IACSIT International Journal of Engineering and Technology Vol.1,No.5,December,2009
ISSN: 1793-8236

Information Security : An Artificial Intelligence And Data Mining Based Approach

P. Chakrabarti , MIACSIT

***Abstract*—Information security plays a major role in case of secured data transmission . In this paper some intelligent techniques have been pointed out regarding effective cipher generation based on SKEY , interlock protocol and SKID . We have also cited cryptosystems based on linear property of propositional operator, the concept of fuzzy logic , optimized function and concept of data mining based on comparison analysis. Various proposed techniques have also been pointed out for shared key evalution in multi-party communication and the areas concerned are shared key generation based on support rule, cent percent confidence rule ,minimal frequent set and sequence mining and feature-based method.**

***Index Terms*—SKEY, Interlock protocol , SKID, propositional operator , fuzzy logic , comparison analysis , optimized function , shared key generation**

I. Introduction

SKEY is mainly a program for authentication and it is based on a one-way function.

A. *Proposed Algorithm*

1) Host computes a Bernoulli trial with biased coin for which p= probability of coming 1.q=(1-p =probability of coming 0

Let number of trials be n.
Assume n=6, and string=110011.

2) Host sends the string to Alice.

3) Alice modifies its own public key based on that the new public key = previous key + (binary equivalent of the number of 1's present in the string).

4) Alice creates a Shared Key.

5) Alice modifies the public key along with modification scheme with shared key.

6) Alice then encrypts the string with her private key and sends back to the host along with her name.

7) Host first decrypts public key and accordingly fetches it from database of Alice and computes the result.

8) If match is found, then it performs another level of verification by decrypting the string with new value of Alice's public key.

9) If that also matches, then authentication of Alice is certified.

B. *Mathematical Analysis*

Encryption

Let us take a super-increasing knapsack sequence, for example {2, 3, 6, 13, 27, 52}, and multiply all of the values by a number n, mod m. The modulus should be a number greater than the sum of all the numbers in the sequence: for example, 105. The multiplier should have no factors in common with the modulus: for example, 31. The normal knapsack sequence would then be

2 * 31 mod 105 = 62
3 * 31 mod 105 = 93
6 * 31 mod 105 = 81
13 * 31 mod 105 = 88
27 * 31 mod 105 = 102
52 * 31 mod 105 = 37

The knapsack would then be {62, 93, 81, 88, 102, 37}.

The super-increasing knapsack sequence is the private key. The normal knapsack sequence is the public key.

If the message is 110011 in binary, encryption using the previous knapsack would proceed like this:

Message = 110011 corresponds to 62+93+102+37=294

No. of 1's=4, Binary form of 4=100

New Message= Old message + (no. of 1's in binary form)=294 + 100=394

The ciphertext would be 394,4.

Decryption

The super-increasing knapsack is {2, 3, 6, 13, 27, 52}, m is equal to 105, and n is equal to 31. The ciphertext message is 394,4. In this case n-1 is equal to 61, so the ciphertext values must be multiplied by 61 mod 105.

Original Ciphertext=394 – (Binary form of 4)=394 – 100= 294

294 * 61 mod 105 = 14 =1+2+5+6, which corresponds to 110011

The recovered plaintext is 110011.

II. Interlock protocol with date as Session Key

A. *Proposed Algorithm*

1) Alice and Bob generate a session key (let it be based on the date) for sharing.

Manuscript received July 6 . 2009.

P.Chakrabarti is with Computer Science and Engineering department , Bengal Institute of Technology and Management , Bolpur-731236 , West Bengal, India

Let KAB.

2) Alice encrypts its public key and sends EKAB(KAPUBLIC) to Bob. Bob sends EKAB(KBPUBLIC) to Alice.

3) Alice decrypts and gets KBPUBLIC. She then sends half of the message for Bob in encrypted form by KBPUBLIC.

4) Similarly Bob does so.

5) Alice then computes,

KAPUBLIC' = Modification of KAPUBLIC , sends EKAB(KAPUBLIC') to Bob.

6) Similarly Bob does so.

7) Alice then sends EKBPUBLIC'(other half of message) to Bob.

8) Similarly Bob performs.

9) Each receiver then decrypts the message in parts by respective keys and retrieve the message sent to him/her.

B. *Mathematical Analysis*

Encryption

Let the session key for sharing be in the form of date. Let the date be 28.08.08 then:
Perform the followings:
1. Take the UNITth place of the day, month and the year i.e. the right most digit in each.

2. Then, Day = RHS of 28=8
Month = RHS of 08=8
Year= RHS of 08=8
(Day * Month) + Year= (8*8) + 8 = 72 [Maximum can be (9*9)+9=90]
3. Again take the TENth place of the result.
i.e. LHS of 72= 7 [Maximum can be 9]
4. Convert it to its Binary Form.
i.e. Binary form of 7 = 111
5. Again Let us take a super-increasing knapsack sequence, for example {2, 3, 6, 13, 27, 52}, and multiply all of the values by a number n, mod m. The modulus should be a number greater than the sum of all the numbers in the sequence: for example, 105. The multiplier should have no factors in common with the modulus: for example, 31. The normal knapsack sequence would then be

2 * 31 mod 105 = 62
3 * 31 mod 105 = 93
6 * 31 mod 105 = 81
13 * 31 mod 105 = 88
27 * 31 mod 105 = 102
52 * 31 mod 105 = 37

The knapsack would then be {62, 93, 81, 88, 102, 37}.
The super-increasing knapsack sequence is the private key. The normal knapsack sequence is the public key. Let the first half of the message be 110011 in binary form encryption using the previous knapsack would proceed like this:
6. Message = 110011
Session Key= 111
Perform ((Message) XOR (Session Key))= ((110011) XOR (111))= 110100
7. Message = 110100 corresponds to 62+93+88=243

The ciphertext would be 243, along with the date (The Date must be given after all the ciphers) i.e. 243, (28.08.08).
Decryption
1. The super-increasing knapsack is {2, 3, 6, 13, 27, 52}, m is equal to 105, and n is equal to 31. The ciphertext message is 287,7. In this case n-1 is equal to 61, so the ciphertext values must be multiplied by 61 mod 105.
2. Cipher Text= 243
243 * 61 mod 105 = 14 =1+2+4, which corresponds to 110100
3. Date= 28.08.08
i. Take the UNITth place of the day, month and the year i.e. the right most digit in
each.

ii. Then, Day = RHS of 28=8
Month = RHS of 08=8
Year = RHS of 08=8
(Day * Month) + Year= (8*8) + 8 = 72 [Maximum can be (9*9)+9=90]

iii. Again take the TENth place of the result.
i.e. LHS of 72= 7 [Maximum can be 9]
iv. Convert it to its Binary Form.
i.e. Binary form of 7 = 111

4. Cipher Text = 110100
Session Key= 111
Perform ((Cipher Text) XOR (Session Key))= ((110100) XOR (111))= 110011
The recovered first half of plaintext is 110011.

III. ANALYSIS OF MODIFIED SKID

A. *Proposed Algorithm*

1) Alice chooses a random number R_A and sends it to Bob.
2) Bob chooses a random number R_B and sends it to Alice.
3) Alice and Bob make a secret shared key K.
4) Bob generates $R_{A'}$, $R_{B'}$, K' and sends $E_K(R_{A'}, R_{B'}, K')$ and $H_K(R_{A'}, R_{B'}, B)$ to Alice, H_K being for the MAC.
5) Alice extracts R_A, R_B, K and then computes $H_K(R_{A'}, R_{B'}, B)$ to find B. Then she matches that with what was sent to her by Bob.
6) If match= true, Alice knows she is communicating with Bob.

B. *Mathematical Analysis*

Let R_A = some prime no.= p = 3 (say)
R_B = some other prime no.= q = 5 (say)
K= p * q = 3 * 5 = 15
Let $R_{A'}$ = ln(p)= ln(3)
$R_{B'}$ = ln(q)= ln(5)

$K' = \ln(K) = \ln(15)$

$E_K(R_{A'}, R_{B'}, K') = \ln(3) * \ln(5) * \ln(15) = \ln(225)$ and

$H_K(R_{A'}, R_{B'}, B) = \ln(3) * \ln(5) * \ln(25) = \ln(375)$, Let B=MAC = 25.

This is $E_K(R_{A'}, R_{B'}, K')$ and $H_K(R_{A'}, R_{B'}, B)$ is being send to Alice.

Alice extracts by $DE_K(R_{A'}, R_{B'}, K') = e^{EK} = e^{\ln(225)} = 225 = 3 * 5 * 15$

$= R_A * R_B * K$

$DH_K(R_{A'}, R_{B'}, B) = e^{HK} = e^{\ln(375)} = 375 = 3 * 5 * 25$

$= R_A * R_B * B$

Since match= true, Alice knows she is communicating with Bob

IV. ENCRYPTION BASED ON PROPOSITIONAL LOGIC

Let message= m1=110111.Key = 111010. If m1 XOR k= 111010 is the encrypted result based on linear property, then if hacker knows k, then by XOR operation m1 will be revealed. To solve this, an intermediate result has to be found out and it will be XOR-ed with m1 to get resultant cipher. The necessary bit-padding is also applied.

Let the intermediate result be based on the truth value of (~m1Uk) ∩ (m1↔k)

m_1	k	$\sim m_1$	$(\sim m_1 U k)$	$(m_1 \leftrightarrow k)$	$(\sim m_1 U k) \cap (m_1 \leftrightarrow k)$
0	0	1	1	1	1
0	1	1	1	0	0
1	0	0	0	0	0
1	1	0	1	1	1

Therefore intermediate result is = 001001
Resultant cipher = m_1 **XOR** intermediate result = 111110

V. ENCRYPTION SCHEME BASED ON FUZZY OPERATORS

Let message = 010100111001000.Let 000→0.1, 001→0.2, 010→0.3, 011→0.4, 100→0.5, 101→0.6, 110→0.7, 111→0.8.Therefore if we can denote message as a fuzzy set and replace each of 3 bits by its corresponding value, then $\tilde{M}$ = {(x1,0.3),(x2, 0.5),(x3,0.8),(x4,0.2),(x5,0.1)}.

Let the encrypted key will be generated based on each value of x and it will be governed by [{(0.3)1/2 + ((0.5)2 + (0.8)2)} + (0.2/0.6 + 0.6/0.6)] in binary form. It should be noted that dilation= √μA(x), concentration= [μA(x)] 2 , normalization= μA(x)/maxx μA(x).

VI. ENCRYPTION SCHEME BASED ON OPTIMIZATION TECHNIQUE

Let the message = m1 = 110101. The values of the possible keys are k1 = 100001, k2 = 110111, k3 = 10010, k4 = 111100.For each key its corresponding optimized value is obtained and the one, whose value is largest, is the ultimate encrypted key. m1 = 110101, k1 = 100001 m1 XOR k1 = k1 ′

Objective_function1 = (mi-k ′ i) 2 + (mi-k ′ i-1) 2 + ………………….

= 1+0+0+0+0+1 =2.

Similarly,objective_function2= 1+1+0+1+1+1=5 ; objective_function3 =1+0+0+0+1+0=2 ;

objective_function4 = 1+1+1+1+0+0=4.So, objective_function2 is the largest. Hence cipher = m1 XOR (k2 in reverse form) = 110101 XOR 111011 = 001110.

VII. THEORY OF COMPARISON ANALYSIS

Let original message is “MOTHER”

For the first alphabet, μvalue = 1/((position of that)+ π /100)

Hence its offset value = ceiling of (the product of μvalue and 10)

The weight is given by its position in alphabet string

Therefore total_value = offset value * weight

From the next character onwards,

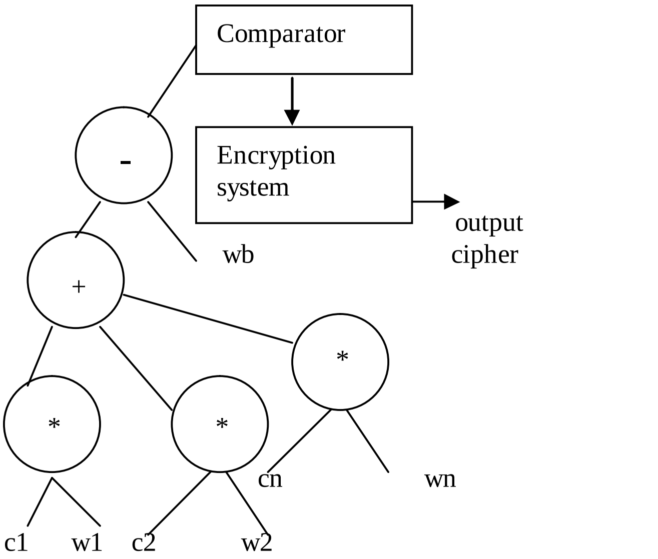
μvalue_next = 1/(mod value of (position of next-position of previous)+ π /100)

Hence total_value is calculated in similar manner.

Now, bias value will be equal to total number of characters in the message.

Compute net_value as (total_value_ first char + total _value_last char)- (bias value) and let it be x (say).

Mode	Operation
0<x <100	Reverse the message
100<x<150	Circular left shift of message by n/2 bits where n= bias value
150<x<200	Circular right shift of message by n/2 bits



ci = offset value, for i = 1 to n, wi = weight , wb = bias value

Fig1: Neuro-fuzzy based cryptosystem

Iteration 1: μ_M= 1/(position of M in alphabet list + π/100)= 0.077.Offset value=ceiling of (0.077*10)=1. Weight=

position of M in alphabet list=13 .Thus , total_value = 1*13 =13.

Iteration 2: μ_O= 1/((position of O – position of M)+ π/100)= 1/2.031=0.492.Offset value = ceiling of (0.492*10)=5. Weight=15 . Thus, total_value= 5*15=75

Iteration 3: μT= 1/((position of T – position of O)+ π/100)= 0.199.Offset value = 2 Weight = 20. Thus total_value= 2 * 20 = 40

Iteration 4: μH= 1/((position of H – position of T)+ π/100)= 0.083.Offset value = 1 Weight = 8. Thus total_value= 1 * 8 = 8

Iteration 5: μE = 1/((position of E – position of H)+ π/100)= 0.33.Offset value = 4 Weight = 5.Thus total_value= 4 * 5 = 20

Iteration 6: μR= 1/((position of R – position of E)+ π/100)= 0.077.Offset value = 1 Weight =18 .Thus total_value= 1 * 18 = 18

Now, wb= bias value = number of bits in MOTHER= 6

So net_value= accumulated sum of all total_value – wb = (13+75+40+8+20+18)-6 =168. It falls in the range 150<x<200. So, “MOTHER” is circular right shifted by 6/2=3 bit

Therefore resultant cipher is “HERMOT”.

VIII. SHARED KEY GENERATION BASED ON COMMUNICATION BASED ON SUPPORT

A. Algorithm

A and B are two parties. K_1, k_2, k_3, k_4, k_5, k_6 are keys) which are protected to A and B only.

A sends messages m_1, m_2, m_3, m_4, m_5,m_6 in encrypted form with the help of one or more keys.

B will decipher each message by error-and-trial technique and form sets. The key having maximum support is the shared key between A and B. If the number of shared keys is more than one, then that one is primary while other one is candidate to it.

B. Analysis

Message	Encrypted key
M_1	SK_1=f(k_1, k_3, k_4, k_6)
M_2	SK_2=f (k_3,k_5)
M_3	SK_3=f (k_4, k_5, k_6)
M_4	SK_4=f (k_2, k_3, k_5)
M_5	SK_5=f (k_1, k_2)
M_6	SK_6=f (k_1, k_2, k_3, k_6)

So, it is seen that k_3 is supported by 4 out of 6 sets of shared key. This support of k_3= 66.6%. Hence shared key of A and B is k_3.

If hacker hacks k_1, k_2, k_3, .. k_6 then by applying error and trial it will get shared key.

So concept of automatic variable shared key is proposed.

The concept is that shared key = (key having maximum support) XOR (XOR of the values of messages where the support is not available)

Hence, k_3= key having maximum support

M_3, m_5= messages encrypted without k_3

Therefore, Shared key= k_3 XOR m_3 XOR m_5

This scheme cannot be revealed to the packer. So it will hack k_3 instead of the modified value of shared key.

IX. SHARED KEY GENERATION COMMUNICATION BASED ON CENT PERCENT CONFIDENCE RULE

C. Algorithm

Input: m_1,m_2, m_3, m_4, m_5, m_6 to A
k_1, k_2, k_3, k_4, k_5, k_6 to A and B

Step 1 : A encrypts each of the messages with combination of the keys and sends it to B

Step 2 : B finds the key which has the confidence level of 100% i.e. key 1=> key 2

If key1 exists, then key2 will also exist and hence confidence of key1 => key2 is 100%.

Step3 : Shared key is key 1

Step 4 : (Application only for enhancing security level)

Shared key = key 1 XOR key-new

Where key-new can be obtained such that key-new =>key1 is minimum

D. Analysis

M_1	SK_1=(k_1, k_3, k_4, k_6)
M_2	SK_2= (k_3, k_5)
M_3	SK_3= (k_4, k_5, k_6)
M_4	SK_4= (k_3, k_3, k_5)
M_5	SK_5= (k_1, k_2)
M_6	SK_6= (k_1, k_2, k_3, k_6)

Only k_4 => k_6 has confidence level of 100% So shared key k_4 (upto step3)

Association scheme	Probability
K_1=> k_4	$1/_3$
K_2=> k_4	0
K_3=> k_4	$1/_4$
K_5=> k_4	$1/_2$
K_6=> k_4	2/3

So key-new = k_2 since it has least probability

Therefore, Shared key = k_4XORk_2

X. SHARED KEY EVALUATION BASED ON MINIMAL FREQUENT SET

The minimal frequent set can be formed based on the minimum probability of the combination of items. The shared key is the XOR of the XOR values of each of the pairs of elements of the set.

Message	Encrypted key
M_1	SK_1= (k_1, k_3, k_4, $k_{6=)}$
M_2	SK_2= (k_3, k_5)
M_3	SK_3= (k_4, $k_{5,}$ k_6)
M_4	SK_4= (k_2, k_3, k_5)
M_5	SK_5= (k_1, k_2)
M_6	SK_6= (k_1, k_2, k_3, k_6)

Sequence	Session	A B C D E F
1	2	1 0 0 1 0 1
2	5	0 0 0 0 1 1
3	8	0 0 0 0 0 1

Among the combination of the keys, only (k_1, k_5) and (k_2, k_4) have least probability and it is zero.

Therefore, minimal frequent set= {k_1, k_5, k_2, k_4}

Therefore, shared key= (k_1XORk_5) XOR (k_2XORk_4)

XI. SHARED KEY GENERATION IN THE LIGHT OF SEQUENCE MINING

Let us suppose that four users viz.U1,U2,U3,U4 are in a network. Each of U1,U2,U3 transmits three messages to U4 in successive sessions.

Sender	Key	Operations
U1	110110	U1(m1) à U4
U2	100101	U2(m1) à U4
U3	001010	U3(m1) à U4
U1	001100	U1(m2) à U4
U2	000011	U2(m2) à U4
U3	100001	U3(m2) à U4
U1	111100	U1(m3) à U4
U2	000001	U2(m3) à U4
U3	110100	U3(m3) à U4

A. Algorithm

Step 1 : Designate each bit of key as a character.

Step 2 : If the character index value is 1 include it in sequence.

Step 3 : else ignore the value.

Step 4 : Identify the pattern that is decided by the communicating party and fetch the combination.

Step 5 : The shared key for each user will be based on the combined result

Step 6 : Repeat the steps 1to5 for other users

Step 7 : Final shared key will be based on shared key in combined form of U1/U2/U3 and computation scheme.

Sequence	Session	A B C D E F
1	1	1 1 0 1 1 0
2	4	0 0 1 1 0 0
3	7	1 1 1 1 0 0

B. Analysis

The bits can be denoted by A,B,C,D, E,F.

Combined sequence of U1:
(A,B,D,E) à (C,D) à (A,B,C,D)

Combined sequence of U2 :
(A,D,F) à (E,F) à (F)

Combined sequence of U3 :
(C,E) à (A,F) à (A,B,D)

Table1: Combined sequence for U1

Table2: Combined sequence for U2

Sequence	Session	A B C D E F
1	3	0 0 1 0 1 0
2	6	1 0 0 0 0 1
3	9	1 1 0 1 0 0

Table3: Combined sequence for U3

C. Method 1

Communicating parties :U1 and U4 (say).Sequence of AB and D are as follows :
AB=2, D=3.Therefore x1=2 and x2=3

Therefore U1 will compute $((\text{A.M.of 2and3})*(\text{H.M. of 2and3}))^{1/2}$ and U4 will compute G.M. of 2and3.So, shared key= $6^{1/2}$. If any occurrence become null, then that parameter value is treated as zero.

D. *Method 2*

Communicating parties : U3 and U4 (say)
In case of U3, Union becomes C E A F B D
So, shared key of U3 and U4 is C E A F B D

E. Method 3

Communicating parties : U2 and U4 (say)
Shared key is based on intersection and it is F.

XII . SHARED KEY USING FEATURE BASED METHOD

Let six messages are to be sent by the sender and those have to be encrypted by combination of one or more keys using some function.

message	Keys associated
M1	SK1 = (K1,K3,K4,K6)
M2	SK2 = (K3,K5)
M3	SK3 = (K4,K5,K6)
M4	SK4 = (K2,K3,K5)
M5	SK5 = (K1,K2)
M6	SK6 = (K1,K2,K3,K6)

Table 4 : Association of keys against each message

Key	Initial value	Count	Value	(Value)^2
K1	0.1	3	0.3	0.09
K2	0.2	3	0.6	0.36
K3	0.3	4	1.2	1.44
K4	0.4	2	0.8	0.64
K5	0.5	3	1.5	2.25
K6	0.6	3	1.8	3.24

Table 5 : Determination of count and value

Now CF = (x , y , z)
where x = number of elements , y = linear sum of the elements and z = sum of the square of the elements

CF1 = (4 , 4.1 , 5.41)
CF2 = (2 , 2.7 , 3.69)
CF3 = (3 , 4.1 , 6.13)
CF4 = (3 , 3.3 , 4.05)
CF5 = (2 , 0.9 , 0.45)
CF6 = (4 , 3.9 , 5.13)
So CFnet = accumulation of maximum of each tuple = (4 , 4.1 , 6.13)
So shared key = floor of modulus of (4.1 – 6.13) = 2

XIII. CONCLUSION

The papers shows how efficiently a neuro-fuzzy approach can be used for information processing. Also it has been shown how encryption schemes can be generated using SKEY , Interlock protocol , SKID, propositional logic , fuzzy operators , optimization technique. We have also demonstrated how the several data mining techniques can be suitably utilized for multi-party cryptosystem.

REFERENCES

[1] Chakrabarti P., Goswami P.S., "Approach towards realizing resource mining and secured information transfer" International Journal of Computer Science and Network Security", pp. 345-350 Vol. 8, No. 7, July 2008. ISSN:1738-7906

[2] Chakrabarti P., Goswami P.S. "An intelligent scheme towards information retrieval" accepted for publication in Asian Journal of Information Technology, 2008. ISSN: 1682-3915, Article ID: 706-AJIT

[3] Chakrabarti P., "Attacking Attackers in Relevant to Information Security", published in COIT08 , RMIT-IET, Mandhi Gobindgarh ,Mar 2008

[4] Chakrabarti P. et. al. ., "Approach towards key generation in multi-party communication and computational complexity of RSA algorithm" published in NCET-08, Integral University, Lucknow, Mar 2008

[5] Chakrabarti P., "Analysis of Cryptic data mining", published in International conference on Emerging Technologies and Applications in Engineering, Technology and Sciences , Rajkot ,Jan 2008

[6] Chakrabarti P., "Shared key evaluation in multiparty communication" published in International Conference on IT, Jabalpur, Dec2007

[7] Chakrabarti P., "Intelligent schemes of neural, cervical cipher generation and congestion control" published in NCSCA-07, ANITS , Vishakapatnam Dec 2007

[8] Pujari A.K. , "Data Mining Techniques", University Press, 2001