International Baccalaureate

Extended essay cover

Candidates must complete this page and then give this cover and their final version of the extended essay to their supervisor.

Candidate session number			
Candidate name			
School number			
School name			
Examination session (May or November)	NOVEMBER	Year	2009

Diploma Programme subject in which this extended essay is registered: COMPUTER SCIENCE

(For an extended essay in the area of languages, state the language and whether it is group 1 or group 2.)

Title of the extended essay: CRYPTOGRAPHY AND ITS APPLIANCE IN TECHNOLOGY OF DIGITAL SIGNATURE

Candidate's declaration

If this declaration is not signed by the candidate the extended essay will not be assessed.

The extended essay I am submitting is my own work (apart from guidance allowed by the International Baccalaureate).

I have acknowledged each use of the words, graphics or ideas of another person, whether written, oral or visual.

I am aware that the word limit for all extended essays is 4000 words and that examiners are not required to read beyond this limit.

This is the final version of my extended essay.

Candidate's signature: Date: 11.9.2009.

IB Cardiff use only: **A:** **B:**

CRYPTOGRAPHY AND ITS APPLIANCE IN TECHNOLOGY OF DIGITAL SIGNATURE

Students name:
Supervisor name:
School number:
Student number:

Word count: 3987 words

Contents

1. Abstract

This essay gives a short overview of cryptography, crypto algorithms – symmetric and asymmetric, and how they are used in technology of digital signature. Because for generating of digital signature hash functions are needed, chapter 4 introduces the short overview about the basics of the most known cryptographic algorithms that use hash functions.

Digital signature has become increasingly popular around the world, and its appliance is widely spread. As well in my country its appliance is increased recently and as time passes it will become standard. Government and Agency for Information Society are working together on legislative for digital signature and digital certificate.

Usage of digital signature have become important, especially since huge amount of jobs introduced computers and have been computerized, as well as most of organizations are using an information system for their work.

This essay, in it most, describes appliance of digital signature and how it can be used in combination with digital certificate like very useful tool for providing security.

2. Introduction

Cryptography has both long and fascinating history. The most complete book of cryptography, un technical book, Kahn's *The Codebreakers* follows cryptography since its first appearance 4000 years ago in Egypt to Cesar's chypher of substitution where every character was substitute with a character third from it, and till twenties century where cryptography played important role in both World Wars.

To reversal in cryptography, after DES, came just after announcing of Diffie's and Hellman's article, named New Directions in Cryptography, which take us to revolutionary concept of PKI and to new direction of secure exchange of keys.

Instead of encryption with only one key – private key, which was problematic in sending it from sender to receiver, Diffie and Hellman suggested public key concept, and also they indicated need for digital signature as electronic equivalent to personal signature. After them, on 1978 RSA was created by three scientists - Rivest, Shamir and Adelman, based on impossibility of factoring large integers. One of the most significant 'products' that came from public key cryptography is digital signature – in 1991 the first international standard for digital signatures (ISO/IEC 9796) was adopted and digital signature was based on asymmetric crypto algorithm RSA.

"Cryptography is the study of mathematical techniques related to aspects of information security such as confidentiality, data integrity, entity authentication, and data origin authentication"[1].

The main purpose of cryptography was finding a system that will provide following:

- ***Confidentiality*** – Confidentiality is used for keeping information contents protected from someone that had no permissions to see them. Confidentiality can be provided from physical protection to mathematical algorithms.
- ***Data integrity*** – provides protection of unauthorized data changing. To provide data integrity there must exists possibility of detecting unauthorized data manipulation (inserting, deleting or replacing of data).
- ***Authentication*** – Two sides of communication must identify itself to each other. Information that is carried over communication channel must be authenticated to original, original data, data contents, time of sending... This aspect of cryptography can split in two parts: entity authentication and data origin authentication. Data origin authentication implicitly provides data integrity
- ***Non-repudiation*** – prevents an entity of denying previous actions. To solve problem of denying, there must exist *Trusted Third Party.*

A fundamental goal of cryptography is to merge these four mention terms in theory and practice. Cryptography can be used something for prevention detection of cheating and other malicious activities.

[1] Alfred J. Menezes, Paul C. van Oorschot and Scott A. Vanstone; Handbook of Applied Cryptography; Chapter 1;Page 4.

In last 20 years cryptography was grown from art to science. Organizations such as *International Association for Cryptologic Research (IACR)* are becoming widely spread and their number is increasing.

Now basic terms in cryptography will explained as following:

- ***Plain text*** – message that will be encrypted,
- ***Transposition*** – characters of plain text are reordered, for example encryption of word SECRET into ETCRSE is called transposition,
- ***Substitution*** – characters of plain text are replacing with some other letters, numbers or symbols, for example word SECRET can be encrypted using substitution as 19 5 3 18 5 20 or XIWOXY,
- ***Cipher text*** – text that was get after appliance of cryptography algorithms on plain text,
- ***Passive attacker***– attacker that can reed only information from unsecured channel,
- ***Active attacker*** – attacker that can carry, insert or delete information from unsecured channel,
- ***Cryptanalysis*** – studying of mathematical techniques in cause of breaking cryptography techniques and services for information security,
- ***Cryptanalyst*** – someone who is using cryptanalysis,
- ***Cryptosystem***– set of cryptography tools used for providing information security.

3. Crypto algorithms

Crypto algorithms can be divided onto symmetric and asymmetric crypto algorithms.

3.1 Symmetric Algorithm

In group of symmetric cryptographic algorithms are algorithms where the same key is used for encryption and decryption (Figure 3.1). Algorithms in this group are also called private key algorithms, because the privacy of key used for encryption and decryption is essential for the security of messages in the system. These systems are based on traditional crypto logical theories. Given that the protection of information, mostly of the application, is in affairs related to state structures (army, police, diplomacy etc.), these systems were the exclusive secret systems, dedicated defined and implemented by the competent state institutions. With increasing intensity and implementation of electronic forms of communication, there was a need for defining public symmetric cryptographic algorithms.

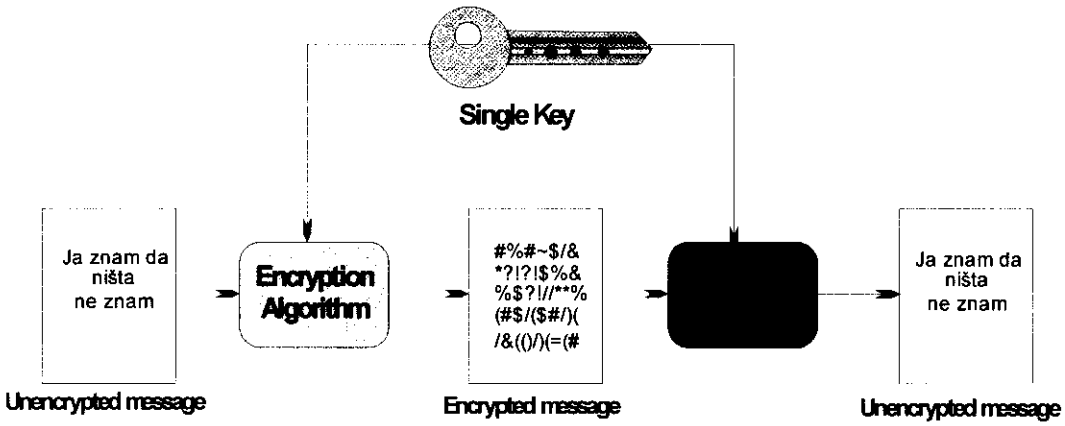

Figure 3.1 Symmetric cryptographic algorithm[2]

Symmetric cryptographic algorithms are generally used in applications related to business systems and financial communications. Given the explosive development of business and financial systems in recent times, public symmetric cryptographic algorithms have become dominant in terms of use. However, none of them adopted as a general standard, but mentioned systems generally use the appropriate list of possible cryptographic algorithms.

Although its number of commercial use of symmetric cryptographic algorithms use far exceeded the secret sector (related to state structure), the main theoretical results are still occurring in the field of cryptology secrets and classified systems. The vast majority of states have specialized organizations dealing with the design and analysis of various chipper systems (e.g. National Security Agency - NSA in the U.S.). Dubious achievements in this area usually are not publicly known and are in the field of assumptions.

[2] This figure was created by a model provided from http://gdp.globus.org/gt4-tutorial/multiplehtm/ch09s03.html

The best known symmetric cryptographic algorithms are AES, DES and 3DES.

3.2 Asymmetric Algorithm

In contrast to symmetric cryptographic algorithms the asymmetric cryptographic algorithms appeared, (Figure 3.2), known as public key algorithms. In this case, each entity has a public key and its corresponding private key. The public key is used for encryption, while the private key is used for decryption. If an entity B wants to send a message(m) to entity A, entity must generate a genuine copy of the entity's public key e by using encryption transformation to calculate encrypted text c = Ee (m) respectively c referring to A . To deciphered c, and applied decryption transformation to calculate the original message m = Dd (c). Public key, as his name suggests, should not be kept secret, can be seen anywhere, only its authentication is required in order to guarantee that A is the only party that knows the corresponding private key. The primary advantage of these algorithms is that the distribution of public keys is generally easier to perform the aspect of security than symmetric cryptographic algorithms. The main objectives of public-key encryption are to provide privacy or the confidentiality. As the transformation of public encryption, public key encryption does not provide data origin authentication or data integrity, but it must ensure the use of additional techniques such as message authentication codes and digital signatures.

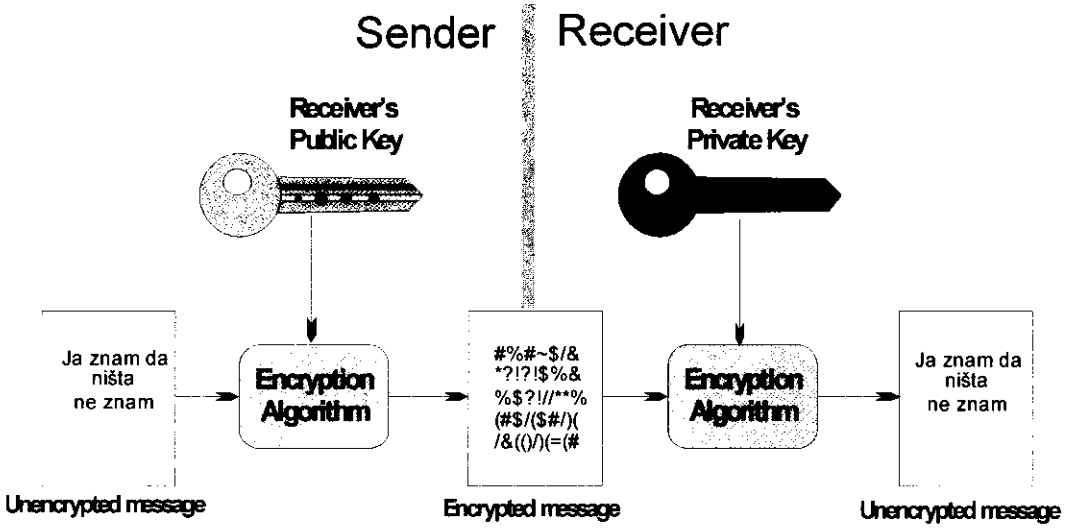

Figure 3.2 Asymmetric cryptographic algorithms[3]

Public key cryptographic algorithms are much slower than symmetric, such as DES. Therefore encryption public key is used mostly to transfer keys used for symmetric data encryption algorithms or other

[3] This figure was created by a model provided from http://gdp.globus.org/gt4-tutorial/multiplehtm/ch09s03.html

applications that include data integrity and authentication and encryption of small data, such as credit card numbers and PIN numbers.

3.2.1 RSA public key encryption

RSA cryptosystem, which is named after its inventors R. Rivest, A. Shamir, and L. Adleman, is the most widely spread public key cryptosystem. It can be used to ensure secrecy and digital signatures, and its security is based on the impossibility of factorization of large integers.

3.2.1.1 Description of the RSA algorithm

Key generation

Each entity creates an RSA public key and its corresponding private key. Each entity should do the following:

1. Generate two large, randomly selected and different areas of p and (bilo je i) q, roughly the same size.
2. Compute $n = PQ$, and $\varphi = (p - 1)(q - 1)$.
3. Select random integer e, $1 < e < \varphi$, such that the greatest common divisor $gcd(e, \varphi) = 1$
4. Use the extended Euclidean algorithm to calculate a unique integer d, $1 < d < \varphi$, such that $ed \equiv 1 \pmod{\varphi}$.
5. The public key of entity A is a **(n, e)** and private key of entity A is **d**.

Definition: whole numbers E and(o5 i) D calculated when generating key RSA encryption are called exponent and decryption exponent, respectively, while n is called module.

RSA algorithm

1. ***Encryption***. B should do the following:

(a) The calculation of an authentic public key (n, e) the entity A.
(b) Feature message as an integer m in the interval [0, n - 1].
(c) Calculate c as $c = m^e \bmod n$.
(d) Send encrypted text to the A.

2. ***Decryption***. To reconstructed plaintext m from c, A should do the following:

(a) Using the private key d to return $m = c^d \bmod n$.

Example (RSA encryption)

Generating key - Entity A selects primes p = 2357, q = 2551, and calculates n = PQ = 6012707 and ϕ = (p-1) (q-1) = 6007800. A chooses e = 3674911, and by using the extended Euclid's algorithm finds d = 422,191 so that ed ≡ 1 (mod ϕ). The public key of entity A is a pair (n = 6012707 e = 3674911), while the private key d = 422191.

Encryption - To encrypt message m = 5234673, B uses the algorithm for modular exponentiation to calculate c = me mod n = $5234673^{3674911}$ = 3650502 mod 6012707, and sends it to A.

Decryption - To decrypt C, calculates the c^d mod n = 3650502^{422191} mod 6012707 = 5234673.

4. The most known cryptographic algorithms that use hash functions

4.1 MD5

MD5 (Message-Digest Algorithm 5) is a cryptographic algorithm that belongs to a group of hash algorithms (these algorithms are also called digest, ireversible or algorithms without key) and is applied in many different areas of data protection, but is now considered vulnerable to cryptographic attacks, so that is rarely applied, but often find in application verifying the integrity of the major media because of its speed. The length of digest is 128 bits.

MD5 algorithm was developed in 1991. Based on the MD4 algorithm, and although it is slightly slower than MD4 algorithm, MD5 is more secure. Size of digest, and the number of additional bits added to the original message remain the same. For reasons that MD4 algorithm has been developed with the intent to be the fastest algorithm, the algorithm was located on the edge "in terms of risk of successful cryptanalysts attacks.

Digest length of this algorithm is 128 bits, in which many cryptanalysts object to this algorithm, so it is thought to be susceptible to brute force birthday attack. One such project under the name MD5CRK running 1st March 2004 with the intent to prove that this algorithm is not secure. Shortly thereafter, 17 August 2004 announced that Xiaoyun Wang, Denguo Feng, Ksuejia Lai and Ksongbo Ju (Xiaoyun Wang, Dengguo Feng, Xuejia Lai, Hongbo Yu) successfully broke the algorithm, that are found in the collision algorithm. For this algorithm, breaking it they need only one hour on an IBM p690 cluster. 1. March 2005. Arjen Lenstra, Xiaoyun Wang and Benne de Veger demonstrated the creation of two X.509 certificates with different public keys, but the same MD5. A few days later Vlastimil Klima has created enhanced algorithm that is able to ordinary laptop for several hours creates a collision of MD5 algorithm.

4.2 SHA-1

SHA-1, the translation safe hash algorithm (eng. Secure Hash Algorithm), with MD5 is currently the most popular algorithm for hashing. The main difference compared to the MD5 is that gives 160-bit hash value, and is considered safer. Algorithm itself is quite similar. The only differences are adding one more round, extending the transformation and add output of the previous steps in the next step to achieve a faster avalanche effect (avalanche effect). The effect of avalanche is one of the most important thing about hash functions. It reflects how many bits of the output is changed by changing only one bit at the entrance.

5. Digital Signature

Digital signatures are the realization of electronic signature, which is defined as a set of data in electronic form which is affiliated to or logically associated with other data in electronic form, which serve to identify the signer and the authenticity of the signed document. The best way to explain the purpose of the digital signature is compared with his personal signature. Holograph signature uniquely determines person, and assured that the document is signed with intent, ensuring the authenticity, personal signature proves that a person signed the document, which provides non-repudiation, and the integrity of the data comes from the fact that the signature cannot be recycled because it is part of the document itself, nor can be moved from one place to another (even though reality is quite far from that). The basic idea of digital signatures is that all these pass into the realm of digital communications and the possibility of eliminating the defects personal signature. Solution which ensures the realization of the basic idea is cryptography.

5.1 Appliance and Basic Terms

Digital signature is a message number that depends on a secret known only to one who signed the message, and, additionally, the content of the message was signed. Signatures must be provable if a dispute arose over whether the party signed a document (whether caused by false signatories who tries to deny the signature is made by him, or misrepresentation), an impartial third party should be able to resolve the matter fairly, without signer request for access to secret information (private key).

Digital signatures are a very large application in information security, including authentication, data integrity and non-repudiation. One of the most important applications of digital signatures is the certification of public keys in large networks. Under the certification means to connect the user with his public key by a third party of confidence, to be used later, other entities can authenticate public key without the help of the third stop of the trust.

The concept and use of digital signatures was visible for several years before it was even possible to realize. The first method was discovered RSA signature scheme, which today remains one of most practical and various techniques currently available. Subsequent research resulted in many alternative digital signature techniques, some of which offer significant advantages related to the very functionality and implementation.

Digital signature is a string of data that connects the message (in digital form) with some entity from whom comes. Generating a digital signature algorithm is a method for creating a digital signature. The algorithm for verification of digital signatures (or verification algorithm) is a method for verifying the authenticity of digital signatures. Digital signature scheme (or mechanism) consists of signed algorithm and related verification algorithm. The process of signing a digital signature (or procedure) consists of (mathematical) generating an algorithm, together with the method for formatting data in a message that can be signed. The process of verification of digital signatures (or procedure) consists of verification algorithm, together with the method of recovering data from the message.

5.2 Algorithms for Creating a Digital Signature

RSA algorithm is described in detail in chapter 3. Below is explained DSA, and the manner in which both the algorithm used in generating digital signatures.

Algorithm: Generation of signature and verification using RSA

The entity A signs a message m from M. Any entity B can verify the signature of a hash function and restore if you know m.

1. *Signature generation and verification.* The entity should calculate the following:

   (a) Compute $\tilde{m} = R(m)$, as an integer in the range [0, n - 1].

   (b) Compute $s = \tilde{m}^d \bmod n$.

   (c) Signature of Entity A message m is s.

2. *Verification.* In order to verify the list are reconstructed message m Entity A Entity B must:

   (a) to obtain the authentic public key (n, e) the entity A.

   (b) calculate the $\tilde{m} = s^e \bmod n$.

   (c) Check if $\tilde{m} \in M_R$, if not, then rejects the signature.

   (d) Reconstruct the $m = R^{-1}(\tilde{m})$.

Example (RSA signature generation with very few parameters)

Generating key - Entity A selects primes p = 7927, q = 6997, and account n = PQ = 55,465,219 and ϕ = 7926 × 6996 = 55450296. Choose e = 5 and accounts 5d = ed ≡ 1 (mod 55450296), from which we get that d = 44,360,237th The public key of entity A in this case (n = 55,465,219, e = 5), and private d = 44,360,237.

Signature generation - For simplicity we assume that M = Zn, and that is a function R: M → Zn identity map R (m) = m for every m ∈ M. To sign a message m = 31229978, a billing entity $\tilde{m} = R(m) =$ 31229978, and then the signature $s = \tilde{m}^d \bmod n = 31229978^{44360237} \bmod 55465219 = 30729435$.

Verification of signatures - B calculates $\tilde{m} = s^e \bmod n = 30729435^5 \bmod 55465219 = 31229978$. Finally, B accepts the signature because $\tilde{m} \in M_R$, and reconstruct $m = R^{-1}(\tilde{m}) = 31229978$.

Digital Signature Algorithm (DSA)

In August 1991, the American National Institute of Standards and Technology - U.S. National Institute of Standards and Technology (NIST) has proposed a digital signature algorithm (DSA). DSA became a United States federal standard for the processing of information - U.S. Federal Information Processing Standard (FIPS 186) called a standard digital signature - Digital Signature Standard (DSS) and the first digital signature scheme recognized by any government. The algorithm is a version of ElGamal's scheme and belongs to the scheme with the addition of a digital inventory. For the mechanism of signature is required hash function h: (0, 1) * → Z_Q for some integer q. DSS explicitly requires the use algorithm Secure Hash (SHA-1), which is described in Chapter 4.

The algorithm of generating DSA key

Each entity creates a public key and its corresponding private key. Each entity A should do the following:

1. Choose a prime number q such that $2^{159} < q < 2^{160}$.

2. Choose t so that 0 ≤ t ≤ 8, and then the prime number p, which meets in $2^{511+64t} < p < 2^{512+64t}$, with a feature that the (p - 1) divisible with q.

3. (Select a generator α for the unique cyclic group q u Z_{*p}.)

3.1 Choose an element $g \in Z_{*p}$ and calculated $\alpha = g^{(p-1)/q} \bmod p$.

3.2 If α = 1, then returns to step 3.1.
4. Select a random integer a such that $1 \leq a \leq q - 1$

5. Calculate the $y = \alpha^a \bmod p$..

6. The public key of entity A **(p, q, α, y)**, a private is **a**.

Algorithm DSA signature generation and verification

The entity A signs a binary message m of arbitrary length. Any entity B can verify this signature using public key entity A.

1. *Signature generation.* Entity A should do the following:
(a) Select a random integer k such that 0 <k <q.

(b) compute $r = (\alpha^k \bmod p) \bmod q$.

(c) compute $k^{-1} \bmod q$.

(d) compute $s = k^{-1}\{h(m) + ar\} \bmod q$.

(e) The signature of entity A message m is a pair **(r, s)**.

2. *Verification.* In order to verify the signature (r, s), entities of a message m, B should do the following:

(a) Obtain an authentic public key of entity A - (p, q, α, y).
(b) Verify that 0 <r <qi 0 <s <q and if not, then rejects the signature.

(c) Compute $w = s^{-1} \bmod q$ *i* $h(m)$.

(d) Calculate $u1 = w \cdot h(m) \bmod q$ i $u2 = rw \bmod q$.

(e) Compute $v = (\alpha^{u1} y^{u2} \bmod p) \bmod q$.

(f) Accept the signature if and only if v = r.

Example (DSA signature generation with very few parameters)

Generating key - Selects a prime p = 124540019 i q = 17389 such that (p -1) divisible with q, so (p - 1) / q = 7162. And then selects random integer g = 110217528 $\in Z_{*p}$ and calculated $\alpha = g^{7162} \bmod p =$ 10,083,255. As $\alpha \neq 1$, α is the generator of the unique cyclic group Z_{*p}. And then randomly selects an integer a = 12,496 such that $1 \leq a \leq q - 1$, and calculated $y = \alpha^a \bmod p = 10083255^{12496} \bmod 124540019 =$ 119946265.

Entity public key is (p = 124,540,019, q = 17,389, α = 10,083,255, y = 119946265), and a private = 12496.

Signature generation - To sign message m, A chooses a random integer k = 9557, and accounts $r = (\alpha^k \bmod p) \bmod q = (10083255^{9557} \bmod 124540019) \bmod 17389 = 27039929 \bmod 17389 = 34$. And then accounts $k^{-1} \bmod q = 7631$, h (m) = 5246, and eventually obtain s = (7631) (5246 (12496) (34)) mod q = 13049th

The signature for m is a pair (r = 34, s = 13049).

Verification of signatures - B calculate $w = s^{-1} \bmod q = 1799$, $u1 = w \cdot h(m) \bmod q =$ (5246)(1799) mod 17389 = 12716, and $u2 = rw \bmod q =$ (34)(1799) mod 17389 = 8999. B then calculates $v = (\alpha^{u1} y^{u2} \bmod p) \bmod q =$ (10083255^{12716} · 119946265^{8999} mod 124540019) mod 17389 = 27039929 mod 17389 = 34. As v = r, B accepts the signature.

Note: (recommended size parameters) q should be the size of 160 bits, while the size of p can be any multiple of 64 between 512 and 1024 bits. 512-bit prime number p provides marginal security against specific attacks. Since 1996 it is recommended that modules be at least 768 bits. FIPS 186 does not allow primes p larger than 1024 bits.

5.3 Digital Certificate. Example of Creating a Digital Certificate

"In cryptography, a digital certificate (also known as a public key certificate or identity certificate) is an electronic document which uses a digital signature to bind together a public key with an identity — information such as the name of a person or an organization, their address, and so forth. The certificate can be used to verify that a public key belongs to an individual.

In a typical public key infrastructure (PKI) scheme, the signature will be of a certificate authority (CA). In a web of trust scheme, the signature is of either the user (a self-signed certificate) or other users ("endorsements"). In either case, the signatures on a certificate are attestations by the certificate signer that the identity information and the public key belong together.

For provable security this relies on something external to the system has the consequence that any public key certification scheme has to rely on some special setup assumption, such as the existence of a certificate authority"[4].

Because we were studying Java, I've used Java's tool for generating digital certificate to make a simple digital certificate for me. In following sentences I will describe how can we generate a digital certificate using a *key tool*.

Generating digital certificate using *keytool* is very simple, all we need is to start Command Prompt and enter following command (as it is shown on Figure 5.1) – `keytool -genkey -v -alias GvozdenovKljuc -keyalg RSA`, which generate a digital certificate using RSA cryptoalgorithm described in Chapter 3. After entering that command, we'll be asked for some personal information

[4] http://en.wikipedia.org/wiki/Public_key_certificate

about person that will use this certificate (in this case, I made certificate for myself). When it is all done, we can check is there a certificate, using command - `keytool -list -v -alias GvozdenovKljuc` (as it is shown on Figure 5.2). And finaly we can export it in .cer file, that can be used on Windows operating systems - using command - `keytool -export - keystore .keystore -alias GvozdenovKljuc -file Gvozden_Micic.cer` (as it is shown on Figure 5.3). In Figure 5.4 is shown that digital certificate.

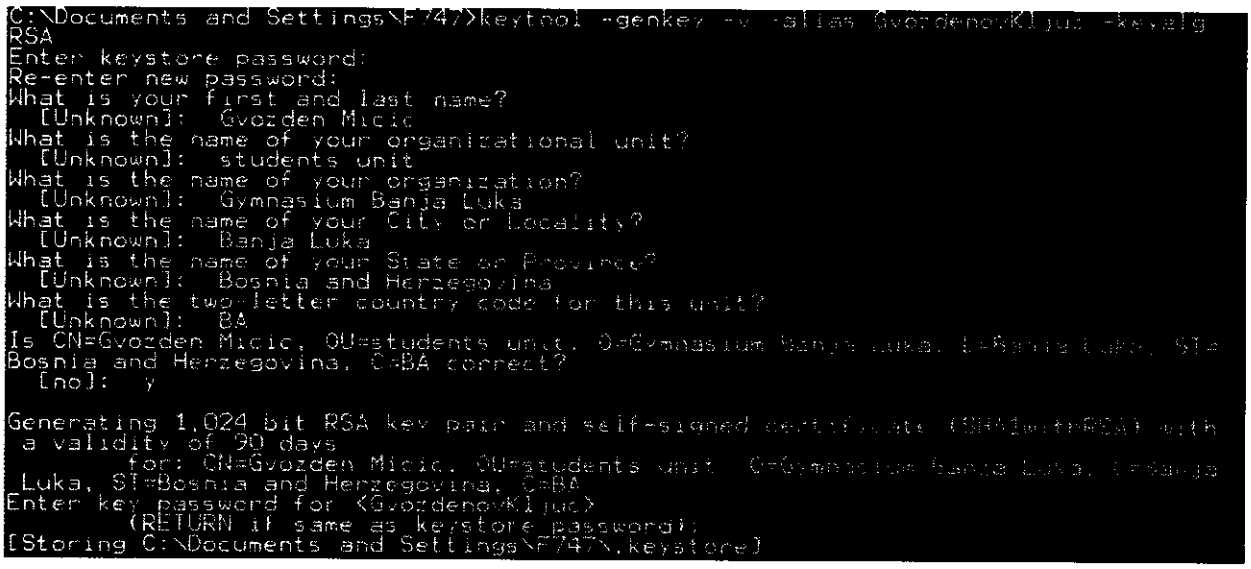

Figure 5.1 generating a digital certificate

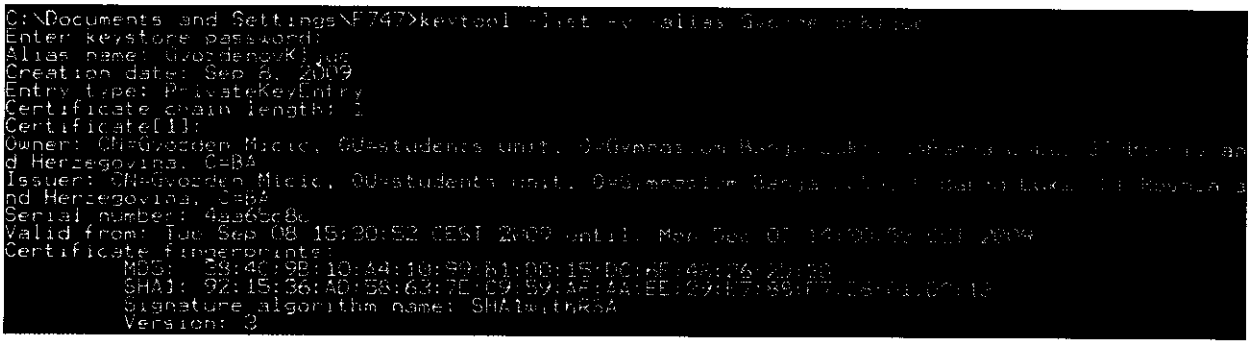

Figure 5.2 checking a digital certificate

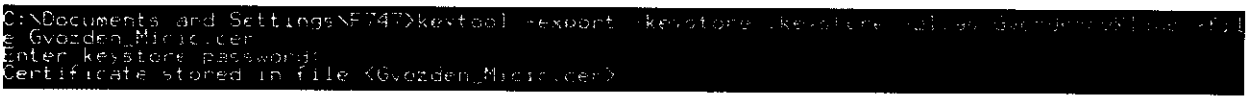

Figure 5.3 exporting a digital certificate in .cer file

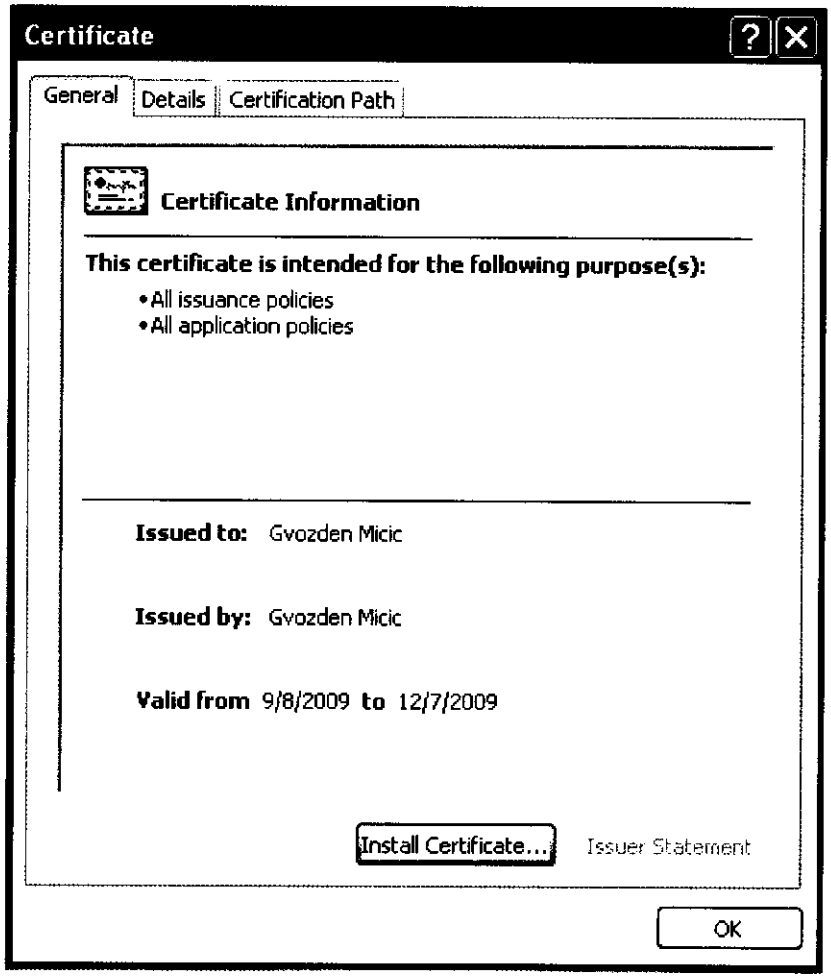

Figure 5.4 Digital certificate

6. Conclusion

Cryptography has a lot of appliance in computing – starts with using a crypto algorithms (such as MD5 and SHA-1) for data crypting when they are storing in some database (usually data for identification – for example: passwords, PIN codes or something) or logging data for connecting on network device, and going so far to digital signature like equivalent of self-signature and digital certificate as personal identity card in electronic world.

Usage of cryptography in computing allows us to get more security for data, keeping them safer from unsolicited attacks. The most logical way is to make a system that use good sides of both – symmetric and asymmetric crypto algorithms, asymmetric for authentication, integrity and non-repudiation, and symmetric for privacy.

Due to this matter this essay gave some basics of cryptography, describing how it works and where its appliance is the most important. As well technologies that are most popular, and have great functioning in area of identification in digital world: digital signature and digital certificate are described in order to understand the rising impact and appliance of such methods in world of safety and protection of our data, given that digital signature will, if not already, replace the ordinary hand made signatures.

7. Bibliography

- Alfred J. Menezes, Paul C. van Oorschot and Scott A. Vanstone - *Handbook of Applied Cryptography*, CRC Press, 1996.
- David Kahn – *The Codebreakers, The Story of Secret Writing*, Macmillian, 1967.
- Whitfield Diffie and Martin E. Hellman – *New Directions in Cryptography*, 1976.
- Goran Vojković – *Elektronički potpis*, journal Mreža, Zagreb, Jun 2006.
- http://en.wikipedia.org/wiki/Public_key_certificate (accesed 02.09.2009.).
- http://gdp.globus.org/gt4-tutorial/multiplehtm/ch09s03.html (accesed 02.09.2009.).