Современное шифрование для backend разработчика

Григорий Скобелев Java Dev

Григорий Скобелев

О спикере

- Java Backend Developer
- Директор программного комитета Podlodka Backend Crew
- Организатор книжного клуба { между скобок }
- Лектор Нетологии
- Активный спикер

Книжный клуб { между скобок }

книжный клуб для backend разработчиков

Цели доклада

- Разберемся, что такое симметричное и асимметричное шифрование
- Обсудим современные алгоритмы шифрования
- Посмотрим, что нового в Java 17 с точки зрения криптографии
- Поговорим, на что еще стоит обратить внимание

План

- Шифрование
  - Как работает шифрование
  - Актуальные алгоритмы
- Java 17
- Стоит обратить внимание 🤔
  - Поиск по зашифрованному значению
  - Ротация ключей/алгоритма

Шифрование

Шифрование - обратимое преобразование информации в нечитаемый вид с целью обеспечения конфиденциальности

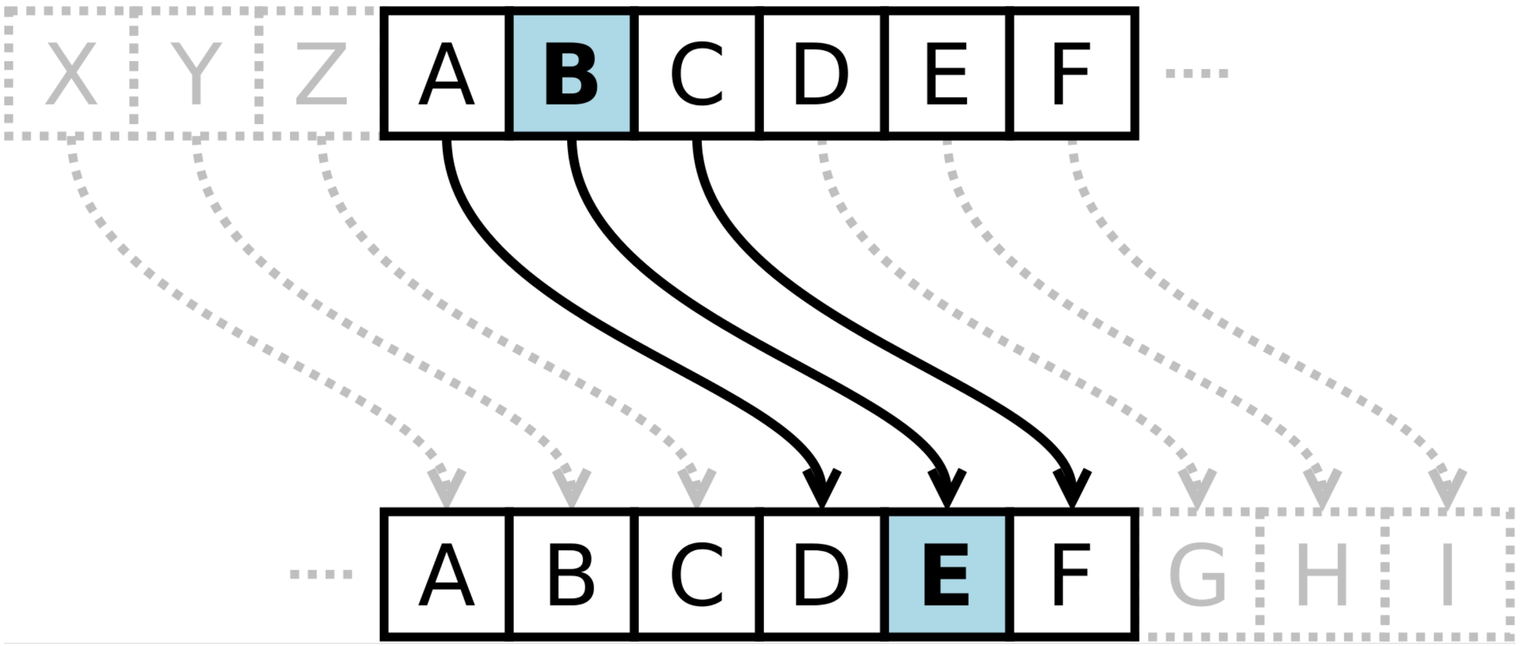
Шифрование

Обеспечивает защиту информации:

- Конфиденциальность
- Целостность
- Проверку подлинности

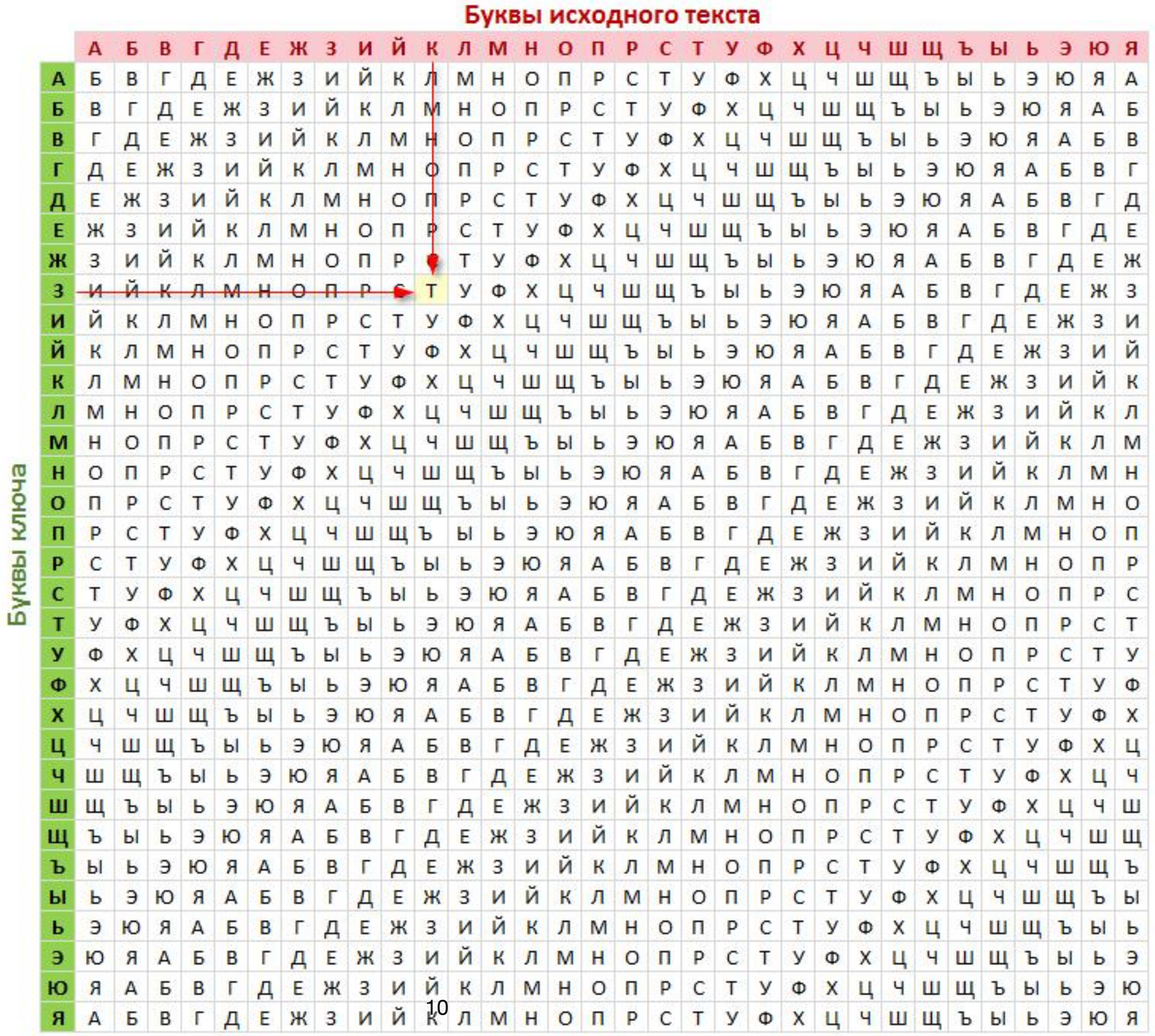
Шифрование

Шифр Цезаря



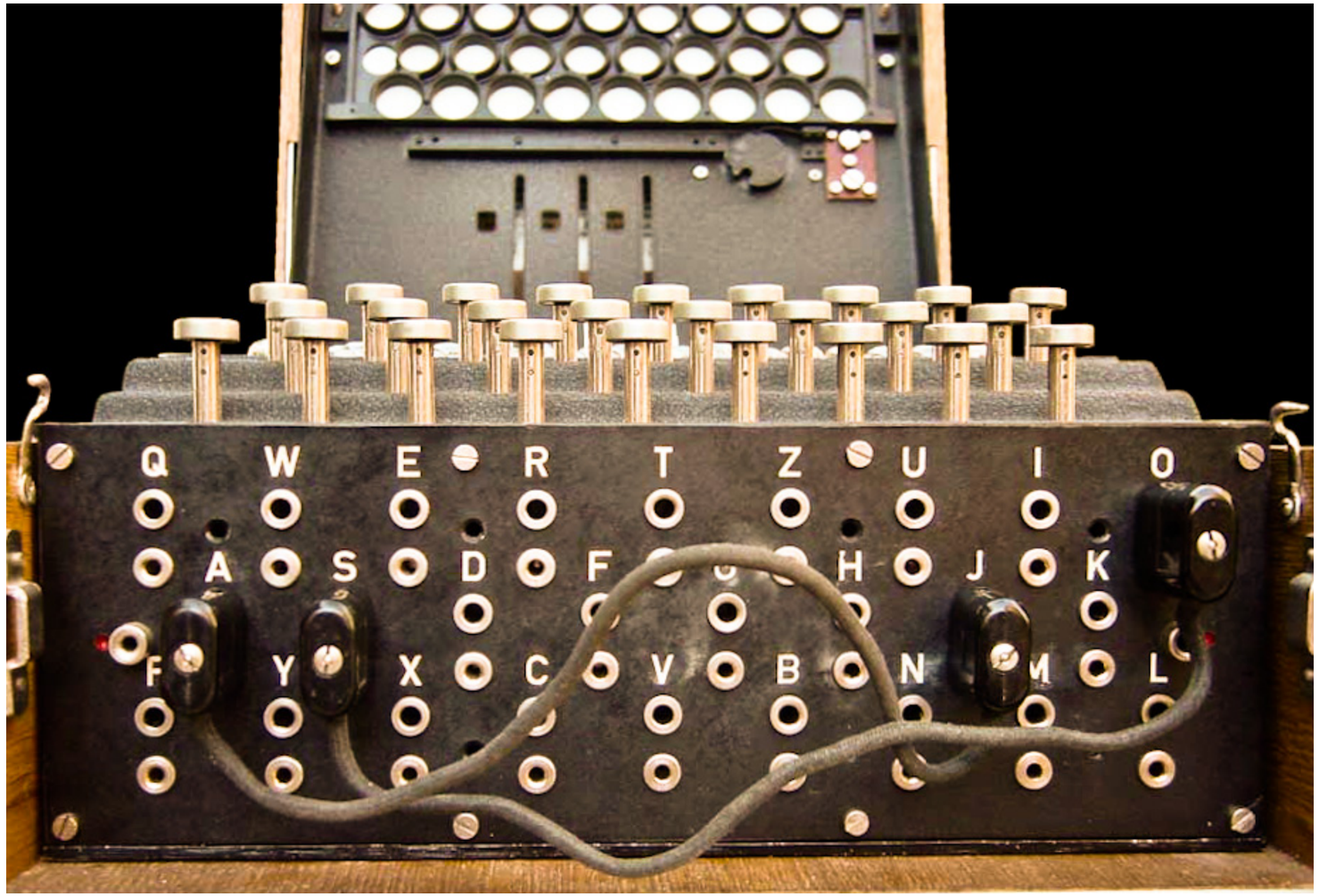
Шифрование

Шифр Виженера



Шифрование

Энигма

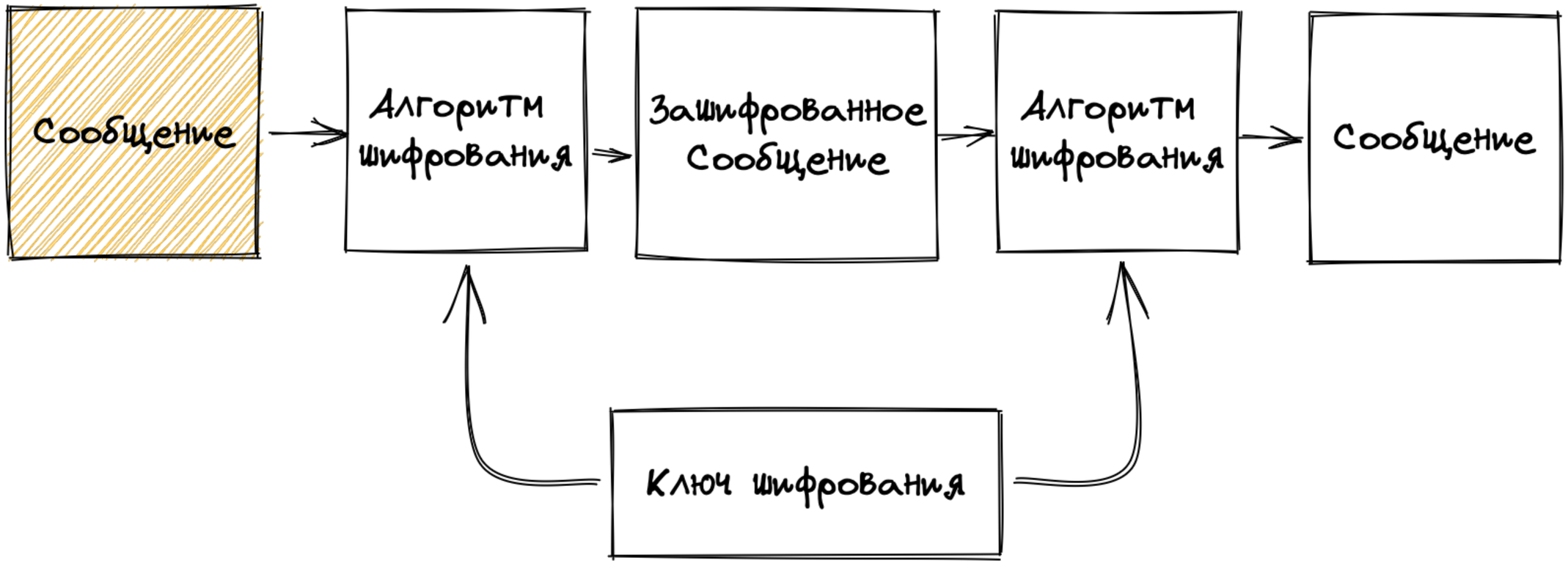


Шифрование

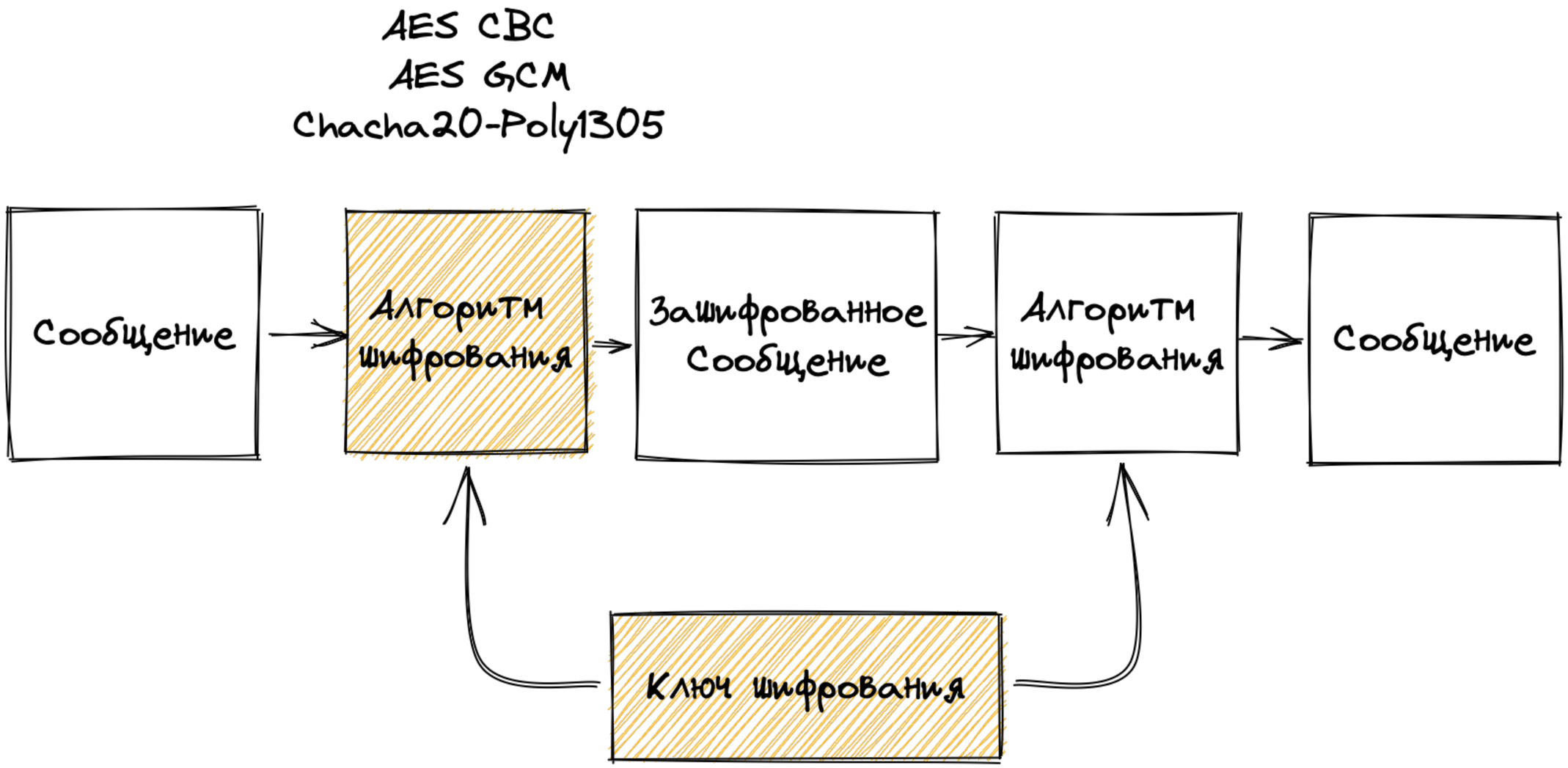
- Симметричное
  - один ключ для всего
  - более быстрый
- Асимметричное
  - 2 ключа - публичный для шифрования, приватный для расшифровки
  - требует больше ресурсов

Симметричное шифрование

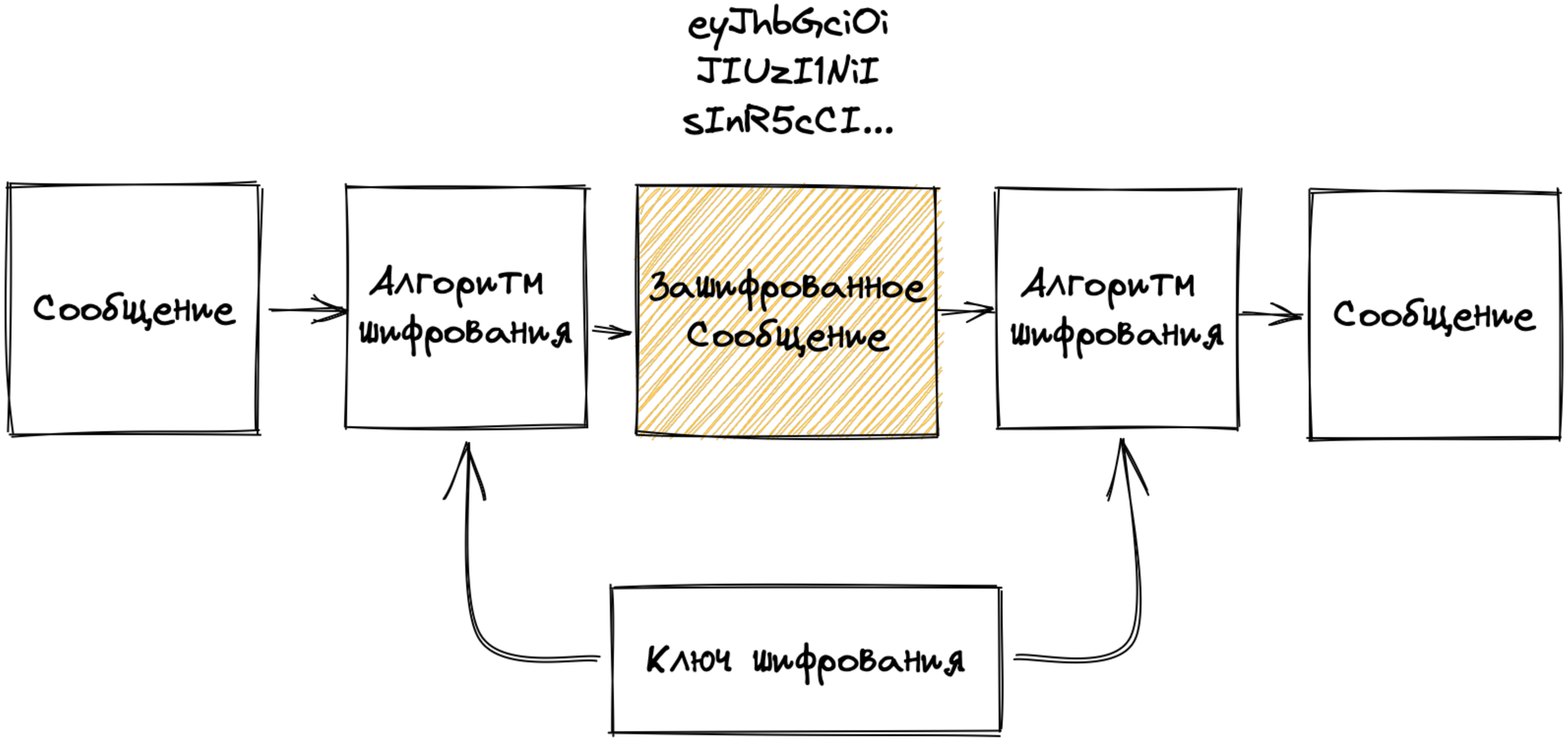
Привет, joker 2022!



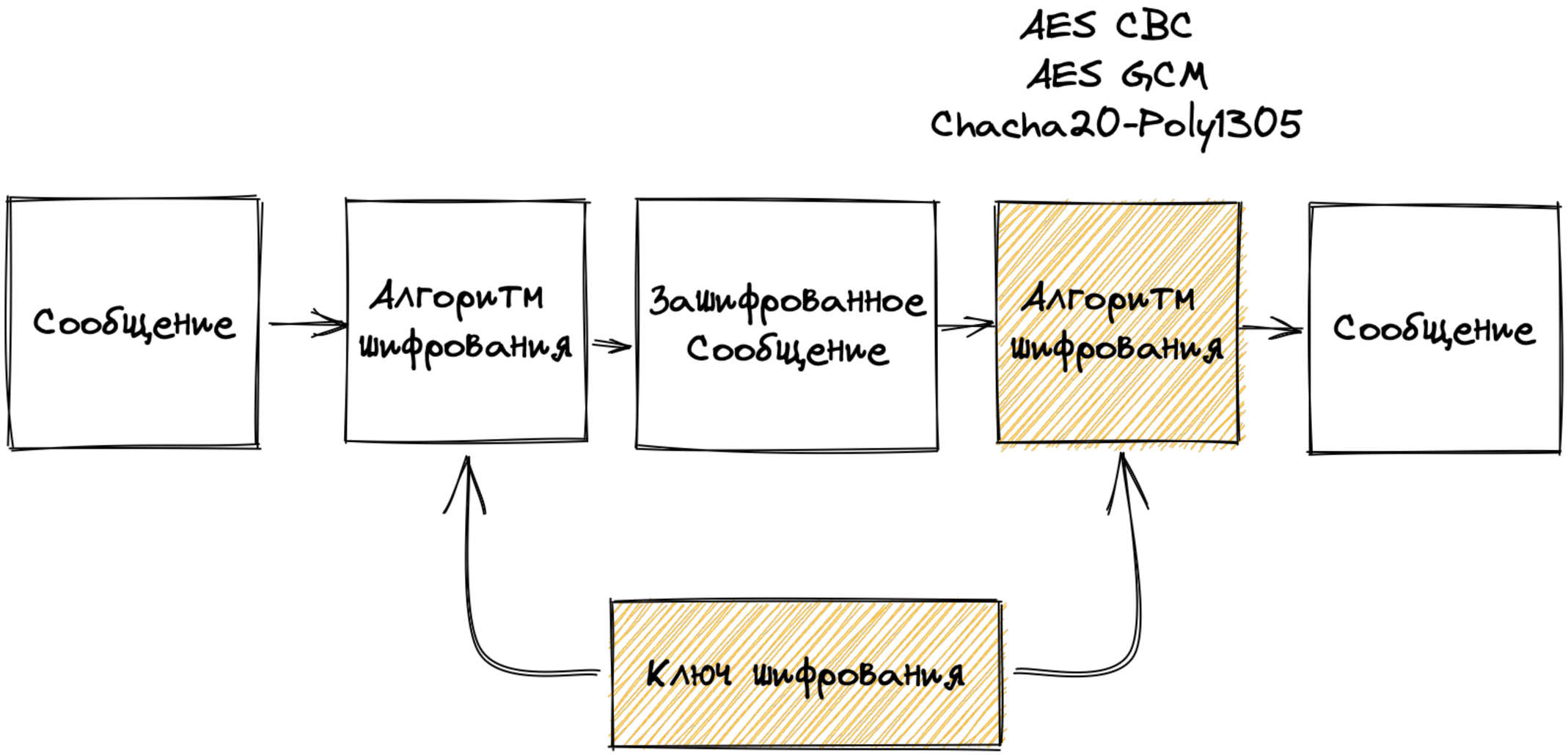
Симметричное шифрование



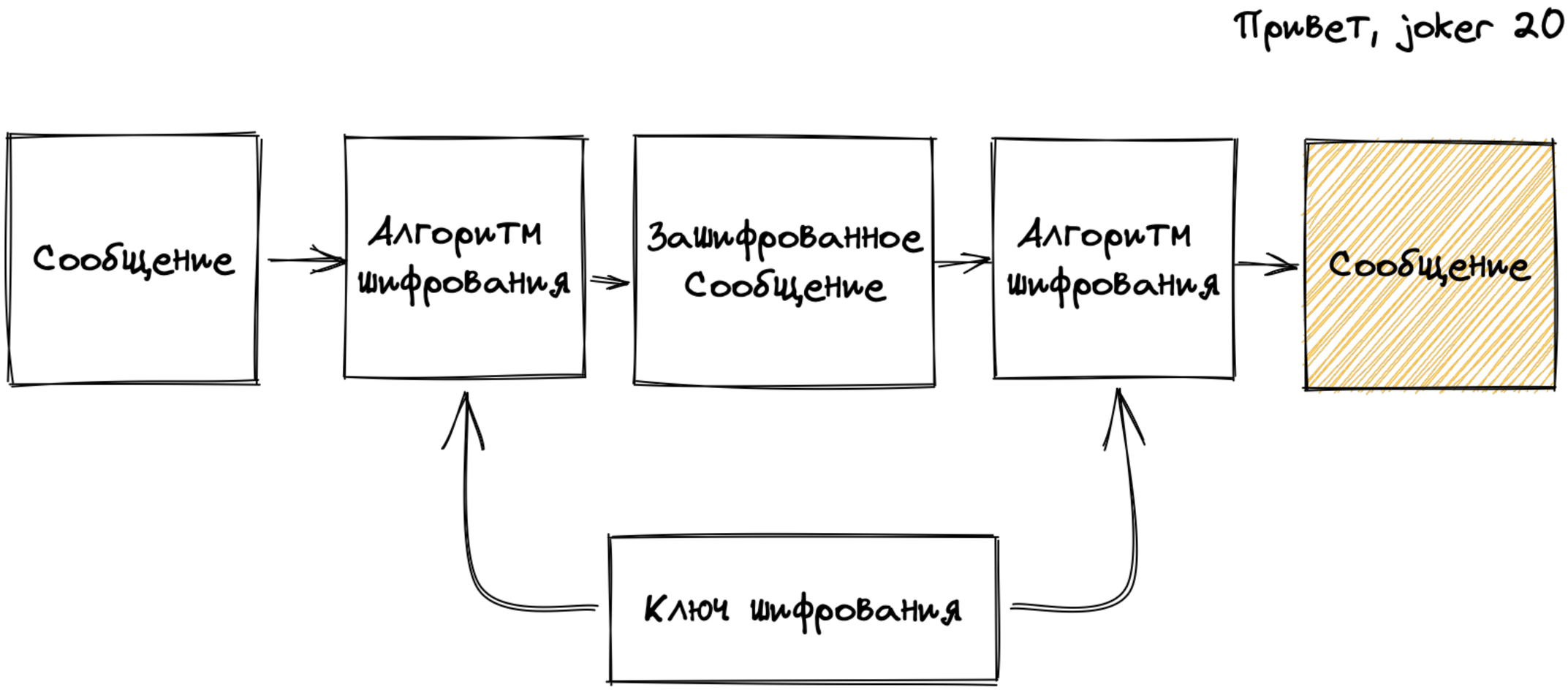
Симметричное шифрование



Симметричное шифрование



Симметричное шифрование



Симметричное шифрование

Примеры

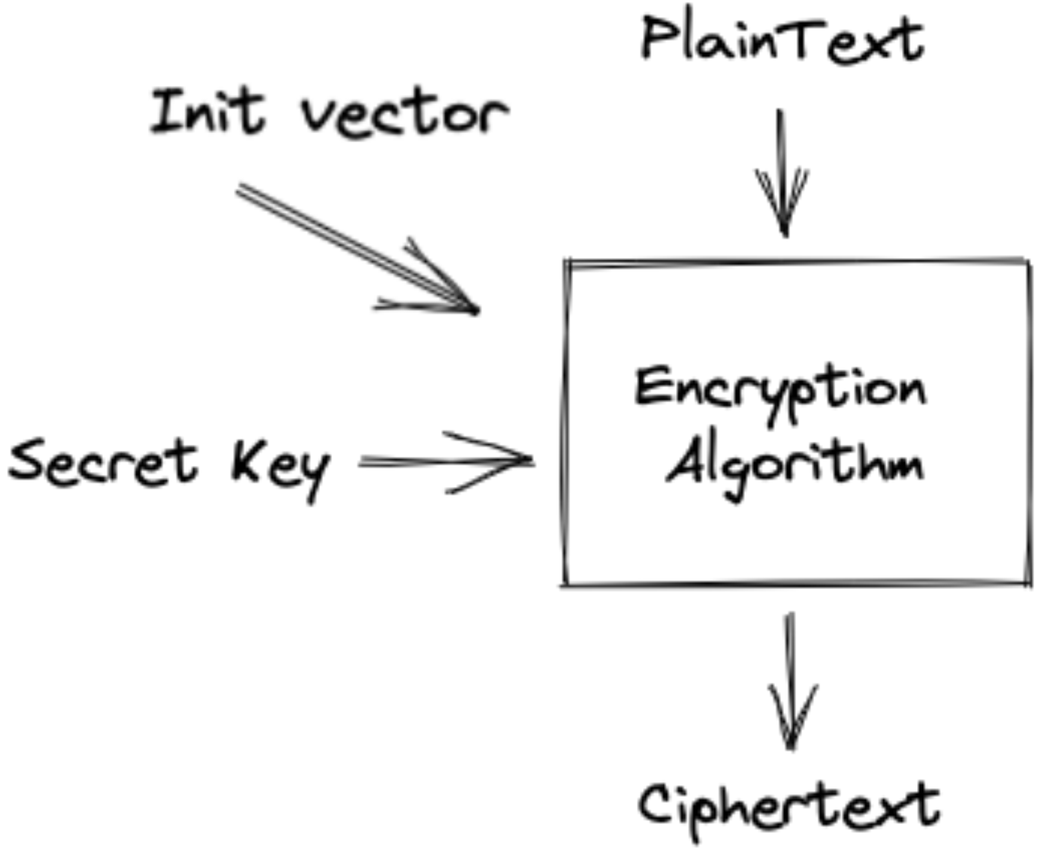
- Мессенджеры
- Банковские платежи, переводы и онлайн оплата
- Защита персональных данных на уровне БД
- Аутентификация
- Подпись запросов
- TLS*

Симметричное шифрование

- Шифрование с помощью блоков (Block cipher)
- Потоковое шифрование (Stream cipher)

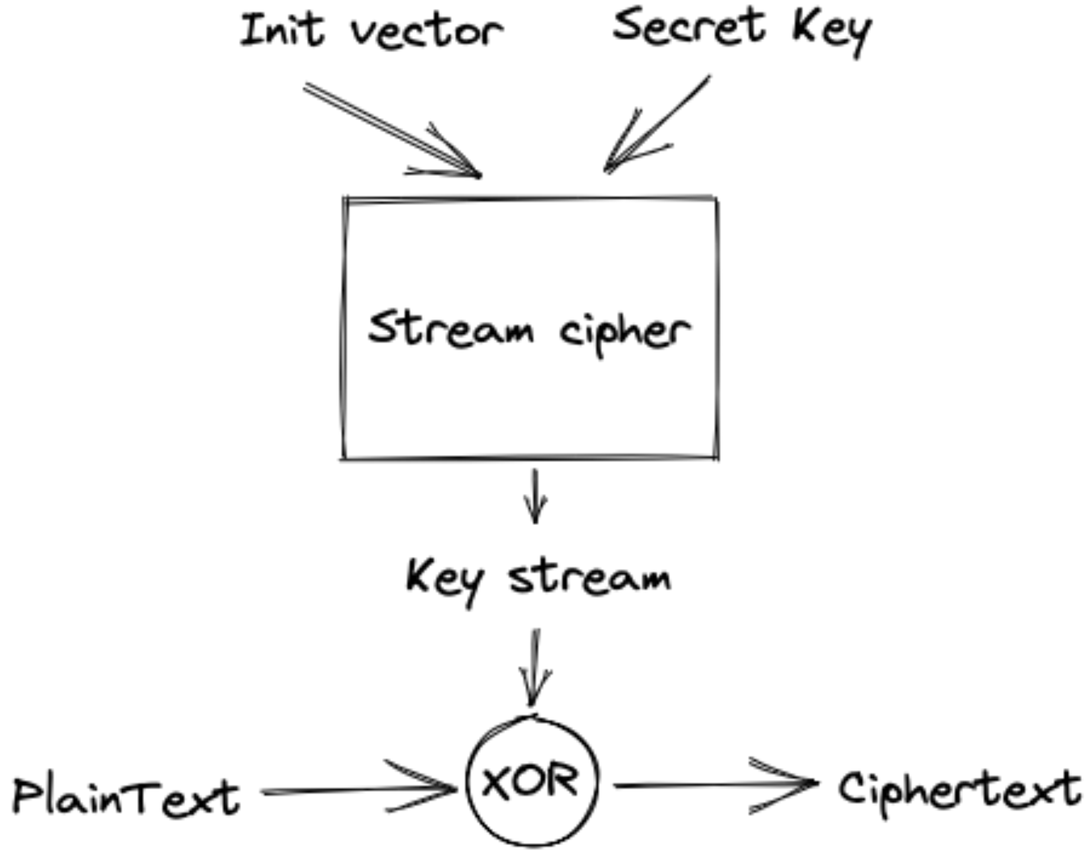
Блочное шифрование

- Шифртекст это последовательность блоков
- Сообщение шифруется блоками фиксированного размера
- Размер блока зависит от алгоритма
- Если размер сообщения не кратен размеру блока, в последний блок добавляется Padding



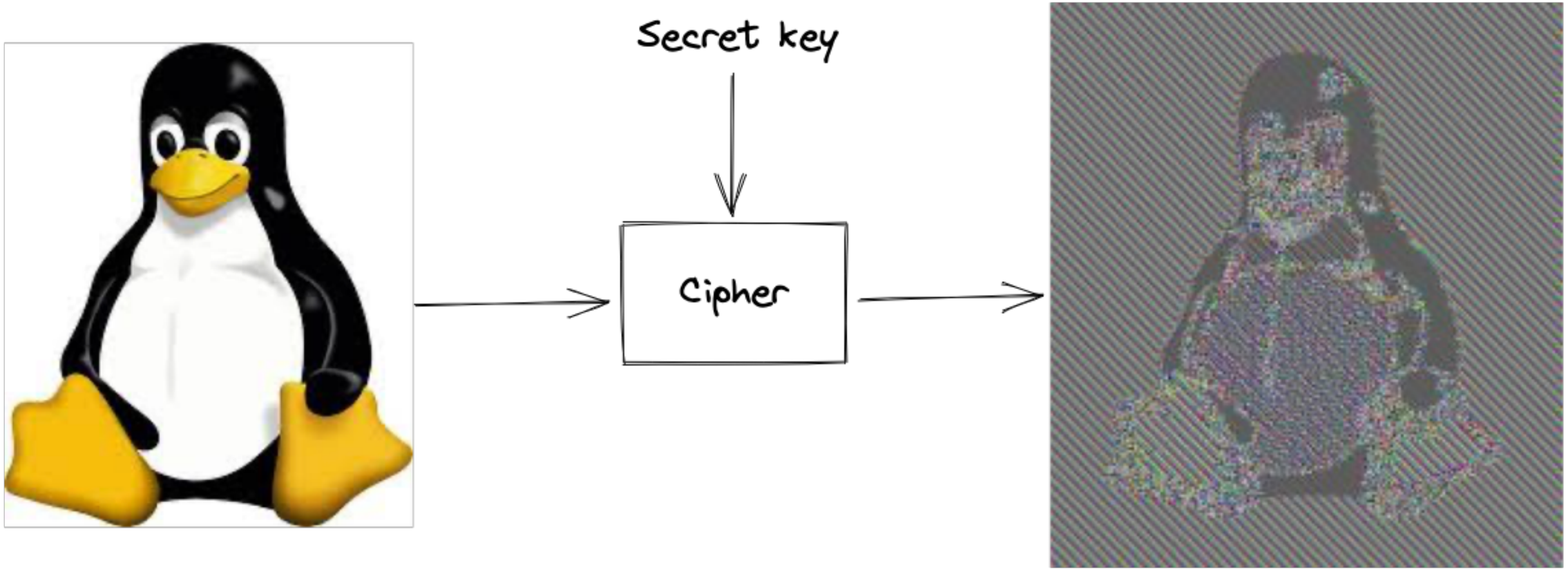
Потоковое шифрование

- KeyStream зависит от алгоритма шифрования
- KeyStream формируется из ключа и секрета
- Генерация псевдослучайного потока данных (KeyStream) и объединения данных

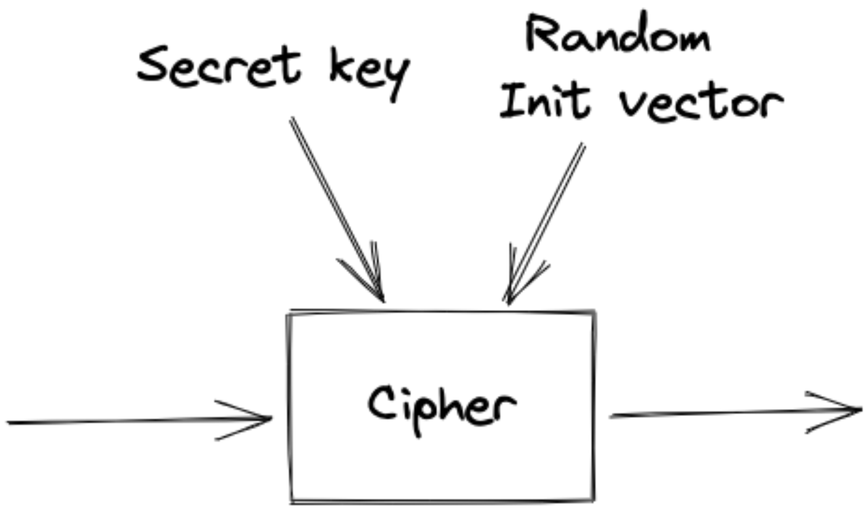


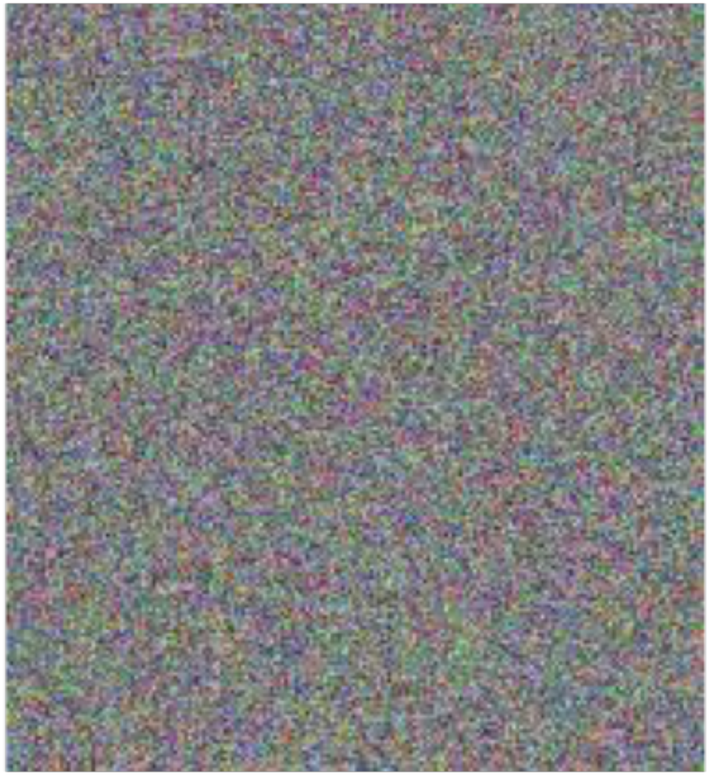
Использование случайной последовательности (IV) внутри алгоритма не позволяет вычислять корреляцию между данными

Без IV vs IV



Без IV vs IV





Случайные числа

Свойства случайных чисел

- Непредсказуемость
- Отсутствие связи с аргументом вызова
- Равномерное распределение

Случайные числа

Как генерируются рандомные числа?

- Pseudo-Random Number Generator (PRNG)
  - математические алгоритмы
- True Random Number Generator (TRNG)
  - аудио вход, различные датчики
  - энтропия в /dev/random (случайный пул)
  - радиоактивный источник

Требования к алгоритму шифрования

- Ключ шифра 256 bit
- Использование псевдослучайного вектора (IV) инициализации 128 bit
- IV должен быть уникален для каждого шифртекста

Алгоритмы шифрования

- Актуальные
  - AES CBC (самый популярный), AES GCM, AES CFB, AES OFB, AES CTR, Chacha20-Poly1305
- Устаревшие
  - AES ECB (128 bit), DES (64 bit), IDEA (128 bit), Blowfish (64 bit)

Алгоритмы шифрования

- Cipher Block Chaining (CBC) mode
  - iv
- Output Feedback (OFB) mode
  - iv
- Counter (CTR) mode
  - nonce
  - параллелизация шифра 🔥🔥🔥

Как проверить целостность шифра 🤔

Алгоритмы с блоком авторизации

AES Galois/Counter Mode (AES-GCM)

- Реализует CTR mode и AEAD
- Индустриальный стандарт
- Эффективный алгоритм + hardware

ChaCha20-Poly1305 (RFC-8439)

- Реализует CTR mode и AEAD
- ChaCha20 алгоритм шифрования
- Poly1305 функция MAC
- Быстрее AES-GCM
- Включен в TLS

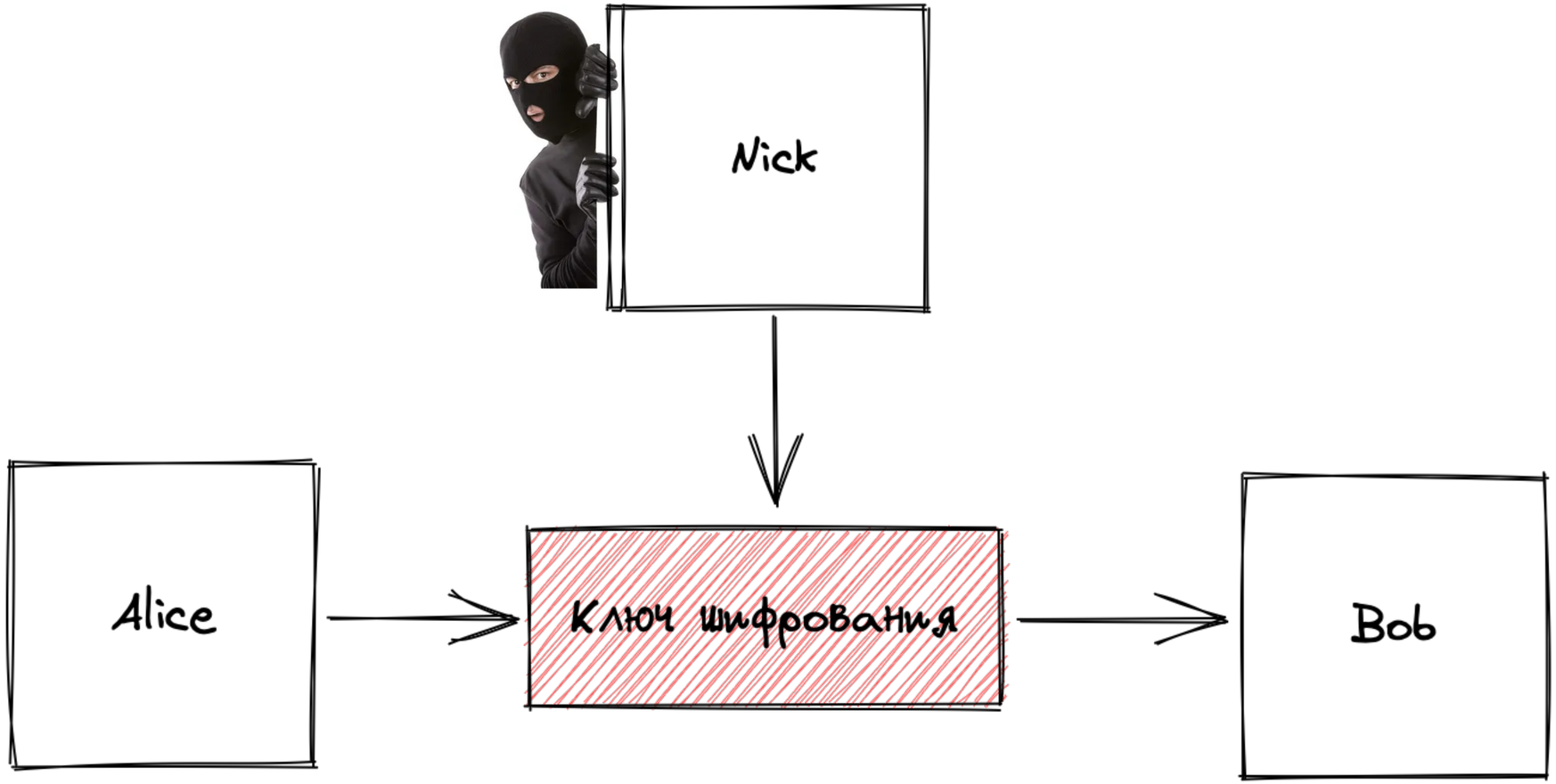
Общие рекомендации

- Используйте AES-GCM или ChaCha20-Poly1305 с размером ключа 256 бит и IV размером 128 бит
- IV должен быть уникален для каждого шифр текста

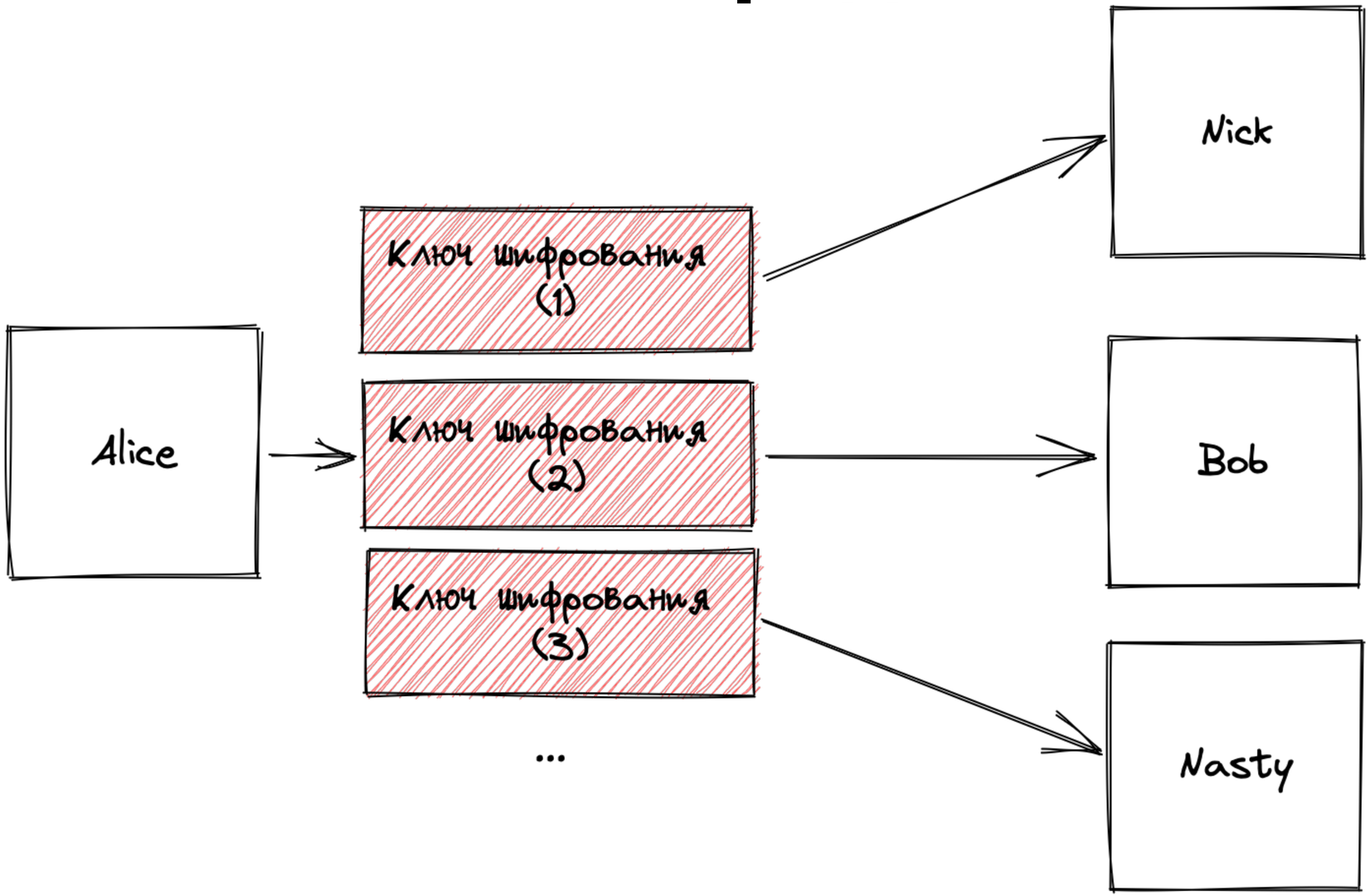
Фундаментальная проблема: как безопасно передать ключ?

Как безопасно передать ключ?

Как безопасно передать ключ?

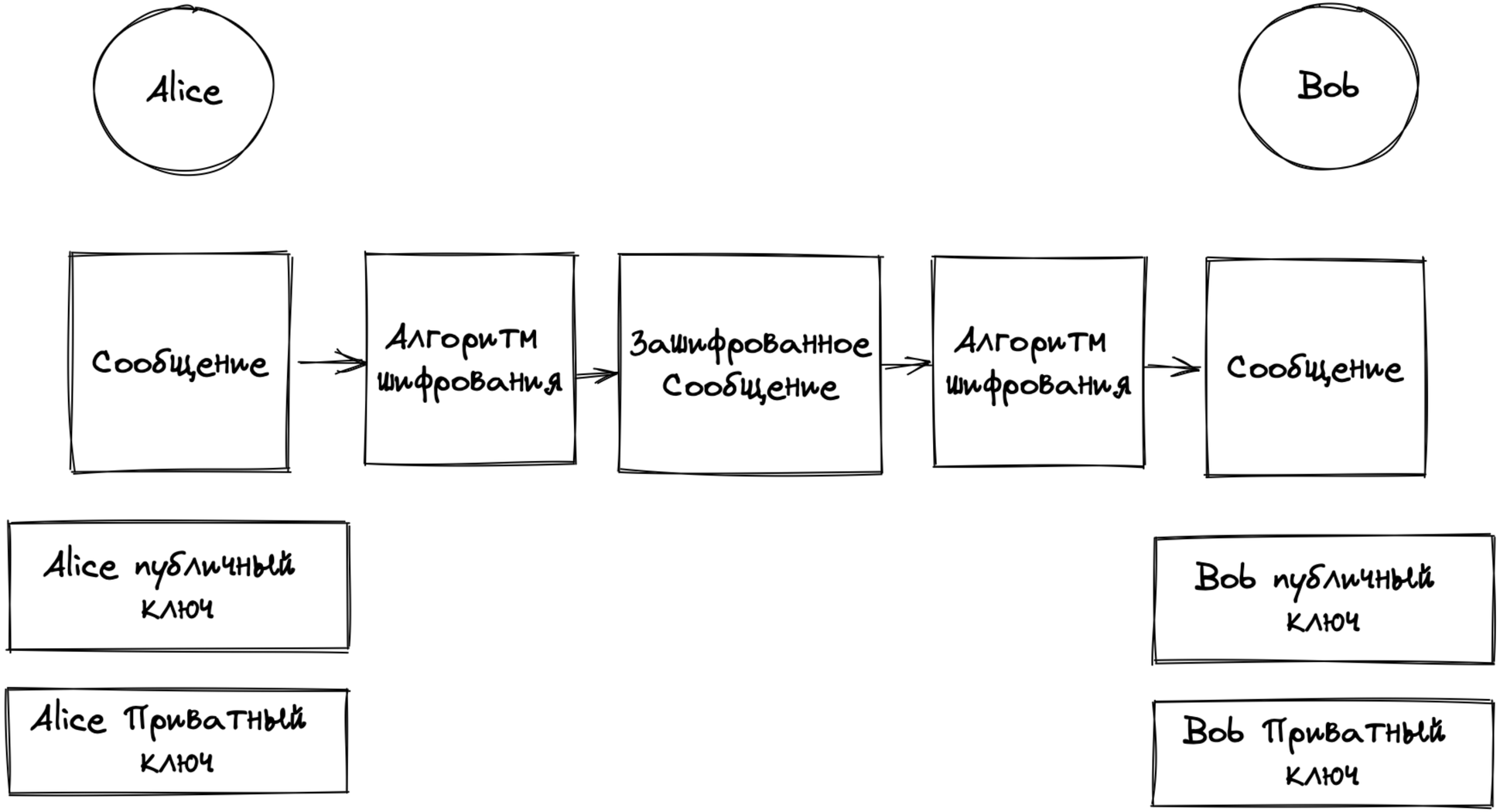


Как безопасно передать ключ?

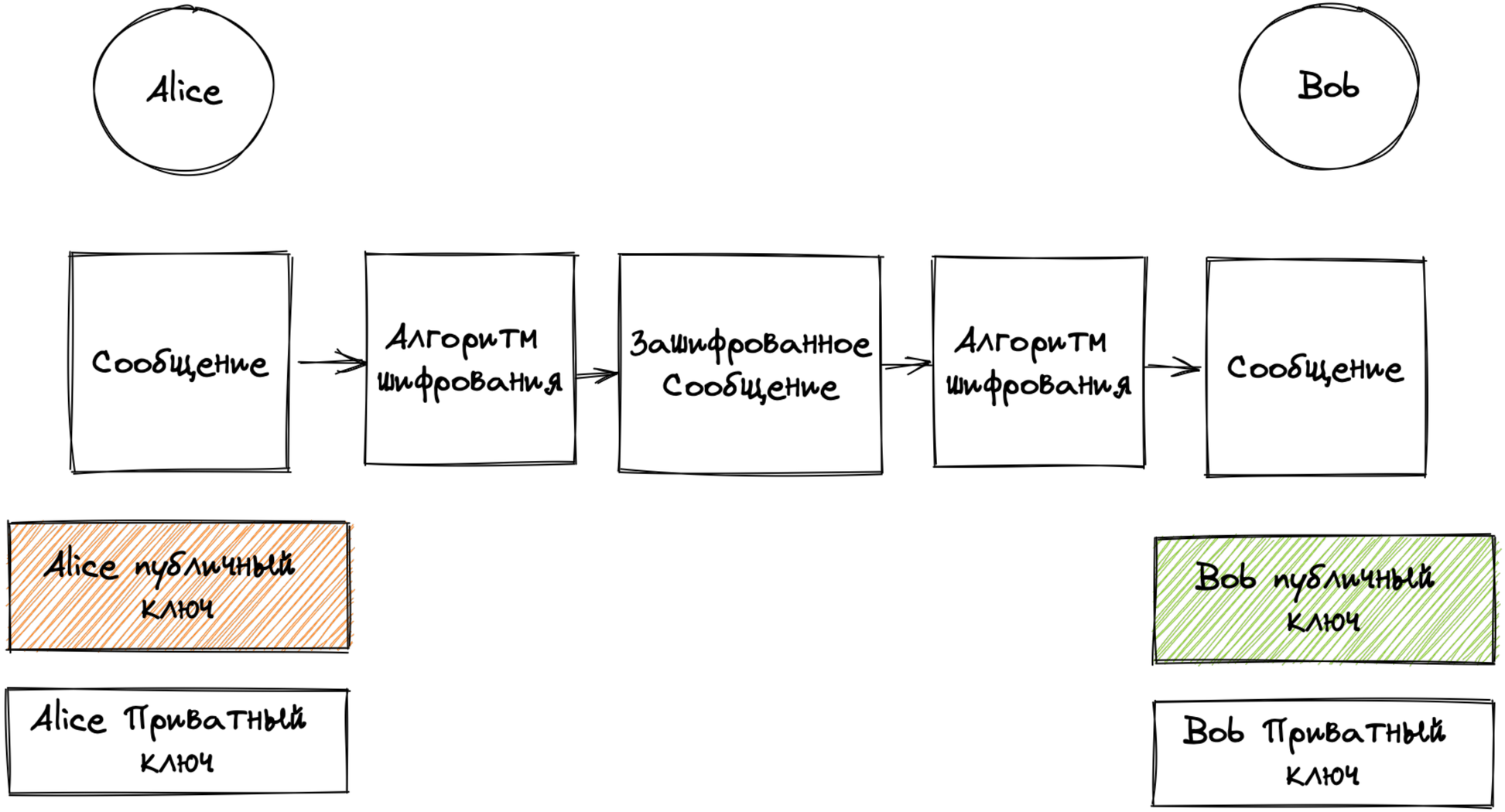


Асимметричное шифрование

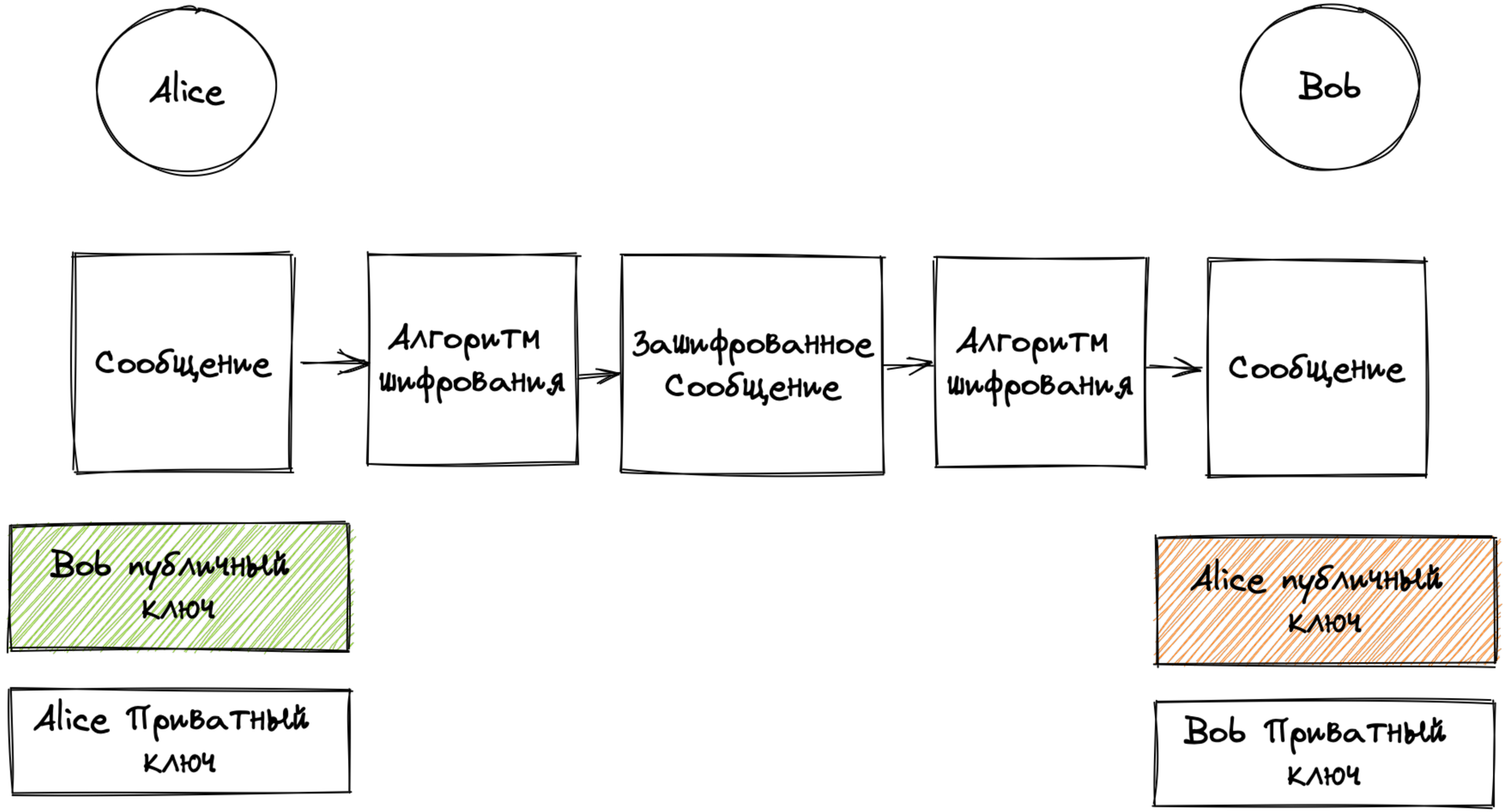
Асимметричное шифрование



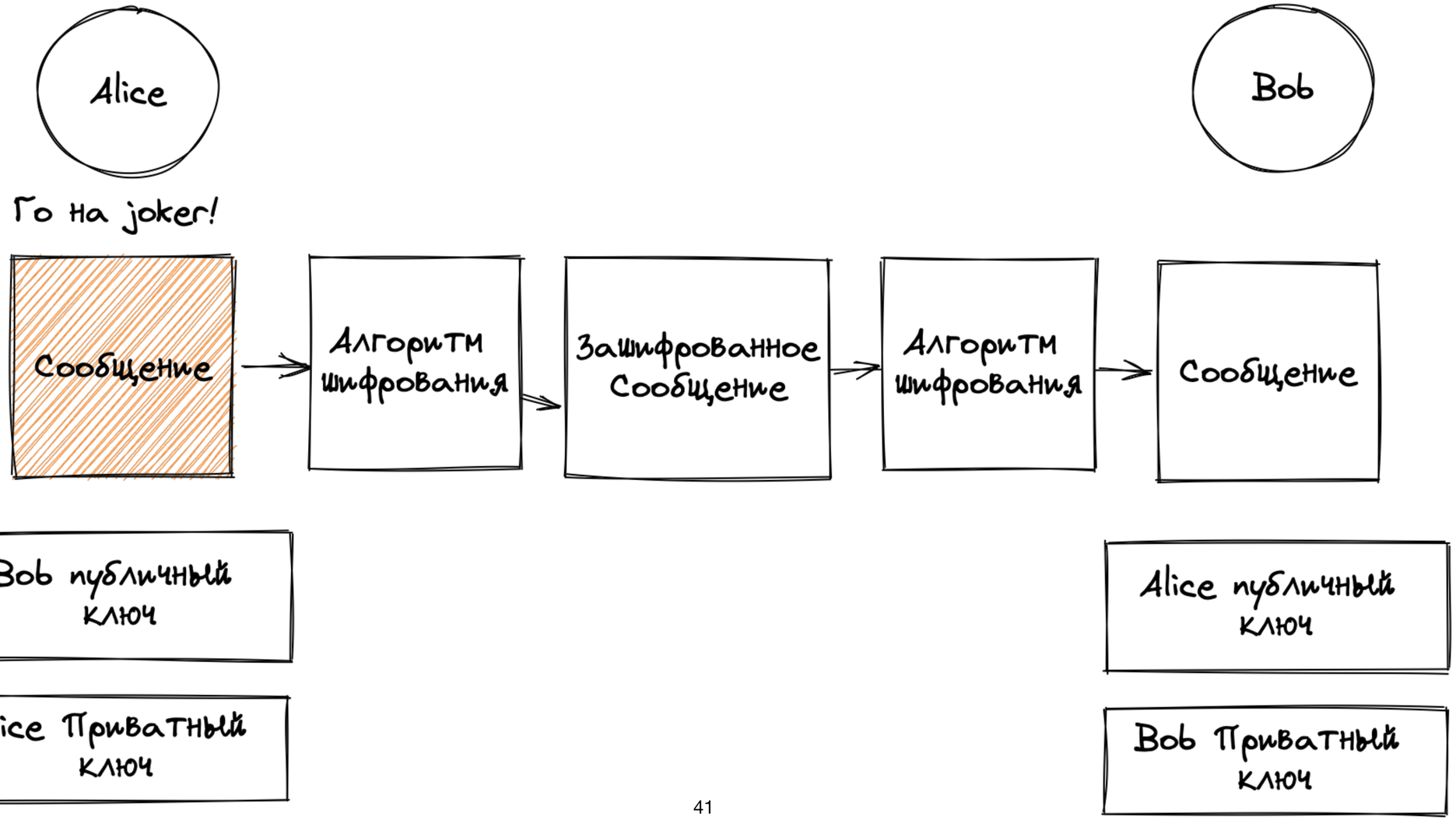
Асимметричное шифрование



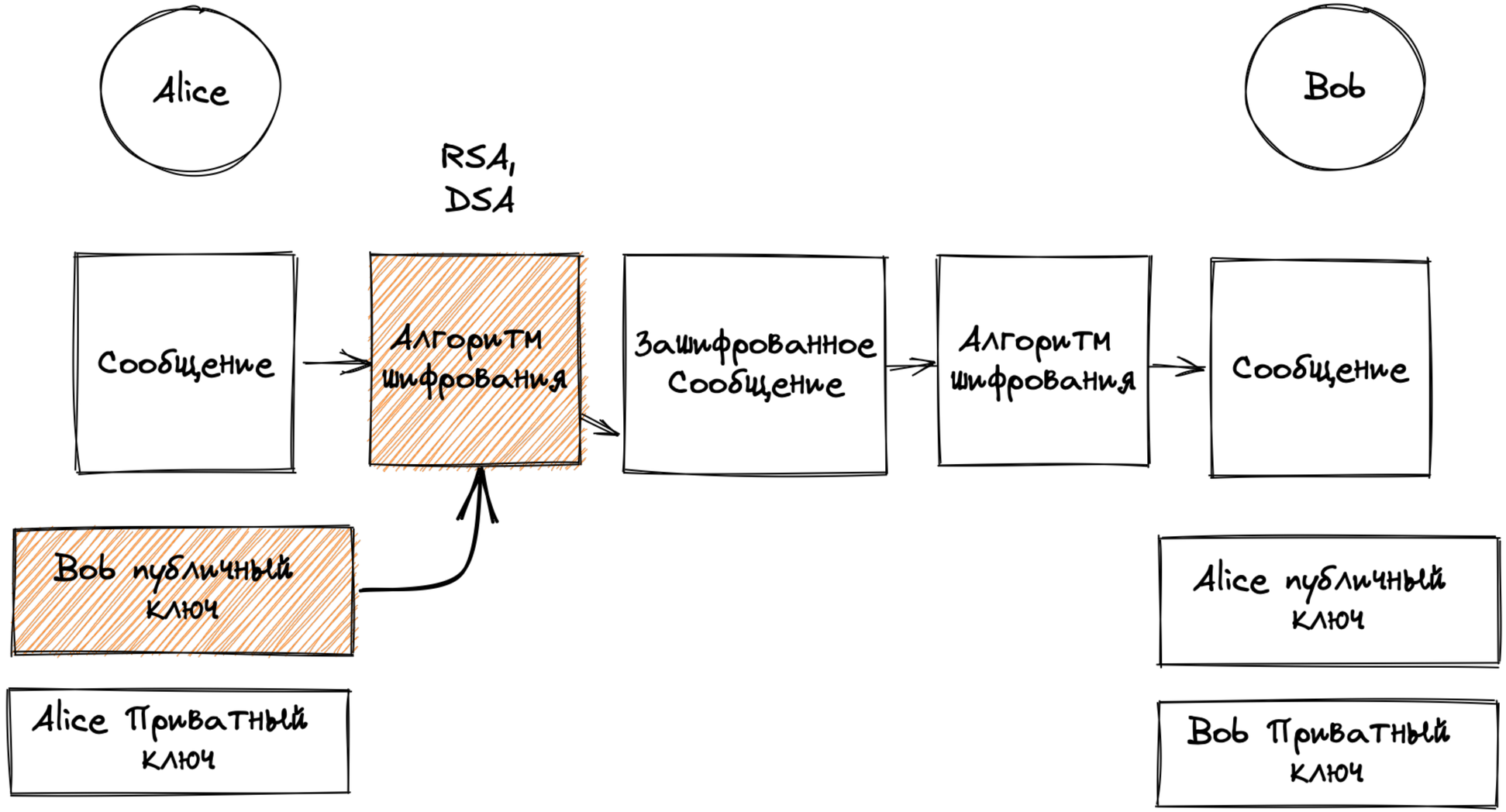
Асимметричное шифрование



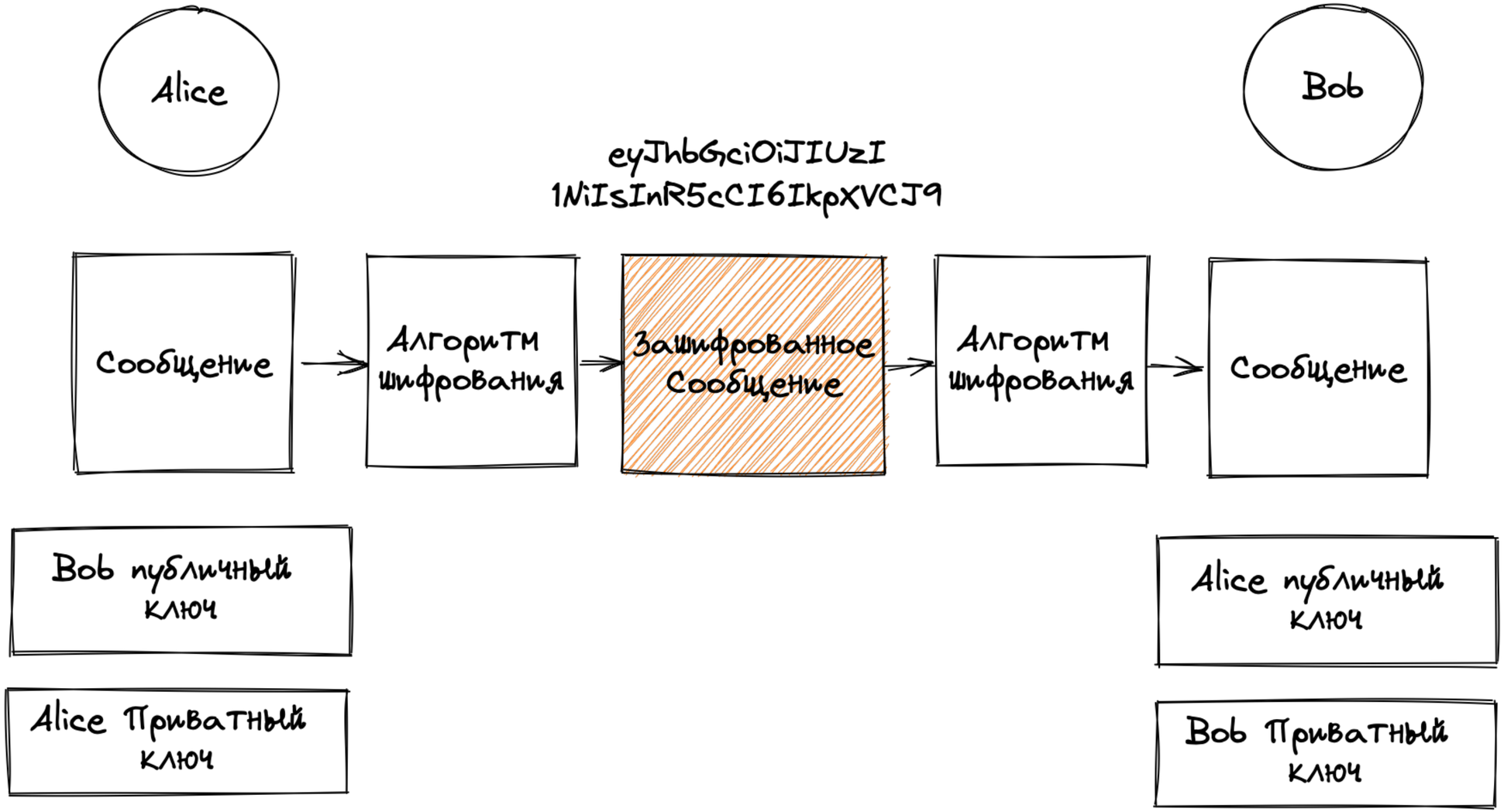
Асимметричное шифрование



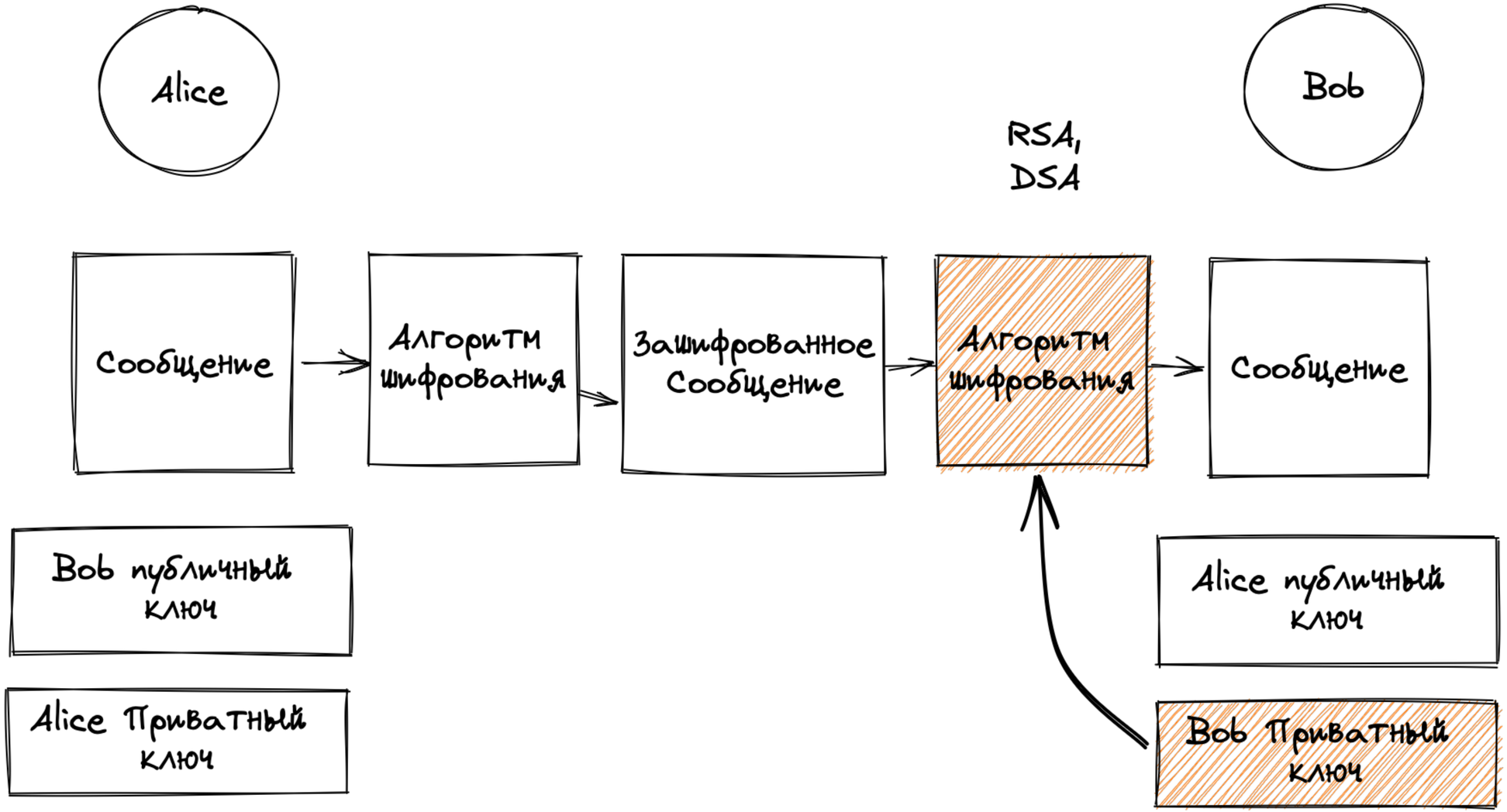
Асимметричное шифрование



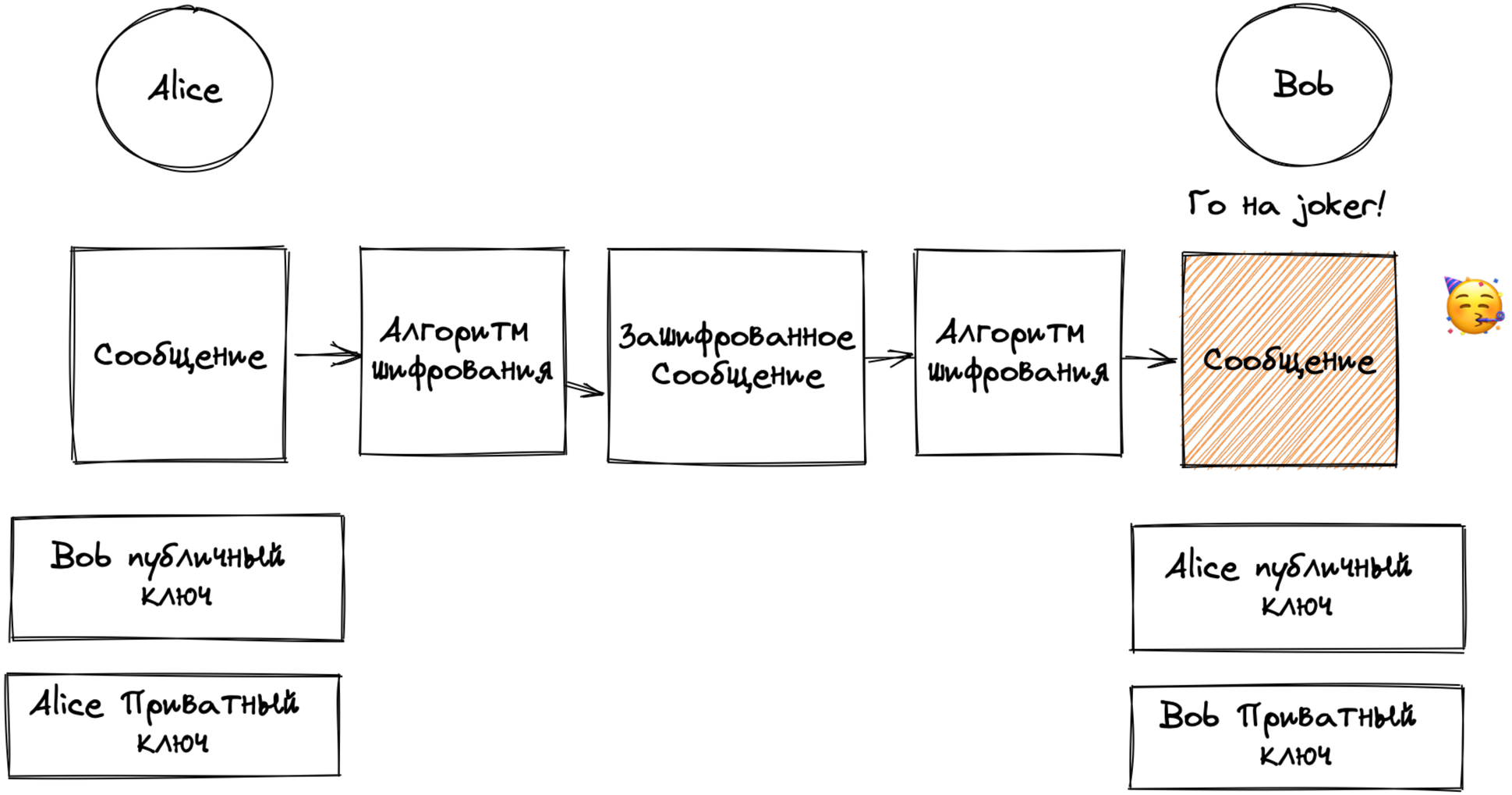
Асимметричное шифрование



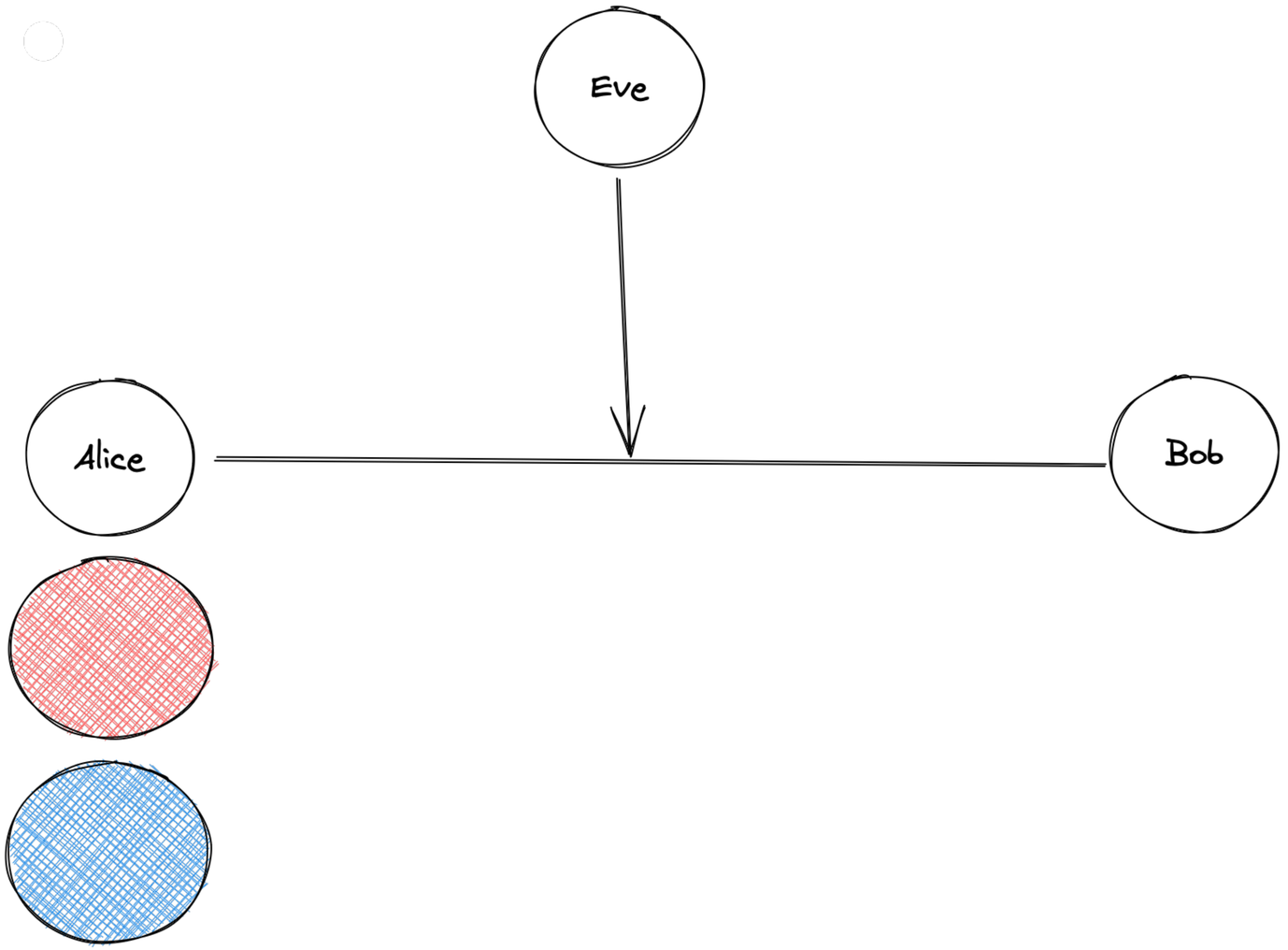
Асимметричное шифрование



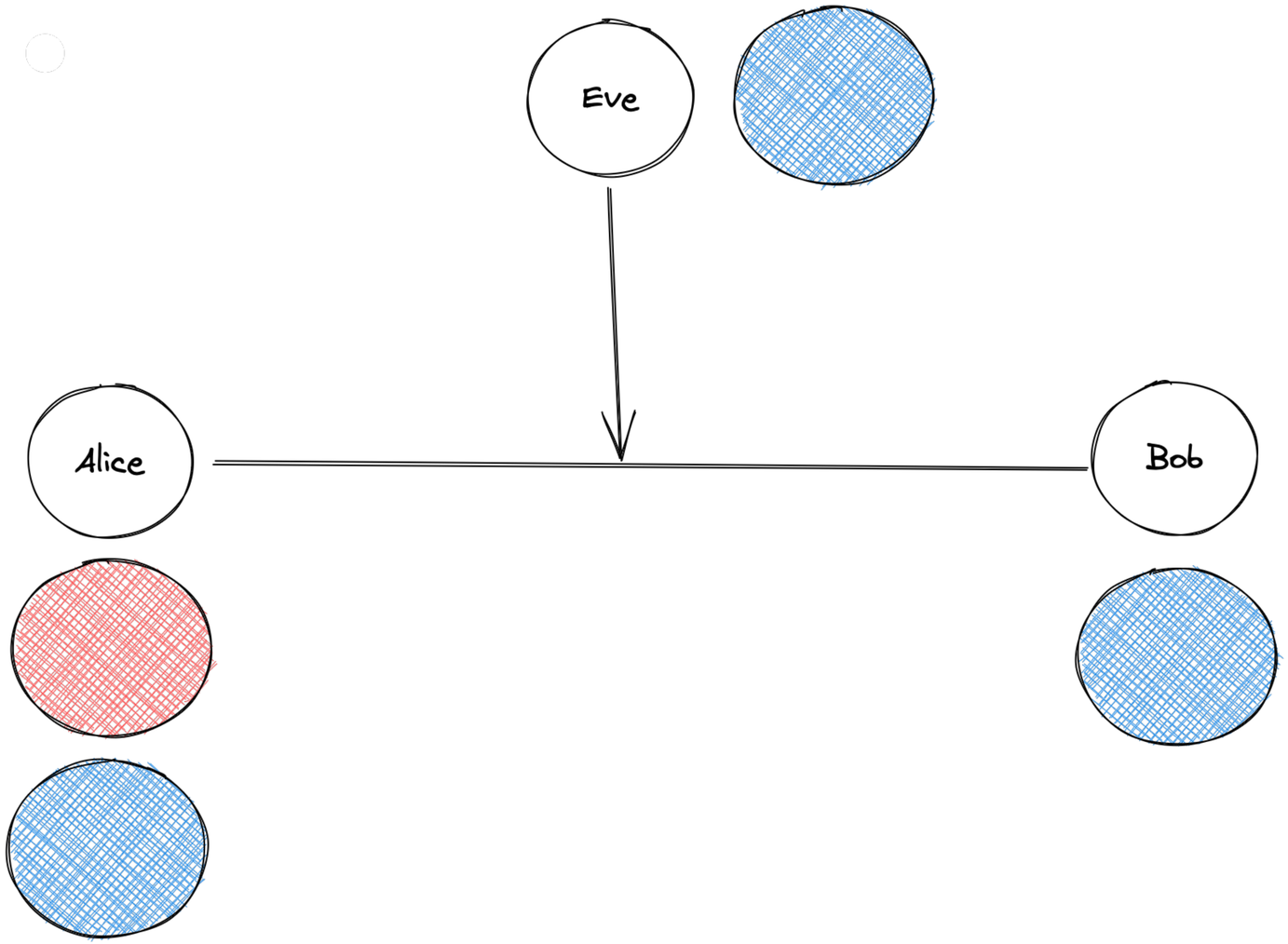
Асимметричное шифрование



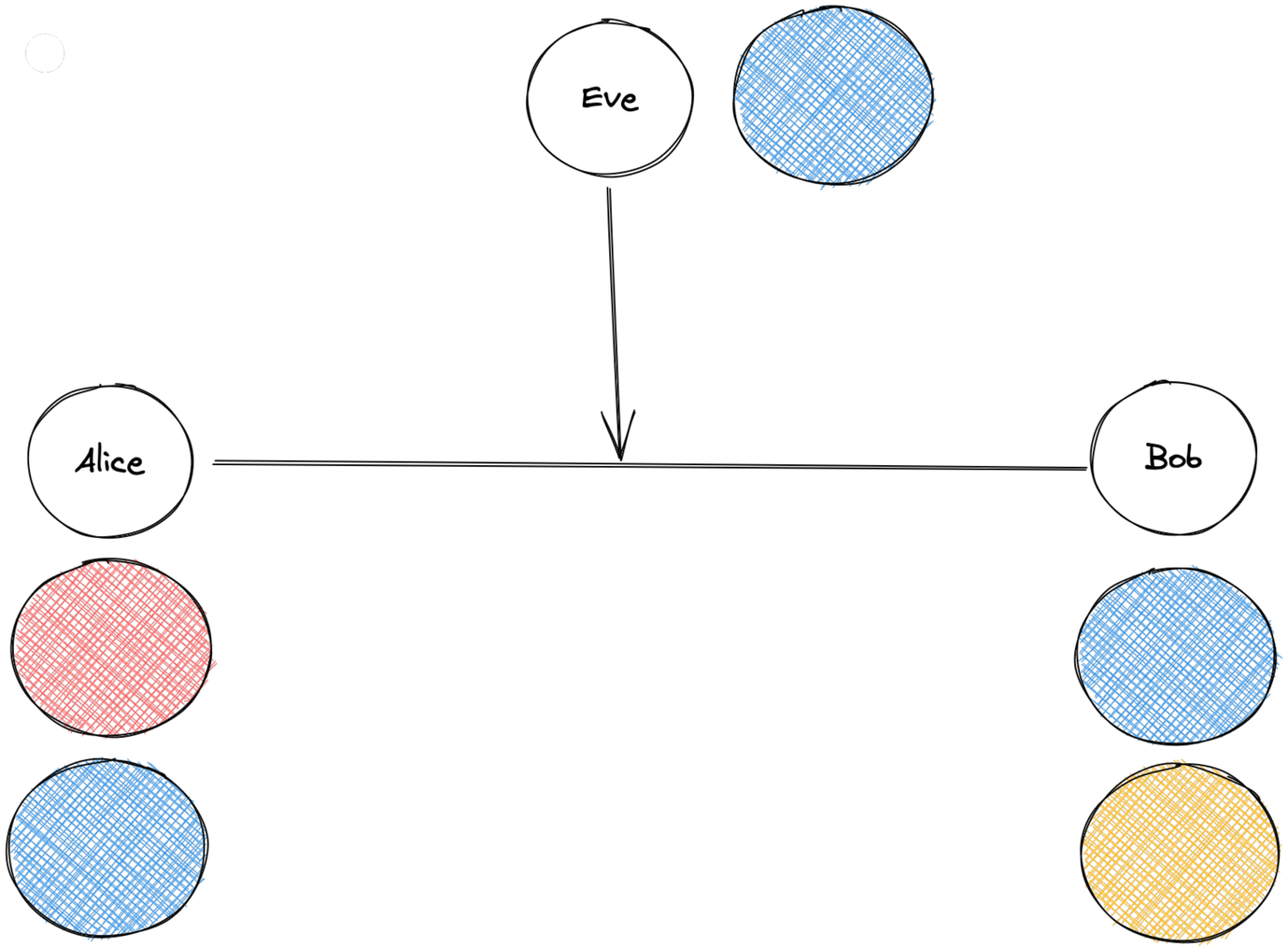
Асимметричное шифрование



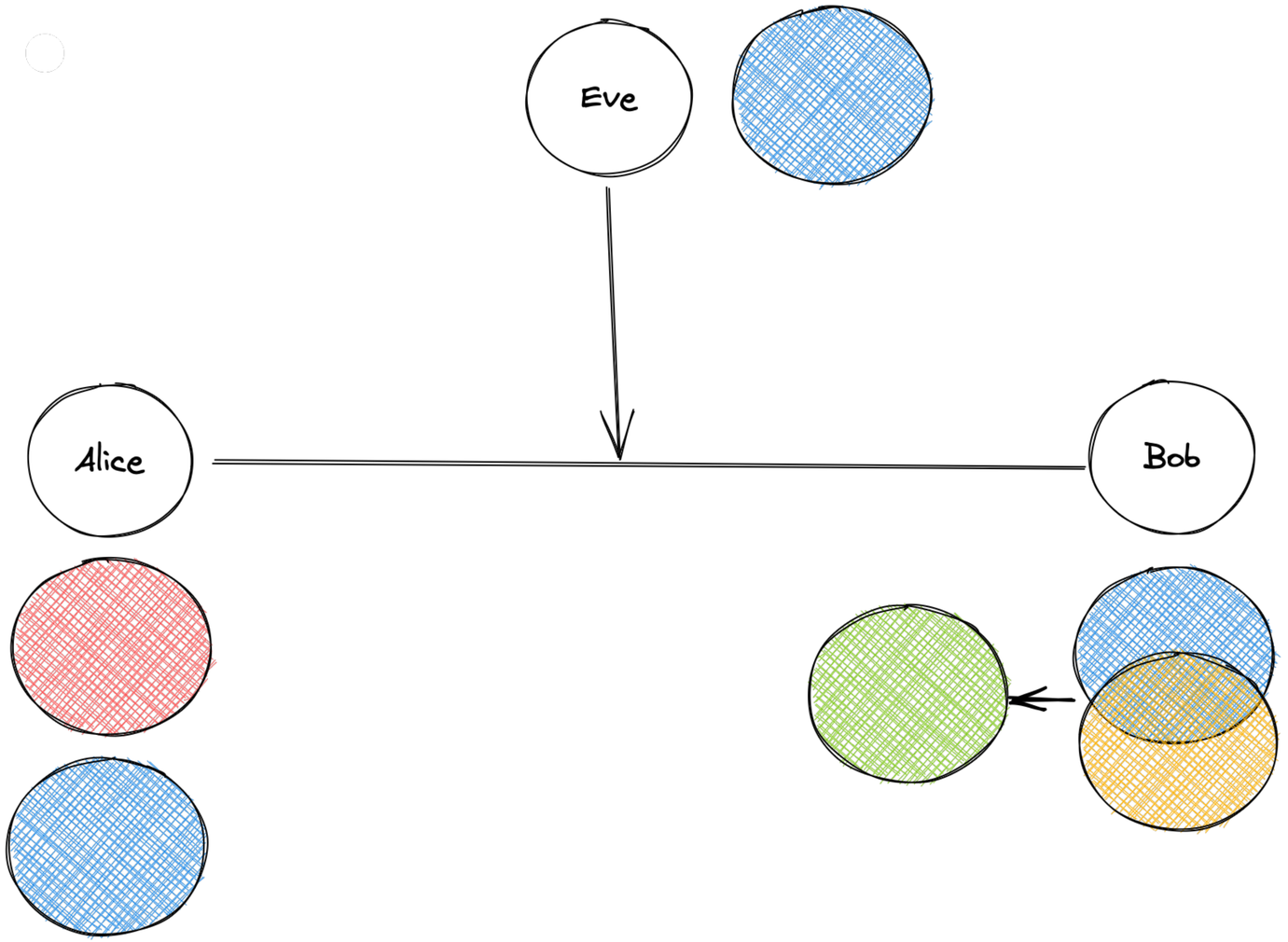
Асимметричное шифрование



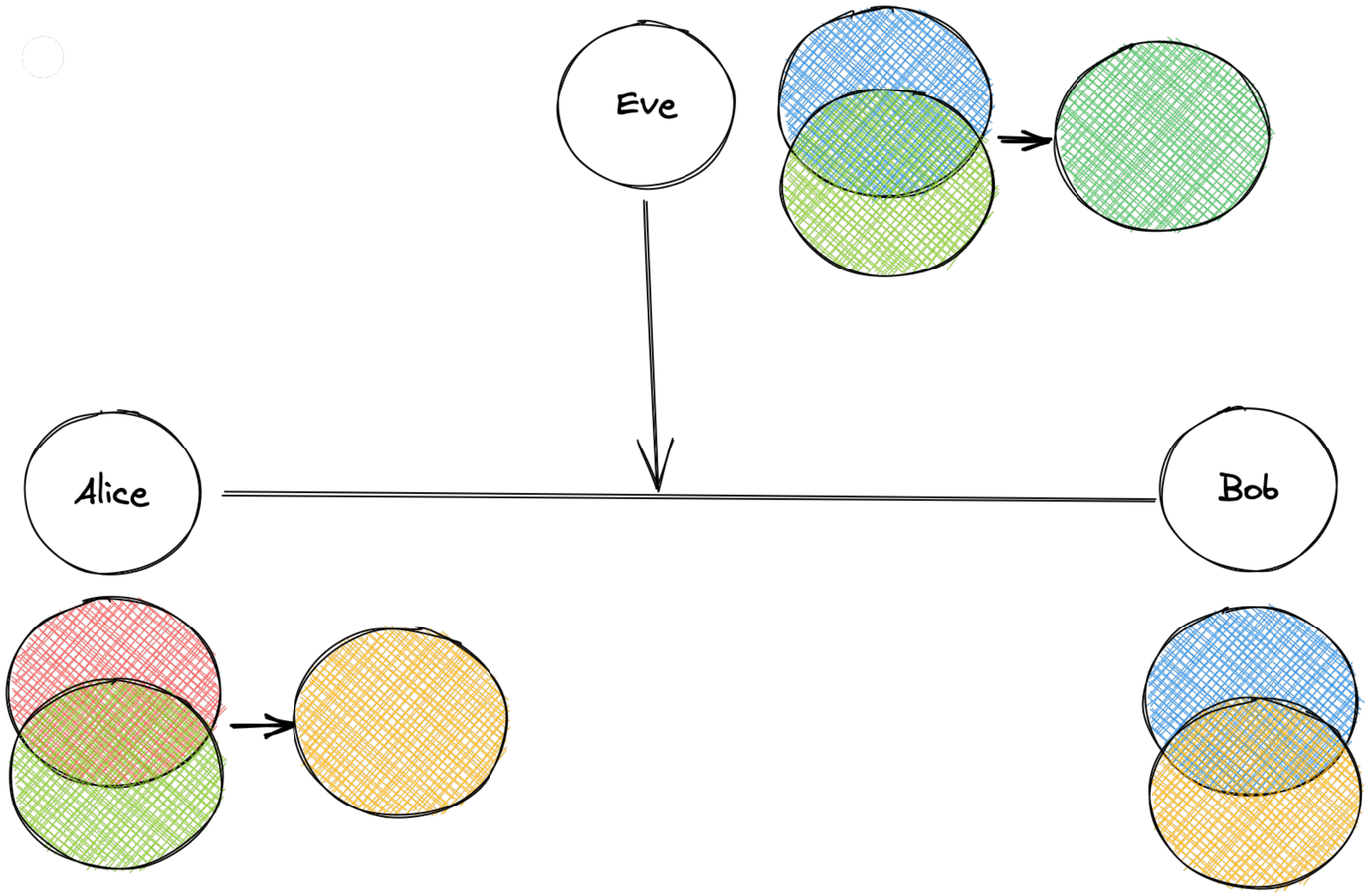
Асимметричное шифрование



Асимметричное шифрование



Асимметричное шифрование



Асимметричное шифрование

Примеры

- Интернет
  - HTTPS
  - TLS*
  - SSL

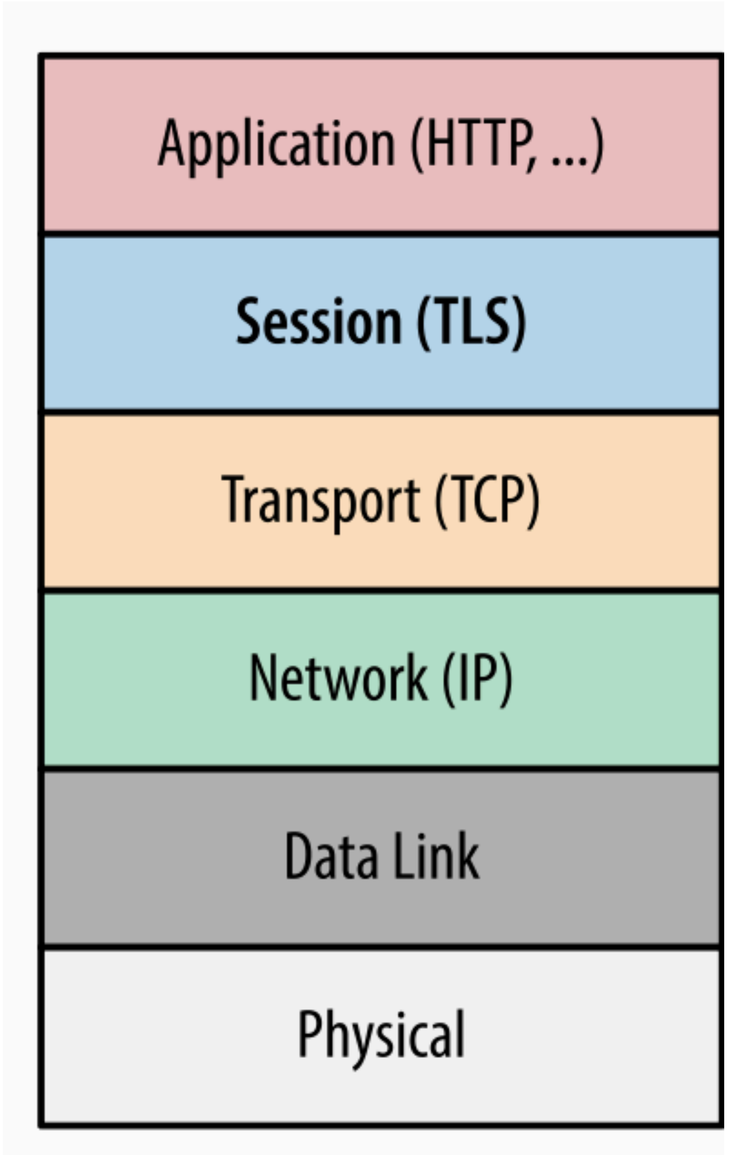
Алгоритмы асимметричного шифрования

- RSA
- DSA (Digital Signature Algorithm)
- ECDSA (Elliptic Curve Digital Signature Algorithm)

Шифрование

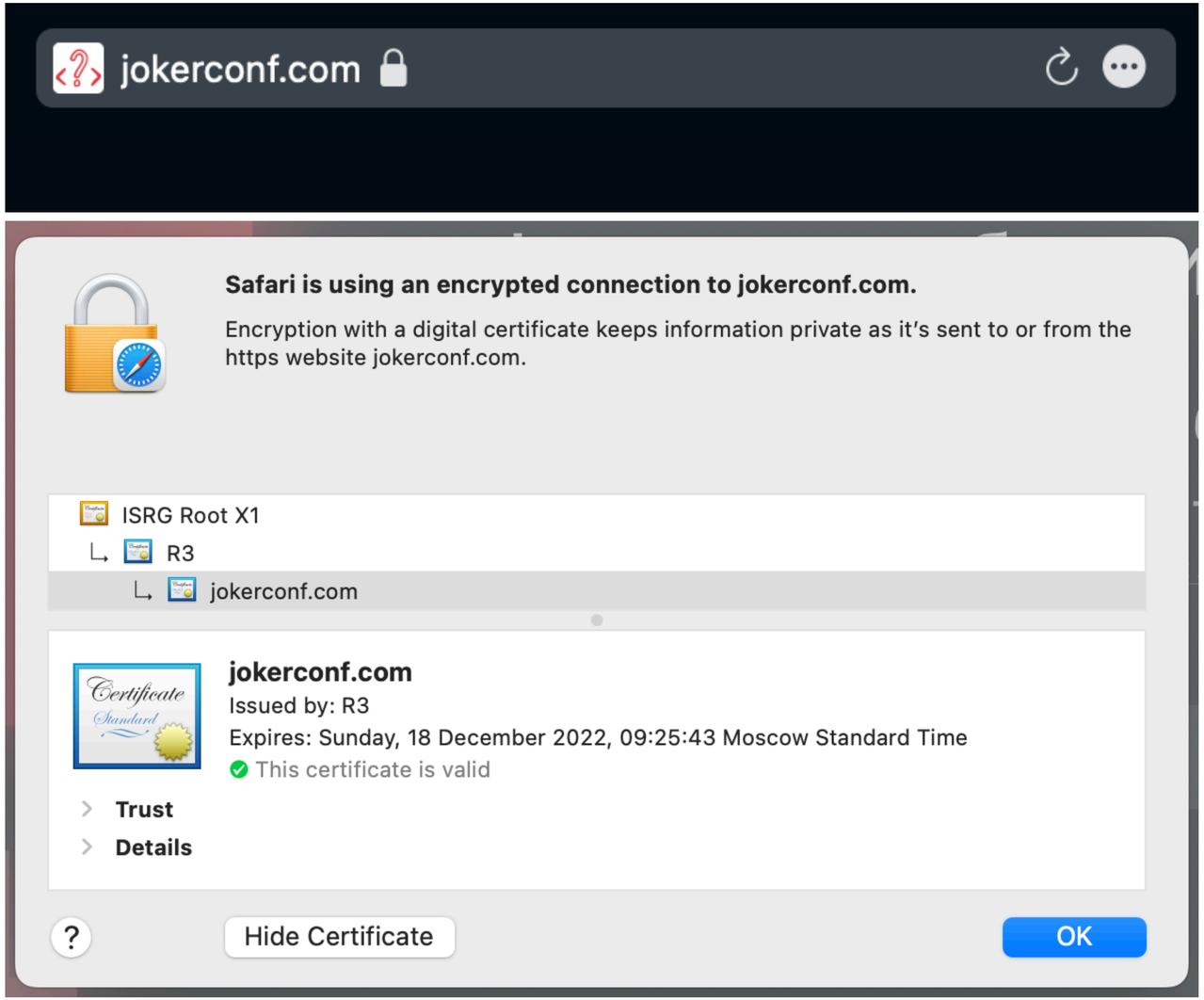
- Симметричное шифрование
  - Быстрее - слабая нагрузка на CPU
  - PlainText и CipherText имеют одинаковую длину
  - Секретный ключ надо как то передать
- Асимметричное шифрование
  - Медленнее - требуется более длинный ключ
  - CipherText гораздо больше чем PlainText
  - Секретный ключ не надо передавать, есть публичный ключ

TLS

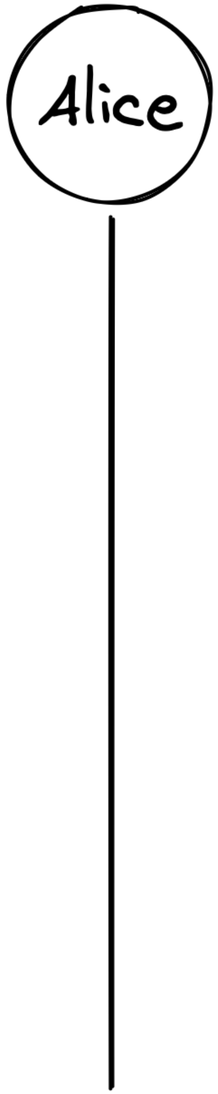

Application (HTTP, ...)
Session (TLS)
Transport (TCP)
Network (IP)
Data Link
Physical

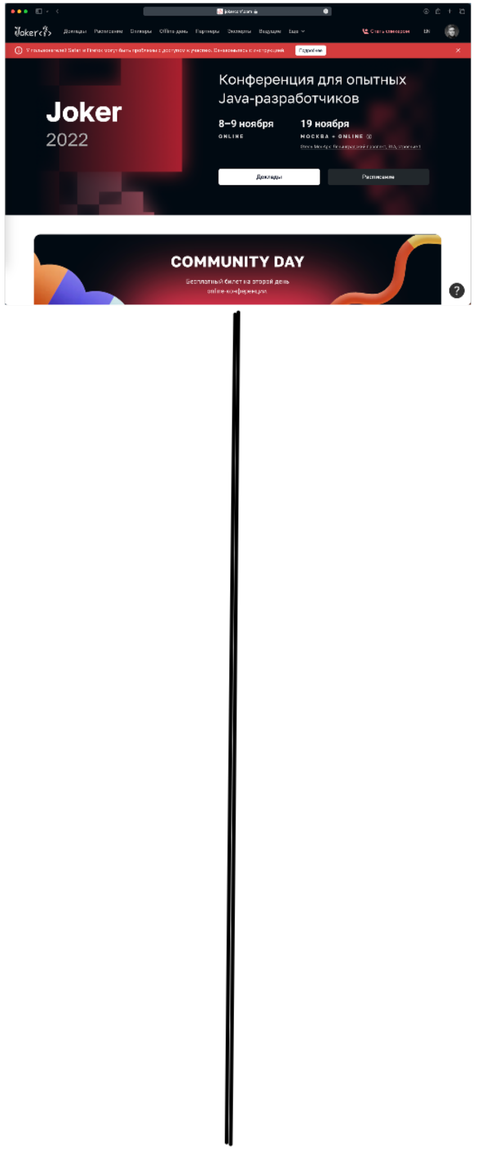
TLS + HTTP = HTTPS

TLS


jokerconf.com
Safari is using an encrypted connection to jokerconf.com.
Encryption with a digital certificate keeps information private as it's sent to or from the https website jokerconf.com.
ISRG Root X1
R3
jokerconf.com
jokerconf.com
Issued by: R3
Expires: Sunday, 18 December 2022, 09:25:43 Moscow Standard Time
This certificate is valid
Trust
Details
Hide Certificate
OK

TLS

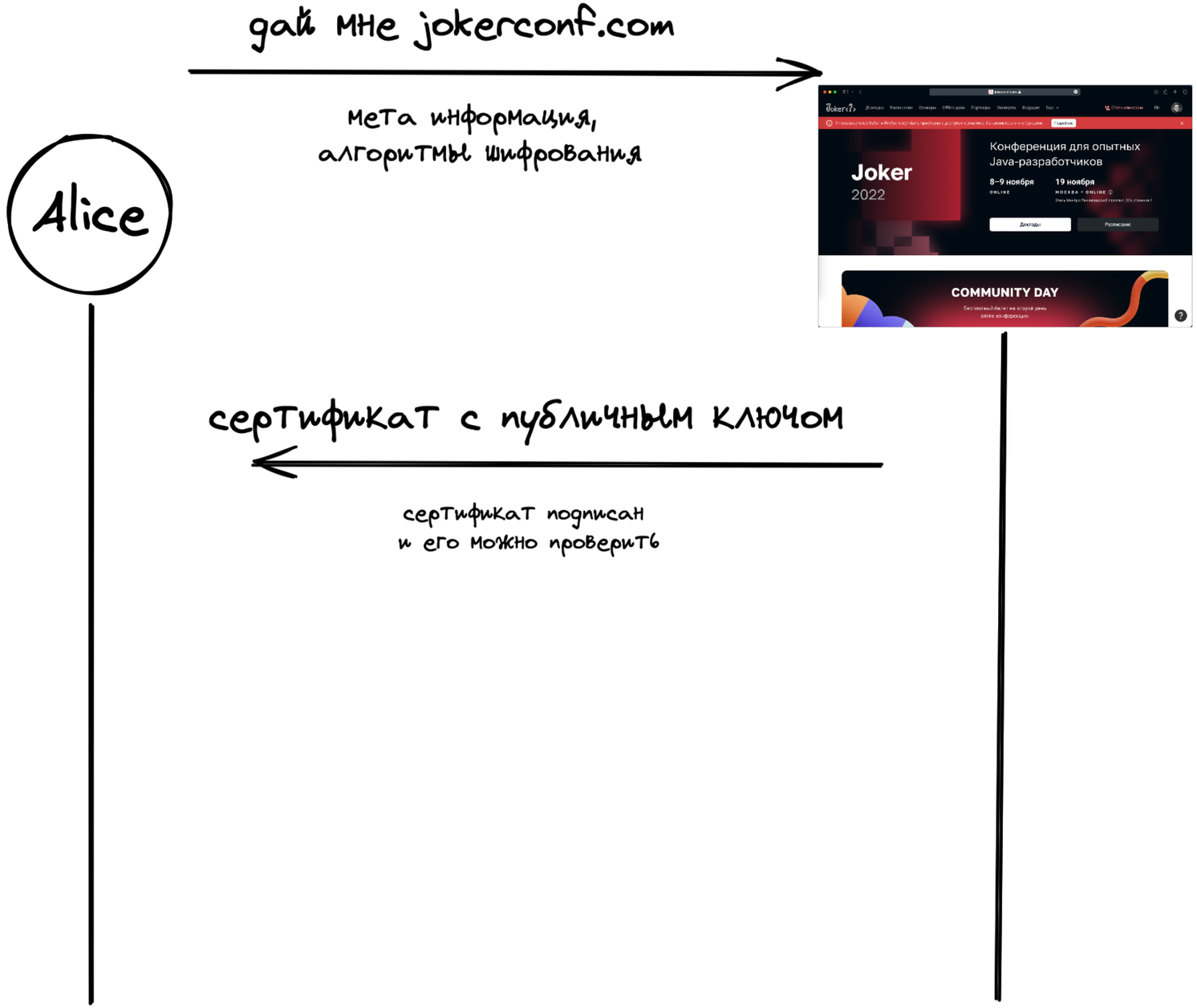

Alice


Joker
2022
Конференция для опытных
Java-разработчиков
8–9 ноября
19 ноября
COMMUNITY DAY

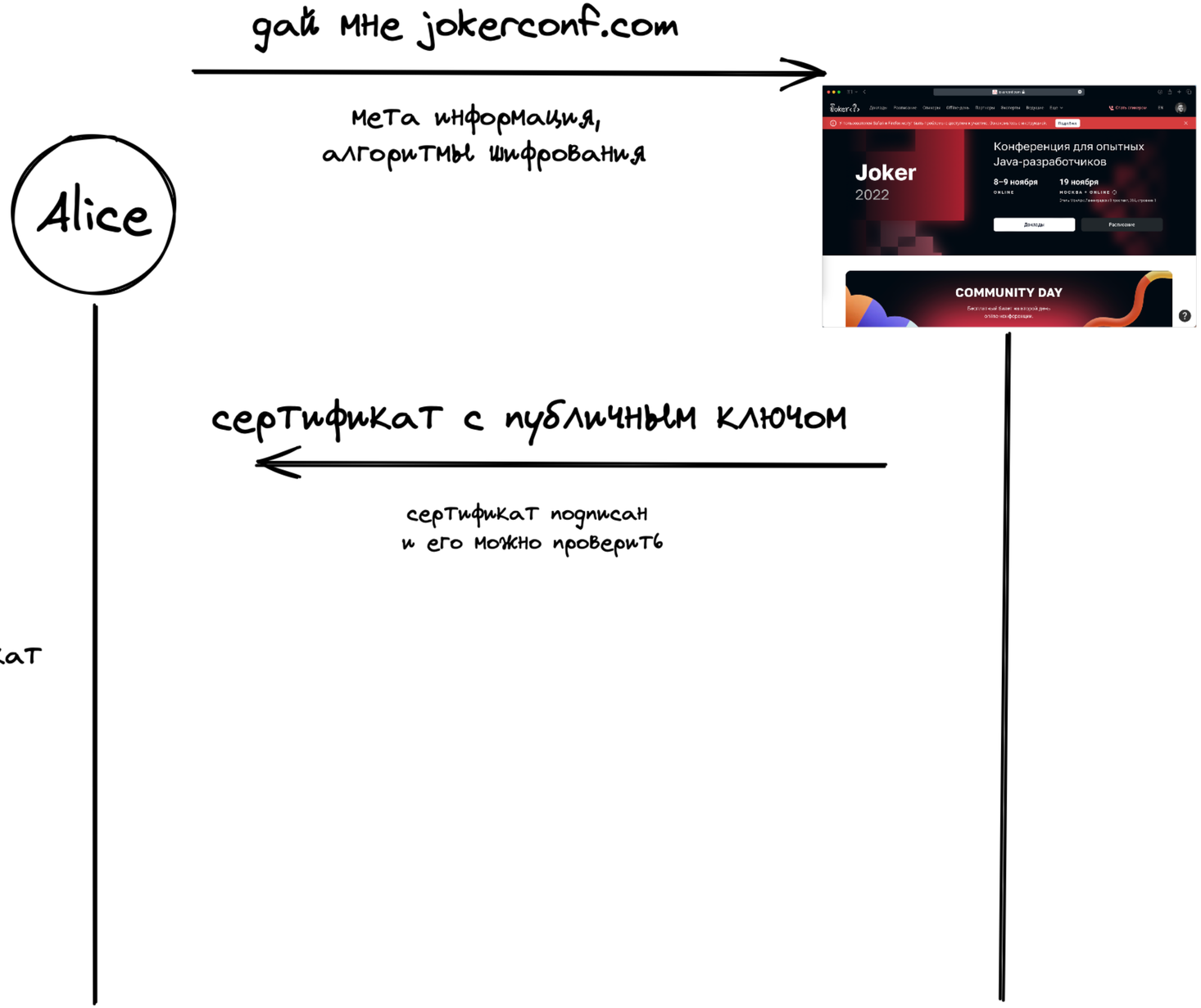
TLS


дай мне jokerconf.com
мета информация,
алгоритмы шифрования
Alice
Joker
2022
Конференция для опытных
Java-разработчиков
8–9 ноября
ONLINE
19 ноября
Доклады
Расписание
COMMUNITY DAY

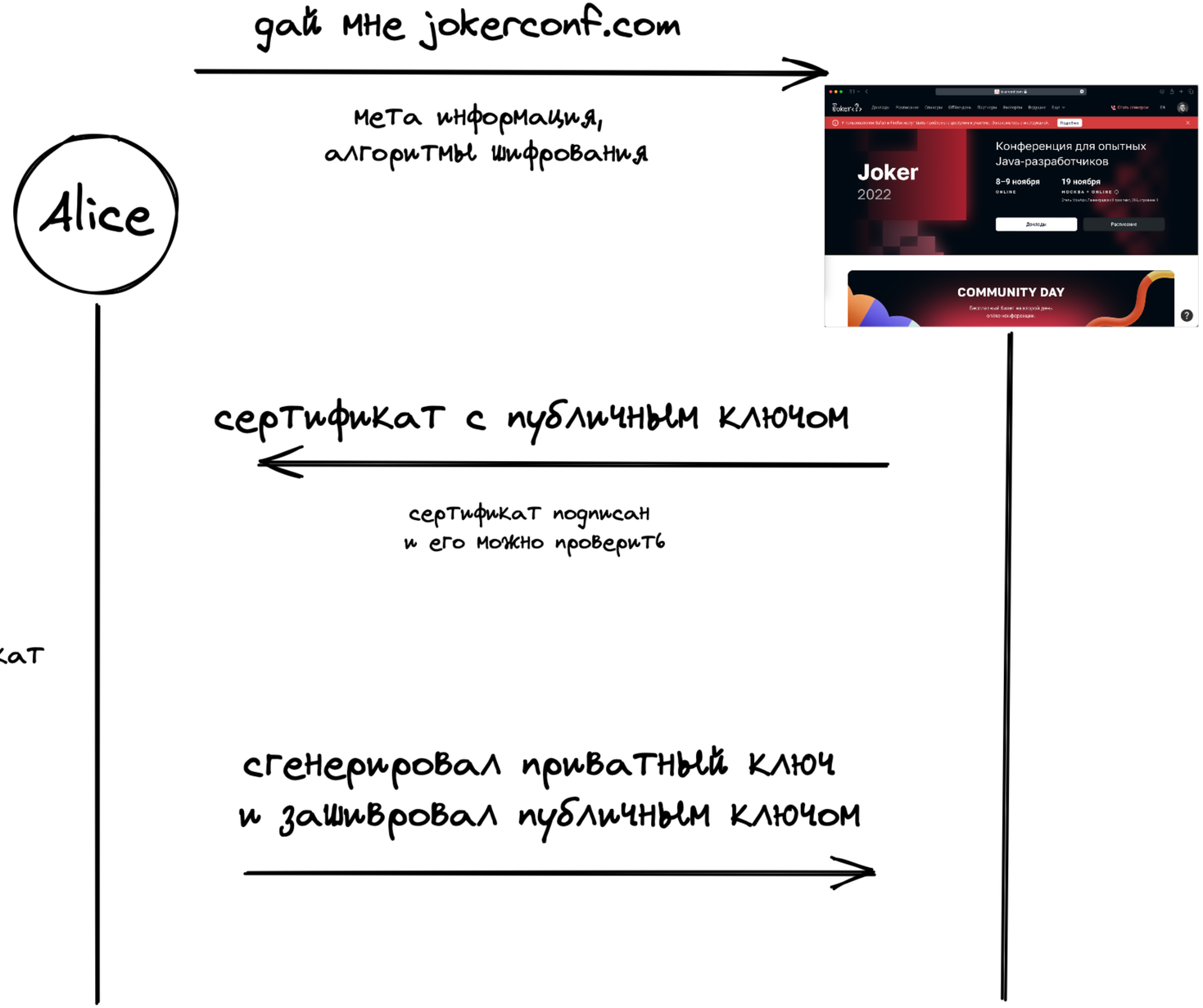
TLS


дай мне jokerconf.com
мета информация,
алгоритмы шифрования
Alice
Joker
2022
Конференция для опытных
Java-разработчиков
8–9 ноября
ONLINE
19 ноября
МОСКВА + ONLINE
Доклады
Расписание
COMMUNITY DAY
сертификат с публичным ключом
сертификат подписан
и его можно проверить

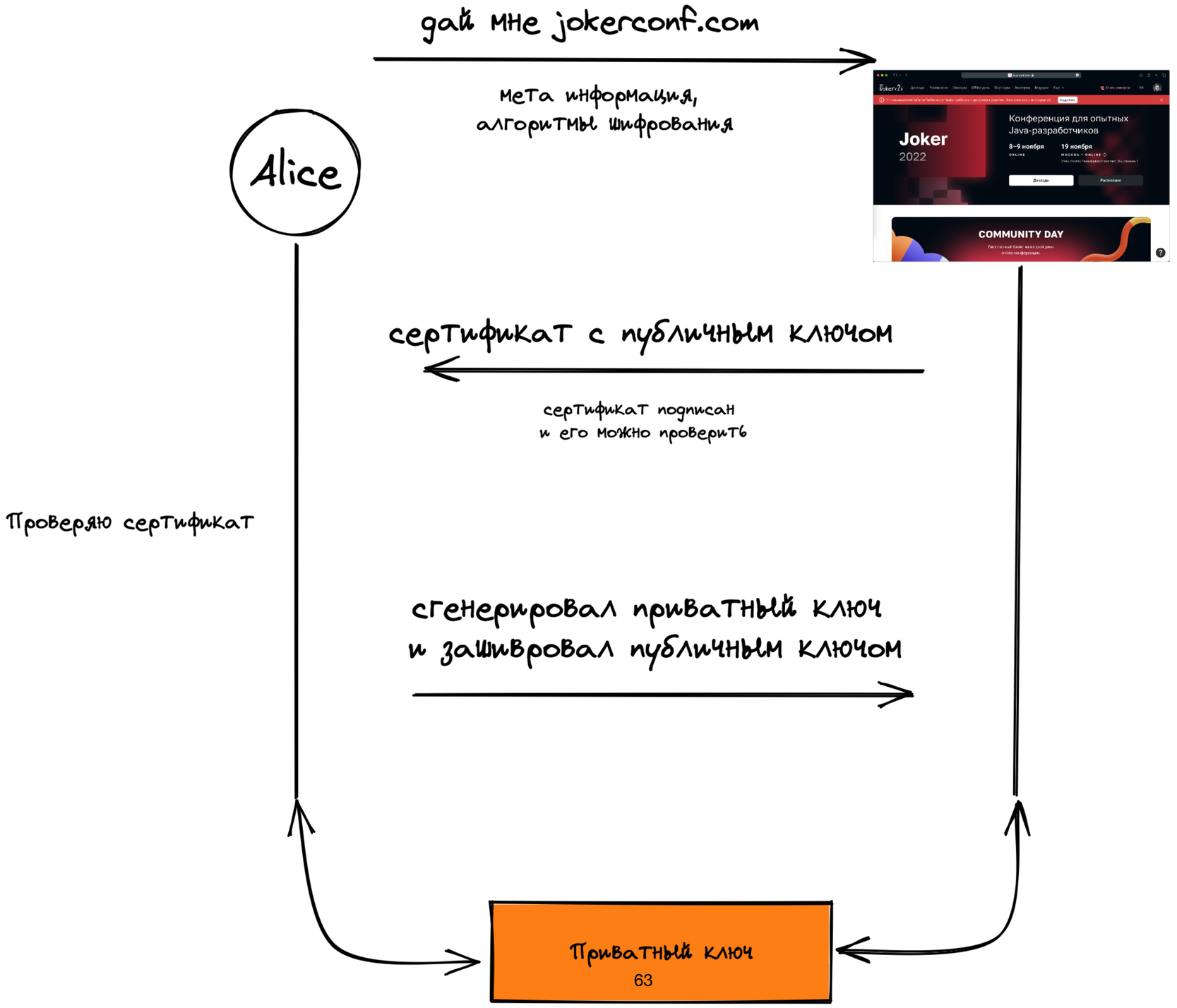
TLS


дай мне jokerconf.com
мета информация,
алгоритмы шифрования
Alice
Joker
2022
Конференция для опытных
Java-разработчиков
8–9 ноября
ONLINE
19 ноября
COMMUNITY DAY
сертификат с публичным ключом
сертификат подписан
и его можно проверить
Проверяю сертификат

TLS


дай мне jokerconf.com
мета информация,
алгоритмы шифрования
Alice
Joker
2022
Конференция для опытных
Java-разработчиков
8–9 ноября
ONLINE
19 ноября
МОСКВА + ONLINE
COMMUNITY DAY
сертификат с публичным ключом
сертификат подписан
и его можно проверить
Проверяю сертификат
сгенерировал приватный ключ
и зашифровал публичным ключом

TLS


дай мне jokerconf.com
мета информация,
алгоритмы шифрования
Alice
Joker
2022
Конференция для опытных
Java-разработчиков
8–9 ноября
19 ноября
COMMUNITY DAY
сертификат с публичным ключом
сертификат подписан
и его можно проверить
Проверяю сертификат
сгенерировал приватный ключ
и зашифровал публичным ключом
Приватный ключ

Useful links

- ChaCha20-Poly1305 https://datatracker.ietf.org/doc/html/rfc8439
- AES-GCM https://www.rfc-editor.org/rfc/rfc7714.html
- AES-CBC https://www.rfc-editor.org/rfc/rfc3602.html
- Random https://www.random.org/randomness/

Java 17

Oracle JRE and JDK Cryptographic Roadmap

Released Changes

Release Date	Release(s) Affected	Algorithm/Protocol	Action	Additional Information	Change Log
2022-10-18	17.0.5+9, 11.0.17+10, 8u351 b10	SHA-1	Disabled SHA-1 JARs signed after 2019-01-01	Disabling SHA-1 signed jars (updated)	• 2022-10-18 Released. • 2022-04-19 Target date narrowed from 2022 to 2022-10-18. • 2021-10-19 Updated Additional Information. • 2021-06-11 Target date changed from 2021-07-20 to 2022. • 2020-12-18 Target date changed from 2021-04-20 to 2021-07-20. • 2020-08-24 Announced.
2022-10-18	11.0.17+10, 8u351 b10	SHA-2	Upgraded the default PKCS12 MAC algorithm	Upgrading the default PKCS12 MAC algorithm	• 2022-10-18 Released. • 2022-04-19 Target date narrowed from 2022 H2 to 2022-10-18. • 2021-06-11 Announced.
2022-10-18	11.0.17+10, 8u351 b10	3DES, RC4	Disabled 3DES and RC4 in Kerberos	Disabling 3DES and RC4 in Kerberos	• 2022-10-18 Released. • 2022-04-19 Announced.
2022-07-19	8u341 b10	TLSv1.3	Enabled TLSv1.3 by default on the client	Enabling TLSv1.3 by default on the client	• 2022-07-19 Released. • 2021-10-19 Announced.

Oracle JRE and JDK Cryptographic Roadmap

Уже вышли

- TLS v1.3 default
- ChaCha20-Poly1305 by Default for TLS
- PKCS12 Keystores by Default

Oracle JRE and JDK Cryptographic Roadmap

Черновики

- Certificate Transparency
- Distributed TLS Sessions
- Edwards-Curve Digital Signature Algorithm (EdDSA)

Стоит обратить внимание 🤔

Стоит обратить внимание 🤔

- Поиск по зашифрованному полю
- Ротация ключей шифрования
- Ротация алгоритма шифрования

Поиск по зашифрованному полю

Поиск по зашифрованному полю

Id	User_id	User_token	Create_at
1	43	[111010...10]	2022-05-21 13:00
1	46	[111010...10]	2022-05-20 10:00
1	45	[111010...10]	2022-05-11 15:00
1	44	[111010...10]	2022-05-06 14:00
1	42	[111010...10]	2022-05-06 10:00

```
select * from user_data where user_token = ?
```

Поиск по зашифрованному полю

- Надо шифровать публичный идентификатор
- Использовали алгоритм шифрования 128 битный (без IV)
- Понимаете что данное поле стоит начать шифровать
- Для ускорения существующих запросов надо знать значение шифрованного поля

Хэширование - необратимое преобразование информации произвольной длины в блок данных фиксированной длины с целью проверки целостности данных

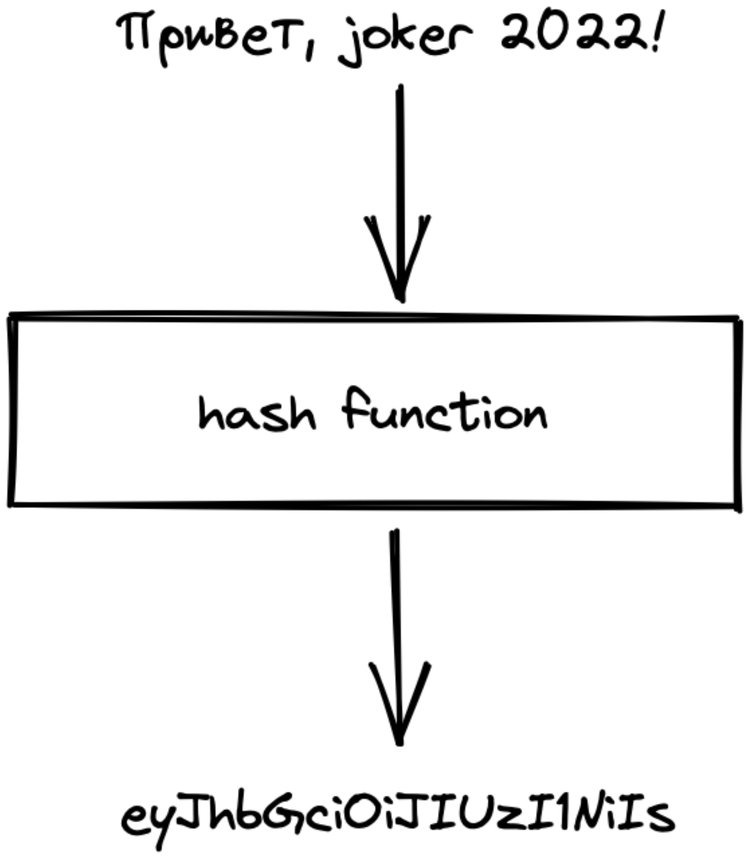
Хэширование

- Контрольные суммы данных (checksum)
- Ассоциативные массивы (хэш-таблицы)
- Отпечатки ключей (certificate/key fingerprint)
- Хранение паролей
- Аутентификация и контроль целостности
- Формирование ключей для шифрования

Свойства хэш функций

- Являются детерминированными
- Необратимость
- Возможные коллизии
- Любое незначительное изменение входных данных изменяет полностью значение хэш функции

Хэш функции

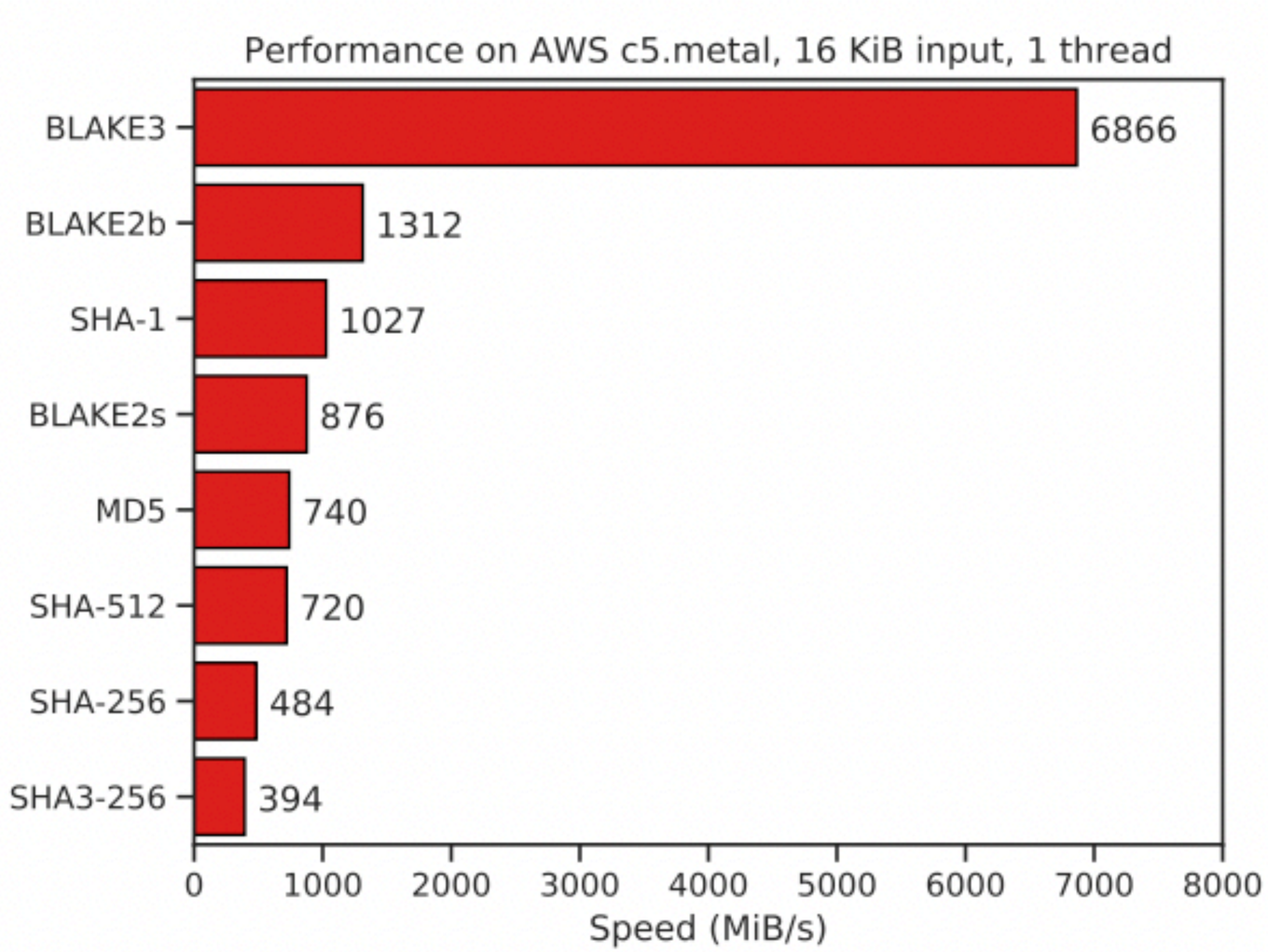


Хэш функции

- Актуальные
  - Argon, Blake2, Blake2b, Blake3, SHA-2* (SHA-256, SHA-384, SHA-512)
- Устаревшие
  - SHA-1, MD*, MD5

Blake3

- Гораздо быстрее чем MD5, SHA-* и Blake2b
- Более безопасный в отличии от MD5 и SHA1
- Большой запас значений (при тесте в 20 млн записей было 0 коллизий)



Поиск по зашифрованному полю

Id	User_id	User_token	Token_hash	Create_at
1	42	[111010...10]	123	2022-05-06
1	43	[111010...10]	852	2022-05-21 1
1	44	[111010...10]	325	2022-05-06
1	45	[111010...10]	142	2022-05-11 1
1	46	[111010...10]	10	2022-05-20

```
select * from user_data where token_hash = 10
```

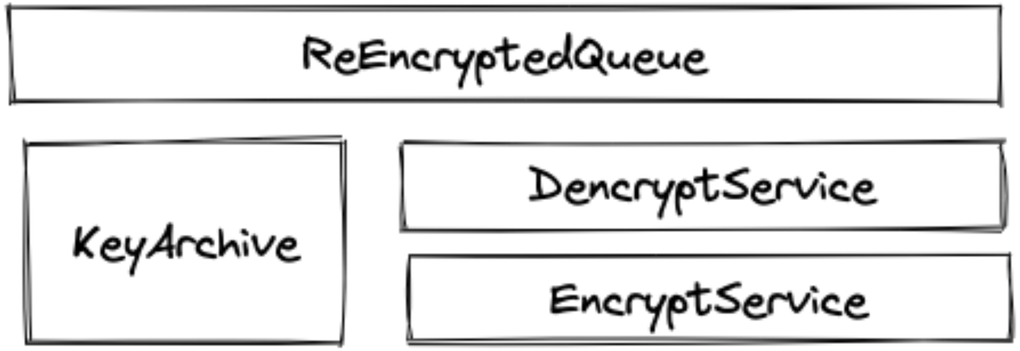
Ротация ключей шифрования

Ротация ключей шифрования - генерация нового ключа и перешифровка всех зашифрованных данных.
Фиксированный интервал ротации раз в квартал или раз в полгода

Ротация ключей/алгоритма шифрования

User_id	User_token	Token_hash	Secret_key_id	Algoritm_id
43	[111010...10]	852	first_key	AES-ECB
46	[111010...10]	10	first_key	AES-ECB
45	[111010...10]	142	first_key	AES-GCM
44	[111010...10]	325	first_key	AES-CBC
42	[111010...10]	123	first_key	ChaCha20-Poly1305

User_id	User_token	Token_hash	Secret_key_id	Algoritm_id
43	[111010...10]	852	second_key	ChaCha20-Poly1305
46	[111010...10]	10	second_key	ChaCha20-Poly1305
45	[111010...10]	142	second_key	ChaCha20-Poly1305
44	[111010...10]	325	second_key	ChaCha20-Poly1305
42	[111010...10]	123	second_key	ChaCha20-Poly1305



Сохранение meta данных

```
1   -----BEGIN RSA PRIVATE KEY-----
2   Proc-Type: 4,ENCRYPTED
3   DEK-Info: AES-256-CBC,F3F98287700F5641CE2C0EEE74D0DCE6
4
5   8Q0LxTA15Y6WXYp2+hTPPov+q3Ju2QdMbqjPMKSfPRGxVDVc2iUB+3/JvStYGc8h
6   Ozc+J1lKWi+c08n51vJaLlCITR8cRpaSMrOLPFQUugwCh53nLisOa3iH2kezXKpj
7   mu8BkSPMiSiX12j0b7agmsQlHYwR0iSvBAFvlK2j188eVsUSaBJUGFT0tQEZla0g
8   3CI4SKuMlvppFJjW3wK4mnGJqA2Yyv49vNT96tYvAANJUUWdFrN8t4+7Ugph2nCR
9   dH6XudvxCRcXeqnynhkFaQppzxtGUS3nnwptvFCmka5NvGT/vTiUaf41yu5hedIe
10  12qIaBdnUgD2yg516TOxbfqi3X00nATk5PopkMq+rQ895q7qRGlf6UrWLziHMGCl
11  YLB+2Qk0EAZ7W8g2l6q4EgXAWgsthPVL5u3xVohIWzpGBVZRTSJA7PobEYGNP/XF
12  u5CRWzDaZd9J36kvXbZTMBu9SO0glN7RRFkkeXZvsmJfxQWGcPYg+EkSLnRiQ25K
13  jV8/NTd5rnKyIwLzGDcZDvMX3UOCpW3CZD8NycGu96Ep8IHVmBIfe4PDV6x2FtDh
14  fH1lMitXU3/adYAgK+82CLmIZG2TEyMfLKrKJ3dhKGwqmZc7Cyd1LRDF0x2Kr2+I
15  V6V5TNoGlCQPGUz/I6egeJLUDsWitlzyJkOZIGHSZIknfB34FacUtuFgGfi87NcF
16  hKob084YyJttUiQG3xrMeyShv0fG85HkixGXIUOQho6Z6cNSEQ+WeXqIDmibzyyb
17  cFIvIdcZPotJYW17J382sews4bn5dwgfLZG9RJ2DBf/nsMABNSxMf+0TMdRaI6o/
18  kCGw42FbCko3ZZDh0W43kgZRqbLSB04Wg8lMExKdSatqyunqCJ+HdDOK27lq3EPw
```

Useful links

- В целом про хэш функции http://www.azillionmonkeys.com/qed/hash.html
- Blake3 https://github.com/BLAKE3-team/BLAKE3
- Argon2 https://datatracker.ietf.org/doc/html/draft-irtf-cfrg-argon2-04
- DB-Queue https://github.com/db-queue/db-queue

Итоги

- Рассмотрели варианты поиска уязвимостей
- Требования к алгоритмам шифрования
- Актуальные алгоритмы шифрования
- Поиск по зашифрованному значению
- Ротация ключей/алгоритма

Задавайте вопросы

Скобелев Григорий
telegram @Gskoba
twitter @gskobeleff
книжный клуб { между скобок }