International Journal of Engineering and Technical Research (IJETR)
ISSN: 2321-0869, Volume-2, Issue-12, December 2014

Detection of Malicious Behaviour in P2P Networks

G.Michael

***Abstract*— The malicious behavior and virus are always a biggest threat to a system. They are more vulnerable. When a malicious enters a system, it will immediately affect the entire system. Because of this, the system gets affected and also tends to work very slowly. The more danger is that, if it is a distributed system and when they are connected in peer to peer, the infected system will affect the other systems. To prevent this type of malicious attack, malicious behavior detection method is applied. This paper proposes a method in which one of the systems behaves as a protector system and the other system in the network acts a child system through this method, if any system is affected by malicious, the request is given to the protector system. The protector system sends an alert message to all other systems in the network. By applying patch framework, the protector system protects the child from the malicious attack**

***Index Terms*— integrity, malicious, p2p, patch, repudiation**.

I. INTRODUCTION

Peer to Peer (P2P) applications as we know them today inform of contribute the major chunk of the Internet traffic[1]. Being technically categorized as unstructured and structured, the P2P networks have diversified applications like file sharing, collaborations, process sharing (e.g. Distributed.net and Adhoc Networks) and distributed computing. Decentralized nature of P2P networks benefits through the properties like scalability, reliability, fault tolerance and load balancing, while in presence of no centralized authority, these networks are prone to many security threats in respect to breaches of confidentiality, integrity, authentication, access control and non-repudiation.

Over the years, malicious behaviors have emerged as a main source of trouble in P2P networks. Malicious s can be categorized mainly as scanning and non-scanning. Scanning malicious s always keep on probing addresses for new victims. They do waste time in probing unused addresses and may potentially have a high rate of failed connections[2]. Moreover, they do not blend with the normal P2P traffic. Due to the circumstances discussed, the non scanning malicious could sometimes be more dangerous than the scanning ones as they chose the vulnerable nodes through the neighbor lists and are hence more successful in acquiring precise and fast knowledge of their prey. We focus on malicious behaviors that hide themselves in popular P2P resources by embedding malicious code in executable files.

This strategy of selecting the targets has made malicious behaviors unpopular & less attended in history because most of the files shared in the early P2P networks were non-executable files like MP3 or some other media files.

However, more recent popular P2P systems, like Bit Torrent, Kazaa, eDonkey2000 & others provide the users with much easier access to executable files, and make malicious behaviors become a major threat yet again to the safety of the P2P networks [1]. The malicious behaviors operate in a purely epidemic manner to spread in the network. Firstly they embed themselves in the popular executable files in the P2P network and make a few copies in the sharing folder of the infected user. Once another user downloads the files and executes them, the malicious behaviors duplicate themselves and create a few new copies in the sharing folder, which increases their possibility of being downloaded by the other vulnerable users. Since the user can only be infected after the file is executed, the downloading of the malicious behaviors are, most of the time, treated as legitimate P2P network behavior and this actually makes it quite difficult to detect. Some researchers define the malicious behaviors as the ones that attach to files and propagate with user activities as viruses. We would use terms “malicious” and “viruses” alternatively for the malicious behaviors[2,3].

There have been lot of efforts to study propagation of P2P active malicious and defaces against them but a little has been done in regards to malicious behaviors[4]. Although such malicious s may propagate in a slower passion, the P2P networks are themselves the vehicles for fast malicious behavior propagation.. the P2P malicious propagate as a part of legitimate network activity and hence are difficult to detect than scanning malicious. Many studies are underway to analyzing the patterns of virus propagation in P2P networks to better understand malicious behavior. For this article, we mainly focused on unstructured file sharing P2P networks such as Kazaa and BitTorrent because most of the existing P2P malicious s targets these kinds of systems[5,6].

II. PROBLEM DEFINITION

In P2P networks have diversified applications like file sharing, collaborations, process sharing and distributed computing. Over the years, malicious s have emerged as a main source of trouble in P2P networks. In this paper we mainly focus on malicious behaviors that hide themselves in popular P2P resources by embedding malicious code in executable files. More recent popular P2P systems, like Bit Torrent, Kazaa, eDonkey2000 & others provide the users with much easier access to executable files, and make malicious behaviors become a major threat yet again to the safety of the P2P networks. Hence we propose a distributed framework for malicious behavior throttling in P2P networks and discuss its feasibility and efficiency keeping in view different design considerations.

Manuscript received on December 15, 2014.
G.Michael, Dept of CSE,Bharath University,Chennai,India

III. . System Analysis

A. Existing System

In the existing system, if a system is affected by malicious behavior it is cleared by using antivirus software. But if the operating system of a system gets affected by malicious behavior it is impossible to clear it. As a result the operating system has to be formatted and a new operating system only should be installed.

B. Proposed System

In the proposed system, one system acts as a guardian system and other systems in the network acts as child system. If any system in the network gets affected by malicious behavior, the request is given to the guardian system. The patch framework is given to the affected system by the guardian system and with the help the patch framework, the malicious in the affected system is cleared.

IV. Modules

A. Detection phase

As an integral part of the framework, the guardian node is equipped with observation software to identify any malicious behavior. The guardian node detects some malicious code, it would request the malicious definition database to look for the malicious definition and confirm it. Besides detection of attacks, locating the nodes responsible for vulnerability in the networks is important to make this activity rather efficient in identifying the threats

B. Analysis & confirmation of threat

The guardian node, by looking at the virus definitions confirms the threat; it would generate the alert to the entire P2P network. This alert generation would have different meanings for the peers and other guardian nodes in the network. The guardian nodes would get the patch ready and they could simply push the patch to other devices or wait for this patch to be pulled by the devices.

C. Patch selection

Selection of a proper patch from the patch reservoir is a key task when we look at the malicious throttling process. Prompt and proper patch availability could let the network recover quickly from the attack.

D. Patch propagation

A better strategy is required to be deployed to make the patch dissemination process fast to an extent that it could take over the malicious in the network[11]. Hence when the patch is ready, it could either be propagated straightaway to the peers or the guardian node would wait for the peers to download it in response to the alert

V. System Maintenance

The objectives of this maintenance work are to form certain that the system gets into work all time with none bug. Provision should be for environmental changes which can have an effect on the pc or software system. This can be referred to as the upkeep of the system. These days there's the speedy modification within the code world. Because of this speedy modification, the system ought to be capable of adapting these changes. In our project the method are often added without affecting different components of the system. Maintenance plays a significant role. The system prone to settle for any modification once its implementation. This method has been designed to favor all new changes. Doing this cannot have an effect on the system's performance or its accuracy.

VI. System Implementation

As an integral part of the framework, the guardian node is equipped with observation software to analyze the traffic patterns and to identify any malicious behavior. In our case the guardian node detects some malicious code, it would request the malicious definition database to look for the malicious definition and confirm it. Besides the content, the threat could also be detected through the behavior of the network or traffic suppose by an alarmingly increased number of connections. This activity may be traced by the firewalls and reported to the guardian nodes for the remedy.

Analysis & Confirmation of Threat In this phase, if the guardian node, by looking at the virus definitions confirms the threat, I would generate the alert to the entire P2P network. This alert generation would have different meanings for the peers and other guardian nodes in the network. The guardian nodes would get the patch ready and they could simply push the patch to other devices or wait for this patch to be pulled by the devices [9, 10].

Selection of a proper patch from the patch reservoir is a key task when we look at the malicious throttling process. Prompt and proper patch availability could let the network recover quickly from the attack. While the definitions for some malicious s are not there, techniques used to deploy to convert the malicious into anti-malicious[8]. Failure to which could require a human intervention prevents the further propagation of the virus from that infected machine.

Hence the addresses from which the malicious attack is being generated could be blocked for some duration to at least contain this epidemic while the recovery process would be underway in parallel. The alert messages could be made more effective if they also carry the information that could result in probing all the peers to block the traffic from some particular addresses. Doing so, these alerts could play a vital part in malicious containment process. Meanwhile the major recovery process through patch propagation and malicious scans on the individual peers is done.

A better strategy is required to be deployed to make the patch dissemination process fast to an extent that it could take over the malicious s in the network. As described by, the speed of epidemiological behavior of malicious s has always been a hard question. Hence when the patch is ready, it could

either be propagated straightaway to the peers or the guardian node would wait for the peers to download it in response to the alert.

An important phase in this regard is the communication between guardian nodes upon receiving the patch. When a guardian node detects a threat directly or through any peer, in an alert message, it is assumed that it would also announce the identity of the malicious so that the peers that may already have the patch could start taking care of the malicious[7] . The guardian nodes receiving the alert would make the patch available in their shared folders or even reactively flood the patch into the network.

VII. Implementation in Action

A. *Execution phase*

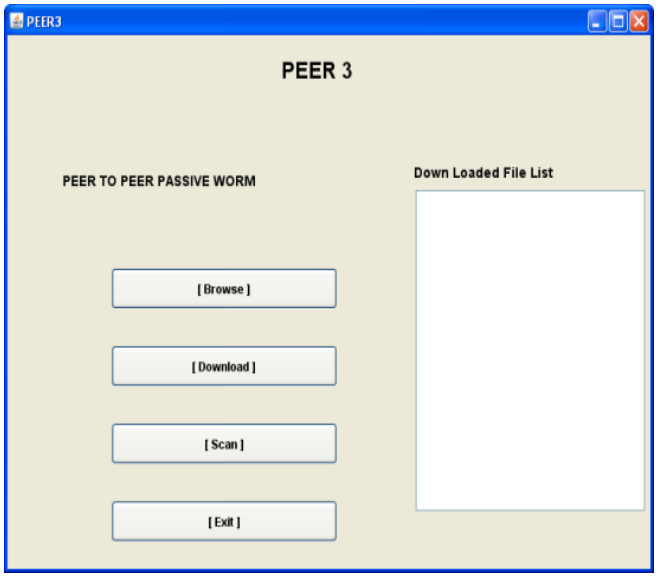

Fig 1.Execution Phase

B. *Detection Phase*

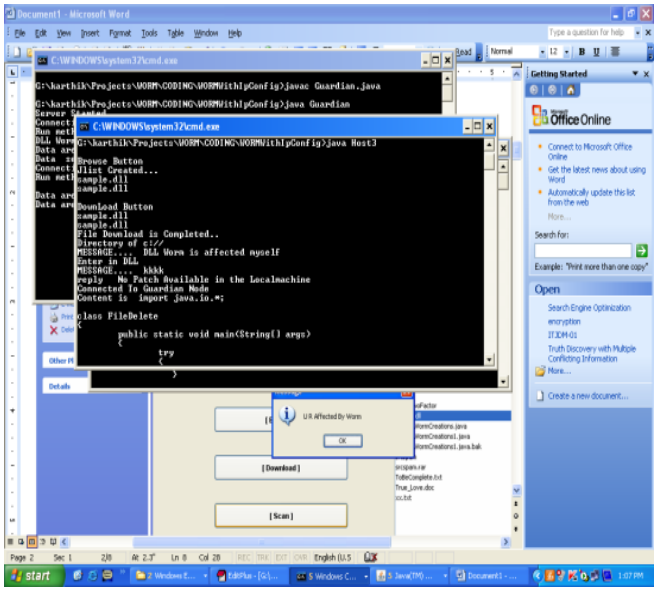

Fig 2. Peer is affected by DLL Malicious

C. *Analysis and Confirmation*

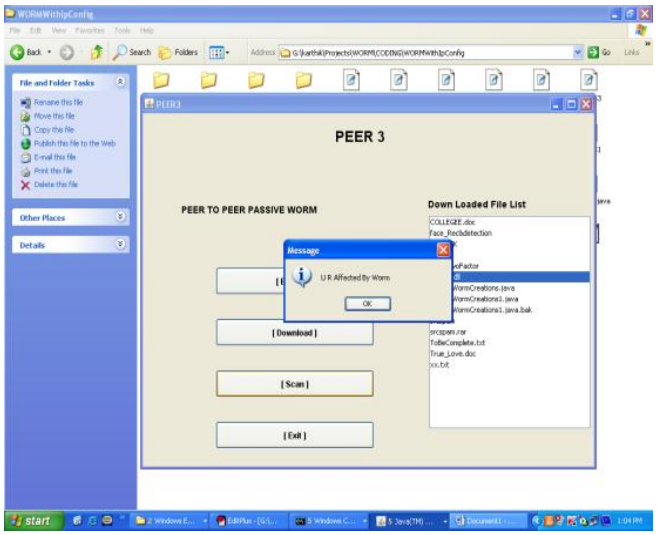

Fig 3. Analysis and confirmation

D. *Patch Selection*

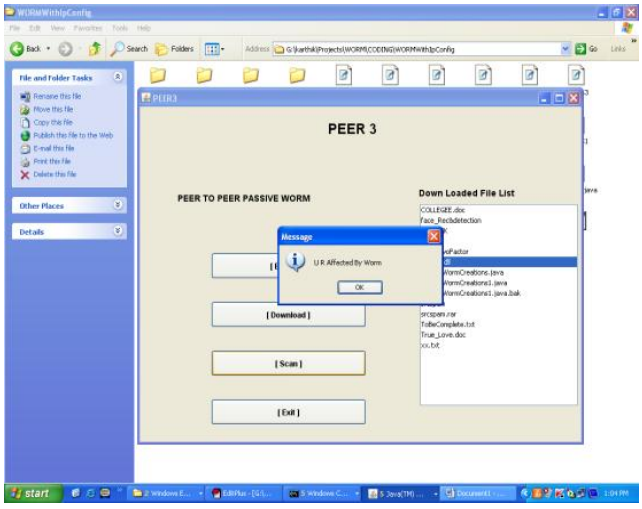

Fig 4. Patch Selection

E. *Patch Propagation*

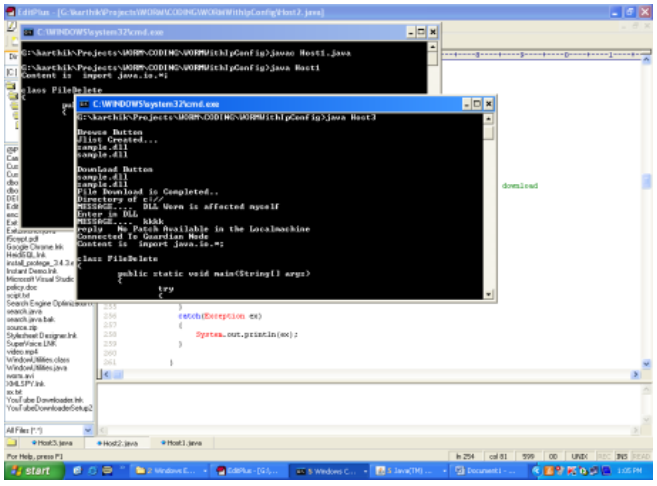

Fig 5. Patch Propagation

VIII. FUTURE ENHANCEMENT

The same project can be extended to detect many type of malicious behavior. Thus malicious detection could be very effective if done in a distributed manner for all type of malicious behavior .It will make the P2P network data transfer secure. The systems will be malicious free systems. We can use the same idea for different kinds of viruses and malicious behaviors.

IX. CONCULSION

After in brief analyzing the malicious and patch modeling work and a substantial review of malicious detection systems, we tend to conclude that malicious detection may be terribly effective if drained a distributed manner. We tend to argue that for the Scalable P2P networks, the distributed or technically hybrid detection mechanisms may prove even more effective than typical centralized detection. We tend to project a distributed threat detection and malicious behavior choking framework and deducing from the previous work in the field we could safely say that the performance of this framework would depend upon the prompt and intelligent threat detection, efficiency in sharing the threat information with the entities that matter, and a awfully robust recovery strategy.

REFERENCES

[1] Gnutella homepage,http://www.gnutella.com/

[2] Ramandeep kaur and jaswinder singh,"Towards security against malicious node attack in mobile Ad Hoic Network",IJARCSSE, vol,july 2013.

[3] S.Parameswari , G.Michael" Intrusion Detection System in MANET : A Survey " IJETR, Vol-2, April 2014.

[4] Lidong Zhou et al., "A First Look at Peer-to-Peer Worms: Threats and Defenses", Book Chapter, Peer-to-Peer Systems IV, Springer Publishing, 2005. [5] Peerbox Mobile, http://www.peerboxmobile.com

[6] Bo Zhan et al .,"Defense against passive in P2P Networks",Proceedings of Networking & Electronic Commerce Research Conference(NAEC 2008), 2008.

[7] Jamie Twycross, "Implementing and Testing a VirusThrottle" Proceedings of the 12th USENIX SecuritySymposium, Washington DC, USA, 2003. [8] Frank Castaneda et al., "WORM vs. WORM: Preliminary Study of an Active Counter-Attack Mechanism", Proceedings of WORM'04, Washington,2004.

[9] Guanling Chen et al., "Simulating Non-Scanning Worms on Peer-to-Peer Networks", Proceedings of INFOSCALE '06, Hong Kong, 2006.

[10] Zhiguang Qin , "Propagation Models of passive wormsin P2P Networks", IEEE International Conference onMachine Learning and Cybernetics (ICMLC), 2008.

[11] G. Micheal and A.R. Arunachalam "EAACK: Enhanced Adaptive Acknowledgment for MANET" Middle-East Journal of Scientific Research 19 (9), 2014.

G.Michael is currently working as Assistant Professor at Bharath University, Chennai, India. He has around 10 years of experience in the field of teaching. He has guided many PG and UG projects. He is currently pursuing his Ph.D in Computer Science & Engineering.