**МОЛДАВСКИЙ ГОСУДАРСТВЕННЫЙ УНИВЕРСИТЕТ**

**Факультет Математики и Информатики**

**Департамент Информатики**

Лабораторная работа № 2

Криптография

Алгоритм хэширования whirlpool

Проверил: профессор Gheorghe Latu

Выполнил: Mamaliga Artur grupa I2302-1

Кишинев, 2024

Алгоритм хеширования Whirlpool

# Введение

Алгоритм криптографического хеширования Whirlpool был разработан Пауло С. Л. М. Баррето и Винсентом Рийменом. Он основан на модифицированной версии стандарта шифрования AES и работает с сообщениями произвольной длины, производя хеш фиксированного размера — 512 бит. Whirlpool используется в различных приложениях, связанных с безопасностью, благодаря своей высокой криптостойкости.

# Математические концепции

## Поле Галуа

Алгоритм Whirlpool выполняет операции в конечном поле Галуа GF(2^8), что играет ключевую роль при выполнении операций, таких как сложение и умножение в этом поле. В GF(2^8) элементы представляются как 8-битные векторы, и арифметические операции выполняются по модулю неприводимого многочлена, обычно x^8 + x^4 + x^3 + x + 1.

Формально, операции умножения и сложения в поле Галуа могут быть выражены следующим образом:  
  
Сложение: a + b = c, где каждая операция выполняется побитово.  
Умножение: a \* b = c (мод x^8 + x^4 + x^3 + x + 1).

Пример из кода Whirlpool, где выполняются арифметические операции над данными в поле Галуа:

static const u64 C0[256] = {  
 LL(0x18186018c07830d8), LL(0x23238c2305af4626), LL(0xc6c63fc67ef991b8), LL(0xe8e887e8136fcdfb),  
 ...  
 LL(0xffffdbffab24e354), LL(0x7a7af57af78ff48d), LL(0x90907a90f4ea3d64), LL(0x5f5f615fc23ebe9d),  
 };

## S-box (Таблица подстановок)

S-box (таблица подстановок) представляет собой нелинейное преобразование, применяемое к каждому байту данных. Whirlpool использует таблицу подстановок 8x8, оптимизированную для аппаратных реализаций. Каждый байт из матрицы состояния заменяется на соответствующее значение из таблицы подстановок.

В математическом смысле S-box можно выразить через функцию замены байта:  
S(b) = T[b], где T — это таблица подстановок, а b — байт, подлежащий замене.

Пример замены байтов с использованием S-box в коде:

for (i = 0; i < 256; i++) {  
 S[i] = C0[i] ^ C1[i] ^ C2[i] ^ C3[i];  
 }

## Перестановки

Whirlpool выполняет серию перестановок строк и столбцов для рассеивания входных данных по матрице состояния. Эти операции помогают достичь так называемого эффекта лавины, когда небольшое изменение во входных данных приводит к значительным изменениям на выходе.

Перестановки могут быть выражены как сдвиг строк и столбцов в матрице состояния.

## Линейная диффузия

Шаг линейной диффузии использует умножение матрицы над полем Галуа GF(2^8), чтобы гарантировать равномерное распределение битов по всей матрице. Это значительно усиливает эффект лавины и делает алгоритм более устойчивым к атакам на коллизии.

Пример выполнения диффузии в коде:

static void diffusion(u64 state[8]) {  
 int i;  
 for (i = 0; i < 8; i++) {  
 state[i] ^= C0[i] ^ C1[i] ^ C2[i] ^ C3[i];  
 }  
 }

# Этапы хеширования Whirlpool

## Генерация ключа

На этапе генерации ключа входные данные XOR'ятся с раундовым ключом, который получается из сообщения. Эта операция выполняется перед началом основных раундов и вносит нелинейность.

Пример кода, в котором ключ используется для добавления к состоянию:

for (i = 0; i < 8; i++) {  
 state[i] ^= key[i];  
 }

## Функция сжатия

Whirlpool использует функцию сжатия, основанную на блочном шифре, который обрабатывает данные в серии из 10 раундов. Каждый раунд включает операции подстановки, перестановки и диффузии, которые последовательно преобразуют входные данные.

Пример кода для одного из раундов функции сжатия:

for (i = 0; i < 10; i++) {  
 add\_key(state, round\_keys[i]);  
 sub\_bytes(state);  
 shift\_columns(state);  
 diffusion(state);  
 }

# Заключение

Whirlpool является мощным криптографическим инструментом для генерации хешей. Его сложная структура, включающая таблицы подстановок, перестановки и диффузию, обеспечивает высокий уровень безопасности против криптоатак. Проведенный анализ алгоритма и его реализации в коде позволяет лучше понять его работу и криптографические свойства.