INTERNATIONAL JOURNAL OF ADVANCE RESEARCH AND DEVELOPMENT
IJARnD
(Volume3, Issue3)
Available online at: www.ijarnd.com

Secure and Avoid Inter-User Collision for Medium Access Control

Ramya. P[1], Vijayakumari[2] , Vijayanand. S[3], Anitha. S[4], Nathiya. G[5], Ponmozhi. C[6]

[1,4,5,6]*Student, The Kavery Engineering College, M. Kalipatti, Tamil Nadu*

[2,3]*Assistant Professor, The Kavery Engineering College, M. Kalipatti, Tamil Nadu*

ABSTRACT

Remote system powerfully designates channel assets to enhance otherworldly effectiveness and, to keep away from impacts, has its clients coordinate with each other utilizing a medium access control (MAC) convention. Nonetheless, MAC accepts client consistence furthermore, can be inconvenient when a client gets out of hand. An assailant who traded off the system can dispatch additionally destroying black hole attack(BHA) assaults than a system untouchable by sending unreasonable reservation solicitations to squander data transfer capacity, by tuning in to the control messages and leading force effective sticking, by distorting data to control the system control, etc. We construct SecureMAC to guard against such insider dangers while holding the advantages of coordination between the helpful clients. Secure MAC is contained four parts: channelization to anticipate inordinate reservations, randomization to upset receptive focused on sticking, coordination to counter control-message mindful sticking and resolve over reserved also, under-held range, and power attribution to decide every hub's commitment to the got control. Our hypothetical examinations and execution assessments illustrate better execution over past methodologies, which either disregard security issues or surrender the advantage of participation when under assault by handicapping client coordination, (for example, the Nash balance of nonstop wideband transmission).

Keywords: *Secure, Avoid, Medium Access Control.*

1. INTRODUCTION

Wireless networks are widely deployed today for governmental, commercial, and personal uses. As the demand for wireless communication increases, securing wireless systems against malicious behavior has taken on increasing importance. Previous protocols have been designed to use a shared key or an authentication server to provide link confidentiality, including Wired Equivalent Privacy (WEP), Wi-Fi Protected Access (WPA), and IEEE 802.11i (also called WPA2). Though WPA and WPA2 are quite robust in environments where the base station and client can share a key, such as home, small business, and enterprise settings, they cannot afford the same properties in open hotspot environments because of the lack of a properly shared secret key. In particular, in an open and public environment, such as coffee shops, bookstores, and restaurants, a single key may be known by an attacker, and individually shared secrets are difficult to distribute in a small and public environment. Some hotspots are owned or administered by a service provider such as AT&T that also owns and administers many other access points. In these environments, a subscriber to that service provider may be able to use WPA2-Enterprise to gain secure access. However, users that do not subscribe to that service provider do not have a shared key with that provider. Furthermore, unlike in cellular systems, the various service providers do not have a common roaming mechanism. Thus a user that joins to an access point but subscribes to a service other than locally overseeing service cannot create a secure MAC layer joining to that access point. CNSA final problem with trying to deploy WPA2-Enterprise in a hotspot environment is that ease of use and configuration is key to a successful deployment. In general, due to the abilities of users, a large number of Wi-Fi service providers, and the inherent need for open access, existing solutions to MAC-layer security in Wi-Fi are not applicable to many deployments of commercial hotspot service. Due to the lack of applicable MAC-layer security solutions, current 802.11-based hotspots choose one of two security strategies. The first strategy is to use no security whatever so that any user can connect directly to the Internet. Small coffee shops often use this strategy, sometimes in combination with a single WEP key that is distributed to all of the customers of that coffee shop. The second strategy is to use a captive portal.

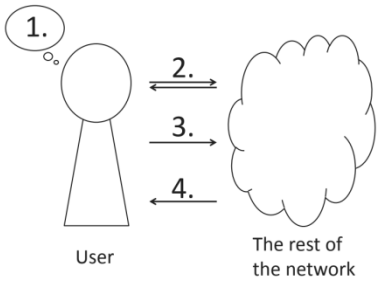


Fig.1

Fig. 1: Handshake-based MAC framework: 1. MAC control decision; 2. Control communication; 3. Data communication; 4. Feedback from receiver and network.

A captive portal is a router or a gateway host that will not allow traffic to pass until a user has authenticated himself. In a captive portal environment, a client device acquires an Internet Protocol (IP) address using Dynamic Host Configuration Protocol (DHCP) and any web request from the client device is redirected to the captive portal. The captive portal presents a web page; the user authenticates himself to the web page, possibly paying an access fee, the portal stops redirecting that client's traffic, so the client can now contact the rest of the Internet.

In this paper, we study the security of public wireless hotspots that use imprisoned portals. Our techniques are also applicable to other environments, but in this paper, we focus on their use in captive portal environments. We aim to address the problem that captive portals encrypt only the authentication phase, where the user supplies login credentials or other payment data and transmits user data in the clear [3]. As a result, many service providers recommend that their users use a Virtual Private Network (VPN) to secure their traffic, and some previous research attempts to use VPN to develop a secure public hotspot service [4]. However, not all people are able to use VPN. In addition, these approaches attempt to solve a MAC-layer problem at the network layer, and are fundamentally unable to address attacks at the MAC layer [3]. One important property for a MAC-layer security mechanism is that they must be resilient to attacker collusion.

In a collusion attack, attackers share information in an attempt to break the security of a victim node. In a commodity wireless system, an attacker can easily purchase a large number of wireless network interfaces and access points, increasing the importance of collusion-resistant protocols. We design a protocol that allows a client to establish a secure connection with an access point in the presence of malicious entities.

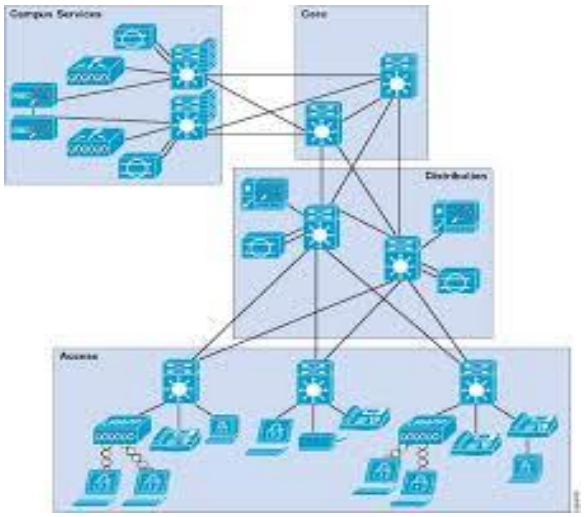


Fig. 2

When a client connects to an access point, our protocol provides a secure authentication and key exchange process. Our scheme constructs a protocol for establishing a secure connection on top of hierarchical identity-based cryptography [5]. Our scheme is scalable, easy to deploy, and provides secure authentication of both the client user and the access point, and is resistant to attacker collusion. The rest of the paper is organized as follows: we review some related work. We overview the hierarchical identity-based cryptography on which the key distribution of our proposed protocol based. We detail the proposed scheme and conclude.

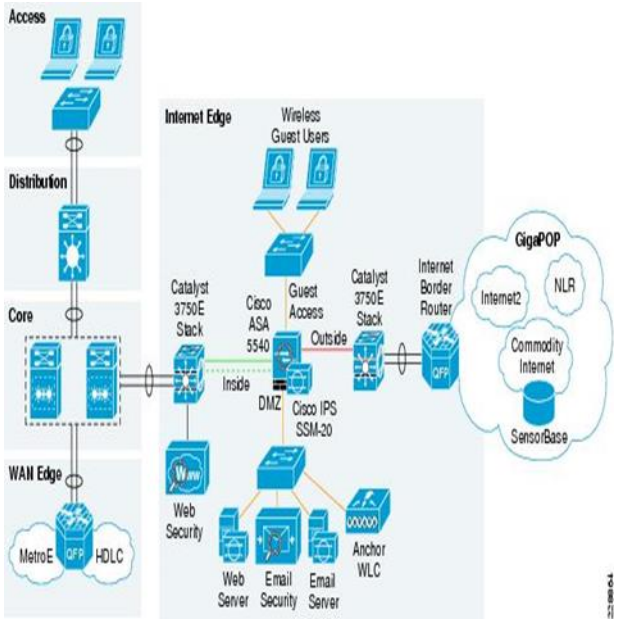


Fig .3

2. MAC OVERVIEW

MAC Framework In order to handle bursty traffic patterns characteristic of data transmissions, Medium Access Control (MAC) protocols is typically dynamic, rapidly adapting resource allocations based on user demand. One common approach is to have each node explicitly announce and share its channel usage intentions, which we call handshaking, before transmitting data; other users will avoid using that channel. Figure 1 illustrates a general handshake-based MAC framework; to send a packet, the transmitter (1) makes a MAC-layer decision based on its observations and the history from previous transmission rounds, (2) reserves channels for data transmission and shares its channel usage intention with other users in a control packet, (3) transmits the data packet using the reserved channels, and (4) receives feedback from the receiver and the network.

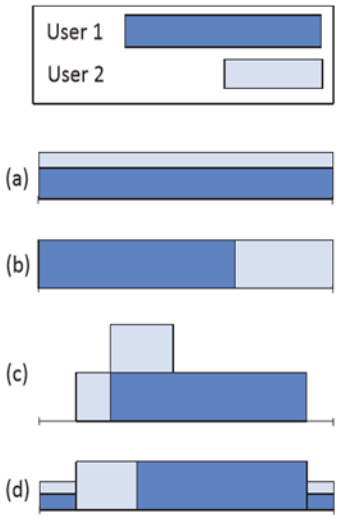


Fig.4

Our wireless MAC framework is applicable to many standardized protocols in last-mile networking and single-hop ad hoc networking, such as IEEE 802.11 (WiFi), IEEE 802.16 (WiMAX), and Bluetooth. In WiFi, users handshake using virtual carrier sense in which they access the channel via backoff-based random access and reserve the channel by exchanging Request to Send (RTS) and Clear to Send (CTS) messages before the data transmission. In WiMAX, a base station uses centralized scheduling to assign time slots and bandwidth to each users using a control channel, common and published as a part of the standard. In Bluetooth, a master. Channel access(Figure 4(b)) except for when the channel noise dominates the signal. SecureMAC randomization causes a partial collision and underutilization (Figure 4(c)) while SecureMAC coordination resolves the collision and waterfills the underutilized spectrum (Figure 4(d)).

SecureMAC Power Attribution & Commit-and-Reveal

We now describe the commit-and-reveal protocol in greater detail. In step one, each node commits to its data, nonce, randomization pattern, and handshaking list. For efficiency and

Attack-resilience reasons, we commit to these as follows. First, the data is subdivided into small blocks and committed using a Merkle hash tree [8]. Second, a commitment is made to the nonce. Next, the randomization pattern and handshaking list can be combined in a third commitment. Finally, the three commitments (the root of the Merkle Hash Tree for the data, the commitment to the nonce, and the commitment to the randomization pattern and handshaking list) are reliably disseminated to all nodes with authentication, for example, through Byzantine agreement with digital signatures [9].When a node receives a handshaking message in step two, the node checks to make sure that it is on the handshaking list declared in the message, and that the handshaking message is consistent with the sender's commitment from step one. If either of these checks fails, the node disregards the message.

After data transmission, each node provides enough information so that every other node can determine the state at that node. In particular, it composes a message with the nonce, randomization pattern, handshaking list, and the set of nodes from which it received a valid coordination message. This message is also distributed reliably and authentically, for example, through Byzantine agreement with digital signatures. In the event that any node F does not send such a message, the network detects the lack of such a message, and any node that received a coordination message from F can reveal F's randomization pattern and handshaking list. Once all nodes have the same subset of the nonce (ensured through the use of Byzantine agreement and previous commitment), they combine those nonces (e.g., by computing exclusive-or across them) to compute a network-wide random value. Each node A then determines the time slot tA in which it is to be audited by using a function on this random value, such as a pseudorandom function, keyed with the random value, computed on the node's node identifier. This time slot can, but need not, be the same for all users. Each node A then reveals the data it transmitted during slot tA, together with sufficient nodes in the Merkle Tree, to allow each receiver to verify that the data is the same as the data that has been committed. Each user can also determine that A sent the correct part of the data, because A's transmission rate does not depend on the data, but only on the presence of conflicts in A's reserved bandwidth and on the unused space detected by A.

At this point, each user can verify:

- The randomization pattern and handshaking list claimed by each node that either sent a coordination message to a legitimate host or participated in the protocol
- The information from which node A claimed to perform coordination at each point in time, for each participating node A, and
- The data that node A transmitted at time tA, for each participating node A.

The user then computes the cross-correlation described above to attribute a portion of the power received at time tA to A.

3. EXISTING SYSTEM

In wireless MAC security, previous work considers a black hole Attack attacker capable of either jamming, sending bogus requests to reserve channels, or falsifying information at the communication feedback.

However, these prior work focus on their respective threats and remain vulnerable when facing a more comprehensive threat model that introduces an attacker capable of performing all of the aforementioned threats.

Disadvantages:

Does not detect the collusion attack by compromised nodes. Detected compromised nodes cannot be performed due to a high risk of false positive. Increase the traffic load and conserve energy of the sensors. So the data is will be not secure.

4. SYSTEM MODEL

Our model supports the MAC framework described and applies generally across the protocols that implement the framework. There are T non-idle transmitters, which form the set T (each user is indexed with i where $i \in T = \{1, 2,...,T\}$), that share a frequency band with a total bandwidth W via frequency division1. In T, there are M malicious attackers, each identified by an index $k \in M = \{1, 2,...,M\}$, and the rest of them are protocol-compliant and collaborative. The network is a single-hop network, in which users communicate directly without any need for relaying, and each transmission is heard by all users. Thus, when two or more users operate on the same channel, they collide. The users do not favor any particular subset of the spectrum, and every part experiences equal path loss in expectation. Furthermore, users operate in a repeated game with infinite-horizon; that is, the transmitters do not run out of queued packets. Also, all users are time-synchronized at the packet level, and they operate in the same phase in the protocol (e.g., control communication phase) at any given time.

We build our scheme on pre-established keys, such as Diffie-Hellman key exchange and those used in resource-constrained sensor networks [12]–[10], and each pair of nodes share a secret key; our main contributions lie after node registration and key establishment. We also timestamp and authenticate control packets either by using digital signatures or by authenticating them to an online trusted authority (the reservation messages need only be authenticated to that online authority); this authentication eliminates forged MAC control Messages, thus ensuring that a user can be held responsible for the channels it has reserved. We further assume that each node knows which users are valid (e.g., based on a certificate signed by an offline trusted authority), which prevents the Sybil attack (one entity faking multiple identities).

5. PROPOSED SYSTEM

Our proposed Secure MAC randomization and coordination, and the centralized scheme that offers fully orthogonal access, either by using no randomization or perfectly orthogonal randomization. Because the behavior of each scheme depends on the handshaking list in use, we use three attacker strategies to represent different categories of handshaking list: an attacker that behaves like other

legitimate users and contains its transmission within its reserved bandwidth, an attacker that reserves as much spectrum as it can and performs wideband jamming outside that spectrum, since the user uses the ideal handshaking list that excludes the attacker and includes all benign users, and an attacker that performs narrowband jamming on the highest priority User.

Thus, to take advantage of the cooperative nature of most users, we focus on a group of compliant users sharing spectrum with malicious users (whose goal is to disrupt the network operations and have the option of behaving like a greedy user if other attacks fail).

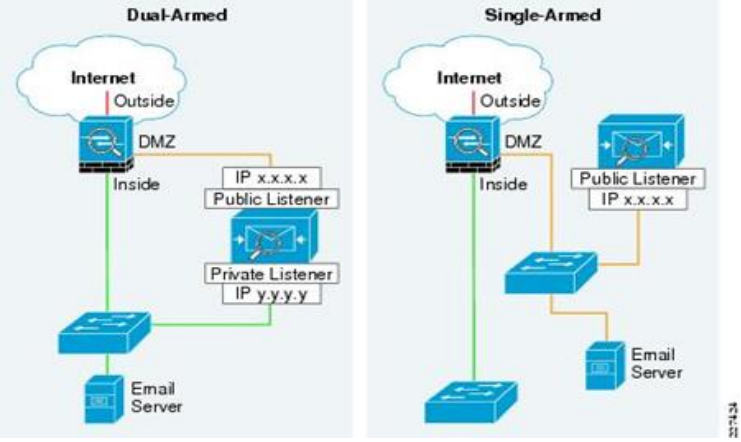


Fig.5

The attackers compromise a fraction of the network and have legitimate rights to vote, the distributed channelization is vulnerable to an attack where attackers attempt to distort the consensus to their advantage.

Its available information gains no additional advantage on the channels used by any other node, except that already available through any assurances of orthogonally.

6. ALGORITHM

Generalized linear model:

Networks are a useful representation for data on connections between units of interests, but the observed connections are often noisy and/or include missing values. One common approach to network analysis is to treat the network as a realization from a random graph model and estimate the underlying edge probability matrix, which is sometimes referred to as network denoising. Here we propose a generalized linear model identifying malicious node and avoid data loss. This model can be applied to data transmission time We develop an efficient projected gradient ascent algorithm to fit the model, establish asymptotic consistency, and demonstrate the empirical performance of the method on both simulated and real networks.

JVM LINKER

The JVM linker is used to add the compiled class or interface to the runtime system.

It creates static fields and initializes them.

And it resolves names. That is it checks the symbolic names and replaces it with the direct references.

JVM VERIFIER

The JVM verifier checks the byte code of the class or interfaces before it is loaded.

If any error occurs then it throws Verify Error exception.

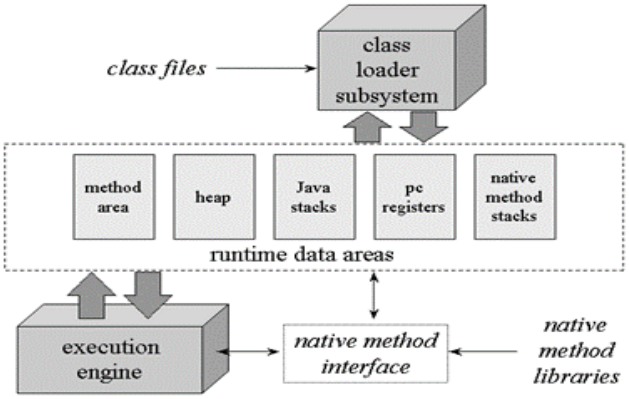
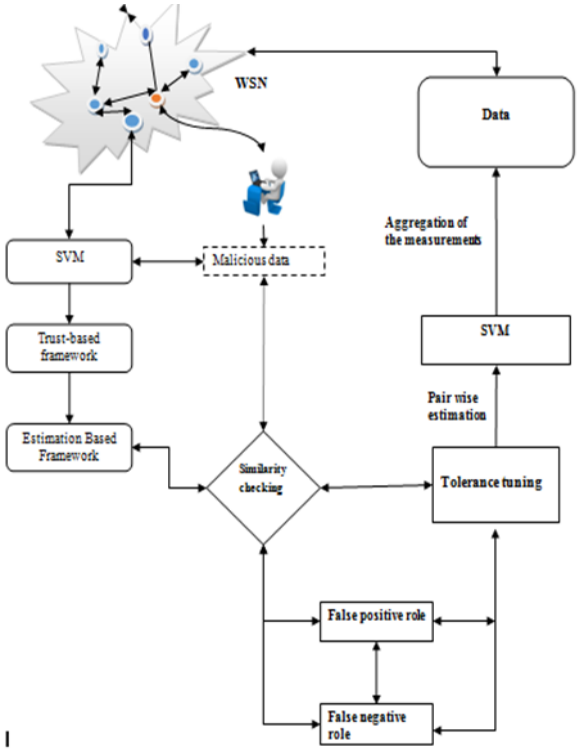


Fig.6

7. SYSTEM ARCHITECTURE DESIGN



8. CONCLUSION

This paper studies the inherent vulnerabilities of MAC against attackers who have the credentials of legitimately registered users. Threats that have been left unresolved in such environments comprise false proviso injection, forged feedback division, and intelligent jamming. Our scheme defends against such threats using a combination of four mechanisms: channelization that allocates bandwidth based on the usage in the previously reserved spectrum, randomization to defend against reactive and outsider jamming, coordination to resolve collisions caused by randomization and power attribution to make future MAC control decisions. Our evaluations show that, in practical scenarios, both centralized and distributed versions of our work are successful in nullifying the attackers' advantages of compromising the network while having the benign users retain the benefit of user collaboration in MAC. In particular, in our implementation environment, our work outperforms security-oblivious MAC with entity fair channelization by 159%, FHSS (without coordination) and entity-fair channelization by 149%, and the Nash equilibrium of wideband access by 76%.

9. REFERENCE

[1] S.-Y. Chang, Y.-C.Hu, and Z. Liu, "Securing wireless medium access control against insider denial-of-service attackers," in Proceedings of the IEEE Conference on Communications and Network Security, ser. CNS '15.IEEE, 2015.
[2] S.-Y. Chang and Y.-C. Hu, "Secure Channel Reservation for Wireless Networks," in Technical Report. [Online]. Available: https://www.ideals.illinois.edu/bitstream/handle/2142/17098/20102509 Chang.pdf?sequence=2
[3] R. Etkin, A. Parekh, and D. Tse, "Spectrum Sharing for Unlicensed Bands," vol. 25, no. 3, p. 517, 2007.
[4] J. Chiang and Y. Hu, "Dynamic Jamming Mitigation for Wireless Broadcast Networks," in INFOCOM 2008. The 27th Conference on Computer Communications. IEEE, 2008, pp. 1211–1219.

[5] J. Bellardo and S. Savage, "802.11 Denial-of-Service Attacks: Real Vulnerabilitiesand PracticalSolutions," in USENIX Security Symposium, August 2003
[6] Z. Liu, H. Liu, W. Xu, and Y. Chen, "Exploiting jamming-caused neighbor changes for jammer localization," Parallel and Distributed Systems, IEEE Transactions on, vol. 23, no. 3, pp. 547–555, 2012.
[7] R. Pickholtz, D. Schilling, and L. Milstein, "Theory of spread-spectrum communications–a tutorial," IEEE Transactions on Communications, pp. 855–884, May 1982.
[8] M. Simon, J. Omura, R. Scholtz, and B. Levitt, Spread spectrum communications handbook. McGraw-Hill: New York, Mar. 1994.
[9] L. Baird, W. Bahn, M. Collins, M. Carlisle, and S. Butler, "Keyless jam resistance," in Information Assurance and Security The workshop,2007. IAW '07. IEEE SMC, June 2007, pp. 143–150.
[10] M. Strasser, S. Capkun, C. Popper, and M. Cagalj, "Jamming-resistant key establishment using uncoordinated frequency hopping," Security and Privacy, 2008.SP 2008. IEEE Symposium on, pp. 64–78, May 2008.
[11] W. Xu, W. Trappe, and Y. Zhang, "Channel surfing: defending wireless sensor networks from interference," in IPSN '07: Proceedings of the 6th international conference on Information processing in sensor networks. New York, NY, USA: ACM, 2007, pp. 499–508.
[12] R. Gummadi, D. Wetherall, B. Greenstein, and S. Seshan, "Understanding and mitigating the impact of rf interference on 802.11 networks," in SIGCOMM '07: Proceedings of the 2007 conference on Applications, technologies, architectures, and protocols for computer communications. New York, NY, USA: ACM, 2007, pp. 385–396.
[13] J. H. Reed and M. Lichtman, Letter Response to FirstNet Conceptual Network NOI (Docket No. 120928505250501; RIN 0660XC002), Nov. 2012.

