Delivering Attribute Certificates over GPRS

Georgios Kambourakis
Department of Information and Communication Sys. Engineering
University of the Aegean
Samos 83200, Greece
Tel: +30 2730 82000
gkamb@aegean.gr

Angelos Rouskas
Department of Information and Communication Sys. Engineering
University of the Aegean
Samos 83200, Greece
Tel: +30 2730 82236
arouskas@aegean.gr

Stefanos Gritzalis
Department of Information and Communication Sys. Engineering
University of the Aegean
Samos 83200, Greece
Tel: +30 2730 82234
sgritz@aegean.gr

ABSTRACT
Attribute Certificates (ACs) have been developed and standardized by the ANSI X9 committee as an alternative and better approach, to X.509 public key certificates, for carrying authorization information. Attribute Authorities (AA) bind the characteristics of an entity (called attributes) to that entity by signing the appropriate AC. Therefore, ACs can be used for controlling access to system resources and employing role-based authorization and access controls policies accordingly. Although ACs are widely used and standardized, to the best of our knowledge, no mobile infrastructure or service currently utilizes them. In this paper, we first examine how basic Public Key Infrastructure (PKI) can be incorporated into mobile networks and especially the Universal Mobile Telecommunications System (UMTS). As a case study, we then experiment with ACs in the GPRS network, using a prototype implementation. In particular, we investigate and measure the performance in terms of service and transfer times when ACs are introduced in the mobile environment. Our measurements show that ACs technology not only is feasible to implement in present and future mobile networks, but at the same time can deliver flexible and relatively fast services to the subscribers, without compromising security.

Categories and Subject Descriptors
C.2.0 & C.2.2 [**Computer – Communication Networks**]: General – *Security and Protection,* Network Architecture and Design - *Wireless communication.*

General Terms
Security, Measurement, Performance, Design.

Keywords
GPRS; UMTS; PKI; Attribute Certificates; Performance Evaluation.

SAC '04, March 14-17, 2004, Nicosia, Cyprus.

1. INTRODUCTION
In numerous cases mobile users want to access specific time-limited services, like buying something from an on-line store, settle down some stock transactions with a bank, or download a file from a protected site. This can be accomplished by using temporary or attribute short-lived certificates [1]. Attribute Certificates (ACs) are particularly well suited to control access to system resources, and to implement role-based authorization and access controls accordingly [2], [3].

ACs are theoretically similar to Privilege Access Certificate (PACs), as used in SESAME and Windows 2000 operating system. The use of ACs has been included into both the ANSI X9.57 standard and the X.509 standards and recommendations of both ITU-T and ISO/IEC. AC based authorization is also an extension to the IETF Transport Layer Security Protocol (TLS). One of the advantages of these temporary certificates having a short life is that they do not usually need to be revoked and will therefore need not be included in any Certificate Revocation List (CRL). If they are issued in respect of a pre-paid subscription service, they certainly not require any revocation at all. Finally, this mechanism can support non-repudiation services.

Another application area for ACs is mobile code technology, used by applications in wired and wireless computer networks in the last few years. Making code mobile means that programs or code segments are exchanged between computer networks and systems and the heterogeneity of platforms is hidden by a common language in which the program code is actually written [4]. A solution to protect the execution environment (e.g. mobile device), against potentially malicious mobile code is to authenticate the mobile code before it is actually executed. This approach is known as "*Shrink-Wrap*". So, although it is not possible for someone to decide if a portion of mobile code contains malicious code, he can at least "authenticate" it. This can be very useful to a software developer who digitally signs the mobile code and distributes it together with the "attribute" certificate that is needed to verify the signature.

For example, let us consider a mobile Palm user who connects via GPRS in a mobile-portal and seeks for games. He wants to be sure that the gaming-code he decided to download is authentic. On the other hand, a hobbyist developer programs an application for specific phones. He wants to sign his code and put it along with the matching certificate in a mobile-portal. Therefore, he needs to obtain an AC. In such an environment, we assume that there are many Attribute Authorities, which can issue that kind of

certificates, certainly in collaboration with the service-offering parties. For example, if an organization already runs a directory service for public key certificates and related status information, this service can also be used to distribute ACs.

Certainly, to implement such scenarios with ACs, we need to examine interworking alternatives between the mobile core network and the presumable public key infrastructure. Using an existing GPRS operator network, we developed a test bed to measure the performance of a simple protocol to obtain ACs. We stretched AA with various levels of AC request load, and the measurement results show that AC issuing is attainable in terms of service time, while simultaneously can deliver flexible and scalable solutions to both future mobiles operators and users.

The rest of the paper is organized as follows. In Section 2, we briefly present two feasible solutions to interwork between UMTS core network and PKI, taking into account that the user may roam between different (serving) and probably heterogeneous network domains. Section 3 gives an overview of our experimental test bed and procedures, while Section 4 presents the derived performance results. The last section concludes the paper and introduces future work.

2. INCORPORATION OF PKI IN MOBILE NETWORKS

To support ACs and public key services in general, some sort of Public Key Infrastructure (PKI) is necessary. Currently 2.5G and 3G systems lack such a large-scale of infrastructure to authorize and consequently charge mobile users for new services, as well as to provide digital signatures and non-repudiation services. However, in the years to come it is very likely that mobile operators will incorporate PKI technology or become associated to Trusted Third-PKI Parties (TTPs).

Successful wireless PKI implementations and solutions from companies like Sonera Smarttrust, Lucent Technologies and Entrust, strengthens the assertion that PKI has become an acknowledged and promising component of standards. Projects like ASPeCT [5] and USECA [6], Third Generation Partnership Project (3GPP) discussion papers especially for UMTS Release 6 [7], as well as other papers [8], foresees that evolution. The eNorge 2005 strategy [9] calls for a shared PKI for Norway, while advanced standards such MexE, WAP and i-mode from NTT DoCoMo have moved forward to introduce public key methods. Finally, WAP specifications [10] mention the use of "role-certificates" to be included in later versions.

Integration between 3G mobile systems and PKI has not been standardized yet. However, recent 3GGP discussion documents [7] deal with that particular subject. Figure 1 depicts two possible alternative architecture approaches that integrate PKI elements with UMTS. Of course, other possible and perhaps more optimal solutions e.g. gateway or proxy oriented that does not affect SGSN or GGSN signaling can be proposed.

In both cases, all CA/AA network elements are assumed to be part of one Network Domain Security (NDS) and message exchanging between CA/AA and SGSN/GGSN is performed over IP. Moreover, both require standardization of new signaling interfaces between AA and SGSN/GGSN accordingly.

In the left side of this figure, the user sends the certificate request always to SGSN, indicating the network (home or serving) he wants the certificate from. SGSN checks which type of certificate is requested (home or serving), adds appropriate parameters (like International Mobile Subscriber Identity (IMSI), and other parameters from the subscriber profile) to the request message,

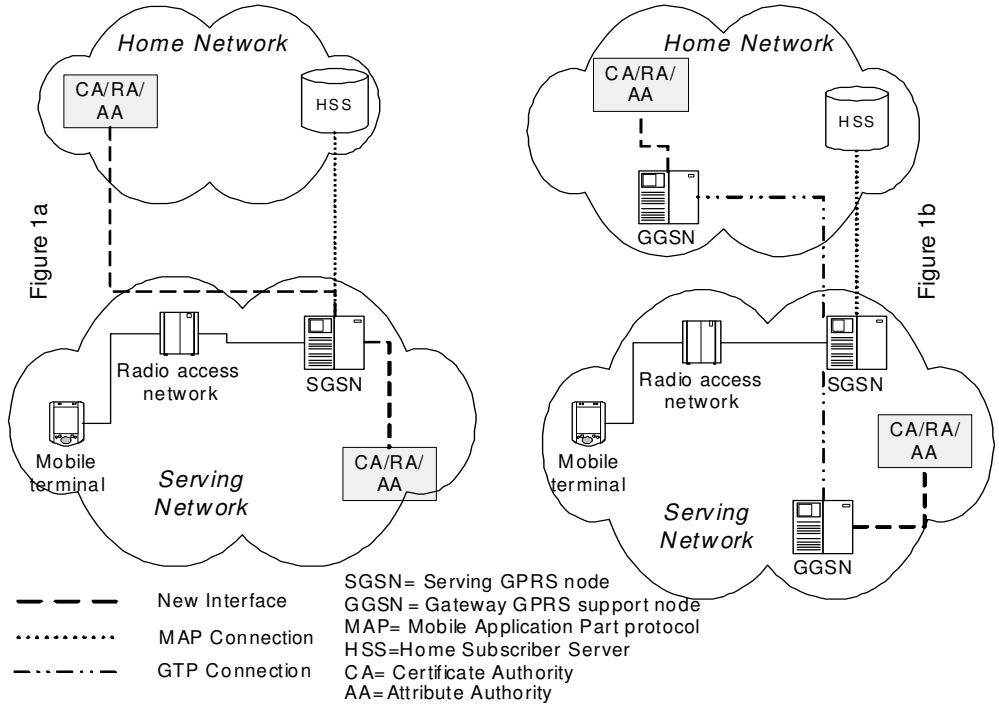

Figure 1. Two alternative architectures for PKI and UMTS integration

and then forwards it to the CA/AA at the home or serving network. CA/AA needs to check, based on the subscriber data, whether certificate issuing is allowed or not. If allowed, CA/AA generates and signs the certificate, updates its database and delivers a pointer or the certificate itself back to SGSN.

In the right side of the same figure, the home or serving GGSN is directly connected to the corresponding CA/AA. The user chooses the network he wants the certificate from, and the SGSN will route the certificate request to the correct CA/AA. If certificate issuing is allowed, CA/AA generates and signs the certificate, updates its database and delivers the certificate back to GGSN.

Both options have their benefits and drawbacks and extensive evaluation of the proposed architectures is beyond the scope of this paper. For the left alternative, addressing of the local CA is easy as SGSN is always located in the serving network, while at the same time there is no need to define new security procedures between SGSN and User Equipment (UE). Moreover, SGSN can easily take over (or deliver the appropriate information to CA/AA) the subscriber information verification, as subscriber profile is downloaded to SGSN. In the following, we will attempt to further evaluate this architecture, in terms of service times, using the GPRS network. On the other hand, a new inter-operator interface between the serving SGSN and the home network has to be defined. Additionally, the address of the Home CA/AA must either be stored to the UE or added to the subscriber profile.

The right option has the main benefit that GGSN is the "natural" element to access network entities that are external to Packet Switched (PS) domain. In contrast, standardization of new messages between UE - GGSN and SGSN - GGSN is required.

3. FEASIBILITY CASE STUDY IN GPRS

To test the feasibility of the aforementioned architecture (Figure 1a) in future mobile networks we used as a case study the delivery of ACs over a GPRS network.

3.1 Test Bed Setup

We constructed an experimental network architecture, which is illustrated in Figure 2. The presumed mobile device is an IBM ThinkPad 380 laptop computer that uses Windows 95B operating system. Contemporary wireless devices are featuring advanced architectures with Strong-Arm processors up to 400 MHz, memory capacities of 64MB RAM and 48 MB ROM, support for various applications and strong operating systems. The "client" uses a Siemens ME45 mobile phone, in order to connect to the Internet over GPRS. The GPRS coding scheme was CS1 (9.05 kb/s) and the time slots for GPRS were varying from 3 to 4, thus having wireless network speeds in the range from 27 to 36 kb/s. Network speeds for 3G will be 144 kb/s up to 348 kb/s for wide and up to 2 Mb/s for low coverage and mobility, which will substantially reduce transfer times.

The IBM 380 incorporates a 150 MHz Pentium CPU and has 16 MB of RAM available. At the other end, the Attribute Authority machine has a Pentium III 733 MHz processor with 256 MB RAM, running the Windows 2000 professional SP2 operating system. CA/AA server process S1 is multi-threaded. It opens a TCP-SSL listening socket and waits for transactions. When it receives a message, it dispatches a thread to process and respond to the request. AA has also a WAN connection available. The multi-threaded process S2 that loads the AA with virtual requests for AC certificate issuing, is running on another laptop machine that incorporates a Celeron 1.2GHz processor with 256 MB RAM and is wired to the local network with a speed up to 10 Mbps. The inter-arrival times between successive AC issuing requests, generated by process S2, follow the negative exponential distribution. Comparable test-beds for GPRS and WAP performance evaluation can be found in the literature [11],[12],[13]

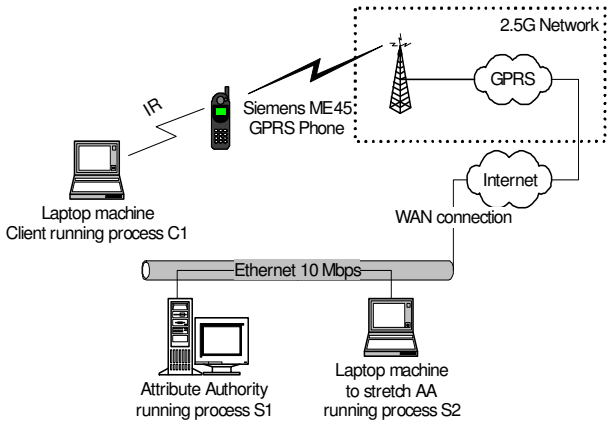

Figure 2. Experimental hardware architecture for AC issuing

We wrote the client and CA/AA server console applications in Java 2 and employed the well-known open-source Apache-style license OpenSSL toolkit in version 0.9.6g [14], [15] to make them public key enabled. In the following, we describe in detail the procedure of obtaining a temporary certificate from the AA (Figure 3).

3.2 Procedure for Attribute Certificate Retrieval

Depending on the nature of the request, the user constructs a request-certificate locally filling up values in the following fields (here we supply some demo values):

COUNTRY_NAME = "US"
STATE_OR_PROVINCE_NAME = "VA"
LOCALITY_NAME = "FAIRFAX"
ORGANIZATION_NAME = "ZORG.ORG"
ORGANIZATIONAL_UNIT_NAME = "SERVER DIVISION"
COMMON_NAME = "TMSI/IMSI"
SUBJECT_ALT_NAME = "DNS:195.251.161.167"
TYPE_OF_REQUEST = "1.1.2.2"

The sixth field appoints the Packet-Temporary Mobile Subscriber Identity (P-TMSI), stored in the nonvolatile memory of the MS, while the last field designates the type of the attribute certificate that the user wants to be issued (last three numbers) and by which network's (home or serving) CA/AA (first number).

When the request-certificate creation phase is completed, the application incorporates the user's public key, hashes the entire block and signs the derived hash with the user's RSA 1024 bits

private key[1]. Note that the user's private key is stored in his UMTS-SIM (USIM) card and can be generated and associated with him during registration time. The requested block (Request_Cert. ‖ User's_Public_Key ‖ Digital_Signature) is then transmitted in clear-text, as it is actually useless to anyone who intercepts it. The total client's request size is about 733 bytes. According to Figure 1a, SGSN checks the first number of the last field, adds the appropriate parameters from user's profile and routes the request to the corresponding CA/AA.

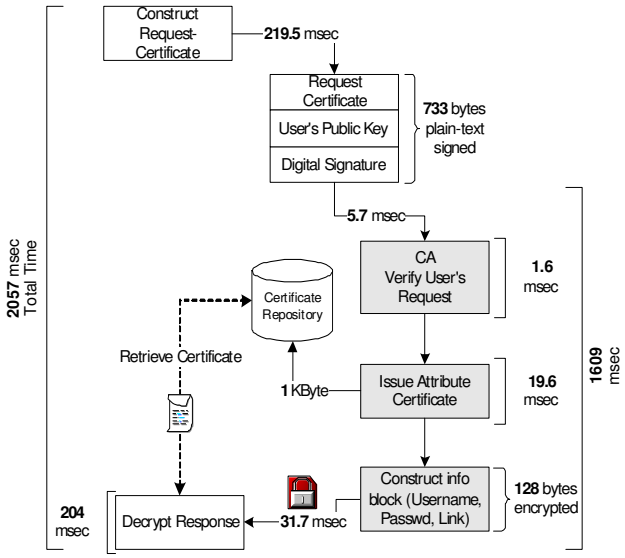

Figure 3. Issuing Attribute Certificates Procedure and Service times

As soon as the CA/AA receives the request, it validates it, by re-calculating its hash, thus certifying that it has not been altered in any way. AA's next step is to issue and sign with its private key a temporary / attribute certificate based on user's certificate values. The CA/AA stores the certificate[2], which is about 1 Kbyte, locally or in a corresponding certificate repository (LDAP directory or other repository). To end up, AA creates a "certificate pointer message" encoded with user's public key, which contains the necessary information for the user to retrieve the certificate. The message has a size of 128 bytes (always the size of the ciphering key) and it contains the following fields:

USERNAME = "HELLO"
PASSWORD = "1111"
LINK = WWW.HELLO.ORG

When the user receives the encrypted message, he decrypts it, and directly or at a later time, retrieves via HTTP or HTTPS the corresponding certificate, from the link provided, using the analogous username and password. Another option is to pass to the application server the corresponding link, instead of the actual certificate. Note that the certificate retrieval and forwarding are safer to be done by the user. If the client delivers a certificate URL, rather than the certificate itself to an application server, he implicitly requests from the server to do the work (retrieve the certificate). The danger is obvious: a denial of service attack is possible when a client deliberately passes an invalid certificate URL.

Closing, issued certificates records held by CA/AA, can effectively provide, if needed, no-repudiation services. For example, assume that an individual has an account at an on-line brokerage. The individual buys shares of some stock using an application on his mobile phone, and then deliberately denies the transaction. It is difficult for the on-line broker to prove that the individual did indeed buy those shares. However, the on-line broker can easily verify the transaction if the application uses an AC when the trade is requested.

4. MEASUREMENT RESULTS

We experimented with various values for the arrival rate of AC requests, which determines the virtual load offered to the AA. We varied this parameter from 20 to 100 requests per minute and the affect on the server performance was negligible. Measurements were gathered from a set of 1000 transactions between the server and the client (MS). Our experiments were conducted in different days and hours during a week period and 50% of the measurements were gathered during peak hours. The average values of the time durations measured are presented in Table 1 and the probability density function of client's total time is shown in Figure 4.

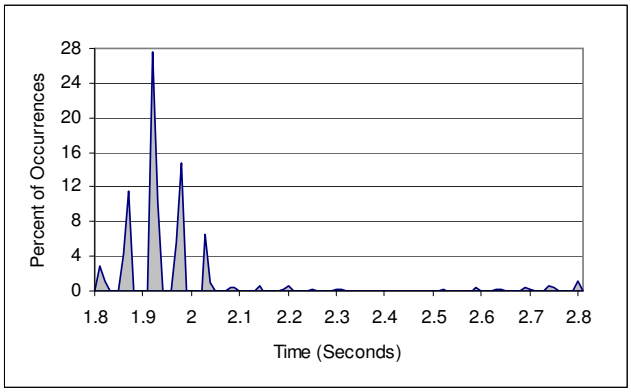

Figure 4. Client service total time

As we see, the average client's total time for one transaction is about 2.05 sec, with a standard deviation of 0.35, which is generally acceptable by a user who demands "a fast *and* secure service". Last but not least, we note that the extra network delay, derived from the fact that our CA/AA server did not reside inside the provider's core network, was measured with a ping tool and found to be about 100 milliseconds.

[1] $Hash_{(16\_bytes)}$ = MD5(Request_Cert. ‖ User's_Public_Key) and Digital_Signature = $(Hash_{16\_bytes})_{User's\_Private\_Key}$.

[2] Another scenario is to deliver the published certificate directly back to the user. Either ways does not affect the generality of the approach, as certificate pointer is 128 bytes, while certificate is about 1Kbyte. For a description of possible attribute certificate fields refer to [3], [16].

Table 1. Average service times in milliseconds

Client					CA/AA (Server)			
Time to create request	Time to send request	Time to send request and receive response	Time to decrypt response	Total Time	Time to verify client's request	Time to create Attr. Cert	Time to send certificate back to client	Total time
219.5	5.7	1609	204	2057 Max: 4.18 Min: 1.81	1.6	19.6	31.7	149.3

5. CONCLUSIONS

As users rush to adopt IP technology and want mobile access to IP networks, they also become aware of the need for security features and protection of their privacy. The constantly increasing population of users expects from mobile operators to provide features that will protect their data while in transit, safeguard their billing and customer information, provide reliable Authentication, Authorization and Accounting (AAA) mechanisms and offer availability and quality comparable to that of the wired services. Thus, more flexible, dynamic and scalable mechanisms are necessary in order to support on-demand services and all-IP end-to-end solutions in a many-to-many trust model integrated with the Internet environment.

In this paper we argued that the necessary, to support ACs, public key infrastructure is about to be incorporated to mobile networks in the near future. Moreover two feasible network architectures were discussed. We experimented with on-the-fly attribute certificate generation, testing the performance of a prototype implementation based on one of the discussed architectures. Results showed that ACs issuing is attainable in terms of service time, while simultaneously can deliver flexible and scalable solutions to both future mobiles operators and users.

6. REFERENCES

[1] 3GPP TSG, "Support of certificates in 3GPP security Architecture", Discussion Document S3-010353 SA WG3 Security – S3#19, July 2001.

[2] Oppliger, R., Internet and Intranet Security, 2nd Edition, Artech House, 2002.

[3] Oppliger, R., Pernul, G. & Strauss, C., "Using Attribute Certificates to Implement Role Based Authorization and Access Control Models", In the Proc. of 4. Fachtagung Sicherheit in Informationsystemen (SIS 2000), pp. 169 – 184, Oct. 2000.

[4] Oppliger, R., Security Technologies for the World Wide Web, Artech House, 2000.

[5] ASPeCT Project, Securing the future of mobile communications;

[6] USECA Project, UMTS Security Architecture: Intermediate report on a PKI architecture for UMTS, Public Report, July 1999.

[7] 3GPP TSG, "Architecture proposal to support subscriber certificates", Discussion and Approval document, Tdoc S2-022854, Oct. 2002.

[8] Kambourakis G., Rouskas A., Gritzalis S., "Introducing PKI to enhance Security in Future Mobile Networks", in the Proc. of the IFIPSEC'2003 18th IFIP Int'l Information Security Conf., pp.109-120, Athens, Greece May 2003.

[9] "eNorge 2005", Naerings –og handelsdepartmentet, 2002.

[10] Wireless Application Protocol, "WAP Certificate and CRL Profiles Specification", WAP-211-WAPCert, May 2001.

[11] Chakravorty, R. & Pratt, I., "Performance Issues with General Packet Radio Service", paper under submission to Journal of Communication and Networks (JCN), 2002.

[12] Chakravorty, R., Cartwright, J. & Pratt, I., "Paractical Experience with TCP over GPRS", in the Proc. of IEEE GLOBECOM 2002, Taipei, Noe 2002.

[13] Korhonen, J., Aalto, O., Gurtov, A., & Laamanen, H., "Measured Performance of GSM HSCSD and GPRS", in the Proc. of the IEEE Int'l Conf. On Communications (ICC'01), Helsinki Finland, June 2001.

[14] The OpenSSL project Web page, http://www.openssl.org.

[15] Viega, J., Messier, M. & Chandra, P., Network Security with OpenSSL, O'Reilly and Associates, 2002.

[16] Housley, R. & Polk, T., "Planning for PKI", Wiley, 2001.