A Link Layer Security Protocol for Suburban Ad-Hoc Networks

Muhammad Mahmudul Islam, Ronald Pose, Carlo Kopp
School of Computer Science and Software Engineering, Monash University, Australia
{sislam,rdp,carlo}@csse.monash.edu.au

Abstract— **Networks using wireless links are more vulnerable to various security threats than a wired network since wireless transmissions are prone to interception by anyone within the transmission range. Therefore participating entities should incorporate robust systems to withstand the attacks as rigorously as possible. In this paper we have presented a possible framework of a link level security protocol (LLSP) to be deployed in a Suburban Ad-hoc Network (SAHN [1][2][3][4][5][6]). LLSP provides authentication, integrity assurance and encryption for ensuring security at the data link layer. We have analysed various security aspects of LLSP to validate its effectiveness. To determine LLSP's practicability, we have estimated the timing requirement for each authentication process. Our initial work indicate that LLSP can be a suitable link-level security service for an ad-hoc network similar to a SAHN.**

Index Terms– Ad-Hoc Network, SAHN, Security, Authentication, Encryption, LLSP

I. Introduction

Wireless networks are more vulnerable to various security threats than their wired counterparts. Due to the nature of the wireless medium, security violation may occur over extended periods. While sophisticated security schemes may be able to prevent attacks from malicious nodes completely, it would be too expensive and hence impractical to implement. Alternatively a less sophisticated, yet robust, security scheme is feasible which is capable of preventing intrusions to a certain extent, e.g. prevent the damage caused by attackers from spreading through the entire network. This paper proposes a robust link layer security scheme to be employed in a SAHN without compromising the overall network performance.

Since SAHN is a quasi-static (i.e. nodes are not mobile) ad-hoc network, it is possible to implement more sophisticated security features that would otherwise be infeasible if nodes were mobile. Taking this into consideration, customized security protocols should be designed to provide the necessary security services for end-to-end as well as for per link communications. LLSP is responsible for ensuring security service for each link in a SAHN. Since the network layer is built on top of the link layer, routing becomes available only to legitimate nodes. LLSP in a SAHN provides the following features in order to address weaknesses in existing solutions: (a) Provision for security for each link, (b) Reduction in communication overhead, (c) Scaling properly with change in network topology and (d) Independent of a central administrator.

We have organised this paper as follows. In Section II, we have done some background studies which indicate the necessity of LLSP. In Section III we have described the mechanisms of LLSP. We have analyzed the robustness of LLSP against possible attacks in Section IV. Furthermore, we have estimated the timing requirement of the authentication process of LLSP in Section V. Finally we have concluded our paper with future research directions.

II. Background Studies

The Wired Equivalent Privacy (WEP) protocol of the IEEE 802.11b is based on a one-way authentication scheme using smaller sized shared keys. It also lacks of any key management protocol. Key management is necessary to prevent nodes from reusing keys so that intruders can not get enough time to break in. These vulnerabilities have led WEP to both shared key and man-in-the- middle attacks. The IEEE 802.11i (or its subset WPA) has been introduced as an improvement over WEP. The IEEE 802.11i relies on a trusted third part authentication server for its two-way authentication scheme [7]. Provision for authentication servers may not be possible in an ad-hoc network.

Research has been conducted to secure the route discovery process. Binkley and Trost[8] have integrated the link-level authentication with Mobile-IP. MAC and IP addresses of the sender are augmented with ICMP Router Advertisement packets and authenticated using similar mechanisms used in the UDP registration scheme of Mobile-IP. However, it is not impossible to spoof a MAC address since some LAN controllers (e.g. WaveLAN controllers) can be configured to use an arbitrary MAC address. Though an attacker is unable to get responses to its packets, it may still have its packets routed on to other nodes. Additionally, every member of the network has to know the network authentication key to join the network. Once a secret in known to all, it can not be regarded as a secret anymore. Moreover, this protocol is responsible only for protecting routing connectivity. Packet confidentiality is dependant upon security services from upper layers (e.g. IPSec) which are done usually on an end-to-end basis. SAR (secure aware ad-hoc routing protocol)[9] uses a negotiable metric to discover routes securely. An intermediate node processes or forwards a RREQ/RREP if and only if the transmitting node has the required authorization, i.e. level of trust, to provide the required service. However, it is unclear how this protocol can handle other types of packets, such as broadcast packets, coming from malicious nodes. Sanzgiri et al.[10] have proposed a secured routing protocol (known as ARAN) for ad-hoc networks. Since ARAN requires a trusted certification authority, a single point of failure (due to system faults or compromise) may expose the whole network. Signing each RREQ or RREP by intermediate nodes increases the size of the routing message at each hop. If clocks are not synchronized, the proposed system may become less effective. Papadimitratos and Haas[11] have proposed a proactive secure link state routing protocol (SLSP) that secures the discovery and the distribution of link state information

across participating mobile nodes in an ad-hoc network. It is robust against Byzantine behavior. However, like the secure link state routing protocols of wired Internet, SLSP relies on the distribution of all keys by a central authority and the reliable flooding of link state updates throughout the entire network. Kong et al.[12] describe a ubiquitous authentication service for mobile nodes by distributing the CA's functionality through a threshold secret sharing mechanism (e.g. [13]) to each local neighborhood. Since each authentication service requires a coalition of K nodes, this protocol is well suited for non-real-time events such as authenticating new nodes, updating session keys etc. However, authenticating every packet in intermediate nodes may not be practical for real-time traffic.

Like IPSec, Venkatraman and Agrawal[14] provide an end-to-end data authentication scheme for mobile nodes. Their proposal focuses on cluster based networks in order to reduce replay attacks. Since a session key is negotiated for each TCP session, this scheme may not be feasible for interactive and real-time traffic. Moreover, the end-to-end authentication mechanism ignores the possibility of replay attacks in intermediate cluster heads. This may enable malicious nodes to use the network as a carrier of their messages.

LLSP does not need to rely on any trusted third part authentication server, MAC-IP binding and central CA. It does not require any clock synchronization to perform effectively. Unlike [9], LLSP provides the authentication service for all types of packet (see Section III). It is not dependent upon any particular routing protocol and does not require flooding of any information throughout the whole network. It does not suffer from initial setup delay before each session like IPSec and [14]. Moreover, the watchdog module of LLSP (see Section III) can prevent a SHAN from flooding with excessive traffic coming from authorized but non-cooperative nodes and provide link-level encryption service on-demand for added security.

III. Link Layer Security Protocol (LLSP)

LLSP is responsible for authentication and encryption in the link layer. Authentication enables the receiver of a digital message to be confident of the identity of the sender. It also provides for guaranteeing the integrity of information. On the other hand, encryption ensures that the transmitted information are readable only by authorized recipients. To enhance the security feature of the link layer, the Watchdog module of LLSP monitors channel usage of each neighbor and informs the MAC layer to take necessary steps for misbehaving neighbors. The security services provided by LLSP can be classified into five types: (Type 1) Authenticating a new node, (Type 2) Updating the capability[1](CAP) of a link, (Type 3) Updating the shared key (SHK) of a link, (Type 4) Authenticating received packets and (Type 5) Encrypting payload.

The once off authentication in Type 1 and the periodical renewal of CAP and SHK in Type 2 and Type 3 respectively rely on a digital signature mechanism using asymmetric cryptosystem such as RSA [18] cryptography. For authenticating regular packets (i.e. packets not used in Type 1-3) in Type 4, LLSP applies symmetric cryptography, e.g. AES (Advanced Encryption System), on a sequence number (SEQ) and the checksum of given message. Encryption in Type 5 is accomplished by symmetric cryptography (e.g. AES) using the periodically updated SHK. The provision for encrypted sequence number and checksum provides better protection against replay attacks.

[1]SAHN is based on a capability-based platform. A capability is a token that not only identifies an object/resource but also authorizes its use [15]. A password-capability model is employed in [16][17].

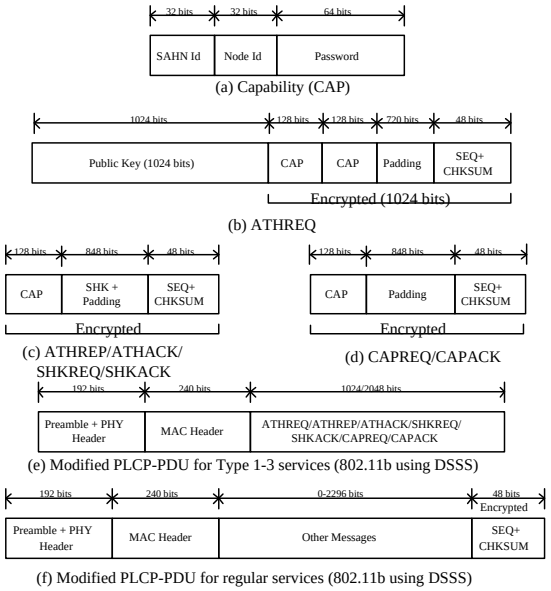

Fig. 1. Various packet formats.

• **Authenticating a new node:** Unlike base stations and mobile phones, any member of a SAHN having an authenticating capability can authorize a new node to join the network. If a new node N is known to one of its neighbors O, O will generate a capability $\mathrm{CAP_{NO}}$ to be used over the link (N-O). If N is unknown to any of its direct neighbors, it can request a known member K to generate a capability from O. Since O and K belong to the same network, O must respond to N's request via K. Now O will create $\mathrm{CAP_{NO}}$ and send it to N through K. The process of distributing a capability is accomplished by an out-of-band capability distribution service.
Once N gets $\mathrm{CAP_{NO}}$ and $\mathrm{PBK_O}$ (O's public key), it creates a $\mathrm{CAP_{ON}}$ to be used in the link (O-N). Now N encrypts $[\mathrm{CAP_{NO}} + \mathrm{CAP_{ON}}]$ using $\mathrm{PBK_O}$. Then it transmits an ATHREQ packet containing the ciphertext and $\mathrm{PBK_N}$. It expects an authentication reply (ATHREP) packet from O within time ATHREP_T.
If O receives an ATHREQ, it retrieves $\mathrm{CAP_{NO}}$ and $\mathrm{CAP_{ON}}$ using its private key $\mathrm{PVK_O}$. If $\mathrm{CAP_{NO}}$ is valid, O sends a $\mathrm{SHK_{ON}}$ along with $\mathrm{CAP_{ON}}$ to N encrypted with $\mathrm{PBK_N}$ and waits for an acknowledgement (ATHACK) until timeout ATHACK_T. ATHACK is expected to contain $[\mathrm{CAP_{NO}} + \mathrm{SHK_{ON}}]$ encrypted with $\mathrm{PBK_O}$.
• **Updating capability and shared key:** LLSP updates the CAP/SHK of each link regularly. This process prevents intruders from getting enough time to perform cryptanalysis on eavesdropped packets to decipher associated CAP/SHK.

When a member N wants to renew the existing CAP_{ON}, it updates the password field (Figure 1(a)) and transmits $\text{E}_{\text{PBK}_\text{O}}[\text{CAP}_{\text{NO}} + \text{CAP}_{\text{ON}_{\text{new}}}]$ in a capability-key-update-request (CAPREQ) packet. Now N waits for an acknowledgement (CAPACK) containing $\text{E}_{\text{PBK}_\text{N}}[\text{CAP}_{\text{ON}_{\text{new}}}]$ and ignores any CAPACK arriving after the timeout limit CAPACK_T.

Updating a shared key is similar to the previous process except that the shared-key-update-request (SHKREQ) contains $\text{E}_{\text{PBK}_\text{O}}[\text{CAP}_{\text{NO}} + \text{SHK}_{\text{NO}_{\text{new}}}]$, the acknowledgement (SHKACK) contains $\text{E}_{\text{PBK}_\text{N}}[\text{CAP}_{\text{ON}} + \text{SHK}_{\text{NO}_{\text{new}}}]$ and the timeout limit is set by SHKACK_T.

• **Authenticating received packets:** LLSP ensures that each received packet, except the control packets required for accessing the channel, comes from an authentic source. One way to achieve this is to use monotonically increasing sequence numbers (SEQs), a one-way checksum of the message (CK(MSG)) and an encryption service. A SEQ is used to verify correct sequence of packets received. Combining a one-way checksum with each SEQ enables detection of replay attacks. This can only work if the authentic transmitter/receiver knows the next possible number in the sequence and the [SEQ+checksum] pair is transmitted encrypted. This is how this method works. Let us assume that N want to send a message (MSG) to O. N calculates the one-way checksum CK(MSG) of the message and appends a sequence number SEQ_N to it. If the message belongs to Type 1-3, $[\text{SEQ}_\text{N} + \text{CK(MSG)}]$ is appended to the message (Figure 1(b-d)) and the resultant payload is encrypted with PBK_O. Otherwise (i.e. in Type 4) a digital signature (Figure 1(f)) is computed by encrypting $[\text{SEQ}_\text{N} + \text{CK(MSG)}]$ with SHK_{NO}. When O receives a packet from N, it applies $\text{PVK}_\text{O}/\text{SHK}_{\text{NO}}$ to the message/signature part to yield SEQ_N and CK(MSG). If SEQ_N is greater than the previously received sequence number from N and the checksum computed on MSG agrees with the received CK(MSG), N can assume that the message has not been tempered with. Otherwise N suspects that the message has been forged by a malicious node.

• **Encrypting payload:** LLSP may be required to provide a link level encryption service to some messages coming from its upper layers or from other members. Before sending a message through a link, the sender encrypts the message using the link's SHK. It also calculates the encrypted checksum of the encrypted message using the method described in the previous section. Now the transmitted packet contains the encrypted message and checksum. The encryption process hides the original message from intruders and the authentication mechanism ensures that the sender is authentic. SHK can be periodically updated between the corresponding nodes to minimize the chance of being hacked.

• **LLSP Watchdog:** The LLSP Watchdog keeps track of what percentage of the total available bandwidth is being consumed by each neighbor member. Should a neighbor exceed its usage limit[2], the LLSP Watchdog informs the MAC protocol to ignore any more incoming packets from that neighbor until it behaves rationally.

IV. Analysis of Potential Attacks

LLSP is able to address the following security attacks, common in an ad-hoc wireless environment:

• **Identity theft:** Intruders steal identity of an existing member of the network and hence can gain access to the network. Possibility of identity theft during out-of-the-band capability distribution process is very low since it does not involve any insecure communication medium.

• **Man-in-the-Middle attacks:** The cryptographic algorithms and their associated encryption/decryption keys used in LLSP are expected to be robust enough to not to be breakable by any cryptanalysis within any feasible time period [20]. However if an intruder is somehow successful, the retrieved authentication key (i.e. capability and encryption key) may timeout by that time since LLSP updates the authentication key of each link regularly.

• **Denial-of-Service (DoS) attacks:** If a malicious node or an unlawful member somehow establishes a connection with a remote member and tries to jam the associated links with unnecessary traffic, the LLSP Watchdog module in one of the members along the affected path will notify[3] the MAC protocol to discard incoming packets over the compromised links. Hence the affected links can be prevented from excessive flooding which could otherwise refrain other members from getting legitimate services.

• **Replay attacks:** LLSP is robust enough to guard against replay attacks targeted at the link layer. An intruder may modify an eavesdropped packet and then replay to the receiver. Since each packet is tagged with an encrypted pair of monotonically increasing sequence number and the checksum of the message, the modified packet will not pass the integrity check and hence be ignored by the receiver. If the intruder wishes to replay the received packet unmodified at a later time inorder to create other problems (e.g. replaying a stale state information may create inconsistency in the network), the LLSP module at the receiver will ignore it as the received sequence number will be out of order by that time.

V. Duration of Authentication Process

The authentication processes in Type 1-3 require encryption and decryption using the RSA method. The time required for encryption and decryption, using the RSA scheme, depends on the size of the key and the message. Each RSA key in the SAHN link layer is 1024 bits long and applied to a block of the same size (see Figure 1(b-d)). Current hardware implementations of RSA can decrypt a 1 Kb block in less than 10 ms [21][22][23][24]. Using similar techniques we can expect to decrypt a 1 Kb message, containing CAP or both CAP and SHK, within 10 milliseconds. For simplicity we can assume that encryption process also takes 10 ms. So the total time needed for encryption and decryption in Type 1 and Type 2-3 are $6 \times 10 = 60$ ms and $4 \times 10 = 40$ ms respectively.

[2]This limit can be set and updated by a sophisticated fairness scheme (e.g. [19]) implemented at the MAC layer.

[3]Provided that the associated incoming link exceeds its usage limit.

Authentication Type	Transmission rate(Mbps)	Total Duration(ms)
Type 1	1	69.86
	2	67.244
	5.5	65.578
	11	65.102
Type 2, Type 3	1	45.922
	2	44.504
	5.5	43.6
	11	43.342

TABLE I

DURATION OF AUTHENTICATION PROCESS IN TYPE 1, TYPE 2 AND TYPE 3 USING IEEE 802.11b.

Adding the times taken for encrypting, decrypting, accessing the channel (using data from [25]), getting a link layer acknowledgement and receiving PLCP-PDU gives the total time needed in each authentication process. Table I summarizes the result for Type 1-3. Since nodes are static, authenticating new nodes and updating SHKs will not be needed as frequently as in a mobile network. Therefore, authentication overheads in Type 1-3 should not have any noticeable adverse effect on normal traffic flow.

A hardware implementation of a symmetric encryption system, such as an AES, can achieve a Gb/s throughput [26][23][24]. This is sufficient enough to cope with the transmission rates of existing wireless technologies. Hence computing a digital signature and encrypting/decrypting payloads using AES should not unduly degrade the maximum throughput achievable considering the delays incurred at the link and physical layers.

VI. CONCLUSION

We have presented a link layer security protocol suitable for an ad-hoc network similar to a SAHN. LLSP does not depend on a single entity to issue and distribute capabilities. It requires each member to store and compute security information for a limited number[4] of one-hop neighbors. Hence LLSP scales properly with network size. Provision for authentication, integrity assurance and encryption for each link prevents a malicious node from spreading its damage through the entire network. Security services of other layers (e.g. end-to-end authentication service of application layer) may be deployed on top of LLSP to enhance overall security. At present we are trying to find the communication overhead of LLSP in a simulated environment using GloMoSim. In future we plan to implement an efficient and decentralized capability distribution protocol. We will conduct more research to enhance the effectiveness and robustness of LLSP. Furthermore, we would like to integrate LLSP with channel access mechanisms of other wireless technologies (e.g. IEEE 802.16) and measure performance.

REFERENCES

[1] R. Pose, C. Kopp. Bypassing the Home Computing Bottleneck: The Suburban Area Network. *3rd Australasian Computer Architecture Conference*, pages 87–100, Feb 1998.

[2] M. M. Islam, R. Pose, C. Kopp. Efficient Routing in Suburban Ad-Hoc Networks (SAHN). In *CIC*, pages 188–194, June 23-26 2003. Las Vegas.

[3] M. M. Islam, R. Pose, C. Kopp. Routing In Suburban Ad-Hoc Networks. *International Conference on Computer Science and its Applications(ICCSA 2003)*, pages 72–76, July 1-2 2003. San Diego , California , USA.

[4] M. M. Islam, R. Pose, C. Kopp. A Hybrid QoS Routing Strategy for Suburban Ad-Hoc Networks. In *ICON*, pages 225–230, Sep 28 - Oct 1 2003. Sydney, Australia.

[5] M. M. Islam, R. Pose, C. Kopp. A Router Architecture to Achieve Link Rate Throughput in Suburban Ad-Hoc Networks. *A Omondi and S Sedukhin (eds), Lecture Notes in Computer Science, Springer-Verlag, Berlin, Germany*, pages 395–407, 2003.

[6] M. M. Islam, R. Pose, C. Kopp. Multiple directional antennas in suburban ad-hoc networks. In *ITCC*, pages 385–399, Apr 2004.

[7] B. Brown. 802.11: the security differences between b and i. *IEEE Potentials*, 22(4):23–27, Oct-Nov 2003.

[8] Jim Binkley and William Trost. Authenticated ad hoc routing at the link layer for mobile systems. *Wirel. Netw.*, 7(2):139–145, 2001.

[9] Robin Kravets Seung Yi, Prasad Naldurg. Security-aware ad hoc routing for wireless networks. In *ACM Symposium on Mobile Ad Hoc Networking and Computing*, 2001.

[10] Kimaya Sanzgiri, Bridget Dahill, Brian Neil Levine, Clay Shields, and Elizabeth M. Belding-Royer. A secure routing protocol for ad hoc networks. In *Proceedings of the 10th IEEE International Conference on Network Protocols*, pages 78–89. IEEE Computer Society, 2002.

[11] Zygmunt J. Haas Panagiotis Papadimitratos. Secure link state routing for mobile ad hoc networks. In *SAINT*, pages 27–31, Jan 2003.

[12] Haiyun Luo Songwu Lu Lixia Zhang Jiejun Kong, Petros Zerfos. Providing robust and ubiquitous security support for mobile ad hoc networks. In *Proceedings of the Ninth International Conference on Network Protocols (ICNP'01)*, pages 251–260. IEEE Computer Society, Nov 2001.

[13] Y. Desmedt and Y. Frankel. Shared generation of authenticators and signatures. In *Advances in Cryptology - CRYPRO*, pages 457–469, 1991.

[14] Lakshmi Venkatraman and Dharma P. Agrawal. A novel authentication scheme for ad hoc networks wireless. In *WCNC*, volume 3, pages 1268–1273, 2000.

[15] R. S. Fabry. Capability-based addressing. *Commun. ACM*, 17(7):403–412, 1974.

[16] R D Pose. Password-capabilities: their evolution from the password-capability system into walnut and beyond. In G Heiser, editor, *ACSAC*, volume 23, pages 105–113. IEEE Computer Society Press, Jan 2001.

[17] M. D. Castro. The Walnut Kernel : a password-capability based operating system, 1996. Ph.D. Thesis, Monash University, Clayton, Australia.

[18] R. L. Rivest, A. Shamir, and L. Adleman. A method for obtaining digital signatures and public-key cryptosystems. *Commun. ACM*, 21(2):120–126, 1978.

[19] Raghupathy Sivakumar Karthikeyan Sundaresan, Hung-Yun Hsieh. Ieee 802.11 over multi-hop wireless networks: problems and new perspectives. *Elsevier*, pages 109–132, 2004.

[20] Arjen K. Lenstra and Eric R. Verheul. Selecting cryptographic key sizes. *Journal of Cryptology: the journal of the International Association for Cryptologic Research*, 14(4):255–293, 2001.

[21] Ho Won Kim and Sunggu Lee. Design and implementation of a private and public key crypto processor and its application to a security system. *IEEE Transactions on Consumer Electronics*, 50(1):214–224, Feb 2004.

[22] Y.C. Tekmen M. Askar S. Yesil, A.N. Ismailoglu. Two fast RSA implementations using high-radix montgomery algorithm. In *International Symposium on Circuits and Systems (ISCAS 04)*, volume 2, pages 557–560, May 2004.

[23] Corrent. http://www.corrent.com/Products/Product_chip.htm.

[24] Infineon Technologies. http://www.infineon.com.

[25] Mihail Sichitiu Jangeun Jun, Pushkin Peddabachagari. Theoretical maximum throughput of ieee 802.11 and its applications. In *Second IEEE International Symposium on Network Computing and Applications*, pages 16–18, Apr 2003.

[26] S. Morioka and A. Satoh. A 10-gbps full-aes crypto design with a twisted bdd s-box architecture. *IEEE Transactions on Very Large Scale Integration Systems*, 12(7):686–691, July 2004.

[4] A SAHN allows a limited number of links per node[1].