ISSN No. 0976-5697

Volume 2, No. 5, Sept-Oct 2011

International Journal of Advanced Research in Computer Science

RESEARCH PAPER

Available Online at www.ijarcs.info

Efficient Invalidation Attack Resistance on Bootstrapping and Cache Consistency on MANET

P.Parameswari*
Research Scholar,
Anna University of Technology,
Coimbatore-641 047, India
param_pnr2000@yahoo.co.in

Dr.C.Chandrasekar
Reader,
Periyar University,
Salem-636 011, India
ccsekar@gmail.com

Abstract: Bootstrap security scheme is capitalizing neighbor node relationship and share knowledge information of other nodes in dynamic MANET. In this paper, we identify possible security attacks on bootstrapping and cache consistency and propose schemes for intrusion detection, damage recovery and intruder identification. We also address other security issues in cooperative cache based data access. Also, we propose a solution based on the improved IR-based cache invalidation and bootstrapping strategy to prevent intruders from dropping or modifying the invalidation messages. Digital signatures can be used to protect IRs but it has significantly high overhead in terms of computation and bandwidth consumption. To rectify this problem, we propose invalidation based bootstrapping and cache consistency schemes for intrusion detection and damage recovery. Experimental simulation is carried out to evaluate the proposed schemes. Our scheme can significantly improve the throughput and reduce the query latency, the number of uplink request, and the broadcast bandwidth requirements.

Keywords: Bootstrap, IR, MANET

I. INTRODUCTION

Deny-by-default computer systems promise enhanced information security (in comparison to their allow-by-default counterparts) by relying on policy rules to explicitly define the various actions that system components are allowed to take. The promise of enhanced security at the expense of open functionality is particularly appropriate in the context of mission-critical applications with the potential of an adversarial presence e.g., military Mobile Ad-hoc Networks (MANETs. The previous work addressed this challenge by defining, an axiomatic set of policies from which nodes can obtain additional policies or update outdated policies. It provided bootstrapping by which nodes form, the neighbor relationships among themselves in a manner consistent with current policy. However, this initial analysis neglected the possibility of nodes or wireless channel being subverted by a knowledgeable adversary. The previous work further handles the knowledgeable adversaries to secure the nodes effectively.

In a mobile ad hoc network (MANET), data caching is essential as it reduces contention in the network, increases the probability of nodes getting desired data, and improves system performance. The major issue that faces cache management is the maintenance of data consistency between the client cache and the server. In a MANET, all messages sent between the server and the cache are subject to network delays, thus, impeding consistency by download delays that are considerably noticeable and more severe in wireless mobile nodes. All cache consistency algorithms are developed with the same goal in mind to increase the probability of serving data items from the cache that is identical to those on the server.

When caching is used, data from the server is replicated on the caching nodes. Since a mobile node may return the cached data, or modify the route and forward a request to a caching node, it is very important that mobile nodes do not maliciously modify data, drop or forward the request to the wrong destination. The proposed research will study methods to avoid or detect such malicious nodes via authentication mechanisms. One common approach for data authentication is based on digital signature. With this approach, the data source can sign the data with its private key, so that intermediate routers cannot modify the data. However, the digital signature approach has high overhead, both in terms of time to sign and verify, and in terms of bandwidth. In this research, we design and evaluate techniques to reduce such overhead and balance system performance and security strength. Further, we identify possible security attacks on cache consistency and propose viable mechanisms to defend against such attacks.

II. RELATED WORKS

Enhanced information security is promised by Deny-by-default computer systems [3][4][6] by relying on policy rules to explicitly define the various actions that system components are allowed to take. For example, policy in a deny-by-default network [7][2][1][8] would specify which nodes can forward data or control traffic to which other nodes, datagrams for which no policy is provided are dropped. In contrast, policy in today's Internet [5] typically specifies only what traffic should be blocked (via firewalls or ingress filters at routers), datagrams outside of policy are forwarded. The promise of enhanced security at the expense of open functionality is particularly appropriate in the context of mission-critical applications with the potential of an adversarial presence e.g., military Mobile Ad-hoc NETworks (MANETs).

Several cache consistency (invalidation) schemes have been proposed in the literature [1], [2] for MANETS. In general, these schemes fall into three types i.e., pull or client model (caching node (CN) asks for updates from server),

push or server model, (server sends updates to CN), and cooperative model (CN and server cooperate to keep the data up-to-date). Pull-based strategies achieve smaller query delay times at the cost of higher traffic load, whereas push-based strategies achieve lower traffic load at the cost of larger query delays. Cooperative-based strategies tend to be halfway between both ends.

Recently, many researchers start to look into security issues in ad hoc networks. Hubaux et al. [13] addressed the issues of distributing public keys in ad hoc networks, by proposing to let users issue certificates for each other based on their personal acquaintances. Zhou and Haas [5] proposed a solution based on threshold cryptography. Based on a trusted certificate authority, the authors of [9] proposed a solution to secure the routing protocol of ad hoc wireless networks. In their protocol, nodes get certificates from the CA to identify themselves to avoid spoofing and malicious route updates. To address the high overhead associated with obtaining and verifying the digital certificates, Hu et al. proposed a protocol [11] to secure on-demand routing protocols based on TESLA [14], an efficient broadcast authentication scheme that requires loose time synchronization. They also identified the wormhole attack [12], which may make most routing protocols unable to find routes longer than one or two hops. Based on the intuition that a receiver can determine if the packet has traversed a distance that is unrealistic with precise timestamp or location information, they provided a packet leash solution to solve the wormhole attack.

III. INVALIDATION BASED BOOTSTRAPPING AND CACHE CONSISTENCY SCHEME

The bootstrap phase is run only once so that all initial node agree on a definition of normal traffic based on their personal experiences. During bootstrap, the nodes broadcast their output models to all the others. Each node calculates the similarity between its own input model and the output models from the other nodes, in order to compare the traffic that other nodes are sending out to the traffic that it is receiving. This similarity measure will be used in the future to either accept or reject new nodes to the MANET.

The bootstrap phase is based on the assumption that the initial nodes used to create the MANET represent well-behaved and well-intentioned nodes, and that the nodes have an accurate representation of what typical traffic in the MANET type being monitored looks like. We assume that initial behavior models for the nodes at the bootstrap phase come either from training in a simulated environment, or from real time deployment, i.e., nodes are meant for utility with a built-in pre-defined model of a certain application. We also assume that the application models have a measure of the typical density for each type of application. This application cardinality can be used as a countermeasure against attackers sending out fake models containing all 1's that would trick the AND similarity calculations.

A. Invalidation Attacks:

We consider two types of attacks that a malicious node can launch on the invalidation-based cache consistency scheme. First one, the node may drop some invalidation messages, such that its descendants can not receive the message and hence may unknowingly use stale cached data. Second, the node may modify some invalidation messages that it forwards, such that its descendants may receive wrong invalidation messages and hence may unknowingly use stale cached data or unnecessarily invalidate cached data.

To prevent malicious nodes from dropping invalidation messages, we borrow ideas from the IR-based cache invalidation scheme. In this approach, the server periodically broadcasts an invalidation report (IR) in which the changed data items are indicated. The IR consists of the current timestamp Ti and a list of tuples (dx, tx) such that $tx > (Ti - w \times L)$, where dx is the data item id, tx is the most recent update timestamp of dx, L is the length of the invalidation broadcast interval, and w is the invalidation broadcast window size. In other words, IR contains the update history of the past w broadcast intervals. Based on the value of w, clients can still validate their local cache even after missing w IRs. Similar to the original invalidation based approach, clients use the invalidation message (IR) to invalidate their local cache. Different from the original invalidation-based approach, the IR is sent out regularly, and the clients expect the IR at regular time interval. Therefore, if a malicious node drops an IR, its descendant nodes can detect it. A node may also miss an IR due to multicast tree partition caused by node movement or node failure. In either case, the node will re-join the multicast tree using some existing multicasting protocols, and then get the missed IR.

B. Invalidation Attack Resistance with faster cache updates and consistency:

We consider a wireless ad hoc network, which consists of a data center and many ordinary nodes. The data center (also called server) stores data that is updated now and then. Some ordinary nodes (called clients) frequently access the data, and cache some data locally to reduce network traffic and data access delay. We assume strong cache consistency is required, and the invalidation based bootstrapping and cache consistency model is used.

In the invalidation-based scheme, the server control node needs to send invalidation messages to clients. The most reliable method to ensure that all co-operative neighbor nodes with cached data, receive the invalidation messages is to use flooding, which has very high overhead. Another option is to use multicast by setting up a multicast group (tree), where the server control node is the root of the tree. With this approach, a node can find out whether its local cache is valid or not. If not, the node has to send a request to the data cache node to ask for the updated data. If the data will be accessed by many nodes, this approach not only increases the access delay, but also creates a large amount of network traffic.

To further improve performance, the multicast tree can also be used to push updates. Nodes receiving these updates do not need to send uplink requests and can reduce the access delay and bandwidth consumption. However, if no node is interested in this data update, pushing data down only consumes extra bandwidth.

The invalidation resistance techniques identify frequently accessed data in which updates are pushed to its maximal. It is easy to see that the multicast tree can be used for pushing cache invalidation and frequently accessed data to improve the system performance. In this paper, we assume that a multicast tree has been built to validate the cached data with faster updates in ad hoc networks.

IV. EXPERIMENTAL PERFORMANCE OF INVALIDATION BASED CACHE UPDATES AND CONSISTENCY

The experimentation is conducted on the MANET for evaluating the security scheme in terms of attack detection rate, false positive and false negative rate, bandwidth and query response time. These are the differences between the corresponding measures when no cache updating is in place and when server update mechanism is employed. Requests for data in the ad hoc network and data updates at the server are assumed to be random processes and may be represented by exponential random variables.

All the nodes are stationary and form a tree rooted at the server. The server maintains 100 data items, and each item is updated at an interval which is uniformly distributed with the mean value of 300s. Each client access a data item at an interval which is uniformly distributed with the mean value of 30s. Since a client may concentrate on a specific set of data items, we assume that 80% queries issued by a specific client are for a set of 10 data items, and the other 20% queries issued by the client are for other data items. Each client has a cache that can store 20 data items

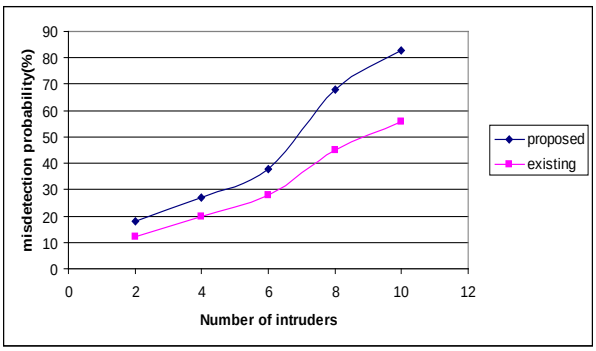

Figure 1.Number of Intruders Vs Misdetection Probability

Figure 1 shows the number of intruders with Misdetection probability. Therefore, as shown in the figure, the misdetection probability is large and increases rapidly as the number of intruder increases. However, as M increases, the number of group keys that are not compromised also increases, which reduces them is detection probability. Also, when M is large, more innocent nodes become the descendants of an intruder as number of intruders increases. Meanwhile, increasing Number of intruders does not rapidly increase the number of compromised keys. Consequently, the misdetection probability decreases as Number of intruders increases.

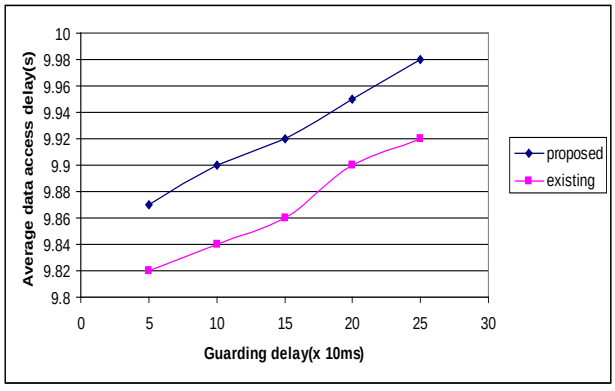

Figure 2.Guarding delay Vs Average data access delay

Figure 2 shows the average data access delay as the system parameters vary. From the figure, we can find that using the proposed schemes increases the average data access delay, compared to the existing scheme. From the figure, we can see that the data access delay is increased when using our scheme.

V. CONCLUSION

The previous work presented security mechanism safely maintains query node to maintain cache consistency and attack resistance in Mobile Ad hoc Network. The approach was built on top of the cooperative cache architecture for caching data items in MANETs and searching for them. Then analyzed the performance of the system through a gain loss model and then evaluated it by comparing it with the Updated Invalidation Report mechanism.

In this paper, we propose an Invalidation Based Bootstrapping and Cache Consistency Scheme which can significantly reduce the query latency and efficiently utilize the broadcast bandwidth. Detailed simulation experiments are carried out to evaluate the proposed methodology. Compared to previous schemes, our scheme can significantly improve the throughput and reduce the query latency, the number of uplink request, and the broadcast bandwidth requirements.

VI. REFERENCES

[1]. X. Yang, D. Wetherall, and T. Anderson, “A DOS-limiting network architecture,” in ACM SIGCOMM’05, pp.21–26, August 2005.

[2]. M. Srivatsa, D. Agrawal and S. Balfe, "Bootstrapping Coalition MANETs" in ITA Technical Report, February 2008.

[3]. H. Artail and K. Mershad, “MDPF: Minimum Distance Packet Forwarding for Search Applications in Mobile Ad Hoc Networks,” IEEE Trans. Mobile Computing, vol. 8, no. 10, pp. 1412- 1426, Oct. 2009.

[4]. H. Jin, 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, pp. 650-660, Dec. 2007.

[5]. X. Kai and Y. Lu, “Maintain Cache Consistency in Mobile Database Using Dynamical Periodical Broadcasting Strategy,” Proc. Second Int’l Conf. Machine Learning and Cybernetics, pp. 2389- 2393, 2003.

[6]. W. Li, E. Chan, Y. Wang, and D. Chen, “Cache Invalidation Strategies for Mobile Ad Hoc Networks,” International Conference on Parallel Processing (ICPP), pp.57, 2007.

[7]. S. Buchegger and J-Y. Boudec, “Performance Analysis of the CONFIDANT Protocol: Cooperation Of Nodes Fairness In Dynamic Adhoc NeTworks,” ACM Mobihoc, pp. 80–91, June 2002.

[8]. K. Sanzgiri, B. Dahill, B. Levine, C. Shields, and E. Belding-Royer. A Secure Routing Protocol for Ad Hoc Networks. IEEE Int’l Conf. on Network Protocols (ICNP),pp.78 – 87, Nov. 2002.

[9]. L. Xiao, L. Ni, and A. Esfahanian. Prioritized Overlay Multicast in Mobile Ad Hoc Environments. IEEE Computer, pp. 67 – 74, Feb. 2004.

[10]. L. Yin and G. Cao, "Supporting Cooperative Caching in Ad Hoc Networks," Proc. IEEE INFOCOM '04, pp. 2537-2547, 2004.

[11]. W. Zhang and G. Cao, "Group rekeying for filtering false data insensor networks: a predistribution and local collaboration-based approach," in Proceedings of the 24th Conference of the IEEE Communications Society (INFOCOM '05), vol. 1, pp. 503–514, Miami, Fla, USA, March 2005.