NON PATH-BASED MUTUAL ANONYMITY APPROACH FOR DECENTRALIZED AND UNSTRUCTURED DUMMY TRAFFIC FOR PEER-TO-PEER SYSTEMS

Abdul Majid[1], Amit Chaturvadi[2]
[1]Ph.D, Research Scholar, Bhagwant University,Ajmer,Rajasthan,India.
[2]HOD of MCA , Government Engineering College Ajmer, Rajasthan
Email: Dr.majid.wahab@gmail.com[1],amit0581@gmail.com[2]

Abstract

Anonymous peer-to-peer systems frequently incurred additional expenses in order to do efficient transfer. Each and every Last node tries to copy themselves of which the information receive and they need all of their users for privacy considerations. Existing models of anonymous approaches are mainly non- path-based with unstructured: peers does not need any anonymous path before transmission of any files and queries. We propose Rumor Riding (RR), a approach for decentralized nodes in P2P systems with a dummy traffic and speed up the queries. Applying random walk mechanism for lower overhead systems is mainly by using the symmetric cryptographic algorithm along with ELGamal algorithm in this approach. Dummy traffic generation will be used to hide the actual Cipher text and cipher data to confuse the attackers and accelerating query speed limited but in the new format speed increased genuinely. Dummy traffic generates traffic in a similar message with cipher and key to confuse the attacker. Evaluation is done by using anonymity approaches RSA, ElGamal and AES. we can show the how much effective by simulations by trace driven this protocol is very effective and efficient than previous protocols and this is illustrated with the experimental and analytical results.

Index Terms: Dummy traffic; Query Speed; non path based; random walk; peer-to-peer;

I. INTRODUCTION

PEER-TO-PEER (P2P) networks, such as Napster, Gnutella, and Bit Torrent, have become essential media for information dissemination and sharing over the Internet. When thought about the privacy along with rapid development of P2P systems. In the present scenario of the distributed and decentralized P2P environment, each and every user cannot depend on a trusted and centralized authority, example of this a certificate Authority (CA) centre, for protecting their privacy. There is no trustworthy entities, the P2P user hide their identities and their behaviors. It becomes critical for both requesters and providers for their requirement of anonymity.

A number of methods [1], [2], [3], [4] have been proposed to provide anonymity. Many of them achieve anonymous message delivery through non traceable paths including both multiple proxies and middle agent peers. The approaches mentioned above are also called as path-based approaches and users setup anonymous path before transmission. Many of the paths are Layer-Encrypted data structure. Path based protocol had strong anonymity and this path is to be pre-constructed that requires to collect a large number of IP addresses and public keys by the initiator. The initiator performs asymmetric key based cryptographic encryption, for example RSA[5], when wrapping layer-encrypted packets. The peer collection and content encryption will be high cost. The users want to establish a long anonymous path and update this

path at time to time to defend from the attackers. In dynamic P2P systems, if a single node leave the peer, whole path will be failed and this type of failure occur means, it become very difficult to identify by the initiator. This ―blindly-assigned‖ path is very unreliable and user has to absorb regularly the path before retransmit the messages. To address the above issue, we propose anonymous P2P protocol called Rumor Riding on non-path based, first of all initiator initiate the encrypted message query with the help of asymmetric key and send along with cipher text to different neighbors. These two (Key and Cipher) take random walks separately in the system, and this separate walk is called a Rumor. The random walks by the cipher and key will meet at some peers together and this peer has authority to recover the original query message. Sower word is used in this paper for the agent peer.

The similar idea we are using during the response, i.e. File Delivery process and Confirmation query. The primitive is to achieve mutual anonymity protocol and that going to meet the design objective. It will reduce overhead of initiator, responder and middle nodes, the peer does not has the additional information and it is not able to build that path or threat of information leakage and request to IP addresses of anonymity but proxies delete that link.

II. LITERATURE SURVEY

A. Path-based anonymous delivery and anonymous multicast:

Since Chaum [4] pioneered the concept of anonymity, many approaches have been proposed to support anonymous communications. The approaches fall into two categories: path-based anonymous delivery and anonymous multicast. Onion Routing [7], as well as its second generation, Tor[8], are the most popular path-based protocols providing initiator anonymity based on a layered encryption method. They mainly focus on the IP layer rather than the application level. APFS [9] adopts the initiator anonymous protocols like Onion Routing to provide responder anonymity in P2P systems. Shortcut Protocol [2] provides P2P mutual anonymity with a reduced response delay. Crowds [1] introduces a random forwarding mechanism to intermediate nodes. When receiving a packet, a peer has two options: forwarding the packet to a randomly chosen peer or directly sending it to the destination peer.

B. P^5 Employ a virtual tree to construct anonymous broadcasting groups.

P^5[3] is based on the anonymous multicast. To make the broadcasting scalable, P^5 employs a virtual tree to construct anonymous broadcasting groups. P^5 also utilizes a noise mechanism, which enables peers within a group to send packets at a fixed rate for concealing the initiator‘s ID. Anonymous multicast-based approaches, however, are not suitable for P2P systems as the initiator has to know the destination node‘s ID. Since random walk is the building block of our protocol design, we provide a brief overview of random walk approaches. Lv et al. [10] propose a multiple random walk based query algorithm to replace the flooding method for reducing the network traffic in Gnutella-like systems. In[11], Academic et al. propose an algorithm, which works well in power-law graphs.

C. Ramdom based search protocol for unstructred P2P systems.

The work attempts to make the search scalable and reduce the network traffic. Gkantsidis et al [12] study the properties of random walk in depth via statistical methods and reveal some factors for improving the system performance. Bisnik et al. [13] provide a mathematical model to analyze the performance of random walk, and develop an adaptive algorithm to reduce the search overhead. Bhattacharjee et al. present a random based search protocol for unstructured P2P systems . Sybil guard employs random routes in a social network trust topology to defend against Sybil attacks. All of these prior studies strongly support the workability and efficiency of random walk in P2P systems.

III. EXISTING SYSTEM:

Napster, Guutella and Bit Torrent become essential media for information sharing dissemination over the Internet, these network are P2P network. The plain-text query message and direct-downloading behavior encountered problem traced on non-anonymous P2P users. the current P2P applications for both content requests and providers the requirement of anonymity increasing. The delivery of message via non-traceable paths with several anonymous proxies or middle agent peers by Most or not all of them. Before transmission users usually need

to construct anonymous paths in these approaches, these approaches known as path-based approaches. Layer encrypted data structure is path for most of the cases. the path-based protocols provide strong anonymity but an anonymous path require to constructed a pre-constructed for all node in the path cooperate to forward data to a receiver. The initiator before transmission the data to the receiver, the data is pre-wrapped in layer-encryption packet which will be peeled off along the path to the receiver. Path-based protocols provide strong anonymity, they have the following problems.

A. Drawback:

- The large number of IP addresses and public keys from other peers in advance require by the user in the Pre-construction of paths. The preparation packet and collection of information Both incur high costs.
- The anonymous paths periodically update middle node by Initiators.
- The invariable paths might otherwise become increasingly vulnerable under the analysis of traffic attackers and other similar attack.

IV. PROPOSED SYSTEM:

In the proposed RR system anonymous paths are automatically constructed through the rumors random walks. the construction and maintenance of paths should not worry by initiator and responder. key rumors and cipher rumors are the two important point of this protocol to achieve mutual anonymity and meet the design objectives. the trace-drive simulation is used to demonstrating the effectiveness of this design. all most results through analytical and experimental show that RR is more efficient that existing protocols. dummy traffic which introduce duplicate traffic between the topology to confuse the attacker.

A. Advantages:

- For the P2P systems the RR generating lightweight and non-path-based anonymity protocol.
- Uses a symmetric cryptographic algorithm and ElGamal algorithm to replace the asymmetric to reduce the cryptographic overhead and make the protocol more practical.
- Introducing new approach called Dummy traffic.
- The Dummy traffic generates duplicate cipher and key rumor to confuse the attackers.
- These duplicate cipher rumor and key rumor take random walk in the constructed topology. it does not interrupt the original rumors random walk.
- The main goal of the dummy traffic to introduce the duplicate traffic and to confuse the attackers.

V. RUMOR GENERATION AND RECOVERY:

Here, AES algorithm as well as ElGamal cryptosystem is used to encrypt the original message. The size of key is 128-bits to take action on the pair of Cipher and key Rumor hits, and then apply a Cyclic Redundancy Check (CRC) and used CRC value CRC(M) to the Message M. The Sower, S_a ,who receive the Key and Cipher Rumor to Decrypt by AES algorithm and ElGamal Cryptosystem to recover the Message M^I and the checksum CRC (M^I), then apply the CRC function to recover M^I and compare the result with CRC (M^I), if the result matched, the Sower, S_a , aware that the Message M Successfully Recovered.

A. Query Issuance:

The Initiator ‗I' want to send the anonymous query, it then create content ‗q' having request for some services example for some file.

The Initiator generates two pairs of asymmetric keys:

- Private Key K_I^-
- Public key K_I^+ by using Cramer Soup cryptosystem

The query ‗q' made a requested service along with Initiator public key K_I+. The Initiator want the number of feedbacks, then tag the request before sending. By using the ElGamal cryptosystem the node ‗I' encrypt the query ‗q' and its public key K_I^+ into cipher text pair (C_1,C_2). To decrypt the two cipher text pair (C_1,C_2), the initiator prepare public value ‗p'

and private key ‗K' as a key pair (p,x) into two query rumors, ‗q_k' and ‗q_c'. IDqk and IDqc are the two random number strings used as two rumors labels. The rumor message forwarded to two randomly selected neighbors. After generating ‗I', they together start their own random walk (Cipher rumor and key rumor).

RR needs to maintain the Local Cache for each node to store the received rumors. It performed all the procedure for cipher in all cached when it receive the rumor. The plaintext and CRC value are matched and then it decrypt the rumor recovered successfully. Either they matched or not, transitional node decrease the TTL value of received rumor by one and kept temporary evidence consisting the ID of rumor in local cache and ahead to a randomly selected neighbor. To confuse the adversary, this process is used and no ways to suspect that the current node is a rumor. When the TTL value of rumor become to zero the process is going up, and this process will be same when cipher rumor query received. There is no particular sequence, if rumors query pair reach to not exacting node further node would be recover the unique ‗q'. The main issue of this procedure is that they select one pair of rumor, now required rumors and initial TTL values carefully along with key and cipher meets. Before send out in to the Hops, first the RR initialized non-zero positive number w(1<w<127). The undersized number in between 9 and 12 is enough to confuse to attacker who could try to determine the location of the initiator. For Example the length of rumor walks is L and L + W is TTL value.

B. Sower

The Sower S_a randomly select a trusted agent called subset S_t to send the request. The request received from Initiator, have tagging will dictate the no of subset agent. It will select one rumor pair, initial TTL values and number of rumors along with key and cipher meet. Sower agent change the value of TTL and Hop count and then prepare a query send to get feedback from a subset trusted agent S_t. Sower S_a now attach the original query message qc plus ‗I' and public key K_I^+ called Ciphertext pair (C_1,C_2). Q_k (p,x) and tag request with a label IDsk and IDsk respectively plus its address in a plaintext. This operation is useful to avoid invisible flooding.

C. Query Response

Responders in a group can hide their identities by having all message sent to same address by using AEs or ElGamal cryptosystem. These keys will be not able to use forever. For doing this, a group of Recipients agree on one value of ‗p'. To calculate the values of ‗x', each entity chooses their various secret generator ‗g' corresponding to their private key. The individual select a private key and publishes multiple public keys using it. The responder will not have good support of Sower and initiator with a public key.

Subset of target node S_t has copy of the file requested received from receiving node and it willing to respond to the node, called the responder. Query message will copy and release the message to continue its random walk. Before sending, the RR create a non-zero positive number w (1<w<127) in the hops rumors field. The undersized number between 7 and 10, or 8 and 11, or 9 and 12 these values are enough to confuse the adversaries who try to identify the location the initiator. The marked C_{r1} and C_{r2} are two confirm numbers which will back L_{sk} and L_{sc} path to Sower S_a. This Sower will flood in group of two ciphertexts. To decrypt the message, correct responder only able to do this because it possesses the corresponding private key.

Responder will calculate the values of ‗x' corresponding to its private key using secret generator ‗g'. Preparation for Response ‗R' as follows:

Start with two responders:

- r_k having public key pair (p,g,x)
- r_c having encrypted response with initiator public key K_I+

It sends rk and rc back to Sower Agent S_a through TCP connection. The Responder will be never revealing its identity due to power of the generator ‗g'. The noise packet will be introduced to passive correlation attack. When Sower agent got the reply r_k and r_c , then it will send to the original peers of ‗q_c' and ‗q_k'. The L_{qk} and L_{qc} arrive at I, when two rumors response create it. To confuse the adversary the initiator ―I' copy received r_k and r_c and add

few hop count before sending them out randomly to two different recipients. Using private key K_I+ , the two responder rumors ‗I' to get original response message.

D. Query Confirm:

The Initiator ‗I' uses responders public key to encrypt the confirm message ‗C' forming two Ciphertexts (C_1,C_2). The initiator will initialize a positive number would be non zero W(1<W<127) in rumor hops field to confuse the adversaries. Again the small number between 9.

E. File Delivery

The responder using private key and ‗R' will encrypt the file with initiator public key to get data cipher rumor divided in to two (C1, C2) and (e,v) and Labeled Dc1 and DC2 when the confirm message is Received. There are number of Algorithm is there for example AES as well as Cramer Soup Cryptosystem. Then the initiator keys are generated using above method. So that the integrating check will be perform for decrypt. Through the TCP connection the ‗R' will send two cipher to Sower S_a. following the reversed paths of L_{ck} an L_{cc} the cipher finally arrive ‗I'. the initiator ‗I' using the private key to Receiver desired file after confirming the integrity of the file. If the file size is bigger than responder its split in to multiple segment.

VI. Generation of Dummies Traffic

A. *Deciding Pool Algorithm*

The Deciding Pool Algorithm determined the performance of pool mix (delay and anonymity). Chaum's design mix flushes all the messages it contains.

Afterward concept of pool was added to the mix, original mix to keeping the fixed number of messages.

The proposals of mixes keep a number of messages in the pool.

- Then mixes that kept a variable number of messages were design.
- It enhances the anonymity by extending the anonymity set size to. Potentially and Infinite number of users.
- Probability distributions obtained by an attacker trying to trace a message will not be uniform for all recipients of messages.
- Parameters that taken into account.
- Number of message kept in the pool. Number of message sent.
- The mix represented at the time of flushing.
- Shows the percentage of messages contained in the mix that are sent in a round.
- As a function P(n)of the total number of messages in the mix.

B. Thread pool pattern

- The number of threads created to perform number of tasks that are organized in queue. Executed results from tasks that also placed in queue, otherwise task may return no result.
- As the thread completed its task request, next task from queue will be allotted until all tasks must be completed.
- Thread terminated or sleep until new task provided.
- The number of threads can be dynamic based on number of waiting tasks.
- java pool memory used to store buffered queue message like Dummies.

C. Uses of Algorithm

When to create or destroy thread, this will have impact on the overall performance.

- Create too many threads and resource are wasted
- Time also wasted creating any unused threads
  Destroy too many threads more time will spent later creating again.
- Creating threads too slowly resut in poor node performance.
- Destroying threads too slowly may starve other processes of resources.

The first question that arises when designing a dummy traffic policy is whether the dummies generated should depend on the incoming traffic or not. Generating dummies depending on the traffic load may make a more efficient use of the resources, but this dependency can be exploited by an active attacker to maximize the effectiveness of his attack by generating his own messages in such a way that he minimizes the number of dummies generated by the mix.

Therefore, dummy traffic policies that are independent from the traffic load seem to be

more secure. One of the issues that needs to be decided is the average number of dummies we want to generate (for pool mixes we will choose an average number of dummies per round, while in continuous mixes we will generate dummies per fixed time unit). These dummies can be generated following a deterministic or random distribution. Random distributions increase the uncertainty of the attacker, specially when combined with binomial mixes, as pointed out in Continuous mixes.

These mixes may generate a certain number of dummies every period of time, selecting their delay (amount of time they are kept in the mix from their generation until the moment in which they are sent) from a random distribution. This is the approach followed by Reliable, one of the mixes that composes the Mixmaster network. Other dummy policies may be explored, for example, the mix could keep always one dummy inside, and generate a new one (with its corresponding delay) when the dummy is sent. Another policy would be that the mix decides every certain amount of time on whether to generate a dummy or not.

D. Pool mixes

The design of dummy policies for pool mixes implies making decisions on the following issues: The dependency on the traffic load. The average number of dummies generated per round. The distribution followed to select the number of dummies in a particular round (binomial, uniform, geometrical, etc.).

E. Insertion in the Pool

With this technique, the mix inserts the dummies it generates for a round in the pool. These dummies are treated as real messages by the mix after being placed in the pool.

F. Insertion at the Output

If the mix is to insert the dummies at the output, then it adds the dummies to the batch of real messages taken from the pool.

The mix does not modify the number of messages contained in the pool. The advantages and disadvantages of these two dummy insertion options have been discussed in. Here, we summarize the conclusions presented in inserting the dummies in the pool provide less anonymity and less delays that inserting them at the output. When dummies are inserted at the output, binomial mixes with a random dummy policy offer more protection against the (n-1) attack than deterministic mixes. Inserting dummies in the pool protects deterministic mixes better than inserting them at the output when an n-1 attack is deployed.

G. Route Length and Selection of Path

Dummy messages, just like real messages, travel in the mix network going through a number of mixes. The route length of the dummy determines the number of mixes a dummy is going through. Regarding this issue, we should decide on the average number of mixes in the path of the dummy and on the distribution of this route length. Random distributions increase the uncertainty of the attacker with respect to a deterministic distribution (i.e., fixed number of mixes in the path) when the attacker wants to find out whether a message is a dummy or not. Normally the path of a dummy is selected randomly among the mixes of the network. The last mix in the path of the dummy can be the mix that actually generated it, preventing this way that corrupted mixes can help the attacker (when they are the last in the path of the dummy) providing the information on which messages were dummies. Note that intermediate mixes (i.e., except for the first and last in the path of the dummy) cannot distinguish dummy messages from real messages. Note that, in order to increase the anonymity provided by the mix, the mix should maximize the number of possible destinations for every message, meaning that the mix should check if it is sending messages to all the possible neighbors. If it is not, then it should generate some extra dummies to send to those mixes. This way an attacker wanting to trace a message will have to follow more possible paths.

VII CONCLUSIONS

A new approach is employed a lightweight and non-path-based mutual anonymity protocol for P2P systems, Rumor Riding (RR).Employing a random walk concept, RR issues key rumors and cipher rumors separately, and expects that they meet in some random peers. The results of trace-driven

simulations and simple implementations show that RR provides a high degree of anonymity and out performs existing approaches in terms of reducing the traffic overhead and processing systems, such as grid systems and ad-hoc networks. In this paper we have presented a thorough analysis of the parameters of mixes and dummy traffic policies, distinguishing between continuous and pool mixes. We have discussed the advantages and disadvantages of different design options. We have introduced anonymity metrics and mix network topologies

REFERENCES

[1] M.K. Reiter and A.D. Rubin, "Crowds: Anonymity for WebTransactions," ACM Trans. Information and System Security,vol. 1, no. 1, pp. 66-92, Nov. 1998.

[2] L. Xiao, Z. Xu, and X. Zhang, "Low-Cost and Reliable MutualAnonymity Protocols in Peer-to-Peer Networks," IEEE Trans. Parallel and Distributed Systems, vol. 14, no.9, pp. 829-840,Sept. 2003.

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

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

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

[6] M.K. Wright, M. Adler, B.N. Levine, and 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.

[7] D. Goldschlag, M. Reed, and P. Syverson, "Onion Routing,"Comm. ACM, vol. 42, no. 2, p. 39, 1999.

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

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

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

[11] L.A.Adamic, R.M.Lukose, A.R.Puniyani, and B.A. Huberman, "Search in Power-Law Networks," Physical Rev. E., vol. 64,p. 046135, 2001.

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

[13] N. Bisnik and A. Abouzeid, "Modeling and Analysis of Random Walk Search Algorithms in P2P Networks," Proc. Second in p2p Systems, 2005.

[14] R.Morselli, B.Bhattacharjee, A. Srinivasan, and M.A.Marsh,"Efficient Lookup on Unstructured Topologies," Proc. ACM Symp. Principles of Distributed Computing, 2005.

[15] H.Yu, M.Kaminsky, P.B.Gibbons, and A. Flaxman, "SybilGuard:Defending against Sybil Attacks via Social Networks," IEEE/ACM Trans. Networking, vol. 16, no. 3, pp. 576-589, June 2008.

[17] K. Sripanidkulchai, "The Popularity of Gnutella Queries and Its Implications on Scalability," http://www- 2.cs.cmu.edu/ ~kunwadee/research/p2p/gnutella.html, 2009.

[18] J. Han, Y. Liu, and J. Wang, "Rumor Riding: Anonymizing Unstructured Peer-to-Peer Systems," technical report, http://www.cse.ust.hk/~jasonhan/RR-TR.pdf, 2009.

[19] S. Jiang, L. Guo, X. Zhang, and H. Wang, "LightFlood: MinimizingRedundant Messages and Maximizing Scope of Peer-to-Peer Search," IEEE Trans. Parallel and Distributed Systems, vol. 19, no. 5, pp. 601-614, May 2008.

[20] Abraham and D. Malkhi, "Probabilistic Quorums for Dynamic Systems,"Proc. Int'l Symp. Distributed Computing, 2003.

[21] D. Stutzbach, R. Rejaie, and S. Sen, "Characterizing Unstructured Overlay Topologies in Modern P2P

File-SharingSystems," IEEE/ ACM Trans. Networking, vol. 16, no. 2, pp. 267-280, Apr. 2008.

[22]A. Medina, A. Lakhina, I. Matta, andJ. Byers, "BRITE: An Approach to Universal Topology Generation," Proc. Int'l Workshop Modeling, Analysis and Simulation of Computer and Telecomm. Systems (MASCOTS), 2001.

[23]S. Saroiu, P. Gummadi, and S. Gribble, "A Measurement Study of Peer-to-Peer File Sharing Systems," Proc. Multimedia Computing and Networking (MMCN) Conf., 2002.

[24]S. Sen and J. Wang, "Analyzing Peer-to Peer Traffic across Large Networks," IEEE/ACM Trans. Networking, vol. 12, no. 2, pp. 219-232, Apr. 2004.