Лекции

Краткий обзор развития методов криптографии

Первая криптография появилась в Древнем Египте.

Научные появились в Арабских Эмиратах.

Элементарная криптография

Под катом: Шифр Цезаря Шифр пар Шифр четырех квадратов Матричный шифр Шифр ADFGX Шифр Виженера Шифр Цезаря Каждую букву заменяют на третью (или N-ю) по алфавиту после нее. а б в г д е ё ж з и й к л м...

https://habr.com/ru/articles/116716/

1. Шифр сдвига:
   - Шифр Цезаря — сдвиг на 3 позиции:
     hello ⇒ khoor
   - Август — 1 позиция
2. Шифр замены
   Одному символу ставится в соответствие произвольный символ
3. Полиалфавитный шифр
   Используется несколько алфавитов замены.
   1 буква 1 алфавит, 2 буква 2 алфавит и т.д.
4. Таблица Тритемия
   Квадратная таблица. Каждая строка идёт со сдвигом в 1 позицию.
   1 буква 1 строка, 2 буква, та, которая соответствует 1 строке и т.д.
5. Шифр Виженера
   есть ключ ABC
   hello ⇒ ABCAC
   h e l l o
   шифр —

ключ	а	б	в
а			
б			
в			

6. Шифр Гронсфельда
   СЕКРЕТНОЕ
   Ключ: 1234
   С Е К Р Е Т Н О Е
   1 2 3 4 1 2 3 4 1
   У буквы происходит сдвиг на столько позиций, сколько соответствует значению её ключа.
7. Квадрат Полибия
   А Б В Г Д Е
   Ё Ж З И Й
   А И Г Д = 11 24 14 15
   11 → первая цифра строка; вторая цифра столбец
8. Перестановочные шифры
   Слово записывается в таблицу. Шифр представляет собой буквы из столбцов. Последовательность столбцов может меняться.

И Н Ф О Р М
А Ц И О Н Н
Ы Е

ИНФОРМАЦИОННОЕ
ИАЫ НЦЕ ФИ ОО РН НН

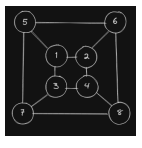
9. Магический квадрат
   Сумма по каждому столбцу, строке, диагонали равна определённому числу

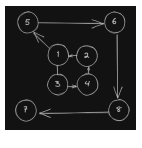
|16| 3| 2|13|
5	10	11	8
9	6	7	12
4	15	14	1

|О|И| Р|М
|З |Е| Д|Ю
|С|Ж|А|6
|Е|Г| О|Г

Приезжаю седьмого
ОИРМ ЗЕДЮ СЖАЬ ЕГОП
Составляется магический квадрат, после буквы из сообщения ставятся в соответствие этому квадрату. Шифр состоит из столбцов, квадраты из букв

10. Маршруты Гамельтона
    Буква ставится в соответсвие вершине. Шифр складывается по маршруту в графе





11. Решето Кардано
    $4 \times 4; 6 \times 6$
    Создаётся трафарет. В отверстия вписываются буквы. Трафарет поворачивается на 90, 180, 270.
    Полученные буквы записываются в таблицу.
    Шифр состоит из столбцов этой таблицы.

12. Одноразовые блокноты
    Сообщение + ключ
    Ш + Т = 25 + 19 = 44; 44-33=11
    Буква ставится в соответствие цифре. Значение буквы из сообщения + значение буквы из ключа = буква равная полученному числу.
    Если число больше 33, то - 33.

- Собственник информационных ресурсов — субъект с полномочиями владения, пользования и распоряжения информационными ресурсами
- Владелец информационных ресурсов — субъект с полномочиями владения и пользования информационных ресурсов
- Пользователь информационных ресурсов — субъект, обращающийся к инфорационной системе за получением необходимой ему информации и использующий её

Цель защиты информации — предотвращение ущерба собственнику, владельцу или пользователю информации.

Защита информации — комплекс правовых, огранизационных и технических мероприятий и действий по предотвращению угроз информационной безопасности и устранению их обработки и передачи информации в информационных системах.

Свойства информации:

- Конфиденциальность — известность её содержания только имеющим соответствующие полномочия субъектом.
- Целостность — неизменность информации в условиях её случайного и преднамеренного искажения или разрушения.
- Доступность — способность обеспечения беспрепятственного доступа субъектов к интересующей их информации

Угрозы информационной безопасности
— совокупность условий и факторов, создающих потенциальную или реальную существующую опасность нарушения безопасности информации.

Угрозы безопасности информации

1. Угрозы нарушения конфиденциальности
2. Угрозы нарушения целостности
3. Угрозы нарушения доступности информации

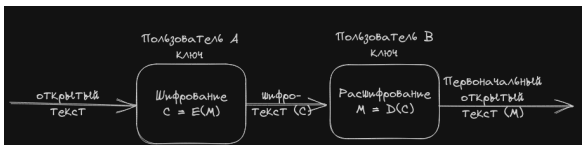
Методы и средства защиты информации:

- Законодательные методы защиты информации
  - 1-й уровень:
    - Международные конвенции
    - Конституция РФ
    - Кодексы РФ
      - Гражданский кодекс РФ
      - Уголовный кодекс РФ
    - Законы РФ
      - Об информации, информационных технологий по защите информации
      - О персональных данных
      - Об электронной подписи
      - О государственной тайне
      - О коммерческой тайне
  - 2-й уровень:
    - Указы Президента РФ
    - Постановление Правительства РФ
    - Письма Высшего Арбитражного Суда РФ
    - Постановление пленумов Верховного Суда РФ
  - 3-й уровень:
    - Государственные стандарты в области защиты информации (ГОСТы):
      - ГОСТ Р50922-96 “Защита информации. Основные термины и определения”
      - ГОСТ Р50739-95 “Средства вычислительной техники. Защита от несанкционированного доступа и информации. Общие технические требования”
      - ГОСТ 28147-89 “Системы обработки информации. Защита криптографическая. Алгоритс криптографического преобразования и другие”
    - Руководящие документы
  - 4-й уровень
    - Приказ об утверждении перечня сведений составляющих коммерческую тайну предприятия

- Трудовые и гражданские правовые договоры, в которые включены пункты об обязанности возмещения ущерба за разглашение сведений, составляющих коммерческую тайну предприятия и др.
- Административно-технические методы защиты информации:
  - Ограничение физического доступа к объектам комрьютерной системы и реализация режимных мер
  - Разграничение доступа к информационным ресурсам и процессам
  - Резервное копивание наиболее важных с точки зрения утраты массивов документов
  - Профилактика заражения компьютерными вирусами
  - Защита территорий и помещений от проникновения нарушителей
  - Организация доступа в помещение сотрудников
  - Защита аппаратных ___ и носителей информации от хищения
  - Предотвращение возможности удалённого видеонаблюдения за работой персонала
  - Контроль над режимом работы персонала
- Программно-аппаратные методы защиты информации
  - Устройства для ввода идентифицирующей пользователем информации
  - Устройства для шифрования информации
  - Устройства для восприпятствия несанкционированному включению рабочих станций и серверов
  - Программы идентификации и аутентификации пользователя
  - Программы разграничения доступа пользователей к ресурсам компьютерной системы
  - Программы шифрования информации
  - Программы защиты информационных ресурсов от несанкционированного изменения, использования и копирования

Криптографические методы и средства защиты информации

Основные понятия криптографии



$C = E_k(M)$ $M = D_k(C)$ $M = D_k(E_k(M))$

Попытка получить открытый текст соответствующий данному шифротексту при неизвестном ключе, называется дешифрацией. Лицо, которое пытается украсть текст, называется злоумышленником.

Криптографические операции позволяют зашифровать или расшифровать открытый текст.

Принцип Керкгоффса — неважен способ - важен ключ.

Классификация криптографических алгоритмов:

Алгоритмы шифрования:

- Семметричные
  - Блочные
  - Поточные
- Ассиметричные

Симметричные криптографические методы

Таблица частот

А	Б	В	Г	Д
Е,Ё	Ж	З	И	Й
К	Л	М	Н	О
П	Р	С	Т	У
Ф	Х	Ц	Ч	Ш
Щ	Ь,Ъ	Ы	Э	Ю
Я	<space>			

Принцип рассеивания и перемешивания

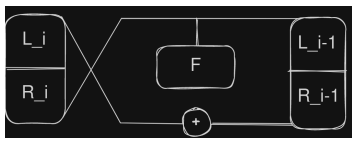
Рассеивание — это распространение влиятия одного знака открытого текста на много знаков шифротекста, что позволяет скрыть статистическое свойство открытого текста

Перемешивание — это использование таких шифрующих преобразований, которые усложняют восстановление взаимосвязи статистических свойств открытого и зашифрованного текста.

Схема одного раунда сети Фейстеля

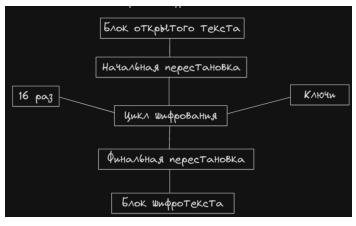
$L_i = R_{i+1} \oplus F(L_{i+1}, K)$

$R_i = L_{i-1}$

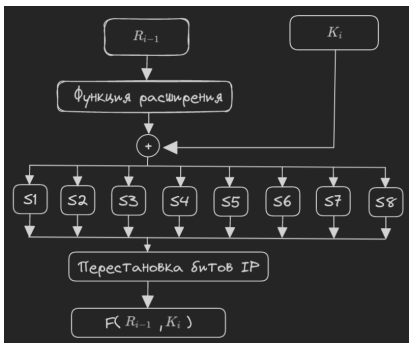


- Алгоритм шифрования TEA (Ting Encryption Algorithm)
  Размер блока — 64 бита
  Длина ключа — 128 бит
  Кол-во раундов — 32

- Алгоритм шифрования DES
  Размер блока — 64 бита
  Длина ключа — 56 бит
  Кол-во раундов — 16



- Преобразующая функция F



- Режимы выполнения алгоритмов симметричного шифрования
- Режим шифрования — это метод применения блочного шифра, позволяющий преобразовать последовательность блоков открытых данных в последовательность блоков зашифрованных данных.
  - Режим электронной кодовой книжки ECB

шифр-
$M = M_1, M_2, M_3, \dots$
$C_1 = (E_1(M_1))$
$i = 1, \dots, n$

Расшифровка
$M_i = D_k(C_i)$
$i = 1, \dots, n$

- Режим сцепления блоков

$C_i = E_k(M_i \oplus C_{i-1})$
$i = 1, \dots, n$
$C_0 = IV$

$M_1 = D_k(C_i) \oplus C_{i-1}$
$i = 1, \dots, n$
$C_0 = IV$

1. ШИФР_СДВИГА
   ЬМШФГХИЖМЗД
2. ДОГОВОР_В_СТАДИИ_ПОДПИСАНИЯ
   КФЙФИФЦЕИЕЧШЖКООЕХФКХОЧЖУОД
3. ЙУИУЗУХДТКДФУЙФНЦЕТ
   ДОГОВОР_НЕ_ПОДПИСАН
4. ТЖУКЖЦОУ
   МАНДАРИН
5. ЮБРМШУХ
   УЧЕБНИК
6. ТХЫЮАШЯХФ
   В ЛОС
7. ШИФР_ЗАМЕНЫ
   ЛФЦВШЮМИАЭЖ
8. ВЫЕЗЖАЮ_ПОЕЗДОМ
   ЫЖАЮРМПШЪДАЮУИ
9. СТУДЕНТЫ_ФСД
10. ИСПОЛЬЗУЕМ_ШИФР_ВИЖЕНЕРА
    ПШЫОТГУУМФЛШПЬЬ_ЙРТЕФНЬА
11. ШИФРШИФРШИФРШИФРШИФРШИФРШИФР
    ДОК У МЕНТ ЫВБЕЗ ОПАС НОММЕСТЕ
    ЬЦЮВГНАБТЗЦПЩНЫЮЖИДЭЕФУЬЭЩЕХ
12. ДЕКАРТДЕКАРТДЕКАРТДЕКАРТДЕК
    ЖЦЬРХИДДЫОАГТНЬСОСЖДЩОЫЦЙТЕ
    ВСТРЕЧА_СОСТОИТСЯ_В_ПОЛДЕНЬ
13. ЯСНОЕ
    _СТАН
    ОВИТС
    Я НЕП
    ОНЯТН
    ЫМ

    ЯСНОЕ
    СТАНО
    ВИТСЯ
    НЕПОН
    ЯТНЫМ

    ЯСНО СТИЕТ

    НАТПН ОНСОЫ

    ЕОЯНМ
14. Ы У Т Е
    Е К А Ы
    К Н Т Ф
    Д Р Д С

    ЫЕ_Д УКНРТАТД ЕЫФС
15. О Ф И Т
    - В А Р
    Ы С К Е
    Р С - Ш

    Р Е С
    А Т Е Е
    - Н И К
    Ж А Т Д
16. В С Т Р
    Е И А -
    В - д Е
    С Я Т Ь

    ВСТРЕЧА В 10

- CFB

Шифрование
$C_i = E_k(C_{i-1}) \oplus M_i$
$i = 1, \dots, n$
$C_0 = IV$

Расшифрование
$M_i = E_k(C_{i-1}) \oplus C_i$
$i = 1, \dots, n$
$C_0 = IV$

- OFB

Шифрование
$Z_i = E_k(Z_{i-1})$
$C_i = M_i \oplus Z_i$
$Z_0 = IV$

Расшифрование
$Z_i = E_k(Z_{i-1})$
$M_i = C_i \oplus Z_i$
$Z_0 = IV$

- ____ ГОСТ 28147-89
  Размер блока — 64 бита

Длина ключа — 256 бит
Кол-во раундов — 32

ш и ф р ш и ф р ш
с о о б щ е н и е
Очень похоже на схему Фейстеля
Генерация раундовых ключей
Раунд: 1 2 3 4 5 6 7 8 9 10 11 — 16
Подключ: 1 2 3 4 5 6 7 8 1 2 3 — 8

17 - 24 25 - 32
1 8 8 1

Таблица замены

4 10 9 2 13 8 0 14 6 11 1 12 7 15 5 3

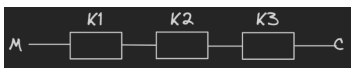
Вход: 0100
Выход: 1101

Различия DES и ГОСТ 28147-89

1. В DES применяется 56-битный ключ, а в ГОСТ — 256-битный
2. В DES 16 раундов, в ГОСТ — 32 раунда
3. DES более сложная процедура создания подключей
4. Таблицы замены DES имеют 6-битные входы и 4-битные выходы,
   таблица замены ГОСТ — 4-битный вход и выход ⇒ ГОСТ меньше
5. В DES нерегулярные перестановки
   В ГОСТ 11-битный цикл сдвига
6. ГОСТ удобнее для программной реализации

- Triple DES (3DES)

1. DES-EEE3



2. DES-EDE3

3. DES-EEE2

4. DES-EDE2

Расшифровывание — аналогично шифрованию, только ключи подаются в обратном порядке.

- DESX

Шифрование

$C_1 = E_1(M_i \oplus K_1) \oplus K_2$

Расшифрование

$M_2 = D_2(C_i \oplus K_2) \oplus K_1$

Размер ключа — 184 бита

- AES

Длина ключа — 128, 192, 256

Кол-во раундов — 10, 12, 14

128-битный блок делит на сегменты по 16 бит

$$S = \begin{pmatrix} m_0 & m_4 & 8 & 12 \\ m_1 & m_5 & 9 & 13 \\ m_2 & m_6 & 10 & 14 \\ m_3 & m_7 & 11 & 15 \end{pmatrix}$$

$$S = \begin{pmatrix} 0,0 & 0,1 & 0,2 & 0,3 \\ 1,0 & 1,1 & 1,2 & 1,3 \\ 2,0 & 2,1 & 2,2 & 2,3 \\ 3,0 & 3,1 & 3,2 & 3,3 \end{pmatrix}$$

4 преобразования

- Табличная замена каждого битового массива
- Сдвиг строк массива
- Операция над столбцами
- Операция с ключом

Раундовые ключи

- ГОСТ 34.12-2015

Размер блока — 128 бит

Длина ключа — 256 бит

Кол-во раундов — 10

Поточные шифры

Шифр построенный на основе формулы $C_i = m_i \oplus x_i, i = 1, 2, \dots$ называется поточным шифром.

C_i — элемент шифра
m_i — элемент открытого текста
x_i — элемент ключа
$m_i = c_i \oplus x_i, i = 1, 2, \dots$

- Линейный конгруэнтный генератор:

$x_i = (a * x_{i-1} + b)\ mod\ n$
x_i — i-й член последовательности
a — множитель
b — приращение
n — модуль
$(x_0 > 1), (a, b < n), (a, b, n > 0)$

Пример: $A = 5$, $B = 12$, $N = 23$, $x_0 = 4$
$x_1 = (5 * 4 + 12)\ mod_{23} = 32\ mod\ 23 = 9$

$x_2 = (5 * 9 + 12)\ mod_{23} = 57\ mod\ 23 = 11$
$x_3 = 11$
$x_4 = 2$
$x_5 = 22$
$x_6 = 7$
...
Генератор имеет период — 22

Генератор Парка-Миллера
$n = 2^{31} - 1 = 2147483647, n = 0$

- Квадратичный конгруэнтный генератор

$x_i = (a * x_{i-1}^2 + b * x_{i-1} + c)\ mod\ n$

- Кубический конгруэнтный генератор

$x_i = (a * x_{i-1}^3 + b * x_{i-1}^2 + c * x_{i-1} + d)\ mod\ n$

Достоинства и недостатки

- '+' — простота и высокая скорость получения псевдослучайных значений
- '-' — потенциальная возможность восстановления всей последовательности псевдослучайный чисел.

Алгоритм RC4

1. Подготовительный этап

```
for i=0 to 2^(n-1) do S[i] <- i
j <- 0
for i=0 to 2^(n-1) do
  j <- (j + S[i] + K[i mod L]) mod 2^n
  S[i] <-> S[j]
end for
```

2. Основной этап

```
i <- 0, j <- 0
while
  i <- (i+1) mod 2^n
  j <- (j + S[i]) mod 2^n
  S[i] <-> S[j]
  T <- (S[j] + S[i]) mod 2^n
  Z <- S[T]
end while
```

Пример:

i	j	S
	0	{0,1,2,3,4,5,6,7}
0	0+0+2	{2,1,0,3,4,5,6,7}
1	2+1+5=0	{1,2,0,3,4,…}
2	2	{1,2,0,3,4,5,6,7}
3	2+3+5≥2	{1,2,3,0,4,5,6,7}
…	…	…
1	2+7+5>6	{4,2,6,0,1,5,7,5}

Вычисление псевдослучайный последовательностей

i	j	S	$+$	z_i
0	0	{4,2,6,0,1,3,7,5}		
1	2	{4,6,2,0,1,3,7,5}	2+6=0	$z_1 = 4$
2	4	{4,6,1,0,2,3,7,5}	1+2=3	$z_2 = 0$
3	4	{4,6,1,2,0,3,7,5}	2+0=2	$z_3 = 1$
…	…	…	…	…
6	7+7=6	{4,6,1,2,0,5,7,3}	1+7=6	$z_6 = 1$

z = 100 00 001.100.100.111

Математические основы криптографических методов

- Модульная арифметика

Для любого числа a и любого натульрального n величина $a\ mod\ n$ есть остаток от деления a на n

Более строгое определение:
$a\ mod\ n = a - \left[\frac{a}{n}\right] * n$

Все числа можно представить: $a = b + k * n$
k — целое число;
b — целое число, принадлежащее множеству модуля n.

Основные свойства

Операции: сложение, вычистание, умножение.

1. $a\ mod\ a > 0$
2. $(a + b)\ mod\ n = ((a\ mod\ n) + (b\ mod\ n))\ mod\ n$
3. $(a - b)\ mod\ n = ((a\ mod\ n) - (b\ mod\ n))\ mod\ n$
4. $(a * b)\ mod\ n = ((a\ mod\ n) * (b\ mod\ n))\ mod\ n$

Выполняются законы коммутативности, ассоциативности и дистрибутивности.

Теоремы:

1. Целое положительное число p называется простым, если оно не делится ни на какое другое число, кроме самого себя и на единицу.
   a. Любое целое положительное число может быть представлено в виде произведения простых чисел, причём единственным образом.
   b. Два числа называются взаимно простыми, если они не имеют ни одного общего делителя, кроме 1.
   c. Пусть дано целое число N ≥ 1. Значение функции Эйлера $\varphi(N)$ равно кол-ву чисел в ряду $1, 2, 3, \ldots, N$ взаимнопростых с N
2. Если p-простое число, то $\varphi(p) = p - 1$
3. Пусть p и q — два различных числа ↔ простых $(p \neq q)$.
   Тогда $\varphi(p * q) = (p - 1)(q - 1)$
4. Для всякого простого числа p и натурального числа a $(0 < a < p)$ взаимно простого с p, имеет место равенство $a^{p-1}\ mod\ p = 1$
5. (Эйлер). Пусть a и n — натуральные взаимно простые числа. Тогда справедливо соотношение: $a^{p(n)}\ mod\ n = 1$
6. Пусть p и q — простые числа такие, что $p \neq q$ и a — натуральное число $(0 < a < p * q)$, то для произвольного числа k справедливо ссоотношение: $a^{k*\varphi(p*q)+1}\ mod\ (p * q) = a$

Шифр Гронсфельда

1. Вышлите_новое_задание
   3 1 4 7 3 1 4 7 3 1 4 7 3 1 4 7 3
   Е Ы Ь Т А У Й Ж Р П Ж У А Л З З Б С П И
2. Д Х Т Ъ Р Ю О В Л Ж Ю Т У Ц С Н У Ш
   2 7 1 6 2 7 1 8 2 7 1 8 2 7 1 8 2 7

В О С Т О Ч Н Ы Й _ Э К С П Р Е С С

3. П Р И М Е Р_Н А Д Ё Ж Н О Г О_Ш И Ф Р А
   М А Т Е М А Т И К А_Ц А Р И Ц А_Н А У К
   Ь С Ы Т Т С Т Ц А Е В Э О Я Н д А Ш ц Х Г Л

Наибольший общий делитель

Пусть a и b — два целых положительных числа. Наибольших общий делитель чисел a и b есть наибольшее число c, которое делится и на a, и на b
$c = \text{НОД}(a, b)$

Алгоритм Евклида

Вход: положительные целые числа a и b $(a > b)$

Выход: наибольший общий делитель ($\text{НОД}(a, b)$)

```
while b != 0 do
  r <- a mod b
  a <- b
  d <- b
end while
return a
```

Пример:

a	b	r
28	8	4
8	4	0
4	0	

Пусть a и b → два целых положительных числа. Тогда существуют целые числа x и y такие, что:

$$a * x + b * y = \text{НОД}(a, b) \quad (1)$$

Обобщённый алгоритм Евклида

Вход: a, b $(a > b)$

Выход: $\text{НОД}(a, b), x, y$, удовлетворяющие (1)

```
u <- (a,1,0); v <- (b,0,1)
while v1 != 0 do
  q <- u1 div v1
  T <- (u1 mod v1, u2 - q*v2, u3-q*v3)
  u <- v; v <- T
end while
return u=(НОД(a,b),x,y)
```

Пример:

				НОД	x	y	q
u				28	1	0	
v	u			19	0	1	1
T	v	u		9	1	-1	2
	T	v	u	1	-2	3	9
		T	v	0	19	-28	
			T	0	19	-28	

$$c * d \ mod \ m = 1 \quad (2)$$

5. Число d, удовлетворяющее уравнению (2), называется инверсией c по модулю m и часто обозначается $c^{-1} \ mod \ m$
   $c * c^{-1} \ mod \ m = 1$

Пример:
$c = 3, m = 11;$
$3 * 4 \ mod \ 11 = 1;$
$c^{-1} \ mod \ m = 4$

Равенство (2) означает, что для некоторого целого k (3)
$c * d - k * m = 1$

Учитывая, что c и m $(m > c)$ взаимнопростые, уравнение (3) можно переписать в виде (4)
$m * (-k) + c * d \geq \text{НОД}(m, c)$

Если получилось отрицательное числа, нужно прибавить значение модуля

Возведение степени по модулю

$a^x \ mod \ p$

Здесь a, x, p — целые числа

$y = a^x \ mod \ p$
$a^x = a * a * \ldots * a$
$x = b_{n-1} * 2^{n-1} + b_{n-2} * 2^{n-2} + \ldots + b_0 * 2^0$
$a^x = a^{b_{n-1} * 2^{n-1}} + a^{b_{n-2} * 2^{n-2}} + \ldots + a^{b_0 * 2^0} =$
$(a^{2^{n-1}})^{b_{n-1}} * (a^{2^{n-2}})^{b_{n-2}} * \ldots * (a^{2^0})^{b_0} =$
$(a^{2^0})^{b_0}, \ (a^{2^1})^{b_1}, \ (a^{2^2})^{b_2}$
$a^{2^0} = a^1 = a$
$a^2 = a^2 \ldots a^{2^2} = (a^{z^{z-1}})^2$

Возведение в степень (←)

Вход: $a, x = (b_i, b_{i+1}, \ldots, b_0)_2, p$
Выход: $y = a_x \ mod \ p$

```
y <- 1; s <- a
for i=0,1,...,x do
  if b_i = 1 then y <- y * s mod p
    s <- s * s mod p
end for
return y
```

Пример: $y = a^{100} \ mod \ p$
$x = 1100100$ в двоичном коде
$y = 1, s = a$

i:	0	1	2	3	4	5	6
x_i:	0	0	1	0	0	1	1

$y = a^{100} \ mod \ p$

y:	1	1	a^4	a^4	a^4	a^{36}	a^1
s:	a^2	a^4	a^8	a^{16}	a^{32}	a^{64}	a^{128}

Возведение в степень (→)

Вход: $a, x = (b_t, b_{t-1}, \ldots, b_0)_2, p$
Выход: $y = a^x \ mod \ p$

```
y <- 1; s <- a
for i=t,t-1,...,0 do
  y <- y*y mod p
  if b_i = 1 then y <- y * mod p
end for
return y
```

Пример: $y = a^{100} \ mod \ p$

i:	6	5	4	3	2	1	0
y_i:	1	a^2	a^6	a^{12}	a^{24}	a^{50}	a^{100}
x_i:	1	1	0	0	1	0	0
y:	a	a^3	a^6	a^{12}	a^{25}	a^{50}	a^{100}

Односторонняя функция

$$y = f(x) \quad (1)$$

$$x = f^{-1}(y) \quad (2)$$

6. Функция наызвается односторонней, если вычисление по формуле (1) является вычислительно простой задачей, не требующей много времени, а вычисление по формуле (2) — вычислительно сложной задачей, требующей привлечения больших вычислительныъ ресурсов.

Пример:
$y = a^x \ mod \ p$ односторонняя
$a^x = y \ mod \ p$ обратная
$x = log_a y \ mod \ p$ логорифмическая

Пример односторонней функции: y

Криптография с открытым ключом

- задача распределения ключей
- эффективный способ подтверждения о подлинности передаваемой информации.

Основные термины и требования

- Закрытый ключ — K_S (secret key)
- Открытый ключ — K_p (public key)
- Секретный ключ —

1. Вычислительно легко создать пару ключей — (K_p, K_S)
2. Вычислительно легко, имея открытый ключ и незашифрованное сообщение M, создать соответствующее зашифрованное сообщение. $C = E_{kp}(M)$
3. Вычислительно легко расшифровать сообщение, используя закрытый ключ $M = D_{ks}(C)$
4. Вычислительно невозможно, зная открытый ключ K_p, определить закрытый ключ K_S
5. Вычислительно невозможно, зная открытый ключ K_p и зашифрованное сообщение C, восстановить исходное сообщение M.

Основные способы использования

- форматирование ключа сессии
- шифрование / расшифрование
- создание и проверка подписи

Распределение ключей

Ключи распределяются: “каждый с каждым”
Всего ключей можно вычислить по формуле: $\frac{(n-1)*n}{2}$

Ключи должны:

1. Оптимальность, целостность
2. Конфиденциальность

Меняются ключи по 2 правилам:

1. Шифрование с открытым ключом для защиты секретного
2. Метод открытого распределения Диффи-Хеллмена

Диффи-Хеллмена
По открытому каналу передаются секретные ключи между двумя пользователями

Пример: пользователи $\mathcal{A}$ и $\mathcal{B}$ выбирают две величины p и g. p — большое число, а g — целое число меньшее p.

1. $\mathcal{A}: x_\mathcal{A} : (1 < x_\mathcal{A} < p-1)$
   $y_\mathcal{A} = g^{x\mathcal{A}} \mod p$
   $\mathcal{X}\mathcal{A}$ — закрытый ключ
2. $\mathcal{B}: x_\mathcal{B} : (1 < x_\mathcal{B} < p-1)$
   $y_\mathcal{B} = g^{x\mathcal{B}} \mod p$
3. $\mathcal{A}$ пересылает $y_A \ B$
4. $\mathcal{B}$ пересылает $y_B \ A$
5. $\mathcal{A}: k_\mathcal{A} = y_\mathcal{B}^{x\mathcal{A}} \mod p$
6. $\mathcal{B}: k_\mathcal{B} = y_\mathcal{A}^{x\mathcal{B}} \mod p$

Утверждение: значения, полученные пользователями $\mathcal{A}$ и $\mathcal{B}$ по алгоритму Д-Х равны между собой.

Доказательство:
$k_A = y_B^{xA} \mod p = (q^{xg} (\mod p))^{xA} (\mod p) = y^{xgxA(\mod p)} = y_A^{xg(\mod p)} = kB$

Пример: $p = 47$, $g = 23$

1. $\mathcal{A}: x_\mathcal{A} = 12 y_\mathcal{A} = 23^{12} \mod 47 = 27$
2. $\mathcal{B}: x_\mathcal{B} = 33 y_\mathcal{B} = 33^{12} \mod 47 = 33$
3. $\mathcal{A}: k_\mathcal{A} = 33^{12} \mod 47 = 25$
4. $\mathcal{B}: k_\mathcal{B} = 27^{33} \mod 47 = 25$

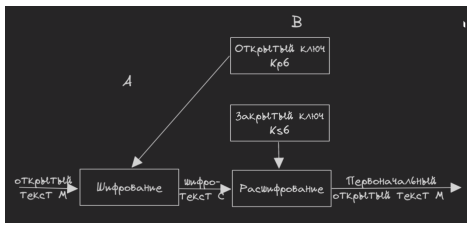
Чтобы взломать, злоумышленнику нужны формулы:

$g^{x_\mathcal{A}} = y_\mathcal{A} \mod p$ $\qquad$ $g^{x_\mathcal{B}} = g_\mathcal{B} \mod p$

$x_\mathcal{A} = log_g y_\mathcal{A} \mod p$ $\qquad$ $x_\mathcal{B} = log_g y_\mathcal{B} \mod p$

но эти вычисления слишком сложные

Шифрование с открытым ключом



- Алгоритм RSA

1. Формирование открытого и закрытого ключей
   - Выбрать p и q $(p \neq q)$
   - $n = p * q$
   - $(p-1)(q-1)$
   - Выбрать число e — взаимно простое: НОД$(e * (p-1) * (q-1)) = 1$
   - $d = (e * d) \mod (p-1) * (q-1) = 1$

   (e, n) — открытый ключ

   (d, n) — закрытый ключ
2. Шифрование

   $$\left.\begin{array}{l}\text{Открытый ключ — } (e,n)\\ \text{Открытое сообщение — } M\end{array}\right\} \Rightarrow \boxed{C = M^e \mod n} \Rightarrow \boxed{C - \text{шифротекст}}$$
3. Расшифрование

   $$\left.\begin{array}{l}\text{Зашифрованное сообщение — } C\\ \text{Закрытый ключ — } (d,n)\end{array}\right\} \Rightarrow \boxed{M = C^d \mod n} \Rightarrow \boxed{M - \text{открытый текст}}$$

Пример: $\mathcal{A}$ хочет передать $\mathcal{B}$ сообщение $M = 15$

1. Создание открытого и закрытого ключей

   $p = 3; q = 11$
   $n = p; q = 33$
   $e = 3; \text{НОД}(3, 20) = 1$
   $d = 7$
   $k_p = (3, 33); k_S = (7, 33)$
2. Шифрование
   $C = 15^3 \mod 33 = 9$
3. Расшифрование
   $M = 9^7 \mod 33 = 15$

- Алгоритм Шамира

1. Подготовительный этап
   - Выбрать общий параметр p
   - $\mathcal{A}$: Вычисляет C_A и D_A
     $(C_A * D_A) \mod (p-1) = 1$
   - $\mathcal{B}$: Выбирает C_B и D_B
     $(C_B * D_B) \mod (p-1) = 1$
2. Передача сообщения
   - $\mathcal{A}$: $x_1 = M^A C_A \mod p$
     $\mathcal{A} ==== x_1 ======== \mathcal{B}$
   - $\mathcal{B}$: $x_2 = x_1 C_B \mod p$
     $p ===== x_2 ====== \mathcal{A}$
   - $\mathcal{A}$: $x_3 = x_2 D_A \mod p$
     $\mathcal{A} ===== x_\mathcal{A} ====== \mathcal{B}$
   - $\mathcal{B}$: $x_4 = x_3 D_B \mod p$
     $M = x_4$

Пример: $\mathscr{A}$ передаёт $\mathscr{B}$ сообщение $M = 10$

Общий параметр $p = 23$
$\mathcal{A}: \mathcal{C}_\mathcal{A} = 7, d_\mathcal{A} = 19$
$\mathcal{B}: C_B = 5, d_B = 9$

1. $x_1 = 10^7 \mod 23 = 14$
2. $x_2 = 14^5 \mod 23 = 15$
3. $x_3 = 15^{19} \mod 23 = 19$
4. $x_4 = 19^9 \mod 23 = 10$

◯ 1

$x_1 = (4 * 1 + 7) \mod 19 = 11 \mod 19 = 11$
$x_2 = (4 * 11 + 7) \mod 19 = 51 \mod 19 = 13$
$x_3 = (4 * 13 + 7) \mod 19 = 2$
$x_4 = (4 * 2 + 7) \mod 19 = 15$
$x_5 = (4 * 15 + 1) \mod 19 = 10$
$x_6 = (4 * 10 + 7) \mod 19 = 9$
$x_7 = (4 * 9 + 7) \mod 19 = 5$
$x_8 = (4 * 7 + 7) \mod 19 = 8$
$x_9 = (4 * 6 + 7) \mod 19 = 1$
$x_{10} = 4 * 3 = 0$

Период — 9

◯ 2

$x_1 = (3 * 1 + 7) \mod 19 = 10$
$x_2 = (3 * 10 + 7) \mod 19 = 18$
$x_3 = (3 * 18 + 7) \mod 19 = 4$
$x_4 = (3 * 4 + 7) \mod 19 = 0$
$x_5 = (3 * 0 + 7) \mod 19 = 7$
$x_6 = (3 * 7 + 7) \mod 19 = 9$
$x_7 = (3 * 9 + 7) \mod 19 = 15$
$x_8 = (3 * 15 + 7) \mod 19 = 14$
$x_9 = (3 * 14 + 7) \mod 19 = 11$
$x_{10} = (3 * 11 + 7) \mod 19 = 2$
$x_{11} = (3 * 2 + 7) \mod 19 = 13$
$x_{12} = (3 * 13 + 7) \mod 19 = 8$

◯ 3

$x_1 = (3 * 5 + 4) \mod 17 = 2$
$x_2 = (3 * 2 + 4) \mod 17 = 10$
$x_3 = (3 * 10 + 4) \mod 17 = 0$
$x_4 = (0 * 10 + 4) \mod 17 = 4$
$x_5 = (3 * 4 + 4) \mod 17 = 16$
$x_6 = (3 * 16 + 4) \mod 17 = 1$
$x_7 = (3 * 1 + 4) \mod 17 = 7$
$x_8 = (3 * 7 + 4) \mod 17 = 8$
$x_9 = (3 * 8 + 4) \mod 17 = 11$
$x_{10} = (3 * 11 + 4) \mod 17 = 3$
$x_{11} = (3 * 3 + 4) \mod 17 = 13$
$x_{12} = (3 * 13 + 4) \mod 17 = 9$

$x_{13} = (3 * 8 + 7) \mod 19 = 12$
$x_{14} = (3 * 12 + 7) \mod 19 = 5$
$x_{15} = (3 * 5 + 7) \mod 19 = 3$
$x_{16} = (3 * 3 + 7) \mod 19 = 16$
$x_{17} = (16 * 3 + 7) \mod 19 = 17$
$x_{18} = (3 * 17 + 7) \mod 19 = 1$

$x_{13} = (3 * 9 + 4) \mod 17 = 14$
$x_{14} = (3 * 14 + 4) \mod 17 = 12$
$x_{15} = (3 * 12 + 4) \mod 17 = 6$
$x_{16} = (3 * 6 + 4) \mod 17 = 5$

Алгоритм RC4

◯ 1: $n = 3, k = 37 (L = 2)$
$S = \{0, 1, 2, 3, 4\}$

◯ 2 Шифр Эль-Гамаля

Формирование открытого
и закрытого ключей

Общие параметры — p,g

↓

X — случайное целое число($1<x<p$)

↓

$y=g^x \mod p$

↓

x – закрытый ключ
y – открытый ключ

Шифрование сообщения

Открытое сообщение — M
Открытые параметры — p,g
Открытый ключ — y

↓

$k-$ случайное число ($1<k<p$)

↓

$R=g^k \mod p$

↓

$E=(M^x * y^k) \mod p$

↓

Шифротекст - (R,E)

Расшифрование сообщения

Шифротекст - (R, E)
Общие параметры - p, g
Закрытый ключ - X

↓

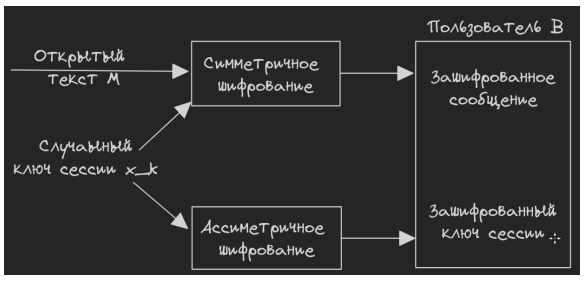
$M=E^x R^{p-1-x} \mod p$

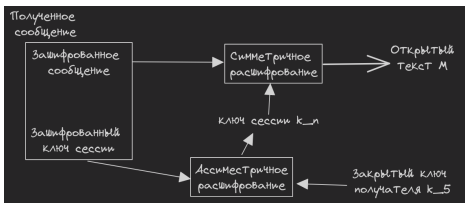
↓

Открытое сообщение - M

Пример: $M = 15$ от $\mathscr{A}$ к $\mathscr{B}$
$p = 23, g = 5$
$\mathscr{B}: x^B = 14;$
$y_B = 5^{13} \mod 23 = 21$
$\mathscr{A}: k = 7$
$R^7 \mod 23 = 17;$
$E = 15 * 21^7 \mod 23 = 15 * 10 \mod 23 = 12$
$\mathscr{B}: M = 12 * 17^{23-1-13} \mod 23 = 12 * 17^9 \mod 23 = 15$

Комбинированные криптосистемы





\+ Высокая скорость
\+ Высокая надёжность

Криптографические хеш-функции

Используется:

- При генерации
- При реализации электронной подписи

Хеш-функция — это необратимое преобразование данных, обладающая свойствами:

- на вход функции может поступать двоичный блок данных производной длины;
- на выходе функции получается двоичный блок данных фиксированной длины;
- значения на вызоде алгоритма распределяются пл равномерному закону по всему диапазону возможных результатов;
- при изменении хотя бы одного бита во входных данных хеш-функции её выход значительно изменяется

Требования:

- для любого заданного сообщения M вычисление хеш-функции $hash(M)$ должно выполняться относительно быстро
- для любого заданного хеш-значения y должно быть быть вычислительно невозможно найти M такое, что $hash(M) = y$
- для любого заданного сообщения M должно быть вычислительно невозможно найти другое сообщение $M_2 \neq M$ такое, что $hash(M_2) = hash(M)$
- вычислительно невозможно найти произвольную пару различных сообщений M и M_2, для которых $hash(M_2) = hash(M)$

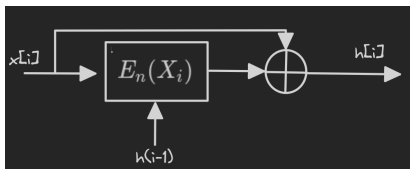
Итеративныя схема Меркле-Дамгаарда

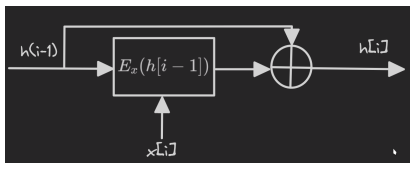
$$
\begin{array}{ccccccccccc}
 & & M_1 & & M_2 & & \dots & & M_n & & \\
 & & \downarrow & & \downarrow & & & & \downarrow & & \\
H_1 & \longrightarrow & F(H_1,M_1) & \longrightarrow & H_2 & \longrightarrow & F(H_2,M_2) & \longrightarrow & \dots & \longrightarrow & F(H_n,M_n) & \longrightarrow & H_n+1
\end{array}
$$

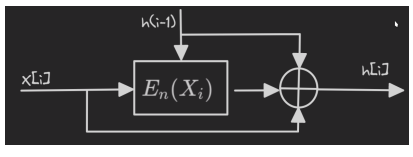
$$
\begin{array}{ccccc}
 & & M_i & & \\
 & & \downarrow n & & \\
H_i & \xrightarrow{k} & F(H_i,M_i) & \xrightarrow{k} & H_{i+1}
\end{array}
$$

Хеш-функция на основе блочных шифров

1. Схема Мейера-Матиаса:
2. Схема Девиса-Мейера:
3. Схема Миачури-Пренеля







Пример:

1. $h \leftarrow E_{n_{i-1}}(X_i) \oplus X_i, h_0 = 0$

   Алгоритм:

   h ← 0
   for i = 1,2,…,n do
   h ← $E_n(X_i) \oplus X_i$
   end for

2. $h \leftarrow E_{x_i}(h_{i-1}) \oplus h_{i-1}, h_0 = 0$

   Алгоритм:

   h ← 0
   for i =1,2,…,m do
   $h \leftarrow E_x(h) \oplus h$
   end for

Функция хэширования MD4

1. Исходный файл данных
2. Этап 1: Дополнение файла
3. Этап 2: Инициализация внутреннего буфера
4. Этап 3: Поблочная обработка входного файла данных
5. Этап 4: Формирование выходного значения

Этап 1

$$\underbrace{\text{Исходное сообщение}\ \underbrace{100\ldots000}_{\text{доп.биты}}\ \underbrace{\text{Длина исходного сообщения}}_{64\ \text{бита}}}_{\text{Дополненное сообщение, кратное } 532}$$

Этап 2

$H = (H_1, H_2, H_3, H_4)$
H_1 = ‘01 23 45 67’
H_2 = ‘89 AB CD EF’
H_3 = ‘FE DC BA 98’
H_4 = ‘76 54 32 10’

Этап 3

1. Загрузить очередной блок данных
2. $(A, B, C, D) = (H_1, H_2, H_3, H_4)$
3. Выполнить первый раунд
4. Выполнить второй раунт
5. Выполнить третий раунд
6. $(H_1, H_2, H_3, H_4) = (H_1 + AH_2 + B_1H_3 + C_1H_4 + D)$

Первый раунд

```
for i=0 to 15 do
  T = A + F(B,C,D) + M[Z_i] + K_i
  (A,B,C,D) = (D,T <<< S_i, B, C)
end for
```

где $F(B, C, D) = (B \wedge C) \vee (\neg B \wedge D))$

Второй раунд

```
for i=16 to 31
  F = G
```

где $G = (B, C, D) = (B \wedge C) \vee (B \wedge D) \vee (C \wedge D)$

Третий раунд

```
for i=32 to 47
  F=H
```

где $H = (B, C, D)$

Константы:

1. Сложение в раундах
   K_i = 0, 0 ≤ i ≤ 15
   5A827999’(16), 16 ≤ i ≤ 31
   6ED9EBAI’(16), 32 ≤ i ≤ 47

Функция хеширования MD5

Используются 4 логические функции
Длина сообщения — 128 бит
Кол-во раундов — 4
Кол-во шагов — 16

Функция хеширования SHA-1

Основан на функции MD4
Выходная строка — 160 бит
Кол-во раундов — 4
Кол-во шагов — 20

Функция хеширования RIPEMD-160

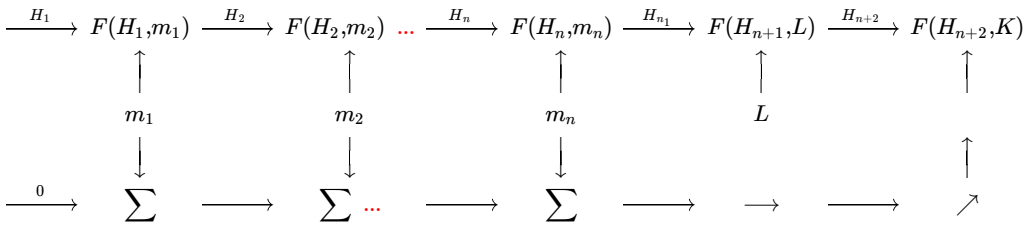
Основан на функции MD4
Выходная строка — 160 бит
Кол-во раундов — 5
Кол-во шагов — 16

Функция хеширования SHA-256(-512)

Выходная строка — 256 бит
Кол-во раундов — 64
Кол-во шагов — 1

Выходная строка — 512
Кол-во раундов — 80
Кол-во шагов — 1

Функция хеширования ГОСТ Р34.11-94



Функция хеширования ГОСТ Р34.11-2012

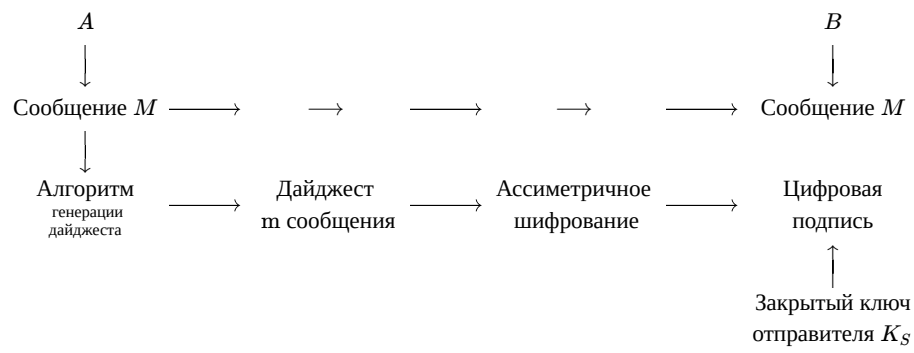
- Две функции хеширования с длинной хеш-кода 256 и 512 бит
- Размер блоков — 512 бит

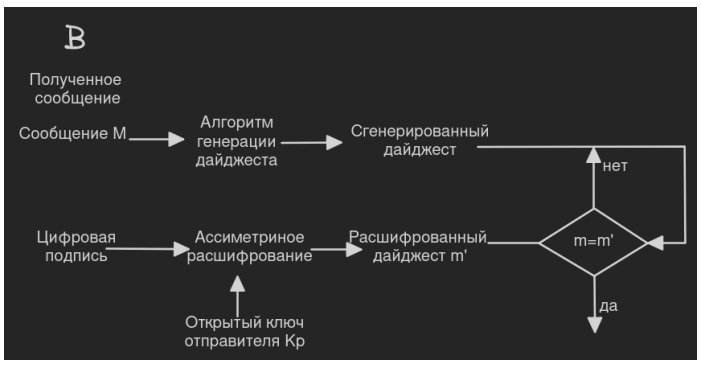
Электронная цифровая подпись

- Аутентификация пользователя
- Проверка целостности данных
- Невозможность отказа
- Зависит от подписываемого текста, практически всегда разная
- Определяется секретным ключом, принадлежит лицу, может быть утеряна
- Легко отделима от документа
- Требует дополнительных механизмов, реализующих алгоритмы её вычисления и проверки
- Требует создания доверенной инфораструктуры сертификатов открытых ключей

Создание и проверка подписи

Формирование электронной цифровой подписи



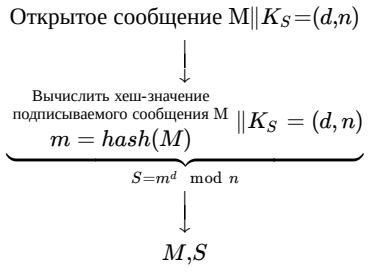


Цифровая подпись RSA

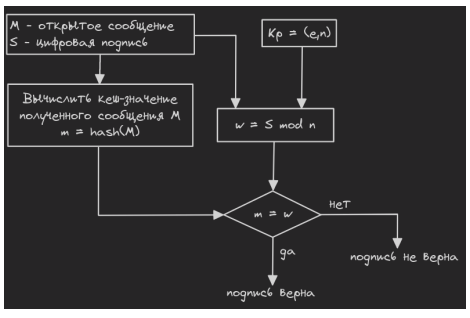
1. Этап: формирование открытого и закрытого ключей

- Выбрать p, q
- $n = p * q$
- $(p-1)(q-1)$
- Выбрать целое число e: $\text{НОД}(e,(p-1)(q-1)) = 1$
- Вычислить d: $(e * d) \mod (p-1)(q-1) = 1$
- $k_p = (e, n)$ — открытый ключ
  $k_s = (d, n)$ — закрытый ключ

2. Этап: формирование цифровой подписи



3. Этап: проверка цифровой подписи



Цифровая подпись Эль-Гамаля

Выбрать P (0 < g < P)

1 Этап: формирование ключей

Общие параметры — p,g

↓

X — случайное целое число$(1<x<p)$

↓

$y=g^x \mod p$

↓

x – закрытый ключ

y – открытый ключ

2 Этап: формирование подписи

Открытое сообщение — M

Открытые параметры — p,g

Открытый ключ — y

↓

Вычислить хеш-значение M

$\mathrm{m = hash(M)}$

↓

$k-$ случайное число $(1<k<p)$

↓

$R=g^k \mod p$

↓

$S = k^{-1} * (m - x * R) \mod (p-1)$

$k * k^{-1} \mod (p-1) = 1$

↓

Цифровая подпись: (R,S)

3 Этап: проверка

M,R,S,p,g,y

↓

кеш-значение M

$\mathrm{m = hash(M)}$

↓

$v_1=y^R * R^S \mod p$

↓

$v_2=g^m \mod p$

↓

$v_1=v_2$

↓

да: подпись+

нет: подпись-

Цифровая подпись DSA

1 Этап — формирование ключей

Выбрать: p(1024 бит), g(106 бит) $(1<g<p-1)$

↓

x — случайное целое число $(1<x<q)$

↓

$y=g^x \mod p$

↓

x – закрытый ключ

y – открытый ключ

2 Этап — подпись

Открытое сообщение — M

Открытые параметры — p,q

Открытый ключ — g

↓

Вычислить хеш-значение M

$\mathrm{m = hash(M)}$

$SHA-1$

↓

$k-$ случайное число $(1<k<q)$

↓

$R=(g^k \mod p) \mod q$

↓

$S = k^{-1} * (m + x * R) \mod q$

↓

Цифровая подпись: (R,S)

3 Этап

M,R,S,p,q,g,ay

↓

кеш-значение M

$\mathrm{m = hash(M)}$

↓

$u_1=m*S^{-1} \mod q$

↓

$u_2=R*S^{-1} \mod q$

↓

$v=(g^{u_1}*y^{u_2} \mod p) \mod q$

↓

$v=R$

↓

да: подпись+

нет: подпись-

DES

- Блок открытого текста: 64 бита
- Начальная перестановка
- Финальная перестановка

$5^2 * 5 * 5^{10} \mod 11 = 125$

$3^4 \mod 5 * 3^13 \mod 5 = 3^4 \mod 5 * 3^4 \mod 5 * 4^9 \mod 5 = 3^4 \mod 5 * 3^5 \mod 5 = 3^4 \mod 5 * 3^1 = 3$

1. 3^9\mod 15 =
   15 = 3 * 5 \phi (15) = 2 * 4 = 8
   3^9\mod 15 = (3\mod 15)(3^8 \mod 15) = 3
2. 2^{14}\mod 21
   21 = 3 * 7
   \phi (21) = 2*6=12
   (2^2\mod 21)(2^{12}\mod 21) = 4
3. 2^{107}\mod 159
   159 = 53 * 3
   \phi (53) = 52 * 2 = 103
   (2^3 \mod 159)(2^{104} \mod 159) = 8

ЕВКЛИД

1. A: 30 12 6
   B: 12 6 0
   R: 6 0
2. A: 33 12 6
   B: 12 6 0
   R: 1 0
3. …..

Цифровая подпись ГОСТ Р3410-94

Используется хеш-функция ГОСТ Р3411-94

1 Этап — формирование ключей

Общие параметры — p, q(256 бит), g

↓

X — случайное целое число$(1<x<q)$

↓

$y=g^x \mod p$

↓

x – закрытый ключ

y – открытый ключ

2 этап — формирование подписи

Открытое сообщение — M

Открытые параметры — p,q

Открытый ключ — g,x

↓

Вычислить хеш-значение M

$\mathrm{m} = \mathrm{hash}(\mathrm{M})$

↓

$k-$ случайное число $(1<k<q)$

↓

$R=(g^k \mod p) \mod q$

↓

$S = (k * m + x * R) \mod q$

↓

Цифровая подпись: (R,S)

3 Этап — проверка подписи

M,R,S,p,q,g,y

↓

кеш-значение M

$\mathrm{m} = \mathrm{hash}(\mathrm{M})$

↓

$u_1=S*m^{-1} \mod q$

↓

$u_2=-R*m^{-1} \mod q$

↓

$v=(g^{u_1}*y^{u_2} \mod p) \mod q$

↓

$v=R$

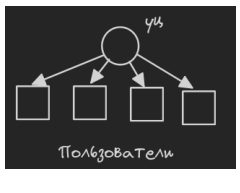
↓

да: подпись+

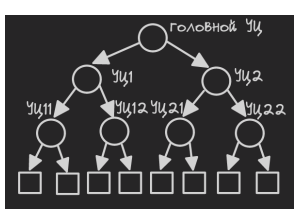
нет: подпись-

Инфратсруктура открытых ключей (PKI)

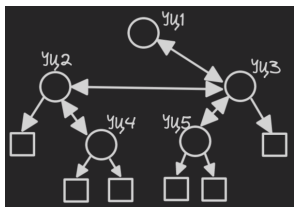
- Сертификат открытого ключа
  - Назначение: сделать достостоверным открытый ключ
  - У каждого сертификата есть:
    1. Серийный номер
    2. Мдентификатор алгоритма подписи
    3. Имя удостоверяющего центра
    4. Срок действия
    5. Имя владельца
    6. Открытый ключ
    7. ID издателя
    8. ID владельца
    9. Расширения
    10. Кол алгоритма
    11. Электронная цифровая подпись
- Основные компоненты PKI:
  - Удосстоверяющий центр
  - Регистрационный центр
  - Реестр сертификатор
  - Архив сертификатов
  - Центр запросов
  - Конечные пользователи
- Архитектура PKI:
  1. Простая PKI



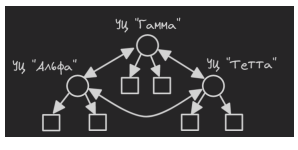
  2. Иерархическая структура



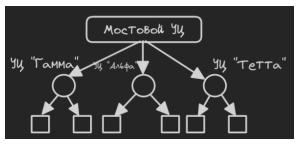
  3. Сетевая PKI



  4. Кросс-сертифицированные корпоративные



  5. Мостовые УЦ



Методы идентификации и проверки подлинности пользователей компьютерных систем

- Аутентификация, авторизация и идентификация
  - Идентификация — процедура распознавания пользователя по его идентификатору
  - Аутентификация — процедура проверки подлинности заявленного пользователя
  - Авторизация — процедура представления пользователю определённых полномочий и ресурсов в данной системе
  - Администрирование — регистрация действий пользователя в системе, включая его попытки доступа к ресурсам
- Криптографический протокол
  Протокол — совокупность действий, выполняемых в заданной последовательности двумя или более субъектами с целью достижения определённого результата.
  Свойства протокола:
  - корректность
  - полнота и однозначность
  - непротиворечивость
  - осведомлённость и согласие участников

  Криптографический протокол — это протокол, в котором используется криптографическое преобразование данных.

Протоколы аутентификации

Пользователь может предоставить различные сущности.
Они строятся на:

- Основе знания чего-либо
- Основе обладания чем-либо
- Основе каких-либо неотделимых характеристик

Обобщённый алгоритм Евклида

1. …
2. …
3. …
4. …

Возведение в степень

1. …\table … \table
2. … \table
3. … \table … \table
4. … \table … \table

Метод Диффи-Хеллмона

1. …
2. …

RSA

1. …
   Решение:
   ……..
2. …….
3. ……...
4. ……
5. ……

Межсетевые экраны

МЭ — комплекс межсетевой защиты.

МЭ — располагается между интернетом и внутренней сетью. Фильтрация осуществляется по определённым фильтрам.

Фильтр 1…n:

- Блокировать поток данных
- Обработать данные от имени пользователя
- Передать на следующий фильтр
- Пропустить данные

Типы МЭ:

- Шлюз прикладного уровня
- Шклюз сеансового уровня
- Экранирующий маршрутизатор

Схема функционирования экранирующего маршрутизатора

- (SMTP, POP3, IMAP4)
- Посредник-служба (NNTP)
- Посредник службы удалённого управления компьютером (Telnet)

+ плюсы: обеспечивает высокий уровень защиты молкальной сети; защиту на уровне приложений
- минусы: высокие требования производительности;
отсутствие прозрачности для пользователя;
снижение пропускной способности

Основные схемы подключения МЭ

1. Схемы защиты сети с использованием экранизирующего маршрутизатора
   Является самым распространённым и простым. Он обеспечивает фильтрацию на основе анализа адресов и портов.

2. Схема единой защиты локальной сети.
   Является наиболее простым решением для локальной сети. МЭ является единственным видимым средством защиты. “Запрещено всё, что явно не разрешено”. Есть доступ по полномочиям. Она неостаточно гибкая.
3. Схема с защищаемой закрытой и не защищащённой открытой подсетями.
   Открытая подсеть никак не защищена
4. Схема с разделённой защитой закрытой и открытой подсетей
   Может быть 2 МЭ у крупных организаций.
   1 МЭ обеспечивает защиту обеих подсетей, а 2 МЭ защищает только закрытую

Виртуальная частная сеть — это объединение мокальных сетей и отдельных компьютеров через открытую внешнюю среду передачи информации в единую виртуальную сеть, которая обеспечивает конфиденциальность и келостность данных.

Туннель VPN — это виртуальное соединение, которое проходит через открытую сеть, по которому передаются криптографически-защищённые пакеты сообщений.

При обработке сообщений:

1. Из заголовка IP-пакета выделяются информация об адресе. VPN-агент выбирает алгоритм защиты и криптографические ключи.
2. С помощью выбранного алгоритма защиты, формируется и добавляется в IP-пакет электронно цифровая подпись.
3. Выбирается алгоритм шифрования и зашифровывается IP-пакет.
4. Щашифрованный IP-пакет помещется в другой IP-пакет и заголовок нового содержит VPN-агента адресата и VPN-агента отправителя.
5. Пакет отправляется VPN-агенту получателю

1) Из заголовка IP-пакета выделяется его отправитель

2) Выбираются алгоритмы защиты и ключи, которые будут расшифровывать. Идёт проверка целостности пакета.

3) Выделяется информационная часть и расшифровывается.

4) Проверяется целостность.

5) Пакет отправляется получателю.

Безопасность VPN-агентов

1. Конфиденциальность;
2. Целостность;
3. Доступность.