A Lightweight and Non-Path-Based Mutual Anonymity Protocol for Peer- to-Peer Systems

[1]B Krishnaveni Reddy, [2]Md Sarfraz Ahmed
[1,2]Dept. of CSE, NIZAM Institute of Engg & Technology, Desh Mukhi, Hyderabad, AP, India

Abstract
An anonymous P2P communication system is a peer-to-peer distributed application in which the nodes or participants are anonymous or [pseudonymous. Anonymity of participants is usually achieved by special routing overlay networks that hide the physical location of each node from other participants. Existing anonymity approaches are mainly path based: peers have to pre-construct an anonymous path before transmission. The overhead of maintaining and updating such paths is significantly high. In highly dynamic P2P systems, when a chosen peer leaves, the whole path fails. Unfortunately, such a failure is often difficult to be known by the initiator. Therefore, a "blindly-assigned" path is very unreliable, and users have to frequently probe the path and retransmit messages. To address the above issues, we propose a non-path-based anonymous P2P protocol called Rumor Riding (RR).It is a lightweight and non-path-based mutual anonymity protocol for decentralized P2P systems. Employing a random walk mechanism, RR takes advantage of lower overhead by mainly using the symmetric cryptographic algorithm.

Keywords
Mutual Anonymity, Non-Path-Based, Random Walk, Peer-To-Peer

I. Introduction

Traditional network (internet/intranet) applications are client-server based, where many clients communicate with a common shared server for application services. Examples of such application services include e-mail servers, web servers, and file servers. These centralized servers have two fundamental problems: scalability and resiliency. In the present state of the Internet, millions of users may be using the same server simultaneously. It is difficult to host a server for millions of users and remain online continuously.
An alternative to the client-server architecture is the peer-to-peer model. Every client in a peer-to-peer network is also a server. The coordination and discovery issues of these decentralized networks are central. To better understand these issues the protocols of two popular peer-to-peer applications Napster and Gnutella are closely examined. Each protocol presents a distinct approach to the coordination of information exchange between peers, and discovery of the information contained on those peers.
Napster hosts act as clients as well as servers for the exchange of music files. A host first joins the network by connecting to a central server known as a broker. Once connected, the host passes information on all the music files it serves to the broker. This information is known as metadata. The broker stores a database of the metadata; this metadata contains the information of all the hosts currently logged into the broker. In Figure 1 there are six computers logged into the broker.
In addition to searching and sharing music, Napster also provides peer-to-peer messaging, chat rooms, and user hot lists. Peer-to-peer messaging allows one peer to talk another peer. Chat rooms allow groups of users to share information. A message posted to a chat room is seen by all users connected to the chat room. Hot lists contain a list of popular peers with whom a client has been in contact. Peers can add each other to their own hot lists. This hot list will provide information on a peer's metadata, as well as when the peer is online. The broker performs all the coordination of these extra features.
An anonymous P2P communication system is a peer-to-peer distributed application in which the nodes or participants are anonymous or [pseudonymous. Anonymity of participants is usually achieved by special routing overlay networks that hide the physical location of each node from other participants.
Interest in anonymous P2P systems has increased in recent years for many reasons, ranging from the desire to share files without revealing one's network identity and risking litigation to distrust in governments, concerns over mass surveillance and data retention, and lawsuits against bloggers

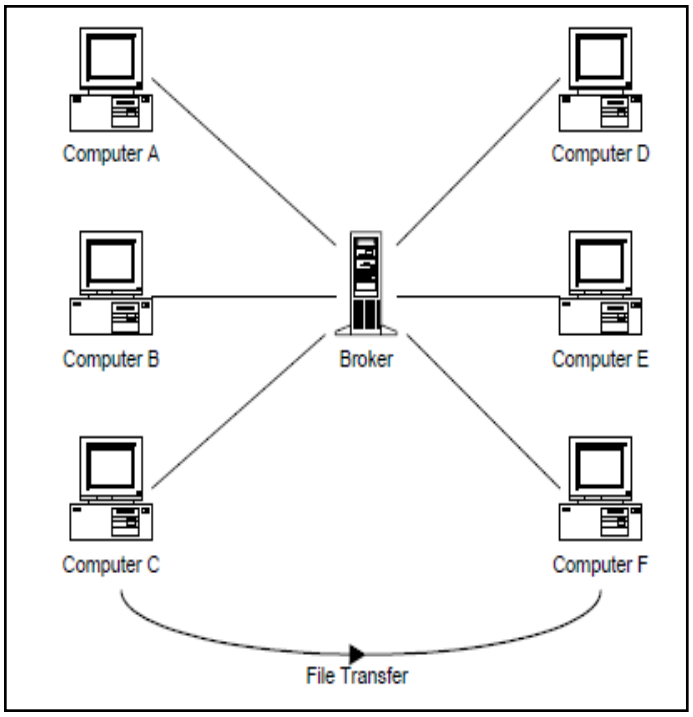

Fig. 1: A Napster Network the Broker Coordinates Music File Sharing Among Peers

While anonymous P2P systems may support the protection of unpopular speech, they may also protect illegal activities, such as fraud , libel, the exchange of illegal pornography, the unauthorized copying of copyrighted works or the planning of criminal activities. Critics of anonymous P2P systems hold that these disadvantages outweigh the advantages offered by such systems, and that other communication channels are already sufficient for unpopular speecpular speech.

A. Functioning of Anonymous P2P

Some of the networks commonly referred to as "anonymous P2P" are truly anonymous, in the sense that network nodes carry no identifiers. Others are actually instead of being identified by their IP Addresses; nodes are identified by pseudonyms such as cryptographic keys. For example, each node in the mute network

has an overlay address that is derived from its public.This overlay address functions as a pseudonym for the node, allowing messages to be addressed to it. In , on the other hand, messages are routed using keys that identify specific pieces of data rather than specific nodes; the nodes themselves are anonymous.

The term anonymous is used to describe both kinds of network because it is difficult—if not impossible—to determine whether a node that sends a message originated the message or is simply forwarding it on behalf of another node. Every node in an anonymous P2P network acts as a universal sender and universal receiver to maintain anonymity. If a node was only a receiver and did not send, then neighbouring nodes would know that the information it was requesting was for itself only.

In distributed and decentralized P2P environments, the individual users cannot rely on a trusted and centralized authority, for example, a Certificate Authority (CA) center, for protecting their privacy. Without such trustworthy entities, the P2P users have to hide their identities and behaviors by themselves. Hence, the requirement for anonymity has become increasingly critical for both content requesters and providers.

In highly dynamic P2P systems, when a chosen peer leaves, the whole path fails. Unfortunately, such a failure is often difficult to be known by the initiator. Therefore, a "blindly-assigned" path is very unreliable, and users have to frequently probe the path and retransmit messages.

II. Related Works

Many efforts have been made to acquire anonymity in P2P systems. Earlier work falls into two categories: anonymous publishing and anonymous communication.

Anonymous communication includes initiator and responder anonymity and anonymous data transferring. In MorphMix [4] and Onion [3], initiator can determine an anonymous path in advance to hide some identification information. Instead of building a path by an initiator, Crowds [9] and Hordes [6] let middle nodes select the next hop on the path and deploy multicast trees by themselves. Freedom [7], Tarzan and Tor [1] are implemented based on Onion Routing to provide anonymous services.

Some studies, such as Peer-to-Peer Personal PrivacyProtocol (P5) [6], Anonymous Peer-to-Peer File Sharing (APFS) [3], and Shortcut-responding Protocol , proposed to provide mutual anonymity in P2P systems. The basic idea of P5 is to let all participants in the channel send fixed length encrypted packets at a fixed rate as if all participants are in a logic ring. This protocol introduces noise packets to maintain a fixed communication rate to confuse the traffic analyzers or attackers. Meanwhile, refines the broadcast channel size and builds a hierarchy overlaid spanning tree to keep the communication scalable. P5 users can join different groups based on the tradeoff between anonymity degree and the communication bandwidth and reliability. As P5 assumes that every initiator knows the public keys of all possible responders, it cannot be directly employed in P2P networks.

APFS is designed for a decentralized system, like Gnutella. Some coordinator nodes act as a superior peer and maintain a list of all the peering nodes. Some peers in these lists volunteer to issue queries for others. When an initiator queries a coordinator for available servers, the coordinator returns a list of current servers. The initiator then contacts some servers to send the request and receive replies from the servers. It finally sends the request and receives the reply with the help of the tail node of the matching responder. All communications of this framework are based on the onion path to guarantee the anonymity and hence no centralized authority exists in this system

III. Proposed System

We propose a non-path-based anonymous P2P protocol called Rumor Riding (RR). In RR, we first let an initiator encrypt the query message with a symmetric key, and then send the key and the cipher text to different neighbors. The key and the cipher texts take random walks separately in the system, where each walk is called a rumor. Once a key rumor and a cipher rumor meet at some peer, the peer is able to recover the original query message and act as an agent to issue the query for the initiator. We call the agent peer as sower in this paper. The similar idea is also employed during the query response, confirm, and file delivery processes. Thus, the rumors serve as the primitives of this protocol to achieve mutual anonymity and meet the design objectives.

In RR, anonymous paths are automatically constructed via the rumors' random walks. Neither the initiator nor the responder needs to be concerned with path construction and maintenance. Extending the scope of anonymous servents from a small clique of nodes to the entire P2P network, RR significantly increases the anonymity degree of a system.

RR employs a symmetric cryptographic algorithm to achieve anonymity, which significantly reduces the cryptographic overhead for the initiator, the responder, and the middle nodes. In addition, as initiating peers have no requirement on extra information for constructing paths, the risk of information leakage, caused by links that are used for peers to request the IP addresses of anonymous proxies, is eliminated.

In Anonymity Peer-to-Peer (P2P) networks, many systems try to mask the identities Rumor Riding consists of five major components: Rumor Generation and Recovery, Query Issuance, Query Response, Query Confirm, and File Delivery.

A. Rumor Generation and Recovery

RR employs the AES algorithm to encrypt original messages. The key size is 128-bit. To determine whether a pair of cipher and key rumors hit, we employ a Cyclic Redundancy Check (CRC) function to attach a CRC value, CRC(M), to the message M. For received key rumors and cipher rumors, the sower S uses AES to recover a message M' and the checksum CRC (M'). It then performs the CRC function to the recovered M' and compares the result. If they match, the sower S is aware that it has successfully recovered a message M.

B. Query Issuance

When an initiator I wishes to issue an anonymous query, it first generates the query content q, and a public key K. Node I then uses an AES cryptographic algorithm to encrypt q into a cipher text C with a symmetric key K. It organizes the key K and the cipher text C into two query rumors, qK and qC. In Gnutella, each packet is labeled with a Descriptor ID. RR also uses the descriptors to identify rumors. Thus, two random number strings, IDqK and IDqC, are used to label the two rumors. After generation, I forward the rumor messages to two randomly chosen neighbors.

C. Query Response

When a receiving node the query has a copy of the desired file, it becomes a responder R. To respond to the query, R encrypts the plain text of the response message r, using the initiator's public key K. It encrypts key using AES, where KR is the public key generated by R, and encloses the cipher text and the key into two response rumors, rK and rC. They are then assigned with IDrK

and IDrC, respectively

D. Query Confirm

In the query confirm phase, I uses the responder's public key to encrypt the confirm message c. It then encrypts (KþR,IDrK, IDrC, IPSb) and obtains two confirm rumors, cK and cC, which take random walks in the system. Note that two confirm rumors are marked with new descriptors: IDcK and IDcC. We assume that cK and cC collide in a new sower S'a . We denote their paths from I to S'a by lcK and lcC . When S'a recovers the IP address of Sb from cK and cC, it directly contacts Sb to forward cK and cC attached with IDrK and IDrC via a TCP line.

E. File Delivery

It employs a digital envelope technique to encrypt the file into cipher CF. Instead of including CF into the rumor generation, R encrypts (IDcK, IDcC,IPS0a) to generate the data cipher rumor and the data key rumor, and attaches the digital envelop payload to the data cipher rumor. The large data cipher rumor and the small data key rumor first take random walks to meet each other at a sower.

The tradeoff is that for each query, RR requires a number of sowers randomly distributed in the entire system, but too many sowers will lead to unacceptable overhead. In order to guarantee the diversity of the sowers, a simple and lightweight method is needed for estimating O(log n), where n is the size of the P2P overlay. Also, to reduce the unnecessary overhead, peers need to observe the diversity of sampling sowers to adjust the TTL value of rumors. The adaptive TTL determination of RR comprises two phases: (a) setting initial TTL value, and (b) adaptively adjusting TTL.

In phase (a), during the P2P Bootstrapping process, each fresh peer retrieves a list of neighbors from the bootstrapping server. The fresh node can set the mean value of those in the responses as its initial TTL value of rumors.

In phase (b), RR employs random walk based schemes similar to the ones used in [5]. A peer can periodically insert several pairs of "sampling" rumors into the network. Those rumors are set with the peer's current TTL setting and its IP address. The initiator sets a timer for each rumor. Any sower of such a pair of sampling rumors records the retained TTL of two rumors and then directly sends the TTL information as well as a timestamp to the initiator through a TCP connection. During the process, we do not need to consider the anonymity. The initiator observes the ID (IP address) of the responding sower and the distribution of the reported TTL. Also, the mechanism facilitates to balance the tradeoff between the length of anonymous `paths and the latencies of query, response, or file delivery by adaptively tuning the setting of TTL.

IV. Conclusion

Rumor Riding is a lightweight and non-path-based mutual anonymity protocol for P2P systems, Rumor Riding (RR). Using a random walk concept, RR gives key rumors and cipher rumors separately, and expects that they meet in some random peers. Sower is a peer where key rumor and cipher rumor meet and decryption can be done and send to responder. Rumor Riding (RR) provides a high degree of anonymity and outperforms existing approaches in terms of reducing the traffic overhead and processing latency. It eliminates to collect large number of IP addresses when sending a data.

References

[1] L. Xiao, Z. Xu, X. Zhang,"Low-Cost and Reliable Mutual Anonymity Protocols in Peer-to-Peer Networks", IEEE Trans. Parallel and Distributed Systems, Vol. 14, No. 9, pp. 829-840, Sept. 2003.

[2] R. Sherwood, B. Bhattacharjee, A. Srinivasan,"P5: A Protocol for Scalable Anonymous Communication", Proc. IEEE Symp. Security and Privacy, pp. 58-70, 2002.

[3] D. Chaum,"Untraceable Electronic Mail Return Addresses, and Digital Pseudonyms", Comm. ACM, Vol. 24, No. 2, pp. 84-90, Feb.1981.

[4] R. Rivest, A. Shamir, L. Adleman,"A Method for Obtaining Digital Signatures and Public-Key Cryptosystems", Comm. ACM, Vol. 21, No. 2, pp. 120-126, 1978

. [5] M.K. Wright, M. Adler, B.N. Levine, C. Shields,"The Predecessor Attack: An Analysis of a Threat to Anonymous\ Communications Systems", ACM Trans. Information and System Security, Vol. 7, No. 4, pp. 489-522, Nov. 2004.

[6] D. Goldschlag, M. Reed, P. Syverson,"Onion Routing", Comm. ACM, Vol. 42, No. 2, p. 39, 1999.

[7] R. Dingledine, N. Mathewson, P. Syverson,"Tor: The Second-Generation Onion Router", Proc. 13th USENIX Security Symp., pp. 303-320, 2004.

[8] V. Scarlata, B.N. Levine, C. Shields,"Responder Anonymity and Anonymous Peer-to-Peer File Sharing", Proc. IEEE Int'l Conf. Network Protocols (ICNP), pp. 272-280, Nov. 2001

[9] Q. Lv, P. Cao, E. Cohen, K. Li, S. Shenker,"Search and Replication in Unstructured Peer-to-Peer Networks", Proc. 16th ACM Int'l Conf. Supercomputing, pp. 84-95, 2002..

[10] C. Gkantsidis, M. Mihail, A. Saberi,"Random Walks in Peer-to-Peer Networks", Proc. IEEE INFOCOM, 2004

B Krishnaveni Reddy received her B.Tech degree from MRITS Engg College,Udayagiri, Nellore (D.t) in 2006. The M.TECH degree in CSE from NIZAM Institute of Engg & Technology, Deshmukhi, Near Ramoji film city, Hyderabad, India, in 2012.At present, She is engaged with "Anonymzing Unstructured Peer-to-Peer Systems and its Securities".

Mohd Sarfaraz Ahmed has received his B.Tech. (IT) and M.Tech (S/W ENGG.) from JNTUH, AP, India. He is presently working as an Asst.Professor in the Department of Computer Science Engineering, Nizam Institute Of Engineering and Technology, Deshmukhi, Nalgonda, A.P, India. His area of interests includes Web Technologies, Artificial Intelligence, Software Requirements and Estimation, DataMining, Peer-to-PeerNetworks.