

ОПЫТ ПРИМЕНЕНИЯ ПЛАТФОРМЫ РАСПРЕДЕЛЁННЫХ РЕЕСТРОВ МАСТЕРЧЕЙН ДЛЯ ФИНАНСОВЫХ ПРИЛОЖЕНИЙ

Алексей Цветков,
Ассоциация ФинТех



Разрабатываем и внедряем
новые технологические решения
для развития финансового рынка:

программное обеспечение,
стандарты и протоколы, а также
готовим предложения по созданию
и изменению законодательства
в области цифровой экономики.



Банк России



ГАЗПРОМБАНК



НСПК



РоссельхозБанк



ВТБ



открытие | БАНК



QIWI



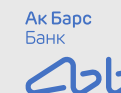
Альфа-Банк



СБЕРБАНК



Тинькофф
Банк



Ак Барс
Банк



Райффайзен
БАНК



МОСКОВСКАЯ
БИРЖА



ПСБ



МОСКОВСКИЙ
КРЕДИТНЫЙ БАНК



РНКО ПЛАТЕЖНЫЙ ЦЕНТР



ВСК СТРАХОВОЙ ДОМ



Абсолют Банк



МЕГАФОН



АЛЬФА
СТРАХОВАНИЕ



РОСБАНК
SOCIETE GENERALE GROUP



СОВКОМБАНК



Ростелеком



МТС



ГРУППА КОМПАНИЙ
РАГИОН



СКБ-БАНК



Ренессанс.
СТРАХОВАНИЕ



РУССКИЙ
СТАНДАРТ
БАНК



Яндекс



ПОЧТА
БАНК



РЕСО
ГАРАНТИЯ

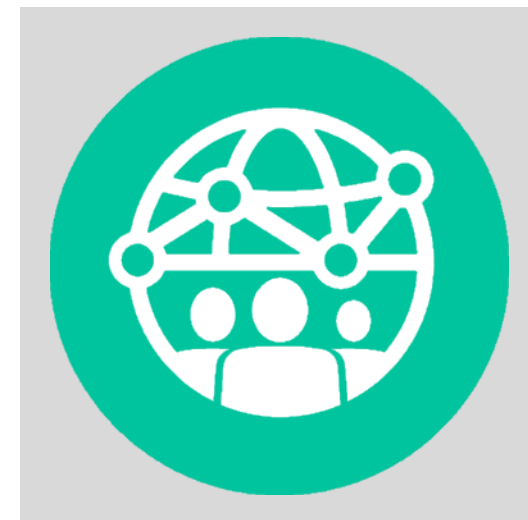
РОСГОССТРАХ



Платформа распределенных реестров для финансовых институтов, сертифицирована ФСБ и соответствует требованиям Банка России.

На базе ГОСТ-криптографии реализованы:

- протокол исполнения смарт-контрактов (EVM)
- сервис передачи конфиденциальных сообщений
- создание закрытых сетей с управляемым доступом
- средства мониторинга и администрирования





Взаимодействие участников сети Мастерчейн происходит по единым для всех правилам, фиксируемым в реестре.

Пилотные и промышленные кейсы:

- регистрация и учёт ценных бумаг (закладные)
- торговое финансирование (гарантии, аккредитивы)
- передача платёжной информации





Программный код и результаты выполнения смарт-контрактов в сети Мастерчейн имеют доступность, целостность и неотказуемость.

Как оператор сети мы требуем использовать в прикладных контрактах разработанные нами механизмы отключения и обновления кода.

```
1 // Контракт с возможностью отключения
2 contract Suspendable {
3
4     // Флаг отключения
5     bool public disabled;
6
7     // Барьер: отключен ли контракт?
8     modifier enabled() {
9         require(disabled == false);
10    }
11
12     // Функция проверки прав администратора,
13     // реализуется в дочерних контрактах
14     function canDisable()
15     public view returns(bool);
16
17     // Отключить для изменения
18     function disable() public enabled {
19         require(canDisable());
20         disabled = true;
```

```
1 // Контракт с возможностью обновления
2 contract Upgradable is Suspendable {
3
4     // Адрес предыдущей версии
5     Upgradable public precursor;
6
7     // Адрес следующей версии
8     Upgradable public successor;
9
10    // Инициализировать с указанием
11    // адреса предыдущей версии
12    function Upgradable(Upgradable _precursor)
13    public {
14        if (_precursor != address(0))
15            _precursor.upgradeWith(this);
16        precursor = _precursor;
17    }
```

```
18     // Указать адрес следующей версии
19     // и отключить
20     function upgradeWith(Upgradable _successor)
21     public {
22         disable();
23         successor = _successor;
24     }
25
26     // Название смарт-контракта
27     function contractName()
28     public pure returns(string);
29
30     // Версия смарт-контракта
31     function contractVersion()
32     public pure returns(string);
33
34 }
```



Обеспечить безопасность взаимодействия финансовых институтов.

Участники финансового рынка:

- самостоятельно управляют своей инфраструктурой
- строят бизнес-процессы обособленно

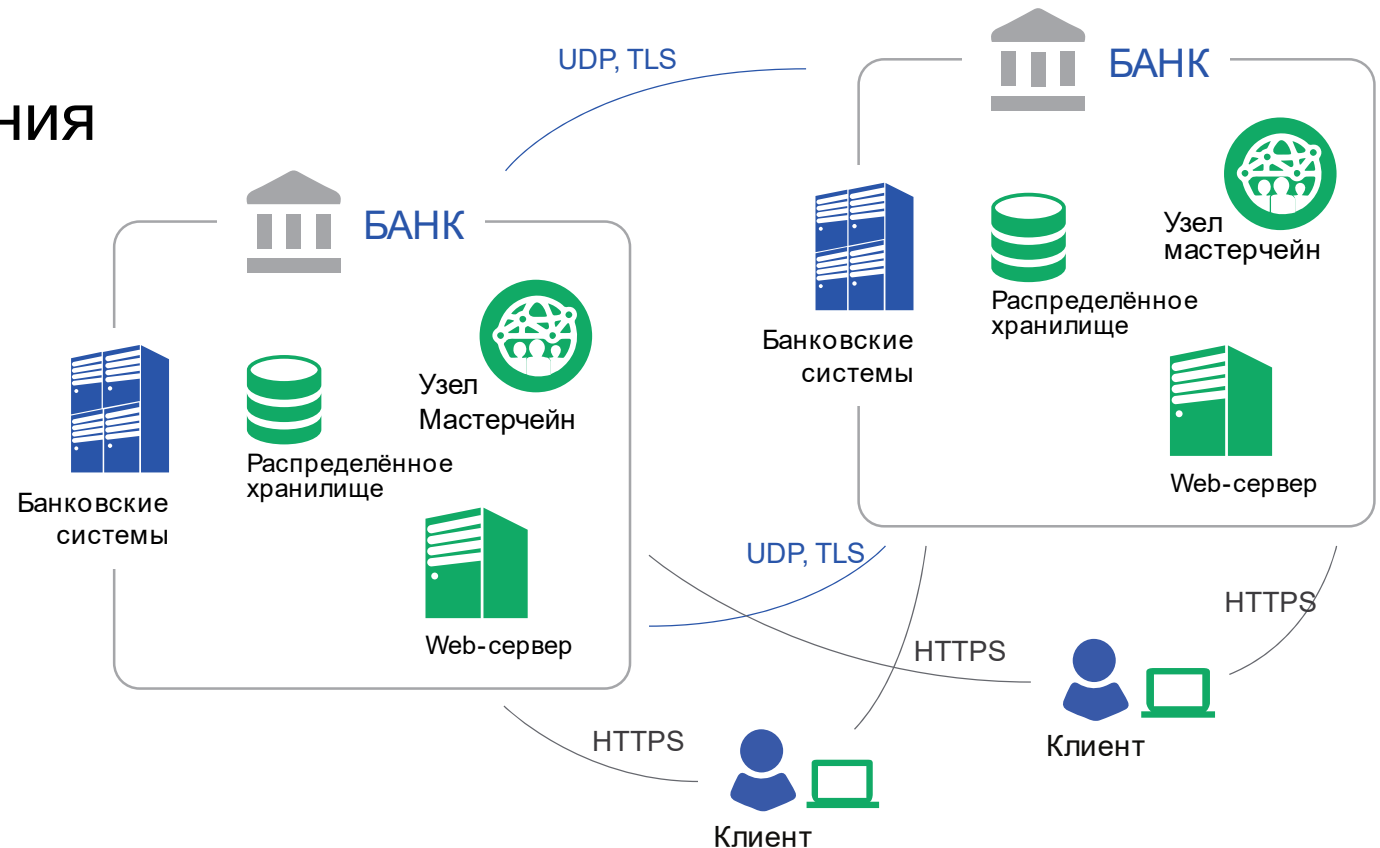
Зрелость процессов обеспечения безопасности
сильно различается между организациями.





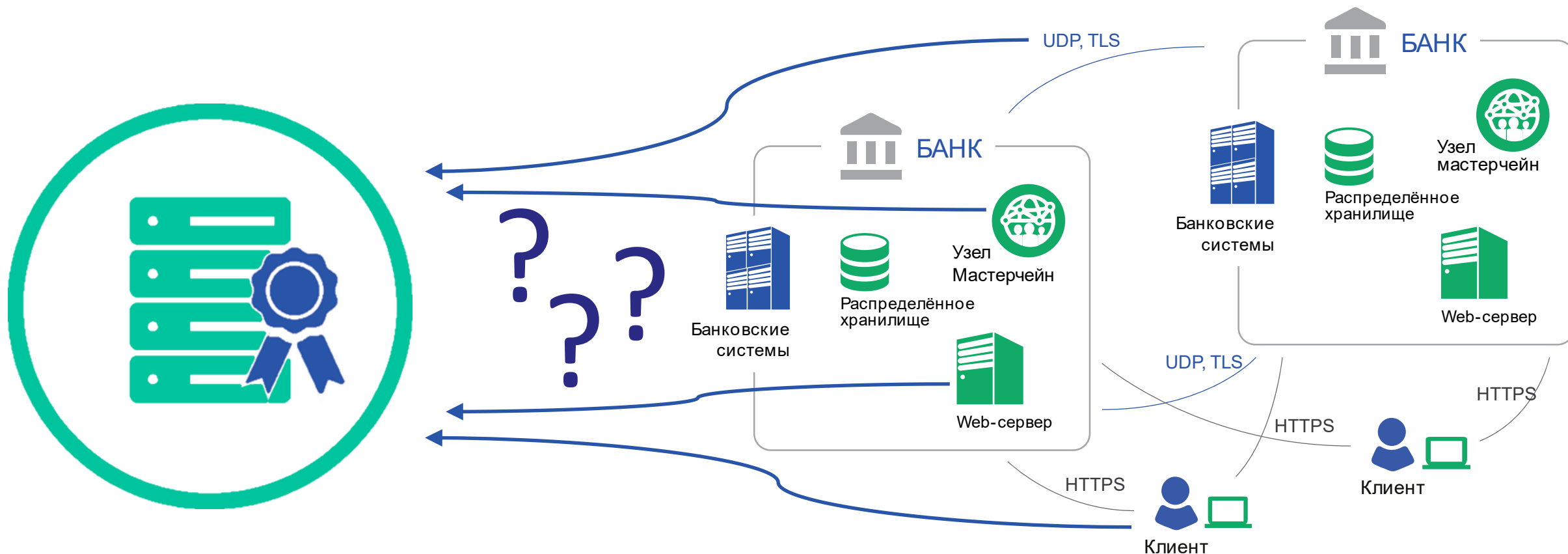
Системы распределённых реестров позволяют объединить ресурсы каждого участника в доверенную цифровую среду.

Для согласованного управления
криптографическими
операциями и для проверки
используется PKI.





В среде распределённых реестров РКИ должна быть встроена так, чтобы сохранить отказоустойчивость и производительность.





Внедрение PKI в логику бизнес-процессов осложняется, так как необходимо заменять ключи с течением времени.

Авторизация бизнес-операций должна происходить на основании отношений между организациями.

Структура отношений в кейсе может быть многоуровневой.





Оператор сети является оператором распределённого КриптоПро УЦ 2.0.
Он управляет выпуском и отзывом сертификатов участников сети.

В шаблоны выпускаемых сертификатов добавлены идентификаторы расширенного назначения для разделения ключей по типам операций:

1.2.643.6.57.1.1.1.1	[TLS Server]	Узел Мастерчейн	1.2.643.6.57.1.3.1.3	[TLS Client]	АРМ Админ сети
1.2.643.6.57.1.1.1.2	[TLS Client]	Узел Мастерчейн	1.2.643.6.57.1.3.2.1	[Sign]	Админ сети АФТ
1.2.643.6.57.1.1.1.3	[TLS Server]	API узла Мастерчейн	1.2.643.6.57.1.3.2.2	[Sign]	Админ Whitelist
1.2.643.6.57.1.1.3.1	[Sign]	Узел Мастерчейн	1.2.643.6.57.1.5.1.1	[TLS Server]	API ИС
1.2.643.6.57.1.2.1.1	[TLS Server]	API СПКС	1.2.643.6.57.1.5.1.2	[TLS Client]	API ИС
1.2.643.6.57.1.2.1.2	[TLS Client]	API СПКС	1.2.643.6.57.1.5.1.3	[TLS Client]	АРМ ИС
1.2.643.6.57.1.2.2.1	[Sign]	Узел СПКС	1.2.643.6.57.1.5.2.1	[Sign]	АРМ ИС Админ АФТ
1.2.643.6.57.1.2.4.1	[Encryption]	Хранилище СПКС	1.2.643.6.57.1.5.2.2	[Sign]	АРМ ИС Робот
1.2.643.6.57.1.2.4.2	[Encryption]	Робот ИС	1.2.643.6.57.1.5.2.3	[Sign]	АРМ ИС Админ
1.2.643.6.57.1.2.4.3	[Encryption]	Менеджер ИС	1.2.643.6.57.1.5.2.4	[Sign]	АРМ ИС Менеджер



Цели и причины использования:

- совместимость с протоколом Ethereum
- сокращение объёма хранимой и передаваемой информации





В качестве идентификатора сертификата в смарт-контракте реестра участников используется поле X.509 v2 Subject Key Identifier. Его значение не должно иметь коллизии в рамках одного УЦ.

- 160 бит SKID соответствуют формату адреса Ethereum
- для использования сертификатов разных УЦ в одной сети требуется включить в структуру транзакции поле Authority KID
- сертификаты УЦ и списки отзыва доступны в реестре по AKID
- компоненты ИС могут обновлять СОС из реестра, а не из CDP



Распределённый реестр с авторизацией, автоматизацией операций в смарт-контрактах, с контролем целостности и доступностью в LAN.

Отметка об отзыве сертификата, о сроках годности полномочий и ОРГН владельца сертификата доступны для логики смарт-контрактов.

```
1  /// @notice Реестр AKID => SKID => запись о ключе
2  mapping(address => mapping(address => Key)) public keys;
3
4  /// @notice Запись о ключе
5  struct Key {
6      bool        enabled;          /// флаг действительности/блокировки
7      bytes32     owner;             /// ID организации из сертификата
8      bytes       certificate;       /// байты сертификата ключа
9      bytes       revocationList;    /// байты списка отзыва ключа УЦ
10     bytes32     revocationHMAC;    /// HMAC списка отзыва в СПКС
11     address      profile;          /// ID контейнера с профилем ключа
12     /// ...
```



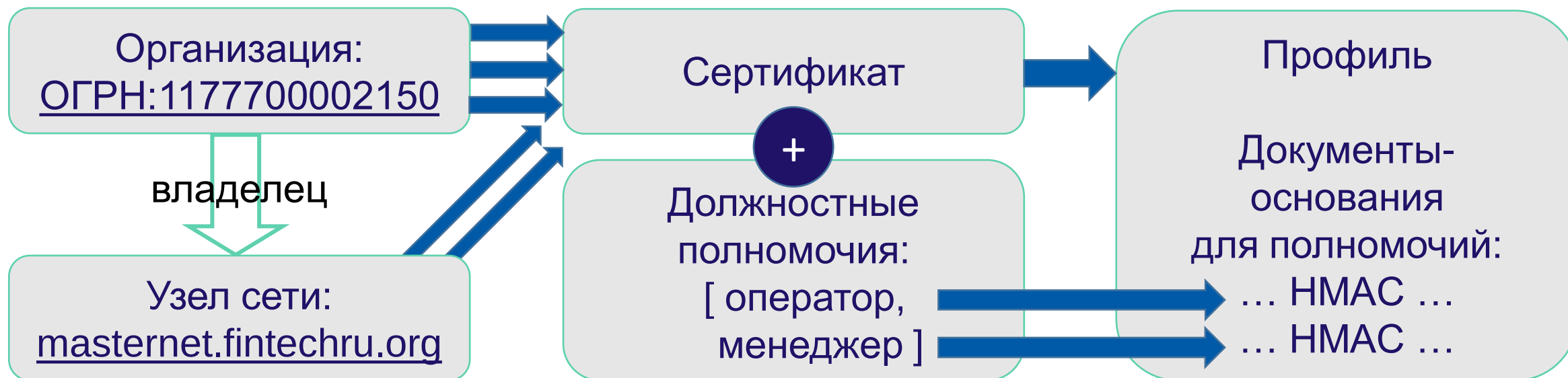

Распределённый реестр с авторизацией, автоматизацией операций в смарт-контрактах, с контролем целостности и доступностью в LAN.

```
12  /// ...
13  bytes32[] functionIds;    /// список ID функций ключа
14  /// реестр записей о функциях ключа
15  mapping(bytes32 => Function) functions;
16  }
17
18  /// @notice Запись о функции ключа
19  struct Function {
20      bytes32 reasonHMAC; /// HMAC архива с основанием для функции
21      uint64  startTime;  /// метка времени начала действия функции
22      uint64  expiryTime; /// метка времени окончания действия функции
23  }
```

Документы-основания для назначенных ключам полномочий,
а также СОС для ключей УЦ доступны в распределённом файловом слое.

⊕ В системном контракте в блокчейне перечислены:

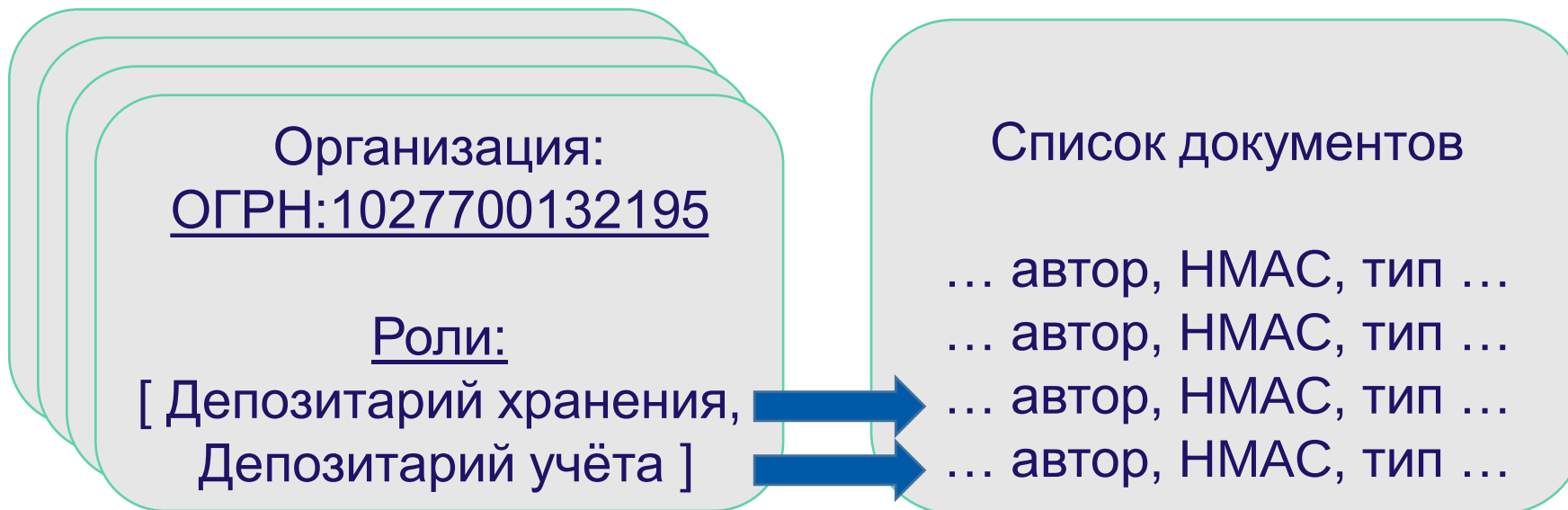
- веб-сервисы сети (узлы) с указанием организации-владельца
- организации участники-сети с перечислением их сертификатов
- сертификаты X.509 и назначенные им должностные полномочия





Каждая сделка в блокчейне представлена перечислением:

- организаций-участников и их ролей в процессе
- конфиденциальных документов (сертификат автора, НМАС, тип, ...)



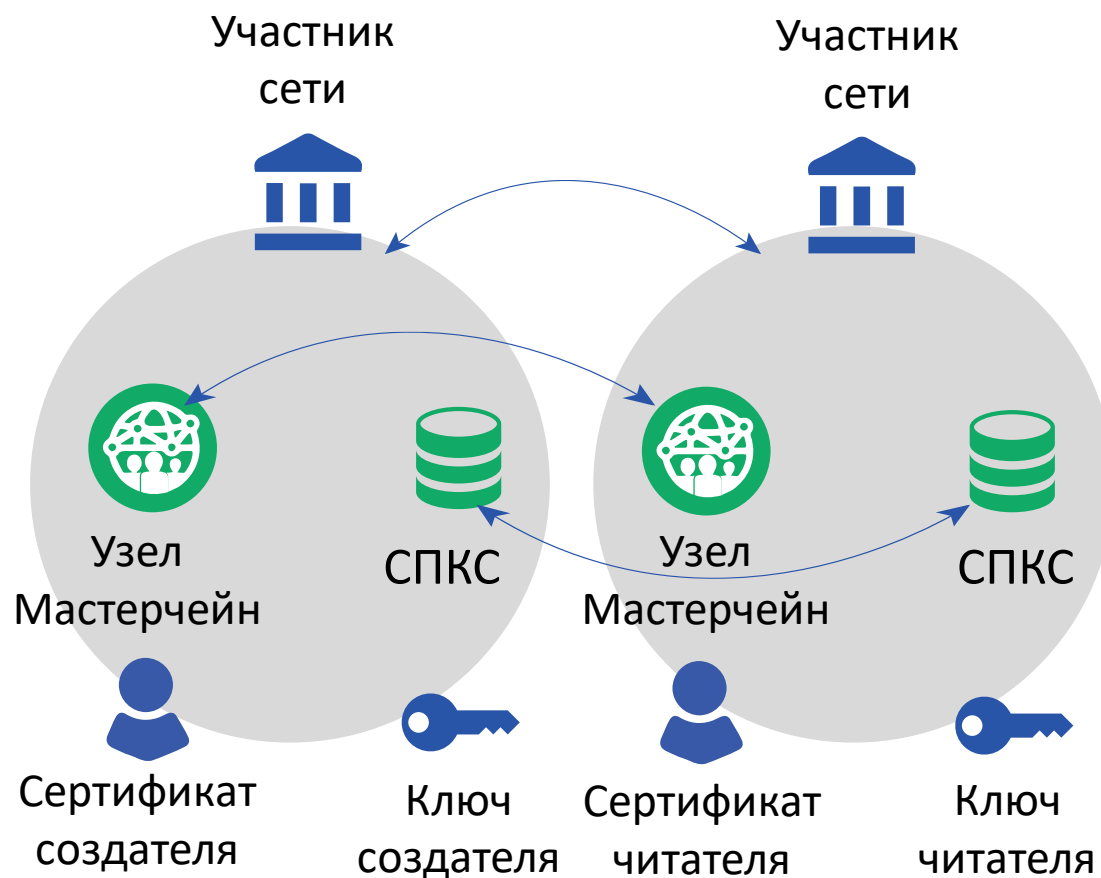


Разработчик приложения записывает матрицу прав доступа в контракт.

Права	Тип объекта доступа	Полномочия ключа	Роли организаций
Чтение	Ипотечная закладная	Менеджер	[Депозитарий хранения]
Запись	Черновик гарантии	Операционист	[Банк-гарант]
Чтение	Документ-основание для назначения роли	*Любые*	[*Любые*]
Запись	Назначение полномочий ключа	Администратор	[Оператор сети]



Веб-сервис предоставляет доступ к документам согласно ролевой модели.



1. Аутентификация запросов и ответов производится по сертификатам.
2. В реестре публикуются события о сохранении документов на узлах.
3. Читатель по записи в реестре находит узел и запрашивает документ.
4. Копию документа читатель сохраняет на своём узле сети.



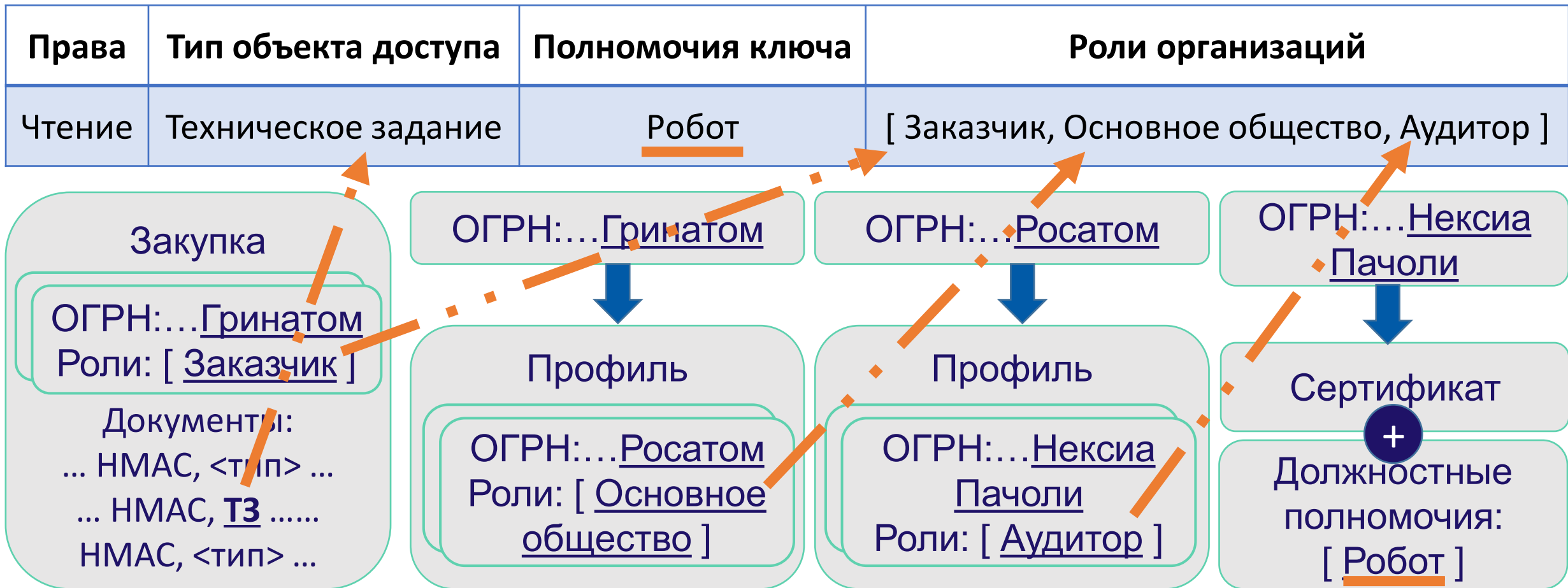
Системные смарт-контракты платформы обеспечивают:

- независимость бизнес-процессов от процесса управления ключами
- конфиденциальность документов, их целостность и неотказуемость





В матрице прав доступа возможно указать последовательность ролей.





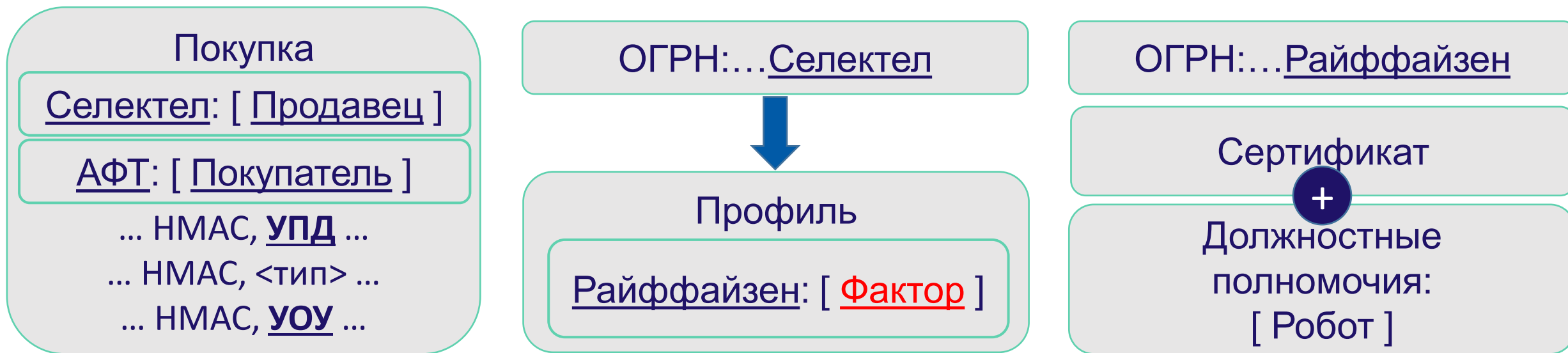
Прямое дебетование (разрешение на списание оплаты за услуги).

Права	Тип объекта доступа	Полномочия ключа	Роли организаций
Запись	Платёж	Робот	[Владелец, <u>Поставщик услуг</u>]
Чтение	Платёж	Робот	[Банк]
Запись	Баланс	Робот	[Банк]



Замкнутый факторинг (с уведомлением в случае невыплаты).

Права	Тип объекта доступа	Полномочия ключа	Роли организаций
Запись	Универсальный передаточный документ	Директор	[Продавец]
Чтение	Универсальный передаточный документ	Робот	[Продавец, Фактор]
Запись	Уведомление об уступке	Робот	[Продавец, Фактор]





Управление реестром участников сети как бизнес-процесс.

Права	Тип объекта доступа	Полномочия ключа	Роли организаций
Запись	Регистрация сервисов	Администратор	[Оператор]
Запись	Регистрация организаций-клиентов	Администратор	[Банк]
Запись	Назначение полномочий сертификата	Администратор	[Клиент]

Смарт-контракт реестра Мастерчейн наследует контракт бизнес-процесса.

Проверки прав на изменения реестра выполняются согласно ролевой модели.

Приложения могут определять новые типы объектов и проверять права доступа к ним в своих смарт-контрактах или веб-сервисах.



Мастерчейн полноценно использует и расширяет концепцию PKI.

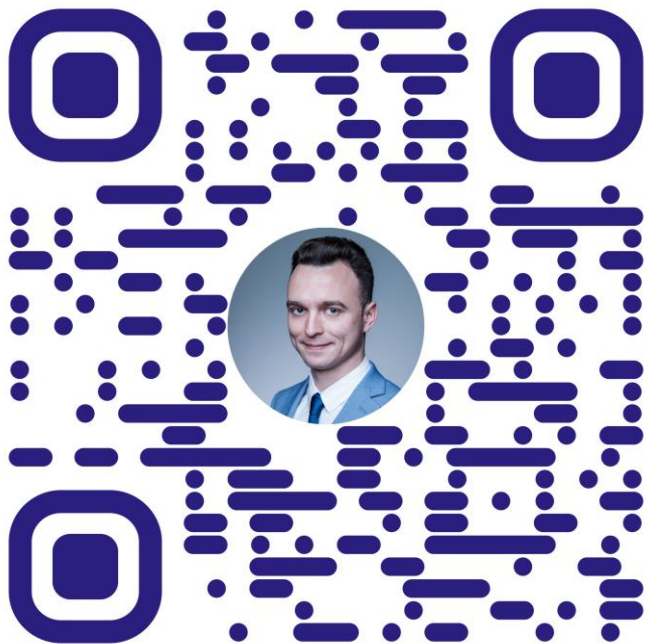
Средства платформы позволяют повысить доступность и удобство использования сертификатов.

Ролевая модель и средства администрирования позволяют отделить управление сертификатами ключей от бизнес-логики.

Дополнительные механизмы авторизации реализованы в виде программного кода в распределённом реестре.

СПАСИБО ЗА ВНИМАНИЕ!

 Добро пожаловать для вопросов и предложений!



Алексей Цветков

Руководитель разработки,
Развитие технологии
распределённых реестров

Ассоциация ФинТех

alexey.tsvetkov@fintechru.org

