

МИНИСТЕРСТВО НАУКИ И ВЫСШЕГО ОБРАЗОВАНИЯ РОССИЙСКОЙ ФЕДЕРАЦИИ
федеральное государственное автономное образовательное учреждение высшего образования
«САНКТ-ПЕТЕРБУРГСКИЙ ГОСУДАРСТВЕННЫЙ УНИВЕРСИТЕТ
АЭРОКОСМИЧЕСКОГО ПРИБОРОСТРОЕНИЯ»

КАФЕДРА № 51

ОТЧЕТ
ЗАЩИЩЕН С ОЦЕНКОЙ
ПРЕПОДАВАТЕЛЬ

ассистент

должность, уч. степень, звание

подпись, дата

М.Н. Исаева

инициалы, фамилия

ОТЧЕТ ПО ЛАБОРАТОРНОЙ РАБОТЕ

АЛГОРИТМИЧЕСКИЕ ХЕШ-ФУНКЦИИ
по курсу: КРИПТОГРАФИЧЕСКИЕ МЕТОДЫ ЗАЩИТЫ ИНФОРМАЦИИ

РАБОТУ ВЫПОЛНИЛ

СТУДЕНТ ГР.

5912

подпись, дата

В.И.Исаева

инициалы, фамилия

Санкт-Петербург
2022

Задача

Вариант 12.

Реализовать алгоритм электронной цифровой подписи DSA.

Требования к работе:

- При постановке подписи использовать хеш-функцию MD5.
- Реализовать атаку на систему DSA в случае, если стал известен случайный параметр k .

Тестируемый алгоритм

DSA представляет собой криптографический алгоритм с использованием закрытого ключа для создания электронной подписи, но не для шифрования. Подпись создается секретно (закрытым ключом), но может быть публично проверена (открытым ключом). Это означает, что только один субъект может создать подпись сообщения, но любой может проверить её корректность. Алгоритм основан на вычислительной сложности взятия логарифмов в конечных полях.

Описание алгоритма

DSA включает в себя два алгоритма (S , V): для создания подписи сообщения (S) и для ее проверки (V).

Оба алгоритма вначале вычисляют хеш сообщения, используя криптографическую хеш-функцию MD5. Алгоритм S использует хеш и секретный ключ для создания подписи, алгоритм V использует хеш сообщения, подпись и открытый ключ для проверки подписи.

Стоит подчеркнуть, что фактически подписывается не сообщение (произвольной длины), а его хеш (128 бит), поэтому неизбежны коллизии и одна подпись, вообще говоря, действительна для нескольких сообщений с одинаковым хешем.

Параметры схемы цифровой подписи:

1. Хеш-функция – MD5
2. Простое число q размерностью 128 бит
3. Простое число p , такое, что $(p-1)$ делится на q
4. Число $g = 2^{(p-1)/q}$

5. Секретный ключ $x \in (0; q)$
6. Открытый ключ $y = g^x \bmod p$

Подпись сообщения m :

1. Выбор случайного числа $k \in (0; q)$
2. Вычисление $r = (g^k \bmod p) \bmod q$
3. Выбор другого k , если $r = 0$
4. Вычисление $s = k^{-1}(MD5(m) + x * r) \bmod q$
5. Выбор другого k , если $s = 0$
6. Подписью является пара (r, s) общей длины $2N$

Проверка подписи:

1. Вычисление $w = s^{-1} \bmod q$
2. Вычисление $u_1 = MD5(m) * w \bmod q$
3. Вычисление $u_2 = r * w \bmod q$
4. Вычисление $v = (g^{u_1} * y^{u_2} \bmod p) \bmod q$
5. Подпись верна, если $v=r$

Пример использования алгоритма

Исходное сообщение: «sign»

```
q = 270069038677997592033695731025064434477
p = 13166691828078804146195296381046017440853320051791679313527650100635102845257698288744364711355938801543913644020125646474073661196849815934677298670992467
g = 12238650980896238355305173443595273376825031421316715135079056635838355209037557517099173190765679608853782684270357750402200333000975846277739060160153955
open key = 1331957369139238413418613915583134728654413012631537706136450996114358132467309195318365329716766614133344517170962725554998433184842396175076967088941671
private key = 7167067600072887826172321568769705036
sign 253031700470184288550248677308632059468 157344976583155211942179495682954414480
```

Проверим подпись:

sign is correct

Изменим подпись и проверим ее:

```
sign 253031700470184288550248677308632059469 157344976583155211942179495682954414480
sign is not correct
```

Атака

Когда случайный параметр k становится известен, узнать секретный ключ становится достаточно

просто:

$$xr \bmod q = ks - MD5(m) \bmod q$$

finded x = 7167067600072887826172321568769705036

Вывод

В данной лабораторной работе был реализован алгоритм цифровой подписи DSA, по которому была проведена атака при известном секретном ключе k . Тем не менее, если все секретные параметры остаются секретными, взломать данный алгоритм не так просто, а его стойкость увеличивается с увеличением длины хеша и простых чисел p и q .