Introducing smartcards for Wireless LAN security

Pascal Urien[1], Marc Loutrel[1], Karen Lu[2]

Abstract

Wireless LANs based on the IEEE 802.11b have emerged as a new standard but still a lot of security issues remain. One of them issues is the authentication of a terminal to an access point (AP). The IEEE is currently finalizing a framework for security in IEEE 802 networks: IEEE 802.1x that is an approved draft, based on the extensible authentication protocol (EAP). On another hand, among token technologies, smartcards currently offer the best combination of flexibility, security, and cost among token technologies. They have been chosen by the DoD to be widely deployed. Therefore it becomes difficult to speak about security without evoking smartcards. We have focused our work on the integration of smartcards for authentication in wireless networks. In this paper, we will present a smartcard embedding EAP and will show that smartcards could constitute the de-facto device for authentication in Wireless LAN as they have been for GSM and will be for UMTS.

1. Introduction

The explosive growth of 802.11b networks has been possible because of the convenience and flexibility that the wireless network provides and the simplicity to install. However 802.11b networks' weaknesses in security mechanisms present a major problem for many companies adopting the network. Most access points have not even turned on the encryption algorithm.

At the same time a growing number of important private and public sectors have adopted smartcards, which are secure tokens capable of authenticating a person and storing personal information. There are many smartcard applications including bank payment cards, government smartcard rollouts, mobile telephony (USIM), and E-Commerce. It is likely that smartcards will constitute a significant component of most I.T. systems in the future.

[1] {purien,mloutrel}@slb.com, SchlumbergerSema 36-38 Rue de la Princesse BP45 78431 - Louveciennes Cedex

[2] karenlu@slb.com, Austin Product Center, Schlumberger, 8311 North FM 620 Rd,Austin, TX 78726,USA

In this paper, we introduce smartcard for solving some security issues in wireless LANs. The paper first reviews some basics information on smartcard technology. It also reviews the authentication in WLANs in the first and second generations. Then we present a new type of smartcards dedicated to authentication in WLANs.

2.Smartcard technology

A smartcard is a portable and tamper-resistant computer. It holds sensitive data and is capable for data processing. Smartcards provides data security, data integrity, personal privacy and mobility. Applications of smartcards are widespread and rapidly increasing. Major application areas include mobile communication to convey user subscription and identification information; financial services to convey identity and account information and to provide services for credit, debit and cash; and IT environment to provide user identity, building and office access, and computer and network access.

A smartcard consists of common physical elements of a computer i.e. a microprocessor, memory, a secure communication bus, and an I/O connector. Current smartcards' CPU is based on a quite basic 8-bit microprocessor. Nevertheless new types of 32-bit microprocessors offer computational capability up to 30Mips at a frequency of 33MHz. Moreover, to improve smartcards performance crypto-coprocessors may be replaced by dedicated cryptographic instruction in CPU cores [1]. Smartcards will be able to process key with larger lengths and to generate asymmetric keys in an acceptable delay.

Smartcard memory includes a ROM, a RAM, and an EEPROM (Electrically Erasable Programmable Read-Only Memory). The EEPROM memory can be erased and programmed a finite number of times and is a persistent memory offering storage capability. Entry-level memory capacities are 64 to 128Ko of ROM memories, 32 to 64Ko of E2PROM and 4Ko of RAM. Writing data in E2PROM is relatively slow. It takes 1ms to write a 32 or a 64-byte word and it can be done only a million of time.Time to access these memories was quite painful but it's no longer an issue with new memories such as FeRAM (10^9 writing operations allowed, memory capacity around one Mo and writing delay less than 200ns). Performance of components will be crucial for smartcards future, as they would be considered as the most secure communication token on the Internet or will be limited to more classical applications.

Hopefully, as announced by manufacturers smartcards with few Mo of memories will come out by 2003.

Until recently smartcards weren't able to communicate without a smartcard reader (Card Acceptance Device) through an ISO 7816 interface. The ISO 7816 standard offers a maximum data rate of 230,400 bauds in a half duplex mode through a serial link. However, most chips are limited to the speed of 9600 bauds. USB smartcards [2] provides direct connection and dramatically improve transfer rates. Evolution of smartcards' components is summarized in table 1.

	Currently used by smart cards	Available in labs	Under development
Memories	EEPROM, SRAM, ROM	FLASH	FRAM
Cores	8-bit, 16-bit, 32-bit		
Coprocessors	Cryptographic	DSP, Baseband, MPEG	
Input/Output	ISO7816, ISO14443, USB, I^2C, LPC	SPI High Speed, IEEE1284, Bluetooth, Ethernet	

ST, 2001

Table 1 – Evolution of smartcards performance

2. Authentication in WLAN

A. Authentication in IEEE 802.11b

Two types of authentication methods have been defined in 802.11b: open and shared keys. The open authentication process is done in clear-text, and a terminal may associate with an access point even without supplying the correct WEP (Wired Equivalent Privacy) key. In shared-key authentication, the access point sends the terminal a challenge text packet that the terminal must encrypt with the correct WEP key and returns to the access point. Privacy protects data streams on wireless LAN by encrypting them and allowing decryption only with the correct WEP keys.

B. Flaws in the IEEE 802.11b

The shared key authentication is only a one-way authentication that allows an access point to authenticate a user. If a rogue access point is placed on a Wireless LAN, denial-of-service attacks may be launched.

It is common to assign a static WEP key to a terminal. Therefore, the owner of a terminal has possession of the terminal's MAC address and WEP key and can use those components to gain access to the Wireless LAN. If multiple users share a terminal, then these users effectively share the MAC address and WEP key [3, 4]. If a terminal is lost, the intended user or users of the terminal do no have longer access to the MAC address or WEP key, but an unintended user does.

WEP supports per-packet encryption but not per-packet authentication. A hacker can reconstruct a data stream from responses to a known data packet and then can spoof packets.

Besides WEP is based on RC4 encryption algorithm that provides fast symmetric block ciphering suitable both for hardware and software implementations. New weaknesses in this algorithm have recently been reported [5].

C. Security in the next generation of WLAN

IEEE 802.1X [6] is a new security framework that has been released recently. It concerns all types of LANs technologies. Of course IEEE 802.11 enters in the scope of this standard.

Corporate LANs are often permitted unauthorized devices to be physically attached to the LAN infrastructure providing connectivity. It is desirable to restrict access to the services offered by the LAN to some users and terminals. Port based network access control makes use of the physical access to provide a mean of authenticating and authorizing devices attached to a LAN port that has point-to-point connection characteristics. The use of port authentication includes associations between stations and access points in IEEE 802.11 Wireless LANs.

The architecture of IEEE 802.1X defines three main components: the Supplicant, the Authenticator, and the Authenticator Server. The Supplicant is a port configured to access the services offered by the Authenticator's system. The Authenticator is a port configured to enforce authentication and authorization before allowing access to services that are accessible via that port. The Authentication Server performs the authentication function necessary to check the credentials of the Supplicant on behalf of the Authenticator, and indicates whether or not the Supplicant is authorized to access the Authenticator's services. Typically in 802.11, the terminal plays the role of the Supplicant, the access point plays the role of the Authenticator and a RADIUS [7] server will be the Authenticator Server.

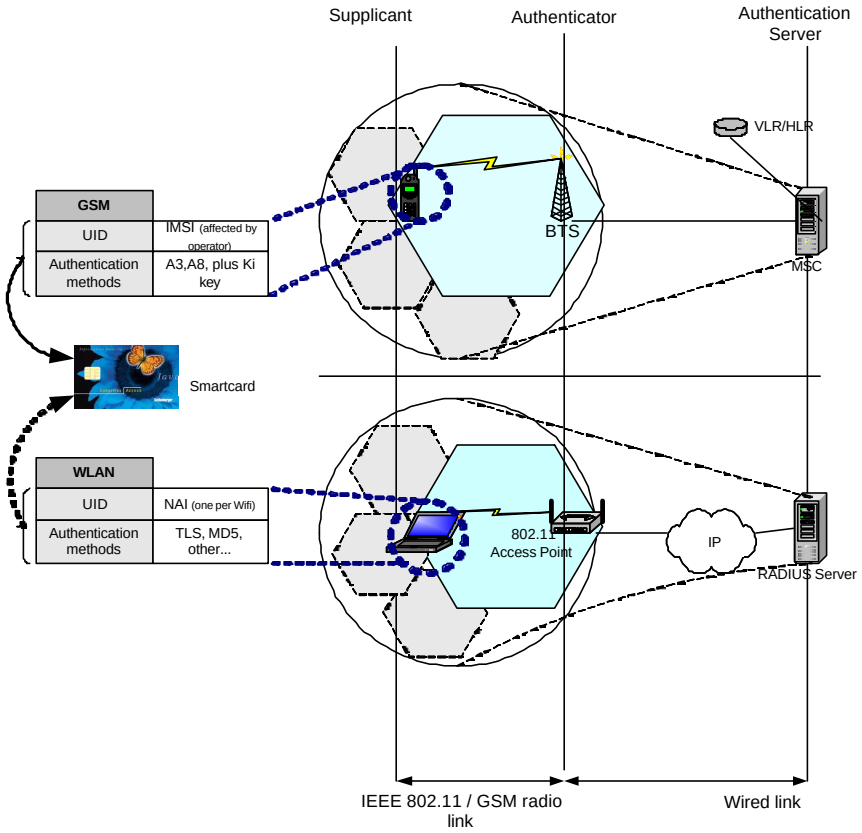

Figure 1 Comparaison between WLAN and GSM architecture.

IEEE 802.1X defines the encapsulation techniques that shall be used in order to carry **E**xtensible **A**uthentication **P**rotocol (EAP) packets [8] between Supplicant **P**ort **A**ccess **E**ntities (PAEs) and Authenticator PAEs in a LAN environment. The encapsulation is known as EAP over LANs. EAP is a general protocol for authentication that supports multiple authentication methods, such as token cards, Kerberos, one-time passwords, certificates, public key authentication and smartcards. It will also easily allow adoption of new authentication protocols.

Nevertheless session hijacking and man-in-the-middle are two flaws in the design of 802.1X that have already been identified and operationally verified [9]. The 802.1X standard has been amended and will hopefully better match requirements of the 802.11 to ensure a safe deployment. Corporate information is too sensible to be exposed so WLAN security will be a key issue to ensure a wide acceptance.

D. Comparison with GSM networks

The IEEE 802.1X will likely be the heart of authentication in next generation of WLAN; even though it is incomplete and is not exactly adapted to wireless networks yet [10]. Based on that assumption, we try to find similarities between WLAN and GSM networks as shown in Figure 1. Currently, WLANs choose RADIUS for remote authentication. A RADIUS server is the equivalent of the AuC (Authentication Center) in GSM architecture. To use radio resources in a visited GSM network, a user need to be authenticated. Thus before being added to the VLR (Visited Local Registration), a mobile phone will be authenticated by his HLR (Home Local Registration). WLANs may apply the same kind of procedures. A RADIUS server will act as a proxy and will forward an access request to the home RADIUS server (equivalent to an HLR).

In a centralized architecture, secret keys used for authentication have to be stored in a safe place. Because terminals are typically insecure and cannot be trusted, we may store sensible information, such as secret keys, in a tamper-resistant device, with which a terminal can communicate. This is exactly what happens in GSM networks. To authenticate a user to a GSM network, a SIM card (a kind of smartcard) stores personal information, such as user identity (IMSI number), and provides authentication mechanism using stored secret key and cryptographic functions. Similarly in WLANs keeping secret information in a safety-box should ensure security and privacy. Token technologies such as

smartcards offer great advantages and the best combination of flexibility, security, and cost among token technologies.

3. EAP Integration in smartcard

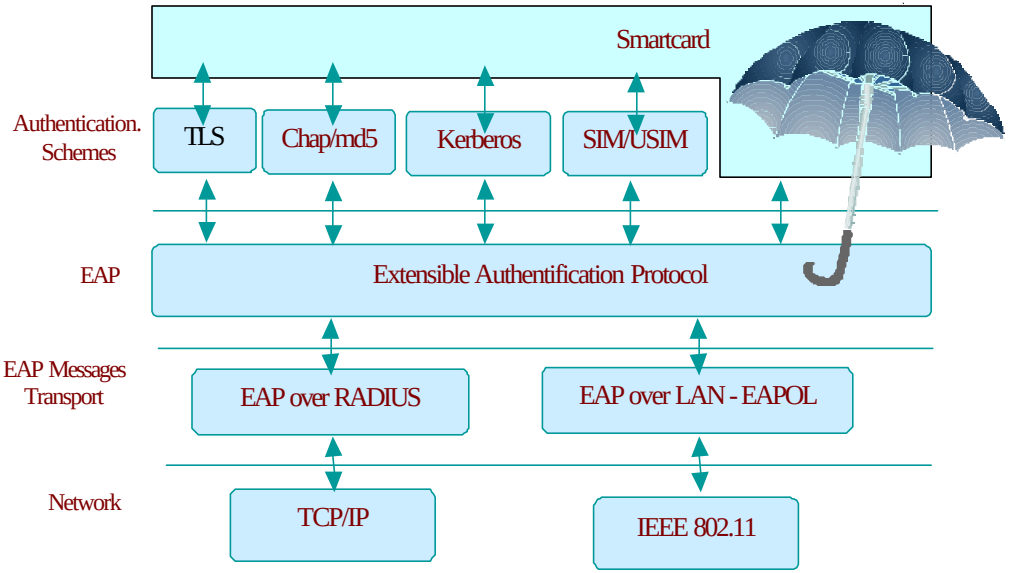

Figure 2. Overview of the EAP secure umbrella.

A. EAP overview

EAP [7] is a general protocol for authentication that also supports multiple authentication schemes, such as token cards, Kerberos, one-time passwords, certificates, public key authentication and SIM cards. As mentioned above, IEEE 802.1x specifies how EAP are encapsulated in LAN frames.

The EAP protocol (cf. figure 2) can support multiple authentication mechanisms without having to pre-negotiate a particular one. Authenticator do not necessarily have to understand each request type and may be able to simply act as a pass through agent for a "back-end" server on a host. The device only needs to look for the success/failure code to terminate the authentication phase.

EAP packets include all relevant information about the required authentication scheme, e.g. authentication method, packet type (request, response, success or failure) and/or challenge. The content of these packets is up to the chosen EAP authentication scheme. The progression of an authentication procedure depends also on the chosen authentication mechanism.

As EAP has emerged as a standard layer to support various authentication protocols, we propose to integrate it in smartcards.

B. EAP functions

An EAP packet consists of two parts,

- A header that includes the following parameters: Code, Identifier, Length and Type.
- A payload that includes information relative to a particular authentication scheme. The standard suggests organizing the payload with one byte prefix indicating the data length, followed by data bytes.

The *Code* field is one octet that identifies EAP packet functions. The *Code* has following values: 1 for request, 2 for response, 3 for Success, and 4 for failure. The *Identifier* is one octet and aids in matching responses with requests. The *Length* field is two octets and indicates the total length of the EAP packet. The *Type* field is one octet and identifies the structure of an EAP Request or Response packet. All EAP implementations support types 1-4, 1 for identity, 2 for notification, 3 for NAK (Response only) and 4 for MD5 challenge. The payload is zero or more octets. The code field determines the format of the payload.

The Network Access Identifier (NAI) is the identity parameter in wireless networks [11]. NAI is similar to an email address. It comprises two parts, a left element equivalent to a username and a right part, separated by the @ character, which indicates an authentication server name (for example superman@comics.com). To use a smartcard in several wireless networks with different NAIs, it is necessary to select an appropriate identity before any authentication operation.

A smartcard connects to a station (a laptop for example), which acts as a supplicant entity in the 802.1x architecture. The station forwards EAP requests and notification messages to the smartcard. The Smartcard produces responses

according to its predefined identity. An EAP authentication scenario begins by an Identity request message, followed by one or more request/response messages relative to fix authentication type. It normally ends by a notification indicating either Success or Failure. Keying materials could be deduced from these messages and stored in smartcard files. The station uses these keys for packet signing or enciphering.

We define a concept of EAP session that begins with a start method, which specifies an identity and optionally an authentication type. The scenario state machine is also reset.

EAP_Start(Identity,Type)

Conceptually EAP messages are sent to the smartcard, which returns a response by means of methods like,

$$Message_{Code,Identifier,Length,Type,Payload} = EAP\_Process(Message_{Code,Identifier,Length,Type,Payload})$$

Optional computed cryptographic key are obtained by the method below.

EAP_GetKey(Identity,Type,Key_Index)

C. API : Java Card Forum.

Javacard could implement a dedicated API in order to support EAP protocol. A network service, identified by an AID and implemented by a java application, could use these API to set the user identity, reset the EAP state machine, handle EAP messages, produce and store optional cryptographic keys.
This new WiFi service should be supported independently of the communication scheme (ISO 7816, non ISO protocols) that are under definition for the next generation of Java Cards 3.x.RMI [12] is another alternative, in a full java environment, for interfacing embedded EAP services.

4. Conclusion

We have demonstrated that smartcards can definitely be the ultimate secure device for authentication in WLAN.
Because security issues have not been solved yet, a business model in WLANs doesn't emerge.
Strong from the experience in GSM networks where smartcards are used for authentication, using smartcards in WLANs may considerably accelerate their deployments in companies.. We have recently presented the integration of EAP [13] in JavaCard technology and proposed an extension for JavaCard to the JavaCard Forum. We are currently working on an **A**pplication **P**rogramming **I**nterface for EAP in smartcard.

5. References

[1] Jean-François Dhem and Nathalie Feyt, "Hardware and software symbiosis helps smart card evolution", IEEE Micro p14-25, November/December 2001.
[2] SchlumbergerSema, e-gate, http://www1.slb.com/smartcards/infosec/egate.html.
[3] William A. Arbaugh, Narendar Shankar, and Y.C. Justin Wan, "Your 802.11 Wireless Network has No Clothes," In IEEE International Conference on Wireless LANs and Home Networks, Singapore, Dec 2001
[4] N. Borisov, I. Goldberg, and D. Wagner. "Intercepting Mobile Communications: The Insecurity of 802.11", In *Proceedings of the 7th InternationalConference on Mobile Computing and Networking*, July 2001 in Rome Italy.
[5] Fluhrer, S., Mantin, I., Shamir, A., "Weaknesses in the Key Scheduling Algorithm of RC4", Eighth Annual Workshop on Selected Areas in Cryptography, Toronto, Canada, August 2001.
[6] IEEE 802.1X specification (IEEE Standard), http://standards.ieee.org/reading/ieee/std/lanman/802.1X-2001.pdf
[7] RFC 2865, "Remote Authentication Dial In User Service (RADIUS)", June 2000.
[8] RFC 2284, "PPP Extensible Authentication Protocol (EAP)", March 1998.
[9] William A. Arbaugh, Arunesh Mishra, "An Initial Security analysis of the 802.1X standard", www.cs.umd.edu/%7Ewaa/1x.pdf.
[10] Implementing 802.1x on Wireless Networks with Cisco and Microsoft, http://www.cs.umd.edu/~mvanopst/8021x/howto/.
[11] RFC 2486, "The Network Access Identifier", January 1999.
[12] Sun Microsystem. "Java Card™ 2.2 Remote Method Invocation Design. Draft 1 Revision 1.1". May 07th 2001
[13] Pascal Urien, Adel Tizraoui, Marc Loutrel, "Integrating EAP in SIM-IP smartcards", ASWN IEEE workshop on Applications and Services in Wireless networks, July 2002, Paris.