ISSN 2067 – 4074

Future Developments in Non-Repudiation in GSM WAP Applications

Cristian TOMA

Faculty of Cybernetics, Statistics and Economic Informatics
Department of IT&C Technologies
Academy of Economic Studies Bucharest, Romania
cristian.toma@ie.ase.ro

Abstract: The paper presents issues and architectures for mobile applications and GSM infrastructure. The paper shows the redesign of the solution for avoiding denial of service from WAP applications using WIM features. The first section contains the structure of GSM network from voice and data point of view. The security in GSM network is presented in second chapter. The third chapter presents a solution for realizing mobile subscriber non-repudiation. The solution is based on the HTTP protocol over WAP.

Key-Words: m-application security, GSM – Global System for Mobile Communication, WAP – Wireless Application Protocol, WIM – Wireless Identification Module, SAWNR - Secure Application for Wireless Non Repudiation.

1. GSM Overview

The GSM – Global System for Mobile Communication includes many technologies for voice and data transmission. For GSM there are few distinctive types of applications:

- Pull typical applications – Web/WAP applications over HTTP (the web browser of the mobile device is requesting a resource and the service provider responses with the resource using usually HTTP); For more information please see [9], [5], [6];
- Push typical applications (the subscribers have signed up for a service and the information is going to the mobile device using SMS – Short Message Service, MMS – Multimedia Message Service, Push SI – Service Indication, SL – Service Location and CO – Cache Operation); For more information please see [6];
- SIM Toolkit– Subscriber Identity Module Toolkit applications that are running in the SIM Smart Card using native code or Java Card technology; For more information please see [3], [4], [5], [6];
- Native applications which are running on the top of operating system of the mobile device. Usually these applications are developed in C/C++ for mobile OS such as: Symbian OS, Linux ALP and Microsoft Mobile OS;
- Applications written in Java Micro-Edition or in C# for .NET Compact Framework that are running in proper virtual machines on the mobile device. For more information please see [5], [6];
- Hybrid Applications that provide complex services such as SIM Sentry for Multimedia Mobile Content Digital Rights Management, Midlets with Java Smart Card technology solution for mobile banking or electronic purses, Web WAP applications for mobile streaming over RTP and RTSP in GSM networks. For more information please see [3], [4], [5], [6], [7], [8].

In this paper we focus on Web WAP – Wireless Application Protocol applications security and in particular we focus on the solution to avoid repudiation.

The basic GSM network architecture is presented in figure 1. The main components involved in voice and data transmission are the following:

ISSN 2067 – 4074

- BSS – Base Station Subsystem. It controls the quality of the links from GSM radio interface and contains BTS and BCS.
- BTS – Base Transceiver Station. Controls the "antennas" and maintains the communication through a duplex radio channel. It supports configurations for: electro-magnetic power, radio channel used for broadcasting, BSIC – Base Station Identity Code. Its main functionalities are: message encryption, channel coding and modularization.
- BCS – Base Station Controller. Administrates and controls base stations and radio channels. It provides the coding implementation for voices messages and manages the data localization.
- NSS – Network Sub System. It contains MSCs, Databases such as VLR, HLR, EIR and AuC and adaptation modules such as XC, IWF, EC. NSS provides the following functionalities: management of communication link with other mobiles, land and satellite networks, management of mobile subscribers from other BSCs, and the management of the subscribers using data from AuC, EIR, VLR and HLR databases
- MSC – Mobile Switching Center. Contains switching subsystems (e.g. for PBX signaling and for communication signaling over SS7 with other MSCs) and control subsystems
- HLR – Home Location Register. Stores the subscribers' parameters including the MSISDN and the service type (e.g. 10 SMS, 80 minutes for 1 month).
- VLR – Visitors Location Register. It is a mirroring database of HLR for temporary subscribers of another VLR area.
- EIR – Equipment Identity Register. It is the centralized database with IMEI (unique number for each provider-device) for each mobile device.
- AuC – Authentication Center. The following functionalities and responsibilities are included: authorization process for the subscriber access into mobile network for encrypting transmission on radio path and for assign of the temporary identity - TMSI (Temporary Mobile Subscriber).

ISSN 2067 – 4074

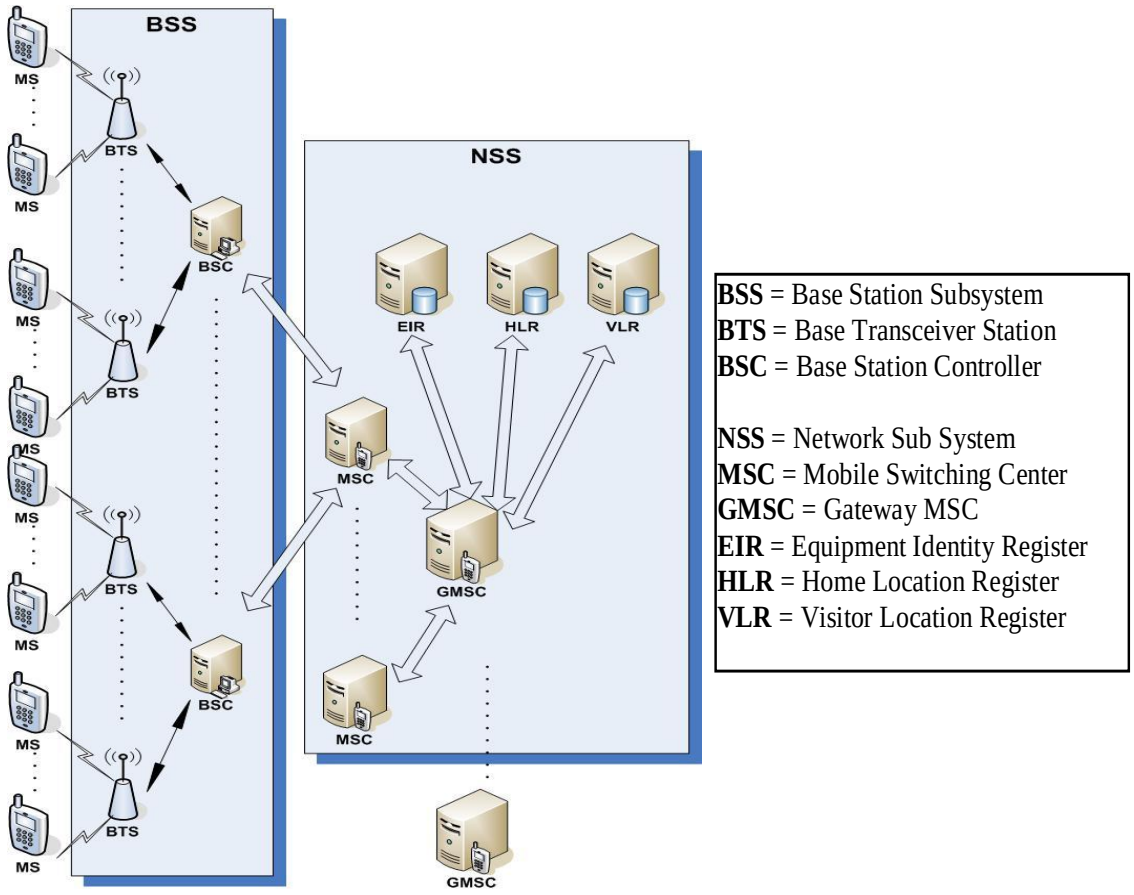
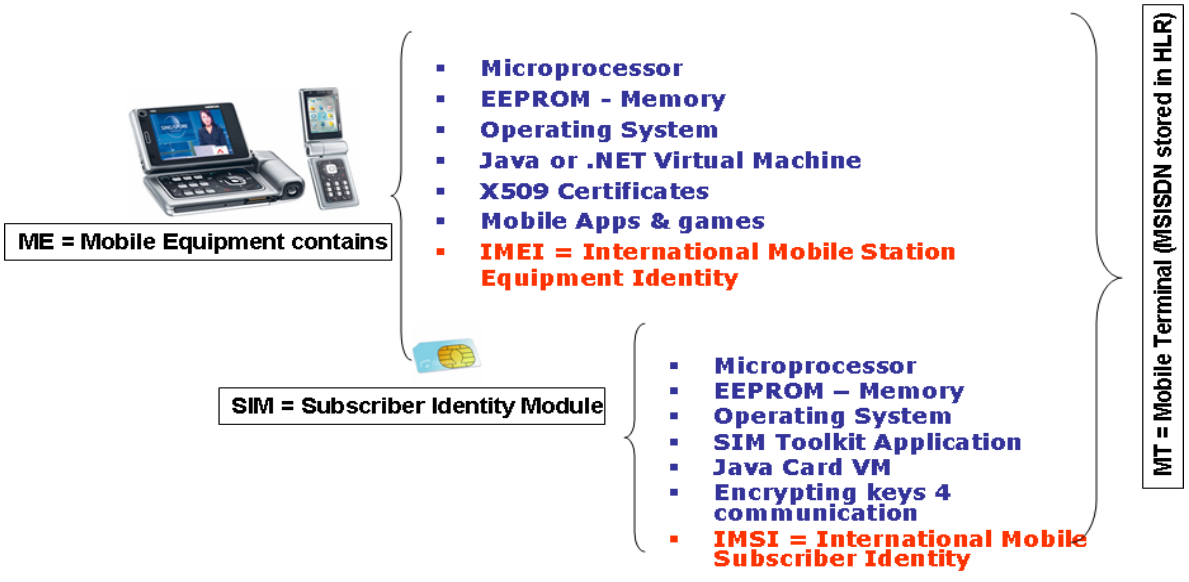


Fig. 1. GSM Network Architecture

Figure 2 presents the main concept used in GSM for end-user device: the mobile is a 2 in 1 computer. The first computer is represented by the SIM – Subscriber Identity Module. Actually, the SIM is a smart card with a microcontroller, three types of memory area (ROM, EEPROM and RAM) and I/O ports for outside communication (usually in half duplex mode). The mobile device itself is the second computer. It also includes a microprocessor, different types of memory areas and an operating system.



ISSN 2067 – 4074

Fig. 2. GSM Mobile Equipment Structure

The GSM evolution is based on this infrastructure and the security of the entire system depends on many factors.

2. GSM Voice and Data Security

Figure 3 presents the processes used in authentication of the voice and data transmission initiated by MT – Mobile terminal.

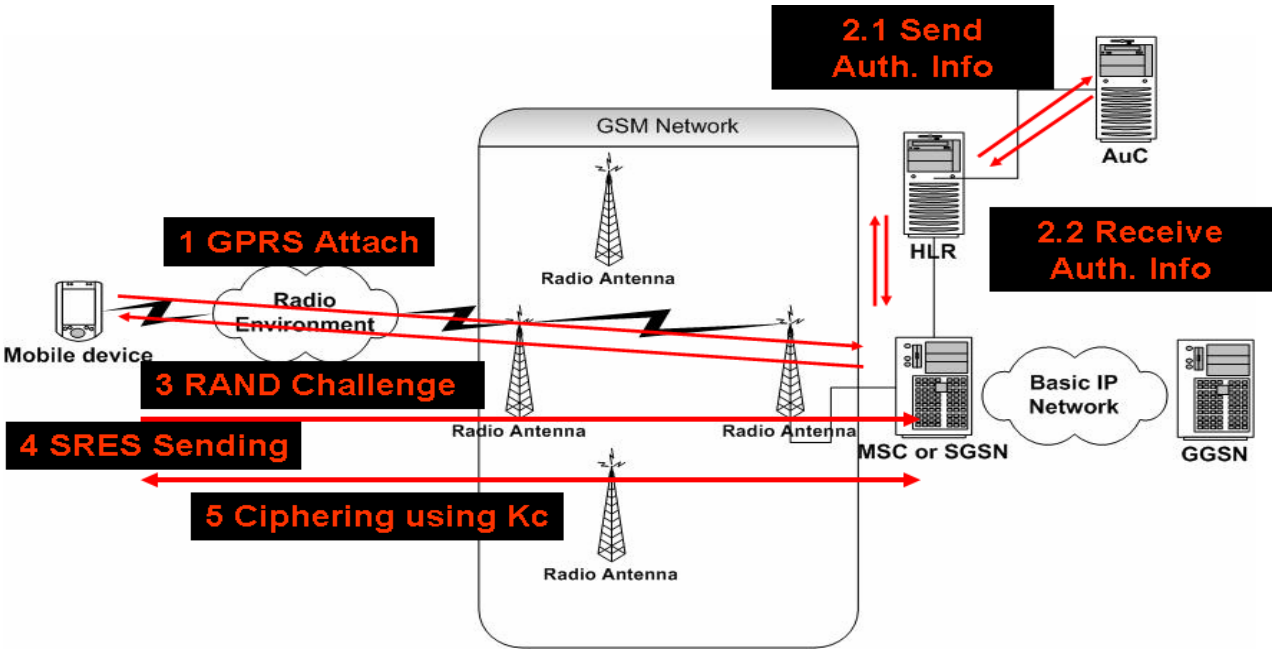


Figure 3. Processes for authentication and confidentiality of mobile voice and data transmission

Fig. 4 and 5 present a more detailed technical view of the systems' security process. Fig. 4 shows the main network items involved in voice and data security, and fig. 5 presents the logical flow of the data.

ISSN 2067 – 4074

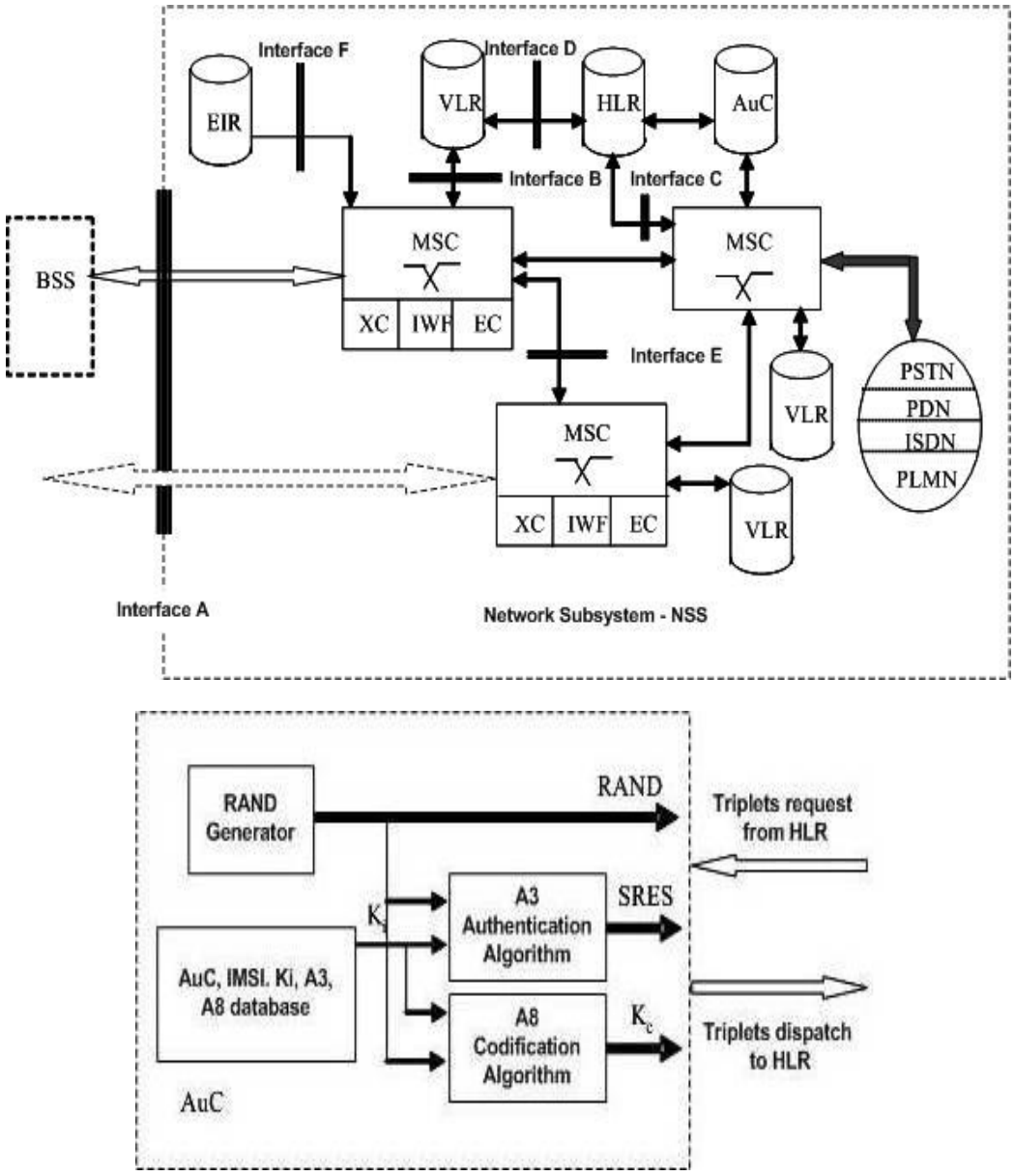


Fig. 4. The details of the security mechanisms

The AuC – Authentication Center plays the main part in voice and data authentication and integrity (figures 3, 4, and 5). Its main functionalities and responsibilities consist in: authorizing the subscribers' access into the mobile network; encrypting transmission on radio path and assign of the temporary identity - TMSI (Temporary Mobile Subscriber Identity). The authentication process of the mobile subscriber is described as it follows:

- The mobile station sends the SIM's IMEI to the HLR through the "Net Attach" (or GPRS Attach).
- A triplets request is sent to the AuC from HLR (all HLR, AuC and SIM suppose to store same Ki) through the "Send Auth. Info" message (contains SIM's IMSI)
- The AuC generates a response that contains:
  - RAND (random number – challenge)
  - Kc – encryption key that is a result of the A8 algorithm. The A8 algorithm uses the stored identity key – Ki from AuC corresponding to the received IMEI.

ISSN 2067 – 4074

  - SRES – Signed RESponse generated through A3 Authentication Algorithm with RAND and Ki as input
- The HLR receives the triplets and sends to the mobile only RAND
- The mobile device must be enabled using Ki from SIM, A3 and A8 algorithms in order to reconstruct Kc and SRES. It then sends the SRES to the HLR via MSC or SGSN (in GPRS only).
- If the SRES received by the HLR from the mobile device is the same with the one that is received from the AuC, then the authentication is done and Kc will be used for ENCRYPTION.

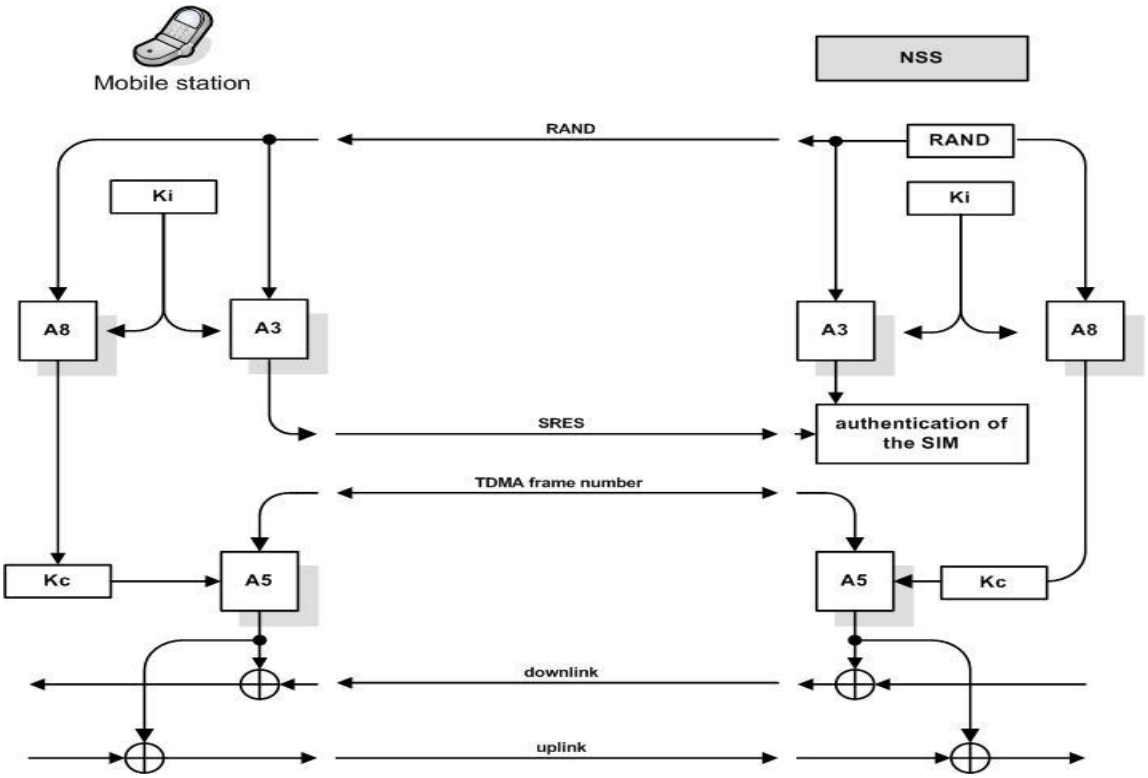


Fig. 5. Overview of data workflow for authentication and confidentiality [3]

The described architecture had been considered at the beginnings of the GSM and has been used since (including in GPRS, EDGE, UMTS networks). The figure with some modifications is copyright of [3].
So in terms of authentication and encryption the path from mobile device to the WAP Gateway is secured. In figure 6 is a typical GSM GPRS architecture. Basically the voice is going via MSC over SS7 protocol and the data traffic goes over SGSN and GGSN. Some certain features of the SGSN and GGSN equipments must map and relate the TCP/IP protocols stack over radio from the mobile phone with the TCP/IP protocols stack over Ethernet from the service provider. In practical approaches the hardware and software components which map and relate TCP/IP protocols over the radio and Ethernet are generically called WAP Gateway. Figure 6 is very simple to understand if it is analyzed together with the previous chapters and figures from this paper. If the link between the WAP Gateway and service provider is secured using SSL/TLS or IPSec and there are secure policies implemented at router level, then the problem of encryption in minimal terms of security requirements is fulfilled.

ISSN 2067 – 4074

3. Future developments of SAWNR

This solution – SAWNR – **Solution for Non-Repudiation in GSM WAP Applications** has been published within WSEAS Conference's proceedings and extended version into WSEAS Journal from Cambridge/UK 2008 [8].

The problem is about the mobile subscriber non-repudiation. How a mobile subscriber can be indubitable linked with a web HTTP request? How a mobile subscriber can not sustain that the mobile device disappeared and someone used its device for requesting services? For the answer at these kinds of questions, we provide a solution – SAWNR – Secure Application for Wireless Non Repudiation – which uses J5EE Servlets in order to send WML – Wireless Markup Language and WMLS – WML Script code [9], [10] to the mobile device. The WML and WMLS code helps the mobile Internet browser to digitally sign the data with the private key from the WIM – Wireless Identification Module.

In figure 6 is shown the architecture used in SAWNR solution.

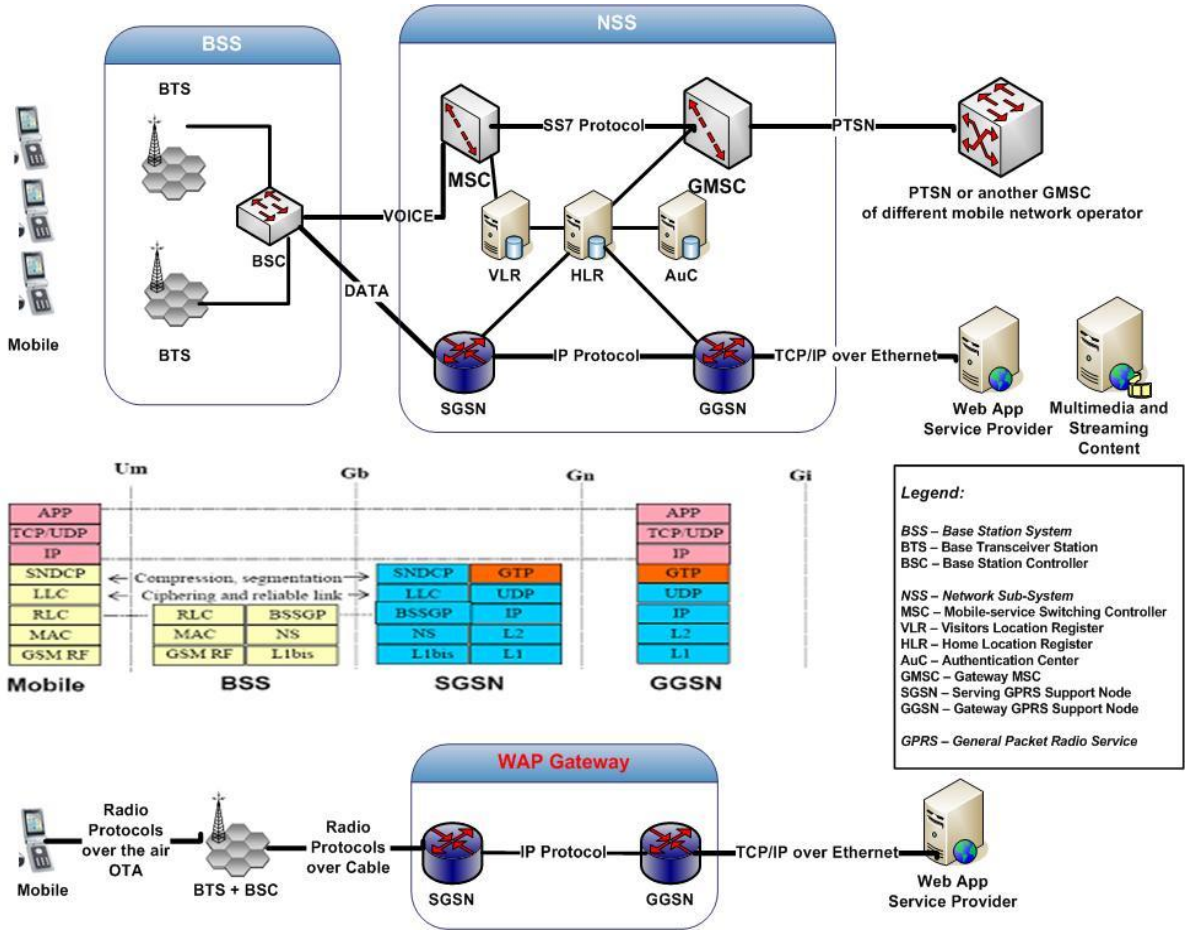


Fig. 6. WAP Gateway features in a GPRS network

In this section it is presented SAWNR – Secure Application for Wireless Non Repudiation in terms of network traffic analysis and in terms of business flow.

In figure 7 from the mobile browser is generate a HTTP GET Request to the Java servlet 'SignDeck' from URL Context 'NokiaServletSecurity' (The complete URL in our tests

ISSN 2067 – 4074

were: http://10.2.4.244:8090/NokiaServletSecurity/SignDeck). The web server (Apache Tomcat) is running on machine with IP address 10.2.4.244 and listen port 8090.

The Java web server servlet generates the HTTP Response from figure 7, starting with header 'HTTP/1.1 200 OK'. The response is a WML code which contains two pairs of <card>…</card> tags (two decks).

The second phone screen from figure 7 is obtained using 'Confirm' option from first deck. Because the mobile web browser display only one pair of <card>…</card> tag (a deck) in a certain period of time, the <go href='#Sign' /> tag instructs the mobile web browser to go in the deck <card id='Sign'>…</card>.

HTTP Network Analysis **Mobile Web Page**

Fig. 7 HTTP Network Analysis for requesting 2 books using a mobile Web WAP application

When the mobile subscriber click 'Sign' option in mobile screen from figure 8, then the mobile web browser request the 'CryptoScript' resource and run the 'sign()' function received in figure 8 HTTP traffic analysis section (because of tag: <go href='CryptoScript#sign()' />).

HTTP Network Analysis **Mobile Web Page**

ISSN 2067 – 4074

```
GET /NokiaServletSecurity/CryptoScript HTTP/1.1
Host: 10.2.4.244:8090
user-agent: Nokia Mobile Browser 4.0
Accept: application/x-wap-prov.browser-bookmarks, application/x-wap-prov.browser-
settings, application/vnd.wap.signed-certificate, application/vnd.wap.hashed-
certificate, application/vnd.nokia.ringing-tone, application/vnd.wap.cert-
response, application/vnd.wap.mms-message, application/vnd.wap.wmlscriptc,
application/vnd.wap.xhtml+xml, application/vnd.syncml+wbxml, application/x-
nokiagamedata, application/vnd.wap.wbxml, application/vnd.wap.wmlc, application/
xhtml+xml, image/vnd.wap.wbmp, text/x-vcalendar, text/x-co-desc, text/x-vcard,
image/gif, text/html, text/css, application/*, multipart/*, text/
vnd.wap.wmlscript, text/vnd.wap.wml, text/plain
accept-charset: ISO-8859-1, US-ASCII, UTF-8; Q=0.8, ISO-10646-UCS-2; Q=0.6
accept-language: en, fi
Via: Nokia Activ Server 2.0 Professional (build 2451A)
X-Network-Info: UDP,127.0.0.1,security=0
Accept-Encoding:
Connection: Close

HTTP/1.1 200 OK
Server: Apache-Coyote/1.1
Content-Type: text/vnd.wap.wmlscript;charset=ISO-8859-1
Content-Length: 206
Date: Thu, 15 Nov 2007 10:31:20 GMT
Connection: close

extern function sign() {
var signed;
signed = Crypto.signText('Java Card Programming Handbook: $99.50, Smart Card
Handbook: $199.50',5,0,'');
WMLBrowser.setVar('signed',signed);
WMLBrowser.refresh();}
```

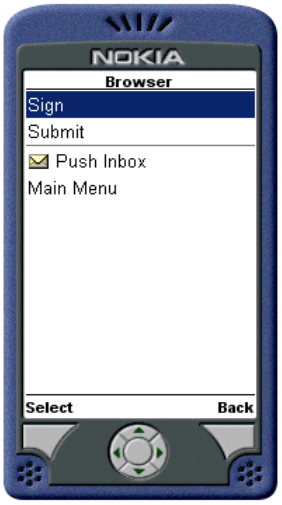


Fig. 8 HTTP Network Analysis for obtain the function 'sign()' from Java Servlet 'CryptoScript'

The figure 8 presents the HTTP Request and Response for the resource: (complete URL = http://10.2.4.244:8090/NokiaServletSecurity/CryptoScript). The mobile browser receives the WML Script code which contains only the 'sign()' function. The mobile browser runs 'sign()' function and it is obtained the behavior from figure 9.

ISSN 2067 – 4074

Fig. 9 Behavior for running the 'sign()' function by the mobile web browser

The mobile subscriber is instructed to choose a proper digital X509 Certificate which is stored in WIM. After that the end-user inserts the password '12345' and the WML Script variable 'signed' contains now the text 'Java Card Programming Handbook: $99.50, Smart Card Handbook: $199.50' which is now digitally signed.

In the last mobile screen from the figure 9 the browser returns in the last deck (<card> tag pair). The last deck was obtained through HTTP GET request from figure 8. In figure 10 the mobile subscriber will choose option 'Submit', after the signing process in order to avoid repudiation. The mobile web browser do a HTTP POST Request accordingly with the WML code (<do type='options' name='a' label='Submit'><go href='SignDeck' method='post'><postfield name='signed' value='$(signed)'/></go></do>).

HTTP Network Analysis

```
POST /NokiaServletSecurity/SignDeck HTTP/1.1
Host: 10.2.4.244:8090
Content-Type: application/x-www-form-urlencoded
user-agent: Nokia Mobile Browser 4.0
Accept: application/x-wap-prov.browser-bookmarks, application/x-wap-prov.browser-
settings, application/vnd.wap.signed-certificate, application/vnd.wap.hashed-
certificate, application/vnd.nokia.ringing-tone, application/vnd.wap.cert-response,
application/vnd.wap.mms-message, application/vnd.wap.wmlscriptc, application/
vnd.wap.xhtml+xml, application/vnd.syncml+wbxml, application/x-nokiagamedata,
application/vnd.wap.wbxml, application/vnd.wap.wmlc, application/xhtml+xml, image/
vnd.wap.wbmp, text/x-vcalendar, text/x-co-desc, text/x-vcard, image/gif, text/html,
text/css, application/*, multipart/*, text/vnd.wap.wmlscript, text/vnd.wap.wml, text/
plain
accept-charset: ISO-8859-1, US-ASCII, UTF-8; Q=0.8, ISO-10646-UCS-2; Q=0.6
accept-language: en, fi
Via: Nokia Activ Server 2.0 Professional (build 2451A)
X-Network-Info: UDP,127.0.0.1,security=0
Accept-Encoding:
Date: Thu, 15 Nov 2007 12:38:25 GMT
Content-Length: 1685
Connection: Close

signed=AQEAgMR568nU%2Bwf%2BsKL0LxLrkBVYYiNE%2FJBXlzfF5G%2FEUEflbD%
2FOAhqHIFNXTstKBgEec0%2BoU8yF7%0Dnh8Pr0%2B1mqaNwQMLFaE%2FFU%
2FGPR2GuxE93PywTx3LrTFwVR62rcRaFaTTVaC83MKmZxENmgKUSA4h%0DYJu1HQH%2BzB9qJsV%2BAhza%
2FvxA7cDA7QwggOwMIIDGaADAgECAgEAMA0GCSqGSIb3DQEBBQUAMIGOM%
0DQswCQYDVQQGEwJVUzELMAkGA1UECBMCTUExDzANBgNVBAcTBkJvc3RvbjEOMAwGA1UEChMFTm9ra%
0DWExDDAKBgNVBAsTA05NUDEkMCIGA1UEAxMbTW9iaWxlIEludGVybmV0IFRvb2xraXQgMy4xMR0wG%
```

Mobile Web Page

ISSN 2067 – 4074

```
0Dp4MaqmekLVhxiGkoxAB%2BtgZKGwHd1cFsQGPRthLlA9ievH%2FaYN3hgbq6hdhbKG8P%2BZupFnJwPj23%
0D6sBAGoBAERKYXZhIENhcmQgUHJvZ3JhbW1pbmcgSGFuZGJvb2s6ICQ5OS41MCwgU21hcnQgQ2FyZ%
0DCBIYw5kYm9vazogJDE5OS41MA0BB9cACwAPAAwAJgAUHTTP/1.1 200 OK
Server: Apache-Coyote/1.1
Content-Type: text/vnd.wap.wml;charset=ISO-8859-1
Content-Length: 220
Date: Thu, 15 Nov 2007 10:31:30 GMT
Connection: close

<?xml version="1.0"?>
<!DOCTYPE wml PUBLIC "-//WAPFORUM//DTD WML 1.1//EN" "http://www.wapforum.org/DTD/
wml_1.1.xml">
<wml><card id='sign' title='Sign-Result'>
<p>
Thank you for the order<br/>
</p>
</card>
</wml>
```

Figure 10 HTTP POST Request for sending the digitally signed text in order to avoid repudiation

The HTTP POST Request contains the variable signed = **AQEAg…AJgAU** , which represents the text signed with the public key RSA – Rivest, Shamir, Adleman cryptographic algorithm. The RSA algorithm uses the private key stored in WIM – Wireless Identification Module and it access with the PIN – Personal Identification Number.

The 'SignDeck' servlet through HTTP POST Request (generated by the mobile browser) receives the 'signed' parameter and verifies its validity searching in a database. The validity of the signature is processed using the public key from X509 Certificate of the mobile subscriber. If everything is ok, the Java servlet generates HTTP Response which contains a WML code (mobile web page) that informs the end-user that everything was successfully processed.

The Java servlets were developed in optimal mode in order to generate as few as possible HTTP Request (two HTTP GET Request and a HTTP POST Request). There are only two Java servlets: 'SignDeck' which have different implementations for HTTP GET and POST Requests, and the 'CryptoScript' servlet which generates the function 'sign()' in WML Script code.

4. Conclusions

Based on the analysis that were made in lab using an Nokia N95 device and a NetFront mobile browser it has been highlighted that the difference between processing time and access time are not significant. On average the process of digital signing takes at most three seconds. Future research will be conducted on beta testing the smart electronic signature solution that has been embedded in a SIM Java cardlet.

This approach is developed to be accessed by a Java Micro-Edition Midlet, defined by JSR177. Another research direction is to develop a web browser which permits WMLS or similar language for other murkup languages such as: XHTML, cHTML, HTML. The solution presented here has many utilities in the public online services which are extended on mobile devices because it allows developing secure environments with fewer resources because everything is taking place at software level and it doesn't interfere with the existing infrastructure.

ISSN 2067 – 4074

References

[1] William Stallings, Cryptography and Network Security, 3/E, Prentice Hall, 2003.
[2] Bruce Schneier, Applied Cryptography 2nd Edition: protocols, algorithms, and source code in C, John Wiley & Sons, Inc. Publishing House, New York 1996.
[3] Wolfgang Rankl & Effing, "Smart Card Handbook 3rd Edition", John Wiley & Sons Publishing House, USA 2004
[4] Zhiqun Chen, "Java Card Technology for Smart Cards – Architectures and Programmer's Guide", Addison Wesley, 2004
[5] Cristian TOMA, "Tutorial on Java Smart Card electronic Wallet Application", Informatics Security Handbook, AES Publishing House, Romania 2006
[6] Cristian TOMA, "Secure Patterns and Smart-card Technologies used in e-Commerce, e-Payment and e-Government", Informatics Security Handbook, AES Publishing House, Romania 2006
[7] Ion IVAN, Cristian TOMA, Catalin BOJA, Marius POPA, "Secure Architecture for the Digital Rights Management of the M-Content", ISP'06 of the WSEAS Conference in Venice, Nov. 2006.
[8] Ion IVAN, Cristian TOMA, Catalin BOJA, Marius POPA, "Secure Platform for Digital Rights Management Distribution", WSEAS Transaction on Computers, 2006.
[9] http://www.forum.nokia.com – the links for WAP, Browsing, Symbian OS, Java Micro Edition
[10] http://www.w3schools.com – the links for WAP, WML, XHTML and XML