УГАТУ

Лекция №9

ИНФРАСТРУКТУРЫ И ПРОТОКОЛЫ

Дисциплина: Криптографические методы защиты информации
Преподаватель: Миронов Константин Валерьевич
Поток: БПС-3, ИКТ-5
Учебный год: 2020/21

УГАТУ

Уфимский государственный авиационный технический университет

Содержание лекции

- **Общие сведения о криптографических протоколах**
  - **Особенности**
  - **Пример**
- Управление ключами средствами симметричной криптографии
- Инфраструктуры открытых ключей

УГАТУ

Уфимский государственный авиационный технический университет

Криптографические протоколы

- **Криптографический протокол** - набор правил, описывающих взаимодействие узлов некоторой сети с использованием КЗИ
  - под *сетью* может подразумеваться как реальная компьютерная сеть, так и абстрактная среда взаимодействия субъектов
- Правила включают:
  - Порядок обмена данными между узлами
  - Вычисления, выполняемые на этих узлах перед отправкой / после получения данных
- Примеры:
  - Протоколы установления защищенного соединения (http + TLS = https)
  - Протоколы электронных финансовых операций (оплата товаров банковской картой, онлайн-банкинг и т.д.)
  - Протоколы систем распределенного реестра (Bitcoin, Etherium, Мастерчейн и др.)
  - и т.д.

УГАТУ

Уфимский государственный авиационный технический университет

Криптографические протоколы

Особенности

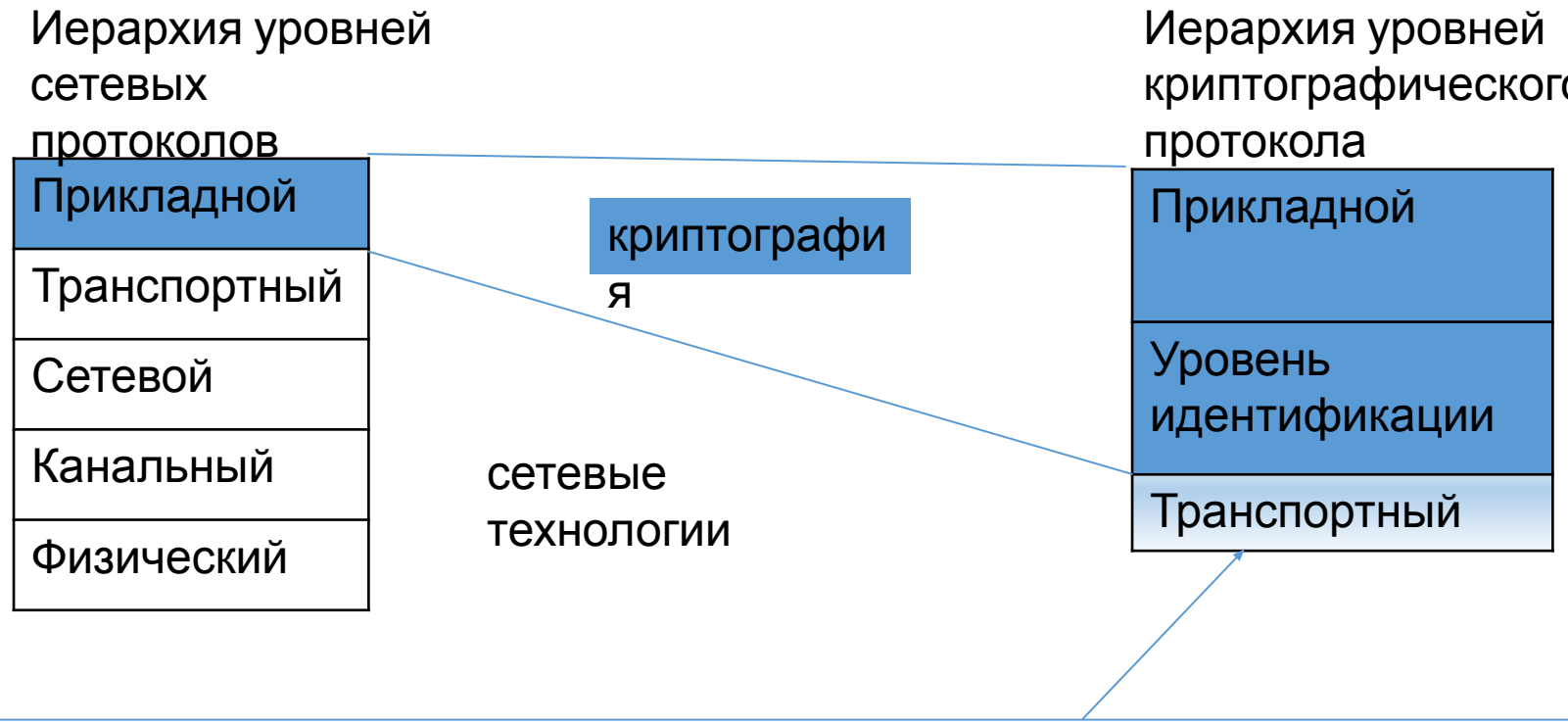
- Цель – минимизация уровня доверия, необходимого для успешного взаимодействия узлов
- **Параноидальная модель** – каждый из участников взаимодействия предполагает, что остальные могут объединиться против него
- При разработке протокола может встать выбор, кого сильнее защищать
  - Обычно выбор в пользу того, кто больше дорожит репутацией честного игрока или несет большие риски
  - Пример: при отправке товара по почте деньги надо заплатить вперед, потому что онлайн-магазин больше дорожит репутацией чем покупатель, который может оказаться вором

УГАТУ

Уфимский государственный авиационный технический университет

Криптографические протоколы

Особенности



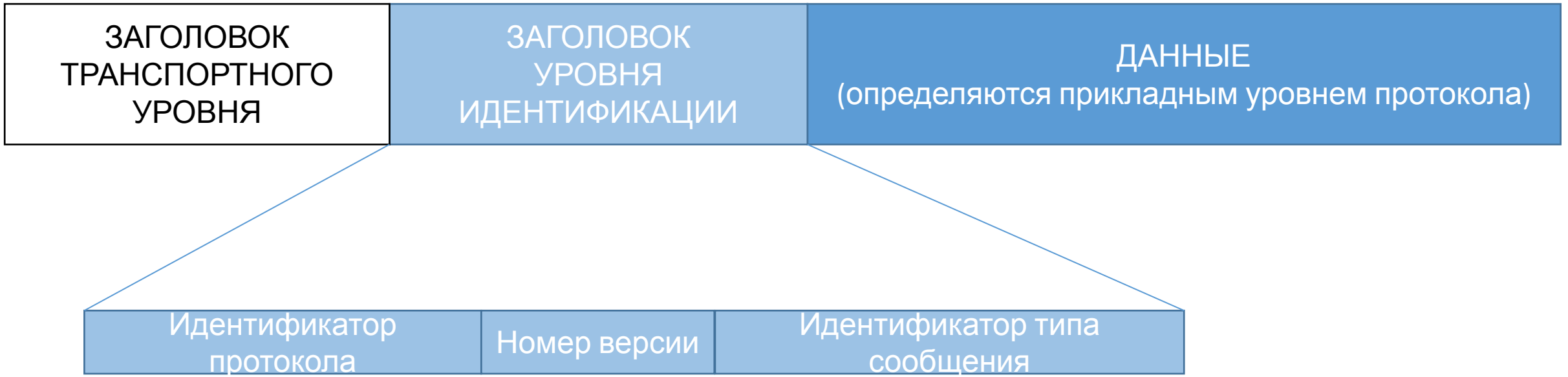
Под транспортным уровнем понимается технологии обмена данными между узлами.
Например, сетевые протоколы UDP или TCP. При этом часть функций КЗИ может реализовываться на транспортном уровне, например, расчет MAC

УГАТУ

Уфимский государственный авиационный технический университет

Криптографические протоколы

Структура передаваемого сообщения



Данный уровень не защищает от злоумышленника, но позволяет избежать случайных ошибок

УГАТУ

Уфимский государственный авиационный технический университет

Криптографические протоколы

Особенности

- Кодирование информации перед отправкой и декодирование после получения
  - Преобразование чисел, текстовых данных и т.п. в набор байт
  - Обратное преобразование по возможности должно быть однозначным и не зависеть от контекста
- Протоколы выполняются в рамках событийно-ориентированной программной архитектуры [Event Driven Architecture, EDA]
  - Информация о состоянии протокола содержит список допустимых событий, которые могут происходить в рамках протокола при данном состоянии

Криптографические протоколы

Особенности

- Время ожидания:
  - При стабильном соединении пакет данных в Интернете обычно либо доходит за секунду, либо не доходит вообще
  - Соединение может быть нестабильным
  - Стандартный подход: ожидать отклика 3-5 секунд, затем послать пакет заново и так до 3-5 раз
- При обработке ошибок важно не давать потенциальному злоумышленнику лишней информации
  - Идеальный вариант – прерывать выполнение протокола без объяснений
  - В реальности система должна быть дружественной к пользователю, поэтому она выдает скупые сообщения об ошибках
  - Время выдачи сообщения об ошибке может быть использовано при тайминг-атаке, поэтому оно должно быть независимым от данных

Криптографические протоколы

Пример

Протокол Фергюссона-Шнайера для установления защищенного соединения с помощью алгоритма Диффи-Хеллмана

Соглашение об аутентификации:

- Каждое следующее сообщение содержит аутентификатор (имитовставку или ЭП) совокупности всех предыдущих переданных по каналу данных (включая заголовок уровня идентификации)
- В дальнейшем такой аутентификатор обозначается как $AUTH_n$

УГАТУ

Уфимский государственный авиационный технический университет

Криптографические протоколы

Пример

АЛИСА

s_a

Сгенерировать $nonce$

s_a, $nonce$ →

БОБ

s_b

Проверить, что $s_a \leq 2s_b$

Сгенерировать параметры алгоритма (p, q, α) такие, что $\log_2 p \approx \max(s_a, s_b)$

Сгенерировать x_b

Вычислить $y_b = f_{DH}(x_b, \alpha, p)$

Вычислить $AUTH_1$

← $p, q, \alpha, y_b, AUTH_1$

Проверить корректность параметров

Сгенерировать x_a

Вычислить $y_a = f_{DH}(x_a, \alpha, p)$

Вычислить $AUTH_2$

y_a, $AUTH_2$ →

Вычислить $K = hash(f_{DH}(x_a, y_b, p))$

Проверить корректность параметров

Вычислить $K = hash(f_{DH}(x_b, y_a, p))$

УГАТУ

Уфимский государственный авиационный технический университет

Криптографические протоколы

Пример

АЛИСА

s_a

БОБ

s_b

s_a и s_b - минимально допустимые длины модуля по версии Алисы и Боба

Сгенерировать $nonce$

s_a, $nonce$ →

Проверить, что $s_a \leq 2s_b$

Сгенерировать параметры алгоритма (p, q, α) такие, что $\log_2 p \approx \max(s_a, s_b)$

Сгенерировать x_b

Вычислить $y_b = f_{DH}(x_b, \alpha, p)$

Вычислить $AUTH_1$

Перед началом работы Алиса и Боб согласовывают требования к безопасной длине параметров

Нонс нужен, чтобы убедиться, что Боб отвечает именно на это сообщение Алисы, а не на какое-то иное

← $p, q, \alpha, y_b, AUTH_1$

Проверить корректность параметров

Сгенерировать x_a

Вычислить $y_a = f_{DH}(x_a, \alpha, p)$

Вычислить $AUTH_2$

y_a, $AUTH_2$ →

Вычислить $K = hash(f_{DH}(x_a, y_b, p))$

Проверить корректность параметров

Вычислить $K = hash(f_{DH}(x_b, y_a, p))$

УГАТУ

Уфимский государственный авиационный технический университет

Криптографические протоколы

Пример

АЛИСА

s_a

БОБ

s_b

Сгенерировать $nonce$

s_a, $nonce$ →

Проверить, что $s_a \leq 2s_b$

Защита от попытки Алисы перегрузить Боба вычислениями
Значение привязано к s_b, чтобы при апгрейде системы достаточно было изменить только один параметр

Сгенерировать параметры алгоритма (p, q, α) такие, что $\log_2 p \approx \max(s_a, s_b)$

Сгенерировать x_b

Вычислить $y_b = f_{DH}(x_b, \alpha, p)$

Вычислить $AUTH_1$

← $p, q, \alpha, y_b, AUTH_1$

Проверить корректность параметров

Сгенерировать x_a

Вычислить $y_a = f_{DH}(x_a, \alpha, p)$

Вычислить $AUTH_2$

y_a, $AUTH_2$ →

Вычислить $K = hash(f_{DH}(x_a, y_b, p))$

Проверить корректность параметров

Вычислить $K = hash(f_{DH}(x_b, y_a, p))$

Криптографические протоколы

Пример

АЛИСА

s_a

Сгенерировать $nonce$

заголовок_0, s_a, $nonce$ →

БОБ

s_b

Проверить, что $s_a \leq 2s_b$

Сгенерировать параметры алгоритма (p, q, α) такие, что $\log_2 p \approx \max(s_a, s_b)$

Сгенерировать x_b

Вычислить $y_b = f_{DH}(x_b, \alpha, p)$

Вычислить $AUTH_1$

$AUTH_1 = auth(\text{заголовок\_0}||s_a||nonce||\text{заголовок\_1}||\ p||q||\alpha||y_b)$

← заголовок_1, p, q, α, y_b, $AUTH_1$

Проверить корректность параметров

Сгенерировать x_a

Вычислить $y_a = f_{DH}(x_a, \alpha, p)$

Вычислить $AUTH_2$

$AUTH_2 = auth(\text{заголовок\_0}||s_a||nonce||\text{заголовок\_1}||\ p||q||\alpha||y_b||\text{заголовок\_2}||\ y_a)$

заголовок_2, y_a, $AUTH_2$ →

Вычислить $K = hash(f_{DH}(x_a, y_b, p))$

Проверить корректность параметров

Вычислить $K = hash(f_{DH}(x_b, y_a, p))$

Криптографические протоколы

Пример

АЛИСА		БОБ
s_a		s_b
Сгенерировать $nonce$	$s_a,\ nonce$ →	
		Проверить, что $s_a \leq 2s_b$
		Сгенерировать параметры алгоритма (p, q, α) такие, что $\log_2 p \approx \max(s_a, s_b)$
		Сгенерировать x_b
		Вычислить $y_b = f_{DH}(x_b, \alpha, p)$
		Вычислить $AUTH_1$
Проверить корректность $AUTH_1$ Проверить, что $\log_2 p \in [s_a - 1, 2s_a]$ Проверить параметры алгоритма Диффи-Хеллмана	← $p, q, \alpha, y_b, AUTH_1$	
Проверить корректность параметров		
Сгенерировать x_a		
Вычислить $y_a = f_{DH}(x_a, \alpha, p)$		
Вычислить $AUTH_2$		
	$y_a, AUTH_2$ →	
		Проверить корректность $AUTH_2$ Проверить, что $y_a \neq 1$ и $y_a^q = 1$
		Проверить корректность параметров
Вычислить $K = hash(f_{DH}(x_a, y_b, p))$		Вычислить $K = hash(f_{DH}(x_b, y_a, p))$

Криптографические протоколы

Пример

Заключительные положения:

- ОБЩИЙ ВЫВОД: Протокол и алгоритм – разные вещи; в протоколе к алгоритму добавляется ряд нюансов, позволяющих повысить безопасность, эффективность, удобство и т.д.
- ЧАСТНЫЙ ВЫВОД: Использование алгоритма Диффи-Хеллмана само по себе не решает проблему распределения ключей (требуется ключ для аутентификации); необходим механизм, позволяющий убедиться, что данный ключ принадлежит именно Алисе, а не кому-либо иному

УГАТУ

Уфимский государственный авиационный технический университет

Содержание лекции

- Общие сведения о криптографических протоколах
- **Управление ключами средствами симметричной криптографии**
  - **Иерархии ключей**
  - **Серверы ключей**
- Инфраструктуры открытых ключей

Иерархии ключей

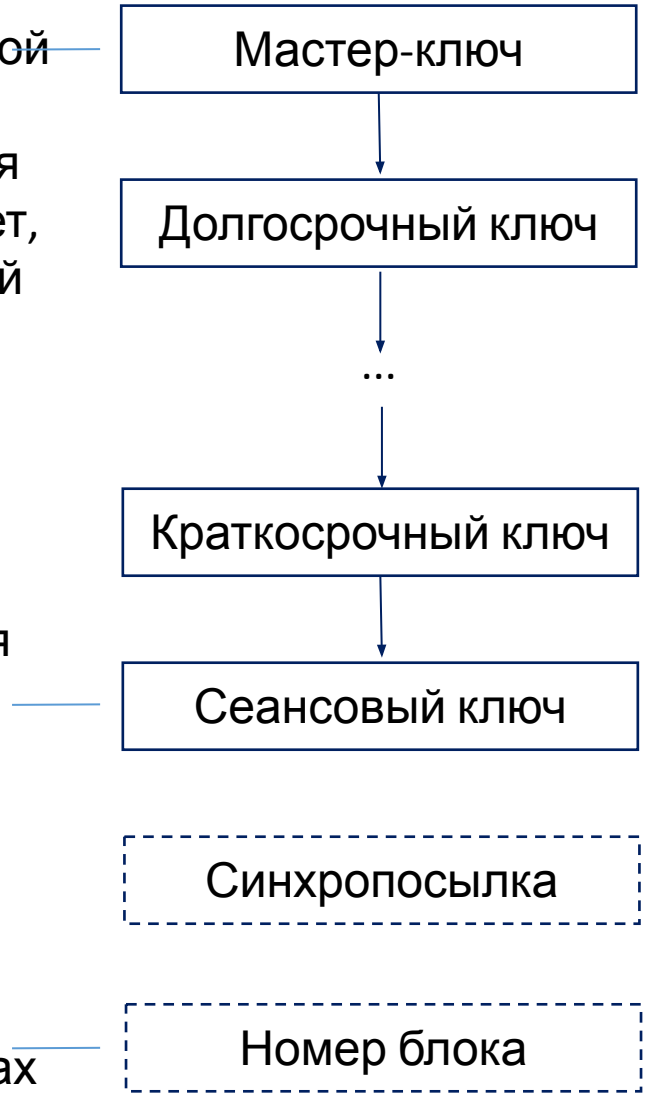
Условия применения:

- Вариант 1: сеть относительно небольшая, каждый абонент может хранить ключи для общения со всеми остальными абонентами
- Вариант 2: сеть может быть больше (до 10 000 узлов), при этом жестко заданы потоки данных
  - Пример: сеть датчиков, измеряющих некоторые величины. Датчики связываются с сервером или точкой доступа, но не друг с другом
  - У датчиков зачастую ограниченные ресурсы, не позволяющие реализовывать распределение средствами асимметричной криптографии

УГАТУ

Уфимский государственный авиационный технический университет

Иерархии ключей



Задается вручную при начальной настройке конфигурации сети; Используется для согласования долгосрочного ключа; Действует, пока система работает в данной конфигурации (если не был скомпрометирован)

Пересылается зашифрованным на мастер-ключе; Используется для установки ключа следующего уровня; Действует в течение нескольких месяцев или лет

Пересылается зашифрованным на ключе предыдущего уровня; Используется для установки сеансового ключа; Действует в течение нескольких дней или недель

Пересылается зашифрованным на краткосрочном ключе; Используется для шифрования данных в рамках одного сеанса связи; Действует в течение сеанса (несколько минут)

Пересылается в открытом виде; Используется при шифровании отдельного сообщения в рамках сеанса связи

Используется при шифровании отдельного блока данных в рамках

УГАТУ
Уфимский государственный авиационный технический университет

Иерархии ключей

- Причины использования иерархий:
  - Защита от атак, требующих знания злоумышленником большого массива шифротекстов
  - Мастер-ключ может быть слабее ключей более нижних уровней (например, пароль)
  - Компрометация отдельного ключа не приводит к компрометации всей системы
- Опции
  - Сеансовый ключ может не пересылаться зашифрованным на краткосрочном ключе виде, а независимо вычисляться на его основе Алисой и Бобом с помощью ГПСП
  - Ключ для общения каждого устройства с сервером может генерироваться на основе некоторого общего ключа и ID конкретного устройства; При этом в памяти сервера хранится только общий ключ, в памяти каждого устройства – присланный сервером ключ для общения сервера с этим устройством

УГАТУ
Уфимский государственный авиационный технический университет

Серверы ключей

Сервер ключей, центр распределения ключей [Key Distribution Center, KDC], **сервер выдачи мандатов** [Ticket Granting Server, TGS]

- Протокол Нидхэма-Шредера, 1978
- Протокол Kerberos, 1983
- Условия:
  - Сеть достаточно мала, чтобы в ней был сервер, хранящий в памяти ключи для общения с каждым из узлов (~до 10 000 узлов)
  - Сеть слишком велика, чтобы каждый из рядовых узлов хранил в памяти ключи для общения со всеми остальными
  - Узлы сети полностью доверяют серверу

УГАТУ

Уфимский государственный авиационный технический университет

Серверы ключей

БОБ

$К_{b,s}$

АЛИСА

$К_{a,s}$

$ШТ_0 = Ш('хочу ключ для связи с Бобом', К_{a,s})$

KDC

$К_{b,s}$, $К_{a,s}$

$ШТ_0$ →

$К_{a,b} = RAND$

$ШТ_1 = Ш(К_{a,b}, К_{b,s})$

$ШТ_2 = Ш(К_{a,b}||ШТ_1, К_{a,s})$

$ШТ_2$ ←

$К_{a,b}||ШТ_1 = РасШ(ШТ_2, К_{a,s})$

$DELETE\ К_{a,b}$

'хочу открыть защищенный кана

$ШТ_1$ ←

$К_{a,b} = РасШ(ШТ_1, К_{b,s})$

УГАТУ

Уфимский государственный авиационный технический университет

Серверы ключей

БОБ

$К_{b,s}$

АЛИСА

$К_{a,s}$

$ШТ_0 = Ш('хочу ключ для связи с Бобом', К_{a,s})$

$ШТ_0$

KDC

$К_{b,s}$, $К_{a,s}$

$К_{a,b} = RAND$

$ШТ_1 = Ш(К_{a,b}, К_{b,s})$

$ШТ_2 = Ш(К_{a,b}||ШТ_1, К_{a,s})$

$ШТ_2$

$К_{a,b}||ШТ_1 = РасШ(ШТ_2, К_{a,s})$

$DELETE\ К_{a,b}$

Мандат или **билет** [Ticket]

'хочу открыть защищенный кана

$ШТ_1$

$К_{a,b} = РасШ(ШТ_1, К_{b,s})$

УГАТУ

Уфимский государственный авиационный технический университет

Содержание лекции

- Общие сведения о криптографических протоколах
- Управление ключами средствами симметричной криптографии
- **Инфраструктуры открытых ключей**
  - **Общие сведения**
  - **Проблема отозванных сертификатов**
  - **Самоподписанные сертификаты**
  - **Установление защищенного соединения**

УГАТУ
Уфимский государственный авиационный технический университет

Инфраструктуры открытых ключей

[Public Key Infrastructure, PKI]

- Инфраструктура позволяет определять, кому принадлежит тот или иной открытый ключ
- **Сертификационное агентство, центр сертификации, удостоверяющий центр** [Certification Authority, CA] – субъект инфраструктуры (организация, сервер и т.п.), который публикует утверждения вида «открытый ключ X принадлежит пользователю A» и эти утверждения расцениваются остальными узлами как истина

Такое утверждение называется:

- **Сертификат открытого ключа** – файл подписанный CA и содержащий следующую информацию:
  - Персональные данные субъекта
  - Открытый ключ субъекта
  - Срок действия сертификата
  - и т.д. (определяется приложением)

Инфраструктуры открытых ключей

- Абоненту необязательно доверять ключ серверу, чтобы тот мог подтвердить его принадлежность
- Алисе не требуется связываться с УЦ каждый раз, когда она хочет установить связь с Бобом
- УЦ объединяются в иерархию
  - Пример: федеральный УЦ выдает сертификат региональным, региональные ведомственным и т. д.
  - Величина сертификата увеличивается и структура усложняется:

$$CERT(A) = [CERT(0 \to 1), CERT(1 \to 2), \dots, CERT(n \to A)]$$

Сертификат, выданный корневым центром центру следующего уровня

Сертификат, выданный центром нижнего уровня Алисе

УГАТУ

Уфимский государственный авиационный технический университет

Инфраструктуры открытых ключей

Практические особенности

- На основе PKI можно строить систему мандатного контроля доступа
  - Пользователь получает сразу несколько сертификатов (мандатов), каждый из которых дает ему право выполнять те или иные действия в системе
  - Мандат обязательно содержит информацию о полномочиях владельца
- Для чего устанавливают время действия ключа:
  - Время действия полномочий пользователя конечно
  - Чем меньше действует ключ тем меньше вероятность, что он попадет к злоумышленнику
  - Если ключ взломан незаметно для системы, то связанные с этим потери прекратятся после обновления ключа

УГАТУ
Уфимский государственный авиационный технический университет

Инфраструктуры открытых ключей

Проблема отзыва сертификата

УЦ публикует и обновляет **список отозванных сертификатов** [Certificate Revocation List, CRL]

- Список сертификатов, утративших силу ранее окончания своего срока действия
- **Пример 1:** использовавшийся закрытый ключ оказался слит в открытый доступ
- **Пример 2:** сотрудник получил сертификат для доступа к корпоративной VPN со сроком действия год, а через три месяца уволился

Альтернатива 1: **протокол состояния сертификата** [Online Certificate Status Protocol, OCSP] – CRL не рассылается, а хранится в УЦ; при получении сертификата Алисы Боб связывается с УЦ и узнает, не отозван ли этот сертификат

- Меньше объем трафика, но требуется постоянная связь

Альтернатива 2: использование краткосрочных сертификатов

- Срок действия: несколько минут или часов
- Предполагается, что за это время с ними ничего не случится
- Подходит далеко не для всех приложений

Инфраструктуры открытых ключей

Проблема отзыва сертификата

Нужно ли Алисе и Бобу поддерживать постоянную связь с УЦ?

- Вариант 1 *«идеальная система»* (отзыва сертификатов не происходит) – постоянная связь с УЦ **не требуется**, достаточно один раз скачать ОК УЦ, а потом проверять подписанные им сертификаты, присланные собеседником
- Вариант 2 *«идеальный злоумышленник»* - при проверке сертификата Алисы Боб **должен** связываться с УЦ, чтобы проверить отсутствие ключа Алисы в CRL онлайн (т.е. де факто переход на OCSP)
- Частый практический вариант - Боб хранит и периодически обновляет офлайн-копию CRL, чтобы иметь возможность проверять сертификаты без связи с УЦ
  - Частота обновления и действия, допустимые без связи с УЦ, определяются рисками, возникающими при работе конкретного приложения

Инфраструктуры открытых ключей

Самоподписанные сертификаты

- Сертификат, у которого подписывающий и подписываемый ключи совпадают (т.е. «сам себе выдал»)
- Используется не для удостоверения пользователя, а для представления сведений о ключе в формате, принятом в рамках данной PKI
- Частные случаи использования:
  - отладка системы
  - хранение ключа корневого УЦ
  - подключение рядового пользователя к общедоступным сервисам
- Перед истечением срока действия очередного сертификата сначала старым ключом подписывают новый, затем выпускают самоподписанный сертификат нового ключа
  - либо подписывают новый сертификат и старым, и новым ключами

УГАТУ
Уфимский государственный авиационный технический университет

Установление защищенного соединения

- Примеры: SSL, TLS, https = http + (SSL или TLS)

№ этапа	Что происходит	Какой примитив нужен	Какие алгоритмы применяются
1	Алиса и Боб обмениваются сертификатами и проверяют их подлинность	Электронная подпись	RSA, (EC)DSA, ГОСТ Р 34.10
2	Алиса и Боб договариваются об общем секретном ключе	Алгоритм распределения ключей	RSA, (EC)DH
3	Алиса и Боб обмениваются данными, зашифрованными на секретном ключе	Симметричный шифр	AES, ГОСТ Р 34.12

(EC)DH – алгоритм Диффи-Хеллмана (на эллиптических кривых)

УГАТУ
Уфимский государственный авиационный технический университет

Установление защищенного соединения

{RSA & RSA} vs {DSA & Диффи-Хеллман}

- При первом варианте алгоритм один и тот же, при втором – две немного отличающиеся модификации схемы Эль-Гамаля
- Сравнение подписей:
  - В DSA ЗК выбирается случайным образом и в целом генерация ключей проще
  - Постановка подписи одинаково сложная в обоих алгоритмах
  - В DSA проверка подписи в 10-40 раз медленнее
- Сравнение алгоритмов распределения ключей
  - При малой длине модуля эффективнее RSA, при большой – Диффи-Хеллман
  - При 256-битной безопасности граничная величина модуля ~3000 бит
- DSA и алгоритм Диффи-Хеллмана можно перевести на эллиптические кривые
- Примерно до середины 2000-х предпочитали первый вариант, затем перешли на второй