

# A flexible, large-scale authentication policy for WLAN roaming users using IPsec and public key certification

André Zúquete  
IEETA/UA  
avz@det.ua.pt

Carlos Ribeiro  
CIIST/IST/INESC ID Lisboa  
carlos.ribeiro@ist.utl.pt

## Abstract

With the growing deployment of WLAN hot-spots there is a real need of a large-scale, easy-to-use authentication policy for enabling hot-spot providers to allow access to the Internet to authorized users. This paper presents an authentication schema based on asymmetric cryptography and public key certification. These mechanisms are used to establish IPsec tunnels between WLAN supplicants and gateways, providing both mutual authentication and secure communication in the WLAN link. The main novelty of the authentication policy is that we associate the authentication of users to the authentication of IPsec peers and we use highly flexible certification hierarchies to validate certificates. The management of user's certificates is also simplified, not requiring full-featured PKIs or complex management policies, such as distribution and checking of CRLs.

**Keywords:** Wireless networks, mobility, ubiquitous authentication, IPsec VPNs, public key certification, flexible certification agreements

## 1 Introduction

With the growing deployment of WLAN networks, users have more possibilities to connect to their home institutions and to the Internet through wireless hot-spots provided by other institutions. In this paper we describe a solution for enabling users to authenticate themselves securely when

connecting to a hosting wireless hot-spot and, simultaneously, to ensure the security of the wireless connection. This solution was mainly conceived for managing the roaming of students and professors within schools or research institutions.

Several solutions have been proposed to provide authentication and safe encryption for WLAN networks in order to overcome the limitation of WEP based security [2]. Some are based on standards (802.1X [4]) and some are non-standard variations of standard protocols (TTLS e PEAP). The standard ones are still not safe [10], while non-standard ones may become deprecated when 802.11i – the most recent IEEE standard for wireless security – becomes ubiquitous. Therefore, using layer 2 authentication mechanisms at this time for supporting a large-scale authentication schema for WLAN networks is not advised, both because several protocols are emerging and because it depends heavily on the services provided by WLAN Access Points.

This paper describes an alternative solution using a higher-level approach. It operates at IP level and uses IPsec tunnels negotiated with client and server asymmetric key pairs and public key certificates. These secure channels are used to get Virtual Private Networks (VPNs) between mobile user's computers (supplicants) and a security gateway in the WLAN access network. IPsec is a mature standard [13, 15, 16], being part of both the IPv4 and IPv6 standards, and is deployed in several different platforms. These facts ensure ubiquity and stability for the present, and longevity.

By using IPsec we don't have any special requirements from WLAN Access Points, which is useful for assuring the stability and the ubiquity of the solution. Our solution also works for roaming users using different network interfaces to connect to hosting institutions' networks, such as the usual 802.3 Ethernets.

IPsec VPNs can provide authentication of end-to-end peers, which is normally used for authenticating hosts, not people. In this paper we present a novel approach for this end-to-end authentication, since we authenticate mutually hosting institutions and client users. For this purpose, client's hosts will authenticate themselves using users' credentials and the security gateways of hosting institutions will authenticate themselves as institution's representatives.

IPsec VPNs can also provide IP-level data integrity and confidentiality. In our approach these mechanisms are fully explored to avoid any dependency on layer-2 security mechanisms, both in wired or in wireless scenarios.

Our distributed authentication and authorization policy is highly flexible, allowing a grass-root approach for evolving with time. Hosting institutions may authenticate and authorize wireless connections from roaming users belonging to individual institutions or to sets of institutions certified by a common trusted third-party, or Certification Authority (CA hereafter). Each institution may use its own local authentication mechanism for local users (e.g. UNIX, Kerberos [17], Windows Domains) without any cooperation with authentication mechanisms used by other institutions. A roaming user only needs to prove to a hosting institution that he (she) is a legitimate user of an authorized roaming institution. Such proof is achieved using only asymmetric key pairs and client certificates issued by the user's home institution.

Roaming authorization mechanisms often require some sort of online remote action to validate clients' credentials. In GSM [1], for instance, roaming operators get data from subscribers' operators for authenticating roaming subscribers. In our approach, roaming institutions and users can

work alone to authenticate themselves, no interaction is needed with any external entity for validating both authentication credentials. Furthermore, roaming issues are transparent for roaming users, they are only required to have credentials issued by their home institutions. In fact, users use their credentials for authenticating in any hosting institution, which can be either their home institution or a roaming institution.

Dealing with IPsec and authentication of hosts with asymmetric key pairs is usually painful, because some sort of well-established certification infra-structure (Public Key Infrastructure, PKI) must be deployed and maintained to manage the certificates of public keys. One of our goals was to minimize this problem. Therefore, we conceived a schema for managing users' certificates without having to maintain any online, full-featured PKI. Consequently, besides the specific hot-spot security gateways, there is not much more specific physical infrastructure to manage or complex PKI policies to implement.

This paper is structured as follows. Section 2 describes the authentication architecture. Section 3 describes the authentication of roaming users. Section 4 describes the management of supplicant certificates. Section 5 describes the authentication of institutions by supplicants. Section 6 presents some related work. Finally, Section 7 presents the conclusions.

## 2 Architecture

The solution assumes that each institution uses several WLAN Access Points to provide a convenient set of hot-spots on the same Virtual Local Address Network (VLAN). This VLAN is connected to the outside world — the remaining network of the institution and the Internet — through an IPsec security gateway. This security gateway prevents any communication between supplicants in the hot-spot and the outside world unless the supplicant and the gateway had established an authenticated IPsec VPN between themselves. In order to establish a VPN, both sup-

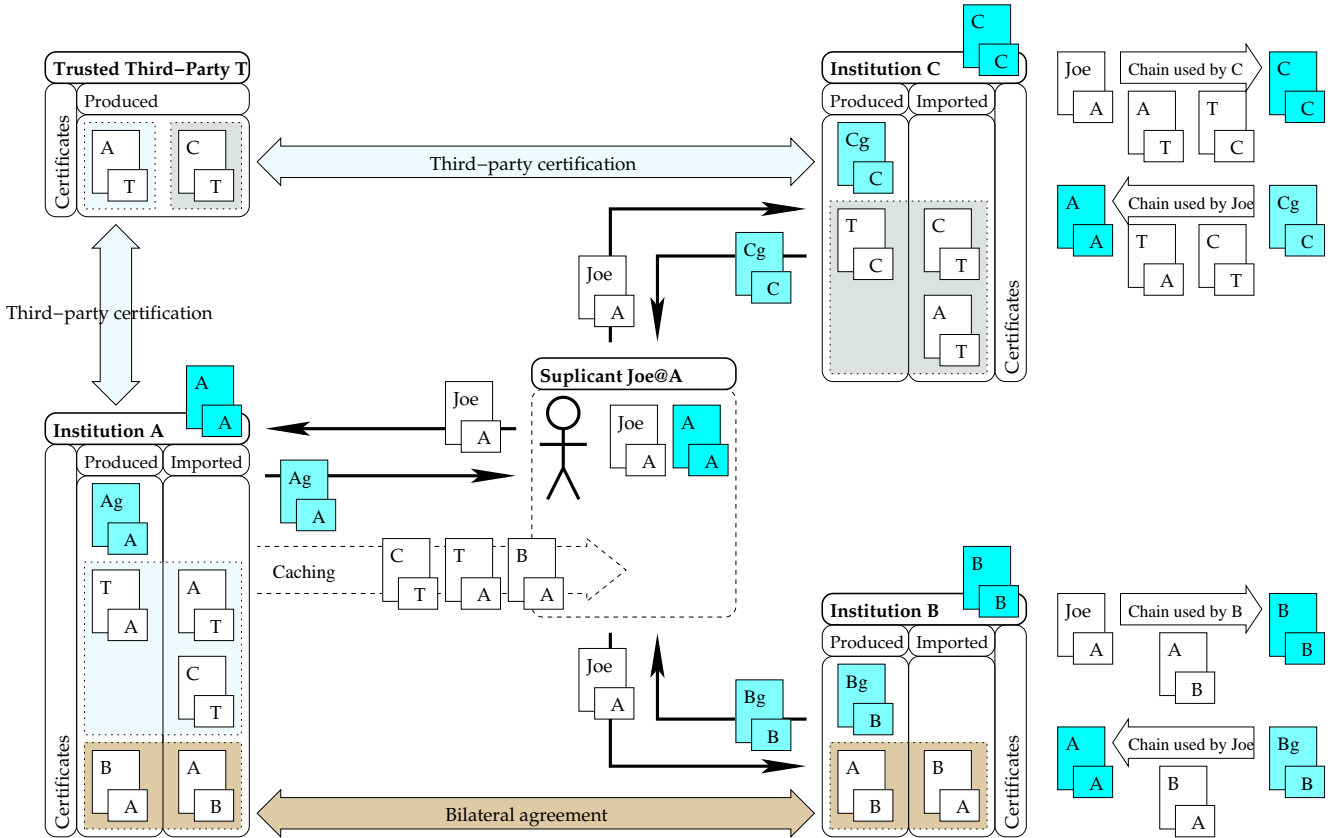


Figure 1: Management of certificates for authenticating IPSec VPN entities (supplicant and institution gateway) for both local and roaming users. User Joe belongs to Institution A and uses hot-spots of Institutions B and C to connect to the Internet.

supplicant and gateway authenticate themselves between each other using asymmetric key pairs and certified public keys. Supplicants can be either local or roaming users.

Along this text we will use the expression *home institution* to refer to the institution that is responsible for doing a proper authentication of a particular person. We will use the expression *hosting institution* to refer to an institution that allows authenticated and authorized, local or roaming users to access local or Internet services.

Along this text we will also refer to Fig. 1 to explain all the details about the management and use of certificates to authenticate IPSec VPN negotiators. We will use the expression  $X/Y$  to refer to a *certificate of X's public key issued by Y*. The figure shows the certificates stored and exchanged by supplicant Joe from Institution A, when establishing VPNs with Institution A (local authentication), Institution B (remote authentication using bilateral agreement), and Institution C (remote authentication using a third-party entity). We will use the term supplicant to refer to a user, or user computer, trying to get connected to the Internet through an institution's hot-spot.

tion), Institution B (remote authentication using bilateral agreement), and Institution C (remote authentication using a third-party entity). We will use the term supplicant to refer to a user, or user computer, trying to get connected to the Internet through an institution's hot-spot.

A key architectural characteristic of the authentication policy is that *each hosting institution acts as root certification authority for all users that try to use its hot-spots*. For local users this means that both users and the IPSec gateway exchange certificates issued by the local institution when setting up an IPSec VPN between the supplicant and the gateway (certificates  $Joe/A$  and  $Ag/A$  in Fig. 1). If they both share the same home institution then both certificates are verifiable by the same Institution's Root Certificate (IRC) that they both have and trust ( $A/A$  in Fig. 1). If the supplicant is roaming, then the gateway needs to build a certificate

chain until finding a trusted certification authority, which must be, again, the local institution's root certification authority ( $B/B$  and  $C/C$  in Fig. 1).

This model assumes that a user must get in advance an asymmetric key pair and a certificate for the public component prior to establish a WLAN IPSec VPN in his home institution or in another hosting institution. The certificate must be issued by the home institution and clearly identifies the user and its home institution. This enables hosting institutions to deploy per user or per institution authentication and authorization policies.

### 3 Authentication of roaming users

The roaming property results naturally, provided that each gateway is able to verify the certificates issued by other institutions. This can be done with bilateral agreements between institutions — using cross-certification of certificates from pairs of institutions, like between A and B in Fig. 1 — or it can be done by a hierarchical mechanism in which both certificates are signed by a third entity which has an agreement with all of them, like between A and C in Fig. 1. Both models may coexist with each other, thus providing the institutions with the flexibility of establishing private agreements with other institutions without being dependent of a central certification authority.

The coexistence of both models is possible because both use the same method to validate certificates. Both use a local certificate database in each institution with one trustworthy root certificate - the IRC - and several other, possibly untrustworthy, certificates. The IRC and other certificates signed with IRC's private key are generated locally. The remaining ones are generated elsewhere and can be acquired through any channel (trustworthy or not).

For the bilateral agreement model, each hosting institution needs to issue, and store locally, a certificate of the other institution's public key. With this certificate the gateway will be able to verify certificates of roaming supplicants from the other

institution by following a certification chain until the local IRC. In Fig. 1 this is exemplified by  $A/B$ , issued and used by B in certification chains for authenticating supplicants from A.

For the hierarchical model, each institution needs to issue, and store locally, a certificate of the public key of the CA, and to get and store locally a certificate, issued by that CA, for each institution participating in the agreement. As for the bilateral agreement, these certificates are going to be used to build certification chains from supplicants' certificates until the local IRC. In Fig. 1 this is exemplified by  $A/T$ , issued by the CA T and cached and used by C in certification chains for authenticating supplicants from A.

Both models have advantages and disadvantages. The bilateral agreement model is more flexible, giving to each institution the ability to choose their partners independently. The hierarchical model is more scalable, because each institution's certificate only needs to be signed once - by the CA - and can be universally distributed through unsecured channels (e.g. HTTP, SMTP, etc.).

The hierarchical model can be made even more scalable by adding further hops in the certificate chain, e.g. international CAs that sign certificates for national CAs, thus reducing the number of certificates that need to be signed by each entity. The difficulty of this schema is the process of building it from the ground, since each international CA needs to be certified by each institution and each national CA needs to be certified by an international CA. However, such large-scale certifications chains can be incrementally created and added to each institution, providing several levels of certification, without breaking the local authentication. Supplicants start by being able to connect with only their home institution and progressively become able to connect with more institutions, linked with certificate chains, without renewing their certificates.

## 4 Supplicant Certificates

The use of supplicant certificates is often avoided due to the complexity to deploy and manage full-featured PKIs, which are usually required to manage certificates. However, it is possible to have supplicant certificates without having a PKI because these certificates require specific characteristics that simplify its management. The characteristics are the following:

- The certificates should be usable only for setting up secure communication channels, and not for checking digital signatures on documents. The rationale for this limitation is that it allows home institutions to both generate key pairs and issue the certificate of the public component for each local, authenticated user.
- User certificates should have a short validity period for avoiding the management of certificate revocation lists (CRLs) on issuing institutions. The rationale is that it is more simple to force the users to authenticate frequently on their home institution and get a fresh key pair and a certificate than to manage and check CRLs. This is analogous to Kerberos tickets [17], that ordinarily are valid for about a day.

This does not mean that revocation certificates and CRLs cannot be used for specific cases. It means that the entire authentication system can work pretty well and securely without them provided that user certificates have short life times.

Therefore, all it is needed on each institution, besides the IPSec gateway, is a server for generating and distributing key pairs and public key certificates to authenticated local users. This task can be accomplished by an HTTPS server, a CGI script protected by a "username, password" pair, or any other form of personal authentication, and a user directory service (LDAP, SQL or Active Directory). Because this infrastructure often exists to provide WebMail and other protected con-

tents, the addition of certificates is easily deployed and has a negligible impact on management.

Using supplicant certificates for authentication is also very convenient for managing visitors without a valid certificate. This is a very common situation that can be handled appropriately by giving a group of people the ability to issue one-day certificates for their visitors. For example, University professors could issue several one-day certificates for their visitors.

## 5 Authentication of institutions

In order to establish an IPSec VPN with its home institution, or with any other hosting institution, a supplicant must be able to validate the certificate presented by the institution's IPSec gateway ( $A_g/A$ ,  $B_g/B$  and  $C_g/C$  in Fig. 1). To do so the supplicant uses its home IRC as trusted root and a set of other certificates to build certification chains until the trusted root.

The home IRC allows him to authenticate the IPSec gateway of its home institution. The other certificates allow him to build authentication chains, ending in the trusted IRC, to authenticate the IPSec gateways of hosting institutions. Considering the example of Fig. 1, the supplicant must have and trust the home IRC  $A/A$  to authenticate the gateway of A, must have certificate  $B/A$  to authenticate the gateway of B and must have certificates  $T/A$  and  $C/T$  to authenticate the gateway of C.

The trusted home IRC must be obtained when the supplicant user gets its own credentials (asymmetric key pair and public key certificate). This is mandatory because the IRC is self-signed, therefore the only way to trust its value in order to tag it as a trusted certification root is to get it within a properly authenticated interaction.

Different policies can be devised to get all the other certificates but, in our opinion, that should be as simple and transparent as possible to supplicant users. These are possible policies:

**Enable a roaming supplicant to look for the required certificates just-in-time for accessing a hosting institution.** However, this implies that supplicants should be able to get institution's certification agreements in order to know which are the missing certificates. And, furthermore, supplicants should be able to browse many Internet servers for getting the missing certificates before establishing an IPSec VPN with the hosting institution, i.e. before authenticating themselves.

**Rely on institutions to provide the missing certificates to roaming supplicants.** An extension of IPSec is under evaluation for providing this service [18]. Without such extension roaming supplicants would have to run some sort of bootstrap application to interact with the hosting institution in order to get the required certificates. In either case, however, hosting institutions have the burden and the responsibility to keep an updated cache of certificates and certification hierarchies for helping roaming supplicants to authenticate the institution's gateway, which is not logic.

**Rely on home institutions to provide the necessary certificates to enable local users to roam seamlessly.** In this case home institutions have the responsibility to provide a good roaming support, in terms of certificate distribution, for their local users, which seems logical.

We decided for the last policy because it simplifies the task of both supplicants and institutions' gateways. All that it is necessary is to have a certificate distribution server, in each institution, that provides the following certificates to local users: (i) all the ones that the institution issued for other institutions/CAs and (ii) all the ones issued by CAs certifying public keys of other institutions/CAs. Considering the example of Fig. 1, Joe gets from Institution A the certificates  $B/A$  and  $T/A$ , for the first case, and the certificate  $C/T$  for the second case. Naturally, each institution using the certification services of a CA, to build third-

party certification bridges with other institutions, must get, and cache locally, all the other certificates issued by the CA (Institution A caches  $C/T$  and Institution C caches  $A/T$ ).

The distribution of all these certificates to users can be done by a simple directory service using an insecure distribution channel. Hosting institutions can also provide a simple gateway service allowing roaming users to get just-in-time the certificates (from their home institutions) necessary to authenticate the hosting gateway.

## 6 Related work

Large-scale authentication schemas, and in particular using asymmetric cryptography, are not common place. However, we can compare our schema with three other existing solutions, the first two using exclusively secret, shared-key authentication, and the last one using both symmetric and asymmetric cryptography: (i) the SIM-based authentication, used in the existing GSM infrastructure, enabling mobile stations to roam between different suppliers [1], (ii) a similar infrastructure that could be implemented with Kerberos [17], and (iii) 802.1X with a AAA (Authentication, Authorization and Accounting [7]) hierarchy [11] with support for EAP [6], usually a RADIUS hierarchy [20].

The SIM-based authentication and key agreement protocol is actually being adapted to the EAP protocol (EAP-SIM [14]) for enabling users with a SIM card to authenticate themselves when accessing WLAN networks. Kerberos is actually used to authenticate users in MS Windows 2000 Domains [3] or AFS cells [5] and was designed since the beginning for large-scale authentication. For that purpose it allows Kerberos realms to establish trust relationships to allow principals registered in one realm to get an authenticated interaction with principals registered in other realms. In fact, the trust relationship between Kerberos realms can support both the bilateral and hierarchical agreements that we allow with public key certificates.

The 802.1X authentication starts with an authentication request sent by the supplicant to the authenticator - a switch or a wireless access point. The authenticator redirects the request to the authentication server - the RADIUS server [21]. If the supplicant is a local user and is an authorized client, the authenticator enables the network connection. When the server receives an authentication request from a roaming supplicant, it simply acts as a proxy RADIUS. In this case, the local RADIUS checks if the user institution is an authorized one and, if this is the case, it simply forwards the authentication requests to the home institution [9]. The 802.1X uses EAP (Extended Authentication Protocol) as a meta-protocol, or framework to encapsulate specific authentication mechanisms. Therefore, EAP can be used with several authentication protocols: PAP, CHAP, MS-CHAPv2, MD5, TLS, etc. Most of them are shared secret based, but TLS uses asymmetric cryptography [8, 12] and can also benefit from the public key certification infrastructure described in this paper.

Thus, what are the advantages of using asymmetric cryptography and certificates for authenticating local or roaming supplicants instead of shared key systems, like in GSM, Kerberos or with a RADIUS hierarchy?

The first advantage is that asymmetric cryptography and certificates are more straightforward to use to establish IPsec security associations (SAs) between mobile computers and WLAN gateways. Symmetric, shared keys could also be used to negotiate such SAs but that would require the development and deployment of some middleware to use shared session keys, provided by an authentication mechanism like the one of GSM or Kerberos, in the creation of security associations.

The second advantage is that all the issues concerning the correct establishment of trust relationships between institutions are managed solely by institutions and are transparent for local or roaming supplicants. Users only have to get and carry a valid key pair and a certificate of the public component issued by the home institution. The same happens with GSM and the RADIUS hier-

archy. But with Kerberos, on the contrary, supplicant users, or software running on supplicants' computers, would have to get acquainted with the trust relationships between institutions and trusted third-parties to perform the right number of hops along TGS services of several Kerberos realms to get the credentials — a session key and a ticket — for connecting the hosting institution. Even worse, such ticket could not be used in other institutions, making roaming a non-transparent issue for users.

The third advantage is that we don't require any new, ubiquitous personal authentication mechanism, or some middleware to manage and accommodate several existing ones, like in [19]. Instead, we only use the authentication mechanisms that probably exist in home institutions, to authenticate users requiring asymmetric key pairs and certificates. This means that we just provide a different set of credentials using the existing user authentication policies. By using a RADIUS hierarchy, GSM, or a Kerberos-like approach, users could be forced to deal with another authentication mechanism.

The fourth advantage is that authentication with asymmetric cryptography allows a simpler and more fault-tolerant, just-in-time fetching of fresh supplicants' credentials. If supplicants have valid credentials when they connect hosting institutions, then no interaction with other institutions is required to authenticate them. If, on the contrary, supplicants need fresh credentials, then the only requirement for allowing them to fetch credentials just-in-time it is to allow unauthenticated supplicants to get in contact with home institutions to get new key pairs and certificates.

On the contrary, GSM authentication always requires fresh authentication tuples, that hosting institutions must fetch, within properly authenticated sessions, from the supplicants' home institutions. The RADIUS hierarchy authentication solution is potentially more complicated, and less fault-tolerant, since it requires the intervention of several servers, namely every RADIUS server in the hierarchy chain from the host institution RADIUS up to a root RADIUS server and from the

root down to the home RADIUS server (cf. [19]). Finally, Kerberos supplicants should get a new ticket for each hosting institution, forcing supplicants to contact several TGS services to get such ticket and forcing the hosting institution to allow several interactions between unauthenticated supplicants and Kerberos servers.

Finally, the fifth advantage is that we totally eliminate the risks of password guessing attacks. Off-line dictionary attacks are a major threat in large-scale authentication infrastructures using user-supplied passwords. By using asymmetric cryptography for authenticating users while setting up IPsec VPNs, we totally eliminate the risks of password guessing attacks because no passwords are used. The non-TLS EAP protocols and the Kerberos authentication protocol are vulnerable to such attacks. On the contrary, this is not an issue for mobile phones using GSM because users' passwords are not chosen or modifiable by users, they are 128 bit keys stored and protected by the SIM card.

## 7 Conclusion

We have described a solution for local and roaming authentication for WLAN supplicants, using IPsec based VPNs with client and server certificates. The solution has several interesting properties, namely:

- The solution solely uses IPsec VPNs, therefore it has no special requirements from WLAN Access Points. IPsec is a mature standard, is deployed in several different platforms and is part of the IPv6 standard, which ensures stability for the present and longevity. On the contrary, layer 2 authentication and access control mechanisms are evolving rapidly, preventing a large-scale integrated solution for authenticating local and roaming supplicants. Furthermore, by using only mandatory IPsec features, like certificate-based negotiation of SAs, the solution promotes maximum compatibility between several IPsec implementations.

- By using client certificates we simplify the physical infra-structure required for roaming, because it uses the certificate chain ability to build on each gateway a logical hierarchy equivalent to the physical hierarchy of other solutions (e.g. multi-hop contact with RADIUS hierarchies).
- Because authentication is done locally, a roaming user may be authenticated even when it is not possible to contact his home institution, thus being more fault-tolerant. But getting just-in-time credentials is also easy, provided that unauthenticated supplicants can contact the credentials' supplying server in their home institution.
- Because authentication does not require user intervention — everything is done automatically in the WLAN gateway using the certificates — the authentication process and the roaming issues are transparent for supplicant users, thus improving the user perception of resilience to temporary disconnections (the authentication upon reconnection is transparent).
- By using client certificates with short validity periods and without the ability for checking signed documents, our solution does not require the deployment of full-featured PKIs. Our solution offers also a simple method to allow limited-time access to foreign visitors to hosting institutions.

This solution was first deployed in two widely-separated campus of the same Faculty, both equipped with several WLAN hot-spots. It has been successfully used for several months by students and professors. The experience was conducted using a Linux gateway with the FreeSwan Open Source IPsec stack and four different IPsec stacks on supplicants: Windows 2000 IPsec stack; Windows XP IPsec stack; Racoon IPsec stack for MacOS X; and FreeSwan for Linux.

## References

- [1] GSM Technical Specification GSM 03.20 (ETS 300 534): "Digital cellular telecommunication system (Phase 2); Security related network functions". Technical report, European Telecommunications Standards Institute, August 1997.
- [2] Information technology — Telecommunications and information exchange between systems — Local and metropolitan area networks — Specific requirements — Part 11: Wireless LAN Medium Access Control (MAC) and Physical Layer (PHY) Specifications. ANSI/IEEE Std 802.11, 1999 Edition, 1999.
- [3] Windows 2000 kerberos authentication. White paper, Microsoft Corporation, 1999. <http://www.microsoft.com/windows2000/docs/kerberos.doc>.
- [4] IEEE Standard for Local and metropolitan area networks — Port-Based Network Access Control. IEEE Std 802.1X-2001, 2001.
- [5] OpenAFS Home Page, 2001. <http://www.openafs.org>.
- [6] B. Aboba, L. Blunk, J. Vollbrecht, J. Carlson, and H. Levkowitz. Extensible Authentication Protocol (EAP). RFC 3748, June 2004.
- [7] B. Aboba, P. Calhoun, S. Glass, T. Hiller, P. McCann, H. Shiino, P. Walsh, G. Zorn, G. Dommety, C. Perkins, B. Patil, D. Mitton, S. Manning, M. Beadles, X. Chen, S. Sivalingham, A. Hameed, M. Munson, S. Jacobs, B. Lim, B. Hirschman, R. Hsu, H. Koo, M. Lipford, E. Campbell, Y. Xu, S. Baba, and E. Jaques. Criteria for Evaluating AAA Protocols for Network Access. RFC 2989, November 2000.
- [8] B. Aboba and D. Simon. PPP EAP TLS Authentication Protocol. RFC 2716, October 1999.
- [9] B. Aboba and J. Vollbrecht. Proxy Chaining and Policy Implementation in Roaming. RFC 2607, June 1999.
- [10] William Arbaugh, Narendar Shankar, and Y. C. Justin Wan. Your 802.11 wireless network has no clothes. In *Proceedings of the First IEEE International Conference on Wireless and Home Networks*, December 2001.
- [11] P. Congdon, B. Aboba, A. Smith, G. Zorn, and J. Roese. IEEE 802.1X Remote Authentication Dial In User Service (RADIUS) Usage Guidelines. RFC 3580, September 2003.
- [12] T. Dierks and C. Allen. The TLS Protocol Version 1.0. RFC 2246, January 1999.
- [13] D. Harkins and D. Carrel. The Internet Key Exchange (IKE). RFC 2409, November 1998.
- [14] H. Haverinen and J. Salowey. Extensible Authentication Protocol Method for GSM Subscriber Identity Modules (EAP-SIM). Work in progress (Internet-Draft), April 2004. Expires at Oct. 4, 2004.
- [15] S. Kent and R. Atkinson. IP Authentication Header. RFC 2402, November 1998.
- [16] S. Kent and R. Atkinson. IP Encapsulating Security Payload (ESP). RFC 2406, November 1998.
- [17] J. Kohl and C. Neuman. The Kerberos Network Authentication Service (V5). RFC 1510, September 1993.
- [18] Brian Korver and Eric Rescorla. The Internet IP Security PKI Profile of IKE/ISAKMP and PKIX. Work in progress (Internet-Draft), February 2004. Expired at July 2004.
- [19] Y. Matsunaga, A. S. Merino, T. Suzuki, and R. H. Katz. Secure Authentication System for Public WLAN Roaming.

In *ACM International Workshop on Wireless Mobile Applications and Services on WLAN Hotspots (WMASH)*, pages 113–121, September 2003.

- [20] D. Mitton, M. St.Johns, S. Barkley, D. Nelson, B. Patil, M. Stevens, and B. Wolff. Authentication, Authorization, and Accounting: Protocol Evaluation. RFC 3127, June 2001.
- [21] C. Rigney, A. Rubens, W. Simpson, and S. Willens. Remote authentication dial in user service (RADIUS). RFC 2138, April 1997.