International Journal of Electronics and Communications System
Volume 1, Issue 1, 11-15.
ISSN: 2798-2610
http://ejournal.radenintan.ac.id/index.php/IJECS/index

Data integrity and security of digital signatures on electronic systems using the digital signature algorithm (DSA)

Asep Saepulrohman*
Universitas Pakuan, Bogor, INDONESIA

Agus Ismangil
Universitas Pakuan, Bogor, INDONESIA

Article Info

Article history:

Received: February 18, 2021
Revised: April 12, 2021
Accepted: May 9, 2021

Keywords:

Data integrity,
DSA,
ElGamal,
hash algorithm,
signature

Abstract

The digital signature generation process begins with the creation of a public key and a private key. A public key is generated and published to verify the signature and calculate the hash value of the received document. At present, in the very fast development of information technology, quantum computers have emerged the ability to solve very large and complex amounts of data calculated by qubits, which when compared to quantum computers can work 10 minutes to work on a process that takes 1025 years on a computer. Therefore, the research focuses on how electronic signatures on documents have a reliable security system. The Digital Signature Algorithm (DSA) is a key algorithm used for digital signatures, which uses the Secure Hash Algorithm (SHA-1) to convert messages into message digest and parameters based on the ElGamal signature algorithm. The author also shows an example of digital signature encryption and decryption process by taking any numbers p = 59419 and q = 3301 to prove that the message can be formed and verified its authenticity.

To cite this article: A. Saepulrohman, and A. Ismangil, "Data integrity and security of digital signatures on electronic systems using the digital signature algorithm (DSA)," Int. J. Electron. Commun. Syst., vol. 1, no. 1, pp. 1–9, Jun. 2021

INTRODUCTION

Data is confidential and remains secure when sent on digital channels, is often known to unauthorized persons. This is what motivates the researchers to think of a reliable algorithm so that the information remains safe. Several algorithms have been developed and one of which has been discovered is the Shor algorithm which was invented by Peter Shor in 1995. Through this Shor algorithm, a quantum computer can decode a secret code that is currently commonly used to secure data transmission. This code is called an RSA code. RSA (Rivest–Shamir–Adleman) is a public-key cryptosystem that is widely used for secure data transmission. If encoded via RSA code, the data transmitted will be safe because the RSA code cannot be cracked in a short time. In addition, solving the RSA code requires the work of thousands of computers in parallel. So, this solving work is not effective. In this study, there is various literature that forms the basis of the author's theory in conducting research, the literature consists of various scientific books and papers which according to research. The following are some of the main literature on which the theory is based, namely: the application of digital signatures in government electronic systems to support e-government [1], a post-quantum digital signature scheme based on Supersingular Isogenies [2], an efficient signature scheme of Isogenies of Supersingular Elliptic Curves [3], and mathematical concepts and models based on Isogen-Based Cryptographic Mathematics [4]and data integrity and security using keccak and Digital Signature Algorithm (DSA) [5] and elliptic curve Diffie-Hellman cryptosystem for public exchange process [6], [7].

Our Contribution. In this paper, the use of information technology includes activities related to the development of cryptographic mathematical concepts and models. DSA is public-key cryptography that is used for authentication, data security, and anti-deny tools [8], the parameters used are dynamic public key and private key parameters, which have different values for each digital signature creation process [9]. The DSA algorithm is used for message signing, the SHA function is

• **Corresponding author:**
Asep Saepulrohman, Universitas Pakuan, Bogor, INDONESIA. ✉ asepspl@unpak.ac.id

also involved as a message digest generator of messages [10]. A process for optimizing digital signature hash function files, then management systems and work processes electronically, and public services so that documents can be easily and cheaply verified.

Digital Signature Algorithm (DSA)

A signature on electronic data is called a digital signature, but a digital signature is not a sign that is digitized by a scanner or electronic pen. A digital signature is a mathematical scheme that relies on the message content and the sender of the message to prove the authenticity of the message or document. This scheme is carried out as a guarantee that the data and information come from the correct source [11]. A digital signature consists of a series of has functions that are generated from a specific hash function algorithm which is then encoded (encrypted) with an asymmetric key cryptographic algorithm. To verify this, the algorithm's public key is used.

DSA is a standard for digital signatures that was formalized in August 1991 by NIST (The National Institute of Standards and Technology) which consists of two main components, namely: the digital signature algorithm and the Secure Hash Algorithm (SHA) [12].

DSA is included in the public key cryptographic algorithm, and not used for encryption. It means more specifically for digital signatures. DSA has two main functions, namely the signature generation and signature validity checks developed from the ElGamal algorithm. DSA uses two keys, namely the public key and the private key. The formation of signatures uses the private key, while to verify a digital signature we can use the public key which can be illustrated in Figure 1 [13].

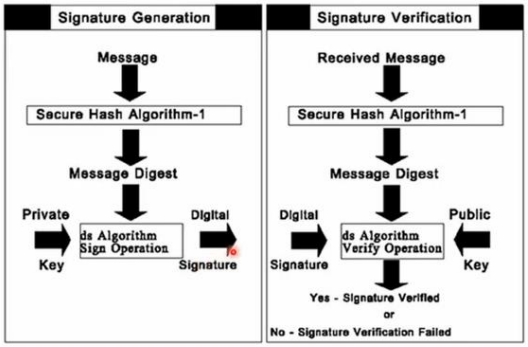

Figure 1. Generate and verification of a digital signature algorithm.

Secure Hash Algorithm (SHA)

SHA is a one-way hash function (a one-way hash) created by NIST and used in conjunction with DSA which is the standard for digital signature generation. The National Security Agency (NSA) states that SHA is a standard one-way that hash function which is a continuation of its predecessor, namely MD4 made by Ronald L. Rivest from MIT, who had found the collision and developed MD5, in other words, SHA is safe because it is designed in such a way computationally. You may find messages that correspond to the message digest.

The SHA algorithm accepts 2^6464 bit (2,147,483,648 gigabytes) input by producing a message digest that is 160 bits in length. SHA has six variants that refer to a family of one-way hash functions, namely SHA-0, SHA-1, SHA-224, SHA-256, SHA-384, SHA-512. The numbers behind the SHA indicate the length of the digest messages. In this research, only SHA-1 will be discussed. Following Table 1 the sizes in bits for the SHA family.

Table 1. The digest SHA message sizes

Name	Size (bit)			
	Output	Internal state	Block	Max message
SHA-0				
SHA-1	160	160	512	$2^{64}-1$
SHA-256/224	256/224	256	512	$2^{64}-1$
SHA-512/384	512/384	512	1024	$2^{128}-1$

The steps for making a message digest with SHA-1 are generally similar to MD-5, which are as follows: First add padding bits, then increase the value of the original message length and initialize the MD buffer, and processing messages in blocks of 512 size.

METHOD

The public key cryptographic algorithm for DSA is the creation of digital signatures using the private key and for verification using the public key [14], [15]. DSA uses the hash SHA function to convert a message into a message digest that is 160 bits in size. Algorithm Digital Signature Parameters:

1. Public key p, q, g with
   a. p is a prime number 512 to 1024 and is in the form of 64 blinks;
   b. q prime numbers are 160 bits and are a factor of$p-1$;

c. $g = h^{(p-1)/q} \text{mod } p$ with $1 < h < p - 1$;
d. $y = g^x \text{ mod } p$

2. The private key x which is an integer less than q;
3. Message m which will be signed.

Before generating digital signatures, the user must have a key first, namely a private key and a public key. The procedure for generating keys in a digital signature is as follows:

Algorithm Key generation for the DSA.
This procedure will generate the public key $(p, q, g, y)$and the private key x by:

1. Select p and q with $(p-1) \text{ mod } q = 0$.
2. Compute $g = h^{(p-1)/q} \text{mod } p$ which is greater than 1.
3. Specify private key x with $x < q$
4. Compute public key $y = g^x \text{ mod } p$.

Algorithm DSA signature generation

1. Compute the message digest of m messages with a hash function SHA-1, $H(m)$.
2. Compute a random secret integer $k < q$.
3. Signature of the message m is r and s, compute:
   $r = (g^k \text{mod } p) \text{ mod } q$
   $s = (k^{-1}(H(m) + x.r)) \text{ mod } q$
4. Sent message m along with the signature r and s.

Algorithm DSA signature verification

1. Compute the message digest of m messages with a hash function SHA-1, $H(m)$.
2. Verify that digital signature $0 < r < q$ and $0 < s < q$; if not, then reject the signature and compute:
   $w = s^{-1} \text{mod } q$
   $u_1 = (H(m).w) \text{ mod } q$
   $u_2 = (r.w) \text{ mod } q$
   $v = ((g^{u1}.y^{u2}) \text{ mod } p) \text{ mod } q$
3. If $v = r$, then the signature verified.

The process of encrypting and decrypting messages using the digital signature algorithm with the help of the program on the portal https://asecuritysite.com/encryption/md5

RESULTS AND DISCUSSION

The value of p, q are prime numbers that have a length limit of 512 to 1024 bits. The q is 160 bits, making DSA almost impossible to solve by software in other words difficult to solve. Therefore, DSA is implemented in a software device, where the bit length limits p and q are changed to a maximum value of p and q is 2^{32}.

DSA based on the ElGamal algorithm has the following parameter properties:

1. p is a prime number with length L bits, in this case, $512 \leq L \leq 1024$ and L must be a multiple of 64 bits and is public.
2. q 160-bit prime number is a factor of $(p-1)$, in other words $(p-1)$ mod q = 0, is public.
3. g parameter is public with $q = h^{(p-1)/q} \; mod \; p$ with $h < (p-1)$.
4. x is a number less than q and is private
5. m is the message to be signed
6. y is the public key with $y = g^x \; mod \; p$.
7. r, s is a signature, where the message is always accompanied by a signature, written (m, s).
8. v, w a compute procedure to verify a signature if $v = r$ then a signature is valid.

Examples of message characteristics change processes for MD5, SHA-1, and SHA-2 (SHA-224, SHA-256, SHA-384, SHA-512) which generates a hash signature, and the output is usually displayed in hex or Base-64 format. For example, the message is given as follows.

Message:

> Nearly nine months after the pandemic, Indonesia finally welcomed the first shipment of the COVID-19 vaccine produced by China Sinovac Biotech at Soekarno-Hatta International Airport in Tangerang, Banten, on December 6. The first batch contained 1.2 million doses of vaccine. This is not an all-around cure solution for a country of 270 million people. Even so, the arrival of the first wave marks an important step in the country's efforts to develop a vaccine for COVID-19.

Generate hash:

MD5	780AD7A4AF9E608CDC58710960964FFA
SHA-1	E17FF7B6320F7036B8AFF6A8DDAEBFA66D0F3FB2
SHA-256	9E04339AA01D0AF7061C39A83D709235B7DA341901CEDC734FDD53AD7D9EB792
SHA-382	F4BD69D781381A0C353F7CA5D1A006BBB9EA182B2EB8942140218C467303D34ED6863ADE7023F2F4D76FAEC562C36914
SHA-512	F80D7D21175D5A7662539BEF77A8AFABDDA0F284F931BCE2D3EF360E9D673A4794C1F636053DA7C82EB6E5BF974E9E31F5811A1DAD8C57B0E9969E0AA614A195

Next, we will generate a key pair in DSA, first select the prime numbers p and q, which in this case $(p-1) \bmod q = 0$. For example $p = 59419$ and $q = 3301$ this satisfies (59419-1) mod 3301 = 0. Then calculate $g = h^{(p-1)/q} \bmod p$ in this case $1 < h < p-1$, for example, h = 100, then $g = 100^{(59419-1)/3301} \bmod 59419 = 18870$. Find the private key x that satisfies $x < q$, for example, $x = 3223$ then compute the public key $y = g^x \bmod p$ or $y = 18870^{3223} \bmod 59419 = 2945$.

The next procedure is to generate a digital signature by calculating the hash value of the message m, for example, $H(m) = 4321$. Find a random number $k < q$, for example, $k = 997$ and $k^{-1} \equiv 2907 \pmod{3301}$ with its public parameters $p = 59419$ and $q = 3301, g = 18870$ and private parameters $x = 3223$. Compute digital signature, r and s as follows:

$$\begin{aligned} r &= (g^k \bmod p) \bmod q \\ &= (18870^{997} \bmod 59419) \bmod 3301 \\ &= 848 \\ s &= (k^{-1}(H(m) + x.r)) \bmod q \\ &= (2907(4321 + 3223.848)) \bmod 3301 \\ &= 7957694475 \bmod 3301 \\ &= 183 \end{aligned}$$

Send message m and signature (r, s) namely (848,183).

The next procedure is to verify the validity of the digital signature. First, calculate the value of the message m, for example, H (m) 4321, then sign $(r, s) = (848,183)$ as follows:

$$\begin{aligned} s^{-1} &\equiv 469 \pmod{3301} \\ w &= s^{-1} \bmod q = 469 \bmod 3301 = 469 \\ u_1 &= (H(m).w) \bmod q \\ &= (4321.469)\ 2026549 \bmod 3301 \\ &= 3036 \\ u_2 &= (r.w) \bmod q \\ &= (848.469) \bmod 3301 \\ &= 397712 \bmod 3301 \\ &= 1592 \\ v &= ((g^{u1}.y^{u2}) \bmod p) \bmod q \\ &= (18870^{3086}.29245^{1592}) \bmod 3301 \\ &= 3036\ 848 \bmod 3301 \\ &= 848. \end{aligned}$$

Since $v = r$ the signature is valid, the signature is verified to prove whether the message is genuine or not by the steps as outlined above; first, the digital signature (r, s) is decrypted using the sender's public key, producing the original message digest. Second, the receiver then converts message m into message digest by using the DSA algorithm, which is the same as the hash function used by the sender. Finally, if the signature is received it is genuine and comes from the correct sender.

CONCLUSION

In this work, we introduce a digital signature data security system with the ElGamal algorithm applicable to the Digital Signature Algorithm (DSA). The built scheme has rich dynamics as confirmed by the software implementing the DSA key exchange algorithm and the encryption-decryption algorithm has been successfully built. The author also shows an example of digital signature encryption and decryption process by taking any numbers p = 59419 and q = 3301 to prove that the message can be formed and verified its authenticity.

REFERENCES

[1] A. Nugraha and A. Mahardika, "Penerapan tanda tangan elektronik pada sistemelektronik pemerintahan guna mendukung e-government," in *Seminar Nasional Sistem Informasi Indonesia*, 2016.

[2] Y. Yoo, R. Azarderakhsh, A. Jalali, D. Jao, and V. Soukharev, "A post-quantum digital signature scheme based on supersingular isogenies," in *Lecture Notes in Computer Science (including subseries Lecture Notes in Artificial Intelligence and Lecture Notes in Bioinformatics)*, 2017, vol. 10322 LNCS, pp. 163–181, doi: 10.1007/978-3-319-70972-7_9.

[3] Y. Huang, F. Zhang, Z. Liu, and H. Zhang, "An efficient signature scheme from supersingular elliptic curve isogenies," *IEEE Access*, vol. 7, no. 1, pp. 129834–129847, 2019, doi: 10.1109/ACCESS.2019.2938682.

[4] L. De Feo, "Mathematics of Isogeny Based Cryptography," Thiès, Nov. 2017.

[5] M. A. Nazal, R. Pulungan, and M. Riasetiawan, "Data integrity and security using keccak and digital signature algorithm (DSA)," *IJCCS (Indonesian J. Comput. Cybern. Syst.*, vol. 13, no. 3, p. 273, Jul. 2019, doi: 10.22146/ijccs.47267.

[6] A. Saepulrohman, A. Denih, and A. Talib Bon, "Elliptic curve Diffie-Hellman cryptosystem for public exchange process," in *The 5th NA International Conference on Industrial Engineering and Operations Management*, 2020, pp. 1–6.

[7] A. Saepulrohman and T. P. Negara, "Implementation of elliptic curve diffie-hellman (ECDH) for encoding messeges becomes a point on the GF(í µí²í µí²) 1*," *Int. J. Adv. Sci. Technol.*, vol. 29, no. 6, pp. 3264–3273, May 2020.

[8] M. B. Yassein, S. Aljawarneh, E. Qawasmeh, W. Mardini, and Y. Khamayseh, "Comprehensive study of symmetric key and asymmetric key encryption algorithms," in *Proceedings of 2017 International Conference on Engineering and Technology, ICET 2017*, 2018, vol. 2018-January, pp. 1–7, doi: 10.1109/ICEngTechnol.2017.8308215.

[9] F. Nurhasanah and R. Sulaiman, "Pembuatan tanda tangan digital menggunakan digital signature algorithm," *Mathunesa J. Ilm. Mat.*, vol. 2, no. 2, pp. 1–7, May 2013.

[10] B. R. Pajčin and P. N. Ivanis, "Analysis of software realized DSA algorithm for digital signature," *Electronics*, vol. 15, no. 2, pp. 73–78, 2011.

[11] A. Supriyanto, "Pemakaian kriptografi kunci publik untuk proses enkripsi dan tandatangan digital pada dokumen e-mail," *Din. Inform.*, vol. 1, no. 1, pp. 14–19, 2019.

[12] A. Mali, C. Mahalle, M. Kulkarni, T. Nangude, and P. G. Navale, "Digital signature authentication and verification on smartphones using CRPT algorithm," *Int. Res. J. Eng. Technol.*, vol. 4, no. 5, pp. 332–338, 2017.

[13] D. K. Black, "The digital signature standard: Overview and current status," *Comput. Secur.*, vol. 12, no. 5, pp. 437–446, Aug. 1993, doi: 10.1016/0167-4048(93)90062-A.

[14] S. Kazmirchuk, A. Ilyenko, S. Ilyenko, O. Prokopenko, and Y. Mazur, "The Improvement of digital signature algorithm based on elliptic curve cryptography," in *Advances in Intelligent Systems and Computing*, 2021, vol. 1247 AISC, pp. 327–337, doi: 10.1007/978-3-030-55506-1_30.

[15] A. Sajjad, M. Afzal, M. M. W. Iqbal, H. Abbas, R. Latif, and R. A. Raza, "Kleptographic attack on elliptic curve based cryptographic protocols," *IEEE Access*, vol. 8, pp. 139903–139917, 2020, doi: 10.1109/ACCESS.2020.3012823.