HYBRID BLRC ALGORITHM FOR ENCRYPTION

Hevisha Shah1, Rachana Patel2
1,2Assistant Professor, 1,2L.D. College of Engineering (I.T.), Gujarat, India

Abstract: Faster data transfer and Security both are very important for Wi-Fi. At present, Advanced Encryption Standard (AES) is used for Wi-Fi that is more secured than other encryption algorithms. Blowfish is a faster encryption algorithm but it cannot apply on Wi-Fi because of security problems. Main aim of this research is proposing the hybrid algorithm of Blowfish and Rivest Cipher 6 (RC6) that solves the security problems of Blowfish and maintain the fastness of Blowfish.

Keywords: Blowfish, RC6, Collision key attack, Known plaintext attack, reflectively weak key attack (key words).

I. INTRODUCTION

Wi-Fi is a very popular technology but security is a matter of great concern for the field of Wi-Fi. Among many security processes, Cryptography is very popular network security process where the message of any formats is converted into an encrypted version that is unreadable by a human or computer. There are two types of cryptography Algorithms are found: one is Symmetric key cryptography where same key is used for both encryption and decryption (e.g. AES, Blowfish, RC6) and the another is Asymmetric key cryptography where different keys are used for encryption and decryption (e.g. RSA). At present, AES [2] Symmetric key encryption algorithm is used for Wi-Fi network security but it is not so fast. On the other hand, Blowfish algorithm is so fast but it has some security problems. In this paper, a 128 bit hybrid algorithm of Blowfish and RC6 is proposed that removes the security problems of Blowfish and also take less Encryption decryption time than AES.

A. Blowfish

In [1], Blowfish is 64 bit symmetric key algorithm which contains eighteen 32 bit sub keys and four 32bit S-boxes with 256 entries each. The main function of it is given below: The Encryption process of Blowfish is , The input is a 64-bit data element, x. Divide x into two 32-bit halves: xL, xR. Then, for i= 1 to 16

XL = XL XOR Pi
XR = F(XL) XOR XR Swap XL and XR

After the sixteenth round, swap XL and XR again to undo the last swap. Then,

XR = XR XOR P17
XL = XL XOR P18.

Finally, recombine XL and XR to get the cipher text.

The F function is: F(XL) = ((S1,A + S2,B mod 2^32) XOR S3,C) + S4,D mod 2^32. Here 64 bit is divided among A,B,C,D registers where each of the register contains 8 bit.

B. RC6

In [3], RC6 is a 128 bit symmetric key encryption algorithm. The procedure is given below:

Input: Plaintext is stored in four w-bit input registers A,B,C,D .Number r of rounds. w-bit round keys S[0,…….,2r+3]. Output: cipher text is stored in A,B,C,D. Procedure: B=B+S[0] D=D+S[1] For i=1 to r do

{ t=(B+(2B+1)) <<< lgw u=(D+(2D+1))<<< lgw A=((A XOR t) <<< u)+S[2i] C=((C XOR u) <<< t)+S[2i+1] (A,B,C,D) =(B,C,D,A)

} A = A+S[2r+2] C =  C+S[2r+3]

II. RELATED WORKS

In[5], The B-R algorithm is also a mixer of Blowfish and RC6. It is also a 128 bit algorithm and the algorithm uses two S-boxes with 259 entries each. But this algorithm’s time complexity is too large because in every iteration it uses two functions: one is Blowfish function and the other is RC6 function and it also contains the risk of Reflectively weak key attack and collision key attack for using the same function and two s-boxes.

Figure 1-B-R Algorithm for Encryption

Figure 2-B-R Algorithm for Decryption

Avalanche Effect of Existing Algorithm(B-R Algorithm)

Input: hevisagopalbhais
Key:15
Changed Input: ievisagopalbhais
III. PROPOSED ALGORITHM

A possible faster and secure encryption algorithm is proposed here: In this proposed algorithm, 128 bit block of plaintext will be used as input. Here sub key generation of blowfish is used for making cipher text more powerful against brute force attack. The p-array consists of eighteen 64 bit sub keys from p1, p2……p18. Here will be used one 64bit s-box with 263 entries for substitution purpose. F(XL) that is used for Blowfish encryption function for finding next XR will be adjusted here with one s-box. Using one s-box can able to risk of collision attack between more than one S-boxes. 1-16 round of iteration is divided between Blowfish and RC6 modified using a variable “a”. The value of ‘a’ is only known by sender and receiver. This type of variation will be able to reduce the risk of reflectively weak key attack[4]. The rotation number ‘w’ that is used at the portion of RC6 is also a variable that only known by sender and receiver.

Proposed Algorithm Input: Plaintext 128-bits P
Output: Cipher text 128-bits C. P-array consists of 18 64 bits subkeys P1,p2……p18 
Split plaintext into two 64-bit halves: XL, XR For I= 1 to a 
Do 
XL=XL XOR P[I] 
XR = F (XL) XOR XR Swap XL and XR
For I= a to 16 Do 
u= (XR × (2XR +1)) <<<w 
XL= (XL <<< u) + P [I] XR= XR + P [I] 
Swap XL and XR 
XR = XR XOR P17 and XL = XL XOR P18 End.

F function of Proposed Algorithm
Input: XL, (64-bits)
Example: If XL (64-bits) contains 0X11083ae37809123 in hexadecimal then it is devided into eight parts 
a=0X11, b=0X08, c=0X3a, d=0Xeb, e=0X47, f=0X80, g=0X91, h=0X23.
Z2=(((S-box1[47] + S-box1[80]) MOD 2^32) XOR S-box1[91]) + S-box1[23]) MOD 2^32 Combined Z1, Z2 into Z.
Output: Z ( 64 bits).

Implementation of proposed Algorithm

Figure 5-Flowchart of Proposed Algorithm

Figure 6-Hybrid BLRC Algorithm for Encryption

Figure 7-Hybrid BLRC Algorithm for Decryption
### IV. RESULT ANALYSIS

The proposed and Existing Algorithm are implemented in C#.

<table>
<thead>
<tr>
<th>Size of Data (bytes)</th>
<th>B-R Algorithm Execution Time</th>
<th>BLRC Algorithm Execution Time</th>
</tr>
</thead>
<tbody>
<tr>
<td>1024</td>
<td>0.127</td>
<td>0.124</td>
</tr>
<tr>
<td>2048</td>
<td>1.010</td>
<td>0.661</td>
</tr>
<tr>
<td>3072</td>
<td>2.586</td>
<td>2.155</td>
</tr>
<tr>
<td>4096</td>
<td>9.974</td>
<td>6.473</td>
</tr>
<tr>
<td>5120</td>
<td>14.002</td>
<td>11.330</td>
</tr>
<tr>
<td>6144</td>
<td>34.111</td>
<td>25.132</td>
</tr>
</tbody>
</table>

Table 3. Comparison of Execution time

![Bar Chart](image-url)
V. CONCLUSION

Proposed algorithm produce a faster algorithm like blowfish and more secure than it. The proposed algorithm improves the faster algorithm Blowfish by adding the edited function of RC6 and removing it’s different attacks. It also uses “a” and “w” random variable to confuse the intruders by making different cipher text. It’s one S-box criteria makes the time complexity little higher than Blowfish but it reduces the memory requirement.

REFERENCES


