Development of Collaborative Defensible Services based on P2P Systems

Aarti.R, Aishwarya.M.J, Bhavana.B ,Vanaja Gokul
Department of Computer Science & Engineering,
Sri Venkateswara College of Engineering,
Sriperumbudur, Chennai, India

ABSTRACT
With the immense development of computer networks, the network security problems have also developed simultaneously, causing severe damage, thus reducing the efficiency of the networks and also the nodes in the networks. Protection provided at the nodes is not sufficient to completely protect the node from external attacks, hence a collaborative approach, is used to solve the network security problems. This paper explains the implementation of collaborative network security platform based on a Peer-to-Peer (*P2P*) network. An experiment is demonstrated where-in a peer can download a required file from a central server; these files may contain malicious software that could affect the peer. The services required to deal with the malicious software is provided by another server.

General Terms
Network Security

Keywords
Peer-to-Peer, Collaborative network security, malicious software.

1. INTRODUCTION
The growth of the network security problems has been so devastating thus causing huge damage. The conventional technique of providing protection at the node by including defensible services such as packets sniffing, filtering, logging, forwarding or by setting firewall or *IDS* (Intrusion Detection System) have proved to be insignificant with the growth of the network security problems. Thus a collaborative network security platform is developed where unavailable services can be obtained from other members of the network. For example, in a network if a node detects that it is a victim of *TCP/SYN* attack, and if the node does not have the services required to deal with the attack then the node can obtain the service from any other neighboring node by requesting for the service. This technique reduces the overhead at each node since it does not require each node to have all the required services.

The major threat to any network is caused by Distributed Denial of Service (*DDOS*) attacks and virus worm spreading. In Denial of Service attack a legitimate user is prevented from using the services of a resource they would normally expect to receive. In a Distributed Denial of Service attack, an attacker takes control of a large number of systems on the Internet and launches an attack on the victim. A computer worm is a malware computer program, which is self-replicating by nature and uses a computer network to send copies of itself to other systems on the network. The worm does not require any kind of user intervention while affecting the nodes on a network. Worms do not attach themselves to existing programs and the most important threat caused by worms is that they consume the network bandwidth. The types of computer worms fall into four major categories. They are Email worms, Instant Messaging Worms, Internet Worms, and File-sharing Network Worms.

Protocols for building heterogeneous unstructured P2P networks are proposed in literature [1].The protocol works in two parts, the joining process and the rebuilding process. In [1], it is proved that the topology structure of the peer-to-peer network depends heavily on the node heterogeneity. In any peer-to-peer system the overall system performance can be improved by fair resource sharing that is the peer's bandwidth contribution (amount of data contributed by the peer to its neighboring peers) is used to determine whether a peer can download data from the other peers. Since only limited resources are available at the peers to detect and respond to the threats posed on it, a collaborative approach is used in detecting intruders [2]. In this approach the information from across the internet is integrated to detect the intruder. In [2] worminators are used to extract relevant information and these information are encoded in Bloom filters.

The major problems to a peer-to-peer network is in the form of Distributed Denial of Service (*DDoS*) attacks [3], in which an attacker first sets up a *DDoS* attack network comprising of attacking hosts and a large number of agents. The resources of the internet and the hosts are exploited by sending useless packets either destined to the host or its router. Protocols for detection of *DDoS* attacks collaboratively have been developed in literature [4]. In [4] Change aggregation trees (CAT) are used to develop a distributed change-point detection (DCD). Using the developed architecture, abrupt traffic changes across many networks are detected at the earliest time.

The attacks caused by the worms are not less damaging; the worms are usually very rapid in causing attack hence fast worm containment is crucial for reducing the damage. In [5] a NetSheild CyberSpace defense system is developed. The

defense system restricts the spread of the worms and also protects a node against the *DDoS* attacks.

Collaborative techniques are used in Internet Protocol (*IP*) trace back mechanisms [6, 7] which traces the attack path back to the attacker. [6] Describes a trace back mechanism based on the probabilistic packet marking in the network. This method enables the victim to reconstruct the attack path without the intervention of the Internet Service Provider (*ISP*).

2. RELATED WORK AND MOTIVATION

The need for collaborative network security in the existing peer to peer systems has been presented in this section. The most attacks such as the *DDoS* and worm attack will also be briefly introduced here in order to understand its effects on the proposed network. The importance of a peer to peer network and its applications are also discussed here since our proposed work involves the development of the peer to peer network and providing security to the nodes comprised in it.

2.1 Peer-to-Peer (*P2P*) networks

P2P network, as mentioned in [8] is an equipotent network; every node in a P2P network has equivalent capabilities. Unlike client/server architecture where a central server manages the client nodes, the nodes in a P2P network share equivalent responsibilities. P2P networks are simple in structure but when they are subjected to heavy loads they do not offer the same performance. According to [9], P2P networks are of three types, they are

1. Purely decentralized P2P architectures: In this type, every node in the network acts as both the server and the client and the central server is absent. The nodes are often called as "servents" (SERVers+clieENTS). Original Gnutella and Freenet networks are examples of purely decentralized P2P architectures.

2. Purely centralized systems: The nodes in this type are similar to the purely decentralized nodes except for the fact that certain nodes in this system acquire a "more important" role than the rest of the nodes. These nodes are called "Super nodes" and they act as local central indexes for files shared by the local peers in the system. The methods of selecting the "supernodes" vary from system to system. Kazaa and Morpheus are examples.

3. Hybrid decentralized architectures: In this type of architecture, there is a central server that co-ordinate the interaction between peers and also maintains directories of shared files that are present in the corresponding peers. The central server facilitates the end–to-end communication between two peer clients and also identifies the peers that contain the required files for sharing purposes. Napster is an example for this architecture.

2.2 Distributed Denial of Service (*DDoS*) attack

As defined in [10], a Distributed Denial of Service attack is one in which an attacker takes control of a large number of systems on the Internet and launches an attack on the target machine that acts as the victim. Victims of a DDoS attack include both the end targeted system that is affected by the attacker and also those systems that are controlled by the attacker and used in performing the DDoS attack.

2.2.1 TYPES OF *DDoS* ATTACKS

In a **Direct *DDoS*** attack, the attacker implants the zombie software on a number of hosts, throughout the Internet. The two levels of zombies involved in this attack are master zombies and slave zombies. The hosts of both the levels of zombies are affected with malicious code and the attacker triggers the master level zombies. The master level zombies, in turn trigger the slave zombies which perform the attack on the victim. It is difficult to trace the attack back to its source because of the use of two levels of zombies. In **reflector *DDoS* attack**, the slave zombies send packets to uninfected machines called reflectors. The packets contain the victim's IP address as the source IP address and these packets require a certain kind of response. The uninfected machines send a response back to the victim machine and results in lot of damages. The reflector attack is worse than the direct attack as it introduces a large amount of traffic in the network.

2.2.2 *DDoS* COUNTER MEASURES

1. Measures to prevent and preempt the attack (before the attack)
2. Measures to detect and filter the attack (during the attack)
3. Tracing the source and identification (during and after the attack)

2.3 Worm attack

A computer worm, as stated in [11], is a program that replicates itself and sends copies of itself from one node to another node, across several network connections. As explained by Xiang Fan etal (2010), worms are different from viruses as worms do not require a host program to run.

2.3.1 TYPES OF WORMS

Email Worms

Email worms are those which use a node's email client to spread it. The worm sends a link in that email, which when clicked causes infection or it sends an attachment that affects the computer upon opening.

Internet Worms

Internet worms, with the help of an infected machine scan the Internet for vulnerable machine. When the worm locates a vulnerable machine it immediately affects the machine. These worms are completely autonomous programs.

File-sharing Networks Worms

When file sharing takes place, the file sharers **usually** do not know what **they** are downloading**.** The worm copies itself into a shared folder using an anonymous name. When another user downloads the file from the network the worm gets copied into the user system and starts replicating. Phatbot is a type worm that affected millions of computers in 2004.

Instant Message and Chat Room Worms

The worm will use the contact list of the user's chat-room profile or instant-message program to send links to various websites on the Internet**.**

3. PROPOSED WORK

This section presents the modules included such as browse, download, scan and the proposed algorithms for scanning the system for the worm attacks. The theoretical analysis helps to evaluate the performance of the proposed network.

3.1 Modules used

1. Initially a peer-to-peer network is formed which consists of a few peers, a central server which consists of files that can be downloaded by the peers in the network. Another server which consists of the services to defend the worms is also included in the network.

2. Every peer is provided with the browse and downloads facility through which it communicates with the file sharing server.

3. If in case a peer is affected by a worm, the detection module facilitates the identification of the worm and its effects on the peer.

4. On detection of the worm attack on the peer, the peer checks if it has the service to defend the worm. When the peer is deprived of the service then it requests the main server for the required service.

5. The server, on having the required service to defend the worm, responds back to the peer with the requested service.

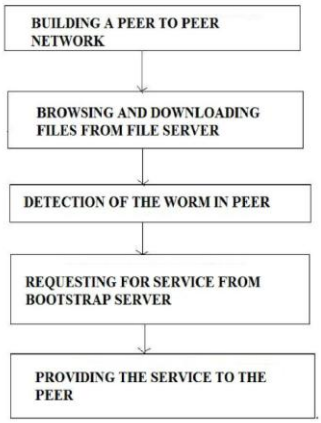

Figure 1: Overview of Modules

3.1.1 BUILDING A *P2P* NETWORK

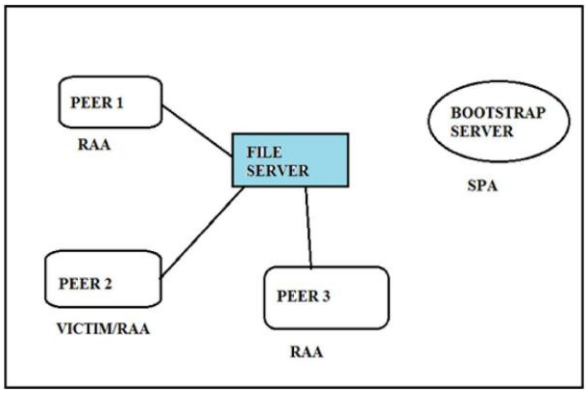

Figure 2: Basic Network

The basic network as shown in Figure 2, includes components such as,

1. SERVICE-PROVIDER
2. FILE SERVER
3. P2P AGENT NODES

SERVICE-PROVIDER: It is initialized first when the system is starting and its main function is to provide service to the worm infected peer. The service provider is the main component of the network since it acts as a repository of the services required to act against the worm attack. The service provider is used only when a peer detects that it has been infected and when it does have the service required to prevent itself.

FILE SERVER: Maintains the list of all the files that can be browsed, downloaded and shared by the peers. File sharing is the prime application of any peer-to-peer network on the internet, allowing users to easily contribute, search and obtain content. A File server is used when the P2P network is used in the hybrid decentralized mode where-in is a central server facilitating the interaction between peers by maintaining directories of the shared files stored on the respective PCs of registered users to the network, in the form of meta-data.

P2P AGENT NODES: The network consists of the agent nodes which constitute the basic entity of the network. The agent nodes are provided with the basic functionalities of the network such as browse and download to interact with the server. It is also provided with certain advanced features such as "scan" which enables the interaction with the main server.
The agent nodes are classified into 3 categories,

1. Service Passive Agent (SPA): The agent nodes are designed to provide the service when requested by any other neighboring peer. In this network the Service-provider acts as the Service Passive Agent.

2. Request Active Agent (RAA): These agent nodes obtain benefits from the network by utilizing the network resources. The nodes can request their neighboring nodes for their required file. RAAs can only request other agent to provide services but they cannot publish new services.

3. Publish Active Agent (PAA): A node in the network makes available it's newly created service by publishing it. Such nodes are known as Publish Active agents. These published services can be used by the Request Active Agents.

3.1.2 BROWSING AND DOWNLOADING FILE FROM SERVER

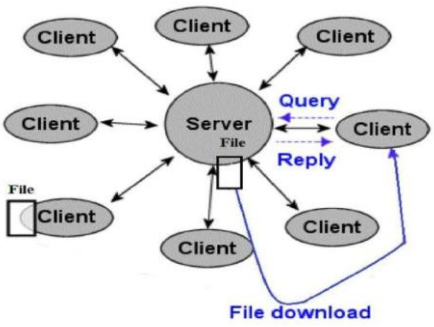

Figure 3: Browse and Download

Browse and download as shown in the Figure 3, are the basic features provided to every node in the network. Every peer in the network can browse for a particular file present in the server and download the file. The file which is downloaded gets saved in the "downloads" folder present in the "C" drive. The communication between the client and the server is two ways i.e. files can be moved from the server to the client and from the client to the server. Hence files from the peer can be moved to the server, this in turn provides communication between the neighboring peers via the main file sharing server.

BROWSE: The browse option allows the peer to view the list of files that are present on the file sharing server. A client-server socket is created where-in the peer acts as the client and the file-sharing server acts as the server. The server socket is kept open and it listens to the incoming client requests to establish a connection. The client sends a connection request to the server via the server IP address and the server port number. The server accepts the client's connection request. The input stream and the output streams are used to form a list of files that are present on the server. The list thus formed is displayed on the user-interface (UI) created for the client i.e. the peer in the network. The browse option also allows the peer to select a file on the server. The selected file can be downloaded from the server by the peer. The browse option also creates another list i.e. the list of all the files that is present on the local peer, this list is known as the local list. The local list is used in the case of the "**SEND**" button which facilitates the transfer of the files from the peer to the main file-sharing server.

DOWNLOAD: The download button allows the peer to move the file present in the server to the peer. The files that are downloaded are saved in the "download" folder of the "C" drive present in the local peer. Using the browse button the peer selects the file that it wants to download. If the file is a normal text file then it is downloaded normally by the peer. If the file is a worm file then the worm performs the actions based on its characteristics. Two types of worm files are included. They are the MSIL and the shutdown worm. The "dll" files are included. When the MSIL worm file is downloaded, the worm file causes the creation of a number of empty folders, one inside another. This type of attack causes the utilization of the resources of the local peer. When the shutdown worm file is downloaded, the worm file causes the peer to shutdown. On selecting the download option a timer for the shutdown is started. The peer's system gets shutdown if no action is taken to abort the shutdown process. The dll files are normally downloaded in the peer. This type of file is included to illustrate the concept that most of the anti-viruses don't allow the dll files to enter into the system.

3.1.3 SCANNING AND DETECTION OF WORM

Every peer is provided with a facility to scan the "C" drive which contains the files that the peer downloads. Two kinds of worms that may infect the system are considered, which includes the 'shutdown worm' that causes the system to shut down and the other worm which creates folders inside folders in a continuous pattern. The "scan" option allows the peer to check all its folders and to detect if it has been affected by the worm or if any dll file is present in the system. On detecting the worm attack the peer is notified the type of the attack. The peer checks itself if it has the mechanism to act against the worm, if it does, the peer takes action against the attack either by deleting it or by aborting the action of the worm. If the service is not available in the peer, it requests for the service to the neighboring components of the network.

ALGORITHM FOR SCAN

Input: Files present in download folder of respective peer.

Output: Action based on the attack.

1. Check the download folder of peer to determine if any malicious software or if any abnormalities caused by worm is found.

2. **If** (attack == folder worm)
    For each folder present in the "downloads" of the peer, the subfolders are determined.
        If (folder name == subfolder name) then
            MSIL WORM is detected
        End If
    End For
End If

3**. Else if** (attack==dll files)
    For each folder in the downloads of the peer, the subfolders are determined. All files in the folders and subfolders are checked for their extension.
        If (.dll extension found) then
            DLL file is detected
        End If
    End For
End If

4**. Else** (attack==shutdown worm)

For each folder in the downloads of the peer, the subfolders are determined. All files in the folders and subfolders are checked for the shutdown code.

If found then

SHUTDOWN WORM is detected

End If

End for

End If

Once the peer detects that it has been infected by a particular kind of worm, it checks if it already has the required services. If the peer does not have the services then the peer requests the bootstrap server for the service to handle the worm. This module brings about the action of the Request Active Agent (RAA) and the Service Passive Agent (SPA).

3.1.4 REQUESTING THE BOOTSTRAP SERVER FOR SERVICE

Once the peer detects that it has been infected by a particular kind of worm, it checks if it already has the required services. If the peer does not have the services then the peer requests the bootstrap server for the service to handle the worm. This module brings about the action of the Request Active Agent (RAA) and the Service Passive Agent (SPA).

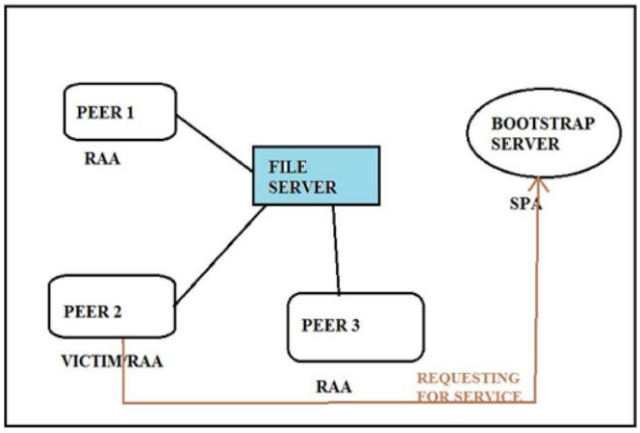

Figure 4: Requesting the bootstrap server for service

3.1.5 PROVIDING THE SERVICE TO THE VICTIM

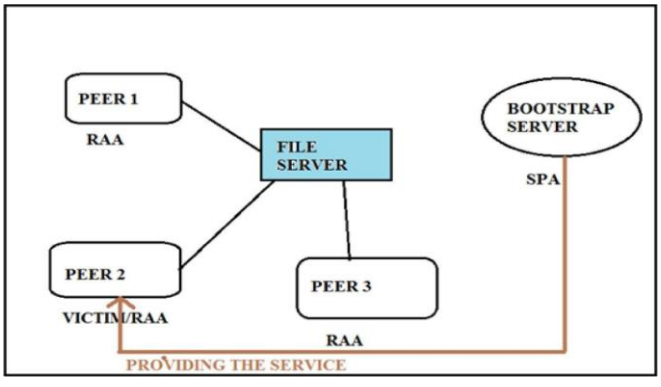

Figure 5: Providing the service to the victim

When the peer requests for service, the bootstrap server provides the peer with patch file, as in Figure, that deletes the worm and saves the system from malicious attack.

4. RESULTS AND ANALYSIS

This section analyses the comparison between a conventional anti-virus and our proposed work theoretically.

4.1 Experimental Results and Proof

The Files that are downloaded by the peer from the server are downloaded at the location "C:/Download". The folder now contains the malicious "Dll" file and the MSIL worm. As shown in the Figure 6, a scan run by the antivirus Kaspersky shows the result that the two malicious files remain undetected and are not deleted from the download folder.

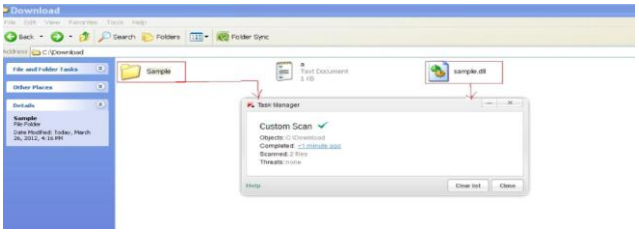

Figure 6: Scan by antivirus-Kaspersky

As shown in the Figure 7, the anti-viruses are incapable of detecting the MSIL worm i.e. the worm that creates folder inside folder. The only solution as demonstrated by the forum is to format the entire system and re-install the OS which is quite a bit of a time-consuming process.

The service is requested by the victim from the service provider and the MSIL worm is deleted from the system.

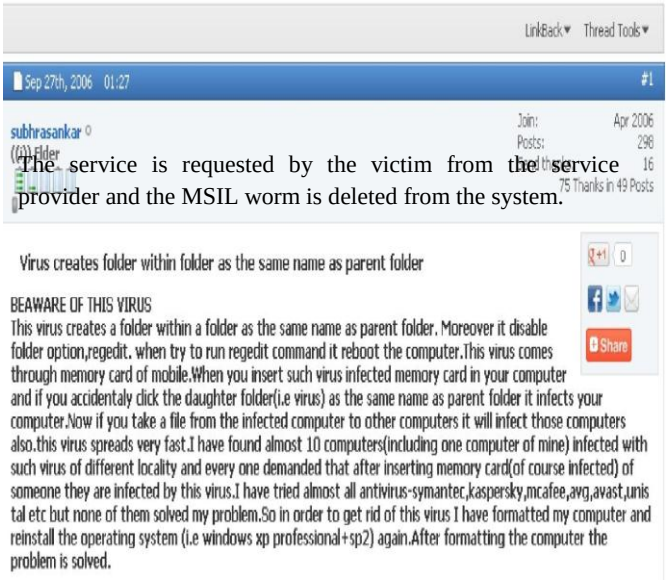

Figure 9: MSIL worm deleted from system

Similarly the malicious 'dll' file, when infected into the

system, is detected and removed from the system.

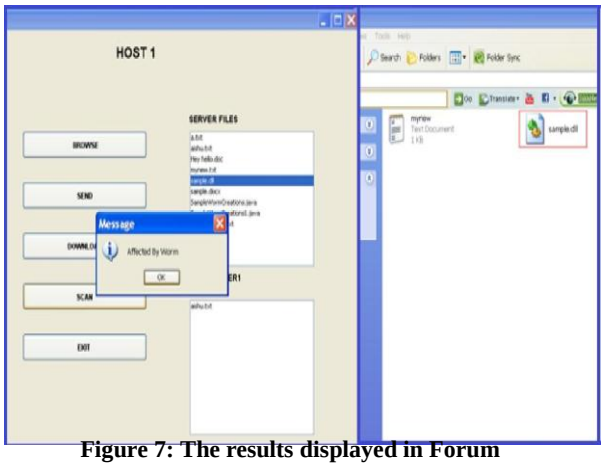

Figure 7: The results displayed in Forum

On the other hand our proposed work provides a technique by which the MSIL worm and the 'dll' files are detected and removed from the system thereby ensuring security against worm attacks. As shown in the figure below our work detects the attack of MSIL worm.

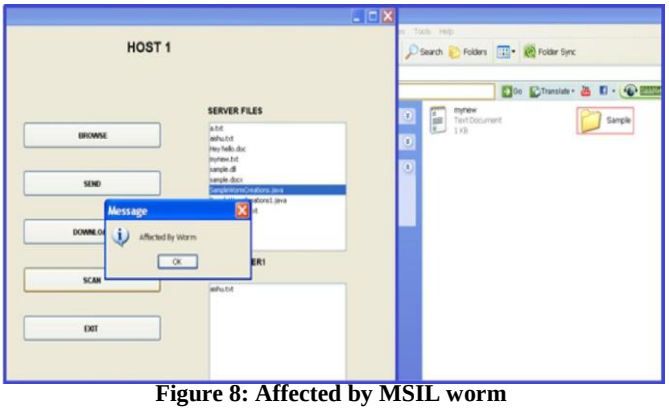

Figure 8: Affected by MSIL worm

5. CONCLUSION

The network security threats (here worm attacks) are thus solved by using a collaborative method. In a network, it is difficult to meet and solve the security issues using anti-virus software. Hence the method of collaborative technique is adopted which provides service to the victim by acquiring the service form its neighboring peers. In the future, the work can be extended by providing a facility of IP trace back. By this method of IP Trace back, a detailed description about the peers, their list of files are maintained in a database. The central server tracks the activities of the peers. Once a malicious activity is identified and rectified, the server can track the peer that provided the malicious file and forbid the acceptance of any kind of file from that peer. Another future work can be providing read, write privilege to the files that the entire peer holds.

6. REFERENCES

[1] Kin-Wah Kwong and Danny H. K. Tsang,"Building Heterogenous Peer-to-Peer Networks: Protocol and Analysis", IEEE/ACM Transactions On Networking, Vol. 16, No. 2, April 2008

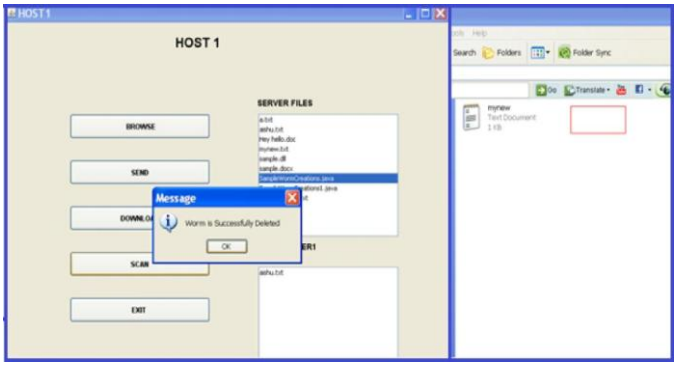

Figure 10 : Dll file detected

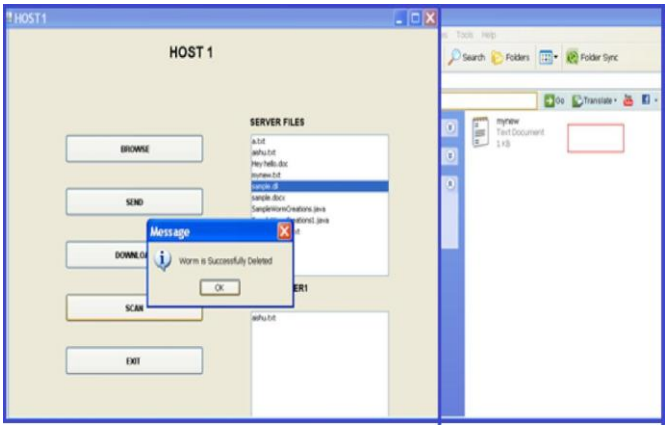

Figure 11: Dll file deleted from the system

[2] Michael E. Locasto, Janak J. Parekh, Angelos D. Keromytis and Salvatore J. Stolfo,"TowardsCollaborative Security and P2P Intrusion Detection", Information Assurance Workshop, 2005. IAW '05. Proceedings from the Sixth Annual IEEE SMC.

[3] Rocky K. C. Chang,"Defending against Flooding-Based Distributed Denial-of-Service Attacks: A Tutorial", Communications Magazine, IEEE Vol. 40,pp. 42-51.

[4] Yu Chen, Member, Kai Hwang and Wei-Shinn Ku,"Collaborative Detection of DDoS Attacks over Multiple Network Domains" IEEE Transactions On Parallel and Distributed Systems, Vol. 18, No. 12, December 2007,pp. 1649-1662.

[5] Min Cai, Kai Hwang, Yu-Kwong Kwok,Shanshan Song, And Yu Chen," Collaborative Internet Worm Containment", Security & Privacy, IEEE,Vol. 3,pp.25-33.

[6] Stefan Savage, David Wetherall and Anna Karlin and Tom Anderson," Network Support for IP Traceback", IEEE/ACM Transactions On Networking, Vol. 9, No. 3, June 2001,pp. 226-237.

[7] Andrey Belenky and Nirwan Ansari," On IP Traceback", New Jersey Institute of Technology

[8] Peer to Peer network:http://www.doubleeaglesserviceinc.com

[9] Stephanos Androutsellis-Theotokis "A Survey of Peer-to-Peer File Sharing Technologies",pp.8.

[10] William Stallings," Cryptography and Network Security Principles and Practices",Fourth Edition,pp.615-619.

[11] Worm and its types: http://www.ehow.com.