



Центр сертификатов доступа

Aladdin Enterprise Certificate Authority Certified Edition KG

Руководство администратора. Часть 1. Установка и обслуживание
Центра сертификации Aladdin Enterprise Certification Authority

Изделие	33714370.03.01.001
Документ	33714370.03.01.001 32 01-1
Версия	2.3.0
Листов	113
Дата	30.05.2025

АННОТАЦИЯ

Настоящий документ представляет собой первую часть руководства администратора программного средства «Центр сертификатов доступа Aladdin Enterprise Certificate Authority Certified Edition KG» 33714370.03.01.001.

Документ содержит сведения об области применения, составе, основных функциях и комплектности программного средства «Центр сертификатов доступа Aladdin Enterprise Certificate Authority Certified Edition KG».

Документ предназначен для администраторов программного средства «Центр сертификатов доступа Aladdin Enterprise Certificate Authority Certified Edition KG», регламентирующих права доступа субъектов к объектам и введение ограничений на действия пользователей, а также на изменение условий эксплуатации, состава и конфигурации программных и программно-аппаратных средств.

Руководство также определяет порядок подготовки и установки программного комплекса «Центр сертификации Aladdin Enterprise Certification Authority» из состава программного средства «Центр сертификатов доступа Aladdin Enterprise Certificate Authority Certified Edition KG». Перед эксплуатацией программного комплекса «Центр сертификации Aladdin Enterprise Certification Authority» рекомендуется внимательно ознакомиться с настоящим руководством.

Инструкции по установке стороннего программного обеспечения приведены в ознакомительных целях, для получения более точной информации рекомендуем ознакомиться с актуальной инструкцией по установке и настройке продуктов на официальных сайтах производителей.

Характер изложения материала данного руководства предполагает, что вы знакомы с операционными системами семейства Linux, на которых работает программа и владеете базовыми навыками администрирования для работы в них.

Документ рекомендован как для последовательного, так и для выборочного изучения.

СОДЕРЖАНИЕ

Аннотация.....	2
Содержание	3
1 Введение	6
1.1 Область применения	6
1.2 Состав программного средства	6
1.3 Основные функции программного средства.....	8
1.4 Комплект поставки программного средства.....	12
1.5 Имя пакета компонентов поставки	13
1.6 Роли управления.....	13
1.7 Режимы функционирования программы	16
1.8 Действия по безопасной установке и настройке программного средства.....	16
1.9 Действия по реализации функций безопасности среды функционирования программного средства	16
2 Условия выполнения программы.....	18
2.1 Требования к программному обеспечению	18
2.1.1 Требования к среде функционирования серверной части центра сертификации.....	18
2.1.2 Требования к среде функционирования клиентской части центра сертификации	19
2.2 Требования к аппаратным средствам.....	19
3 Действия по приёму программного средства	21
3.1 Проверка комплектности	21
3.2 Контроль целостности установочных пакетов	21
4 Подготовка к установке программы	22
4.1 Подготовка среды функционирования программы	23
4.2 Подготовка среды функционирования с ОС РЕД ОС.....	23
4.2.1 Подключение репозитория и установка зависимостей.....	23
4.2.2 Установка среды исполнения Java	24
4.2.3 Установка и настройка СУБД.....	24
4.2.4 Установка веб-сервера	27
4.3 Подготовка среды функционирования с ОС Astra Linux Special Edition	28
4.3.1 Подключение репозитория и установка зависимостей.....	28
4.3.2 Установка среды исполнения Java	30
4.3.3 Установка и настройка СУБД.....	30
4.3.4 Установка веб-сервера	33
4.4 Подготовка среды функционирования с ОС Альт Сервер	34
4.4.1 Подключение репозитория и установка зависимостей.....	34
4.4.2 Установка среды исполнения Java	34
4.4.3 Установка и настройка СУБД.....	35
4.4.4 Установка веб-сервера	37

4.5 Подготовка среды функционирования с ОС «Platform V SberLinux OS Server»	38
4.5.1 Установка среды исполнения Java	38
4.5.2 Установка и настройка СУБД	39
4.5.3 Установка веб-сервера	42
4.6 Установка веб-сервера Cppnginx	42
4.7 Установка JC-WebClient	43
4.1 Установка ПО «Рутокен Плагин» и его расширения	43
4.2 Установка программного средства «Криптографический модуль Aladdin JCP»	44
5 Установка программы	45
5.1 Распаковка инсталляционного комплекта программы	45
5.2 Настройка параметров конфигурации программы	47
5.3 Создание и настройка базы данных	56
5.3.1 Создание и настройка базы данных в автоматическом режиме	56
5.3.2 Создание и настройка базы данных PostgreSQL в ручном режиме	57
5.3.3 Создание и настройка базы данных Jatoba в ручном режиме	58
5.4 Установка программы	59
5.5 Порядок совместной установки компонентов программного средства на одном сервере	61
6 Запуск и остановка программы	64
7 Подключение к веб-интерфейсу	67
7.1 Общие сведения	67
7.2 Установка сертификата администратора инициализации	68
7.3 Подключение к веб-интерфейсу	70
8 Контроль целостности исполняемых файлов программы	72
9 Сбор диагностической информации	73
10 Резервное копирование и восстановление данных программы	75
10.1 Расписание резервного копирования	75
10.2 Восстановление данных из резервной копии	76
11 Восстановление доступа к программе	77
12 Обновление программы	78
13 Удаление программы	81
14 Удаление базы данных Postgres	82
14.1 Удаление базы данных	82
14.2 Удаление пользователя базы данных	82
15 Поиск и устранение неисправностей	83
Приложение 1. Разрешение конфликта при установке СУБД Postgres и PostgresPro	84
Приложение 2. Настройка подключения к внешней СУБД	85
2.1 Настройка на хосте СУБД	85
2.2 Настройка на хосте Центра сертификации Aladdin eCA	86
Приложение 3. Настройка TLS-соединения с СУБД	87

3.1 Настройка СУБД.....	87
3.2 Настройка Центра сертификации Aladdin eCA.....	88
Приложение 4. Развертывание кластера	89
4.1 Развертывание кластера в виртуальной инфраструктуре	89
4.2 Развертывание кластера Центр сертификации Aladdin eCA с помощью переноса контейнеров закрытого ключа основного узла.....	92
4.3 Обновление ПО узлов кластера Aladdin eCA	104
Приложение 5. Настройка взаимодействия с криптопровайдером СКЗИ «КриптоПро CSP».....	105
5.1 Настройка взаимодействия с СКЗИ «КриптоПро CSP»	106
5.2 Индикация об отсутствии связи с СКЗИ «КриптоПро CSP»	107
Перечень документации для ознакомления	109
Обозначения и сокращения.....	110
Термины и определения	111
ЛИСТ РЕГИСТРАЦИИ ИЗМЕНЕНИЙ	113

1 ВВЕДЕНИЕ

1.1 Область применения

Программное средство «Центр сертификатов доступа Aladdin Enterprise Certificate Authority Certified Edition KG» 33714370.03.01.001 (далее по тексту - программное средство или Центр сертификатов доступа) применяется как элемент систем защиты информации автоматизированных (информационных) систем и используется совместно с другими средствами защиты информации при идентификации и строгой аутентификации субъектов ¹ и объектов доступа ² в автоматизированной (информационной) системе.

1.2 Состав программного средства

Центр сертификатов доступа включает:

- Программный комплекс «Центр сертификации Aladdin Enterprise Certification Authority» 33714370.03.01.003 (далее по тексту - программа или Центр сертификации Aladdin eCA), состоящий из следующих программных компонентов:
 - Программный компонент «Серверная часть Центра сертификации» 33714370.03.01.004.
Программный компонент реализует функции программного средства, для выполнения которых оно предназначено в заданных условиях применения, в части формирования идентификационной информации, необходимой для выпуска сертификатов безопасности цифровых сертификатов) (далее - сертификаты), выпуска и обслуживания сертификатов, приостановки и/или возобновления действия сертификатов, предоставления информации о сертификатах и их статусах.
 - Программный компонент «Клиентская часть Центра сертификации» 33714370.03.01.005.
Программный компонент реализует интерфейс (веб-интерфейс), с помощью которого обеспечивается взаимодействие пользователя и программного компонента «Серверная часть Центра сертификации» 33714370.03.01.004.
- Программный комплекс «Центр валидации Aladdin Enterprise Validation Authority» 33714370.03.01.006 (далее по тексту - Центр валидации Aladdin eVA), состоящий из следующих программных компонентов:
 - Программный компонент «Серверная часть Центра валидации» 33714370.03.01.007.
Программный компонент реализует функции программного средства, для выполнения которых оно предназначено в заданных условиях применения, в части предоставления информации о сертификатах и их статусах.
 - Программный компонент «Клиентская часть Центра валидации» 33714370.03.01.008.
Программный компонент реализует интерфейс (веб-интерфейс), с помощью которого обеспечивается взаимодействие пользователя и программного компонента «Серверная часть Центра валидации» 33714370.03.01.007.
- Программный комплекс «Центр регистрации Aladdin Enterprise Registration Authority» 33714370.03.01.009 (далее по тексту - Центр регистрации Aladdin eRA), состоящий из следующих программных компонентов:

¹ Субъектами доступа могут являться как физические лица (пользователи), так и ресурсы автоматизированной информационной системы, а также вычислительные процессы, инициирующие получение и получающие доступ от имени пользователей, программ, средств вычислительной техники и других программно-аппаратных устройств информационно-телекоммуникационной инфраструктуры.

² Объект доступа представляет собой одну из сторон информационного взаимодействия, предоставляющую доступ.

- Программный компонент «Серверная часть Центра регистрации» 33714370.03.01.010.

Программный компонент реализует функции программного средства, для выполнения которых оно предназначено в заданных условиях применения, в части формирования идентификационной информации, необходимой для выпуска сертификатов, выпуска и обслуживания сертификатов.

- Программный компонент «Клиентская часть Центра регистрации» 33714370.03.01.011.

Программный компонент реализует интерфейс (веб-интерфейс), с помощью которого обеспечивается взаимодействие пользователя и программного компонента «Серверная часть Центра регистрации» 33714370.03.01.010.

- Программное средство «Утилита контроля целостности 2.0» 33714370.03.01.012.

Программное средство предназначена для контроля целостности исполняемых файлов программных комплексов из состава Центра сертификатов доступа.

- Программное средство «Криптографический модуль Aladdin JCP» 33714370.03.10.001. Программное средство предназначено для создания ключевых пар (открытый и закрытый ключи) центров сертификации инфраструктуры открытых ключей, пользователей и средств вычислительной техники (устройств), генерации и проверки цифровой подписи.

Центр сертификатов доступа может взаимодействовать со следующими криптопровайдерами:

- Средство криптографической защиты информации «КриптоПро CSP» версии 5.0 R3 KC1 (исполнение 1-Base) ЖТЯИ.00101-03 или версии 5.0 R3 KC2 (исполнение 2-Base) ЖТЯИ.00102-03 ¹.

Средство криптографической защиты информации (далее - СКЗИ) предназначено для создания, хранения и удаления ключевых пар (открытый и закрытый ключи) центров сертификации инфраструктуры открытых ключей, создания ключевых пар (открытый и закрытый ключи) пользователей и средств вычислительной техники (устройств), генерации и проверки цифровой подписи, а также для идентификации, аутентификации, шифрования и имитозащиты TLS-соединений.

- Программно-аппаратный криптографический модуль «КриптоПро HSM» версии 2.0 R3 ЖТЯИ.00096-01 (исполнение 1K, комплектация 1 или 2) ².

Программно-аппаратный криптографический модуль (далее - ПАКМ) предназначен для создания, хранения и удаления ключевых пар (открытый и закрытый ключи) центров сертификации инфраструктуры открытых ключей, создания ключевых пар (открытый и закрытый ключи) пользователей и средств вычислительной техники (устройств), а также генерации и проверки цифровой подписи.

¹ СКЗИ «КриптоПро CSP» не является обязательным программным средством, не входит в состав и комплект поставки Центра сертификатов доступа и, при необходимости, приобретается заказчиком самостоятельно. Порядок настройки взаимодействия Центра сертификации Aladdin eCA с СКЗИ «КриптоПро CSP» описан в Приложении 5. Порядок настройки взаимодействия Центра регистрации Aladdin eRA с СКЗИ «КриптоПро CSP» описан в документе «Центр сертификатов доступа Aladdin Enterprise Certificate Authority Certified Edition. Руководство администратора. Часть 5. Центр регистрации Aladdin Enterprise Registration Authority». Порядок настройки взаимодействия Центра валидации Aladdin eVA с СКЗИ «КриптоПро CSP» описан в документе «Центр сертификатов доступа Aladdin Enterprise Certificate Authority Certified Edition. Руководство администратора. Часть 4. Центр валидации Aladdin Enterprise Validation Authority».

² ПАКМ «КриптоПро HSM» не является обязательным программным средством, не входит в состав и комплект поставки Центра сертификатов доступа и, при необходимости, приобретается заказчиком самостоятельно.

1.3 Основные функции программного средства

Центр сертификатов доступа реализует следующие функции:

- Формирование идентификационной информации, необходимой для выпуска сертификатов безопасности (цифровых сертификатов) пользователей и средств вычислительной техники (устройств) (далее по тексту - СБТ) на основе данных, полученных при первичной идентификации непосредственно от пользователей и СБТ через заявку на выпуск сертификатов, либо полученных от доменной службы каталогов или уполномоченных пользователей. Первичная идентификация пользователей и СБТ в программном средстве завершается созданием для них субъектов. Идентификационная информация, необходимая для выпуска сертификатов, представляет собой атрибуты субъекта, значения которых записываются в поля сертификатов, создаваемых для данного субъекта.

- Выпуск и обслуживание сертификатов безопасности (цифровых сертификатов) пользователей и средств вычислительной техники (устройств), в том числе:

- Создание ключевых пар (открытый и закрытый ключи) пользователей и СБТ.

Создание ключевых пар для пользователей и средств вычислительной техники (устройств) выполняется при формировании для них сертификатов с закрытым ключом (PKCS#12) ¹.

- Формирование сертификатов для пользователей и СБТ.

В программном средстве реализовано формирование сертификатов для пользователей и СБТ:

- С закрытым ключом (PKCS#12).
- На основании запроса PKCS#10 ².

- Формирование заявок на выпуск сертификатов для пользователей и СБТ.

В программном средстве реализовано:

- Создание заявок пользователями с ролями «Администратор» и «Оператор» через программный компонент «Клиентская часть Центра регистрации» и программный интерфейс программного компонента «Серверная часть Центра регистрации»
- Создание заявок пользователем с ролью «Получатель сертификата» через программный компонент «Клиентская часть Центра регистрации» и программный интерфейс программного компонента «Серверная часть Центра регистрации», включая заявки, создаваемые через программный интерфейс по протоколу WS-Trust X.509v3 Token Enrollment Extensions (WSTEP) ³.
- Создание заявок через программный интерфейс программного компонента «Серверная часть Центра регистрации» по протоколу Simple Certificate Enrollment Protocol (SCEP) ⁴.
- Автоматическое создание заявок на основании запросов PKCS#10 из локального или сетевого каталога в соответствии с настройками Offline-выпуска.

- Выдача сертификатов для их использования владельцами.

Выдача сертификатов для их использования владельцами доступна:

- Путем их экспорта за пределы программного средства пользователями с ролями «Администратор» или «Оператор».
- Путем их экспорта за пределы программного средства инициатором заявки на выпуск сертификата, если по данной заявке успешно выпущен сертификат.
- Путем их автоматического экспорта за пределы программного средства в локальный или сетевой каталог в соответствии с настройками Offline-выпуска.

¹ В соответствии с документом «RFC 7292. PKCS #12: Personal Information Exchange Syntax v1.1».

² В соответствии с документом «RFC 2986. PKCS #10: Certification Request Syntax Specification Version 1.7».

³ В соответствии с документом «OASIS WS-Trust 1.3. WS-Trust X.509 Token Profile (WSTEP)».

⁴ В соответствии с документом «RFC 8894. Simple Certificate Enrollment Protocol».

- Централизованное автоматическое (автоматизированное) отслеживание актуальности (с уведомлением владельцев о сроках действия) сертификатов.

Уведомление владельцев о сроках действия их сертификатов выполняется по электронной почте. По умолчанию программное средство уведомляет владельца сертификата в случае, если срок его действия истекает через 30 суток, через 7 суток или через 1 сутки. В программном средстве доступно формирование шаблонов рассылок уведомлений владельцев о сроках действия их сертификатов. Для каждого шаблона рассылки доступно указание времени, отслеживаемого до окончания действия сертификата, а также текста отправляемого уведомления.

- Выпуск и обслуживание сертификатов центров сертификации инфраструктуры открытых ключей, в том числе:

- Создание, экспорт, импорт и удаление ключевой пары (открытый и закрытый ключи) центра сертификации (корневого и/или подчиненного).

Создание ключевой пары центра сертификации (корневого и/или подчиненного) выполняется при создании собственного центра сертификации в программном средстве. В программном средстве доступно создание центра сертификации (корневого и/или подчиненного) на основании импортированного контейнера закрытого ключа PKCS #12 центра сертификации, содержащего его ключевую пару. Для центра сертификации доступен экспорт ключевой пары за пределы программного средства, если его ключевая пара уже не экспортирована за пределы программного средства, и для данной ключевой пары при ее создании не был установлен запрет на экспорт. При экспорте ключевая пара центра сертификации удаляется из программного средства. Для центра сертификации, ключевая пара которого экспортирована за пределы программного средства, доступна возможность импорта его ключевой пары в программное средство. Удаление ключевой пары центра сертификации выполняется при удалении данного центра сертификации из программного средства, если его ключевая пара уже не экспортирована за пределы программного средства.

- Создание, импорт, просмотр, экспорт и удаление корневого (самоподписанного) сертификата центра сертификации.

Создание корневого (самоподписанного) сертификата центра сертификации выполняется при создании в программном средстве собственного корневого центра сертификации. Импорт корневого (самоподписанного) сертификата центра сертификации выполняется при создании в программном средстве корневого центра сертификации на основании импортированного контейнера закрытого ключа PKCS #12 корневого центра сертификации. В программном средстве доступен просмотр значений полей корневого (самоподписанного) сертификата центра сертификации. Для каждого корневого центра сертификации доступен импорт его самоподписанного сертификата. Удаление корневого (самоподписанного) сертификата центра сертификации выполняется при удалении данного корневого центра сертификации.

- Создание, просмотр, экспорт и удаление запроса на сертификат центра сертификации в вышестоящий центр сертификации.

Создание запроса на сертификат центра сертификации в вышестоящий центр сертификации выполняется при создании в программном средстве подчиненного центра сертификации. Запрос на сертификат центра сертификации в вышестоящий центр сертификации доступен для просмотра средствами из состава операционной системы (среды функционирования) (далее - ОС). Запрос на сертификат центра сертификации в вышестоящий центр сертификации доступен для экспорта за пределы программного средства. Удаление запроса на сертификат центра сертификации в вышестоящий центр сертификации выполняется при удалении в программном средстве данного подчиненного центра сертификации.

- Создание на основании запроса, импорт, просмотр, экспорт, удаление и отзыв сертификата для подчиненного центра сертификации.

Создание сертификата для подчиненного центра сертификации на основании запроса выполняется в вышестоящем центре сертификации при подписании запроса на сертификат данного подчиненного центра сертификации. Импорт сертификата для подчиненного центра сертификации выполняется при создании в программном средстве подчиненного центра сертификации на основании импортированного контейнера закрытого ключа PKCS #12 подчиненного центра сертификации. В программном средстве доступен просмотр значений полей созданного сертификата для подчиненного центра сертификации. Сертификат для подчиненного центра сертификации доступен для экспорта за пределы программного средства как отдельно, так и в составе его цепочки сертификатов. Удаление сертификата подчиненного центра сертификации выполняется при удалении данного центра сертификации из программного средства. В вышестоящем центре сертификации, подписавшем запрос на сертификат подчиненного центра сертификации, доступен отзыв сертификата данного подчиненного центра сертификации.

- Приостановка и/или возобновление действия пользователей и CBT, в том числе:

- Блокирование, возобновление действия, отзыв и перевыпуск сертификатов.

Блокирование, возобновление действия и отзыв сертификатов выполняется путем формирования списка (основного и разностного) отозванных сертификатов. В данный список программным средством заносятся заблокированные и отозванные сертификаты. Операция блокирования сертификата обратима путем возобновления действия данного сертификата. Операция отзыва сертификата необратима. В программном средстве доступен повторный выпуск сертификатов пользователей и CBT на основании ранее использованной идентификационной информации.

- Формирование, экспорт и публикация списка отозванных сертификатов.

Формирование списка отозванных сертификатов выполняется автоматически с задаваемой пользователем с ролью «Администратор» периодичностью и/или при любом изменении статуса сертификата. В программном средстве доступен экспорт списка отозванных сертификатов. При каждом формировании списка отозванных сертификатов безопасности выполняется его публикация в зарегистрированные точки распространения. В программном средстве доступна публикация списка отозванных сертификатов и сертификатов центров сертификации в точки распространения центров валидации, создаваемых в Центре валидации Aladdin eVA, и точки распространения доменной службы каталогов.

- Предоставление информации о сертификатах центров сертификации, пользователей и CBT, а также информации об их статусах, в том числе:

- Формирование и экспорт реестра сертификатов.

В программном средстве реализовано формирование реестра сертификатов, содержащего значения полей всех созданных сертификатов. При экспорте реестра сертификатов доступен выбор критериев, которым должны соответствовать сертификаты в экспортируемом реестре.

- Проверка статусов сертификатов на основании данных, опубликованных в точке распространения.

Программное средство позволяет экспортировать опубликованные списки отозванных сертификатов и сертификаты центров сертификации из точек распространения, реализованных программным средством.

- Проверка статусов сертификатов в режиме реального времени.

Программное средство позволяет выполнять проверку статусов сертификатов в режиме реального времени по протоколу Online Certificate Status Protocol (OCSP) ¹.

¹ В соответствии с документом «RFC 6960. X.509 Internet Public Key Infrastructure Online Certificate Status Protocol - OCSP».

Центр сертификатов доступа выпускает сертификаты в следующих форматах:

- Формат сертификата открытого ключа X.509v3 ¹.
Сертификат включает в себя следующие данные:
 - Версия сертификата.
 - Серийный номер сертификата.
 - Идентификатор алгоритма подписи сертификата.
 - Отличительное имя издателя сертификата.
 - Период действия сертификата.
 - Отличительное имя субъекта.
 - Информация об открытом ключе, включающая алгоритм открытого ключа и сам открытый ключ.
 - Расширения сертификата, включая следующие возможные поля:
 - Идентификатор ключа издателя сертификата.
 - Идентификатор ключа субъекта.
 - Идентификаторы использования ключа.
 - Политики сертификата.
 - Альтернативное имя субъекта.
 - Альтернативное имя издателя сертификата.
 - Базовые ограничения.
 - Точки распространения списков отзыва.
 - Доступ к информации о центрах сертификации.
 - Идентификаторы расширенного использования ключа.
 - Подпись сертификата.
- Формат списка отозванных сертификатов безопасности (CRL) ².
Список отозванных сертификатов включает в себя следующие данные:
 - Версия CRL.
 - Отличительное имя издателя CRL.
 - Дата и время издания текущего CRL.
 - Дата и время издания следующего CRL.
 - Расширения CRL, включая следующие возможные поля:
 - Идентификатор ключа издателя CRL.
 - Номер CRL.
 - Перечень отозванных сертификатов, где для каждого сертификата указаны:
 - Серийный номер.
 - Дата и время отзыва.
 - Причина отзыва (может отсутствовать).
 - Алгоритм подписи CRL.
 - Подпись CRL.

¹ Формат определяется документом «ITU-T Recommendation X.509 (10/2019). Information technology - Open Systems Interconnection - The Directory: Public-key and attribute certificate frameworks».

² Формат определяется документом «ITU-T Recommendation X.509 (10/2019). Information technology - Open Systems Interconnection - The Directory: Public-key and attribute certificate frameworks».

- Формат контейнера закрытого ключа PKCS #12 ¹.

Контейнеры закрытого ключа включают в себя следующие данные:

- Цепочка сертификатов владельца закрытого ключа.
- Закрытый ключ.

Центр сертификатов доступа реализовывает следующие криптографические алгоритмы:

- Алгоритмы генерации ключевой пары:
 - RSA с длинами ключей 1024, 1536, 2048, 3072, 4096, 6144 и 8192 бит.
 - ECDSA с длинами ключей 256, 384 и 521 бит.
 - ГОСТ Р 34.10-2012 с ключом 256 или 512 бит.
- Алгоритмы генерации цифровой подписи:
 - RSA PKCS#1 Ver 1.5 (длины ключей: 1024, 1536, 2048, 3072, 4096, 6144 и 8192 бит; хэш-алгоритмы: SHA256, SHA384, SHA512).
 - ECDSA (длины ключей: 256, 384, 521 бит; хэш-алгоритмы: SHA256, SHA384, SHA512).
 - ГОСТ Р 34.10-2012 с ключом 256 или 512 бит (хэш-алгоритм: ГОСТ Р 34.11-2012 с длиной хэш-кода 256 или 512 бит).

1.4 Комплект поставки программного средства

Программное средство «Центр сертификатов доступа Aladdin Enterprise Certificate Authority Certified Edition KG» 33714370.03.01.001 поставляется в следующей комплектации:

- Программный комплекс «Центр сертификации Aladdin Enterprise Certification Authority» 33714370.03.01.003 на носителе оптической записи (rpm-пакет и deb-пакет).
- Программный комплекс «Центр валидации Aladdin Enterprise Validation Authority» 33714370.03.01.006 на носителе оптической записи (rpm-пакет и deb-пакет).
- Программный комплекс «Центр регистрации Aladdin Enterprise Registration Authority» 33714370.03.01.009 на носителе оптической записи (rpm-пакет и deb-пакет).
- Контрольные суммы исполняемых файлов программного средства «Центр сертификатов доступа Aladdin Enterprise Certificate Authority Certified Edition KG».
- Контрольные суммы установочных пакетов (дистрибутивов) программного средства «Центр сертификатов доступа Aladdin Enterprise Certificate Authority Certified Edition KG»
- Эксплуатационная документация в составе:
 - «Центр сертификатов доступа Aladdin Enterprise Certificate Authority Certified Edition KG. Формуляр. Часть 1. Общие сведения» 33714370.03.01.001 30 01-1.
 - «Центр сертификатов доступа Aladdin Enterprise Certificate Authority Certified Edition KG. Формуляр. Часть 2. Свидетельства о приёме, упаковке и маркировке» 33714370.03.01.001 30 01-2.
 - «Центр сертификатов доступа Центр сертификатов доступа Aladdin Enterprise Certificate Authority Certified Edition KG. Описание применения» 33714370.03.01.001 31 01.
 - «Центр сертификатов доступа Aladdin Enterprise Certificate Authority Certified Edition KG. Руководство администратора. Часть 1. Установка и обслуживание Центра сертификации Aladdin Enterprise Certification Authority» 33714370.03.01.001 32 01-1.

¹ Формат определяется документом «RFC 7292. PKCS #12: Personal Information Exchange Syntax v1.1»

- «Центр сертификатов доступа Aladdin Enterprise Certificate Authority Certified Edition KG. Руководство администратора. Часть 2. Функции управления Центра сертификации Aladdin Enterprise Certification Authority» 33714370.03.01.001 32 01-2.
 - «Центр сертификатов доступа Aladdin Enterprise Certificate Authority Certified Edition KG. Руководство администратора. Часть 3. Описание REST API Центра сертификации Aladdin Enterprise Certification Authority» 33714370.03.01.001 32 01-3.
 - «Центр сертификатов доступа Aladdin Enterprise Certificate Authority Certified Edition KG. Руководство администратора. Часть 4. Центр валидации Aladdin Enterprise Validation Authority» 33714370.03.01.001 32 01-4.
 - «Центр сертификатов доступа Aladdin Enterprise Certificate Authority Certified Edition KG. Руководство администратора. Часть 5. Центр регистрации Aladdin Enterprise Registration Authority» 33714370.03.01.001 32 01-5.
 - «Центр сертификатов доступа Aladdin Enterprise Certificate Authority Certified Edition KG. Руководство администратора. Часть 6. Описание методов REST API Центра регистрации Aladdin Enterprise Registration Authority» 33714370.03.01.001 32 01-6.
 - «Центр сертификатов доступа Aladdin Enterprise Certificate Authority Certified Edition KG. Руководство оператора» 33714370.03.01.001 34 01.
 - «Центр сертификатов доступа Aladdin Enterprise Certificate Authority Certified Edition KG. Руководство получателя сертификатов» 33714370.03.01.001 34 02.
- Потребительская упаковка.

1.5 Имя пакета компонентов поставки

Имя пакета компонентов поставки представлено в формате:

• <name>	- название компонента;
• <major_version>	- мажорная версия компонента;
• <minor_version>	- минорная версия компонента;
• <release>	- номер релиза компонента;
• <build_number>	- номер сборки;
• <arch>	- целевая архитектура.

1.6 Роли управления

В Центре сертификации Aladdin eCA определены следующие роли пользователей:

- Оператор

Пользователь с ролью «Оператор» имеет доступ к Центру сертификации Aladdin eCA через веб-интерфейс и программный интерфейс API. Пользователь с данной ролью обладает правами на работу с субъектами группы, над которой он может осуществлять свои ролевые права, и принадлежащими им сертификатами (выпуск, отзыв, приостановка и возобновление сертификата), имеет полномочия запуска обновления списка субъектов из ресурсной системы. Для конкретного «Оператора» можно определить перечень субъектов, над которыми он может осуществлять свои ролевые права, а также перечень групп субъектов, над элементами которых он может осуществлять свои ролевые права.

- Администратор

Пользователь с ролью «Администратор» имеет неограниченные права доступа к ОС и серверу, на котором развёрнут Центр сертификации Aladdin eCA, а также доступ через веб-интерфейс или программный интерфейс API к функциональным задачам и к функциям управления учетными записями. Все учётные записи могут быть созданы, отредактированы, удалены или заблокированы только пользователем с ролью «Администратор».

Доступные действия пользователей в соответствии с назначенными ролями приведены в таблице 1.

Таблица 1 - Полномочия пользователей в соответствии с назначенной ролью

Тип действия, осуществляемого пользователем, над объектом программы	Возможные роли пользователей	
	Оператор	Администратор
Установка или обновление программы	-	+
Установка лицензии на программу	-	+
Внесение изменений в конфигурационную информацию лицензии на программу	-	+
Просмотр информации о лицензии на ПО	-	+
Инициализация центра сертификации	-	+
Чтение конфигурационной информации о планах архивации в автоматическом режиме из базы данных	-	+
Внесение изменений в конфигурационную информацию о планах архивации в автоматическом режиме из базы данных	-	+
Чтение конфигурационной информации о уведомлениях об истечении срока действия сертификата	-	+
Внесение изменений в конфигурационную информацию о уведомлениях об истечении срока действия сертификата	-	+
Просмотр журнала событий	-	+
Архивация журнала событий	-	+
Экспорт журнала событий	-	+
Просмотр списка сертификатов центра сертификации (свои и подчинённые)	-	+
Импорт и экспорт закрытого ключа центра сертификации	-	+
Удаление сертификата центра сертификации	-	+
Просмотр цепочки сертификатов центра сертификации	-	+
Скачивание цепочки сертификатов центра сертификации	-	+
Скачивание сертификата центра сертификации	-	+
Скачивание сертификата центра сертификации в контейнере #pkcs12	-	+
Подписание запроса на сертификат подчинённого центра сертификации	-	+
Импортирование сертификата центра сертификации (активация центра сертификации)	-	+
Создание сертификатов доступа для полного набора субъектов ресурсных систем	-	+
Создание сертификатов доступа для ограниченного набора субъектов ресурсных систем	+	+
Просмотр списка сертификатов доступа для полного набора субъектов ресурсных систем	-	+
Просмотр списка сертификатов доступа для ограниченного набора субъектов ресурсных систем	+	+

Тип действия, осуществляемого пользователем, над объектом программы	Возможные роли пользователей	
	Оператор	Администратор
Экспорт списка выпущенных сертификатов для полного набора субъектов ресурсных систем	-	+
Экспорт списка выпущенных сертификатов для ограниченного набора субъектов ресурсных систем	+	+
Скачивание сертификата доступа для полного набора субъектов ресурсных систем	-	+
Скачивание сертификата доступа для ограниченного набора доступных субъектов ресурсных систем	+	+
Скачивание сертификата доступа субъекта в контейнере #pkcs12 для полного набора субъектов ресурсных систем	-	+
Скачивание сертификата доступа субъекта в контейнере #pkcs12 для ограниченного набора субъектов ресурсных систем	+	+
Скачивание цепочки сертификатов для полного набора субъектов ресурсных систем	-	+
Скачивание цепочки сертификатов для ограниченного набора субъектов ресурсных систем	+	+
Управление статусом сертификата доступа субъекта для полного набора субъектов ресурсных систем	-	+
Управление статусом сертификата доступа субъекта для ограниченного набора субъектов ресурсных систем	+	+
Создание учётной записи с определением роли для субъектов ресурсных систем	-	+
Управление учётными записями субъектов ресурсных систем	-	+
Просмотр учётных записей субъектов ресурсных систем	-	+
Просмотр ограниченного списка субъектов ресурсных систем	+	+
Просмотр полного списка субъектов ресурсных систем	-	+
Просмотр списка полного набора зарегистрированных ресурсных систем	-	+
Просмотр списка ограниченного набора зарегистрированных ресурсных систем	+	+
Регистрация ресурсных систем	-	+
Обновление полного набора субъектов ресурсных систем	-	+
Обновление ограниченного набора субъектов ресурсных систем	+	+
Просмотр списка зарегистрированных центров валидации	-	+
Управление настройкой обновления списков отозванных сертификатов	-	+
Экспорт списка отозванных сертификатов	-	+
Моментальная публикация списка отозванных сертификатов	-	+
Просмотр шаблонов сертификатов	-	+
Создание нового шаблона сертификата	-	+
Импорт шаблонов сертификатов	-	+
Редактирование созданных шаблонов сертификатов	-	+
Удаление созданных шаблонов сертификатов	-	+
Просмотр идентификаторов расширенного использования ключа	-	+
Просмотр ограниченного набора идентификаторов расширенного использования ключа	+	+

Тип действия, осуществляемого пользователем, над объектом программы	Возможные роли пользователей	
	Оператор	Администратор
Создание пользовательских идентификаторов расширенного использования ключа	-	+
Удаление пользовательских идентификаторов расширенного использования ключа	-	+
Просмотр списка разрешённых издателей	-	+
Управление проверкой издателя	-	+
Перезагрузка веб-сервера	-	+
Контроль целостности исполняемых файлов программы	-	+

1.7 Режимы функционирования программы

Основным режимом функционирования Центра сертификации Aladdin eCA является нормальный режим. В нормальном режиме должны исправно функционировать клиентская и серверная части программы, обеспечивая возможность круглосуточного функционирования, с перерывами на обслуживание (обновление программы).

Функционирование корневого и/или подчинённого Центра сертификации Aladdin eCA предусматривает автономный режим (Stand alone operation) или сетевой режим работы.

Сетевой режим работы Центра сертификации Aladdin eCA обеспечивает возможность кластеризации с целью отказоустойчивости ¹.

1.8 Действия по безопасной установке и настройке программного средства

Установка компонентов Центра сертификатов доступа производится только с диска, получаемого от разработчика, после выполнения действий по приёме поставленных компонентов Центра сертификатов доступа.

Установка (изменение) программного обеспечения компьютеров и локальной вычислительной сети должна осуществляться только в присутствии и под контролем администратора информационной безопасности того технологического участка, в котором эксплуатируется Центр сертификатов доступа.

Настройка Центра сертификатов доступа должна проводиться привилегированным пользователем с ролью «Администратор», допускаемым к установке и настройке Центра сертификатов доступа.

1.9 Действия по реализации функций безопасности среды функционирования программного средства

Для безопасной работы Центра сертификатов доступа в среде ОС должно обеспечиваться:

- Предотвращение несанкционированного доступа к идентификаторам и паролям привилегированных пользователей (администратора).
- Разделение полномочий (ролей) пользователей.
- Порядок обработки, хранения и передачи аутентификационной информации пользователей, созданной Центра сертификатов доступа.
- Срок хранения информации о зарегистрированных событиях безопасности не менее трех месяцев.
- Синхронизация внутренних системных часов информационной системы для регистрации всех событий безопасности в журнале событий.

¹ Справочная информация по разворачиванию кластера Центра сертификации Aladdin eCA приведена Приложении 4 «Разворачивание кластера» настоящего руководства.

- Защита аппаратного обеспечения с функционирующими компонентами Центра сертификатов доступа от внешних воздействий (воздействий окружающей среды, нестабильности электроснабжения, кондиционирования и иных внешних факторов).

2 УСЛОВИЯ ВЫПОЛНЕНИЯ ПРОГРАММЫ

2.1 Требования к программному обеспечению

2.1.1 Требования к среде функционирования серверной части центра сертификации

Среда функционирования серверной части Центра сертификации Aladdin eCA:

- Поддерживаемые ОС:
 - Astra Linux Special Edition версия 1.7, уровень защищённости «Смоленск».
 - Astra Linux Special Edition версия 1.7, уровень защищённости «Воронеж».
 - Astra Linux Special Edition версия 1.7, уровень защищённости «Орел».
 - Astra Linux Special Edition версия 1.8, уровень защищённости «Смоленск».
 - Astra Linux Special Edition версия 1.8, уровень защищённости «Воронеж».
 - Astra Linux Special Edition версия 1.8, уровень защищённости «Орел».
 - РЕД ОС версия 7.3, конфигурация «Сервер».
 - РЕД ОС версия 8, конфигурация «Сервер».
 - ОС Альт 8 СП, релиз 10, вариант исполнения Сервер.
 - ОС Platform V SberLinux OS Server.
- Поддерживаемые СУБД:
 - PostgreSQL из состава ОС.
 - Postgres Pro.
 - Jatoba.
- Поддерживаемая среда исполнения Java:
 - Java Axiom JDK Certified.
 - Open JDK версии 17 и выше из состава поддерживаемых ОС.
- Поддерживаемые веб-серверы:
 - Apache2 из состава ОС.
 - Nginx из состава ОС.
 - Cpnginx ¹.
- Поддерживаемые ресурсные системы:
 - Samba DC.
 - Free IPA.
 - ALD PRO.
 - РЕД АДМ.
 - Microsoft AD.
 - Альт Домен.
- Поддерживаемые криптопровайдеры, обеспечивающие формирование электронной подписи ответов службы OCSP по алгоритму ГОСТ Р 34.10-2012:
 - Программное средство «Криптографический модуль Aladdin JCP» ².

¹ Из состава средства криптографической защиты (далее - СКЗИ) «КриптоПро CSP». СКЗИ «КриптоПро CSP» не является обязательным программным средством, не входит в состав и комплект поставки Центра сертификатов доступа и, при необходимости, приобретается заказчиком самостоятельно.

² Входит в состав программного средства.

- СКЗИ «КриптоПро CSP»¹

2.1.2 Требования к среде функционирования клиентской части центра сертификации

Среда функционирования клиентской части Центра сертификации Aladdin eCA:

- Поддерживаемые ОС:
 - Astra Linux Special Edition версия 1.7, уровень защищённости «Смоленск».
 - Astra Linux Special Edition версия 1.7, уровень защищённости «Воронеж».
 - Astra Linux Special Edition версия 1.7, уровень защищённости «Орел».
 - Astra Linux Special Edition версия 1.8, уровень защищённости «Смоленск».
 - Astra Linux Special Edition версия 1.8, уровень защищённости «Воронеж».
 - Astra Linux Special Edition версия 1.8, уровень защищённости «Орел».
 - РЕД ОС версия 7.3, конфигурация «Сервер».
 - РЕД ОС версия 8, конфигурация «Сервер».
 - ОС Альт 8 СП, релиз 10, вариант исполнения Рабочая станция.
 - ОС Platform V SberLinux OS Server.
- Веб-браузер из состава ОС.
- JC-WebClient (для 64-битных систем)².
- ПО «Рутокен Плагин» и веб-браузерное расширение «Адаптер Рутокен Плагин»³.

2.2 Требования к аппаратным средствам

Минимальные аппаратные требования, необходимые для стабильного функционирования Центра сертификации Aladdin eCA:

- Системные требования, предъявляемые к конфигурации серверного оборудования, зависят от количества выпускаемых сертификатов и количества одновременных обращений к серверу Центра сертификации Aladdin eCA приведены в таблице 2:

- Малое внедрение - до 1000 сертификатов и 5 одновременных соединений.
- Среднее внедрение - до 20000 сертификатов и 15 одновременных соединений.
- Крупное внедрение - до 100000 сертификатов и 50 одновременных соединений.

Таблица 2 - Системные требования, предъявляемые к серверу Центра сертификации Aladdin eCA

Приложение	Системные требования	Тип внедрения		
		Малое внедрение	Среднее внедрение	Крупное внедрение
СУБД	ОЗУ, Гбайт	2	3	4
	Количество ядер процессора, шт.	2	4	4
	Накопитель HDD, Гбайт	6	12	18
Центр сертификации Aladdin eCA	ОЗУ, Гбайт	6	8	16
	Количество ядер процессора, шт.	2	4	6
	Накопитель HDD, Гбайт	40	60	300

¹ СКЗИ «КриптоПро CSP» не является обязательным программным средством, не входит в состав и комплект поставки Центра сертификатов доступа и, при необходимости, приобретается заказчиком самостоятельно. Порядок настройки взаимодействия Центра валидации Aladdin eCA с СКЗИ «КриптоПро CSP» описан в приложении 5 настоящего руководства.

² JC-WebClient обеспечивает выпуск сертификатов на электронных ключах (ключевых носителях) JaCarta. Официальный сайт производителя [JC-WebClient](#).

³ ПО «Рутокен Плагин» через браузерное расширение «Адаптер Рутокен Плагин» обеспечивает выпуск сертификатов на электронных ключах (ключевых носителях) Рутокен. Официальный [сайт производителя](#).

Приложение	Системные требования	Тип внедрения		
		Малое внедрение	Среднее внедрение	Крупное внедрение
ОС	ОЗУ, Гбайт	4	4	4
	Количество ядер процессора, шт.	2	2	2
	Накопитель HDD, Гбайт	20	20	20
Итого:	ОЗУ, Гбайт	12	15	24
	Количество ядер процессора, шт.	6	10	12
	Накопитель HDD, Гбайт	66	92	338

- Устройства взаимодействия с пользователем: Клавиатура и мышь.
- USB 2.0 тип А или совместимые.
- Поддерживаемые модели электронных ключей:
 - JaCarta:
 - JaCarta PKI.
 - JaCarta PRO.
 - JaCarta-2 PKI/ГОСТ.
 - JaCarta-2 ГОСТ.
 - JaCarta-3.
 - Рутокен¹:
 - Рутокен ЭЦП 3.0.
 - Рутокен ЭЦП 2.0.
 - Рутокен ЭЦП 2.0 Flash.
 - Рутокен ЭЦП PKI.

¹ Возможность использования ключевых носителей Рутокен может быть ограничена лицензией.

3 ДЕЙСТВИЯ ПО ПРИЁМКЕ ПРОГРАММНОГО СРЕДСТВА

Приёмка Центра сертификатов доступа предусматривает проверку комплектности и контроль целостности установочных пакетов (дистрибутивов) Центра сертификации Aladdin eCA, Центра регистрации Aladdin eRA и Центра валидации Aladdin eVA

3.1 Проверка комплектности

Проверку комплектности программного средства выполняют путём сверки комплектности поставленного программного средства с комплектностью, указанной в разделе 3 документа «Центр сертификатов доступа Aladdin Enterprise Certificate Authority Certified Edition KG. Формуляр. Часть 1. Общие сведения» 33714370.03.01.001 30 01-1 (далее - Формуляр).

3.2 Контроль целостности установочных пакетов

Расчет КС установочных пакетов (дистрибутивов) программного средства, расположенных на носителе оптической записи из комплекта поставки, выполняется по алгоритму SHA256.

Эталонные КС установочных пакетов (дистрибутивов) программного средства приведены в таблице 2 Формуляра, а также содержатся в следующих файлах на носителе оптической записи из комплекта поставки:

- `aeca-ca_[версия]_[номер сбоки].deb.txt` - для deb-пакета;
- `aeca-ca_[версия]_[номер сбоки].rpm.txt` - для rpm-пакета.
- `aeca-ra_[версия]_[номер сбоки].deb.txt` - для deb-пакета;
- `aeca-ra_[версия]_[номер сбоки].rpm.txt` - для rpm-пакета.
- `aeca-va_[версия]_[номер сбоки].deb.txt` - для deb-пакета;
- `aeca-va_[версия]_[номер сбоки].rpm.txt` - для rpm-пакета.

Для расчета КС установочных пакетов используйте утилиту «Sha256sum - GNU coreutils» из состава ОС. Для расчета КС файла выполните следующую команду с правами суперпользователя:

```
sudo sha256sum /путь к файлу/<имя файла>
```

Сравните рассчитанную КС установочного пакета с эталонной КС, приведенной в таблице 2 Формуляра на программное средство.

Внимание! При нарушении целостности установочного пакета (дистрибутива) дальнейшая установка Центра сертификации Aladdin eCA запрещена.

4 ПОДГОТОВКА К УСТАНОВКЕ ПРОГРАММЫ

При установке Центра сертификации Aladdin eCA осуществляется конфигурирование установленного в среде функционирования веб-сервера, в результате чего для внешнего доступа открывается порт сервера, используемый для подключения по протоколу HTTPS (по умолчанию 443). Изменение порта веб-сервера для подключения к нему по протоколу HTTPS осуществляется путем редактирования конфигурационного файла Центра сертификации Aladdin eCA (см. раздел 5.2 настоящего руководства).

В таблице 3 приведён список портов, которые должны быть открыты в Центре регистрации Aladdin eCA.

Таблица 3 - Таблица сетевого взаимодействия

Порт	Транспорт	Протокол	Назначение	Возможность изменения
443	TCP	TLS/HTTPS	Порт для подключения к веб-интерфейсу Центра сертификации Aladdin eCA, а также для взаимодействия с Центром регистрации Aladdin eRA и Центром валидации Aladdin eVA.	Да
80	TCP	HTTP	С данного порта выполняется переадресация пакетов на порт 443.	Да
389	TCP	LDAP	Порт для взаимодействия с доменной службой каталогов (ресурсной системой) по протоколу LDAP.	Нет
5432	TCP	TCP	Порт для подключения к СУБД.	Да
	TCP	TLS		
514	UDP/TCP	Syslog	Порт для отправки сообщений на Syslog-серверы (порт 514, как правило, используется по умолчанию).	Да
25	TCP	SMTP	Порт для подключения к почтовому серверу (значение 25 задано по умолчанию).	Да

В таблице 4 приведён список портов, которые открывает для локальной передачи данных внутри сервера и использует Центр сертификации Aladdin eCA. Доступ к данным портам для внешних подключений ограничивается автоматически при установке Центра сертификации Aladdin eCA с помощью утилиты «iptables» из состава ОС сервера. Во избежание возникновения ошибок в работе Центра сертификации Aladdin eCA переназначение данных портов запрещено.

Таблица 4 - Таблица входящих сетевых портов

Порт	Транспорт	Протокол	Назначение
1100	TCP	HTTP	Внутренний интерфейс для подключения к внутреннему сервису «certificate-authority-service» (Сервис сертификатов)
1150	TCP	HTTP	Внутренний интерфейс для подключения к внутреннему сервису «storage-service» (Сервис хранения)
1200	TCP	HTTP	Внутренний интерфейс для подключения к внутреннему сервису «templates-service» (Сервис шаблонов)
1250	TCP	HTTP	Внутренний интерфейс для подключения к внутреннему сервису «security-service» (Сервис безопасности)
1300	TCP	HTTP	Внутренний интерфейс для подключения к внутреннему сервису «licenses-service» (Сервис лицензирования)
1350	TCP	HTTP	Внутренний интерфейс для подключения к внутреннему сервису «routes-service» (Сервис маршрутизации)
1400	TCP	HTTP	Внутренний интерфейс для подключения к внутреннему сервису «external-integration-service» (Сервис внешних интеграций)
1450	TCP	HTTP	Внутренний интерфейс для подключения к внутреннему сервису «validation-authority-service» (Сервис валидации)

Порт	Транспорт	Протокол	Назначение
1500	TCP	HTTP	Внутренний интерфейс для подключения к внутреннему сервису «publisher-service» (Сервис публикации)
1550	TCP	HTTP	Внутренний интерфейс для подключения к внутреннему сервису «subjects-service» (Сервис субъектов)
1600	TCP	HTTP	Внутренний интерфейс для подключения к внутреннему сервису «ldap-service» (Сервис синхронизации по LDAP)
1650	TCP	HTTP	Внутренний интерфейс для подключения к внутреннему сервису «logs-service» (Сервис журнализации)
1700	TCP	HTTP	Внутренний интерфейс для подключения к внутреннему сервису «export-service» (Сервис экспорта)
1750	TCP	HTTP	Внутренний интерфейс для подключения к внутреннему сервису «event-delivery-service» (Сервис доставки событий)
1800	TCP	HTTP	Внутренний интерфейс для подключения к внутреннему сервису «settings-service» (Сервис настройки)
1850	TCP	HTTP	Внутренний интерфейс для подключения к внутреннему сервису «x509-provider-service» (Сервис аутентификации по сертификату)
1900	TCP	HTTP	Внутренний интерфейс для подключения к внутреннему сервису «api-gateway-service» (Сервис проксирования)

4.1 Подготовка среды функционирования программы

Порядок подготовки среды функционирования Центра сертификации Aladdin eCA:

- Установка ОС (выполняется в соответствии с документацией производителя).
 - Подключение репозитория и установка зависимостей ОС.
 - Развертывание среды исполнения Java.
 - Установка СУБД.
 - Установка веб-сервера.
 - Установка программного средства «Криптографический модуль Aladdin JCP» (для использования алгоритмов ГОСТ Р 34.10-2012).
 - Установка СКЗИ «КриптоПро CSP» (для использования алгоритмов ГОСТ Р 34.10-2012 и RSA).
- Порядок установки и настройки СКЗИ «КриптоПро CSP» представлен в Приложении 5. Установка и настройка СКЗИ «КриптоПро CSP» могут быть выполнены после установки Центра сертификации Aladdin eCA в процессе его эксплуатации.

При использовании СКЗИ «КриптоПро CSP» канал взаимодействия клиентской и серверной части Центра сертификации Aladdin eCA должен быть организован по протоколу TLS ГОСТ, а также должна обеспечиваться TLS-аутентификация пользователей в Центре сертификации Aladdin eCA с использованием отечественных криптографических алгоритмов. Для этого в качестве веб-сервера должен использоваться веб-сервер **Cpnginx** из состава СКЗИ «КриптоПро CSP». Установка веб-сервера выполняется после установки СКЗИ «КриптоПро CSP». Порядок установки веб-сервера **Cpnginx** из состава СКЗИ «КриптоПро CSP» приведен в подразделе 0. После установки веб-сервера необходимо установить на СКЗИ «КриптоПро CSP» серверную лицензию, обеспечивающую возможность использования СКЗИ «КриптоПро CSP» в качестве TLS-сервера.

4.2 Подготовка среды функционирования с ОС РЕД ОС

4.2.1 Подключение репозитория и установка зависимостей

Для РЕД ОС репозитории настроены по умолчанию для скачивания из сети Интернет. Для проверки доступности и готовности к дальнейшим командам следует установить необходимые пакеты из состава ОС, выполнив команды:

```
sudo dnf install tar unzip iptables
```

Если доступ к сети Интернет отсутствует, зависимости возможно установить с USB-носителя из комплекта поставки ОС выполнив следующие действия:

- Перейдите в каталог USB-носителя.
- Выполните следующую команду с правами суперпользователя:

```
sudo dnf install tar unzip iptables
```

4.2.2 Установка среды исполнения Java

4.2.2.1 Установка Axiom JDK Certified

Для установки Axiom JDK Certified воспользуйтесь [инструкцией с официального сайта производителя](#).

Внимание! В РЕД ОС Axiom JDK Certified версии 21 работает некорректно. В результате установка Центра сертификации Aladdin eCA прерывается. Проблема была выявлена в Axiom JDK Certified версии 21.0.4+10.

4.2.2.2 Установка OpenJDK

Для установки OpenJDK воспользуйтесь инструкцией по установке пакета с официального сайта РЕД ОС:

- [инструкция для РЕД ОС 7.3](#);
- [инструкция для РЕД ОС 8](#).

Внимание! В РЕД ОС OpenJDK определенных версий работает некорректно. В результате установка Центра сертификации Aladdin eCA прерывается. Проблема была выявлена для OpenJDK версий 17.0.15.0.6 и 21.0.7.0.6.

4.2.3 Установка и настройка СУБД

Выполните установку и настройку одной из нижеприведённых СУБД:

- PostgreSQL из состава ОС.
- Postgres Pro.
- Jatoba.

Порядок разрешения конфликтов между СУБД PostgreSQL приведен в Приложении 1.

Порядок настройки взаимодействия с СУБД, размещенной на отдельном узле, приведен в Приложении 2.

Центр сертификации Aladdin eCA может быть настроен на взаимодействие с СУБД по протоколу TLS.

Программа не будет функционировать, если ему не удалось установить TLS-соединение с СУБД. Порядок настройки TLS-соединения с СУБД приведен в Приложении 3.

4.2.3.1 Установка СУБД PostgreSQL¹

Порядок установки СУБД PostgreSQL:

- Установите последнюю доступную версию СУБД PostgreSQL, выполнив команду:

```
sudo dnf install postgresql-server
```

- Выполните установку последней доступной версии пакета `postgresql-contrib`, выполнив команду:

```
sudo dnf install postgresql-contrib
```

- Произведите инициализацию БД, выполнив команду:

```
sudo postgresql-setup --initdb
```

¹ Подробное описание приведено в официальной документации на PostgreSQL, размещённой по адресу <https://postgrespro.ru/docs>.

В случае ошибки `Data directory in '/var/lib/pgsql/data' is not empty...` следует очистить каталог командой ниже и повторить инициализацию БД.

```
sudo rm -rf /var/lib/pgsql/data
```

- Запустите PostgreSQL, выполнив команду:

```
sudo systemctl start postgresql
```

- Добавьте запуск PostgreSQL в автозагрузку, выполнив команду:

```
sudo systemctl enable postgresql
```

- Отредактируйте файл `/var/lib/pgsql/data/postgresql.conf`¹ под администратором - установите число подключений `max_connections` в значение `1000`².

- Отредактируйте файл `/var/lib/pgsql/data/pg_hba.conf`³ под администратором - измените параметры для успешного локального подключения пользователя к базе данных:

```
host all all 127.0.0.1/32 ident      на host all all 127.0.0.1/32 password
```

```
host all all ::1/128 ident         на host all all ::1/128 password
```

- Выполните перезапуск СУБД PostgreSQL для вступления изменений в силу, выполнив команду:

```
sudo systemctl restart postgresql
```

4.2.3.2 Установка СУБД Postgres Pro⁴

Порядок установки СУБД Postgres Pro:

- Загрузите скрипт для добавления репозитория, выполнив команду⁵:

```
wget --user [ключ] --password='' https://repo.postgrespro.ru/std/std-16/keys/pgpro-repo-add.sh
```

- Запустите скрипт, выполнив команду с правами суперпользователя (root или sudo):

```
sudo sh pgpro-repo-add.sh
```

- Обновите список пакетов, выполнив команду с правами суперпользователя (root или sudo):

```
sudo dnf update
```

- Установите Postgres Pro, выполнив команду с правами суперпользователя (root или sudo):

```
sudo dnf install postgrespro-std-16
```

- Отредактируйте файл `/var/lib/pgpro/std-16/data/postgresql.conf`⁶ под администратором - установите число подключений `max_connections` в значение `1000`⁷.

¹ Расположение файла может отличаться, для поиска можно использовать команду `sudo find / -type f -name postgresql.conf`

² Значение `max_connections` равное `1000` является рекомендуемым, при необходимости можно установить и большее значение.

³ Расположение файла может отличаться, для поиска можно использовать команду `sudo find / -type f -name pg_hba.conf`

⁴ Подробное описание приведено в официальной документации на Postgres Pro, размещённой по адресу <https://postgrespro.ru/docs>

⁵ Команды ниже приведена для Postgres Pro версии 16.

⁶ Расположение файла указано для 16 версии Postgres Pro, для поиска можно использовать команду `sudo find / -type f -name postgresql.conf`

⁷ Значение `max_connections` равное `1000` является рекомендуемым, при необходимости можно установить и большее значение.

- Отредактируйте файл `/var/lib/pgpro/std-16/data/pg_hba.conf1` под администратором - измените параметры для успешного локального подключения пользователя к базе данных:

```
host all all 127.0.0.1/32 ident на host all all 127.0.0.1/32 password
host all all ::1/128 ident на host all all ::1/128 password
```

- При отсутствии создайте символические ссылки на утилиты `psql` и `pg_dump`, выполнив команды с правами суперпользователя (root или sudo):

```
sudo ln -s /opt/pgpro/std-16/bin/psql /usr/bin/psql
sudo ln -s /opt/pgpro/std-16/bin/pg_dump /usr/bin/pg_dump
```

- Перезапустите Postgres Pro, выполнив команду с правами суперпользователя (root или sudo):

```
sudo systemctl restart postgrespro-std-16.service
```

4.2.3.3 Установка СУБД Jatoba²

- Создайте каталог `/localrepo`, выполнив команду:

```
sudo mkdir /localrepo
```

- В каталог `/localrepo` скопируйте необходимые файлы для установки СУБД Jatoba.

Внимание! Требуется скопировать полную структуру файлов и каталогов из дистрибутива. Также допускается установка с оптического диска напрямую. В этом случае, пользователю не требуется копировать файлы, а вместо каталога `/localrepo` во всех шагах далее указывать соответствующий путь до носителя и директорию репозитория СУБД на носителе для соответствующей ОС.

- Дистрибутив СУБД Jatoba содержит:

- каталог `/packages`;
- каталог `/repodata`;
- файл ключа `RPM-GPG-KEY-Jatoba`.

- Проверьте результат копирования всех файлов дистрибутива, перейдя в каталог `/localrepo` и выполнив команду:

```
ls -l
```

- Установите открытый ключ репозитория командой:

```
sudo rpm --import /localrepo/RPM-GPG-KEY-Jatoba
```

- Создайте файл `/etc/yum.repos.d/jatoba-[версия].repo` с описанием локального репозитория в системе, в котором разместите следующее описание:

```
[jatoba-[версия]]
name=Jatoba [версия] Official Repository
baseurl=file:///localrepo
enabled=1
gpgcheck=0
```

¹ Расположение файла указано для 16 версии Postgre Pro, для поиска можно использовать команду `sudo find / -type f -name pg_hba.conf`

² Подробное описание приведено в официальной документации на Jatoba, размещённой по адресу <https://www.gaz-is.ru/produkty/inform-sistemy/subd-jatoba.html#materialy>

```
gpgkey=file:///localrepo/RPM-GPG-KEY-Jatoba
```

- Обновите описания пакетов командой:

```
sudo dnf makecache
```

- Установите основные пакеты СУБД Jatoba 4 командой:

```
sudo dnf install jatoba[версия]-client jatoba[версия]-contrib jatoba[версия]-libs
jatoba[версия]-server
```

Внимание! Пакеты `jatoba[версия]-client`, `jatoba[версия]-contrib`, `jatoba[версия]-libs` и `jatoba[версия]-server` являются обязательными для установки СУБД.

- Перейдите в директорию расположения исполняемых файлов СУБД Jatoba посредством команды:

```
cd /usr/jatoba-[версия]/bin/
```

- Инициализируйте каталог данных СУБД Jatoba при помощи команды:

```
sudo ./jatoba-setup initdb jatoba-[версия]
```

- Отредактируйте файл `/var/lib/jatoba/[версия]/data/postgresql.conf` под администратором - установите число подключений `max_connections` в значение `1000`¹.

- Отредактируйте файл `/var/lib/jatoba/[версия]/data/pg_hba.conf` под администратором - измените параметры для успешного локального подключения пользователя к базе данных:

```
host all all 127.0.0.1/32 ident на host all all 127.0.0.1/32 password
```

```
host all all ::1/128 ident на host all all ::1/128 password
```

- Добавьте СУБД Jatoba в автозагрузку командой:

```
sudo systemctl enable jatoba-[версия]
```

- Выполните перезапуск СУБД Jatoba для вступления изменений в силу, выполнив команду:

```
sudo systemctl restart jatoba-[версия]
```

4.2.4 Установка веб-сервера

4.2.4.1 Установка веб-сервера Apache

- Установите пакет, выполнив команду с правами суперпользователя (от имени пользователя root, либо с использованием sudo):

```
sudo dnf install httpd
```

- Установите дополнительный модуль для использования протокола SSL, выполнив команду с правами суперпользователя (от имени пользователя root, либо с использованием sudo):

```
sudo dnf install mod_ssl
```

- Добавьте веб-сервер в автозагрузку, выполнив команду с правами суперпользователя:

```
sudo systemctl enable httpd
```

4.2.4.2 Установка веб-сервера nginx

Порядок установки веб-сервера nginx:

- Установите пакет, выполнив следующую команду с правами суперпользователя:

¹ Значение `max_connections` равно `1000` является рекомендуемым, при необходимости можно установить и большее значение.

```
sudo dnf install nginx
```

- Запустите установленный веб-сервер, выполнив команду:

```
sudo systemctl start nginx
```

- Добавьте веб-сервер в автозагрузку, выполнив команду:

```
sudo systemctl enable nginx
```

4.3 Подготовка среды функционирования с ОС Astra Linux Special Edition

4.3.1 Подключение репозитория и установка зависимостей

4.3.1.1 Подключение репозитория и установка зависимостей Astra Linux Special Edition 1.7¹

Порядок подключения репозитория и зависимостей:

- Для установки зависимостей через сеть Интернет перед началом установки компонентов необходимо установить пути нахождения необходимых репозитория², отредактировав файл `/etc/apt/sources.list`, выполнив следующую команду с правами суперпользователя:

```
sudo nano /etc/apt/sources.list
```

- Укажите ссылки на следующие репозитории³:

```
deb https://dl.astralinux.ru/astra/frozen/1.7_x86-64/1.7.6/repository-main/
1.7_x86-64 main contrib non-free
deb https://dl.astralinux.ru/astra/frozen/1.7_x86-64/1.7.6/repository-update/
1.7_x86-64 main contrib non-free
deb https://dl.astralinux.ru/astra/frozen/1.7_x86-64/1.7.6/repository-base/
1.7_x86-64 main contrib non-free
```

- Укажите нижеприведённый репозиторий для развёртывания веб-сервера Nginx:

```
deb https://dl.astralinux.ru/astra/frozen/1.7_x86-64/1.7.6/repository-extended/
1.7_x86-64 main contrib non-free
```

Для установки необходимых компонентов в офлайн-режиме предварительно необходимо настроить использование установочных оптических дисков в качестве репозитория, отредактировав файл `/etc/apt/sources.list`. Пример:

```
deb cdrom:[OS Astra Linux 1.7.6 1.7_x86-64 DVD ]/ 1.7_x86-64 contrib main non-free
```

- Зарегистрируйте физический оптический диск, установленный в оптический привод, выполнив команду:

```
apt-cdrom add
```

- Выполните обновление пакетов для ОС из указанных репозитория, выполнив следующую команду с правами суперпользователя:

```
sudo apt update
```

- Для проверки доступности и готовности к дальнейшим командам следует установить необходимые пакеты из состава ОС, выполнив следующую команду с правами суперпользователя:

¹ Подробнее см. на [официальном сайте производителя](#).

² Ссылки на репозитории приведены для Astra Linux SE версии 1.7.6

³ При использовании доменной службы каталогов ALD Pro необходимо указывать адреса репозитория в соответствии с [инструкцией по подготовке и присоединению хоста к домену ALD Pro](#).

```
sudo apt install tar unzip iptables
```

В процессе установки в офлайн режиме может потребоваться замена оптических дисков с нужным репозиторием («диск 1», «диск 2», «develop»).

4.3.1.2 Подключение репозитория и установка зависимостей Astra Linux Special Edition 1.8¹

Порядок подключения репозитория и зависимостей:

- Для обновления посредством сети Интернет перед началом установки компонентов необходимо установить пути нахождения всех необходимых репозиториях², отредактировав файл `/etc/apt/sources.list`, выполнив следующую команду с правами суперпользователя:

```
sudo nano /etc/apt/sources.list
```

- Укажите ссылки на следующие репозитории³:

```
deb https://dl.astralinux.ru/astra/frozen/1.8_x86-64/1.8.1/main-repository/  
1.8_x86-64 main contrib non-free
```

- Укажите нижеприведённый репозиторий для развёртывания веб-сервера Nginx:

```
deb https://dl.astralinux.ru/astra/frozen/1.8_x86-64/1.8.1/extended-repository/  
1.8_x86-64 main contrib non-free
```

Для установки необходимых компонентов в офлайн режиме предварительно необходимо настроить использование установочных дисков в качестве репозитория, отредактировав файл `/etc/apt/sources.list`. Пример:

```
deb cdrom:[OS Astra Linux 1.7.6 1.7_x86-64 DVD ]/ 1.7_x86-64 contrib main non-free
```

- Зарегистрируйте физический оптический диск, установленный в оптический привод, выполнив команду:

```
apt-cdrom add
```

- Выполните обновление пакетов для операционной системы из указанных репозиториях, выполнив следующую команду с правами суперпользователя:

```
sudo apt update
```

- Для проверки доступности и готовности к дальнейшим командам следует установить необходимые пакеты из состава ОС, выполнив следующую команду с правами суперпользователя:

```
sudo apt install tar unzip iptables
```

В процессе установки в офлайн режиме может потребоваться заменить и вставить диск с нужным репозиторием («диск 1», «диск 2», «develop»).

4.3.1.3 Поддержка активного режима замкнутой программной среды

Центр сертификации Aladdin eCA обеспечивает работу ОС Astra Linux Special Edition 1.7 и Astra Linux Special Edition 1.8 в [активном режиме замкнутой программной среды \(далее - ЗПС\)](#). Для этого в состав установочных пакетов Центра сертификации Aladdin eCA включен публичный открытый ключ ОсОО «Аладдин КГ» - `aladdin_pub.key`. После распаковки установочного пакета ключ находится в каталоге `/opt/aecaCa/digsig/keys/aladdin_pub.key`.

¹ Подробнее см. на [официальном сайте производителя](#).

² Ссылки на репозитории приведены для Astra Linux SE 1.8.1

³ При использовании доменной службы каталогов ALD Pro необходимо указывать адреса репозиториях в соответствии с [инструкцией по подготовке и присоединению хоста к домену ALD Pro](#).

Для обеспечения режима ЗПС открытый ключ необходимо переместить¹ в каталог `/etc/digsig/keys/`.

4.3.2 Установка среды исполнения Java

4.3.2.1 Установка Axiom JDK Certified

Для установки Axiom JDK Certified воспользуйтесь [инструкцией с официального сайта производителя](#).

Внимание! В Astra Linux Special Edition Axiom JDK Certified версии 21 работает некорректно. В результате установка Центра сертификации Aladdin eCA прерывается. Проблема была выявлена в Axiom JDK Certified версии 21.0.4+10.

4.3.2.2 Установка OpenJDK

Для установки OpenJDK воспользуйтесь инструкцией с официального сайта Astra Linux:

- [инструкция для Astra Linux SE 1.7](#) (в инструкции описана установка Open JDK 11, установка Open JDK 17 и 21 аналогична).
- [инструкция для Astra Linux SE 1.8](#) (в инструкции описана установка Open JDK 17, установка Open JDK 21 аналогична).

4.3.3 Установка и настройка СУБД

Выполните установку и настройку одной из нижеприведённых СУБД:

- PostgreSQL из состава ОС.
- Postgres Pro.
- Jatoba.

Порядок разрешения конфликтов между СУБД PostgreSQL приведен в Приложении 1.

Порядок настройки взаимодействия с СУБД, размещенной на отдельном узле, приведен в Приложении 2.

Центр сертификации Aladdin eCA может быть настроен на взаимодействие с СУБД по протоколу TLS. Программа не будет функционировать, если ему не удалось установить TLS-соединение с СУБД. Порядок настройки TLS-соединения с СУБД приведен в Приложении 3.

4.3.3.1 Установка СУБД PostgreSQL²

Порядок установки СУБД PostgreSQL:

- Установите последнюю доступную версию СУБД PostgreSQL, выполнив следующую команду с правами суперпользователя:

```
sudo apt install postgresql
```

- Выполните установку последней доступной версии пакета `postgresql-contrib`, выполнив следующую команду с правами суперпользователя:

```
sudo apt install postgresql-contrib
```

- Установите пакет `postgresql-client`, выполнив следующую команду с правами суперпользователя:

```
sudo apt install postgresql-client
```

- Запустите PostgreSQL, выполнив следующую команду с правами суперпользователя:

```
sudo systemctl start postgresql
```

¹ Данное действие необходимо выполнять после распаковки установочных пакетов Центра сертификации Aladdin eCA.

² Подробное описание приведено на [официальном сайте производителя](#).

- Добавьте запуск PostgreSQL в автозагрузку, выполнив следующую команду с правами суперпользователя:

```
sudo systemctl enable postgresql
```

- При наличии мандатных политик¹:
 - выдайте полномочия пользователю `postgres`, выполнить следующую команду с правами суперпользователя:

```
sudo pdpl-user -l 0:0 -i 63 postgres
```

- предоставьте служебному пользователю `postgres` право на чтение файла, содержащего классификационную метку пользователя, поочерёдно выполнив следующие команды с правами суперпользователя:

```
sudo usermod -a -G shadow postgres
sudo setfacl -d -m u:postgres:r /etc/parsec/macdb
sudo setfacl -R -m u:postgres:r /etc/parsec/macdb
sudo setfacl -m u:postgres:rx /etc/parsec/macdb
```

- Отредактируйте файл `/etc/postgresql/11/main/postgresql.conf`², установите число подключений `max_connections` в значение `1000`³.
- Выполните перезапуск СУБД PostgreSQL для вступления изменений в силу, выполнив следующую команду с правами суперпользователя:

```
sudo systemctl restart postgresql
```

4.3.3.2 Установка СУБД Postgres Pro⁴

- Загрузите скрипт для добавления репозитория, выполнив команду⁵:

```
wget --user [ключ] --password='' https://repo.postgrespro.ru/std/std-16/keys/pgpro-repo-add.sh
```

- Запустите скрипт, выполнив команду с правами суперпользователя (root или sudo):

```
sudo sh pgpro-repo-add.sh
```

- Обновите список пакетов, выполнив команду с правами суперпользователя (root или sudo):

```
sudo apt update
```

- Установите Postgres Pro, выполнив команду с правами суперпользователя (root или sudo):

```
sudo apt install postgrespro-std-16
```

- При наличии мандатных политик⁶:
 - выдайте полномочия пользователю `postgres`, выполнить команду:

¹ Подробная информация по аутентификации в СУБД PostgreSQL приведена на [официальном сайте производителя](#).

² Расположение файла может отличаться. В инструкции расположение указано для PostgreSQL версии 11. Для поиска файла можно использовать команду `sudo find / -type f -name postgresql.conf`

³ Значение `max_connections` равное `1000` является рекомендуемым, при необходимости можно установить и большее значение.

⁴ Подробное описание приведено в официальной документации на Postgres Pro, размещённой по адресу <https://postgrespro.ru/docs>

⁵ Команды ниже приведены для 16-ой версии Postgres Pro.

⁶ Подробная информация по аутентификации в СУБД PostgreSQL для Astra Linux Special Edition приведена в <https://wiki.astralinux.ru/pages/viewpage.action?pagelId=238751148>

```
sudo pdpl-user -l 0:0 -i 63 postgres
```

- предоставьте служебному пользователю `postgres` право на чтение файла, содержащего классификационную метку пользователя, поочередно выполнив команды:

```
sudo usermod -a -G shadow postgres
sudo setfacl -d -m u:postgres:r /etc/parsec/macdb
sudo setfacl -R -m u:postgres:r /etc/parsec/macdb
sudo setfacl -m u:postgres:rx /etc/parsec/macdb
```

- Отредактируйте файл `/var/lib/pgpro/std-16/data/postgresql.conf`¹ с правами администратора - установите число подключений `max_connections` в значение `1000`².
- При отсутствии создайте символические ссылки на утилиты `psql` и `pg_dump`, выполнив команды с правами суперпользователя (root или sudo):

```
sudo ln -s /opt/pgpro/std-16/bin/psql /usr/bin/psql
sudo ln -s /opt/pgpro/std-16/bin/pg_dump /usr/bin/pg_dump
```

- Перезапустите Postgres Pro, выполнив команду с правами суперпользователя (root или sudo):

```
sudo systemctl restart postgrespro-std-16.service
```

4.3.3.3 Установка СУБД Jatoba³

- Создайте каталог `/localrepo`, выполнив команду:

```
sudo mkdir /localrepo
```

- В каталог `/localrepo` скопируйте необходимые файлы для установки СУБД Jatoba.

Внимание! Требуется скопировать полную структуру файлов и каталогов из дистрибутива. Также допускается установка с оптического диска напрямую. В этом случае, пользователю не требуется копировать файлы, а вместо каталога `/localrepo` во всех шагах далее указывать соответствующий путь до носителя и директорию репозитория СУБД на носителе для соответствующей ОС.

- Дистрибутив СУБД Jatoba содержит:

- каталог `/pool`;
- каталог `/dists`;
- файл ключа `DEB-GPG-KEY-Jatoba`.

- Проверьте результат копирования всех файлов дистрибутива, перейдя в каталог `/localrepo` и выполнив команду:

```
ls -l
```

- Установите открытый ключ репозитория командой:

```
sudo apt-key add /localrepo/DEB-GPG-KEY-Jatoba
```

- Создайте файл `/etc/apt/sources.list.d/jatoba-[версия].list` с описанием локального репозитория в системе, в котором разместите следующее описание:

¹ Расположение файла указано для 16 версии Postgres Pro, для поиска можно использовать команду `sudo find / -type f -name postgresql.conf`

² Значение `max_connections` равно `1000` является рекомендуемым, при необходимости можно установить и большее значение.

³ Подробное описание приведено в официальной документации на Jatoba, размещённой по адресу <https://www.gaz-is.ru/produkty/inform-sistemy/subd-jatoba.html#materialy>


```
deb file:///localrepo stable non-free
```

- Обновите описания пакетов командой:

```
sudo apt update
```

- Установите основные пакеты СУБД Jatoba командой:

```
sudo apt install jatoba[версия]-client jatoba[версия]-contrib jatoba[версия]-libs  
jatoba[версия]-server
```

Внимание! Пакеты `jatoba[версия]-client`, `jatoba[версия]-contrib`, `jatoba[версия]-libs` и `jatoba[версия]-server` являются обязательными для установки СУБД.

- При наличии мандатных политик¹:
 - выдайте полномочия пользователю `postgres`, выполнить команду:

```
sudo pdpl-user -l 0:0 -i 63 postgres
```

- предоставьте служебному пользователю `postgres` право на чтение файла, содержащего классификационную метку пользователя, поочередно выполнив команды:

```
sudo usermod -a -G shadow postgres  
sudo setfacl -d -m u:postgres:r /etc/parsec/macdb  
sudo setfacl -R -m u:postgres:r /etc/parsec/macdb  
sudo setfacl -m u:postgres:rx /etc/parsec/macdb
```

- Перейдите в директорию расположения исполняемых файлов СУБД Jatoba посредством команды:

```
cd /usr/jatoba-[версия]/bin/
```

- Инициализируйте каталог данных СУБД Jatoba при помощи команды:

```
sudo ./jatoba-setup initdb jatoba-[версия]
```

- Отредактируйте файл `/var/lib/jatoba/[версия]/data/postgresql.conf` с правами администратора - установите число подключений `max_connections` в значение `1000`².
- Добавьте СУБД Jatoba в автозагрузку командой:

```
sudo systemctl enable jatoba-[версия]
```

- Выполните перезапуск СУБД Jatoba для вступления изменений в силу, выполнив команду:

```
sudo systemctl restart jatoba-[версия]
```

В случае возникновения ошибки запуска следует обратиться к внутренним логам:

```
ls /var/lib/jatoba/[версия]/data/log  
cat /var/lib/jatoba/[версия]/data/log/[weekDay]
```

4.3.4 Установка веб-сервера

4.3.4.1 Установка веб-сервера Apache

- Установите пакет, выполнив команду с правами суперпользователя (от имени пользователя `root`, либо с использованием `sudo`):

¹ Подробная информация по аутентификации в СУБД PostgreSQL для Astra Linux Special Edition приведена в <https://wiki.astralinux.ru/pages/viewpage.action?pageId=238751148>

² Значение `max_connections` равное `1000` является рекомендуемым, при необходимости можно установить и большее значение.

```
sudo apt install apache2
```

- Активируйте модули, выполнив поочередно команды:

```
sudo a2enmod ssl
sudo a2enmod proxy
sudo a2enmod proxy_http
sudo a2enmod headers
sudo a2enmod cgi
sudo a2enmod rewrite
sudo a2enmod http2
```

- Перезапустите веб-сервер, выполнив команду с правами суперпользователя:

```
sudo systemctl restart apache2.service
```

- Добавьте веб-сервер в автозагрузку, выполнив команду с правами суперпользователя:

```
sudo systemctl enable apache2
```

- Для проверки корректности запуска модулей выполните команду с правами суперпользователя:

```
sudo apachectl -M | grep -E 'ssl|proxy|proxy_http|headers|cgi|rewrite|http2'
```

4.3.4.2 Установка веб-сервера Nginx

Установите пакет из расширенного репозитория ОС, выполнив команду с правами суперпользователя:

```
sudo apt install nginx
```

- Запустите установленный веб-сервер, выполнив команду:

```
sudo systemctl start nginx
```

- Добавьте веб-сервер в автозагрузку, выполнив команду:

```
sudo systemctl enable nginx
```

4.4 Подготовка среды функционирования с ОС Альт Сервер

4.4.1 Подключение репозитория и установка зависимостей

- Для развёртывания Центра сертификации Aladdin eCA с использованием веб-сервера Apache перед началом установки компонента необходимо установить путь нахождения необходимого репозитория, отредактировав файл `/etc/apt/sources.list`, выполнив команду:

```
sudo nano /etc/apt/sources.list.d/aptsp.list
```

Укажите ссылку на следующий репозиторий:

```
rpm [cert8] http://update.altsp.su/pub/distributions/ALTlinux c10f/branch/x86_64-i586 classic
```

- После этого обновите список доступных пакетов, выполнив команду:

```
sudo apt-get update
```

4.4.2 Установка среды исполнения Java

4.4.2.1 Установка Axiom JDK Certified

Для установки Axiom JDK Certified 17 воспользуйтесь [инструкцией с официального сайта производителя](#).

Для установки Axiom JDK Certified 21 воспользуйтесь [инструкцией с официального сайта производителя](#).

4.4.2.2 Установка OpenJDK

Для установки OpenJDK воспользуйтесь инструкцией с [официального сайта производителя ОС](#).

4.4.3 Установка и настройка СУБД

Выполните установку и настройку одной из нижеприведённых СУБД:

- PostgreSQL из состава ОС;
- Postgres Pro;
- Jatoba.

Порядок разрешения конфликтов между СУБД PostgreSQL приведен в Приложении 1.

Порядок настройки взаимодействия с СУБД, размещенной на отдельном узле, приведен в Приложении 2.

Центр сертификации Aladdin eCA может быть настроен на взаимодействие с СУБД по протоколу TLS.

Программа не будет функционировать, если ему не удалось установить TLS-соединение с СУБД. Порядок настройки TLS-соединения с СУБД приведен в Приложении 3.

4.4.3.1 Установка СУБД PostgreSQL¹

- Установите последнюю доступную версию СУБД PostgreSQL, выполнив команду²:

```
sudo apt-get install postgresql15-server
```

- Выполните установку последней доступной версии пакета `postgresql-contrib`, выполнив команду:

```
sudo apt-get install postgresql15-contrib
```

- Установите пакет `postgresql`, выполнив команду:

```
sudo apt-get install postgresql15
```

- Произведите инициализацию БД, выполнив команду:

```
sudo /etc/init.d/postgresql initdb
```

В случае ошибки `Data directory in '/var/lib/pgsql/data' is not empty...` следует очистить директорию командой ниже и повторить инициализацию БД.

```
sudo rm -rf /var/lib/pgsql/data
```

- Запустите PostgreSQL, выполнив команду:

```
sudo systemctl start postgresql
```

- Добавьте запуск PostgreSQL в автозагрузку, выполнив команду:

```
sudo systemctl enable postgresql
```

- Отредактируйте файл `/var/lib/pgsql/15/data/postgresql.conf`³ - установите число подключений `max_connections` в значение `1000`⁴.

- Выполните перезапуск СУБД PostgreSQL для вступления изменений в силу, выполнив команду:

```
sudo systemctl restart postgresql
```

¹ Подробное описание приведено в официальной документации на PostgreSQL, размещённой по адресу <https://postgrespro.ru/docs>

² Команды ниже приведены для установки 15-ой версии PostgreSQL.

³ Расположение файла может отличаться. В инструкции расположение указано для 15 версии PostgreSQL. Для поиска можно использовать команду `sudo find / -type f -name postgresql.conf`

⁴ Значение `max_connections` равно `1000` является рекомендуемым, при необходимости можно установить и большее значение.

4.4.3.2 Установка СУБД Postgres Pro¹

- Загрузите скрипт для добавления репозитория, выполнив команду²:

```
wget --user [ключ] --password=' ' https://repo.postgrespro.ru/std/std-16/keys/pgpro-repo-add.sh
```

- Запустите скрипт, выполнив команду с правами суперпользователя (root или sudo):

```
sudo sh pgpro-repo-add.sh
```

- Обновите список пакетов, выполнив команду с правами суперпользователя (root или sudo):

```
sudo apt-get update
```

- Установите Postgres Pro, выполнив команду с правами суперпользователя (root или sudo):

```
sudo apt-get install postgrespro-std-16
```

- Отредактируйте файл `/var/lib/pgpro/std-16/data/postgresql.conf`³ с правами администратора - установите число подключений `max_connections` в значение `1000`⁴.

- При отсутствии создайте символические ссылки на утилиты `psql` и `pg_dump`, выполнив команды с правами суперпользователя (root или sudo):

```
sudo ln -s /opt/pgpro/std-16/bin/psql /usr/bin/psql
sudo ln -s /opt/pgpro/std-16/bin/pg_dump /usr/bin/pg_dump
```

- Перезапустите Postgres Pro, выполнив команду с правами суперпользователя (root или sudo):

```
sudo systemctl restart postgrespro-std-16.service
```

4.4.3.3 Установка СУБД Jatoba⁵

- Создайте каталог `/localrepo`, выполнив команду:

```
sudo mkdir /localrepo
```

- В каталог `/localrepo` скопируйте необходимые файлы для установки СУБД Jatoba.

Внимание! Требуется скопировать полную структуру файлов и каталогов из дистрибутива. Также допускается установка с оптического диска напрямую. В этом случае, пользователю не требуется копировать файлы, а вместо каталога `/localrepo` во всех шагах далее указывать соответствующий путь до носителя и директорию репозитория СУБД на носителе для соответствующей ОС.

- Дистрибутив СУБД Jatoba содержит:

- каталог `/base`;
- каталог `/RPMS.classic`;
- файл ключа `RPM-GPG-KEY-Jatoba`.

- Проверьте результат копирования всех файлов дистрибутива, перейдя в каталог `/localrepo` и выполнив команду:

¹ Подробное описание приведено в официальной документации на Postgres Pro, размещённой по адресу <https://postgrespro.ru/docs>

² Команды ниже приведены для 16-ой версии Postgres Pro.

³ Расположение файла указано для 16 версии Postgres Pro, для поиска можно использовать команду `sudo find / -type f -name postgresql.conf`

⁴ Значение `max_connections` равно `1000` является рекомендуемым, при необходимости можно установить и большее значение.

⁵ Подробное описание приведено в официальной документации на Jatoba, размещённой по адресу <https://www.gaz-is.ru/produkty/inform-sistemy/subd-jatoba.html#materialy>.

```
ls -l
```

- Создайте файл `/etc/apt/sources.list.d/jatoba-[версия].list` под администратором с описанием локального репозитория в системе, в котором разместите следующее описание:

```
rpm file:///localrepo x86_64 classic
```

- Обновите описания пакетов командой:

```
sudo apt-get update
```

- Установите основные пакеты СУБД Jatoba командой:

```
sudo apt-get install jatoba[версия]-client jatoba[версия]-contrib jatoba[версия]-libs jatoba[версия]-server
```

Внимание! Пакеты `jatoba[версия]-client`, `jatoba[версия]-contrib`, `jatoba[версия]-libs` и `jatoba[версия]-server` являются обязательными для установки СУБД.

- Перейдите в директорию расположения исполняемых файлов СУБД Jatoba посредством команды:

```
cd /usr/jatoba-[версия]/bin/
```

- Инициализируйте каталог данных СУБД Jatoba при помощи команды:

```
sudo ./jatoba-setup initdb jatoba-[версия]
```

- Отредактируйте файл `/var/lib/jatoba/[версия]/data/postgresql.conf` под администратором - установите число подключений `max_connections` в значение `1000`¹.

- Добавьте СУБД Jatoba в автозагрузку командой:

```
sudo systemctl enable jatoba-[версия]
```

- Выполните перезапуск СУБД Jatoba для вступления изменений в силу, выполнив команду:

```
sudo systemctl restart jatoba-[версия]
```

4.4.4 Установка веб-сервера

4.4.4.1 Установка веб-сервера Apache

- Установите пакет, выполнив команду с правами суперпользователя (от имени пользователя root, либо с использованием sudo):

```
sudo apt-get install apache2-mod_http2
```

- Установите модуль ssl, выполнив команду с правами суперпользователя (от имени пользователя root, либо с использованием sudo):

```
sudo apt-get install apache2-mod_ssl
```

- Создайте файлы:

- `/etc/httpd2/conf/mods-available/http2.load`, выполнив команду с правами суперпользователя:

```
sudo nano /etc/httpd2/conf/mods-available/http2.load
```

Внесите следующий текст в созданный файл:

```
LoadModule http2_module /usr/lib64/apache2/modules/mod_http2.so
```

¹ Значение `max_connections` равно `1000` является рекомендуемым, при необходимости можно установить и большее значение.

- `/etc/httpd2/conf/mods-available/http2.conf` выполнив команду с правами суперпользователя:

```
sudo nano /etc/httpd2/conf/mods-available/http2.conf
```

Внесите следующий текст в созданный файл:

```
# mod_http2 doesn't work with mpm_prefork
<IfModule !mpm_prefork>
    Protocols h2 h2c http/1.1
</IfModule>
```

- Активируйте модули, выполнив поочередно команды:

```
sudo a2enmod ssl
sudo a2enmod proxy
sudo a2enmod proxy_http
sudo a2enmod headers
sudo a2enmod cgi
sudo a2enmod rewrite
sudo a2enmod http2
```

- Включите https порт по умолчанию, выполнив команду с правами суперпользователя:

```
sudo a2enport https
```

4.4.4.2 Установка веб-сервера Nginx

- Установите пакет из официального репозитория ОС, выполнив команду с правами суперпользователя:

```
sudo apt-get install nginx
```

- Запустите установленный веб-сервер, выполнив команду:

```
sudo systemctl start nginx
```

- Добавьте веб-сервер в автозагрузку, выполнив команду:

```
sudo systemctl enable nginx
```

4.5 Подготовка среды функционирования с ОС «Platform V SberLinux OS Server»

4.5.1 Установка среды исполнения Java

4.5.1.1 Установка Axiom JDK Certified

Для установки Axiom JDK Certified воспользуйтесь [инструкцией с официального сайта производителя](#).

Внимание! В Platform V SberLinux OS Server Axiom JDK Certified версии 21 работает некорректно. В результате установка Центра сертификации Aladdin eCA прерывается. Проблема была выявлена в Axiom JDK Certified версии 21.0.4+10.

4.5.1.2 Установка OpenJDK

Для установки OpenJDK воспользуйтесь инструкцией по установке пакета [с официального сайта Platform V SberLinux OS Server](#).

Внимание! В ОС «Platform V SberLinux OS Server» OpenJDK определенных версий работает некорректно. В результате установка Центра сертификации Aladdin eCA прерывается. Проблема была выявлена для OpenJDK версий 17.0.15.0.6 и 21.0.7.0.6.

4.5.2 Установка и настройка СУБД

Выполните установку и настройку одной из нижеприведённых СУБД:

- PostgreSQL из состава ОС.
- Postgres Pro.
- Jatoba.

Порядок разрешения конфликтов между СУБД PostgreSQL приведен в Приложении 1.

Порядок настройки взаимодействия с СУБД, размещенной на отдельном узле, приведен в Приложении 2.

Центр сертификации Aladdin eCA может быть настроен на взаимодействие с СУБД по протоколу TLS.

Программа не будет функционировать, если ему не удалось установить TLS-соединение с СУБД. Порядок настройки TLS-соединения с СУБД приведен в Приложении 3.

4.5.2.1 Установка СУБД PostgreSQL¹

Порядок установки СУБД PostgreSQL:

- Установите последнюю доступную версию СУБД PostgreSQL, выполнив команду:

```
sudo dnf install postgresql-server
```

- Выполните установку последней доступной версии пакета `postgresql-contrib`, выполнив команду:

```
sudo dnf install postgresql-contrib
```

- Произведите инициализацию БД, выполнив команду:

```
sudo postgresql-setup --initdb
```

В случае ошибки `Data directory in '/var/lib/pgsql/data' is not empty...` следует очистить каталог командой ниже и повторить инициализацию БД.

```
sudo rm -rf /var/lib/pgsql/data
```

- Запустите PostgreSQL, выполнив команду:

```
sudo systemctl start postgresql
```

- Добавьте запуск PostgreSQL в автозагрузку, выполнив команду:

```
sudo systemctl enable postgresql
```

- Отредактируйте файл `/var/lib/pgsql/data/postgresql.conf`² под администратором - установите число подключений `max_connections` в значение `1000`³.

- Отредактируйте файл `/var/lib/pgsql/data/pg_hba.conf`⁴ под администратором - измените параметры для успешного локального подключения пользователя к базе данных:

```
host all all 127.0.0.1/32 ident      на host all all 127.0.0.1/32 password
```

```
host all all ::1/128 ident          на host all all ::1/128 password
```

- Выполните перезапуск СУБД PostgreSQL для вступления изменений в силу, выполнив команду:

¹ Подробное описание приведено в официальной документации на PostgreSQL, размещённой по адресу <https://postgrespro.ru/docs>.

² Расположение файла может отличаться, для поиска можно использовать команду `sudo find / -type f -name postgresql.conf`

³ Значение `max_connections` равное `1000` является рекомендуемым, при необходимости можно установить и большее значение.

⁴ Расположение файла может отличаться, для поиска можно использовать команду `sudo find / -type f -name pg_hba.conf`

```
sudo systemctl restart postgresql
```

4.5.2.2 Установка СУБД Postgres Pro¹

Порядок установки СУБД Postgres Pro:

- Загрузите скрипт для добавления репозитория, выполнив команду²:

```
wget --user [ключ] --password='' https://repo.postgrespro.ru/std/std-16/keys/pgpro-repo-add.sh
```

- Запустите скрипт, выполнив команду с правами суперпользователя (root или sudo):

```
sudo sh pgpro-repo-add.sh
```

- Обновите список пакетов, выполнив команду с правами суперпользователя (root или sudo):

```
sudo dnf update
```

- Установите Postgres Pro, выполнив команду с правами суперпользователя (root или sudo):

```
sudo dnf install postgrespro-std-16
```

- Отредактируйте файл `/var/lib/pgpro/std-16/data/postgresql.conf`³ под администратором - установите число подключений `max_connections` в значение 1000⁴.

- Отредактируйте файл `/var/lib/pgpro/std-16/data/pg_hba.conf`⁵ под администратором - измените параметры для успешного локального подключения пользователя к базе данных:

```
host all all 127.0.0.1/32 ident на host all all 127.0.0.1/32 password
```

```
host all all ::1/128 ident на host all all ::1/128 password
```

- При отсутствии создайте символические ссылки на утилиты `psql` и `pg_dump`, выполнив команды с правами суперпользователя (root или sudo):

```
sudo ln -s /opt/pgpro/std-16/bin/psql /usr/bin/psql
sudo ln -s /opt/pgpro/std-16/bin/pg_dump /usr/bin/pg_dump
```

- Перезапустите Postgres Pro, выполнив команду с правами суперпользователя (root или sudo):

```
sudo systemctl restart postgrespro-std-16.service
```

4.5.2.3 Установка СУБД Jatoba⁶

- Создайте каталог `/localrepo`, выполнив команду:

```
sudo mkdir /localrepo
```

- В каталог `/localrepo` скопируйте необходимые файлы для установки СУБД Jatoba.

¹ Подробное описание приведено в официальной документации на Postgres Pro, размещённой по адресу <https://postgrespro.ru/docs>

² Команды ниже приведена для Postgres Pro версии 16.

³ Расположение файла указано для 16 версии Postgres Pro, для поиска можно использовать команду `sudo find / -type f -name postgresql.conf`

⁴ Значение `max_connections` равно 1000 является рекомендуемым, при необходимости можно установить и большее значение.

⁵ Расположение файла указано для 16 версии Postgre Pro, для поиска можно использовать команду `sudo find / -type f -name pg_hba.conf`

⁶ Подробное описание приведено в официальной документации на Jatoba, размещённой по адресу <https://www.gaz-is.ru/produkty/inform-sistemy/subd-jatoba.html#materialy>

Внимание! Требуется скопировать полную структуру файлов и каталогов из дистрибутива. Также допускается установка с оптического диска напрямую. В этом случае, пользователю не требуется копировать файлы, а вместо каталога `/localrepo` во всех шагах далее указывать соответствующий путь до носителя и директорию репозитория СУБД на носителе для соответствующей ОС.

- Дистрибутив СУБД Jatoba содержит:
 - каталог `/packages`;
 - каталог `/repdata`;
 - файл ключа `RPM-GPG-KEY-Jatoba`.
- Проверьте результат копирования всех файлов дистрибутива, перейдя в каталог `/localrepo` и выполнив команду:

```
ls -l
```

- Установите открытый ключ репозитория командой:

```
sudo rpm --import /localrepo/RPM-GPG-KEY-Jatoba
```

- Создайте файл `/etc/yum.repos.d/jatoba-[версия].repo` с описанием локального репозитория в системе, в котором разместите следующее описание:

```
[jatoba-[версия]]
name=Jatoba [версия] Official Repository
baseurl=file:///localrepo
enabled=1
gpgcheck=0
gpgkey=file:///localrepo/RPM-GPG-KEY-Jatoba
```

- Обновите описания пакетов командой:

```
sudo dnf makecache
```

- Установите основные пакеты СУБД Jatoba 4 командой:

```
sudo dnf install jatoba[версия]-client jatoba[версия]-contrib jatoba[версия]-libs
jatoba[версия]-server
```

Внимание! Пакеты `jatoba[версия]-client`, `jatoba[версия]-contrib`, `jatoba[версия]-libs` и `jatoba[версия]-server` являются обязательными для установки СУБД.

- Перейдите в директорию расположения исполняемых файлов СУБД Jatoba посредством команды:

```
cd /usr/jatoba-[версия]/bin/
```

- Инициализируйте каталог данных СУБД Jatoba при помощи команды:

```
sudo ./jatoba-setup initdb jatoba-[версия]
```

- Отредактируйте файл `/var/lib/jatoba/[версия]/data/postgresql.conf` под администратором - установите число подключений `max_connections` в значение `1000`¹.
- Отредактируйте файл `/var/lib/jatoba/[версия]/data/pg_hba.conf` под администратором - измените параметры для успешного локального подключения пользователя к базе данных:

¹ Значение `max_connections` равно `1000` является рекомендуемым, при необходимости можно установить и большее значение.

```
host all all 127.0.0.1/32 ident
```

на

```
host all all 127.0.0.1/32 password
```

```
host all all ::1/128 ident
```

на

```
host all all ::1/128 password
```

- Добавьте СУБД Jatoba в автозагрузку командой:

```
sudo systemctl enable jatoba-[версия]
```

- Выполните перезапуск СУБД Jatoba для вступления изменений в силу, выполнив команду:

```
sudo systemctl restart jatoba-[версия]
```

4.5.3 Установка веб-сервера

4.5.3.1 Установка веб-сервера Apache

- Установите пакет, выполнив команду с правами суперпользователя (от имени пользователя root, либо с использованием sudo):

```
sudo dnf install httpd
```

- Установите дополнительный модуль для использования протокола SSL, выполнив команду с правами суперпользователя (от имени пользователя root, либо с использованием sudo):

```
sudo dnf install mod_ssl
```

- Добавьте веб-сервер в автозагрузку, выполнив команду с правами суперпользователя:

```
sudo systemctl enable httpd
```

4.5.3.2 Установка веб-сервера Nginx

Порядок установки веб-сервера nginx:

- Установите пакет, выполнив следующую команду с правами суперпользователя:

```
sudo dnf install nginx
```

- Запустите установленный веб-сервер, выполнив команду:

```
sudo systemctl start nginx
```

- Добавьте веб-сервер в автозагрузку, выполнив команду:

```
sudo systemctl enable nginx
```

4.6 Установка веб-сервера Cppnginx

Пакеты веб-сервера `cppnginx` расположены в дистрибутиве СКЗИ «КриптоПро CSP». Установка веб-сервера выполняется после установки СКЗИ «КриптоПро CSP» (см. Приложение 5).

Порядок установки веб-сервера `cppnginx`:

- распакуйте архив с дистрибутивом СКЗИ «КриптоПро CSP», выполнив команду с правами суперпользователя:

```
sudo tar -zxvf <имя_дистрибутива>.tgz && cd <имя_дистрибутива>
```

- установите следующие пакеты:

- для ОС Astra Linux SE командой `sudo dpkg -i <наименование пакета>.deb`:
 - `cprocsp-nginx-64_5.0.13000-7_amd64.deb`;
 - `lsb-cprocsp-rcrypt-64_5.0.13300-7_amd64.deb`;
 - `cprocsp-pki-plugin-64_2.0.15000-1_amd64.deb`.
- для ОС РЕД ОС командой `sudo dnf install <наименование пакета>.rpm`:
 - `cprocsp-nginx-64-5.0.13000-7.x86_64.rpm`;

- о `lsb-cprocsp-rcrypt-64-5.0.13000-7.x86_64.rpm`.
- для ОС SberLinux OS Server командой `sudo dnf install <наименование пакета>.rpm`:
 - о `cprocsp-nginx-64-5.0.13000-7.x86_64.rpm`;
 - о `lsb-cprocsp-rcrypt-64-5.0.13000-7.x86_64.rpm`.
- для ОС Альт Сервер командой `sudo apt-get install <наименование пакета>.rpm`:
 - о `cprocsp-nginx-64-5.0.13000-7.x86_64.rpm`;
 - о `lsb-cprocsp-rcrypt-64-5.0.13000-7.x86_64.rpm`.
- установите на СКЗИ «КриптоПро CSP» соответствующую лицензию (TLS-сервер) командой:

```
sudo /opt/cprocsp/sbin/amd64/cpconfig -license -set "Номер лицензии"
```

- выполните проверку активации лицензии командой:

```
sudo /opt/cprocsp/sbin/amd64/cpconfig -license -view
```

- Запустите установленный веб-сервер, выполнив команду:

```
sudo systemctl start cpnginx.service
```

- Добавьте веб-сервер в автозагрузку, выполнив команду:

```
sudo systemctl enable cpnginx.service
```

4.7 Установка JC-WebClient

JC-WebClient необходимо установить на компьютер, с которого будет выполняется управление серверной частью Центра сертификации Aladdin eCA через веб-интерфейс. JC-WebClient обеспечивает выпуск сертификатов на электронных ключах (ключевых носителях).

Скачайте дистрибутив JC-WebClient [с веб-сайта производителя](#) и установите зависимости.

Установите JC-WebClient, выполнив команду:

РЕД ОС `sudo dnf install JC-WebClient-x64-x.x.x.xxxx.rpm`

SberLinux OS Server `sudo dnf install JC-WebClient-x64-x.x.x.xxxx.rpm`

Astra Linux SE `sudo apt install -f JC-WebClient-x64-x.x.x.xxxx.deb`

Альт Сервер `sudo apt-get install JC-WebClient-x64-x.x.x.xxxx.rpm`

Перейдите в каталог `/etc/rc.d/init.d/`, выполнив команду:

```
cd /etc/rc.d/init.d/
```

Выполните запуск JC-WebClient, выполнив следующую команду с правами суперпользователя:

```
sudo sh jcmon start
```

4.1 Установка ПО «Рутокен Плагин» и его расширения

ПО «Рутокен Плагин» и его веб-браузерное расширение «Адаптер Рутокен Плагин» обеспечивает выпуск сертификатов на электронных ключах (ключевых носителях) Рутокен. ПО «Рутокен Плагин» и его браузерное расширение «Адаптер Рутокен Плагин» необходимо установить на компьютер, с которого будет выполняется подключение к Клиентской части Центра сертификации Aladdin eCA.

Скачайте дистрибутив ПО «Рутокен Плагин» с [официального сайта производителя](#).

Установите ПО «Рутокен Плагин» и его браузерное расширение «Адаптер Рутокен» по инструкции с [официального сайта производителя](#).

4.2 Установка программного средства «Криптографический модуль Aladdin JCP»

Порядок установки криптопровайдера «Aladdin JCP»:

- Получите от ОсОО «Аладдин КГ» набор файлов «Aladdin JCP».
- При отсутствии создайте каталог `/opt/aecaCa/services/cryptoproviders` командой:

```
sudo mkdir -p /opt/aecaCa/services/cryptoproviders
```

- Переместите в каталог `/opt/aecaCa/services/cryptoproviders` все файлы криптопровайдера «Aladdin JCP».
- Назначьте права доступа на скопированные файлы:
 - Если выполняется первоначальная установка Центра сертификации Aladdin eCA, то назначьте файлам права доступа (`chmod 777`).
 - Если Центр сертификации Aladdin eCA был ранее установлен, то назначьте владельцем данных файлов пользователя «aeca» и предоставьте ему права доступа к файлам (`chmod 700`).
- Выполните установку программы (см. раздел 5), если Центр сертификации Aladdin eCA не был ранее установлен.
- Если Центр сертификации Aladdin eCA был ранее установлен необходимо запустить скрипт с правами суперпользователя в режиме обновления программы:

```
sudo bash /opt/aecaCa/scripts/install.sh
```

5 УСТАНОВКА ПРОГРАММЫ

Перед установкой Центр сертификации Aladdin eCA необходимо выполнить подготовку сервера, где предполагается развертывание Центра сертификации Aladdin eCA, в соответствии с разделом 4 настоящего руководства.

Внимание! В случае повторной установки ПО рекомендуется произвести очистку кэш используемого веб-браузера.

5.1 Распаковка инсталляционного комплекта программы

Распакуйте инсталляционный rpm/deb-пакет, находясь в папке, где расположен пакет, выполнив команду с правами суперпользователя:

РЕД ОС	<code>sudo dnf install <наименование пакета>.rpm</code>
SberLinux OS Server	<code>sudo dnf install <наименование пакета>.rpm</code>
Astra Linux SE	<code>sudo dpkg -i <наименование пакета>.deb</code>
Альт Сервер	<code>sudo apt-get install <наименование пакета>.rpm</code>

Инсталляционный rpm/deb-пакет будет автоматически распакован в каталог `/opt/aecaCa`.
Структура распакованного инсталляционного rpm/deb-пакета приведена в таблице 5.

Таблица 5 - Структура установочного комплекта Центра сертификации Aladdin eCA

Структурный элемент	Назначение элемента
<code>../opt/aecaCa</code>	Установочный комплект программы, а также используемые дополнительные инструменты
<code>/opt/aecaCa/digsig/keys/aladdin_pub.key</code>	Публичный открытый ключ производителя для обеспечения ЗПС ОС Astra Linux SE
<code>../opt/aecaCa/bin</code>	Каталог с дополнительными утилитами
<code>../opt/aecaCa/bin/jcverify</code>	Каталог утилиты контроля целостности «jcverify»
<code>../opt/aecaCa/bin/jcverify/jcverify</code>	Утилита контроля целостности «jcverify»
<code>../opt/aecaCa/bin/jcverify/jcverify.txt</code>	Вспомогательный файл для работы утилиты целостности «jcverify»
<code>../opt/aecaCa/dist</code>	Путь развертывания продукта, содержит создаваемые временные файлы
<code>..dist/archive/</code>	Архивы, сформированные в результате очистки журнала событий
<code>..dist/backup/</code>	Созданные резервные копии Центра сертификации
<code>..dist/certificates/account</code>	Расположение pkcs#12 контейнера сертификата администратора инициализации
<code>..dist/certificates/ssl</code>	Расположение сертификатов для управления ssl-соединением
<code>..dist/cryptotoken/</code>	Расположение pkcs#12 контейнеров, содержащих открытый и закрытый ключи Центров сертификации
<code>..dist/environment/</code>	Расположение переменных окружения сервисов

Структурный элемент	Назначение элемента
../dist/logs/	Расположения технических журналов сервисов
../opt/aecaCa/eula	Файл лицензионного соглашения
../opt/aecaCa/samples	Содержит шаблоны файлов конфигурации для внутреннего использования программой
../opt/aecaCa/scripts	Содержит скрипты управления Центра сертификации Aladdin eCA
../scripts/external	Содержит скрипт для экспорта шаблонов MSCS
../scripts/internal	Скрипты для внутреннего использования программы, запускаемые автоматически при выполнении скриптов из каталога скриптов
/opt/aecaCa/scripts/internal/aeca/selinux	Политики, подключаемые к selinux, необходимые для функционирования Центра сертификации Aladdin eCA
../scripts/backup.sh	Скрипт резервного копирования конфигурации Центра сертификации Aladdin eCA
../scripts/config.sh	Конфигурационный файл Центра сертификации Aladdin eCA (развертывание продукта, настройка подключения к БД, управление конфигурацией сервисов)
../scripts/database_create.sh	Скрипт создания базы данных на разворачиваемом сервере Центра сертификации с предустановленными параметрами по умолчанию (именем пользователя, наименованием БД и т.д.)
../scripts/diagnostics.sh	Скрипт сбора диагностических данных
../scripts/email_config.sh	Конфигурация управления шаблонами email-рассылки
../scripts/install.sh	Скрипт установки и обновления текущей версии Центра сертификации Aladdin eCA
../scripts/integrity_check.sh	Скрипт контроля целостности исполняемых файлов
../scripts/restore.sh	Скрипт восстановления из резервной копии конфигурации Центра сертификации Aladdin eCA
../scripts/restore_access.sh	Скрипт восстановления доступа к Центру сертификации Aladdin eCA
../scripts/export-ca-data.sh	Скрипт экспорта файлов CRL, Delta CRL, AIA из Центра сертификации Aladdin eCA
../scripts/uninstall.sh	Скрипт удаления Центра сертификации Aladdin eCA
../scripts/jc_checksum	Файл эталонных хэш-сумм сервисов и список сервисов, подвергаемых контролю целостности
../opt/aecaCa/services	Сервисы Серверной части Центра сертификации
../opt/aecaCa/services/cryptoproviders	Каталог файлов для взаимодействия со сторонним криптопровайдером
/opt/aecaCa/scripts/key	Файл, содержащий ключ шифрования пароля пользователя СУБД в конфигурационном файле
../opt/aecaCa/static	Артефакты Клиентской части Центра сертификации Aladdin eCA

Владельцем распакованных файлов будет являться пользователь «root», другие пользователи не будут иметь прав доступа к инсталляционному комплекту.

5.2 Настройка параметров конфигурации программы

- Конфигурация Центра сертификации Aladdin eCA задается с помощью параметров конфигурационного файла `/opt/aecaCa/scripts/config.sh`.

- Перед установкой программного компонента определите значения следующих параметров:

- `webserver` - укажите используемый веб-сервер (nginx, apache или cpnginx). О выборе веб-сервера смотри в подразделе 5.3. Также значение параметра можно будет ввести при запуске инсталлятора в интерактивном режиме;
- `webserver_path` - укажите папку с файлами для развёртывания веб-сервера. Также значение параметра можно будет ввести при запуске инсталлятора, в интерактивном режиме указав путь к файлам веб-сервера (конфигурация nginx располагается по пути `etc/nginx`; конфигурация apache располагается: для Astra Linux по пути `/etc/apache2`, для РЕД ОС и SberLinux OS Server по пути `/etc/httpd`; для Альт Сервера конфигурация `apache` располагается по пути `/etc/httpd2/conf`; конфигурация `cpnginx` располагается по пути `/etc/opt/cprosp/cpnginx`);
- `database_password` - укажите пароль пользователя базы данных. После создания и настройки базы данных (см. подраздел 5.3) пароль пользователя базы данных отображается в конфигурационном файле в зашифрованном виде (алгоритм шифрования AES-256 с использованием сгенерированного в файле `/opt/aecaCa/scripts/key` ключа шифрования);

Внимание! Пароль не должен содержать специальные символы «|» и «\».

- `root_cert_path` - абсолютный путь к сертификату корневого Центра сертификации из цепочки сертификатов сервера СУБД. Значение параметра необходимо заполнить только при включенном флаге обязательного использования TLS для подключения к СУБД (при значении параметра `use_tls=true`). Иначе (при `use_tls=false`) следует оставить параметр незаполненным;
- `hostname` - укажите полное имя сервера Центра сертификации Aladdin eCA. Установленное значение заносится в атрибуты «Common name» и «DNS Name» автоматически создаваемых при развёртывании локального субъекта веб-сервера и сертификата для него.

- При использовании СКЗИ «КриптоПро CSP» канал взаимодействия клиентской и серверной части Центра сертификации Aladdin eCA должен быть организован по протоколу TLS ГОСТ, а также должна обеспечиваться TLS-аутентификация пользователей в Центре сертификации Aladdin с использованием отечественных криптографических алгоритмов. Для этого настройте конфигурационный файл в соответствии с таблицей 6.

Таблица 6 - Параметры для настройки TLS ГОСТ

Параметр	Значение
<code>webserver</code>	'cpnginx'
<code>webserver_path</code>	'/etc/opt/cprosp/cpnginx'
<code>initial_cryptography_provider</code>	'CRYPTO_PRO'
<code>initial_cryptography_key_algorithm</code>	'GOST_R_34_10_2012'
<code>initial_cryptography_key_bits</code>	'256' или '512'
<code>initial_cryptography_hash_algorithm</code>	'GOST_R_34_11_2012'
<code>initial_ca_common_name</code>	пример значения: 'INITIAL_CA_GOST'
<code>initial_admin_principal</code>	пример значения: 'INITIAL_ADMIN_GOST'
<code>sign_provider</code>	'CRYPTO_PRO'
<code>sign_key_algorithm</code>	'GOST_R_34_10_2012'

Параметр	Значение
sign_key_length	'256' или '512'
sign_hash_algorithm	'GOST_R_34_11_2012'

Отредактируйте конфигурационный файл `/opt/aecaCa/scripts/config.sh`, выполнив команду:

```
sudo nano /opt/aecaCa/scripts/config.sh
```

Настраиваемые параметры конфигурационного файла позволяют задавать:

- параметры конфигурации развёртывания сервисов центра сертификации;
- параметры e-mail уведомлений пользователям об истечении срока действия выданного сертификата;
- параметры конфигурации центра валидации;
- параметры конфигурации технического центра сертификации, создаваемого по умолчанию в процессе развёртывания сервера центра сертификации;
- параметры сертификата технического центра сертификации;
- параметры сертификата учётной записи администратора инициализации;
- параметры сертификата веб-сервера технологического центра сертификации;
- расписание синхронизации ресурсных систем;
- расписание публикации списка отозванных сертификатов;
- расписание проверки срока действия сертификатов центров сертификации и выпущенных сертификатов субъектов;
- расписание архивации журнала событий;
- конфигурацию базы данных.

Описание параметров конфигурационного файла приведено в таблице 7.

Таблица 7 - Описание параметров конфигурации

Параметр	Значение параметра по умолчанию	Описание
Конфигурация развёртывания		
webserver	'#CHANGEIT'	Используемый веб-сервер (nginx, apache или cpnginx)
webserver_path	'#CHANGEIT'	Папка с файлами для развёртывания веб-сервера. Конфигурация Nginx располагается по пути <code>etc/nginx</code> , конфигурация Apache располагается для Astra Linux по пути <code>/etc/apache2</code> , для РЕД ОС и SberLinux OS Server по пути <code>/etc/httpd</code> , для Альт Сервер конфигурация Apache располагается по пути <code>/etc/httpd2/conf</code> , конфигурация Cpnginx располагается по пути <code>/etc/opt/cprosp/cpnginx</code> .
aeca_path	'/opt/aecaCa/dist'	Папка с файлами для развёртывания Центра сертификации Aladdin eCA.
environment_path	'/opt/aecaCa/dist/environment'	Папка с переменными окружения для сервисов.
cryptotoken_path	'/opt/aecaCa/dist/cryptotoken'	Папка, содержащая открытый и закрытый ключи для доступа (аутентификации) к центру сертификации Aladdin eCA.
webserver_config_path	'/opt/aecaCa/dist/webserver'	Расположение конфигурации Центра сертификации Aladdin eCA для веб-сервера.

Параметр	Значение параметра по умолчанию	Описание
encryption_key_path	'/opt/aecaCa/scripts/key'	Ключ для шифрования конфигурационного файла
ssl_protocols	'TLSv1.2 TLSv1.3'	Поддерживаемые версии протокола TLS. Доступно использование только TLS v1.2 и/или TLS v1.3 (при использовании обеих версий необходимо указывать их через пробел).
ssl_ciphers	По умолчанию не задано	<p>Поддерживаемые наборы шифров для TLS-соединения. Данный параметр позволяет ограничить наборы шифров (cipher suites), которые могут использоваться при TLS-соединении. Разделитель между наборами - «:». Если клиент не поддерживает ни один из указанных в данном параметре наборов, TLS-соединение не будет установлено. По умолчанию значение в данном параметре не задано, что означает отсутствие управления со стороны Центра сертификации перечнем допустимых наборов шифров (cipher suites) TLS-соединения для веб-сервера. В данном параметре могут быть указаны любые наборы шифров, поддерживаемые используемой на сервере Центра сертификации версией Openssl для TLS версии 1.2. Получить список поддерживаемых используемым Openssl наборов шифров для TLS версии 1.2 можно с помощью команды:</p> <pre>openssl ciphers -tls1_2 -s</pre> <p>Данный параметр учитывается только при использовании веб-серверов Nginx или Apache. Конфигурирование наборов шифров TLS-соединения для Cpnginx осуществляется с помощью утилиты «cpconfig» из состава СКЗИ «КриптоПро CSP»¹.</p>
Параметры проксирования nginx		
proxy_connect_timeout	'320'	Время ожидания подключения к прокси-серверу перед тем, как будет выдано сообщение об ошибке. Только для nginx. Настраивается разработчиком Центра сертификации Aladdin eCA, редактировать не следует.
proxy_send_timeout	'320'	Время ожидания ответа от прокси-сервера после отправки запроса. Если ответ не получен в течение этого времени, запрос считается неудачным. Только для nginx. Настраивается разработчиком Центра сертификации Aladdin eCA, редактировать не следует.
proxy_read_timeout	'720'	Время ожидания чтения ответа от прокси-сервера после получения успешного запроса. Если ответ не получен в течение этого времени, запрос считается неудачным. Только для nginx. Настраивается разработчиком Центра сертификации Aladdin eCA, редактировать не следует.

¹ Описание порядка конфигурирования наборов шифров представлено в разделе 6 [инструкции по установке и настройке cpnginx](#).

Параметр	Значение параметра по умолчанию	Описание
Путь до резервных копий		
backup_path	'/opt/aecaCa/dist/backup'	Папка, в которую сохраняются резервные копии Центра сертификации Aladdin eCA.
Путь хранения архива журнала событий		
logs_base	'/opt/aecaCa/dist/logs'	Папка, в которую сохраняется журнал событий (лог-файлы).
archive_path	'/opt/aecaCa/dist/archive'	Папка, в которую сохраняется архив журнала событий, сформированный в результате автоматической архивации по заданным параметрам.
logs_file_max_size	'10MB'	Максимальный размер лог-файла (файла с диагностической информацией) сервиса перед его архивацией. При достижении данного значения текущий лог-файл (access.log или service.log) будет заархивирован. Файл будет сохранен в текущем каталоге хранения лог-файлов данного сервиса с именем {access или service}-{дата в формате YYYY-MM-DD}.{индекс лога}.log.
logs_max_history	'10'	Максимальный срок хранения архивов с лог-файлами в днях. Архивы, срок хранения которых превышает указанное в данном параметре значение, будут автоматически удаляться.
logs_total_size_cap	'100MB'	Максимальный общий объем лог-файлов, включая архивы, каждого типа (access или service) для каждого сервиса. При достижении данного объема наиболее старые архивы данного типа будут удаляться.
Путь хранения контейнера сертификата и ключа веб-сервера, а также цепочек сертификатов разрешенных издателей		
certificates_ssl_path	'/opt/aecaCa/dist/certificates/ssl'	Папка, содержащая сертификат веб-сервера и цепочки сертификатов разрешенных издателей.
certificates_account_path	'/opt/aecaCa/dist/certificates/account'	Папка, содержащая сертификат администратора инициализации в контейнере .p12.
Конфигурация пользователя		
aeca_user	'aeca'	Имя пользователя Центра сертификации Aladdin eCA, используемое для работы программы.
aeca_group	'aeca'	Группа, в которой состоит пользователь Центра сертификации Aladdin eCA.
Конфигурация памяти		
memory	'8192'	Лимит оперативной памяти для программы. Значение в Мб. Центр сертификации Aladdin eCA при запуске резервирует указанное в данном параметре количество оперативной памяти для своих сервисов. При значении параметра менее 8 Гб Центр сертификации Aladdin eCA не запустится - будет выдано сообщение об ошибке. При значении параметра, превышающем количество оперативной памяти хоста, будет использована вся доступная оперативная память хоста. Необходимо изменять при крупном внедрении.

Параметр	Значение параметра по умолчанию	Описание
enable_gc_diagnostic	'false'	Флаг сбора диагностической информации о памяти. При включении данного флага и выполнении скрипта сбора диагностических данных в архиве диагностических данных ¹ будет содержаться лог сборщика мусора и дампы памяти для упавших приложений Центра сертификации Aladdin eCA.
enable_heap_dump	'false'	Флаг сбора дампов памяти для упавших приложений Центра сертификации Aladdin eCA.
Конфигурация базы данных		
max_db_pool_size	'200'	Максимальный размер пула подключений к СУБД. Настраивается разработчиком Центра сертификации Aladdin eCA, редактировать не следует.
use_tls	f'false'	Флаг обязательного использования TLS для подключения к СУБД (true или false).
database_username	'aeca'	Имя пользователя базы данных, используемое для работы Центра сертификации Aladdin eCA.
database_password	'#CHANGEIT'	Пароль пользователя базы данных, используемый для работы Центра сертификации Aladdin eCA. Пароль не должен содержать специальные символы « » и «\».
database_host	'localhost'	Сетевой адрес базы данных.
database_port	'5432'	Порт, используемый для подключения к базе данных.
database_name	'aecaca'	Имя базы данных, используемой Центром сертификации Aladdin eCA.
root_cert_path	'#CHANGEIT'	Абсолютный путь к сертификату корневого Центра сертификации из цепочки сертификатов сервера СУБД.
Конфигурация аеса-са		
http_port	'80'	Порт для подключения к программному компоненту «Центр Сертификации» по протоколу HTTP.
https_port	'443'	Порт для подключения к Центру сертификации Aladdin eCA по протоколу HTTPS.
number_of_services	'17'	Количество активных сервисов в системе. Настраивается разработчиком Центра сертификации Aladdin eCA, редактировать не следует.
hostname	'localhost'	Имя сервера, на котором развёртывается Центр сертификации Aladdin eCA. Также заносится в атрибуты «Common name» и «DNS Name» автоматически создаваемых (при развёртывании Центра сертификации Aladdin eCA) сертификата веб-сервера и локального субъекта. Должно совпадать с hostname сервера.
Переменные окружения, используемые всеми сервисами		

¹ Размещение скрипта сбора диагностических данных - /opt/aecaCa/scripts/diagnostics.sh

Параметр	Значение параметра по умолчанию	Описание
logging_response	'false'	Флаг для сбора и регистрации ответов сервисов.
logging_sql	'false'	Флаг для сбора и регистрации информации о подключениях и запросах к базе данных PostgreSQL.
internal_http_read_timeout	'240'	Максимальный таймаут ожидания ответа методов сервисов при внутреннем взаимодействии. Единица измерения - секунды.
internal_http_connection_timeout	'60'	Максимальный таймаут ожидания подключения к сервисам при внутреннем взаимодействии. Единица измерения - секунды.
Ключ для внутренней аутентификации		
api_key	'2d2ec9b4-ad3d-4ed0-8961-d2a4ab99d810'	Значение ключа для внутренней аутентификации. Для служебного пользования, доступ к учётной записи Системного администратора ограничен, установку программы выполняет ответственный специалист
Переменные окружения, используемые certificate-authority-service		
pkcs12_key_protection_algorithm	'PBEWithHmacSHA256AndAES_256'	Алгоритм хеширования для ключа контейнера PKCS12. Допустимые значения: <ul style="list-style-type: none"> PBEWithHmacSHA256AndAES_256 - рекомендуется; PBEWithSHA1AndDESede - устаревший
pkcs12_mac_protection_algorithm	'HmacPBESHA256'	Алгоритм хеширования MAC контейнера PKCS12. Допустимые значения: <ul style="list-style-type: none"> HmacPBESHA256 - рекомендуется; HmacPBESHA1 - устаревший
pkcs12_certificate_protection_algorithm	'PBEWithHmacSHA256AndAES_256'	Алгоритм хеширования для сертификата контейнера PKCS12. Допустимые значения: <ul style="list-style-type: none"> PBEWithHmacSHA256AndAES_256 - рекомендуется; PBEWithSHA1AndRC2_40 - устаревший
Переменные окружения, используемые ldap-service		
ldap_sync_connection_point	'0 */30 * * * *'	CRON-выражение, по которому запускается частичная синхронизация зарегистрированных точек подключения (значение по умолчанию: '0 */30 * * * *' - запуск каждые полчаса).
ldap_sync_resource	'0 0 0 * * *'	CRON-выражение, по которому запускается полная синхронизация ресурсных систем (значение по умолчанию: '0 0 0 * * *' - запуск каждую полночь).
ldap_clean_queues	'0 */30 * * * *'	CRON-выражение, по которому запускается очистка необработанных элементов очередей на синхронизацию.
ldap_partition_size	'1000'	Максимальное количество объектов, получаемых из ресурсных систем при каждом запросе.
Переменные окружения, используемые publisher-service		
crl_scheduler	'0 */1 * * * *'	CRON-выражение, по которому запускается служба выпуска CRL.
crl_clean_queues	'0 */30 * * * *'	CRON-выражение, по которому очищаются очереди службы выпуска CRL.

Параметр	Значение параметра по умолчанию	Описание
Переменные окружения, используемые event-delivery-service		
email_host	'127.0.0.1'	Хост почтового сервера.
email_port	'25'	Порт почтового сервера.
email_login	'aeca'	Логин пользователя.
email_password	'aeca'	Пароль пользователя.
email_from	'no_reply@aeca.kg'	Почтовый адрес, с которого отправлено сообщение. Может не работать. Google предоставляет логин.
email_schedule	'0 0 12 * * *'	CRON-выражение для запуска метода отправки почтовых уведомлений.
email_enabled	'true'	Флаг отправки почтовых уведомлений. Если отправка выключена, то сообщения не отправляются, но помечаются, как отправленные.
email_protocol	'smtp'	Протокол подключения к почтовому серверу.
email_smtp_auth	'false'	Флаг использования SMTP-авторизации.
email_start_tls	'false'	Флаг использования директивы «start tls» при подключении к почтовому серверу.
Переменные окружения, используемые settings-service		
initial_cryptography_provider	'EMBEDDED'	Криптопровайдер (используется для технологического Центра Сертификации, сертификатов веб-сервера и администратора инициализации). Доступные для выбора значения: 'EMBEDDED' и 'CRYPTO_PRO'.
initial_cryptography_key_algorithm	'RSA'	Алгоритм ключа (используется для технологического Центра сертификации и сертификатов веб-сервера и администратора инициализации). Доступные для выбора значения алгоритмов ключа: <ul style="list-style-type: none"> Для стандартного провайдера (EMBEDDED) - 'RSA' и 'ECDSA' Для провайдера СКЗИ «КриптоПро CSP» (CRYPTO_PRO) - 'RSA' и 'GOST_R_34_10_2012'
initial_cryptography_key_bits	'4096'	Длина ключа (используется для технологического Центра сертификации и сертификатов веб-сервера и администратора инициализации).
initial_cryptography_hash_algorithm	'SHA512'	Хеш алгоритм (используется для технологического Центра сертификации). Для стандартного провайдера (EMBEDDED): <ul style="list-style-type: none"> для алгоритма ключа 'RSA' доступны алгоритмы хэширования 'SHA1', 'SHA256', 'SHA384', 'SHA512' для алгоритма ключа 'ECDSA' доступны алгоритмы хэширования 'SHA1', 'SHA256', 'SHA384', 'SHA512' Для провайдера КриптоПро (CRYPTO_PRO): <ul style="list-style-type: none"> для алгоритма ключа 'GOST_R_34_10_2012' доступен алгоритм хэширования 'GOST_R_34_11_2012' для алгоритма ключа 'RSA' доступны алгоритмы хэширования 'SHA1', 'SHA256', 'SHA384', 'SHA512'

Параметр	Значение параметра по умолчанию	Описание
initial_ca_common_name	'INITIAL_CA'	Subject DN сертификата технологического Центра сертификации.
initial_admin_principal	'INITIAL_ADMIN'	Имя учетной записи администратора инициализации.
initial_client_password	'INITIAL'	Пароль от pkcs12 контейнера сертификата администратора инициализации.
initial_server_password	'INITIAL'	Пароль от pkcs12 контейнера сертификата веб-сервера.
certificate_server_name	'server'	Шаблон имени файлов сертификата и закрытого ключа сертификата Веб-сервера.
issuers_name	'issuers'	Шаблон имени файла активных издателей.
Переменные окружения, используемые logs-service		
archive_cron	'0 0 0 1 * *'	CRON-выражение, по которому запускается - архивация журнала событий.
archive_enabled	'true'	Флаг включения архивации (true или false).
archive_millis_ago	'15778800000'	Период архивации, мс (архивировать записи старше...).
Переменные окружения, используемые security-service		
session_max_count	'100'	Максимальное число одновременных сессий аккаунта в виде натурального числа. При указании значения «-1» ограничение на количество одновременных сессий пользователя будет отсутствовать.
token_expire	'18000'	Срок действия JWT-токена (маркера доступа), мс. Маркер доступа предназначен для подтверждения подлинности учетной записи после аутентификации по сертификату. Маркер доступа содержится в заголовке каждого запроса пользователя.
refresh_expire	'86400000'	Срок действия JWT-токена обновления (маркер обновления), мс. Маркер обновления предназначен для выработки нового маркера доступа после истечения срока его действия.
sign_provider	'EMBEDDED'	Провайдер подписи маркера доступа (выбирается между стандартным - 'EMBEDDED', СКЗИ «КриптоПро CSP» - 'CRYPTO_PRO' и 'ALADDIN_JCP' для кппровайдера Aladdin JCP).
sign_key_algorithm	'RSA'	Алгоритм ключа подписи маркера доступа. Для стандартного провайдера доступны алгоритмы 'RSA' и 'ECDSA'. Для провайдера СКЗИ «КриптоПро CSP» доступны алгоритмы 'RSA' и 'GOST_R_34_10_2012'. Для провайдера Aladdin JCP доступен алгоритм 'GOST_R_34_10_2012'.
sign_key_length	'2048'	Длина ключа подписи маркера доступа

Параметр	Значение параметра по умолчанию	Описание
sign_hash_algorithm	'SHA512'	Алгоритм хэширования подписи. Для стандартного провайдера (EMBEDDED): <ul style="list-style-type: none"> для алгоритма ключа 'RSA' доступны алгоритмы хэширования 'SHA1', 'SHA256', 'SHA512', 'SHA384' для алгоритма ключа 'ECDSA' доступны алгоритмы хэширования 'SHA1', 'SHA256', 'SHA512', 'SHA384' Для провайдера СКЗИ «КриптоПро CSP» (CRYPTO_PRO): <ul style="list-style-type: none"> для алгоритма ключа 'GOST_R_34_10_2012' доступен алгоритм хэширования 'GOST_R_34_11_2012' для алгоритма ключа 'RSA' доступны алгоритмы хэширования 'SHA1', 'SHA256', 'SHA512', 'SHA384' Для провайдера Aladdin JCP (ALADDIN_JCP): <ul style="list-style-type: none"> для алгоритма ключа 'GOST_R_34_10_2012' доступен алгоритм хэширования 'GOST_R_34_11_2012'
block_inactive_account_delay	'0'	Период неактивности (мс), после которого учетные записи операторов блокируются. Значение по умолчанию - 0, обозначающее отсутствие ограничения на неактивность учетных записей операторов. Операциями, обновляющими дату и время последней активности пользователя, являются: <ul style="list-style-type: none"> успешная аутентификация, включая аутентификацию в Центре сертификации и Центре регистрации; успешное обновление маркера доступа, включая его обновление в Центре сертификации и Центре регистрации.
block_inactive_account_cron	'0 0 0 * * *'	CRON-выражение, определяющее расписание запуска блокировки учетных записей операторов, период неактивности которых равен или превышает указанное в параметре «block_inactive_account_delay» значение. Значение по умолчанию - запуск каждую полночь.
Переменные окружения, используемые api-gateway-service		
max_requests_count	'30'	Максимальное число параллельных HTTP запросов. При превышении числа запросов в систему данного значения, для последующих запросов будет возвращаться HTTP код ошибки 429 (Слишком много запросов). Настраивается разработчиком Центра сертификации Aladdin eCA, редактировать не следует.
Переменные окружения, используемые subjects-service		
ldap_automatically_certificates_publication_enable	'true'	Флаг включения автоматической публикации сертификатов, требующих публикации. Возможные значения: true/false.
ldap_automatically_certificates_publication_cron	'0 0 * * * *'	CRON выражение, по которому запускается автоматическая публикация сертификатов, требующих публикации. Значение по умолчанию - '0 0 * * * *', обозначающее запуск публикации сертификатов, ожидающих её, каждый час.

5.3 Создание и настройка базы данных

Перед установкой Центра сертификации Aladdin eCA необходимо создать и настроить базу данных одним из следующих способов:

- В автоматическом режиме посредством запуска скрипта (в результате будет создана база данных с параметрами, указанными в конфигурационном файле `/opt/aecaCa/scripts/config.sh`). Порядок создания и настройки базы данных в автоматическом режиме приведен в подразделе 5.3.1.
- В ручном режиме (в результате будет создана база данных с параметрами, указанными в конфигурационном файле `/opt/aecaCa/scripts/config.sh`). Порядок создания и настройки базы данных в ручном режиме для PostgreSQL приведен в подразделе 5.3.2, а для Jatoba - в 5.3.3.

После создания и настройки базы данных пароль пользователя базы данных, заданный в конфигурационном файле `/opt/aecaCa/scripts/config.sh` в параметре `database_password`, отображается в зашифрованном виде. Шифрование пароля выполняется по алгоритму AES-256 с использованием автоматически сгенерированного в файле `/opt/aecaCa/scripts/key` ключа шифрования.

База данных предназначена для хранения информации:

- об учетных записях;
- о сертификатах;
- сведений о субъектах;
- сведений о ресурсных системах;
- о шаблонах;
- журнала событий;
- сведений о лицензии;
- профили сертификатов;
- профили конечных сущностей;
- центры сертификатов;
- настройки оповещения пользователей по e-mail об истечении срока действия сертификата;
- о ролях пользователей;
- о группах субъектов;
- о дискретных правах, определенных для ролей пользователей;
- Security Groups.

5.3.1 Создание и настройка базы данных в автоматическом режиме

- Предварительно необходимо:
 - распаковать инсталляционный пакет программного компонента в соответствии с подразделом 5.1 настоящего документа;
 - указать параметры создаваемой базы данных в конфигурационном файле `/opt/aecaCa/scripts/config.sh` (см. подраздел 5.2 настоящего руководства).
- Запустите скрипт создания и настройки базы данных с параметрами по умолчанию, выполнив команду от имени суперпользователя (с правами `sudo` или `root`)¹:

```
sudo bash /opt/aecaCa/scripts/database_create.sh
```

¹ Выполнение скрипта требует наличия утилиты `psql` из пакета СУБД (`postgresql`, `postgresql-client`, `postgrespro-std`, `jatoba4-client`). Установка и настройка СУБД описана в разделах 4.2.3, 4.3.3 или 4.4.3 в зависимости от ОС.

В результате выполнения скрипта будет создана База данных с параметрами, указанными в конфигурационном файле `/opt/aecaCa/scripts/config.sh` (имя пользователя, пароль, имя базы данных).

5.3.2 Создание и настройка базы данных PostgreSQL в ручном режиме

Требования к настройке предварительно установленной СУБД PostgreSQL:

- создание пользователя, от имени которого будет осуществляться взаимодействие с СУБД;
- создание базы данных, используемой программным компонентом в процессе работы;
- назначение созданному пользователю полных прав доступа к созданной базе данных.

Возможно использование локальной СУБД или удаленной, доступной для подключений.

- Запустите PostgreSQL, выполнив команду:

```
sudo systemctl start postgresql
```

- Добавьте запуск PostgreSQL в автозагрузку, выполнив команду:

```
sudo systemctl enable postgresql
```

- Зайдите под пользователем «postgres» в СУБД, выполнив команду:

```
sudo -u postgres psql
```

- Создайте пользователя базы данных, выполнив команду:

```
CREATE USER aeca;
```

где `aeca` - задаваемое имя пользователя по умолчанию, в случае указания отличного имени пользователя, требуется соответственно отредактировать конфигурационный файл (см. подраздел 5.2).

- Задайте пароль пользователю, выполнив команды:

```
ALTER USER aeca WITH PASSWORD 'aeca';
```

где `'aeca'` - задаваемый пароль пользователя по умолчанию. В случае указания отличного пароля, требуется соответственно отредактировать конфигурационный файл (см. подраздел 5.2).

Внимание! Пароль не должен содержать специальные символы «|» и «\».

- Создайте базу данных, выполнив команду:

```
CREATE DATABASE aecaca;
```

где `aecaca` - задаваемое имя базы данных по умолчанию, в случае указания отличного имени базы данных, требуется соответственно отредактировать конфигурационный файл (см. подраздел 5.2).

- Назначьте владельцем созданной базы данных созданного пользователя, выполнив команду:

```
ALTER DATABASE aecaca OWNER TO aeca;
```

Наделите созданного пользователя полными правами доступа к созданной базе данных и завершите действия, выполнив команды:

```
GRANT ALL PRIVILEGES ON DATABASE aecaca TO aeca;
```

```
\q
```

- Завершите работу под пользователем «postgres» и выйдите из терминала, выполнив команду:

```
exit
```

- Перезапустите СУБД PostgreSQL, выполнив команду:

```
sudo systemctl restart postgresql
```

Установите расширение `pgcrypto` в БД PostgreSQL, выполнив команду от имени пользователя «postgres» (с правами root):

```
sudo -u postgres psql -c "CREATE EXTENSION IF NOT EXISTS pgcrypto WITH SCHEMA pg_catalog;" -d aecaca
```

где **aecaca** - имя созданной базы данных.

5.3.3 Создание и настройка базы данных Jatoba в ручном режиме

Требования к настройке предварительно установленной СУБД Jatoba:

- создание пользователя, от имени которого будет осуществляться всё взаимодействие с СУБД;
- создание базы данных, используемой Программой в процессе работы;
- назначение созданному пользователю полных прав доступа к созданной базе данных.

Возможно использование локальной СУБД или удаленной, доступной для подключений.

- Запустите Jatoba, выполнив команду:

```
sudo systemctl start jatoba-[версия]
```

Добавьте запуск Jatoba в автозагрузку, выполнив команду:

```
sudo systemctl enable jatoba-[версия]
```

- Зайдите под пользователем «postgres» в Jatoba, выполнив команду:

РЕД ОС

```
sudo -u postgres psql
```

SberLinux OS Server

```
sudo dnf install <наименование пакета>.rpm
```

Astra Linux SE

```
sudo -u postgres psql
```

Альт Сервер

```
sudo - postgres -s /bin/bash
-bash-4.4$ /usr/jatoba-[версия]/bin/psql
psql
```

- Создайте пользователя базы данных, выполнив команды:

```
CREATE USER aeca;
```

где **aeca** - задаваемое имя пользователя.

- Задайте пароль пользователю, выполнив команды:

```
ALTER USER aeca WITH PASSWORD 'aeca';
```

где **'aeca'** - задаваемый пароль пользователя.

Внимание! Пароль не должен содержать специальные символы «|» и «\».

- Создайте базу данных, выполнив команду:

```
CREATE DATABASE aecaca;
```

где **aecaca** - задаваемое имя базы данных.

- Назначьте владельцем созданной базы данных созданного пользователя, выполнив команду:

```
ALTER DATABASE aecaca OWNER TO aeca;
```

Наделите созданного пользователя полными правами доступа к созданной базе данных и завершите действия, выполнив команды:

```
GRANT ALL PRIVILEGES ON DATABASE aecaca TO aeca;
```

```
\q
```

- Завершите работу под пользователем «postgres» и выйдите из терминала, выполнив команду:

```
exit
```

- Перезапустите СУБД Jatoba, выполнив команду:

```
sudo systemctl restart jatoba-[версия]
```

- Установите расширение pgcrypto в БД Jatoba, выполнив команду от имени пользователя «postgres» (с правами root):

```
sudo -u postgres psql -c "CREATE EXTENSION IF NOT EXISTS pgcrypto WITH SCHEMA pg_catalog;" -d aecaca
```

где **aecaca** - имя созданной базы данных.

5.4 Установка программы

Для инициализации процесса установки Центра сертификации Aladdin eCA необходимо запустить скрипт с правами суперпользователя (от имени пользователя root, либо с использованием sudo)¹:

```
sudo bash /opt/aecaCa/scripts/install.sh
```

В случае запуска от имени пользователя, не имеющего соответствующих привилегий, будет выведено сообщение, после которого работа инсталлятора завершится:

```
"This script must be run as root!"
```

При использовании Astra Linux Special Edition и наличии мандатных политик² может быть выведено сообщение:

```
ВАЖНО: error obtaining MAC configuration for user "aeca"
```

В данном случае явно назначьте классификационную метку пользователю **aeca**, выполнив команду:

```
sudo pdpl-user -l 0:0 aeca
```

Повторно запустите скрипт установки. После инициализации процесса установки интерактивный инсталлятор будет запущен и пользователю будет предложено (в случае, если ранее на сервере был установлен Центр сертификации Aladdin eCA):

- Установить Центр сертификации Aladdin eCA.
- Установить обновление Центра сертификации Aladdin eCA.
- Завершить работу инсталлятора.

Подтвердите выбор действия, вводом цифры «1» и процесс установки продукта будет запущен.

В случае, если в конфигурационном файле **/opt/aecaCa/scripts/config.sh** не определён используемый веб-сервер или введено неверное значение параметра **webserver**, то в процессе установки пользователю будет предложено выбрать используемый веб-сервер:

- apache;
- nginx;
- cpnginx.

¹ Выполнение скрипта требует наличия утилиты **psql** из пакета СУБД (**postgresql**, **postgresql-client**, **postgrespro-std**, **jatoba4-client**). Установка и настройка СУБД описана в разделах 4.2.3, 4.3.3 или 4.4.3 в зависимости от ОС.

² Подробная информация по аутентификации в СУБД PostgreSQL для Astra Linux Special Edition приведена в <https://wiki.astralinux.ru/pages/viewpage.action?pagelId=238751148>

Подтвердите выбор действия, вводом цифры «1», «2» или «3».

В случае, если в конфигурационном файле `/opt/aecaCa/scripts/config.sh` не определено расположение конфигурации выбранного веб-сервера (параметр `webserver_path`), то в процессе установки пользователю будет предложено ввести расположение (конфигурация `nginx` располагается по пути `etc/nginx`; конфигурация `apache` располагается: для Astra Linux по пути `/etc/apache2`, для РЕД ОС и SberLinux OS Server по пути `/etc/httpd`; для Альт Сервера конфигурация `apache` располагается по пути `/etc/httpd2/conf`; конфигурация `cpnginx` располагается по пути `/etc/opt/cprosp/cpnginx`).

В процессе установки программы осуществляется:

- создание системного пользователя и соответствующей группы, от имени которых функционирует продукт;
- установка прав для создаваемого пользователя продукта;
- подготовка, установка параметров и служебных сервисов;
- запуск служебных сервисов;
- запись номера сборки Центра сертификации Aladdin eCA в базу данных¹;
- создание и выпуск сертификата технологического центра сертификации;
- выпуск сертификата веб-сервера технологического центра сертификации;
- создание учётной записи и выпуск сертификата администратора инициализации.

Ход установки программного компонента отображен в виде горизонтальной шкалы с указанием процентов выполнения установки.

В результате успешной установки программы:

- В каталоге `/opt/aecaCa/dist/certificates/account` (значение по умолчанию параметра `certificates_account_path` конфигурационного файла `config.sh`) будет выпущен сертификат администратора инициализации (с использованием выбранного алгоритма - RSA, ECDSA или ГОСТ²) - контейнер закрытого ключа `INITIAL_ADMIN.p12` (`INITIAL_ADMIN_GOST.p12`) (имя контейнера задано в параметре `initial_admin_principal` конфигурационного файла).

- Выпущен технологический сертификат веб-сервера (с использованием выбранного алгоритма - RSA, ECDSA или ГОСТ) и применён в качестве сертификата веб-сервера технологического Центра сертификации;

- Создан технологический центр сертификации `INITIAL_CA` (`INITIAL_CA_GOST`) (значение задано в параметре `initial_ca_common_name` конфигурационного файла).

Внимание! Технологический Центр сертификации использовать для выпуска сертификатов запрещено.

После первичной установки Центра сертификации Aladdin eCA системному пользователю `aeca` будет назначена командная оболочка `/sbin/nologin`, которая запрещает интерактивный вход в ОС. При обновлении ПО командная оболочка не меняется. Чтобы сменить командную оболочку, выполните команду:

```
sudo usermod -s /bin/bash aeca
```

В случае возникновения ошибки установка будет прекращена, сообщение об ошибке будет выведено в консоль пользователя.

¹ Значение номера сборки записывается в таблицу «build_info» схемы «aeca_info».

² Маркеры доступа будут подписаны по алгоритму ГОСТ Р 34.11-2012/34.10-2012 256 Бит.

5.5 Порядок совместной установки компонентов программного средства на одном сервере

В Центре сертификатов доступа поддерживается совместная работа Центра сертификации Aladdin eCA, Центра регистрации Aladdin eRA и Центра валидации Aladdin eVA (далее в разделе - компоненты программного средства) на одном хосте. Также поддерживается совместная работа двух выбранных компонентов программного средства на одном хосте.

Порядок совместной установки всех компонентов программного средства на одном хосте ¹:

- Выполните подготовку среды функционирования компонентов программного средства на хосте (см. раздел 4 настоящего руководства).
- Введите хост в домен ресурсной системы в соответствии с документацией производителя используемой ОС.
- Определите имя хоста (hostname) для каждого компонента программного средства. Имя хоста компонента формируется путем добавления к реальному сетевому имени хоста, на котором выполняется совместная установка компонентов программного средства, префикса, идентифицирующего данный компонент. Например, имя хоста `ra.eca-host` для Центра регистрации Aladdin eRA, в котором `ra` это префикс, а `eca-host` это реальное имя хоста (hostname), на котором выполняется совместная установка компонентов программного средства.
- Отредактируйте файл `etc/hosts`, сопоставив в нем имена хостов компонентов программного средства, включая доменную часть, IP-адресу `127.0.0.1`.

В указанном ниже примере содержания файла `etc/hosts` префикс `ca` идентифицирует Центр сертификации Aladdin eCA, `va` - Центр валидации Aladdin eVA, `ra` - Центр регистрации Aladdin eRA, а `eca-host.ad.local` - это полное доменное имя хоста (FQDN), на котором выполняется совместная установка компонентов программного средства.

```
127.0.0.1 ca.eca-host.ad.local
127.0.0.1 va.eca-host.ad.local
127.0.0.1 ra.eca-host.ad.local
```

При совместной установке компонентов программного средства в среде ОС Astra Linux Special Edition и взаимодействии с доменной службой каталогов Samba DC, Альт Домен или MS AD заполнение файла `etc/hosts` не выполняется. При этом на в DNS-сервисе домена ресурсной системы необходимо добавить DNS-записи, сопоставив выбранные имена хостов компонентов программного средства IP-адресу хоста, на котором выполняется совместная установка компонентов программного средства.

Формат команды для добавления DNS-записи на контроллере домена:

```
sudo samba-tool dns add [IP-адрес контроллера домена] [Домен] [Имя хоста компонента] A [IP-адрес хоста] -U [Имя учетной записи администратора домена]
```

Пример команды:

```
sudo samba-tool dns add 192.168.86.129 ad.local ca.eca-host A 192.168.86.138 -U admin_dc
```

- Установите Центр сертификации Aladdin eCA, указав в конфигурационном файле в параметре `hostname` выбранное для Центра сертификации Aladdin eCA имя хоста, включая доменную часть (см. разделы 5.1-5.4 настоящего руководства) (например, `ca.eca-host.ad.local`).

¹ Аналогичным образом выполняется совместная установка двух выбранных компонентов на одном сервере.

- Установите сертификат администратора инициализации технологического Центра сертификации (см. раздел 7 настоящего руководства):
 - Контейнер закрытого ключа `INITIAL_ADMIN.p12` в хранилище сертификатов веб-браузера, если сертификат был выпущен с использованием алгоритмов RSA или ECDSA.
 - Контейнер закрытого ключа `INITIAL_ADMIN_GOST.p12` в личное хранилище пользователя с помощью утилиты `cptools` из состава СКЗИ «КриптоПро CSP, если сертификат был выпущен с использованием алгоритмов ГОСТ.
- Подключитесь к веб-интерфейсу Центра сертификации Aladdin eCA (см. раздел 7 настоящего руководства) и пройдите аутентификацию с помощью сертификата администратора инициализации. В качестве адреса Центра сертификации Aladdin eCA необходимо указать имя хоста компонента, включая доменную часть (например, `https://ca.eca-host.ad.local`).
- Установите лицензию и выполните инициализацию Центра сертификации Aladdin eCA (см. разделы 2.2 и 3 документа «Центр сертификатов доступа Aladdin Enterprise Certificate Authority Certified Edition. Руководство администратора. Часть 2. Функции управления Центром сертификации Aladdin Enterprise Certification Authority» (далее в разделе - часть 2 руководства администратора)).
- Выполните подключение ресурсной системы к Центру сертификации Aladdin eCA (см. раздел 7.9.1 части 2 руководства администратора).
- Создайте учетную запись пользователя локального ресурса (см. раздел 7.6.1 части 2 руководства администратора) или субъекта ресурсной системы (см. раздел 7.8.8 части 2 руководства администратора) с ролью «Администратор» для управления компонентами программного средства и выпустите для нее сертификат с закрытым ключом по шаблону «User» (см. разделы 7.6.7 или 7.8.7 части 2 руководства администратора).
- Создайте субъект локальной ресурсной системы для веб-сервера центра сертификации, указав в атрибутах «Common Name» и «DNS Name» имя хоста компонента, выбранное для Центра сертификации Aladdin eCA, включая доменную часть (см. раздел 7.8.5 части 2 руководства администратора), выпустите для созданного субъекта сертификат с закрытым ключом по шаблону «WEB-Server» и установите данный сертификат в качестве сертификата веб-сервера Центра сертификации Aladdin eCA (см. разделы 7.8.7 и 7.13.1 части 2 руководства администратора).
- Создайте субъект локальной ресурсной системы для веб-сервера Центра регистрации Aladdin eRA, указав в атрибутах «Common Name» и «DNS Name» имя хоста компонента, выбранное для Центра регистрации Aladdin eRA, включая доменную часть (см. раздел 7.8.5 части 2 руководства администратора) и выпустите для созданного субъекта сертификат с закрытым ключом по шаблону «WEB-Server» (см. раздел 7.8.7 части 2 руководства администратора).
- Создайте для Центра регистрации Aladdin eRA учетную запись пользователя с ролью «Администратор» и выпустите для нее сертификат с закрытым ключом по шаблону «User» (см. разделы 7.6.1 и 7.6.7 части 2 руководства администратора).
- Создайте субъект локальной ресурсной системы для веб-сервера Центра валидации Aladdin eVA, указав в атрибутах «Common Name» и «DNS Name» имя хоста компонента, выбранное для Центра валидации Aladdin eRA, включая доменную часть (см. раздел 7.8.5 части 2 руководства администратора) и выпустите для созданного субъекта сертификат с закрытым ключом по шаблону «WEB-Server» (см. раздел 7.8.7 части 2 руководства администратора).
- Создайте для Центра валидации Aladdin eVA учетную запись пользователя с ролью «Администратор» и выпустите для нее сертификат с закрытым ключом по шаблону «User» (см. разделы 7.6.1 и 7.6.7 части 2 руководства администратора). Данный сертификат будет использоваться в дальнейшем для подключения к Центру сертификации Aladdin eCA.

- Создайте пользователя-службу HTTP и keytab-файл¹ на контроллере домена ресурсной системы:
 - Для Центра регистрации Aladdin eRA (см. раздел 3.4 документа «Руководство администратора. Часть 5. Центр регистрации Aladdin Enterprise Registration Authority» (далее в разделе - часть 5 руководства администратора).
 - Для Центра валидации Aladdin eVA (см. раздел 3.4 документа «Руководство администратора. Часть 4. Центр валидации Aladdin Enterprise Validation Authority» (далее в разделе - часть 4 руководства администратора).

При совместной установке компонентов в среде ОС Astra Linux Special Edition и взаимодействии с доменной службой каталогов Samba DC, Альт Домен или MS AD создайте только один keytab-файл. При создании keytab-файла в соответствующей для доменной службы каталогов команде укажите полное доменное имя хоста (FQDN), на котором выполняется совместная установка компонентов программного средства (например, eca-host.ad.local).

Настройку HTTP-службы в доменной службе каталогов ALD PRO необходимо выполнять через интерфейс Free IPA.

- Установите Центр регистрации Aladdin eRA (см. раздел 4 части 5 руководства администратора), указав в конфигурационном файле:

- В параметре `hostname` выбранное для Центра регистрации Aladdin eRA имя хоста.
- В параметре `aeca_ca_host` выбранное для Центра сертификации Aladdin eCA имя хоста.
- Значения параметров `aeca_user`, `aeca_group` и `database_username` должны отличаться от значений этих же параметров в конфигурационном файле Центра сертификации Aladdin eCA.

Остальные параметры конфигурационного файла указываются в соответствии с разделом 4.2 части 5 руководства администратора.

- Установите Центр валидации Aladdin eVA (см. раздел 4 части 4 руководства администратора), указав в конфигурационном файле:

- В параметре `hostname` выбранное для Центра валидации Aladdin eVA имя хоста.
- Значения параметров `aeca_user`, `aeca_group` и `database_username` должны отличаться от значений этих же параметров в конфигурационных файлах Центра сертификации Aladdin eCA и Центра регистрации Aladdin eRA.

Остальные параметры конфигурационного файла указываются в соответствии с разделом 4.3 части 4 руководства администратора.

¹ Keytab-файл используется для аутентификации доменных пользователей в Центре регистрации Aladdin eRA с использованием Kerberos без ввода пароля.

6 ЗАПУСК И ОСТАНОВКА ПРОГРАММЫ

Центр сертификации Aladdin eCA запускается автоматически:

- в случае выполнения успешной установки программы.
- в случае выполнения успешного обновления программы.
- после запуска ОС.

Для проверки состояния Центра сертификации Aladdin eCA в терминале выполните команду с правами суперпользователя:

```
sudo systemctl status aeca-ca.service
```

Возможные варианты ответа:

- active (running) - сервис запущен, с перечислением модулей и их статуса (ожидание запуска, успешно запущен, не удалось запустить сервис);
- inactive (dead) - сервис остановлен, с выводом информации о последних запущенных модулях.

Для проверки автозагрузки программы выполните команду с правами суперпользователя:

```
sudo systemctl is-enabled aeca-ca.service
```

Для добавления программы в автозагрузку выполните команду с правами суперпользователя:

```
sudo systemctl enable aeca-ca.service
```

Для запуска программы выполните команду с правами суперпользователя:

```
sudo systemctl start aeca-ca.service
```

Для перезапуска программы выполните команду с правами суперпользователя:

```
sudo systemctl restart aeca-ca.service
```

При запуске Центра сертификации Aladdin eCA выполняются следующие проверки:

- Проверка возможности подключения к базе данных. Если не удаётся подключиться к базе данных, то программа не запускается.
- Проверка соответствия номера своей сборки и значения номера сборки, указанной в базе данных:
 - Если в базе данных отсутствует номер сборки, то программа не запускается.
 - Если номер сборки не равен номеру сборки программы, то программа завершает запуск с ошибкой «Текущая версия схемы базы данных не позволяет выполнить запуск службы. Текущая версия схемы базы данных: X.X.X.X. Необходимая версия схемы базы данных: Y.Y.Y.Y.», где «X.X.X.X» - номер сборки указанный в базе данных, а «Y.Y.Y.Y» - номер сборки запускаемой программы.
- Проверка целостности контейнеров закрытого ключа всех Центров сертификации программы. Результат проверки целостности для каждого контейнера закрытого ключа записывается в журнал событий:
 - Событие с кодом CAENV076 - при успешной проверке целостности.
 - Событие CAENV077 - при неуспешной проверке целостности.

Для завершения работы Центра сертификации Aladdin eCA выполните команду с правами суперпользователя:

```
sudo systemctl stop aeca-ca.service
```

Центр сертификации Aladdin eCA при остановке отключает от веб-сервера свою конфигурацию. В

результате отключения от веб-сервера конфигурации закрываются порты¹, используемые для доступа к программе (определяются параметрами «http_port» и «https_port» конфигурационного файла /opt/aecaCa/scripts/config.sh), если данные порты не используются иными программами.

Модули Центра сертификации Aladdin eCA запускаются поочередно в порядке, приведенном в таблице ниже (Таблица 8).

Таблица 8 - Модули программы

Порядок запуска	Исполняемый файл	Наименование	Назначение
1	logs-service.jar	Модуль журнала событий	Обеспечивает фиксацию событий в журнале и получение событий из журнала, просмотра и поиск Записей журнала событий, экспорт и архивацию записей журнала событий
2	storage-service.jar	Модуль хранения данных	Обеспечивает хранение и управление файлами сертификатов
3	templates-service.jar	Модуль шаблонов	Обеспечивает просмотр, создание, редактирование и удаление шаблонов сертификатов
4	subjects-service.jar	Модуль работы с субъектами	Обеспечивает взаимодействие с группами безопасности и субъектами
5	license-service.jar	Модуль лицензирования	Обеспечивает управление лицензиями программы
6	export-service.jar	Модуль экспорта данных	Обеспечивает управление экспортом файлов программы
7	security-service.jar	Модуль безопасности	Предназначен для идентификации и аутентификации пользователей программы, управления учетными записями пользователей программы, предоставления информации о пользователях программы.
8	ldap-service.jar	Модуль работы с LDAP	Обеспечивает взаимодействие с ресурсными системами и обеспечивает публикацию сертификатов в ресурсную систему, а также получение данных из ресурсной системы
9	event-delivery-service.jar	Модуль оповещения пользователей	Предназначен для оповещения посредством рассылки уведомлений по адресам электронной почты владельцев сертификатов
10	certificate-authority-service.jar	Модуль сертификатов	Обеспечивает создание сертификата, подпись сертификата (включая цепочки сертификатов), генерацию CRL, валидацию сертификата, взаимодействие уполномоченного пользователя с контейнерами и точками распространения.
11	publisher-service.jar	Модуль публикации	Обеспечивает обслуживание точек публикации CRL, Delta CRL и AIA
12	validation-authority-service.jar	Модуль валидации	Обеспечивает взаимодействия с точками распространения, а также для валидации сертификатов
13	external-integration-service.jar	Модуль внешних интеграций	Предназначен для предоставления пользователям или внешним системам доступа к программным интерфейсам (публичный API) программы, реализуемым другими модулями.
14	settings-service.jar	Модуль настроек	Обеспечивает управление жизненным циклом программы, её состоянием и параметрами (данные о продукте, конфигурация серверного сертификата SSL, разрешенные издатели сертификатов)
15	routes-service.jar	Модуль управления	Предоставляет пользовательские веб-интерфейсы, обеспечивает разграничение доступа на основе ролей пользователей

¹ Порты будут закрыты только в том случае, если они были открыты Центром сертификации Aladdin eCA.

Порядок запуска	Исполняемый файл	Наименование	Назначение
16	api-gateway-service.jar	Модуль проксирования	Предназначен для перенаправления поступающих в программу запросов в нужный сервис (на основании данных, указанных в URL запроса), а также для перенаправления запросов к модулю безопасности с целью аутентификации пользователя
17	x509-provider-service.jar	Модуль аутентификации по сертификату	Предназначен для аутентификации пользователей в программе по сертификату доступа.

7 ПОДКЛЮЧЕНИЕ К ВЕБ-ИНТЕРФЕЙСУ

7.1 Общие сведения

Веб-интерфейс представляется собой графический интерфейс в виде совокупности динамических веб-страниц, отображаемых в веб-браузере. Веб-интерфейс реализован клиентским компонентом Центра сертификации Aladdin eCA и предназначен для управления серверным компонентом Центра сертификации Aladdin eCA (выполнения доступных пользователю в рамках его полномочий действий).

Подключение к веб-интерфейсу Центра сертификации Aladdin eCA выполняется из веб-браузера удаленно по сети передачи данных с выделенного компьютера, на котором развернута среда функционирования.

Канал управления является защищенным — организован по протоколу HTTPS/TLS с двусторонней аутентификацией и шифрованием передаваемых данных. Идентификация и аутентификация пользователей выполняется по предъявленному сертификату, который должен быть предварительно установлен в хранилище веб-браузера или хранилище сертификатов используемой ОС. Пример установки сертификата администратора инициализации из контейнера закрытого ключа PKCS#12 (по умолчанию `INITIAL_ADMIN.p12`) приведен в подразделе 7.2.

При использовании СКЗИ «КриптоПро CSP» канал взаимодействия клиентского и серверного компонента программного средства должен быть организован по протоколу TLS ГОСТ с использованием отечественных криптографических алгоритмов.

Для этого на компьютере, предназначенном для подключения к веб-интерфейсу, должны быть выполнены следующие действия:

- Установлен криптопровайдер СКЗИ «КриптоПро CSP» в соответствии с инструкцией, описанной в разделе 2 документа «СКЗИ «КриптоПро CSP». Руководство администратора безопасности. Использование СКЗИ под управлением ОС Linux» ЖТЯИ.00101-03 91 03.
- Установлена клиентская лицензия СКЗИ «КриптоПро CSP», дающая право использовать двустороннюю аутентификацию по протоколу TLS. Порядок установки лицензии описан в разделе 4 документа «СКЗИ «КриптоПро CSP». Руководство администратора безопасности. Использование СКЗИ под управлением ОС Linux» ЖТЯИ.00101-03 91 03.
- Сертификат администратора инициализации из контейнера закрытого ключа PKCS#12, выпущенного с использованием алгоритмов ГОСТ при установке программы (по умолчанию `INITIAL_ADMIN_GOST.p12`), должен быть установлен в личное хранилище пользователя с помощью утилиты `cptools` из состава СКЗИ «КриптоПро CSP». Порядок установки сертификата из контейнера закрытого ключа приведен в подразделе 2.6.5 документа «СКЗИ «КриптоПро CSP». Инструкция по использованию графического приложения Инструменты КриптоПро (cptools)» ЖТЯИ.00101-03 92 06.
- Установлен веб-браузер Chromium с поддержкой TLS ГОСТ из состава ОС. Данный веб-браузер входит в состав базовых репозиториях ОС Astra Linux SE, Альт Сервер, РЕД ОС и SberLinux OS Server.

7.2 Установка сертификата администратора инициализации

После установки Центра сертификации Aladdin eCA сформирован контейнер закрытого ключа PKCS12, содержащий сертификат администратора инициализации технологического центра сертификации. По умолчанию контейнер расположен в каталоге `/opt/aecaCa/dist/certificates/account/` (каталог определен параметром `certificates_account_path` конфигурационного файла). Пароль от контейнера с сертификатом определен в параметре `initial_client_password` конфигурационного файла (по умолчанию - «INITIAL»).

Установите сертификат администратора инициализации `INITIAL_ADMIN.p12` (имя контейнера по умолчанию) в доверенное хранилище сертификатов веб-браузера¹.

Порядок установки сертификата администратора инициализации в хранилище веб-браузера Firefox:

- Откройте браузер Firefox - Настройки - Приватность и Защита - Сертификаты (см. Рисунок 1). Нажмите кнопку <Просмотр сертификатов>.

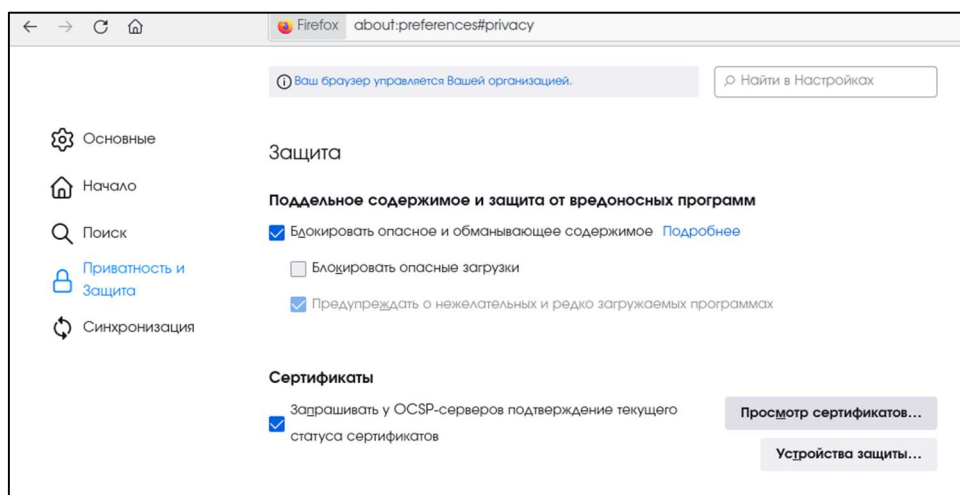


Рисунок 1 - Окно настроек браузера

- На вкладке «Ваши сертификаты» нажмите кнопку <Импортировать> (см. Рисунок 2).

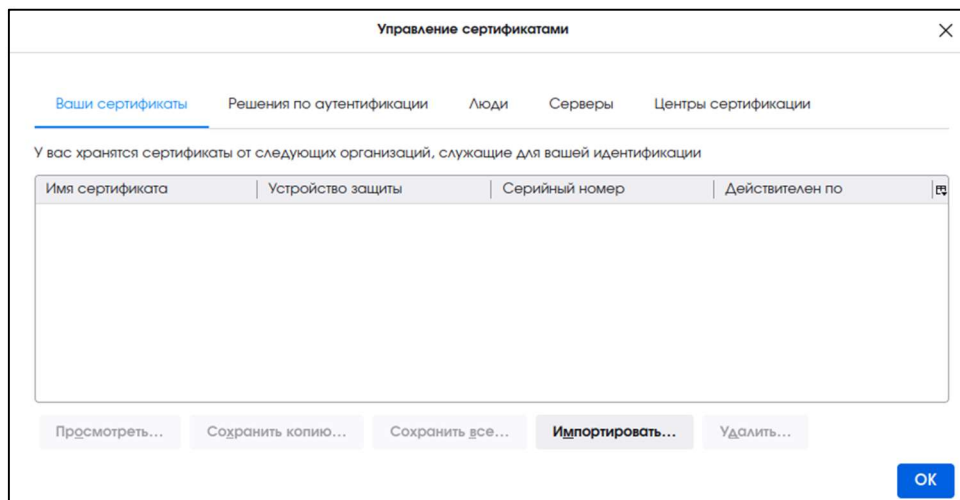


Рисунок 2 - Окно управления сертификатами

- Укажите путь к контейнеру с сертификатом администратора инициализации и нажмите кнопку <Открыть> (см. Рисунок 3).

¹ Сертификат администратора инициализации из контейнера закрытого ключа PKCS#12, выпущенного с использованием алгоритмов ГОСТ, устанавливается в личное хранилище пользователя с помощью утилиты `crttools` из состава СКЗИ «КриптоПро CSP» (см. раздел 6.1).

Внимание! Запрещается каким-либо образом удалять сертификат технологического центра сертификации «INITIAL_CA», созданного при развёртывании Центра сертификации.

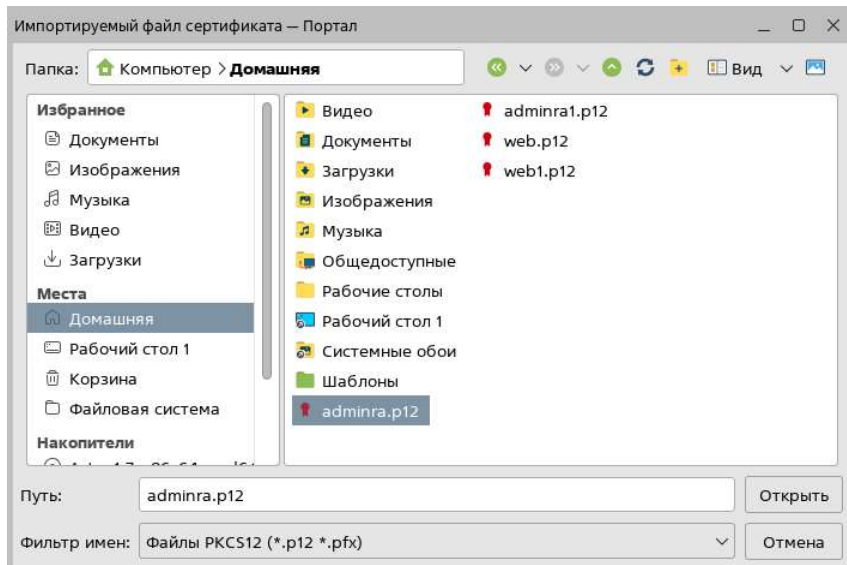


Рисунок 3 - Окно выбора импортируемого файла сертификата

- В открывшемся окне введите пароль от контейнера и нажмите кнопку <Ок> (см. Рисунок 4).

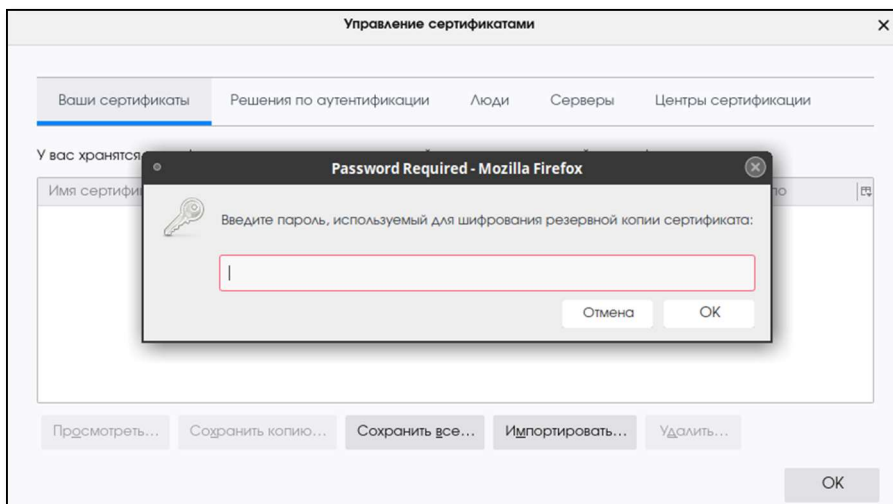


Рисунок 4 - Окно ввода пароля от контейнера

- В результате в окне «Управление сертификатами» появится запись об импортированном сертификате (см. Рисунок 5). Завершите установку сертификата, нажав кнопку <ОК>.

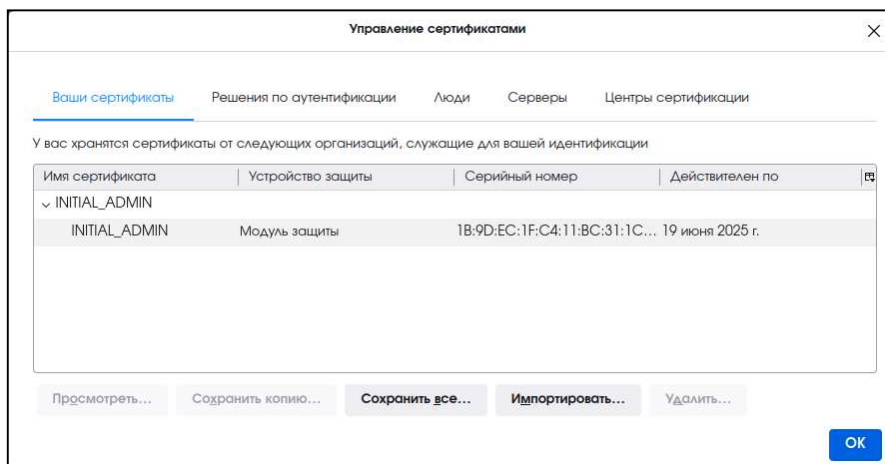


Рисунок 5 - Окно «Управление сертификатами»

7.3 Подключение к веб-интерфейсу

Порядок подключения к веб-интерфейсу:

- Запустите веб-браузер и в адресной строке введите IP-адрес или доменное имя сервера, на котором установлен Центр сертификации Aladdin eCA. Например, `https://172.22.5.21`.
- В открывшемся окне выберите сертификат администратора инициализации (см. Рисунок 6) и нажмите кнопку <OK>.

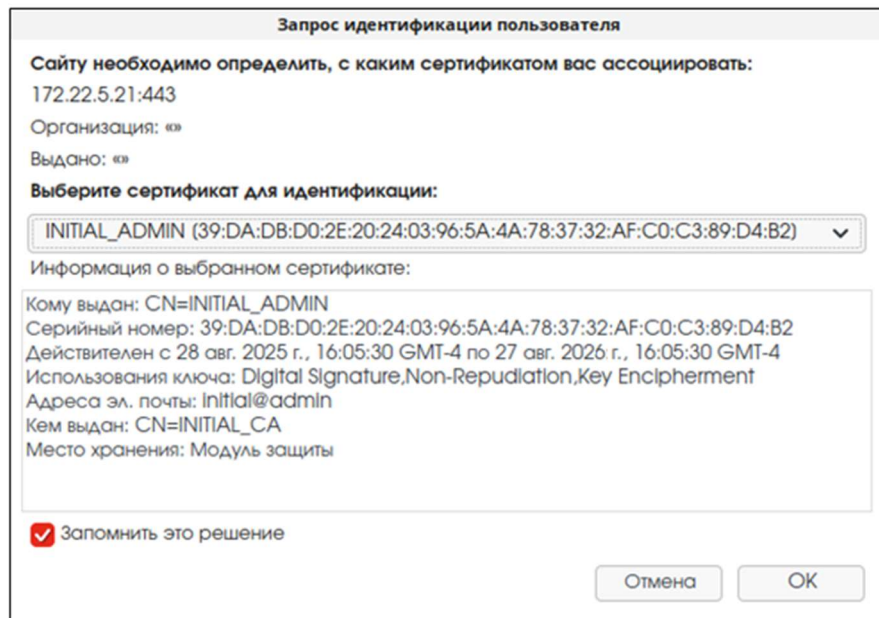


Рисунок 6 - Окно выбора сертификата

- Далее на открывшейся странице с предупреждением системы безопасности (см. Рисунок 7) нажмите кнопку <Дополнительно>, примите риск и продолжите подключение.

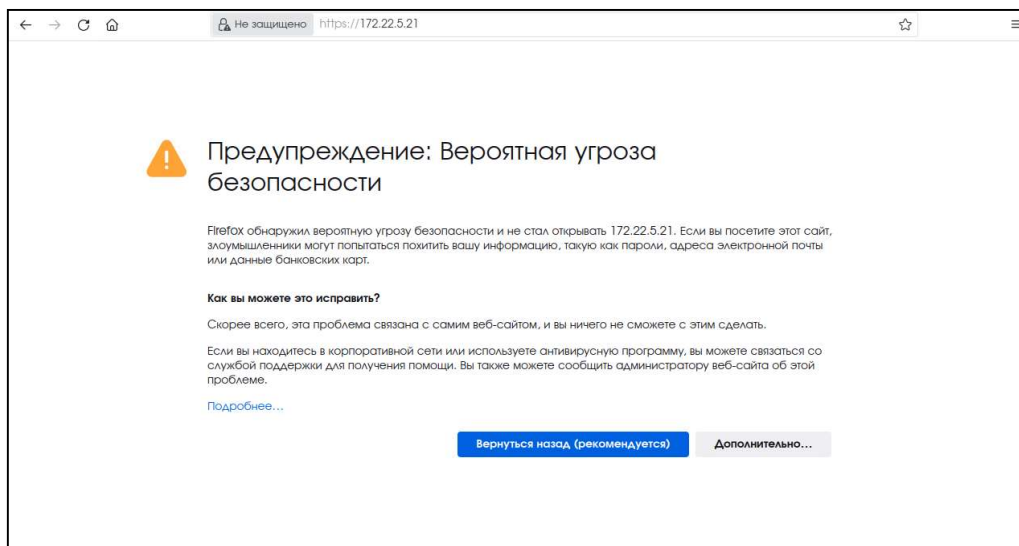


Рисунок 7 - Страница с предупреждением системы безопасности

- В результате вы подключитесь к веб-интерфейсу Центра сертификации Aladdin eCA, где запущен Мастер инициализации (первый шаг - установка лицензии) (см. часть 2 настоящего руководства администратора).

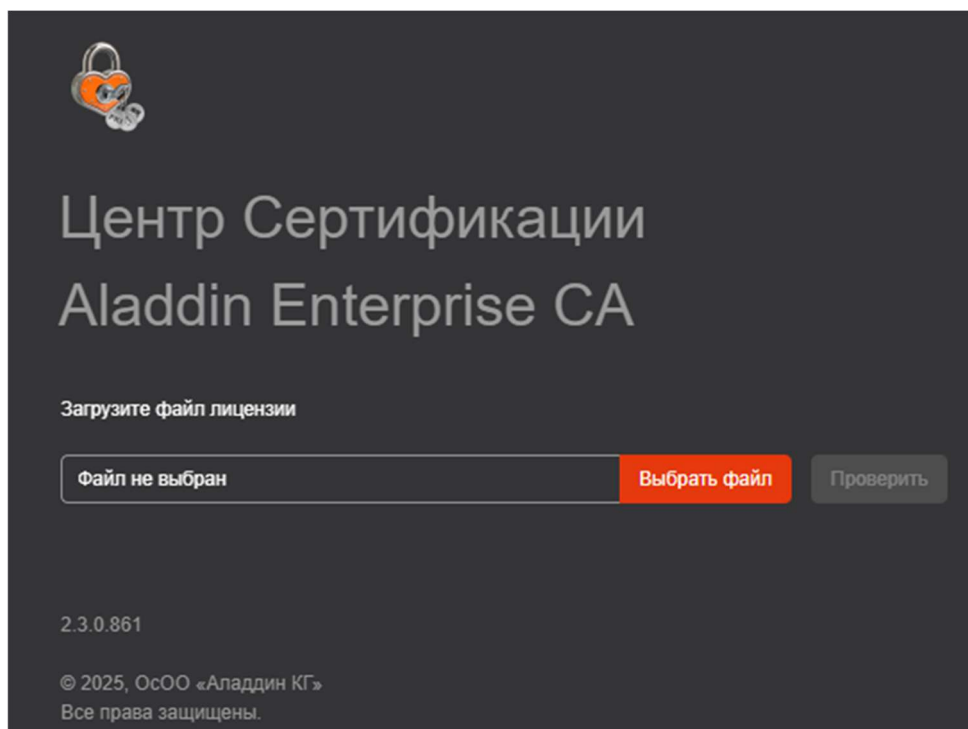


Рисунок 8 - Окно инициализации Центра сертификации. Шаг 1 - выбор лицензии

8 КОНТРОЛЬ ЦЕЛОСТНОСТИ ИСПОЛНЯЕМЫХ ФАЙЛОВ ПРОГРАММЫ

Контроль целостности исполняемых файлов Центра сертификации Aladdin eCA необходим для отслеживания неизменности и контроля состояния файлов, перечень которых приведен ниже:

- все файлы из каталога «/opt/aecaCa/samples» и его подкаталогов;
- все файлы из каталога «/opt/aecaCa/scripts» и его подкаталогов, кроме файлов «config.sh» и «jc_checksum»;
- все «.jar» файлы в каталоге «/opt/aecaCa/services» и его подкаталогах;
- все файлы в каталоге «/opt/aecaCa/static» и его подкаталогах;
- все файлы в каталоге «/opt/aecaCa/bin» и его подкаталогах;
- все файлы в каталоге /opt/aecaCa/digsig и его подкаталогах.

Контроль целостности осуществляется с помощью скрипта integrity_check.sh, находящегося в каталоге скриптов /opt/aecaCa/scripts. Скрипт integrity_check.sh осуществляет проверку целостности исполняемых файлов Центра сертификации Aladdin eCA средствами утилиты «Утилита контроля целостности 2.0» - jcverify¹.

Скрипт integrity_check.sh принимает в качестве опционального входного параметра путь к файлу с контрольными суммами, на основании которого должна выполняться проверка. В случае, если путь к файлу не указан, то по умолчанию будет использоваться файл /opt/aecaCa/scripts/jc_checksum.

Файл с эталонами контрольными суммами jc_checksum формируется при сборке программы с помощью утилиты контроля целостности jcverify.

Для выполнения контроля целостности исполняемых файлов запустите скрипт integrity_check.sh с правами суперпользователя (root или sudo):

```
sudo bash /opt/aecaCa/scripts/integrity_check.sh
```

В данном случае будет использован файл с эталонами контрольных сумм по умолчанию - /opt/aecaCa/scripts/jc_checksum.

После завершения работы скрипта необходимо проанализировать полученные данные.

При успешной проверке целостности будет выведено сообщение: «Успешная проверка контрольных сумм». При этом в журнале событий будет зафиксировано событие с кодом CAENV074 (событие «Успешная проверка контрольных сумм»).

При ошибке проверки целостности будет выведено сообщение «Неуспешная проверка контрольных сумм», а также сообщение об ошибке, генерируемое утилитой jcverify. При этом в журнале событий будет зафиксировано событие с кодом CAENV075 (событие «Неуспешная проверка контрольных сумм»).

¹ Данная утилита включена в состав Центра сертификации (каталог «/opt/aecaCa/bin/jcverify»).

9 СБОР ДИАГНОСТИЧЕСКОЙ ИНФОРМАЦИИ

Сбора диагностической информации компонентов необходим для предоставления в службу поддержки пользователей информации о проблемах в работе программы.

В процессе работы Центра сертификации Aladdin eCA системные службы и компоненты программы регистрируют все производимые действия. Произошедшие события записываются в файлы регистрации событий¹ с расширением `.log`, расположенные в папках соответствующих сервисов, которыми были инициированы события по пути `/opt/aecaCa/dist/logs/` (определяется параметром `logs_base` конфигурационного файла). Максимальный размер лог-файла каждого сервиса перед его архивацией составляет 10 Мбайт (определяется параметром `logs_file_max_size` конфигурационного файла). Срок хранения архивов составляет 10 дней (определяется параметром `logs_max_history` конфигурационного файла). Максимальный общий объем файлов регистрации событий, включая архивы, каждого типа (`access.log` или `service.log`) для каждого сервиса составляет 100 Мбайт (определяется параметром `logs_total_size_cap` конфигурационного файла).

В процессе автоматизированного сбора диагностической информации будет собрана следующая информация:

- О работе сервисов программы (файлы в формате `.log`).
- Конфигурационный файл `/opt/aecaCa/scripts/config.sh`.
- О работе веб-сервера Nginx/Apache (в формате `.log` и `.gz`).
- О работе системы управления базой данных PostgreSQL.
- О работе системы управления базой данных Jatoba.
- О работе ОС (системная).
- Данные системных логов, представленные в таблице 9.

Таблица 9 - Данные системных логов

Системный лог	РЕД ОС и SberLinux OS Server	Astra Linux SE	Альт Сервер
<code>/var/log/audit/</code>	+	+	+
<code>/var/log/samba/</code>	+	+	+
<code>/var/log/httpd/</code>	+	-	-
<code>/var/log/messages/</code>	+	+	+
<code>/var/log/secure/</code>	+	-	-
<code>/var/log/cron/</code>	+	+	-
<code>/var/log/auth/</code>	-	+	-
<code>/var/log/syslog/</code>	-	+	+
<code>/var/log/httpd2/</code>	-	-	+
<code>/var/log/ahttpd/</code>	-	-	+

При включенном флаге сбора диагностической информации о памяти (параметр `enable_gc_diagnostic` конфигурационного файла `/opt/aecaCa/scripts/config.sh` архив диагностических данных дополнительно содержит:

- Лог сборщика мусора.
- Дампы памяти для упавших приложений Центра сертификации Aladdin eCA.

¹ Файлы регистрации событий, создаваемые в подкаталогах `/opt/aecaCa/dist/logs/`, имеют права доступа 640 (rw-r-----).

Предварительно выполните переход в директорию, где будет сохранён архив с диагностической информацией в формате `tar.gz`, выполнив команду:

```
cd /`папка размещения архива`
```

Для выполнения сбора диагностической информации запустите скрипт от имени суперпользователя:

```
sudo bash /opt/aecaCa/scripts/diagnostics.sh
```

Сформированный архив в формате `tar.gz` с диагностической информацией будет сохранён в каталог, из которого запускался скрипт.

Для вывода текущей рабочей директории используйте команду: `pwd`

10 РЕЗЕРВНОЕ КОПИРОВАНИЕ И ВОССТАНОВЛЕНИЕ ДАННЫХ ПРОГРАММЫ

Создание резервных копий является неотъемлемой частью работы администратора Центра сертификации Aladdin eCA.

Перед выполнением каких-либо настроек, изменений и обновлений программного компонента следует в обязательном порядке выполнить резервное копирование.

Резервные копии создаются для:

- содержимого каталога, содержащего сертификаты и ключи веб-сервера, разрешённых издателей, путь к которому определён значением параметра `certificates_ssl_path` конфигурационного файла `/opt/aecaCa/scripts/config.sh` (по умолчанию - `/opt/aecaCa/dist/certificates/ssl`);
- закрытого и открытого ключей центра сертификации из каталога, путь к которому определён значением параметра `«cryptotoken_path»` конфигурационного файла `/opt/aecaCa/scripts/config.sh` (по умолчанию - `/opt/aecaCa/dist/cryptotoken`);
- базы данных, имя которой указано в значении параметра `«database_name»` конфигурационного файла `/opt/aecaCa/scripts/config.sh` (по умолчанию - `aecaca`);
- конфигурационного файла `/opt/aecaCa/scripts/config.sh`;

Резервное копирование осуществляется на локальный диск в папку, путь к которой определён значением параметра `«backup_path»` конфигурационного файла `/opt/aecaCa/scripts/config.sh` (по умолчанию - `/opt/aecaCa/dist/backup/`) с указанием даты и времени создания резервной копии в имени архива. Каталог хранения архивов выбран исходя из того, что необходимо хранить резервные копии временно и не увеличивать размер занятого пространства жесткого диска. Для постоянного хранения требуется создать механизм переноса файлов.

Для постоянного хранения резервных копий следует:

- определить каталог для хранения резервных копий;
- составить сценарий для создания резервной копии;
- настроить расписание вызова сценариев.

Создание резервной копии Центра сертификации Aladdin eCA осуществляется запуском скрипта с правами суперпользователя (root):

```
bash /opt/aecaCa/scripts/backup.sh
```

После запуска скрипта резервного копирования создаётся каталог `/opt/aecaCa/dist/backup`, где будет размещён архив, содержащий в имени дату и время создания полной резервной копии.

При успешном создании резервной копии в журнале событий продукта должно регистрироваться событие с кодом «CAENV086». В случае ошибки создания резервной копии в журнале событий продукта должно регистрироваться событие с кодом «CAENV087».

10.1 Расписание резервного копирования

Для снижения потерь данных во время сбоя выполните настройку автоматического резервного копирования, настроив системный планировщик расписания `crontab`.

- Выполните переход в режим редактирования `crontab`, выполнив команду:

```
sudo nano /etc/crontab
```

- Укажите время и период запуска сценариев создания резервных копий:

```
0 0 1 * * /opt/aecaCa/scripts/backup.sh
0 0 1 12 * /opt/aecaCa/scripts/backup.sh
```

где:

- первая строка описывает запуск резервного копирования один раз в месяц,
- вторая строка описывает запуск резервного копирования один раз в год.

Для просмотра настроенного расписания используется команда:

```
crontab -l
```

Внимание! В случаях, когда изменений между резервными копиями обнаружено не было, возможно отображение сообщения о некорректном срабатывании функции stat следующего вида: tar: /tmp/1/inc/copia_*: Функция stat завершилась с ошибкой: No such file or directory

10.2 Восстановление данных из резервной копии

Восстановление данных производится из папки, путь к которой определен значением параметра `backup_path` конфигурационного файла `/opt/aecaCa/scripts/config.sh` (по умолчанию - `/opt/aecaCa/dist/backup/`), на сервере Центра сертификации Aladdin eCA.

Если восстановление происходит на том же сервере, для которого ранее создана резервная копия, и путь к папке не изменен (значение по умолчанию), выполните команду:

```
sudo bash /opt/aecaCa/scripts/restore.sh `путь к папке сохранения резервной копии`/архив резервной копии.tar
```

где `путь к папке сохранения резервной копии` определен значением параметра «`backup_path`» конфигурационного файла `/opt/aecaCa/scripts/config.sh` (по умолчанию - `/opt/aecaCa/dist/backup/`).

Если восстановление происходит после переустановки ОС, выполните:

- подготовку к установке программного компонента в соответствии с разделом 4 настоящего документа;
- установку программного компонента в соответствии с разделом 5 настоящего документа;
- создание каталога хранения резервных копий, путь к которому определен значением параметра `backup_path` конфигурационного файла `/opt/aecaCa/scripts/config.sh` (по умолчанию - `/opt/aecaCa/dist/backup/`), выполнив команду:

```
sudo mkdir -p /opt/aecaCa/dist/backup
```

- копирование в созданный каталог файла резервной копии;
- восстановление данных из резервной копии, выполнив команду:

```
sudo bash /opt/aecaCa/scripts/restore.sh /opt/aecaCa/dist/backup/архив резервной копии.tar
```

При успешном восстановлении из резервной копии в журнале событий продукта должно регистрироваться событие с кодом «CAENV088». В случае ошибки восстановления из резервной копии в журнале событий продукта должно регистрироваться событие с кодом «CAENV089».

11 ВОССТАНОВЛЕНИЕ ДОСТУПА К ПРОГРАММЕ

Восстановление доступа к Центру сертификации Aladdin eCA необходимо выполнить в случае отсутствия ранее созданной резервной копии и блокировки доступа к Центру сертификации Aladdin eCA, возникшей в результате:

- Некорректного удаления технологических составляющих.
- Истечения срока действия сертификата Центра сертификации.
- Истечения срока действия сертификата администратора.
- Удаления Корневого Центра сертификации с автоматическим удалением активного Подчиненного Центра сертификации.

Внимание! После обновления ПО с версий 2.1.2 до версии 2.3.0 перед восстановлением доступа установите параметру `initial_cryptography_key_bits` конфигурационного файла `/opt/aecaCa/scripts/config.sh` (в случае, если параметр был изменен ранее при эксплуатации) значение в соответствии со следующими условиями:

– Если в параметре `initial_cryptography_key_algorithm` выбран алгоритм `RSA`, установите параметру `initial_cryptography_key_bits` значение не менее 4096.

– Если в параметре `initial_cryptography_key_algorithm` выбран алгоритм `ECDSA`, установите параметру `initial_cryptography_key_bits` значение не менее 384.

– Если в параметре `initial_cryptography_key_algorithm` выбран алгоритм `GOST_R_34_10_2012`, установите параметру `initial_cryptography_key_bits` значение 512.

После изменения значения параметра запустите скрипт с правами суперпользователя (`root` или `sudo`) командой `sudo bash /opt/aecaCa/scripts/install.sh` и выберите режим обновления.

Описанные действия можно выполнить и перед обновлением Центра сертификации Aladdin eCA до версии 2.3.0.

Для восстановления доступа к Центру сертификации Aladdin eCA запустите скрипт от имени суперпользователя (`root` или `sudo`):

```
sudo bash /opt/aecaCa/scripts/restore_access.sh
```

В результате выполнения скрипта восстановления доступа к программе:

- Будут созданы и выпущены технологический Центр сертификации «INITIAL_CA» (по умолчанию статус «активирован») и сертификат технологического Центра сертификации «INITIAL_CA».
- Будет заменена учётная запись администратора «INITIAL_ADMIN».
- Будут выпущены и заменены сертификат учётной записи администратора «INITIAL_ADMIN» и сертификат технологического веб-сервера.

Для дальнейшего доступа к Центру сертификации выполните аутентификацию по выпущенному сертификату учётной записи «INITIAL_ADMIN» (см. раздел 7 настоящего документа).

12 ОБНОВЛЕНИЕ ПРОГРАММЫ

Обновление базы данных и модулей Центра сертификации Aladdin eCA обеспечивает актуальность версии программного обеспечения.

При обновлении программы решаются следующие задачи:

- Исправление обнаруженных за время существования программы недочетов и ошибок.
- Устранение выявленных уязвимостей.
- Изменение или улучшение функций программы.
- Добавление новых функций и возможностей.

Компания ведет учет покупателей Центра сертификатов доступа. Выполняется регистрация следующей информации:

- наименование организации;
- адрес организации;
- контактная информация (содержит электронный почтовый адрес лица, обеспечивающего администрирование программы).

Уведомление пользователей о выпуске новой версии Центра сертификатов доступа выполняется путем публикации информации на [официальном сайте Компании](#) и (или) рассылкой электронных почтовых сообщений на электронные адреса потребителей. Рассылка может происходить за счет применения средств, обеспечивающих доведение уведомлений до потребителя автоматически. Вместе с файлами новой версии программного средства может предоставляться обновленная документация для использования программы.

Получение файлов для обновления программного средства и соответствующих им контрольных сумм возможно:

- С использованием электронной почты.
- Путем загрузки с [веб-сайта изготовителя \(производителя\)](#).

Проверка квалифицированной электронной подписи изготовителя (производителя) файлов для обновления программного средства и файлов соответствующих им контрольных сумм выполняется любым доступным способом, если сведения о наличии обновления не предписывают иной порядок проверки подлинности и целостности обновления.

Контроль целостности файлов для обновления программы выполняется путем расчета КС полученных установочных пакетов (дистрибутивов) с использованием предварительно установленного программного обеспечения «ФИКС-Unix 1.0», и её сравнением со значением контрольной суммы для этого обновления (см. раздел 3.2 настоящего документа).

Внимание! На случай, если во время процесса обновления произойдёт сбой, рекомендуем предварительно сделать резервную копию данных программы (см. раздел 10 настоящего руководства).

Схема обновления программного средства представлена на рисунке ниже.

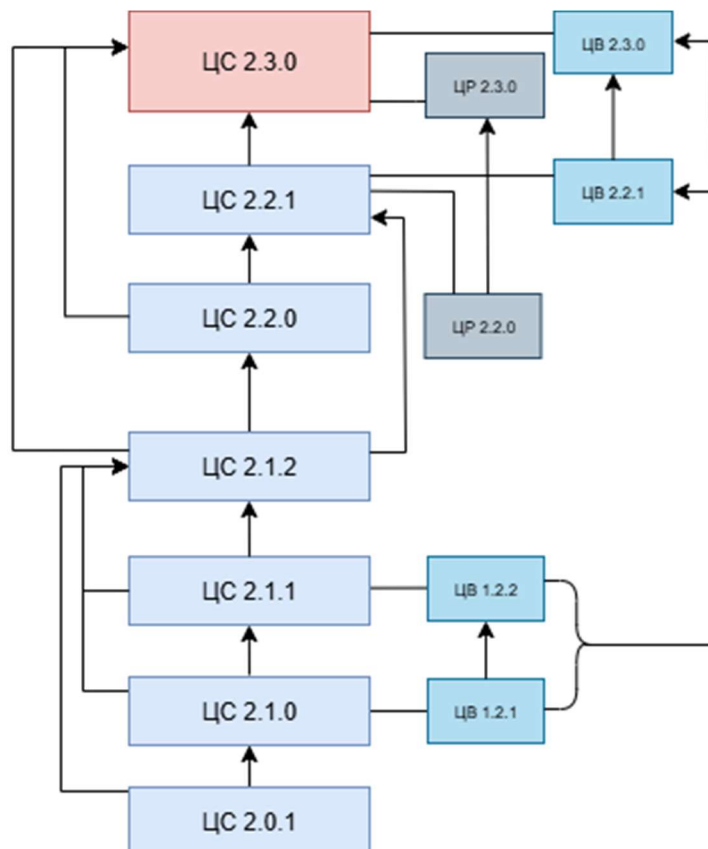


Рисунок 9 – Схема обновления программы

Порядок обновления программы:

- Перенесите дистрибутив с новой версией программы на компьютер с установленным Центром сертификации Aladdin eCA.
- Проверьте целостность дистрибутива путем подсчёта КС (см. подраздел 3.2 настоящего документа);
- Выполните распаковку установочного пакета:
 - для РЕД ОС и SberLinux OS Server командой: `sudo dnf install aeca-*.rpm`;
 - для ОС Astra Linux SE командой: `sudo dpkg -i aeca-*.deb`;
 - для Альт Сервер командой: `sudo apt-get install aeca-*.rpm`.
- Запустите процесс установки продукта в режиме обновления, выполнив команду:

```
sudo bash /opt/aecaCa/scripts/install.sh
```

Установщик обнаружит текущую версию Центра сертификации Aladdin eCA и предложит выбрать необходимое действие в интерактивном режиме:

- Удалить установленную версию со всеми данными и выполнить чистую установку актуальной версии программы.
- Выполнить обновление установленной версии до актуальной версии программы.
- Прервать процесс установки.

Для продолжения процесса обновления введите в терминале цифру «2».

При обновлении программа проверяет соответствие номера сборки и значения номера сборки, указанной в БД¹, имя которой указано в значении параметра `database_name` конфигурационного файла `/opt/aecaCa/scripts/config.sh`:

¹ Значение номера сборки указано в таблице «build_info» схемы «aeca_info».

- Если на момент обновления в БД отсутствует номер сборки, то программа записывает в БД номер устанавливаемой сборки.
- Если на момент обновления в базе данных присутствует номер сборки, и он меньше номера устанавливаемой сборки, то Центр сертификации Aladdin eCA перезаписывает номер сборки в БД, заменив его номером устанавливаемой сборки.
- Если на момент обновления в БД записан номер сборки, и он равен номеру устанавливаемой сборки, программа не изменяет его.
- Если на момент обновления в БД записан номер сборки, и он больше номера устанавливаемой сборки, то программа завершает процесс обновления с ошибкой «Текущая версия схемы базы данных не позволяет выполнить установку или обновление службы. Текущая версия схемы базы данных: X.X.X.X. Необходимая версия схемы базы данных: Y.Y.Y.Y.», где X.X.X.X - номер сборки, записанный в БД, а Y.Y.Y.Y - номер устанавливаемой сборки программы. Номер сборки в БД при этом не меняется.

После обновления программы запустите веб-браузер и очистите его данные.

Запустите обновленный Центр сертификации Aladdin eCA¹, подключитесь к веб-интерфейсу и проверьте версию программы в окне «О программе».

Внимание! После обновления Центра сертификации Aladdin eCA с версий 2.1.2 до версии 2.3.0 пароль пользователя базы данных, заданный в конфигурационном файле `/opt/aecaCa/scripts/config.sh` в параметре `database_password`, отображается в зашифрованном виде. Шифрование пароля выполняется по алгоритму AES-256 с использованием автоматически сгенерированного в файле `/opt/aecaCa/scripts/key` ключа шифрования.

¹ Описание проверок при запуске, выполняемых Центром сертификации Aladdin eCA, см. в разделе 6 настоящего руководства.

13 УДАЛЕНИЕ ПРОГРАММЫ

Для инициализации процесса удаления программы выполните команду с правами суперпользователя (root или sudo):

```
sudo bash /opt/aecaCa/scripts/uninstall.sh
```

В результате выполнения данного действия будут полностью уничтожены:

- Все добавленные при установке программы системные службы.
- Все добавленные при установке программы пользователи и группы.
- Все добавленные при установке программы файлы и структура каталогов.

Процесс удаления выполняется вне зависимости от наличия соединения с БД, имя которой указано в значении параметра `database_name` конфигурационного файла `/opt/aecaCa/scripts/config.sh`.

14 УДАЛЕНИЕ БАЗЫ ДАННЫХ POSTGRES

14.1 Удаление базы данных

Для удаления ранее созданной базы данных «аесаса» (по умолчанию) необходимо выполнить команды с правами суперпользователя (root или sudo):

- Зайдите под пользователем «postgres» в Postgres, выполнив команду:

```
sudo -u postgres psql
```

- Для предотвращения возможности новых подключений выполните команду:

```
UPDATE pg_database SET datallowconn = 'false' WHERE datname = 'aecaca';
```

- Для закрытия всех текущих сессий выполните команду:

```
SELECT pg_terminate_backend(pg_stat_activity.pid)
FROM pg_stat_activity
WHERE pg_stat_activity.datname = 'aecaca' AND pid <> pg_backend_pid();
```

- Удалите базу данных, выполнив команду:

```
DROP DATABASE aecaca;
```

- Завершите работу под пользователем «postgres» и выйдите из терминала, выполнив команду:

```
exit
```

14.2 Удаление пользователя базы данных

Для удаления ранее созданного пользователя базы данных «аеса» (по умолчанию) необходимо выполнить команды с правами суперпользователя (root или sudo):

- Зайдите под пользователем «postgres» в Postgres, выполнив команду:

```
sudo -i -u postgres
```

- Удалите пользователя «аеса» в Postgres, выполнив команду:

```
dropuser aeca -i
```

- Завершите работу под пользователем «postgres» и выйдите из терминала, выполнив команду:

```
exit
```

- Перезапустите СУБД Postgres, выполнив команду:

```
sudo systemctl restart postgresql
```

15 ПОИСК И УСТРАНЕНИЕ НЕИСПРАВНОСТЕЙ

Проблема	Возможная причина	Способы решения
Ошибка при запуске скрипта установки <code>install.sh</code> «error obtaining MAC configuration for user «аеса»»	У пользователя postgres нет прав на чтение БД атрибутов конфиденциальности	Для предоставления дополнительных прав пользователю postgres выполните команды: <pre>sudo usermod -a -G shadow postgres sudo setfacl -d -m u:postgres:r /etc/parsec/macdb sudo setfacl -R -m u:postgres:r /etc/parsec/macdb sudo setfacl -m u:postgres:rx /etc/parsec/macdb</pre>
Ошибка запуска сервисов после запуска скрипта <code>install.sh</code> для установки программы	На сервере была установлена и удалена более ранняя версия программы Не хватка аппаратных ресурсов	Очистите конфигурацию nginx, выполнив команды: <pre>sudo rm -rfv /etc/nginx/general-configs sudo rm -rfv /etc/nginx/conf.d/default.conf</pre> Проверьте показатель загруженности оперативной памяти. Для корректной работы программы требуется не менее 6 Гб свободной оперативной памяти
Ошибка при запуске скрипта установки <code>install.sh</code> «Минимальное количество оперативной памяти для развертывания системы составляет 8192 мегабайта!»	Значение параметра <code>memory</code> в файле конфигурации меньше 8192 или не задано	В файле <code>/opt/aecaCa/scripts/config.sh</code> для параметра <code>memory</code> указать значение 8192: <pre>memory='8192'</pre>
Ошибка при запуске скрипта установки <code>install.sh</code> «psql: FATAL: remaining connection slots are reserved for non-replication superuser connections»	Недостаточное число соединений к БД	В файле <code>postgresql.conf</code> необходимо установить в <code>max_connections</code> значение 1000 и более ¹ .

¹ Параметр `max_connections` задается в инструкциях из разделах про установке и настройке СУБД: 4.2.3, 4.3.3 и 0.

ПРИЛОЖЕНИЕ 1. РАЗРЕШЕНИЕ КОНФЛИКТА ПРИ УСТАНОВКЕ СУБД POSTGRES И POSTGRES PRO

В случае, если другой продукт Postgres¹ установлен, то для разрешения конфликта необходимо выполнить следующие команды:

- Создайте начальную базу данных, запустив вспомогательный скрипт `pg-setup` с правами суперпользователя (`root` или `sudo`) и ключом `initdb`:

```
/opt/pgpro/std-16/bin/pg-setup initdb [--tune=конфигурация] [параметры_initdb]
```

где аргумент `tune` выбирает вариант конфигурации базы данных; параметры `_initdb` – обычные параметры `initdb`.

- Для настройки автозапуска сервера запустите скрипт `pg-setup` со следующими параметрами:

```
/opt/pgpro/std-16/bin/pg-setup service enable
```

- Запустите сервер с помощью скрипта `pg-setup`, выполнив команду с правами суперпользователя (`root` или `sudo`):

```
/opt/pgpro/std-16/bin/pg-setup service start
```

¹ Подробное описание приведено в [официальной документации на PostgreSQL](#).

ПРИЛОЖЕНИЕ 2. НАСТРОЙКА ПОДКЛЮЧЕНИЯ К ВНЕШНЕЙ СУБД

Для подключения Центра сертификации Aladdin eCA к внешней СУБД необходимо:

- выполнить настройку на хосте СУБД в соответствии с разделом 2.1 настоящего приложения.
- выполнить настройку на хосте Aladdin eCA в соответствии с разделом 2.2 настоящего приложения.

2.1 Настройка на хосте СУБД

На внешнем хосте с установленной СУБД) в зависимости от используемой на нем ОС необходимо выполнить следующие настройки:

- Если в качестве ОС на хосте СУБД используется Astra Linux Special Edition 1.7, необходимо разрешить подключение по протоколу TCP для порта СУБД, выполнив в терминале на данном хосте следующую команду:

```
sudo iptables -A INPUT -p tcp --destination-port port -j ACCEPT
```

где `port` - порт для подключения к СУБД (по умолчанию в поддерживаемых СУБД используется порт 5432). Данная команда разрешит подключение к СУБД с любого IP-адреса.

- В случае, если необходимо ограничить доступ к порту СУБД, предоставив его только для определенного IP-адреса, необходимо использовать следующую команду:

```
sudo iptables -A INPUT -s IP -p tcp --destination-port port -j ACCEPT
```

где `IP` - IP-адрес, доступ с которого необходимо разрешить, а `port` - порт для подключения к СУБД (по умолчанию в поддерживаемых СУБД используется порт 5432).

- Если в качестве ОС на хосте с СУБД используется РЕД ОС, SberLinux OS Server и ОС Альт 8 СП, необходимо отредактировать файл `/var/lib/pgsql/15/data/pg_hba.conf` или `var/lib/jatoba/[версия]/data/pg_hba.conf`, если используется СУБД Jatoba)¹, приведя его к следующему виду:

#	TYPE	DATABASE	USER	ADDRESS	METHOD
# "local" is for Unix domain socket connections only					
local	all		all		peer
# IPv4 local connections:					
host	all		all	0.0.0.0/0	password
# IPv6 local connections:					
host	all		all	:::1/128	password
# Allow replication connections from localhost, by a user with the					
# replication privilege.					
local	replication		all		peer
host	replication		all	127.0.0.1/32	ident
host	replication		all	:::1/128	ident

¹ Путь к файлу может отличаться в зависимости от версии PostgreSQL или Jatoba.

Кроме того, необходимо отредактировать файл `/var/lib/pgsql/15/data/postgresql.conf` (или `/var/lib/jatoba/4/data/postgresql.conf`, если используется СУБД Jatoba)¹, указав для параметра `listen_addresses` значение `''`:

```
listen_addresses = ''
```

Значение `''` позволит подключаться к СУБД с любого IP-адреса. В случае, если необходимо ограничить доступ к СУБД, предоставив его только для определенного IP-адреса, необходимо указать данный IP-адрес в параметре `listen_addresses`, например:

```
listen_addresses = '192.168.111.100'
```

- Затем на хосте СУБД необходимо перезапустить используемую СУБД, выполнив команду `sudo systemctl restart postgresql` (или `sudo systemctl restart jatoba-[версия]` если используется СУБД Jatoba).
- Затем на хосте СУБД необходимо создать и настроить базу данных. Действия по созданию и настройке базы данных описаны в разделе 5.3. В результате должна быть создана база данных с выбранными параметрами (имя пользователя, пароль, имя базы данных).

2.2 Настройка на хосте Центра сертификации Aladdin eCA

Внимание! На хосте Центра сертификации Aladdin eCA предварительно должна быть установлена СУБД. При этом не нужно настраивать СУБД, установленную на хосте Центра сертификации Aladdin eCA.

На хосте Центра сертификации Aladdin eCA необходимо отредактировать конфигурационный файл `/opt/aecaCa/scripts/config.sh`, указав в нем значения следующих параметров:

Параметр	Значение по умолчанию	Описание
<code>use_tls</code>	<code>false</code>	Флаг обязательного использования TLS для подключения к СУБД ² . Допустимые значения: <code>true</code> , <code>false</code> .
<code>database_username</code>	<code>'aeca'</code>	Имя пользователя базы данных, используемое для работы Центра сертификации Aladdin eCA. Необходимо внести значение, указанное при создании и настройке базы данных на хосте СУБД.
<code>database_password</code>	<code>#CHANGEIT</code>	Пароль пользователя базы данных, используемый для работы Центра сертификации Aladdin eCA. Необходимо внести значение, указанное при создании и настройке базы данных на хосте СУБД.
<code>database_host</code>	<code>'localhost'</code>	Сетевой адрес хоста СУБД.
<code>database_port</code>	<code>'5432'</code>	Порт, используемый для подключения к базе данных.
<code>database_name</code>	<code>'aecaca'</code>	Имя базы данных, используемой Центром сертификации Aladdin eCA. Необходимо внести значение, указанное при создании и настройке базы данных на хосте СУБД.
<code>root_cert_path</code>	<code>#CHANGEIT</code>	Абсолютный путь к сертификату корневого Центра сертификации из цепочки сертификатов сервера СУБД ³ .

- Затем на хосте Aladdin eCA необходимо применить изменения конфигурационного файла путем запуска команды `sudo bash /opt/aecaCa/scripts/install.sh` и дальнейшего выбора действия «[Update]». Если Aladdin eCA не был установлен ранее, выбор действия не потребует, и будет выполнена установка с указанными в конфигурационном файле параметрами.

¹ Путь к файлу может отличаться в зависимости от версии PostgreSQL или Jatoba.

² Подробная информация о параметре `use_tls` приведена в Приложении 3.

³ Подробная информация о параметре `root_cert_path` приведена в Приложении 3.

ПРИЛОЖЕНИЕ 3. НАСТРОЙКА TLS-СОЕДИНЕНИЯ С СУБД

Внимание! Для настройки TLS-соединения Центра сертификации Aladdin eCA с СУБД необходимо в предварительно развернутом и инициализированном Центре сертификации Aladdin eCA создать сертификат с закрытым ключом (PKCS#12) для сервера СУБД. При этом в сертификате сервера СУБД в атрибуте «Common Name» или в атрибуте «Subject Alternative Name» типа «dNSName» обязательно должно быть указано доменное сервера СУБД (или IP-адрес)¹, так как программный компонент Центр сертификации Aladdin eCA аутентифицирует сервер СУБД в режиме «verify-full», который предполагает проверку соответствия имени узла сервера имени, записанному в сертификате. Для создания сертификата может быть использован шаблон «WEB-Server» (необходимо предварительно создать локальный субъект, указав ему необходимые атрибуты «Common Name» и «DNS Name»).

Во избежание ошибок в работе Центра сертификации Aladdin eCA перед началом настройки TLS-соединения с СУБД рекомендуется остановить работу программы путем выполнения команды `sudo systemctl stop aeca-ca.service`.

Для настройки TLS-соединения Центра сертификации Aladdin eCA с СУБД необходимо:

- Выполнить настройку СУБД в соответствии с разделом 3.1 настоящего приложения;
- Выполнить настройку Центра сертификации Aladdin eCA в соответствии с разделом 3.2 настоящего приложения.

3.1 Настройка СУБД

На хосте с установленной и настроенной СУБД отредактировать файл `/var/lib/pgsql/15/data/postgresql.conf` (или `var/lib/jatoba/4/data/postgresql.conf`, если используется СУБД Jatoba)², указав:

- в параметре «ssl» значение «on»;
- в параметре «ssl_cert_file» абсолютный путь к файлу сертификата сервера СУБД³;
- в параметре «ssl_key_file» абсолютный путь к файлу закрытого ключа сервера СУБД⁴;
- в параметре «ssl_ca_file» абсолютный путь к файлу цепочки сертификатов издателя сертификата СУБД⁵.

При этом указанные выше файлы должны иметь метку доступа «600», установить которую можно с помощью команды `sudo chmod 600 путь_к_файлу` для каждого файла. Владелец всех указанных выше файлов необходимо назначить пользователя «postgres», выполнив команду `sudo chown postgres:postgres путь_к_файлу` для всех перечисленных файлов. Указанные файлы должны располагаться в каталоге, к которому имеет доступ пользователь postgres (например, /tmp). В случае использования ОС РЕД ОС или SberLinux OS Server на хосте СУБД указанные выше файлы должны располагаться в каталоге `/var/lib/pgsql` (или `/var/lib/jatoba`, если используется СУБД Jatoba). При этом указанные выше файлы должны быть скопированы в нужный каталог, а не перемещены.

¹ Указанное в сертификате доменное сервера СУБД (или IP-адрес) должно соответствовать значению параметра «database_host» конфигурационного файла Центра сертификации Aladdin eCA.

² Путь к файлу может отличаться в зависимости от версии PostgreSQL или Jatoba.

³ Файл сертификата сервера СУБД может быть скачан из пользовательского интерфейса Центра сертификации Aladdin eCA. Например, в карточке локального субъекта сервера СУБД.

⁴ Файл закрытого ключа сервера СУБД может быть получен из контейнера закрытого ключа сервера СУБД путем выполнения команды `openssl pkcs12 -in container.p12 -out key.key -nocerts -nodes`, где `container.p12` - путь к контейнеру закрытого ключа сервера СУБД, а «key.key» - путь к файлу для сохранения закрытого ключа.

⁵ Файл цепочки сертификатов издателя сертификата СУБД может быть скачан в карточке Центра сертификации, выпустившего сертификат сервера СУБД.

Пример значений отредактированных параметров конфигурационного файла СУБД `postgresql.conf`:

```
# - SSL -
ssl = on
ssl_cert_file = '/tmp/cert.pem'
ssl_key_file = '/tmp/key.key'
ssl_ca_file = '/tmp/chain.pem'
```

На хосте СУБД перезапустить СУБД, выполнив команду `sudo systemctl restart postgresql` (или `sudo systemctl restart jatoba-[версия]` если используется СУБД Jatoba).

3.2 Настройка Центра сертификации Aladdin eCA

На хосте Центра сертификации Aladdin eCA отредактировать конфигурационный файл `/opt/aecaCa/scripts/config.sh`, указав в нем в параметре конфигурации БД `use_tls` значение `true`, а в параметре `root_cert_path` абсолютный путь к файлу сертификата корневого издателя из цепочки сертификатов сервера СУБД¹.

При этом указанный выше файл сертификата корневого издателя из цепочки сертификатов сервера СУБД должен иметь метку доступа «600», установить которую можно с помощью команды `sudo chmod 600 путь_к_файлу`. Владелец файла сертификата корневого издателя из цепочки сертификатов сервера СУБД необходимо назначить пользователя «аеса», выполнив команду `sudo chown аеса:аеса путь_к_файлу`. Указанный файл должен располагаться в каталоге, к которому имеет доступ пользователь `аеса` (например, `/tmp`). В случае использования ОС РЕД ОС или SberLinux OS Server на хосте Центра сертификации Aladdin eCA файл сертификата корневого издателя из цепочки сертификатов сервера СУБД должен располагаться в каталоге `/opt/aecaCa` (или в его подкаталогах). Кроме того, в случае использования ОС РЕД ОС или SberLinux OS Server на хосте Центра сертификации Aladdin eCA необходимо дополнительно выполнить команду `restorecon -Rv "путь_к_файлу_сертификата_корневого_издателя_из_цепочки_сертификатов_сервера_СУБД"`.

На хосте Центра сертификации Aladdin eCA применить изменения конфигурационного файла путем запуска команды `sudo bash /opt/aecaCa/scripts/install.sh` и дальнейшего выбора действия «[Update]».

По завершению выполнения указанной команды дальнейший обмен данными Центра сертификации Aladdin eCA с СУБД будет осуществляться только по протоколу TLS. Если в СУБД, к которой выполняется подключение, отключен TLS, то Центр сертификации Aladdin eCA не будет выполнять обмен данными с такой СУБД. При этом Центра сертификации Aladdin eCA сможет установить соединение с СУБД только в случае, если ее сертификат издан издателем, путь к сертификату которого указан в конфигурационном файле Центра сертификации Aladdin eCA и только в случае, если имя хоста сервера СУБД соответствует указанному в сертификате.

¹ Если сертификат сервера СУБД выпущен подчиненным Центром сертификации, необходимо указать путь до сертификата корневого Центра сертификации.

ПРИЛОЖЕНИЕ 4. РАЗВЕРТЫВАНИЕ КЛАСТЕРА

Программное средство обеспечивает объединение нескольких Центров сертификации Aladdin eCA в кластер. Кластеризация обеспечивается в отказоустойчивом режиме с использованием внешнего средства балансировки нагрузки HAProxy¹. Отказоустойчивый режим кластеризации обеспечивает как холодное «active-passive»², так и горячее «active-active»³ резервирование. Горячее «active-active» резервирование возможно только при «source»⁴ балансировке.

Развертывания кластера Центра сертификации Aladdin eCA возможно в следующих вариантах:

- В виртуальной инфраструктуре путем клонирования виртуальной машины основного узла.
- С помощью переноса контейнеров закрытого ключа основного узла.

4.1 Развертывание кластера в виртуальной среде с холодным резервированием «active-passive»

Кластер включает следующие узлы:

- Виртуальная машина с установленным и инициализированным Центром сертификации Aladdin eCA (далее - BM1) - основной узел кластера.
- Клон BM1, созданный сразу после завершения инициализации на BM1 Центра сертификации Aladdin eCA (далее - BM2) - резервный узел кластера.
- Виртуальная машина с установленной и настроенной СУБД (далее - BM3).
- Виртуальная машина с установленным и настроенным средством балансировки нагрузки HAProxy (далее - BM4).
- Клон BM1, созданный при необходимости при эксплуатации кластера (далее - BMP) – дополнительный резервный узел кластера.

На всех указанных выше виртуальных машинах допускается использование только ОС, определенных требованиями в разделе 2.1.1 настоящего руководства. Допускается использование одной виртуальной машины для реализации BM3 и BM4.

Порядок развертывания кластера:

- Выполните следующие действия на BM3:
 - Выполнить установку одной из нижеприведённых СУБД:
 - PostgreSQL из состава ОС
 - Jatoba.
 - Увеличьте максимальное количество подключений к СУБД, указав в параметре `max_connections` значение 2000⁵ в файле⁶:
 - `/var/lib/pgsql/15/data/postgresql.conf` для СУБД PostgreSQL.
 - `var/lib/jatoba/[версия]/data/ostgresql.conf` для СУБД Jatoba.

¹ Серверное программное обеспечение для обеспечения высокой доступности и балансировки нагрузки для TCP- и HTTP-приложений посредством распределения входящих запросов на несколько обслуживающих серверов

² Это конфигурация отказоустойчивых кластеров, в которой одни узлы назначаются активными, а другие — резервными, готовыми взять на себя работу в случае отказа активного узла.

³ Это архитектурный подход построения кластера, при котором оба или все узлы активны и работают одновременно, обрабатывая запросы и трафик.

⁴ Это режим, при котором балансировщик выбирает узел кластера на основе хэш-суммы источника IP-адреса, с которого клиенты отправляют запросы. Это гарантирует, что одни и те же пользователи используют один и тот же узел кластера.

⁵ Значение 200 указано из необходимости наличия 1000 подключений для каждого экземпляра Центра сертификации Aladdin eCA, взаимодействующего с СУБД.

⁶ Путь к файлу может отличаться в зависимости от версии PostgreSQL или Jatoba.

- Перезапустите используемую СУБД, выполнив команду с правами суперпользователя:
 - o `sudo systemctl restart postgresql` для СУБД PostgreSQL.
 - o `sudo systemctl restart jatoba-[версия]` для СУБД Jatoba.
- Выполните следующие действия на ВМ1:
 - Выполните установку Центра сертификации Aladdin eCA (см. разделы 4 - 5 настоящего руководства) с подключением внешней СУБД, установленной на ВМ3 (см. Приложение 2 настоящего руководства).
 - Подключитесь к веб-интерфейсу Центра сертификации Aladdin eCA под учетной записью администратора инициализации технологического центра сертификации.
 - Выполните первичное лицензирование Центра сертификации Aladdin eCA (см. раздел 2.2 документа «Центр сертификатов доступа Aladdin Enterprise Certificate Authority Certified Edition KG. Руководство администратора. Часть 2. Функции управления Центра сертификации Aladdin Enterprise Certification Authority»).
 - Выполните инициализацию Центра сертификации (см. раздел 3.1 документа «Центр сертификатов доступа Aladdin Enterprise Certificate Authority Certified Edition KG. Руководство администратора. Часть 2. Функции управления Центра сертификации Aladdin Enterprise Certification Authority»).
- Средствами используемого гипервизора клонируйте ВМ1, тем самым создав ВМ2.
- Запустите ВМ2 и дождитесь завершения запуска службы `aeca-ca.service`.

Внимание! В случае, если на ВМ1 был создан Центр сертификации, у которого криптопровайдером какого-либо алгоритма является СКЗИ «КриптоПро CSP» или программное средство «Криптографический модуль Aladdin JCP», на ВМ2 необходимо выполнить аналогичную ВМ1 установку программного средства «Криптографический модуль Aladdin JCP» и СКЗИ «КриптоПро CSP» и подключение внешней гаммы.

В случае, если на ВМ1 был создан Центр сертификации, местом хранения закрытого ключа которого является ПАКМ «КриптоПро HSM», необходимо выполнить на ВМ2 подключение СКЗИ «КриптоПро CSP» к тому же ПАКМ «КриптоПро HSM».

- Выполните следующие действия на ВМ4:
 - Установите средство балансировки нагрузки HAProxy, выполнив следующую команду с правами суперпользователя:
 - o `sudo dnf install haproxy`- для РЕД ОС и SberLinux OS Server.
 - o `sudo apt install haproxy`- для ОС Astra Linux SE.
 - o `sudo apt-get install haproxy`- для ОС Альт Сервер.
 - Выполните редактирование конфигурационного файла `/etc/haproxy/haproxy.cfg`, приведя его к следующему виду:

```
global
    log /var/log/haproxy/log local0
    log /var/log/haproxy/log local1 notice
    chroot /var/lib/haproxy
    stats socket /run/haproxy/admin.sock mode 660 level admin
    stats timeout 30s
    user haproxy
    group haproxy
    daemon

defaults
    log global
```

```
mode http
option httplog
option dontlognull
timeout connect 5000
timeout client 50000
timeout server 50000

frontend ft_app
bind *:443
mode tcp
default_backend bk_app

backend bk_app
mode tcp
server main IP_VM1:443 check
server clone IP_VM2:443 check backup

listen stats
bind *:8404
stats enable
stats uri /stats
stats auth admin:password
```

где:

- IP_VM1 – IP-адрес ВМ1.
- IP_VM2 – IP-адрес ВМ2.
- admin:password – имя и пароль учетной записи, которые будут использоваться для доступа к панели мониторинга HAProxy.
- Перезапустите HAProxy, выполнив следующую команду с правами суперпользователя `sudo systemctl restart haproxy.service`.

В кластер можно подключать дополнительные резервные узлы. Для подключения нового резервного узла необходимо выполнить действия, аналогичные действиям по подключению узла ВМ2:

- Средствами используемого гипервизора клонируйте ВМ1, тем самым создав ВМР.
- Запустите ВМР и дождитесь запуска службы `aeca-ca.service`.

Внимание! В случае, если на ВМ1 был создан Центр сертификации, у которого криптопровайдером какого-либо алгоритма является СКЗИ «КриптоПро CSP» или программное средство «Криптографический модуль Aladdin JCP», на ВМР необходимо выполнить аналогичную ВМ1 установку программного средства «Криптографический модуль Aladdin JCP» и СКЗИ «КриптоПро CSP» и подключение внешней гаммы.

В случае, если на ВМ1 был создан Центр сертификации, местом хранения закрытого ключа которого является ПАКМ «КриптоПро HSM», необходимо выполнить на ВМР подключение СКЗИ «КриптоПро CSP» к тому же ПАКМ «КриптоПро HSM».

- Выполните на ВМ4 редактирование конфигурационного файла `/etc/haproxy/haproxy.cfg`, добавив в секцию `backend bk_app` информацию об IP-адреса ВМР в соответствии с примером, представленном ниже:

```
backend bk_app
    mode tcp
    server main IP_VM1:443 check
    server clone IP_VM2:443 check backup
    server clone IP_VMR:443 check backup
```

где IP_VMR - это IP-адрес BMP.

- Перезапустить HAProxy на BM4, выполнив следующую команду с правами суперпользователя:

```
sudo systemctl restart haproxy.service.
```

В результате в кластер будет добавлен дополнительный резервный узел.

В результате выполненной настройки кластера все запросы, направляемые к Центру сертификации Aladdin eCA через средство балансировки нагрузки HAProxy, будут перенаправляться на основной узел кластера BM1. При недоступности основного узла кластера все запросы будут перенаправляться на резервный узел кластера BM2. При недоступности BM2 все запросы будут перенаправляться на дополнительный резервный узел кластера BMP. Для мониторинга состояния узлов кластера используйте панель мониторинга HAProxy. Для подключения к панели мониторинга введите в адресной строке веб-браузера http://IP_VM4:8404/stats, где IP_VM4 - IP-адрес BM4. Пройдите идентификацию и аутентификацию с помощью имени и пароля учетной записи администратора, указанных при настройке конфигурационного файла.

Внимание! В случае дальнейшего создания Центров сертификации в развернутом в виртуальной инфраструктуре кластере необходимо сохранять соответствие перечня закрытых ключей, хранимых локально и в хранилище HDIMAGE СКЗИ «КриптоПро CSP» на BM1, BM2 и всех дополнительных резервных узлах. Например, если активным узлом кластера в момент создания Центра сертификации являлся BM2, скопируйте созданные закрытые ключи Центров сертификации с BM2 на BM1, а затем перезапустите `aesa-ca.service` на BM1.

Закрытые ключи Центров сертификации, для которых при создании Центра сертификации было выбрано место хранения «Локально», хранятся в каталоге `/opt/aesaCa/dist/cryptotoken`. При копировании файлов необходимо назначать владельцем данных файлов на конечной ВМ пользователя «aesa».

Закрытые ключи Центров сертификации, для которых при создании Центра сертификации было выбрано место хранения «Жесткий диск (HDIMAGE)», по умолчанию хранятся в каталоге `/var/opt/cproscsp/keys/aesa`. Имена файлов контейнеров закрытых ключей в данном каталоге соответствуют первым 8 символам идентификатора Центра сертификации. При копировании файлов необходимо назначать владельцем данных файлов на конечной ВМ пользователя «aesa», затем перезапускать на данной ВМ СКЗИ «КриптоПро CSP».

Закрытые ключи Центров сертификации, для которых при создании Центра сертификации было выбрано место хранения ПАКМ «КриптоПро HSM», не требуют копирования. Для поддержки работы с такими ключами необходимо сохранять подключение всех узлов кластера к одному ПАКМ «КриптоПро HSM».

4.2 Развертывание кластера с холодным резервированием «active-passive» путем переноса контейнеров закрытого ключа основного узла

Кластер включает следующие узлы:

- Сервер с установленным и инициализированным Центром сертификации Aladdin eCA (далее - APM1) - основной узел кластера.
- Сервер с установленным и инициализированным Центром сертификации Aladdin eCA, для которого будет выполнен перенос контейнеров закрытого ключа (далее - APM2) - резервный узел кластера.
- Сервер с установленным и инициализированным Центром сертификации Aladdin eCA, для которого будет выполнен перенос контейнеров закрытого ключа (далее - APMР) – дополнительный резервный узел кластера.

- Сервер с установленной и настроенной СУБД (далее - АРМ3).
- Сервер с установленным и настроенным средством балансировки нагрузки HAProxy (далее - АРМ4).

На всех указанных выше серверах допускается использование только следующих ОС, определенных требованиями в разделе 2.1.1.

Допускается использование одного сервера для реализации АРМ3 и АРМ4.

Порядок развертывания кластера:

- Выполните следующие действия на АРМ3:
 - Выполнить установку одной из нижеприведенных СУБД:
 - PostgreSQL из состава ОС
 - Jatoba.
 - Увеличьте максимальное количество подключений к СУБД, указав в параметре `max_connections` значение 2000¹ в файле ²:
 - `/var/lib/pgsql/15/data/postgresql.conf` для СУБД PostgreSQL.
 - `var/lib/jatoba/[версия]/data/ostgresql.conf` для СУБД Jatoba.
 - Перезапустите используемую СУБД, выполнив команду с правами суперпользователя:
 - `sudo systemctl restart postgresql` для СУБД PostgreSQL.
 - `sudo systemctl restart jatoba-[версия]` для СУБД Jatoba.
- Выполните следующие действия на АРМ1:
 - Выполните установку Центра сертификации Aladdin eCA (см. разделы 4 - 5 настоящего руководства) с подключением внешней СУБД, установленной на АРМ3 (см. Приложение 2 настоящего руководства).
 - Подключитесь к веб-интерфейсу Центра сертификации Aladdin eCA под учетной записью администратора инициализации технологического центра сертификации.
 - Выполните первичное лицензирование Центра сертификации Aladdin eCA (см. раздел 2.2 документа «Центр сертификатов доступа Aladdin Enterprise Certificate Authority Certified Edition KG. Руководство администратора. Часть 2. Функции управления Центром сертификации Aladdin Enterprise Certification Authority»).
 - Выполните инициализацию Центра сертификации (см. раздел 3.1 документа «Центр сертификатов доступа Aladdin Enterprise Certificate Authority Certified Edition KG. Руководство администратора. Часть 2. Функции управления Центром сертификации Aladdin Enterprise Certification Authority»).

• На АРМ2 выполните установку Центра сертификации Aladdin eCA (см. разделы 4 - 5 настоящего руководства) с подключением внешней СУБД ³, установленной на АРМ3 (см. Приложение 2 настоящего руководства).

Внимание! В случае, если на АРМ1 был создан Центр сертификации, у которого криптопровайдером какого-либо алгоритма является СКЗИ «КриптоПро CSP» или программное средство «Криптографический модуль Aladdin JCP», на АРМ2 необходимо выполнить аналогичную АРМ1 установку программного средства «Криптографический модуль Aladdin JCP» и СКЗИ «КриптоПро CSP» и подключение внешней гаммы.

¹ Значение 200 указано из необходимости наличия 1000 подключений для каждого экземпляра Центра сертификации Aladdin eCA, взаимодействующего с СУБД.

² Путь к файлу может отличаться в зависимости от версии PostgreSQL или Jatoba.

³ В конфигурационном файле Aladdin eCA на АРМ2 необходимо указывать параметры СУБД, аналогичные указанным СУБД АРМ1.

В случае, если на APM1 был создан Центр сертификации, местом хранения закрытого ключа которого является ПАКМ «КриптоПро HSM», необходимо выполнить на APM2 подключение СКЗИ «КриптоПро CSP» к тому же ПАКМ «КриптоПро HSM».

- Если на APM1 был создан Центр сертификации, закрытый ключ которого хранится локально, скопируйте с APM1 содержимое каталога `/opt/aecaCa/dist/cryptotoken` в каталог `/opt/aecaCa/dist/cryptotoken` APM2.

- Если на APM1 был создан Центр сертификации, закрытый ключ которого расположен в хранилище HDIMAGE СКЗИ «КриптоПро CSP», скопируйте с APM1 контейнер Центра сертификации (имя файла будет соответствовать первым 8 символам идентификатора Центра сертификации) из каталога `/var/opt/cproscsp/keys/aeca` в каталог `/var/opt/cproscsp/keys/aeca` APM2. При этом необходимо назначить владельцем данного файла на APM2 пользователя «aeca», и перезапустить на APM2 СКЗИ «КриптоПро CSP».

- Скопируйте с APM1 содержимое каталога `/opt/aecaCa/dist/certificates` в каталог `/opt/aecaCa/dist/certificates` APM2.

- В случае, если на APM2 установлена РЕД ОС или SberLinux OS Server, выполните с правами суперпользователя следующие команды в терминале на APM2:

- `sudo restorecon -Rv /opt/aecaCa/dist/cryptotoken`
- `sudo restorecon -Rv /opt/aecaCa/dist/certificates`

- Выполните на APM2 перезапуск `aeca-ca.service` для обеспечения работы Центра сертификации с перенесенными контейнерами, выполнив с правами суперпользователя следующую команду:

```
sudo systemctl restart aeca-ca.service
```

- Выполните следующие действия на BM4:

- На APM4 выполните установку средства балансировки нагрузки HAProxy, выполнив следующую команду с правами суперпользователя:
 - o `sudo dnf install haproxy`- для РЕД ОС и SberLinux OS Server.
 - o `sudo apt install haproxy`- для ОС Astra Linux SE.
 - o `sudo apt-get install haproxy`- для ОС Альт Сервер.
- На APM4 выполните редактирование конфигурационного файла `/etc/haproxy/haproxy.cfg`, приведя его к следующему виду:

```
global
    log /var/log/haproxy/log local0
    log /var/log/haproxy/log local1 notice
    chroot /var/lib/haproxy
    stats socket /run/haproxy/admin.sock mode 660 level admin
    stats timeout 30s
    user haproxy
    group haproxy
    daemon

defaults
    log global
    mode http
    option httplog
    option dontlognull
    timeout connect 5000
```

```

        timeout client 50000
        timeout server 50000

frontend ft_app
    bind *:443
    mode tcp
    default_backend bk_app

backend bk_app
    mode tcp
    server main IP_ARM1:443 check
    server clone IP_ARM2:443 check backup

listen stats
    bind *:8404
    stats enable
    stats uri /stats
    stats auth admin:password
    
```

где:

- IP_ARM1 – IP-адрес APM1.
- IP_ARM2 – IP-адрес APM2.
- admin:password – имя и пароль учетной записи, которые будут использоваться для доступа к панели мониторинга HAProxy.
- На APM4 перезапустите HAProxy, выполнив следующую команду с правами суперпользователя:
`sudo systemctl restart haproxy.service`

В кластер можно подключать дополнительные резервные узлы. Для подключения нового резервного узла (далее - APMР) необходимо выполнить действия, аналогичные действиям по подключению узла APM2:

- Выполните для APMР действия, производимые для APM2 данного раздела (при их выполнении вместо APM2 использовать APMР).

Внимание! В случае, если на APM1 был создан Центр сертификации, у которого криптопровайдером какого-либо алгоритма является СКЗИ «КриптоПро CSP» или программное средство «Криптографический модуль Aladdin JCP», на APMР необходимо выполнить аналогичную APM1 установку программного средства «Криптографический модуль Aladdin JCP» и СКЗИ «КриптоПро CSP» и подключение внешней гаммы.

В случае, если на APM1 был создан Центр сертификации, местом хранения закрытого ключа которого является ПАКМ «КриптоПро HSM», необходимо выполнить на APMР подключение СКЗИ «КриптоПро CSP» к тому же ПАКМ «КриптоПро HSM».

- Выполните на APM4 редактирование конфигурационного файла `/etc/haproxy/haproxy.cfg`, добавив в секцию `backend bk_app` информацию об IP-адреса APMР в соответствии с примером, представленном ниже:

```

backend bk_app
    mode tcp
    server main IP_ARM1:443 check
    server clone IP_ARM2:443 check backup
    server clone IP_ARMR:443 check backup
    
```

где IP_ARMR - это IP-адрес APMР.

- На APM4 перезапустите HAProxy, выполнив следующую команду с правами суперпользователя:
`sudo systemctl restart haproxy.service`

В результате в кластере появится дополнительный резервный узел. Все описанные выше рекомендации и уточнения по работе с узлом APM2, также относятся и к узлу APMР.

В результате приведенной настройки кластера все запросы, направляемые к Центру сертификации Aladdin eCA через средство балансирования нагрузки HAProxy, будут перенаправляться на основной узел кластера APM1. В случае недоступности основного узла кластера все запросы, направляемые к Центру сертификации Aladdin eCA через средство балансирования нагрузки HAProxy, будут перенаправляться на резервный узел кластера APM2. Для мониторинга состояния узлов кластера может быть использована панель мониторинга, доступная по адресу `http://IP_ARM4:8404/stats`, где IP_ARM4 - IP-адрес APM4 (для входа в панель мониторинга потребуется ввод логина и пароля, указанных в файле `/etc/haproxy/haproxy.cfg` на APM4).

Внимание! В случае дальнейшего создания Центров сертификации в развернутом кластере необходимо сохранять соответствие перечня закрытых ключей, хранимых локально и в хранилище КриптоПро HDIMAGE на APM1, APM2 и дополнительных узлах. Например, если активным узлом кластера в момент создания Центра сертификации являлся APM2, необходимо скопировать созданные закрытые ключи Центров сертификации с APM2 на APM1, затем перезапустить службу `aeca-ca.service` на APM1.

Закрытые ключи центров сертификации, для которых при создании Центра сертификации было выбрано место хранения «Локально», хранятся в каталоге `/opt/aecaCa/dist/cryptotoken`. При копировании файлов необходимо назначать владельцем данных файлов на конечном APM пользователя «aeca».

Закрытые ключи центров сертификации, для которых при создании Центра сертификации было выбрано место хранения «Жесткий диск (HDIMAGE)», по умолчанию хранятся в каталоге `/var/opt/cproscsp/keys/aeca`. Имена файлов контейнеров закрытых ключей в данном каталоге соответствуют первым 8 символам идентификатора Центра сертификации. При копировании файлов необходимо назначать владельцем данных файлов на конечном APM пользователя «aeca», затем перезапускать на данном APM СКЗИ «КриптоПро CSP».

Закрытые ключи центров сертификации, для которых при создании Центра сертификации было выбрано место хранения ПАКМ «КриптоПро HSM», не требуют копирования. Для поддержки работы с такими ключами необходимо сохранять подключение всех узлов кластера к одному ПАКМ «КриптоПро HSM».

4.3 Развертывания кластера в виртуальной среде с горячим резервированием «active-active»

Кластер включает следующие узлы:

- Виртуальная машина с установленным и инициализированным Центром сертификации Aladdin eCA (далее - BM1) - первый узел кластера.
- Клон BM1, созданный сразу после завершения инициализации на BM1 Центра сертификации Aladdin eCA (далее - BM2) - второй узел кластера.
- Виртуальная машина с установленной и настроенной СУБД (далее - BM3).
- Виртуальная машина с установленным и настроенным средством балансировки нагрузки HAProxy (далее - BM4).
- Клон BM1, созданный при необходимости при эксплуатации кластера (далее - BMP) - дополнительный узел кластера.

На всех указанных выше виртуальных машинах допускается использование только ОС, определенных требованиями в разделе 2.1.1 настоящего руководства.

Допускается использование одной виртуальной машины для реализации BM3 и BM4.

Порядок развертывания кластера:

- Выполните следующие действия на ВМ3:
 - Выполнить установку одной из нижеприведённых СУБД:
 - PostgreSQL из состава ОС
 - Jatoba.
 - Увеличьте максимальное количество подключений к СУБД, указав в параметре `max_connections` значение 2000¹ в файле ²:
 - `/var/lib/pgsql/15/data/postgresql.conf` для СУБД PostgreSQL.
 - `var/lib/jatoba/[версия]/data/ostgresql.conf` для СУБД Jatoba.
 - Перезапустите используемую СУБД, выполнив команду с правами суперпользователя:
 - `sudo systemctl restart postgresql` для СУБД PostgreSQL.
 - `sudo systemctl restart jatoba-[версия]` для СУБД Jatoba.
- Выполните следующие действия на ВМ1:
 - Выполните установку Центра сертификации Aladdin eCA (см. разделы 4 - 5 настоящего руководства) с подключением внешней СУБД, установленной на ВМ3 (см. Приложение 2 настоящего руководства).
 - Подключитесь к веб-интерфейсу Центра сертификации Aladdin eCA под учетной записью администратора инициализации технологического центра сертификации.
 - Выполните первичное лицензирование Центра сертификации Aladdin eCA (см. раздел 2.2 документа «Центр сертификатов доступа Aladdin Enterprise Certificate Authority Certified Edition KG. Руководство администратора. Часть 2. Функции управления Центром сертификации Aladdin Enterprise Certification Authority»).
 - Выполните инициализацию Центра сертификации (см. раздел 3.1 документа «Центр сертификатов доступа Aladdin Enterprise Certificate Authority Certified Edition KG. Руководство администратора. Часть 2. Функции управления Центром сертификации Aladdin Enterprise Certification Authority»).
- Средствами используемого гипервизора клонируйте ВМ1, тем самым создав ВМ2.
- Запустите ВМ2 и дождитесь завершения запуска службы `aeca-ca.service`.

Внимание! В случае, если на ВМ1 был создан Центр сертификации, у которого криптопровайдером какого-либо алгоритма является СКЗИ «КриптоПро CSP» или программное средство «Криптографический модуль Aladdin JCP», на ВМ2 необходимо выполнить аналогичную ВМ1 установку программного средства «Криптографический модуль Aladdin JCP» и СКЗИ «КриптоПро CSP» и подключение внешней гаммы.

В случае, если на ВМ1 был создан Центр сертификации, местом хранения закрытого ключа которого является ПАКМ «КриптоПро HSM», необходимо выполнить на ВМ2 подключение СКЗИ «КриптоПро CSP» к тому же ПАКМ «КриптоПро HSM».

- Выполните следующие действия на ВМ4:
 - Установите средство балансировки нагрузки HAProxy, выполнив следующую команду с правами суперпользователя:
 - `sudo dnf install haproxy`- для РЕД ОС и SberLinux OS Server.
 - `sudo apt install haproxy`- для ОС Astra Linux SE.
 - `sudo apt-get install haproxy`- для ОС Альт Сервер.

¹ Значение 200 указано из необходимости наличия 1000 подключений для каждого экземпляра Центра сертификации Aladdin eCA, взаимодействующего с СУБД.

² Путь к файлу может отличаться в зависимости от версии PostgreSQL или Jatoba.

- Выполните редактирование конфигурационного файла `/etc/haproxy/haproxy.cfg`, приведя его к следующему виду:

```
global
    log /var/log/haproxy/log local0
    log /var/log/haproxy/log local1 notice
    chroot /var/lib/haproxy
    stats socket /run/haproxy/admin.sock mode 660 level admin
    stats timeout 30s
    user haproxy
    group haproxy
    daemon

defaults
    log global
    mode http
    option httplog
    option dontlognull
    timeout connect 5000
    timeout client 50000
    timeout server 50000

frontend ft_app
    bind *:443
    mode tcp
    default_backend bk_app

backend bk_app
    mode tcp
    balance source
    hash-type consistent
    server main IP_VM1:443 check
    server clone IP_VM2:443 check

listen stats
    bind *:8404
    stats enable
    stats uri /stats
    stats auth admin:password
```

- o `IP_VM1` – IP-адрес VM1.
 - o `IP_VM2` – IP-адрес VM2.
 - o `admin:password` – имя и пароль учетной записи, которые будут использоваться для доступа к панели мониторинга HAProxy.
- Перезапустите HAProxy, выполнив следующую команду с правами суперпользователя: `sudo systemctl restart haproxy.service`.

В кластер можно подключать дополнительные узлы. Для подключения нового резервного узла необходимо выполнить действия, аналогичные действиям по подключению узла VM2:

- Средствами используемого гипервизора клонируйте VM1, тем самым создав BMP.
- Запустите BMP и дождитесь запуска службы `aeca-ca.service`.

Внимание! В случае, если на VM1 был создан Центр сертификации, у которого криптопровайдером какого-либо алгоритма является СКЗИ «КриптоПро CSP» или программное средство «Криптографический модуль Aladdin JCP», на BMP необходимо выполнить аналогичную VM1 установку программного средства «Криптографический модуль Aladdin JCP» и СКЗИ «КриптоПро CSP» и подключение внешней гаммы.

В случае, если на VM1 был создан Центр сертификации, местом хранения закрытого ключа которого является ПАКМ «КриптоПро HSM», необходимо выполнить на BMP подключение СКЗИ «КриптоПро CSP» к тому же ПАКМ «КриптоПро HSM».

- Выполните на VM4 редактирование конфигурационного файла `/etc/haproxy/haproxy.cfg`, добавив в секцию `backend bk_app` информацию об IP-адреса BMP в соответствии с примером, представленном ниже:

```
backend bk_app
    mode tcp
    balance source
    hash-type consistent
    server main IP_VM1:443 check
    server clone IP_VM2:443 check
    server clone IP_VMR:443 check
```

где `IP_VMR` - это IP-адрес BMP.

- Перезапустить HAProxy на VM4, выполнив следующую команду с правами суперпользователя:
`sudo systemctl restart haproxy.service`.

В результате в кластер будет добавлен дополнительный резервный узел.

В результате выполненной настройки средство балансировки нагрузки HAProxy будет выбирать узел кластера на основе хэш-суммы источника IP-адреса и перенаправлять на него запросы. Это будет гарантирует, что одни и те же пользователи используют один и тот же узел кластера. Для мониторинга состояния узлов кластера используйте панель мониторинга HAProxy. Для подключения к панели мониторинга введите в адресной строке веб-браузера `http://IP_VM4:8404/stats`, где `IP_VM4` - IP-адрес VM4. Пройдите идентификацию и аутентификацию с помощью имени и пароля учетной записи администратора, указанных при настройка конфигурационного файла.

Внимание! В случае дальнейшего создания Центров сертификации в развернутом в виртуальной инфраструктуре кластере необходимо сохранять соответствие перечня закрытых ключей, хранимых локально и в хранилище HDIMAGE СКЗИ «КриптоПро CSP» на VM1, VM2 и всех дополнительных узлах. Например, если активным узлом кластера в момент создания Центра сертификации являлся VM2, скопируйте созданные закрытые ключи Центров сертификации с VM2 на VM1, а затем перезапустите `aeca-ca.service` на VM1.

Закрытые ключи Центров сертификации, для которых при создании Центра сертификации было выбрано место хранения «Локально», хранятся в каталоге `/opt/aecaCa/dist/cryptotoken`. При копировании файлов необходимо назначать владельцем данных файлов на конечной VM пользователя «aeca».

Закрытые ключи Центров сертификации, для которых при создании Центра сертификации было выбрано место хранения «Жесткий диск (HDIMAGE)», по умолчанию хранятся в каталоге `/var/opt/cproscsp/keys/aeca`. Имена файлов контейнеров закрытых ключей в данном каталоге соответствуют первым 8 символам идентификатора Центра сертификации. При копировании файлов необходимо назначать владельцем данных файлов на конечной VM пользователя «aeca», затем перезапускать на данной VM СКЗИ «КриптоПро CSP».

Закрытые ключи Центров сертификации, для которых при создании Центра сертификации было выбрано место хранения ПАКМ «КриптоПро HSM», не требуют копирования. Для поддержки работы с такими ключами необходимо сохранять подключение всех узлов кластера к одному ПАКМ «КриптоПро HSM».

4.4 Развертывание кластера с горячим резервированием «active-active» путем переноса контейнера закрытого ключа первого узла

Кластер включает следующие узлы:

- Сервер с установленным и инициализированным Центром сертификации Aladdin eCA (далее - APM1) – первый узел кластера.
- Сервер с установленным и инициализированным Центром сертификации Aladdin eCA, для которого будет выполнен перенос контейнеров закрытого ключа (далее - APM2) – второй узел кластера.
- Сервер с установленным и инициализированным Центром сертификации Aladdin eCA, для которого будет выполнен перенос контейнеров закрытого ключа (далее - APMР) – дополнительный узел кластера.
- Сервер с установленной и настроенной СУБД (далее - APM3).
- Сервер с установленным и настроенным средством балансировки нагрузки HAProxy (далее - APM4).

На всех указанных выше серверах допускается использование только следующих ОС, определенных требованиями в разделе 2.1.1.

Допускается использование одного сервера для реализации APM3 и APM4.

Порядок развертывания кластера:

- Выполните следующие действия на APM3:
 - Выполнить установку одной из нижеприведённых СУБД:
 - PostgreSQL из состава ОС
 - Jatoba.
 - Увеличьте максимальное количество подключений к СУБД, указав в параметре `max_connections` значение 2000¹ в файле²:
 - `/var/lib/pgsql/15/data/postgresql.conf` для СУБД PostgreSQL.
 - `var/lib/jatoba/[версия]/data/ostgresql.conf` для СУБД Jatoba.
 - Перезапустите используемую СУБД, выполнив команду с правами суперпользователя:
 - `sudo systemctl restart postgresql` для СУБД PostgreSQL.
 - `sudo systemctl restart jatoba-[версия]` для СУБД Jatoba.
- Выполните следующие действия на APM1:
 - Выполните установку Центра сертификации Aladdin eCA (см. разделы 4 - 5 настоящего руководства) с подключением внешней СУБД, установленной на APM3 (см. Приложение 2 настоящего руководства).
 - Подключитесь к веб-интерфейсу Центра сертификации Aladdin eCA под учетной записью администратора инициализации технологического центра сертификации.
 - Выполните первичное лицензирование Центра сертификации Aladdin eCA (см. раздел 2.2 документа «Центр сертификатов доступа Aladdin Enterprise Certificate Authority Certified Edition KG. Руководство администратора. Часть 2. Функции управления Центром сертификации Aladdin Enterprise Certification Authority»).

¹ Значение 200 указано из необходимости наличия 1000 подключений для каждого экземпляра Центра сертификации Aladdin eCA, взаимодействующего с СУБД.

² Путь к файлу может отличаться в зависимости от версии PostgreSQL или Jatoba.

- Выполните инициализацию Центра сертификации (см. раздел 3.1 документа «Центр сертификатов доступа Aladdin Enterprise Certificate Authority Certified Edition KG. Руководство администратора. Часть 2. Функции управления Центра сертификации Aladdin Enterprise Certification Authority»).

- На АРМ2 выполните установку Центра сертификации Aladdin eCA (см. разделы 4 - 5 настоящего руководства) с подключением внешней СУБД¹, установленной на АРМ3 (см. Приложение 2 настоящего руководства).

Внимание! В случае, если на АРМ1 был создан Центр сертификации, у которого криптопровайдером какого-либо алгоритма является СКЗИ «КриптоПро CSP» или программное средство «Криптографический модуль Aladdin JCP», на АРМ2 необходимо выполнить аналогичную АРМ1 установку программного средства «Криптографический модуль Aladdin JCP» и СКЗИ «КриптоПро CSP» и подключение внешней гаммы.

В случае, если на АРМ1 был создан Центр сертификации, местом хранения закрытого ключа которого является ПАКМ «КриптоПро HSM», необходимо выполнить на АРМ2 подключение СКЗИ «КриптоПро CSP» к тому же ПАКМ «КриптоПро HSM».

- Если на АРМ1 был создан Центр сертификации, закрытый ключ которого хранится локально, скопируйте с АРМ1 содержимое каталога `/opt/aecaCa/dist/cryptotoken` в каталог `/opt/aecaCa/dist/cryptotoken` АРМ2.

- Если на АРМ1 был создан Центр сертификации, закрытый ключ которого расположен в хранилище HDIMAGE СКЗИ «КриптоПро CSP», скопируйте с АРМ1 контейнер Центра сертификации (имя файла будет соответствовать первым 8 символам идентификатора Центра сертификации) из каталога `/var/opt/cproscsp/keys/aeca` в каталог `/var/opt/cproscsp/keys/aeca` АРМ2. При этом необходимо назначить владельцем данного файла на АРМ2 пользователя «aeca», и перезапустить на АРМ2 СКЗИ «КриптоПро CSP».

- Скопируйте с АРМ1 содержимое каталога `/opt/aecaCa/dist/certificates` в каталог `/opt/aecaCa/dist/certificates` АРМ2.

- В случае, если на АРМ2 установлена РЕД ОС или SberLinux OS Server, выполните с правами суперпользователя следующие команды в терминале на АРМ2:

- `sudo restorecon -Rv /opt/aecaCa/dist/cryptotoken`
- `sudo restorecon -Rv /opt/aecaCa/dist/certificates`

- Выполните на АРМ2 перезапуск `aeca-ca.service` для обеспечения работы Центра сертификации с перенесенными контейнерами, выполнив с правами суперпользователя следующую команду:

```
sudo systemctl restart aeca-ca.service
```

- На АРМ4 выполните установку средства балансировки нагрузки HAProxy, выполнив следующую команду с правами суперпользователя:

- `sudo dnf install haproxy`- для РЕД ОС и SberLinux OS Server.
- `sudo apt install haproxy`- для ОС Astra Linux SE.
- `sudo apt-get install haproxy`- для ОС Альт Сервер.

- На АРМ4 выполните редактирование конфигурационного файла `/etc/haproxy/haproxy.cfg`, приведя его к следующему виду:

```
global
    log /var/log/haproxy/log local0
    log /var/log/haproxy/log local1 notice
```

¹ В конфигурационном файле Aladdin eCA на АРМ2 необходимо указывать параметры СУБД, аналогичные указанным СУБД АРМ1.

```

chroot /var/lib/haproxy
stats socket /run/haproxy/admin.sock mode 660 level admin
stats timeout 30s
user haproxy
group haproxy
daemon

defaults
    log global
    mode http
    option httplog
    option dontlognull
    timeout connect 5000
    timeout client 50000
    timeout server 50000

frontend ft_app
    bind *:443
    mode tcp
    default_backend bk_app

backend bk_app
    mode tcp
    balance source
    hash-type consistent
    server main IP_VM1:443 check
    server clone IP_VM2:443 check

listen stats
    bind *:8404
    stats enable
    stats uri /stats
    stats auth admin:password
    
```

где:

- IP_ARM1 – IP-адрес APM1.
- IP_ARM2 – IP-адрес APM2.
- admin:password – имя и пароль учетной записи, которые будут использоваться для доступа к панели мониторинга HAProxy.

- На APM4 перезапустите HAProxy, выполнив следующую команду с правами суперпользователя:

```
sudo systemctl restart haproxy.service
```

В кластер можно подключать дополнительные резервные узлы. Для подключения нового резервного узла (далее - APMР) необходимо выполнить действия, аналогичные действиям по подключения узла APM2:

- Выполните для APMР действия, производимые для APM2 данного раздела (при их выполнении вместо APM2 использовать APMР).

Внимание! В случае, если на APM1 был создан Центр сертификации, у которого криптопровайдером какого-либо алгоритма является СКЗИ «КриптоПро CSP» или программное средство «Криптографический модуль Aladdin JCP», на APM» необходимо выполнить аналогичную APM1 установку программного средства «Криптографический модуль Aladdin JCP» и СКЗИ «КриптоПро CSP» и подключение внешней гаммы.

В случае, если на APM1 был создан Центр сертификации, местом хранения закрытого ключа которого является ПАКМ «КриптоПро HSM», необходимо выполнить на APM2 подключение СКЗИ «КриптоПро CSP» к тому же ПАКМ «КриптоПро HSM».

- Выполните на APM4 редактирование конфигурационного файла `/etc/haproxy/haproxy.cfg`, добавив в секцию `backend bk_app` информацию об IP-адреса APMР в соответствии с примером, представленном ниже:

```
backend bk_app
    mode tcp
    server main IP_ARM1:443 check
    server clone IP_ARM2:443 check backup
    server clone IP_ARMR:443 check backup
```

где `IP_ARMR` - это IP-адрес APMР.

- На APM4 перезапустите HAProxy, выполнив следующую команду с правами суперпользователя:
`sudo systemctl restart haproxy.service`

В результате в кластере появится дополнительный резервный узел. Все описанные выше рекомендации и уточнения по работе с узлом APM2, также относятся и к узлу APMР.

В результате выполненной настройки средство балансировки нагрузки HAProxy будет выбирать узел кластера на основе хэш-суммы источника IP-адреса и перенаправлять на него запросы. Это будет гарантирует, что одни и те же пользователи используют один и тот же узел кластера. Для мониторинга состояния узлов кластера может быть использована панель мониторинга, доступная по адресу `http://IP_ARM4:8404/stats`, где `IP_ARM4` - IP-адрес APM4 (для входа в панель мониторинга потребуется ввод логина и пароля, указанных в файле `/etc/haproxy/haproxy.cfg` на APM4).

Внимание! В случае дальнейшего создания Центров сертификации в развернутом кластере необходимо сохранять соответствие перечня закрытых ключей, хранимых локально и в хранилище КриптоПро HDIMAGE на APM1, APM2 и дополнительные узлы. Например, если активным узлом кластера в момент создания Центра сертификации являлся APM2, необходимо скопировать созданные закрытые ключи Центров сертификации с APM2 на APM1, затем перезапустить `aesa-ca.service` на APM1.

Закрытые ключи центров сертификации, для которых при создании Центра сертификации было выбрано место хранения «Локально», хранятся в каталоге `/opt/aesaCa/dist/cryptotoken`. При копировании файлов необходимо назначать владельцем данных файлов на конечном APM пользователя «aesa».

Закрытые ключи центров сертификации, для которых при создании Центра сертификации было выбрано место хранения «Жесткий диск (HDIMAGE)», по умолчанию хранятся в каталоге `/var/opt/cproscsp/keys/aesa`. Имена файлов контейнеров закрытых ключей в данном каталоге соответствуют первым 8 символам идентификатора Центра сертификации. При копировании файлов необходимо назначать владельцем данных файлов на конечном APM пользователя «aesa», затем перезапускать на данном APM СКЗИ «КриптоПро CSP».

Закрытые ключи центров сертификации, для которых при создании Центра сертификации было выбрано место хранения ПАКМ «КриптоПро HSM», не требуют копирования. Для поддержки работы с такими ключами необходимо сохранять подключение всех узлов кластера к одному ПАКМ «КриптоПро HSM».

4.3 Обновление ПО узлов кластера Aladdin eCA

Процесс обновления кластера Центра сертификации Aladdin eCA:

- Выполните резервное копирование данных на всех узлах кластера (см. раздел 10 настоящего руководства).
- Для кластера по схеме «active-passive» на всех резервных узлах выполните остановку службы Центра сертификации Aladdin eCA, выполнив следующую команду с правами суперпользователя: `sudo systemctl stop aeca-ca.service`.
- Для кластера по схеме «active-active» на всех узлах, на которые был перенесен закрытый ключ, выполните остановку службы Центра сертификации Aladdin eCA, выполнив следующую команду с правами суперпользователя: `sudo systemctl stop aeca-ca.service`.
- Для кластера по схеме «active-passive» выполнить обновление ПО Центра сертификации Aladdin eCA на основном узле (см. раздел 12 настоящего руководства).
- Для кластера по схеме «active-active» выполнить обновление ПО Центра сертификации Aladdin eCA на узле, на котором был создан закрытый ключ (см. раздел 12 настоящего руководства).
- Вне зависимости от схемы кластера выполните обновление ПО Центра сертификации Aladdin eCA на всех остальных узлах кластера (см. раздел 12 настоящего руководства).

Критерием правильности установки обновления ПО кластера является отображение информации о новой версии в окне «О программе» веб-интерфейса и работоспособность всех узлов кластера. Работоспособность узлов можно посмотреть в панели мониторинга HAProxy, доступной по адресу `http://IP_ARM4:8404/stats`, где `IP_ARM4` - IP-адрес APM4 (для входа в панель мониторинга потребуется ввод логина и пароля, указанных в файле `/etc/haproxy/haproxy.cfg`).

ПРИЛОЖЕНИЕ 5. НАСТРОЙКА ВЗАИМОДЕЙСТВИЯ С КРИПТОПРОВАЙДЕРОМ СКЗИ «КРИПТОПРО CSP»

Для использования алгоритмов ГОСТ Р 34.10-2012 и RSA Центр сертификации Aladdin eCA может взаимодействовать со средством криптографической защиты информации (СКЗИ) - криптопровайдером СКЗИ «КриптоПро CSP» из состава программного средства с целью реализации следующих возможностей:

- создание ключевой пары (открытый и закрытый ключи) Центра сертификации (корневого или подчиненного);
- подписание сертификата Центра сертификации (самоподписанный сертификат);
- создание контейнеров закрытого ключа Центра сертификации с возможностью указания места хранения;
- подписание запроса на сертификат Центра сертификации в вышестоящем центре сертификации;
- создание ключевой пары (открытый и закрытый ключи) для субъектов (пользователей или технических средств);
- подписание сертификатов доступа для субъектов (пользователей или технических средств - владельцев сертификатов доступа);
- создание контейнеров закрытого ключа субъектов (пользователей или технических средств);
- подписание списка отозванных сертификатов.

Взаимодействие Центра сертификации с криптопровайдером СКЗИ «КриптоПро CSP» из состава программного средства осуществляется через модуль «КриптоПро Java CSP»¹. При каждом запуске Центр сертификации автоматически определяется наличие на его хосте активного криптопровайдера СКЗИ «КриптоПро CSP».

Также Центр сертификации может интегрироваться с программно-аппаратным криптографическим модулем (ПАКМ) «КриптоПро HSM»² для обеспечения возможности генерации и хранения в последнем закрытых ключей Центров Сертификации. Взаимодействие программного средства с ПАКМ «КриптоПро HSM» осуществляется посредством криптопровайдера СКЗИ «КриптоПро CSP». При каждом запуске Центр сертификации автоматически определяется наличие подключения криптопровайдера СКЗИ «КриптоПро CSP» к ПАКМ «КриптоПро HSM», при наличии подключения будет доступен выбор ПАКМ «КриптоПро HSM» в качестве места хранения закрытых ключей создаваемых Центров Сертификации.

Установка и настройка ПАКМ «КриптоПро HSM» выполняется в соответствии с документом «ПАКМ «КриптоПро HSM». Инструкция по использованию» ЖТЯИ.00096-01 90 01.

Настройка СКЗИ «КриптоПро CSP» в качестве клиентского приложения ПАКМ «КриптоПро HSM» выполняется в соответствии с документом «СКЗИ «КриптоПро CSP». Руководство администратора безопасности. Использование СКЗИ под управлением ОС Linux» ЖТЯИ.00101-03 91 03.

¹ Модуль «КриптоПро Java CSP» входит в состав СКЗИ «КриптоПро CSP».

² Сетевое устройство, подключаемое либо непосредственно к серверу (хосту), использующему криптографические сервисы ПАКМ «КриптоПро HSM», либо в сегмент локальной сети через стандартные сетевые устройства (коммутаторы, маршрутизаторы, концентраторы) для обслуживания групп серверов и компьютеров пользователей сети.

5.1 Настройка взаимодействия с СКЗИ «КриптоПро CSP»

До выполнения настройки взаимодействия СКЗИ «КриптоПро CSP» с Центром сертификации Aladdin eCA необходимо подготовить внешнюю гамму¹. Подключение внешней гаммы необходимо для генерации ключевых пар центров сертификации, субъектов и пользователей по алгоритмам, криптопровайдером которых является СКЗИ «КриптоПро CSP». При этом, внешняя гамма не используется для генерации ключевой пары центра сертификации, если при его создании в качестве места хранения закрытого ключа выбран ПАКМ «КриптоПро HSM»².

Внимание! При развертывании нескольких экземпляров Центра сертификации Aladdin eCA под одним средством балансирования нагрузки необходимо для каждого экземпляра программного средства подготовить уникальную внешнюю гамму, чтобы исключить совпадения ключевых пар.

Порядок настройки взаимодействия СКЗИ «КриптоПро CSP» с Центром сертификации Aladdin eCA:

- На сервере Центра сертификации Aladdin eCA выполнить установку криптопровайдера СКЗИ «КриптоПро CSP» в соответствии с инструкцией, описанной в разделе 2 документа «СКЗИ «КриптоПро CSP». Руководство администратора безопасности. Использование СКЗИ под управлением ОС Linux» ЖТЯИ.00101-03 91 03.

Внимание! Перед установкой СКЗИ «КриптоПро CSP» в ОС Альт 8 СП Сервер установите пакет `newt52` командой `sudo apt-get install newt52`.

- При отсутствии создайте каталог `/opt/aecaCa/services/cryptoproviders` командой:

```
sudo mkdir -p /opt/aecaCa/services/cryptoproviders
```

- Переместите в каталог `/opt/aecaCa/services/cryptoproviders` файлы `ASN1P.jar`, `asn1rt.jar`, `JCP.jar`, `JCSP.jar`, `cpSSL.jar` и `sspiSSL.jar` из состава дистрибутива ПО «КриптоПро Java CSP» и «КриптоПро Java TLS» командой:

```
sudo cp {ASN1P.jar,asn1rt.jar,JCP.jar,JCSP.jar,cpSSL.jar,sspiSSL.jar}
/opt/aecaCa/services/cryptoproviders
```

- Назначьте права доступа на скопированные файлы:
 - Если выполняется первоначальная установка Центра сертификации Aladdin eCA, то назначьте файлам права доступа (`chmod 777`) командой:

```
sudo chmod 777
/opt/aecaCa/services/cryptoproviders/{ASN1P.jar,asn1rt.jar,JCP.jar,JCSP.jar,cpSSL.
jar,sspiSSL.jar} -R
```

- Если Центр сертификации Aladdin eCA был ранее установлен, то назначьте владельцем данных файлов пользователя «aeca» и предоставьте ему права доступа к файлам (`chmod 700`) командами:

```
sudo chown aeca:aeca
/opt/aecaCa/services/cryptoproviders/{ASN1P.jar,asn1rt.jar,JCP.jar,JCSP.jar,cpSSL.
jar,sspiSSL.jar} -R
```

¹ Заранее сформированный набор случайных данных, необходимых для генерирования закрытых ключей. При создании сертификатов на КН и с закрытым ключом (PKCS#12) для субъектов с использованием алгоритмов ключей, для которых в активном центре сертификации выбран криптопровайдер СКЗИ «КриптоПро CSP», Центр сертификации использует внешнюю гамму, заранее подготовленную на биологическом датчике случайных числе (БДСЧ) криптопровайдера «КриптоПро CSP».

² Выбор места хранения осуществляется в разделе 3.1.3 документа «Центр сертификатов доступа Aladdin Enterprise Certificate Authority Certified Edition. Руководство администратора. Часть 2. Функции управления Центра сертификации Aladdin Enterprise Certification Authority» 33714370.03.01.001 32 01-2.

```
sudo chmod 700
/opt/aecaCa/services/cryptoproviders/{ASN1P.jar,asn1rt.jar,JCP.jar,JCSP.jar,cpSSL.
jar,sspiSSL.jar} -R
```

- Если используется уже заранее подготовленная внешняя гамма, то пропустите этот пункт. Иначе подготовьте внешнюю гамму с помощью утилиты `/opt/cprocsp/bin/amd64/genkpin` (утилита `genkpin` входит в состав дистрибутива СКЗИ «КриптоПро CSP») командами:

```
mkdir -p ~/gamma /opt/cprocsp/bin/amd64/genkpin <количество ключей> 0x12345678
~/gamma
```

- На хосте Центра сертификации Aladdin eCA поместите каталог с заранее подготовленной внешней гаммой в каталог `/opt/aecaCa/dist/` командой:

```
sudo cp -a ~/gamma/. /opt/aecaCa/dist/gamma
```

- В результате в каталоге `/opt/aecaCa/dist/gamma` появятся подкаталоги `db1`, `db2`, `kpin`.
 - Если выполняется первоначальная установка Центра сертификации Aladdin eCA, то назначьте права доступа файлам (`chmod 777`) командой:

```
sudo chmod 777 /opt/aecaCa/dist/gamma -R
```

- Если Центр сертификации Aladdin eCA был ранее установлен, то назначьте владельцем данных файлов пользователя «aeca» и предоставьте ему права доступа (`chmod 700`) командами:

```
sudo chown aeca:aeca /opt/aecaCa/dist/gamma -R
sudo chmod 700 /opt/aecaCa/dist/gamma -R
```

- Подключить данную внешнюю гамму к СКЗИ «КриптоПро CSP» посредством следующих команд¹:

```
sudo ./cpconfig -hardware rndm -add cpsd -name 'cpsd rng' -level 3
sudo ./cpconfig -hardware rndm -configure cpsd -add string /db1/kis_1
/opt/aecaCa/dist/gamma/db1/kis_1
sudo ./cpconfig -hardware rndm -configure cpsd -add string /db2/kis_1
/opt/aecaCa/dist/gamma/db2/kis_1
```

- Правильность настройки гаммы можно, выполнив команду с правами суперпользователя:

```
sudo /opt/cprocsp/bin/amd64/csptest -keyset -newkeyset -cont '\\.\HDIMAGE\1'
```

Если после выполнения команды произошел запрос пароля, гамма настроена правильно.

- Если Центр сертификации Aladdin eCA был ранее установлен, перезапустите сервис `aeca-ca.service` командой:

```
sudo systemctl restart aeca-ca.service
```

Если в дальнейшем к СКЗИ «КриптоПро CSP» будет подключен ПАКМ «КриптоПро HSM», для обнаружения Центром сертификации Aladdin eCA наличия такого подключения необходимо перезапустить сервис `aeca-ca.service`.

5.2 Индикация об отсутствии связи с СКЗИ «КриптоПро CSP»

При отсутствии на хосте Центра сертификации Aladdin eCA активного криптопровайдера «КриптоПро CSP» для центров сертификации, закрытый ключ которых создан с его помощью, в пользовательском интерфейсе будет отображаться следующая индикация:

¹ Подключение осуществляется с помощью файла `cpconfig` (находится в `/opt/cprocsp/sbin/amd64`). Путь к файлу в командах приведен с учётом нахождения в каталоге `/opt/cprocsp/sbin/amd64`.

- В разделе «Центр сертификации» в списках для центров сертификации в соответствующих строках слева от имени отображаемого центра сертификации будет присутствовать индикация вида «треугольник с восклицательным знаком», при наведении на которую курсора будет отображаться всплывающее сообщение «Закрытый ключ центра сертификации недоступен».
- При переходе на карточку центра сертификации справа от индикации состояния центра сертификации будет присутствовать индикация «треугольник с восклицательным знаком», при наведении на которую курсора будет отображаться всплывающее сообщение «Закрытый ключ центра сертификации недоступен».
- В карточке Центра сертификации в подразделе «Криптопровайдеры» справа от названия криптопровайдера СКЗИ «КриптоПро CSP» в полях алгоритмов, для которых он был выбран в качестве криптопровайдера при создании центра сертификации, будет присутствовать индикация «треугольник с восклицательным знаком», при наведении курсора на которую будет отображаться всплывающее сообщение «Криптопровайдер недоступен».

ПЕРЕЧЕНЬ ДОКУМЕНТАЦИИ ДЛЯ ОЗНАКОМЛЕНИЯ

Перед началом работы следует ознакомиться со следующей документацией, относящейся к программному обеспечению:

- официальная документация РЕД ОС 7.1
(адрес: <https://redos.red-soft.ru/base/manual/?ysclid=l5gg69co40129982631>);
- официальная документация Astra Linux Special Edition 1.7
(адрес: <https://wiki.astralinux.ru/pages/viewpage.action?pageId=137563555&ysclid=l5gg3t48tj885563182>);
- официальная документация Astra Linux Special Edition 1.8
(адрес: <https://wiki.astralinux.ru/pages/viewpage.action?pageId=302043140>);
- официальная документация Альт Сервер 8, релиз 10
(адрес: <https://www.basealt.ru/alt-server/docs>);
- официальная документация Postgres
(адрес: <http://www.postgresql.org/docs/12/index.html>);
- официальная документация Jatoba 4
(адрес: <https://www.gaz-is.ru/produkty/inform-sistemy/subd-jatoba.html#materialy>);
- официальная документация JC-Web Client Руководство пользователя
(адрес: <https://www.aladdin.kg>).

ОБОЗНАЧЕНИЯ И СОКРАЩЕНИЯ

БД	- База данных
ЗПС	- Замкнутая программная среда
КС	- Контрольная сумма
ОС	- Операционная система
ПАКМ	- Программно-аппаратный криптографический модуль
ПО	- Программное обеспечение
СВТ	- Средство вычислительной техники
СКЗИ	- Средство криптографической защиты информации
СУБД	- Система управления базами данных
ЦС	- Центр сертификатов
CSP	- Cryptography Service Provider
HTTP	- Hypertext Transfer Protocol
HTTPS	- Hyper Text Transfer Protocol Secure
LDAP	- Lightweight Directory Access Protocol
API	- Application Programming Interface
CRL	- Certificate Revocation List
AIA	- Authority Information Access
URL	- Uniform Resource Locator
TCP	- Transmission Control Protocol
TLS	- Transport Layer Security
VGA	- Video Graphics Array
WSTEP	- WS-Trust X.509v3 Token Enrollment Extensions
SCEP	- Simple Certificate Enrollment Protocol
HDD	- Hard (magnetic) Disk Drive
SMTP	- Simple Mail Transfer Protocol
OCSP	- Online Certificate Status Protocol

ТЕРМИНЫ И ОПРЕДЕЛЕНИЯ

Администратор инициализации - сотрудник (специалист), ответственный за приёмку и ввод в эксплуатацию изделия, которому доступны все функции роли «Администратор» в центре сертификации.

Артефакт - объект, применяемый или создаваемый в процессе разработки программного обеспечения.

Аутентификация - действия по проверке подлинности идентификатора пользователя. Под аутентификацией понимается ввод пароля или PIN-кода на средстве вычислительной техники в открытом контуре, а также процессы, реализующие проверку этих данных.

Ключевой носитель - это сущность в центре сертификации, соответствующая физическому токenu, программному или аппаратному модулю безопасности Hardware Security Module (HSM). С помощью крипто-токена ЦС осуществляет хранение ключей и выполнение криптографических операций.

Контрольный список - это текстовый файл, в котором содержатся контрольные суммы всех файлов, входящих в дистрибутив ПО «Центр сертификатов доступа Aladdin Enterprise Certificate Authority Certified Edition KG, записанный на компакт-диск с размещённым на нём дистрибутивом программы и комплектом документации.

Корневой ЦС - экземпляр центра сертификации в информационной системе, имеющий абсолютное доверие со стороны всех участников процесса строгой аутентификации. С точки зрения службы безопасности предприятия должен быть обеспечен максимальным уровнем защиты (отдельный ПК, отключённый от сети, с доступом ограниченного круга лиц). Корневой ЦС владеет само подписанным сертификатом, который должен распространяться доверенным способом в информационной системе.

Оператор - сотрудник (специалист) или система (приложение, сервис) и соответствующая роль в центре сертификации, отвечающая за управление жизненным циклом сертификатов субъектов.

Подчиненный ЦС - экземпляр центра сертификации в информационной системе, обладающий функцией управления политиками строгой аутентификации или функцией управления жизненным циклом сертификатов субъектов информационной системы. Подчиненный ЦС владеет сертификатом, выданным вышестоящим ЦС (Корневым или другим Подчиненным), который используется для проверки всей цепочки доверия сертификатов.

Расширение pgcrypto - предоставляет криптографические функции, которые позволяют администраторам баз данных PostgreSQL хранить определенные столбцы данных в зашифрованном виде.

Сервис валидации - служба, составная часть Центра сертификации, отвечающая за предоставление информации о действительности сертификатов. Предоставляет сервисы CRL DP, OCSP.

Сертификат - выпущенный центром сертификации цифровой документ в форматах x509v3 или другом поддерживаемом формате, подтверждающий принадлежность владельцу закрытого ключа или каких-либо атрибутов и предназначенный для аутентификации в информационной системе.

Событие безопасности - идентифицированное возникновение состояния системы, сервиса или сети, указывающего на возможное нарушение политики информационной безопасности, или сбой средств контроля, или ранее неизвестную ситуацию, которая может быть значимой для безопасности.

Список отозванных сертификатов (Certificate Revocation List - CRL) - список аннулированных (отозванных) сертификатов, издается центром сертификации по запросу или с заданной периодичностью на основании запросов об отзыве сертификатов.

Субъект - пользователь информационной системы или устройство (сервер, шлюз, маршрутизатор). Субъекту для строгой аутентификации в информационной системе в центре сертификации выдается сертификат. Синоним - конечная сущность (end entity).

Технологический ЦС - экземпляр центра сертификации в информационной системе, обладающий функцией первичной настройки программного компонента «Центр сертификации Aladdin Enterprise Certification Authority».

Центр сертификации - комплекс средств, задача которых заключается в обеспечении жизненного цикла сертификатов пользователей и устройств информационной системы, а также в создании инфраструктуры для обеспечения процессов идентификации и строгой аутентификации в информационной системе. Программный компонент «Центр сертификации» является частью Центра сертификатов Aladdin Enterprise Certificate Authority Certified Edition.

Шаблон субъекта - шаблон, на основании которого необходимо создавать субъекты. Шаблон определяет свойства субъекта (subject name, alternative name), свойства сертификата (криптографию, срок действия, назначение, политики и проч.), а также инфраструктурные характеристики (реквизиты для доставки сертификатов, возможности отзыва, хранения и проч.).

ЛИСТ РЕГИСТРАЦИИ ИЗМЕНЕНИЙ

[illegible]