VOL. 13, NO. 11, JUNE 2018 ISSN 1819-6608
ARPN Journal of Engineering and Applied Sciences

www.arpnjournals.com

ENHANCED SECURITY FOR DATA TRANSACTION WITH PUBLIC KEY SCHNORR AUTHENTICATION AND DIGITAL SIGNATURE PROTOCOL

M. Mesran[1], Muhammad Syahrizal[1] and Robbi Rahim[2]
[1]Department of Informatics Engineering, STMIK Budi Darma, Medan, Indonesia
[2]School of Computer and Communication Engineering, Universiti Malaysia Perlis, Malaysia
E-Mail: usurobbi85@zoho.com

ABSTRACT

Authentication and digital signatures need to be done to identify each other in communication, the Schnorr scheme algorithm is an algorithm that can be used for authentication and digital signatures, this paper provides an understanding of how authentication and digital signatures work to make it easier for readers to know the application Schnorr algorithm on information security process, and the result show the message was more secure from any attacker.

Keywords: authentication and digital signatures, secure communication, Schnorr Scheme, application schnorr.

INTRODUCTION

Authentication is an identification by each communicating party, meaning that some communicating parties must identify each other (Wang & Song, 2016) (Chakraborty, Rahman, & Rahman, 2016) (Rahim, Man-In-The-Middle-Attack Prevention Using Interlock Protocol Method, 2017). Information obtained by a party from another party should be identified to ensure the authenticity of the information received (Chakraborty, Rahman, & Rahman, 2016). Identification of information can be the date of making information, information content, time of delivery and other matters relating to that information. Message authentication is successful in protecting both parties from exchanging messages from third parties (Sravanthi & Prasad, 2011). However, message authentication cannot prevent the possibility of both sides attacking each other. In situations where there is no complete trust between the sender and the recipient of the message, a mechanism is required rather than authentication (Wang & Song, 2016) (Sravanthi & Prasad, 2011). The most interesting solution to this problem is the digital signature (Wang & Song, 2016) (Sravanthi & Prasad, 2011) (Shiralkar & S. Vijayaraman, 2033). A digital signature is an authentication mechanism that allows the message maker to add code that acts as its signature. The signature ensures the integrity and source of a message (K. Doke & M Patil, 2012).

Claus Schnorr's authentication and digital signature scheme take securities from the problem of calculating discrete logarithms (Chakraborty, Rahman, & Rahman, 2016). This scheme uses prime numbers and modulo in the process of forming the key. The authentication scheme can modify into a digital signage scheme. The process of establishing the private and public keys is the same as the authentication scheme, only in the digital signature scheme added a hash function (Chakraborty, Rahman, & Rahman, 2016) (K. Doke & M Patil, 2012).

THEORY

Information security

Information Security is a protection against information when the information is sent from one system to another (Hariyanto & Rahim, 2016) (Legito & Rahim, 2017) (Nofriansyah & Rahim, 2016) (Rahim & Ikhwan, Cryptography Technique with Modular Multiplication Block Cipher and Playfair Cipher, 2016) (Rahim & Ikhwan, Study of Three Pass Protocol on Data Security, 2016) (Siahaan & Rahim, 2016). The information security system has four very basic goals, such as:

a) Confidentiality

Guarantee, whether the information sent, cannot be opened or cannot be known by others who are not eligible. For sensitive data, a high degree of confidentiality is required, which is only accessible to certain parties (the rightful party).

b) Integrity

Ensure the integrity and authenticity of data, so that the efforts of irresponsible parties to duplicate and destroy data can be avoided.

c) Availability

Ensures legitimate users to access their information and resources. The aim is to ensure that the proper parties are not rejected access to the information that they are entitled.

d) Legitimate use

Ensure certainty that the source not used or information not accessed by untrustworthy parties (unauthorized parties).

Authentication

Authentication (Braz & Robert, 2006) (Guidorizzi, 2013) is a term applied in a broad sense. The word implicitly means more than conveying the idea that the tool has provided assurance that unauthorized parties are not manipulating information. Authentication is unique to the security topic; examples include access control, entity authentication, message authentication, data integrity, non-repudiation, and key authentication (Wang & Song, 2016)

(Chakraborty, Rahman, & Rahman, 2016) (Sravanthi & Prasad, 2011) (Guidorizzi, 2013).

Authentication is one of the most important things in information security. Until the mid-1970s, it remained thought that secrecy and authentication were closely connected. With the discovery of hash and digital signatures (Rahim, Dahria, Syahril, & Anwar, 2017), it continues recognized that confidentiality and authentication are separate and independent issues (K. Nair, Navin K.S, & Chandra C.S, 2015). At first, it did not seem important to separate them, 6but there were situations where they were not only useful but necessar (Shiralkar & S. Vijayaraman, 2033) (Haddaji, Ouni, Bouaziz, & Mtibaa, 2016) (Sadikin & Wardhani, 2016), the formulas used in the authentication process are as follows:

a) r (r < q)
b) z = a^r mod p
c) n = (r + se) mod q
d) z = ((a^y).(v^e)) mod p

Digital signature

A digital signature is an authentication mechanism that allows the message maker to add code that acts as its signature. The signature ensures the integrity and source of a message (Sravanthi & Prasad, 2011) (Mali, Mahalle, Kulkarni, Nangude, & Navale, 2017).

The digital signing of a document is the fingerprint of the document, and its timestamp is encrypted using the private key of the signing party (Ratha, Connell, & Bolle, 2001). The digital signature utilizes a one-way hash function to ensure that the signature applies only to the concerned document (Wang & Song, 2016) (Chakraborty, Rahman, & Rahman, 2016) (Rahim, Dahria, Syahril, & Anwar, 2017). The receiving party can check the validity of the digital signature.

Schnorr algorithm

Claus Schnorr Authentication and Digital Signature scheme take securities from the problem of calculating discrete logarithms (Xin, Wang, Shao, Wang, & Zhang, 2015) (Cao & Markowitch, 2009). This scheme also uses prime numbers and modulo in its key forming process. The difficulty level for solving this algorithm is about 2t, where this t value can be determined by itself (Cao & Markowitch, 2009).

The authentication scheme can be changed into a digital signage scheme. The process of establishing the private and public keys is the same as the authentication scheme, only in the digital signature scheme added a hash function (Xin, Wang, Shao, Wang, & Zhang, 2015).

RESULT AND DISCUSSIONS

The Schnorr Authentication Protocol and Digital Signature Testing Process The protocol will be explained gradually in the following process, starting with the establishment of the key (Alice) as follows:

1. Alice selects p, q and a as below:

p = 166853
q = 59
a = 56937
These values meet the requirements that:
a. P and q are primes,
b. GCD (q, p-1) should not be worth 1,
c. The value of the operation (a ^ q) mod p should be 1.

2. Alice selects the value of s (s <q).

s = 58 (s are private key)

3. Alice calculates the value of v with the following formula:

v = a^(-s) mod p
v = 56937^(-58) mod 166853
v = ((56937^(-1) mod 166853)^58) mod 166853
Complete the operation (56937 ^ (- 1) mod 166853) with extended euclidean algorithm
(56937 ^ (- 1) mod 166853) = 76260
V = (76260 ^ 58) mod 166853 (finish with fast exponentiation)
V = 56937 (v is the public key)

The key that is formed will be used on the authentication process between Alice and bob; here is the process of authentication scheme with Schnorr algorithm:

1. Alice selects r value as follows:

r = 45
2. Alice calculates the value of x
x = a^r mod p (Finish with fast exponentiation)
x = 56937^45 mod 166853
x = 94651
Alice sends X to Bob
3. Bob chose the e value as follows:
e = 3598229
Bob sends e to Alice
4. Alice calculates the n value as follows:
n = (r + se) mod q
n = (45 + 58. 3598229) mod 59
n = 49
Alice sends n to Bob
5. Bob did the following verification:
z = ((a^y).(v^e)) mod p
z = ((a^y) mod p . (v^e) mod p) mod p
z = (56937^49 mod 166853). (56937^3598229 mod 166853) mod 166853
z = (102788.166427) mod 166853
94651 = 94651 (TRUE)
The result of the operation (((a ^ y). (V ^ e)) mod p) is equal to the value of z. The authentication process succeeded.
The next process is the process of digital signature scheme between alice and bob, here is the step with p, q, a, s (private), v (public) is known:
p= 166853
q= 59
a= 56937
s (private)= 58

v (public)= 56397
Message (M) = PENGUJIAN DIGITAL
1. Alice selects r value as follows:
R = 21
Alice calculates the value of z
z = a ^ r mod p (finish fast Exponentiation)
z = 56937 ^ 21 mod 166853
z = 153196

2. Alice combines M and z and calculates the hash value: e = H (M, z)
M(1) = ASCII from 'P' = 80
(M(1),z) = M(1) Combined with z
(M(1),z) = 80153196
e(1) = H(80153196)
e(1) = 65565550

M(2) = ASCII from 'E' = 69
(M(2),z) = M(2) Combined with z
(M(2),z) = 69153196
e(2) = H(69153196)
e(2) = 67515350
M(3) = ASCII from 'N' = 78
(M(3),z) = M(3) Combined with z
(M(3),z) = 78153196
e(3) = H(78153196)
e(3) = 49665455

M(4) = ASCII from 'G' = 71
(M(4),z) = M(4) Combined with z
(M(4),z) = 71153196
e(4) = H(71153196)
e(4) = 69575053

M(5) = ASCII from 'U' = 85
(M(5),z) = M(5) Combined with z
(M(5),z) = 85153196
e(5) = H(85153196)
e(5) = 51536752

M(6) = ASCII from 'J' = 74
(M(6),z) = M(6) Combined with z
(M(6),z) = 74153196
e(6) = H(74153196)
e(6) = 69545551

M(7) = ASCII from 'I' = 73
(M(7),z) = M(7) Combined with z
(M(7),z) = 73153196
e(7) = H(73153196)
e(7) = 66655650

M(8) = ASCII from 'A' = 65
(M(8),z) = M(8) Combined with z
(M(8),z) = 65153196
e(8) = H(65153196)
e(8) = 50655754

M(9) = ASCII from 'N' = 78
(M(9),z) = M(9) Combined with z
(M(9),z) = 78153196
e(9) = H(78153196)
e(9) = 49665455

M(10) = ASCII from ' ' = 32
(M(10),z) = M(10) Combined with z
(M(10),z) = 32153196
e(10) = H(32153196)
e(10) = 57576753

M(11) = ASCII from 'D' = 68
(M(11),z) = M(11) Combined with z
(M(11),z) = 68153196
e(11) = H(68153196)
e(11) = 51656755

M(12) = ASCII from 'I' = 73
(M(12),z) = M(12) Combined with z
(M(12),z) = 73153196
e(12) = H(73153196)
e(12) = 66655650

M(13) = ASCII from 'G' = 71
(M(13),z) = M(13) Combined with z
(M(13),z) = 71153196
e(13) = H(71153196)
e(13) = 69575053

M(14) = ASCII from 'I' = 73
(M(14),z) = M(14) Combined with z
(M(14),z) = 73153196
e(14) = H(73153196)
e(14) = 66655650

M(15) = ASCII from 'T' = 84
(M(15),z) = M(15) Combined with z
(M(15),z) = 84153196
e(15) = H(84153196)
e(15) = 69545256

M(16) = ASCII from 'A' = 65
(M(16),z) = M(16) Combined with z
(M(16),z) = 65153196
e(16) = H(65153196)
e(16) = 50655754

M(17) = ASCII from 'L' = 76
(M(17),z) = M(17) Combined with z
(M(17),z) = 76153196
e(17) = H(76153196)
e(17) = 51677049

3. Alice calculates the n value as follows: n = (r + se) mod q
n(1) = (r + (s . e(1))) mod q
n(1) = (21 + (58 . 65565550)) mod 59
n(1) = 50
n(2) = (r + (s . e(2))) mod q
n(2) = (21 + (58 . 67515350)) mod 59
n(2) = 23

n(3) = (r + (s . e(3))) mod q
n(3) = (21 + (58 . 49665455)) mod 59
n(3) = 58

n(4) = (r + (s . e(4))) mod q
n(4) = (21 + (58 . 69575053)) mod 59
n(4) = 10

n(5) = (r + (s . e(5))) mod q
n(5) = (21 + (58 . 51536752)) mod 59
n(5) = 5

n(6) = (r + (s . e(6))) mod q
n(6) = (21 + (58 . 69545551)) mod 59
n(6) = 12
n(7) = (r + (s . e(7))) mod q
n(7) = (21 + (58 . 66655650)) mod 59
n(7) = 34

n(8) = (r + (s . e(8))) mod q
n(8) = (21 + (58 . 50655754)) mod 59
n(8) = 15

n(9) = (r + (s . e(9))) mod q
n(9) = (21 + (58 . 49665455)) mod 59
n(9) = 58

n(10) = (r + (s . e(10))) mod q
n(10) = (21 + (58 . 57576753)) mod 59
n(10) = 11

n(11) = (r + (s . e(11))) mod q
n(11) = (21 + (58 . 51656755)) mod 59
n(11) = 8

n(12) = (r + (s . e(12))) mod q
n(12) = (21 + (58 . 66655650)) mod 59
n(12) = 34

n(13) = (r + (s . e(13))) mod q
n(13) = (21 + (58 . 69575053)) mod 59
n(13) = 10

n(14) = (r + (s . e(14))) mod q
n(14) = (21 + (58 . 66655650)) mod 59
n(14) = 34

n(15) = (r + (s . e(15))) mod q
n(15) = (21 + (58 . 69545256)) mod 59
n(15) = 12

n(16) = (r + (s . e(16))) mod q
n(16) = (21 + (58 . 50655754)) mod 59
n(16) = 15

n(17) = (r + (s . e(17))) mod q
n(17) = (21 + (58 . 51677049)) mod 59
n(17) = 10
Digital signatures generated as follows:

|65565550,50|67515350,23|49665455,58|69575053,10|515
36752,5|69545551,12|66655650,34|50655754,15|4966545
5,58|57576753,11|51656755,8|66655650,34|69575053,10|
66655650,34|69545256,12|50655754,15|51677049,10
Digital signatures are e and n. Alice sends Bob a message and digital signature.

4. Bob did the following calculations:
z'=((a^y).(v^e)) mod p

z'(1) = ((a^y(1)) mod p . (v^e(1)) mod p) mod p
z'(1) = (56937^50 mod 166853) .
(56937^65565550 mod 166853)
mod 166853
z'(1) = 153196

z'(2) = ((a^y(2)) mod p . (v^e(2)) mod p) mod p
z'(2) = (56937^23 mod 166853) .
(56937^67515350 mod 166853)
mod 166853
z'(2) = 153196

z'(3) = ((a^y(3)) mod p . (v^e(3)) mod p) mod p
z'(3) = (56937^58 mod 166853) .
(56937^49665455 mod 166853)
mod 166853
z'(3) = 153196

z'(4) = ((a^y(4)) mod p . (v^e(4)) mod p) mod p
z'(4) = (56937^10 mod 166853) .
(56937^69575053 mod 166853)
mod 166853
z'(4) = 153196

z'(5) = ((a^y(5)) mod p . (v^e(5)) mod p) mod p
z'(5) = (56937^5 mod 166853) .
(56937^51536752 mod 166853)
mod 166853
z'(5) = 153196

z'(6) = ((a^y(6)) mod p . (v^e(6)) mod p) mod p
z'(6) = (56937^12 mod 166853) .
(56937^69545551 mod 166853)
mod 166853
z'(6) = 153196

z'(7) = ((a^y(7)) mod p . (v^e(7)) mod p) mod p
z'(7) = (56937^34 mod 166853) .
(56937^66655650 mod 166853)
mod 166853
z'(7) = 153196

z'(8) = ((a^y(8)) mod p . (v^e(8)) mod p) mod p
z'(8) = (56937^15 mod 166853) .
(56937^50655754 mod 166853)
mod 166853
z'(8) = 153196

z'(9) = ((a^y(9)) mod p . (v^e(9)) mod p) mod p
z'(9) = (56937^58 mod 166853) .

(56937^49665455 mod 166853)
mod 166853
z'(9) = 153196

z'(10) = ((a^y(10)) mod p . (v^e(10)) mod p) mod p
z'(10) = (56937^11 mod 166853) .
(56937^57576753 mod 166853)
mod 166853
x'(10) = 153196

z'(11) = ((a^y(11)) mod p . (v^e(11)) mod p) mod p
z'(11) = (56937^8 mod 166853) .
(56937^51656755 mod 166853)
mod 166853
x'(11) = 153196

z'(12) = ((a^y(12)) mod p . (v^e(12)) mod p) mod p
z'(12) = (56937^34 mod 166853) .
(56937^66655650 mod 166853)
mod 166853
z'(12) = 153196

z'(13) = ((a^y(13)) mod p . (v^e(13)) mod p) mod p
z'(13) = (56937^10 mod 166853) .
(56937^69575053 mod 166853)
mod 166853
z'(13) = 153196

z'(14) = ((a^y(14)) mod p . (v^e(14)) mod p) mod p
z'(14) = (56937^34 mod 166853) .
(56937^66655650 mod 166853)
mod 166853
z'(14) = 153196

z'(15) = ((a^y(15)) mod p . (v^e(15)) mod p) mod p
z'(15) = (56937^12 mod 166853) .
(56937^69545256 mod 166853)
mod 166853
z'(15) = 153196

z'(16) = ((a^y(16)) mod p . (v^e(16)) mod p) mod p
z'(16) = (56937^15 mod 166853) .
(56937^50655754 mod 166853)
mod 166853
z'(16) = 153196

z'(17) = ((a^y(17)) mod p . (v^e(17)) mod p) mod p
z'(17) = (56937^10 mod 166853) .
(56937^51677049 mod 166853)
mod 166853
z'(17) = 153196

5. Bob combines M and z' and performs verification: e = H (M, z')

M(1) = ASCII from 'P' = 80
(M(1),z'(1)) = M(1) combined with z'(1)
(M(1),z'(1)) = 80153196
e(1) = H(80153196)
65565550 = 65565550 (TRUE)
M(2) = ASCII from 'E' = 69
(M(2),z'(2)) = M(2) combined with z'(2)
(M(2),z'(2)) = 69153196
e(2) = H(69153196)
67515350 = 67515350 (TRUE)

M(3) = ASCII from 'N' = 78
(M(3),z'(3)) = M(3) combined with z'(3)
(M(3),z'(3)) = 78153196
e(3) = H(78153196)
49665455 = 49665455 (TRUE)

M(4) = ASCII from 'G' = 71
(M(4),z'(4)) = M(4) combined with z'(4)
(M(4),z'(4)) = 71153196
e(4) = H(71153196)
69575053 = 69575053 (TRUE)

M(5) = ASCII from 'U' = 85
(M(5),z'(5)) = M(5) combined with z'(5)
(M(5),z'(5)) = 85153196
e(5) = H(85153196)
51536752 = 51536752 (TRUE)

M(6) = ASCII from 'J' = 74
(M(6),z'(6)) = M(6) combined with z'(6)
(M(6),z'(6)) = 74153196
e(6) = H(74153196)
69545551 = 69545551 (TRUE)

M(7) = ASCII from 'I' = 73
(M(7),z'(7)) = M(7) combined with z'(7)
(M(7),z'(7)) = 73153196
e(7) = H(73153196)
66655650 = 66655650 (TRUE)

M(8) = ASCII from 'A' = 65
(M(8),z'(8)) = M(8) combined with z'(8)
(M(8),z'(8)) = 65153196
e(8) = H(65153196)
50655754 = 50655754 (TRUE)

M(9) = ASCII from 'N' = 78
(M(9),z'(9)) = M(9) combined with z'(9)
(M(9),z'(9)) = 78153196
e(9) = H(78153196)
49665455 = 49665455 (TRUE)

M(10) = ASCII from ' ' = 32
(M(10),z'(10)) = M(10) combined with z'(10)
(M(10),z'(10)) = 32153196
e(10) = H(32153196)
57576753 = 57576753 (TRUE)

M(11) = ASCII from 'D' = 68
(M(11),z'(11)) = M(11) combined with z'(11)
(M(11),z'(11)) = 68153196
e(11) = H(68153196)
51656755 = 51656755 (TRUE)

M(12) = ASCII from 'I' = 73

(M(12),z'(12)) = M(12) combined with z'(12)
(M(12),z'(12)) = 73153196
e(12) = H(73153196)
66655650 = 66655650 (TRUE)

M(13) = ASCII from 'G' = 71
(M(13),z'(13)) = M(13) combined with z'(13)
(M(13),z'(13)) = 71153196
e(13) = H(71153196)
69575053 = 69575053 (TRUE)

M(14) = ASCII from 'I' = 73
(M(14),z'(14)) = M(14) combined with z'(14)
(M(14),z'(14)) = 73153196
e(14) = H(73153196)
66655650 = 66655650 (TRUE)

M(15) = ASCII from 'T' = 84
(M(15),z'(15)) = M(15) combined with z'(15)
(M(15),z'(15)) = 84153196
e(15) = H(84153196)
69545256 = 69545256 (TRUE)

M(16) = ASCII from 'A' = 65
(M(16),z'(16)) = M(16) combined with z'(16)
(M(16),z'(16)) = 65153196
e(16) = H(65153196)
50655754 = 50655754 (TRUE)

M(17) = ASCII from 'L' = 76
(M(17),z'(17)) = M(17) combined with z'(17)
(M(17),z'(17)) = 76153196
e(17) = H(76153196)
51677049 = 51677049 (TRUE)

The result of the operation H (M, z') is equal to the value e so that the digital signature verification process succeeds.

The following is the process of experimenting the delivery of messages with different message lengths in the network as well as the amount of time required for the authentication process.

Table-1. Experimental result.

No.	p, q, a Key	Length message	Time process
1	p=427249 q=43 a=58978	25	1.35 ms
2	p=427249 q=43 a=58978	35	2.765 ms
3	p=9895759 q=641 a=54251	40	4.31 ms
4	p=704461 q=59 a=34253	55	6.292 ms
5	p=490201 q=43 a=20191	55	4.13 ms
6	p=1740527 q=419 a=38820	55	2.19 ms
7	p=145361 q=23 a=70968	55	2.16 ms
8	p=582449 q=23 a=82637	60	7.13 ms
9	p=207763 q=31 a=58336	25	1.31 ms
10	p=799789 q=73 a=65266	65	7.61 ms

Based on experimental results conducted in Table-1 found that the process time of the use of protocols depends on the value of p and a as prime numbers, from the table above generated graphs of process time as in Figure-1 below:

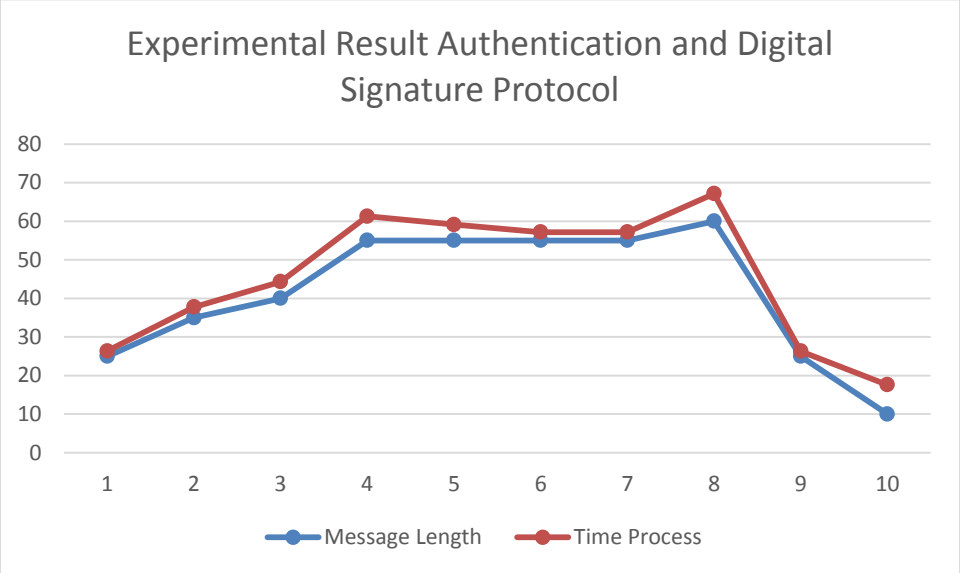

Figure-1. Experimental result authentication and signature protocol.

Figure-1 shows that the time it takes to secure a message depends on the value of p and a, the greater value of p and a, the longer it takes time to process.

CONCLUSIONS

After completing the experiment of Schnorr Authentication and Digital Signature Scheme, the authors draw the conclusion that the use of the Schnorr scheme can serve as a additional security mechanism especially in two-way communication process and the process is also not complicated if implemented, one of the weaknesses that may occur is to take time much longer than cryptography process, and for next research this weakness could be solved.

REFERENCES

[1] L. Wang and T. Song. 2016. An Improved Digital Signature Algorithm and Authentication Protocols in Cloud Platform. in IEEE International Conference on Smart Cloud, New York, NY, USA.

[2] N. R. Chakraborty, M. T. Rahman and M. E. Rahman. 2016. Generation and verification of digital signature with two factor authentication. in International Workshop on Computational Intelligence (IWCI), Dhaka, Bangladesh.

[3] R. Rahim. 2017. Man-In-The-Middle-Attack Prevention Using Interlock Protocol Method. ARPN Journal of Engineering and Applied Sciences. 12(22): 6483-6487.

[4] J. Sravanthi and M. K. Prasad. 2011. Robust and Secure Digital Signature for Image Authentication over Wireless CHANNELS. International Journal of Computer Trends and Technology. 1(3): 245-250.

[5] P. Shiralkar and B. S. Vijayaraman. 2003. Digital Signature: Application Development Trends in E-Business. Journal of Electronic Commerce Research. 4(3): 94-101.

[6] K. K. Doke and S. M Patil. 2012. Digital Signature Scheme for Image. International Journal of Computer Applications. 49(16): 1-6.

[7] E. Hariyanto and R. Rahim. 2016. Arnold's Cat Map Algorithm in Digital Image Encryption. International Journal of Science and Research (IJSR). 5(10): 1363-1365.

[8] Legito and R. Rahim. 2017. SMS Encryption Using Word Auto Key Encryption. International Journal of Recent Trends in Engineering & Research (IJRTER). 3(1): 251-256.

[9] D. Nofriansyah and R. Rahim. 2016. Combination of Pixel Value Differencing Algorithm with Caesar Cipher Algorithm for Steganography. International Journal of Research in Science & Engineering. 2(6): 153-159.

[10] R. Rahim and A. Ikhwan. 2016. Cryptography Technique with Modular Multiplication Block Cipher and Playfair Cipher. IJSRST. II(6): 71-78.

[11] R. Rahim and A. Ikhwan. 2016. Study of Three Pass Protocol on Data Security. International Journal of Science and Research (IJSR). 5(11): 102-104.

[12] A. P. U. Siahaan and R. Rahim. 2016. Dynamic Key Matrix of Hill Cipher Using Genetic Algorithm. International Journal of Security and its Applications. 10(8): 173-180.

[13] C. Braz and J. Robert. 2006. Security and Usability: The Case of the User Authentication Methods. in Proceedings of the 18th International Conference of the Association - IHM '06 (2006).

[14] R. P. Guidorizzi. 2013. Security: Active authentication. IT Professional. 15(4): 4-7.

[15] R. Rahim, M. Dahria, M. Syahril and B. Anwar. 2017. Combination of the Blowfishand Lempel-Ziv-Welch algorithms for text compression. World Transactions on Engineering and Technology Education. 15(3): 292-297.

[16] N. K. Nair, Navin K.S and S. Chandra C.S. 2015. Digital Signature and Advanced Encryption Standard for Enhancing Data Security and Authentication in Cloud Computing. International Journal for Research in Applied Science & Engineering Technology (IJRASET). 3(3): 240-244.

[17] R. Haddaji, R. Ouni, S. Bouaziz and A. Mtibaa. 2016. Comparison of Digital Signature Algorithm and Authentication Schemes for H.264 Compressed Video. International Journal of Advanced Computer Science and Applications, 7(9): 357-363.

[18] M. A. Sadikin and R. W. Wardhani. 2016. Implementation of RSA 2048-bit and AES 256-bit with digital signature for secure electronic health record application. In: 2016 International Seminar on Intelligent Technology and Its Applications (ISITIA), Lombok, Indonesia.

[19] A. Mali, C. Mahalle, M. Kulkarni, T. Nangude and G. Navale. 2017. Digital Signature Authentication and Verification on Smart Phones using CRiPT Algorithm. International Research Journal of Engineering and Technology (IRJET). 4(5): 332-338.

[20] N. K. Ratha, J. H. Connell and R. M. Bolle. 2001. Enhancing security and privacy in biometrics-based authentication systems. IBM Systems Journal. 40(3): 614-634.

[21] W. Xin, M. Wang, S. Shao, Z. Wang and T. Zhang. 2015. A variant of schnorr signature scheme for path-checking in RFID-based supply chains. In: IEEE 12th International Conference on Fuzzy Systems and Knowledge Discovery (FSKD), Zhangjiajie, China.

[22] Z. Cao and O. Markowitch. 2009. Security Difference between DSA and Schnorr's Signature. In: IEEE International Conference on Networks Security, Wireless Communications and Trusted Computing, Wuhan, Hubei, China.