**Отчет по лабораторной работе №2**

Выполнила студентка 4 курса 9 группы

Стащенко Ксения

**Принцип работы разработанного клиент-серверного приложения**

Клиент имеет возможность сгенерировать открытый и закрытый ключ RSA, установить соединение с сервером, прервать соединение с сервером и отправить серверу запрос с именем файла и открытым ключом. Сервер же хранит текстовые файлы, генерирует сеансовый ключ по запросу клиента, отправляет клиенту сеансовый ключ и зашифрованный текстовый файл.

Клиент отправляет логин и пароль серверу и, если они верны, пользователь должен ввести полученный код, сгенерированный отдельным классом для корректной аутентификации и ждет ответа сервера. Сервер получает открытый ключ, генерирует ключ сессии, получает секретный код и отправляет ответ об авторизации(разрешение или нет). После успешной авторизации клиент может отправить запрос серверу на получение файла или закрытие канала. Впоследствии сервер и клиент осуществляют взаимодействие с помощью алгоритма Serpent с режимом обратной связи по выходу OFB.

**Принцип работы алгоритма Serpent.**

*B*0 = *НП*(*P*) ,

*Bi*+1 = *Ri*(*Bi*) для *i*= 0,1,…, 31 ,

*C*= *КП*(*B*32 ) ,

где

*Ri*( *X*) = *ЛП*(*Si*mod8 ( *X Ki*)) для *i*= 0,1,…, 30 ,

*R*31 ( *X*) = *S*7 ( *X K*31 ) *K*32 .

Алгоритм использует начальную подстановку НП и конечную подстановку КП, которая является обратной к НП.  
Используемые в алгоритме *S-box*преобразуют четыре входных бита в четыре выходных. Всего используется восемь различных *S-box* . Для обработки 128-битного информационного блока в *i*-цикле ( *i*= 0, …, 31 ) используются 32 одинаковых *S-box*— *Si*mod8 , то есть, для каждых четырех бит цикла используется одна и та же *S-box*. Все циклы имеют одну и ту же схему, за исключением последнего, в котором вместо линейного преобразования *ЛП*используется операция сложения по модулю два с блоком ключа *K*32 .

Для создания *S-box*используется матрица *sbox*[ , ], состоящая из восьми *S-box*алгоритма DES. Затем в *l*-й строке выполняется перестановка местами *i*-го и *k*-го элементов, при этом значение *k*определяется значением *j*-го элемента в *l*+1 строке. При этом *j*определяется вектором ключа и номером *i*. Эта процедура при последовательном увеличении *l*повторяется до тех пор, пока не будут получены все 8 *S-box*алгоритма.

128-битное слово *Bi*, полученное после выполнения преобразований в блоке из 32 *S-box*, состоит из четырех 32-битных слов *X*0 , *X*1 , *X*2 , *X*3 , которые и подвергаются линейному преобразованию:

*X*0 = ЦСЛ( *X*0 ,13) ;

*X*2 = ЦСЛ( *X*2 , 3) ;

*X*1 = *X*1 *X*0 *X*2 ;

*X*3 = *X*3 *X*2 СЛ( *X*0 ,3) ;

*X*1 = ЦСЛ( *X*1,1) ;

*X*3 = ЦСЛ( *X*3 , 7) ;

*X*0 = *X*0 *X*1 *X*3 ;

*X*2 = *X*2 *X*3 СЛ( *X*1, 7) ;

*X*0 = ЦСЛ( *X*0 ,5) ; *X*2 = ЦСЛ( *X*2 , 22) .

Полученные 32-битные блоки формируют 128-битный блок *Bi*+1 :

*Bi*+1 = *X*0 , *X*1, *X*2 , *X*3 ,

СЛ( , ) — сдвиг влево, при этом освобождающиеся разряды заполняются нулями.

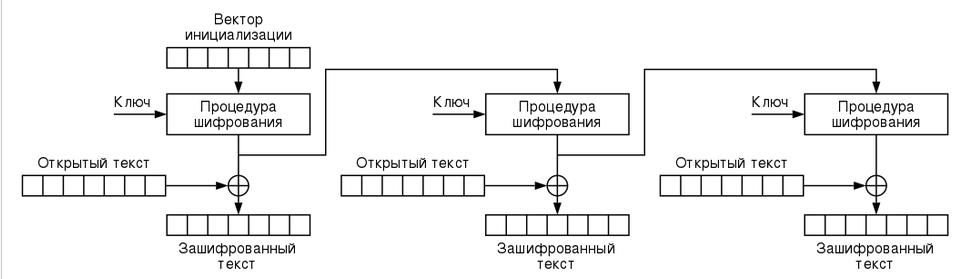
Для генерации ключей: ключ расширяется до 256 бит. Затем из полученного таким образом массива, элементами которого являются слова *w*−8 , *w*−7 , ..., *w*−1 получается промежуточный ключ *w*0 , *w*1 , ..., *w*131 по рекуррентной формуле: *wi*= ЦСЛ(*wi*−8 *wi*−5 *wi*−3 *wi*−1 *φ i*, 11) ,

| где *φ*— 9E3779B9 в шестнадцатеричной форме. |
| --- |
| форме 9E3779B9. Для получения очередного значения *wi* используется многочлен *x*8 + *x*7 + *x*5 + *x*3 +1 |  |

Ключи для циклов алгоритма Serpent вычисляются из промежуточных ключей с использованием *S-box*. Сначала генерируется 132 вспомогательных ключевых блока *k j*по 32 бита каждый, из полученных таким образом *k j*и формируются 128-битные ключи *Ki*для каждого цикла алгоритма:

*Ki*= *IP*({*k*4*i*, *k*4*i*+1, *k*4*i*+2 , *k*4*i*+3}) , для *i*= 0,…, 31.

**Схема шифрования в режиме OFB** определяется работает следующим образом:



**Скриншот работы программы** представлен ниже

