Центр
Управления
Гетерогенными
Инфраструктурами

Система централизованного анализа и управления гетерогенными инфраструктурами — MiniCA

Архитектура программного обеспечения

ООО «Клируэй Текнолоджис»

Москва 2025

Оглавление

Перечень терминов и сокращений

Сокращение	Расшифровка	Описание
API	Application Programming Interface	Прикладной программный интерфейс компонента ИС.
PKI (ИОК)	Public Key Infrastructure	Инфраструктура открытых ключей (ИОК) — набор средств, распределенных служб и компонентов, используемых для поддержки крипто-задач (шифрования, аутентификации, подписи) на основе закрытого и открытого ключей.
JWT	JSON Web Token	Открытый стандарт (RFC 7519), определяющий компактный и самодостаточный способ передачи информации между сторонами в виде JSON-объекта, подписанного цифровой подписью или зашифрованного.
RSA	Rivest - Shamir - Adleman	Реализация криптосистемы на основе закрытого и открытого ключей.
TLS / mTLS	Transport Level Security / mutual Transport Level Security —	Развитие протокола SSL, криптографический протокол на основе сертификатов x509, обеспечивающий организацию безопасной сетевой связи и взаимную аутентификацию клиента и сервера.
SSL	Secure Socket Layers	Криптографический протокол, обеспечивающий организацию безопасной сетевой связи между клиентом и сервером и упрощенной проверкой подлинности сторон связи с применением сертификатов x509.
X.509v3		Версия международного стандарта, определяющая структуру цифровых сертификатов, содержащих информацию о субъекте (пользователе, устройстве или организации), открытый ключ субъекта и дополнительные расширения, используемые для настройки поведения сертификата и улучшения функциональности. Является основой современных методов аутентификации и защиты данных в интернет-коммуникациях, включая TLS/SSL и электронную почту.
ИС	Информационная система	
ОС	Операционная система	

Сокращение	Расшифровка	Описание
УЦ	Удостоверяющий Центр	Удостоверяющий Центр — программный или программно-аппаратный комплекс, обеспечивающий управление жизненным циклом сертификатов x509.
УЦ (CA)	Центр Сертификации	Центр Сертификации (Certification Authority) — сервер, предназначенный для выпуска и отзыва сертификатов, а также публикации CRL (COC). Часть Удостоверяющего Центра.
HSM	Hardware Security Module	Специализированное устройство для генерации, хранения и управления криптографическими ключами.
KRA	Key Recovery Agent	Пользователь, имеющий специальный сертификат и права для резервного копирования и восстановления криптографических ключей и цифровых сертификатов.

Определения

Термин	Определение
Микросервис	Отдельная программа, являющаяся частью программной системы или проекта, реализующая отдельный функциональный или информационный блок и тесно связанная с другими частями системы или проекта через сетевые вызовы API.
Роль	Набор системных и объектных прав, которые могут быть выданы и отозваны как единое целое, и после добавления этой Роли, могут быть временно активированы и деактивированы во время существования сессии.

1. Введение

Настоящий документ относится к эксплуатационной документации ПО "Система централизованного анализа и управления гетерогенными инфраструктурами — MiniCA" (далее — MiniCA). Разработчиком ПО MiniCA является ООО «Клируэй Текнолоджис».

MiniCA — это центр сертификации, который выполняет выпуск сертификатов X.509v3 на основе CSR-запросов в формате PKCS#10, используя готовые шаблоны. Система позволяет создавать и настраивать эти шаблоны, а также отзыв сертификатов с указанием причины. MiniCA формирует списки отозванных сертификатов (CRL и DeltaCRL), предоставляет статус сертификатов через протокол OCSP и обеспечивает доступ к ним по серийному номеру или SHA1-отпечатку. Все запросы, сертификаты, журналы событий и CRL/DeltaCRL сохраняются в базе данных. Управление осуществляется через Панель управления и API, а также поддерживается работа с протоколом SCEP для выпуска сертификатов.

ПО MiniCA является развитием одного из модулей ПО «Система централизованного анализа и управления гетерогенными инфраструктурами (ЦУГИ)», зарегистрированной в Реестре российского ПО (Реестровая запись №13033 от 21.03.2022), обладает расширенным составом функций и позволяет применяться в отдельно устанавливаемом исполнении.

2. Функции MiniCA

Основной функцией MiniCA является управление жизненным циклом сертификатов:

1) Выпуск сертификатов:
   - Выпуск сертификатов X.509v3 на основе CSR запросов в формате PKCS#10;
   - Поддержка шаблонов сертификатов для наполнения и контроля атрибутов сертификатов;
   - Поддержка выпуска сертификатов по различным протоколам – SCEP, WSTEP и т.п.
2) Предоставление сертификата по запросу:
   - Возможность получения выпущенных сертификатов по запросу;
   - Возможность получения различных выборок и отчетов о выпущенных сертификатах.
3) Отзыв сертификата:
   - Отзыв сертификатов с указанием причины;
   - Формирование Списка Отозванных Сертификатов (COC, CRL);
   - Формирование разностных списков отозванных сертификатов DeltaCRL;
   - Предоставление информации о статусе сертификата по протоколу OCSP.
4) Вспомогательные функции:
   - Сохранение информации о всех запросах, сертификатах, CRL/DeltaCRL в базе данных;
   - Текстовый журнал событий;
   - Панель управления с поддержкой ролевой модели;
   - API-интерфейс с обеспечением авторизации и разделения ролей;
   - Поддержка функций самодиагностики;
   - Поддержка хранения ключа УЦ на аппаратных носителях HSM.

3. Логическая архитектура системы

3.1. Программные компоненты системы

MiniCA реализует автономный многопоточный HTTP сервер с возможностями установления TLS или mTLS (mutual TLS) в зависимости от параметров конфигурации. Сервис эксплуатирует утилиту OpenSSL для выполнения всех криптографических операций, а также операций с различными контейнерами ASN.1 (CSR запросы PKCS#10, сертификаты X.509v3, CRL/DeltaCRL).

MiniCA построен на базе модульных компонентов, написанных на языках программирования C# (.Net 8.0.x) и Go (1.21 и выше).

Ядром системы является сервис центра сертификации на базе OpenSSL. Система реализует программный интерфейс с использованием протокола HTTP (или HTTPS).

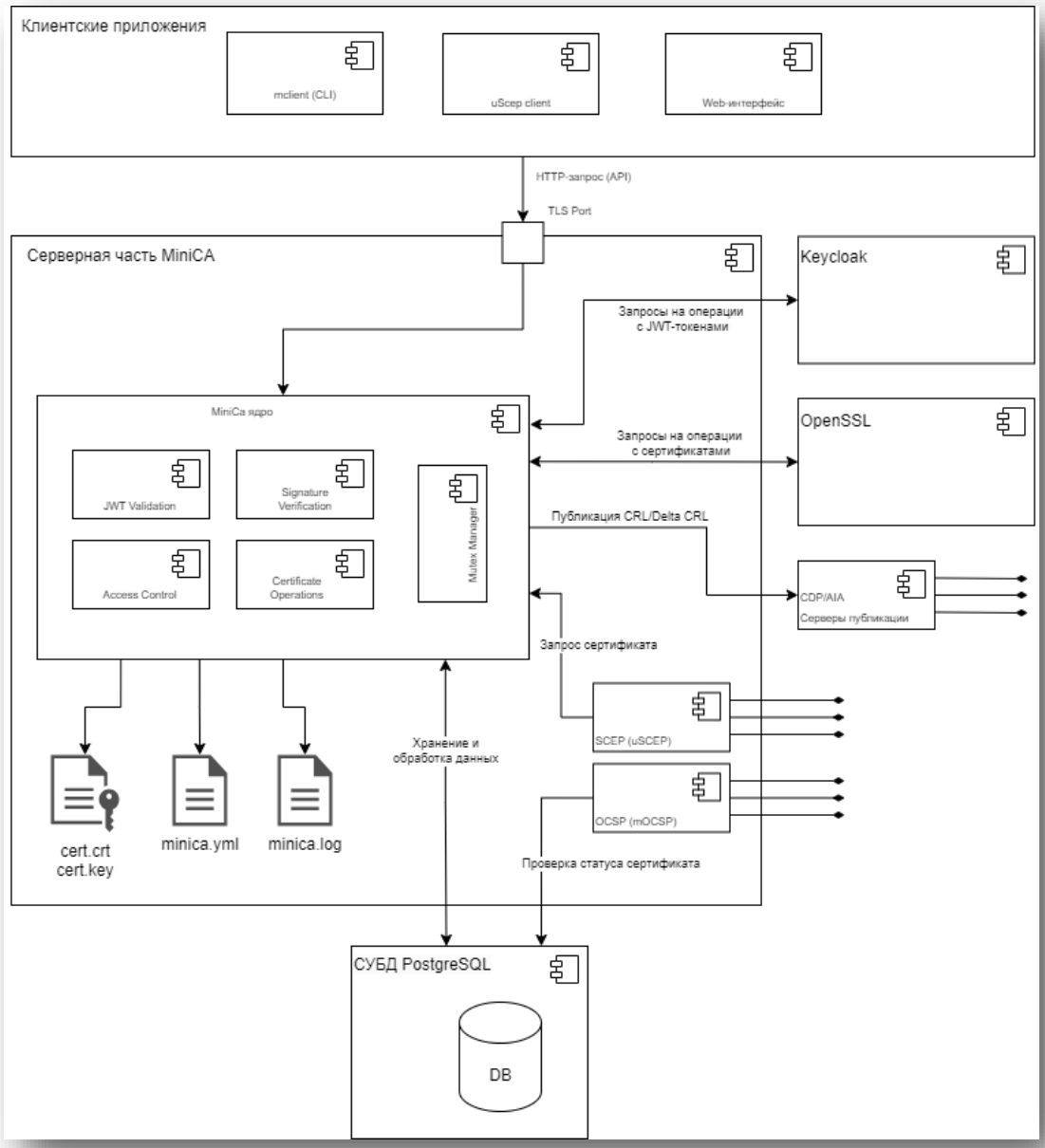

Рисунок 1 — Архитектурная схема MiniCA

Также имеется ряд дополнительных программных компонентов, которые дополняют сервис MiniCA. В таблице ниже приведено описание функций компонентов.

Таблица 1 — Компоненты MiniCA и их функции

Компонент	Тип	Необходимые сторонние компоненты	Конфигурационный файл	Функции
MiniCA	сервис (демон)	PostgreSQL, OpenSSL	MiniCA.yml	Обеспечивает выполнение всех необходимых функций по управлению жизненным циклом сертификатов.
mclient	консольное приложение		mclient.yml	Внутреннее клиентское приложение для вызова функций MiniCA с использованием интерфейса командной строки.
mOCSP	сервис (демон)		mocsp.yml	Реализация сервиса OCSP с предоставлением статуса сертификатов в реальном времени.
Веб-интерфейс (контрольная панель)	сервис (демон)	Nginx, KeyCloak	itc_Api_MiniCA_controlPanel_config.json	Обеспечение основных функций по управлению сертификатами через графический интерфейс пользователя.
uSCEP	сервис (демон)		uscep.yml	Обеспечивает интерфейс для запроса сертификатов по протоколу SCEP.

На время выполнения операций с OpenSSL применяется блокировка (мьютекс). Система может быть масштабирована использованием нескольких микросервисов MiniCA, работающих параллельно с одной базой данных.

Протокол информационного взаимодействия между сервисом и клиентом основан на передаче структур данных JSON (запрос → ответ). Все запросы к микросервису должны быть подписаны закрытым ключом, известным только клиенту. Открытый ключ для проверки подписи запросов сервис извлекает из X.509v3 сертификата, заданного конфигурацией. Опционально проверка подписи может быть отключена.

Операционная система — GNU/Linux. Целевая система Astra Linux SE 1.7 «Воронеж». Возможна эксплуатация с использованием других распространенных дистрибутивов ОС Linux (Debuian/Ubuntu/Fedora) и их производных. При необходимости микросервис может быть запущен в контейнере docker (podman).

Микросервис реализует программный интерфейс с использованием протокола HTTP (или HTTPS). Основные функции микросервиса:

- выпуск сертификатов X.509v3 на основе CSR запросов в формате PKCS#10;
- отзыв сертификатов в соответствии с заданной причиной;
- формирование списка отозванных сертификатов CRL (RFC 5280);
- формирование разностных списков отозванных сертификатов DeltaCRL;
- предоставление информации о статусе сертификата;
- предоставление сертификатов по их серийному номеру или SHA1 отпечатку;
- сохранение информации о всех запросах, сертификатах, CRL/DeltaCRL и журнала событий в базе данных.

3.2. Описание архитектуры MiniCA

Дистрибутив состоит из двух исполняемых программ:

- MiniCA — приложение микросервиса;
- mclient — клиентское приложение для вызова функций сервиса с использованием интерфейса командной строки.

Для управления конфигурацией OpenSSL используется традиционный файл конфигурации `openssl.cnf` (имя и путь могут быть заданы в конфигурационном файле MiniCA). Параметры выпускаемых сертификатов (шаблоны/атрибуты) определяются конфигурацией OpenSSL.

Для автоматизации запуска микросервиса рекомендуется применение службы systemd, входящей в большинство современных дистрибутивов Linux.

В качестве СУБД используется PostgreSQL. СУБД может быть развернута как на локальной, так и на удаленной машине. Реквизиты доступа к базе данных указываются в параметрах конфигурации.

Конфигурация микросервиса описывается в конфигурационном файле в формате JSON. Некоторые опции могут быть заданы опциями командной строки (путь к конфигурационному файлу, опции журнала). Единственная используемая переменная окружения MINICA_CONF может использоваться для указания пути в конфигурационном файл микросервиса.

Диаграмма взаимодействия программных компонентов приведена на следующем рисунке:

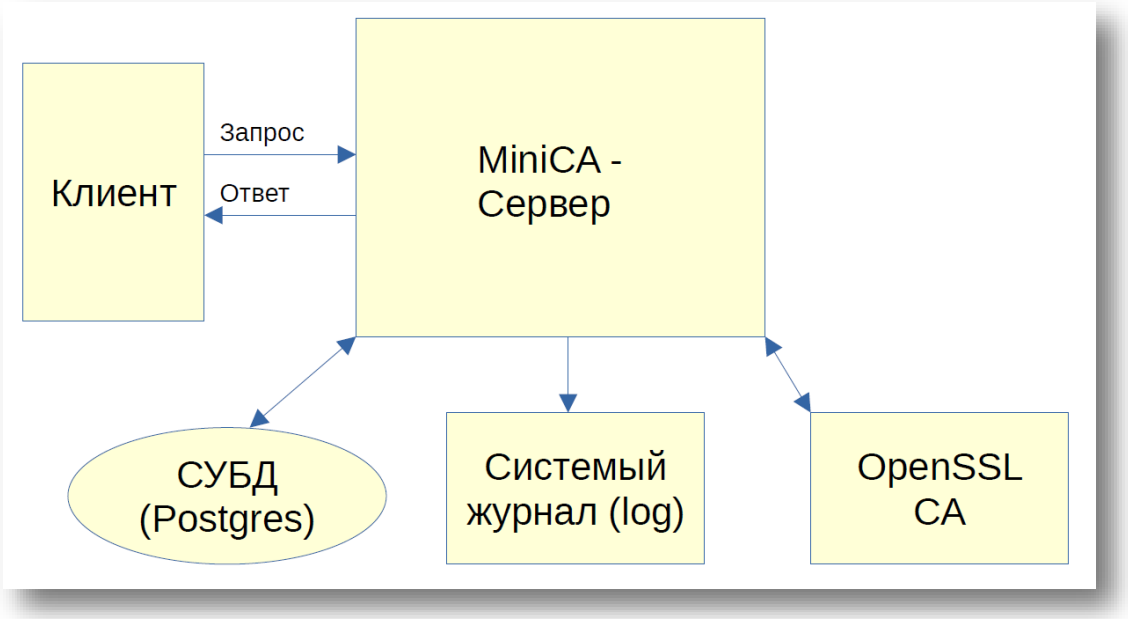

Рисунок 2 — Диаграмма взаимодействия программных компонентов

3.3. Описание архитектуры MiniCA, панель управления

Графическая панель управления MiniCA-web предназначена для доступа с помощью веб-интерфейса к функциям выдающего УЦ, а также размещения CDP, OCSP и прочих компонентов.

MiniCA-web состоит из двух пакетов и одной исполняемой программы:

- minica-cp-service— приложение микросервиса;
- minica-spa — интерфейс приложения и конфигурационные файлы.
- Микросервис также использует программу Nginx для перенаправления веб-интерфейса.

Для автоматизации запуска микросервиса рекомендуется применение службы `systemd`, входящей в большинство современных дистрибутивов Linux.

В качестве СУБД используется PostgreSQL. Реквизиты доступа к базе данных указываются в параметрах конфигурации.

Конфигурация микросервиса описывается в конфигурационном файле в формате JSON. Некоторые опции могут быть заданы опциями командной строки (путь к конфигурационному файлу, опции журнала). Диаграмма взаимодействия программных компонентов приведена на следующем рисунке:

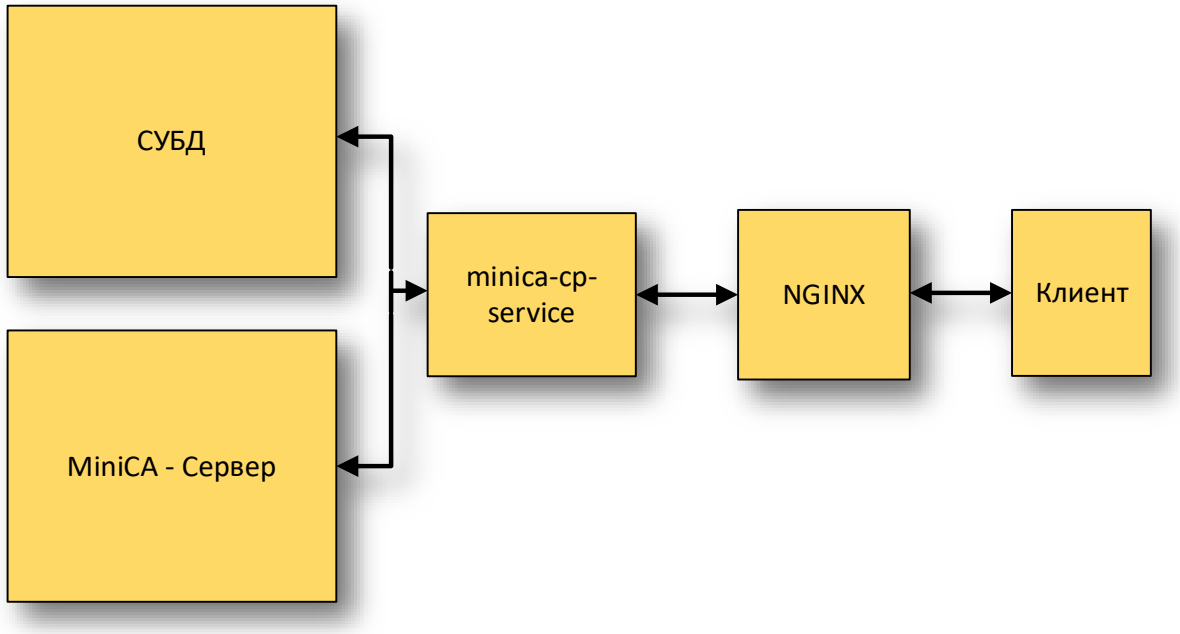

Рисунок 3 — Диаграмма взаимодействия программных компонентов MiniCA-web

3.4. Описание архитектуры mOSCP

MOCSP специализированный микросервис для быстрой онлайн-проверки текущего статуса цифровых сертификатов, используемый в инфраструктуре открытых ключей (PKI). Состоит из одного пакета и одной исполняемой программы:

- mocsp — приложение микросервиса;
- для автоматизации запуска микросервиса рекомендуется применение службы `systemd`, входящей в большинство современных дистрибутивов Linux.

В качестве СУБД используется PostgreSQL. Реквизиты доступа к базе данных указываются в параметрах конфигурации.

Конфигурация микросервиса описывается в конфигурационном файле в формате JSON. Некоторые опции могут быть заданы опциями командной строки (путь к конфигурационному

файлу, опции журнала). Диаграмма взаимодействия программных компонентов приведена на следующем рисунке:

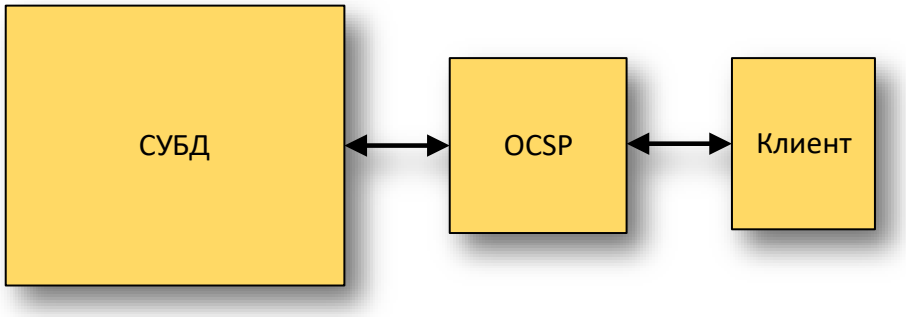

Рисунок 4 — Диаграмма взаимодействия программных компонентов MOSCP

3.5. Сторонние программные компоненты

Все программные компоненты MiniCA протестированы на следующих операционных системах:

- Astra Linux 1.7.4 и выше;
- Debian 12.

Для работы MiniCA используется база данных PostgreSQL версии 15 и выше.
Для работы веб-консоли необходим Nginx версии 1.26.0 и выше.

3.6. Алгоритм типовой операции

MiniCA работает как веб-сервис и принимает на вход HTTP-запрос. При этом выполняется ряд проверок. Ниже приведен порядок обработки типового запроса к MiniCA:

1) Клиент отправляет HTTP-запрос на микросервис MiniCA. Запрос содержит два компонента: данные в виде base64-кода и цифровую подпись этих данных.
2) Микросервис проверяет заголовок Authorization, где содержится JWT-токен. Проверяется:
    1) Алгоритм подписи токена (HS256, RS256, ES256 или EdDSA);
    2) Значения полей `iss` и `sub`:
        - `iss` содержит уникальный идентификатор ключа сервера;
        - `sub` содержит идентификатор открытого ключа клиента.

3) После успешной проверки токена, микросервис декодирует данные и цифру (`signature`) из запроса. Данные представляют собой JSON-объект, закодированный в base64.
4) Проверяется цифровая подпись данных — MiniCA вычисляет SHA-256-хеш распакованного JSON-объекта и сравнивает его с цифровой подписью, приложенной клиентом.
5) Если подпись совпадает, проходит дополнительная проверка прав доступа клиента, исходя из ролевой модели или списка разрешенных операций.
6) Производится непосредственное выполнение запрошенной операции с сертификатами: создание, обновление, аннулирование и т.п.
7) Применяется блокировка (мьютекс) для исключения возможных проблем параллельного доступа к общим данным (конфликты записи/чтения) и предотвращения ситуации, когда разные потоки одновременно пытаются изменить одни и те же ресурсы.
8) Результат выполненной операции сохраняется в базе данных PostgreSQL.
9) Формируется ответ клиенту. Ответ состоит из двух частей:
    - Основной объект данных (результат операции), закодированный в base64;
    - Новая цифровая подпись этого объекта, созданная сервером.
10) Ответ отправляется клиенту, завершая весь цикл обработки запроса.

4. Физическая архитектура

Физическая архитектура MiniCA включает в себя следующие серверные компоненты:

- Серверы с ролью УЦ — MiniCA;
- Сервер или кластер серверов базы данных (PostgreSQL);
- Серверы публикации CDP&AIA;
- Серверы OCSP;
- Серверы SCEP;
- Серверы Панели управления

Все компоненты могут размещаться на отдельных серверах или совмещаться в произвольном сочетании.

Существует несколько типовых конфигураций, рекомендуемых для развертывания в инфраструктурах организаций:

1) базовая – на одном сервере (см. Рисунок 5);

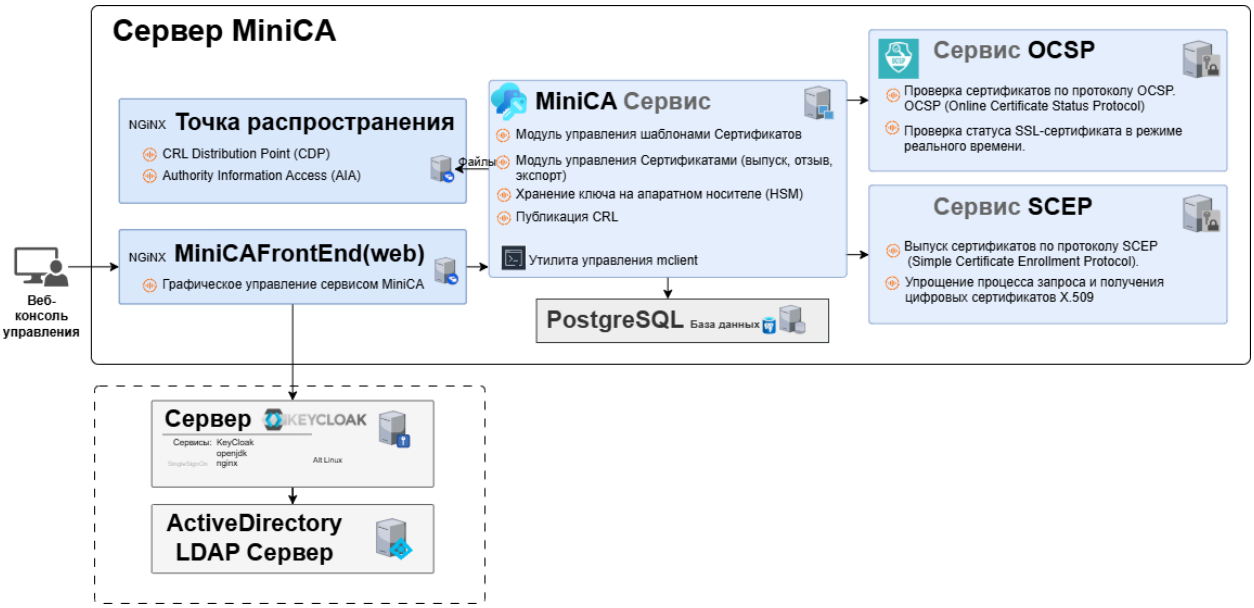

Рисунок 5 — Базовая схема физической архитектуры MiniCA на одном сервере

11) типовая – без отказоустойчивости (см. Рисунок 6);

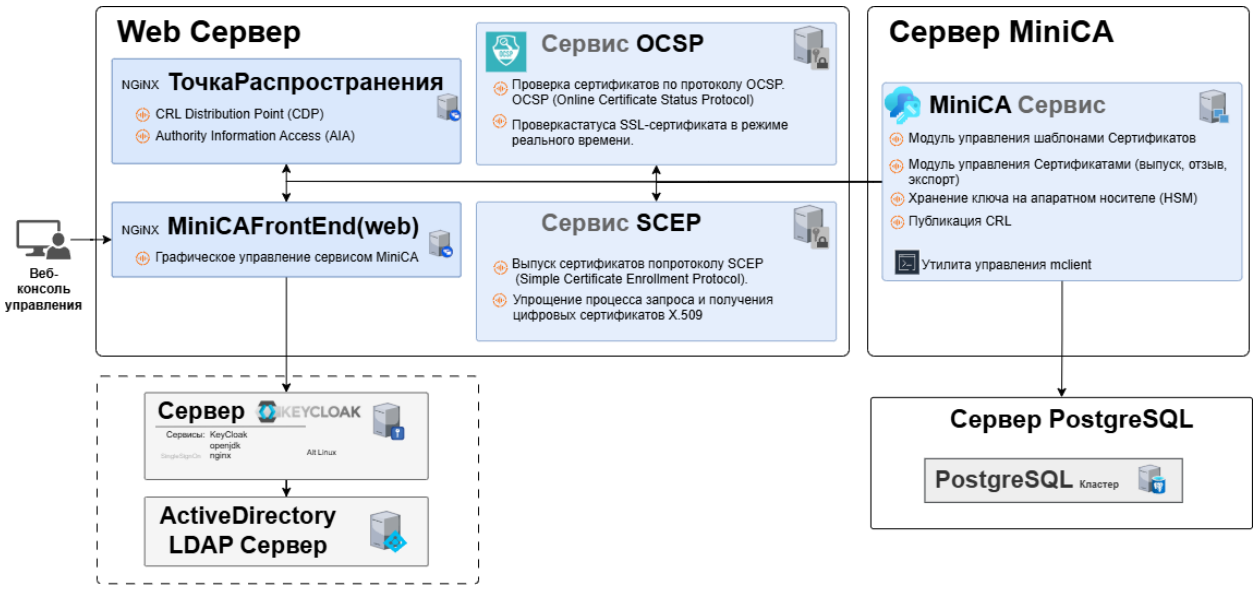

Рисунок 6 — Схема физической архитектуры MiniCA без отказоустойчивости

12) распределенная – с отказоустойчивостью (см. Рисунок 7);

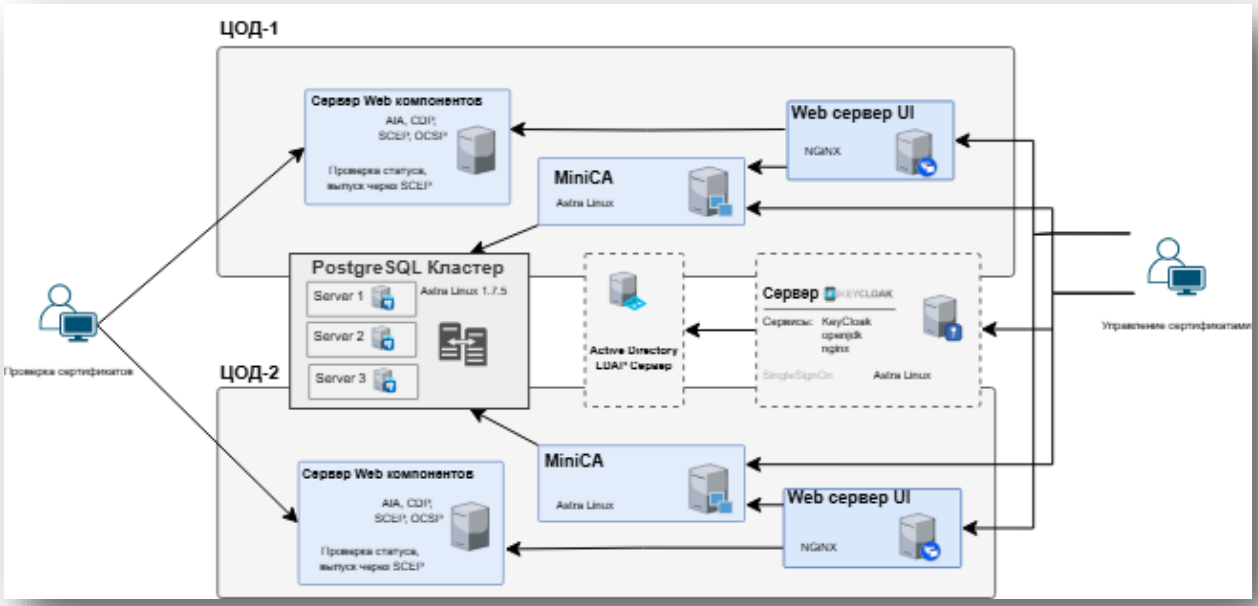

Рисунок 7 — Схема физической архитектуры MiniCA с отказоустойчивостью

Таблица 2 — Типовые конфигурации

Название	Список серверов	Рекомендация к установке
Базовая установка	Все компоненты MiniCA и СУБД PostgreSQL размещаются на одном сервере	Рекомендуется для тестовых стендов или инфраструктур с небольшой нагрузкой на УЦ без высоких требований к доступности.
Типовая, без отказоустойчивости	– Сервер MiniCA, сервис CDP&AIA, сервис OCSP, веб-сервер, сервис SCEP – Сервер PostgreSQL	Описанная конфигурация обладает высокой производительностью, но не обеспечивает отказоустойчивости. Рекомендуется для УЦ без высоких требований к доступности. Допускается размещение компонентов MiniCA как на одном сервере, так и на нескольких серверах (см. Рисунок 6)
Распределенная, с отказоустойчивостью	– 2 или более серверов MiniCA; – 2 или более серверов CDP&AIA + OCSP; – 1 или 2 сервера Веб-консоли; – 1 или более серверов SCEP (при необходимости) – Кластер PostgreSQL.	Рекомендуется для высоконагруженных УЦ, имеющих высокие требования к отказоустойчивости.

В таблице ниже приведен список межкомпонентных взаимодействий и требуемых портов.

Таблица 3 — Информационные потоки MiniCA

№	Инициатор	Получатель	Протокол	Порт по умолчанию	Описание
1)	Сервер MiniCA	Сервер PostgreSQL	TCP	5432	Чтение и запись данных
2)	Веб сервер точек распространения	Сервер MiniCA	HTTPS	443	Получение CRL
3)	SCEP	Сервер MiniCA	HTTPS	443	Запрос сертификатов
4)	Панель управления	Сервер MiniCA	HTTPS	443	Запрос сертификатов, настройка УЦ и шаблонов
5)	Панель управления	Сервер PostgreSQL	TCP	5432	Получение списка сертификатов
6)	OCSP	Сервер PostgreSQL	TCP	5432	Чтение данных о сертификатах

№	Инициатор	Получатель	Протокол	Порт по умолчанию	Описание
7)	Драйвер УЦ	Сервер MiniCA	HTTPS	443	Запросы на выпуск сертификатов
8)	Все потребители	Веб сервер точек распространения и OCSP	HTTP	80	Доступ пользователей к AIA, CDP, OCSP
9)	Клиенты SCEP	SCEP	HTTPS	443	Запрос сертификатов
10)	АРМ администратора	Все серверы компонентов MiniCA	SSH	22	Доступ администраторов для настройки и аудита

4.1. Требования к комплексу технических средств MiniCA

4.1.1. Рекомендуемые аппаратные требования к серверам (виртуальным или физическим)

Таблица 4 — аппаратные требования к минимальной конфигурации

№	Наименование ТС	Кол-во серверов	CPU, кол-во ядер	CPU, тактовая частота, не менее, ГГЦ	RAM, Гб	SSD, Гб	ОС	Описание
1)	Сервер MiniCA + СУБД PostgreSQL	1	4	2	4	256	Astra Linux 1.7.5 Orel	Включает роли PostgreSQL, Панель управления, AIA, CDP и OCSP. Рекомендации по применению: корневой УЦ, выдающий УЦ для небольших организаций (до 100 сертификатов в сутки).

Таблица 5 — рекомендуемые аппаратные требования к базовой конфигурации

№	Наименование ТС	Кол-во серверов	CPU, кол-во ядер	CPU, тактовая частота, не менее, ГГЦ	RAM, Гб	SSD, Гб	ОС	Описание
2)	Сервер MiniCA + СУБД PostgreSQL + сервер панели управления	1	4	2	8	512	Astra Linux 1.7.5 Orel	Включает роли PostgreSQL, панель управления, AIA, CDP и OCSP. Рекомендации по применению: выдающий УЦ для организаций со средней нагрузкой (до 5000 сертификатов в сутки).

Таблица 6 — рекомендуемые аппаратные требования к типовой конфигурации

№	Наименование ТС	Кол-во серве-ров[1]	CPU, кол-во ядер	CPU, тактовая частота, не менее, ГГЦ	RAM, Гб	SSD, Гб	ОС	Описание
1)	Сервер MiniCA	1 или 2	8	2	16	256	Astra Linux 1.7.5 Orel	Основной сервер MiniCA. Рекомендации по применению: выдающий УЦ, для высоконагруженных систем, с использованием K8S, service mesh
2)	Сервер СУБД PostgreSQL	1 или кластер СУБД	4	2	8	512	Astra Linux 1.7.5 Orel	Сервер PostgreSQL
3)	Сервер панели управления	1 или 2	2	2	4	100	Astra Linux 1.7.5 Orel	Панель управления, AIA, CDP и OCSP

[1] 2 сервера используются для распределения нагрузки и обеспечения отказоустойчивой работы компонентов MiniCA

4.1.2. Требования к АРМ оператора

Таблица 7 — Требования к АРМ оператора

№	Требования к ПК для рабочих мест	Ядер	Тактовая частота не менее,ГГЦ	RAM, ГБ	SSD, ГБ	Браузер	Операционная система
1)	АРМ оператора	2	2	4	80	Google Chrome версия не ниже 113, браузеры на Chromium (Microsoft Edge, Яндекс.Браузер).	Windows 10+

4.1.3. Требования к системному ПО

Таблица 8 — Требования к системному ПО

Сервер / АРМ, на который устанавливается ПО	Наименование ПО	Минимальная версия	Рекомендуемая версия	Назначение
Сервер панели управления	dot.Net	8.0	8.0	Компонент системы
	nginx	Установка из репозитория		Веб сервер
	curl	Установка из репозитория		Обращение к API системы
	jq	Установка из репозитория		Для скрипта формирования токена для обращения к API
Сервер MiniCA	psql	Установка из репозитория	Не ниже версии БД	Для создания БД и таблиц
Keycloak сервер	KeyCloak	19.0.3	24.0.2 и выше	Авторизация
	openjdk	11 (зависит от версии Keycloak)	17 (зависит от версии Keycloak)	Для работы keycloak

Сервер / АРМ, на который устанавливается ПО	Наименование ПО	Минимальная версия	Рекомендуемая версия	Назначение
	nginx	Установка из репозитория		Веб сервер
Сервер СУБД	PostgreSQL	11	15	База данных
	etcd	Установка из репозитория		Обеспечение работы кластера PostgreSQL
	patroni	Установка из репозитория		Обеспечение работы кластера PostgreSQL
АРМ администратора	Dbeaver +pg utils(pg_dump, pg_dumpall,pg_restore,psql) +driver PostgeSQL	21.3.1	Самая последняя	Для настройки и эксплуатации БД
	WinSCP	5.13.7	Самая последняя	Доступ к файлам серверов
	putty	0.70	Самая последняя	Доступ к серверам по ssh
	liquibase	4.7.1	Самая последняя	Первичное наполнение/обновление БД
	Google Chrome	113	Самая последняя	Доступ к веб интерфейсу системы

4.1.4. Требование к сети

Источник	Назначение	Порты	Описание
АРМы (операторов, администраторов, пользователей и т.д)	MiniCA сервер, сервер панели управления, Keycloak сервер	443	Доступ к Веб-консоли Системы

Источник	Назначение	Порты	Описание
Сервер панель управления, Keycloak сервер	Контроллеры домена	389,636	Доступ к AD для чтения пользователей, авторизации
АРМ администратора	Сервер панели управления, Keycloak сервер	22,443	Доступ для настройки и сопровождения
АРМ администратора	Сервер БД	5432 (или порты, выделенные для данного подключения)	Доступ для настройки и сопровождения

4.1.5. Требования к учетным записям

Название	Расположение	Права	Описание
Itc-mca-svc	Домен	Active Directory: чтение Центры сертификации Microsoft: чтение УЦ, запуск в качестве задания	Доступ к Active Directory на чтение (обычная УЗ без каких-либо прав) Чтение сертификатов для импорта в MiniCA
Itc-mca-mail	Доменная (или иная – зависит от почтовой системы)	Отрывка почтовых уведомлений	УЗ с правом отправлять почтовые уведомления
Itc-svc	Локальная MiniCA сервер	Запуск служб Домашняя папка Запуск crontab Папка /app (RW)	Локальная УЗ для запуска служб системы
keycloak	Локальная Keycloak сервер	Запуск служб Домашняя папка Запуск crontab Папка /app (RW)	Локальная УЗ для запуска службы Keycloak

Название	Расположение	Права	Описание
Itc-svc	Локальная БД сервер	БД owner: MiniCA	Локальная УЗ для доступа к БД
УЗ администратора	Доменная	АРМ администратора: RDP MiniCA сервер: ssh, root или переключение в контекст Itc-svc, nginx(www-data) Keycloak сервер: ssh, root или переключение в контекст keycloak	УЗ для доступа к серверам

4.1.6. Группы

Имя	Тип	Описание
ITC_MCA_Administrators	Доменная	Администраторы инфраструктуры
ITC_MCA_Operators	Доменная	Операторы
ITC_MCA_Auditors	Доменная	Аудиторы

4.1.7. Дополнительно

Для нормальной работы сертификатов, выданных MiniCA, необходимо внести сертификат MiniCA в контейнеры TrustedRoot на серверах и APM.

Предпочтительны права Root на серверах для развертывания системы.

5. Обеспечение безопасности

5.1. Защита ключа УЦ

В таблице ниже описаны варианты хранения ключа УЦ и меры по обеспечению его безопасности.

Таблица 9 — Варианты хранения ключа УЦ и меры по обеспечению его безопасности

Способ хранения	Методы защиты	Примечание
При помощи аппаратных устройств HSM	Ключ внутри HSM не может быть извлечен, поэтому защита сводится к ограничению доступа пользователей к интерфейсу HSM.	
На файловой системе	– Разрешение на чтение ключа имеет только технологическая учетная запись, от которой запущен сервис MiniCA; – Шифрование файла ключа.	В случае шифрования файла ключа при запуске MiniCA должен быть введен ключ шифрования.

5.2. Защита файлов системы

Исполняемые, конфигурационные и вспомогательные файлы компонентов MiniCA защищены при помощи разрешений операционной системы:

- Владельцем файлов и папок является технологическая учетная запись, от которой запускается MiniCA;
- На файлы и папки установлены разрешения 700.

5.3. Защита БД

Для защиты базы данных MiniCA должны быть предприняты следующие меры:

1) Владельцем БД MiniCA является учетная запись, от имени которой происходит взаимодействие MiniCA и PostgreSQL;
2) Для чтения БД выделяется отдельная учетная запись с доступом только для чтения;
3) В случае хранения в БД чувствительной информации (например, при использовании KRA), она хранится в зашифрованном виде.

5.4. Защита сетевых взаимодействий

Все взаимодействия между компонентами системы, а также между компонентами и пользователями, защищены при помощи TLS. Также возможна настройка mTLS.

5.5. Ролевая модель

Контроль доступа пользователей к функциям УЦ является важнейшим элементом безопасности всей информационной инфраструктуры организации. MiniCA имеет продвинутую систему разделения полномочий пользователей на разных уровнях.

В разделах ниже описана реализация ролевой модели для MiniCA и Веб-консоли управления.

5.5.1. Ролевая модель, Панель управления

Аутентификация и авторизация в графическом Веб-консоли осуществляется при помощи системы Keycloak и службы каталогов, в частности, Microsoft Active Directory.

Keycloak — это система идентификации и управления доступом, которая позволяет управлять идентификацией пользователей, контролировать доступ к приложениям и данным, обеспечивая единую точку входа.

Роли пользователя присваиваются группам AD в консоли администрирования Keycloak. Для получения соответствующей роли учетная запись пользователя должна быть добавлена в ролевую группу AD. В составе токена аутентификации Панель управления получает роль пользователя и предоставляет разрешенные в соответствии с ролью элементы управления.

Реализованы следующие роли пользователей:

- Администратор УЦ;
- Менеджер сертификатов;
- Аудитор;
- Пользователь.

Роль "Пользователь" предоставляется всем пользователям по умолчанию и может быть совмещена с другими ролями. В таблице указана доступность функций системы для каждой роли.

Таблица 10 — Роли MiniCA

Наименование роли	Описание доступных действий и полномочий
Администратор	– Изменение настроек и параметров УЦ;

Наименование роли	Описание доступных действий и полномочий
	– Настройка шаблонов.
Аудитор	– Просмотр журналов событий, отчётов. – Просмотр конфигурации ИС.
Менеджер сертификатов	– Отзыв сертификатов; – Публикация CRL.
Пользователь	– Доступ на чтение к информации CDP&AIA; – Формирование запросов на сертификаты.

5.5.2. Ролевая модель сервиса MiniCA

Ниже описана реализация ролевой модели MiniCA на основе подписи запросов и JWT-токенов.

Она построена на проверке и обработке JSON Web Token (JWT), обеспечивающих управление доступом и контроль над функциональностью клиентского приложения. Конфигурируется блок JWT, включающий следующие параметры:

Таблица 11 — Параметры блока JWT

Параметр	Описание	Пример
`no-check`	Логический переключатель, управляющий проверкой JWT-токенов. Если установлен в true, проверка токенов отключается, однако остается доступна проверка подписанных запросов главным ключом.	no-check: false
`key`	Файл с приватным ключом для подписи JWT-токенов, расположенный в папке conf. Формат файла — PEM.	key: "conf/jwt.key"
`keys`	Каталог с файлами ключей для проверки JWT-токенов. Названия файлов формируются по стандарту SKI (Subject Key Identifier) и имеют формат XX...XX.PEM.	keys: "conf/jwt-keys"
`cache-minutes`	Время хранения ключей в кэше (в минутах).	cache-minutes: 5

Параметр	Описание	Пример
`roles`	Перечень ролевых моделей и назначаемых им прав. Каждый профиль имеет свою область ответственности и функциональные возможности.	Подробнее в таблице Таблица 12 — Описание ролей JWT.
`sign-off`	Логическая опция отключения проверки подписи запросов. При значении true проверка подписей также отключается.	sign-off: false
`master-key`	Файл с главным ключом проверки подписи запросов. Обычно является открытым ключом или HMAC-секретом.	master-key: "conf/mclient.pem"
`chain-file`	Файл с цепочкой сертификатов (PEM-формат).	chain-file: "ca/ca-chain.pem"

Таблица 12 — Описание ролей JWT

Роль	Наименование	Описание	Права
admin	Администратор	Возможность просмотра статуса системы, выдачи сертификатов, обновления списков отмены, восстановления повреждённых записей и управления JWT-токенами.	– ping; – ca; – revoke; – status; – get; – updcrl; – crl; – updeltacrl; – deltacrl; – find; – delete; – export; – templates; – csr; – repair; – stat; – gettempl; – addtempl; – deltempl;

Роль	Наименование	Описание	Права
			– mkjwt; – revokejwt; – repairjwt; – deljwt; – getjwt; – renewjwt; – revivejwt; – chain.
causer	Центр регистрации	Доступ к функционалу центра сертификации (выпуск, отзыв, статус, восстановление сертификатов).	– ping; – ca; – revoke; – status; – get; – repair; – csr; – temlates; – gettempl; – chain.
crluser	Пользователь списка аннулирования	Обновление и получение списков отзыва сертификатов (CRL и DeltaCRL).	– ping; – updcrl; – crl; – updeltacrl; – deltacrl; – chain.
archuser	Архивариус	Удаление ненужных записей из базы данных.	– ping; – status; – get; – crl; – deltacrl; – find; – delete; – export; – templates; – stat; – gettempl; – csr;

Роль	Наименование	Описание	Права
			– getjwt.
exporter	Экспортёр	Экспорт данных из хранилища.	– ping; – status; – get; – crl; – deltacrl; – find; – export; – templates; – stat; – gettempl; – csr; – getjwt.
templadmin	Администратор шаблонов	Управление шаблонами сертификата.	– ping; – templates; – gettempl; – addtenpl; – deltempl.
jwtadmin	Администратор JWT	Полный доступ к управлению JWT-токенами (выдача, отзыв, восстановление, удаление).	– ping; – mkjwt; – revokejwt; – repairjwt; – getjwt; – deljwt; – revivejwt.

В таблице ниже описаны операции MiniCA.

Каждой операции соответствует отдельная команда сервиса MiniCA.

Таблица 13 — Описание операций

Операция	Пояснение
`ping`	Проверка доступности службы
`ca`	Управление центром сертификации
`revoke`	Отзыв выданных сертификатов
`status`	Мониторинг текущего состояния системы

Операция	Пояснение
`get`	Получение данных
`updcrl`	Обновление основных списков отозванных сертификатов
`crl`	Получение списков отозванных сертификатов
`updeltacrl`	Обновление инкрементальных списков отозванных сертификатов
`deltacrl`	Получение частичных списков отозванных сертификатов
`find`	Поиск данных в системе
`delete`	Удаление ресурсов
`export`	Экспорт информации
`templates`	Управление шаблонами сертификатов
`csr`	Обработка запросов на сертификаты (CSR-запросы)
`repair`	Исправление ошибок в системе
`stat`	Получение статистики
`gettempl`	Получение существующих шаблонов сертификатов.
`addtempl`	Добавление новых шаблонов сертификатов.
`deltempl`	Удаление шаблонов сертификатов.
`mkjwt`	Создание новых JWT-токенов
`revokejwt`	Отзыв действующих JWT-токенов
`repairjwt`	Восстановление JWT-токенов
`deljwt`	Принудительное удаление JWT-токенов
`getjwt`	Получение сведений о JWT-токене
`renewjwt`	Продление срока действия JWT-токена
`revivejwt`	Возобновление временного блокировки JWT-токена
`chain`	Управление цепочками сертификатов