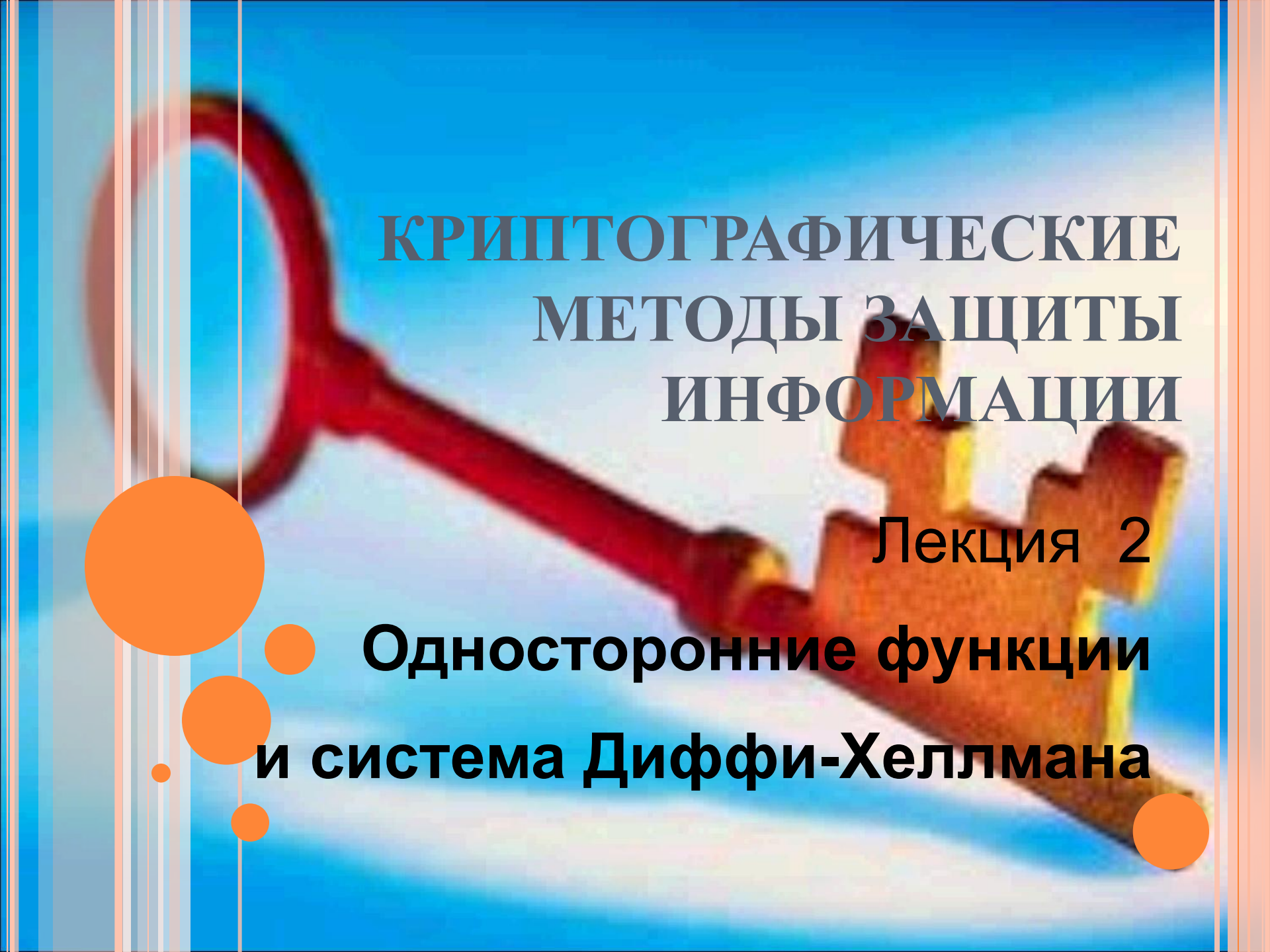


КРИПТОГРАФИЧЕСКИЕ МЕТОДЫ ЗАЩИТЫ ИНФОРМАЦИИ

Лекция 2

Односторонние функции и система Диффи-Хеллмана

ПРЕДЫСТОРИЯ И ОСНОВНЫЕ ИДЕИ

Для того, чтобы лучше понять идеи, лежащие в основе ряда криптографических схем и алгоритмов, рассмотрим три практически важные проблемы.

Чуть позже мы увидим, насколько легко и красиво они решаются при помощи так называемых *односторонних функций*.

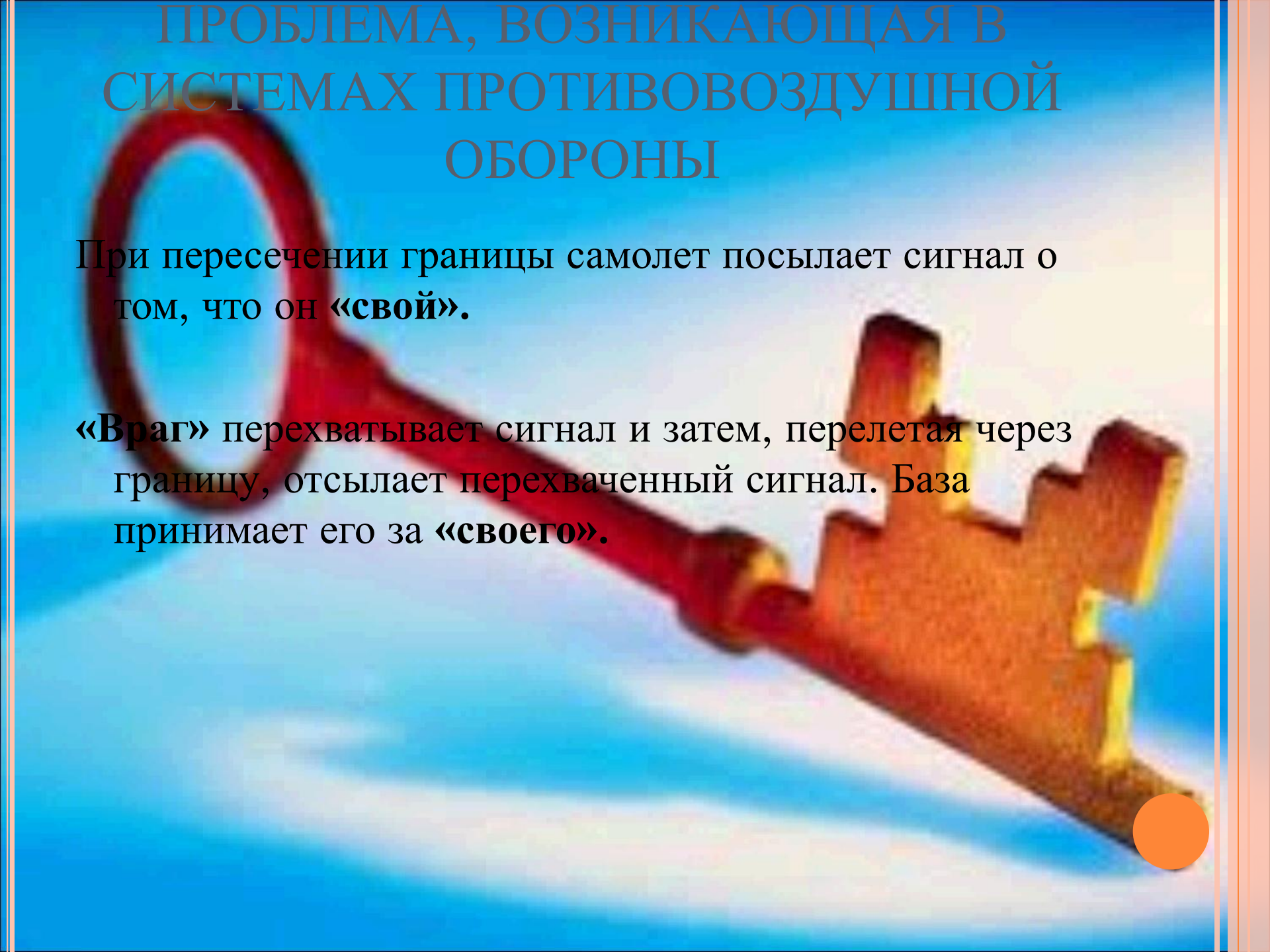
Проблемы следующие:

- Проблема хранения паролей в компьютере;
- Проблема ПВО;
- Проблема, возникающая в сетях с удаленным доступом.

ПРОБЛЕМА ХРАНЕНИЯ ПАРОЛЕЙ В КОМПЬЮТЕРЕ

При хранении **логинов** и **паролей** в компьютере **администратор** может прочитать их и воспользоваться в своих целях.

ПРОБЛЕМА, ВОЗНИКАЮЩАЯ В СИСТЕМАХ ПРОТИВОВОЗДУШНОЙ ОБОРОНЫ

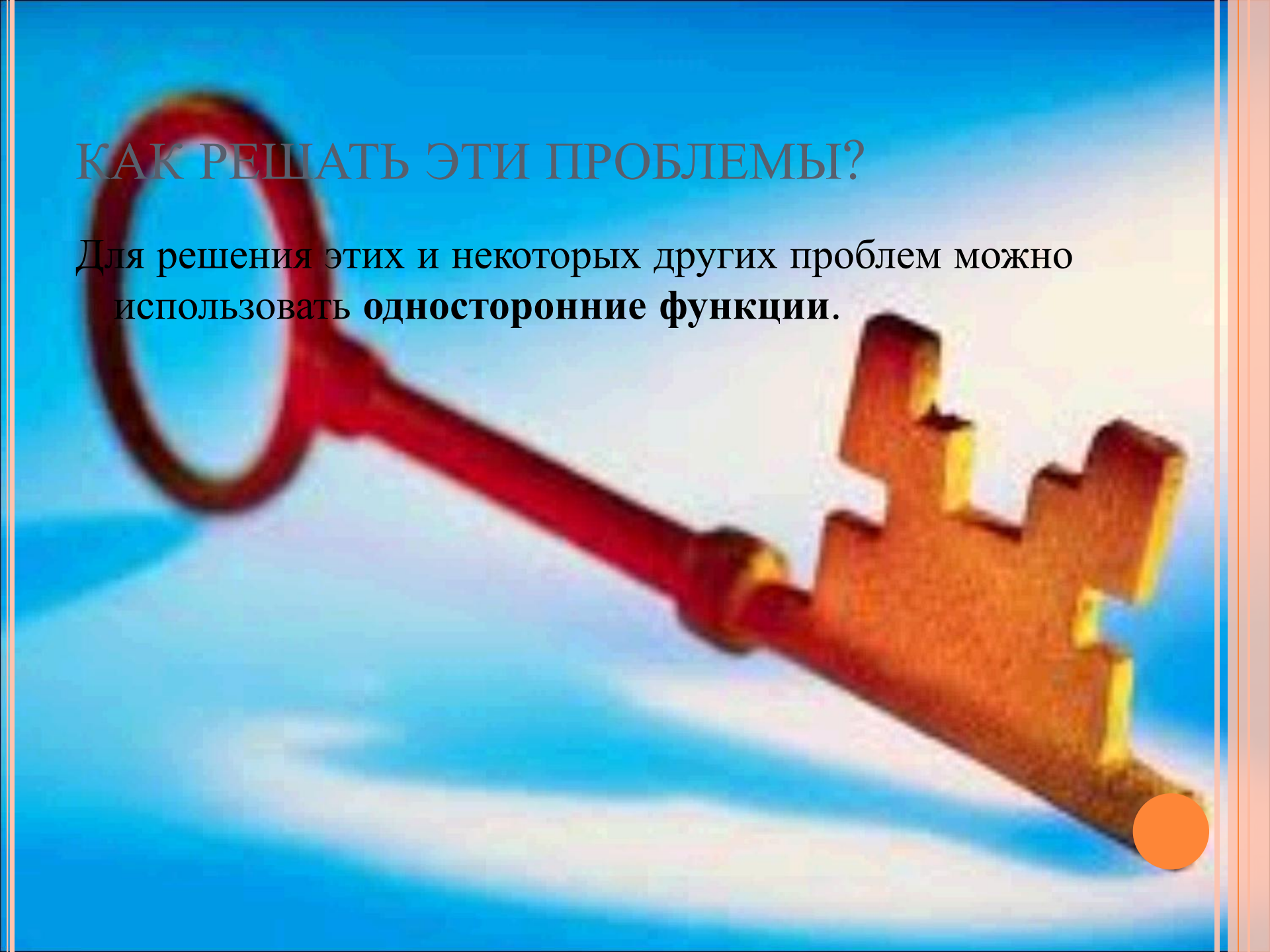


При пересечении границы самолет посылает сигнал о том, что он **«свой».**

«Враг» перехватывает сигнал и затем, перелетая через границу, отсылает перехваченный сигнал. База принимает его за **«своего».**

КАК РЕШАТЬ ЭТИ ПРОБЛЕМЫ?

Для решения этих и некоторых других проблем можно использовать **односторонние функции**.



ОДНОСТОРОННЯЯ ФУНКЦИЯ (ОПРЕДЕЛЕНИЕ)

Функция называется **односторонней**, если она вычисляется относительно быстро, а обратную к ней вычислить за реальное время невозможно.

То есть теоретически можно, но практически нельзя.

Например,

$y=f(x)$ вычисляется за 10 секунд,

$x=f^{-1}(y)$ вычисляется за 100000 лет.

ОДНОСТОРОННЯЯ ФУНКЦИЯ, КОТОРУЮ МЫ БУДЕМ ИСПОЛЬЗОВАТЬ

Возведение в степень по модулю

$$y = a^x \bmod p.$$

Пример. Вычислим a^{64}.

Медленный (наивный) способ: $a^{64} = a*a*a* \ldots *a$

(63 умножения).

БЫСТРЫЙ СПОСОБ УМНОЖЕНИЯ

Быстрый способ: $a^{64} = (((((a^2)^2)^2)^2)^2)^2$

(6 умножений)

$64 = 2^6$.

Степень	Количество умножений для медленного способа	Количество умножений для быстрого способа
64	63	6
512	511	9
16 000 001	16 000 000	24
1000 000 001	1000 000 000	30
1 000 000 000 001	1 000 000 000 000	40

Недостаток рассмотренного способа – он работает только для степеней двойки.

- Можно ли расширить его так, чтобы возводить в степень можно было любые числа?

- **Идея.**
- $7^{68169} = 7^{65536} * 7^{2048} * 7^{512} * 7^{64} * 7^{8} * 7^{0}$

БЫСТРЫЙ АЛГОРИТМ ВОЗВЕДЕНИЯ В СТЕПЕНЬ ПО МОДУЛЮ (ОПИСАНИЕ АЛГОРИТМА)

Для описания алгоритма введем величину $t = [\log_2 x]$ — целую часть $\log_2 x$ (далее все логарифмы будут двоичные, поэтому в дальнейшем мы не будем писать индекс 2). Вычисляем числа ряда

$$a, \quad a^2, \quad a^4, \quad a^8, \quad \ldots, \quad a^{2^t} \mod p. \tag{2.5}$$

В ряду (2.5) каждое число получается путем умножения предыдущего числа самого на себя по модулю p. Запишем показатель степени x в двоичной системе счисления:

$$x = (x_t x_{t-1} \ldots x_1 x_0)_2.$$

Тогда число $y = a^x \bmod p$ может быть вычислено как

$$y = \prod_{i=0}^{t} a^{x_i \cdot 2^i} \mod p \tag{2.6}$$

БЫСТРЫЙ АЛГОРИТМ ВОЗВЕДЕНИЯ В СТЕПЕНЬ ПО МОДУЛЮ (ПРИМЕР)

Пример 2.1. Пусть требуется вычислить $3^{100} \bmod 7$. Имеем $t = [\log 100] = 6$. Вычисляем числа ряда (2.5):

a	a^2	a^4	a^8	a^{16}	a^{32}	a^{64}
3	2	4	2	4	2	4

Записываем показатель в двоичной системе счисления: $100=(1100100)_2$. Проводим вычисления по формуле (2.6):

$$a^{64} \quad a^{32} \qquad\qquad a^4$$
$$4 \cdot \quad 2 \cdot \quad 1 \cdot \quad 1 \cdot \quad 4 \cdot \quad 1 \cdot \quad 1 \quad = \quad 4$$

Нам потребовалось всего 8 операций умножения.

БЫСТРЫЙ АЛГОРИТМ ВОЗВЕДЕНИЯ В СТЕПЕНЬ ПО МОДУЛЮ (ПСЕВДОКОД)

Алгоритм 2.1 ВОЗВЕДЕНИЕ В СТЕПЕНЬ (СПРАВА-НАЛЕВО)

ВХОД: Целые числа a, $x = (x_t x_{t-1} \ldots x_0)_2$, p.
ВЫХОД: Число $y = a^x \bmod p$.

1. $y \leftarrow 1$, $s \leftarrow a$.
2. FOR $i = 0, 1, \ldots, t$ DO
3. IF $x_i = 1$ THEN $y \leftarrow y \cdot s \bmod p$;
4. $s \leftarrow s \cdot s \bmod p$.
5. RETURN y.

ДИСКРЕТНЫЙ ЛОГАРИФМ

Дискретный логарифм – это функция, обратная к $y = a^x \bmod p$.

$$x = \log_a y \bmod p.$$

Не существует эффективных алгоритмов ее вычисления.

Определение.

Вычисление дискретного логарифма называется *дискретным логарифмированием*.

МЕТОДЫ ДИСКРЕТНОГО ЛОГАРИФМИРОВАНИЯ

Название метода	**Необходимое количество умножений (в среднем)**
Метод полного перебора (метод грубой силы)	$p/2$
Метод «Шаг младенца-шаг великана»	$2p^{1/2}$
Метод исчисления порядка	Еще меньше

СРАВНЕНИЕ СЛОЖНОСТИ ПРЯМОЙ И ОБРАТНОЙ ФУНКЦИИ БЫСТРОГО ВОЗВЕДЕНИЯ В СТЕПЕНЬ ПО МОДУЛЮ

Утверждение 2.1 (о сложности вычислений (2.3)). *Количество операций умножения при вычислении по (2.3) не превосходит* $2 \log x$.

Таблица 2.1: Количество умножений для вычисления прямой и обратной функции

Количество десятичных знаков в записи p	Вычисление (2.3) ($2 \log p$ умножений)	Вычисление (2.4) ($2 \cdot \sqrt{p}$ умножений)
12	$2 \cdot 40 = 80$	$2 \cdot 10^6$
60	$2 \cdot 200 = 400$	$2 \cdot 10^{30}$
90	$2 \cdot 300 = 600$	$2 \cdot 10^{45}$

РЕШЕНИЕ ПРОБЛЕМЫ ХРАНЕНИЯ ПАРОЛЕЙ В КОМПЬЮТЕРЕ

На компьютере хранится не ***логин*** и ***пароль***, а ***логин*** и ***y(пароль)*** т.е.

$$y = a^{\text{пароль}} \bmod p.$$

(параметры ***a*** и ***p*** как-то выбираются и могут быть известны всем)

Когда пользователь входит в систему, то от его введенного пароля вычисляется односторонняя функция и сравнивается с хранящимся на компьютере значением.

РЕШЕНИЕ ПРОБЛЕМЫ ПВО

База генерирует случайное число ***R*** и передает (открыто) его самолету.

Самолет вычисляет

$$y = a^{\text{пароль} \,|\, R} \bmod p.$$

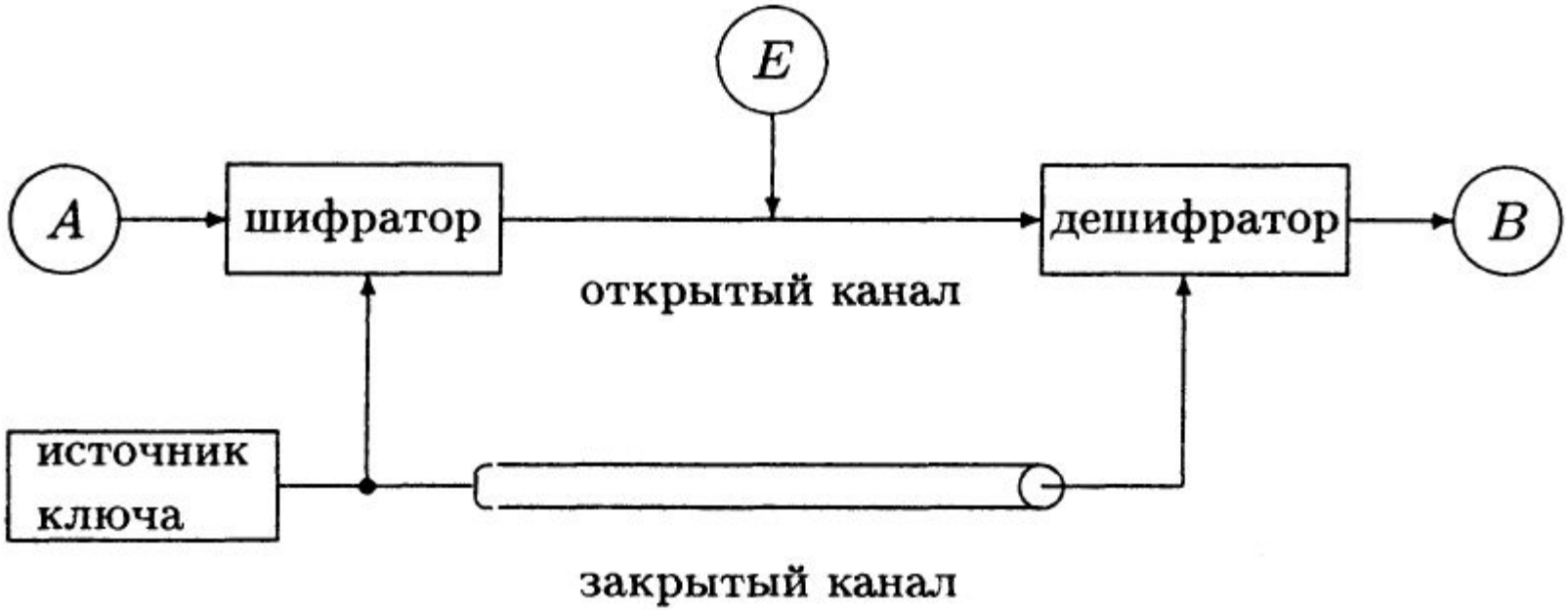
и передает сигнал на базу.

Если **«враг»** перехватит ***y*** и отошлет его на базу, то за **«своего»** не сойдет. Потому что для него число R уже будет другим.

ВЫВОДЫ

- Надежность рассмотренных криптосистем основана на том, что враг **не может практически вскрыть** систему.
- Фактически **мы предлагаем врагу решить задачу дискретного логарифмирования** для больших чисел.
- Однако не доказано, что более эффективных алгоритмов не существует. Поэтому может быть кто-то придумает очень быстрый алгоритм дискретного логарифмирования, и вся криптография устареет в один миг.

КРИПТОСИСТЕМА С ЗАКРЫТЫМ (СЕКРЕТНЫМ) КЛЮЧОМ



КРИПТОСИСТЕМА С ОТКРЫТЫМ КЛЮЧОМ

Отличие криптосистемы с закрытым ключом от криптосистемы с открытым ключом

- Криптосистема с *закрытым (секретным) ключом* **подразумевает** наличие защищенного канала, по которому передается секретный ключ.
- Криптосистема с *открытым ключом* **не подразумевает** наличие защищенного канала, по которому передается секретный ключ.

ПРИМЕРЫ СЕКРЕТНЫХ КАНАЛОВ (ЭТО ДОРОГИЕ КАНАЛЫ)

- Личная встреча
- Курьерская почта
- Охраняемый поезд
- ……

Дорогой канал – это значит труднодоступный, медленный, имеющий высокую стоимость. Им нельзя воспользоваться в любой момент.

ПРИМЕРЫ НЕЗАЩИЩЕННЫХ КАНАЛОВ (ЭТО ДЕШЕВЫЕ КАНАЛЫ)

- Интернет
- Телефон
- Обычная почта
- E-mail
- ……

Дешевый канал – это значит легкодоступный, быстрый, имеющий невысокую стоимость. Им можно воспользоваться в любой момент.

ЕЩЕ ОДИН НЕДОСТАТОК ОБМЕНА СЕКРЕТНЫМИ КЛЮЧАМИ

Вопрос.

Сколько нужно ключей, если ***N*** абонентов хотят общаться попарно безопасно?

Ответ

N*(N-1)/2

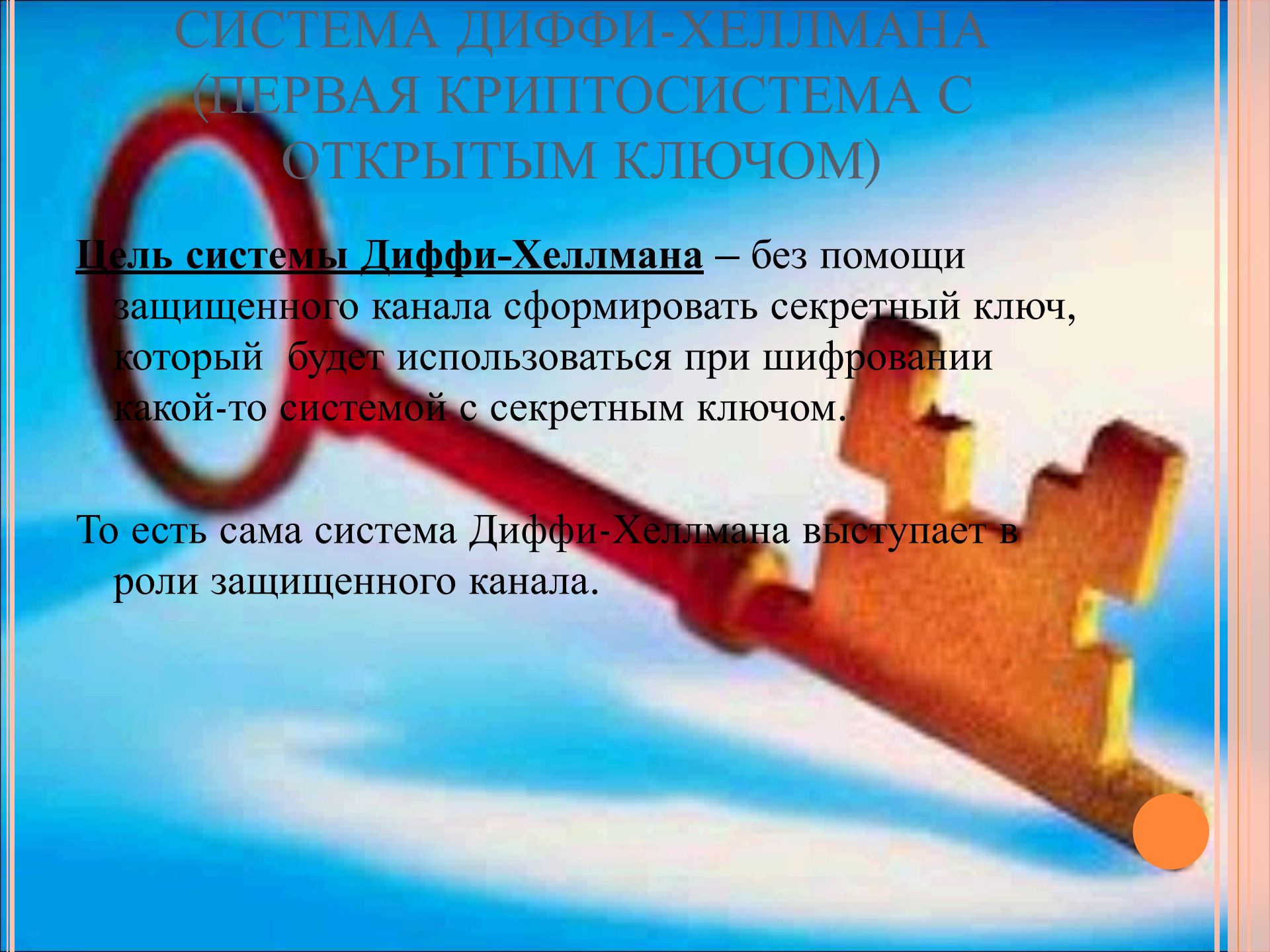
Примерно $N^2/2$

N	Количество ключей
2	1
10	45
100	≈5000
1000	≈500 тыс.
10000	≈50 млн.

СИСТЕМА ДИФФИ-ХЕЛЛМАНА (ПЕРВАЯ КРИПТОСИСТЕМА С ОТКРЫТЫМ КЛЮЧОМ)

Цель системы Диффи-Хеллмана – без помощи защищенного канала сформировать секретный ключ, который будет использоваться при шифровании какой-то системой с секретным ключом.

То есть сама система Диффи-Хеллмана выступает в роли защищенного канала.



СИСТЕМА ДИФФИ-ХЕЛЛМАНА ДЛЯ АБОНЕНТОВ A, B, C…. (ВЫБОР ПАРАМЕТРОВ)

- Выбрать большое простое p.
- Выбрать g, такое что числа 1, 2, …. ,p-1 могут быть представлены как степени g по модулю p. Алгоритм, как это сделать, описан далее.
- Каждый абонент выбирает свое число X и хранит его в секрете.
- Каждый абонент вычисляет число Y и публикует его.
- Общий секретный ключ вычисляется на основании открытого ключа собеседника и своего секретного ключа.

ПАРАМЕТРЫ ПОЛЬЗОВАТЕЛЕЙ В СИСТЕМЕ ДИФФИ-ХЕЛЛМАНА

$$\begin{cases} Y_A = g^{X_A} \bmod p, \\ Y_B = g^{X_B} \bmod p, \\ Y_C = g^{X_C} \bmod p. \end{cases} \tag{2.7}$$

В результате получаем следующую таблицу.

Таблица 2.2: Ключи пользователей в системе Диффи-Хеллмана

Абонент	Секретный ключ	Открытый ключ
A	X_A	Y_A
B	X_B	Y_B
C	X_C	Y_C

ВЫЧИСЛЕНИЕ ОБЩЕГО КЛЮЧА С ПОМОЩЬЮ СИСТЕМЫ ДИФФИ-ХЕЛЛМАНА

$$Z_{AB} = (Y_B)^{X_A} \bmod p \tag{2.8}$$

(никто другой кроме A этого сделать не может, так как число X_A секретно). Абонент B вычисляет число

$$Z_{BA} = (Y_A)^{X_B} \bmod p. \tag{2.9}$$

Утверждение 2.2. $Z_{AB} = Z_{BA}$.

Доказательство. Рассмотрим следующие равенства:

$$Z_{AB} = (Y_B)^{X_A} \bmod p = (g^{X_B})^{X_A} \bmod p =$$

$$= g^{X_A X_B} \bmod p = (Y_A)^{X_B} \bmod p = Z_{BA}.$$

ВЫБОР ПАРАМЕТРА *G*

Выбор числа g при произвольно заданном p может оказаться трудной задачей, связанной с разложением на простые множители числа $p-1$. Дело в том, что для обеспечения высокой стойкости рассмотренной системы число $p-1$ должно обязательно содержать большой простой множитель (в противном случае алгоритм Полига-Хеллмана, описанный, например, в [26], быстро вычисляет дискретный логарифм). Поэтому часто рекомендуют использовать следующий подход. Простое число p выбирается таким, чтобы выполнялось равенство

$$p = 2q + 1,$$

где q — также простое число. Тогда в качестве g можно взять любое число, для которого справедливы неравенства

$$1 < g < p-1 \quad \text{и} \quad g^q \bmod p \neq 1.$$

СИСТЕМА ДИФФИ-ХЕЛЛМАНА (ПРИМЕР)

Пример 2.2. Пусть $p = 23 = 2 \cdot 11 + 1$ $(q = 11)$. Выберем параметр g. Попробуем взять $g = 3$. Проверим: $3^{11} \bmod 23 = 1$ и значит, такое g не подходит. Возьмем $g = 5$. Проверим: $5^{11} \bmod 23 = 22 \neq 1$. Итак, мы выбрали параметры для системы Диффи-Хеллмана: $p = 23$, $g = 5$. Теперь каждый абонент выбирает секретное число и вычисляет соответствующее ему открытое число. Пусть $X_A = 7$, $X_B = 13$. Вычисляем $Y_A = 5^7 \bmod 23 = 17$, $Y_B = 5^{13} \bmod 23 = 21$. Пусть A и B решили сформировать общий секретный ключ. Для этого A вычисляет $Z_{AB} = 21^7 \bmod 23 = 10$, а B вычисляет $Z_{BA} = 17^{13} \bmod 23 = 10$. Теперь они имеют общий ключ 10, который не передавался по каналу связи.

ЛИТЕРАТУРА

Рябко, Фионов

Основы современной криптографии

Глава 2

ПРАКТИЧЕСКОЕ ЗАДАНИЕ

1. Реализуйте одностороннюю функцию – быстрое возведение в степень по модулю.

2. Реализуйте систему Диффи-Хеллмана. Не забудьте, что для возведения в степень нужно использовать созданную одностороннюю функцию.