Available online at www.sciencedirect.com

SciVerse ScienceDirect

AASRI Procedia 1 (2012) 100 – 105

AASRI
Procedia

www.elsevier.com/locate/procedia

2012 AASRI Conference on Computational Intelligence and Bioinformatics

Security Module in Information Unilateral Transmission System among Networks

Shanshan Liu[a*], Peng Wang[a], Lei Luo[a], Yong Cai[a]

[a]Military Transportation University, Chenglin Road Hedong Area Tianjin, 300161, China

Abstract

Physical isolation among different confidential levels Networks are always being used to keep data unrevealed, information unilateral transmission system transmits data across networks. Information security module is essential in system and being used to make sure data untouched in low confidential level network. MD5 algorithm is used in digital signature and RSA arithmetic is used encoding and decoding process. Experiments approve its efficiency and security.

Keywords：Physical Isolation, Unilateral Transmission, RSA, MD5

1. Introduction

According to《Law of PRC Keeping National Secret》rule: “Computer which recorded national secrets cannot link to internet or other public networks directly, must isolate from them, it is called “physical isolation”. Networks of government, military and bank are independent to keep data safe. Physical isolation solved disclosure in a way, but set obstacles to obtain information from external networks. Unilateral Transmission System transmits data from low security level to high unilaterally, and prevents data from leaking out. Security module uses RSA and MD5 algorithm to insure data against destroying, tampering and replacing, promises its security and reliability. [1]

[*] * Shanshan Liu. Tel.: 18622616583; fax: 022-84657714.
E-mail address: 870166743@qq.com.

2212-6716 © 2012 Published by Elsevier Ltd. Open access under CC BY-NC-ND license.
doi:10.1016/j.aasri.2012.06.018

2. System Design

Unilateral Transmission System includes gap and software system, gap is laid between different security level networks, data “flows” through it unilaterally from low security level network to high, and there is no direct physical link in it. Host computer in low security level network (external network in short) encodes data to QR codes, camera in gap catches QR codes which include data and display on screen, then host computer in high security level network (internal network in short) decodes them back to data. Security module consists of RSA encryption and MD5 signature process which applied to terminal computer in external network, and decryption, signature authentication process applied to host computer in internal network.

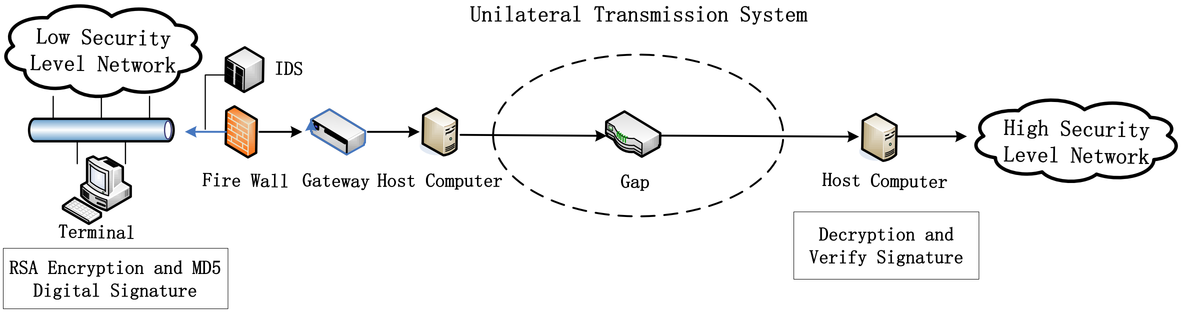
Unilateral Transmission System
Low Security Level Network
IDS
Fire Wall
Gateway
Host Computer
Gap
Host Computer
High Security Level Network
Terminal
RSA Encryption and MD5 Digital Signature
Decryption and Verify Signature

Fig. 1. system structure

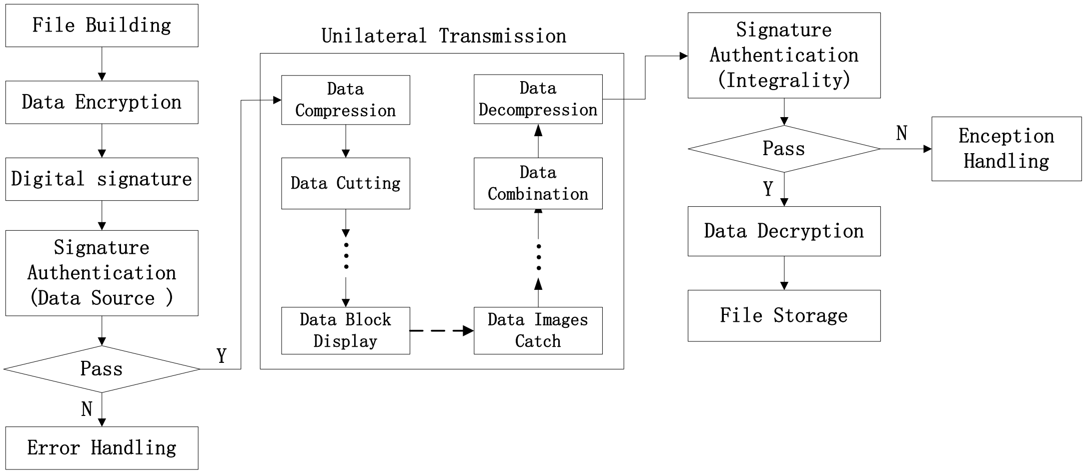

Fig. 2. software system

3. Digital Signature based on MD5 Algorithm

3.1. Fill in bits

Fill in bits of input data, make the length of data “m”, divides 512, cumulative remainder is 448, fill in one “1” and n “0” , $(m+1+n)\bmod 512 = 448$. After filling bits, length $m = N * 512 + 448(\mathrm{bit})$, then add a 64 bit binary number which demonstrates former length “m” back to the data. After that, $N * 512 + 448 + 64 = (N + 1) * 512$.

3.2. Structure initialization

Define a structure in process, the structure consists plain text block (512bit), and hashed value (128bit). During hashing process, all the hashed values are computed from plain text are passed by it.

3.3. Group data

Group filled data, 512 bits a group, N groups.

3.4. Compute groups

Assign primitives to 4 integer chaining variable, which are 32 bits, they are:
A=0x01234567，B=0x89ABCDEF，C=0xFEDCBA98，D=0x76543210。
Assign primitive A, B, C, D to variable a, b, c, d. Divide 512 bits data block to 16 strings of 32 bits. Compute Each 32 bit string 4 rounds. There are many processes in a round, the first process in 4 rounds are different, the rest process are the same. These are 4 nonlinear functions used in process.

F(X,Y,Z) =(X&Y)|((~X)&Z)
G(X,Y,Z) =(X&Z)|(Y&(~Z))
H(X,Y,Z) =X^Y^Z
I(X,Y,Z) =Y^ (X| (~Z))

If the counterpart bits of X, Y and Z are independent and symmetrical, the results are independent and symmetrical too.

FF(a, b, c, d, M′, s, n) shows a = b + ((a + F(b, c, d) + M′ + n) << s)
GG(a, b, c, d, M′, s, n) shows a = b + ((a + G(b, c, d) + M′ + n) << s)
HH(a, b, c, d, M′, s, n) shows a = b + ((a + H(b, c, d) + M′ + n) << s)
II(a, b, c, d, M′, s, n) shows a = b + ((a + I(b, c, d) + M′ + n) << s)

After these 4 rounds compute process, add A, B, C, D to a, b, c, d separately. Then compute next group of data, the output are the cascades of A, B, C and D. When all the groups of data are computed, cascade the outcome, we get the MD5 results. [2]

4. Working Theory of Internal Network Terminal

RSA encryption algorithm has high intensity, being difficult to crack, it costs much time to encrypt data by comparing with DES and other encryption algorithm, as computer hardware improved day by day, RSA encryption speed is receptible. The more secret key it uses, the more time program runs, in the system, 1024 bit secret key is suitable.[3]

Secret key generation

- Prime number p, q, order n=p*q；
- Get integer e which relatively prime with (p-1)×(q-1)，get answer d from equation d*e=1(mod(p-1)×(q-1))；
- Binary group (e, n) is public secret key, binary group (d, n) is private secret key.

Table 1. RSA algorithm diagram

Public secret key KU	n: p*q (p and q are prime numbers and secret) e: relatively prime with (p-1)×(q-1)
Private secret key KR	d: e-1[mod(p-1)(q-1)] n: p*q (p and q are prime numbers and secret)
Encrypt	c ≡ me mod n
Decrypt	m ≡ cd mod n

Order n=p×q，p and q are both prime numbers， n=(p-1)(q-1) is the Euler number of n.[4]

Euler theorem deduce：

If n=p×q，p and q are both prime numbers, k is integer, then

$m^{k(p-1)(q-1)+1} = m \bmod n$， 0<=m<=n

If choose e, d，and e×d=k (n)+1，and：

$$ed = 1 \bmod \phi(n) \Rightarrow d = e^{-1} \bmod \phi(n)$$

Public secret key：PK={e, n}，private secret key：SK={d, n}

Text encryption process input: Clear text D, modulus n, encryption exponent e (public secret key encryption) or decryption exponent d (private secret key encryption).

Clear text encryption process:

The output is secret text. The length of D is less than [log2n]-11, to make sure the number of filling cluster is not null when it is transformed to PKCS format.

- Format clear data. Use PKCS format: EB=00||BT||PS||00||D, BT means type of block, PS is filling cluster, D is clear text. The beginning is 0 to insure the length of EB is more than k. public key encryption BT=02, private key decryption BT=01. When BT=02, PS is non-null random number; when BT=01, the value of PS is FF.
- Clear text is char data type, need to be transformed to integer.
- RSA computing process. x is integer encryption number, $y = x^c \bmod n$， 0<=y<n, y is secret text, when encrypting with public secret key, c is public secret key encryption exponent e; When encrypting with private secret key, c is private secret key encryption exponent d.

Secret text decryption process:

Inputs of decryption process are: secret text ED, module n, encryption exponent e (public secret key decryption) or decryption exponent d (private secret key decryption), the result is clear text.

- Transform secret text to integer.
- RSA computing process. $x = y^c \bmod n$， 0<=x<n, x is clear text.
- Now the clear text is in integer type, transform it to ASCII type, acquire clear text in PKCS format.
- Divide original clear text from clear text in PKCS format. The process is also check integrality of data. If problems below happen then decryption failed: cannot divide clearly; filling cluster is less than 64 bit or mismatch with type labeled by BT; BT is not accord with practical type.

5. Experiment

Experiment is held in campus network and LAN in laboratory, uses 2 normal PC as external network terminal and internal network terminal. Main computer hardware is as below:

CPU: AMD 9650 Processing Unit; RAM：Kingston DDR2 800MHz, 2G; GPU: NVIDIA GeForce 9800GT，1G;

Txt file is used to be transmitted in experiment, according file size there are 3groups and testing 300 times in each group, result is as below:

Table 2. Time consume of 32 kb file in system running with different size of secret key

File size（kb）	function	Time of running 512bit secret key（ms）	Time of running 768 bit secret key（ms）	Time of running 1024 bit secret key（ms）
32	File encryption	15	16	19
32	Digital signature	15	78	99
32	File decryption	73	121	156
32	Signature authentication	78	135	242

Table 3. Time consume of 64 kb file in system running with different size of secret key

File size（kb）	function	Time of running 512bit secret key（ms）	Time of running 768 bit secret key（ms）	Time of running 1024 bit secret key（ms）
64	File encryption	21	28	33
64	Digital signature	21	92	122
64	File decryption	99	147	233
64	Signature authentication	139	205	279

Table 4. Time consume of 128 kb file in system running with different size of secret key

File size（kb）	function	Time of running 512bit secret key（ms）	Time of running 768 bit secret key（ms）	Time of running 1024 bit secret key（ms）
128	File encryption	27	72	156
128	Digital signature	30	133	174
128	File decryption	125	169	257
128	Signature authentication	241	311	383

Result analysis：From 3 tabs above we find time consume of file encryption and digital signature increases nonlinearly with increase of size of file and length of secret key.

Suppose time internal of file transmission T, time consume t_1, time consume of file encryption and digital signature t_2. In order to make the system use in practice, it fulfils:

$$T > t_1 + t_2 \tag{1}$$

Because it may resend the file if necessary, there is must be redundancy time during transmission, so:

$$T > 3t_1 + t_2 \tag{2}$$

Suppose time consume of encryption and decryption f_1, and digital signature and authentication f_2, so:

$$t_2 = f_1 + f_2 \tag{3}$$

then：

$$T > 3t_1 + f_1 + f_2 \tag{4}$$

6. Conclusion

The paper actualizes security module which consists of digital signature and RSA encryption and decryption in unilateral system. The module insures security and integrality of system and it is also efficient and available.

References

[1] Communication technology [J]. Shanshan Liu, Rui Zhao 2009.11.

[2] Hongjun Yi, Minggao She. MD5 algorithm and digital signature. [J]. Computer and digital engineer. 2006.5

[3] Hanae Nozaki,Masahiko Motoyama,Atsushi Shimbo,Shinichi Kawamura[M]. Cryptographic Hardware and Embedded Systems . DOI: 10.1007/3-540-44709-1_30 .2001

[4] Wang Qian, Ni Jianwei, a kind of RSA encryption algorithm [J]. Chong Qing University Transaction，2005.1