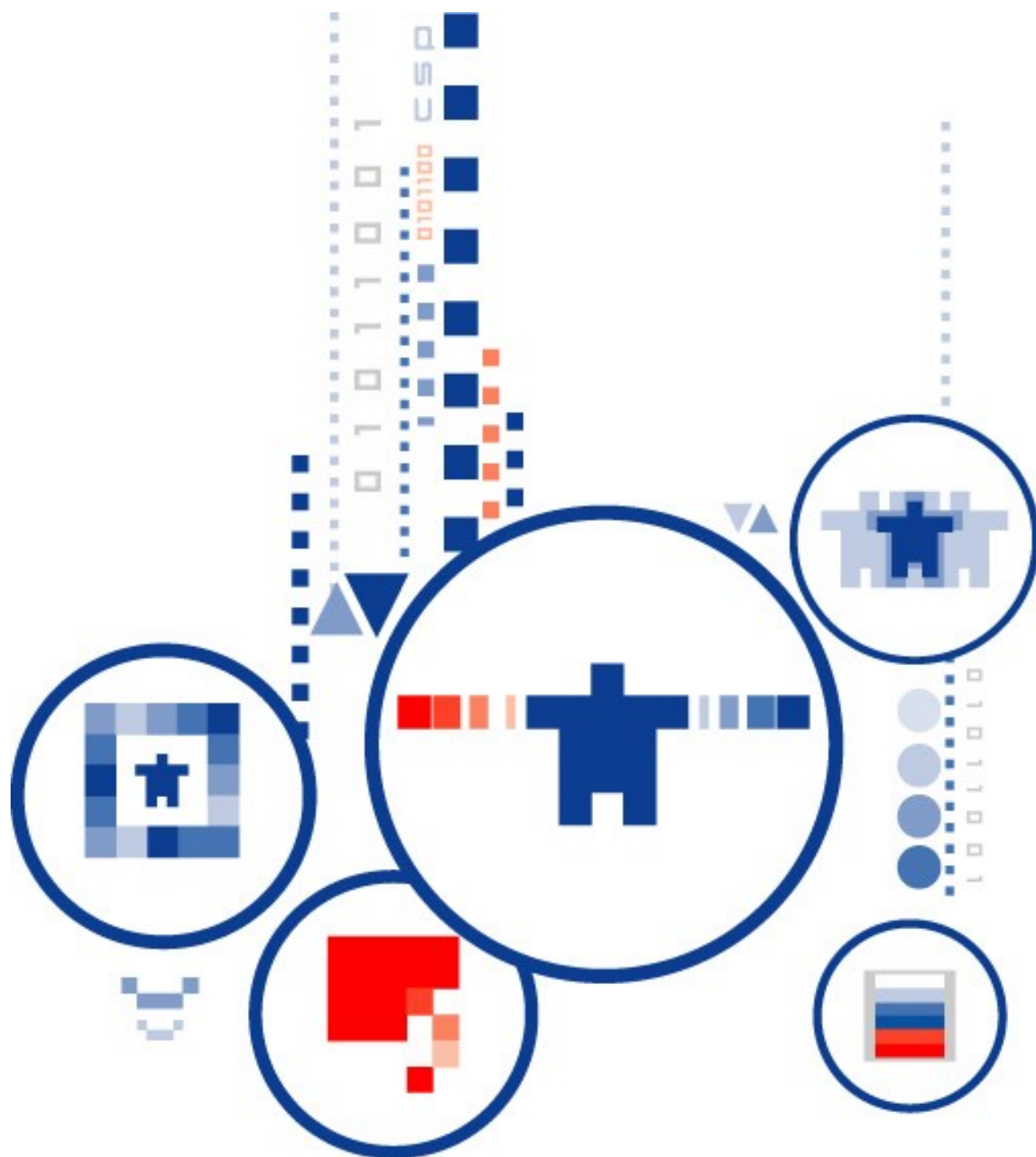




КриптоПро **УЦ**

программно-аппаратный комплекс
удостоверяющий центр

Руководство по установке

Оглавление

Термины и определения	4
Перечень сокращений	6
1 Общий порядок выполнения работ по разворачиванию УЦ.....	7
2 Подготовка ОС	9
2.1 Установка ОС	9
2.1.1 Организационно-технические меры при разворачивании КриптоПро УЦ в ОС CH Astra Linux SE.....	9
2.2 Установка КриптоПро CSP	9
2.3 Установка СУБД.....	9
2.4 Установка Программы настройки УЦ (pkica)	9
2.5 Автоматический запуск сервисов УЦ	10
2.6 Установка CRL в хранилище	10
3 Настройка сервера ЦС.....	12
3.1 Установка ПО ЦС.....	12
3.2 Настройка подключения к БД.....	12
3.3 Установка и настройка службы очередей NATS Streaming	13
3.4 Настройка подключения к NATS Streaming без использования TLS	13
3.5 Ввод лицензии ЦС	14
3.6 Создание базы данных ЦС.....	14
3.7 Запуск сервиса ЦС	14
3.8 Создание экземпляра ЦС.....	14
3.9 Настройка расписания выпуска CRL.....	14
3.10 Настройка шаблонов сертификатов.....	15
3.11 Прочие настройки ЦС	17
3.12 Создание ключа и сертификата ЦС	17
3.12.1 Создание ключа и сертификата корневого ЦС	17
3.12.2 Создание ключа и сертификата подчиненного ЦС.....	18
3.12.3 Установка сертификата ЦС	20
3.13 Добавление ключа подписи ЦС.....	21
3.14 Загрузка ключа подписи ЦС	22
3.15 Выпуск и установка служебных TLS сертификатов	22
3.15.1 Создание запросов.....	23
3.15.2 Выпуск служебных сертификатов.....	23
3.15.3 Установка служебных сертификатов	23
3.16 Выпуск и установка CRL.....	23
3.16.1 Выпуск CRL	24
3.16.2 Установка CRL в хранилище.....	24
3.17 Настройка подключения к NATS Streaming с использованием TLS.....	24
3.18 Восстановление работоспособности при пропуске выпуска очередного CRL.....	25
4 Настройка сервера ЦР	26
4.1 Подготовка сервисных сертификатов	26
4.2 Установка ПО ЦР	26
4.3 Настройка подключения к БД	26
4.4 Настройка подключения к NATS Streaming	27
4.5 Настройка публикации сертификатов подписи в AIA.....	27
4.6 Настройка публикации выпущенных CRL в CDP.....	28
4.7 Ввод лицензии ЦР	28

4.8	Создание базы данных ЦР	28
4.9	Развертывание веб-сервера	28
4.9.1	Подготовка служебных ключей и сертификатов	28
4.9.2	Развертывание веб-сервера <i>nginx</i> в замкнутой программной среде (ЗПС) ОС <i>Astra Linux SE</i>	29
4.9.3	Развертывание веб-сервера <i>IIS</i> в ОС <i>Microsoft Windows</i>	29
4.9.4	Конфигурация <i>IIS</i> для Центра регистрации	29
4.10	Настройка папок и создание операторов	30
4.10.1	Создание папок Центра регистрации	30
4.10.2	Создание групп Центра регистрации	30
4.10.3	Конфигурация <i>nginx</i> для Центра регистрации	31
4.10.4	Создание операторов Центра регистрации	32
4.10.5	Назначение разрешений на папку	33
4.11	Добавление шаблонов сертификатов	34
4.12	Настройка печатных форм	34
5	Настройка рабочего места Оператора	36
5.1	Подготовка рабочего места Оператора	36
5.2	Подключение к веб-интерфейсу ЦР	36
6	Проверка работоспособности	37
7	Настройка сервера Реестра сертификатов	38
7.1	Установка ПО Реестра сертификатов	38
7.2	Настройка подключения к БД	38
7.3	Настройка подключения к NATS Streaming	38
7.4	Настройка публикации сертификатов подписи в AIA	39
7.5	Настройка публикации выпущенных CRL в CDP	39
7.6	Создание базы данных Реестра сертификатов	39
7.7	Развертывание и конфигурация веб-сервера	39

ТЕРМИНЫ И ОПРЕДЕЛЕНИЯ

Используемые в настоящем документе термины описаны в таблице ниже.

Таблица 1. Список терминов

Термин	Определение
Владелец сертификата ключа проверки электронной подписи	Лицо, которому в установленном Федеральным законом (N 63-ФЗ) порядке выдан сертификат ключа проверки электронной подписи.
Демилитаризованная зона	Технология обеспечения защиты информационного периметра сети предприятия, при которой серверы, отвечающие на запросы из внешней сети, находятся в особом сегменте сети, называемом демилитаризованной зоной (ДМЗ). Компьютеры в ДМЗ ограничены в доступе к основным сегментам сети предприятия с помощью межсетевого экрана (МЭ) с целью минимизировать ущерб при взломе одного из общедоступных сервисов, находящихся в ДМЗ.
Межсетевой экран	Комплекс аппаратных или программных средств, осуществляющий контроль и фильтрацию проходящих через него сетевых пакетов в соответствии с заданными правилами.
Сертификат ключа проверки электронной подписи	Электронный документ или документ на бумажном носителе, выданные удостоверяющим центром либо доверенным лицом удостоверяющего центра и подтверждающие принадлежность ключа проверки электронной подписи владельцу сертификата ключа проверки электронной подписи.
Квалифицированный сертификат ключа проверки электронной подписи (квалифицированный сертификат)	Сертификат ключа проверки электронной подписи, выданный аккредитованным удостоверяющим центром или доверенным лицом аккредитованного удостоверяющего центра либо федеральным органом исполнительной власти, уполномоченным в сфере использования электронной подписи.
Средства электронной подписи	Шифровальные (криптографические) средства, используемые для реализации хотя бы одной из следующих функций - создание электронной подписи, проверка электронной подписи, создание ключа электронной подписи и ключа проверки электронной подписи.
Список сертификатов, досрочно прекративших действие	Электронный документ с электронной подписью уполномоченного лица удостоверяющего центра, включающий в себя список серийных номеров сертификатов, действие которых было прекращено до окончания срока их действия. По тексту далее также может использоваться термин "список отозванных сертификатов".
Удостоверяющий центр	Юридическое лицо или индивидуальный предприниматель, осуществляющие функции по созданию и выдаче сертификатов ключей проверки электронных подписей, а также иные функции, предусмотренные настоящим Федеральным законом (N 63-ФЗ).

Электронная подпись	Информация в электронной форме, которая присоединена к другой информации в электронной форме (подписываемой информации) или иным образом связана с такой информацией и которая используется для определения лица, подписывающего информацию.
Электронный документ	Документ, информация в котором представлена в электронной форме, способной быть обработанной средствами вычислительной техники.
Представитель юридического лица	Указываемое наряду с наименованием юридического лица, физическое лицо, действующее от имени юридического лица без доверенности в соответствии с гражданским законодательством Российской Федерации.

ПЕРЕЧЕНЬ СОКРАЩЕНИЙ

Устойчивые русскоязычные сокращения, используемые в этом документе, описаны в таблице ниже.

Таблица 2. Список сокращений

Сокращение	Значение
АРМ	Автоматизированное рабочее место
ДМЗ	Демилитаризованная зона
МЭ	Межсетевой экран
НСД	Несанкционированный доступ
ОС	Операционная система
ПАК	Программно-аппаратный комплекс
ПО	Программное обеспечение
СКЗИ	Средство криптографической защиты информации
СКПЭП	Сертификат ключа проверки электронной подписи
СУБД	Системы управления базами данных
УЦ	Удостоверяющий центр
ЦС	Центр сертификации
ЦР	Центр регистрации
ЭП	Электронная подпись

Используемые иностранные сокращения, как правило, англоязычные, приведены в таблице ниже.

Таблица 3. Список иностранных сокращений

Сокращение	Значение	Перевод
AIA	Authority Info Access	Доступ к информации о Центре сертификации
CRL	Certificate Revocation List	Список отозванных сертификатов, СОС
CDP	CRL Distribution Point	Пункт распространения CRL
DN	Distinguished Name	Отличительное имя
IETF	Internet Engineering Task Force	Специальная комиссия интернет разработок
ITU	International Telecommunication Union	Международный союз электросвязи, МСЭ
ITU-T	Telecommunication Standardization Sector (ITU's)	Сектор стандартизации электросвязи (в МСЭ), МСЭ-Т
LDAP	Lightweight Directory Access Protocol	Облегченный протокол доступа к Справочнику
RDN	Relative Distinguished Name	Относительное отличительное имя

1 Общий порядок выполнения работ по разворачиванию УЦ

Разворачивание УЦ выполняется с помощью командной строки. Разворачивание с использованием командной строки имеет ряд преимуществ. Например, оно позволяет выполнить более гибкую настройку компонентов УЦ. Также можно оформить последовательность команд в виде файла скрипта и впоследствии запускать его на выполнение.

Вне зависимости от выбранной логической схемы размещения компонентов УЦ для разворачивания УЦ потребуется:

- сервер для компонента «ПАК КриптоПро УЦ 2.0. Сервер ЦС»;
- сервер для компонента «ПАК КриптоПро УЦ 2.0. Сервер ЦР»;
- ПЭВМ для «ПАК КриптоПро УЦ 2.0. АРМ оператора ЦР» (рабочее место оператора).

Для взаимодействия компонентов УЦ между собой посредством криптографического протокола TLS, а также для подключения операторов ЦР потребуются сертификаты безопасности (технологические сертификаты). Для создания сертификатов безопасности необходимо использовать другие экземпляры ЦС и ЦР, отличные от экземпляров, предназначенных для выпуска квалифицированных сертификатов ключей проверки ЭП. Экземпляр ЦС (ЦР) для создания сертификатов безопасности может размещаться на технических средствах с установленным ЦС (ЦР), предназначенным для выпуска квалифицированных сертификатов ключей проверки ЭП.

Разворачивание технологических экземпляров ЦС и ЦР для выпуска сертификатов безопасности выполняется по аналогии с разворачиванием экземпляров для выпуска квалифицированных сертификатов ключей проверки ЭП. В данном Руководстве, для простоты изложения информации, разворот технологических экземпляров ЦС и ЦР отдельно не описывается.

Предполагается, что сервера подготовлены согласно разделу 2 данного документа, в зависимости от используемой ОС. В процессе настройки можно ввести купленные лицензии на ПО, либо воспользоваться временными.

1. На серверах, предназначенных для развертывания ЦС и ЦР, установить соответствующие модули.
2. Выполнить настройку сервера ЦС (раздел 3). В процессе настройки будет выпущен сертификат для работы службы сервера ЦР, который нужно сохранить и использовать при выполнении настройки сервера ЦР.
3. Выполнить настройку сервера ЦР (раздел 4).
4. Выполнить настройку рабочего места Оператора (раздел 5).
5. Провести проверку работоспособности развернутого комплекса (раздел 6).

Дополнительные компоненты и модули настраиваются в соответствии с настоящей документацией. Дальнейшая настройка прав и ролей пользователей производится согласно «ЖТЯИ.00078-03 90 01 ПАК КриптоПро УЦ 2.0. Общее описание» и «ЖТЯИ.00078-03 90 03 ПАК КриптоПро УЦ 2.0. Руководство по эксплуатации».

Работа с данными в БД компонентов УЦ допускается только с использованием механизмов, приведенных в документации на УЦ. Прямое изменение данных в БД компонентов УЦ, а также изменение структуры БД запрещено.

В данном Руководстве, для удобства изложения информации, используются пути файловой системы в стиле ОС Astra Linux. Для другой ОС пути файловой системы должны быть записаны в соответствующем стиле.

В данном Руководстве предполагается использование следующих папок:

- `/opt/cpca` – папка установки УЦ;
 - `/opt/cpca/CryptoPro.Ca.Service` – сервис ЦС;
 - `/opt/cpca/CryptoPro.Ra.Service` – сервис ЦР;

- `/opt/cpca/CryptoPro.Ra.Web` – веб-приложение ЦР (API);
- `/opt/cpca/CryptoPro.Ra.Web.Ui` – пользовательский интерфейс (UI) ЦР;
- `/opt/cpca/CryptoPro.CertRegistry.Service` – сервис Реестра сертификатов;
- `/opt/cpca/CryptoPro.CertRegistry.Web` – веб-приложение Реестра сертификатов (API);
- `/opt/cpca/nats-streaming` – служба очередей;
- `/var/log/cpca` – папка для хранения логов УЦ;
- `/var/lib/nats-streaming/pkica-store` – папка для хранения очередей;
- `/var/lib/cpca/aia` – папка для публикации сертификатов подписи УЦ;
- `/var/lib/cpca/cdp` – папка для публикации актуальных CRL.

Однако, допускается использование других папок.

2 Подготовка ОС

2.1 Установка ОС

Дистрибутив необходимо получить самостоятельно. Установка выполняется согласно эксплуатационной документации на ОС.

2.1.1 Организационно-технические меры при развертывании КриптоПро УЦ в ОС CN Astra Linux SE

При развертывании КриптоПро УЦ в замкнутой программной среде (ЗПС) ОС Astra Linux Special Edition требуется соблюдение перечня организационно-технических мер, описанных в «ЖТЯИ.00078-03 90 04 ПАК КриптоПро УЦ 2.0. Руководство по безопасности».

2.2 Установка КриптоПро CSP

Дистрибутив необходимо получить самостоятельно. Установка выполняется согласно эксплуатационной документации на СКЗИ КриптоПро CSP версий, указанных в Разделе 3 ЖТЯИ.00078-03 30 01 ПАК КриптоПро УЦ 2.0. Формуляр в соответствии с комплектацией исполнения.

2.3 Установка СУБД

Установка и настройка СУБД PostgreSQL или Microsoft SQL Server выполняется согласно эксплуатационной документации на СУБД.

2.4 Установка Программы настройки УЦ (pkica)

Программа настройки УЦ позволяет выполнять настройку компонентов УЦ, обеспечивает управление их БД и проч.

Программа настройки УЦ представляет собой приложение командной строки и позволяет выполнять настройку УЦ. Программа является самодокументирующей и имеет подробную встроенную справку по всем командам и ключам запуска. Для получения справки необходимо при запуске указать ключ `--help`.

Программа настройки УЦ (pkica) находится в дистрибутиве УЦ. Установка выполняется простым копированием. Необходимо дать права на выполнение следующего файла:

```
chmod u+x /opt/cpca/pkica/pkica
```

Перед запуском необходимо настроить в конфигурационном файле `appsettings.json` подключение к БД и службе очередей (настройка выполняется аналогично описанию в п.п. 3.2, 3.4, 3.17, 4.3, 4.4). Для конкретной установки Программы допускается не настраивать подключение к базам данных всех компонентов, достаточно указать подключение к БД только того компонента, управление которым будет производиться (например, при установке на сервере ЦС для управления ЦС достаточно указать только подключение к БД ЦС и не указывать подключение к БД ЦР).

2.5 Автоматический запуск сервисов УЦ

Запуск сервисов УЦ может выполняться путем запуска их исполняемых файлов. Однако, чтобы обеспечить автоматический старт сервисов после перезагрузки сервера можно настроить их автоматическую загрузку с помощью systemd как описано в этом разделе.

Для настройки автоматического запуска сервисов необходимо выполнить следующие действия.

- создать для каждого сервиса файл в следующей директории:

```
/etc/systemd/system/<имя сервиса>.service
```

Пример файла:

```
[Unit]
Description=nats-streaming-server
[Service]
WorkingDirectory=/opt/cpca/nats-streaming
ExecStart=/opt/cpca/nats-streaming/nats-streaming-server -sc /opt/cpca/nats-streaming/nats.conf
Restart=always
# Restart service after 10 seconds if the dotnet service crashes:
RestartSec=10
KillSignal=SIGINT
SyslogIdentifier=nats-streaming-server
User=cp
[Install]
WantedBy=multi-user.target
```

- Применить изменения:

```
sudo systemctl daemon-reload
```

- Разрешить автозагрузку:

```
sudo systemctl enable <имя юнит-сервиса>
```

- Запустить сервис:

```
sudo systemctl start <имя юнит-сервиса>
```

- Посмотреть информацию о работе сервиса можно с помощью следующих команд:

```
sudo systemctl status <имя приложения>
sudo journalctl -u <имя приложения>
```

2.6 Установка CRL в хранилище

Для подключения к NATS/NATS Streaming с использованием TLS необходимо, чтобы в локальном хранилище компьютера «Промежуточные Центры сертификации» (Local Machine\CA) были установлены действующие CRL. Сервис ЦС (CryptoPro.Ca.Service) и другие сервисы, взаимодействующие с ним через NATS Streaming, поддерживают автоматическую установку выпущенных CRL в это хранилище, но для этого требуется произвести дополнительную настройку прав доступа:

1. Создать группу безопасности, которой будут даны права для установки CRL в хранилище LocalMachine\CA:

```
sudo groupadd crl-writers
```

2. Сделать группу владельцем файла из состава КриптоПро CSP, в котором это хранилище хранится на диске:

```
sudo chown :crl-writers /var/opt/cprosp/users/stores/ca.sto
```

3. Дать созданной группе права на запись:

```
sudo chmod g+w /var/opt/cprosp/users/stores/ca.sto
```

В дальнейшем учетные записи, под которыми будут выполняться сервисы, устанавливающие CRL в хранилище, будут добавляться в эту группу:

```
sudo usermod --append --groups crl-writers <имя учетной записи>
```

Если же такая настройка безопасности не будет произведена, то при установке CRL в хранилище возникнет ошибка и в журнал сервиса будет записано сообщение «Доступ запрещен».

3 Настройка сервера ЦС

Последовательность действий при первоначальной установке:

- установка, настройка и запуск NATS Streaming без использования TLS;
- разворачивание ЦС (включает создание БД ЦС, создание экземпляра ЦС и сертификата ключа подписи);
- выпуск служебных сертификатов для подключения по TLS;
- выпуск CRL;
- настройка NATS Streaming и сервиса ЦС для работы с использованием TLS.

3.1 Установка ПО ЦС

На сервере ЦС подлежит установке и запуску сервис `CryptoPro.Ca.Service`. Установка сервиса выполняется простым копированием. Необходимо дать права на выполнение следующего файла:

```
chmod u+x /opt/cpca/CryptoPro.Ca.Service/CryptoPro.Ca.Service
```

Перед запуском необходимо настроить в конфигурационном файле `appsettings.json` приложения подключение к БД и службе очередей, указать серийный номер лицензии и наименование компании в разделе `License`, а также пути к файлам лога и журнала аудита.

3.2 Настройка подключения к БД

Перед настройкой подключения должна быть установлена соответствующая СУБД согласно п. 2.3 настоящего руководства.

Настройка подключения к СУБД задается в конфигурационных файлах `appsettings.json`, которые находятся в каталогах рядом с исполняемыми модулями `CryptoPro.Ca.Service` и `pkica`.

Для настройки подключения в `appsettngs.json` в разделе **CaDb** необходимо указать тип сервера и строку подключения.

Пример настройки подключения к БД PostgreSQL:

```
"CaDb": {  
  "StorageType": "PostgreSql",  
  "ConnectionString": "Server=localhost;Database=Ca;Username=postgres;Pooling=True"  
}
```

Обратите внимание, здесь в строке подключения не указан пароль для подключения к БД PostgreSQL. Рекомендуется хранить пароль подключения к БД PostgreSQL в файле паролей `.pgpass`. Подробнее о файле паролей `.pgpass` и рекомендациях по его защите см. эксплуатационную документацию СУБД PostgreSQL.

Пример настройки подключения к БД Microsoft SQL Server:

```
"CaDb": {  
  "StorageType": "SqlServer",  
  "ConnectionString": "Data Source=.;Initial Catalog=Ca;Integrated  
Security=True;Pooling=True"  
}
```

3.3 Установка и настройка службы очередей NATS Streaming

Служба очередей NATS Streaming с поддержкой ГОСТ TLS находится в дистрибутиве УЦ (папка nats-streaming). Установка выполняется простым копированием. Необходимо дать права на выполнение следующего файла:

```
chmod u+x /opt/cpca/nats-streaming/nats-streaming-server
```

В дистрибутиве прилагаются два файла с примерами конфигурации:

- nats.no-tls.conf – конфигурация без использования TLS (применяется только на этапе начальной настройки для выпуска сертификата ЦС, служебных TLS сертификатов, а также первого CRL);
- nats.conf – конфигурация с использованием TLS (применяется после выполнения начальной настройки ЦС и выпуска необходимых служебных сертификатов TLS).

В файле конфигурации необходимо указать адрес, на котором сервис должен принимать входящие подключения. Пример:

```
listen: ca.example:4222
```

А также каталог хранения сообщений (параметр dir). Пример:

```
dir: "/var/lib/nats-streaming/pkica-store"
```

Пользователь, под которым запускается процесс NATS Streaming должен иметь право на запись в каталог хранения сообщений.

```
sudo chown ca:ca --recursive /var/lib/nats-streaming/pkica-store
```

На этапе настройки ЦС для выпуска сертификата ЦС, служебных TLS сертификатов, а также первого CRL допускается запуск и подключение к службе очередей NATS Streaming без использования TLS. После выполнения начальной настройки ЦС и выпуска необходимых служебных сертификатов TLS требуется обязательно настроить службу очередей NATS Streaming на прием только TLS подключений. Настройка служебных сертификатов для TLS соединения выполняется в конфигурационном файле в разделе tls (см. пример в п. 3.17).

Запуск службы очередей NATS Streaming должен выполняться с указанием соответствующего конфигурационного файла. Пример:

```
/opt/cpca/nats-streaming/nats-streaming-server -sc /opt/cpca/nats-streaming/nats.conf
```

3.4 Настройка подключения к NATS Streaming без использования TLS

Настройка подключения к NATS Streaming задается в конфигурационном файле appsettings.json, который находится в каталоге рядом с исполняемым модулем CryptoPro.Ca.Service.

Для настройки подключения в appsettings.json в разделе необходимо указать адрес сервера, имя кластера и имя клиента. TLS на этом этапе надо отключить – установить параметр Secure в значение false (после выпуска необходимых технологических сертификатов требуется в обязательном порядке настроить подключение с использованием TLS).

Пример:

```
"Nats": {
  "Url": "nats://ca.example:4222",
  // настройки TLS
  "Secure": false, // включить TLS
  "Thumbprint": "89afdf6aaf47b23d9d19e63c280daf1b6c8b6e80", // отпечаток
  клиентского сертификата
  "StoreLocation": "CurrentUser" // имя хранилища
},
```

```
"Stan": {
  "Url": "nats://ca.example:4222",
  "ClusterID": "pkica-cluster",
  "ClientID": "ca-service",
  // настройки TLS
  "Secure": false, // включить TLS
  "Thumbprint": "89afdf6aaf47b23d9d19e63c280daf1b6c8b6e80", // отпечаток
клиентского сертификата
  "StoreLocation": "CurrentUser" // имя хранилища
}
```

3.5 Ввод лицензии ЦС

Данные лицензии необходимо указать в секции License конфигурационного файла appsettings.json CryptoPro.Ca.Service.

Если наименование компании в лицензии содержит кавычки, то их необходимо экранировать обратным слешем согласно правилам json.

Пример:

```
"License": {
  "SerialNumber": "C0210-00000-00000-00000-00000",
  "Company": "ООО \"КРИПТО-ПРО\""
}
```

3.6 Создание базы данных ЦС

Для создания БД Центра сертификации необходимо выполнить команду:

```
./pkica ca db new
```

Перед выполнением команды следует остановить все сервисы УЦ.

3.7 Запуск сервиса ЦС

После установки и настройки ЦС необходимо запустить сервис ЦС CryptoPro.Ca.Service. Пример:

```
/opt/cpca/CryptoPro.Ca.Service/CryptoPro.Ca.Service
```

3.8 Создание экземпляра ЦС

Экземпляр ЦС создается командой **pkica ca authority new --name <имя экземпляра>**. Имя экземпляра ЦС является уникальным для серверов ЦС, использующих общую базу данных.

Пример:

```
./pkica ca authority new --name "CryptoPro CA"
```

3.9 Настройка расписания выпуска CRL

ЦС выпускает CRL согласно настроенному расписанию. Пример команды получения текущего расписания выпуска CRL:

```
./pkica ca crl schedule get --name "CryptoPro CA"
```

Команда вернет текущее расписание выпуска базового и разностного CRL. Расписание выпуска указывается в формате CRON и позволяет выполнять гибкую настройку периодичности выпуска CRL.

Пример настройки расписания:

```
./pkica ca crl schedule update --name "CryptoPro CA" --base "0 0 18 * * *" --delta -
```

Здесь настраивается выпуск базового CRL каждый день в 18 часов, выпуск разностного CRL отключен.

Подробнее команды настройки и формат расписания выпуска CRL описаны в «ЖТЯИ.00078-03 90 03 ПАК КриптоПро УЦ 2.0. Руководство по эксплуатации».

3.10 Настройка шаблонов сертификатов

Шаблон сертификата – это совокупность правил формирования сертификата по запросу на сертификат. При выпуске сертификата ЦС обрабатывает запрос на сертификат и формирует сертификат в соответствии с шаблоном сертификата. Для обработки запросов на ЦС должны быть настроены соответствующие шаблоны.

Для управления шаблонами есть несколько команд:

- `./pkica ca template add` – добавление шаблона сертификата;
- `./pkica ca template get` – получение шаблона сертификата;
- `./pkica ca template list` – отображение списка шаблонов сертификатов;
- `./pkica ca template remove` – удаление шаблона сертификата;
- `./pkica ca template update` – обновление шаблона сертификата.

Получить подробную справку по каждой из этих команд можно с помощью ключа `--help`.

Пример добавления шаблонов для выпуска технологических TLS сертификатов:

```
./pkica ca template add --default-template Tls
```

Для добавления шаблонов квалифицированных сертификатов используются команды:

- `./pkica ca template add --default-template Qual` – добавляет шаблоны для выпуска квалифицированных сертификатов:
 - сертификат физического лица, срок действия 16 лет (1.2.643.2.2.46.0.10.0.16);
 - сертификат физического лица, срок действия 18 лет (1.2.643.2.2.46.0.10.0.18);
 - сертификат представителя юридического лица, срок действия 16 лет (1.2.643.2.2.46.0.10.1.16);
 - сертификат представителя юридического лица, срок действия 18 лет (1.2.643.2.2.46.0.10.1.18);
 - сертификат ИП, срок действия 16 лет (1.2.643.2.2.46.0.10.2.16);
 - сертификат ИП, срок действия 18 лет (1.2.643.2.2.46.0.10.2.18);
 - сертификат юридического лица, срок действия 16 лет (1.2.643.2.2.46.0.10.3.16);
 - сертификат юридического лица, срок действия 16 лет (1.2.643.2.2.46.0.10.3.18);
 - сертификат представителя иностранного юридического лица, срок действия 16 лет (1.2.643.2.2.46.0.10.7.16);
 - сертификат представителя иностранного юридического лица, срок действия 18 лет (1.2.643.2.2.46.0.10.7.18);
 - сертификат иностранного юридического лица, срок действия 16 лет (1.2.643.2.2.46.0.10.8.16);
 - сертификат иностранного юридического лица, срок действия 18 лет (1.2.643.2.2.46.0.10.8.18).

- `./pkica ca template add --default-template QualOcsdp` – добавляет шаблоны для выпуска сертификатов подписи службы актуальных статусов сертификатов (OCSP). Эти сертификаты содержат значение «Подпись ответов службы OCSP» (1.3.6.1.5.5.7.3.9) в расширении «Улучшенный ключ» и расширение «Признак доверия службе OCSP» (1.3.6.1.5.5.7.48.1.5):
 - сертификат физического лица для службы OCSP, срок действия 16 лет (1.2.643.2.2.50.1.11.0.16);
 - сертификат физического лица для службы OCSP, срок действия 18 лет (1.2.643.2.2.50.1.11.0.18);
 - сертификат представителя юридического лица для службы OCSP, срок действия 16 лет (1.2.643.2.2.50.1.11.1.16);
 - сертификат представителя юридического лица для службы OCSP, срок действия 18 лет (1.2.643.2.2.50.1.11.1.18);
 - сертификат ИП для службы OCSP, срок действия 16 лет (1.2.643.2.2.50.1.11.2.16);
 - сертификат ИП для службы OCSP, срок действия 18 лет (1.2.643.2.2.50.1.11.2.18);
 - сертификат юридического лица для службы OCSP, срок действия 16 лет (1.2.643.2.2.50.1.11.3.16);
 - сертификат юридического лица для службы OCSP, срок действия 16 лет (1.2.643.2.2.50.1.11.3.18);
 - сертификат представителя иностранного юридического лица для службы OCSP, срок действия 16 лет (1.2.643.2.2.50.1.11.7.16);
 - сертификат представителя иностранного юридического лица для службы OCSP, срок действия 18 лет (1.2.643.2.2.50.1.11.7.18);
 - сертификат иностранного юридического лица для службы OCSP, срок действия 16 лет (1.2.643.2.2.50.1.11.8.16);
 - сертификат иностранного юридического лица для службы OCSP, срок действия 18 лет (1.2.643.2.2.50.1.11.8.18).
- `./pkica ca template add --default-template QualTsp` – добавляет шаблоны для выпуска сертификатов подписи службы штампов времени (TSP). Эти сертификаты содержат значение «Подпись штампов времени» (1.3.6.1.5.5.7.3.8) в критическом расширении «Улучшенный ключ»:
 - сертификат физического лица для службы TSP, срок действия 16 лет (1.2.643.2.2.50.1.12.0.16);
 - сертификат физического лица для службы TSP, срок действия 18 лет (1.2.643.2.2.50.1.12.0.18);
 - сертификат представителя юридического лица для службы TSP, срок действия 16 лет (1.2.643.2.2.50.1.12.1.16);
 - сертификат представителя юридического лица для службы TSP, срок действия 18 лет (1.2.643.2.2.50.1.12.1.18);
 - сертификат ИП для службы TSP, срок действия 16 лет (1.2.643.2.2.50.1.12.2.16);
 - сертификат ИП для службы TSP, срок действия 18 лет (1.2.643.2.2.50.1.12.2.18);
 - сертификат юридического лица для службы TSP, срок действия 16 лет (1.2.643.2.2.50.1.12.3.16);
 - сертификат юридического лица для службы TSP, срок действия 16 лет (1.2.643.2.2.50.1.12.3.18);
 - сертификат представителя иностранного юридического лица для службы TSP, срок действия 16 лет (1.2.643.2.2.50.1.12.7.16);
 - сертификат представителя иностранного юридического лица для службы TSP, срок действия 18 лет (1.2.643.2.2.50.1.12.7.18);
 - сертификат иностранного юридического лица для службы TSP, срок действия 16 лет (1.2.643.2.2.50.1.12.8.16);
 - сертификат иностранного юридического лица для службы TSP, срок действия 18 лет (1.2.643.2.2.50.1.12.8.18).

Для редактирования шаблона необходимо сохранить его с помощью команды **pkica ca template get** в файл json, отредактировать файл с помощью любого текстового редактора и загрузить шаблон из файла с помощью команды **pkica ca template update**.

Пример:

```
./pkica ca template get --name User --file user.json
vi user.json
./pkica ca template update --file user.json
```

3.11 Прочие настройки ЦС

Перед выпуском сертификата ЦС и загрузкой ключей рекомендуется произвести все остальные настройки ЦС согласно ЖТЯИ.00078-03 90 03 ПАК КриптоПро УЦ 2.0. Руководство по эксплуатации, раздел **Настройки ЦС** (настроить адреса CDP и AIA, задать сроки действия сертификатов и т.д.). Эти настройки влияют на состав выпускаемых сертификатов и CRL, поэтому их важно установить заранее.

3.12 Создание ключа и сертификата ЦС

3.12.1 Создание ключа и сертификата корневого ЦС

Для создания ключа и сертификата корневого ЦС используется команда **pkica ca new-root-cert**.

Параметры (обязательные параметры помечены звездочкой *):

-o --out-file	Имя файла для сохранения сертификата. Если не задано, то сертификат будет выведен на консоль в кодировке base64.
-v --validity	Срок действия корневого сертификата. Задается в формате U:N, где U - единицы измерения срока действия, а N - количество таких единиц. N - целое число, U может принимать одно из значений Years, Months, Weeks, Days, Hours, Minutes. Значение по умолчанию: Years:16 (16 лет).
-V --key-validity	Срок действия ключа подписи корневого сертификата. Задается в формате U:N, где U - единицы измерения срока действия, а N - количество таких единиц. N - целое число, U может принимать одно из значений Years, Months, Weeks, Days, Hours, Minutes. Значение по умолчанию: Years:3 (3 года).
--path-length	Ограничение на длину пути.
-i --key-index	* Индекс ключа ЦС.
-s --subject	* Субъект.
--extensions-file	Файл с набором расширений сертификата. Формат файла описан в «ЖТЯИ.00078-03 90 03 ПАК КриптоПро УЦ 2.0. Руководство по эксплуатации».
--serial-number-	Алгоритм серийного номера. Допустимые значения:

algorithm	CryptoPro, Msca. Значение по умолчанию: CryptoPro.
-c --container-name	* Имя контейнера закрытого ключа.
--hash-algorithm	Имя или OID алгоритма хэширования. Если не задано, то будет использован алгоритм по умолчанию для провайдера.
-l --key-length	Длина ключа. Если не задана, то будет использовано значение по умолчанию для провайдера.
--use-existing-key	Использовать существующий ключ.
-e --exportable	Сделать ключ экспортируемым.
--prompt-for-media	Всегда показывать окно выбора носителя (только для КриптоПро CSP).
-p --provider-type	Тип провайдера закрытого ключа. Значение по умолчанию: 80 (ГОСТ 2012 256 бит).
-n --provider-name	Имя провайдера закрытого ключа сертификата ЦС. Если не указано, то будет использован провайдер заданного типа по умолчанию.
--pin	ПИН-код на контейнер. Если не задан, то будет запрошен во время выполнения.
--no-pin	Флаг указывает, что контейнер не имеет ПИН-кода. Подавляет запрос ПИН-кода во время выполнения.
-k --key-spec	Тип ключа. Допустимые значения: Signature, KeyExchange. Значение по умолчанию: Signature (ключ подписи).
--key-location	Расположение контейнера закрытого ключа. Допустимые значения: CurrentUser, LocalMachine. Значение по умолчанию: CurrentUser.
--silent	Работа с криптопровайдером в "тихом" режиме.

После выполнения команды, новый сертификат ЦС, в зависимости от наличия параметра --out-file, будет сохранен файл или выведен на консоль. На следующем шаге этот сертификат потребуется установить в хранилище сертификатов.

Пример:

```
./pkica ca new-root-cert -s "CN=CryptoPro CA;O=CryptoPro;C=RU" -c cont_ca -i 0
```

3.12.2 Создание ключа и сертификата подчиненного ЦС

Для создания сертификат подчиненного ЦС вначале необходимо создать запрос на этот сертификат. Команда создания запроса сертификат **pkica ca new-cert-req**.

Параметры (обязательные параметры помечены звездочкой *):

-o --out-file	Имя файла для сохранения сертификата. Если не задано, то
---------------	--

	сертификат будет выведен на консоль в кодировке base64.
-t --template	OID шаблона сертификата.
--template-major-version	Старшая версия шаблона. Значение по умолчанию: 1.
--template-minor-version	Младшая версия шаблона. По умолчанию не задана.
--is-ca	Запрос на сертификат ЦС (добавить расширение Основные ограничения и указать признак ЦС).
--path-length	Ограничение на длину пути (для расширения Основные ограничения).
--subject-sign-tool	Наименование средства ЭП. Если задано, то будет добавлено соответствующее расширение.
--sign-tool-class	Класс средства ЭП. Если задан, то будет добавлено расширение Политики сертификата с соответствующим значением. Допустимые значения: KC1, KC2, KC3, KB1, KB2, KA1.
--extensions-file	Файл с набором расширений запроса. Формат файла описан в «ЖТЯИ.00078-03 90 03 ПАК КриптоПро УЦ 2.0. Руководство по эксплуатации».
-s --subject *	Субъект сертификата ЦС.
-c --container-name *	Имя контейнера закрытого ключа сертификата ЦС.
--hash-algorithm	Имя или OID алгоритма хэширования. Если не задано, то будет использован алгоритм по умолчанию для провайдера.
-l --key-length	Длина ключа. Если не задана, то будет использовано значение по умолчанию для провайдера.
-i --key-index	Индекс ключа ЦС. Если задан, то будет добавлено расширение Версия ЦС.
--use-existing-key	Использовать существующий ключ.
-e --exportable	Сделать ключ экспортируемым.
--prompt-for-media	Всегда показывать окно выбора носителя (только для КриптоПро CSP).
-p --provider-type	Тип провайдера закрытого ключа сертификата ЦС. Значение по умолчанию: 80 (ГОСТ-2012).
-n --provider-name	Имя провайдера закрытого ключа сертификата ЦС. Если не указано, то будет использован провайдер заданного типа по умолчанию.

--pin	ПИН-код на контейнер. Если не задан, то будет запрошен во время выполнения.
--no-pin	Флаг указывает, что контейнер не имеет ПИН-кода. Подавляет запрос ПИН-кода во время выполнения.
-k --key-spec	Тип ключа. Допустимые значения: Signature, KeyExchange. Значение по умолчанию: Signature (ключ подписи).
--key-location	Расположение контейнера закрытого ключа. Допустимые значения: CurrentUser, LocalMachine. Значение по умолчанию: CurrentUser.
--silent	Работа с криптопровайдером в "тихом" режиме.

После выполнения команды, запрос сертификат ЦС в зависимости от наличия параметра --out-file, будет сохранен файл или выведен на консоль. Этот запрос необходимо обработать на вышестоящем УЦ. На следующем шаге выпущенный по этому запросу сертификат потребует-ся установить в хранилище сертификатов.

Пример:

```
./pkica ca new-cert-req --subject 'CN="ООО ""Вектор""", O="ООО ""Вектор""", OU="Отдел
УЦ", S="г. Москва 77", L=Москва, STREET="Московская ул. д.42", C=RU,
OGRN=4128724831913, INNLE=7768132748' --container-name ca --provider-type 80 --key-
spec Signature --is-ca --path-length 0 --key-index 0 --subject-sign-tool "КриптоПро
CSP 5.0" --sign-tool-class KC2 --out-file ca.req
```

3.12.3 Установка сертификата ЦС

Для установки сертификата ЦС, полученного на одном из предыдущих шагов, используются утилиты из состава КриптоПро CSP.

Обратите внимание, сервис ЦС должен иметь доступ к ключу подписи ЦС, поэтому сертификат ЦС должен быть установлен в хранилище пользователя, от имени которого запускается сервис ЦС.

3.12.3.1 Установка сертификата корневого ЦС

Для установки используется утилита **certmgr**.

```
/opt/cproscsp/bin/amd64/certmgr -install -store uMy -file <имя файла сертификата> -
provtype 80 -container cont_ca -at_signature
```

Также корневой сертификат нужно установить в хранилище доверенных корневых сертификатов (Root) без ссылки на ключ:

```
/opt/cproscsp/bin/amd64/certmgr -install -store uRoot -file <имя файла сертификата>
```

3.12.3.2 Установка сертификата подчиненного ЦС

Помимо, описанного в предыдущем пункте, способа использования утилиты **certmgr**, для установки сертификата в хранилище «Личные» можно использовать утилиту **cryptcp**. В этом случае не требуется указывать параметры контейнера ключа подписи, так как он будет найден автоматически по болванке сертификата в хранилище Request, созданной командой new-cert-req:

```
/opt/cprocsp/bin/amd64/cryptcp -instcert -from_request <имя файла сертификата>
```

Также сертификат подчиненного ЦС нужно добавить в хранилище промежуточных сертификатов ЦС (Ca) без ссылки на ключ:

```
/opt/cprocsp/bin/amd64/certmgr -install -store uCa -file <имя файла сертификата>
```

3.13 Добавление ключа подписи ЦС

Чтобы использовать ключ подписи для подписи сертификатов и CRL, необходимо добавить его экземпляру ЦС (предварительно сертификат должен быть установлен в хранилище со ссылкой на ключ согласно п. 3.12.3).

Для этого используется команда **pkica ca signing-key add --name <имя ЦС> --thumbprint <отпечаток сертификата> --store-name <имя хранилища> --store-location <расположение хранилища>**.

Параметры (обязательные параметры помечены звездочкой *):

-n --name *	Имя экземпляра ЦС, которому надо добавить ключ.
-t --thumbprint *	Отпечаток сертификата добавляемого ключа.
--store-name	Имя хранилища сертификатов, в которое требуется установить сертификат. Значение по умолчанию: My.
--store-location	Расположение хранилища сертификатов, в котором находится сертификат ключа. Допустимые значения: CurrentUser, LocalMachine. Значение по умолчанию: CurrentUser.
-u --usage	Разрешенное использование ключа подписи. Допустимые значения: SignCertAndCrl - ключ может использоваться для подписи сертификатов и CRL; SignCrl - ключ может использоваться только для подписи CRL; Retired - ключ выведен из эксплуатации и не может использоваться. Значение по умолчанию: SignCertAndCrl.
--hash-algorithm	Имя или OID алгоритма хэширования. Если не задано, то будет использован алгоритм по умолчанию для провайдера.
--accept-new-subject	По умолчанию субъект сертификата должен совпадать с субъектом текущего сертификата ЦС. Флаг отключает это поведение и позволяет добавить сертификат с отличающимся субъектом.
--planned-key-change-date	Плановая дата смены ключа. В течение 30 дней до наступления этой даты, а также после наступления, при загрузке ключа будет выводиться сообщение о необходимости плановой смены.
--planned-retirement-date	Плановая дата вывода ключа из обращения. В течение 30 дней до наступления этой даты, а также после наступления, при загрузке ключа будет выводиться сообщение о необходимости вывода ключа из обращения.

--pin	ПИН-код на контейнер. Если не задан, то будет запрошен во время выполнения.
--no-pin	Флаг указывает, что контейнер не имеет ПИН-кода. Подавляет запрос ПИН-кода во время выполнения.
--skip-container-check	Пропустить проверку контейнера ключа (в этом случае ПИН-код на контейнер не потребуется).

3.14 Загрузка ключа подписи ЦС

Для выпуска сертификатов и CRL добавленный ключ должен быть загружен на ЦС. Для этого используется команда **pkica ca signing-key load --name <имя ЦС> --index <индекс ключа> --pin <ПИН контейнера>**.

Параметры (обязательные параметры помечены звездочкой *):

-n --name	Имя экземпляра ЦС, ключ которого надо загрузить.
-i --index *	Индекс загружаемого ключа.
--pin	ПИН-код на контейнер. Если не задан, то будет запрошен во время выполнения.
--no-pin	Флаг указывает, что контейнер не имеет ПИН-кода. Подавляет запрос ПИН-кода во время выполнения.
-u --usage	Требуемое использование, для которого загружается ключ. Допустимые значения: • SignCertAndCrl - использовать для подписи сертификатов и CRL; • SignCrl - использовать только для подписи CRL; Если этот параметр отсутствует, то ключ будет загружен для всех использований, для которых он зарегистрирован.

3.15 Выпуск и установка служебных TLS сертификатов

Для подключения к NATS Streaming по TLS необходимо выпустить сертификат сервера NATS Streaming и сертификат клиентской аутентификации. Выпуск сертификатов TLS должен выполняться по соответствующим шаблонам. Перечень шаблонов сертификатов и процесс их настройки описан в п. 3.10.

Для взаимодействия с ЦР оператору (персоналу) УЦ необходимо иметь два ключа: ключ электронной подписи и ключ для обеспечения взаимодействия по протоколу TLS.

ЦС и ЦР, предназначенные для создания квалифицированных сертификатов ключей проверки ЭП, не могут быть использованы для создания сертификатов безопасности (технологических сертификатов), предназначенных для обеспечения взаимодействия компонентов ПАК «КриптоПро УЦ» между собой посредством криптографического протокола TLS с использованием сервера NATS Streaming. Для создания сертификатов безопасности необходимо использо-

вать другие экземпляры ЦС и ЦР. Экземпляр ЦС (ЦР) для создания сертификатов безопасности может размещаться на технических средствах с установленным ЦС (ЦР), предназначенным для выпуска квалифицированных сертификатов ключей проверки ЭП.

3.15.1 Создание запросов

Для создания запросов используется команда **pkica ca new-cert-req** (см. подробное описание команды в п. 3.12.2) без указания флага **--is-ca**, параметр **--key-спец** должен иметь значение **KeyExchange**. Для запроса на сертификат сервера в параметре «Шаблон сертификата» нужно указать 1.2.643.2.2.46.0.3, для запроса на сертификат клиента – 1.2.643.2.2.46.0.6. В компоненте «Общее имя» субъекте запроса на сертификат сервера нужно указать DNS имя компьютера, по которому будет осуществляться подключение к серверу NATS Streaming. Созданные запросы нужно сохранить в файлы (параметр **-o|--out-file**) для их дальнейшей обработки на ЦС.

3.15.2 Выпуск служебных сертификатов

Для обработки на ЦС запросов, созданных на предыдущем шаге, используется команда **pkica ca issue-cert**.

Параметры (обязательные параметры помечены звездочкой *):

-f --file *	Имя файла запроса на сертификат.
-o --out-file	Имя файла для сохранения сертификата. Если не задано, то сертификат будет выведен на консоль в кодировке base64.
-n --name	Имя экземпляра ЦС.
-t --timeout	Таймаут ожидания выпуска сертификата. Задается в формате ISO 8601: [d.]hh:mm[:ss]. Значение по умолчанию: 00:01 (1 минута). Если равен -1, то ожидание будет бесконечным; если 0, то будет выполнена отправка команды без ожидания выпуска сертификата.

3.15.3 Установка служебных сертификатов

Установка выпущенных служебных сертификатов осуществляется с помощью утилиты **cryptcp**:

```
/opt/cproscsp/bin/amd64/cryptcp -instcert -from_request <имя файла сертификата>
```

Если сертификат требуется установить в хранилище локального компьютера, то:

```
/opt/cproscsp/bin/amd64/cryptcp -instcert -from_request -dm -km <имя файла сертификата>
```

3.16 Выпуск и установка CRL

Компонент TLS при подключении к серверу NATS Streaming осуществляет проверку действительности клиентских сертификатов. Для того, чтобы эта проверка проходила, необходимо, чтобы в локальном хранилище был установлен действующий CRL. Поэтому предварительно перед настройкой подключения к NATS Streaming по TLS необходимо выпустить CRL, даже если на ЦС еще нет отозванных сертификатов.

3.16.1 Выпуск CRL

Выпуск CRL осуществляется командой **pkica ca crl issue**.

Параметры (обязательные параметры помечены звездочкой *):

-n --name	Имя экземпляра ЦС.
-d --delta-only	Выпустить только разностный CRL (без базового).
-u --next-update	Время до которого действует CRL. Формат ISO 8601: yyyy-MM-ddThh:mm:ss[±hh:mm].
-o --output-folder	Имя папки, в которую надо сохранить выпущенные CRL.
--force	Флаг подавляет запрос подтверждений перезаписи файлов.

Пример:

```
./pkica ca crl issue --name "CryptoPro CA" --output-folder /var/lib/cpca/cdp
```

3.16.2 Установка CRL в хранилище

После выпуска CRL нужно установить в локальное хранилище CA (сертификаты промежуточных ЦС) текущего пользователя. Это делается командой

```
/opt/cprosp/bin/amd64/certmgr -install -store uCa -crl -file <имя файла CRL>
```

Если производится установка подчиненного ЦС, то помимо выпущенного непосредственного самим ЦС CRL аналогичным образом в хранилища нужно установить CRL, выпущенные всеми вышестоящими ЦС.

3.17 Настройка подключения к NATS Streaming с использованием TLS

Для настройки подключения к NATS Streaming с использованием TLS нужно остановить сервис NATS Streaming и CryptoPro.Ca.Service.

Далее необходимо внести изменения в конфигурационный файл NATS Streaming: включить режим требования TLS и проверки клиентских сертификатов, а также указать серверный сертификат NATS и клиентский сертификат, используемый для внутреннего подключения NATS Streaming к NATS. Эти сертификаты были выпущены и установлены согласно п. 3.15 настоящего руководства. Сертификаты нужно сохранить в кодировке DER (утилиты из состава КриптоПро CSP и КриптоПро УЦ сохраняют сертификаты в кодировке DER) и указать путь до них в конфигурационном файле:

```
...
tls: {
  # серверный сертификат NATS
  cert_file:      "/opt/cpca/nats-streaming/ssl/nats-server.cer"
  key_file:       "/opt/cpca/nats-streaming/ssl/nats-server.cer"
  verify_and_map: true
}
...

# NATS Streaming Configuration
streaming: {
```

```
cluster_id: pkica-cluster

tls: {
  # клиентский сертификат NATS Streaming для подключения к сервису NATS
  client_cert: "/opt/cpca/nats-streaming/ssl/nats-client.cer"
  client_key:  "/opt/cpca/nats-streaming/ssl/nats-client.cer"
}
```

После настройки и запуска NATS Streaming нужно настроить сервис ЦС CryptoPro.Ca.Service и pkica для подключения по TLS. Для этого нужно отредактировать конфигурационные файлы appsettings.json: выставить параметр Secure в true и задать сертификат подключения, указав отпечаток и имя хранилища.

Пример

```
"Nats": {
  "Url": "nats://ca.example:4222",
  // настройки TLS
  "Secure": true, // включить TLS
  "Thumbprint": "89afdf6aaf47b23d9d19e63c280daf1b6c8b6e80", // отпечаток
клиентского сертификата
  "StoreLocation": "CurrentUser" // имя хранилища
},

"Stan": {
  "Url": "nats://ca.example:4222",
  "ClusterID": "pkica-cluster",
  "ClientID": "ca-service",
  // настройки TLS
  "Secure": true, // включить TLS
  "Thumbprint": "89afdf6aaf47b23d9d19e63c280daf1b6c8b6e80", // отпечаток
клиентского сертификата
  "StoreLocation": "CurrentUser" // имя хранилища
}
```

После этого нужно запустить сервис ЦС CryptoPro.Ca.Service (п. 3.7 настоящего руководства).

3.18 Восстановление работоспособности при пропуске выпуска очередного CRL

В случае если по тем или иным причинам очередной CRL не был выпущен вовремя и срок действия предыдущего CRL истек, перестанут проверяться на отзыв сертификаты, выпущенные на данном УЦ. Если пропуск времени выпуска CRL случился на технологическом УЦ, который выпускает технологические TLS сертификаты, то TLS соединения перестанут устанавливаться.

Для восстановления работоспособности необходимо выполнить следующие действия:

- Просмотреть записи в журналах, найти причину невыпуска CRL и устранить ее.
- Если настроена автозагрузка ключа ЦС (не должно быть ПИН-кода на контейнере ключа), то ключ загрузится автоматически.
- Если автозагрузка ключа ЦС не настроена, то:
 - отключить требование TLS в NATS, руководствуясь разделом 3.4;
 - загрузить ключ согласно п. 3.14.
- CRL будет выпущен автоматически в течение минуты после устранения проблем и загрузки ключа. Также можно выпустить CRL вручную согласно п. 3.16.1.
- Выпущенный актуальный CRL перенести вручную на ЦР и установить его согласно п. 3.16.2.

4 Настройка сервера ЦР

4.1 Подготовка сервисных сертификатов

Для обеспечения функционирования Центра регистрации необходимо подготовить следующие сервисные сертификаты:

- сертификат веб-сервера ЦР;
- сертификат TLS клиента для подключения сервиса ЦР к службе очередей;
- сертификаты Операторов ЦР.

4.2 Установка ПО ЦР

На сервере ЦР подлежит установке и запуску сервис ЦР CryptoPro.Ra.Service и веб-приложение API ЦР CryptoPro.Ra.Web. Установка сервиса ЦР и веб-приложения API ЦР выполняется простым копированием. Необходимо дать права на выполнение следующим файлам:

```
chmod u+x /opt/cpca/CryptoPro.Ra.Service/CryptoPro.Ra.Service
chmod u+x /opt/cpca/CryptoPro.Ra.Web/CryptoPro.Ra.Web
```

Перед запуском необходимо настроить в конфигурационном файле appsettings.json приложений подключение к БД и службе очередей, указать серийный номер лицензии и наименование компании в разделе License, а также пути к файлам лога и журнала аудита.

Также, необходимо выполнить установку веб-приложения UI ЦР CryptoPro.Ra.Web.Ui (выполняется простым копированием). В конфигурационном файле init.json необходимо в параметре "API_HOST" указать URL для доступа к API ЦР. Веб-приложение UI ЦР состоит из набора статических HTML страниц, не требует отдельного запуска и должно быть опубликовано с помощью веб-сервера.

Веб-приложения API и UI ЦР должны быть опубликованы за веб-сервером. Подробнее про развертывание и настройку веб-сервера см. п. 4.9.

4.3 Настройка подключения к БД

Перед настройкой подключения должна быть установлена соответствующая СУБД согласно п. 2.3 настоящего руководства.

Настройка подключения к СУБД задается в конфигурационных файлах appsettings.json, которые находятся в каталогах рядом с исполняемыми модулями CryptoPro.Ra.Service, CryptoPro.Ra.Web и pkica.

Для настройки подключения в appsettngs.json в разделе **RaDb** необходимо указать тип сервера и строку подключения.

Пример настройки подключения к PostgreSQL:

```
"RaDb": {
  "StorageType": "PostgreSql",
  "ConnectionString": "Server=localhost;Database=Ra;Username=postgres;Pooling=True"
}
```

Обратите внимание, здесь в строке подключения не указан пароль для подключения к БД PostgreSQL. Рекомендуется хранить пароль подключения к БД PostgreSQL в файле паролей .pgpass. Подробнее о файле паролей .pgpass и рекомендациях по его защите см. эксплуатационную документацию СУБД PostgreSQL.

Пример настройки подключения к Microsoft SQL Server:

```
"RaDb": {
  "StorageType": "SqlServer",
  "ConnectionString": "Data Source=.;Initial Catalog=Ra;Integrated
Security=True;Pooling=True"
}
```

4.4 Настройка подключения к NATS Streaming

Необходимо настроить подключение сервера ЦР к сервису NATS Streaming, запущенному на сервере ЦС.

На сервере ЦР в конфигурационных файлах appsettings.json для CryptoPro.Ra.Service, CryptoPro.Ra.Web и pkica необходимо указать адрес подключения (Url) к сервису NATS Streaming, выставить параметр Secure в true и задать сертификат подключения, указав отпечаток и имя хранилища.

Пример:

```
"Nats": {
  "Url": "nats://ca.example:4222",
  // настройки TLS
  "Secure": true, // включить TLS
  "Thumbprint": "89afdf6aaf47b23d9d19e63c280daf1b6c8b6e80", // отпечаток
  клиентского сертификата
  "StoreLocation": "CurrentUser" // имя хранилища
},

"Stan": {
  "Url": "nats://ca.example:4222",
  "ClusterID": "pkica-cluster",
  "ClientID": "ra-service",
  // настройки TLS
  "Secure": true, // включить TLS
  "Thumbprint": "89afdf6aaf47b23d9d19e63c280daf1b6c8b6e80", // отпечаток
  клиентского сертификата
  "StoreLocation": "CurrentUser" // имя хранилища
}
```

После этого нужно запустить сервис ЦС CryptoPro.Ca.Service (п. 3.7 настоящего руководства).

4.5 Настройка публикации сертификатов подписи в AIA

Настройка публикации осуществляется в конфигурационных файлах appsettings.json, которые находятся в каталогах рядом с исполняемым модулем CryptoPro.Ra.Service. Если задать параметру PublishCaCerts значение true, то сертификаты подписи будут публиковаться в папку, указанную в параметре AiaFolder. Для публикации сертификатов в несколько разных папок, необходимо запустить соответствующее количество экземпляров сервиса. При этом требуется задать им разные значения NodeId.

```
"Aia": {
  "PublishCaCerts": true,
  "AiaFolder": "/var/lib/cpca/aia",
  "NodeId": "ra"
}
```

4.6 Настройка публикации выпущенных CRL в CDP

Настройка публикации осуществляется в конфигурационных файлах `appsettings.json`, которые находятся в каталогах рядом с исполняемым модулем `CryptoPro.Ra.Service`. Если задать параметру `PublishCrls` значение `true`, то выпускаемые CRL будут публиковаться в папку, указанную в параметре `CdpFolder`. Для публикации CRL в несколько разных папок, необходимо запустить соответствующее количество экземпляров сервиса. При этом требуется задать им разные значения `NodeId`.

```
"Cdp": {
  "PublishCrls": true,
  "CdpFolder": "/var/lib/cpsa/cdp",
  "NodeId": "ra"
}
```

4.7 Ввод лицензии ЦР

Данные лицензии необходимо указать в секции `License` конфигурационных файлов `appsettings.json` `CryptoPro.Ra.Service` и `CryptoPro.Ra.Web`.

Если наименование компании в лицензии содержит кавычки, то их необходимо экранировать обратным слешем согласно правилам `json`.

Пример:

```
"License": {
  "SerialNumber": "R0210-00000-00000-00000-00000",
  "Company": "ООО \"КРИПТО-ПРО\""
}
```

4.8 Создание базы данных ЦР

Для создания БД Центра регистрации необходимо выполнить команду:

```
./pkica ra db new
```

Перед выполнением команды следует остановить все сервисы УЦ.

4.9 Развертывание веб-сервера

Необходимо выполнить настройку веб-сервера `nginx`, `Apache` или `IIS` согласно эксплуатационной документации на СКЗИ КриптоПро CSP в соответствии с версиями СКЗИ, указанными в Разделе 3 ЖТЯИ.00078-03 30 01 ПАК КриптоПро УЦ 2.0. Формуляр.

4.9.1 Подготовка служебных ключей и сертификатов

Перед настройкой веб-сервера Центра регистрации необходимо подготовить сертификат веб-сервера TLS. Требования к нему такие же, как к серверным сертификатам — наличие в субъекте или расширении «Дополнительное имя субъекта» (SAN) имени хоста и наличие Проверка подлинности сервера (1.3.6.1.5.5.7.3.1) в расширении «Улучшенный ключ».

Ниже описан пример установки сертификата из `pfx`. В этом варианте, сертификат веб-сервера должен быть получен заранее на ЦС и скопирован на сервер с компонентами ЦР в виде `pfx`-файла.

```
sudo -u nginx /opt/cprosp/bin/amd64/certmgr -install -pfx -store uMy -file
<path>/certificate.pfx -pin <пароль от файла pfx>
```

Для успешной проверки серверных и клиентских сертификатов необходимо установить сертификаты всех вышестоящих УЦ. Ниже пример установки корневых и промежуточных сертификатов УЦ (где mRoot — хранилище корневых сертификатов, mCa – хранилище промежуточных сертификатов).

```
sudo /opt/cprocsp/bin/amd64/certmgr -inst -store mRoot -file /<path>/root.cer
sudo /opt/cprocsp/bin/amd64/certmgr -inst -store mCa -file /<path>/to/ca.cer
```

4.9.2 Развертывание веб-сервера nginx в замкнутой программной среде (ЗПС) ОС Astra Linux SE

В составе дистрибутива УЦ поставляется установочный пакет веб-сервера nginx 1.18.0 (nginx_1.18.0~astra-gost_amd64_signed.deb), подготовленный для использования в замкнутой программной среде (ЗПС) ОС Astra Linux SE.

Пример команды установки:

```
sudo dpkg -i nginx_1.18.0~astra-gost_amd64_signed.deb
```

4.9.3 Развертывание веб-сервера IIS в ОС Microsoft Windows.

Развертывание IIS производится средствами операционной системы, в **Диспетчере сервера** через мастер добавления ролей и компонентов.

4.9.4 Конфигурация IIS для Центра регистрации

Конфигурация IIS осуществляется с использованием оснастки MMC **Диспетчер служб IIS** из состава ОС Windows. Оснастка устанавливается в процессе добавления роли IIS.

Для обеспечения работоспособности ЦР на IIS необходимо настроить веб-приложение CryptoPro.Ra.Web и виртуальный каталог с веб-интерфейсом оператора CryptoPro.Ra.Web.UI. Для работы по двухстороннему TLS на IIS должна быть настроена привязка в соответствии с разделом **4.4 Установка сертификата IIS** из документации «ЖТЯИ.00103-02 92 01. Инструкция по использованию. Windows» КриптоПро CSP.

Перед настройкой веб-приложения нужно создать пул приложений (например **Ra**), работающий под учетной записью, у которой есть права доступа к ключу и сертификату, используемым для подключения к сервису NATS Streaming и к базе данных Центра регистрации.

При создании непосредственно веб-приложения CryptoPro.Ra.Web в качестве физического пути необходимо выбрать локальную папку, содержащую бинарные модули CryptoPro.Ra.Web, а в качестве пула приложений – пул, созданный выше.

Для созданного веб-приложения необходимо включить **Требовать SSL** и **Требовать** для сертификатов клиентов в **Параметрах SSL**.

Для настройки UI оператора ЦР необходимо создать виртуальный каталог, указав в качестве физического пути локальную папку, содержащую файлы CryptoPro.Ra.Web.UI. Для этого каталога так же необходимо включить **Требовать SSL** и **Требовать** для сертификатов клиентов в **Параметрах SSL**.

В файле конфигурации **init.json** в CryptoPro.Ra.Web.UI в параметре **API_HOST** раздела **config** нужно указать URL адрес созданного веб-приложения CryptoPro.Ra.Web.

4.10 Настройка папок и создание операторов

Для упорядоченного хранения пользователей, запросов и сертификатов на Центре регистрации предусмотрена одноуровневая структура папок. По умолчанию она отсутствует и должна быть создана администратором после настройки подключения к БД Центра регистрации. Для работы Центра регистрации обязательно должна существовать хотя бы одна папка.

Операторы Центра регистрации так же работают с этими папками в соответствии с назначенными разрешениями. Но конкретному оператору разрешения на папку дать нельзя, разрешения даются группе операторов, в которой этот оператор состоит. Если оператор состоит в нескольких группах, то применяется сумма всех разрешений (например, если для одной группы, в которой состоит оператор, разрешена отправка запросов на сертификат в какую-то папку, а для другой группы этого же оператора разрешено рассмотрение запросов на сертификат в эту же папку, то этому оператору будут доступны и отправка и рассмотрение запросов на сертификат). У каждого оператора есть папка по умолчанию — эта папка используется, если оператор явно не указал папку при создании пользователя, получении данных и т.д.

Вариант последовательности настройки:

- создать одну или несколько папок
- создать одну или несколько групп
- создать оператора, с указанием папки по умолчанию и группы, в которой он состоит. При создании можно указать только одну группу, если требуется добавить оператора в другие группы, то это нужно сделать отдельными командами.
- назначить необходимые разрешения группам на папки, с которыми будет работать оператор.

4.10.1 Создание папок Центра регистрации

Для добавления папки, в которой будут содержаться объекты ЦР (пользователи, запросы, сертификаты) используется команда **pkica ra folder add**:

Параметр `-n|--name` (имя папки) является обязательным и помечен звездочкой (*). Имя папки должно быть уникальным (на Центре регистрации не должно существовать папок с тем же именем).

Параметры (звездочкой * помечены обязательные):

<code>-n --name</code>	* Имя папки.
------------------------	--------------

Пример:

```
./pkica ra folder add --name Ra
```

4.10.2 Создание групп Центра регистрации

Для создания группы операторов используется команда **pkica ra group add**.

Параметр `-n|--name` (название группы) является обязательным и помечен звездочкой (*). Название группы должно быть уникальным (на Центре регистрации не должно существовать групп с тем же именем).

Параметры (звездочкой * помечены обязательные):

-n --name	* Название группы.
-----------	--------------------

Пример:

```
./pkica ra group add --name Operators
```

4.10.3 Конфигурация nginx для Центра регистрации

Для обеспечения работоспособности ЦР nginx необходимо настроить в режимах двусторонней аутентификации по сертификату (Mutual TLS) и обратного прокси-сервера (reverse-proxy). Настройка выполняется путем редактирования файла конфигурации nginx (/etc/nginx/nginx.conf).

Пример конфигурации nginx:

```
worker_processes auto;
error_log /var/log/nginx/error.log;
pid /var/run/nginx.pid;
worker_rlimit_nofile 8192;

events {
    worker_connections 4096;
}

http {
    include mime.types;
    default_type text/html;

    log_format main '$remote_addr - $remote_user [$time_local] $status '
        '$request' $body_bytes_sent "$http_referer" '
        '$http_user_agent' "$http_x_forwarded_for";

    access_log /var/log/nginx/access.log main;

    server {
        listen 443;
        ssl on;

        # DNS-имя сервера ЦР
        server_name astra-ca;

        # серийный номер сертификата веб-сервера
        ssl_certificate 0x486CE9B87EEF169A4F011E3FC59AF01C;

        ssl_protocols TLSv1 TLSv1.1 TLSv1.2;
        ssl_verify_client on;
        ssl_client_certificate root;

        proxy_set_header Host $host;
        proxy_set_header X-Real-IP $remote_addr;
        proxy_set_header X-Forwarded-For $proxy_add_x_forwarded_for;
        proxy_set_header X-Forwarded-Proto $scheme;
        proxy_set_header X-SSL-CERT $ssl_client_certificate;

        add_header X-Frame-Options "SAMEORIGIN";
        add_header X-XSS-Protection "1; mode=block";
        add_header X-Content-Type-Options "nosniff";

        charset utf-8;

        location / {
```

```

    root /opt/cpca/CryptoPro.Ra.Web.Ui;
    try_files $uri $uri/ /index.html;
}

location /api/ {
    proxy_pass http://localhost:5000;
}

location /swagger/ {
    proxy_pass http://localhost:5000;
}

location /docs/ {
    proxy_pass http://localhost:5000;
}
}
}

```

Пример секции `server` (должна быть включена в секцию `http` конфигурационного файла `nginx`) для настройки точки публикации CDP и AIA:

```

server {
    default_type application/octet-stream;

    # DNS-имя сервера ЦР
    server_name astra-ca;

    charset utf-8;

    location /cdp/ {
        root /var/lib/cpca;
    }

    location /aia/ {
        root /var/lib/cpca;
    }
}

```

После изменения конфигурации необходимо перезапустить `nginx`:

```
sudo systemctl restart nginx
```

4.10.4 Создание операторов Центра регистрации

Для создания оператора используется команда **`pkica ra operator add`**:

Перед созданием оператора требуется, чтобы были выпущены сертификат TLS аутентификации на Центре Регистрации (неквалифицированный) и сертификат подписи (квалифицированный), удовлетворяющие следующим требованиям:

- сертификат подписи должен содержать биты «Цифровая подпись» (0) и «Неотрекаемость» (1) в расширении «Использование ключа»
- сертификат аутентификации должен содержать назначение «Проверка подлинности клиента» (идентификатор 1.3.6.1.5.5.7.3.2) в расширении «Улучшенный ключ»

Имя входа (логин) рекомендуется задавать так, чтобы оно было понятным и легко запоминающимся, но при этом уникальным. Подходящие варианты - сокращение ФИО или адрес email.

Дополнительно можно указать одну из созданных папок в качестве папки по умолчанию для оператора, тогда эта папка будет использоваться при создании пользователя, получении данных и т.д., если явно не указана другая папка.

Параметры (звездочкой * помечены обязательные):

-l --last-name	* Фамилия оператора.
-f --first-name	* Имя оператора.
-m --middle-name	Отчество оператора.
--tls-cert-file	Путь к файлу сертификата аутентификации оператора.
--signing-cert-file	Путь к файлу сертификата подписи оператора.
-g --group	Название группы, в которую надо добавить оператора.
--default-folder	Имя папки по умолчанию. Эта папка будет использоваться, когда оператор явно не указал папку при создании пользователя, получении данных и т.п.
--login	Имя входа (логин) оператора. Является уникальным отличительным свойством каждого оператора, служит для идентификации оператора на ЦР. Если не задано, то будет сгенерировано автоматически из ФИО.

Пример:

```
./pkica ra operator add --last-name "Петров" --first-name "Петр" --tls-cert-file operator-auth.cer --signing-cert-file operator-sign.cer --group Operators --default-folder Ra --login Petrov
```

4.10.5 Назначение разрешений на папку

Для добавления разрешений группе на ранее созданную папку необходимо выполнить команду **pkica ra permission grant**:

В качестве параметров указывается папка (либо отдельным флагом можно указать все папки), на которой назначаются разрешения, группа (либо отдельным флагом можно указать все группы), для которой назначаются разрешения, и перечень дополнительных разрешений. Разрешение на чтение добавляется всегда при выполнении этой команды, даже если перечень дополнительных разрешений отсутствует (т.е. нельзя разрешить, к примеру, управление пользователями в папке, не разрешив при этом чтение в этой папке).

Параметры:

--folder	Имя папки.
--all-folders	Назначить разрешения для всех папок.
--group	Название группы.
--all-groups	Назначить разрешения для всех групп.

<code>--manage-user</code>	Управление пользователями.
<code>--submit-cert-request</code>	Отправка запросов на сертификат.
<code>--resolve-cert-request</code>	Рассмотрение (одобрение/отклонение) запросов на сертификат.
<code>--submit-rev-request</code>	Отправка запросов на отзыв.
<code>--resolve-rev-request</code>	Рассмотрение (одобрение/отклонение) запросов на отзыв.
<code>--all</code>	Разрешить всё.

Пример:

```
./pkica ra permission grant --folder Ra --group Operators --manage-user --submit-cert-request --resolve-cert-request --submit-rev-request --resolve-rev-request
```

4.11 Добавление шаблонов сертификатов

Для создания запросов на сертификаты пользователей Оператором ЦР необходимо настроить список шаблонов сертификатов. Параметры этих шаблонов будут использоваться при создании запросов на рабочем месте Оператора.

Предварительно надо получить список настроенных шаблонов на Центре сертификации, сохранив его в файл формата JSON командой

```
./pkica ca template list --detailed --file <файл с шаблонами> --format Json
```

Далее полученный файл с шаблонами нужно перенести на Центр регистрации и добавить с помощью команды

```
./pkica ra template add --file <файл с шаблонами>
```

После добавления шаблонов вручную с использованием файла с шаблонами возможно настроить автоматический перенос шаблонов с ЦС на ЦР. Для этого нужно включить режим автообновления на ЦР командой команды **pkica ra template auto-update --enable**. После включения режима автообновления шаблоны на ЦР будут автоматически обновляться после каждого изменения списка шаблонов на ЦС.

4.12 Настройка печатных форм

АРМ Оператора поддерживает печатные формы сертификата, заявлений на регистрацию пользователя, выпуск и отзыв сертификата.

Печатные формы являются настраиваемыми представлениями, их можно отредактировать на сервере ЦР. Шаблоны печатных форм представляют собой файлы html. Они расположены на сервере в подпапке **templates** папки установки веб-приложения UI ЦР **CryptoPro.Ra.Web.Ui**.

Перечень файлов печатных форм:

- **user.html** – печатная форма заявления на регистрацию пользователя;
- **certificate.html** – печатная форма сертификата;
- **certRequest.html** – печатная форма заявления на изготовление сертификата;
- **revocation.html** – печатная форма заявления на отзыв сертификата;
- **tpl.header.html** – общая «шапка» для печатных форм заявлений.

Редактирование печатных форм может выполняться в простом текстовом редакторе.

5 Настройка рабочего места Оператора

5.1 Подготовка рабочего места Оператора

Перед подключением к веб-интерфейсу необходимо выполнить следующие действия:

- Установить СКЗИ «КриптоПро CSP» версии, указанной в Разделе 3 ЖТЯИ.00078-03 30 01 ПАК КриптоПро УЦ 2.0. Формуляр в соответствии с комплектацией исполнения.
- Установить КриптоПро ЭЦП Browser plug-in из состава дистрибутива УЦ (PKI SDK).
- Установить сертификаты аутентификации и подписи Оператора в хранилище Личное текущего пользователя.
- Для подключения с использованием браузера Chromium Gost необходимо установить расширение CryptoPro Extension for CAdES Browser Plug-in. Расширение будет установлено автоматически при первом запуске браузера после установки КриптоПро ЭЦП Browser plug-in, после установки нужно убедиться, что оно включено.

5.2 Подключение к веб-интерфейсу ЦР

Подключение Оператора к веб-интерфейсу ЦР осуществляется с использованием браузера Chromium Gost или Яндекс Браузер по защищенному TLS-соединению с двусторонней аутентификацией.

Перед подключением Оператора к веб-интерфейсу ЦР необходимо зарегистрировать оператора согласно п. 4.10.

Подробнее подключение Оператора к веб-интерфейсу ЦР и работа в веб-интерфейсе описана в «ЖТЯИ.00078-03 90 03 ПАК КриптоПро УЦ 2.0. Руководство по эксплуатации».

6 Проверка работоспособности

После выполнения установки и настройки ПО, согласно настоящему Руководству, необходимо выполнить проверку работоспособности. Для этого выполните следующие действия с помощью АРМ Оператора согласно «ЖТЯИ.00078-03 90 03 ПАК КриптоПро УЦ 2.0. Руководство по эксплуатации»:

- Выполнить вход в АРМ Оператора, используя сертификат аутентификации оператора.
- Создание пользователя.
- Выпуск сертификата:
 - создать и одобрить запрос на сертификат;
 - убедиться, что сертификат выпущен.
- Отзыв сертификата:
 - создать и одобрить запрос на отзыв сертификата;
 - убедиться, что сертификат отозван;
 - выполнить выпуск CRL;
 - убедиться, что отозванный сертификат присутствует в выпущенном CRL.

7 Настройка сервера Реестра сертификатов

7.1 Установка ПО Реестра сертификатов

На сервере ЦС подлежит установке и запуску сервис `CryptoPro.CertRegistry.Service`. Установка сервиса выполняется простым копированием. Необходимо дать права на выполнение следующим файлам:

```
chmod u+x /opt/cpca/CryptoPro.CertRegistry.Service/CryptoPro.CertRegistry.Service
chmod u+x /opt/cpca/CryptoPro.CertRegistry.Web/CryptoPro.CertRegistry.Web
```

Перед запуском необходимо настроить в конфигурационном файле `appsettings.json` приложений подключение к БД и службе очередей, а также пути к файлам лога и журнала аудита.

7.2 Настройка подключения к БД

Перед настройкой подключения должна быть установлена соответствующая СУБД согласно п. 2.3 настоящего руководства.

Настройка подключения к СУБД задается в конфигурационных файлах `appsettings.json`, которые находятся в каталогах рядом с исполняемыми модулями `CryptoPro.CertRegistry.Service` и `pkica`.

Для настройки подключения в `appsettngs.json` в разделе **CertRegistryDb** необходимо указать тип сервера и строку подключения.

Пример настройки подключения к БД PostgreSQL:

```
"CertRegistryDb": {
  "StorageType": "PostgreSql",

  "ConnectionString": "Server=localhost;Database=CertRegistry;Username=postgres;Pooling=
True"
}
```

Обратите внимание, здесь в строке подключения не указан пароль для подключения к БД PostgreSQL. Рекомендуется хранить пароль подключения к БД PostgreSQL в файле паролей `.pgpass`. Подробнее о файле паролей `.pgpass` и рекомендациях по его защите см. эксплуатационную документацию СУБД PostgreSQL.

Пример настройки подключения к БД Microsoft SQL Server:

```
"CertRegistryDb ": {
  "StorageType": "SqlServer",
  "ConnectionString": "Data Source=.;Initial Catalog=CertRegistry;Integrated
Security=True;Pooling=True"
}
```

7.3 Настройка подключения к NATS Streaming

Необходимо настроить подключение сервера Реестра сертификатов к сервису NATS Streaming, запущенному на сервере ЦС.

На сервере ЦР в конфигурационных файлах `appsettings.json` для `CryptoPro.CertRegistry.Service` необходимо указать адрес подключения (Url) к сервису NATS Streaming, выставить параметр `Secure` в `true` и задать сертификат подключения, указав отпечаток и имя хранилища.

Пример:

```
"Stan": {
  "Url": "nats://ca.example:4222",
  "ClusterID": "pkica-cluster",
  "ClientID": "cert-registry-service",
  // настройки TLS
  "Secure": true, // включить TLS
  "Thumbprint": "89afdf6aaf47b23d9d19e63c280daf1b6c8b6e80", // отпечаток
  // клиентского сертификата
  "StoreLocation": "CurrentUser" // имя хранилища
}
```

7.4 Настройка публикации сертификатов подписи в AIA

Настройка публикации осуществляется в конфигурационных файлах appsettings.json, которые находятся в каталогах рядом с исполняемым модулем CryptoPro.CertRegistry.Service. Если задать параметру PublishCaCerts значение true, то сертификаты подписи будут публиковаться в папку, указанную в параметре AiaFolder. Для публикации сертификатов в несколько разных папок, необходимо запустить соответствующее количество экземпляров сервиса. При этом требуется задать им разные значения NodeId.

```
"Aia": {
  "PublishCaCerts": true,
  "AiaFolder": "/var/lib/cpca/aia",
  "NodeId": "certRegistry"
}
```

7.5 Настройка публикации выпущенных CRL в CDP

Настройка публикации осуществляется в конфигурационных файлах appsettings.json, которые находятся в каталогах рядом с исполняемым модулем CryptoPro.CertRegistry.Service. Если задать параметру PublishCrls значение true, то выпускаемые CRL будут публиковаться в папку, указанную в параметре CdpFolder. Для публикации CRL в несколько разных папок, необходимо запустить соответствующее количество экземпляров сервиса. При этом требуется задать им разные значения NodeId.

```
"Cdp": {
  "PublishCrls": true,
  "CdpFolder": "/var/lib/cpca/cdp",
  "NodeId": "certRegistry"
}
```

7.6 Создание базы данных Реестра сертификатов

Для создания БД Центра регистрации необходимо выполнить команду:

```
./pkica cert-registry db new
```

Перед выполнением команды следует остановить сервис CryptoPro.CertRegistry.Service.

7.7 Развертывание и конфигурация веб-сервера

Развертывание и конфигурация веб-сервера Реестра сертификатов выполняется аналогично развертыванию и конфигурации веб-сервера Центра регистрации, как описано в разделе 4.9. У Реестра сертификатов нет пользовательского интерфейса, поэтому шаги, связанные с его развертыванием (*.Web.UI) следует пропустить.

