Security on Mobile Phones with Lightweight Cryptographic Message Syntax

Murat Yasin KUBİLAY[1], Albert LEVİ[2], Atilla ÖZGİT[3]
[1]UEKAE, TUBITAK Kocaeli/TURKEY, mkubilay@uekae.tubitak.gov.tr
[2]FENS, Sabanci University Istanbul/TURKEY, levi@sabanciuniv.edu
[3]CENG, METU Ankara/TURKEY, ozgit@ceng.metu.edu.tr

***Abstract*—Cryptographic Message Syntax (CMS) is a standard for protecting messages cryptographically. Using CMS, messages can be protected in different content types such as signed-data, enveloped-data, digested-data and authenticated-data. CMS is architected around certificate based key management and symmetric cryptography. In this article, a lightweight CMS envelope is proposed for the mobile phones which have limited memory and processing power, in order to provide the privacy of the data either stored on them or exchanged by means of multimedia messaging (MMS) and e-mail. A sample prototype is implemented on mobile phone which makes use of the proposed lightweight CMS. The prototype is compared with standard CMS in terms of application size and performance. The application size decreases approximately by 35% and the envelope preparation duration and resolution is much shorter with lightweight CMS in comparison to standard CMS.**

***Index Terms*—CMS, Mobile Phone, PKI, Security**

I. INTRODUCTION

For the last decade mobile phones have become an indispensable part of our life and being carried as an accessory. As the time passes, like every other technological device, the features and capabilities of the mobile phones are also evolved. Nowadays, they are used as music player, camera, camcorder, radio, voice recorder, document reader/writer, game box, internet browser, e-mail client, messenger (SMS/MMS) etc. in addition to their traditional roles of voice communication. With the increasing features of mobile phones, the security of the information stored on them has become an important concern. In case of lost or stolen mobile phones, the stored information can be easily compromised. Actually a greater security deficiency is not the data on a lost or stolen mobile phone. Everyday, millions of information is shared between mobile phone users by means of MMS, e-mail. Photos, videos, office documents, voice records are sent with almost no security. These data are encrypted on the air with a very weak encryption algorithm and can be easily decrypted with a PC processing power by a person who is sniffing them. A better known scenario is, either authorized or unauthorized employees of the GSM [1] operators can easily break the privacy of the communication.

In this article, a Public Key Infrastructure (PKI) [2] [3] [4] [5] based solution will be proposed to the problem explained in the previous paragraph. In the solution, an application running on the mobile phone prepares a cryptographic envelope for a file with the help of another application running on the subscriber identity module (SIM) [6] card. The sample implementations will be compared in terms of feasibility and performance.

The rest of this article is organized as follows. In Section 2, the modern cryptographic methodologies will be shortly explained. Then, X.509 Certificates [7] [8], which binds public key's to distinguished names will be discussed. And then, CMS (Cryptographic Message Syntax) [9] will be explained which can be used for preparing encrypted data envelope for different recipients. In Section 3, the design and implementation of a sample application, which makes use of the technologies discussed in Section 2, will be explained. In Section 4, different sample implementations will be compared, and benefits and drawbacks of them will be further argued. And finally in Section 5, there will be a summary and conclusion for secure communication with mobile phones.

II. LITERATURE

The desire of keeping communication confidential from unintended recipients exists since the human being started to communicate [10]. Countless methods for hiding data have been developed for thousands of years. These methods attempt to transform the words, letters, and bits to meaningless messages. The intended recipient must be able to transform the meaningless message to its original form in order to read the sender's message. There are two modern mechanisms to make such transformation, symmetric (secret key) ciphers and asymmetric (public key) ciphers.

Users of a public key require confidence that the associated private key is owned by the correct remote subject (person or system) with which an encryption or digital signature mechanism will be used. This confidence is obtained through the use of X.509 certificates, which are data structures that bind public key values to subjects [7]. The goal is to provide

single mechanism by which a relying party is assured that

- The integrity of the public key is provided
- The public key has been bound to the claimed owner in a trusted way

The Cryptographic Message Syntax (CMS) [9] is a standard published by IETF [11], which describes a general syntax for the data that may have cryptography applied to it, such as digital signatures, digital envelopes, digested-data, encrypted-data and authenticated-data. Recursion can be used in the syntax, so that, for example one can sign a previously enveloped data or one envelope can be nested inside another. CMS is architected around certificate-based key management.

Both X.509 Certificates and CMS types are DER encoded in Abstract Syntax Notation One (ASN.1) [12]. ASN.1 is the method used by Open Systems Interconnection [13] [16] for specifying abstract objects and Distinguished Encoding Rules (DER) [14] [15] is a set of rules for representing abstract objects as strings of ones and zeros.

III. A PROPOSAL FOR MOBILE PHONES SECURITY WITH KORUGAN

As explained in the first Chapter, in today's world mobiles phones are used for many purposes such as listening music, taking pictures, recording videos, watching TV, reading, generating documents, sending e-mail etc. in addition to voice communication. In the mobile phones, a wide range of digital files are stored and shared by means of MMS and e-mail. The privacy and security of the stored and shared files in the mobile phones is a major concern for many people since these files can easily be compromised by either unauthorized or authorized person due to the insufficient security precautions. In this section, a PKI based solution will be proposed in order to fix these concerns.

Today applications can be developed and installed on the mobile phones and as well as on the SIM cards. The proposed solution, named Korugan, is mainly composed of two applications. One of the applications will run on the SIM card which will handle the cryptographic operations and the other application will run on the mobile phone which will prepare and resolve a lightweight CMS envelope for the encrypted data. The application running an the SIM card is implemented as a javacard applet, while the application running on the mobile phone is implemented as a java midlet.

The steps for preparation of lightweight CMS envelope in Korugan is described below and depicted in Figure 1.

- A random symmetric encryption key is generated.
- The plain file is encrypted with the symmetric encryption key.
- The symmetric encryption key is encrypted for each recipient with their asymmetric public keys which are extracted from their X.509 certificates.
- The encrypted file and the encrypted symmetric keys are packed into the CMS envelope.

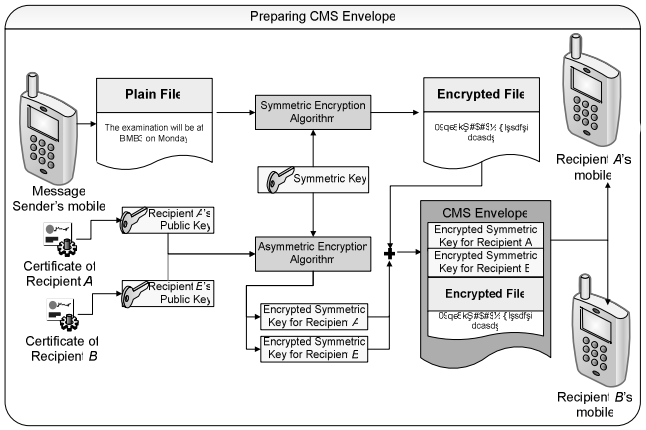

Fig. 1. Preparing CMS Envelope

Opening the steps of the lightweight CMS envelope in Korugan is described below and depicted in Figure 2.

- The lightweight CMS envelope is parsed, the recipient list and the encrypted content is identified.
- The recipient list is searched if the envelope is prepared for the message recipient.
- If the envelope is prepared for the message recipient, the encrypted symmetric encryption key is decrypted with the private key of the recipient which is stored in the SIM card.
- The encrypted message is decrypted with the symmetric key.

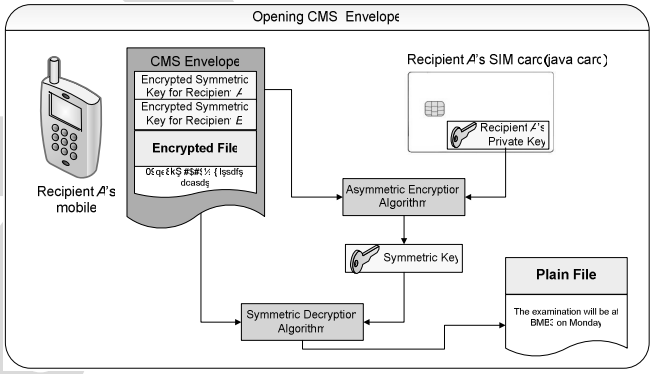

Fig. 2. Opening CMS Envelope

Since lightweight CMS envelopes are prepared using the X.509 certificates of the mobile phone users, the secure distribution of the X.509 certificates is very important. How can one know if a certificate belongs to an intended recipient? There may be two solutions two this problem. Either a certificate can be shared by some means like MMS, e-mail etc. between users, or can be downloaded from a trusted repository.

A. Lightweight CMS

CMS is very ideal for enveloping encrypted data for any number of recipients. Since it is used not only for enveloped data content type but also, signed data, encrypted data, authenticated data content types, its ASN.1 structure is quite complex. But in mobile phones we are only interested in the

enveloped data content type and do not need the rest. The required memory and processing power for DER encoding and decoding of the standard CMS structure is excessive for the mobile phones which have limited memory and processing power. For this reason a lightweight CMS structure is proposed in the following sections.

B. ContentInfo Type

For lightweight CMS, only the enveloped data content type will be supported.

C. EnvelopedData Type

In the EnvelopedData type the optional fields originatorInfo and unprotectedAttributes will be omitted. The ASN.1 structure of the EnvelopedData will be

```
EnvelopedData ::= SEQUENCE {
    version CMSVersion,
    recipientInfos RecipientInfos,
    encryptedContentInfo EncryptedContentInfo}
```

D. RecipientInfo Type

There are several key management techniques for encrypting content encryption keys. In lightweight CMS, only the key transfer is enough as a key management technique, and the other choices will be omitted. In order to keep the conformance to the standard, the ASN.1 choice structure of RecipientInfo is preserved.

```
RecipientInfo ::= CHOICE {
    ktri KeyTransRecipientInfo}
```

E. RecipientIdentifier Type

In the standard CMS, recipient can be identified either by certificate issuer distinguished name and the certificate serial number or by subjectKeyIdentifier which uniquely identifies the certificate by a key identifier. This is actually the subjectKeyIdentifier extension of the X.509 certificate. In lightweight CMS, both choices are omitted. Since, in the first choice issuerAndSerialNumber, issuer is represented in the ASN.1 name structure. Encoding, decoding and making comparison with this structure is quite complex and time consuming. Also in order to get the subjectKeyIdentifier of a certificate, the extensions of a certificate have to be decoded and identified. In addition, this extension may not be present in some of the certificates at all. In order to get rid of the unnecessary and heavy ASN.1 encoding, decoding, comparison process, a new alternative, the choice issuerHashAndSerialNumber is introduced. The ASN.1 structure of the type IssuerHashAndSerialNumber is:

```
IssuerHashAndSerialNumber ::= SEQUENCE {
    issuerHash OCTET String,
    issuerHashAlgorithm AlgorithmIdentifier
    serialNumber INTEGER}
```

The fields of type IssuerHashAndSerialNumber have the following meanings:

issuerHash is the result of hashing the certificate issuer with the specified hash algorithm. issuerHashAlgorithm is the object id of the hash algorithm used for hashing the certificate issuer. serialNumber is the certificate serial number which is unique for a certificate authority.

IV. PERFORMANCE ANALYSIS

There are several dozens of mobile phone manufacturers and hundreds of different mobile phone models in the market [17][18]. These mobile phones differ from each other according to their technical specifications such as operating system, memory size, data storage media size, processing power etc. Korugan can be used by only the mobile phones which support java and use SIM card having java card capabilities. One of the motivations to implement two versions of Korugan is to get rid of the dependency to the limitations of the mobile phone by handling all cryptographic operations in the java card. A second motivation is to compare the performance of the application when the cryptographic operations are done in the mobile phone and in the java card.

Another issue of interest is the benefits of Korugan by introducing the lightweight CMS. If Korugan was implemented according to standard CMS, what would be the size and performance of the application? In order to make this comparison, a third version of Korugan is implemented which is preparing the envelope according to standard CMS. In order to answer these questions, Korugan versions will be compared in terms of size and performance in the following paragraphs.

The installed application size is one of the important criteria for the mobile phones which have limited storage media. Table 1 and Table 2 compare media storage size for two versions of Korugan and Korugan implementation with standard CMS.

TABLE 1
JAVA CARD APPLET SIZE (IN KB) COMPARISON

Java Card Applet Size (in KB) Comparison

Whole Cryptography in Applet	4 2
Only Asymmetric Decryption in Applet	3 1

The size of the java card applet is almost the same for the two versions. The whole cryptography for the CMS envelope preparation and resolution can be done in the java card. But the memory limitations of the java card for symmetric encryption of big sized data must be considered as a drawback of this option.

TABLE 2
MIDLET APPLICATION SIZE (IN KB) COMPARISON

Midlet Application Size (in KE) Comparison

Cryptography in Applet with Lightweight CMS	88 2
Cryptography (except asymmetric decryption) in Midlet with Lightweight CMS	91 7
Cryptography in Applet with Standard CMS	133 2
Cryptography (except asymmetric decryption) in Midlet with Standard CMS	136 7

TABLE 1
KORUGAN PERFORMANCE COMPARISON FOR DIFFERENT NUMBER OF RECIPIENTS

Korugan Performance Comparison for Different Number of Recipients

	Number of Recipient	Data Size	RecipientInfo Preparation (ms)	Data Encryption (ms)	Total CMS Envelope Preparation (ms)	Opening the CMS Envelope and Finding the Recipient (ms)	Data Decryption (ms)
Cryptograhy in Midlet with Lightweight CMS	1	1 KB	52	11	168	83	15
	3		147	11	237	137	15
	5		166	11	291	143	15
	10		293	11	491	176	15
Cryptograhy in Applet with Lightweight CMS	1	1 KB	66	24	190	87	27
	3		189	24	292	141	27
	5		336	24	374	147	27
	10		433	24	644	180	27
Cryptograhy in Midlet with Standard CMS	1	1 KB	76	11	176	116	15
	3		300	11	411	208	15
	5		526	11	624	252	15
	10		729	11	1024	274	15
Cryptograhy in Applet with Standard CMS	1	1 KB	90	24	203	120	27
	3		342	24	464	212	27
	5		596	24	707	256	27
	10		869	24	1177	278	27

TABLE 2
KORUGAN PERFORMANCE COMPARISON FOR DIFFERENT SIZE OF DATA

Korugan Performance Comparison for Different Size of Data

	Data Size	Number of Recipient	RecipientInfo Preparation (ms)	Data Encryption (ms)	Total CMS Envelope Preparation (ms)	Opening the CMS Envelope and Finding the Recipient (ms)	Data Decryption (ms)
Cryptograhy in Midlet with Lightweight CMS	50 KB	1	34	45	91	41	49
	100 KB			86	172	72	91
	1000 KB			771	1272	477	949
Cryptograhy in Midlet with Standard CMS	50 KB	1	65	45	145	76	49
	100 KB			86	221	134	91
	1000 KB			771	1412	665	949

The java midlet size differs by 3.5 KB according to the application handling the cryptographic operations. Due to the drawback of the java card application in symmetric ciphering mentioned in the above paragraph, if the mobile phone is capable of handling cryptographic operations, it is a better media for handling these operations. Nevertheless, asymmetric decryption has always to be done by the java card.

The application size decreases approximately by 35 % (45 KB) with lightweight CMS in comparison to standard CMS. For the old generation mobile phones, this may be a significant amount of storage size, but for the new generations it may not be so important since the storage media size is increasing in chunks of MB as the time passes.

A second comparison criterion is the average speed of the envelope preparation and extraction for the CMS versions. In order to make the comparison, the midlets of Korugan versions are deployed into a Nokia N73 mobile phone. Unfortunately, the applet could not be deployed into the javacard due to the insufficient support from the mobile phone operators. The comparison data is collected as an average of 10 runs on the mobile phone. For the versions which cryptography is handled in the javacard applet and the asymmetric decryption operation, the cryptographic operations' durations are gathered from the javacard emulation environment.

In Table 3, Korugan versions are compared according to envelopes prepared for different numbers of recipients. The envelope preparation and extraction process are divided into components and the execution durations are measured for each component separately. When the cryptographic operations are handled in the mobile phone, CMS envelope preparation and extraction is apparently faster because of thememory and processing power limitations of the java card. The recipientInfo preparation duration is faster in the lightweight CMS versions, since the issuer and the serial field values from X.509 certificates are extracted in a more efficient way and unnecessary fields are not parsed. With the increasing number of the recipients in the envelope, the difference in the preparation time increases between versions.

Envelope extraction time for a recipient depends on two factors. The first factor is the position of the recipient in the RecipientInfos structure and the second factor is the comparison criteria to find a recipient. In Table 3, the extraction is done for the last recipient in the RecipientInfos structure in the envelope. As described in the previous section, the recipient is identified according to the issuerAndSerialNumber structure in standard CMS. The ASN.1 syntax of issuer structure in issuerAndSerialNumber is Name, which can be composed of many different attributes

such as common name, organization, organizational unit, domain component, locality, country etc. The identification, encoding, decoding and comparison of these attributes are bypassed in lightweight CMS with the introduction of issuerHashAndSerial structure which results in the performance increase in Korugan compared to standard CMS implementations. With increasing number of comparisons, the performance of lightweight CMS is getting more obvious.

Since the javacard is not suitable for bulk data symmetric encryption, in Table 4 only Korugan midlet versions are compared according to size of the data to be enveloped. Due to the reasons described in the above paragraph, recipientInfo preparation and envelope extraction duration is shorter in lightweight CMS version. With the increasing size of the enveloped data, the envelope preparation and extraction duration is also increasing.

Another comparison criterion is the limit of the number of recipients in the envelope. For each recipient, the size of the envelope is increasing about 185 bytes. Since the data is encrypted once for all of the recipients, the number of recipients does not impact data encryption process. As a result, the number of recipients in an envelope is limited with the size of storage media of the mobile phone which it will be stored.

V. CONCLUSION

During the last decade, mobile phones have become an indispensable part of our daily life. Nowadays, mobile phones comprise of several features, such as music player, camera, camcorder, e-mail client, etc. Several types of documents and multimedia content are being shared widely by means of MMS and e-mail between mobile phone users. Since the files in the mobile phone are generally not encrypted and the standard on-the-air encryption is very weak, these files can be easily captured via several types of attacks.

Despite commonly accepted and severe privacy concerns in mobile phone communication, there are no widely used and effective security measures against these privacy breaches. In order to prevent these breaches, the low level security standards of the network can be bypassed. The security of the multimedia content in the mobile phones can be provided from end to end by enveloping them with CMS. CMS is a standard published by IETF for protecting messages cryptographically. Any form of digital data can be digitally signed, encrypted and enveloped by using CMS which is architected around certificate-based key management.

Mobile phones have limited storage media, memory and processing power as compared to computers. Since the CMS ASN.1 structure is quite complex, its implementation is costly in the mobile phones. In this article, in order to meet the performance restrictions of mobile phones a lightweight CMS is proposed. The lightweight CMS is used to envelope the files to be stored in mobile phones or to be shared between users by means of MMS and e-mail. In lightweight CMS, unnecessary syntax components irrelevant for enveloping are removed and ASN.1 Name related syntax is simplified. A sample mobile application, which makes use of the lightweight CMS, named Korugan is developed. Using Korugan, any file in the mobile phone can be digitally enveloped for any number of recipients. In the envelope, the plain file is encrypted with symmetric encryption and the symmetric encryption key is encrypted with the public keys of each recipient extracted from their X.509 certificates. The recipients can resolve the envelope using their private keys, and obtain the plain file using Korugan.

In Korugan, cryptographic operations can be implemented either in the mobile phone or in the SIM card except the asymmetric decryption which requires a private key operation stored on the SIM card. Two different versions of Korugan are implemented. These implementations differ according to the media where cryptographic operations are handled. In order to make comparisons, another base implementation is also developed which uses standard CMS. The Korugan versions and the base implementation are compared to each other in terms of application size and performance. When the cryptographic operations are handled in the mobile phone, CMS envelope preparation and extraction are obviously faster due to the memory limitations of the SIM card. The application size and the envelope preparation duration decrease significantly when the lightweight CMS is employed in Korugan instead of standard CMS.

Lightweight CMS proposes syntax only for data envelopment for any number of recipients, although standard CMS enables encapsulation syntax for digital data signing, encryption for local storage, digestion for content integrity, authentication for content integrity for any number of recipients in addition to enveloping data. As the secret key distribution method, among the options key transfer, key agreement, key encryption key and password transfer which are available in standard CMS, lightweight CMS only supports the option key transfer.

Despite the privacy of the voice is not provided by Korugan; nevertheless it can be a good candidate for providing the privacy of the stored and transmitted huge amount of data by means of MMS and e-mail within the mobile phone.

REFERENCES

[1] GSM, Global System for Mobile Communications, http://en.wikipedia.org/wiki/GSM , Last Access Date: 21.08.2007.
[2] PKI: Implementing and Managing E-Security , Andrew Nash, William Duane, Celia Joseph, Derek Brink, Osborne/McGraw-Hill, 2001.
[3] Cryptography and Public Key Infrastructure on the Internet , Klaus Schmeh, John Wiley & Sons, 2003.
[4] Public-Key Cryptography , Salomaa, Arto, Springer-Verlag, 1990.
[5] Authentication: From Passwords to Public Keys , Richard E. Smith , Addison-Wesley, 2002.
[6] SIM, Subscriber Identity Module, http://en.wikipedia.org/wiki/Subscriber_Identity_Module, Last Access Date: 21.08.2007.
[7] RFC 2459 Internet X.509 Public Key Infrastructure Certificate and Certificate Revocation List (CRL) Profile.
[8] ITU-T X.509 The Directory: Public-key and Attribute Certificate Frameworks.
[9] RFC 3852 Cryptographic Message Syntax (CMS).
[10] Understanding PKI second Edition Concepts, Standards and Deployment Considerations, Carliste Adams and Steve Lloyd.

[11] IETF, Internet Engineering Task Force, www.ietf.org , Last Access Date: 21.08.2007.
[12] CCITT. Recommendation X:208: Specification of Abstract Syntax Notation One (ASN.1). 1988.
[13] CCITT. Recommendation X.200: Reference Model of Open Systems Inteconnection for CCITT Applications. 1984.
[14] CCITT. Recommendation X.209: Specification of Basic Encoding Rules for Abstract Syntax Notation One (ASN.1). 1988.
[15] ITU-T X.690 ASN.1 encoding rules: Specification of Basic Encoding Rules (BER), Canonical Encoding Rules (CER) and Distinguished Encoding Rules (DER).
[16] ITU-T Recommendation X.501, Information Technology - Open Systems Interconnection – The directory models, 1993.
[17] Nokia, www.nokia.com.tr , Last Access Date: 21.08.2007.
[18] Sony Ericsson, www.sonyericsson.com , Last Access Date: 21.08.2007.