Цель

Реализовать алгоритм электронной цифровой подписи DSA. При постановке подписи использовать хеш-функцию SHA2. Реализовать атаку на систему DSA в случае, если атакующий получил две подписи, при создании которых использовалось одно и то же значение k.

1. Алгоритм

DSA — криптографический алгоритм с использованием закрытого ключа (из пары ключей: открытый и закрытый) 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.

2. 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 SHA2. 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 (256 бит), поэтому неизбежны коллизии и одна подпись, вообще говоря, действительна для нескольких сообщений с одинаковым хешем.

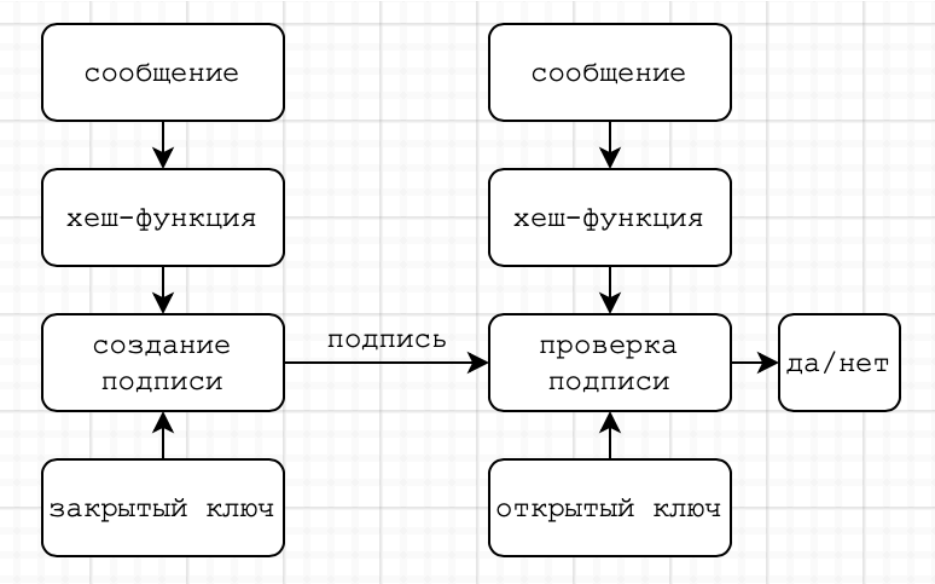

Рисунок 1. Иллюстрация работы алгоритма DSA

3. Параметры схемы цифровой подписи

1. Хеш-функция – SHA2
2. Простое число q размерностью 256 бит
3. Простое число p, такое, что (p-1) делится на q
4. Число $g = 2^{(p-1)/q}$
5. Секретный ключ x ϵ (0; q)
6. Открытый ключ $y = g^x mod\ p$

3.1. Подпись сообщения m:

1. Выбор случайного числа k ϵ (0; q)
2. Вычисление $r = (g^k mod\ p) mod\ q$
3. Выбор другого k, если r = 0
4. Вычисление $s = k^{-1}(SHA2(m) + x * r)\ mod\ q$
5. Выбор другого k, если s = 0
6. Подписью является пара (r, s) общей длины 2N

3.2. Проверка подписи:

1. Вычисление $w = s^{-1}\ mod\ q$

2. Вычисление $u_1 = SHA2(m) * w \ mod \ q$
3. Вычисление $u_2 = r * w \ mod \ q$
4. Вычисление $v = (g^{u_1} * y^{u_2} \ mod \ p) \ mod \ q$
5. Подпись верна, если $v = r$

4. Примеры работы программы

Таблица 1. Результат работы программы

q	57008909191164406479706701297555312662629216285303530325741150410195292544473
p	80511688033310948178445497258056825586652580294384725916217476744976260515065303747804705288674337741469509749581399129602919118490392480255963220377699333
g	74833257574597462181343839533804074276423180844497257627699147798667039745562391337970272792459268657057088555580250085968378586188016061146249133493604233
открытый ключ	78082388177133068921497364178429773636479591271548951828601737419985694904455084114674513939190582391872506165651341141231189384269735884948494664501639283
закрытый ключ	37329649249082264054059598327219224163116296592043618386802668109301138018824
подпись	r: 34580602341404303708238030990742109895998314865466413058538919593675884263413 s: 47196838163277508180573482455518989570890804642507501287587951509508187654908

Проверка подписи:

```
sign is correct
```

Изменим подпись:

Таблица 2. Результат работы программы

подпись	r: 34580602341404303708238030990742109895998314865466413058538919593675884263413 s: 47196838163277508180573482455518989570890804642507501287587951509508187654908

Проверим ее:

```
sign is not correct
```

5. Атака

Если для двух сообщений использовался один и тот же параметр k, тогда их подписи (*r, s*) будут иметь одинаковые *r*, но разные *s*, назовем их s_1, s_2.

Из выражения для *s* можно выразить общий *k*:

$$k = (H(m) \ + \ xr)s_1^{-1} \ mod \ q$$

И приравнять общий k для разных сообщений:

$$(H(m_1) + xr)s_1^{-1} \, mod \, q = (H(m_2) + xr)s_2^{-1} \, mod \, q$$

Отсюда легко выразить секретный ключ x:

$$x = \frac{H(m_1)s_1^{-1} - H(m_2)s_2^{-1}}{r(s_2^{-1} - s_1^{-1})}$$

Таблица 3. Результат работы программы

x	37329649249082264054059598327219224163116296592043618386802668109301138018824
закрытый ключ	37329649249082264054059598327219224163116296592043618386802668109301138018824

Вывод

В ходе выполнения данной лабораторной работы был реализован алгоритм цифровой подписи DSA, а также проведена атака, когда были получены две подписи, при создании которых использовалось одно и то же значение k. Повторение параметра для двух сообщений ведет к простому взлому системы.