Peer to Peer File Sharing

Security Concerns, Unwanted Traffic Detection and Filtering

Leonardo Carvajal
Department of Computer Science
Sam Houston State University
Huntsville TX, USA

Lei Chen
Department of Computer Science
Sam Houston State University
Huntsville TX, USA

Abstract—Peer to Peer (P2P) file sharing systems have been introduced in business networks yet they are causing certain security problems in the network infrastructure. Securing P2P networks can be challenging as data traffic in such networks is difficult to manage and peers may have very different security settings and configurations. For example, P2P applications can use different port numbers as a configuration parameter, or simply use a random port number. Furthermore, not every P2P application supports encryption and decryption. This paper summarizes the existing solutions for detecting and stopping unwanted data traffic in P2P networks.

Keyword: Peer to Peer, P2P, sharing, security, threats, attacks

I. Introduction

The use of P2P file sharing applications has greatly increased in recent years. Peer to Peer networks support hundreds of millions of users and generate the majority of Internet traffic [1]. However, P2P applications are also causing problems, including security threats, vulnerabilities, excessive network usage, and legal problems associated with copyrights. Peer to Peer file sharing applications establish multiple TCP connections using different ports between peers to transfer data making it difficult to control network saturation [11]. Moreover, malicious code can take advantage of the regular use of the P2P networks to propagate messages, introduce executable files into a system and trick users into downloading and executing infected files. Therefore, network administrators and Internet service providers are required to monitor unwanted network traffic and create policies on the usage of these applications in order to enforce data integrity, confidentiality and availability, as well as the illicit trade of copyrighted material. In the past few years, there has been a vast amount of research towards enforcing network security in peer to peer networks by combining existing mechanisms to detect unwanted network traffic and reinforce organizational policies.

There are different solutions for different companies. The type of organization can vary in size, e.g. small or large, and type, e.g. public or private. This paper expands on existing solutions to identify and stop unwanted peer to peer network traffic, as well as other tools to protect peer to peer applications against attacks. From the network administrator point of view, these solutions would protect the network infrastructure from infringing activities due to vulnerabilities in P2P applications [9].

This paper is structured as follows: section II presents the security issues in peer to peer file sharing networks; section III describes solutions proposed by other researchers about how to detect malicious peer to peer activity; section IV presents other tools to protect peer to peer networks against unwanted traffic; and the paper finishes with conclusions.

II. Issues in peer to peer file sharing networks

Unlike client/server architecture, P2P network services are provided by many nodes simultaneously functioning as both clients and servers. In a P2P networks, nodes play an important role: they control the exchange of data, allow users to share resources, support communication, and provide directory services as well as real time collaboration tools [4]. Decentralized P2P networks spread services among all nodes. An effective attack to peer to peer networks may shut down the nodes offering specific file resources, and attacks to a single node may or may not have an effect on the entire network. There are several issues that are found in P2P file sharing systems.

A. *Unpredictable Network Usage*

Peer to Peer applications normally take as much bandwith as available [12]. Files available in P2P networks are generally larger. Typical peers serve multi-megabytes of files overloading the network. For example, an audio file is usually from 3 to 5 megabytes, and a video file can be hundreds of megabytes.

B. *Exposure of Sensitive Data or Personal Information*

P2P users have been observed unintentionally or intentionally sharing private files, including sensitive corporate information [1]. Some users of peer to peer networks do not know about basic computer security. Therefore, these users can share their entire hard drive, allowing attackers to obtain sensitive data, such as operating system files, applications files, and registries. In addition, P2P networks are well known for the distribution of malicious code. Many of the shared files are infected with malware and are spread to peers.

C. DDOS Attacks

Many attackers are looking for controllable peer to peer networks users, or zombies [3]. These zombies send packages to selected targets in order to get the victim's resources. As a result, the victims will not be able to provide its services. In addition, downloading files can consume bandwidth and may decrease the availability of other network services or systems.

D. Danger of Legal Action

In many cases, P2P file sharing networks are used to support illegal activities because many available files in P2P networks are copyrighted [1]. These illegal activities may not be limited to the end user and may be extended to the network sponsor. While many countries do not enforce penalty and punishment on copyright infringement offenses, other countries do. P2P networks provide sharing infrastructure that is harder to track and difficult to block, providing cover for espionage and criminal activity [2].

E. Content Verification Susceptible to Attack

Attackers can introduce files without content, modify files, or share files with malicious code [6]. Therefore, integrity verification of the requested content should be verified. P2P file sharing applications use different ports. Moreover, opening these ports may give access to attackers to the computer network, or attackers can take advantage of the P2P applications vulnerabilities [5].

F. Malware

Malicious code also exists in peer to peer networks [14]. Malware has the ability to spread across P2P infrastructures by replicating themselves. Malware is placed in shared folders and has names of popular movies, music, or applications in order to catch the attention of the users. Moreover, malicious code also uses other attack vectors including denial of service and has the ability to open backdoors making users' confidential files available to other peers [14].

III. DECTECTION OF P2P ACTIVITIES

This section presents two mechanisms which are used to detect P2P activities.

A. Intrusion Detection Systems

Based on the detection of encrypted traffic generated by one of the most popular P2P applications, GoalBit, authors in [8] propose a method to detect peer to peer traffic using intrusion detection systems, specifically using a set of rules. Due to the nature of the analysis, the proposed rules are signature based, focusing on identifying patterns. This method relies on the findings of repetitive string series on the data field of the IP packets, during the link phase or other critical connection points when encryption is not used. This approach is implemented in Snort (the most popular intrusion detection system) to detect signatures and block traffic matching from the protocol signatures. The following rules have been taken from [8] to demonstrate how this method works:

Rule 1000506/TCP traffic: alert tcp any any → any any (msg: "LocalRule: GoalBit tracker Cookie"; content:" | 47 67 61 6c 42 69 74 20 70 72 6f 74 6f 74 6f 63 6f 6c |"; dsize: 77; threshold: type both, track by_src, count 1, seconds 10, sid: 1000506; rev:1;)

In this rule, all TCP traffic coming into the network is scanned to find a GoalBit signature on IP packets. Once the signature is detected, the IDS track the source's IP address and if at least 1 event of the SID is fired, this rule alerts once every 10 seconds.

Rule 1000509/TCP traffic: alert tcp any any → any any (msg: "LocalRule: GoalBit Pattern | 00 0d 06 00 00 |"; flow: established; content:" | 00 0d 06 00 00 | " ; threshold: type both, track by_src, count 3, seconds 10, sid: 1000509; rev:2;)

Rule 1000565/TCP traffic: alert tcp any any → any any (msg: "LocalRule: GoalBit Pattern TCP payload size (1460 bytes)"; content:" | 62 72 6f 61 64 63 61 74 65 72 7b 70 69 65 | " ; depth:90; dsize: 1460; threshold: type both, track by_src, count 3, seconds 10, sid: 1000565; rev:1;)

Rule 1000566/TCP traffic: alert tcp any any → any any (msg: "LocalRule: GoalBit Pattern TCP payload size (1456 bytes)"; content:" | 67 6f 61 6c 62 69 74 5f 74 72 61 63 | " ; dsize: 1460; threshold: type both, track by_src, count 3, seconds 10, sid: 1000566; rev:1;)

The above rules were created during periods when encryption was not being used, such as at the start and in the middle of the transmissions [8]. Rule 1000506 was triggered when it receives the first bytes of the TCP session. This rule checks any encrypted or non encrypted communication to find any GoalBit signatures. Rules 1000565 and 1000566 were created for large payload sizes. Packets with this payload signature, for the most part, were not encrypted. It is very important to mention that the total accuracy of this detection method rate is 96% [8].

B. Multi-Phased P2P Flow Model

Authors in [10] proposed a method that consists of three steps based on detecting malicious traffic. First, the flow grouping step involves clustering of TCP/UDP connections. In this step, authors track packets to determine if they are normal transmissions or flooding attacks. The segmented connection is the unit of the grouping. If there is a TCP session, ACK packets are discarded with a payload size of zero. Flows are processed to determine the similarity of each flow. Flows are considered different if their time gap is longer than 240 seconds and the threshold is greater than 0.5. Clustering of a flow occurs if a flow is link to at least one other flow, even though it is not link to all flows in a cluster. Second, the flow compression computes the state value of each flow of group and extracts the transition information. In order to define a state, the authors in [10] use seven features of the clustered flow. Each value of the features can only be 0 or 1. The features and values are as follows: protocol (TCP (0), UDP (1)), port (inside port (random port (0), reserved port (1)), outside port (random port (0), reserved port (1)), connection count (connections ≥ minimum count (0), connections < minimum count (1)), connection interaction (round trip (0), one way(1)), packet count comparison (inbound ≤ outbound (0), inbound > outbound (1)), and traffic volume comparison (inbound ≤ outbound (0), inbound > outbound (1)). The F values are the following: protocol (PT (64)), port (IP (32),OP(16), CC(8), traffic (CI (4), PC(2), TV(1)). In the last step, the algorithm constructs a matrix based on transitions of flow modeling. The detection engine uses the ratio computed from the probability-based models [10]. The detection rate of this model is 97% [10].

IV. Approaches against worms and unwanted P2P traffic

This section presents several ways to detect passive and active worms, application management tools to monitor and control unwanted P2P traffic, and network security policies to prevent the use of P2P applications for illegal file sharing. In addition, P2P topologies are also considered to obtain accurate information from other peers.

A. *Passive Worm Detector Based on Hash Values*

Based on hash values, authors in [13] proposed a method of detecting passive worm and malware. Files acquire an identifier for the content of each file that is hashed. Therefore, different versions of the same file have distinct hash values. Passive worms have the ability to propagate to other peers as files are copied to other hosts. Although worms are replicated with different file names, their code will be the same. This means that the hash value of the infected files will be the same [13]. Extracting the hash values and looking to see if those values represent multiple files indicates that malicious code might be present in a P2P infrastructure. However, having multiple files and same hash values detecting worms will not be easy. So, the way to detect worms is based on the popularity of the hash which increases in a short period of time [13]. This detection system has the following elements: data collector, which acquire the IP addresses of peers, shared folders, hash values, and obtain file names and other information; finding hash values that increase over time is a task of the popularity analyzer; worm detectors track hash values which increase over time to determine if malware is present on the P2P infrastructure.

B. *Active Worm Attacks*

In [16] authors define a propagation P2P attack model based on three worm attack strategies: random based, attack, offline based attack, and online based attack. The random-based attack happens when the worm peer chooses IP addresses randomly of victim peers in order to launch the attack. In the offline P2P-based attack, infected peers obtain the IP addresses of offline peers. This information is maintained in a list called the hit-list [16]. The attack is launched based on the hit-list and the infected peers can continue launching the attack using the random-based attack. In the online P2P-based attack, after adhering to the P2P at the system's initial time, infected peers launch the attack to their neighbors. At the same time infected peers can infect other peers using the random-based attack [16]. In order to evaluate how the active worms attack affects and propagates on P2P systems, authors take into account the P2P characteristics or parameters and attacker parameters. The P2P system characteristics are P2P size, P2P vulnerability, P2P topology degree, and structured and unstructured P2P. The attacker parameters are the attack scan rate and the system's initial infected worm instances [16]. The following are the results obtained by the authors in [16] based on the topology degree in structured P2P systems, this P2P parameter only has impact on the online P2P –based attack. Based on the topology degree in unstructured P2P systems, the power-law distribution was used to determine the degree distribution to other P2P network hosts. Therefore, this method only determines the topology degree. Based on P2P size, this characteristic will have impact on both offline and online based attacks. Based o P2P vulnerabilities, this parameter will depend on how well protected the peers are in home environments as well as in organization environments.

Authors in [17] analyze the impact of how a new worm propagation threat is spread in BitTorrent due to its vulnerable topology to active worms. In contrast with [16], where authors do not take into account the cooperation of worm infected peers to share the attack information, authors in [17] consider the level of cooperation on the infected peers. Based on the same parameters or characteristics of the P2P systems and attacker parameters mentioned in [16], in [17] authors include the Internet parameters. These parameters are the connection speed, patch rate when an infected machine becomes impenetrable, and death rate when an infection is detected on a peer and removed without patching [17]. These are the results obtained in [17]: based on the impact of the attack strategy, the BitTorrent Worm (BTW) attack can reach its speed of propagation up to 300% compared with traditional scanning method. Based on the impact of P2P system size, the results show that BTW performance can differ. If the network size is large, the attack performance is higher. Based on the impact of P2P topology degree, the results shows that if the topology degree increases, peers are open to the BTW and the speed of propagation also increases.

C. *P2P Aplication Management Tools*

In [12] Lai mentions that almost 2.5 billon downloads occur every month using P2P applications. Organizations are making request to ISPs and network administrators to eliminate potential threats and illegal P2P file sharing. However, most companies have insufficient budgets to employ enough staff members for their network operation and even less resources to manage P2P usage [12].

An unsuccessful method that many companies use to block peer to peer traffic is blocking P2P traffic ports using hardware or software firewalls. P2P applications can use different ports to overcome port blocking [12].

Monitoring tools such as Network Instruments Observer can identify the top users of the network, break down web traffic and generate Internet traffic activity reports per user or by department. These tools give a real time picture of actual protocols running across the network, and help network administrators collect information and troubleshoot network issues.

Bandwidth management tools allow network administrators to detect and stop P2P traffic [12]. NetEnforcer can limit the use of bandwidth, prioritize network traffic per application and per user, control the bandwidth utilization and costs associated, while protecting and enhancing service quality for all network users. Using NetEnforcer or similar tools, companies can prioritize business-critical applications. This tool also includes application layer protocol monitoring and application signature detection to control P2P applications [12].

Another bandwidth management tool, Packet Shaper allows administrators to set policies that provide a limit on the bandwidth usage on application type identifying peer to peer application traffic. It permits bandwidth management according to the priority of the application. This tool can prevent denial of service attack. It detects and stops SYN floods and ICMP packets [12].

A new P2P detection tool called Watchdog can detect encrypted peer to peer traffic. SSL encrypted peer to peer file transfer sessions on any port as well as sessions that are hidden behind HTTP proxies can be detected and tracked by this tool [12]. This tool is capable of blocking file transfers.

Audible Magic's CopySense appliance handles illegal peer to peer file sharing of copyrighted works [12]. This tool filters illicit traffic of copyrighted content, allowing network administrators to manage and control network traffic. Audible Magic's CopySense utilizes a database of file signatures for copyrighted media. It can identify over 3.5 million recorded songs [12]. Infractions are tracked and addressed in real time, reducing the use of the network for unwanted traffic. The Integrated Computer Application for Recognizing User Services (ICARUS) tool can block the infringing and non-infringing P2P traffic. In contrast, Audible Magic's CopySense Network Appliance only blocks the infringing use of P2P file sharing applications.

D. P2P Network Security Policies

Network security standards, policies and procedures must be followed and enforced to prevent the use of P2P applications for illegal file sharing. Policies should focus on the prevalent use of this technology that is only for distribution of copyrighted content. Other concerns of peer to peer file sharing applications include network utilization, network security, malicious code and inappropriate content. Policies will support the primary usage of the network for operations of organizations' daily business. In most cases, violations of security policies can result in firing employess and criminal prosecution under state and federal statutes.

E. P2P Topology Path Length and Hierachy

Authors in [15] present aspects that will affect the DoS resilience of P2P systems based on: hierarchy and k - regular topology. These models are based on the probability of obtaining accurate information. In a P2P hierarchy, supernodes are target for DoS attacks because they store the directory of the files that are shared, as well as the information about the connectivity with other supernodes in order to replay clients' file petitions. The following are the results obtained in [15] with the hierarchy model: having a P2P infrastructure with corrupted nodes, if client's petition needs just one supernode hops to reach the solicited file, the probability of obtaining the correct replies is 81%. However, if a client's petition needs 5 supernode hops to reach the solicited file, the probability of obtaining correct replies is 53.1%. Therefore, while paths are longer, the possibility of obtaining correct replies is reduced. In K-regular Topologies, topologies are adjacent nodes that have the same number of neighbors (k) [15]. However, the number of neighbors may not be the same if the peers consider an anonymous connection. Therefore if a peer's file petition is requested, it may require a higher number of hops. This means that the probability of obtaining correct replies is lower. Therefore, attacks to peer to peer file sharing systems are higher [15].

V. Conclusion

The popularity of P2P file sharing applications has increased security risks for organizations. Most organizations are concerned about how these kinds of applications saturate the network infrastructure with music, videos and other organizations' resources not related to the goals of the organization. Another problem is that files downloaded to organizations' computers might be illegal copies of copyrighted material. Information Technology departments use a variety of mechanisms to prevent the unauthorized use of P2P applications within the organization. This paper has presented the most relevant approaches and tools to detect and prevent unwanted P2P activities, including strong policies and other mechanisms such as scanning and blocking network traffic of suspicious activities.

References

[1] Danny Hughes, Kevin Lee, and James Walkerdine , ”On the Penetration of Business Networks by P2P File Sharing,” in *Second International Conference on Internet Monitoring and Protection*, 2007.

[2] Rosslin John Robles, Min-Kyu Choi, and Eun-suk Cho, “A Paradigm Solution to P2P Security Issues,” in *Int. e-Commerce Advance Science and Technology*, 2009, pp. 3-7.

[3] Jiri Schafer and Kamil Malinka,"Security in Peer to Peer Networks Empiric Model of File Diffusion in BitTorrent," in *4th Int. Conf. Internet Monitoring and Protecting*, 2009, pp. 39-44.

[4] Huu Tran, Michael Hitchens, Vijay Varadharajan, and Paul Watters, "A trusted Access Control Framework for P2P File-Sharing Systems," in *Proceedings of the 38th Hawaii International Conference on Systems Science*, 2005, pp. 1-10.

[5] Amuthan A, Marimuthu.G, and Kaliaperumal.G, "Secure Trust Management Model for Peer to Peer File Sharing System," in *Int. J. Recent Trends Engineering and Technology*, 2010, pp. 90-96.

[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," in *Proceedings of the 41st Hawaii International Conference on System Science*, 2008, pp. 1-10.

[7] Alexis Ulliac and Bogdan V. Ghita, " Non Intrusive Identification of Peer to Peer Traffic," in *2010 Third International Conference on Communication Theory, Reliability, and Quality of Service*, pp. 116-121.

[8] Andre F. Esteves, Pedro R. M. Inacio, Manuela Pereira, and Mario M. Freire, "On-Line Detection of Encrypted Traffic Generated by Mesh-Based Peer-to-Peer Live Streaming Applications: The Case of GoalBit," in *2011 IEEE International Symposium on Network Computing and Applications*, pp. 223-228.

[9] Kevin W. Hamlen and Bhavani Thuraisingham, "Secure Peer-to-peer Networks for Trusted Collaboration," in *2007 Collaborative Computing; Networking, Applications and Worksharing*.

[10] Sang-Kyun Noh, Joo-Hyung Oh, Jae-Seo Lee, Bong-Nam Noh, and Hyun-Cheol Jeong, "Detecting P2P Botnets using a Multi-Phased Flow Model," in *2009 Third International Conference on Digital Society*, pp. 247-253.

[11] Wei Li, Shanzhi Chen, Yaning Liu, and Xin Li, "Aggregate Congestion Control for Peer-to-Peer File Sharing Applications," in *Ninth ACIS International Conference on Software Engineering, Artificial Intelligence, Networking, and Parallel/Distributed Computing*, 2008, pp. 700-705.

[12] Wayne Lai. (2004, January 5). *Managing Peer-to-Peer Applications in Dormitory* [Online]. Available: http://www.sans.org/reading _ room/whitepapers/tools/managing-peer-to-peer-applications-dormitory-networks_1348

[13] Sahar Fahimian, Amirvala Movahed, and Medhi Khrrazi, " Passive Worm and Malware Detection in Peer to Peer Netwroks," in *2010 IEEE/IFIP International Conference on Embedded and Ubiquitous Computing*, pp. 561-565.

[14] 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" in 2008 *Proceedings of the 41st Hawaii International Conference on System Sciences,* pp.1-10.

[15] D. Dumitriu, E. Knightly, A. Kuzmanovic, I. Stoica, and W. Zwaenepoel, "Denial of Service Resilience in Peer to Peer File Sharing Systems," in *Proceedings of the International Conference on Measurement and Modeling of Computer Systems,* 2005 pp. 38-49.

[16] Wei Yu, Corey Boyer, Sriram Chellappan, and Dong Xuan, "Peer to Peer System-Based Active Worm Attacks: Modeling and Analysis," in *Proceedings of IEEE International Conference on Communication ,* 2005, pp. 1-7.

[17] Sinan Hatahet, Abdelmadjid Bouabdallah, and Yacine Challal, "A New Worm Propagation Threat in BitTorrent: Modeling and Analysis," in *Proccedings of the International Multiconference on Computer Science and Information Technology,* 2008, pp.791-798.