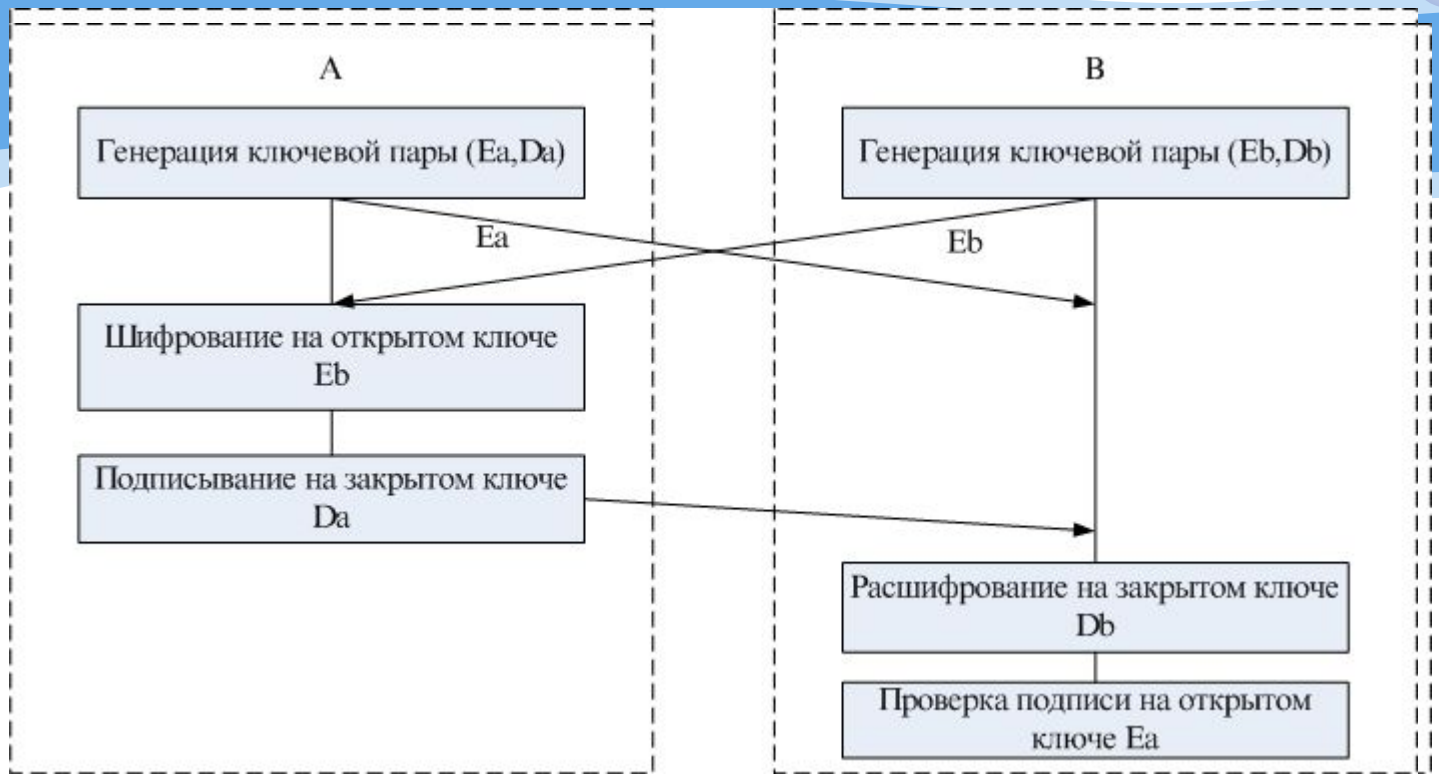
# Инфраструктура открытых КЛЮЧЕЙ

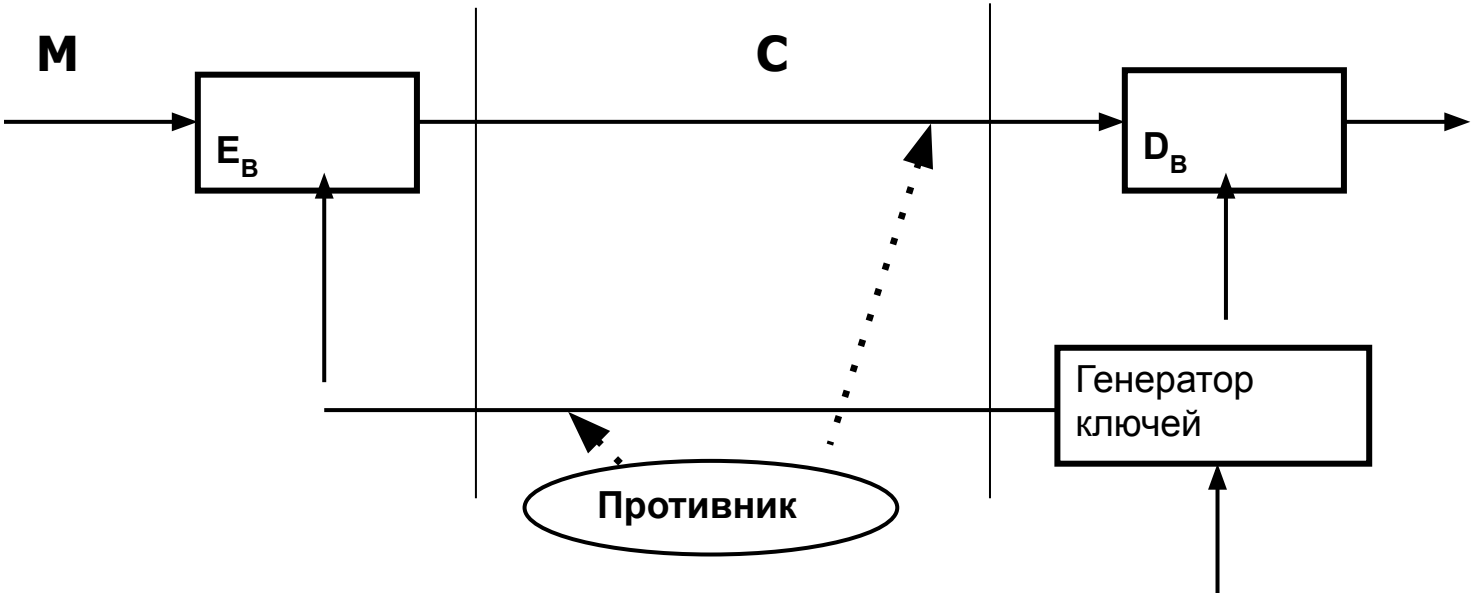
Лекция 9

# Асимметри


A
B
Генерация ключевой пары (Ea,Da)
Генерация ключевой пары (Eb,Db)
Ea
Eb
Шифрование на открытом ключе Eb
Подписывание на закрытом ключе Da
Расшифрование на закрытом ключе Db
Проверка подписи на открытом ключе Ea

# Обобщенная схема асимметричной криптосистемы с открытым ключом

**Отправитель А** **Незащищенный канал** **Получатель В**

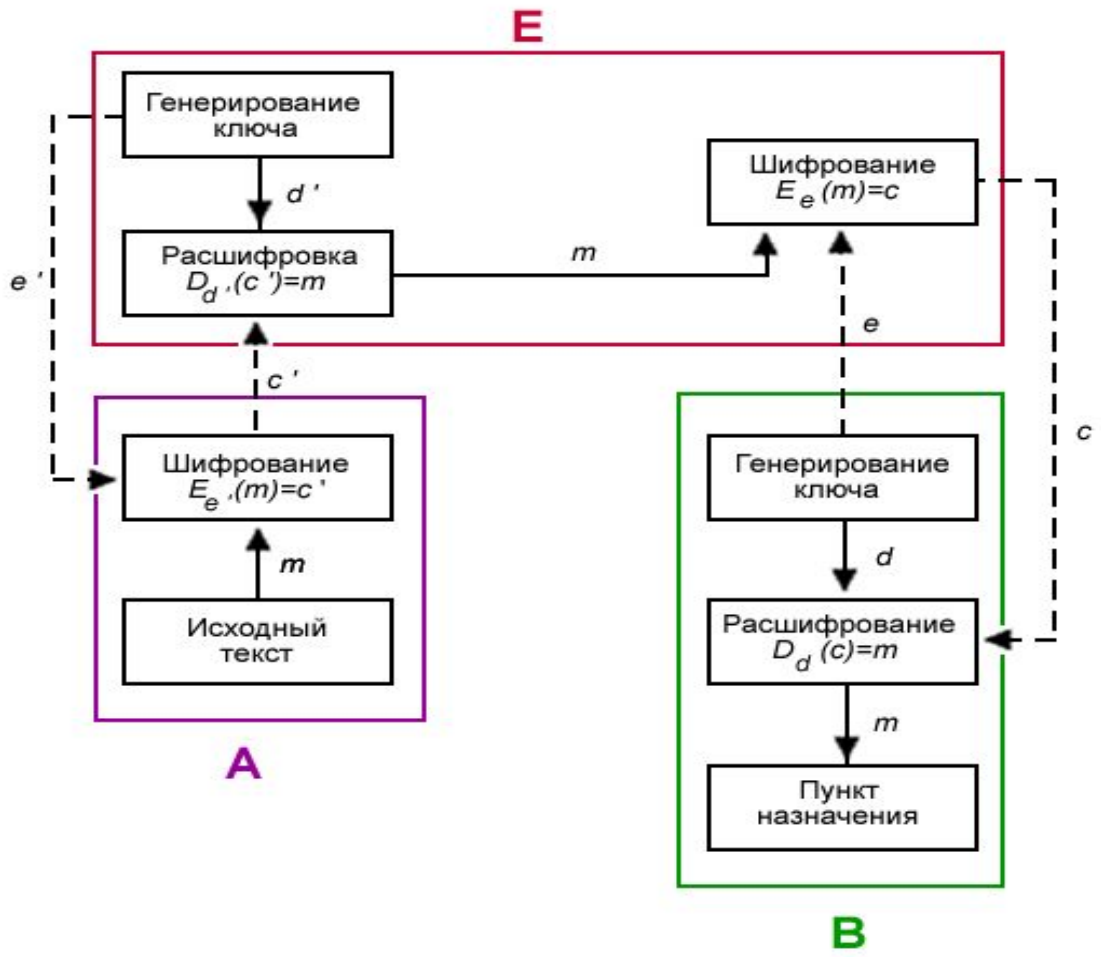



$$E_B : M \rightarrow C$$
$$D_B : C \rightarrow M$$

# Подмена открытого ключа асимметричной системы

А передает шифрованную информацию В. Е перехватывает открытый ключ e, посланный В для А. Затем создает пару ключей e'и d' , «маскируется» под В, посылая А открытый ключ e', который, как думает А, открытый ключ, посланный ему В. Е перехватывает зашифрованные сообщения от А к В, расшифровывает их с помощью секретного ключа d' , заново зашифровывает открытым ключом e участника В и отправляет сообщение В. Таким образом, никто из участников не догадывается, что есть третье лицо, которое может как просто перехватить сообщение , так и подменить его на ложное сообщение . Это подчеркивает необходимость аутентификации открытых ключей. Для этого используются сертификаты ключей.

# Подмена открытого ключа асимметричной системы



# Инфраструктура открытых ключей (1)

- **Инфраструктура открытых ключей** ( ***PKI*** - *Public Key Infrastructure*) - технология аутентификации с помощью открытых ключей. Это комплексная система, которая связывает открытые ключи с личностью пользователя посредством удостоверяющего центра (УЦ).
- Фактически, PKI представляет собой систему, основным компонентом которой является удостоверяющий центр и пользователи, взаимодействующие между собой посредством удостоверяющего центра

# Инфраструктура открытых ключей (2)

В основе PKI лежит использование криптографической системы с открытом ключом и несколько основных принципов:

- закрытый ключ известен только его владельцу;
- удостоверяющий центр создает сертификат открытого ключа, удостоверяя этот ключ;
- никто не доверяет друг другу, но все доверяют удостоверяющему центру;
- удостоверяющий центр подтверждает или опровергает принадлежность открытого ключа заданному лицу, которое владеет соответствующим закрытым ключом.

# Объекты PKI (1)

PKI реализуется в модели клиент-сервер.

**Основные компоненты PKI**

- **Удостоверяющий центр** (УЦ) является основной структурой, формирующей цифровые сертификаты подчиненных центров сертификации и конечных пользователей.
- **Сертификат открытого ключа** (чаще всего просто *сертификат*) - это данные пользователя и его открытый ключ, скрепленные подписью УЦ.

# Сертификаты открытых ключей. Основные понятия

- **X.500** – стандарт службы каталогов.
- Рекомендации **X.509** Международного союза телекоммуникаций (ITU – International Telecommunication Union) – часть рекомендаций серии X.500. Появился в 1988 году. После исправлений – в 1993 году.
- **Обозначения:**
- **Y«X»** - удостоверение пользователя X, выданное центром сертификации Y
- **Y{I}** – подпись I объектом Y. Она состоит из I с добавленным шифрованным хэш-кодом.

# ПРОТОКОЛ ЦЕНТРАЛИЗОВАННОГО РАСПРЕДЕЛЕНИЯ ОТКРЫТЫХ КЛЮЧЕЙ (1)

А – инициатор, запрашивает выдачу средств А и В. Сертификат А нужен, чтобы выработать пару ОК и ЗК. ОК посылает В.

**1.** А → ЦРК: $Id_A$, $Id_B$ «Пришлите сертификаты А и В»

**2.** ЦРК → А: ЦРК передает А два сертификата:

$C_A = E_{K^C_{ЦРК}}(h(L_A, K^O_A, Id_A)), (L_A, K^O_A, Id_A)$; $C_A = C_A \{L_A, K^O_A, Id_A\}$

$C_B = E_{K^C_{ЦРК}}(h(L_B, K^O_B, Id_B)), (L_B, K^O_B, Id_B)$. $C_B = C_B \{L_B, K^O_B, Id_B\}$

А проверяет подлинность сертификата В и берет себе $K^O_B$. Свой ОК у него есть. Проверяет сертификат В путем:

- Проверить подпись;
- Проверить сроки $L_A$, $L_B$ действия сертификатов $C_A$,$C_B$.

Успешная проверка подписи говорит о том, что информация подписана ЦРК и что ключ В $K^0_B$ – подлинный.

Проверка сроков $L_A$, $L_B$ используется для подтверждения актуальности сертификатов.

# ПРОТОКОЛ ЦЕНТРАЛИЗОВАННОГО РАСПРЕДЕЛЕНИЯ ОТКРЫТЫХ КЛЮЧЕЙ (2)

- A проверяет открытым ключом сертификат B
- **3.** $A \rightarrow B: C_A, E_{K^C_A}(T), E_{K^O_B}(r_1)$
- $C_A$ – сертификат открытого ключа A
- $E_{K^C_A}(T)$ –для аутентификации A. $E_{K^O_B}(r_1)$ – для проверки подлинности B.
- $r_1$ – некоторое случайное число
- **4.** $B \rightarrow A: E_{K^O_A}(f(r_1))$
- $K^O_B$ - открытый ключ B, $K^O_A$ - открытый ключ A. Y {I}- подпись I объектом Y. Это I с добавленным шифрованным хэш-кодом

# Объекты PKI (2)

**Регистрационный центр** (РЦ) - необязательный компонент системы, предназначенный для регистрации пользователей. Удостоверяющий центр доверяет регистрационному центру проверку информации о субъекте. Регистрационный центр, проверив правильность информации, подписывает её своим ключом и передаёт удостоверяющему центру, который, проверив ключ регистрационного центра, выписывает сертификат.

# Объекты PKI (3)

- Один **регистрационный центр** может работать с несколькими удостоверяющими центрами (т.е. состоять в нескольких PKI), один удостоверяющий центр может работать с несколькими регистрационными центрами. Иногда, удостоверяющий центр выполняет функции регистрационного центра.
- **Конечные пользователи** - пользователи или приложения, являющиеся владельцами сертификата и использующие инфраструктуру управления открытыми ключами.

# Основные поля сертификата X.509 (1)

V – версия:

SN – порядковый номер;

AI – идентификатор алгоритма подписи (не слишком полезное, т.к. в конце есть поле в подписи);

CA – имя объекта, выдавшего сертификат;

$T_A$ – срок действия;

A – имя субъекта;

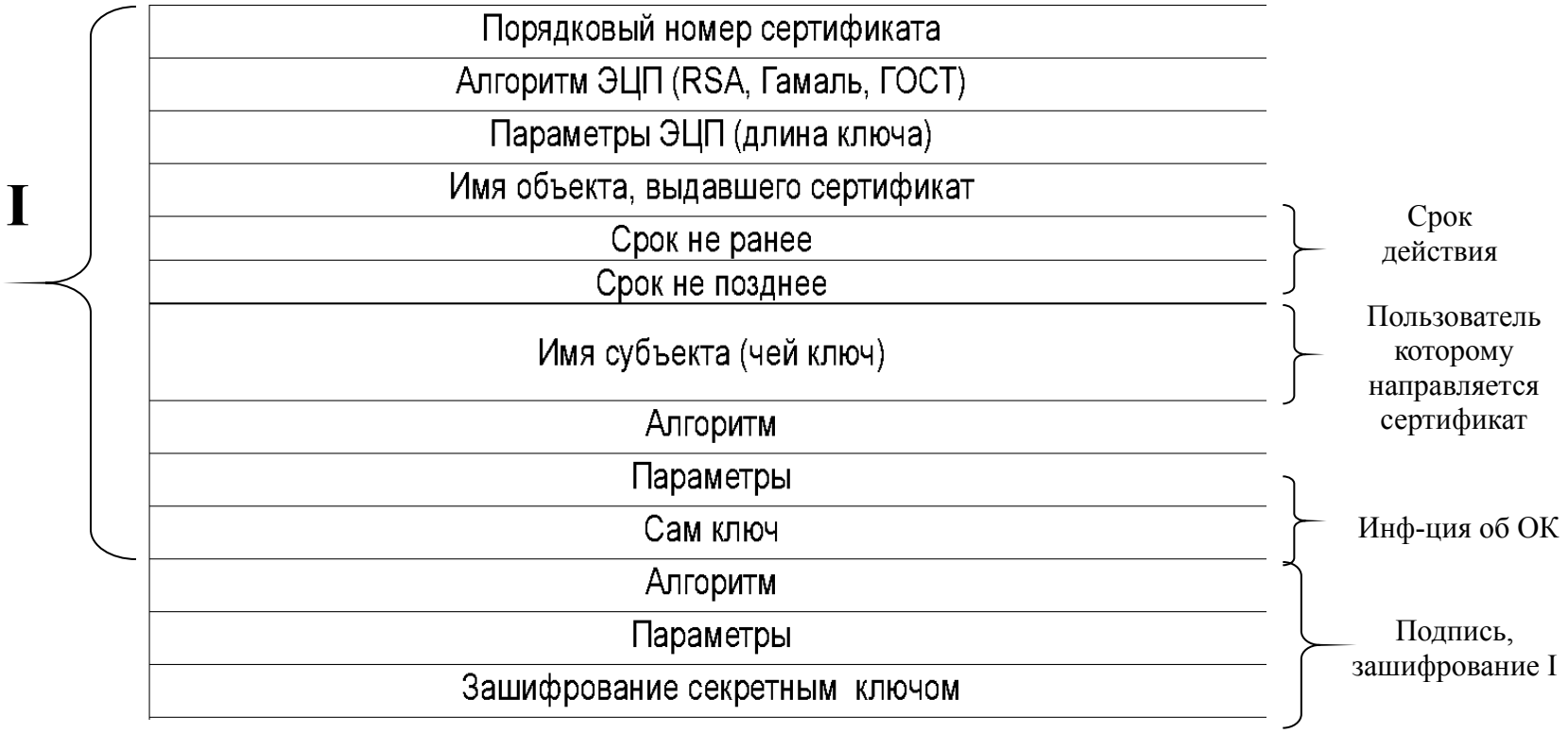
$A_P$ – информация об открытом ключе субъекта.

$CA«A» = CA\{V,SN,AI,CA,T_A, A, A_P\}$, где

Y«X» - удостоверение X, выдан. Y; Y{I} – подпись I объектом Y.

# Структура сертификата X.509

Рекомендация ITU – international telecommunicaon Union



# Стандарт X.509 версия 3

| | | |
|---|---|---|
| Version | Версия сертификата | 3 |
| Certificate Serial Number | Серийный номер сертификата | 40:00:00:00:00:00:00:ab:38:1e:8b:e9:00:31:0c:60 |
| Signature Algorithm Identifier | Идентификатор алгоритма ЭЦП | ГОСТ Р 34.10-94 |
| Issuer X.500 Name | Имя Издателя сертификата | C=RU, ST=Moscow,O=PKI, CN=Certification Authority |
| Validity Period | Срок действия сертификата | Действителен с : Ноя 2 06:59:00 1999 GMT<br>Действителен по : Ноя 6 06:59:00 2004 GMT |
| Subject X.500 Name | Имя Владельца сертификата | C=RU, ST=Moscow, O=PKI, CN=Sidorov |
| Subject Public Key Info | Открытый ключ Владельца | тип ключа: Открытый ключ ГОСТ<br>длина ключа: 1024<br>значение: AF:ED:80:43..... |
| Issuer Unique ID version 2 | Уникальный идентификатор Издателя | |
| Subject Unique ID version 2 | Уникальный идентификатор Владельца | |
| type critical value | | **дополнения (только версия 3)** |
| type critical value | | |
| type critical value | | |
| **CA Signature<br>ЭЦП Центра Сертификации** | | |

Типы дополнений:

- ограничивающие
- информационные

# Отзыв сертификатов (1)

- В некоторых ситуациях желательно иметь возможность отменить действие сертификата до окончания срока его действия по следующим причинам:
- 1. Секретный ключ пользователя оказался скомпрометированным.
- 2. Пользователь больше не сертифицируется в данном центре сертификации.
- 3. Сертификат данного центра сертификации оказался скомпрометированным.

# Отзыв сертификатов (2)

- Каждый центр должен поддерживать список отозванных сертификатов (CRL – Certificate Revocation List). CRL должны размещаться в каталоге, подписываются центром сертификации и включают имя центра, дату создания списка, дату выхода следующей версии CRL и **запись** для каждого отозванного сертификата. **Запись** состоит из порядкового номера сертификата и даты отзыва этого сертфиката.

# Важность стандарта X.509

- Структура сертификатов и протоколов аутентификации, определяемых в X.509, используется в протоколах S/MIME (Secure/Multipurpose Internet Mail Extension) – защищенное многоцелевое расширение электронной почты, IP Security, SSL/TLS, SET.

# Электронная цифровая подпись по RSA

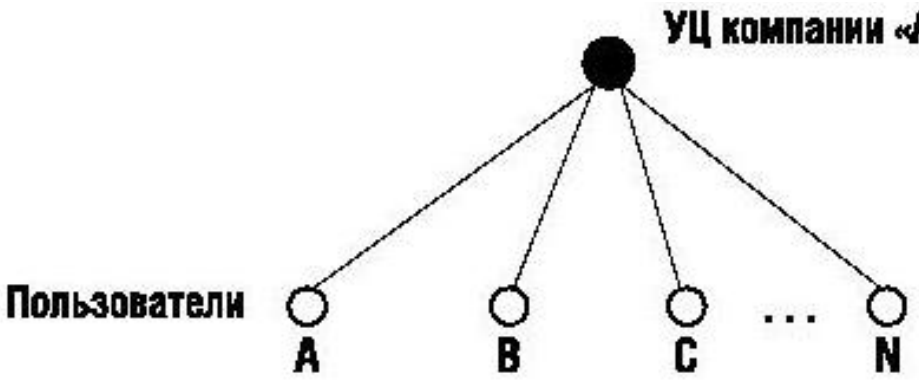
- Подписывание A:
- $SA = m^{d}_{A} \mod n$
- $d_A$ – секретный ключ
- Проверка для B:
- A□ B: SA, M'
- B:m' = h (M')
- $m = (SA)^{e}_{A} \mod n$
- сравнивает m = m'

# Архитектуры PKI

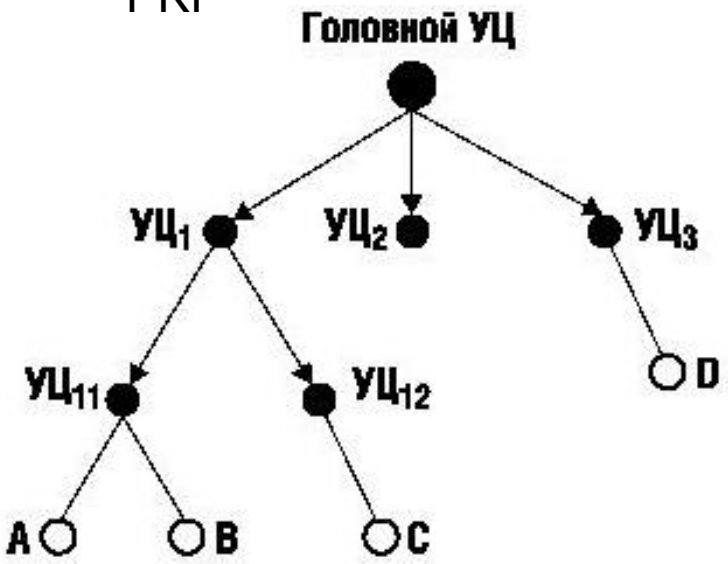
- В основном выделяют 5 видов архитектур PKI, это:
- простая PKI (одиночный УЦ)
- иерархическая PKI (подчинение нескольких УЦ вышестоящему головному УЦ)
- сетевая PKI (объединение одноранговых инфраструктур с перекрестной (кросс-) сертификацией головных УЦ)

# Архитектура Public Key Infrastructure

Одиночный УЦ



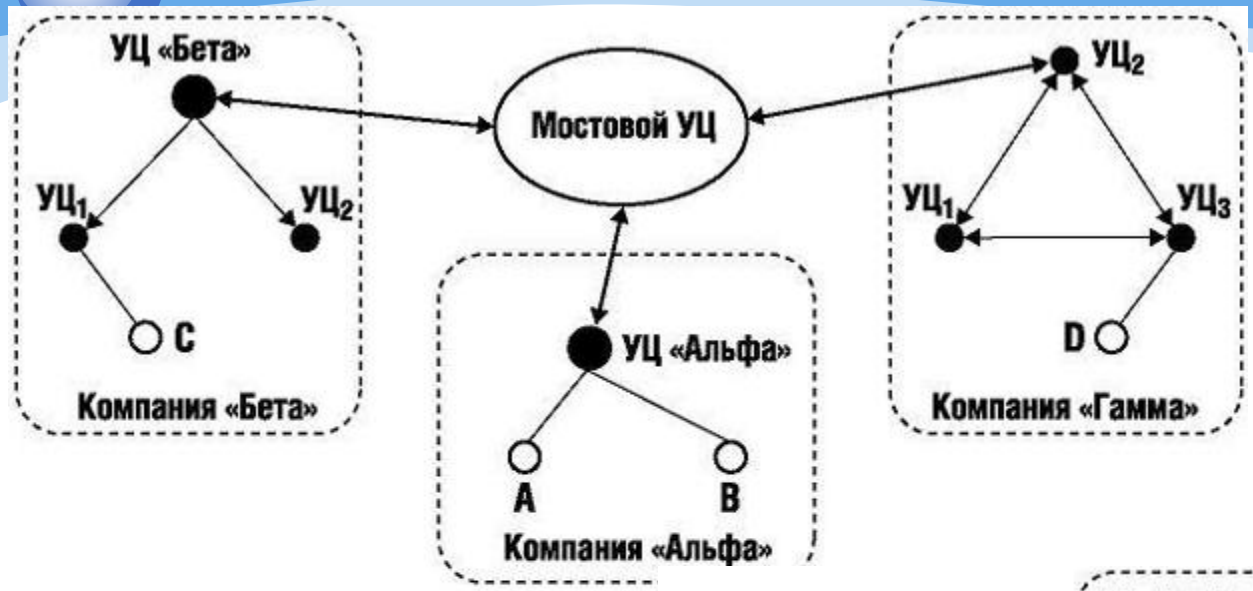

Иерархическая PKI



# Архитектуры PKI

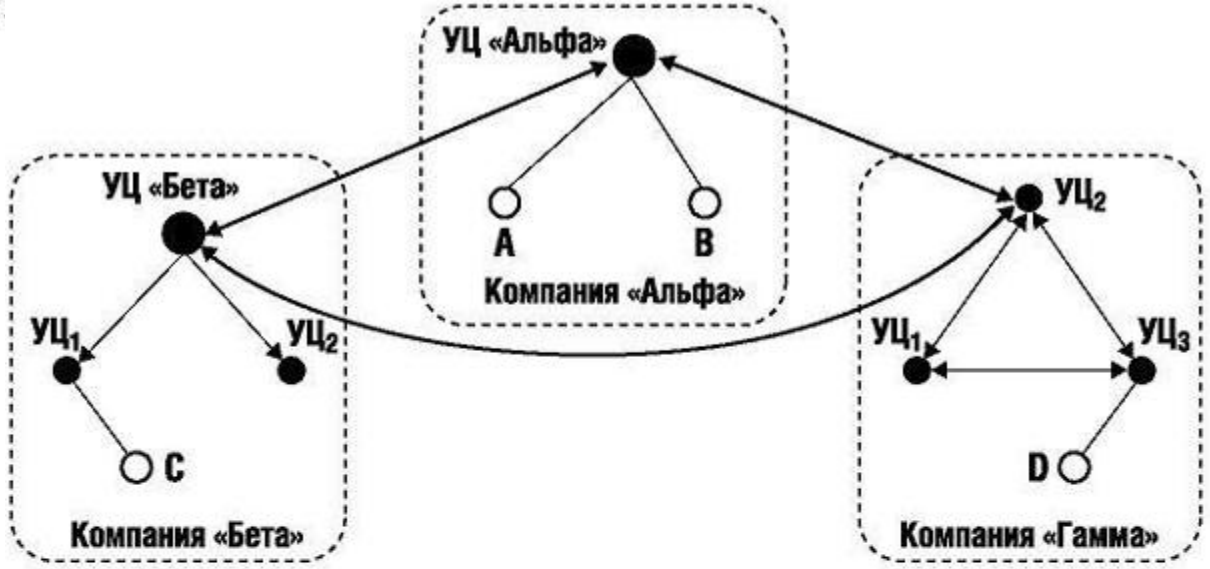
- 4.Кросс-сертифицированные корпоративные PKI (смешанный вид иерархической и сетевой архитектур. Есть несколько фирм, у каждой из которых организована какая-то своя PKI, но они хотят общаться между собой
- 5.Архитектура мостового УЦ (убирает недостатки сложного процесса сертификации в кросс-сертифицированной корпоративной PKI. В данном случае все компании доверяют не какой-то одной или двум фирмам, а одному определённому мостовому УЦ, который является практически их головным УЦ)

# Архитектура PKI (2)

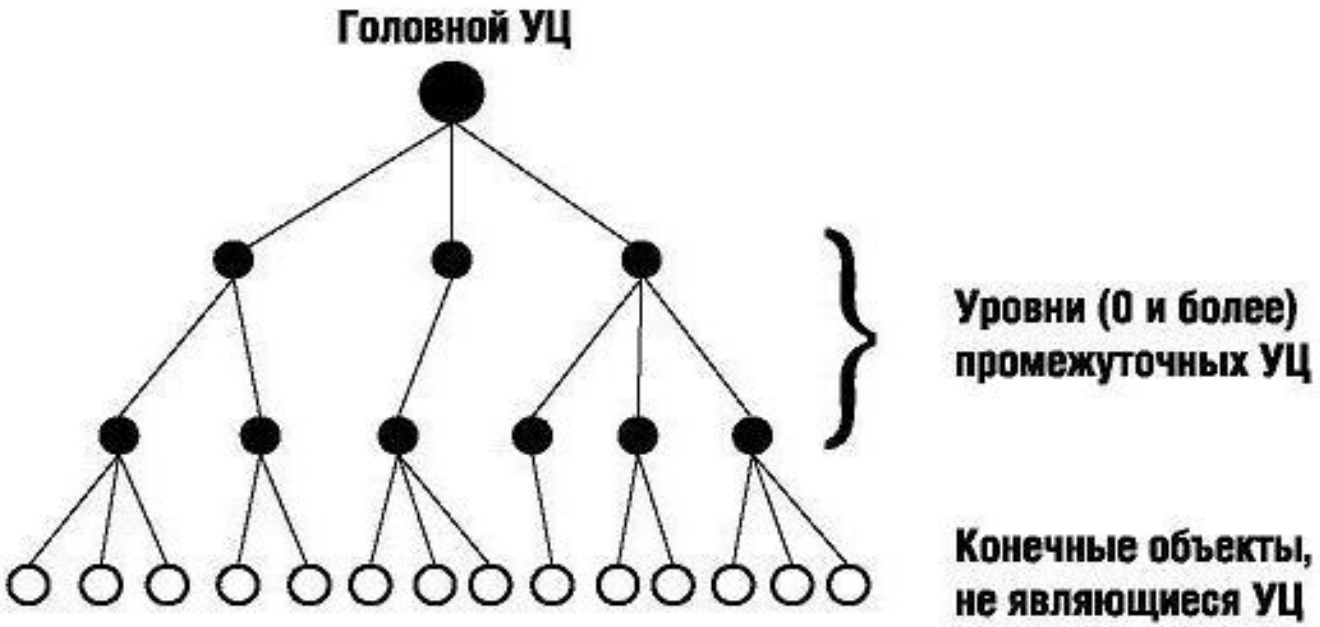



Архитектура мостового УЦ

Кросс-сертифицированные корпоративные PKI

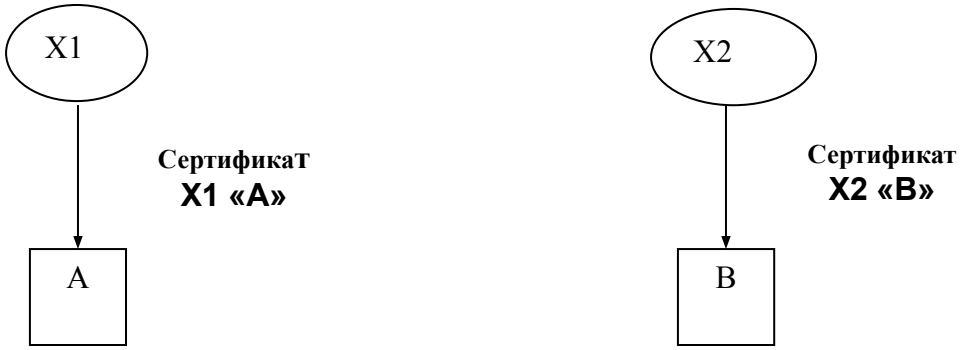


# . Строгая иерархия удостоверяющих центров



# Участники обслуживаются в разных ЦРК.

- Сертифицирующие центры – X1 и X2




X1 «A» - удостоверение пользователя A выданное центром сертификации X1
X2 «B» - удостоверение пользователя B выданное центром сертификации X2

A от B: X1 «X2»  X2 «B»

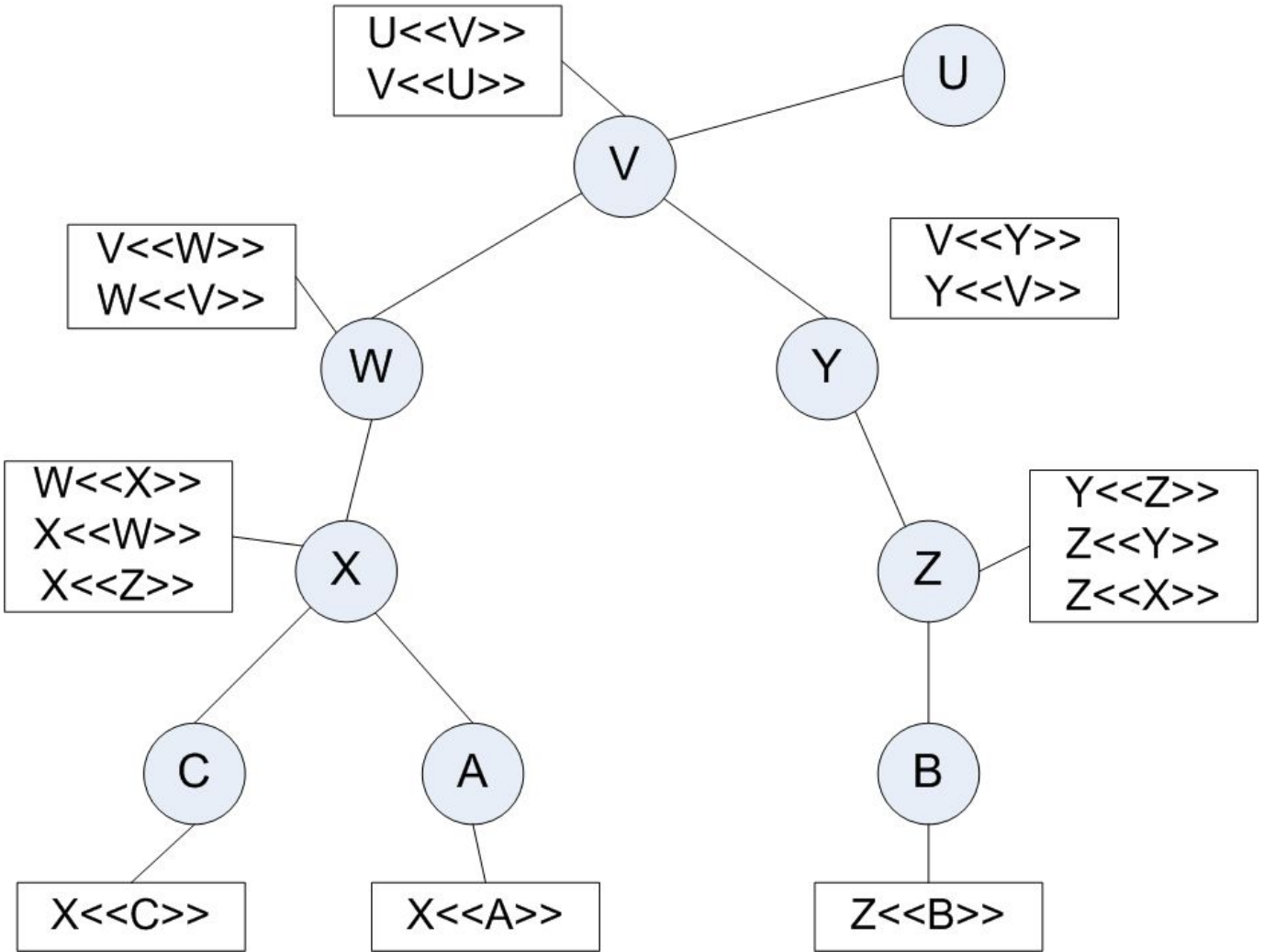
B от A X2 «X1»  X1 «A»

X1 «X2» X2 «X3» … XN «B» - цепочка из N элементов

# Построение цепочки доверия

X<<W>>W<<V>>V<<Y>>Y<<Z>>Z<<B>> **: от A к B**

Z <<Y>> Y <<V>> V <<W>> W <<X>>X <<A>>**: от B к A**



# Прямые, возвратные и самоподписанные сертификаты

- **Прямые сертификаты.** Сертификаты X, выданные другими центрами сертификации.
- **Возвратные сертификаты.** Сертификаты, выданные X для сертификации других центров сертификации.
- **Самоподписанный сертификат.** Открытый ключ для корневой подписи распространяется с автоподписью. Известен всем программным средствам.

# Структура иерархической PKI (1)

- Имеется главный (корневой) управляющий центр (назовем его УЦ$^1$). Подлинность открытого ключа УЦ$^1$ подтверждается соответствующим юридическим документом. УЦ$^1$ составляет справочники открытых ключей и выдает сертификаты пользователям второго уровня $P^2_i$ и управляющим центрам второго уровня УЦ$^2_j$. Эти справочники и сертификаты УЦ$^1$ подписывает своим ключом.

# Структура иерархической PKI (2)

Каждый управляющий центр второго уровня обслуживает свою группу пользователей и управляющих центров третьего уровня, подписывая их открытые ключи своим.

В такой системе может быть произвольное количество уровней.

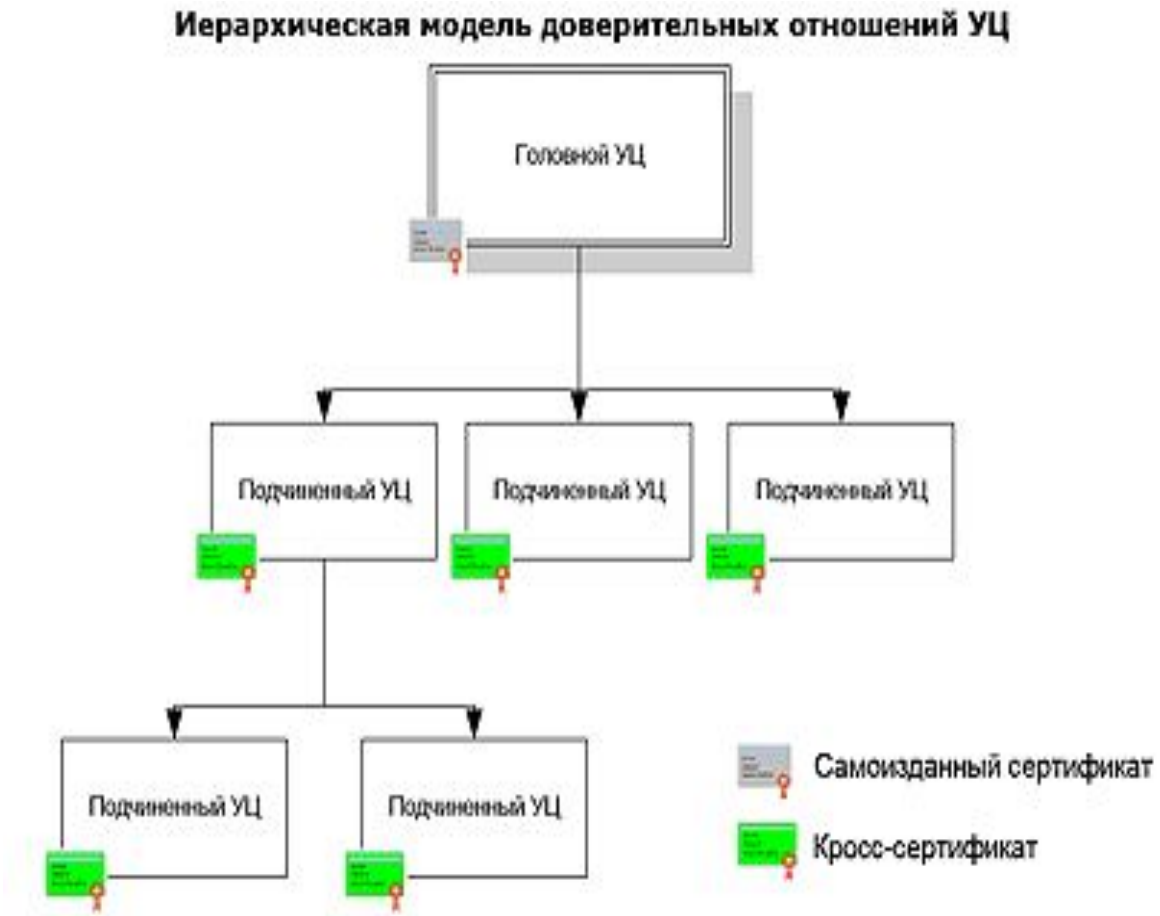
# Проверка сертификатов в иерархической PKI (1)

- Для того, чтобы проверить принадлежность открытого ключа пользователя *n*-го уровня, необходимо проверить сертификат, выданный соответствующим УЦ *n*-1-го уровня. Подпись этого УЦ можно проверить по сертификату, выданному УЦ *n*-2 – го уровня, и т. д., а подлинность подписи корневого УЦ гарантируется юридическим документом.

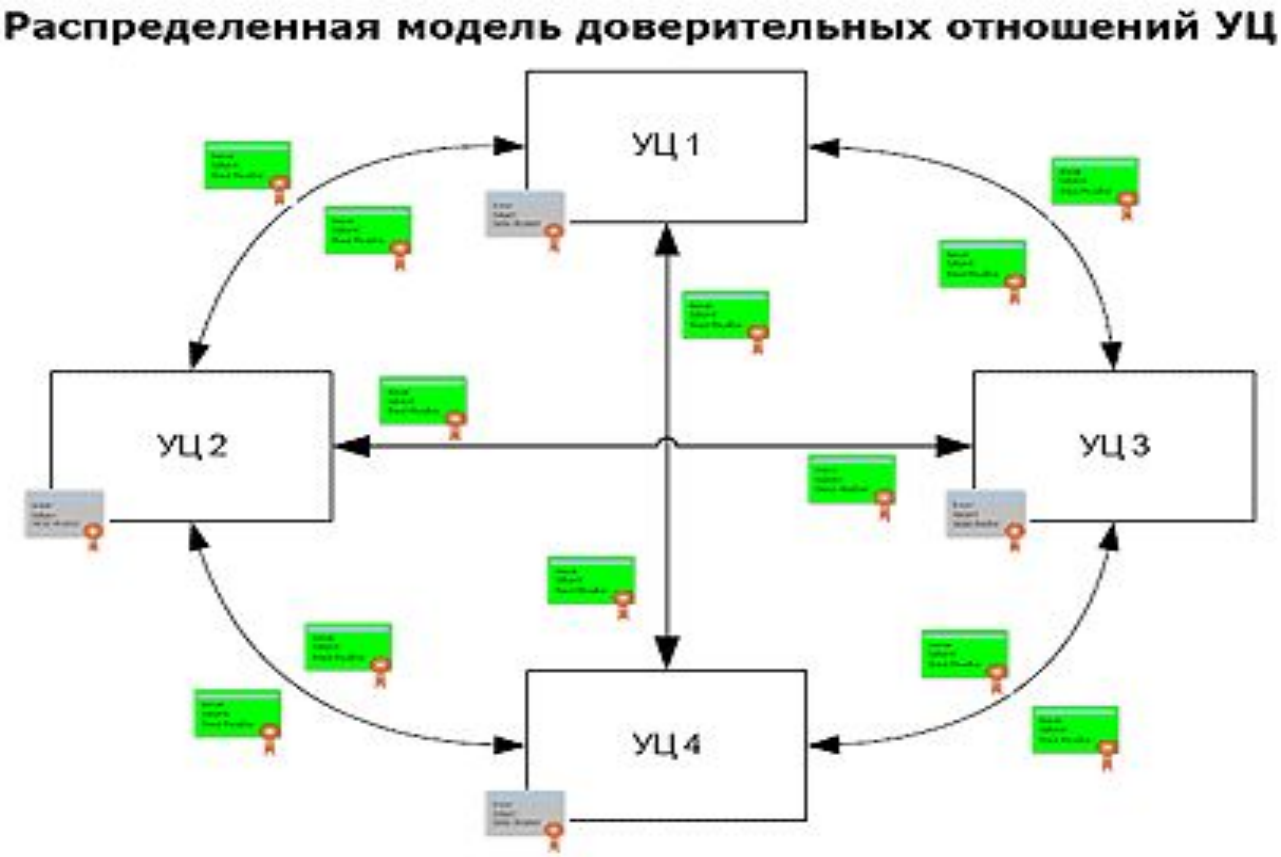
# Проверка сертификатов в иерархической PKI (2)

- Для того чтобы все пользователи системы могли проверить подлинность сертификатов друг друга, каждый из УЦ, к которому они принадлежат, распределяет между пользователями подписанный этим УЦ справочник открытых ключей, в котором указаны открытые ключи главного УЦ и всех подчиненных УЦ, через которые проходит путь от данного пользователя к главному УЦ

# Иерархическая модель доверительных отношений УЦ



# Распределенная модель доверительных отношений УЦ



# Кросс-сертификаты

В распределенной модели доверительных отношения, все Центры Сертификации удостоверяющих центров имеют самоизданные сертификаты. Удостоверяющие центры устанавливают между собой доверительные отношения попарно, путем выпуска кросс-сертификатов Центров Сертификации. Таким образом, каждый Центр Сертификации помимо самоизданного сертификата является владельцем кросс-сертификатов, в количестве, равном числу Центров Сертификации, с кем были установлены доверительные отношения.