POLITECNICO DI TORINO
Repository ISTITUZIONALE

Fast Authentication in Heterogeneous Wireless Networks

Original
Fast Authentication in Heterogeneous Wireless Networks / Albertengo, G., Pastrone, C., Tolu, G.. - (2005). (SPECTS 2005 2005).

Availability:
This version is available at: 11583/1413088 since:

Publisher:
Society For Modeling And Simulation International

Published
DOI:

Terms of use:

This article is made available under terms and conditions as specified in the corresponding bibliographic description in the repository

Publisher copyright

(Article begins on next page)

28 November 2026



Fast Authentication in Heterogeneous Wireless Networks

Journal:	*International Symposium on Performance Evaluation of Computer and Telecommunication Systems*
Manuscript ID:	SPECTS-05-NTS-085.R1
Topic Area:	Networking and Telecommunication System
Date Submitted by the Author:	n/a
Complete List of Authors:	Albertengo, Guido; Politecnico di Torino, Electronics Pastrone, Claudio; ISMB, Networking Tolu, Giacomo; CNIT, ----
Keywords:	Authentication, Security, Mobility , Wireless , Satellite and Terrestrial Networks

powered by ScholarOne
Manuscript Central™

Fast Authentication in Heterogeneous Wireless Networks

Guido Albertengo
Electronics Department
Politecnico di Torino
Turin - Italy

Claudio Pastrone
Istituto Superiore
"Mario Boella" - ISMB
Turin - Italy

Giacomo Tolu
CNIT
Parma – Italy

Keywords: Authentication, Wireless, Security Mobility, Satellite and Terrestrial Networks.

Abstract. The growing diffusion of wireless devices is leading to an increasing demand for mobility and security. At the same time, most applications can only tolerate short breaks in the data flow, so that it is a challenge to find out mobility and authentication methods able to cope with these constraints. This paper aims to propose an authentication scheme which significantly shortens the authentication latency and that can be deployed in a variety of wireless environments ranging from common Wireless LANs (WLANs) to satellite-based access networks.

1 INTRODUCTION

Wireless LANs are widely used due to their easy installation and fast deployment. Inside homes, hotels and enterprises, WLANs are a cost effective alternative to traditional wired LANs. In public areas, shops and lounges, they are used by travelers and guests to get their e-mail, to transfer data with the file servers of their company, or simply to surf the Internet looking for news. Usually these WLANs are connected to the Internet through ADSL broadband connections, but in some cases the WLAN uses a satellite link to connect a small group of residential and business users living in rural areas where ADSL can not be used, either for technical or economic reasons [1].
Thus, the applications of this technology can be extremely different, but they all share the same problems: limited bandwidth, weak security and lack of mobility.

The first issue has already been effectively addressed by standardization bodies and industries, so that we are now able to share a bandwidth of at most 54 Mbps with a simple, low cost 802.11g wireless card. On the contrary, the two remaining issues require better solutions than those currently available [2][3][4]. In particular, the requirements of the applications we are used to run on our portable computers should be carefully taken into account in designing an integrated mobile secure wireless network. So far, however, the increase in security generally comes at the expenses of mobility: the time needed to authenticate increases the total handover time making impossible to support not only real-time applications, but even asynchronous ones.
This led us to study currently available authentication schemes to see whether it could be possible to overcome their present limits and make them usable in this new challenging unwired world.
The rest of this paper is organized as follows. In Section 2 we give a short overview of the main authentication mechanisms used in a wireless environment. Section 3 describes our authentication scheme and in Section 4 an example of its possible extension to a multi-domain environment is reported. Finally, in Section 5 we present our future work while Section 6 summarizes the paper.

2 CURRENT AUTHENTICATION SCHEMES

Before starting the description of our proposal, it is worth recalling the brief story of the security in WLANs.

At the beginning, the only way to protect data from eavesdropping in a WLAN was to use WEP encryption [5]. This system was based on a secret key shared by the network client and the wireless Access Point (AP). Simple, difficult to manage (the key had to be manually written when configuring both the WLAN interface in the client computer and the AP) and unfortunately very weak (the interested reader can see [6]). Often, in order to improve its effectiveness, other mechanisms were used along with the WEP protocol. An example of this is the MAC address authentication where the MAC address of a station attempting to associate to an access point is compared against a list of authorized users stored in an external authentication server or even locally in the access point. The access is granted only if a matching entry is found. Actually, MAC address authentication does not improve the security of the WEP protocol since a valid MAC address can be easily spoofed and forged by any attacker.

The second act was the adoption of the 802.1X Network Access Control scheme [7], which was aimed to both wireless and wired LANs. Unfortunately, also this method proved to be not secure enough. This is basically due to fact that this scheme is still based on WEP keys. Furthermore, 802.1X does not have a smart mechanism to manage keys.

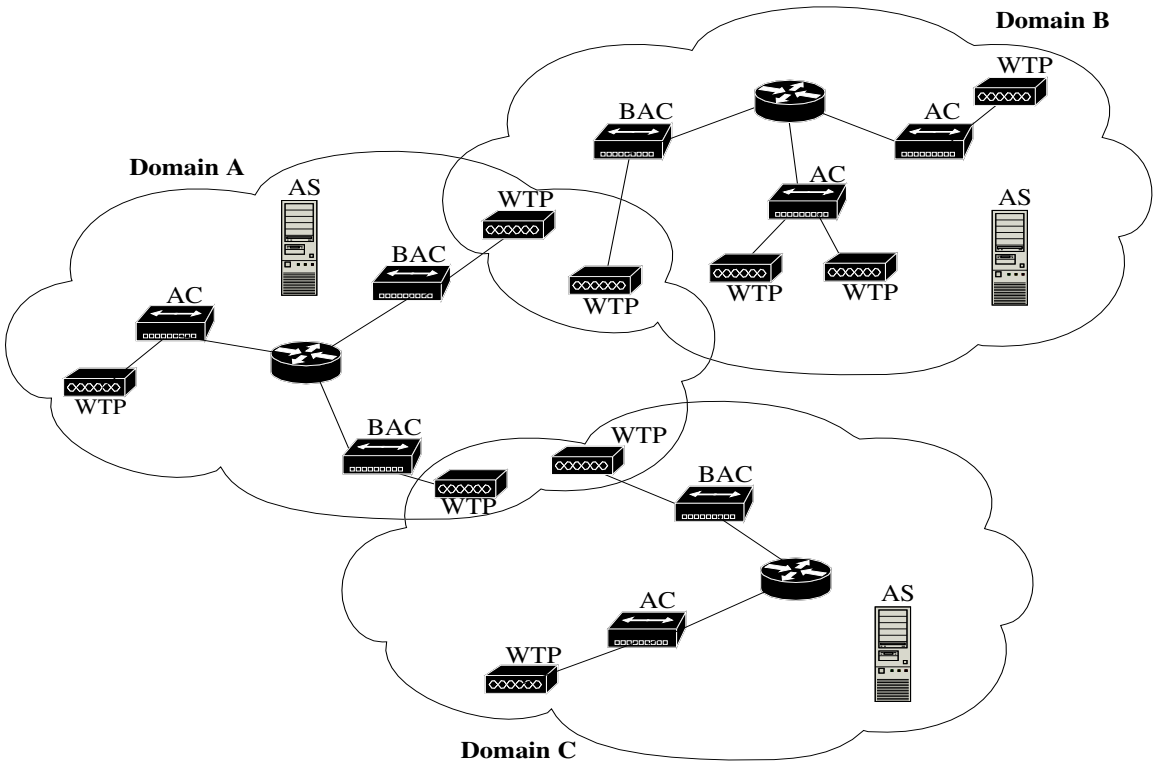


Figure 1 CAPWAP Architecture

At this point, the industry was tired of waiting for a new standard able to overcome the shortcomings of WEP and 802.1X, so that the Wi-Fi Alliance [8] along with members of the IEEE Task Group I, released the Wi-Fi Protected Access (WPA) method. This is a robust wireless LAN security solution that matched the immediate needs of the marketplace, and was readily adopted.

Finally, in June 2004, the IEEE approved the 802.11i standard [9]**,** currently considered as the ultimate solution to wireless security. This standard, for the first time adopts a hierarchy of keys and new strong cryptographic algorithms. Although security is certainly improved by the 802.11i, mobility is surely not: a full 802.11i authentication requires several messages most of which involving a (possibly) remotely located Authentication Server (AS), which usually adopts the de-facto standard RADIUS protocol. The result is a not negligible total handover time that makes this solution not suitable for mobility.

It is worth to notice that after this long authentication procedure the network client (or Mobile Host – MH – as it will be referred to in the following) and the AP, share a common key (the Pairwise Master Key – PMK) that in turns is used to locally generate the Pairwise Transient Key (PTK), which is the key actually used to encrypt data. This latter key is periodically regenerated so that it can not be successfully decoded intercepting enough data on the radio link (as it was possible with WEP and 802.1X). This periodical regeneration of the PTK just requires the exchange of four messages between the MH and the AP so that it can last a few milliseconds. We will refer to this simple procedure as *Four Way Handshake* (FWH).

Therefore, as far as the MH does not change the AP he is connected to, the short break due to the FWH required to regenerate the PTK does not affect any application, being it asynchronous or even real time.

One proposal trying to leverage on this nice property of 802.11i was the Pro-Active Key Distribution (PAKD) scheme [10]. It tries to avoid long authentications by distributing the PMK to the neighbors of the AP the MH host is attached to *before* the MH handoff. In this way, when the MH connects to a new neighboring AP both of them already share the PMK, so that the authentication procedure can be easier. To find out the neighbors of a given AP a data structure called Neighbor Graph is used.

Although the PAKD solution seems to effectively reduce the authentication latency, it unfortunately has two main

drawbacks: it lacks in security due to some modifications of the 802.11i standard and it is intended to be a local solution only, that is it can not be used to authenticate users not registered in the local AS (we will refer to these users as foreigners from now on).

3 OUR AUTHENTICATION SCHEME

To try to solve the latency authentication problem we started from two considerations:

- a hierarchical structure could ease the key management;
- the PAKD mechanism requires a lot of key distributions when the lowest level of the hierarchy is the AP.

A simple solution is to split the AP in two different equipments, each implementing one of the two layer the AP is composed of, as suggested by the CAPWAP [11] paradigm. According to this latter, the resulting infrastructure of a WLAN is composed of a number of Wireless Termination Points (WTPs) all controlled by a single Access Controller (AC). An example of such an architecture is shown in Figure 1

From the authentication point of view this implies that, after the authentication, a MH can freely handoff among all the WTPs laying under the same AC without needing further authentications. Another important point is that this architecture perfectly fits with the topology of the periphery of all enterprise networks: here each PC or AP is directly connected to a layer-2 switch, so that it would just be a matter of hours to replace APs with WTPs and layer-2 switches with ACs.

Another advantage of this two-tier hierarchy is the possibility to identify three different types of mobility:

- *Nano-mobility (n-mobility)*: when a MH moves between two WTPs controlled by the same AC;
- *Micro-mobility (μ-mobility):* when the MH moves between two WTPs controlled by two different ACs;
- *Macro-mobility (M-mobility*): when the MH leaves a WTP belonging to a domain and connects to another WTP of a different domain.

The original PAKD scheme roughly corresponds to a μ-mobility handoff: M-mobility events are not taken into account.

In order to minimize the management overhead due to a PAKD mechanism, notice that these three events likely occur with different frequencies. So, it is worth to find out some way to treat them differently: in our scheme we propose three different authentication mechanisms, each of them being tailored on the actual security needs. The three abovementioned mobility events trigger a zero, fast or full authentication procedure, respectively.

In the following these three authentication procedures will be described in reverse order, starting from the most complex one to better show how our proposed modifications to the original protocol reduce its complexity and latency.

3.1 Full Authentication

In order to describe the full authentication process, we consider a MH and two domains, namely its Home Network (HN) and a Foreign Network (FN). Moreover, we assume that an agreement exist between these two domains, so that their authentication servers (ASs) can exchange messages on a secure link. At the beginning the MH is off line and is in a location within the FN.

When the MH switches on, and connects to the nearest WTP (in our example a WTP in the FN), it is asked to perform a full authentication, since it is unknown to the FN. During the full authentication the keys that will be used for other simpler authentication procedures (fast and zero authentication) are created and distributed.

To better understand this protocol first the authentication and then the key management will be described. The first phase involves a MH, an authenticator (the AC) and a server (the AS). Its goal is to verify the MH credentials and to create the PMK to be shared between the MH and the AC. To start this procedure, the MH sends an *EAPOL-Start* packet, according to the 802.1X standard. This is therefore a supplicant initiated authentication, whose basic message exchange is shown in Figure 2. It is easy to see, without going into many details, that the AC requests the MH identity (*EAP Request-Identity*) and upon receiving it (*EAP Identity-Response*) gets in touch with the local AS to verify the MH credentials.

If the MH is one of its registered users and the credentials are valid the key management procedure starts. Otherwise, as in this example, the local AS should get in touch with the AS of the domain where the MH is registered (the AS in the HN in our case) to verify the MH credentials. Notice that the remote AS must be identified using the MH identity, which should therefore contain the HN domain name. A simple solution is to use the NAI [12] naming convention: *<username>@<domain>*. Obviously, all the AS names should have an identical host name (*LocalAS* for example) so that their full network name will be something like *LocalAS.<domain>*. A simple DNS query is therefore sufficient to get the remote AS IP address. Now a secure channel between the two ASes can be established, thanks to the previously mentioned agreement, so that the key generation and management can start and no information can be eavesdropped.

In the key management phase the FN AS just acts as a relay between the HN AS and the MH, and the procedure is fully in accordance with the 802.11i standard (see Figure 2 again).

At the end of this procedure the MH and the remote (i.e. in the HN) AS share a MK and the initial PMK, which is actually composed of the first 256 bits of the MK. Then, the

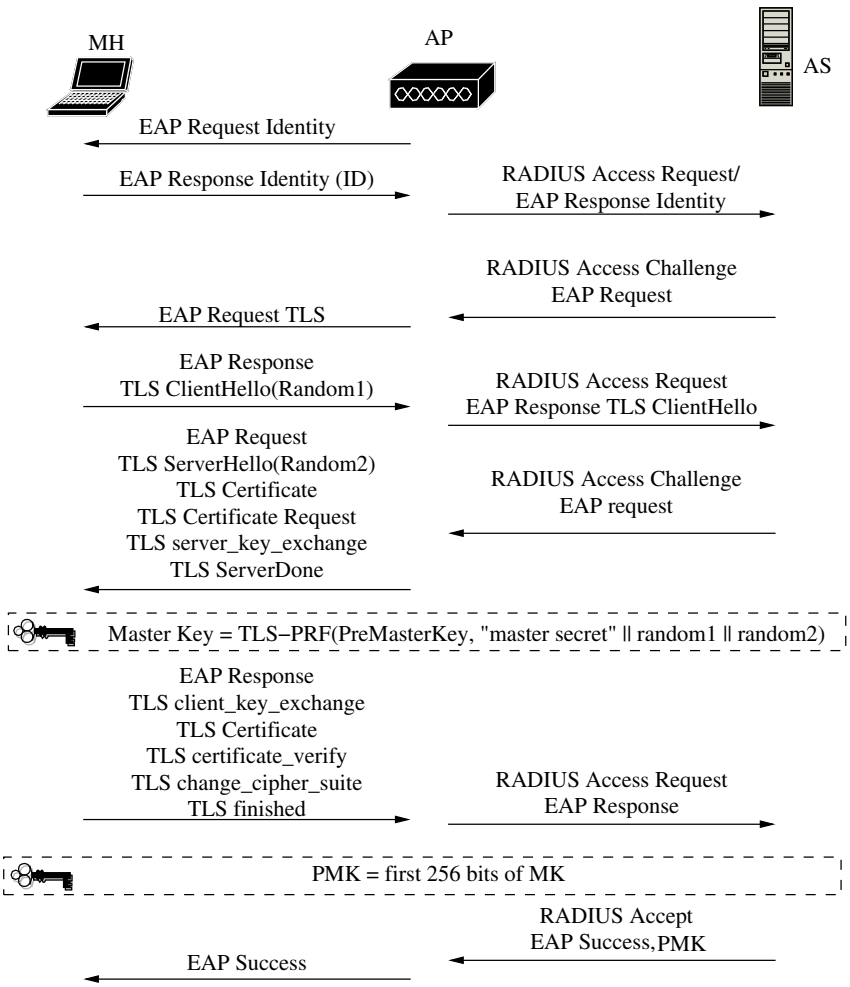


Figure 2 – Authentication phase in IEEE 802.11i

whole *authentication context* of the MH, that is the set of MK, initial PMK, MH identity and MH MAC address, is copied to the local AS (i.e. the one in the FN). Notice that the MH MAC address is required to compute the subsequent PMKs to be shared between MH and AC, according to the following formula, that was originally proposed in PAKD:

$$PMK_n = PRF(MK, PMK_{n-1} \| AC_MAC \| MH_MAC)$$

where *AC_MAC* and *MH_MAC* are the AC and the MH MAC addresses, respectively, *MK* and *PMK* are the keys, and *PRF* is a pseudo random number generating function.

The simple protocol we have described solves the problem of the authentication across different domains by simply transferring an authentication context through a secure channel between two authentication servers. Notice that when the MH is in its HN the authentication context is still created but it is not sent anywhere.

In both cases, the first phase of the authentication procedure ends when the local AS stores the authentication context and transmits the PMK to the AC. Also in this moment the location of the MH can be registered in both the local and the remote servers.

The second phase of the authentication aims to establish a shared PTK for the data traffic between the MH and the AC. The messages exchanged between the MH and the AC are those described in 802.11i with the only difference that in 802.11i the role of the authenticator is played by the Access Point (AP) whereas in our case the authenticator is the AC. This is the FWH and will described in the next section along with the fast authentication procedure.

So far, full authentication is very similar to the authentication procedure described in 802.11i, being the only difference the interaction between the local AS and the remote one when the MH is roaming. From now on, however, our proposal adds new key management procedures to better support mobility.

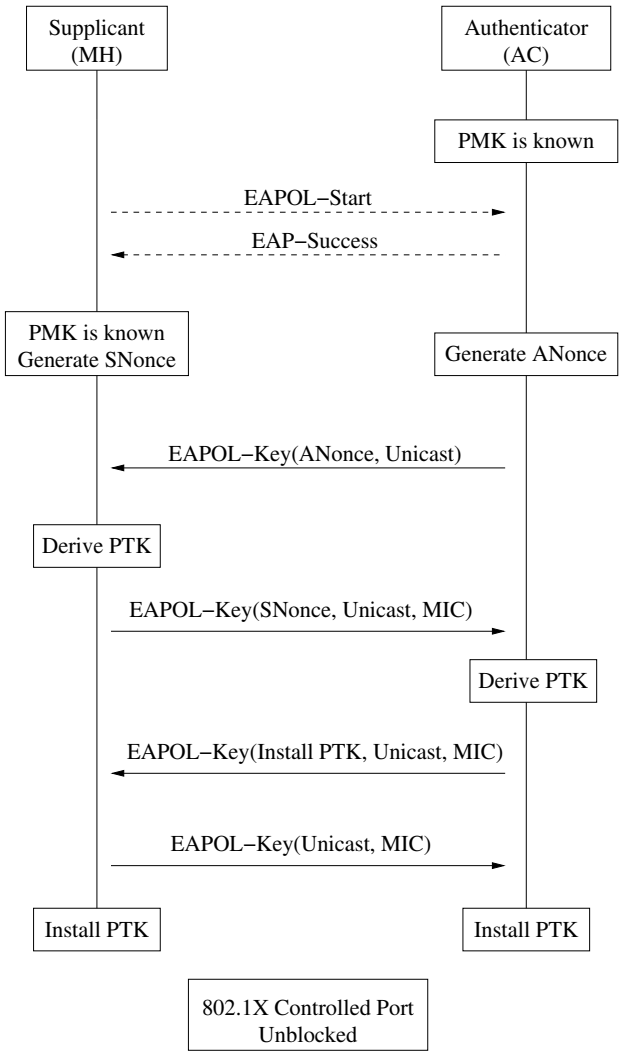


Figure 3 – Fast Authentication with Four Way Handshake (FWH)

The first addendum is based on the PAKD scheme and aims to the creation and distribution of the PMKs for the neighbors of the AC currently hosting the MH. In fact, whenever the AC changes the PMK should be updated by using the previously mentioned formula.

Note that the MH can compute the new PMK as soon as it obtains *AC_MAC* in response to the EAPOL-Start message (which is sent to a standard MAC address as specified in 802.1X), but the AC can not do that since it does not possess neither the PMK nor the MK. Since the MK, according to the 802.11i standard should never be moved to a peripheral device such as the AC, which is potentially less secure than a core device such as the AS, the only possible solution is to compute the PMKs in the AS. In fact, it knows all the required information, including the MAC addresses of the ACs, and can therefore compute these keys and then copy them to every AC, along with the identity of the MH the PMK is related to.

This preliminary key distribution opens the way to a simpler authentication procedure when the MH moves from a WTP controlled by a given AC to another WTP controlled by a neighboring AC, i.e. an AC that already has the right PMK to re-authenticate this MH. We dubbed this new procedure Fast Authentication.

3.2 Fast Authentication

This procedure aims to authenticate a MH which is already known to the AC. Its message exchange is shown in Figure 3. The procedure is activated by the reception at the AC of an implicit (i.e. a successful association) or explicit (i.e. an *EAPOL-Start* message sent by the MH) request of authentication. The AC verifies the possession of a PMK for the requesting MH: if a correct PMK is available, the four way handshake is started after having sent an *EAP-Success* packet in response to the implicit or explicit authentication pending request; otherwise a full authentication is performed.

After the MH authentication, the FWH starts and the PTK is generated and installed. The message exchange in the FHW is shown in Figure 3, lower part. The procedure starts with the local generation of two random numbers, *ANonce* in the AC and *SNonce* in the MH. Then the AC sends *ANonce* to the MH, which computes the PTK using the formula

$$PTK = PRF\left(PMK, \text{"Pairwise key expansion"}, Min\left(AC_MAC, MH_MAC\right) \| Max\left(AC_MAC, MH_MAC\right) \| Min\left(ANonce, SNonce\right) \| Max\left(ANonce, SNonce\right)\right)$$

Then, the MH sends back to the AC *SNonce* and the Message Integrity Code (MIC), i.e. a sort of digital signature computed using the Key Confirmation Key (KCK), which in turns is a portion of the PTK. At this point, also the AC can derive the PTK using the abovementioned formula.

The two last messages are the confirmation of the successful generation of the PTK at the AC and the successful activation of the PTK in the MH.

Now the AC opens the controlled port and the MH can transmit and receive data traffic, which is encrypted for protection on the radio link.

Concurrently with the FWH, the new AC updates the MH location information stored in the local server. Notice that only this server should be updated since the server in the home network of the MH only records the foreign domain this user is authenticated with, and not the exact

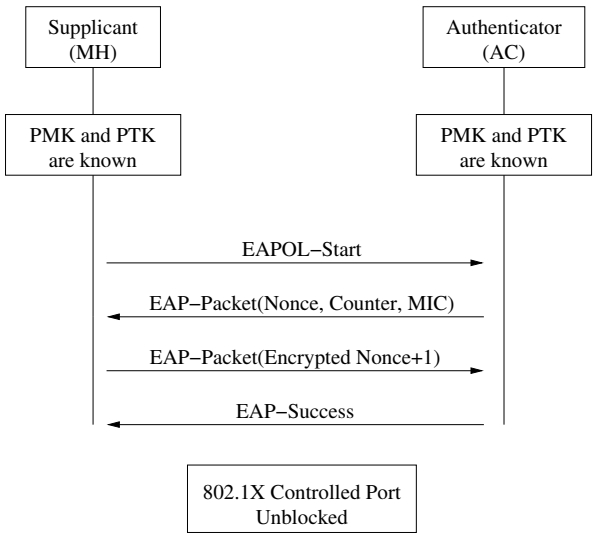
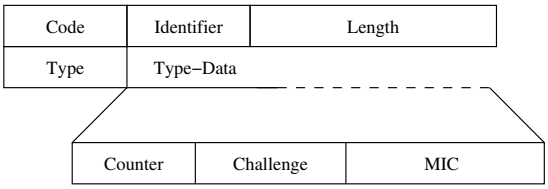
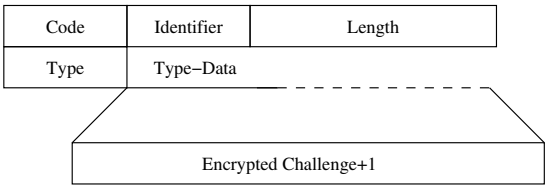


Figure 4 – Zero Authentication message flow



(a) EAP Request Packet Format



(b) EAP Response Packet Format

Figure 5 – Proposed EAP Packets format for Zero Authentication

location within that domain. Upon receiving a location update message from an AC, the AS generates the PMKs for the new neighboring ACs and sends them these keys.

It is straightforward to see that fast authentication is far shorter than full authentication, since it skips all the long lasting message exchange between the MH and the AS. Moreover, in this procedure, some parts are concurrently executed to minimize the latency. In total, fast authentication lasts for only three RTT between MH and AC, and does not add any particular mechanism to the key management scheme of 802.11i. Finally, notice that the PTK update procedure, which requires the interaction between the AS and a group of ACs and thus can potentially last for a quite long time, is started after the MH has been re-authenticated and the data flow has been re-established.

This scheme is therefore particularly attractive when the RTT from MH to AS is high. As an example, the interested reader can refer to the network architecture proposed in the TWISTER project [1], where a mixed wireless and satellite network is being implemented to cover remote and underserviced areas in Europe. In that project mobility is not an issue, but one of the services to be tested is the wireless Internet access for tourists. Notice that in this system authentication is governed by a remotely located server and the RTT is in the order of some hundreds of milliseconds.

3.3 Zero Authentication

As already said, in an n-mobility event only the WTPs change, but the AC does not. This event is expected to take place more frequently than other mobility events, due to the topological characteristic of the networks we are considering.

Thus, the authentication mechanism should be handle it as fast as possible, trying to leverage on the possession of the same (and complete) set of keys by the AC and the MH. The solution we propose is based on a simple challenge started by the AC, as shown in Figure 4**.**

After a successful association to the new WTP, the MH sends an *EAPOL-Start* packet. Although the association to APs or WTPs is not specified as a trigger event for sending an *EAPOL-Start* packet, most of the supplicants have this feature implemented making this behavior a *de facto* standard. As the AC receives such a packet it verifies, just looking at the sender MAC address, whether or not the MH has already been authenticated.

If the MH is unknown, the AC starts the full authentication procedure otherwise it must decide whether to carry out a Zero Authentication or a Fast Authentication: if no PTK is found for that MH, then the AC begins a Fast Authentication; if the AC already shares a PTK with the MH, then a n-mobility occurred and the Zero Authentication is initiated.

In the latter case a simple *EAP-Success* message could be sent, so that there is no further authentication for a MH involved in n-mobility and the data transmission can start immediately. Notice that this scheme is roughly equivalent to a MAC address based authentication: if the authenticator recognizes the MH MAC address as a valid (i.e. belonging to a pre-authenticated MH) address it opens the ports for data traffic. Therefore, the identity of the MH is confirmed by the use of the right PTK: if a rogue MH tries to enter the network using the MAC identity of an authenticated user it does not possess the right PTK, so that the AC can identify the threat and discard all the packets coming from the intruder.

This method loads the AC with the burden of controlling all incoming packets. Another simpler approach is the following: the AC replies to the *EAPOL-Start* message with

a challenge to immediately check the MH identity. In particular, we propose to create a new EAP packet type to perform this task. The format of the proposed EAP request and response packets is shown in Figure 5. It is worth to recall that response packets are always sent replying to request packets in the EAP protocol. As an example, in the message exchange of Figure 4, the second packet, sent by the AC to the MH, is of type request and the third one, sent as a reply from MH to AC, is of type response.

On the receipt of the *EAPOL-Start* message the authenticator sends to the MH an *EAP Request* packet containing a random generated number (nonce/challenge), a counter and the MIC. The counter is used to prevent possible reply attacks while the MIC, which is calculated using the KCK, guarantees the message integrity and, at the same time, confirms to the MH that the AC knows the PTK so that it can be trusted. The MH replies to the AC with the *EAP-Response* packet containing the random number+1 encrypted with the KCK. The *EAP-Success* message is sent from the AC only after a successful response from the MH.

At this same time the AC opens the controlled port for this MH and from now on it does not check the correctness of the incoming packets. In terms of elapsed time, we can observe that our Zero Authentication scheme only requires two local (i.e. between the AC and the MH) RTT against the three of the Fast Authentication and the long message flow of the Full Authentication.

In order to improve key security, we maintain the re-authentication mechanism described in the 802.11i standard. After a timeout set in the authenticator state machine a new PTK should be created. This is achieved by performing a four-way handshake initiated by the AC by means of an *EAPOL-Key* packet.

4 ENHANCING MOBILITY BETWEEN DOMAINS

According to our proposed authentication method, a MH can easily move within a domain, but when it enters a new domain a Full Authentication should be carried out (recall that this is the mobility event referred to as M-mobility). In this last section we propose an enhancement that allows MHs to freely move among different domains without being forced to fully authenticate when an M-mobility event occurs.

The idea is to extend the Fast Authentication scheme to a multi-domain scenario. To do this, the main requirement is that both source and destination domains had previously agreed on a *mutual trust relationship*. We define as mutual trust relationship a particular agreement where the ASes of the involved parties can exchange the authentication information related to their MHs using a protected communication link. In this way, when a MH moves from his HN to a Foreign Network (FN) with whom such an agreement exists, its authentication context is transmitted to the FN AS by the HN AS so that the FN server can directly authenticate the MH using the Fast Authentication procedure. In the following of the paper we will refer to this network as a Trusted Foreign Network (TFN).

The MH authentication context is copied in the TFN server only when the MH reaches the boundary between its current network and the TFN, i.e. when it authenticates to an AC close to the TFN, i.e. to a Border Access Controller (BAC). Recalling the description of the Fast Authentication procedure, it can be noticed that the TFN is a “special neighbor” of the BAC, so that the authentication context transfer can be done along with the location update procedure started by the BAC. When the AS receives an update location request from a BAC, it reads from a list we call Trusted Foreign Network List (TFNL) the address of the AS in the TFN to be get in touch with, establishes a secure communication link and transfers the authentication context of the MH to the remote server. From now on, both the local and the remote ASes are informed of any update concerning the MH keys and location, so that a mirror copy of the authentication context of the MH is kept in the remote server until the MH is at the border of the two domains. Obviously, if the MH moves away from the border the copy of his authentication context is deleted from the remote server.

This latter is therefore able to compute the correct PMK and to distribute it to its ACs according to the rules of the PAKD scheme. Therefore, when the MH leaves its original network and moves towards the TFN, it will find an AC that already has the correct PMK, so that a Fast Authentication can be done.

In its simplest form the trust is transitive: if a domain A has domain B in its TFNL and domain B has domain C in its TFNL, then domain A implicitly trusts domain C. This means than a MH moving from domain A to domain B can then move to domain C without being required to perform a Full Authentication. Another possibility is that any domain has its own TFNL, so that, referring to the previous example, domain C is not implicitly trusted by domain A. A Full Authentication is therefore required, unless domain C is in the TFNL of domain A. The server of domain B should check the trust relationship between A and C before transferring the authentication context of the MH to the server of domain C. The detailed protocol required to move the authentication context is currently under design in our lab.

Going back to the key management, when an AS receives the authentication context of a MH it can handle this roaming user as if it were one of its registered local users. In turns, the roaming MH can compute its PMK retrieving the MAC address of the AC it contacts in the new domain (this MAC address can be found in the response to the EAPOL-Start message sent by the MH) and can carry out a Fast Authentication.

5 FUTURE WORK

At the time of this writing, there are no performance results available for the solution proposed throughout this paper, since our project is still in its definition phase. However, a group of researchers and students of Politecnico di Torino has recently started working on an experimental test bed. The first goal is to implement the four way handshake mechanism. For this to be achieved, some modifications to the FreeRADIUS [13] and hostap [14] code will be necessary, since there are no implementation of the 802.11i standard freely available. This step achieved, the three authentication methods (i.e. Full Authentication, Fast Authentication and Zero Authentication) will be fully implemented and tested. If the empirical results obtained by these tests confirm low authentication latencies, further studies will focus on the multi-domain extension of the Fast Authentication mechanism.

6 CONCLUSIONS

This paper presents a possible solution aiming to reduce the latency introduced by current authentication schemes. The Fast Authentication procedure leverages on the distribution of the authentication key *before* a Mobile Host reaches a new Access Controller. This idea comes from the proposal dubbed Pro-Active Key Distribution. With respect to this latter, our solution overcomes all the problems affecting this mechanism. Along with an even faster authentication scheme dubbed Zero Authentication, our proposal provides a novel method for a complete, scalable, secure and fast authentication of mobile users within a single domain. Finally, we showed that with some not very complex mutual trusts this method can also be applied to a multi-domain network.

REFERENCES

[1] http://www.twister-project.net/

[2] C. Perkins, Ed., “IP Mobility Support for IPv4”, RFC 3344, IETF, August 2002; http://www.ietf.org/rfc/rfc3344.txt

[3] http://www.wi-fi.org/OpenSection/protected_access_archive.asp

[4] “Cisco AVVID Wireless LAN Design Solutions Reference Network Design (SRND)”; http://www.cisco.com/application/pdf/en/us/guest/netsol/ns178/c649/ccmigration_09186a00800d67eb.pdf

[5] IEEE 802.11, 1999 Edition (ISO/IEC 8802-11: 1999)

[6] Nikita Borisov and Ian Goldberg and David Wagner, "Intercepting Mobile Communications: The Insecurity of 802.11", Proceedings of the 7th annual international conference on Mobile computing and networking, ACM, 2001.

[7] IEEE 802.1X, 2001 Edition

[8] http://www.wi-fi.org

[9] IEEE 802.11i-2004 Amendment to IEEE Std 802.11, 1999 Edition

[10] Arunesh Mishra and Min ho Shin and Nick L. Petroni and Jr. and T. Charles Clancy and William A. Arbaugh, "Pro-active Key Distribution using Neighbor Graphs", IEEE Wireless Communications Magazine, February 2004.

[11] L. Yang, P. Zerfos and E. Sadot, “Architecture Taxonomy for Control and Provisioning of Wireless Access Points (CAPWAP)”, Internet-Draft, August 2004.

[12] B. Aboba and M. Beadles, “The Network Access Identifier”, RFC 2486, January 1999

[13] http://www.freeradius.org/

[14] http://hostap.epitest.fi/

BIOGRAPHY

Guido Albertengo is an associate professor with Politecnico di Torino. He has been working in packet networks and data communication protocols since 1980. He published than 50 contributions to journals and conferences and is member of the IEEE and of the AEIT.

Claudio Pastrone graduated from Politecnico di Torino in Telecommunication Engineering in September 2002. Since then He has been working on authentication and mobility in wireless networks. Currently he is a researcher with Istituto Superiore “Mario Boella” (ISMB) in Turin.

Giacomo Tolu is a researcher with CNIT. He graduated from Politecnico Di Torino with honours in Telecommunication Engineering in January 2005. His thesis was on Wireless LANs protocols, mobility, and authentication mechanisms.