Asymmetric Cryptography for Mobile Devices

Eric Neidhardt*
*neidhard@cs.tu-berlin.de
Service-centric Networking
Telekom Innovation Laboratories and TU Berlin
Berlin, Germany

***Abstract*—This paper is meant to give the reader a general overview about the application of asymmetric cryptography in communication, particular in mobile devices. The basics principles of a cryptosystem are addressed, as well as the idea of symmetric and asymmetric cryptography.**

The functional principles of RSA encryption and the Diffie-Hellman key exchange scheme, as well as the general idea of digital signatures are shortly described.

Furthermore some challenges and solution approaches for the application of asymmetric encryption in low power mobile devices are named.

On completing, some the future developments in cryptography are shortly described.

I. Introduction

The ability to transfer sensible messages, while ensuring privacy has always been an important requirement for man. It was crucial for military commanders, diplomatic emissaries or political leaders, as for example a battle plan that falls into enemy hands could easy lead to a devastating defeat [1].

Providing this desired privacy in communication was not easy, since communication channels have always been insecure by definition. Cryptography has always been the most important tool to provide the communicating parties with some aspects of security. Security is often defined as integrity, confidentiality, authentication and availability [2]. Assuming this definition, cryptography alone is not enough to provide the complete area of security. Therefor the next sections of this paper are going to explain that the application of cryptography can provide confidentiality and up to a certain level integrity and authentication.

The used methods to encrypt messages have greatly evolvent during the centuries. Simple rotating ciphers like the Caesar cipher where sufficient during the ancient time, but they are no match for even the slowest modern computer. This encryption schemes simply rotate the characters of the alphabet by key positions. As a result there is only a small number of different keys and a fast computer can simply try them all out in no time [1]. As knowledge increases and tools became more sophisticated, the cryptographic techniques need to be enhanced as well.

Not only cryptographic techniques have been evolved, but also many divererent fields for using cryptography had accrued. In the past it was mainly used by government or military, but worldwide communication and trade via the internet would not be practical without the security provided by cryptography [3].

One primary issue of cryptography has remained since it was first used. This issue is the distribution of the keys. As Section II-A is going to explain, cryptography is all about encrypting and decrypting a message using a key which may look quite different. For a long time the same key was shared by both communicating parties, resulting in so called symmetric cryptography.

How can the key be exchanged between the sender of a message and the according receiver, while still ensuring security? Of course this is an extremely important point to discuss, as compromising the key during the initial key exchange will compromise the whole communication process.

In the last century Diffie and Hellman proposed a new kind of cryptography using two keys, one being private and another one being public. The private does not need to be distributed and the public, which needs to be exchanged may fall into enemy hands without compromising the communication. This new approach in cryptography is called assymmetric and it would not be possible without the increased performance of modern digital hardware [1]. Nevertheless asymmetric encryption is a computational intensive task. Therefor this scheme is difficult to employ on low power mobile devices.

II. Cryptosystem Basics

This section will outline the concept behind a cryptosystem and why it is essential for providing several security aspects. Furthermore the term security is explained in more details and the properties of a good cryptosystem are explained.

According to Bellare and Rogaway [3] a cryptosystem is in general a pair of algorithms that uses a key to convert a plaintext to ciphertext and vice versa. As stated before, this is normally used to provide a secure communication between a sender and a receiver.

The basic communication scheme is shown in Figure 1. The figure represents a communication between the Sender who uses a key K_s to encrypt a message and the receiver who uses another key K_r to decrypt the message. The goal of this communication is to provide a as secure as possible message exchange against a possible adversary. If the decryption key is the same as the encryption key, this kind of communication scheme is called symmetric. By implication the scheme is called asymmetric, if the keys differ [4], [3]. This will be described in detail in the following sections.

As mentioned before security is often defined as confidentiality, integrity, authentication and availability [2]:

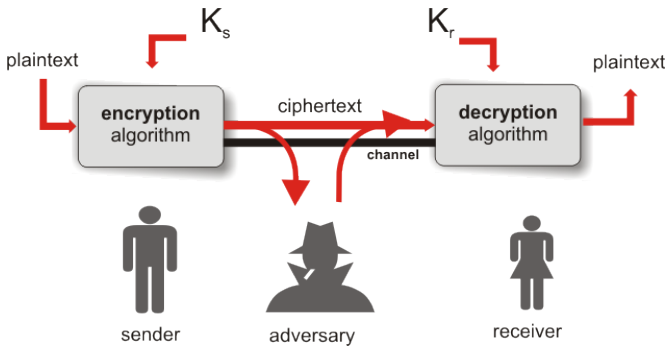

Fig. 1. cryptography in communication

- *Confidentiality.*
  Prevent unauthorized access of data.
- *Integrity.*
  Prevent unauthorized alteration or manipulation of data.
- *Availability.*
  Prevent unauthorized withhold of a data.
- *Authentication.*
  Verify the identity of the communicating parties.

The first aspect of security is confidentiality, which means that unauthorized access of certain vital data must be prevented. The second aspect is integrity. Unauthorized alteration of data must be prevented. This is quite difficult to achieve, so normally integrity is attained by at least detecting unauthorized alteration and retransfering the corrupted data. The third aspect of security is the availability. This is often forgotten, but no system or communication can be secure, if an unauthorized person would be able to prevent the access of this system or communication. The last aspect is authentication. This is a quite important point to consider, as verifying the identity of the sender and the receiver is important in almost every communication [2], [5]. Authentication is very important in the field of electronic commerce and digital trade, where contracts must be verified.

Obviously cryptography can be employed to provide confidentiality. The data is encrypted via a secret key and thereby protected from attackers who intercept the encrypted message. Without the decryption key an unauthorized third person is no able to read the content of the encrypted message. Cryptography can also be used to provide a certain kind of integrity for data [4]. As I stated before this is done by providing the ability to detect unauthorized alteration. If the sender and the receiver in communication are using the same key, the encryption of the message itself can provide some integrity check, as alteration of a message without knowing the key will most often result in a rubbish message after decryption and is therefor easy detected. Of course we can not distinguish between unauthorized alteration or if the sender has send rubbish in the first place. As a result this is no real integrity check. Because of this limitation cryptographic checksums are used on the messages to provide integrity [6], but this explained in detail in the following Section VII. Nevertheless it is not possible to distinguish between a wilful alteration or jitter on the communication channel. As stated before authentication is an important goal in cryptography. Digital signatures can be used in asymmetric encryption to verify the origin of data[7]. Of course cryptography will not be able to provide availability. Other tools are required to ensure this aspect of security.

There are some properties a good cryptosystem should match, otherwise it would be a weak scheme and therefor vulnerable to an attacker even with the key uncompromised [3]:

- There should be enough possible keys to make any attempt to find the key via simply trying out all possible keys infeasible.
- The generated ciphertext should look random.
- Ideally it should be impossible to get the plaintext from the ciphertext without knowing the key, even if the algorithm is known.

The most important requirement, which plays an important role in all modern cryptosystems was stated by Kerckhoffs [8]. He stated, that the security of an encryption should rely on the security of the key alone and not on the obscurity of the used encryption algorithm. This was a major contrast to early approaches were the securiy was achived by keeping the encryption process secret. An example of this is the Caesar cipher where the letters in a message are only rotated by three (A becomes D etc.) [1]. Such a encryption would be of no use if the adversary knows the algorithm.

A. Attacking a Cryptosystem

In general there are three approaches for attacking a cryptosystem [1], defined by the abilities of a possible attacker. As stated above, we must always assume that the attacker is also in possession of the used encryption algorithm:

- *Ciphertext only attack.*
  The attacker is only in possession of the ciphertext. His goal is the plaintext or the used key.
- *Known plaintext attack.*
  The attacker is in possession of both the plaintext and the ciphertext. His goal is to find the used key.
- *Chosen plaintext attack.*
  The attacker has the ability to get corresponding ciphertext from a chosen plaintext. His goal is to find the used key.

Obvious the occurrence of this attacks may differ greatly. It is quite difficult for an attacker to get ciphertext for his chosen plaintext. Usually attacks are ciphertext only [1].

One particular type of attack which is most often used is the so-called brute force attack. Brute force simply mean that an attacker tries out all possible keys to find the matching one. Because of this a good cryptosystem should have enough keys to make this attack infeasible. A cryptosystem being vulnerable to brute force is most often considered very weak.

III. Symmetric Cryptography

As stated before, cryptographic schemes can be distinguished into symmetric and and asymmetric [4], [3]. The symmetric scheme is the classical one.

The idea of symmetric cryptography is simple to understand. As we already know, cryptography is usually employed to provide a secure communication between a sender and a receiver. As described in section I, security in this context means privacy. In symmetric cryptography, a plaintext is encrypted by the sender using a secret key and decrypted be the receiver using the same key. As a consequence the key must be distributed between the sender and the receiver, which leads to a possible security flaw [1], [3].

According to Bellare and Rogaway, a symmetric encryption scheme consists of three parts:

- *encryption algorithm.* The algorithm to generate ciphertext from plaintext using a secret key.
- *decryption algorithm.* The algorithm to decrypt the ciphertext using the same secret key as the encryption algorithm and returning the plaintext.
- *key generation algorithm.* An Algorithm to generate a random, secret key.

A. Substitution Cipher

An important property of a cryptosystem, which was already mentioned in I is that the generated ciphertext should ideally appear to be random. If we consider the classic substitution cipher, this demand becomes obvious.

In a substitution cipher every character of the plaintext is replaced by another character according to some kind of secret map [3]. The map would be the shared key in this encryption scheme. The drawback of this encryption scheme is, that the characteristics of human language are not hidden. For example the letter e appears much more frequently than the letter z. By counting the letters of the ciphertext (frequency analysis) an adversary would be able to make several assumptions of the used map and may be able to break the encryption [3].

Nevertheless there are some approaches to improve the effectiveness of this encryption scheme by changing the substitution map during the operation [9].

B. Block Cipher and Modes of Operation

Usually a message is to large to be encrypted in one pass. Therefor the message can be split into blocks usually each of the size of the key. After that it is encrypted blockwise, resulting in a so-called block cipher [3].

The mode of operation defines how a message is encrypted and decrypted using a block cipher. All blocks in a message are encrypted with the same key. Because of that the encryption algorithm must be applied with care to avoid additional security flaws. Two simple modes of operation are the ECB mode (Electronic Code Book) and the CBC mode (Cipher-block chaining) [3]. This modes are explained to clarify the basic principles. Real world block ciphers would be the Data Encryption Standard and the Advanced Encryption Standard [10].

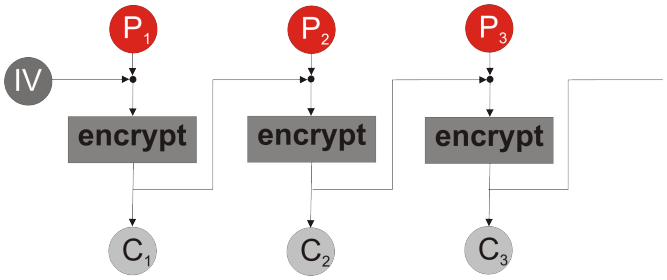

Fig. 2. Cipher-block chaining mode

1) Electronic Code Book: This relatively simple mode represents a direct application of the used block cipher. Each block is encrypted using the key and the resulting ciphertexts are send individually or packed together before. This encryption mode has the drawback that the same plaintext will always encrypted to the same ciphertext [3]. Therefor the overall pattern of the message is not hidden and the ciphertext may do not appear to be random. As a result an eavesdropping attacker may be able to create a table of plaintext and ciphertext pairs, making the ECB encryption very weak.

For more information about modes of operations and their weak spots for attacks, I suggest to consult the paper of Biham [11].

2) Cipher-block chaining: Similar to the ECB mode the CBC mode also works on blocks, but in contrary the generated ciphertext C_n does not only depend on the key and the used block P_n, but also on all previous blocks. This is shown in Figure 2. As a result the final output of the encryption looks much more random. This first block in this scheme is encrypted using some initial value IV.

C. Brute Force

According to Kerckhoffs law [8] we must assume, that an adversary knows the encryption algorithm. As there is a finite number of possible keys, we might consider it to be possible that the adversary may break the encryption by simple trying all possible keys, taking advantage of rapidly increasing computer performance. This is called brute force. Actually the increasing performance favors the encoder, as we can double the amount of possible keys by simply adding one bit to the size of the key. The workload for calculating the enlarged key with the key generation algorithm is most likely compensated by the increased performance of our hardware. Contrary the redoubled key space will increase the workload for a brute force attempt significantly.

IV. One-Time Pad Encryption

This section is going to explain the so-called one-time pad encryption as being a classic symmetric encryption scheme. Nevertheless the one-time pad encryption is stated as being secure [3].

The algorithm itself is quite simple. The message is bitwise encrypted with the key using exclusive OR. Therefor it is necessary that the used key and the message have got the same bit length.

$$m \text{ message to send with n bits}$$

$$K \text{ shared key with n bits}$$

$$c = m \oplus K \text{ ciphertext}$$

The receiver of the encrypted message is now able to decrypt the message using the same shared key.

$$c \text{ ciphertext received with n bits}$$

$$K \text{ shared key with n bits}$$

$$m = c \oplus K \text{ ciphertext}$$

According to Bellare and Rogaway [3] this encryption scheme is perfect secure, unless the same key is used twice. If an adversary would be able to intercept two message, which where encrypted with the same key, he would be able to derive some information from the message.

$$m_1 \text{ message 1}$$

$$m_2 \text{ message 2}$$

$$K \text{ shared key}$$

$$c_1 = m_1 \oplus K \text{ ciphertext 1}$$

$$c_2 = m_2 \oplus K \text{ ciphertext 2}$$

By simply XORing the encrypted messages together, the attacker would be be able to obtain $m_1 \oplus m_2$. Furthermore if the adversary would be able to obtain one decrypted message, he could obtain the other [3].

$$c_1 \oplus c_2 = m_1 \oplus m_2$$

It is recommended to read [3] for a proof of the one-time pad security.

V. Asymmetric Cryptography

Modern symmetric cryptography is sophisticated highly available and quite secure, but the requirement to have both communicating parties to share the same secret key [1] is a great drawback. Obviously this results in security flaws, unless there is some secure way to exchange the key.

To match the prerequirement of a shared key, the key could be sent using another communication channel, for example via mail. This has two major disadvantages. The first one is, that you are required to relay on the confidentiality of the courier, as he may compromise the key. The second disadvantage is of course the additional setup time for a secure communication. Another possible option would be to give all keys to a trusted third party, for example some kind of global key distribution center [1]. Of course this is not a good solution for distributing private keys worldwide, as such a key distribution center would be an extremely valuable target for any attack. Furthermore it is unlikely that everyone is willing to trust such a key distribution center. Of course both options would not scale for the size of the internet.

It is therefor obvious that secure communication via the internet would be impossible using classic cryptographic schemes. Therefor it was certainly a major breakthrough in cryptography when Diffie and Hellman proposed a new encryption scheme in 1976 which was able to counter this problem. This scheme is called asymmetric encryption or public key cryptosystem. In this cryptosystem, one public key is used to encrypt the messages and another private key is used to decrypt them [1].

Using this kind encryption it is possible for the communicating parties to relay on confidential communication without the need to distribute a private key. As a result a public key encryption can be set up quite fast in comparison to a key exchange via mail, for example. This also improves the security compared to a private key exchange via a third party, as there are fewer people involved in the communication process. Another opportunity acquired with public key encryption is the ability to digitally sign messages and data and therefor make digital contract possible. This was not possible using a symmetric encryption scheme, as it must be impossible for anyone else to reproduce such a signature. Be cause of the shared key in a symmetric encryption scheme the person validating the signature using the key would be able to reproduce it [1], [12].

The three main components of an asymmetric encryption scheme are same as in an symmetric encryption scheme, but their application is slightly different [3]:

- *encryption algorithm.* The algorithm to generate ciphertext from plaintext using the public key.
- *decryption algorithm.* The algorithm to decrypt the ciphertext using the secret key and returning the plaintext.
- *key generation algorithm.* The algorithm to generate two random keys. One key being the public key and the other key being the secret one. Usually both keys are very large compared to symmetric keys.

There are some characteristics a public key encryption scheme should match:

- As the encryption key is public it should be impossible to obtain the private decryption key from the encryption key in a reasonable amount of time. Both keys must be very large, otherwise it would be not computationally infeasible to obtain the private key from the public key.
- It should be relatively easy to encrypt and decrypt a message using the appropriate key.

A. Trapdoor Functions

Asymmetric or public key encryption is based on trapdoor functions [13]. These functions are relatively easy to compute in one direction, but the inverse function is hard to compute. Nevertheless the inverse can be much more easier calculated if a so-called trapdoor information is given [14]. This trapdoor information must be hold secret in order to guarantee the safety of the secret key [1], [7].

Current asymmetric encryption schemes make use of the hardness of certain mathematical problems. This paper is not meant to go into the mathematical details, as this would be beyond the scope of this topic. I suggest to consult chapter 10 in [3] for more detailed information.

According to Tillich and Großschädl [14] these mathematical problems are the integer factorization and the discrete logarithm problem. The hardness of the used problem correlates to the required key size. As the performance of the cryptographic algorithm is depending on the size of the keys, a harder mathematical problem as a basis for the algorithm will result in a more efficient encryption scheme [14].

VI. RSA

This sections is meant to give a general idea of one of the widely used public key encryption schemes. This scheme is the RSA encryption developed by Rivest, Shamir and Adleman. This section does not describe the mathematical details, but only the procedure of RSA. All of the following descriptions refer to the work of Rivest et al. [7]. This source should also be consulted for the underlying mathematics.

A. Key Setup

At the beginning of the key setup, each user must randomly select two large prime numbers. The prime numbers are multiplicated and the euler totient function is used.

$$p, q \text{ two prime numbers}$$

$$n = p * q$$

$$\varphi(n) = (p-1) * (q-1) \text{ euler totient}$$

Now the private decryption key d is selected with the condition the key and the result of euler totient have no common divisor except one.

$$gcd(d, \varphi(n)) = 1$$

$$d \text{ encryption key}$$

The public encryption e is the inverse of d multiplicated with the result of the euler totient function. Therefor it is obtained by solving the equation.

$$e * d = 1 \text{ modulo } \varphi(n)$$

$$e \text{ decryption key}$$

As a result we have got the following key pair.

$$(d) \text{ private decryption key}$$

$$(e, n) \text{ public encryption key}$$

B. Using RSA

In the last section the calculation of both the private and the public key was described. The following section will give short overview of how the generated keys are used to encrypt and decrypt a message. To decrypt a message, the sender has to obtain the public encryption key from the receiver.

$$\text{m message to send}$$

$$(e, n) \text{ receivers public encryption key}$$

$$c = m^e \text{ modulo } n \text{ ciphertext}$$

The receiver can now the decrypt the received ciphertext by using his private key. This is describe in detail by Rivest et al. [7].

$$\text{c ciphertext received}$$

$$(d) \text{ private decryption key}$$

$$m = c^d \text{ modulo } n \text{ obtained plaintext}$$

The encryption and decryption process perform flawlessly because of the following equations:

$$c^d = (m^e)^d = m^{e*d}$$

$$c^d = m^{1 \text{ modulo } \varphi(n)}$$

$$c^d = m^1 \text{ if } \varphi(n) > 1$$

$$c^d = m$$

VII. Diffie-Hellman Key Exchange Scheme

One special shape of asymmetric encryption is the so-called Diffie-Hellman Key Exchange Scheme. This cryptographic scheme is not used to encrypt a message in general, but to distribute a shared key, which is derived from the public and the private keys of both communicating parties. This scheme was proposed by Diffie and Hellman in [1].

At the beginning of the key exchange, each one must calculate his public and his private key. A prerequirement of the key generation is the presence of two values, shared by both users. This are not secret, but the communicating parties have to agree on them prior.

$$q \text{ large prime numbers}$$

$$a \text{ primitive element of GF(q)}$$

Using this parameters each user choses his private key K_D and calculates his public key K_E.

$$K_D \text{ with } 1 \leq K_D < q$$

$$K_E = a^{K_D} \text{ modulo } q$$

Subsequently the communicating parties exchange there public keys. Using the exchanged keys, they are both able to compute the same shared key.

$$K_{DA}, K_{DB} \text{ secret key of user A, B}$$

$$K_{EA}, K_{EB} \text{ public key of user A, B}$$

$$K_S = a^{K_{DA} * K_{DB}} \text{ modulo } q$$

User A make calculate the shared key K_S using this equation:

$$K_S = K_{EA}^{K_{DB}} \text{ modulo } q$$

By implication the user B make is able to calculate the key using this equation:

$$K_S = K_{EB}^{K_{DA}} \text{ modulo } q$$

This shared key K_S may now be used for traditional symmetric encryption.

VIII. Authentication and Integrity

As mentioned in I authentication and integrity are two very important aspects of security, which can be provided via cryptography. Digital trade would not be possible without the ability to verify the integrity of contracts and the authentication of signers.

The goal of this section is to give a short overview of message authentication methods. The following explanations refer to the work of Bellare and Rogaway [3] and Rives et al. [7].

In symmetric encryption schemes a at least partial integrity check comes for free as a part of the encryption process itself. As stated before any alteration of the encrypted message will most likely result in rubbish. Of course the receiver can not distinguish if the sender has send a incomprehensibly message in the first place. Therefor another mechanism is used to provide integrity, called message authentication code.

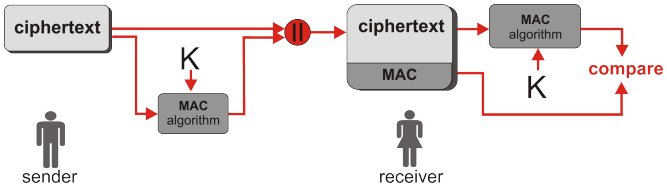

Fig. 3. message authentication code

In asymmetric encryption schemes this verification comes not natural, as the key used for encryption is public. Therefor anyone is able to encrypt messages like the sender. To verify the integrity of the message, the sender uses his private key to provide the message with a signature. This digital signature is also used to provide the authentication aspect of security, as only the holder of the private key is able to sign his message.

A. Message Authentication Code

On easy way to generate a signature is the use of a cryptographic checksum, the message authentication code. The message authentication code generates a small block of a fixed size using some shared key. The basic use of a message authenticate code is shown in Figure 3.

The sender generates the message authentication code using the key K and appends it to the message, which is usually the encrypted ciphertext. The receiver uses the same MAC algorithm with the shared key K on the message without the appended MAC and compares the result with the appended MAC. If both match, the message was not altered.

$$k, m^s \text{ key, message to send}$$

$$t^1 = MAC(k, m^s) \text{ message authentication code}$$

$$t^2, m^r \text{ received mac, received message}$$

$$t^2 = MAC(k, m^r) \text{ message authentication code}$$

$$t^1 = t^2 \text{ unaltered message}$$

Using a message authentication code requires a shared key. Therefor it can only be used in symmetric encryption or if public key encryption was used to distribute a shared key.

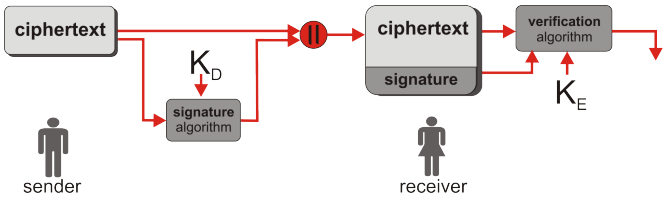

Fig. 4. digital signature

B. Digital Signatures

A digital signature is very important as it allows us to have authentication with the advantages of public key encryption. It is quite similar to a message authentication code but it makes use of the asymmetric keys. Like the message authentication code the generated signature is appended to the message we want to sign. The basic use of a digital signature is shown in Figure 4.

Contrary to the message authentication code two keys are used for authentication. The sender is using his private key K_D to generate the signature and append that signature to the message. The receiver uses the senders public key K_E to verify it. The RSA encryption is often used to generate digital signatures.

$$d, m \text{ senders private key, message}$$

$$s = sign(d, m) \text{ digital signature}$$

$$e, m, s \text{ senders public key, message, signature}$$

$$v = verify(e, m, s) \text{verify the signature}$$

$$v = true \text{ signature accepted}$$

IX. Application of Asymmetric Encryption

This section is meant to give a short summary of the applications of asymmetric encryption schemes. Some of this applications were already been mentioned in the previous sections.

In general asymmetric cryptography is normaly used for:

- *Message encryption.* Encrypting and decrypting a message for confidentiality.
- *Digital signatures.* Digitialy sign a message to provide authentication.
- *Key exchange.* Closly related to simple message encryption, but used to exchange a shared key for symmetric encryption.

Obviously the public key encryption can be used for message encryption and decryption and therefor providing confidentiality.

In the last section the ability to create digital signatures was already mentioned. As stated before this is extremly important in all aspects of modern electronic commerce. Without verifying the identity of the communicating parties it would not be possible to have digital contracts. Signatures are also very important in software distribution. For example Apple inc. wants to make sure, that only software approved by them and therefor usually obtained via their App Store can run on their smart phones. To guarantee this they provide their software packets with digital signatures wich are checked by the mobile device. Therefor the digital signature is used to authenticate the source of software [3], [1].

The last major application for public encryption is the exchange of keys. The asymmetric decryption and encryption process is rather costly compared to a symmetric scheme. As a result a public key scheme has a higher overhead and requires more computation power than a private key scheme [14]. This is especially important when it is used on low power mobile devices [15]. Therefor usually not the whole communication process is encrypted via a public key scheme but only the initial authentification and a key exchange. After both communicating parties have authenticate themself they negotiate a common key for symmetric encrypted communication.

X. Asymmetric Cryptography in Mobile Communication

The goal of this section is to have a more detailed view at the application of asymmetric cryptography in mobile communication and devices.

A. Threats and Challenges

Before some applications of asymmetric or public key cryptography in mobile devices are described, this section will recall the three basic threats to secure communication:

- *Eavesdropping.*
  The communication might be eavesdropped by a third person, therefor breaking the confidentiality.
- *Tampering and Injection.*
  The communication is manipulated by a third person, therefor breaking the integrity.
- *Spoofing.*
  A third person is pretending to be someone else, therefor breaking authentication.

As the communication in mobile devices is done wireless and therefor via a shared medium, mobile communication is especially vulnerable to these threats [5], [16]. Moreover, mobile devices share several characteristics which should be considered when encryption is applied. Most mobile devices lack the computational power of a Desktop PC. As these devices are powered by batteries, their power capacity is quite limited [16], [17].

As a result of this limitations it is possible that these mobile devices have only insufficient resources to apply a public key encryption scheme [14]. As described in Section IV these encryption schemes require intensive computation.

B. Server Based Signature

There are some approaches to reduce the workload for public key encryption on mobile devices. One approach is the Server Based Signature scheme proposed by Asokan et al. in [18]. Others had taken up the idea like Lei et al. in [17] The goal of this encryption scheme is to provide the authentication of digital signatures for mobile devices with feasible costs.

In principle the idea of server based signature is, that the Sender is using a signature server to obtain a digital signature for the message to be send. As a result the computation cost is partly outsourced to the signature server. Therefor the device itself is disencumbered. The drawback in this case is that there is additional communication between the sender and the signature server required [17]. This is especially crucial if the available bandwidth is limited.

Both sources [17], [18] should be studied if a more detailed pleadings is desired.

C. Cryptographic Co-Processor

The benefits of public key cryptography are evident, but there computational cost are also significant. Therefor it is often difficult to implement asymmetric encryption schemes in low power devices. Nevertheless the security provided via asymmetric encryption is desired even in ultra-low power environments like wireless sensor networks [19].

As a result the public key cryptography is emulated trough message authentication codes used in a stack of protocols [19]. This reduces the computation workload with the drawback of increased communication.

According to Gaubatz et al. the public key cryptography is usually implemented in software and the required ressources could be reduced if the encryption is implemented in hardware and employed via co-processor [20].

The authors had come to the conclusion that the reduced communication overhead in comparison to some kind of key management is worth the effort to employ a public key encryption scheme. Using hardware driven encryption they were able to implement the encryption with feasible computational costs [20].

XI. Conclusion

A secure communication has always been desired and cryptography has always been a tool to guarantee such a communication. It arose from the necessity of certain groups to transmit vital messages confidential. Typically these groups were military or governmental organizations. With the spreading of prompter communication, for example the rise of the world wide web, the demand for a secure way to communicate has increased overall. The internet is often considered as a motor for modern economy, because it makes it possible for traders all over the world to have a fast way to communicate. Certainly the internet would not have become so vital for economy if the communication process was not secure, as confidentiality is as important for a stock broker as it is for a diplomatic emissary.

An important aspect of cryptography is secrecy. As stated by Kerckhoff [8] secrecy in cryptography should not mean secrecy of the cryptographic scheme itself, but only on the secrecy of the used key. This is a contrast to early approaches in cryptography where the security was provided through hiding the encryption process. Furthermore not only the encryption algorithm used to be held secret, but also old encrypted and decrypted message must be held secret, otherwise it was possible to reproduce the key or the algorithm from old messages [1]. The development of modern computers and the rising of computational power naturally lead to new sophisticated cryptographic methods. Furthermore the spreading of general purpose computers and world wide communication has intensified the idea that secure communication, protected by cryptographic methods should be freely available.

The development of asymmetric encryption schemes was only the last breakthrough in a long process. Certainly the public key encryption has played a major role in the spreading of the internet, as electronical business and digital trade would certainly not be possible without confidentiality and integrity.

As stated before, asymmetric cryptography make use of mathematical problems. Most public key encryption schemes rely on integer factorization or the discrete logarithm problem. There are known algorithm for both problems to solve them in subexponential-time [14]. A new problem, called elliptic curve, which can not yet be solved in a reasonable amount of time, might dramatically improve the efficiency of modern cryptographic schemes.

Another major breakthrough in cryptography will be the use of quantum physics to generate a shared key for two communicating parties. This approach is called quantum cryptography [21]. It does not aim to improve public key cryptography, but to provide a quantum key distribution for symmetric encryption. Basically this approach makes use of the negative rules of quantum physics. One rule of quantum physics state, that a measurement of a quantum system will alter the system. If the key is a quantum information it can not be eavesdropped by an adversary without being changed. Therefor this approach can be used to safely distribute a key, using a public channel. The result of this scheme is a shared key which may be used for classical symmetric encryption, for example the already mentioned one-time pad encryption [22].

The development of cryptography has definitely not come to an end. Technology might change, but the requirement of confidentiality will remain and therefor the need of new cryptography schemes and techniques will remain too.

REFERENCES

[1] W. Diffie and M. Hellman, "New directions in cryptography," *IEEE Transactions on Information Theory, Volume 22 Issue 6*, pp. 644–654, 1976.

[2] M. Bishop, "What is computer security?" *IEEE Security Privacy, Volume 1 Issue 1*, pp. 67–69, 2003.

[3] M. Bellare and P. Rogaway, "Introduction to modern cryptography," 2005.

[4] G. J. Simmons, "Symmetric and asymmetric encryption," *ACM Computing Surveys Volume 11 Issue 4*, pp. 305–330, 1979.

[5] L. Elbaz, "Using public key cryptography in mobile phones," 2002.

[6] V. Hassler and H. Biely, "Digital signature management," *Internet Research, Volume 9 Issue 4*, pp. 262–271, 1999.

[7] R. Rives, A. Shamir, and L. Adleman, "A method for obtaining digital signatures and public-key cryptosystems," *Communications of the ACM, Volume 21 Issue 2*, pp. 120–126, 1978.

[8] A. Kerckhoffs, "La cryptographie militaire," *Journal des sciences militaires Volume 9*, pp. 5–38, 1883.

[9] T. Ritter, "Substitution cipher with pseudo-random shuffling: The dynamic substitution combiner," *Cryptologia, Volume 14 Issue 4*, pp. 289–303, 1990.

[10] F. P. Miller, A. F. Vandome, and J. McBrewster, *Advanced Encryption Standard*. Alpha Press, 2009.

[11] E. Biham, "Cryptanalysis of multiple modes of operation," *Journal of Cryptology, Volume 11 Issue 1*, pp. 45–58, 1998.

[12] M. Bellare and P. Rogaway, "The exact security of digital signatures-how to sign with rsa and rabin," *Lecture Notes in Computer Science, Volume 1070*, pp. 399–416, 1996.

[13] M. B. und Phillip Rogaway, "Optimal asymmetric encryption," *Lecture Notes in Computer Science, Volume 950*, pp. 92–111, 1995.

[14] S. Tillich and J. Großschädl, "A survey of public-key cryptography on j2me-enabled mobile devices," *Lecture Notes in Computer Science, Volume 3280*, pp. 935–944, 2004.

[15] M. Jakobsson and D. Pointcheval, "Mutual authentication for low-power mobile devices," *Lecture Notes in Computer Science, Volume 2339*, pp. 178–195, 2002.

[16] H. Yang, H. Luo, F. Ye, S. Lu, and L. Zhang, "Security in mobile ad hoc networks: challenges and solutions," *IEEE Wireless Communications, Volume 11 Issue 1*, pp. 38–47, 2004.

[17] Y. Lei, D. Chen, and Z. Jiang, "Generating digital signatures on mobile devices," *18th International Conference on Advanced Information Networking and Applications, Volume 2 Issue 1*, pp. 532–535, 2004.

[18] N. Asokan, G.Tsudik, and M. Waidner, "Server-supported signatures," *Lecture Notes in Computer Science, Volume 1146*, pp. 131–143, 1997.

[19] G. Gaubatz, J.-P. Kaps, and B. Sunar, "Public key cryptography in sensor networks-revisited," *Lecture Notes in Computer Science, Volume 3313*, pp. 2–18, 2005.

[20] G. Gaubatz, J.-P. Kaps, E. Öztürk, and B. Sunar, "State of the art in ultra-low power public key cryptography for wireless sensor networks," *Third IEEE International Conference on Pervasive Computing and Communications Workshops*, pp. 146–150, 2005.

[21] N. Gisin, G. Ribordy, W. Tittel, and H. Zbinden, "Quantum cryptography," *Reviews of Modern Physics, Volume 74*, pp. 145–195, 2002.

[22] F.-G. Deng and G. L. Long, "Secure direct communication with a quantum one-time pad," *Physical Review. A, Volume 69 Issue 5*, 2004.