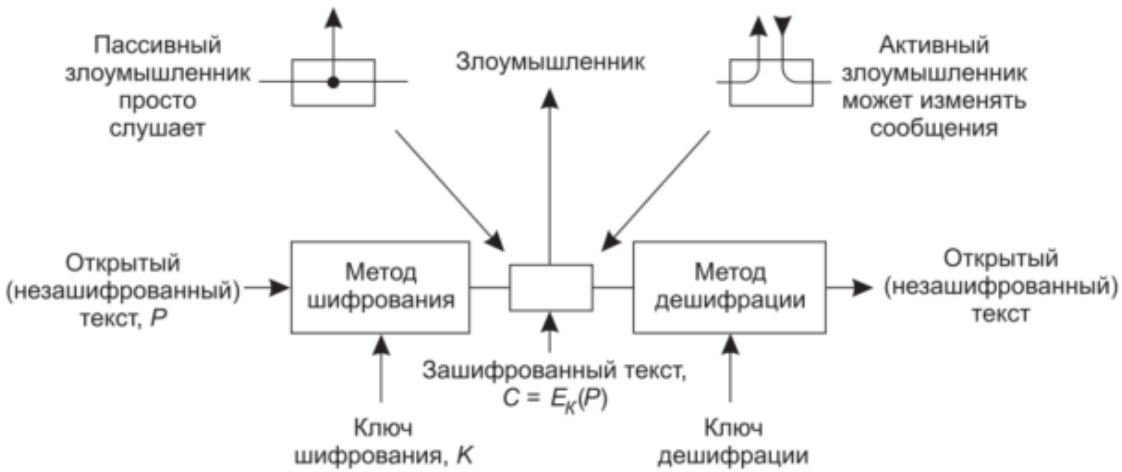
Архитектурные аспекты сетевой безопасности

Часть 2: шифры, подписи, авторизация

Юрий Литвинов
yurii.litvinov@gmail.com

20.05.2020г

Шифрование

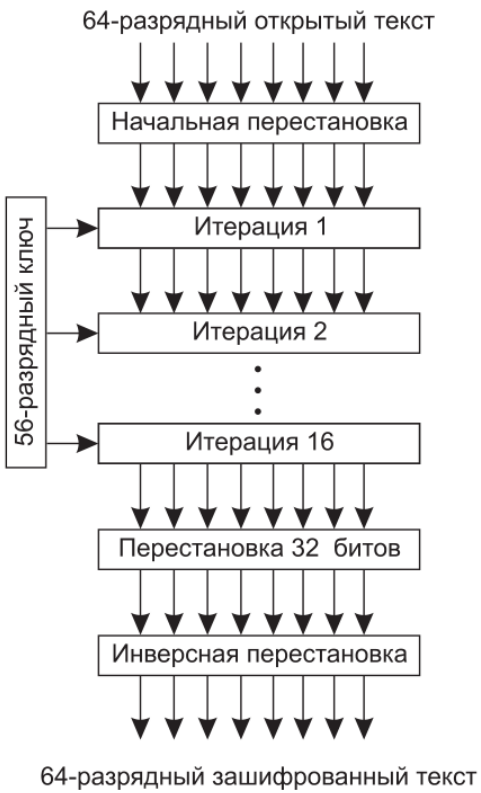


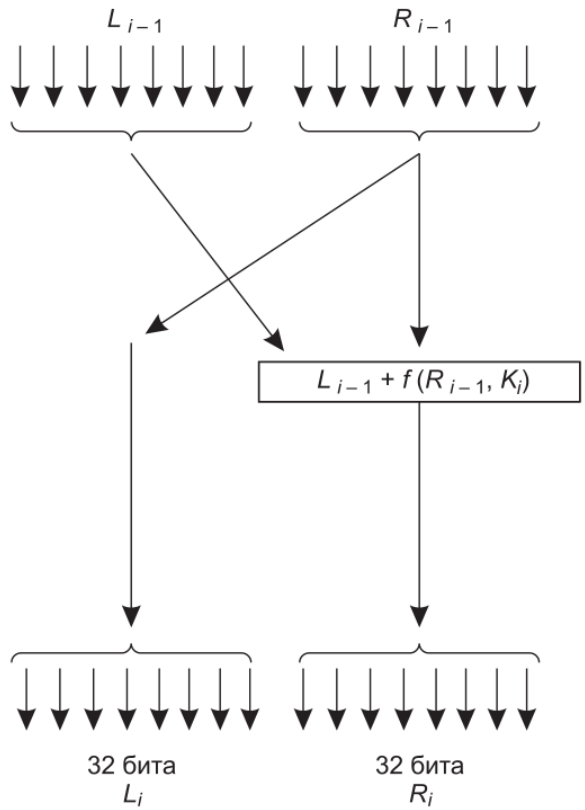
© Э. Таненбаум

- Алгоритм шифрования считается известным, секретен только ключ
- Усложнение алгоритма шифрования не всегда повышает криптостойкость

Шифрование с симметричным ключом

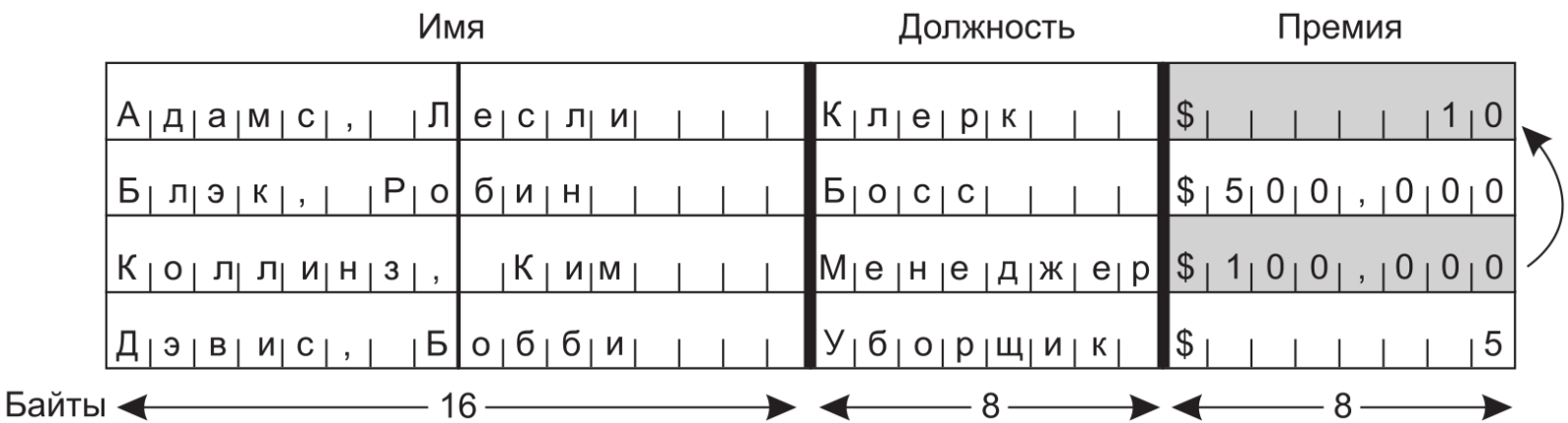
- Data Encryption Standard (DES, Triple DES)
- Advanced Encryption Standard (AES, он же Rijndael)





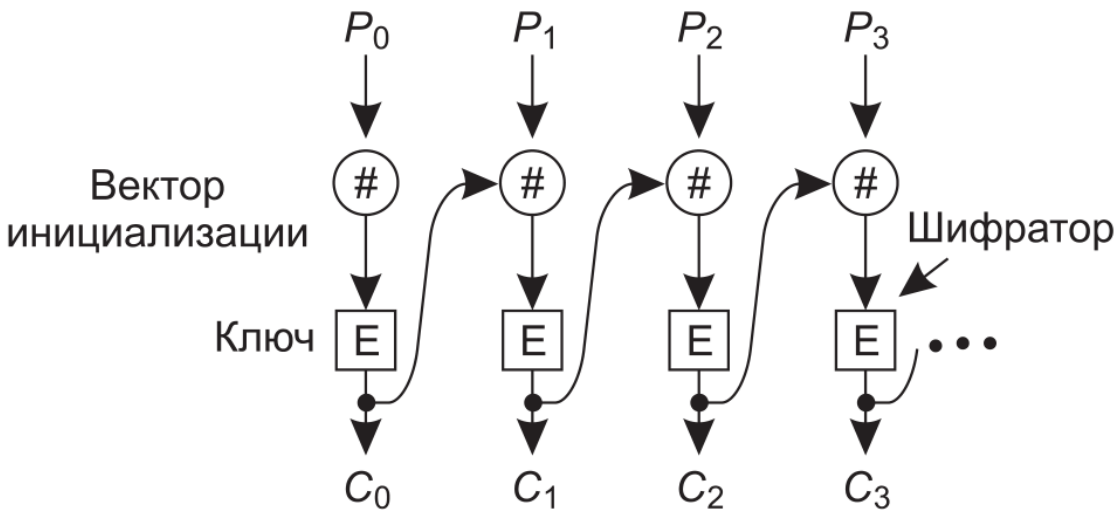
Режимы шифрования, ECB

- Electronic Code Book — один ключ применяется ко всем блокам
  - Быстро, надёжно, но не криптостойко



Режимы шифрования, CBC

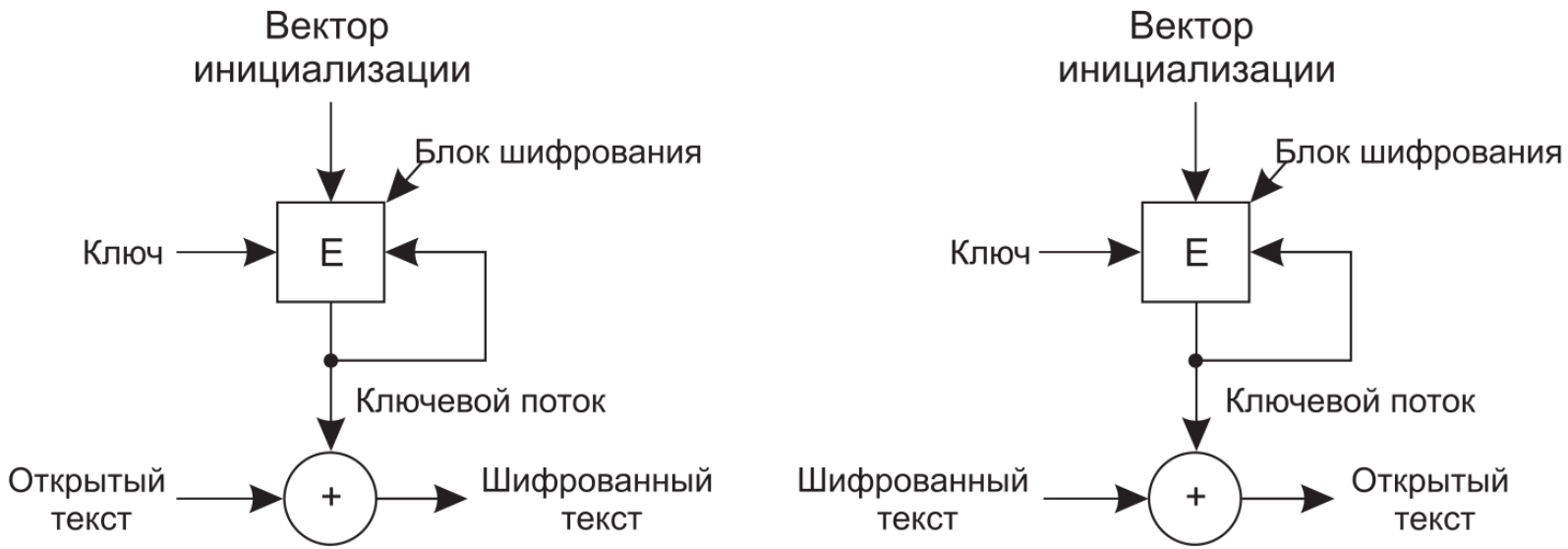
- Cipher Block Chaining — xor-им следующий блок с зашифрованным предыдущим перед шифровкой
  - Более криптостоек, не устойчив к ошибкам передачи
  - Initialization Vector (IV)



© Э. Таненбаум

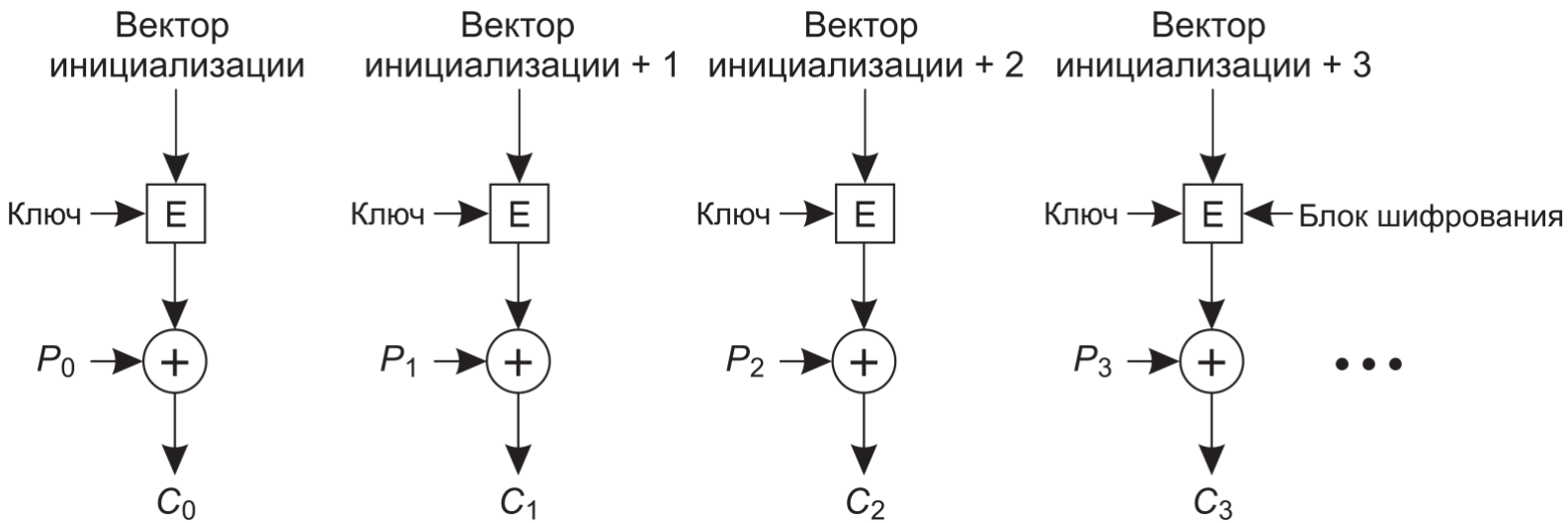
Режимы шифрования, SCM

- Stream Cipher Mode — шифруем IV ключом снова и снова, генерируя ключ бесконечной длины
  - И xor-им его с шифруемым текстом
  - Устойчив к ошибкам передачи, довольно быстр
  - Уязвим к Keystream Reuse Attack $((P_0 \oplus K_0) \oplus (Q_0 \oplus K_0))$



Режимы шифрования, Counter Mode

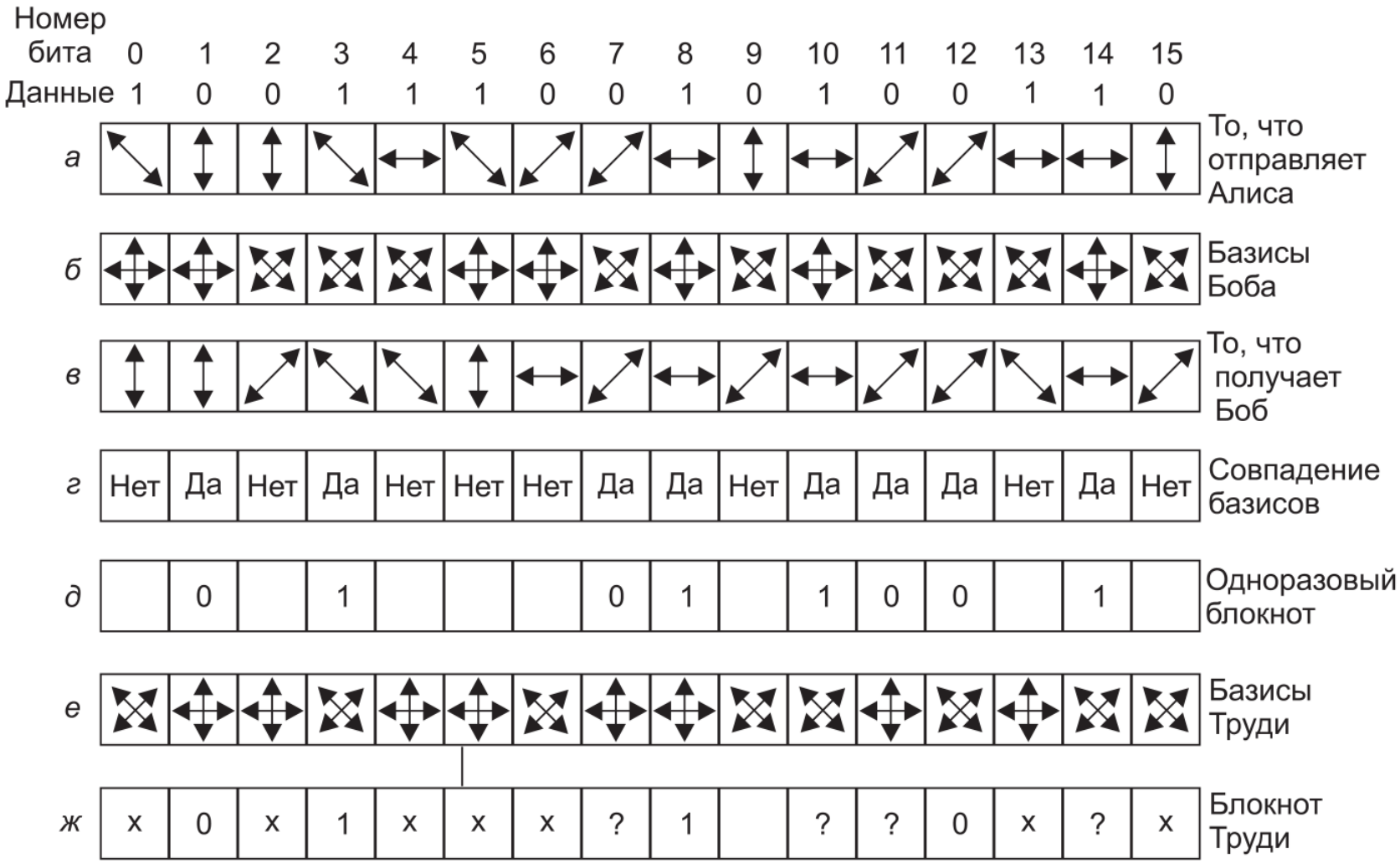
- Counter Mode — шифруем $IV + i$ для каждого i-го блока
  - И xor-им его с шифруемым текстом
  - Для произвольного доступа к зашифрованным блокам



© Э. Таненбаум

Квантовый шифр

Протокол Bennett-Brassard BB84



© Э. Таненбаум

Шифрование с открытым ключом

Или почему нельзя отдать ключи от Telegram

- Алгоритм делится на две части, D и E, так, что D(E(P)) = P
- D очень сложно получить по E
  - Например, найти простые сомножители огромного числа или дискретный логарифм по заданному модулю
- E не ломается атакой “произвольного открытого текста”
- D (ключ от D) держится в секрете, E выкладывается в открытый доступ
- Если Боб хочет послать Алисе сообщение, он берёт её открытый ключ E_A, шифрует им сообщение P и отправляет Алисе
- Алиса дешифрует сообщение, вычисляя $D_A(E_A(P))$
- У каждого пользователя своя пара ключей
- Алгоритмы: RSA, ElGamal, эллиптические шифры

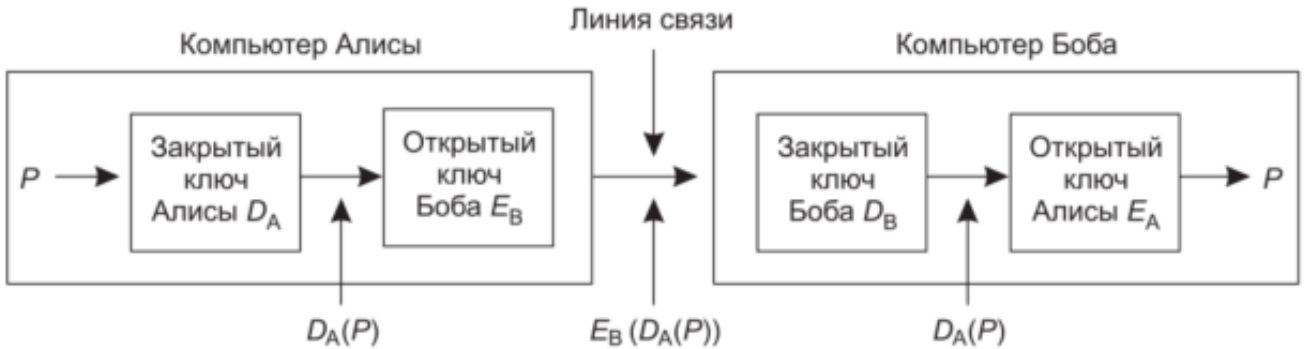
Конкретно Telegram

- Криптопротокол MTProto
  - AES + Диффи-Хеллман
- “Секретные чаты” отключены по умолчанию
  - Для удобства — позволяют только обмен “устройство-устройство”
- Без “секретных чатов” ничего секретного в Telegram нет
  - Коммуникации с сервером шифруются, но если сервер взломают, то всё
- “Атака присутствия”

Цифровые подписи, задачи

- Получатель может установить личность отправителя
- Отправитель не может отрицать, что он подписал сообщение
- Получатель не может сам подделать сообщение и сделать вид, что его послал отправитель

Цифровые подписи, реализация

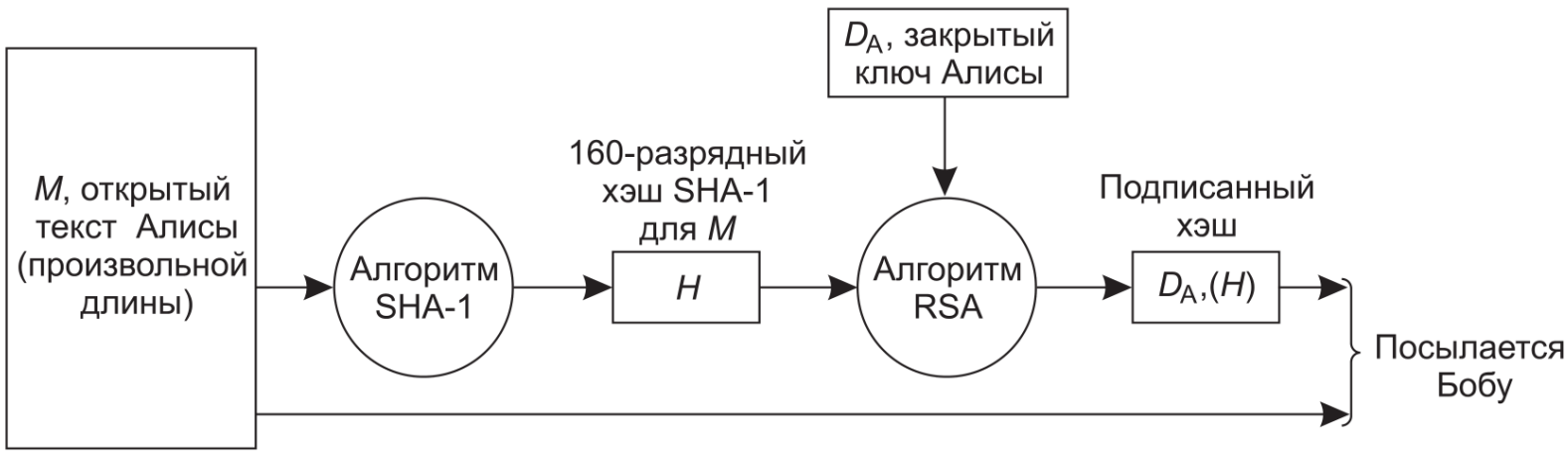


© Э. Таненбаум

- Надо, чтобы $D(E(P)) = P$ (это так для большинства криптосхем)
- Шифровать всё сообщение слишком медленно
- Message Digest-ы — хорошие хеши сообщений
  - MD5, SHA-1
- Подписывается только хеш, это почти так же криптостойко, но в сотни раз быстрее

SHA-1

- Считается блоками по 512 бит, возвращает 160-битный дайджест
- Изменение в одном бите входа даёт совершенно другой выход
- Если известен P, очень сложно найти такой P', что $MD(P') = MD(P)$



Атака дней рождения

Уважаемый господин декан,

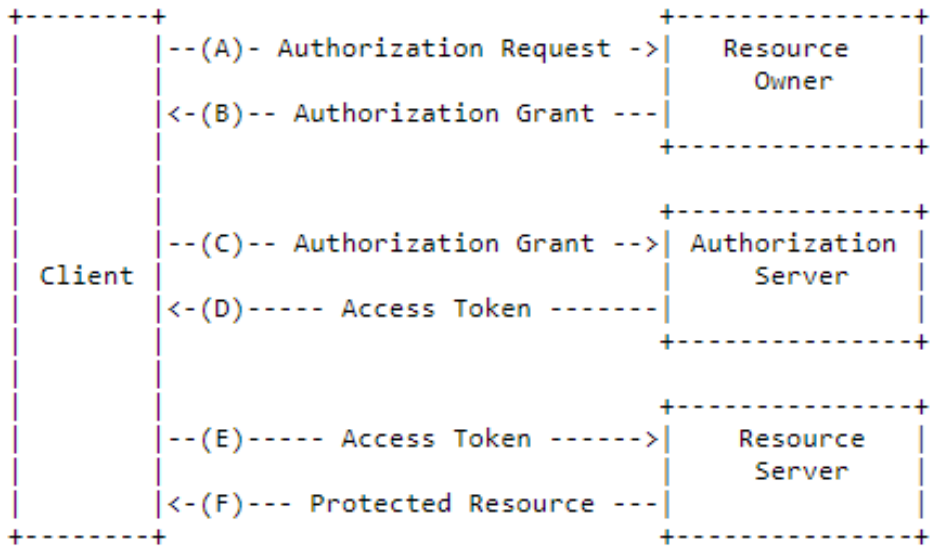
Это [письмо | обращение] отражает мое [искреннее | откровенное] [мнение | суждение] о проф. Томе Уилсоне, являющемся [кандидатом | претендентом] на профессорскую должность в [настоящее время | этом году]. Я [знакома | работала] с проф. Уилсоном в течение [почти | около] шести лет. Он является [слабым | недостаточно талантливым] [исследователем | ученым], почти не известным в той области науки, которой он занимается. В его работах практически не заметно понимания [ключевых | главных] [проблем | вопросов] современности.

[Более | Кроме] того, он также не является сколько-нибудь [уважаемым | ценимым] [преподавателем | педагогом]. Его студенты дают его [занятиям | лекциям] [самые низкие | негативные] оценки. Он самый непопулярный [преподаватель | учитель] нашей кафедры, [славящийся | печально известный] своей [привычкой | склонностью] [высмеивать | ставить в неудобное положение] студентов, осмелившихся задавать вопросы на его [лекциях | занятиях].

Авторизация, OAuth 2

- Позволяет разрешить пользование ресурсом, не раскрывая хозяину ресурса логин и пароль пользователя
  - Логин по аккаунту в Google или аккаунту в VK
- Роли:
  - Client — приложение, пытающееся получить доступ
  - Resource Server — сервер, хранящий защищённую информацию. К нему пытается получить доступ клиент
  - Resource Owner — пользователь, владеющий защищённой информацией
  - Authorization Server — сервер, выдающий клиенту токен на доступ к ресурсному серверу

Протокол



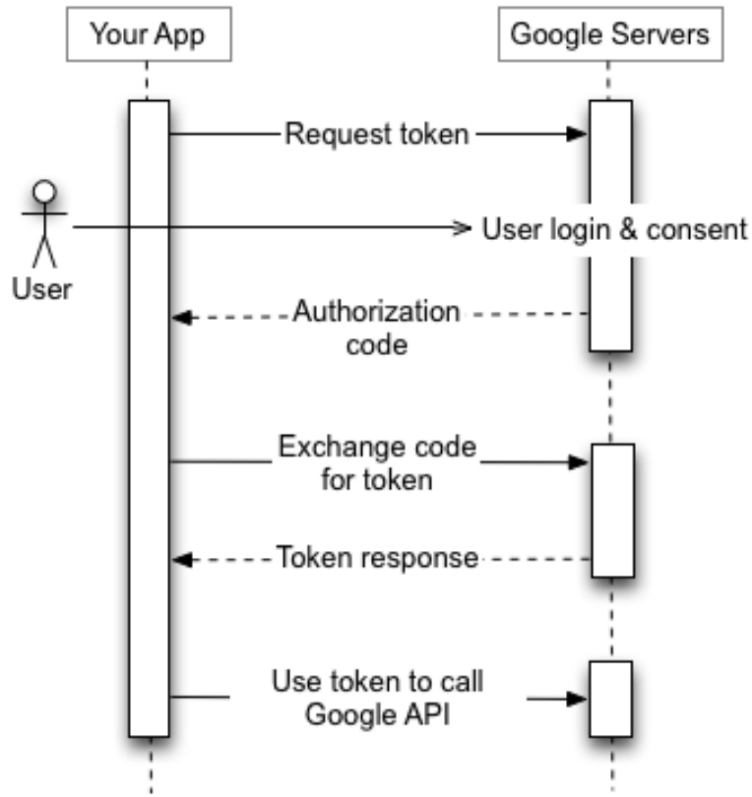
© RFC 6749

Детали

- Access Token — выдаётся авторизационным сервером и посылается с каждым запросом, ограниченное время жизни
- Refresh Token — выдаётся авторизационным сервером, используется для получения нового Access Token
- Scope — к какой части ресурса даёт доступ Access Token
- Authorization Grant — Authorization Code, Implicit, Resource Owner Password Credentials, Client Credentials

Пример: Google OAuth 2.0

- Google Developer Console, Client ID и Client Secret
- Scope
- Consent Screen



© https://developers.google.com

Как это всё отлаживать

И ломать

- Fiddler — кроссплатфоренный отладочный прокси
  - Перехват HTTP-трафика
  - Man-In-The-Middle-атака с самоподписанными сертификатами
    - Расшифровка HTTPS-трафика на лету
  - Возможность модифицировать HTTP-пакеты, повторять пакеты и т.д.
- Wireshark — когда Fiddler-а мало
  - Перехват пакетов на низком уровне
  - Умеет даже ставить себя как драйвер USB и читать USB-пакеты