SECURED CLIENT CACHE SUSTAIN FOR MAINTAINING CONSISTENCY IN MANETS

R.Dhivya[1], V.Kavitha[2]

[1]M.E. –Final year, Communication Systems, M.Kumarasamy College of Engineering, Tamil Nadu, India
[2]Head of the Department, ECE, M.Kumarasamy College of Engineering, Tamil Nadu, India

Abstract

Mobile Ad Hoc Network is a self configured, Infrastructure less network, there will be vast usage among the mobile users. Each and every node in the network is free to move independently in any direction. Due to its dynamic topology, there will be insufficiency in their bandwidth consumption and get affected by several delays. In this the problem of cache Consistency was overcome by Client Cache Mechanism provides broadcasting of desired data to all the entire nodes and to maintain entry table list for each and every node. In order to provide error-free communication, security cryptographic mechanism (authentication) is also implemented to prevent malicious node to inject traffic into the network..

Keywords: *Invalidation report, Cache Path, Cache management, TESLA Key, Authentication.*

---***---

1. INTRODUCTION

Mobile ad hoc networks (MANETs) are gaining more and more popularity in recent years, because it fulfills the people's basic needs at any time at anywhere. MANETs is having an attractive solution in networking region. It is gaining much popularity in recent years. More research will be implementing in future decades. However, Wireless bandwidth and unauthorized node attacks are two important are drawbacks in ad hoc network. In Mobile Ad hoc Network, each of the Mobile User (MU) can retrieve their corresponding data through caching from Mobile Base Station (MBS).Because of its false networking topology, it gets affected by severe delays and bandwidth consumption. Factor which affects functionality is absence of caching the data to access its information and also security mechanism. Caching is necessary in order to provide corresponding information to the client's request from access server. In absence of security mechanism unauthorized node may inject traffic inside of the network and cause vulnerable distortion inside of nodal communication. So in order to provide error-free communication both client caching and authentication is necessary. In order to fetch the data and to maintain delay free communication, three types of basic algorithms were used:
1) Push- or Server-based (2) *Pull- or client based* (3) *Hybrid based*. In First approach, Push- or server-based, content owners (server) keep track of locations and send invalidation reports (messages) or updated contents whenever the contents are modified. It informs client about its updates and its cache current state. Because of server have to maintain all update records, when request arises frequently it is so hardly get the queries. It is a dangerous disadvantage in this approach.
2) In Second approach, Pull-or Client-based methods are client-based mechanism, the client checks the updates, considered outdated [7], are validated before serving new requests. In this client asks server to update or validate its cached data. In this every node maintains it update history, when request arises frequently, it is easy to fetch data from the nodes.
3) In third approach, where both caching node and server cooperate to keep the data to update. Server pushes the updates or client pulls them. General structure of MANET network is shown in fig 1.

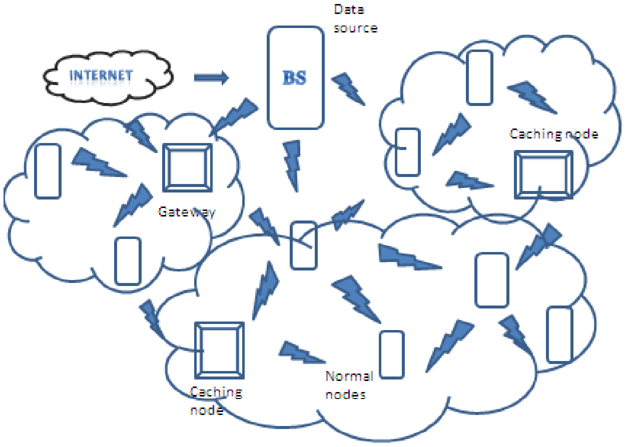


Fig-1 General Structure of mobile ad hoc networks

1.1 Basic Mechanism

In MANET environment, server is connected to external network terminal through router such as gateway. There is no fixed infrastructure, if two nodes want to get same data request from server, Query directory (QD) acts as a local server to fetch the query, Caching Node (CN) checks the nearest QDs, if it finds it's requested data then retrieve the information. If it misses, then it can get its data directly from server through wireless links. Here Server usage is minimal. Three mechanisms were implemented:

- TTL Algorithm: It reduces the Network Traffic by allotting time-to-live value for every packet transmission.
- Piggyback: It avoids frequent path disconnection, by carrying essential information related to node transmission, node ID and Cache response time likewise.
- Prefetch: It restricts long query latency by overwhelming hot data items (frequently used) with separate cache address.

1.2 Security Mechanism

Even though piggybacking concept carries all the essential data it also carries unessential information related to node like current query time, destination terminating address and cache interval longer than one TESLA interval likewise. Because of this unnecessary information leads to network traffic and unauthorized node may inject traffic into the network. In order to avoid this Security mechanism is essential to provide error-free communication. LHAP (Lightweight Hop by Hop Authentication Protocol) is implemented to avoid Masquerading and spoofing inside of the network. LHAP is an authentication mechanism use TESLA key to authenticate every node. It maintains separate MAC Address, Encrypted key and TESLA key for each node to providing security.

2. RELATED WORK

Mobile ad hoc networks (MANETs) have gained a great deal of communication because of its more advantages brought about by multi hop infrastructure-less transmission. Due to the error prone wireless channel and the dynamic network topology reliability of data delivery in MANETs especially biggest challenge in network environment remains an issue.

2.1 Cache Invalidation Scheme

Cache Invalidation techniques, earliest scheme, used in mobile ad hoc networks to maintain consistency of data among cache and to reduce long query latency. Server-based mechanism normally implements invalidation reports (IRs), are broadcasted periodically. It gets divided into two categories: Single-Hop mechanism and Multi-Hop mechanism. This scheme used to achieve an optimized overall performance in terms of packet and power efficiency.

2.1.1 Single-hop Mechanism

Domino Barbara and kupazg Imielinski proposed [6], an algorithm to broadcast the invalidation report based upon updating. Because of its limited size, an IRs can only record the updating history. When disconnection occurs, complete records get deleted. Kun-Lang and Philips K.Ye [15], made some modification on traditional IR-based mechanism, to overcome long disconnection problems. The querying mobile user must listen to the next IR-invalidation status. However this scheme reduces query latency, it can be overcome by inserting several updated invalidation reports between two successive IRS. Aurag kasal and co-authors [5], proposed asynchronous stateful (AS) strategy to maintain cache consistency. In these AS strategy, the Mobile Base Station (MBS) will broadcasts updated data items to neighbour nodes alone in order to avoid unnecessary IRs which cause delay. In Scalable Asynchronous Cache Consistency, developed by Zhijug Weahen and colleagues, the mobile base station (MBS), keeps only minimum state of information instead of storing all mobile users information. This can improve scalability and its performance.

2.2 IR-Based Algorithm

Zhuijg Wang proposed IR-based algorithm [12], in order to reduce network traffic. Server-based approaches generally employ invalidation reports (IRs) that are periodically broadcasted by the server. An IR entry list normally carries the IDs of the updated data items and the time stamps of the updated history. When a query is generated from the requesting node, the node from sender waits for the periodic IR to invalidate its cache (if connected) or not. If it is valid, then the query is transmitted. If the requested data item is invalid or modified, it usually waits for the periodic IR. In some proposed mechanism, like the Modified Time Stamp (MTS) mechanism [5], broadcasting of request packet is forwarded to the server without waiting for their periodic IR. Such schemes generally affected by large average delays due to the waiting mechanism for the periodic IR or from high traffic occurs in case of broadcasts are employed when misses occur and the request rate is high. At the beginning of the proposed scheme, server broadcasts update-Invalidation Reports (UIR), consists of last IR updated ID. Since a node which has to answer the query waits for periodic IRs to see whether the items are updated instead of waiting for next IRs. These approaches consequently reduce the generated network traffic by saving a list of submitted queries.

2.3 Cooperative and Adaptive Caching Schemes

Pietro Michiard proposed [10],caching schemes, the main idea behind these schemes is to analyse passing requests and cache either the data or the address of the node in which it is stored. There are three strategies behind these schemes: Cache Path, Cache management and Hybrid cache.

- **Hybrid Cache:** It is a middle solution where queries are cached by path or by data, depending on its preference. It can perform based on the optimization option.
- **Cache Path:** It saves space by storing locations where the data should be stored. In cache data, forwarding

node checks the frequently passing request s. If same data is fetched frequently then the forwarding node caches the data, sends that data to the requesting nodes, which avoids travelling further to the server.

- **Cache Data:** It saves time by storing the data instead of path. The main advantage is it saves caching space; hence disadvantage of caching path was overcome here.

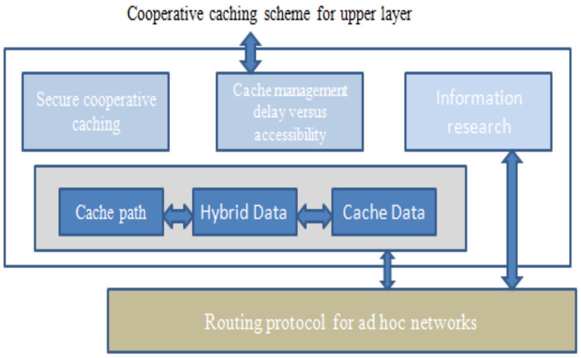

Cooperative caching scheme for upper layer
Secure cooperative caching
Cache management delay versus accessibility
Information research
Cache path
Hybrid Data
Cache Data
Routing protocol for ad hoc networks

Fig -2: Cooperative Cache Scheme

3. PROPOSED WORK

In Proposed work, normally in mobile ad hoc networks frequent disconnection occurs obviously It is the unmeasurable factor that affects entire communication. Even though useful information get hold by each and every node if frequent disconnection may occurs all the essential information get deleted. So in order to avoid this in my proposed work, Client Cache Consistency is used. It is a temporary storage maintained in the network used to provide the data even though disconnection may occurs. In this all the data conversation takes place between Base Station and Client get stored. If disconnection happened, after its longer time reconnection data will not get erased. It will be Wait on Cache storage this mechanism provide error-free communication. It also possess three techniques to provide strong communication 1) TTL Mechanism 2) Piggybacking 3) Prefetching.

- **TTL Mechanism**: Because of the frequent disconnection data may get wander inside of the network instead of reaching its destination.TTL mechanism reduce Network Traffic by allotting time-to-live value for every packet transmission.
- **Piggybacking:** It avoids frequent path disconnection, by carrying or storing essential information related to node transmission, node ID and Cache response time likewise. Disconnected node may get its transmission time, data packets, and frequency availability.

- **Prefetching:** It restricts long query latency by overwhelming hot data items (frequently used) with separate cache address. It stores frequently asked data packets in separate address so client can easily get the data without contacting server.

3.1 DCIM Mechanism

Distributed Cache Invalidation Method for Maintaining Cache Consistency client based algorithm possess low cost of caching the data because each and every node holds the information about the entry list and address ID. Hence it avoids unnecessary delays and network traffic. In this algorithm, in order to decrease the delay, every node holds the information about newly arrived packets and the neighbour should sends the packets immediately after acquiring the request from the adjacent node. It sends notification to the database query and informs about caching node details in sequential manner. Hence if a node sends packet it immediately updates its history. However, caching of data from neighbour node can be easy .i.e. can be easily fetched. Hence it doesn't lead to occur delay and query latency. This algorithm implements three schemes to make efficient communication between nodes: TTL, Piggybacking and Prefetching.

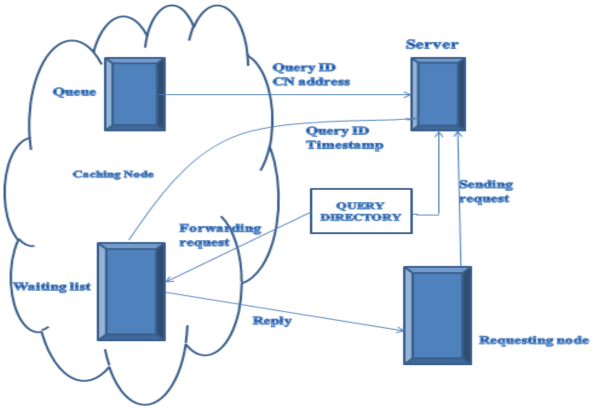


Fig -3: DCIM Mechanism

TTL value is added to each packet in order to reduce delays. If it expires, once it send into the validation requests and get refreshed, where another TTL value will be allotted for that data. The interaction between nodes can be carried by, the requesting node sends Data Rely Packet (DRP) to Query directory. It forwards the data packet to caching node if it holds information about data. If it failed to get the data, then that packets are send to the server. It will be placed in processing thread for to update its request, that it sends to the monitoring thread, where it get assigned with TTL value then it will be sends to the server with Cache Update request(CURP) in order to validate the data items. Then server sends Server Validation Reply (SVRP) to the caching node that which items are valid. Another message send from server is Server Update Data which includes updated data items and

Timestamps.Lastly, caching node release it request from requesting node and starts to transmit the data packets. It is proposing higher advantage compared to existing caching techniques.

3.2 Security Mechanism

Even though Cache Consistency provides lot of benefits to provide error-free communication piggybacking technique stores all the information with respect to that node. Through this unauthorized node may enter into the network may cause spoofing and masquerades the information. In order to avoid this Security is essential.

3.3 LHAP Protocol

It is an authentication mechanism provides separate MAC address, TESLA Key and Encrypted key for every node. LHAP provides a protection mechanism that can prevents many attacks from happening.

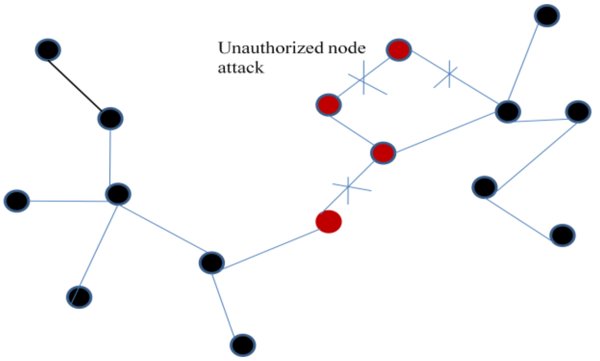


Fig -4: Security for preventing unauthorized attack

In LHAP, each and every node in the network verify every packet it receives from a neighbor before forwarding it (if it is not the destination node). Packets from unauthorized nodes are get dropped immediately, thus preventing them from propagating into the network. A packet that needs multiple hops before reaching its destination is thus get authenticated by each node on its path. This can be referred as *hop-by-hop* authentication.

LHAP's efficiency gains lot over traditional authentication protocols are derived from two techniques: (i) lightweight packet authentication, and (ii) lightweight trust management. Since all packets are authenticated on every hop on their paths, it is important that the packet authentication technique used by LHAP be as inexpensive as possible. LHAP employs a packet authentication technique based on the use of one-way hash chains. Secondly, LHAP uses TESLA to reduce the number of public key operations for bootstrapping trust between nodes, and also use TESLA for maintaining the trust between nodes.

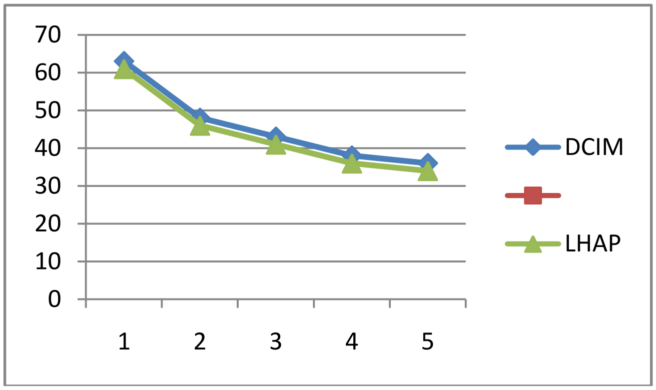
Lightweight Traffic Authentication: In LHAP, each node generates a one-way key chain that is used for authenticating traffic to its immediate neighbors.

Trust Management: Nodes can bootstraps their trust relationship, i.e., exchange authenticated TRAFFIC keys, by using a public key based technique

4. PERFORMANCE ANALYSIS

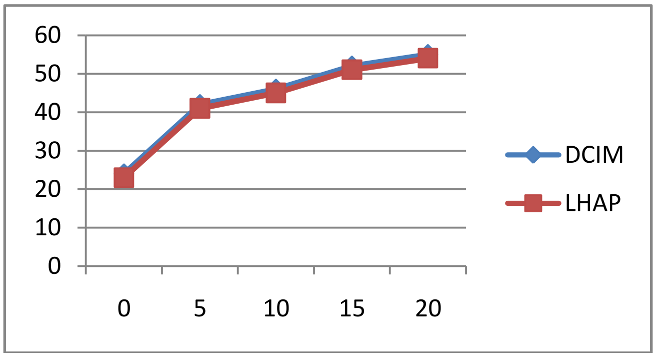
In this DCIM and LHAP Protocol get compared to provide strong error-free communication.

4.1 Traffic Overhead



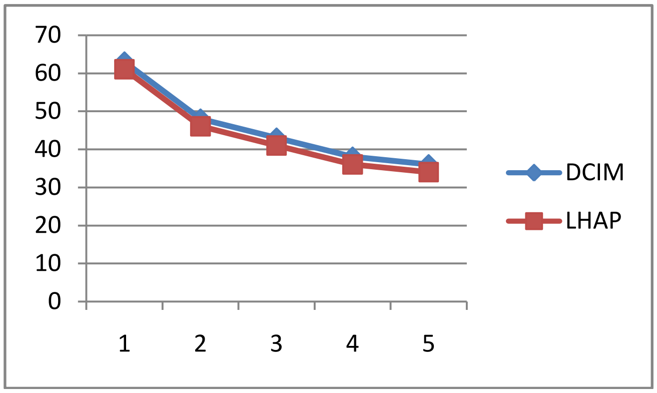
Traffic overhead get reduced in LHAP compared to DCIM because of TESLA key is introduced. Traffic overhead get reduced due to security mechanism holds key way chain for each data transmission. TESLA Key is included to provide effective transmission so as traffic can be reduced shorten.

4.2 Node Mobility



Path gets convergence because of un authorized attack present in DCIM, encrypted key is implemented to resolve this contention mechanism.

4.3 Traffic Delay



Delay is reduced because LHAP protocol add and count header format, destination address and path transmission of every node because it prevent unauthorized attack of injecting this kind of spurious information, hence delay can de reduced because of absence of nodal attack

5. CONCLUSIONS

The main focus is to provide efficient caching scheme to access the information. In MANET, fetching query from cached node cause severe delays and network traffic, it needs entry list to maintain the caching data table to avoid unnecessary delays. Because of arrival of high requesting rate among several nodes there will be cause of delay in order to fetch the data frequently. Security is also implemented to provide error-free communication.

ACKNOWLEDGEMENTS

I would like to give special thanks to Prof. V.Kavitha (Head of ECE) for her guidance, without whose patience and help it would be impossible to complete our project.

REFERENCES

[1] T.Andrel and A. Yasinsac, "On Credibility of MANET Simulations," IEEE Computer, vol. 39, no. 7, pp. 48-54, July 2006.

[2] S. Lim, W.C. Lee, G. Cao, and C. Das, "Cache Invalidation Strategies for Internet-Based Mobile Ad Hoc Networks," Computer Comm., vol. 30, pp. 1854-1869, 2007.

[3] Tang and S.T. Chanson, "The Minimal Cost Distribution Tree Problem for Recursive Expiration-Based Consistency "on IEEE Transaction on mobile computing ,pp no-sep 2008.

[4] J. Cao, and S. Feng, "A Selective Push Algorithm for Cooperative Cache Consistency Maintenance over MANETs," Proc. Third IFIP Int'l Conf. Embedded and Ubiquitous Computing, Dec. 2007.

[5] S. Lim, W.-C. Lee, G. Cao, and C.R. Das, "Performance Comparison of Cache Invalidation Strategies for Internet-Based Mobile-Ad Hoc Networks," Proc. IEEE Int'l Conf. Mobile Ad-Hoc and Sensor Systems, pp. 104-113, Oct. 2004.

[6] J.Jing, A. Elmagarmid, A. Helal, and Alonso, "Bit-Sequences: An Adaptive Cache Invalidation Method in Mobile Client/Server Environments," Mobile Networks and Applications, pp. 115-127.

[7] J. Cao, Y. Zhang, G. Cao, and X. Li, "Data Consistency for Cooperative Caching in Mobile Environments," on pp 453-546.jul-2009.

[8] L. Feeney and M. Nilsson, "Investigating the Energy Consumption of a Wireless Network Interface in an Ad Hoc Networking Environment," Proc. IEEE INFOCOM, 2006.

[9] Understanding the Power of Pull-Based Streaming Protocol: Can We Do Better? Meng ZHANG, , Qian ZHANG, IEEE transaction-pp no 145-189,2003.

[10] A.Idris, H.Artail, and H.Safa, "Query Caching in Manets for Speeding Up Access to Database Data," Proc. Third Int'l Symp. Telecomm. (IST '05), pp. 987-992, Sept. 2005.

[11] K.Mershad and H. Artail, "SSUM: Smart Server Update Mechanism for Maintaining Cache Consistency in Mobile Environments," IEEE Trans. Mobile Computing, vol. 9, no. 6, pp. 778-795, June 2010.

[12] Z.Wang, S. Das, H. Che, and M. Kumar, "A Scalable Asynchronous Cache Consistency Scheme (SACCS) for Mobile Environments," IEEE Trans. Parallel and [1].

[13] Distributed Systems,vol. 15, no. 11, pp. 983-995, Nov. 2004.

[14] G. Cao, L. Yin, and C. Das, "Cooperative Cache-Based Data Access in Ad Hoc Networks," Computer, vol. 37, no. 2, pp. 32-39, 2004.

[15] Q. Hu and D. Lee, "Cache Algorithms Based on Adaptive Invalidation Reports for Mobile Environments," Cluster Computing, vol. 1, pp. 39-50, 1998.

[16] N. Chand, R. Joshi, and M. Misra, "A Zone Co-Operation Approach for Efficient Caching in Mobile Ad Hoc Networks," Int'l J. Comm. Systems, vol. 19, pp. 1009-1028, 2006.

[17] C.Bettstetter and J.Eberspacher, "Hop Distances in Homogeneous Ad Hoc Networks," vol. 4, pp. 2286-2290, Apr. 2003.

[18] J. Cao, Y. Zhang, L. Xie, and G. Cao, "Consistency of Cooperative Caching in Mobile Peer-to-Peer Systems over MANETs," Proc. Third Int'l Workshop Mobile Distributed Computing, vol. 6, pp. 573-579, 2005.H. Jin,

[19] J. Cao, and S. Feng, "A Selective Push Algorithm for "Cooperative Cache Consistency Maintenance over MANETs," Proc. Third IFIP Int'l Conf. Embedded and Ubiquitous Computing, Dec.2007.

[20] O. Bahat and A. Makowski, “Measuring Consistency in TTL Based Caches,” Performance Evaluation, vol. 62, pp. 439-455, 2005.

[21] S. Das, C. Perkins, and E. Royer, “Performance Comparison of Two On-Demand Routing Protocols for Ad Hoc Networks,” Proc. IEEE INFOCOM, pp. 3-12, 2000.