A- Gossip: Secure Routing in Unstructured Peer to Peer Network

Anubhava Srivastava[1] Dharmendra Kumar[2]
[1,2]Department of Computer Science Engineering
[1,2]United College of Engineering and Research Allahabad, India

Abstract— Over the Internet today, computing and communi-cations environments are signicantly more complex and chaotic than classical distributed systems, lacking any centralized organization or hierarchical control. There has been much in-terest in emerging Peer-to-Peer (P2P) network overlays because they provide a good substrate for creating large-scale data sharing, content distribution and application-level multicast applications. These P2P networks try to provide a long list of features such as: selection of nearby peers, redundant storage, efficient search/location of data items, data permanence or guarantees, hierarchical naming, trust and authentication, and anonymity. P2P networks potentially offer an efficient routing architecture that is self-organizing, massively scalable, and robust in the wide-area, combining fault tolerance, load balancing and explicit notion of locality. P2P network have mainly two types, they are structured and unstructured. The structured P2P overlays have defined topology, such as ring topology in Chord [2]. These protocols are maintained by distributed hash table (DHT). On the other hand, unstructured overlays are changing their overlay and have no structure. Random peer sample (RPS) mechanism is efficient to maintain these overlays. The RPS mechanism focuses on exchanging a random set of neighbours for updating the overlay connections, such as in Cyclon [22] or updating while searching a file, such as in FreeNet[3]. Such types of overlays are badly affected by few attacker nodes, which are described in TooLate [18], SPSS [17], etc. We proposed a A-Gossip security protocol. The comparative performance will be done by Peersim simulator [23]. Further, we will analyze the security issue in unstructured P2P routing in FreeNet [21] After that analysis, we will apply our proposed A-Gossip protocol on FreeNet [21] for enhancing its security.

Key words: Peer-to-Peer networks, distributed hash table, Un-structured P2P network; Gossip-based Protocols; Hub Attack; Paralyse

I. INTRODUCTION

The rapid expansion of Internet in terms of speed,size and profusion of network application have been deployed in last few years indicate change toward traditional client-server model to distributed model, The network researcher apprehend that using centralized servers is not good for administering and managing very large scale distributed systems. As a result, considerable effort has been made in designing peer-to-peer (P2P) overlay networks. The main idea behind designing Peer to Peer system is communal collaboration among peers because in peer to peer network each peer provide both services(client and server) , same Peers some time works as client and some time works as server, By this effective nature peers can collectively perform large scale tasks in simple and scalable way. A Peer to Peer system are maintain with Resource Sharing, Networked, Decentralization, Autonomy,Self Organization , Scalable, Stability and Heterogeneity property. P2P system used to handle various complex services such as file sharing system, data containing digital formats etc.

P2P system are designed in such a way that churn (joining and leaving of peers) does not affect the entire network other way we can say these are highly dynamic and symmetric net-works . It refers to network communication without servers and allows host to communicate directly with other peers. P2P system architecture is designed for Internet architecture on application layer implemented as an abstract overlay . Overlay management is key point in peer to peer system , according to overlay management peer to peer system are categorized in two way Structured Peer to Peer system and Unstructured P2P system Structured P2P system has a predefined fix connection between nodes, parameter of connection of nodes is their Id . Structured P2P overlays maintain by distributed hash tables (DHT). DHTs logically organize peers in a well-defined structured and perform an exhaustive and exact search in very large-scale systems [1].Distributed Hash Table (DHT) provide a lookup service same as Hash Ta-ble(key, value),any participating peer can retrieve value with the help of their key Some application based on Structured Peer to Peer are chord [2], pastry [3], can [4]. Chord protocol design in 2001 for Structured P2P network . Goal of algorithm is quickly locate the node by their particular key, for joining and leaving of nodes network stabilization protocol is executed periodically in background of each nodes. Pastry was proposed in 2001 developed at Microsoft Research, Ltd. Similar to Chord It focus that lookup queries routed efficiently and object easily located in Structured P2P overlay In Pastry data item and nodes having unique 128- bit Ids routing Table is maintained by periodically exchanging Keep-alive message between adjacent nodes Unstructured P2P system unlike Structured P2P system don't have predefined connection between nodes in these system peers are linked either randomly or probabilistically connected based on some proximity metric some applica-tion based on unstructured P2P system are Napster [5] , Gnutella [6] , Kazza [7] etc. Napster [5] protocol created on application level using client server architecture over P2P architecture But difference in this server peer maintain only meta information about all peer. A peer query send to server first when server gets query it matches query to its index , server inform query peer about all that peers who hold file related to query now query directly communicate to any of these peers and gets result for his query. But problem in Napster [5] due to high churn some time query file are not accessible. To overcome drawback of Napster , Gnutella Protocol is discussed , Gnutella is pure decentralized P2P file system each peer have both capability server as well as client there is no central system as in client server system. Gnutella protocol is highly fault tolerant means peers can join or leave the system at any time. Gnutella protocol working with five descriptor that are ping, pong , query , query-hit, push. but

problem in Gnutella protocol that it effected from Slashdot Effect and message overhead is high compare to other previous existing protocol . For avoiding message overhead in network Gossip-based protocols [10], [11] is discussed. Gossip based protocol come for efficient communication in unstructured P2P system .These protocol easily create a dynamic structure with the help of few neighbour [8], [9] such neighbour are called view. Communication is starting in periodic cycle in which node A randomly select a node B from his view , A push subset of descriptor with a fresh descriptor of itself , B also communicate to A same way now A and B updates own view by message received from each other, old information is progressively replaced by new information . The compact set of view provides the scope of whole overlay [15], [16]. Natural reduction (churn) han-dled by gossip protocol efficiently But reduction that form forcefully due to malicious peer(attacker) are not efficiently managed by Gossip based protocol.In these protocols, nodes periodically communicate and exchange data and/or infor-mation about their view node. Structurally different protocol are discussed for exchanging views between peers. Cyclone [22] is an example of such type of protocol , In cyclone peer A select a neighbour B from his view which having maximum age and share their information,selected peers B also send their information and updated own information which received by A. In such a way both peer update their information . protocol prevents the duplicate of links and highly focuses that in degree of peers must be unchanged. But big problem with such protocol that malicious peer easily fool the network and come in network as important node, later create a hub with cooperating malicious peers after some time all malicious node leave the system by which network partition occur such types of attack is known as Hub attack, where attacker want a leading position in network topology . Attacker create a hub of malicious peer and later leave overlay that cause havoc to the system . Hence security is an important issue while gossiping in Unstructured peer to peer network.

II. HUB ATTACK IN TRADITIONAL GOSSIP PROTOCOLS

Gossip based protocol form a random graph with his view . View is update and exchanged by gossiping with neighbour peers, so after each gossiping cycle a random graph is formed. Gossip based protocol are basically used in unstructured P2P network because these protocol are highly scalable , robust and fault tolerant. These protocol are best suitable for Unstructured P2P network because these are highly adaptable for dynamic environments. These protocol also work best even 70% of node leave the network or failure in network . update is done by received peers by its neighbours 1 shows a sample of dierent view conditions. A view may be full of non-malicious peers, malicious peers or combination of them. P1, P2, P3 and Q1, Q2, Q3 are sample of peer P and the received view from a communicating peer, Let Q, respectively. Cases (P1 and Q1), (P2 and Q1), and (P3 and Q1) represent set of non-malicious peers inside views. In this case, view of P always lled non-malicious peers. In contrast, case (P1 and O3), (P2 and Q3), (P3 and Q2), and (P3 and Q3) are set of of malicious peers and the resulting view of P is always transferd into full of malicious peers. The gure also shows that the view of P turns completely malicious in those cases which have Q3 received view.This is also assume that if view is completely lled with set of malicious peer,such as P3,maybe recovered with non-malicious nodes when it receives non-malicious peer, such as Q1. Case (P2 and Q2) may ll partially or fully malicious peers into the resulting update view of P. Suppose the received malicious nodes are same as exist malicious nodes in side view. In this case only malicious nodes are updated. But the received malicious nodes are dierent, it may ll completely malicious nodes inside view. The Hub attack is very simple and strong. In Hub attack, the Bad peers send the descriptors of own and capture all over network , after some time these malicious peers leave the network by which network partition occur, and some peers are completely disconnected with network so gossiping between these peers not performed successfully, malicious peers communicate periodically not continuously so in degree of these peers are not become too high because of this nature existing algorithm are not very efficient to capture malicious peers. To protect our network we proposed Advanced Gossip (A-Gossip) protocol, proposed protocol efficiently handle malicious peer in network

III. RELATED WORK

Unfortunately,In starting there is no way to detect that a message comes form reliable source or from an attacker because message format is very simple so we have to identify and ban malicious peer to gossiping in network.

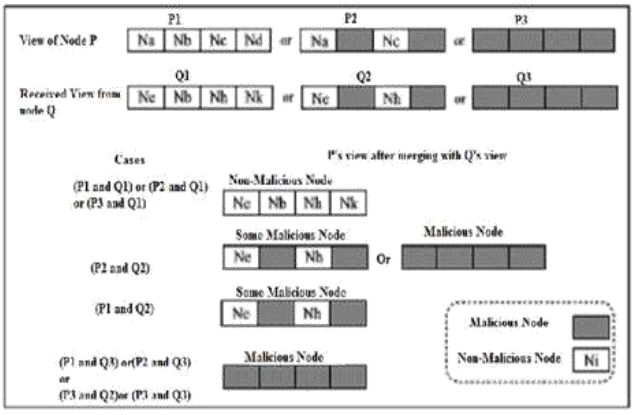

Fig. 1: Sample of Dierent View

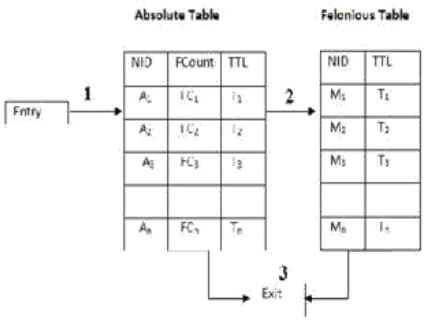

Fig. 2: An Example of A- Gossip Architecture

There is several protocol are present such as SPSS [17], TooLate [18] SPSS is intuition based approach. It works on central authority that maintain two list Black list and White list to maintain secure gossiping between nodes . A peer having in-degree greater than the threshold is put in blacklist and peers having in degree less tan threshold are put in white list . Peers which are present in Black list are supposed to malicious peers. Gossiping is carried out with only those peer which are present in white list. Drawback of SPSS is that Gossiping process is quite fast before peers determine that particular peers present in black list or white list malicious peer pollute the entire network and also there is high chance that central authority may be failed or hacked by some attacker. to overcome drawback of SPSS TooLate Protocol is discussed, TooLate avoid centralized blacklist mechanism TooLate come with concept of decentralization which focus on maintenance of Blacklist AS the name discussed TooLate, capture the malicious node before it being too-late.

IV. THE PROPOSED SOLUTION (A-GOSSIP)

A-Gossip protect network by malicious peer using non-malicious peers. It observed each gossip cycle to maintain and secure overlay from the partition,because Hub attack make network partition, In proposed protocol we capture malicious node before gossiping in the network . A sample of A-Gossip is shown in 2

A. Data Structure Used in A-Gossip:

As in previous existing traditional protocol like cyclone some data structure also used in A- Gossip . One data structure named View . View data structure is maintain by two descriptors named NodeID(NID) and Age. Node ID is always unique and unchanged forever, Age is used for showing oldness of peer within its view. Apart from above data structure proposed protocol advanced Gossip also uses two more data structure Absolute Table(At) and Felonious Table(FT).AT contains three descriptors named NID, FCount and Time-to-Live (TTL). FCount descriptor reects the same occurrence of that node while gossip with neighbours. TTL value indicates the liveliness of a node within the tables. FT has NID and TTL only. The node of AT is treated as non malicious and take part in gossiping while node of FT are treated as malicious . Both Table maintained in such a way that no common node is present in both table at any instance.

$$(mean)\mu_{FCount} = \frac{1}{|N|}\sum_{i=0}^{(N-1)} node_{i.(FCount)} \qquad (1)$$

and

$$(SD)\sigma_{FCount} = \sqrt{\frac{\sum_{i=0}^{(N-1)}(node_{i.(FCount)} - (\mu_{FCount}))^2}{|N|}} \qquad (2)$$

Fig. 3:

B. Gossiping in A-Gossip:

Our protocol(A- Gossip) is produced for reducing communication overhead in protection of network from malicious node. It avoids drawback of other existing protocol such as there is no central authority as in SPSS and use only one instance of protocol while TooLate uses multiple instances of protocol for detecting malicious node in network. For detect-ing malicious node , it uses two table at each node. The table are Absolute Table(AT), and Felonious Table(FT). The table are updated after each cycle. AT has Node Identification (NID), FCount and Time-to-Live (TTL). FT has only NID and TTL values. All tables is maintained an unique NID and is forced to shift when the defined threshold will be crossed. The threshold of AT is defined on FCount. The average or mean ( ) value of FCount is computed through equation ?? and standard deviation ( ) through equation ??. Here, N is the size of overlay. While updating the table entries, the nodes of AT are shifted to FT when the node's FCount will be more than the addition of mean and standard deviation of entire FCount and TTL value of FCount is greater. The value is predefined which is equal to the View Table (ViTab) length of the A-Gossip. ViTab refers as view of a nodes in the traditional gossip mechanism.

The mechanism assumed that the entry of nodes inside

Options	Action Of Receiving Node Ci
A's View	*Insert Node Ci into the A's AT (Absolute Table)*
A's AT	Update Node FCount
A's FT	Reinitialized TTL value with max
No Where	Insert node Ci into view Update FCount of Node Ci, if present

Fig. 4:

Entry/Exit	Cause	Update
1.	Receiving Node Only Present in View	Insert in to AT
2.	AT.Node.FCount >AT(μ+ σ) && AT.Node.FCount > TTL	Declared as Malicious and move inside Felonious Table(FT)
3.	AT.NodeTTL = 0 or FT.NodeTTL =0	Remove From Table

Fig. 5:

Malicious table will be declared as malicious. This table maintains only TTL value for the captured nodes and the value is reinitialized whenever the node reappear while gossip. From this mechanism of A-Gossip, it ensures that the captured actual malicious nodes inside Malicious table will be stuck in a loop. The value of TTL is initialized with the predefined ViTab length size. The nodes are free from tables as soon the TTL value becomes zero.

The mechanism tried to eliminate false +ve and false -ve from the tables. A-Gossip proved through simulation that it can provide secure from Hub attack. But, the mechanism gets in trouble if the malicious nodes will stop gossip for some interval.

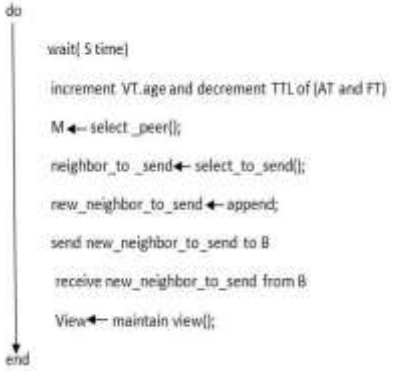

Fig. 6: Algorithm 1

The Gossiping in A -Gossip is carried out as follows and depicted in Algorithm 1. This technique uses Absolute Table(AT) and Felonious Table (FT)to detect and confirm the malicious nodes. To start with, each node (say P) keeps its peers in GT. A node chooses a peer (say Q) for gossiping (with maximum age value) from the AT and initiate gossiping by sending its gossip sequence (GS). GS contains its view appended with @ number of nodeids. Initially due to lack of malicious nodes the said could be 0s. The identical operation is also being executed by peer

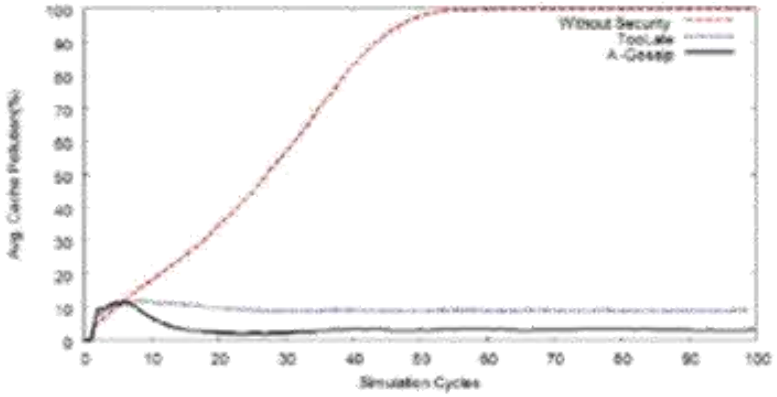

Fig. 7: With and without security mechanism: 2% Malicious Nodes

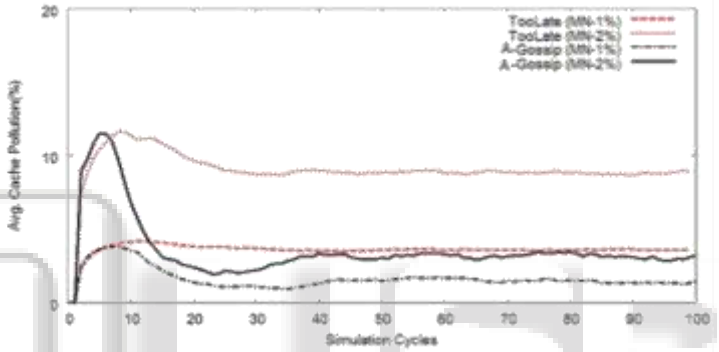

Fig. 8: 1% and 2% Malicious Nodes in 1,000 nodes

Q. Now both the peers P and Q update their view considering the received view in GS from their peer. This operation is called as maintain view(). Let P receives Ci in the view of GS received from Q. This leads to dierent cases as Ci may or may not exist in the tables of P. The action corresponding to each case is highlighted in table 1. Thus the exchange while gossiping is performed in the similar fashion to that of the traditional gossiping mechanism except a few suspicious NIDs are piggybacked. These NIDs are used to conform a node malicious or not by its peers.

V. Conclusion

Proposed paper discuss advanced gossip mechanism named A- Gossip for P2P unstructured networks. In A-

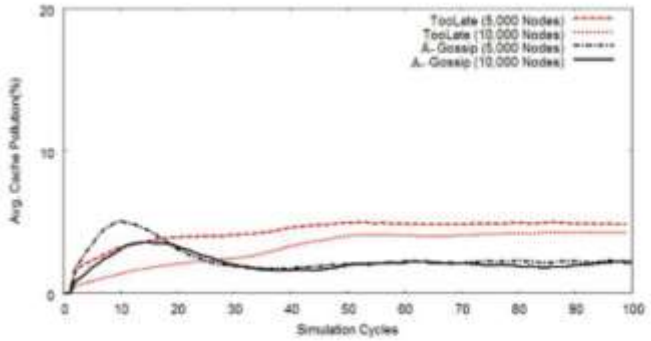

Fig. 9: 2% Malicious Nodes in 5,000 and 10,000 nodes

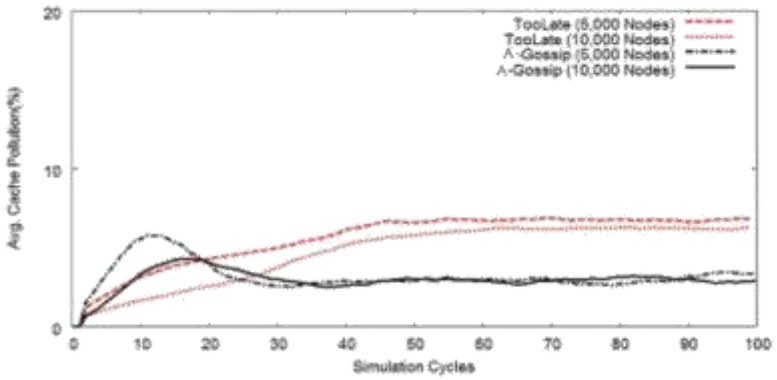

Fig. 10: 2% Malicious Nodes with 1% Churn rate on dierent size of network

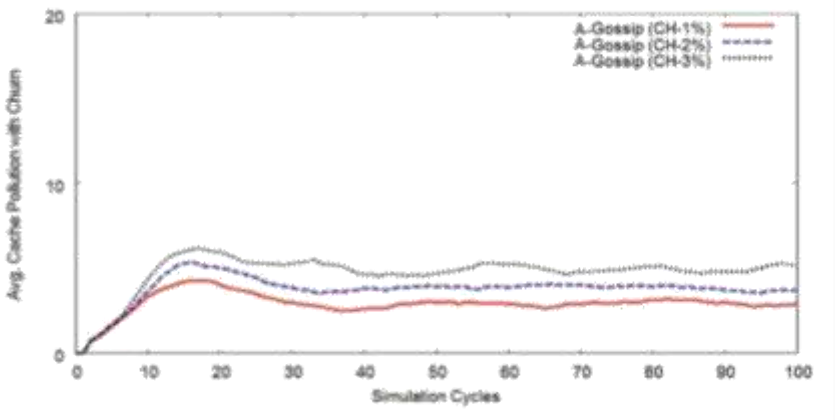

Fig. 11: 10,000 Nodes with Churn 1%, 2% and 3%: 2% Malicious Nodes

Gossip each node observes the behaviour of neighbour peers conforms to the network that peer is malicious or non malicious by feedback obtained from other peer . Proposed protocol captures the malicious peer and restrict them that they can not take participate in gossiping of network. We check performance of proposed protocol by Peersim sim-ulator and compare its performance by existing protocol TooLate . The proposed scheme is proved to be more secure than other existing protocols through Simulation.

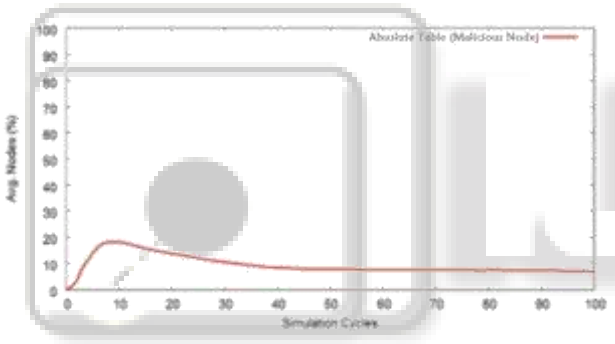

Fig. 12: Effect of Non-Malicious Nodes

REFERENCES

[1] I. Stoica, R. Morris, D. Liben-Nowell, D. R. Karger, M. F. Kaashoek, F. Dabek, and H. Balakrishnan : Chord- A Scal-able Peer-to-Peer Lookup Service for Internet Applications. IEEE/ACM Transactions on Networking, vol. 11, no. 1, pp. 17-32, (2003).
[2] I. Stoica, R. Morris, D. Karger, M. F. Kaashoek, and H. Balakrishnan, Chord: A scalable peer-to-peer lookup service for Internet applications, in Proc. of ACM SIGCOMM, 2001
[3] A. Rowstron and P. Druschel, Pastry: Scalable, distributed object location and routing for large-scale peer-to-peer systems, in IFIP/ACM International Conference on Distributed Systems Platforms (Middleware), 2001
[4] RATNASAMY, S., FRANCIS, P., HANDLEY, M., KARP, R., AND SHENKER, S. A scalable content-addressable network. In SIGCOMM01 (Aug. 2001).
[5] Napster. http://www.napster.com/.
[6] Gnutella. http://gnutella.wego.com/.
[7] KaZaA, http://www.kazaa.com
[8] A. Rowstron and P. Druschel: Pastry- Scalable, decentralized object location and routing for large-scale peer-to-peer systems. in Proc. of IFIP/ACM International Conference on Distributed Systems Platforms (Middleware), pp. 329-350, (2001).
[9] S. Rhea, D. Geels, T. Roscoe, and J. Kubiatowicz: Handling churn in a DHT. in Proc. of the USENIX Annual Technical Conference. Berkeley, CA, USA: USENIX Association, pp. 10-23, (2004).
[10] A. Demers, D. Greene, C. Hauser, W. Irish, J. Larson, S. Shenker, H. Sturgis, D. Swinehart, and D. Terry: Epidemic algorithms for replicated database maintenance. in Proc. of the 6th ACM Symposium on Principles of Distributing Computing (PODC87), pp. 1-12, (1987).
[11] P. T. Eugster, R. Guerraoui, A.-M. Kermarrec, and L. Mas-souli: Epidemic information dissemination in distributed sys-tems. IEEE Computer, vol. 37, no. 5, pp. 60-67, (2004).
[12] M. Jelasity, A. Montresor, and O. Babaoglu: A modular paradigm for building self-organizing peer- to-peer appli-cations. in Proc. of Engineering Self- Organising Systems. Springer, pp. 265-282, (2004).
[13] Mark Jelasity, Alberto Montresor, and Ozalp Babaoglu: The bootstrapping service. in Proc. Of the 26th IEEE International Conference Workshops on Distributed Computing Systems (IDCSW06).IEEE Computer Society, pp. 11-16, (2006).
[14] S. Voulgaris and M. van Steen,: Epidemic-style management of semantic overlays for content-based searching. in Proc. of Euro-Par 2005 Parallel Processing, pp. 1143-1152, (2005).
[15] S. Voulgaris, D. Gavidia, and M. van Steen: Cyclon-Inexpen-sive membership management for unstructured P2P overlays. Journal of Network and Systems Management, vol. 13, no. 2, pp. 197-217, (2005).
[16] Marin Bertier, Francois Bonnet, Anne-Marie Kermarrec, Vin-cent Leroy, Sathya Peri, Michel Raynal : D2HT- The Best of Both Worlds, Integrating RPS and DHT. European Dependable Computing Conference, pp. 135–144, (2010).
[17] G. P. Jesi, A. Montresor and M. van Steen: A Secure Peer Sampling., Elsevier Journal, 54, pp. 2086-2098, (2010).
[18] G. P. Jesi, D. Hales, and M. van Steen: Identifying Malicious Peers Before its TooLate: A Decentralized Secure Peer Sam-pling Service. IEEE SASO, Boston, MA(USA), (2007).

[19] Anceaume, Emmanuelle and Busnel, Yann and Gambs, Se-bastien: Uniform and Ergodic Sampling in Unstructured Peer-to-Peer Systems with Malicious Nodes. Springer,ISBN: 978-3-642-17652-4, Tozeur, Tunisie, (2010).
[20] Bortnikov, Edward and Gurevich, Maxim and Keidar, Idit and Kliot, Gabriel and Shraer, Alexander: Brahms: byzantine resilient random membership sampling. Proceedings of the twenty-seventh ACM symposium on Principles of distributed computing, Toronto, Canada, (2008).
[21] Ian Clarke, Oskar Sandberg, Brandon Wiley, and Theodore W. Hong.Freenet: a distributed anonymous information storage and retrieval system. In International workshop on Designing privacy enhancing technologies, Springer, pp 46-66, (2001).
[22] Voulgaris, S., Gavidia, D., van Steen, M.: Cyclon-Inexpen-sive membership management for unstructured P2P overlays. Journal of Network and Systems Management 13(2), 197217 (2005)
[23] A. Montresor and M. Jelasity: PeerSim: A scalable P2P simulator., IEEE Ninth International Conference, pp. 99-100 (2009)

