♦ Authentication Across Heterogeneous Networks

Miroslav Živković, Milind M. Buddhikot, Ko Lagerberg, and Jeroen van Bemmel

In the beyond third-generation (3G) vision, the convergence of different access network technologies is realized and end-user devices are able to roam seamlessly among them. Seamless roaming requires authentication to a network, which may require interaction with an end user that results in the termination of ongoing service sessions and, therefore, inhibits seamless roaming. Furthermore, some access technologies may support a plethora of authentication protocols, making selection of the appropriate method(s) of authentication challenging for an end user. We propose a solution involving single sign-on authentication that allows the end user to roam between different administrative domains and access network technologies (e.g., wireline and wireless). Our solution integrates a number of authentication mechanisms and does not require any end-user interactions while roaming, thus enabling a seamless roaming experience. We describe a prototype implementation of this solution using General Packet Radio Service (GPRS)/Universal Mobile Telecommunications System (UMTS)/wireless local area network (WLAN) networks and present our conclusions using measurements on this prototype. © 2005 Lucent Technologies Inc.*

Introduction

Current trends indicate that an integrated public network will emerge in which a multitude of access network technologies will be available to enable end users to access their (subscribed) services and applications anywhere and anytime. Such seamless roaming across diverse networks and administrative domains offers significant advantages to network operators, service providers, application providers, and end users [23].

In this paper, we address the problem of transparent authentication that results from the convergence of different wireless access network technologies. We focus primarily on the interworking of wireless local area networks (WLANs) (with data rates of 1 Mb/s to 54 Mb/s and a cover range of up to 100 meters) and third generation (3G) mobile networks (with data rates of 64 Kb/s to 2.4 Mb/s and a cover range of a few kilometers), because these networks (which have complementary characteristics) will coexist and will compete to offer network access to end users. However, the issues, principles, and solutions we outline may be applicable to the integration of other wireline and wireless technologies as well.

Bell Labs Technical Journal 10(2), 39–56 (2005) © 2005 Lucent Technologies Inc. Published by Wiley Periodicals, Inc.
Published online in Wiley InterScience (www.interscience.wiley.com). • DOI: 10.1002/bltj.20093

Panel 1. Abbreviations, Acronyms, and Terms

3G—3rd generation
3GPP—3rd Generation Partnership Project
AAA—Authentication, authorization, and accounting
AES—Advanced encryption standard
AKA—Authentication and key agreement
AP—Access point
AuC—Authentication center
AUTN—Authentication token
CBC—Cipher block chaining
CCMP—Counter mode encryption with CBC-MAC Data Origin Authenticity Protocol
DHCP—Dynamic Host Configuration Protocol
EAP—Extensible Authentication Protocol
EAP-SIM—EAP method for GSM subscriber identity modules
EAPOL—EAP-over-LAN
ETSI—European Telecommunications Standards Institute
F-AAA—Foreign AAA
GPRS—General Packet Radio Service
GSM—Global System for Mobile Communications
H-AAA—Home AAA
HLR—Home location register
IEEE—Institute of Electrical and Electronics Engineers
IETF—Internet Engineering Task Force
IMSI—International mobile subscriber identifier
IP—Internet Protocol
LAN—Local area network
MAC—Medium access control
MAC—Message authentication code
MN—Mobile node
NAI—Network access identifier
PAE—Port access entity
PEAP—Protected EAP
PIN—Personal identification number
RADIUS—Remote Authentication Dial-in User Service
RAN—Radio access network
RAND—Random challenge number
SGSN—Serving GPRS support node
SIM—Subscriber identity module
SKE—Shared key exchange
SRES—Signed response to authentication challenge
SRP—Secure Remote Password
SS7—Switching System 7
SSO—Single sign-on
TKIP—Temporary Key Integrity Protocol
TLS—Transport Layer Security
TTLS—Tunneled Transport Layer Security
UDP—User Datagram Protocol
UMTS*—Universal Mobile Telecommunications System
USIM—Universal subscriber identity module
VLR—Visiting location register
WEP—Wired Equivalent Privacy
WLAN—Wireless local area network
WRAP—Wireless Robust Authentication Protocol
XMAC—Expected MAC
XRES—Expected response to authentication challenge

Two aspects common to WLAN and 3G technologies are [22]:

- Service access is granted when the end user or the device is authenticated by an authentication, authorization, and accounting (AAA) server in the network. At the time of service provisioning, an end user is assigned a home area and an AAA server—called a home AAA (H-AAA) server—where user credentials and additional information are stored in a profile. This information can be service class (e.g., gold, bronze, or silver), service parameters (e.g., minimum bandwidth), or types of services (e.g., virtual private network [VPN], voice, or data).
- The data is encrypted before it is transmitted on the air interface between the base station and the end-user device.

When the end user roams to a portion of the network other than the home area, the authentication process involves a foreign AAA (F-AAA) server that eventually communicates with the user's H-AAA server. This requires the interworking of the different authentication mechanisms used in different networks. While the authentication mechanisms within 3G networks have been standardized since the beginning of the development of these networks, this is certainly not the case for WLAN networks. This may lead to a situation in which an end user has to use different authentication mechanisms to access different networks. Some of these mechanisms may require end-user interaction, making it hard to realize seamless transparent roaming. We define seamless roaming [23] as a continuous process in which the end user is minimally aware of changes occurring at the network level and service disruption is a rare event. Such seamless roaming between different networks and administrative domains requires automating otherwise manual tasks, such as reconfiguring a device, typing in passwords, or buying a new access token to use a particular public WLAN.

Our paper presents a solution to the problem of seamless authentication in converged 3G/WLAN network architecture. This solution meets the following requirements:

- It enables the integration of the different authentication mechanisms that exist within access networks both on the client side and on the side of the access network(s);
- It requires minimal intervention from an end user during roaming; and
- It requires minimal technical knowledge (e.g., the types of authentication mechanisms and their configuration) during set-up time.

Our solution, which is operational in our local testbed, consists of two key components:

- *A network service element* used for network provisioning, and
- *Software* (called the SmartClient) on client devices.

In the following sections, we describe the design and implementation of the SmartClient as well as various network components and authentication methods relevant to our prototype in greater detail. The paper is organized as follows. We start with the choice of architecture for our prototype, explaining why so-called loosely coupled interworking is the architecture on which our prototype has been built. Then we describe concepts and technologies used in the prototype, like single sign-on (SSO) and some of the authentication methods used in Global System for Mobile Communications (GSM), General Packet Radio Service (GPRS), Universal Mobile Telecommunications System (UMTS*), and WLAN networks. Finally, we describe our prototype, give performance results, and briefly explain the limitations of the prototype as a foundation for future work.

WLAN/3G Integration Architecture

Since we focus on the interworking of WLAN and 3G networks, we have to choose between the two main architectural approaches identified in the standardization efforts of the European Telecommunications Standards Institute (ETSI) [9]: tightly coupled interworking and loosely coupled interworking. In the following, we briefly describe the advantages and disadvantages [8, 11] of each architecture and explain why the loosely coupled interworking architecture is more suitable for our prototype.

A tightly coupled approach views WLANs, such as the 802.11 WLAN of the Institute of Electrical and Electronics Engineers (IEEE), as just another wireless access technology within the 3G radio access network (RAN). The 802.11 WLAN network emulates functions natively available in 3G access networks (e.g., authentication, signaling, transport, and billing). One of the drawbacks of this approach is that a mobile node has to implement the 3G protocol stack on top of its standard 802.11 network interface cards. Moreover, all 802.11 traffic is injected into the core network using 3G protocols. With respect to the authentication, only authentication based on the ETSI universal subscriber identity module (USIM) [10] can be used. This requires the use of USIM readers that are built either into the 802.11 cards or into the subscriber device.

The loosely coupled approach does not require the WLAN to conform to the 3G access/core network interfaces. The WLAN network may serve users visiting from other networks as well as local subscribers. This approach completely separates the data paths in the WLAN and 3G networks; the 802.11 data traffic is never injected into the 3G core network, yet the end user still receives seamless access. The loosely coupled integration approach makes it possible to deploy and traffic engineer the WLAN and 3G networks independently. Operators of 3G networks can benefit from other operators' WLAN deployments without extensive capital investments. At the same time, they can continue to deploy 3G networks using well-established engineering techniques and tools. Furthermore, while roaming agreements with many partners can result in widespread coverage, including key hotspot areas, users benefit from having just one subscription for all network access. They no longer need to establish separate accounts with mobile operators that are in different regions or that use different access technologies. Unlike the tightly coupled approach, this architecture allows a mobile operator to provide its own public hotspots, interoperate through roaming agreements with WLAN and 3G operators, and/or manage privately installed enterprise WLANs.

In loosely coupled architecture, some features (e.g., the AAA databases and policies) are shared and maintained during the handover process [12]. The elements of both networks are still the same and changes are minimized, but the end user should have the same level of security for both WLAN and 3G network access. This is an attractive goal for both the subscribers and the operators. If different providers manage both WLAN and 3G networks, a preestablished roaming agreement between those providers is necessary to guarantee easy access, connectivity, and billing.

The use of a single authentication mechanism in the tightly coupled architecture runs counter to our assumption that different WLAN access networks may use different authentication mechanisms. Even more important is the fact that the 3rd Generation Partnership Project (3GPP) standardization work that addresses the feasibility of the interworking and interworking architectures of the UMTS and WLAN networks [2, 3] has adopted the loosely coupled interworking architecture for release 6. Therefore, we have based our solution on the loosely coupled architecture.

Single Sign-On

Single sign-on is a concept that has recently become popular in both the telecommunications and information technology (IT) industries. The term is used in different contexts with diverse meanings, but we use the following definition: single sign-on (SSO) is the ability of a system to authenticate an end user once for access to multiple distinct resources, in our case access networks. The key characteristic of SSO is that the end-user experience is seamless. The end user does not have to interact with the system (e.g., to enter a username and password) or reauthenticate when accessing a different network.

Implementing this so-called non-obtrusive authentication model has some implications for the selection of appropriate authentication protocols. Currently, users are typically required to enter a personal identification number (PIN) code when enabling their mobile phone, and current WLAN networks often use a Web-based login that requires an end user to enter a username and password in a Web page each time the end user associates with a different hotspot as well as after the session has expired. Such an authentication protocol is not acceptable from an SSO perspective.

However, in some cases it is possible to make an authentication protocol suitable for SSO by storing (i.e., caching) the credentials that would normally be entered interactively. The familiar *remember my password* feature of popular Web browsers is an example of this. This approach compromises security, because unauthorized users could gain access, for example, when an end user temporarily leaves a computer unattended. This would be a security risk even without SSO, but an SSO implementation that uses stored credentials increases the risk, because an attacker could access more network resources. In other words, there is a tradeoff between seamlessness and security.

The requirements imposed by an SSO solution are as follows:

- The end-user must interact *at most once* with the system for authentication purposes, and
- This authentication must be valid for the entire usage session.

We adhered to the principle of SSO in the solution we prototyped. Once supplied, credentials were used not only for authentication for network access, but also for authentication to the service provider. The authentication protocols that enabled us to adhere to the principle of SSO in our prototype are described in the following section.

Overview of Authentication Mechanisms

This section provides a brief summary of the authentication mechanisms that are used in the

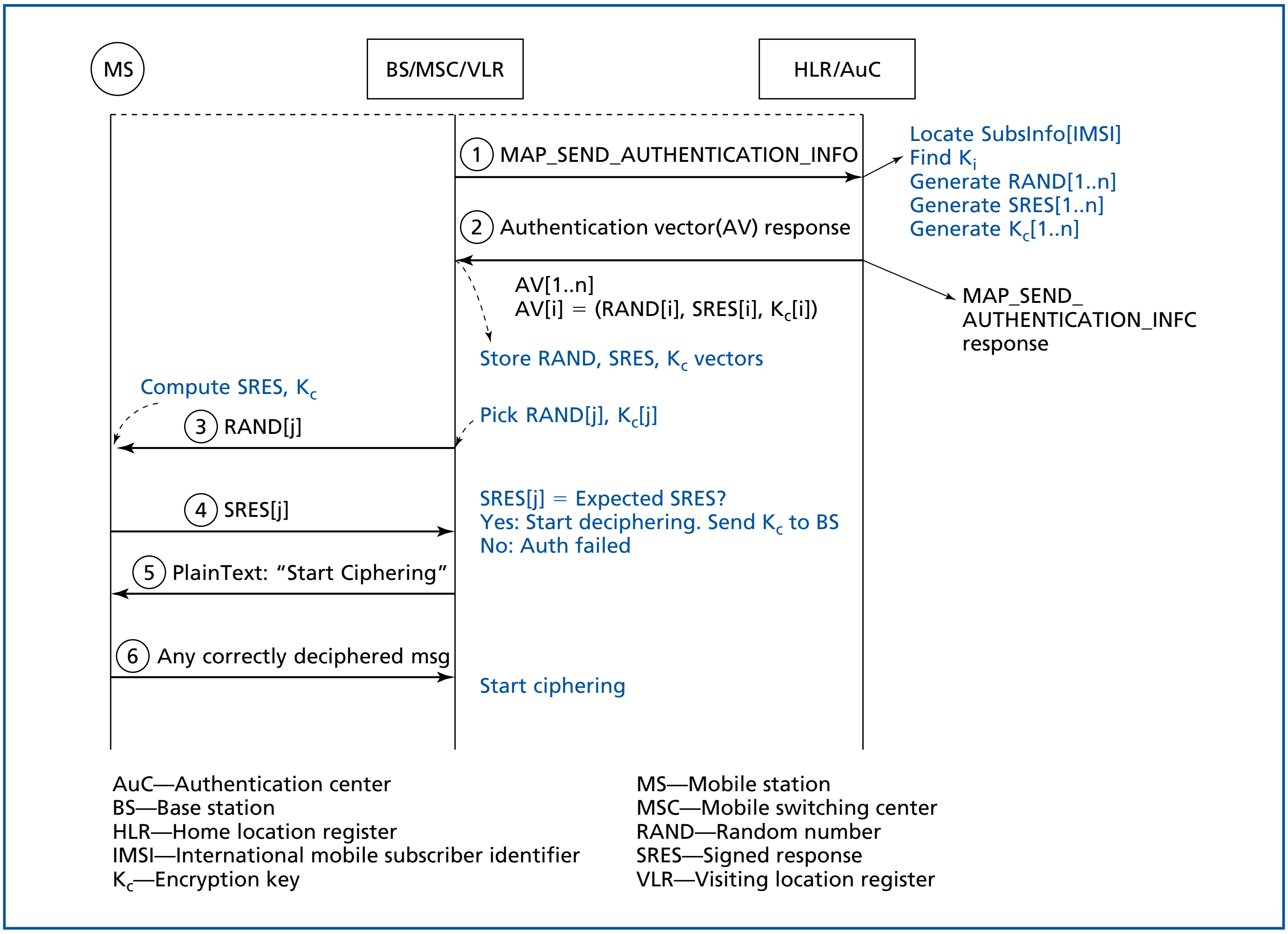

MS
BS/MSC/VLR
HLR/AuC
1 MAP_SEND_AUTHENTICATION_INFO
Locate SubsInfo[IMSI]
Find K_i
Generate RAND[1..n]
Generate SRES[1..n]
Generate K_c[1..n]
2 Authentication vector(AV) response
AV[1..n]
AV[i] = (RAND[i], SRES[i], K_c[i])
MAP_SEND_
AUTHENTICATION_INFC
response
Store RAND, SRES, K_c vectors
Compute SRES, K_c
3 RAND[j]
Pick RAND[j], K_c[j]
4 SRES[j]
SRES[j] = Expected SRES?
Yes: Start deciphering. Send K_c to BS
No: Auth failed
5 PlainText: "Start Ciphering"
6 Any correctly deciphered msg
Start ciphering
AuC—Authentication center
BS—Base station
HLR—Home location register
IMSI—International mobile subscriber identifier
K_c—Encryption key
MS—Mobile station
MSC—Mobile switching center
RAND—Random number
SRES—Signed response
VLR—Visiting location register

Figure 1.
GSM/GPRS SIM authentication.

GSM/GPRS, UMTS, and WLAN networks. Most of these mechanisms are successfully integrated in our prototype.

GSM/GPRS SIM Authentication Mechanisms

In GSM/GPRS, authentication of the end user is based on a challenge-response mechanism that supports only one-way authentication in which the network authenticates the client device. A simplified view of the authentication (see **Figure 1**) follows:

1. A master key K is shared between the end user's subscriber identity module (SIM) and the home network.
2. The visiting location register (VLR) or serving GPRS support node (SGSN) authenticates the terminal by sending an authentication request to the home location register (HLR), which answers with a set of GSM triplets. Each triplet contains a 16-byte random number, *RAND,* a 4-byte signed response, *SRES,* and an 8-byte encryption key, k_c.
3. The VLR/SGSN picks one of the received triplets and sends an authentication challenge in the form of RAND to the terminal.
4. The terminal computes an expected response, *XRES,* and key, k_c using its local SIM card and forwards *XRES* to VLR/SGSN.
5. If the *XRES* matches the *SRES* in the selected triplet, the VLR/SGSN concludes that the client device is authenticated.

Note that the master key, K, which is shared between the HLR and the client SIM, and the encryption key, k_c, are never transmitted over the air.

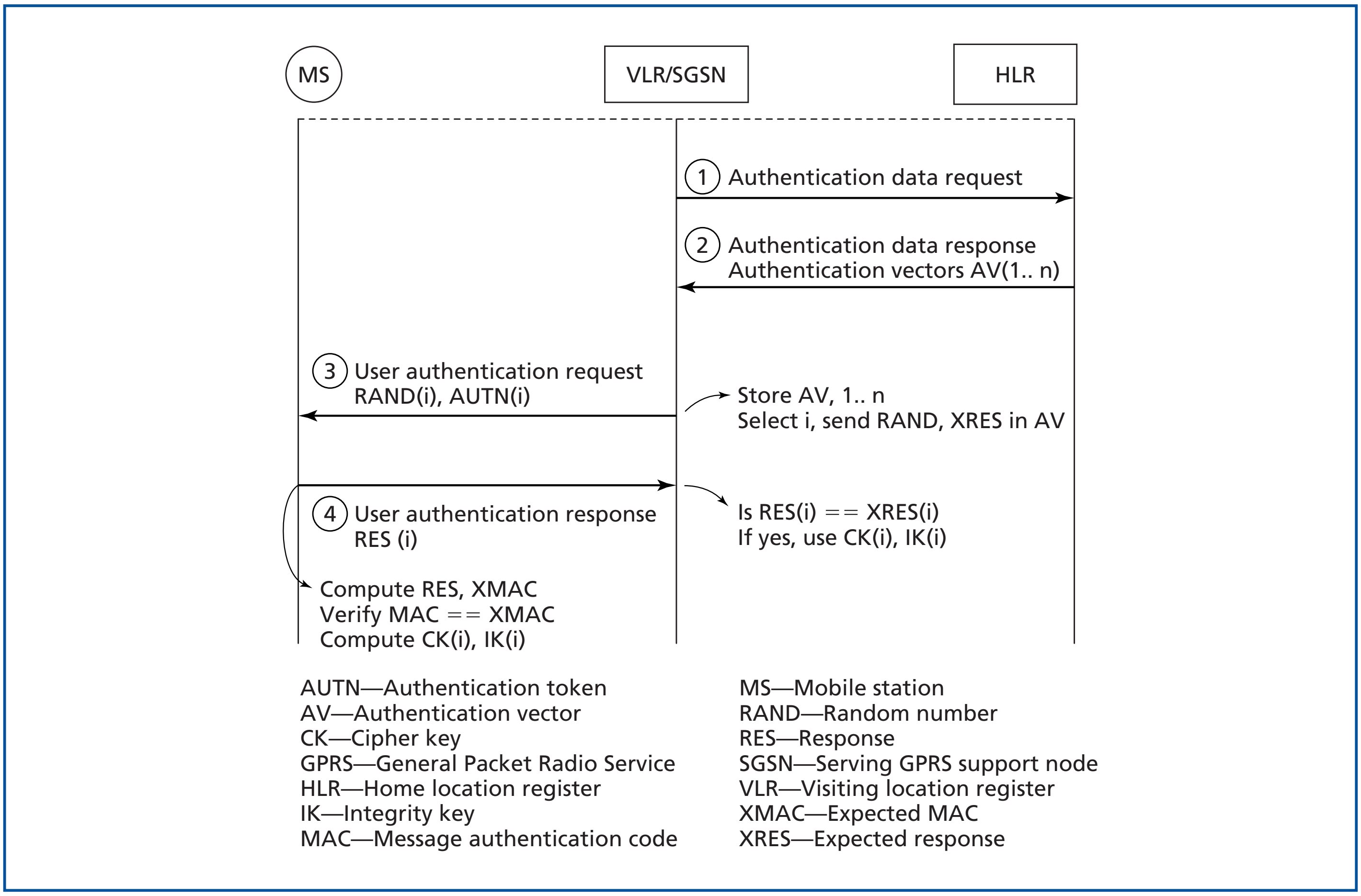

MS
VLR/SGSN
HLR
1 Authentication data request
2 Authentication data response
Authentication vectors AV(1.. n)
3 User authentication request
RAND(i), AUTN(i)
Store AV, 1.. n
Select i, send RAND, XRES in AV
4 User authentication response
RES (i)
Is RES(i) == XRES(i)
If yes, use CK(i), IK(i)
Compute RES, XMAC
Verify MAC == XMAC
Compute CK(i), IK(i)
AUTN—Authentication token
AV—Authentication vector
CK—Cipher key
GPRS—General Packet Radio Service
HLR—Home location register
IK—Integrity key
MAC—Message authentication code
MS—Mobile station
RAND—Random number
RES—Response
SGSN—Serving GPRS support node
VLR—Visiting location register
XMAC—Expected MAC
XRES—Expected response

Figure 2.
UMTS AKA protocol exchange.

UMTS AKA Authentication Mechanism

The UMTS security mechanisms (including authentication) are specified by the 3GPP [1]. The UMTS authentication and key agreement (AKA) method is based on a challenge-response mechanism like that used in GSM/GPRS SIM authentication. The significant difference is that AKA enables mutual authentication: the network authenticates the mobile station and the mobile station authenticates the network. A simplified view of the authentication (see **Figure 2**) follows:

1. A master key, *K,* is shared between the end-user's USIM and the home network. This key is never transmitted over the air.
2. The VLR or SGSN requests authentication data from the HLR, which answers with an authentication vector. Each authentication vector contains a randomly generated number, *RAND,* an expected response, *XRES,* a cipher key, *CK,* an integrity key, *IK,* and a so-called authentication token, *AUTN.* The *AUTN* contains, among other things, the message authentication code (MAC).
3. The VLR/SGSN stores the authentication vector, picks *RAND* from authentication vector, and sends an authentication challenge to the mobile station. The authentication challenge consists of *RAND* together with *AUTN*.
4. The mobile station computes an expected MAC, *XMAC,* from the challenge, and compares *XMAC* with the MAC contained in the received authentication token. If *XMAC* matches MAC, the mobile station assumes that the response is from a network that is trusted by its home environment. After the network authentication has been checked, the mobile station computes

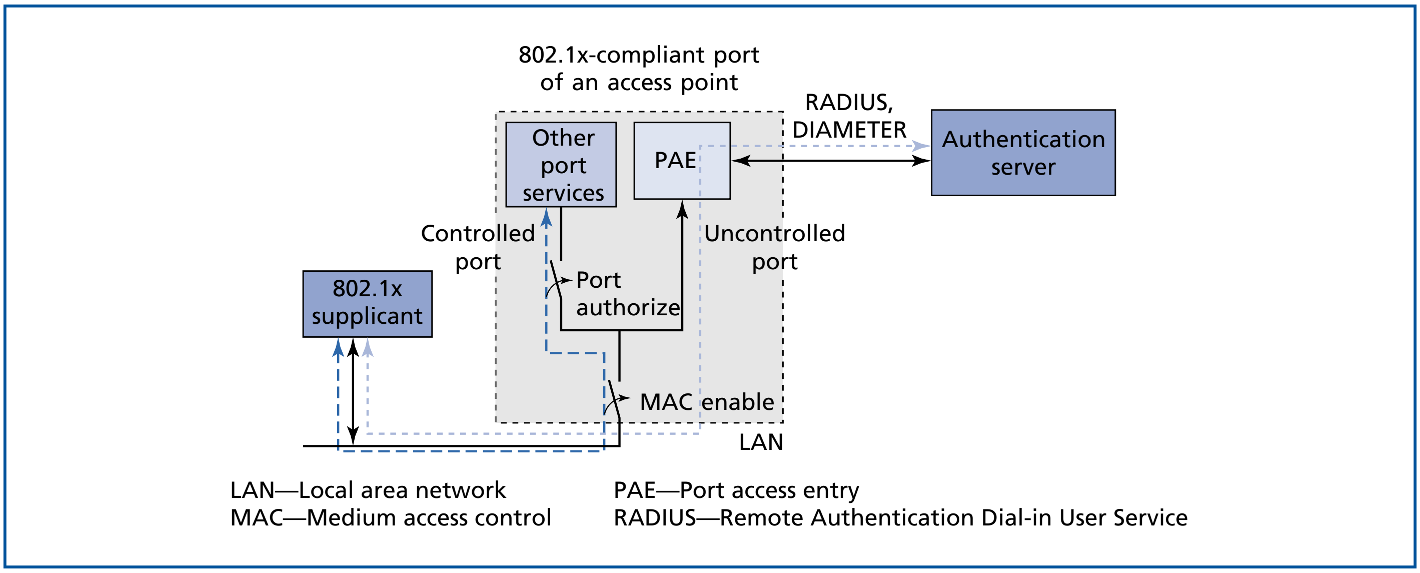


Figure 3.
802.1x Protocol entities.

the cipher key and the response to the authentication challenge, *RES*. The mobile station then sends *RES* to the network.

5. If *RES* matches *XRES*, VLR/SGSN concludes that the mobile station is authenticated. It uses the cipher key from the authentication vector. Therefore, the cipher key is never transmitted over the air.

WLAN Authentication Mechanisms

The IEEE 802.11 standard currently supports two different forms of mobile node (MN) authentication:

- Open system authentication, and
- Shared key authentication.

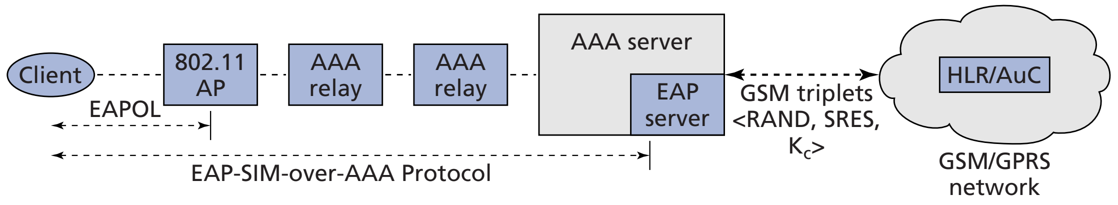
Open system authentication is essentially a simple, null authentication algorithm and is therefore of little practical interest in public networks. Shared key authentication depends on establishing that the MN is in possession of a shared secret key also known to the 802.11 access point (AP). The required secret key must have been delivered to the MN via a secure out-of-band mechanism that is independent of the WLAN being accessed. Once the MN is authenticated, its traffic to the AP and to the other nodes is encrypted using Wired Equivalent Privacy (WEP) [16]. Drawbacks of this scheme are:

- The shared key used for encryption is also used for authentication. This means that, every time a new MN is added to or an existing MN leaves the trusted shared relationship, the key has to be changed and delivered reliably to each MN. This makes the scheme cumbersome and unsuitable for public networks, which require the use of per-user, per-session keys.
- WEP is not an “industrial strength” encryption protocol [7]. It has been shown that an adversary can easily sign on to an 802.11 network and alter or decrypt WEP-encrypted communication without knowledge of the shared key.

The new IEEE 802.11i standard [18] is considered a significant improvement for the public environment. It employs the IEEE 802.1x port access control standard [17], which specifies the use of the EAP-over-LAN (EAPOL) protocol between the MN and the AP to perform per-session user authentication. The EAPOL protocol defines two entities (see **Figure 3**):

- *A port access entity* (PAE) on the authenticator in the AP, and
- *An 802.1x supplicant* (i.e., software) on the client device.

The authenticator has two ports: an uncontrolled port that allows only unencrypted EAPOL packets to pass and a controlled port that allows regular packets to pass only after an Extensible Authentication



AAA—Authentication, authorization, and accounting
AP—Access point
AuC—Authentication center
EAP—Extensible Authentication Protocol
EAPOL—EAP over local area network
EAP-SIM—EAP method for GSM subscriber identity modules

GPRS—General Packet Radio Service
GSM—Global System for Mobile Communications
HLR—Home location register
K_c—Encryption key
RAND—Random number
SRES—Signed response

Figure 4.
Model for EAP-SIM authentication.

Protocol (EAP) [6] authentication exchange through the uncontrolled port is successful. The PAE communicates with an AAA authentication server using the EAP-over-RADIUS protocol to complete the EAP authentication.

EAPOL packets use Ethernet encapsulation and can be one of the following types:

- *EAPOL start* and *EAPOL logoff packets,* which signify the start and end of an EAPOL session;
- *EAP packets,* which correspond to the core authentication exchange; and
- *EAPOL-Key packets,* which are used to set up various keys used in the user session, such as keys for encryption, message authentication, and anonymity.

The specific EAP authentication method used dictates the format and content of the EAP messages. Some of the well-known EAP-schemes are:

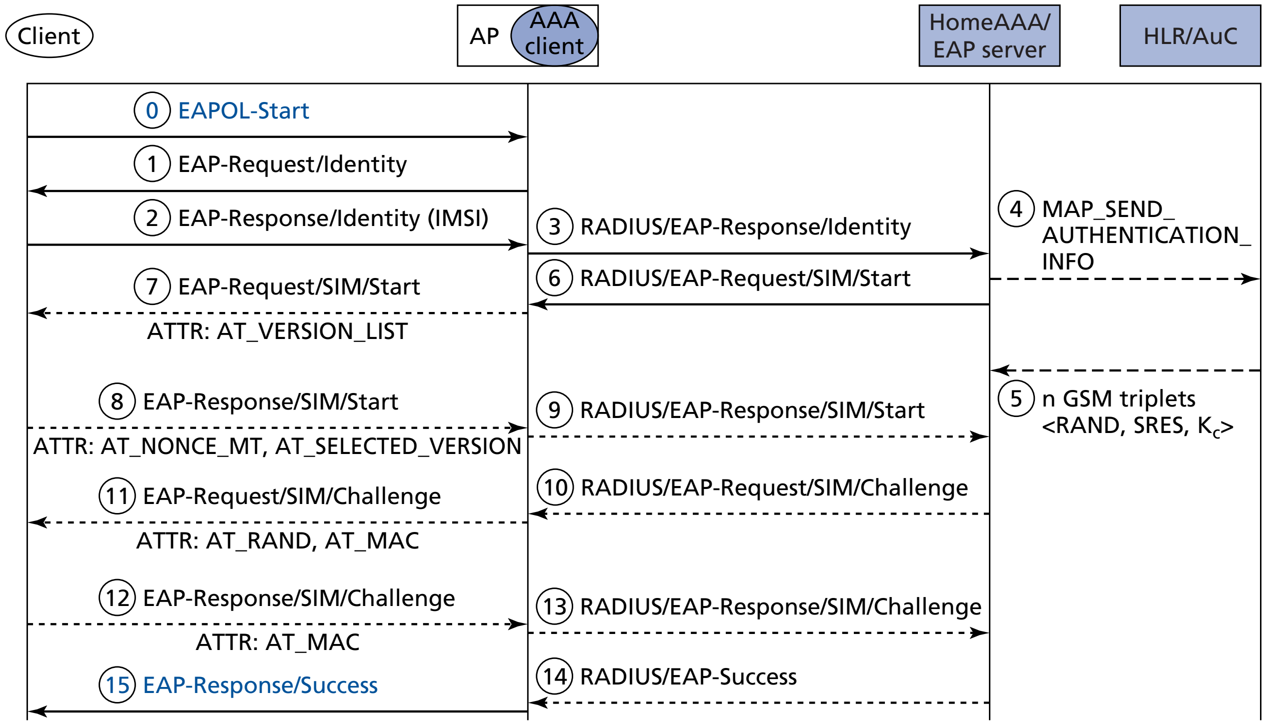
- *Certificate-based schemes:* EAP-Transport Layer Security (TLS) [4], EAP-Tunneled Transport Layer Security (TTLS) [13], and EAP-protected EAP (PEAP) [20];
- *Symmetric-key-based schemes:* EAP-SIM [14], EAP-AKA [5], and EAP-SKE [21]; and
- *Password-based schemes:* EAP-SRP, which is based on Secure Remote Password (SRP) [24], EAP-MD5 [6], and SecureID.

Additionally, individual per-user session keys, which are used for encryption and integrity protection, are derived and distributed during the authentication exchange with the H-AAA server. This eliminates the need for any preconfiguration of keys and medium access control (MAC) addresses in WLAN APs and requires only a security association between the end user and its H-AAA server in the home service provider network.

The 802.11i standard also specifies a key-derivation procedure to derive encryption, authentication, and integrity protection keys. It standardizes three protocols, namely, Temporal Key Integrity Protocol (TKIP), Counter Mode Encryption with Cipher Block Chaining (CBC)-MAC Data Origin Authenticity Protocol (CCMP), and Wireless Robust Authentication Protocol (WRAP), for protecting data transfers. The TKIP protocol improves WEP encryption to acceptable levels by using larger initialization vector (IV) values, per-packet keying, and a keyed message authentication code (MAC) function called Michael. TKIP also provides a graceful migration path for existing infrastructure and client devices, because it can be supported on legacy hardware by simple firmware upgrades. CCMP and WRAP are both 128-bit advanced encryption standard (AES)-based protocols; implementation of CCMP is mandatory for 802.11i compliance. However, both these protocols require new hardware.

EAP-SIM. The primary purpose of EAP-SIM schemes is to allow a GSM/GPRS service provider to use the credentials and user profile stored in the HLR to authenticate and charge end users when they roam to a WLAN network.

Figure 4 shows the entities involved in the EAP-SIM authentication process:


Client
AP
AAA client
HomeAAA/ EAP server
HLR/AuC
0 EAPOL-Start
1 EAP-Request/Identity
2 EAP-Response/Identity (IMSI)
3 RADIUS/EAP-Response/Identity
4 MAP_SEND_ AUTHENTICATION_ INFO
5 n GSM triplets <RAND, SRES, K_c>
6 RADIUS/EAP-Request/SIM/Start
7 EAP-Request/SIM/Start
ATTR: AT_VERSION_LIST
8 EAP-Response/SIM/Start
ATTR: AT_NONCE_MT, AT_SELECTED_VERSION
9 RADIUS/EAP-Response/SIM/Start
10 RADIUS/EAP-Request/SIM/Challenge
11 EAP-Request/SIM/Challenge
ATTR: AT_RAND, AT_MAC
12 EAP-Response/SIM/Challenge
ATTR: AT_MAC
13 RADIUS/EAP-Response/SIM/Challenge
14 RADIUS/EAP-Success
15 EAP-Response/Success

AAA—Authentication, authorization, and accounting
AP—Access point
ATTR—Attribute
AuC—Authentication center
EAP—Extensible Authentication Protocol
EAPOL—EAP over local area network
GSM—Global System for Mobile Communications
HLR—Home location register

IMSI—International mobile subscriber identifier
K_c—Encryption key
RADIUS—Remote authentication dial-in user service
RADIUS/EAP—EAP encapsulated within RADIUS
RAND—Random number
SIM—Subscriber identity module
SRES—Signed response

Figure 5.
EAP-SIM protocol exchange.

- The client device contains a SIM card with the user credentials and runs EAP-SIM supplicant software. The EAP-SIM protocol-related traffic between the client device and the WLAN AP consists of EAP messages.
- The Remote Authentication Dial-in User Service (RADIUS) client (i.e., the AAA client) in the access point encapsulates the EAP-SIM packets in RADIUS messages and sends them to the user's H-AAA server. These RADIUS packets may be forwarded by multiple AAA relays before reaching the H-AAA server.
- The AAA server communicates with a logically or physically distinct EAP server that implements the EAP-SIM protocol. It also interfaces to the Switching System 7 (SS7) network to communicate with the appropriate HLR to obtain EAP-SIM triplets (i.e., $\langle RAND, SRES, K_c \rangle$).

EAP-SIM protocol flow, which is illustrated in **Figure 5**, is as follows (the message numbers refer to the numbers in Figure 5):

1. When the client device MN wants to access a WLAN network, it associates with the AP and sends the EAPOL-Start message (Message 0) to begin the authentication session.
2. The AP sends an EAP-Request/Identity message (Message 1) to the client to determine its network access identifier (NAI).

3. The client supplicant supplies the NAI (e.g., *username*@3goperator.com) in an EAP-Response/Identity message (Message 2).
4. The AP encapsulates this message in a RADIUS access request message (Message 3) and forwards it—either directly or via an AAA broker network containing relays—to the H-AAA server.
5. The EAP-SIM state machine sends the SEND_MAP_AUTH_INFO command (Message 4) to the HLR to request the *n* (where *n* = 1 to 5) GSM triplets for the end user with the international mobile subscriber identifier (IMSI) included in the NAI.
6. The AAA server uses the NAI to locate the user profile and determines that the user needs to be authenticated using EAP-SIM. The EAP-SIM state machine implemented in the AAA server issues an EAP-Request/SIM/Start message that is relayed back to the AP in a RADIUS access challenge message (Message 6). This message marks the start of the EAP-SIM session.
7. The AP decapsulates the EAP message and relays it to the client (Message 7). The SIM/Start message includes a list of EAP-SIM versions in the AT_VERSION_LIST attribute.
8. The HLR forwards the triplets requested in step 5 to the AAA server (Message 5).
9. The MN sends an EAP-Response/SIM/Start response message (Message 8) that includes the version selected in the AT_SELECTED_VERSION attribute. It also includes a NONCE challenge, N1, for the authenticator in the AT_NONCE_MT attribute.
10. This response is relayed to the AAA server (Message 9).
11. The H-AAA server sends an EAP-Request/SIM/Challenge message (Message 10) containing the *n* GSM RANDs in the AT_RAND attribute. It also computes the message authentication code (MAC) byte string (*EAP PKT* | *NONCE_MT*) and includes that in the AT_MAC attribute.
12. The above message is relayed to the MN (Message 11).
13. The MN verifies the AT_MAC. It runs the GSM A3 algorithm in the SIM card on *n*RAND* to get *n*SRES* and computes the MAC over (*EAP PKT* | *n*SRES*). It sends an EAP-Response/SIM/Challenge (Messages 12 and 13) to the AAA server and includes the MAC in the AT_MAC attribute. Note that no SRES is sent.
14. The H-AAA server verifies AT_MAC (using SRES) and sends an EAP-Response/Success message (Message 15) encapsulated in a RADIUS Accept message (Message 14). It also includes the session master secret in two attributes MS-MPPE-RECV and MS-MPPE-SEND. The session master secret is removed by the AP to drive the 802.11i key derivation process.

The communication between the H-AAA server and the HLR can be done in two different ways:

- *Synchronously:* The H-AAA server obtains GSM triplets from the HLR only when the version negotiation using EAP-SIM/Start request/response packets completes. The advantage of this method is that the HLR is contacted only when needed, which improves performance, because the HLR is not unnecessarily loaded. On the other hand, total authentication latency is higher, because the EAP-SIM/Challenge exchange waits until the HLR exchange completes.
- *Asynchronously:* The H-AAA server obtains GSM triplets from the HLR without waiting for the EAP-SIM/Start request/response to complete. The disadvantage of this method is that, when the EAP-SIM/Start exchange fails, triplets supplied by the HLR are wasted load on the HLR. On the other hand, if the EAP-SIM/Start version negotiation is successful, pipelining improves overall authentication latency.

Other WLAN authentication mechanisms. Below, we briefly describe some of the authentication mechanisms integrated in our solution.

- *EAP-TLS:* This authentication protocol is based on client- and server-side certificates. It implements the Internet Engineering Task Force (IETF) TLS protocol in the EAP. It has not found much acceptance, because of the overhead of managing a large number of client-side certificates. However, the EAP-TLS supplicant is standard in the Windows* XP operating system, and EAP-TLS is

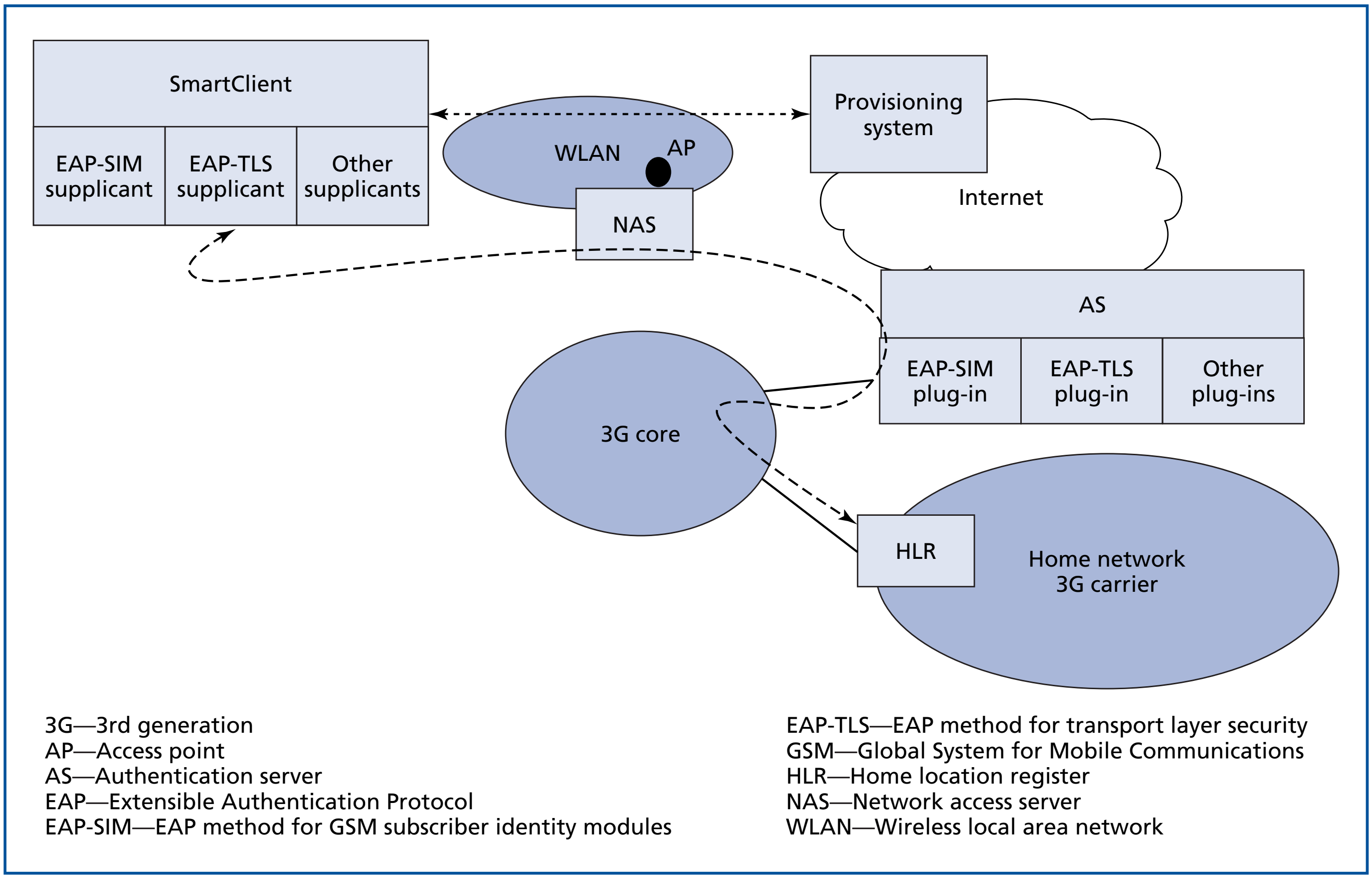

SmartClient
EAP-SIM supplicant
EAP-TLS supplicant
Other supplicants
WLAN
AP
NAS
Provisioning system
Internet
AS
EAP-SIM plug-in
EAP-TLS plug-in
Other plug-ins
3G core
HLR
Home network 3G carrier
3G—3rd generation
AP—Access point
AS—Authentication server
EAP—Extensible Authentication Protocol
EAP-SIM—EAP method for GSM subscriber identity modules
EAP-TLS—EAP method for transport layer security
GSM—Global System for Mobile Communications
HLR—Home location register
NAS—Network access server
WLAN—Wireless local area network

Figure 6.
Architecture of the testbed.

supported in most commercial and open source RADIUS servers, including Lucent's *NavisRadius*™.

- *PEAP, EAP-TTLS:* The PEAP and the TTLS are very similar protocols. Both eliminate the need for client-side certificates and require only a server certificate. Both are two-phase protocols. In phase 1, they use a server-side certificate to authenticate the server and to negotiate a cipher suite that is used to set up a protected/encrypted tunnel between the client and the server. The authentication traffic is then transported in this secure tunnel. This eliminates the problem faced by most EAP methods, in which the EAP/Identity message at the start of an EAP exchange goes in the clear. Phase 2 is used to do such things as client authentication and session key setup. The TTLS protocol exchanges attribute-value pairs (AVPs) that are similar to RADIUS attributes and support any authentication method. On the other hand, the PEAP only allows EAP methods (e.g. EAP-MD5, EAP-SIM, and EAP-AKA) to proceed in phase 2.

This overview indicates the variety of authentication methods deployed in a given access network (especially in a WLAN) and the differences between the methods deployed in different access networks. Our solution enables the integration of virtually any authentication method; its details are described in the next section.

Prototype

The architecture of the testbed we developed is illustrated in **Figure 6**. As explained before, it is based on a loosely coupled 3G/WLAN integration architecture and it uses the SSO principle for authentication to the network. A typical usage scenario for an end user is as follows:

1. The end user switches on a mobile device. The device is equipped with different network

interfaces, and the credentials (e.g., certificates and SIM cards) used in different authentication protocols are preinstalled on it. After the discovery of available networks, the client obtains the available authentication methods from the provisioning system. Other information (e.g., costs) can be obtained as well. The very first time an end user connects to the network there is a bootstrap problem that we discuss later.

2. Based on the information obtained and on user preferences, the client software selects an access network (AN_1). Because the authentication method for AN_1 is obtained from the provisioning system (see previous step), authentication can be performed. In some cases, the end user may be asked to provide the credentials required (e.g., a password) for a particular authentication method. These credentials may be cached for later use.
3. The end user switches to another access network (AN_2). The information about the available authentication methods for this network has already been obtained (see first step). The authentication method for AN_2 may be completely different from the authentication method for AN_1, but the client supplies the necessary credentials; there is no user interaction.

If the client discovers a new access network, it pulls network information from the provisioning system. When information about a particular network (e.g., the supported authentication protocols) changes, the changed information is provided (i.e., pushed) to the client by the provisioning system. If the access network supports multiple authentication mechanisms, the preferred order of applying these authentication methods is supplied by the provisioning system as well, so the client "knows" which authentication method should be applied.

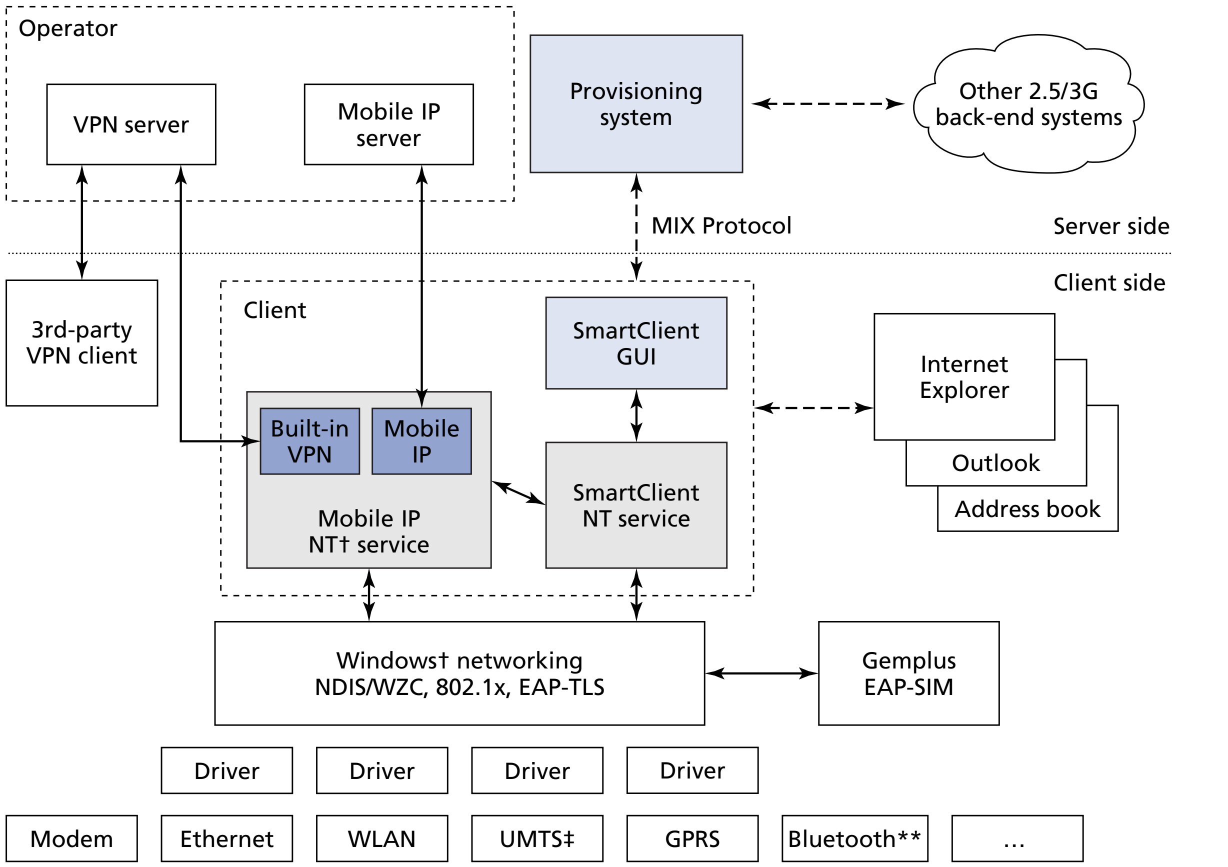
The main components of the architecture are:

- The client software,
- The authentication server,
- The network access server, which proxies the authentication messages between the client and the authentication server,
- The HLR, which is used in case of EAP-SIM authentication, and
- The provisioning system.

The client software (named SmartClient) has been implemented for the Windows XP platform. Since one of the goals was to have an SSO system, it integrates a number of EAP supplicants (e.g., EAP-SIM and EAP-TLS) and it supports the majority of the authentication processes in use today. While some supplicants are already integrated into the Windows XP operating system (e.g., EAP-TLS), we used additional Gemplus*-supplied EAP-SIM supplicant software. The credentials for EAP-SIM-based authentication are accessed using a universal serial bus (USB) SIM card reader. The client software enables easy integration of any other network authentication protocol; we have successfully tested and integrated the EAP-TLS, EAP-TTLS, EAP-SIM, PEAP, and EAP-MD5 authentication protocols. The client also integrates with mobility software solutions (e.g., mobile IP) and supports multiple network interfaces, such as the Ethernet, WLAN, GPRS, and Bluetooth* interfaces. In fact, any network interface supported by the Windows XP operating system is supported. The architecture of the SmartClient is shown in **Figure 7**.

The SmartClient consists of a graphical user interface (GUI) and a SmartClient task that runs in user space. It interacts in user space with the mobile-IP task, which performs mobility management at the network layer. The SmartClient offers the abstraction of an interface to the operating system (OS) protocol stack, which makes possible the integration of different network technologies. It also offers an interface to different authentication supplicants (e.g., EAP-SIM). In this way, the SmartClient hides all the authentication (and mobility) details from the applications. An important feature of the SmartClient is that it assures that the end user always has the best possible connection, as determined by the preferences the end user has defined. These preferences, as well as other relevant information for particular networks, are communicated to the SmartClient using the Mobility Information Exchange (MIX) Protocol.

The authentication server we used is Lucent's implementation of a RADIUS authentication server, the



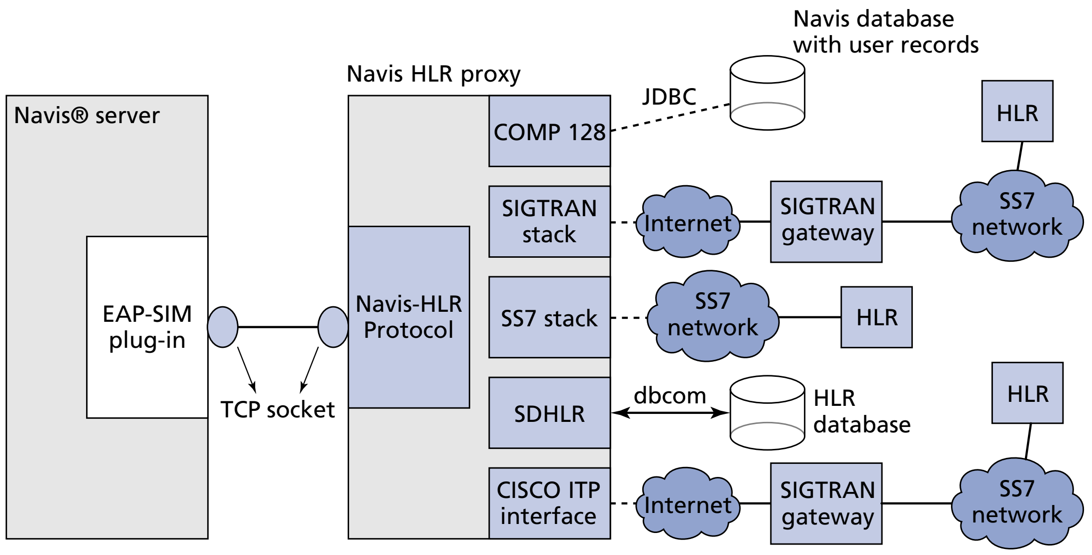
3G—3rd generation
EAP—Extensible Authentication Protocol
EAP-SIM—EAP method for subscriber identity module
EAP-TLS—EAP method for transport security layer
GPRS—General Packet Radio Service
GUI—Graphical user interface
IP—Internet Protocol
MIX—Mobility information exchange
NDIS—Network driver interface specification
UMTS‡—Universal Mobile Telecommunications System
VPN—Virtual private network
WLAN—Wireless local area network
WZC—Windows† zero configuration

† Registered trademark or trademark of Microsoft Corporation in the United States and/or other countries.
‡ Trademark of the European Telecommunications Standards Institute (ETSI) under auspices of the ITU.
** Registered trademark of Bluetooth SIG.

Figure 7.
Architecture of the SmartClient and other testbed components.

NavisRadius server. Because of its flexible architecture, which is based on plug-ins, it supports multiple authentication mechanisms. We have implemented an EAP-SIM plug-in for the *NavisRadius* server as part of the prototype (see **Figure 8**). An interesting issue is that there is no unified approach to communication between the *NavisRadius* server (i.e., the EAP-SIM plug-in) and the HLR. In order to facilitate integration of as many approaches as possible, we have introduced an abstraction in the form of the *Navis*® HLR proxy. This proxy offers a well-defined interface to the plug-in, thereby hiding details of the implementation of communication with the HLR from it.

In our prototype architecture, an end user has a subscription with a *roaming provider* that allows the end user to roam seamlessly across multiple access networks. Each roaming provider operates a provisioning system that provides the client software with the appropriate authentication parameters. The provisioning system offers a Web-based interface for managing data on networks and subscribers. For example, to add a subscriber, a new account is



EAP—Extensible Authentication Protocol
EAP-SIM—EAP method for subscriber identity module
GSM—Global System for Mobile Communications
HLR—Home location register
ITP—Internet Protocol transfer point

JDBC†—Java† database connectivity
SDHLR—Super distributed home location register
SIGTRAN—Signaling transport
SS7—Signaling System 7
TCP—Transport Control Protocol

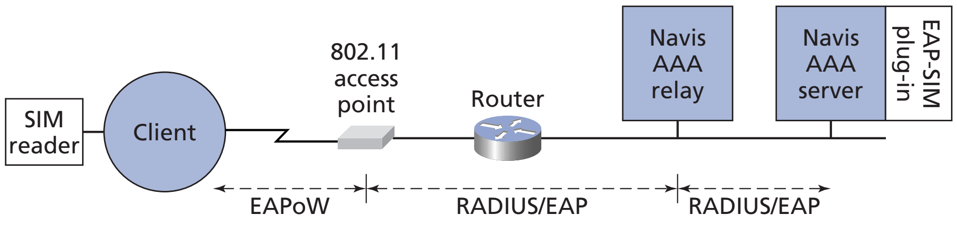
† Trademark of Sun Microsystems, Inc.

Figure 8.
Implementation of the EAP-SIM plug-in for the Navis® server.

provisioned with information such as login, e-mail address, and allowed services. Then an invitation e-mail containing a reference to a Web page where the user can download a certificate for network (and service) authentication is automatically sent to the subscriber. More details on this system (called the SmartServer) can be found in [19].

Performance comparison of EAP-SIM and EAP-TLS

We carried out measurements to characterize the performance of successful authentication using EAP-SIM and EAP-TLS. The test setup used for the measurements is depicted in **Figure 9**. In a test network with no significant load, we performed two sets of measurements for each of the authentication methods.



AAA—Authentication, authorization, and accounting
EAP—Extensible Authentication Protocol
EAPoW—EAP over wireless
EAP-SIM—EAP method for GSM subscriber identity modules

EAP-TLS—EAP method for transport layer security
GSM—Global System for Mobile Communications
RADIUS—Remote authentication dial-in user service
RADIUS/EAP—EAP encapsulated within RADIUS
SIM—Subscriber identity module

Figure 9.
Setup for performance measurements of the EAP-SIM and EAP-TLS Protocols.

In the first set, we measured the elapsed time for the complete authentication procedure. For EAP-TLS, it takes about 2 seconds on average from the first EAPOL-Start to the last EAPOL-Key. For EAP-SIM it takes about 3.5 seconds. A closer look at the EAP-SIM implementation revealed that the client's USB SIM card reader consumes about 1.5 seconds when it fetches the identity information from the SIM card. By contrast, in EAP-TLS the required certificate is accessed quickly from the local hard disk. Clearly, in a refined implementation of the SIM supplicant, identity information could be cached for subsequent authentications (assuming the same SIM card is connected to the device) and the latency could be minimized.

To compare EAP-TLS and EAP-SIM without this effect, in the second set of measurements we measured the time between the arrival of the first EAP request (excluding identity requests) and the arrival of EAPOL-Success at the client. The average time measured for EAP-TLS was 0.21 ms, while the average time measured for EAP-SIM was 0.35 ms. The difference between EAP-TLS and EAP-SIM authentication is striking, because for EAP-TLS the number of messages (i.e., 11) exchanged between client and server is more than twice the number of messages (i.e., 5) for EAP-SIM, while the measured time is around 40% shorter. The difference can be attributed to the complexity of the computations in the EAP-SIM authentication algorithm at the client or the server or both.

Whether the measured performance is acceptable in a deployed service setting depends on how the authentication is embedded in the roaming process and on the capabilities of the terminal. For example, if a terminal is equipped with multiple network adapters, it may be able to perform a "make before break" connection handover (i.e. it may be able to wait for the authentication to complete before switching to another network). If this is not possible, acceptability depends entirely on whether the service or application can cope with the authentication delay. For example, a user browsing the Web will probably not notice a service interruption of a few seconds. On the other hand, it will definitely be an impediment for real-time services like voice over IP (VoIP). In general, if the delay affects the end user's perception of the quality of the service, it is unacceptable.

Limiting Factors

This section describes the limitations of the prototype as well as options for dealing with these limitations.

Migration from a non-authenticated (W)LAN infrastructure. Existing (W)LAN clients typically assume that packets can be sent once a link layer connection has been established with the network. When authentication protocols (such as Web-based login or EAPOL) above the link layer are used, packets sent before successful authentication are discarded. It is necessary to have knowledge of the network authentication protocols being used to be able to prevent this from happening.

Bootstrap. Part of our solution is based on information about the network, in particular about the type of authentication mechanism used. The first time a network is used, this information may not be known, but to be able to fetch this information automatically the client must have a network connection. This bootstrap problem can be solved in several ways, for example, by using static provisioning if a separate 3G connection is available, or by allowing network information to pass through before authentication using EAP-TLV [15] or Dynamic Host Configuration Protocol (DHCP) attributes in case of a Web login. Another possibility would be by using conventions, for example, by encoding the authentication method in the server set ID (SSID) of a WLAN network. The current prototype supports either a secondary connection (i.e., UMTS) for retrieving network authentication settings or manual provisioning at the client side.

Network selection policies. It should be noted that the automatic authentication and selection of a new access network is not always in the user's interest. For example, the user may want to avoid networks that have an additional charge for roaming. Avoiding such networks can typically be achieved by using some kind of network selection policy on the client side. Some selection policies have been implemented as part of the SmartClient (see [23] for more details), but additional policies may be required in a real service setting.

Response time. It is obvious from our time measurements that the authentication process may take up to several (i.e., 3 to 4) seconds in the worst case (i.e., EAP-SIM authentication). Contributing factors must be studied further, but two sources seem to be client-side processing (e.g., in EAP-SIM) and the chaining of RADIUS proxy forwarding messages using the unreliable User Datagram Protocol (UDP). The reauthentication case can be optimized by caching credentials locally (under certain conditions) and/or by extending the reauthentication session timeout. Future research should also include performance measurements in a loaded network, preferably in a real service setting.

Conclusions

We have proposed a solution for the unified and seamless authentication of end users and terminals within heterogeneous networks. The solution offers SSO in a WLAN/3G loosely coupled architecture. Several different authentication protocols (as well as an EAP-SIM server) have been integrated in a prototype as a proof of concept. The limiting factors have been identified and the opportunities to resolve them have been briefly described. Future work should provide an answer to the question of the prototype's performance in a fully loaded network. Further, it would be interesting to verify the solution for access network technologies that are not included in the current testbed.

Acknowledgments

The authors would like to thank the following people for their contribution to this work: Erik Meeuwissen, Harold Teunissen, and Jacco Brok.

*Trademarks

Bluetooth is a registered trademark of Bluetooth SIG.

Gemplus is a registered trademark of Gemplus SA.

UMTS is a trademark of the European Telecommunications Standards Institute (ETSI) under auspices of the ITU.

Windows is either a registered trademark or a trademark of Microsoft Corporation in the United States and/or other countries.

References

[1] 3rd Generation Partnership Project, <http://www.3gpp.org>.

[2] 3rd Generation Partnership Project, "Feasibility Study on 3GPP System to WLAN Interworking," TR 22.934, Sept. 2003, <http://www.3gpp.org>.

[3] 3rd Generation Partnership Project, "3GPP System to WLAN Interworking: System Description," TS 23.234, June 2004, <http://www.3gpp.org>.

[4] B. Aboba and D. Simon, "PPP EAP TLS Authentication Protocol," IETF RFC 2716, Oct. 1999, <www.ietf.org/rfc/rfc2716>.

[5] J. Arkko and H. Haverinen, "Extensible Authentication Protocol Method for UMTS Authentication and Key Agreement (EAP-AKA)," IETF Internet Draft, Apr. 2004, <http://www.ietf.org/internet-drafts/draft-arkko-pppext-eap-aka-12.txt>.

[6] L. Blunk and J. Volbrecht, "PPP Extensible Authentication Protocol (EAP)," IETF RFC 2284, Mar. 1998, <www.ietf.org/rfc/rfc2284>.

[7] N. Borisov, I. Goldberg, and D. Wagner, "Security of the WEP Algorithm," <http://www.isaac.cs.berkeley.edu/isaac/wep-faq.html>.

[8] M. Buddhikot, G. Chandranmenon, S. J. Han, Y. W. Lee, S. Miller, and L. Salgarelli, "Integration of 802.11 and Third Generation Wireless Data Networks," Proc. IEEE Infocom 2003 (San Francisco, CA, 2003), pp. 503–512.

[9] European Telecommunications Standards Institute, "Requirements and Architectures for Interworking Between HIPERLAN/2 and 3rd Generation Cellular Systems," TR 101 957, Aug. 2001, <http://www.etsi.org>.

[10] European Telecommunications Standards Institute, "General Packet Radio Service (GPRS) Service Description, (Stage 2), (Release 99)," 3GPP TS 23.060 v3.14.0 (2002-12) 122 060, <http://www.etsi.org>.

[11] P. M. Feder, N. Y. Lee, and S. Martin-Leon, "A Seamless Mobile VPN Data Solution for UMTS and WLAN Users," Aug. 2003, <http://www.3gamericas.org/English/Technology_Center/WhitePapers/>.

[12] F. Fitzek, M. Munari, V. Pastesini, S. Rossi, and L. Badia, "Security and Authentication Concepts for UMTS/WLAN Convergence," Proc. IEEE Vehicular Technology 2003 Conf. (Orlando, FL, 2003), pp. 2343–2347.

[13] P. Funk and S. Blake-Wilson, "EAP Tunneled TLS Authentication Protocol (EAP-TTLS)," IETF Internet Draft, Aug. 2003, <http://www.ietf.org/

proceedings/04mar/I-D/draft-ietf-pppext-eap-ttls-03.txt>.

[14] H. Haverinen and J. Salowey, "Extensible Authentication Protocol Method for GSM Subscriber Identity Modules (EAP-SIM)," IETF Internet Draft, Apr. 2004, <http://www.ietf.org/internet-drafts/draft-haverinen-pppext-eap-sim-13.txt>.

[15] T. Hiller, A. Palekar, and G. Zorn, "A Container Type for the Extensible Authentication Protocol (EAP)," IETF Internet Draft, Mar. 2003, <http://www.watersprings.org/pub/id/draft-hiller-eap-tlv-01.txt>.

[16] Institute of Electrical and Electronics Engineers, "Wireless LAN Medium Access Control (MAC) and Physical Layer (PHY) Specifications," ISO/IEC 8802-11:1999(E), ANSI/IEEE Std 802.11, 1999, <http://standards.ieee.org>.

[17] Institute of Electrical and Electronics Engineers, "IEEE Standards for Local and Metropolitan Area Networks: Port Based Network Access Control," IEEE Standard 802.1x, June 2001, <http://standards.ieee.org>.

[18] Institute of Electrical and Electronics Engineers, "IEEE Standards for Local and Metropolitan Area Networks: Wireless LAN Medium Access Control (MAC) and Physical Layer (PHY) specifications: Amendment i: Medium Access Control Security Enhancements," IEEE Std. 802.11i/D8.0, Feb. 2004, <http://standards.ieee.org>.

[19] Lucent Technologies, "Managed Seamless Roaming Services," Mar. 2004, <http://www.lucent.com/livelink/0900940380070310_Brochure_datasheet.pdf>.

[20] A. Palekar, D. Simon, J. Salowey, H. Zhou, G. Zorn, and S. Josefsson, "Protected EAP Protocol (PEAP) Version 2," IETF Internet Draft, July 2004, <http://www.watersprings.org/pub/id/draft-josefsson-pppext-eap-tls-eap-08.txt>.

[21] L. Salgarelli, M. Buddhikot, J. Garay, S. Miller, U. Blumenthal, S. Patel, P. Dahl, D. Stanley, and C. Carroll, "EAP SKE Authentication and Key Exchange Protocol," IETF Internet Draft, July 2001, http://watersprings.org/pub/id/draft-salgarelli-pppext-eap-ske-01.txt>.

[22] L. Salgarelli, M. Buddhikot, J. Garay, S. Patel, and S. Miller, "Efficient Authentication and Key Distribution in Wireless IP Networks," IEEE Wireless Commun., 10:6 (2003), 52–61.

[23] J. van Bemmel, J. Brok, and H. Teunissen, "Secure, Smart and Seamless Roaming from an End-User Perspective," Feb. 2004, <http://www.lucent.nl/bell-labs>.

[24] T. Wu, "The SRP Authentication and Key Exchange System," IETF RFC 2945, Sept. 2000, <www.ietf.org/rfc/rfc2945>.

(Manuscript approved December 2004)

MIROSLAV ŽIVKOVIĆ is a member of technical staff in the Bell Labs Europe Department at Lucent Technologies in Enschede, The Netherlands. He holds a Dipl. Ing. degree in electronics and telecommunications from the Faculty of Electrical Engineering in Belgrade, Serbia, and Montenegro. His current research interests are in the area of security of wireless systems.

MILIND M. BUDDHIKOT is a member of technical staff in the Center for Networking Research at Bell Labs in Holmdel, New Jersey. He holds a D. Sc. degree in computer science from Washington University in St. Louis, Missouri, and an M.Tech. degree in communication engineering from the Indian Institute of Technology in Mumbai, India. Dr. Buddhikot's current research interests are in the areas of systems and protocols for integrated public wireless networks, authentication and dynamic key exchange protocols, voice-over-IP (VOIP) networks, and sensor and ad-hoc networks. He has authored over 26 research papers and 9 patent submissions in the areas of design of multimedia systems and protocols, layer-4 packet classification, MPLS path routing, authentication and dynamic key exchange, and 802.11/3G integration. He currently serves as the associate editor of the IEEE/ACM Transactions on Networking. *He also served as a co-guest-editor of* IEEE Network *magazine's March 2001 special issue, "Fast IP Packet Forwarding and Classification for Next Generation Internet Services."*

KO LAGERBERG is a member of technical staff in the Bell Labs Europe Department at Lucent Technologies in Enschede, The Netherlands. He is currently working in several research projects on service platforms for mobile and wireless networks. Since joining Bell Labs, he has worked on 802.11 protocol analysis, prototyping of OSA/Parlay, 3G/WLAN integration and Web services technology. Mr. Lagerberg holds a B.Sc. degree in computer science from the Polytechnic College of Enschede, The Netherlands.

JEROEN VAN BEMMEL is a member of technical staff in the Bell Labs Europe Department at Lucent Technologies in Enschede, The Netherlands. He holds an M.Sc. degree in computer science from the University of Twente in Enschede, The Netherlands. His current research activities are in the areas of Web services security, WLAN VoIP, and OMA/OSA/Parlay in relationship to IMS. ◆