Secure Access Control for Location-Based Applications in WLAN Systems

YounSun Cho, Lichun Bao and Michael Goodrich

1 Introduction

With the growing population of high speed wireless portable devices, wireless LAN (WLAN) systems based on IEEE 802.11 a/b/g are becoming the prevailing access technologies, and increasing being adopted by the retail stores to attract their customers by offering them with on-site Internet access conveniences. Location-based service provisioning is of great interests to wireless Internet service providers to deliver attractive value-added services, such as service advertisements, product marketing, to the special network users according to their geographic locations, along with Internet access benefits. It is usually the case that the basic location information is derived from direct interactions between the infrastructure network and the wireless mobile devices. Unfortunately, the Internet access service easily becomes the target of free-riders and attackers by exploiting the location authentication protocols, and collusively forging their location claims. WLAN systems are vulnerable to misuses and abuses because of their open-air transmissions in untethered environments. Therefore, a natural resort in WLAN systems is to exercise network access control to authenticate network access according to various user certificates.

User identity-based access control is a promising approach. A user's identity [25] can be based on a password, a token, a ticket, an administered access control list (ACL), or biometrics [14, 26]. SecureID is a token-based authentication scheme for a user remotely logging into corporate networks using a combination of both a password and a random number token [27]. Kerberos is another widely used ticket-based network authentication protocol today [14]. It is designed to provide strong authentication for client/server applications based on the secret key cryptography, which is derived from the Needham-Schroeder key distribution protocol [26]. Likewise, the access control list (ACL) is commonly used for access control by modern operating systems [4].

However, identification-based access control does not satisfy certain security requirements that depend on user information such as the location as we mentioned before. Moreover, identification-based approaches may require user-agreement, key distribution, communication overheads in order to procure the related identities.

Various location sensing schemes have been reported. One approach is to estimate the position of a given source based on the received signal strength. A variety of ranging and positioning techniques with different technologies such as RF, ultrasound or infrared, have been proposed to solve this problem [12, 13].

The Cricket system is a decentralized indoor location-support system that requires the combination of RF (radio-frequency) and ultrasonic signals in order to trace user locations and to provide location services to users and applications [20]. PAC is based on the Cricket system for location tracing, and adopts the INS/Twine [18] architecture for scalable resource discovery [22]. In PAC, The client first acquires

a Location ID (LID) along with a time-varying Location Code (LIDCODE) from its surrounding access points' beacons, then sends them to a location authentication server to a service-granting ticket. PAC requires synchronization between beacons and the location authentication server, and keeps track of the corresponding LIDCODEs as they change with time.

Sastry *et al.* described an Echo protocol to compute node location based on the round-trip latency of messages and ultrasonic signals in location computations [23]. They proposed the concept of *Region of Acceptance* in order to combat malicious location-provers from submitting location claims that overstate the true processing delay. Water *et al.* proposed a similar protocol for proving the location of tamper-resistant devices, based on the exchange RF messages [5].

Zhang *et al.* [32] proposed the use of location-based keys using identity based public-key cryptography (ID-PKC) , which solves the Bilinear Diffie-Hellman Problem (BDHP) [7, 24].

Although exact location information meets the goal of location-based access control mechanisms, such localization is not required in many cases. Instead, coarse location information, such as the areas enclosed within airport, Internet cafe, hotel, etc., is sufficient to provide location-based access control. These areas can be easily defined by a set of access points, and are much more static than the previous ones. In these coarse location scenarios, we suggest that the access to a WLAN system is granted if and only if the clients are located within the areas concurrently covered by multiple access points. Using sectored antennas, we can further specify the areas in desired shapes.

We propose a location authentication and authorization protocol, LBAC (Location-Based network Access Control), to securely authenticate the location claims of mobile wireless users, and to securely distribute the shared keys for data encryption purposes. In LBAC, location areas are defined by the shared points of multiple wireless access points (APs). The fact that a mobile node is located at certain places is authenticated by the mobile node collecting all the key information from the corresponding access points. Using Diffie-Hellman algorithm, LBAC authenticates location claims, and derives the shared keys for each mobile node and access point pair. LBAC eliminates the dependence on Global Positioning System (GPS) or ultrasonic devices in order to localize the mobile devices. We enumerate possible attacks to the system and analyze their countermeasures. The computational, communicational, and the memory requirement are also evaluated.

The rest of the chapter is organized as follows. Section 2 provides the basic algorithm and protocol used by LBAC. Section 3 describes the network assumptions and the protocol operations of LBAC. We evaluate the efficiency and security features of LBAC in Section 4. Section 5 concludes the chapter.

2   Background Review

2.1   Diffie-Hellman Key Exchange

Diffie-Hellman key exchange scheme is used to agree on a shared key, K_{AB}, securely between two parties/nodes A and B [9, 19].

In Diffie-Hellman algorithm, two publicly-known numbers are distributed beforehand — a prime number p, and a generator g of the cyclic group $Z_p{}^*$. In order to derive a shared key between nodes A and B, node A chooses random private key value $X_A \in Z_{p-1}$, then computes a public key value $Y_A = g^{X_A} \bmod p$. Similarly, node B chooses random private key value $X_B \in Z_{p-1}$, and computes $Y_B = g^{X_B} \bmod p$. Nodes A and B now exchange the public keys Y_A and Y_B of each other explicitly, and derive the shared secret key
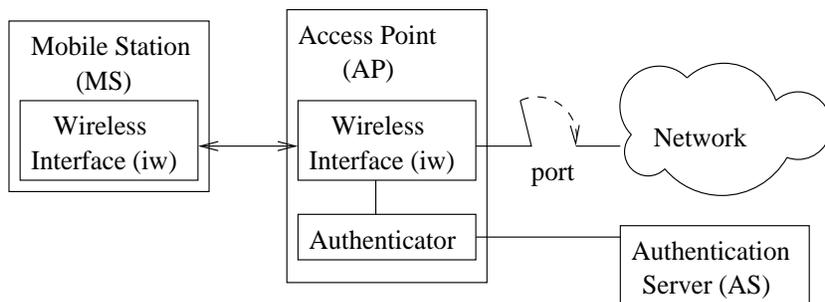
Figure 1: Three Entities for Port-Based Access Control in 802.11i.

using the modulo arithmetic as follows:

$$K_{AB} \equiv Y_A^{X_B} \equiv g^{X_A X_B} \equiv Y_B^{X_A} \equiv K_{BA}(\mathtt{mod}\ p). \tag{1}$$

Once the shared key is established, secure communication between two parties can be established using any symmetric key encryption scheme, like Data Encryption Standard (DES).

2.2 IEEE 802.11i Overview

After the intermediate WEP (Wired Equivalent Privacy) in the original IEEE 802.11 standard [1], the IEEE 802.11i standard has been proposed to specially designed to address WEP's weaknesses [2]. The IEEE 802.11i separates the user authentication process from the message protection process in order to meet the goals of RSN (Robust Security Network) [16, 10], thus contains the following components:

1. The port-based network access control protocol IEEE 802.1x [3] for the authentication purpose, which entails the use of EAP (extensible authentication protocol) and an authentication server, such as a RADIUS (Remote Authentication Dial In User Service) server. The authentication and key distribution process can be in the four-way handshakes mode, or a simplistic pre-shared key (PSK) mode.

2. AES-based encryption protocol, CCMP (Counter Mode with Cipher Block Chaining Message Authentication Code Protocol), to provide confidentiality, integrity and origin authentication.

IEEE 802.11i defines two modes of authentications: IEEE 802.1x authentication (*i.e.* Extensible Authentication Protocol (EAP) authentication) and the pre-shared key (PSK) mode. The former is required and extensible to several authentication protocols using the port-based access control IEEE 802.1x, which was originally designed for the Point-to-Point Protocol (PPP), as in MODEM connections and wired LANs. The latter does not require an authentication server, but requires an static pre-shared keys between access points and mobile stations. A pairwise master key (PMK) is obtained directly from a pre-shared key (PSK) with pseudo-random functions.

The IEEE 802.1x authentication process involves the three entities — the mobile station (MS), the access point (AP) and the authentication server (AS) in case of WLANs as shown in Fig. 1. The AS resides in the network, and the MS, who initially does not have access to the network, is connected to the AP. The AP is the entity that initially blocks the MS's access to the network, and also serves as a broker between the MS and the AS during the authentication process. Only after the MS is authenticated by the authenticator on
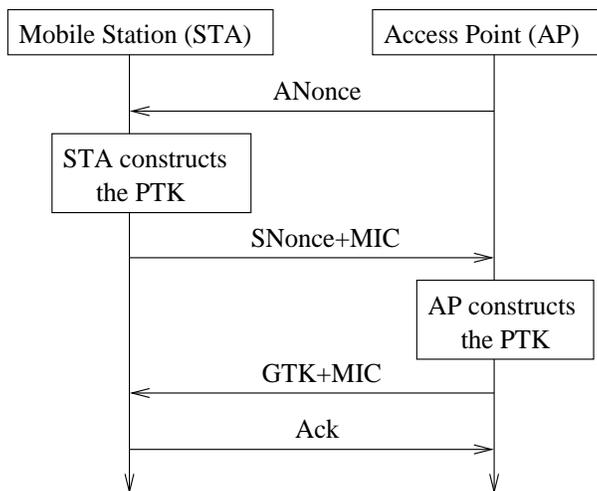
Figure 2: Four Way Handshakes for Mutual Authentication and Key Generation.

the AP to the AS, can the MS access the network. The IEEE 802.1x EAP exchange provides the shared PMK (Pairwise Master Key).

However, the PMK is designed to last for the entire session, and should be exposed as little as possible. Therefore, a four-way handshake is used to establish another key called the PTK (Pairwise Transient Key), and to authenticate the access point (AP) to the mobile station (STA), as shown in Fig. 2. The PTK is generated through a cryptographic hash function with the concatenated product of the following attributes: PMK, AP nonce (ANonce), STA nonce (SNonce), AP MAC address and STA MAC address. The four-way handshake works as follows:

1. The AP sends a nonce-value to the STA (ANonce), which now has all the attributes to construct the PTK.

2. The STA sends its own nonce-value (SNonce) to the AP together with an MIC (message integrity code).

3. The handshake also yields the GTK (Group Temporal Key), used to decrypt multicast and broadcast traffic. The AP sends the GTK and a sequence number together with another MIC. The sequence number is the sequence number that will be used in the next multicast or broadcast frame, so that the receiving STA can perform basic replay detection.

4. The STA sends a confirmation to the AP.

The pairwise transient key (PTK) is then divided into three separate keys: 1) EAPOL-Key Confirmation Key (KCK) to compute the MIC for EAPOL-Key packets, 2) EAPOL-Key Encryption Key (KEK) to encrypt the EAPOL-Key packets, and 3) Temporal Key (TK) to encrypt the actual wireless traffic.
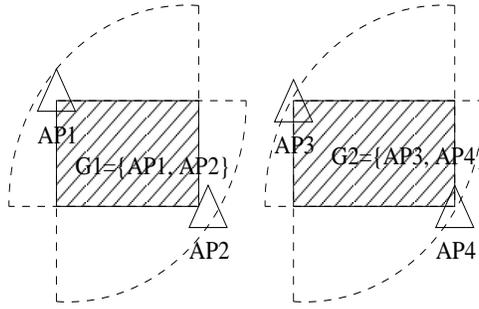
Figure 3: Defining Access-Granted Network Areas.

3 Location-Based Access Control (LBAC)

3.1 Network Assumptions

We address the network access control problem in infrastructure-based WLAN systems based on IEEE 802.11 [1], which involves two types of elements: the access points (APs) and the mobile stations (MSs). In addition, for authentication and key distribution purposes, we have another type of node, called the key server (KS), which provides similar functionalities as the authentication server in IEEE 802.1x.

The access-controlled areas are divided into two kinds — access-granted and access-denied areas. We assume that the network access points are purposefully deployed such that the desired access-granted areas are covered by multiple access points. Specifically, the access-granted areas can be custom-made into special shapes according to customer requirements using directional antennas [8] by adjusting the angle and distance of the signal propagation. Although multi-path effect could change the shapes of the network coverage, we do not address it in this chapter.

We also assume that mobile devices have sufficient computational and communicational capacities to carry out the simple cryptographic operations required in LBAC.

In Fig. 3 illustrates two access-granted areas as defined by the shaded areas of two AP groups $G_1 = \{AP_1, AP_2\}$ and $G_2 = \{AP_3, AP4\}$, where the directional antenna of each access point spreads $90°$. The access-denied areas are those areas outside access-granted areas. The set of access points that cover an access-granted area is called the *location group* of the access-granted area. The location groups that an access point belongs to are designated by the network administrators or an automated bootstrapping process when the WLAN system is initially designed and deployed.

3.2 LBAC Protocol Operation

LBAC may be applied in two phases of the mobile station network accesses: one is during the initial network access authentication phase, in which case LBAC can use the same-format protocol message as in IEEE 802.11i four-way handshaking mechanism, the other is during normal data frame exchange phase, in which case LBAC protocol messages are carried in regular data frames. In either case, LBAC works according to the same protocol. Therefore, we do not differentiate LBAC operations in these scenarios, and focus on the message exchange procedures.

LBAC uses the Diffie-Hellman key exchange scheme for user location authentication, network-access authorization and data encrypting key distribution purposes. We employ the key server for the key exchange
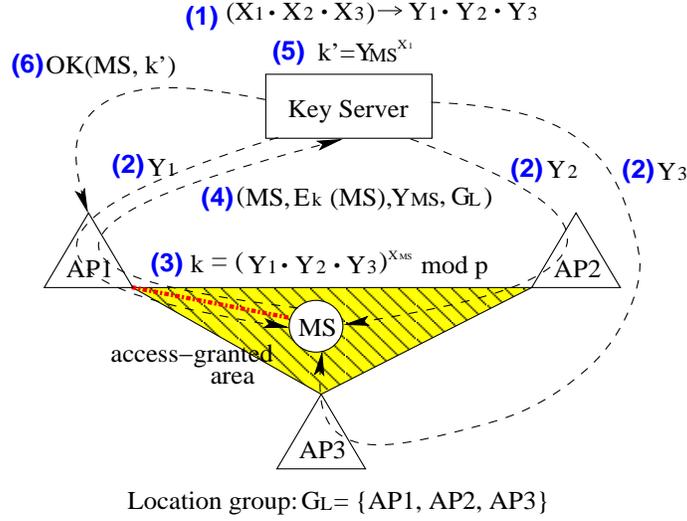
and management purposes.

Figure 4: Key Distribution Protocol in LBAC.

Fig. 4 shows the location-based access control architecture, which includes the three elements – mobile stations (MSs), access points (APs) and a key server (KS). In particular, Fig. 4 illustrates an example network with one mobile station MS located in the access-granted area confined by the location group $G_L = AP_1$, AP_2, AP_3. The key server KS connects to each AP by *low-latency* and *secure* connections, which is assumed to be provided by the infrastructure networks separately from LBAC. The LBAC operational steps are marked by numbers, and the message flows are indicated by arrows. For simplicity, we omitted a location group indicator in all of the messages in Fig. 4.

The key server initiates the location-based access control process, authenticates the MS and generates the data encryption key for the MS to access the network. We describe LBAC protocol operations in two phases: the authentication phase and the key generation phase according to Fig. 4. The authentication of MS location claims is carried out from step (1)-(5), and the key-generation is carried out in step (6).

3.2.1 LBAC Authentication Phase

In LBAC, each AP is assigned a unique *location key*, which is the public key used in the Diffie-Hellman algorithm. The corresponding private key of the location key is kept as a secret value by the key server.

The location keys are generated and distributed to the APs by the key server periodically as shown in Step (1) in Fig. 4. The location keys assigned to AP_1, AP_2 and AP_3 are Y_1, Y_2 and Y_3, respectively.

After receiving these location keys, the APs broadcast these keys via their beacon messages or special broadcast data frames in their wireless cell or BSS (basic service set) as shown in Step (2) of Fig. 4. When a mobile station is located within the access-granted areas, such as MS in Fig. 4, the station can listen to the channel and gather these location keys of the location group, which is defined by $G_L = \{AP_1, AP_2, AP_3\}$.

In Step (3), the MS derives the location claim key k by multiplying all the location keys from the APs of the location group G_L, then raising the product to the power of X_{MS} using modulo arithmetic, thus giving the location claim key k as

$$k = (Y_1 \cdot Y_2 \cdot Y_3)^{X_{MS}} = g^{(X_1+X_2+X_3)\cdot X_{MS}} \pmod{p}.$$

Such computation is similar to the shared key computation in the Diffie-Hellman algorithm, but with a little complication to the base.

After derive the location claim key k, the MS composes the location claim with four-element tuple $(MS, E_k(MS), Y_{MS}, G_L)$, that is, the MS identifier MS, the encrypted mobile station ID $E_k(MS)$ using key k, the current MS public key Y_{MS}, and the location group indicator G_L. The location claim is sent to the key server via the currently contacted or associated access point, AP_1 in Fig. 4 as shown in Step (4). The purpose of sending MS and its encrypted form $E_k(MS)$ is to show the key server that the MS has derived the location claim key, and can encrypt the plaintext using the key. A more complicated location claim is to include the BSS time stamp and location group indicator G_L in the encrypted message so as to defend against the message-replay attack.

In Step (5), the key server exams the location claim of the MS by deriving the same location claim key k. According to the location claim tuple, the key server first retrieves the private keys $\{X_1, X_2, X_3\}$ of the location group G_L, which were generated in Step (1) originally. Then the key server computes k using

$$k = Y_{MS}^{X_1+X_2+X_3} = g^{X_{MS} \cdot (X_1+X_2+X_3)} \ (\texttt{mod } p),$$

and encrypts MS using the key k. If the result is the same as $E_k(MS)$, the key server asserts that the MS has received the location keys of the location group G_L, and is located in the corresponding access-granted area. Therefore, the MS location claim is authenticated. Otherwise, the location claim is wrong, and LBAC access control process stops here.

In case that the MS location claim is authenticated, the key server will generate the encryption key for data communication between the MS and its corresponding AP. In Fig. 4, the corresponding AP is AP_1 for mobile station MS, which is easily derived from who forwarded the location claim.

3.2.2   LBAC Key Generation Phase

Once the MS location claim is authenticated by the key server, the key server generates the shared key k' for the AP-MS communication by again using the Diffie-Hellman algorithm. In this case, the public key belongs to the MS, and the private key belongs to the AP contacted by the MS. That is, the shared key k' between MS and AP_1 in Fig. 4 is

$$k' = Y_{MS}^{X_1} = g^{X_{MS} \cdot X_1} \ (\texttt{mod } p).$$

Then in Step (6) of Fig. 4, the key server directly sends the key k' to AP_1, along with an OK message for the AP to communicate with MS.

The same key k' has already been derived by the MS in Step (3) of Fig. 4 when MS receives the three location keys $\{Y_1, Y_2, Y_3\}$ from the APs by

$$k' = Y_1^{X_{MS}} = g^{X_1 \cdot X_{MS}} \ (\texttt{mod } p).$$

Therefore, the MS and its corresponding AP can use the symmetric key k' for secure data communication later.

3.3   Stronger Location Authentication

Although the capability to gather all the location keys of a location group provides strong evidence that an MS is located with the corresponding access-granted areas, there are still possibilities that the MS strands

outside the access-granted area, but is still able to communicate with the associated AP. In this case, the key server has no means to detect the errand but to ask the MS iteratively connect with all the APs of the location group. This way, the connectivities between the MS and all the APs will be tested out in order to authenticate the true location of the MS.

The iteratively associating with different AP can be achieved by the key server instructing the current associated AP to explicitly disassociate with the MS, and disallow the MS to associate with the same AP again shortly after. Such policy forces the MS to connect with other APs available in the location group. However, such physical location authentication may cause unnecessary control overhead, and could interrupt on-going communication sessions unexpectedly.

The location authentication can be further improved by renewing the location keys of the location groups periodically, thus forcing the MSs to reauthenticate themselves with the key server, and derive new shared keys for data communication purpose.

4   Evaluations

4.1   Efficiency Estimation

We have presented a novel protocol based on location groups and location keys to verify the location claims of mobile stations in WLAN network systems. Our location-based access control (LBAC) protocol does not require specific hardware such as GPS or ultrasonic devices, therefore avoiding the installation of sensors throughout buildings. Therefore, LBAC is able to locate objects and authenticate objects' locations inside or outside buildings efficiently with no additional hardware cost.

Other major performance concerns with LBAC for access control are the storage, computation and communication complexities.

With regard to the storage complexity, we consider the three elements of a typical access controlled WLAN systems.

- The *key server* is required to store all the private keys of the APs in the WLAN system for the Diffie-Hellman algorithm computations, as well as the location group information for access control purposes, from which all other information can be derived, such as the location keys of APs and location claim keys. Therefore, the space requirement for the key server is linear to the number of APs and location groups in the WLAN system.

- The *AP*s are required to store their respective location keys and the shared key between the AP and their associated MSs. Therefore, the storage requirement for each AP is constant for its location key, and linear to the number of MSs associated with the AP.

- The *MS*s are required to store the location keys of their respective location groups, and the shared key between themselves and their associated APs. Therefore, the memory space requirement for the MSs is linear to the size of the location groups that the MSs belong to.

With regard to the computation complexity for location-based access controls, LBAC exerts no computation overhead for the APs, but to the key server and the MSs. The *key server* is required to authenticate each MS of the WLAN system, therefore its computation task increases linearly with the number of MSs in the system. On the other hand, the *MS*s has constant computation overhead in each location group.

With regard to the communication complexity, the complete LBAC protocol operations involve six steps as shown in Fig. 4, in which four messages went through the wireless interfaces, and five messages through the wired infrastructure network for authenticating the particular single MS. Therefore, the communication overhead is a product of the number of MSs and the average size of the location groups of the WLAN system in the basic LBAC operations.

Moreover, LBAC reduces the computation and communication overhead as compared with previous systems [17, 21, 28, 29] because LBAC adopts the Diffie-Hellman key distribution protocol, and does not require any pre-deployment phase for location authentication and key generation purposes.

4.2 Security Analysis

Man-in-the-middle attack is possible in Diffie-Hellman key exchange protocol, but is addressed extensively in the literature. Therefore, we analyze LBAC from different security concerns.

4.2.1 Wormhole attack

The simplest wormhole attack is usually staged by two colluding attackers, one of which intercepts the traffic on one side of the network, and tunnels the packets to the other attacker for replaying on another side of the network. The wormhole attack is very difficult to detect, since it can be launched without compromising any host, or the integrity and authenticity of the communication [11, 30, 31].
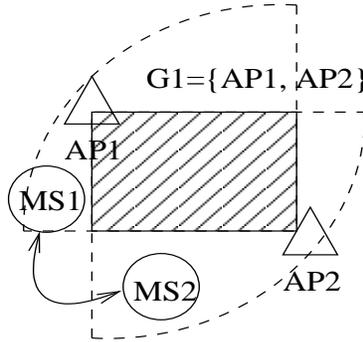

Figure 5: Wormhole Attack to LBAC.

In the LBAC system, wormhole attackers can forge legitimate location claims by gathering the location keys from different points of the network, even though none of the locations is authorized to access the network. For instance in Fig. 5, two mobile stations MS_1 and MS_2 can collect the two location keys of a location group $G_1 = \{AP_1, AP_2\}$ by exchanging the missing keys of each other, and successful access the network through the corresponding AP visible to each MS.

Another possible scenario is that an MS quickly moves between the coverage areas of the APs in a location group, thereby gathering all the location keys for location authenticating purposes.

As mentioned in Section 3.3, by physically and forcefully switching the associated APs of an MS, we can ensure that each access-granted MS can in fact communication with all the APs of the location group. In addition, by periodically changing the location key of each AP, we can also ensure that the location keys are renewed in time before the collusive MSs catch up with the old location keys.

4.2.2 Sybil Attack

In the Sybil attack [12, 15], an adversary impersonates multiple network entities by assuming their identities. Unlike the wormhole attack, the attacker is able to compromise communications, gain access to the cryptographic quantities, obtaining multiple node identities and injecting bogus data into the network [6].

In LBAC, Sybil attack is staged by a malicious MS impersonating an AP (Rogue AP) of a location group by broadcasting bogus location keys as if it were the legitimate access point. Sybil attack to the AP is usually prevented by authenticating the APs. For simplicity in LBAC, we have not provided such authentication mechanisms. However, LBAC can be easily extended to resist the Sybil attack by requiring the APs to include a PKI certificate of the location keys in the location-key broadcast messages. Assuming that the mobile stations have sufficient computational and communication capacities, the location keys can be easily verified by the MSs.

5   Summary

We have described and evaluated LBAC, a secure Location-Based network Access Control based on the location group and location key concepts using the Diffie-Hellman algorithm. LBAC does not depend on expensive location-detection hardware and infrastructure for location tracking and authentication purposes, and improves the protocol efficiency by eliminating the pre-installation phase. Compared to previous systems, LBAC provides the minimum communication and computational overhead, thus is a promising technique for network access control based on coarse-grain location information.

References

[1] IEEE Std 802.11. Wireless LAN Medium Access Controlf (MAC) and Physical Layer (PHY) Specifications. Technical report, IEEE, Jul. 1997.

[2] IEEE Std 802.11i/D3.0. Specification for Enhanced Security. Technical report, IEEE, Nov. 2002.

[3] IEEE Std 802.1x. Port Based Network Access Control. Technical report, IEEE, Jun. 2001.

[4] A. S. Tanenbaum. *Modern Operating Systems, Second Edition*. Prentice Hall, 2001.

[5] B. Waters and E. Felten. Proving the Location of Tamper Resistant Devices. Technical report, Princeton University, 2003.

[6] C. Karlof and D. Wagner. Secure Routing in Wireless Sensor Networks: Attacks and Countermeasures. In *Proceedings of IEEE International Workshop on Wireless Sensor Network Protocols and Applications*, 2004.

[7] D. Boneh and M. Franklin. Identify-based encryption from the weil-pairing. *SIAM Journal of Computing*, 32(3):586–615, 2003.

[8] D. M. Pozar and D. H. Schaubert. *Microstrip Antennas: The Analysis and Design of Microstrip Antennas and Arrays*. Wiley-IEEE Press, May 1995.

[9] W. Diffie and M. Hellman. New Directions in Cryptography. *IEEE Transaction on Information Theory*, 22:644–654, Nov. 1976.

[10] J. Edney and W.A. Arbaugh. *Real 802.11 Security: Wi-Fi Protected Access and 802.11i.* Addison-Wesley, 2003.

[11] I. Khalil, S. Bagchi, and N. B. Shroff. LITEWORP: A Lightweight Countermeasure for the Wormhole Attack in Multihop Wireless Network. In *the International Conference on Dependable Systems and Networks (DSN)*, Yokohama, Japan, 2005.

[12] J. Douceur. The Sybil Attack. In *Proceedings of IPTPS 2002*, Cambridge, MA, USA, March 2002.

[13] J. Hightower and G. Borriello. Location Systems for Ubiquitos Computing, August 2001. IEEE Computer Magazine.

[14] J. Kohl and B. C. Neuman. The Kerberos Network Authentication Service (Version 5), September 1993. Internet Request for Comments RFC-1510.

[15] J. Newsome, E. Shi, D. Song and A. Perrig. The Sybil Attack in Sensor Networks: Analysis and Defenses. In *Proceedings of IPSN 2004*, Berkeley, CA, April 2004.

[16] David Kotz Kwang-Hyun Baek, Sean W. Smith. A survey of wpa and 802.11i rsn authentication protocols. Technical report, Dartmouth College Computer Science, 2004.

[17] L. Lazos and R. Poovendran. SeRLoc: Secure Range-Independent Localization for Wireless Sensor Networks. In *2004 ACM workshop on Wireless security (ACM WiSe 2004)*, Philadelphia, PA, 2004.

[18] M. Balazinska, H. Balakrishnan, and D. Karger. INS/Twine: A Scalable Peer-to-Peer Architecture for Intentional Resource Discovery. In *Proceedings of the First International Conference on Pervasive Computing*, pages 32–43, August 2002.

[19] R. C. Merkle. Secure Communication over an Insecure Channel. *Communication of ACM*, 21:294C99, Apr. 1978.

[20] N. B. Priyantha, A. Chakraborty, and H. Balakrishnan. The Cricket Location-Support System. In *6th ACM International Conference, Mobile Computing and Networking (ACM MOBICOM)*, August 2000.

[21] N. Malhotra, M. Krasniewski, C. Yang, S. Bagchi, and W. Chappell. Location Estimation in Ad-Hoc Networks with Directional Antennas. In *25th International Conference on Distributed Computing Systems (ICDCS 2005)*, 2005.

[22] N. Michalakis. PAC: Location Aware Access Control for Pervasive Computing Environments. Technical report, MIT Laboratory of Computer Science, 200 Technology Square, Cambridge MA, 02139 USA, 2002.

[23] N. Sastry, U. Shankar and D. Wagner. Secure Verification of Location Claims. In *Proceedings of ACM Workshop on Wireless Security (WISE 2003)*, 2003.

[24] P. S. L. M. Barreto, H. Kim, B. Bynn, and M. Scott. Efficient Algorithms for Pairing-Based Cryptosystems. *Advances in Cryptology - Crypto 2002*, Lecture Notes on Computer Science 2442:354–368, 2002.

[25] R. E. Smith. *Authentication: From Passwords to Public Keys.* Addison Wesley, 2002.

[26] R. M. Needham and M. D. Schroeder. Using Encryption for Authentication in Large Networks of Computers. In *Communication of the ACM*, pages 993–999, December 1978.

[27] RSA and Adtron. RSA SecureID. In *http://rsasecurity.agora.com /rsase-cured/guides/imp_pdfs/adtran_ace5.pdf*, June 2003.

[28] S. Capkun, L. Buttyan, and J. Hubaux. SECTOR: Secure Tracking of Node Encounters in Multi-hop Wireless Networks. In *Proceedings of SASN 2003*, Virginia, October 2003.

[29] U. Hengartner and P. Steenkiste. Proceedings of 9th ACM Symposium on Access Control Models and Technologies (SACMAT 2004). In *SACMAT*, pages 11–20, Yorktown Heights, NY, June 2004.

[30] Y. Hu, A. Perrig, and D. Johnson. Packet Leashes: A Defense Against Wormhole Attacks in Wireless Ad Hoc Networks. In *Proceedings of INFOCOM*, San Francisco, CA, USA, April 2003.

[31] Y. Hu, A. Perrig, and D. Johnson. Rushing Attacks and Defense in Wireless Ad Hoc Network Routing Protocols. In *Proceedings of ACM Workshop on Wireless Security (WISE 2003)*, October 2003.

[32] Y. Zhang, W. Liu, W. Lou, and Y. Fang. Securing Sensor Networks with Location-Based Keys. In *IEEE Wireless Communications and Networking Conference (WCNC'2005)*, New Orleans, LA, March 2005.