



Forensic Analysis of Peer-to-Peer Network Traffic with Wireshark

Ahmad Musa

Department of Software Engineering, Bayero University Kano.
asmusa.se@buk.edu.ng

Abstract

The continuing rise of network security threats and network attacks have motivated accelerated studies on network forensics. Typically, data collected in a networked system is often used to investigate security threats. One of the principles and techniques of network security is packet analysis, which is a branch of network security that studies insecure protocols. In order to help with the forensic investigation and facilitate the fight against security and privacy threats, we carried out an active inspection of network packets on a BitTorrent client. This paper proposes a capture and analysis technique for network packets using Wireshark. Network traffic of P2P networks was monitored, captured, and analyzed. The analysis results showed that the proposed technique successfully identified the source and location of threats on the network, which can be verifiable as credible digital evidence for forensic investigations.

Keywords: *Wireshark, Peer-to-Peer Networks, Packet Sniffing, Packet Capture.*

1. Introduction

The Internet has changed our living and working styles by infiltrating every aspect of people's lives due to the advancement of network technology. The openness, diversity, and interconnectivity of the Internet made it a target to constant attacks by hackers. The aspects of network security like the integrity, authenticity, confidentiality, and applicability of data and information mainly involve two complementary issues - techniques and administration. The Internet security techniques put more emphasis on safeguarding and preventing data and information from illegal users; while the network managers control the administration. The modern challenge coming from this advancement is how to ensure the safety of valuable information by improving Internet security.

The growing need to identify P2P users within the network traffic comes with the continuous popularity of P2P due to the bandwidth it consumes. In this study, we will introduce a new approach based on traffic analysis to discover P2P users and identify the type of P2P applications used. The monitoring, capturing, and analysis of network packets is an essential technique of network forensics, and it is a significant approach for assessing the security level of a network. The research on the capturing and analyzing network packets is still a focal issue in academics and industries (Sanders, 2017).

The advancement of network technologies brings the need for more improved digital security on the internet. As Internet users are increasing, the network traffics are also growing. Research is being done daily on how to make data transfer faster. To this end, network monitoring is employed to secure and manage the network. Packet sniffing can be adopted to monitor data packets through capturing for analysis. The captured log traffic will be used to map the communication between various nodes that exist in the system. The functionality of the Wireshark (Wireshark, n.d.) for packet capturing and analysis is investigated in this study. This technique has been proven to be effective for digital investigations in the past but to the best of our knowledge, this is the first study that uses this technique on BitTorrent networks (Das & Tuna, 2017; Bhandari et al., 2018; An & Lu, 2016; Khanchi et al., 2019). The rest of the paper is outlined as follows: The fundamental functions of Peer-to-



Peer networks and their relativity regarding security are discussed in section 2. Section 3 highlights packet sniffing and network traffic analysis using Wireshark and a case study. Section 4 concludes the paper.

2. Background on Peer-To-Peer Networks

Peer-to-Peer networks are often divided into three main categories, viz - file sharing, processor sharing, and instant messaging. Although a particular client can offer beyond one of these features (Jadoon et al., 2019). Each action presents a unique set of security threats that traditional border firewalls, intrusion detection systems, and antivirus software were not determined to challenge.

P2P file-sharing networks are designed precisely to bypass firewalls. The burden is now on independent users configuring access controls accurately to directories and files on their terminal and influencing corporate Internet security away from defending a single point of entry to the network (Washbourne, 2015). Sensitive material tends to leak either deliberately or accidentally from an organization because of the ease with which P2P systems allow files to be shared (Macfarlane, n.d.). Unaware users often allow access to their whole hard drive, endangering all directories and files, including their encrypted passwords and cookie lists, which could be useful to a hacker.

P2P processor-sharing networks are designed to use unused processor cycles of every peer to offer a distributed computing ecosystem. Demanding computational challenges may be broken into independent, small parts, and the processor shared across a large number of computers all over the world (Venčkauskas et al., 2016). When the screen saver gets switched on, each peer operates on its given project when the user does not use it. To partake in these designs, a user must download a binary code program from the Internet and execute it. Consequently, designing the environments, most organizations try to avoid employing antivirus software with strict configurations. In some cases, benign agents are replaced by less helpful software because P2P networks bypass antivirus protection.

Instant messaging P2P networks, such as Skype, ezTalks, and WeChat, have replaced the traditional Internet Relay Chat (IRC) to transmit real-time and online chat services (Liu et al., 2018). The critical security risk associated with these networks is that third parties can monitor the unencrypted messages traveling across the Internet. Many end-users are unaware that people that they have not explicitly established a conversation can view their unencrypted e-mail and communications. Due to the ease of use of P2P networks, the possibility of confidential or restrictive information to leak from an organization in precisely the same form as with file-sharing systems is big.

The most critical security model that can be used for when P2P clients are in use within an organization is to ensure that the client does not execute as a high-privilege user (an administrator or root) (Liu et al., 2018). As such, the impact of an infected workstation will be reduced if it has restricted network privileges. Also, e-mail clients, web browsers, and other software applications that support inbound network connections should be placed as low-privilege users. These policies are set after weighing the security costs against how much the data is worth. If the security consequences of P2P networks are recognized, it can be used as a tool for information and data sharing between an organization and external partners

2. Packet Sniffing and Network Traffic Analysis

All Packet sniffing, often referred to as packet analysis, outlines monitoring, capturing, and analysing live data as it travels over a system's Network Interface Card (NIC) (Bhandari et al., 2018). Understanding how internet routing works would help explain how packet sniffing occurs. Information and data packets are not transmitted through the internet intact as one file. Rather, the sender breaks the information down into many little data packets. These divided packets are addressed to an IP address at the receiving end which must generally send back acknowledgment receipt of every packet it receives (Anjum et al., 2017; Venčkauskas et al., 2015). Each

packet contains the sender and receiver IP address as well as a lot of other information. The packets do not get transferred from the sender to the receiver in one go. Instead, each packet crosses the internet enroute to its target by passing through several traffic control mechanisms such as switches and routers. The packets that pass through these traffic control agents can be captured for analysis (Khan et al., 2014, 2016; Scanlon & Kechadi, 2014). Packet analysis is done to understand better the happenings on a network. Packet analysis is carried out by a packet sniffer - a tool used to capture complete network packets transmitted over the internet. Packet analysis can support the following (Das & Tuna, 2017):

- a) Learning network features
- b) Identifying who is on a network
- c) Learning who, how or what is consuming bandwidth
- d) Determining peak network routine
- e) Determining malicious activity
- f) Finding unsecured application

A. Wireshark

Wireshark is an open source program that can be used for packet capture and network traffic analysis. It can put the NIC into promiscuous mode, which allows it to accept every packet it receives. Wireshark enables users to see all the traffic being transmitted over the network (Kang et al., 2018). Wireshark utilizes packet capture (pcap) application in capturing packets. Pcap is an archive of information about several protocols, their packet structure, and various messages passed to and fro from these protocols. Therefore, packets can only be captured in pcap format.

Network traffic analysis and packet sniffing works together. The packet sniffing method entails a collaborative effort between software and hardware, while the network analysis is achieved with the help of Wireshark. This process can be divided down into three levels (Gong, n.d):

1. Collection: First, Wireshark collects complete binary data from the Internet. This is achieved by initiating the P2P network interface into promiscuous mode. The NIC monitors to all traffic on the P2P client.
2. Conversion: The obtained pcap data is changed into readable information. This is as far as Wireshark goes. At this time, the network data can be translated into various forms. The end-user does the bulk of the analysis with the essential functions of Wireshark.
3. Analysis: Finally, Wireshark is used to investigate the obtained and converted data. Wireshark verifies the protocol of the captured P2P network data as per the information collected and commences its investigation into the protocol's distinct features.

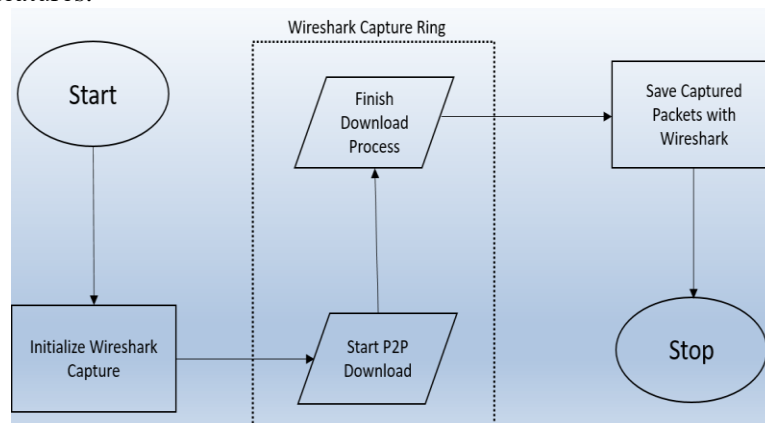
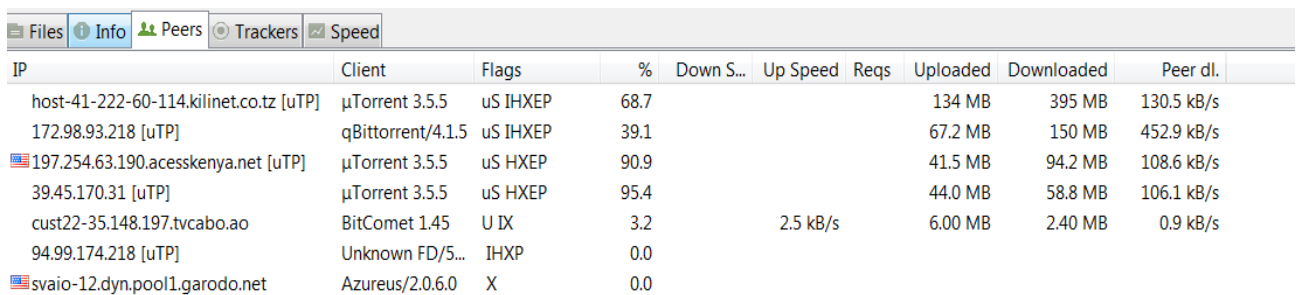


Figure 1: Wireshark Flowchart Model.

Figure 1 shows our research methodology design using a flowchart. Wireshark is a tool that can be used to collect, convert and analyse data all in one place. But due to the complication of the graphical user interface and information overflow shown while and after a capture session, we will be using other methods to further analyse the captured data using a case study.

B. Case Study

In this section, we introduce how P2P client data are captured as network packets using a packet sniffing tool. The tool provides the possibility to monitor, capture and analyze the packets received from P2P clients. Wireshark is the tool that was used for this purpose. Figure 1 shows how our methodology made it possible to capture packets received from and sent to a P2P client



IP	Client	Flags	%	Down S...	Up Speed	Reqs	Uploaded	Downloaded	Peer dl.
host-41-222-60-114.kilinet.co.tz [uTP]	µTorrent 3.5.5	uS IHXEP	68.7				134 MB	395 MB	130.5 kB/s
172.98.93.218 [uTP]	qBittorrent/4.1.5	uS IHXEP	39.1				67.2 MB	150 MB	452.9 kB/s
197.254.63.190.acesskenya.net [uTP]	µTorrent 3.5.5	uS HXEP	90.9				41.5 MB	94.2 MB	108.6 kB/s
39.45.170.31 [uTP]	µTorrent 3.5.5	uS HXEP	95.4				44.0 MB	58.8 MB	106.1 kB/s
cust22-35.148.197.tvcabo.ao	BitComet 1.45	U IX	3.2		2.5 kB/s		6.00 MB	2.40 MB	0.9 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						

Figure 2: Wireshark Capture displaying P2P Clients and IP addresses

The art of packet monitoring from a P2P client starts by identifying the P2P protocol to be used for file sharing purposes. uTorrent is a BitTorrent protocol that was used by first searching for a file tracker on the internet for possible file downloading and sharing. We identified a movie file that is still been shown in the cinema, which will make possessing the movie and sharing it illegal and a crime. We used a movie file for the purpose of our experiment but in real life, forensic investigators can target a particular suspected file. This methodology can be used to track and monitor the activities of users with regards to P2P protocol online and help hold the perpetrators accountable.

To capture packets sent to and received from a P2P client, uTorrent was started at the same time as Wireshark. Wireshark was set to capture mode on a machine that had only uTorrent running for a more precise data collection. The interface of the Wireshark itself is powerful enough to display some basic information of the first peers to participate in the file sharing for few seconds, as shown in figure 2. Our experiment could have ended with the exposure of the few captured peers from figure 2 for analysis and future investigations. But we wanted to know the full information of all and every peer that will participate in the download process. Hence the need for the capture of the whole session.

After we finished capturing the whole session of the suspected file, we exported the data out to a .csv format as shown in figure 3. Then comes the analysis stage and the need for data visualization. We exported our data from fig. 3 into Tableau for data visualization (Tableau, n.d.). We found that most of the communication and file sharing was done by very few peers as opposed to the overflow of data and information we have in figure 3.

Address A	Port A	Address B	Port B	Packets	Bytes	Packets#1/Bytes#1	Packets#2/Bytes#2	Rel Status	Duration	BinA#	BinB#		
27.147.211.187	55146	192.168.1.9	24747	22	2782	11	1407	11	1375	72.77450	13.13103	833.1073	814.1556
41.134.169.95	53982	192.168.1.9	24747	21	3151	10	1632	11	1519	75.63595	11.84201	1162.513	1026.177
41.212.147.95	57963	192.168.1.9	24747	16121	16895545	11104	16585900	5017	309645	0.000005	72.13957	1839311	34338.44
46.40.147.95	52378	192.168.1.9	24747	39	2326	9	1110	10	1216	72.37137	2.337221	3767.345	4126.65
60.284.169.211	53036	192.168.1.9	24747	4259	4685176	3694	453722	1389	75051	34.50465	37.0652	968863	18028.47
103.196.169.211	53036	192.168.1.9	24747	29	2495	9	3321	10	1194	72.36869	2.932104	3593.525	2342.43
164.40.169.99	57915	192.168.1.9	24747	19	2471	9	1336	10	1135	72.51168	1.248790	8558.644	7271.003
192.168.1.9	61386	192.168.1.9	55413	3	218	3	218	0	0	7.542752	8.99880	193.8021	0
192.168.1.9	61387	72.137.211.187	27941	3	218	3	218	0	0	9.152014	6.955760	193.869	0
192.168.1.9	61388	41.134.169.95	80	9	1338	5	1077	4	491	9.787324	0.282167	28996.54	12248.91
192.168.1.9	61389	72.137.211.187	80	3	218	3	218	0	0	9.843793	9.038944	193.5871	0
192.168.1.9	61390	103.196.169.211	27093	15	1518	9	748	6	770	11.14641	0.747952	8000.511	8235.823
192.168.1.9	61391	164.40.169.99	443	1	54	1	54	0	0	17.35740	0	0	0
192.168.1.9	61392	164.40.169.99	443	3	362	2	108	1	54	17.3679	0.139739	7216.5	3808.43
192.168.1.9	61393	54.240.169.215	443	1	54	1	54	0	0	17.36821	0	0	0
192.168.1.9	61394	54.240.169.215	443	2	120	2	120	0	0	17.36844	0.00000	0	0
192.168.1.9	61395	54.240.169.215	443	2	120	2	120	0	0	17.36864	0.00000	0	0
192.168.1.9	61313	216.58.100.66	443	4	264	3	198	1	96	17.36922	0.127213	12006.97	4150.323
192.168.1.9	61394	192.168.1.9	443	4	264	3	198	1	96	17.36939	0.126912	12018.07	4170.022
192.168.1.9	61395	54.240.169.215	443	2	120	2	120	0	0	17.37011	1.40200	0	0
192.168.1.9	61391	60.284.169.211	1030	6	360	3	218	3	162	18.14818	1.918326	909.0000	675.5170
192.168.1.9	61393	216.58.100.66	6881	8	856	5	690	3	206	32.14881	3.870672	1421.902	425.1069
192.168.1.9	61394	54.240.169.215	80	3	218	3	218	0	0	16.04378	6.998892	193.8016	0
192.168.1.9	61393	192.168.1.9	443	1	54	1	54	0	0	44.90889	0	0	0
192.168.1.9	61396	41.134.169.95	24747	6	380	3	218	3	162	45.14413	1.018300	1712.641	1272.696
192.168.1.9	61397	54.240.169.215	6881	10	1427	5	1015	5	412	51.14918	1.540782	2287.464	928.5077
192.168.1.9	61398	54.240.169.215	80	3	218	3	218	0	0	62.23562	8.999992	193.778	0
192.168.1.9	61399	197.84.72.21	18588	3	218	3	218	0	0	69.1783	9.025532	193.6587	0
192.168.1.9	24747	41.134.169.95	9457	19	2443	10	1329	9	1114	72.48888	2.38837	4051.353	3898.754
192.168.1.9	61401	72.137.211.187	80	9	1234	5	769	4	465	72.49391	0.800721	7683.678	4645.813
192.168.1.9	61402	41.134.169.95	80	12	2560	7	2026	5	534	73.10175	0.701661	2898.82	6088.237
192.168.1.9	24747	111.04.170.1	3592	20	2651	10	1259	10	1352	73.3144	2.907047	3502.472	3645.375
192.168.1.9	61403	192.168.1.9	21818	3	218	3	218	0	0	74.21312	6.998934	193.8007	0
192.168.1.9	61405	192.168.1.9	80	9	1495	5	1014	4	401	84.60376	1.297865	6538.827	2918.113
192.168.1.9	57993	192.168.1.9	24747	839	773061	421	384277	418	468794	0.1246537	89.38167	27233.95	41957.95

Figure 3: Download session data exported into a .csv format

Figure 4 shows a clear view of the peers that communicated with our machine. The amount of data contributed for the suspected file download was also displayed for visualization. The mapped figure has now narrowed us down to three suspects through data visualization.

Captured Packets

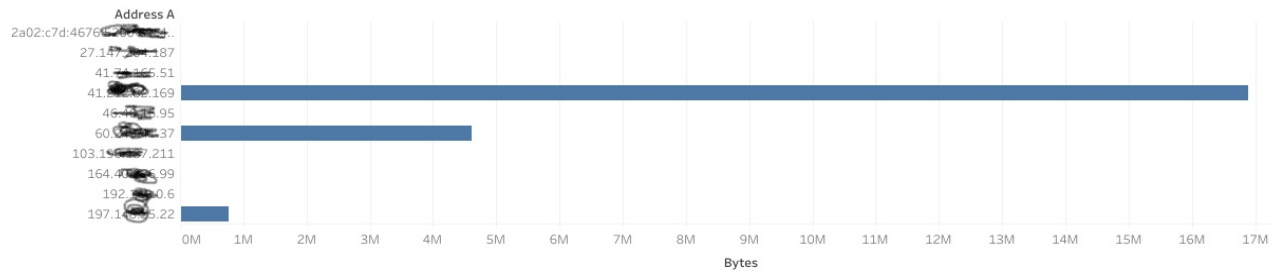


Figure 4. Tableau data visualization

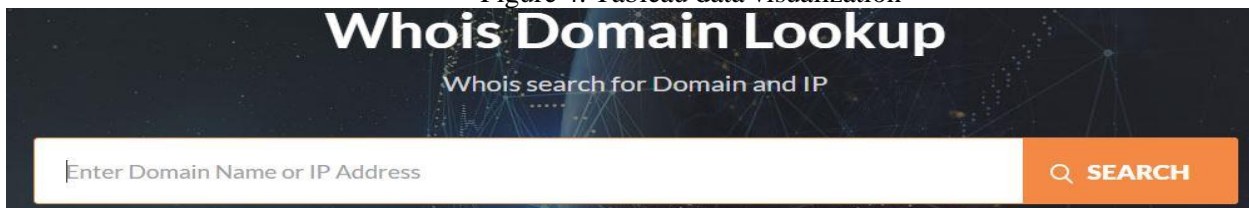


Figure 5: Whois Database used for our experiment

Now that we have identified our suspects, we went ahead with our data analysis in an effort to refine the data and make more sense of it. When any crime is done using a Torrent client, the forensic investigator needs to know all the peers who participated in sharing the copyrighted or illegal item. Hence, we have resolved to support the forensic examiner and provide further evidence based on the captured data. That will be achieved by verifying the three IP addresses obtained from the analyzed files in the open-source Whois database (Whois, n.d.). The database gives first-hand information on the IP provider for each IP address. This initial information allows the forensic examiner to liaise with the IP provider to determine who is using the precise IP address. Figure 5 shows the Whois database we used, while figure 6 shows the amount of information we recovered regarding one of the suspected IP addresses in question. The database alone gives us the country origin, internet provider, state, whether the IP is assigned or not, possible address, contact phone number and the route of the message. With this

information alone, a forensic investigator has gotten more than enough to start off an investigation with a credible lead.

```
Whois IP 41.212.82.169 Updated 2 hours ago

% This is the AfrINIC Whois server.
% Note: this output has been filtered.
% To receive output for a database update, use the "-B" flag.
% Information related to '41.212.82.0 - 41.212.82.255'
% No abuse contact registered for 41.212.82.0 - 41.212.82.255

inetnum:      41.212.82.0 - 41.212.82.255
netname:      KE-NBI-Zuku-HFC
descr:        Wananchi Group Kenya
country:      KE
admin-c:      WL8-AFRINIC
tech-c:       WL8-AFRINIC
status:       ASSIGNED PA
mnt-by:       WOL-MNT
source:       AFRINIC # Filtered
parent:       41.212.0.0 - 41.212.127.255

person:       WGK LIR
address:      P.O Box 10286
address:      Nairobi 00100
address:      Kenya
phone:        tel:+254-20-5205205
nic-hdl:      WL8-AFRINIC
mnt-by:       GENERATED-U2UE79002WQQQGGXEEG9Q4ZFWQXCBOXMG-MNT
source:       AFRINIC # Filtered

% Information related to '41.212.0.0/17AS15399'
route:        41.212.0.0/17
descr:        Wananchi Group (K) LTD
origin:       AS15399
mnt-by:       WOL-MNT
source:       AFRINIC # Filtered
```

Figure 6: Data recovered from Whois database



Figure 7: Possible location of the suspected IP address

Additionally, we used the DNSCHECKER database (Dnschecker.Org, n.d.) that also provides further information of the suspected IP address in question. Along with the same information found by the Whois database, the DNSCHECKER was able to provide us the latitude and longitude coordinates on the map. These coordinates determine the location of the IP address studied. Although the coordinates are not fully accurate; however, they will facilitate the forensic examiner to presume the estimated location of the IP addresses in consideration. Figure 7 showed the result of our analysis on the map. Please note that the IP addresses, recovered details and the coordinates of the suspects were hidden so as to prevent undesirable effects.

3. Conclusion

Network forensics is the art of capturing information that is transmitted over a network and trying to make sense of it in some forensics capacity. Forensic network investigation aims to guarantee the quality of the data



provided and can contribute towards adequately securing the P2P networks. In this study, a case study was presented on how Wireshark can be used to monitor and trace P2P users in detail. The case study shows that packet sniffing can be used to defeat the anonymity of peer-to-peer users online. Hence, by law enforcement, P2P users can be tracked with the help of the credible evidence obtained using Wireshark and our analysis using various other tools. Our future work will focus on analysing the captured data with more accuracy by developing a tool that will incorporate the characteristics of all the tools used in this study for easier accessibility. This will also allow for more credible and verifiable evidence, which is the focal aim of network forensics. Our research is challenged with the possibilities of peers hiding under a proxy server or a virtual private network (VPN) to mask their identities. This challenge can be softened with the access a government-backed digital investigator possesses. The companies involved in the masking can be queried to reveal the identities of the suspects.

Reference

- An, X., & Lu, X. (2016). Packet capture and protocol analysis based on Winpcap. Proceedings - 2016 International Conference on Robots and Intelligent System, ICRIS 2016, 272–275. <https://doi.org/10.1109/ICRIS.2016.55>
- Anjum, N., Karamshuk, D., Shikh-Bahaei, M., & Sastry, N. (2017). Survey on peer-assisted content delivery networks. *Computer Networks*, 116, 1339–1351. <https://doi.org/10.1016/j.comnet.2017.02.008>
- Bhandari, A., Gautam, S., Koirala, T. K., & Ruhul Islam, M. (2018). Packet sniffing and network traffic analysis using TCP—A new approach. In *Lecture Notes in Electrical Engineering* (Vol. 443, pp. 273–280). Springer. https://doi.org/10.1007/978-981-10-4765-7_28
- Das, R., & Tuna, G. (2017). Packet tracing and analysis of network cameras with Wireshark. 2017 5th International Symposium on Digital Forensic and Security, ISDFS 2017. <https://doi.org/10.1109/ISDFS.2017.7916510>
- dnschecker.org. (n.d.). Retrieved October 10, 2019, from <https://dnschecker.org/ip-location.php>
- Gong, Y. (n.d.). identifying-p2p-users-using-traffic-analysis. In Symantec White Paper. <https://www.symantec.com/connect/articles/identifying-p2p-users-using-traffic-analysis>
- Harichandran, V. S., Breitingner, F., Baggili, I., & Marrington, A. (2016). A cyber forensics needs analysis survey: Revisiting the domain's needs a decade later. *Computers and Security*, 57, 1–13. <https://doi.org/10.1016/j.cose.2015.10.007>
- Jadoon, A. K., Iqbal, W., Amjad, M. F., Afzal, H., & Bangash, Y. A. (2019). Forensic Analysis of Tor Browser: A Case Study for Privacy and Anonymity on the Web. *Forensic Science International*, 299, 59–73. <https://doi.org/10.1016/j.forsciint.2019.03.030>
- Kang, S., Kim, S., & Kim, J. (2018). Forensic analysis for IoT fitness trackers and its application. *Peer-to-Peer Networking and Applications*. <https://doi.org/10.1007/s12083-018-0708-3>
- Khan, S., Gani, A., Wahab, A. W. A., Shiraz, M., & Ahmad, I. (2016). Network forensics: Review, taxonomy, and open challenges. *Journal of Network and Computer Applications*, 66, 214–235. <https://doi.org/10.1016/j.jnca.2016.03.005>
- Khan, S., Shiraz, M., Wahid, A., Wahab, A., Gani, A., Han, Q., Bin, Z., & Rahman, A. (2014). A Comprehensive Review on Adaptability of Network Forensics Frameworks for Mobile Cloud Computing. The An, X., & Lu, X. (2016). Packet capture and protocol analysis based on Winpcap. Proceedings - 2016 International Conference on Robots and Intelligent System, ICRIS 2016, 272–275. <https://doi.org/10.1109/ICRIS.2016.55>
- Anjum, N., Karamshuk, D., Shikh-Bahaei, M., & Sastry, N. (2017). Survey on peer-assisted content delivery networks. *Computer Networks*, 116, 1339–1351. <https://doi.org/10.1016/j.comnet.2017.02.008>
- Bhandari, A., Gautam, S., Koirala, T. K., & Ruhul Islam, M. (2018). Packet sniffing and network traffic analysis using TCP—A new approach. In *Lecture Notes in Electrical Engineering* (Vol. 443, pp. 273–280). Springer. https://doi.org/10.1007/978-981-10-4765-7_28
- Das, R., & Tuna, G. (2017). Packet tracing and analysis of network cameras with Wireshark. 2017 5th International Symposium on Digital Forensic and Security, ISDFS 2017.



- <https://doi.org/10.1109/ISDFS.2017.7916510>
- dnschecker.org. (n.d.). Retrieved October 10, 2019, from <https://dnschecker.org/ip-location.php>
- Gong, Y. (n.d.). identifying-p2p-users-using-traffic-analysis. In Symantec White Paper. <https://www.symantec.com/connect/articles/identifying-p2p-users-using-traffic-analysis>
- Harichandran, V. S., Breitinger, F., Baggili, I., & Marrington, A. (2016). A cyber forensics needs analysis survey: Revisiting the domain's needs a decade later. *Computers and Security*, 57, 1–13. <https://doi.org/10.1016/j.cose.2015.10.007>
- Jadoon, A. K., Iqbal, W., Amjad, M. F., Afzal, H., & Bangash, Y. A. (2019). Forensic Analysis of Tor Browser: A Case Study for Privacy and Anonymity on the Web. *Forensic Science International*, 299, 59–73. <https://doi.org/10.1016/j.forsciint.2019.03.030>
- Kang, S., Kim, S., & Kim, J. (2018). Forensic analysis for IoT fitness trackers and its application. *Peer-to-Peer Networking and Applications*. <https://doi.org/10.1007/s12083-018-0708-3>
- Khan, S., Gani, A., Wahab, A. W. A., Shiraz, M., & Ahmad, I. (2016). Network forensics: Review, taxonomy, and open challenges. *Journal of Network and Computer Applications*, 66, 214–235. <https://doi.org/10.1016/j.jnca.2016.03.005>
- Khan, S., Shiraz, M., Wahid, A., Wahab, A., Gani, A., Han, Q., Bin, Z., & Rahman, A. (2014). A Comprehensive Review on Adaptability of Network Forensics Frameworks for Mobile Cloud Computing. *The Scientific World Journal*, 2014.
- Khanchi, S., Zincir-Heywood, N., & Heywood, M. (2019, April 1). Network Analytics for Streaming Traffic Analysis. *IEEE Xplore*. <https://ieeexplore.ieee.org/abstract/document/8717885>
- Kolenbrander, F., Le-Khac, N.-A., & Kechadi, M.-T. (2016). Forensic Analysis of Ares Galaxy Peer-To-Peer Network. *Proceedings of the Conference on Digital Forensics, Security and Law*, 21–35. https://search.proquest.com/docview/1916621172?accountid=12507%0Ahttp://openurl.ac.uk/athens:mmu/?url_ver=Z39.88-2004&rft_val_fmt=info:ofi/fmt:kev:mtx:journal&genre=article&sid=ProQ:ProQ%3Acriminaljusticeperiodicals&atitle=FORENSIC+ANALYSIS+OF+ARES+GALAXY
- Liu, G., Yan, Z., & Pedrycz, W. (2018). Data collection for attack detection and security measurement in Mobile Ad Hoc Networks: A survey. *Journal of Network and Computer Applications*, 105(October 2017), 105–122. <https://doi.org/10.1016/j.jnca.2018.01.004>
- Macfarlane, R. (n.d.). Lab 5 : Packet Capture & Traffic Analysis with Wireshark. 1–16.
- Sanders, C. (2017). *Practical Packet Analysis, 3rd Edition: Using Wireshark to Solve Real-World ...* - Chris Sanders - Google Books.
- Scanlon, M., Farina, J., & Kechadi, M. T. (2015). Network investigation methodology for BitTorrent Sync: A Peer-to-Peer based file synchronisation service. *Computers and Security*, 54, 27–43. <https://doi.org/10.1016/j.cose.2015.05.003>
- Scanlon, M., & Kechadi, T. (2014). The Case for a Collaborative Universal Peer-to-Peer Botnet Investigation Framework. *Proceedings of the 9th International Conference on Cyber Warfare and Security*, 287–293. <https://doi.org/10.1038/nature03184>
- Tableau. (n.d.). Retrieved October 10, 2019, from <https://public.tableau.com/en-us/s/>
- Venčkauskas, A., Jusas, V., Paulikas, K., & Toldinas, J. (2015). Investigation of artifacts left by bittorrent client on the local computer operating under windows 8.1. *Information Technology and Control*, 44(4), 451–461. <https://doi.org/10.5755/j01.itc.44.4.13082>
- Venčkauskas, A., Jusas, V., Paulikas, K., & Toldinas, J. (2016). A methodology and tool for investigation of artifacts left by the BitTorrent client. *Symmetry*, 8(6). <https://doi.org/10.3390/sym8060040>
- Washbourne, L. (2015). A survey of P2P network security. 1–12. <http://arxiv.org/abs/1504.01358>
- Whois. (n.d.). Retrieved October 10, 2019, from <https://www.whois.com/whois/>
- Wireshark. (n.d.). Wireshark. Retrieved June 1, 2019, from <https://www.wireshark.org/>