.63.190.acesskenya.net	µTorrent 3.5.5	U IHXE	91.1		4.1 kB/s		41.5 MB	94.2 MB	
39.45.170.31	µTorrent 3.5.5	U IHXE	95.5		11.0 kB/s	0 | 3	44.5 MB	58.8 MB	
cust22-35.148.197.tvcabo.ao	BitComet 1.45	U IX	3.3		4.5 kB/s		6.15 MB	2.40 MB	0.6 kB/s
94.99.174.218 [uTP]	Unknown FD/5...	IHXP	0.0						
svaio-12.dyn.pool1.garodo.net	Azureus/2.0.6.0	X	0.0						

Fig. 2 uTorrent clients displayed on Wireshark(Source: Wireshark, 2016)

Secondly, the experimental setup described above proves that peers can be monitored effectively and investigated with credible digital evidence. The captured packets are the one of the most critical preliminary stepsthat can be taken before further analysis and pattern recognition (Scanlon et al., 2014). Fig. 3 below illustrates the proposed flow chart of our experimental setup and the processes of data capture.

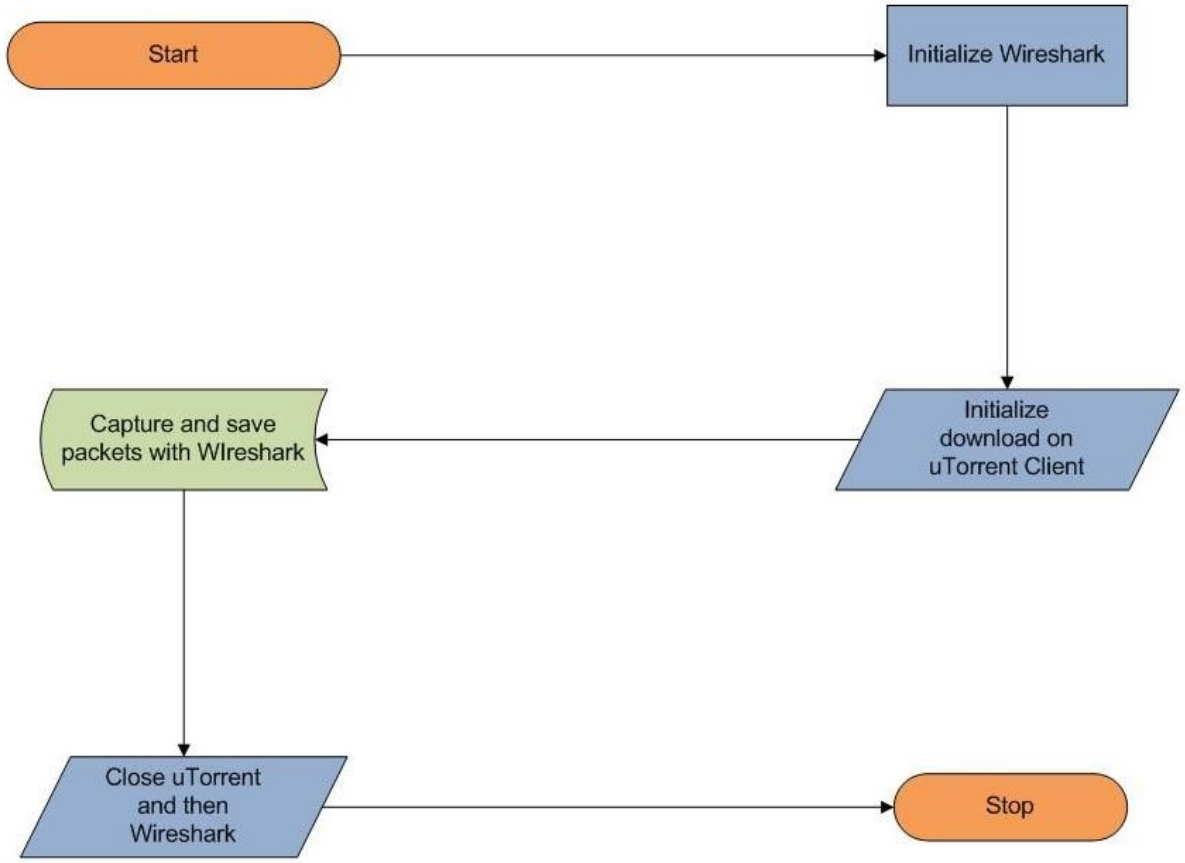


Fig. 3 Flowchart of Experiment

Finally, as stated above, capturing and analysing the data is still an ongoing part of the challenges of investigating peer to peer networks. Several stages of evidence collection are needed during the network data analysis and investigation in peer to peer file networks (Neuner et al., 2016). Problems faced while collecting evidence can be classed as either legal or technical. The legal challenges in peer to peer digital investigation includes jurisdiction, spreading of illegal content, etc. The technical challenges that can derail all of the efforts, as mentioned above, are listed below:

Evidence Validation

Validating and presenting an evidence is one of the most challenging part of a digital investigation. Défense lawyers tend to rely on technical challenges to dismiss any digital evidence presented in court, which made evidence validation a necessary and the most crucial part of all digital investigation. To be admissible in court, validation of evidence collected at every stage is compulsory. The entire investigation could be invalidated if there is a flaw in the initial stages of evidence collection. Therefore, evidence validation is essential. In order to maintain credibility, the initial data collected must be validated in the early stages of the investigation. While gathering information, extreme care must be taken, i.e., at later stage information, the data collected must be able to be verified. To avoid exonerating the accused of criminal charges, the data must be useful, especially in cases where downloading incriminating content takes place due to malicious programs installed on a users' computer(Scanlon et al., 2015). Zeroing in on a particular file must be helped by the information that is collected.While keeping evidence validation in mind, other

complexities that can derail an investigation of p2p systems are recovering an encrypted data and lack of storage space.

Data Encryption

At the network level, peers can decide to complicate the monitoring of P2P network traffic by using encrypted communication(Scanlon et al., 2010). The encryption will prevent obtaining meaningful information from the network and make evidence validation harder. Despite the encryption of network data, Wireshark will still capture all the available data that might be used by the primary evidence collection investigator.

Storage Space

It is crucial to collect only relevant information in the initial stage of the investigation. But logging every outgoing and incoming packet would mean needing a large amount of storage space. Therefore, it is not costly to record all the network traffic. Irrelevant messages must be filter-out if they do not provide any meaningful information for the purpose of investigation (Schrader et al., 2009).

CONCLUSION

The use of peer to peer network has become prevalent for file sharing. Most times, the file shared on these types of networks is related to some sort of unethical activity or cybercrime. Law enforcement has to resort to digital investigations to identify and analyse the suspected data of a crime. Although people think that using a peer to peer network gives them anonymity to share and download files, the security and privacy of being on such a network are questionable. In this paper, we have introduced a new approach that can be used to gather credible evidence during an investigation while keeping in mind thetechnical challenges of monitoring a peer to peer network. Also, the flowchart of our experiment showed the detailed methodology that can be used for future works to ensure confidentiality, reliability and assurance whileanalysing peer to peer traffic.

REFERENCES

Alhazmi, A., Maciá-Fernández, G., Camacho, J. and Salah, S. (2017). *Torrent Forensics: Are your Files Being Shared in the BitTorrent Network?* The Second International Conference on Cyber-Technologies and Cyber-Systems.

Awan, A., Ferreira, R.A., Jagannathan, S. and Grama, A. (2006). Unstructured peer-to-peer networks for sharing processor cycles. *Parallel Computing*, 32(2), pp.115–135.

Bauer, K., McCoy, D., Grunwald, D. and Sicker, D. (2010). BitStalker: Accurately and efficiently monitoring bittorrent traffic. In: *IEEE Xplore*. 2009 First IEEE International Workshop on Information Forensics and Security (WIFS). pp.181–185.

Das, R. and Tuna, G. (2017). Packet tracing and analysis of network cameras with Wireshark. *2017 5th International Symposium on Digital Forensic and Security (ISDFS)*.

Gao, B. and Zhai, J. (2016). A Survey of Covert Channels in BitTorrent Network. *IJISET - International Journal of Innovative Science, Engineering & Technology*, 3(9).

Kim, S., Wang, X., Kim, H., Kwon, T. and Choi, Y. (2010). Measurement and Analysis of BitTorrent Traffic in Mobile WiMAX Networks. *2010 IEEE Tenth International Conference on Peer-to-Peer Computing (P2P)*.

Leong, R.S.C., Lai, P.K.Y., Chow, K.P., Kwan, M.Y.K. and Law, F.Y.W. (2010). *Forensic Investigation of Peer-to-Peer Networks*. Handbook of Research on Computational Forensics, Digital Crime, and Investigation: Methods and Solutions.

Liberatore, M., Erdely, R., Kerle, T., Levine, B.N. and Shields, C. (2010). Forensic investigation of peer-to-peer file sharing networks. *Digital Investigation*, [online] 7, pp.S95–S103. Available at: https://www.dfrws.org/sites/default/files/session-files/paper-forensic_investigation_of_peer-to-peer_file_sharing_network.pdf [Accessed 11 May 2020].

Musa, A., Abubakar, A., Gimba, U.A. and Rasheed, R.A. (2019). An Investigation into Peer-to-Peer Network Security Using Wireshark. *2019 15th International Conference on Electronics, Computer and Computation (ICECCO)*.

Neuner, S., Schmiedecker, M. and Weippl, E.R. (2016). PeekaTorrent: Leveraging P2P hash values for digital forensics. *Digital Investigation*, 18, pp.S149–S156.

Scanlon, M., Farina, J. and Kechadi, M.-T. (2014). *BitTorrent Sync: Network Investigation Methodology*.

Scanlon, M., Farina, J. and Kechadi, M.-T. (2015). Network investigation methodology for BitTorrent Sync: A Peer-to-Peer based file synchronisation service. *Computers & Security*, 54, pp.27–43.

Scanlon, M., Hannaway, A. and Kechadi, M.-T. (2010). A Week in the Life of the Most Popular BitTorrent Swarms. In: *ASIA*. The 5th Annual Symposium on Information Assurance (ASIA'10): Academic Track of 13th Annual NYS Cyber Security Conference,. .

Schrader, K., Mullins, B., Peterson, G. and Mills, R. (2009). TRACKING CONTRABAND FILES TRANSMITTED USING BITTORRENT. In: *Advances in Digital Forensics V*. Springer, pp.159–173.

Teing, Y.-Y., Dehghantanha, A., Choo, K.-K.R. and Yang, L.T. (2017). Forensic investigation of P2P cloud storage services and backbone for IoT networks: BitTorrent Sync as a case study. *Computers & Electrical Engineering*, 58, pp.350–363.

utorrent (2018). *µTorrent - a BitTorrent client*. [online] Utorrent.com. Available at: https://www.utorrent.com/ [Accessed 11 May 2020].

Venčkauskas, A., Damaševičius, R., Jusas, N., Jusas, V., Maciulevičius, S., Marcinkevičius, R., Paulikas, K. and Toldinas, J. (2015). Investigation of Artefacts Left by BitTorrent Client in Windows 8 Registry. *Information Security and Computer Fraud*, 3(2), pp.25–31.

Wireshark (2016). [online] Wireshark.org. Available at: https://www.wireshark.org/ [Accessed 11 May 2020].