УТВЕРЖДЕН

RU.АЛДЕ.03.16.001-05 32 01-ЛУ

ПРОГРАММНОЕ ОБЕСПЕЧЕНИЕ JACARTA MANAGEMENT SYSTEM 4LX

Руководство администратора. Часть 5

Установка и настройка JaCarta Identity Provider (JIP)

RU.АЛДЕ.03.16.001-05 32 01-5

Версия продукта	4LX
Версия документа	1.00
Статус	Публичный
Дата	10 февраля 2026 г.
Листов	131

2026

Подп. и дата

Инв. № дубл.

Взам. инв. №

Подп. и дата

Инв. № подл.

Оглавление

1. О документе 9
1.1 Назначение документа 9
1.2 На кого ориентирован данный документ 9
1.3 Соглашения по оформлению 9
1.4 Обозначения и сокращения 10
1.5 Авторские права, товарные знаки, ограничения 11
1.6 Лицензионное соглашение 12
2. Введение 15
2.1 Состав ПО JIP 15
3. Описание пакетов установки 15
4. Системные требования 17
5. Интерфейсы JIP 17
5.1 Интерфейсы сервера JIP 17
5.2 Интерфейсы web-приложения JIP 17
6. Взаимодействие с другими продуктами 17
6.1 Интеграция с JMS 17
6.2 Интеграция с JAS 17
6.3 Схема взаимодействия 18
7. Развертывание ПО JIP 19
7.1 Начальные условия для развертывания JIP 19
7.2 Порядок развертывания ПО JIP 19
8. Установка JIP-плагинов для компонентов JMS 19
8.1 Установка плагина для сервера JMS 19
8.2 Установка плагина для серверного web-приложения Консоль управления JMS 20
9. Установка сервера JIP 21
10. Установка web-консоли сервера JIP 22
11. Синхронизация данных из JMS в JIP 23
11.1 Требования к пользователю JMS 23
11.2 Синхронизация данных с помощью консольного агента 24
11.3 Синхронизация через API 25
12. Выбор приоритета OTP для JAS 26
13. Установка серверного web-приложения JIP 26
13.1 Установка web-приложения JIP на одном хосте с сервером JIP 26
13.2 Установка web-приложения JIP на отдельном хосте от сервера JIP 27

14. Настройка журналирования событий аутентификации 28
15. Обновление ПО JIP 29
15.1 Обновление JIP-плагина для сервера JMS 29
15.2 Обновление JIP-плагина для серверного web-приложения Консоль управления JMS 29
15.3 Обновление сервера JIP 30
15.4 Обновление Web-консоли сервера JIP 30
15.5 Обновление серверного web-приложения JIP 31
16. Удаление ПО JIP 31
16.1 Удаление JIP-плагина для сервера JMS 31
16.2 Удаление JIP-плагина для серверного web-приложения Консоль управления JMS 32
16.3 Удаление web-консоли сервера JIP 32
16.4 Удаление серверного web-приложения JIP 33
16.5 Удаление сервера JIP 33
17. Изменение настроек JIP 33
17.1 Через консольный агент 34
17.2 Через web-консоль сервера JIP 34
18. Настройки сервера JIP 35
18.1 Основные настройки сервиса 35
18.1.1 Путь до исполняемого файла 35
18.1.2 Адреса API управления 35
18.1.3 Адреса API администрирования 36
18.1.4 Адреса API проверки работоспособности 36
18.1.5 Язык интерфейса 36
18.1.6 Автоматический старт 36
18.1.7 Имя пользователя API управления 36
18.1.8 Пароль пользователя API управления 37
18.1.9 Имя пользователя API администрирования 37
18.1.10 Пароль администратора Administration Api 37
18.2 База данных 37
18.2.1 Тип СУБД 37
18.2.2 Адрес сервера 37
18.2.3 Порт сервера 37
18.2.4 Имя базы данных 37
18.2.5 Режим аутентификации для мастера развертывания 38
18.2.6 Имя пользователя для создания БД 38
18.2.7 Пароль пользователя сервера 38
18.2.8 Режим аутентификации для подключения к БД 38
18.2.9 Имя пользователя БД 38

18.2.10 Пароль пользователя БД 38
18.3 JAS 38
18.3.1 Адрес сервера 39
18.3.2 Тип аутентификации 39
18.3.3 Имя пользователя 39
18.3.4 Пароль 40
18.3.5 Таймаут запроса 40
18.3.6 Домен по-умолчанию 40
18.3.7 Поддерживаемые типы аутентификации 40
18.3.8 Таймаут для Push-аутентификации 41
18.3.9 Идентификатор системы 41
18.3.10 Время жизни кода 41
18.3.11 Таймаут между попытками аутентификации 41
18.3.12 Текст сообщения 41
18.4 JMS 42
18.4.1 Адрес authentication API 42
18.4.2 Адрес integration API 42
18.4.3 Имя домена 42
18.4.4 Имя пользователя 43
18.4.5 Пароль 43
18.5 SSO 43
18.5.1 Время жизни SSO cookie 44
18.5.2 Использовать скользящий срок действия cookie 44
18.5.3 Допустимое отклонение времени 44
18.5.4 SSO кука постоянного хранения 44
18.6 OIDC 45
18.6.1 Поддержка протокола OIDC 45
18.6.2 Выполнять backchannel logout при выходе из системы 46
18.6.3 Выполнять frontchannel logout при выходе из системы 46
18.6.4 Игнорировать prompt=none 46
18.6.5 Разрешить автоматическую переадресацию после выхода из системы 46
18.6.6 Автоматическая переадресация после frontchannel logout 46
18.6.7 Задержка перед автоматической переадресацией 47
18.6.8 Таймаут вызова backchannel logout 47
18.6.9 Издатель токена 47
18.6.10 Максимальное число автоматически отзываемых токенов 47
18.6.11 Таймаут для процесса отзыва 47
18.6.12 Отключить шифрование access token 48
18.6.13 Отпечаток сертификата 48
18.7 SAML 48
18.7.1 Поддержка протокола SAML 49
18.7.2 Издатель токена 49

18.7.3 Адрес для artefact resolution 50
18.7.4 Адрес single sign-on 50
18.7.5 Адрес single sign-out 50
18.7.6 Время жизни артефакта 50
18.7.7 Время жизни метаданных 50
18.7.8 Отпечаток сертификата 51
18.8 CORS 51
18.8.1 CORS включен 51
18.8.2 Разрешенные заголовки 52
18.8.3 Разрешенные методы 52
18.8.4 Разрешенные адреса 52
18.8.5 Разрешить адреса из настроек для клиентов SSO 52
18.8.6 Разрешить передачу credentials 52
18.8.7 Время кеширования результата запроса OPTIONS в браузере 53
18.9 Kerberos 53
18.9.1 Инициализация 54
18.9.2 Просмотр списка keytab-файлов 55
18.9.3 Регистрация keytab-файла 55
18.9.4 Просмотр данных keytab-файла 57
18.9.5 Удаление keytab-файла 57
18.9.6 Обновление keytab-файла 59
18.9.7 Скачивание keytab-файла 59
18.9.8 Управление автоопределением доступности Kerberos аутентификации (shortcut) 59
18.10 Syslog 60
18.10.1 Адрес сервера 61
18.10.2 Порт 61
18.10.3 Протокол 61
18.10.4 Защищенное соединение 61
18.10.5 Формат сообщения 61
18.10.6 Метод фрейминга 62
18.10.7 Приложение 62
18.10.8 Отправка тестового сообщения 62
18.10.9 Пример команды консольного агента 62
18.11 Журналирование 62
18.11.1 Уровень логирования 63
18.11.2 Параметры аутентификации 63
18.11.3 Пример команды консольного агента 63
18.12 API проверки работоспособности (Healthcheck API) 64
18.12.1 Адреса API проверки работоспособности 64
18.12.2 Таймаут проверки 64
18.12.3 Мягкий таймаут проверки 64

18.12.4 Эндпоинт проверки 64
18.12.5 Выполняемые проверки 64
18.12.6 Формат ответа 65
18.12.7 Статусы проверок 66
18.12.8 Swagger-документация 66
19. Настройки клиентов SSO в JMS 66
19.1 OIDC 66
19.1.1 Название 66
19.1.2 Тип клиента 66
19.1.3 ID клиента 66
19.1.4 Секрет 67
19.1.5 Включён 67
19.1.6 Introspection Feature 67
19.1.7 Refresh Token 67
19.1.8 Revocation Endpoint 67
19.1.9 Reference Tokens 67
19.1.10 Время жизни access token (минут) 67
19.1.11 Время жизни refresh token (минут) 67
19.1.12 Адреса переадресации после входа 68
19.1.13 Адреса переадресации после выхода 68
19.1.14 Адреса для back-channel logout 68
19.1.15 Адреса для front-channel logout 68
19.1.16 Режим автоматического отзыва токенов 68
19.1.17 Отзывать access token 68
19.1.18 Отзывать refresh token 68
19.2 SAML 69
19.2.1 Название 69
19.2.2 Issuer 69
19.2.3 Включён 69
19.2.4 Способ определения адресов клиента 69
19.2.5 Адрес метаданных 69
19.2.6 Адрес ACS клиента 69
19.2.7 Адрес Logout клиента 69
19.2.8 Использовать Artefact Binding 69
19.2.9 Отключить проверку подписи на логауте 70
19.2.10 Время жизни access token (минут) 70
19.2.11 Время жизни refresh token (минут) 70
20. Параметры профиля SSO в JMS 70
20.1.1 Имя 70
20.1.2 Описание 70
20.1.3 Клиенты SSO 70
20.1.4 Стратегии входа 70

20.1.5 Требуется повторная аутентификация 71
20.1.6 Утверждения OIDC 71
20.1.7 Утверждения SAML 71
21. Безопасность 72
21.1 Настройка HTTPS для API сервера JIP 72
21.1.1 Настройка локальной (внутренней) точки доступа без HTTPS 72
21.1.2 Генерация самоподписанного сертификата 73
21.1.3 Добавление сертификата в JIP 74
21.1.4 Включение HTTPS в API управления сервера JIP 75
21.1.5 Включение HTTPS в административном API сервера JIP 77
21.1.6 Включение HTTPS в API проверки работоспособности сервера JIP 78
21.2 Настройка HTTPS для web-приложения JIP 79
21.2.1 Генерация самоподписанного сертификата 80
21.2.2 Включение HTTPS в web-приложении JIP 80
21.3 Настройка HTTPS для web-консоли сервера JIP 83
21.3.1 Генерация самоподписанного сертификата 83
21.3.2 Включение HTTPS в web-приложении JIP 83
21.4 Хранение паролей 84
21.4.1 Сервер JIP 85
21.4.2 Web-приложение JIP 85
21.5 Изменение паролей 85
22. Интеграция с внешними приложениями 86
22.1 Интеграция по OIDC 86
22.1.1 Регистрация сервиса в JIP 86
22.1.2 Создание профиля SSO 88
22.1.3 Редактирование профиля SSO 89
22.1.4 Привязка профиля к дереву РС 89
22.1.5 Настройка сервиса в качестве клиента OIDC 89
22.2 Интеграция по SAML 90
22.2.1 Регистрация сервиса в JIP 90
22.2.2 Создание профиля SSO 91
22.2.3 Редактирование профиля SSO 93
22.2.4 Привязка профиля к дереву РС 93
22.2.5 Настройка сервиса в качестве клиента SAML 93
23. Настройка Kerberos 94
23.1 Подготовка к выдаче keytab-файла 94
23.2 Регистрация keytab-файла в JIP 94
23.3 Настройка браузера 94
23.3.1 Google Chrome и Microsoft Edge 94
23.3.2 Mozilla Firefox 95
23.4 Автоматический выбор метода Kerberos 96

24. Специфика работы в мультидоменной среде 97
25. Диагностика 97
25.1 Диагностика проблем с Kerberos 97
25.2 Диагностика событий аутентификации через аудит 98
25.2.1 Перечень аудируемых событий 98
25.2.2 Поля аудит-записей 98
26. Аутентификация 100
26.1 Идентификация пользователя 100
26.1.1 Источники идентификационной информации 100
26.1.2 Этапы процесса идентификации 100
26.1.3 Логика идентификации пользователя 100
26.2 SSO (Single-Sign-On) 102
26.3 SLO (Single Logout) 103
26.3.1 OIDC SLO 103
26.3.2 SAML SLO 104
26.3.3 Кросс-протокольный SLO 104
27. Поиск и устранение неисправностей 104
27.1 Ошибка проверки сертификатов подписи токенов OIDC/SAML. Сервер не стартует 104
28. Работа с web-консолью сервера JIP 106
28.1 Информация о сервере 108
28.2 Конфигурация сервера 109
28.2.1 Интеграция с сервером JAS 110
28.2.2 Интеграция с сервером JMS 112
28.2.3 Общие настройки SSO 113
28.2.4 Настройки OIDC 114
28.2.5 Настройки SAML 116
28.2.6 Настройки CORS 118
28.2.7 Настройки Kerberos 120
28.3 Конфигурация узла 121
29. Приложения 122
29.1 Приложение 1. Полный файл инициализации сервера JIP 122
29.2 Приложение 2. Минимальный файл инициализации сервера JIP 126
Контакты, техническая поддержка 128
Список литературы 129
Полезные web-ресурсы 129
Регистрация изменений 130

1. О документе

1.1 Назначение документа

Настоящий документ является руководства администратора по программному обеспечению JIP (JaCarta Identity Provider) и представляет собой описание операций по установке и настройке данного программного продукта для среды функционирования Linux.

1.2 На кого ориентирован данный документ

Документ предназначен для администраторов корпоративной информационной системы управления средствами аутентификации.

Изложенный материал предполагает наличие у администратора опыта в области системного и сетевого администрирования, информационной безопасности, администрирования СУБД, администрирования ОС Linux и Windows.

1.3 Соглашения по оформлению

В данном документе для представления ссылок, терминов и наименований, примеров кода программ используются различные шрифты и средства оформления. Основные типы начертаний текста приведены в таблице 1.

Табл. 1 — Элементы оформления

Выделение	Используется для выделения наименований полей, кнопок, секций, вкладок экранных форм
`file.exe`	Используется для выделения имен файлов, каталогов, текстов программ
[1]	Ссылка на пункт в списке литературы (приведен в конце документа)
Гиперссылка	Используется для выделения внешних ссылок
Ссылка, с. 9	Используется для выделения перекрестных ссылок
	Важная информация
	Ссылка, примечание, заметка
	Совет
	Рекомендация

1.4 Обозначения и сокращения

Табл. 2— Обозначения и сокращения

JIP	JaCarta Identity Provider, компонент JMS, предназначенный для обеспечения прозрачной аутентификации пользователей в серверных приложениях, поддерживающих протоколы OIDC (OpenID Connect) и SAML (Security Assertion Markup Language)
JMS	То же, что «Программное обеспечение JaCarta Management System 4LX»
JMS Web Admin	Серверное web-приложение Консоль управления JMS
JWA **(JMS Web Agent)**	Программное обеспечение, обеспечивающее взаимодействие web-клиента JMS с ЭК/ЗНИ из среды web-браузера.
JWA Tray **(JMS Web Agent Tray)**	Программа, позволяющая выполнять базовые операции с ЭК/ЗНИ пользователя в фоновом режиме или через простое графическое меню. Запущенное приложение отображается значком в области уведомлений рабочего стола
OIDC	OpenID Connect, протокол аутентификации, реализуемый сервером JIP, обеспечивающий SSO-сервис с использование JWT-токенов
SAML или **SAML 2.0**	Security Assertion Markup Language, протокол аутентификации, реализуемый сервером JIP для обеспечения SSO-сервиса
SSO	Single Sign-On, технология «единого входа», позволяет пользователю получить доступ к нескольким разнородным сервисам или приложениям, используя один набор аутентификационных данных без повторной аутентификации
Консольный агент	Приложение, предназначенное для конфигурирования сервера JMS. Устанавливается вместе с компонентом JMS Server
ПО	Программное обеспечение
РС	Ресурсная система – служба каталога (LDAP) или служба справочника, с которой осуществляется интеграция JMS. Примеры ресурсных систем: Microsoft Active Directory (AD), FreeIPA, ALD Pro, Samba AD
ФСБ	Федеральная служба безопасности Российской Федерации
ФСТЭК	Федеральная служба по техническому и экспортному контролю Российской Федерации

1.5 Авторские права, товарные знаки, ограничения

Ограничение ответственности

Информация, приведённая в данном документе, предназначена исключительно для ознакомления и не является исчерпывающей. Состав продуктов, компонент, их функции, характеристики, версии, доступность и пр. могут быть изменены АО «Аладдин Р. Д.» без предварительного уведомления.

АО «Аладдин Р. Д.» не гарантирует ни отсутствия ошибок в данном документе, ни того, что описанное программное обеспечение (ПО) не содержит дефектов, будет работать в произвольно выбранных условиях и при этом удовлетворять всем требованиям, которые могут быть к нему предъявлены.

АО «Аладдин Р. Д.» не гарантирует работоспособность нелегально полученного программного обеспечения. Нелегальное использование программного обеспечения и документации на него преследуется по закону.

Все указанные данные о характеристиках продуктов основаны на международных или российских стандартах и результатах тестирования, полученных в независимых тестовых или сертификационных лабораториях, либо на принятых в компании методиках. В данном документе АО «Аладдин Р. Д.» не предоставляет никаких ни явных, ни подразумеваемых гарантий.

АО «Аладдин Р. Д.» НЕ НЕСЁТ ОТВЕТСТВЕННОСТИ (КАК В СИЛУ ДОГОВОРА, ГРАЖДАНСКОГО ПРАВОНАРУШЕНИЯ, ВКЛЮЧАЯ ХАЛАТНОСТЬ, ТАК И В ЛЮБОЙ ИНОЙ ФОРМЕ) ПЕРЕД ВАМИ ИЛИ ЛЮБОЙ ТРЕТЬЕЙ СТОРОНОЙ ЗА ЛЮБЫЕ ПОТЕРИ ИЛИ УБЫТКИ (ВКЛЮЧАЯ КОСВЕННЫЕ, ФАКТИЧЕСКИЕ ИЛИ ПОБОЧНЫЕ УБЫТКИ), ВКЛЮЧАЯ БЕЗ ОГРАНИЧЕНИЙ ЛЮБЫЕ ПОТЕРИ ИЛИ УБЫТКИ ПРИБЫЛЬНОСТИ БИЗНЕСА, ПОТЕРЮ ДОХОДНОСТИ ИЛИ РЕПУТАЦИИ, УТРАЧЕННУЮ ИЛИ ИСКАЖЁННУЮ ИНФОРМАЦИЮ ИЛИ ДОКУМЕНТАЦИЮ ВСЛЕДСТВИЕ ИСПОЛЬЗОВАНИЯ ПРОГРАММНОГО ОБЕСПЕЧЕНИЯ И/ИЛИ ЛЮБОГО КОМПОНЕНТА ОПИСАННОГО ПРОДУКТА, ДАЖЕ ЕСЛИ АО «Аладдин Р. Д.» БЫЛО ПИСЬМЕННО УВЕДОМЛЕНО О ВОЗМОЖНОСТИ ПОДОБНЫХ УБЫТКОВ.

Государственное регулирование и экспортный контроль

Описываемый в данном документе продукт (или продукты) может являться или содержать в себе средство криптографической защиты информации (СКЗИ), являющееся предметом экспортного контроля.

Вы соглашаетесь с тем, что продукт не будет поставляться, передаваться или экспортироваться в какую-либо страну, а также использоваться каким-либо противоречащим закону образом.

Вы гарантируете, что будете соблюдать накладываемые на экспорт и реэкспорт продукта ограничения.

Сведения, приведённые в данном документе, актуальны на дату его публикации.

1.6 Лицензионное соглашение

ВАЖНО:

ПОЖАЛУЙСТА, ВНИМАТЕЛЬНО ПРОЧИТАЙТЕ ДАННОЕ ЛИЦЕНЗИОННОЕ СОГЛАШЕНИЕ, ПРЕЖДЕ ЧЕМ ОТКРЫТЬ ПАКЕТ С ПРОГРАММНЫМ ОБЕСПЕЧЕНИЕМ И/ИЛИ ИСПОЛЬЗОВАТЬ ЕГО СОДЕРЖИМОЕ И/ИЛИ ПРЕЖДЕ, ЧЕМ ЗАГРУЖАТЬ ИЛИ УСТАНАВЛИВАТЬ ПРОГРАММНОЕ ОБЕСПЕЧЕНИЕ.

ВСЕ УКАЗАНИЯ ПО ИСПОЛЬЗОВАНИЮ НАСТОЯЩЕГО ПРОГРАММНОГО ОБЕСПЕЧЕНИЯ (включая без ограничений библиотеки, утилиты, файлы для скачивания с Web-сайта, CD-ROM, Руководства, описания и др. документацию), далее «ПО», «Продукт»), ПРЕДОСТАВЛЯЕМЫЕ КОМПАНИЕЙ АО «Аладдин Р.Д.» (или любым дочерним предприятием – каждое из них упоминаемое как «КОМПАНИЯ») ПОДЧИНЯЮТСЯ И БУДУТ ПОДЧИНЯТЬСЯ УСЛОВИЯМ, ОГОВОРЕННЫМ В ДАННОМ СОГЛАШЕНИИ.

ОТКРЫВАЯ ПАКЕТ, СОДЕРЖАЩИЙ ПРОДУКТ И/ИЛИ ЗАГРУЖАЯ ДАННОЕ ПРОГРАММНОЕ ОБЕСПЕЧЕНИЕ как определено далее по тексту) И/ИЛИ УСТАНАВЛИВАЯ ДАННОЕ ПРОГРАММНОЕ ОБЕСПЕЧЕНИЕ НА ВАШ КОМПЬЮТЕР И/ИЛИ ИСПОЛЬЗУЯ ДАННЫЙ ПРОДУКТ, ВЫ ПРИНИМАЕТЕ ДАННОЕ СОГЛАШЕНИЕ И СОГЛАШАЕТЕСЬ С ЕГО УСЛОВИЯМИ.

ЕСЛИ ВЫ НЕ СОГЛАСНЫ С ДАННЫМ СОГЛАШЕНИЕМ, НЕ ОТКРЫВАЙТЕ ЭТОТ ПАКЕТ И/ИЛИ НЕ ЗАГРУЖАЙТЕ И/ИЛИ НЕ УСТАНАВЛИВАЙТЕ ДАННОЕ ПРОГРАММНОЕ ОБЕСПЕЧЕНИЕ И НЕЗАМЕДЛИТЕЛЬНО (не позднее 7 дней с даты получения этого пакета) ВЕРНИТЕ ЭТОТ ПРОДУКТ В АЛАДДИН Р.Д., СОТРИТЕ ДАННОЕ ПРОГРАММНОЕ ОБЕСПЕЧЕНИЕ И ВСЕ ЕГО ЧАСТИ В СВОЕМ КОМПЬЮТЕРЕ И НЕ ИСПОЛЬЗУЙТЕ ЕГО НИКОИМ ОБРАЗОМ.

Лицензионное соглашение на использование программного обеспечения.

Настоящее лицензионное соглашение (далее "Соглашение") является договором, заключенным между Вами (физическим или юридическим лицом) - конечным пользователем (далее "Пользователь") и компанией АО «Аладдин Р.Д.» (далее «компания Аладдин Р.Д.», «Правообладатель») относительно предоставления неисключительного права на использование настоящего программного обеспечения - комплекса программ для ЭВМ, и документации (печатные материалы, носители и файлы с информацией), являющихся неотъемлемой частью ПО, включая все дальнейшие усовершенствования.

Лицензионный договор считается заключенным с момента начала использования Вами ПО любым способом или с момента, когда Вы примете все условия настоящего Лицензионного договора в процессе установки ПО. Лицензионный договор сохраняет свою силу в течение всего срока действия исключительного права на ПО, если только иное не оговорено в Лицензионном договоре или в отдельном письменном договоре между Вами и компанией Аладдин Р.Д. Срок действия Лицензионного договора также может зависеть от объема Вашей Лицензии, описанного в данном Лицензионном договоре.

Права на ПО охраняются действующими законодательством и международными соглашениями. Вы подтверждаете свое согласие с тем, что Лицензионный договор имеет такую же юридическую силу, как и любой другой письменный договор, заключенный Вами. В случае нарушения Лицензионного договора Вы можете быть привлечены в качестве ответчика.

1. **Предмет Соглашения**

1.1. Предметом настоящего Соглашения является передача Правообладателем конечному Пользователю неисключительного права на использование ПО. ДАННОЕ СОГЛАШЕНИЕ НЕ ЯВЛЯЕТСЯ СОГЛАШЕНИЕМ О ПРОДАЖЕ. Все условия, оговоренные далее, относятся как к ПО в целом, так и ко всем его компонентам в отдельности. Данное соглашение не передает Вам права на Программное обеспечение, а лишь предоставляет ограниченное право на использование, которое подлежит отмене согласно условиям данного Соглашения. Ничего в данном Соглашении не подтверждает отказ компании Аладдин Р.Д. от прав на интеллектуальную собственность по какому бы то ни было законодательству.

1.2. Компания Аладдин Р.Д. сохраняет за собой все права, явным образом не предоставленные Вам настоящим Лицензионным договором. Настоящий Лицензионный договор не предоставляет Вам никаких прав на товарные знаки Компании Аладдин Р.Д..

1.3. В случае, если Вы являетесь физическим лицом, то территория, на которой допускается использование ПО, включает в себя весь мир. В случае, если Вы являетесь юридическим лицом (обособленным подразделением юридического лица), то территория на которой допускается приобретение ПО, ограничена страной регистрации юридического лица (обособленного подразделения юридического лица), если только иное не оговорено в отдельном письменном договоре между Вами и Компанией Аладдин Р.Д.

2. **Имущественные права**

2.1. Программное обеспечение, включая все переработки, исправления, модификации, дополнения, обновления и/или усовершенствования к нему (далее по всему тексту и любой его части определяемое как "Программное обеспечение"), и связанная с ним документация предназначается НЕ ДЛЯ ПРОДАЖИ и является и остается исключительной собственностью компании Аладдин Р.Д.

2.2. Все права на интеллектуальную собственность (включая, без ограничений, авторские права, коммерческую тайну, товарные знаки, и т.д.), подтвержденные или включенные в приложенные/взаимосвязанные/имеющие отношение к данному руководству, данные, содержащиеся в нем, а также все права на ПО являются и будут являться собственностью исключительно компании Аладдин Р.Д.

2.3. Вам, конечному Пользователю, предоставляется неисключительное право на использование ПО в указанных в документации целях и при соблюдении приведенных ниже условий.

3. **Условия использования**

3.1. ПО может быть использовано только в строгом соответствии с документами, инструкциями и рекомендациями Правообладателя, относящимися к данному ПО.

3.2. ПО может предоставляться на нескольких носителях, в том числе с помощью сети интернет. Независимо от количества носителей, на которых Вы получили ПО, Вы имеете право использовать ПО только в объеме предоставленной Вам Лицензии.

3.3. После уплаты Вами соответствующего вознаграждения компания Аладдин Р.Д. настоящим предоставляет Вам, а Вы получаете индивидуальное, неисключительное и ограниченное право на использование данного Программного обеспечения только в форме исполняемого кода, как описано в прилагаемой к Программному обеспечению документации и только в соответствии с условиями данного Соглашения:

- Вы можете установить Программное обеспечение и использовать его на компьютерах, расположенных в пределах Вашего предприятия, как описано в соответствующей документации компании Аладдин Р.Д.
- Вы можете добавить/присоединить Программное обеспечение к программам Вашего компьютера с единственной целью, описанной в данном Соглашении.

Продукт должен использоваться и обслуживаться строго в соответствии с описаниями и инструкциями компании Аладдин Р.Д., приведенными в данном и других документах компании Аладдин Р.Д.

3.4. За исключением указанных выше разрешений, Вы обязуетесь:

3.4.1. Не использовать и не выдавать сублицензии на данное Программное обеспечение и любую другую Продукцию компании Аладдин Р.Д., за исключением явных разрешений в данном Соглашении и в Руководстве по интеграции.

3.4.2. Не продавать, не выдавать лицензий или сублицензий, не сдавать в аренду или в прокат, не передавать, не переводить на другие языки, не закладывать, не разделять Ваши права в рамках данного Соглашения с кем-либо или кому-либо еще.

3.4.3. Не модифицировать (в том числе не вносить в ПО изменения в целях его функционирования на технических средствах Конечного пользователя), не демонтировать, не декомпилировать или дизассемблировать, не реконструировать, не видоизменять и не расширять данное Программное обеспечение и не пытаться раскрыть (получить) исходные коды данного Программного обеспечения.

3.4.4. Не помещать данное Программное обеспечение на сервер с возможностью доступа к нему третьих лиц через открытую сеть.

3.4.5. Не использовать какие бы то ни было резервные или архивные копии данного Программного обеспечения (или позволять кому-либо еще использовать такие копии) с любой иной целью, кроме замены его оригинального экземпляра в случае его разрушения или наличия дефектов.

3.4.6. Не пытаться обойти технические ограничения в Программе;

3.4.7. Не использовать Программу для оказания услуг на платной и бесплатной основе;

3.4.8. Не создавать условия для использования ПО лицами, не имеющими прав на использование ПО, в том числе работающими с Вами в одной многопользовательской системе или сети Интернет.

3.4.9. Вы не вправе удалять, изменять или делать малозаметными любые уведомления об авторских правах, правах на товарные знаки или патенты, которые указаны на/в ПО.

3.4.10. Вы обязуетесь соблюдать права третьих лиц, в том числе авторские права на объекты интеллектуальной собственности.

3.5. Компания Аладдин Р.Д. не несет обязательств по предоставлению поддержки, обслуживания, модификации или выходу новых релизов данного Программного обеспечения.

Нелегальное использование, распространение и воспроизведение (копирование) программного обеспечения является нарушением действующего законодательства и преследуется по Закону.

В случае нарушения настоящего Соглашения Правообладатель лишает Пользователя права на использование ПО. При этом Правообладатель полностью отказывается от своих гарантийных обязательств.

4. Ограниченная гарантия

Компания Аладдин Р.Д. гарантирует, что:

Данное Программное обеспечение с момента поставки его Вам в течение двенадцати (12) месяцев будет функционировать в полном соответствии с Руководством Пользователя (Администратора), при условии, что оно будет использоваться на компьютерном аппаратном обеспечении и с операционной системой, для которой оно было разработано.

Правообладатель гарантирует соответствие компонентов ПО спецификациям, а также работоспособность ПО при выполнении Пользователем условий, оговоренных в документации на ПО. ПО поставляется "таким, какое оно есть". Правообладатель не гарантирует, что ПО соответствует вашим требованиям, и что все действия ПО будут выполняться безошибочно. Правообладатель не гарантирует корректную совместную работу ПО с программным обеспечением или оборудованием других производителей.

5. Отказ от гарантии

5.1. КОМПАНИЯ АЛАДДИН Р.Д. НЕ ГАРАНТИРУЕТ, ЧТО ЛЮБОЙ ИЗ ЕГО ПРОДУКТОВ БУДЕТ СООТВЕТСТВОВАТЬ ВАШИМ ТРЕБОВАНИЯМ, ИЛИ ЧТО ЕГО РАБОТА БУДЕТ БЕСПЕРЕБОЙНОЙ ИЛИ БЕЗОШИБОЧНОЙ. В ОБЪЕМЕ, ПРЕДУСМОТРЕННОМ ЗАКОНОДАТЕЛЬСТВОМ РФ, КОМПАНИЯ АЛАДДИН Р.Д. ОТКРЫТО ОТКАЗЫВАЕТСЯ ОТ ВСЕХ ГАРАНТИЙ, НЕ ОГОВОРЕННЫХ ЗДЕСЬ, ОТ ВСЕХ ПОДРАЗУМЕВАЕМЫХ ГАРАНТИЙ, ВКЛЮЧАЯ ГАРАНТИЮ ТОВАРНОГО ВИДА И ПРИГОДНОСТИ ИСПОЛЬЗОВАНИЯ ДЛЯ ОПРЕДЕЛЕННОЙ ЦЕЛИ.

НИ ОДИН ИЗ ДИЛЕРОВ, ДИСТРИБЬЮТОРОВ, ПРОДАВЦОВ, АГЕНТОВ ИЛИ СОТРУДНИКОВ КОМПАНИИ АЛАДДИН Р.Д. НЕ УПОЛНОМОЧЕН ПРОИЗВОДИТЬ МОДИФИКАЦИИ, РАСШИРЕНИЯ ИЛИ ДОПОЛНЕНИЯ К ДАННОЙ ГАРАНТИИ.

5.2. Если Вы произвели какие-либо модификации Программного обеспечения или любой из частей данного Продукта во время гарантийного периода, то гарантия, упомянутая выше, будет немедленно прекращена.

5.3. Гарантия недействительна, если Продукт используется на или в сочетании с иным аппаратным и/или программным обеспечением, отличным от описанных в документации, или используется на компьютере с любым установленным нелицензионным программным обеспечением.

5.4. ПО и обновления предоставляются такими, каковы они есть, и Компания Аладдин Р.Д. не предоставляет на них никаких гарантий. Компания Аладдин Р.Д. не гарантирует и не может гарантировать работоспособность ПО и результаты, которые Вы можете получить, используя ПО.

5.5. За исключением гарантий и условий, которые не могут быть исключены или ограничены в соответствии с применимым законодательством, Компания Аладдин Р.Д. не предоставляет Вам никаких гарантий (в том числе явно выраженных или подразумевающихся в статутном или общем праве или обычаями делового оборота) ни на что, включая, без ограничения, гарантии о не нарушении прав третьих лиц, товарной пригодности, интегрируемости, удовлетворительного качества и годности к использованию ПО. Все риски, связанные с качеством работы и работоспособностью ПО, возлагаются на Вас.

5.6. Компания Аладдин Р.Д. не предоставляет никаких гарантий относительно программами для ЭВМ других производителей, которые могут предоставляться в составе ПО.

6. Исключение косвенных убытков

Стороны признают, что Продукт по сути своей сложный и не может быть полностью лишен ошибок. КОМПАНИЯ АЛАДДИН Р.Д. НЕ НЕСЕТ ОТВЕТСТВЕННОСТИ (КАК В СИЛУ ДОГОВОРА, ГРАЖДАНСКОГО ПРАВОНАРУШЕНИЯ, ВКЛЮЧАЯ ХАЛАТНОСТЬ, ТАК И В ЛЮБОЙ ИНОЙ ФОРМЕ) ПЕРЕД ВАМИ ИЛИ ЛЮБОЙ ТРЕТЬЕЙ СТОРОНОЙ ЗА ЛЮБЫЕ ПОТЕРИ ИЛИ УБЫТКИ (ВКЛЮЧАЯ КОСВЕННЫЕ, ФАКТИЧЕСКИЕ, ПОБОЧНЫЕ ИЛИ ПОТЕНЦИАЛЬНЫЕ УБЫТКИ), ВКЛЮЧАЯ, БЕЗ ОГРАНИЧЕНИЙ, ЛЮБЫЕ ПОТЕРИ ИЛИ УБЫТКИ ПРИБЫЛЬНОСТИ БИЗНЕСА, ПОТЕРЮ ДОХОДНОСТИ ИЛИ РЕПУТАЦИИ, УТРАЧЕННУЮ ИЛИ ИСКАЖЕННУЮ ИНФОРМАЦИЮ ИЛИ ДОКУМЕНТАЦИЮ ВСЛЕДСТВИЕ КАКОГО-ЛИБО ИСПОЛЬЗОВАНИЯ ДАННОГО ПРОГРАММНОГО ОБЕСПЕЧЕНИЯ И/ИЛИ ЛЮБОЙ КОМПОНЕНТЫ ДАННОГО ПРОДУКТА, ДАЖЕ ЕСЛИ АЛАДДИН Р.Д. ПИСЬМЕННО УВЕДОМЛЕН О ВОЗМОЖНОСТИ ПОДОБНЫХ УБЫТКОВ.

7. Ограничение ответственности

В СЛУЧАЕ ЕСЛИ, НЕСМОТРЯ НА УСЛОВИЯ ДАННОГО СОГЛАШЕНИЯ, КОМПАНИЯ АЛАДДИН Р.Д. ПРИЗНАНА ОТВЕТСТВЕННОЙ ЗА УБЫТКИ НА ОСНОВАНИИ КАКИХ-ЛИБО ДЕФЕКТОВ ИЛИ НЕСООТВЕТСТВИЯ ЕГО ПРОДУКТОВ, ПОЛНАЯ ОТВЕТСТВЕННОСТЬ ЗА КАЖДУЮ ЕДИНИЦУ ДЕФЕКТНЫХ ПРОДУКТОВ НЕ БУДЕТ ПРЕВЫШАТЬ СУММУ, ВЫПЛАЧЕННУЮ КОМПАНИИ АЛАДДИН Р.Д. ЗА ЭТИ ДЕФЕКТНЫЕ ПРОДУКТЫ.

Компания Аладдин Р.Д. ни при каких обстоятельствах не несет перед Вами никакой ответственности за убытки, вынужденные перерывы в деловой активности, потерю деловых либо иных данных или информации, претензии или расходы, реальный ущерб, а также упущенную выгоду и утерянные сбережения, вызванные использованием или связанные с использованием ПО, а также за убытки, вызванные возможными ошибками и опечатками в ПО и/или в документации, даже если Компании Аладдин Р.Д. стало известно о возможности таких убытков, потерь, претензий или расходов, равно как и за любые претензии со стороны третьих лиц. Вышеперечисленные ограничения и исключения действуют в той степени, насколько это разрешено применимым законодательством. Единственная ответственность Компании Аладдин Р.Д. по настоящему Лицензионному договору ограничивается суммой, которую Вы уплатили за ПО.

8. Прекращение действия

В случае невыполнения Вами условий данного Соглашения действие Вашей лицензии и настоящего Соглашения будет прекращено.

После прекращения действия данного Лицензионного соглашения:

(i) Лицензия, предоставленная Вам данным Соглашением, прекращает свое действие, и Вы после ее прекращения не сможете продолжать дальнейшее использование данного Программного обеспечения и других лицензионных Продуктов;

(ii) Вы незамедлительно вернете в компанию Аладдин Р.Д. все имущество, в котором используются права Аладдин Р.Д. на интеллектуальную собственность и все копии такового и/или сотрете/удалите любую информацию, содержащуюся в них в электронном виде. Разделы 1, 3, 6-11 будут продолжать действовать даже в случае прекращения действия настоящего Соглашения.

9. **Срок действия Договора**

9.1. Если иное не оговорено в настоящем Лицензионном договоре либо в отдельном письменном договоре между Вами и Компанией Аладдин Р.Д., настоящий Лицензионный договор действует в течение всего срока действия исключительного права на ПО.

9.2. В случае нарушения вами условий настоящего Соглашения или неспособности далее выполнять его условия вы обязуетесь уничтожить все копии ПО (включая архивные, файлы с информацией, носители, печатные материалы) или вернуть все относящиеся к ПО материалы организации, в которой вы приобрели ПО. После этого Соглашение прекращает свое действие.

9.3. Без ущерба для каких-либо других прав Компания Аладдин Р.Д. имеет право в одностороннем порядке расторгнуть настоящий Лицензионный договор при несоблюдении Вами его условий и ограничений. При прекращении действия настоящего Лицензионного договора Вы обязаны уничтожить все имеющиеся у Вас копии ПО (включая архивные, файлы с информацией, носители, печатные материалы), все компоненты ПО, а также удалить ПО и вернуть все относящиеся к ПО материалы организации, в которой вы приобрели ПО.

9.4. Вы можете расторгнуть настоящий Лицензионный договор удалив ПО и уничтожив все копии ПО, все компоненты ПО и сопровождающую его документацию. Такое расторжение не освобождает Вас от обязательств оплатить ПО.

10. **Применимое законодательство**

Данное Соглашение должно быть истолковано и определено в соответствии с законами Российской Федерации (за исключением конфликта применения правовых норм), и только российский суд уполномочен осуществлять правосудие в любых конфликтах и спорах, вытекающих из данного Соглашения. Применение Конвенции Организации Объединенных Наций о Договорах международной купли-продажи товаров (the United Nations Convention of Contracts for the International Sale of Goods) однозначно исключается. Невозможность для любой из сторон воспользоваться любым из прав, предоставленных ей по данному Соглашению, или принять меры против другой стороны в случае любого нарушения своих обязательств по Соглашению не должно рассматриваться как отказ этой стороны от последующего понуждения к признанию своих прав или совершению последующих действий в случае дальнейших нарушений.

11. **Государственное регулирование и экспортный контроль**

Приобретая и/или начиная использовать Продукт, Вы обязуетесь соблюдать все применимые международные и национальные законы, которые распространяются на продукты, подлежащие экспортному контролю. Настоящее ПО не должно экспортироваться или реэкспортироваться в нарушение экспортных ограничений, имеющихся в законодательстве страны, в которой приобретено или получено ПО. Вы также подтверждаете, что применимое законодательство не запрещает Вам приобретать или получать ПО.

12. **Программное обеспечение третьих сторон**

Если Продукт содержит в себе любое программное обеспечение, предоставленное какой-либо третьей стороной, такое программное обеспечение третьей стороны предоставляется "как есть" без какой-либо гарантии, и разделы 2, 3, 6, 8, 9-12 настоящего Соглашения применяются ко всем таким поставщикам программного обеспечения и к поставляемому ими программному обеспечению, как если бы это были Аладдин Р.Д. и Продукт соответственно.

13. **Разное**

13.1. Настоящее Соглашение представляет собой полное соглашение, относящееся к данной лицензии, и может быть изменено только посредством письменного соглашения, подписанного обеими сторонами. Если выполнение какого-либо условия настоящего Соглашения представляется невозможным, такое условие будет скорректировано только в пределах, обеспечивающих возможность выполнения данного условия.

13.2. Все права на материалы, не содержащиеся в ПО, но доступные посредством использования ПО, принадлежат своим законным владельцам и охраняются действующим законодательством об авторском праве и международными соглашениями. Настоящий Лицензионный договор не предоставляет Вам никаких прав на использование такой интеллектуальной собственности.

13.3. ПО содержит коммерческую тайну и иную конфиденциальную информацию, принадлежащую Компании Аладдин Р.Д. и третьим лицам, которая охраняется действующим законодательством Российской Федерации, международными соглашениями и законодательством страны приобретения и/или использования ПО.

13.4. Вы соглашаетесь на добровольную передачу Компании Аладдин Р.Д. в процессе использования и регистрации ПО своих персональных данных и выражаете свое согласие на сбор, обработку, использование своих персональных данных в соответствии с применимым законодательством, на условиях обеспечения конфиденциальности. Предоставленные Вами персональные данные будут храниться и использоваться только внутри Компании Аладдин Р.Д. и ее дочерних компаний и не будут предоставлены третьим лицам, за исключением случаев, предусмотренных применимым законодательством.

13.5. В случае предъявления любых претензий или исков, связанных с использованием Вами ПО Вы обязуетесь сообщить Компании Аладдин Р.Д. о таких фактах в течение трех (3) дней с момента, когда Вам стало известно об их возникновении. Вы обязуетесь совершить необходимые действия для предоставления Компании Аладдин Р.Д. возможности участвовать в рассмотрении таких претензий или исков, а также предоставлять необходимую информацию для урегулирования соответствующих претензий и/или исков в течение семи (7) дней с даты получения запроса от Компании Аладдин Р.Д.

13.6. Вознаграждением по настоящему Лицензионному договору признается стоимость Лицензии на ПО, установленная Компанией Аладдин Р.Д. или Партнером Компании Аладдин Р.Д., которая, подлежит уплате в соответствии с определяемым Компанией Аладдин Р.Д. или Партнером Компании Аладдин Р.Д. порядком. Вознаграждение также может быть включено в стоимость приобретенного Вами оборудования или в стоимость полной версии ПО. В случае если Вы являетесь физическим лицом, настоящий Лицензионный договор может быть безвозмездным.

13.7. В случае если какая-либо часть настоящего Лицензионного договора будет признана утратившей юридическую силу (недействительной) и не подлежащей исполнению, остальные части Лицензионного договора сохраняют свою юридическую силу и подлежат исполнению.

Я ПРОЧИТАЛ И ПОНЯЛ НАСТОЯЩЕЕ ЛИЦЕНЗИОННОЕ СОГЛАШЕНИЕ И СОГЛАСЕН ВЫПОЛНЯТЬ ВСЕ ЕГО УСЛОВИЯ.

Я ПРИНИМАЮ ДАННОЕ ЛИЦЕНЗИОННОЕ СОГЛАШЕНИЕ ЦЕЛИКОМ.

ЕСЛИ Я НЕ ПРИНИМАЮ ЭТО ЛИЦЕНЗИОННОЕ СОГЛАШЕНИЕ ИЛИ ХОТЯ БЫ ОДИН ИЗ ЕГО ПУНКТОВ, ТО ДАННОЕ ЛИЦЕНЗИОННОЕ СОГЛАШЕНИЕ НЕ ВСТУПАЕТ В СИЛУ, И Я ОБЯЗУЮСЬ НЕ УСТАНАВЛИВАТЬ И НЕ ИСПОЛЬЗОВАТЬ ДАННОЕ ПРОГРАММНОЕ ОБЕСПЕЧЕНИЕ.

2. Введение

JaCarta Identity Provider (JIP) – это программный комплекс, предназначенный для организации централизованной аутентификации пользователей и обеспечения единого входа (Single Sign-On, SSO) во внутренние и внешние информационные системы предприятия.

Решение реализует безопасное взаимодействие между пользователями и прикладными системами за счёт поддержки стандартных протоколов аутентификации и федерации — OIDC (OpenID Connect) и SAML (Security Assertion Markup Language).

2.1 Состав ПО JIP

Программное обеспечение JIP состоит из нескольких компонентов, каждый из которых поставляется в виде отдельного deb/rpm пакета:

- **aladdin-jip-web** – серверное web-приложение JIP, выполняет непосредственно аутентификацию пользователей, предоставляет необходимые web-формы и взаимодействует с внешними приложениями по протоколам OIDC и SAML;
- **aladdin-jip-engine** – сервер JIP, обеспечивает доступ для web-приложения JIP к данным, как самого JIP, так и данным хранящимся в JMS. Обеспечивает интеграцию с JMS и JAS. Включает в свой состав консольный агент JIP;
- **jip-agent** – консольный агент JIP, устанавливается вместе с сервером JIP, позволяет изменять настройки, управлять состоянием сервера JIP, инициализировать/обновлять сервер JIP. Требует запуска с sudo или из-под root;
- **aladdin-jip-eap-engine-plugin** – плагин для сервера JMS, расширяет API и БД сервера JMS для поддержки работы с клиентами SSO в Консоли управления JMS, также регистрирует новый тип профиля – профиль SSO;
- **aladdin-jip-eap-web-admin-plugin** – плагин для серверного web-приложения Консоль управления JMS, добавляет в него экраны клиентов SSO (отдельно OIDC и SAML) и редактор профиля SSO;
- **aladdin-jip-wsc** – web-консоль сервера JIP, предназначена для конфигурирования как кластера JIP, так и отдельных его узлов.

3. Описание пакетов установки

В поставку JIP входят следующие пакеты установки.

Табл. 3 – Пакеты установки компонента Сервер JIP

Файл	Описание
`aladdin-jip-engine_x.x.x.xxxx_x64.deb`	deb-пакет установки для среды для ОС Astra Linux
`aladdin-jip-engine_x.x.x.xxxx_x64.rpm`	rpm-пакет установки для среды РЕД ОС
`aladdin-jip-engine_x.x.x.xxxx_alt_x64.rpm`	rpm-пакет установки для среды ОС Альт

Табл. 4 – Пакеты установки компонента Web-консоль сервера JIP

Файл	Описание
aladdin-jip-wsc_x.x.x.xxxx_x64.deb	deb-пакет установки для среды для ОС Astra Linux
aladdin-jip-wsc_x.x.x.xxxx_x64.rpm	rpm-пакет установки для среды РЕД ОС
aladdin-jip-wsc_x.x.x.xxxx_alt_x64.rpm	rpm-пакет установки для среды ОС Альт

Табл. 5 – Пакеты установки компонента Серверное web-приложение JIP

Файл	Описание
aladdin-jip-web_x.x.x.xxxx_x64.deb	deb-пакет установки для среды для ОС Astra Linux
aladdin-jip-web_x.x.x.xxxx_x64.rpm	rpm-пакет установки для среды РЕД ОС
aladdin-jip-web_x.x.x.xxxx_alt_x64.rpm	rpm-пакет установки для среды ОС Альт

Табл. 6 – Пакеты установки компонента JIP-плагин для сервера JMS

Файл	Описание
aladdin-jip-eap-engine-plugin_x.x.x.xxxx_x64.deb	deb-пакет установки для среды для ОС Astra Linux
aladdin-jip-eap-engine-plugin_x.x.x.xxxx_x64.rpm	rpm-пакет установки для среды РЕД ОС
aladdin-jip-eap-engine-plugin_x.x.x.xxxx_alt_x64.rpm	rpm-пакет установки для среды ОС Альт

Табл. 7 – Пакеты установки компонента JIP- плагин для серверного web-приложения Консоль управления JMS

Файл	Описание
aladdin-jip-eap-web-admin-plugin_x.x.x.xxxx_x64.deb	deb-пакет установки для среды для ОС Astra Linux
aladdin-jip-eap-web-admin-plugin_x.x.x.xxxx_x64.rpm	rpm-пакет установки для среды РЕД ОС
aladdin-jip-eap-web-admin-plugin_x.x.x.xxxx_alt_x64.rpm	rpm-пакет установки для среды ОС Альт

4. Системные требования

Серверная ОС: Astra Linux 1.7.7, RedOS 7.3, ALT Linux 11 или более поздние версии данных ОС

СУБД: PostgreSQL 11, MS SQL Server 2016 или более поздние версии данных СУБД

Сервер JMS: 4LX (v.4.1)

Серверное web-приложение Консоль управления JMS: 4LX (v.4.1)

5. Интерфейсы JIP

5.1 Интерфейсы сервера JIP

Сервер JIP предоставляет два интерфейса:

1. API управления (control API) – доступен согласно настройке описанной в разделе 18.1.2 Адреса API управления. Предназначен для управления сервером и его настройками через консольный агент JIP, web-консоль сервера JIP или напрямую
2. API Администрирования (admin API) – доступен согласно настройке описанной в разделе 18.1.3 Адреса API администрирования. Предназначен для обеспечения доступа к данным (в том числе данным JMS) и конфигурации для web-приложения JIP
3. API Проверки работоспособности (healthcheck API) – доступен согласно настройке описанной в разделе 18.1.4 Адреса API проверки работоспособности. Предназначен для получения информации по работоспособности системы

Для обоих API доступен Swagger UI по пути **/swagger**. Для обоих API возможна установка HTTPS согласно инструкциям из раздела 21.1 Настройка HTTPS.

5.2 Интерфейсы web-приложения JIP

Web-приложение JIP предоставляет один интерфейс: Интеграционный API – доступен согласно параметрам, указанным при развертывании (13 Установка серверного web-приложения JIP). Обеспечивает возможность интеграции по протоколам SAML и OIDC.

6. Взаимодействие с другими продуктами

ПО JIP функционирует во взаимодействии с серверами JMS и JAS.

6.1 Интеграция с JMS

В JMS хранятся данные клиентских приложений, интегрированных с JIP по протоколам OIDC и SAML, а также профили задающие политики входа через JIP для пользователей JMS. JIP получает эти данные с помощью фоновой синхронизации. Помимо этого, JIP получает из JMS в момент аутентификации пользователя информацию о действующих на этого пользователя профилях SSO. В связи с чем при отсутствии связи с JMS аутентификация будет невозможна. Более подробное описание синхронизации и её настройки приведено в разделе 11 Синхронизация данных из JMS в JIP.

6.2 Интеграция с JAS

JIP обращается к JAS для прохождения аутентификации по паролю пользователя в ресурсной системе, OTP (SOTP, Messaging) или Push (за счет интеграции JAS с A2FA). При отсутствии связи с JAS аутентификация по указанным методам будет невозможна. JIP позволяет настроить приоритет

для методов аутентификации доступных через JAS, это описано в разделе 12 Выбор приоритета OTP для JAS.

6.3 Схема взаимодействия

Ниже приведена схема взаимодействия различных компонентов JIP (зеленый цвет) между собой и с JMS и JAS (синий цвет).

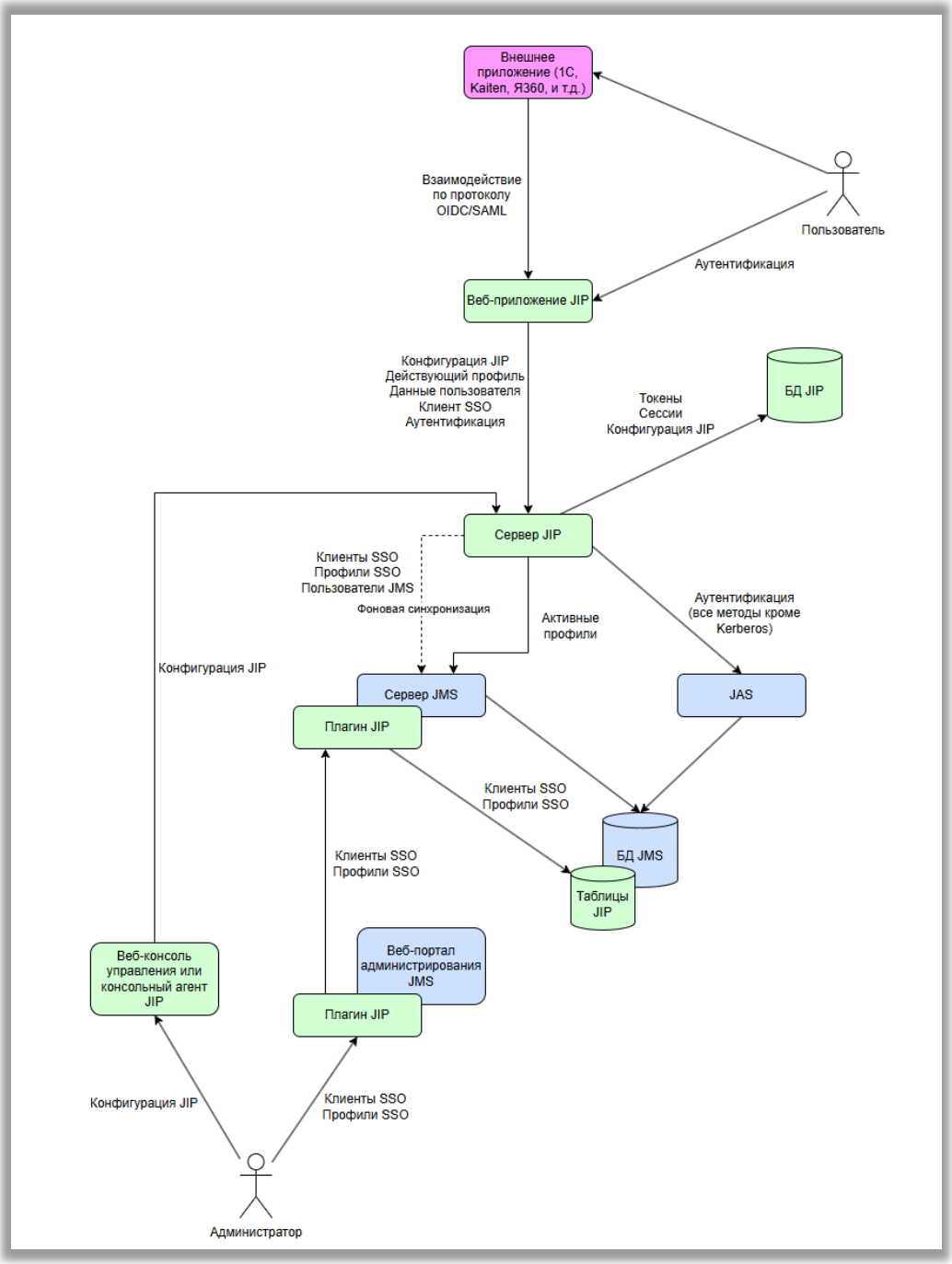

Рис. 1 – Схема взаимодействия JIP с JMS и JAS

7. Развертывание ПО JIP

7.1 Начальные условия для развертывания JIP

Для развертывания ПО JIP должны быть выполнены следующие начальные условия.

1. В сетевой доступности должна быть установлена служба управления учетными записями (сервер ресурсной системы, например AD, FreeIPA и т.п.).
2. В сетевой доступности должен быть развёрнут сервер СУБД, на котором уже функционирует БД связанного сервера JMS (см. руководство по установке и настройке JMS [2]).

7.2 Порядок развертывания ПО JIP

1. Установка плагина для сервера JMS
2. Установка плагина для серверного web-приложения Консоль управления JMS
3. Установка сервера JIP
4. Установка web-консоли сервера JIP
5. Дополнительная настройка сервера JIP средствами web-консоли сервера JIP или консольного агента JIP по необходимости
6. Установка web-приложения JIP
7. Настройка web-приложения JIP путем изменения его файла конфигурации

Примечание. Все развёртывание необходимо производить или с **sudo**, или из-под **root**

8. Установка JIP-плагинов для компонентов JMS

8.1 Установка плагина для сервера JMS

Для установки плагина для сервера JMS выполните следующие действия.

1. Скопируйте с дистрибутивного диска на машину, где установлен сервер JMS, установочный пакет JIP-плагина для сервера JMS согласно Табл. 6, с. 16.
2. Из директории, куда скопирован пакет, выполните следующие команды.

2.1. Команду установки плагина.

2.1.1. Для ОС Astra Linux:

```
dpkg -iE ./<имя файла deb-пакета согласно Табл. 6, с. 16>
```

2.1.2. Для РЕД ОС / ОС Альт:

```
rpm -i ./<имя файла rpm-пакета согласно Табл. 6, с. 16>
```

2.2. Команду инициализации плагина:

```
eap-agent SsoProfile initialize
```

Если инициализация прошла успешно, то в консоль будет выведено соответствующее сообщение.

Помимо регистрации новых команд для API JMS и нового профиля (**Профиль SSO**) плагин добавляет новую роль **Администратор SSO** и соответствующие ей права.

Для работы с клиентами SSO из консоли управления JMS необходимо назначить эту роль пользователю, от лица которого будет производиться управление клиентами SSO.

Примечание. Уже существующим администраторам JMS роль **Администратор SSO** автоматически не назначается. Поэтому перед созданием профилей SSO или назначьте эту роль учетной записи, под которой планируется создавать профили, или создайте новую учетную запись с соответствующими правами для управления SSO и в дальнейшем все операции проводите из-под нее.

В случае использования кластера JMS шаги 1-2 следует повторить для каждой машины, где установлен сервер JMS.

8.2 Установка плагина для серверного web-приложения Консоль управления JMS

Для установки плагина для серверного web-приложения Консоль управления JMS выполните следующие действия.

1. Скопируйте с дистрибутивного диска на машину, где установлено серверное web-приложение Консоль управления JMS, установочный пакет JIP- плагина для данного приложения согласно Табл. 7, с. 16.
2. Из директории, куда скопирован пакет, выполните команду установки плагина:

2.1. Для ОС Astra Linux:

```
dpkg -iE ./<имя файла deb-пакета согласно Табл. 7, с. 16>
```

2.2. Для РЕД ОС / ОС Альт:

```
rpm -i ./<имя файла rpm-пакета согласно Табл. 7, с. 16>
```

3. После установки перезагрузите сервис eap-web-admin следующей командой:

```
systemctl restart eap-web-admin.service
```

Если установка прошла успешно, то после входа в Консоль управления JMS от лица пользователя, включенного в роль **Администратор SSO** или имеющего соответствующие права, можно будет увидеть раздел **SSO** в главном меню, а также новый тип профиля как это показано на Рис. 2. и Рис. 3.

В случае использования кластера JMS шаги 1-3 следует повторить для каждой машины, где установлено серверное web-приложение Консоль управления JMS.

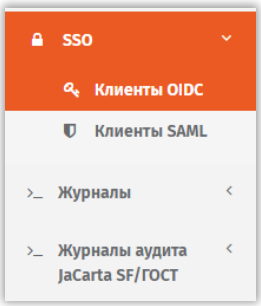

Рис. 2. Новые пункты меню Консоли управления JMS

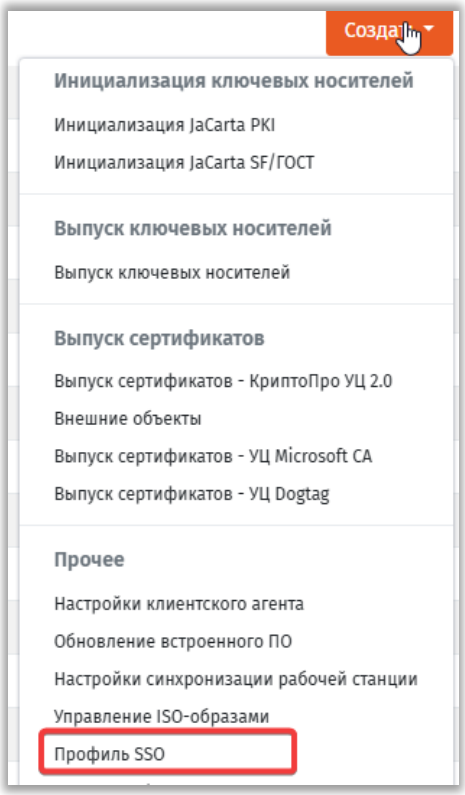

Рис. 3.. Новый тип профиля

9. Установка сервера JIP

Для установки сервера JIP выполните следующие действия.

Примечание. Установку нужно проводить с **sudo** или из-под **root**. Дальнейшая работа сервиса будет возможна только из-под **root**.

1. Подготовьте (создайте или заполните) файл инициализации JIP_*InitialConf.ini*. Параметры файла инициализации описаны в разделе «Настройки сервера JIP», с. 35. Пример файла инициализации приведён в разделе «Приложение 2. Минимальный файл инициализации сервера JIP», с. 126.
2. Скопируйте с дистрибутивного диска на целевую машину с ОС Linux, предназначенную для установки сервера JIP установочный пакет сервера JIP согласно Табл. 3, с. 15. В ту же директорию поместите подготовленный первом шаге файл инициализации.
3. Из директории, где находится файл дистрибутива, выполните следующие команды:

3.1. Команду установки сервера JIP.

3.1.1. Для ОС Astra Linux:

```
dpkg -iE ./<имя файла deb-пакета согласно Табл. 3, с. 15>
```

3.1.2. Для РЕД ОС / ОС Альт:

```
rpm -i ./<имя файла rpm-пакета согласно Табл. 3, с. 15>
```

3.2. Команду инициализации сервера JIP

```
sudo jip-agent server initialize -p ./JIP_InitialConf.ini
```

Если инициализация прошла успешно, то в консоль будет выведено соответствующее сообщение.

Примечание. Перед выполнением команды убедитесь, что значение Subject CN в сертификате используемого для подписи и проверки сообщений соответствующих протоколов такое же как, как в идентификаторе издателя токена в сертификатах для OIDC и SAML.

Например, если значение "Издатель SAML (Issuer)" равно "http://jip.local/", то при выпуске сертификата необходимо задать значение "Subject CN" равное "http://jip.local/". Сделать это можно при помощи правки шаблона сертификата. В имени субъекта следует выбрать: "Предоставляется в запросе". Аналогичным образом следует сделать для OIDC. Без прохождения этой проверки инициализация JIP не будет проходить.

4. Для проверки состояния сервера JIP выполните команду:

```
sudo jip-agent server status
```

10.Установка web-консоли сервера JIP

Для установки web-консоли сервера JIP выполните следующие действия

Примечание. Установку нужно проводить с **sudo** или из-под **root**. Дальнейшая работа сервиса будет возможна только из-под **root**.

1. Скопируйте с дистрибутивного диска на машину, где должна быть установлена web-консоль сервера JIP, установочный пакет для web-консоли согласно Табл. 4, с. 16.
2. Из директории, куда был скопирован пакет, выполните команду установки

2.1. Для ОС Astra Linux:

```
dpkg -iE ./<имя файла deb-пакета согласно Табл. 4, с. 16>
```

2.2. Для РЕД ОС / ОС Альт:

```
rpm -i ./<имя файла rpm-пакета согласно Табл. 4, с. 16>
```

3. Для проверки работоспособность консоли выполите команду:

```
sudo systemctl status aladdin-jip-wsc
```

Если установка прошла успешно, то в сообщении будет содержаться строка “Active: active (running)”.

4. После установки укажите параметры подключения к серверу JIP, отредактировав файл конфигурации консоли **/etc/aladdin/jip-wsc/appsettings.json**. Для этого в параметрах "ControlApiUrl" и "HealthCheckApiUrl" укажите адреса подключения к API сервера JIP:

```
  "ServerConnectionSettings": {
    "ControlApiUrl": "http://localhost:28103",
    "HealthCheckApiUrl": "http://localhost:28105",
    "PingTimeout": 5
},
```

5. При необходимости в том же фале укажите адрес, по которому будет доступна сама web-консоль сервера JIP:

```
  "Kestrel": {
    "Endpoints": {
      "Http": {
        "Url": "http://0.0.0.0:5030"
```

```
        }
      }
    },
```

6. После изменения настроек необходимо перезагрузите сервис следующей командой:

```
sudo systemctl restart aladdin-jip-wsc
```

После применения настроек web-консоль сервера JIP будет доступна по указанному адресу. Для входа потребуются логин и пароль, указанные для API управления при развертывании сервера JIP.

11.Синхронизация данных из JMS в JIP

Помимо настроек подключения JIP к JMS ключевую роль играет настройка синхронизации данных из JMS в JIP. Синхронизация данных из JMS запускается через API управления, либо через консольный агент JIP (который также использует API управления). Синхронизация разделена на две части:

1. Метаданные – профили SSO и клиенты SSO
2. Пользователи – пользователи JMS

Синхронизация метаданных является относительно быстрой и может выполняться, например, каждую минуту. В свою очередь синхронизация пользователей напрямую зависит от их количества и может идти продолжительное время особенно при значениях числа пользователей более 100000. В связи с этим синхронизацию пользователей рекомендуется производить в нерабочее время.

Одновременно может быть запущена только одна синхронизация, пока она не завершится другие попытки запуска будут завершаться с ошибкой.

Примечание. По умолчанию после установки синхронизация выключена. Синхронизацию обязательно нужно включить на одной из нод по инструкции из 11.2 Синхронизация данных с помощью консольного агент.

11.1 Требования к пользователю JMS

Для обеспечения корректного подключения JIP к JMS необходимо создать специального пользователя и назначить ему необходимые права. Это делается через отдельную роль, к примеру, «Администратор JIP», которая должна включать следующие права доступа:

Блок «*Обслуживание сервера*»

- Чтение конфигурации сервера
- Чтение из каталога учетных записей
- Чтение контейнера ресурсной системы

Блок «*Пользователи*»

- Чтение

Блок «*Профили*»

- Чтение типов профилей
- Чтение экземпляров профилей

Блок « *SSO: Клиенты OIDC* »

- Получение списка клиентов OIDC
- Получение клиента OIDC по идентификатору

Блок «*SSO: Клиенты SAML*»

- Получение списка клиентов SAML
- Получение клиента SAML по идентификатору

После создания роли «Администратор JIP», необходимо включить в нее пользователя, который будет использоваться для подключения к JMS. После необходимо прописать его данные в *Web-консоли сервера JIP* в разделе настроек JMS. Проверить корректность введенных данных можно кнопкой «Проверить соединение»

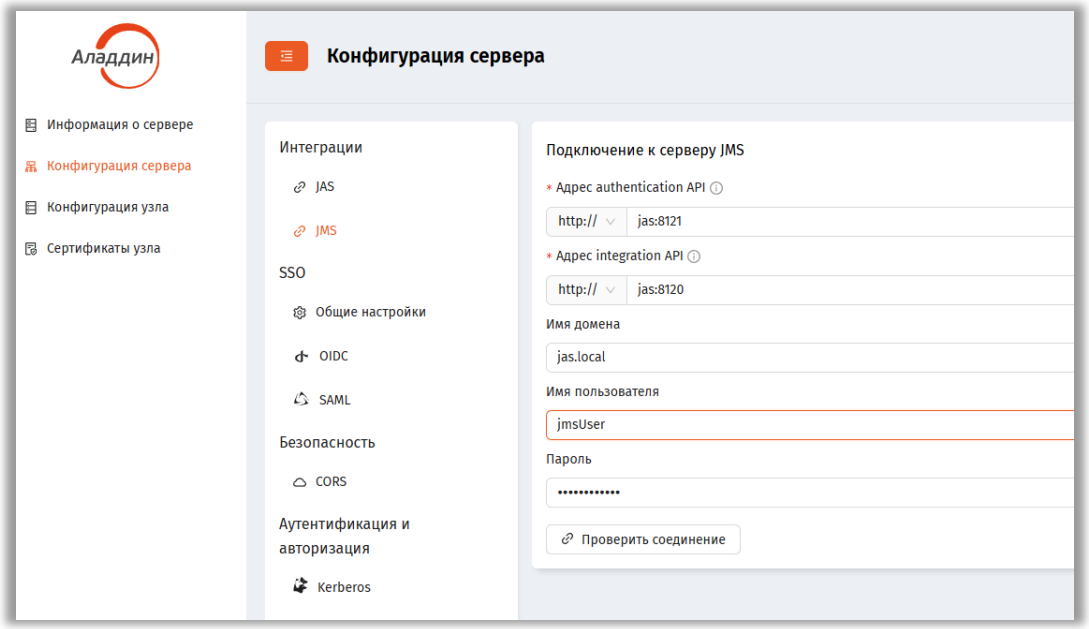

Рис. 4. Данные подключения к серверу JMS

11.2 Синхронизация данных с помощью консольного агента

Синхронизация данных из JMS в JIP представляет собой длящийся процесс, который может быть запущен или остановлен с помощью команд консольного агента. По умолчанию синхронизация данных выключена.

Примечание. Существует также возможность синхронизации данных напрямую, вызывая методы API, см. раздел «Синхронизация через API», ниже).

Выполнение команд консольного агента по синхронизации можно выполнить как в ручном режиме, так и автоматически.

Для управления синхронизацией вручную используются следующие команды.

1. Команда запуска синхронизации:

```
sudo jip-agent sync start --scope <область_синхронизации>
```

где параметр `<область_синхронизации>` может принимать значение:

users - пользователи,
metadata – клиенты SSO и профили SSO).

2. Команда остановки синхронизации:

```
sudo jip-agent sync stop
```

3. Команда отключения поддержки синхронизации (в некоторых отладочных сценариях может потребоваться временное отключение синхронизации):

```
sudo jip-agent sync disable
```

4. Команда включения поддержки синхронизации

```
sudo jip-agent sync enable
```

5. Команда отображения статуса последней синхронизации:

```
sudo jip-agent sync status
```

Для автоматизации процесса синхронизации рекомендуется использовать утилиту **cron**.

Примечание. В случае использования кластерного сценария развёртывания, синхронизация должна быть включена только на одной ноде кластера.

Для настройки необходимо открыть конфигурацию cron:

```
crontab -e
```

Затем вставить в неё фрагмент:

```
# Синхронизация профилей SSO и клиентов SSO каждую минуту с 02:00 до 23:59 (кроме
00:00-02:00 чтобы успели синхронизироваться пользователи)
* 2-23 * * * jip-agent sync start -s metadata
# Синхронизация пользователей один раз в день в 00:10
10 0 * * * jip-agent sync start -s users
```

11.3 Синхронизация через API

Для API управления (Control API) доступен Swagger UI API управления сервера JIP. Например, по адресу:

http://localhost:28103/swagger/index.html

В разделе Sync описаны методы управления синхронизацией данных из JMS в JIP, также можно проверить их работу.

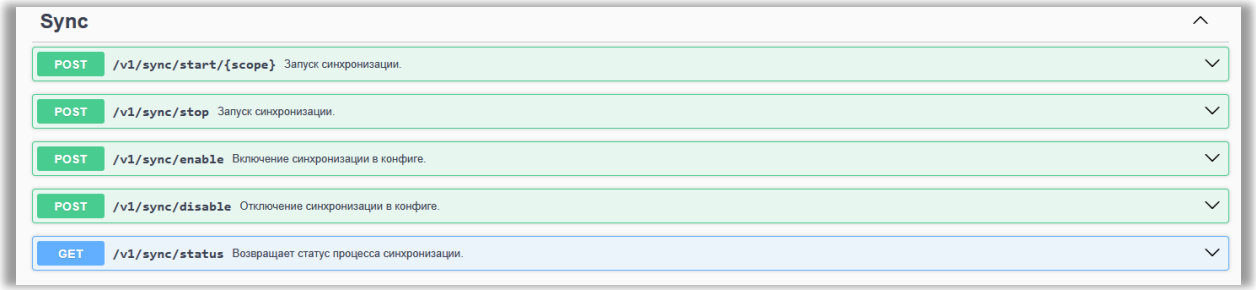

Рис. 5. Swagger UI API управления JIP

Возможна автоматизация запуска этих методов сторонними средствами.

12.Выбор приоритета OTP для JAS

Помимо настроек подключения JIP к JAS ключевую роль играет настройка 18.3.7 Поддерживаемые типы аутентификации. Она доступна как при инициализации сервера JIP, так и после в консольном агенте или web-консоли сервера JIP. Важно корректно указать приоритет и состав доступных типов аутентификации JAS, так как Профиль SSO поддерживает лишь указание общего типа аутентификации “OTP” в свойстве 20.1.4 Стратегии входа. При этом доступность и приоритет конкретных типов OTP определяется в рамках данной настройки глобально.

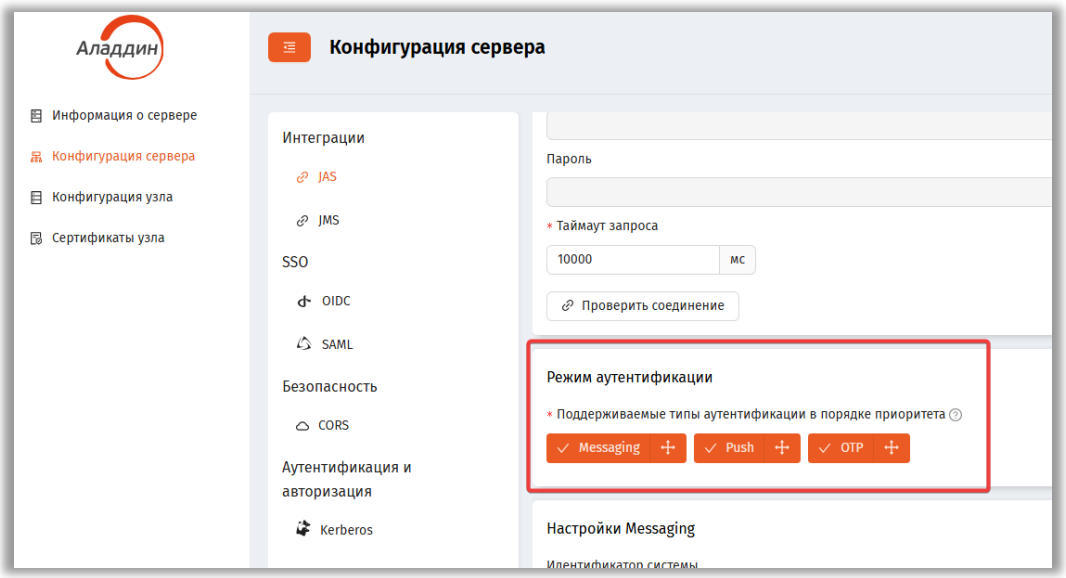

Рис. 6. Выбор приоритета типов OTP в web-консоли сервера JIP

13.Установка серверного web-приложения JIP

Web-приложение JIP может быть установлено как на одном хосте с сервером JIP, так и на разных.

13.1 Установка web-приложения JIP на одном хосте с сервером JIP

Для установки web-приложения JIP выполните следующие действия.

Примечание. Установку нужно проводить с **sudo** или из-под **root**. Дальнейшая работа сервиса будет возможна только из-под **root**.

1. Скопируйте с дистрибутивного диска на машину, где где должно быть установлено серверное web-приложение JIP, установочный пакет для данного приложения согласно Табл. 5, с. 16.
2. Из директории, где находится файл дистрибутива, выполните команду установки web-приложения JIP:

2.1. Для ОС Astra Linux:

```
dpkg -iE ./<имя файла deb-пакета согласно Табл. 5, с. 16>
```

2.2. Для РЕД ОС / ОС Альт:

```
rpm -i ./<имя файла rpm-пакета согласно Табл. 5, с. 16>
```

3. Откройте на редактирование файл конфигурации web-приложения JIP **/etc/aladdin/jip-web/appsettings.json** и выполните следующие настройки:

3.1. Укажите внешний адрес по которому будет доступно для пользователей web-приложение JIP:

```
"Kestrel": {
    "Endpoints": {
      "Http": {
        "Url": "http://*:28100"
      }
    }
  },
```

3.2. Укажите адрес подключения к административному API сервера JIP (по этому адресу web-приложение получает свои основные настройки). Для этого отредактируйте следующую секцию:

```
"WebHost": {
    "ApiBaseUrl": "http://localhost:28102",
    "ApiUsername": "username",
    "ApiPassword": "password",
    "ApiTimeout": "00:01:00",
    "CheckVersionPeriod": "00:01:00"
  }
```

3.3. В параметрах «ApiUsername» и «ApiPassword» укажите логин и пароль, который был задан при инициализации сервера JIP. Подробности в 18.1.7 Имя пользователя API управления и 18.1.8 Пароль пользователя API управления. Параметр CheckVersionTimeout отвечает за таймаут вызова проверки версии настроек для web-приложения JIP и в общем случае не требует изменения.

4. После внесения изменений перезагрузить сервис командой:

```
systemctl restart aladdin-jip-web
```

5. Для проверки работоспособности web-приложения JIP выполните команду:

```
sudo systemctl status aladdin-jip-web
```

Если установка прошла успешно, то в сообщении будет содержаться строка “Active: active (running)”.

13.2 Установка web-приложения JIP на отдельном хосте от сервера JIP

Если для вашей инфраструктуры нужно, чтобы сервер JIP и web-приложение JIP находились на разных компьютерах в рамках локальной сети, выполните следующие действия.

1. Убедитесь, что в конфигурации сервера JIP установлен адрес отличный от localhost.
2. В конфигурационном файле web-приложение JIP укажите параметры подключения к серверу JIP по его сетевому адресу.
3. При помощи утилиты *Dotnet Certificate Tool* (см. https://github.com/stylianosnicoletti/dotnet-certificate-tool) установите в локальное хранилище машины, на которой установлено web-приложение JIP, сертификаты, которые были указаны в настройках сервера JIP для использования в OIDC/SAML.

Ниже приведена краткая инструкция по работе с утилитой *Dotnet Certificate Tool*:

1. Установите утилиту при помощи команды.

```
dotnet tool install --global dotnet-certificate-tool
```

В системе должен быть установлен .NET. Подробности и способы установки пакета описаны в официальной документации его поставщика (см. https://learn.microsoft.com/ru-ru/dotnet/core/install/linux?WT.mc_id=dotnet-35129-website)

2. Для установки сертификата, в зависимости от его формата, используйте одну из команд:

2.1. При использовании PFX файла

```
certificate-tool add --file path/to/cert.pfx --password <ВашПароль>
```

2.2. При использовании pem-файла и ключа

```
certificate-tool add --cert path/to/cert.pem --key path/to/key.pem --password <ВашПароль>
```

2.3. При использовании строки в формате Base64

```
certificate-tool add --base64 "BASE64_STRING" --password <ВашПароль>
```

3. Для удаления сертификата по его отпечатку выполните команду:

```
certificate-tool remove --thumbprint ОтпечатокСертификата
```

14.Настройка журналирования событий аутентификации

Журналирование событий аутентификации настраивается через стандартную конфигурацию логирования Serilog в файле конфигурации web-приложения JIP **/etc/aladdin/jip-web/appsettings.json.**

Основные настройки журналирования:

- События аудита записываются в отдельный файл в формате JSON.
- Путь: /var/log/aladdin/jip-web/audit/auth-audit.log
- Формат: JSON (каждое событие - отдельная строка)
- Ротация: по размеру файла (10 МБ)
- Хранение: 31 последний файл

Дополнительно события дублируются в общий лог /var/log/aladdin/jip-web/Aladdin.JIP.Web.log и консоль.

Пример конфигурации (фрагмент appsettings.json web-приложения JIP):

```
  "Serilog": {
    "WriteTo": [
      {
        "Name": "Logger",
        "Args": {
          "configureLogger": {
            "Filter": [
              {
                "Name": "ByIncludingOnly",
                "Args": {
                  "expression": "StartsWith(SourceContext, 'Aladdin.JIP.Web.Audit')"
                }
              }
            ],
            "WriteTo": [
              {
                "Name": "File",
                "Args": {
                  "path": "/var/log/aladdin/jip-web/audit/auth-audit.log",
                  "formatter": "Serilog.Formatting.Json.JsonFormatter, Serilog",
                  "rollOnFileSizeLimit": true,
```

```
                    "fileSizeLimitBytes": "10485760",
                    "retainedFileCountLimit": 31,
                    "buffered": false
                }
            }
        ]
    }
  }
 }
]
}
```

15.Обновление ПО JIP

Порядок обновления:

1. Обновление плагина для сервера JMS
2. Обновление плагина для серверного web-приложения Консоль управления JMS
3. Обновление сервера JIP
4. Обновление web-консоли сервера JIP
5. Обновление web-приложения JIP

15.1 Обновление JIP-плагина для сервера JMS

Для обновления плагина для сервера JMS выполните следующие действия.

1. Скопируйте на машину, где установлен сервер JMS, обновлённые установочный пакет JIP-плагина для сервера JMS согласно Табл. 6, с. 16.
2. Из директории, куда скопирован пакет, выполните следующие команды.

2.1. Команду обновления плагина.

2.1.1. Для ОС Astra Linux:

```
dpkg -iE ./<имя файла deb-пакета согласно Табл. 6, с. 16>
```

2.1.2. Для РЕД ОС / ОС Альт:

```
rpm -Uvh ./ <имя файла rpm-пакета согласно Табл. 6, с. 16>
```

При этом предыдущая версия будет обновлена до новой.

3. После установки выполните инициализацию:

```
eap-agent SsoProfile initialize
```

Проверить корректность обновления - в консоли должно появиться сообщение об успешной инициализации.

При необходимости убедитесь, что роль «Администратор SSO» и права для пользователя остались корректными.

Повторите действия для каждой машины, где установлен сервер JMS (в случае кластера).

15.2 Обновление JIP-плагина для серверного web-приложения Консоль управления JMS

Для обновления плагина для серверного web-приложения Консоль управления JMS выполните следующие действия.

1. Скопируйте на машину, где установлено серверное web-приложение Консоль управления JMS, обновлённый установочный пакет JIP- плагина для данного приложения согласно Табл. 7, с. 16.
2. Из директории, куда скопирован пакет, выполните команду обновления плагина:

2.1. Для ОС Astra Linux:

```
dpkg -iE ./<имя файла deb-пакета согласно Табл. 7, с. 16>
```

2.2. Для РЕД ОС / ОС Альт:

```
rpm - Uvh ./<имя файла rpm-пакета согласно Табл. 7, с. 16>
```

После обновления зайдите в Консоль управления JMS под учетной записью с ролью «Администратор SSO» и убедитесь, что раздел SSO и новый тип профиля отображаются корректно.

Повторите данную процедуру для каждой машины с серверным web-приложением Консоль управления JMS в кластере.

15.3 Обновление сервера JIP

Для обновления сервера JIP выполните следующие действия.

1. Скопируйте машину с сервером JIP обновлённый установочный пакет сервера JIP согласно Табл. 3, с. 15..
2. Из директории, куда скопирован пакет, выполните следующие команды:

2.1. Команду обновления сервера JIP.

2.1.1. Для ОС Astra Linux:

```
dpkg -iE ./<имя файла deb-пакета согласно Табл. 3, с. 15>
```

2.1.2. Для РЕД ОС / ОС Альт:

```
rpm - Uvh ./<имя файла rpm-пакета согласно Табл. 3, с. 15>
```

3. Обновите версию БД до актуальной следующей командой:

```
sudo jip-agent server update
```

4. Проверьте состояние:

```
sudo jip-agent server status
```

В сообщении должно быть: «Текущее состояние сервера: Работает».

15.4 Обновление Web-консоли сервера JIP

Для обновления web-консоли сервера JIP выполните следующие действия

1. Скопируйте на машину с web-консолью сервера JIP её обновлённый установочный пакет согласно Табл. 4, с. 16.
2. Из директории, куда был скопирован пакет, выполните команду обновления.

2.1. Для ОС Astra Linux:

```
dpkg -iE ./<имя файла deb-пакета согласно Табл. 4, с. 16>
```

2.2. Для РЕД ОС / ОС Альт:

```
rpm - Uvh ./<имя файла rpm-пакета согласно Табл. 4, с. 16>
```

3. Проверьте состояние командой:

```
systemctl status aladdin-jip-wsc
```

Если обновление прошло успешно, то в сообщении будет содержаться строка “Active: active (running)”.

При необходимости обновите параметры подключения к серверу JIP в файле **/etc/aladdin/jip-wsc/appsettings.json** и перезапустить сервис:

```
systemctl restart aladdin-jip-wsc
```

15.5 Обновление серверного web-приложения JIP

Для обновления web-приложения JIP выполните следующие действия.

1. Скопируйте серверным web-приложением JIP обновлённый установочный пакет для данного приложения согласно Табл. 5, с. 16.
2. Из директории, где находится файл дистрибутива, выполните команду обновления серверного web-приложения JIP:

2.1. Для ОС Astra Linux:

```
dpkg -iE ./<имя файла deb-пакета согласно Табл. 5, с. 16>
```

2.2. Для РЕД ОС / ОС Альт:

```
rpm - Uvh ./<имя файла rpm-пакета согласно Табл. 5, с. 16>
```

3. Проверьте состояние командой:

```
systemctl status aladdin-jip-web
```

При необходимости отредактируйте файл **/etc/aladdin/jip-web/appsettings.json** и перезапустите сервис командой:

```
systemctl restart aladdin-jip-web
```

16.Удаление ПО JIP

Порядок удаления:

1. Удаление плагина для сервера JMS
2. Удаление плагина для серверного web-приложения Консоль управления JMS
3. Удаление web-консоли сервера JIP
4. Удаление web-приложения JIP
5. Удаление сервера JIP

16.1 Удаление JIP-плагина для сервера JMS

1. Остановите сервер JMS следующей командой:

```
systemctl stop eap-engine
```

2. Для удаления плагина выполните следующую команду.

2.1. Для ОС Astra Linux:

```
dpkg -r aladdin-jip-eap-engine-plugin
```

2.2. Для РЕД ОС / ОС Альт:

```
rpm -e aladdin-jip-eap-engine-plugin
```

3. При необходимости удалить остаточные конфигурации:

```
dpkg --purge aladdin-jip-eap-engine-plugin
```

4. Запустите сервер JMS:

```
systemctl start eap-engine
```

16.2 Удаление JIP-плагина для серверного web-приложения Консоль управления JMS

1. Остановите серверное web-приложение Консоль управления JMS следующей командой:

```
systemctl stop eap-web-server-console
```

2. Для удаления плагина выполните следующую команду.

2.1. Для ОС Astra Linux:

```
dpkg -r aladdin-jip-eap-web-admin-plugin
```

2.2. Для РЕД ОС / ОС Альт:

```
rpm -e aladdin-jip-eap-web-admin-plugin
```

3. Для полного удаления конфигурационных файлов выполните команду:

```
dpkg --purge aladdin-jip-eap-web-admin-plugin
```

4. Запустите серверное web-приложение Консоль управления JMS следующей командой:

```
systemctl start eap-web-server-console
```

16.3 Удаление web-консоли сервера JIP

1. Остановите web-консоль следующей командой:

```
systemctl stop aladdin-jip-wsc
```

2. Для удаления пакета выполните следующую команду.

2.1. Для ОС Astra Linux:

```
dpkg -r aladdin-jip-wsc
```

2.2. Для РЕД ОС / ОС Альт:

```
rpm -e aladdin-jip-wsc
```

3. При необходимости удалите конфигурацию:

```
rm -rf /etc/aladdin/jip-wsc
```

16.4 Удаление серверного web-приложения JIP

1. Остановите сервис web-приложения JIP следующей командой:

```
systemctl stop aladdin-jip-web
```

2. Для удаления пакета выполните следующую команду.

2.1. Для ОС Astra Linux:

```
dpkg -r aladdin-jip-web
```

2.2. Для РЕД ОС / ОС Альт:

```
rpm -e aladdin-jip-web
```

3. При необходимости удалите конфигурацию:

```
rm -rf /etc/aladdin/jip-web
```

16.5 Удаление сервера JIP

1. Остановите сервер JIP следующей командой:

```
systemctl stop aladdin-jip-engine
```

2. Для удаления пакета выполните следующую команду.

2.1. Для ОС Astra Linux:

```
dpkg -r aladdin-jip-engine
```

2.2. Для РЕД ОС / ОС Альт:

```
rpm -e aladdin-jip-engine
```

3. При необходимости удалите конфигурационные и временные файлы:

```
rm -rf /etc/aladdin/jip-engine
rm -rf /var/log/aladdin/jip-engine
```

17.Изменение настроек JIP

Изменение настроек развернутого сервера JIP и web-приложения JIP доступно двумя способами:

1. Через консольный агент
2. Через web-консоль сервера JIP

Доступные для изменения настройки описаны в разделе 18 Настройки сервера JIP, там же приведены примеры команд для изменения настроек с помощью консольного агента.

Web-приложение JIP не требует отдельного конфигурирования (помимо настроек, указанных при развертывании). Web-приложение JIP получает свою конфигурацию из сервера JIP и автоматически перезапускает свои сервисы для применения изменения настроек. В связи с чем некоторые настройки могут потребовать нескольких секунд для окончательного применения.

17.1 Через консольный агент

Консольный агент поддерживает команды отображения и изменения настроек JIP. Для получения информации о доступных к изменению настройках необходимо вызвать команду:

sudo jip-agent help

Консольный агент выведет все доступные команды:

```
server          Операции над сервером.
certificates    Операции с сертификатами.
sso             Общие настройки SSO.
oidc            Настройка OIDC.
saml            Настройки SAML.
jas             Настройка подключения к JAS.
jms             Настройка подключения к JMS.
kerberos        Настройки Kerberos.
cors            Настройка CORS.
ssl             Настройки SSL.
sync            Управление синхронизацией данных из JMS в JIP
syslog          Настройки Syslog-уведомлений.
journaling      Настройки журналирования
help            Отобразить подробную справку о конкретной команде.
version         Отобразить версию продукта.
```

Рис. 7. Список доступных команд консольного агента

Также можно вызвать **help** для любой из этих команд:

sudo jip-agent server help

```
initialize    Инициализировать\переинициализировать сервер.
start         Запустить сервер.
stop          Остановить работу сервера.
pause         Приостановить работу сервера.
continue      Запустить сервер, если он был приостановлен.
status        Получить текущий статус сервера.
autostart     Управление настройкой автостарта сервера.
update        Выполнить обновление дополнительных пакетов и базы данных.
help          Отобразить подробную справку о конкретной команде.
version       Отобразить версию продукта.
```

Рис. 8. Помощь по разделу "Операции над сервером"

После чего можно вызвать готовую команду, например:

sudo jip-agent server status

17.2 Через web-консоль сервера JIP

Почти все доступные в консольном агенте настройки доступны и в web-консоли сервера JIP. Для входа в неё необходимо перейти по адресу, указанному при развертывании web-консоли сервера JIP.

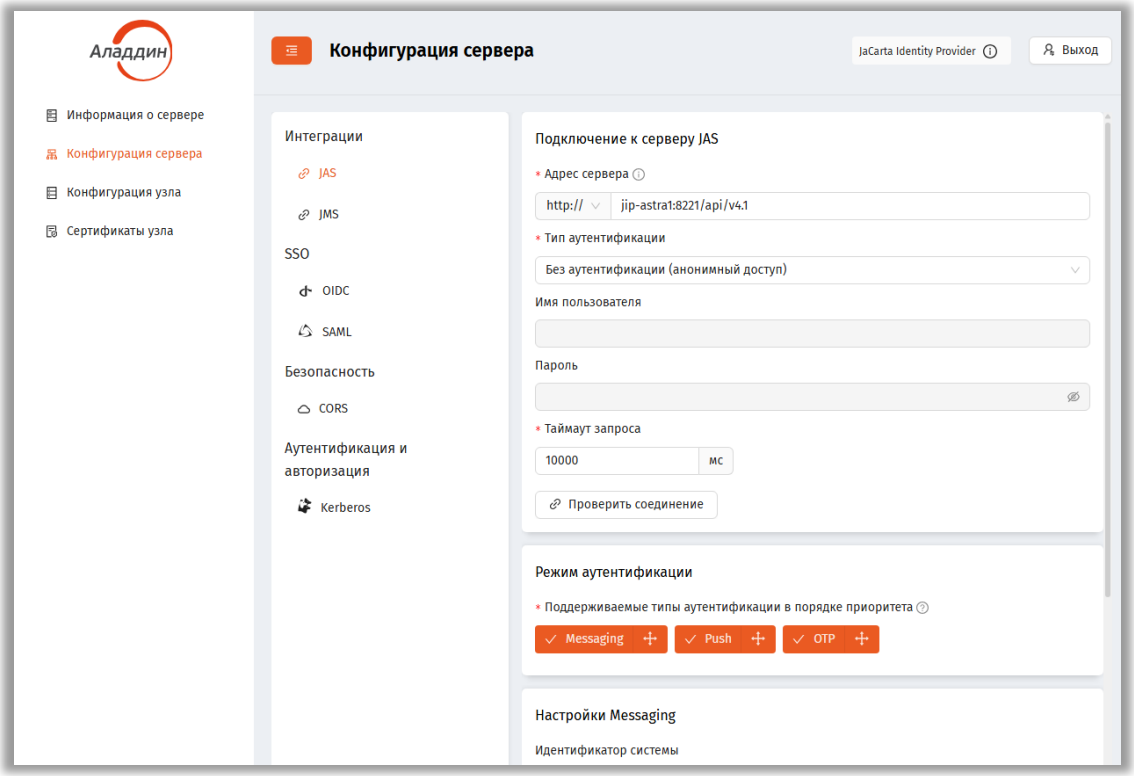

Рис. 9. Web-консоль сервера JIP

Все изменения настроек вступают в силу автоматически после сохранения без необходимости ручной перезагрузки сервисов. Исключение – настройка HTTPS.

18.Настройки сервера JIP

В данном разделе перечислены доступные настройки сервера JIP и web-приложения JIP. Часть настроек доступна к изменению только в процессе развертывания JIP. Способы изменения настроек описаны в разделе 15.

18.1 Основные настройки сервиса

Общие настройки сервиса: пути, URL сервисов, язык интерфейса и учетные данные для авторизации консольного агента и web-консоли сервера JIP в API сервера JIP. Настройки отсутствуют в параметрах консольного агента и web-консоли сервера JIP. Изменение возможно только путем инициализации через ini-файл.

18.1.1 Путь до исполняемого файла

Определяет путь к исполняемому файлу сервера JIP.

Параметр при инициализации: [service] -> execPath
Обязательность при инициализации: Обязателен

18.1.2 Адреса API управления

Адреса API управления сервера JIP. По данным адресам будет также доступен Swagger UI данного API, для доступа к нему необходимо перейти по указанному адресу API управления добавив путь **/swagger**.

Допустимые значения: Можно задать несколько адресов через «;»
Параметр при инициализации: [service] -> controlServiceUrls
Обязательность при инициализации: Обязателен

18.1.3 Адреса API администрирования

Адреса API администрирования сервера JIP. По данным адресам будет также доступен Swagger UI данного API, для доступа к нему необходимо перейти по указанному адресу API управления добавив путь **/swagger**.

Допустимые значения: Можно задать несколько адресов через «;»
Параметр при инициализации: [service] -> administrationServiceUrls
Обязательность при инициализации: Обязателен

18.1.4 Адреса API проверки работоспособности

Адреса API работоспособности (хелсчек) сервера JIP. По данным адресам будет также доступен Swagger UI данного API, для доступа к нему необходимо перейти по указанному адресу API работоспособности добавив путь **/swagger**.

Допустимые значения: Можно задать несколько адресов через «;»

Параметр при инициализации: [service] -> healthcheckUrls

Обязательность при инициализации: Обязателен

18.1.5 Язык интерфейса

Язык, который будет использован при инициализации.

Допустимые значения: en, ru
Параметр при инициализации: [service] -> culture
Обязательность при инициализации: Необязателен
Значение по умолчанию при инициализации: ru

18.1.6 Автоматический старт

Запускать ли сервисы после запуска сервера JIP. Если установлен в false – сервер запустится со статусом «Остановлен» и потребуется его принудительный запуск, путем выполнения команды консольного агента JIP:
sudo jip-agent server start

Допустимые значения: true, false
Параметр при инициализации: [service] -> autoStart
Обязательность при инициализации: Необязателен
Значение по умолчанию при инициализации: true

18.1.7 Имя пользователя API управления

Имя пользователя для подключения к API управления сервера JIP. Произвольное имя, никак не связанное с наличием реального пользователя. Это не доменный и не локальный пользователь ОС. Он определяется только в рамках настроек сервера JIP.

Параметр при инициализации: [controlPrimaryUser] -> username
Обязательность при инициализации: Обязателен

18.1.8 Пароль пользователя API управления

Пароль пользователя для подключения к API управления сервера JIP.

Параметр при инициализации: [controlPrimaryUser] -> password
Обязательность при инициализации: Обязателен

18.1.9 Имя пользователя API администрирования

Имя пользователя для подключения к административному API сервера JIP. Произвольное имя, никак не связанное с наличием реального пользователя. Это не доменный и не локальный пользователь ОС. Он определяется только в рамках настроек сервера JIP.

Параметр при инициализации: [administrationPrimaryUser] -> username
Обязательность при инициализации: Обязателен

18.1.10 Пароль администратора Administration Api

Пароль пользователя для подключения к административному API сервера JIP.

Параметр при инициализации: [administrationPrimaryUser] -> password
Обязательность при инициализации: Обязателен

18.2 База данных

Параметры подключения к базе данных. Настройки отсутствуют в параметрах консольного агента и web-консоли сервера JIP. Изменение возможно только путем инициализации через ini-файл.

18.2.1 Тип СУБД

Тип используемой СУБД.

Допустимые значения: PostgreSQL, MSSQL, JatobaSQL

Параметр при инициализации: [database] -> type

Обязательность при инициализации: Обязателен

18.2.2 Адрес сервера

Адрес сервера базы данных.

Параметр при инициализации: [database] -> serverAddress
Обязательность при инициализации: Обязателен

18.2.3 Порт сервера

Порт подключения к СУБД.

Допустимые значения: Число
Параметр при инициализации: [database] -> serverPort
Обязательность при инициализации: Обязателен

18.2.4 Имя базы данных

Имя создаваемой БД или имя БД, к которой необходимо подключиться, в зависимости от сценария развертывания.

Параметр при инициализации: [database] -> databaseName
Обязательность при инициализации: Обязателен

18.2.5 Режим аутентификации для мастера развертывания

Режим аутентификации для мастера развертывания новой базы данных на сервере.

Допустимые значения: password
Параметр при инициализации: [database] -> serverLoginType
Обязательность при инициализации: Обязателен
Значение по умолчанию при инициализации: password

18.2.6 Имя пользователя для создания БД

Имя пользователя, которое будет использоваться мастером развертывания для создания БД.

Параметр при инициализации: [database] -> serverLogin
Обязательность при инициализации: Обязателен

18.2.7 Пароль пользователя сервера

Пароль пользователя для мастера развертывания.

Параметр при инициализации: [database] -> serverPassword
Обязательность при инициализации: Обязателен

18.2.8 Режим аутентификации для подключения к БД

Режим аутентификации для подключения к БД и дальнейшей работы с ней.

Допустимые значения: password
Параметр при инициализации: [database] -> serverLoginType
Обязательность при инициализации: Обязателен
Значение по умолчанию при инициализации: password

18.2.9 Имя пользователя БД

Логин пользователя, который будет использоваться сервером JIP для доступа с создаваемой БД.

Параметр при инициализации: [database] -> databaseLogin
Обязательность при инициализации: Обязателен

18.2.10 Пароль пользователя БД

Пароль пользователя, который будет использоваться сервером JIP для доступа с создаваемой БД.

Параметр при инициализации: [database] -> databasePassword
Обязательность при инициализации: Обязателен

18.3 JAS

Параметры JAS используются для настройки подключения к сервису JaCarta Authentication Server. Параметры доступны как в консольном агенте JIP, так и в web-консоли сервера JIP.

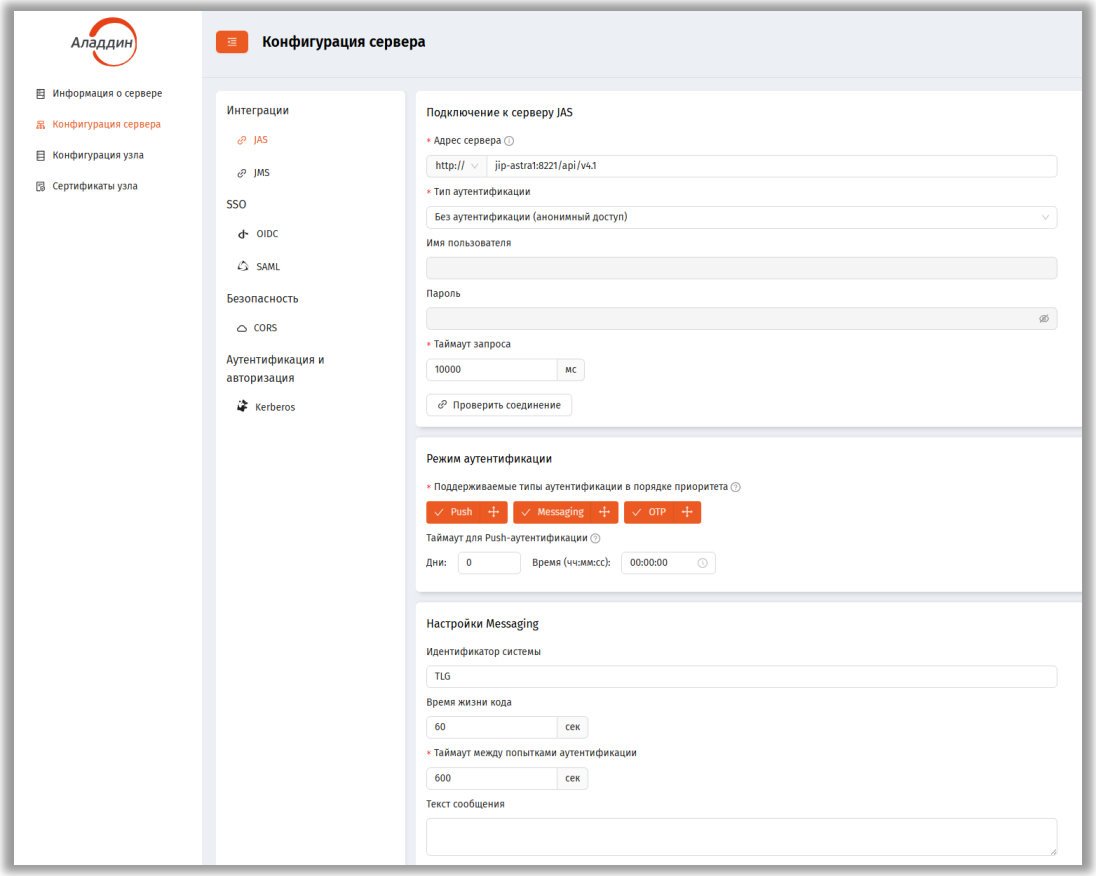

Рис. 10. Раздел JAS web-консоли сервера JIP

18.3.1 Адрес сервера

URL сервера JAS.

Обязательность: Обязателен
Допустимые значения: url
Параметр для команды консольного агента: --url
Параметр при инициализации: [jas] -> url
Обязательность при инициализации: Обязателен

18.3.2 Тип аутентификации

Способ аутентификации в jas

Обязательность: Обязателен
Допустимые значения: None, Basic
Параметр для команды консольного агента: --securityType
Параметр при инициализации: [jas] -> securityType
Обязательность при инициализации: Необязателен
Значение по умолчанию при инициализации: Basic

18.3.3 Имя пользователя

Имя пользователя для доступа к API JAS. Локальный пользователь JAS, который указан в конфигурационном файле, расположенном на компьютере с установленным JAS по пути */etc/aladdin/jas-engine/appsettings.json* в разделе *AuthenticationServiceWebApi*. Если в настройках JAS установлен беспарольный доступ (SecurityType установлено в None и в конфигурационном файле отсутствуют логин/пароль) – поле нужно оставить пустым

Обязательность: Обязателен
Параметр для команды консольного агента: --login
Параметр при инициализации: [jas] -> username
Обязательность при инициализации: Обязателен, если в типе аутентификации выставлен «Basic»

18.3.4 Пароль

Пароль пользователя для доступа к API JAS. Если в настройках JAS установлен беспарольный доступ (SecurityType установлено в None и в конфигурационном файле отсутствуют логин/пароль) – поле нужно оставить пустым

Обязательность: Обязателен
Параметр для команды консольного агента: --password
Параметр при инициализации: [jas] -> password
Обязательность при инициализации: Обязателен, если в типе аутентификации выставлен «Basic»

18.3.5 Таймаут запроса

Максимальное время ожидания ответа от JAS.

Обязательность: Необязателен
Допустимые значения: TimeSpan. Строка формата d.hh:mm:ss.fffffff, где d – дни, hh – часы, mm – минуты, ss – секунды, fffffff – дробная часть секунд.
Параметр для команды консольного агента: --timeout
Параметр при инициализации: [jas] -> timeout
Обязательность при инициализации: Необязателен
Значение по умолчанию при инициализации: 30 секунд

18.3.6 Домен по-умолчанию

Используется для поиска пользователя в ресурсной системе, если пользователь ввел логин без указания домена.

Обязательность: Необязателен

Параметр для команды консольного агента: --defaultDomain

Параметр при инициализации: [jas] -> defaultDomain

Обязательность при инициализации: Необязателен

18.3.7 Поддерживаемые типы аутентификации

Приоритет типов ОТП при вызове авторизации через JAS. Список через запятую. Элементы могут отсутствовать, т.е. “Push” является корректным значением, в этом случае только данный метод будет доступен для аутентификации.

Обязательность: Обязателен
Допустимые значения: Otp, Push и/или Messaging
Параметр для команды консольного агента: --authTypes
Параметр при инициализации: [jas] -> authTypes
Обязательность при инициализации: Обязателен

18.3.8 Таймаут для Push-аутентификации

Максимальное время ожидания реакции на Push-аутентификацию от пользователя. Следует убедиться, что установлено достаточное время, так как подтверждение запроса происходит на мобильном устройстве.

Обязательность: Необязателен
Допустимые значения: TimeSpan. Строка формата d.hh:mm:ss.fffffff, где d – дни, hh – часы, mm – минуты, ss – секунды, ffffff – дробная часть секунд.
Параметр для команды консольного агента: --pushTimeout
Параметр при инициализации: [jas] -> pushTimeout
Обязательность при инициализации: Необязателен
Значение по умолчанию при инициализации: 2 минуты

18.3.9 Идентификатор системы

Уникальный идентификатор системы для messaging. Если заполнен ограничивает возможность использования OTP лишь теми токенами, идентификатор системы которых совпадает с указанных.

Обязательность: Необязателен
Параметр для команды консольного агента: --messagingSystemId
Параметр при инициализации: [jas] -> messagingSystemId
Обязательность при инициализации: Необязателен

18.3.10 Время жизни кода

Срок действия кода авторизации JAS.

Обязательность: Необязателен
Допустимые значения: Число
Параметр для команды консольного агента: --messagingTTL
Параметр при инициализации: [jas] -> messagingTTL
Обязательность при инициализации: Необязателен

18.3.11 Таймаут между попытками аутентификации

Минимальный интервал между попытками аутентификации через JAS.

Обязательность: Необязателен
Допустимые значения: Число
Параметр для команды консольного агента: --messagingRetryDelay
Параметр при инициализации: [jas] -> messagingRetryDelay
Обязательность при инициализации: Необязателен

18.3.12 Текст сообщения

Текст сообщение для messaging токенов.

Обязательность: Необязателен
Параметр для команды консольного агента: --messagingAdditionalInfo
Параметр при инициализации: [jas] -> messagingAdditionalInfo
Обязательность при инициализации: Необязателен

Пример команды на изменение:

```
sudo jip-agent jas configure \
--url=https://jas.example.com \
--securityType=Basic \
--login=admin \
```

```
--password=secret \
--timeout=”00:00:30” \
--defaultDomain='domain.local' \
--authTypes= Otp,Push,Messaging \
--messagingSystemId=sys1 \
--messagingTTL=600 \
--messagingRetryDelay=5 \
--messagingAdditionalInfo='Test server'
```

18.4 JMS

Параметры JMS используются для настройки подключения к серверу JaCarta Management System. Параметры доступны как в консольном агенте JIP, так и в web-консоли сервера JIP.

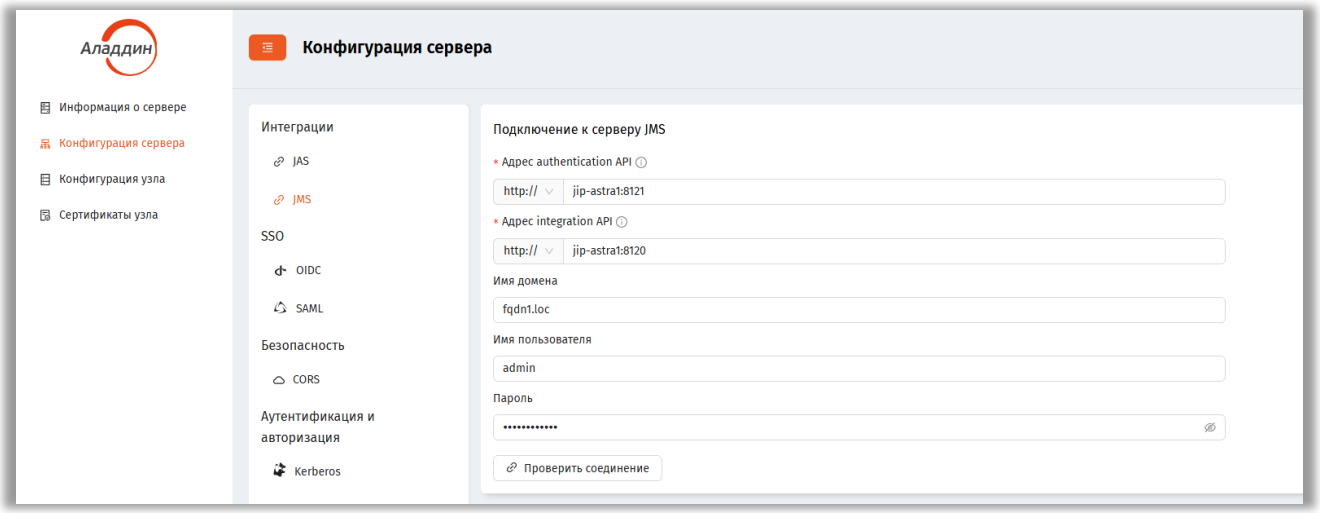

Рис. 11. Раздел JMS web-консоли сервера JIP

18.4.1 Адрес authentication API

URL для обращения к API аутентификации JMS. Используется для получения доступа к интеграционному API JMS.

Обязательность: Обязателен
Допустимые значения: url
Параметр для команды консольного агента: --authenticationApiUrl
Параметр при инициализации: [jms] -> authenticationApiUrl
Обязательность при инициализации: Обязателен

18.4.2 Адрес integration API

URL для обращения к интеграционному API сервера JMS. Используется для получения данных пользователей, профилей SSO и клиентов SSO.

Обязательность: Обязателен
Допустимые значения: url
Параметр для команды консольного агента: --integrationApiUrl
Параметр при инициализации: [jms] -> integrationApiUrl
Обязательность при инициализации: Обязателен

18.4.3 Имя домена

Имя домена учетной записи.

Обязательность: Обязателен
Параметр для команды консольного агента: --authAccountSystemName
Параметр при инициализации: [jms] -> authAccountSystemName
Обязательность при инициализации: Обязателен

18.4.4 Имя пользователя

Имя пользователя для аутентификации. Список необходимых прав указан в разделе 11.1 Требования к пользователю JMS.

Обязательность: Обязателен
Параметр для команды консольного агента: --login
Параметр при инициализации: [jms] -> authId
Обязательность при инициализации: Обязателен

18.4.5 Пароль

Пароль пользователя для подключения.

Обязательность: Обязателен
Параметр для команды консольного агента: --password
Параметр при инициализации: [jms] -> authPassword
Обязательность при инициализации: Обязателен

Пример команды на изменение:

```
sudo jip-agent jms configure \
--authenticationApiUrl=https://jms.example.com/auth \
--integrationApiUrl=https://jms.example.com/integration \
--authAccountSystemName=domain \
--login=user \
--password=pass
```

18.5 SSO

В этом разделе находиться общие параметры для всех SSO протоколов

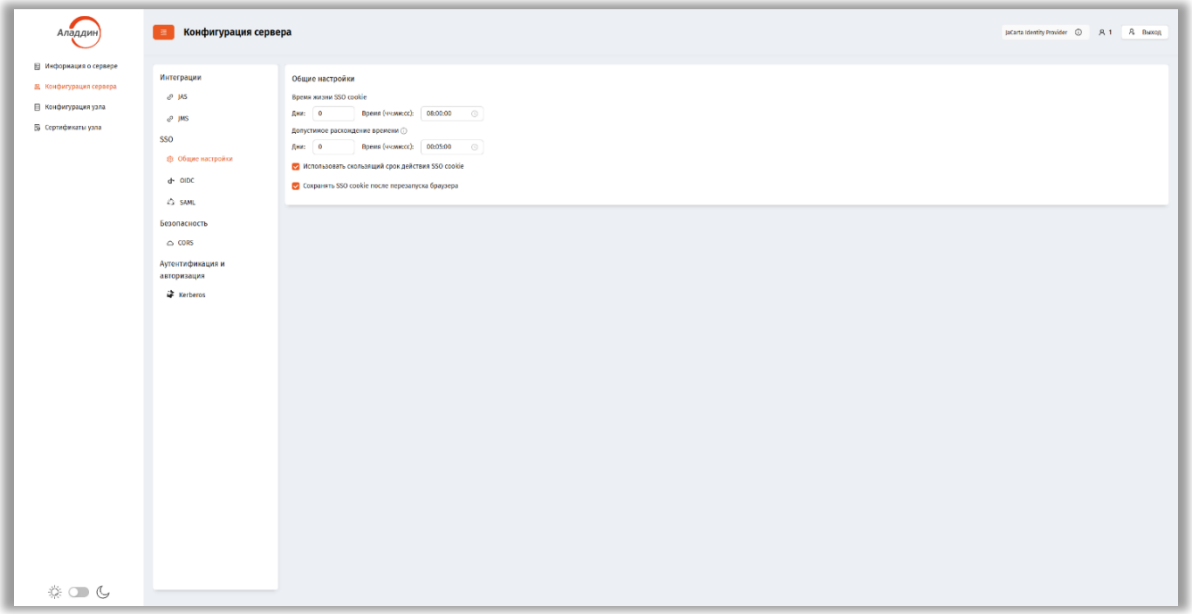

Рис. 12. Раздел SSO web-консоли сервера JIP

18.5.1 Время жизни SSO cookie

Срок действия cookie для аутентификации.

Обязательность: Необязателен
Допустимые значения: TimeSpan. Строка формата d.hh:mm:ss.fffffff, где d – дни, hh – часы, mm – минуты, ss – секунды, ffffff – дробная часть секунд.
Параметр для команды консольного агента: --cookieLifetime
Параметр при инициализации: [sso] -> cookieLifetime
Обязательность при инициализации: Необязателен
Значение по умолчанию при инициализации: 8 часов

18.5.2 Использовать скользящий срок действия cookie

Продление срока действия cookie при каждом использовании.

Обязательность: Необязателен
Допустимые значения: true, false
Параметр для команды консольного агента: --cookieSlidingExpiration
Параметр при инициализации: [sso] -> cookieSlidingExpiration
Обязательность при инициализации: Необязателен
Значение по умолчанию при инициализации: true

18.5.3 Допустимое отклонение времени

Задает допустимую разницу во времени между серверами - участниками аутентификации.

Обязательность: Необязателен
Допустимые значения: TimeSpan. Строка формата d.hh:mm:ss.fffffff, где d – дни, hh – часы, mm – минуты, ss – секунды, ffffff – дробная часть секунд.
Параметр для команды консольного агента: --clockTolerance
Параметр при инициализации: [sso] -> clockTolerance
Обязательность при инициализации: Необязателен
Значение по умолчанию при инициализации: 5 минут

18.5.4 SSO кука постоянного хранения

При влючении данной настройки (значение “true”) SSO кука будет будет доступна после перезапуска браузера. При отключении данной настройки (значение “false”) кука будет автоматически удаляться при закрытии браузера.

Обязательность: Необязателен
Допустимые значения: true, false
Параметр для команды консольного агента: - isPersistent
Параметр при инициализации: [sso] -> isPersistent
Обязательность при инициализации: Необязателен
Значение по умолчанию при инициализации: false

Пример команды на изменение:

```
sudo jip-agent sso configure \
--cookieLifetime=”00:00:30” \
--cookieSlidingExpiration=true \
```

```
--clockTolerance=”00:05:00” \
--isPersistent=true
```

18.6 OIDC

Параметры OIDC позволяют настроить поддержку протокола OpenID Connect. Параметры доступны как в консольном агенте JIP, так и в web-консоли сервера JIP.

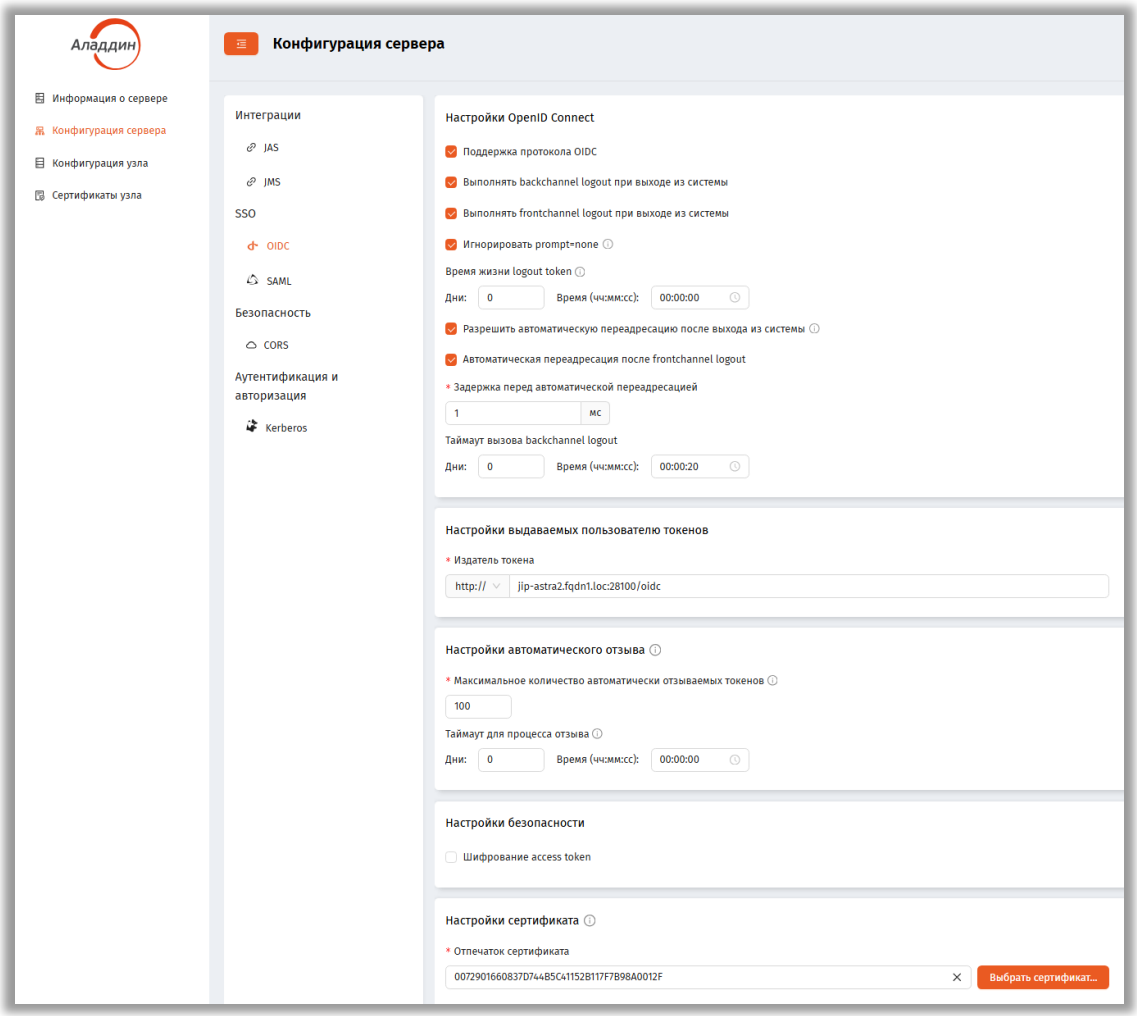

Рис. 13. Раздел OIDC web-консоли сервера JIP

18.6.1 Поддержка протокола OIDC

Отвечает за включение/выключение поддержки протокола OIDC.

Обязательность: Обязателен
Допустимые значения: true, false
Параметр для команды консольного агента: --enabled
Параметр при инициализации: [oidc] -> enabled
Обязательность при инициализации: Обязателен
Значение по умолчанию при инициализации: true

18.6.2 Выполнять backchannel logout при выходе из системы

Разрешает выполнение backchannel logout при выходе из системы согласно настройкам клиента OIDC.

Обязательность: Необязателен
Допустимые значения: true, false
Параметр для команды консольного агента: --enabledBackchannelLogout
Параметр при инициализации: [oidc] -> enabledBackchannelLogout
Обязательность при инициализации: Необязателен
Значение по умолчанию при инициализации: true

18.6.3 Выполнять frontchannel logout при выходе из системы

Разрешает выполнение frontchannel logout при выходе из системы согласно настройкам клиента OIDC.

Обязательность: Необязателен
Допустимые значения: true, false
Параметр для команды консольного агента: --enabledFrontchannelLogout
Параметр при инициализации: [oidc] -> enabledFrontchannelLogout
Обязательность при инициализации: Необязателен
Значение по умолчанию при инициализации: true

18.6.4 Игнорировать prompt=none

Определяет, следует ли игнорировать требование prompt=none запроса авторизации и отображать интерфейс аутентификации при ее необходимости, то есть когда у пользователя нет активной сессии.

Обязательность: Необязателен
Допустимые значения: true, false
Параметр для команды консольного агента: --ignorePromptNone
Параметр при инициализации: [oidc] -> ignorePromptNone
Обязательность при инициализации: Необязателен
Значение по умолчанию при инициализации: false

18.6.5 Разрешить автоматическую переадресацию после выхода из системы

После выхода из системы пользователь будет автоматически переадресован на адрес, указанный в соответствующем клиенте OIDC.

Обязательность: Необязателен
Допустимые значения: true, false
Параметр для команды консольного агента: --enabledAutoRedirectAfterLogout
Параметр при инициализации: [oidc] -> enabledAutoRedirectAfterLogout
Обязательность при инициализации: Необязателен
Значение по умолчанию при инициализации: true

18.6.6 Автоматическая переадресация после frontchannel logout

Определяет, выполнять ли переадресацию после frontchannel logout.

Обязательность: Необязателен
Допустимые значения: Число
Параметр для команды консольного агента: --autoRedirectFrontchannelLogout
Параметр при инициализации: [oidc] -> autoRedirectFrontchannelLogout
Обязательность при инициализации: Необязателен
Значение по умолчанию при инициализации: true

18.6.7 Задержка перед автоматической переадресацией

Время ожидания перед выполнением автоматической переадресации.

Обязательность: Необязателен
Допустимые значения: Число секунд, 0 – без задержки
Параметр для команды консольного агента: --autoRedirectDelay
Параметр при инициализации: [oidc] -> autoRedirectDelay
Обязательность при инициализации: Необязателен
Значение по умолчанию при инициализации: 2 секунды

18.6.8 Таймаут вызова backchannel logout

Максимальное время ожидания ответа при backchannel logout.

Обязательность: Необязателен
Допустимые значения: TimeSpan. Строка формата d.hh:mm:ss.fffffff, где d – дни, hh – часы, mm – минуты, ss – секунды, ffffff – дробная часть секунд.
Параметр для команды консольного агента: --backchannelHttpClientTimeout
Параметр при инициализации: [oidc] -> backchannelHttpClientTimeout
Обязательность при инициализации: Необязателен
Значение по умолчанию при инициализации: 30 секунд

18.6.9 Издатель токена

Определяет идентификатор издателя (Issuer) в рамках протокола OIDC. URL, по которому можно обратиться к web-приложению JIP извне. Может быть, к примеру, http://PC-DOMAIN-NAME/oidc и будет использоваться в качестве Issuer JWT-токенов. Идентификатор издателя (включая протокол, порт и путь) также должен быть прописан в сертификате в Subject CN. При изменении любых данных в издателе не забудьте также обновить и сертификат.

Обязательность: Обязателен
Допустимые значения: url
Параметр для команды консольного агента: --tokenIssuer
Параметр при инициализации: [oidcToken] -> tokenIssuer
Обязательность при инициализации: Обязателен

18.6.10 Максимальное число автоматически отзываемых токенов

Максимальное количество токенов, которые будут отозваны у пользователя за один выход из системы, начиная с тех, что были выданы последними.

Обязательность: Необязателен
Допустимые значения: Число
Параметр для команды консольного агента: --revokeTokenLimit
Параметр при инициализации: [oidcToken] -> revokeTokenLimit
Обязательность при инициализации: Необязателен
Значение по умолчанию при инициализации: 100

18.6.11 Таймаут для процесса отзыва

Максимальная продолжительность процесса автоматического отзыва.

Обязательность: Необязателен
Допустимые значения: TimeSpan. Строка формата d.hh:mm:ss.fffffff, где d – дни, hh – часы, mm – минуты, ss – секунды, ffffff – дробная часть секунд.
Параметр для команды консольного агента: --revokeProcessTimeout
Параметр при инициализации: [oidcToken] -> revokeProcessTimeout

Обязательность при инициализации: Необязателен
Значение по умолчанию при инициализации: 15 секунд

18.6.12 Отключить шифрование access token

Включает или отключает шифрование access token.

Обязательность: Необязателен
Допустимые значения: true, false
Параметр для команды консольного агента: --disableAccessTokenEncryption
Параметр при инициализации: [oidcSecurity] -> disableAccessTokenEncryption
Обязательность при инициализации: Необязателен
Значение по умолчанию при инициализации: true (шифрование отключено)

18.6.13 Отпечаток сертификата

Отпечаток сертификата, используемого в OIDC для подписи и проверки подлинности токенов. Обязательное требование к сертификату: Subject CN должен быть строго равен издателю токена в 18.6.9 Издатель токена, включая протокол, порт и путь.

Обязательность: Обязателен
Допустимые значения: Строка, представляющая отпечаток сертификата в шестнадцатеричном формате
Параметр для команды консольного агента: --certificateThumbprint
Параметр при инициализации: [oidcCertificates] -> certificateThumbprint
Обязательность при инициализации: Обязателен

Пример команды на изменение:

```
sudo jip-agent oidc configure \
--enabled=true \
--enabledBackchannelLogout=true \
--enabledFrontchannelLogout=true \
--ignorePromptNone=false \
--enabledAutoRedirectAfterLogout=true \
--autoRedirectFrontchannelLogout=true \
--autoRedirectDelay=5 \
--backchannelHttpClientTimeout=”00:00:30” \
--tokenIssuer=https://auth.example.com \
--revokeTokenLimit=100 \
--revokeProcessTimeout=”00:00:30” \
--disableAccessTokenEncryption=false \
--disableTransportSecurityRequirements=false \
--certificateThumbprint=1234ABCD
```

18.7 SAML

Параметры SAML позволяют настроить поддержку протокола SAML. Параметры доступны как в консольном агенте JIP, так и в web-консоли сервера JIP.

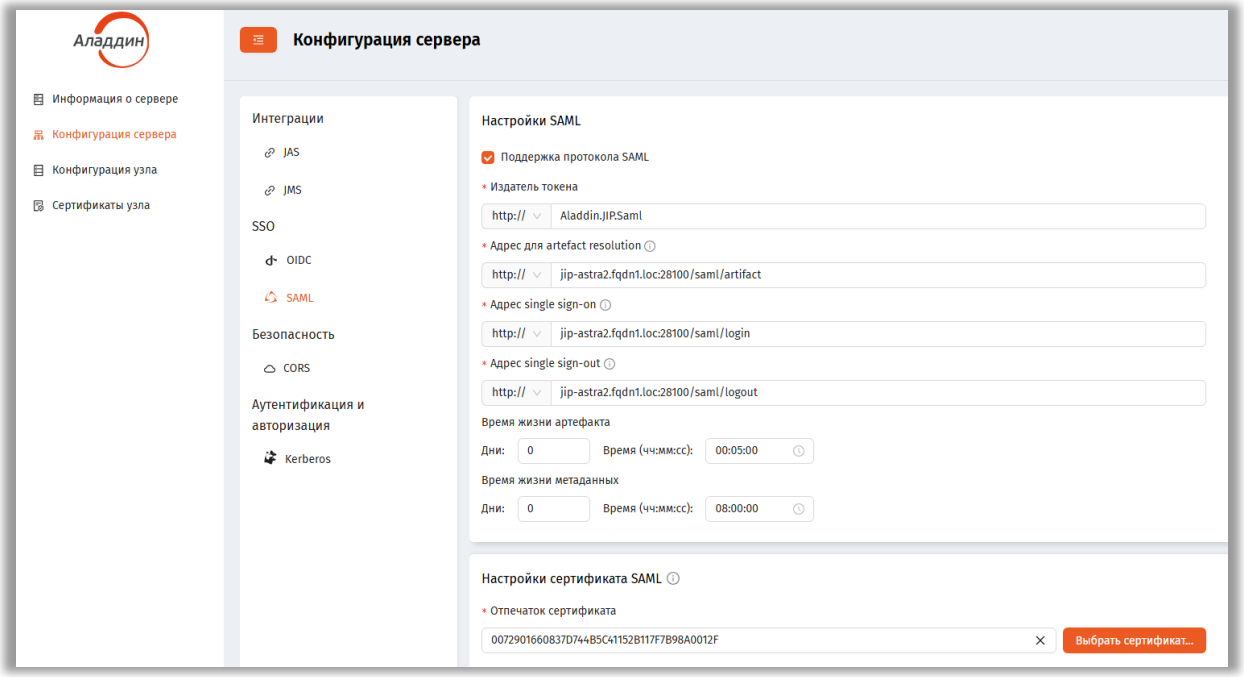

Рис. 14. раздел SAML web-консоли сервера JIP

18.7.1 Поддержка протокола SAML

Отвечает за включение/выключение поддержки протокола SAML.

Обязательность: Обязателен
Допустимые значения: true, false
Параметр для команды консольного агента: --enabled
Параметр при инициализации: [saml] -> enabled
Обязательность при инициализации: Обязателен
Значение по умолчанию при инициализации: true

18.7.2 Издатель токена

Определяет идентификатор издателя (Issuer), который будет использоваться при выпуске утверждений SAML. URL, который будет указан в качестве идентификатора сервера и как Issuer в токене SAML. Попадает в метаданные SAML сервера, которые может читать любой клиент и каждый клиент в своей конфигурации должен будет указать этот Issuer в качестве идентификатора сервера. Главное требование к издателю - должен быть уникален в рамках федерации. По формату должен соответствовать строке URI, но при этом не обязательно должен быть доступной ссылкой. Никак не связан с другими адресами, будь то адресом самого JIP или subject в SSL или OIDC/SAML сертификате. Идентификатор издателя (вся строка, включая протокол, порт и путь) также должен быть прописан в сертификате в Subject CN. При изменении любых данных в издателе не забудьте также обновить и сертификат.

Обязательность: Обязателен
Допустимые значения: url
Параметр для команды консольного агента: --endpointIssuer
Параметр при инициализации: [saml] -> endpointIssuer
Обязательность при инициализации: Обязателен

18.7.3 Адрес для artefact resolution

URL, используемый для обработки запросов разрешения артефактов (Artifact Resolution Service).

Обязательность: Обязателен
Допустимые значения: url
Параметр для команды консольного агента: --artifactResolutionLocation
Параметр при инициализации: [saml] -> artifactResolutionLocation
Обязательность при инициализации: Обязателен

18.7.4 Адрес single sign-on

URL, по которому происходит вход пользователей (Single Sign-On Service).

Обязательность: Обязателен
Допустимые значения: url
Параметр для команды консольного агента: --endpointSingleSignOnDestination
Параметр при инициализации: [saml] -> endpointSingleSignOnDestination
Обязательность при инициализации: Обязателен

18.7.5 Адрес single sign-out

URL, по которому происходит выход пользователей (Single Logout Service).

Обязательность: Необязателен
Допустимые значения: url
Параметр для команды консольного агента: --endpointSingleLogoutDestination
Параметр при инициализации: [saml] -> endpointSingleLogoutDestination
Обязательность при инициализации: Необязателен

Значение по умолчанию при инициализации: Пустая строка

18.7.6 Время жизни артефакта

Указывает, как долго артефакт (SAML Artifact) будет считаться действительным.

Обязательность: Необязателен
Допустимые значения: TimeSpan. Строка формата d.hh:mm:ss.fffffff, где d – дни, hh – часы, mm – минуты, ss – секунды, fffffff – дробная часть секунд.
Параметр при инициализации: [saml] -> artefactLifetime
Обязательность при инициализации: Необязателен
Значение по умолчанию при инициализации: 5 минут

18.7.7 Время жизни метаданных

Определяет срок кэширования метаданных поставщика услуг (SP Metadata).

Обязательность: Необязателен
Допустимые значения: TimeSpan. Строка формата d.hh:mm:ss.fffffff, где d – дни, hh – часы, mm – минуты, ss – секунды, fffffff – дробная часть секунд.
Параметр для команды консольного агента: --spMetadataCacheLifetime
Параметр при инициализации: [saml] -> spMetadataCacheLifetime
Обязательность при инициализации: Необязателен
Значение по умолчанию при инициализации: 8 часов

18.7.8 Отпечаток сертификата

Отпечаток сертификата, используемого для подписи и проверки сообщений SAML. Обязательное требование к сертификату: Subject CN должен быть строго равен издателю токена в 18.7.2 Издатель токена, включая протокол, порт и путь.

Обязательность: Обязателен
Допустимые значения: Строка, представляющая отпечаток сертификата в шестнадцатеричном формате
Параметр для команды консольного агента: --certificateThumbprint
Параметр при инициализации: [samlCertificates] -> certificateThumbprint
Обязательность при инициализации: Обязателен

Пример команды на изменение:

```
sudo jip-agent saml configure \
--enabled=true \
--endpointIssuer=https://idp.example.com \
--endpointSingleSignOnDestination=https://idp.example.com/sso \
--endpointSingleLogoutDestination=https://idp.example.com/slo \
--certificateThumbprint=ABCD1234
```

18.8 CORS

В этом блоке содержатся настройки CORS для интеграционного API web-приложение JIP. Параметры доступны как в консольном агенте JIP, так и в web-консоли сервера JIP.

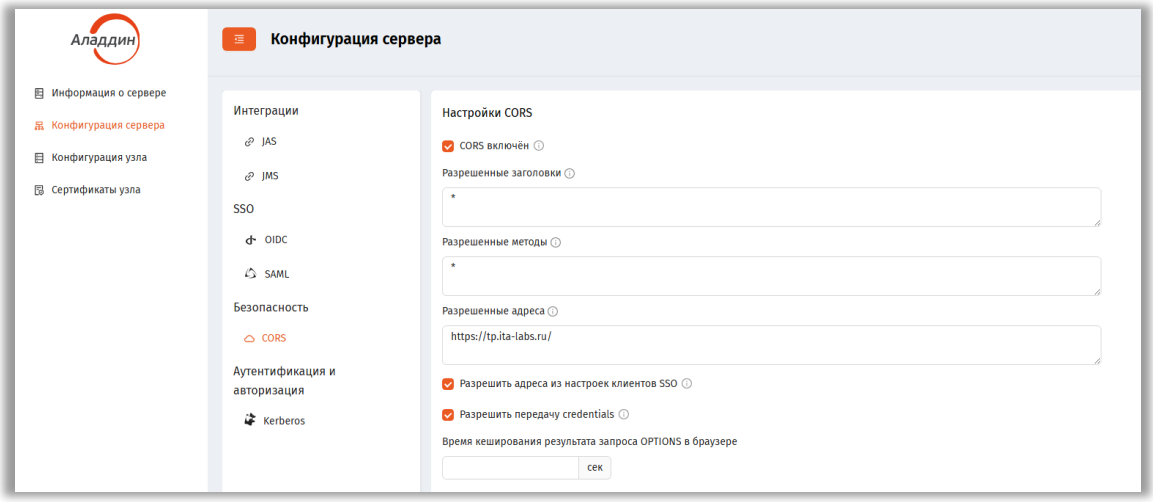

Рис. 15. Раздел CORS web-консоли сервера JIP

18.8.1 CORS включен

Отвечает за включение/выключение CORS в web-приложении JIP для SAML и OIDC endpoint.

Обязательность: Обязателен
Допустимые значения: true, false
Параметр для команды консольного агента: --enabled
Параметр при инициализации: [cors] -> enabled

Обязательность при инициализации: Обязателен
Значение по умолчанию при инициализации: true

18.8.2 Разрешенные заголовки

Соответствует параметру Allowed Headers стандарта CORS.

Обязательность: Необязателен
Допустимые значения: Список заголовков, разделённых запятыми. Например: Authorization, Content-Type
Параметр для команды консольного агента: --allowHeaders
Параметр при инициализации: [cors] -> allowHeaders
Обязательность при инициализации: Необязателен
Значение по умолчанию при инициализации: *

18.8.3 Разрешенные методы

Соответствует параметру Allowed Methods стандарта CORS.

Обязательность: Необязателен
Допустимые значения: Список методов, разделённых запятыми. Например: GET, POST, PUT
Параметр для команды консольного агента: --allowMethods
Параметр при инициализации: [cors] -> allowMethods
Обязательность при инициализации: Необязателен
Значение по умолчанию при инициализации: *

18.8.4 Разрешенные адреса

Соответствует параметру Allowed Origins стандарта CORS.

Обязательность: Необязателен
Допустимые значения: Список адресов, разделённых запятыми. Например: https://example.com, https://app.example.com
Параметр для команды консольного агента: --allowOrigins
Параметр при инициализации: [cors] -> allowOrigins
Обязательность при инициализации: Необязателен
Значение по умолчанию при инициализации: пустое значение

18.8.5 Разрешить адреса из настроек для клиентов SSO

Автоматически добавляет адреса из клиентов SSO в список разрешенных. Если не разрешено, то для корректной переадресации после входа или выхода из системы указанные для переадресации адреса нужно добавить вручную в список разрешенных. В противном случае переадресация будет запрещена браузером.

Обязательность: Необязателен
Допустимые значения: true, false
Параметр для команды консольного агента: --useClientOrigins
Параметр при инициализации: [cors] -> useClientOrigins
Обязательность при инициализации: Необязателен
Значение по умолчанию при инициализации: false

18.8.6 Разрешить передачу credentials

Разрешает серверу включать учётные данные в кросс-доменные HTTP-запросы. К учетным данным относятся: файлы cookie, клиентские сертификаты TLS или заголовки аутентификации, содержащие имя пользователя и пароль. По умолчанию эти учётные данные не отправляются в кросс-доменных

запросах, а включение этой настройки может сделать сайт уязвимым для атак с подделкой межсайтовых запросов(CSRF).

Обязательность: Необязателен
Допустимые значения: true, false
Параметр для команды консольного агента: --allowCredentials
Параметр при инициализации: [cors] -> allowCredentials
Обязательность при инициализации: Необязателен
Значение по умолчанию при инициализации: false

18.8.7 Время кеширования результата запроса OPTIONS в браузере

Определяет максимальный срок жизни результатов preflight (OPTIONS) запросов.

Обязательность: Необязателен
Допустимые значения: TimeSpan. Строка формата d.hh:mm:ss.fffffff, где d – дни, hh – часы, mm – минуты, ss – секунды, ffffff – дробная часть секунд.
Параметр для команды консольного агента: --preflightMaxAge
Параметр при инициализации: [cors] -> preflightMaxAge
Обязательность при инициализации: Необязателен
Значение по умолчанию при инициализации: 10 минут

Пример команды на изменение:

```
sudo jip-agent cors configure \
--enabled=true \
--useClientOrigins=false \
--allowOrigins=https://example.com,https://app.example.com \
--allowMethods=GET,POST,PUT \
--allowHeaders=Authorization,Content-Type \
--allowCredentials=true \
--preflightMaxAge=”01:00:00”
```

18.9 Kerberos

В этом блоке содержатся настройки Kerberos, представляющие собой список keytab-файлов с возможностью регистрации, удаления и просмотра содержимого keytab-файла. Управление списком keytab-файлов доступно как в консольном агенте JIP, так и в web-консоли сервера JIP.

Keytab-файл содержит данные, позволяющие JIP выполнить аутентификацию доменного пользователя по протоколу Kerberos. Один Keytab-файл соответствует одному домену. Realm, в данном случае, является именем домена в формате FQDN.

Важно отметить, что ресурсные системы из JMS никак не связаны с keytab-файлами в JIP, иными словами, они существуют и конфигурируются независимо друг от друга.

Описание процесса создания keytab-файла в доменном окружении выходит за рамки данного руководства.

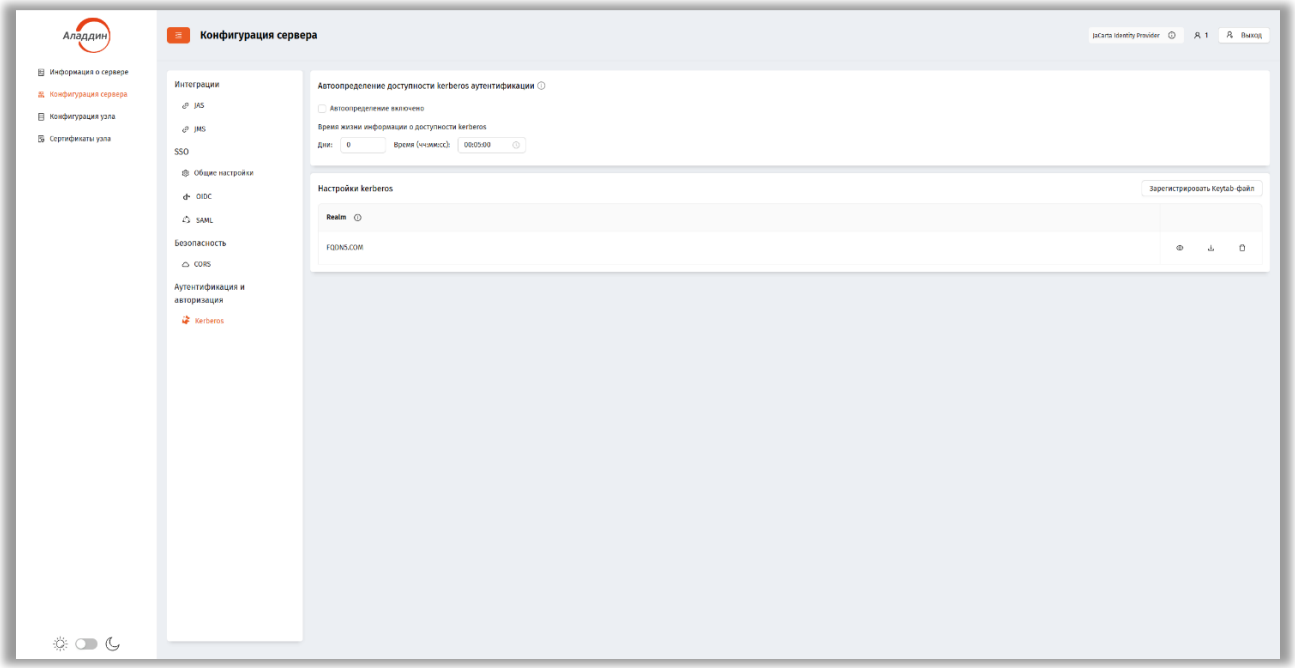

Рис. 16 – Раздел Kerberos web-консоли сервера JIP

18.9.1 Инициализация

При инициализации в ini файле секции [kerberos] можно задать следующие настройки:

18.9.1.1 Список realms

Каждое значение в списке соответствует имени домена в формате FQDN записанное заглавными буквами. Количество элементов в списке realms должно совпадать с количеством путей к файлам, указанном в параметре keyTabFilePaths.

Допустимые значения: Список realm, разделенный запятыми.
Параметр при инициализации: [kerberos] -> realms
Обязательность при инициализации: Не обязателен
Значение по умолчанию при инициализации: “”

18.9.1.2 Список путей к keytab-файлам

Каждое значение в списке соответствует пути к keytab-файлу на локальном компьютере. Количество элементов в списке keyTabFilePaths должно совпадать с количеством realm, указанном в параметре realms.

Допустимые значения: Список путей к файлам, разделенный запятыми.
Параметр при инициализации: [kerberos] -> keyTabFilePaths
Обязательность при инициализации: Не обязателен
Значение по умолчанию при инициализации: “”

18.9.1.3 Включение проверки наличия Kerberos тикета (shortcut)

Данная настройка включает или выключает автоматическую проверку наличия Kerberos тикета при выполнении аутентификации пользователя.

Включение данной настройки позволяет JIP автоматически выбирать стратегию с методом Kerberos если пользователь обладает тикетом и сконфигурирована строго одна стратегия с методом Kerberos. Если в процессе проверки тикет обнаружен не будет и будет в наличии строго одна стратегия без Kerberos, то будет автоматически выбрана она.

Обязательность: Необязателен
Допустимые значения: true, false
Параметр для команды консольного агента: --shortcutEnabled
Параметр при инициализации: [kerberos] -> shortcutEnabled
Обязательность при инициализации: Необязателен
Значение по умолчанию при инициализации: true

18.9.1.4 Время жизни информации о наличии Kerberos тикета

Обязательность: Необязателен
Допустимые значения: TimeSpan. Строка формата d.hh:mm:ss.fffffff, где d – дни, hh – часы, mm – минуты, ss – секунды, fffffff – дробная часть секунд.
Параметр для команды консольного агента: --shortcutLifetime
Параметр при инициализации: [kerberos] -> shortcutLifetime
Обязательность при инициализации: Необязателен
Значение по умолчанию при инициализации: 5 минут

18.9.2 Просмотр списка keytab-файлов

Список зарегистрированных на сервере JIP keytab-файлов отображается в разделе «Аутентификация и авторизация» - «Kerberos» в web-консоли сервера JIP.

Для просмотра списка зарегистрированных keytab-файлов с помощью консольного агента JIP необходимо выполнить команду:

```
sudo jip-agent kerberos list
```

18.9.3 Регистрация keytab-файла

Для регистрции keytab-файла в web-консоли сервера JIP необходимо открыть раздел «Аутентификация и авторизация» - «Kerberos», нажать на кнопку «Зарегистрировать Keytab-файл», переместить keytab-файл в область «Нажмите или переместите Keytab-файл для загрузки», нажать кнопку «Зарегистрировать».

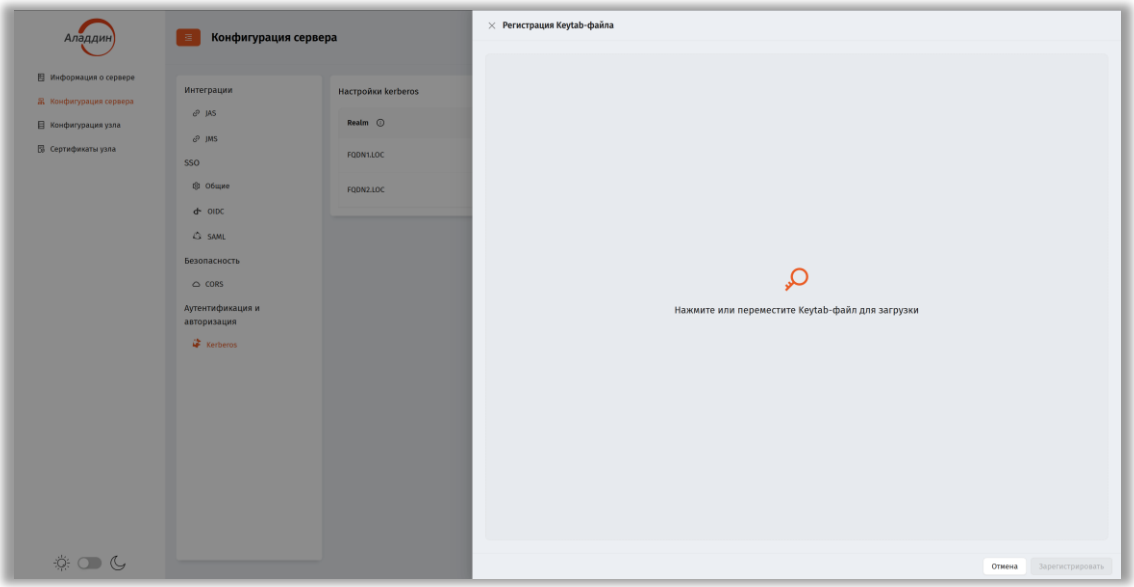

Рис. 17. Диалог регистрации Keytab-файла

Чтобы сервер JIP начал использовать загруженный keytab-файл необходимо нажать на кнопку «Сохранить».

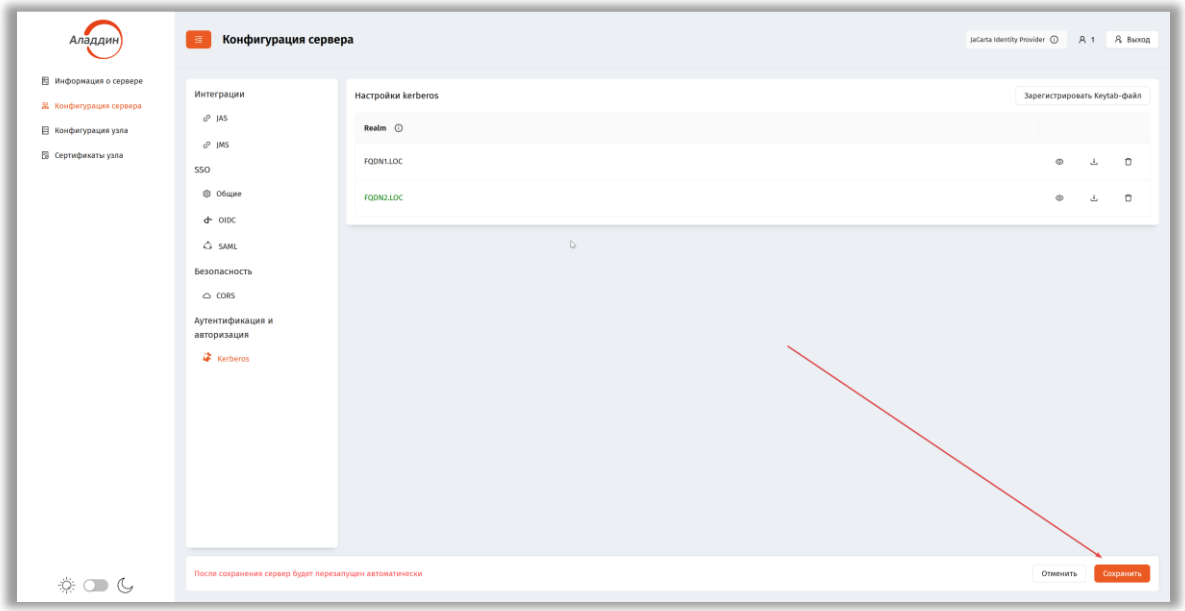

Рис. 18. Применение изменений на сервере JIP после регистрации keytab-файла

Регистрацию keytab-файла можно выполнить в консольном агенте JIP, для этого необходимо воспользоваться следующей командой:

```
sudo jip-agent kerberos register \
--realm=FQDN2.LOC \
--keytabPath=/home/fqdn2.keytab
```

где, realm – имя домена, keytabPath – путь до keytab-файла.

18.9.4 Просмотр данных keytab-файла

Существует возможность просмотреть данные keytab-файл без сохранения в JIP.

Для этого в web-консоли сервера JIP необходимо нажать на кнопку «Зарегистрировать Keytab-файл», выбрать keytab-файл, после чего на экране появится информация из выбранного файла. Если сохранять не требуется, то нужно нажать кнопку «Отмена».

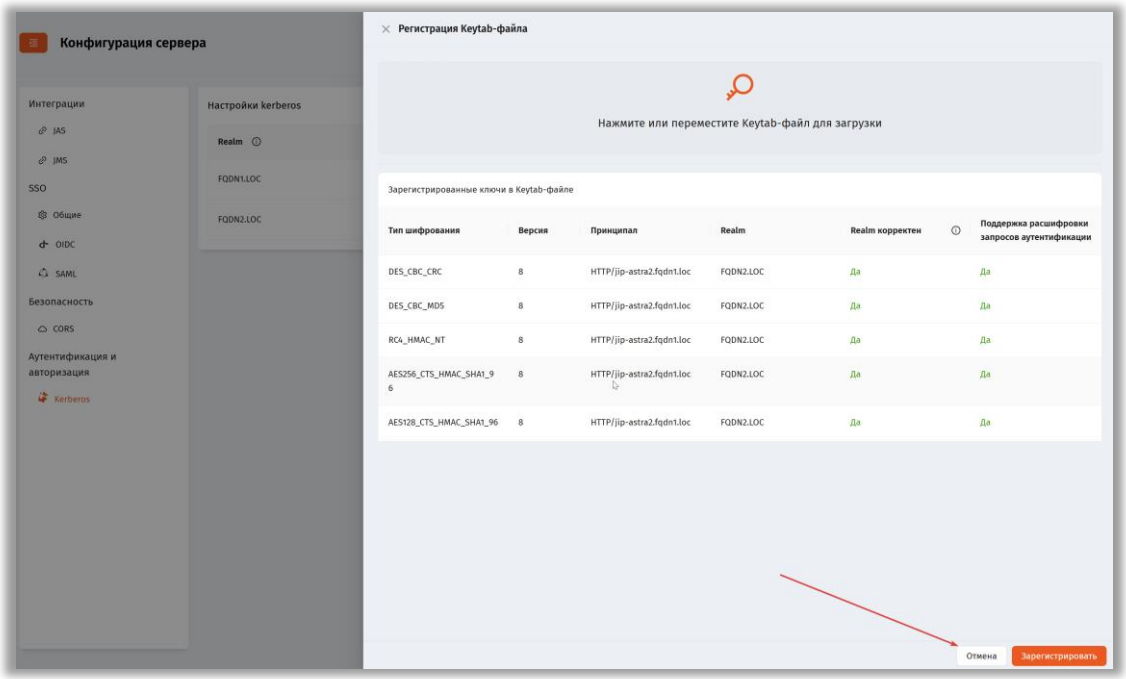

Рис. 19. Просмотр данных keytab-файла в web-консоли сервера JIP

Данные keytab-файла отображаются в виде списка таблицы ключей. По каждому ключу отобразится следующая информация:

- Тип шифрования
- Версия
- Принципал
- Realm
- Realm корректен
- Поддержка расшифровки запросов аутентификации.

Для просмотра данных keytab-файла в консольном агенте JIP можно воспользоваться следующей командой:

```
sudo jip-agent kerberos parse \
--keytabPath=/home/fqdn2.keytab
```

где keytabPath – путь до keytab-файла.

После выполнения команды на экране отобразится информации из данного файла в виде таблицы, по формату аналогичной таблице из web-консоли сервера JIP.

18.9.5 Удаление keytab-файла

Для удаление keytab-файла в web-консоли сервера JIP необходимо в списке keytab-файлов нажать на иконку удаления, после чего в диалоге подтверждения удаления нажать на кнопку «Удалить».

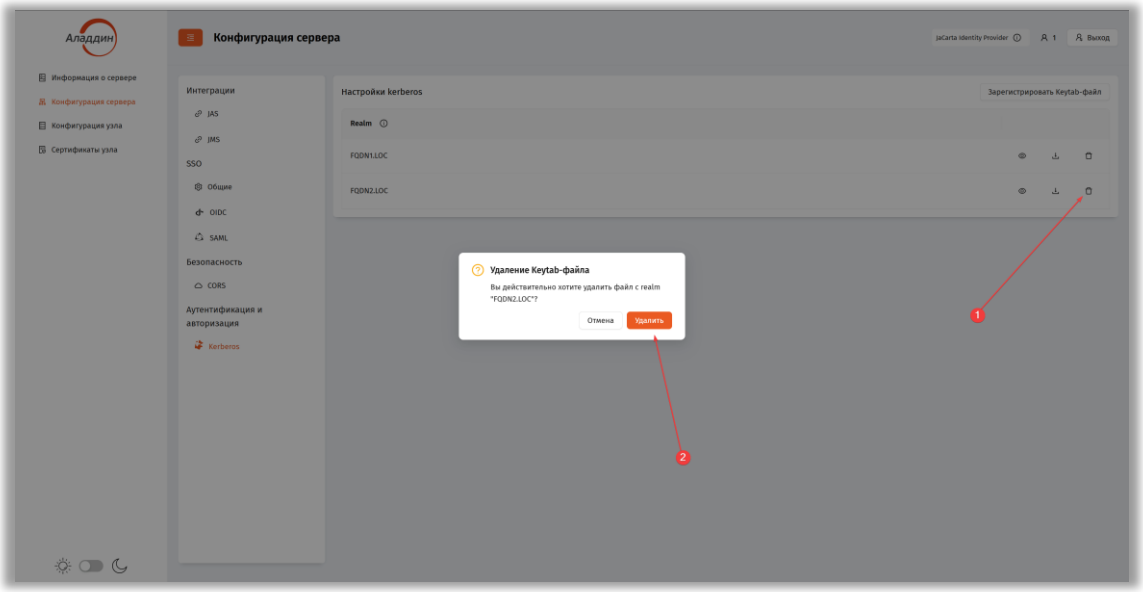

Рис. 20. Удаление keytab-файла в web-консоли сервера JIP

Для того чтобы изменения вступили в действие на сервере JIP необходимо нажать на кнопку «Сохранить».

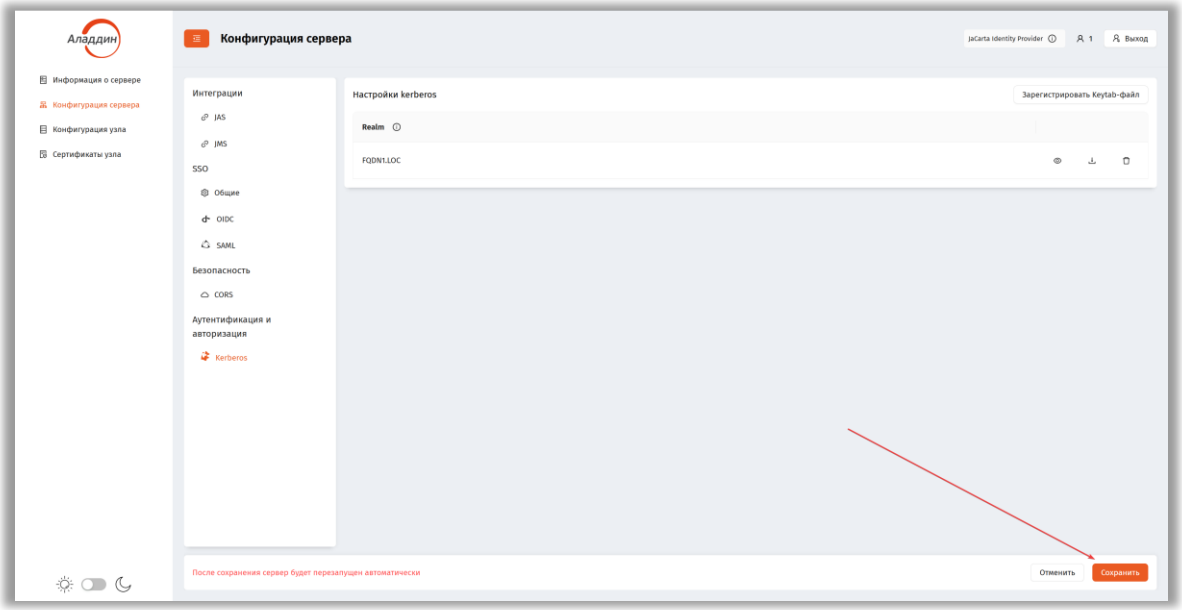

Рис. 21. Применение изменений на сервере JIP после удаления keytab-файла

Для удаления keytab-файла в консольном агенте JIP можно воспользоваться следующей командой:

```
sudo jip-agent kerberos remove \
--realm=FQDN2.LOC
```

18.9.6 Обновление keytab-файла

Обновление существующего keytab-файла в web-консоли сервера JIP возможно только через удаление существующего и регистрации нового keytab-файла.

Для обновления существующего keytab-файла в консольном агенте JIP можно воспользоваться следующей командой:

```
sudo jip-agent kerberos update \
--realm=FQDN2.LOC \
--keytabPath=/home/fqdn2.keytab
```

где – realm – значение Realm существующего keytab-файла, keytabPath – путь до keytab-файла.

18.9.7 Скачивание keytab-файла

Скачивание keytab-файл возможно только в web-консоли сервера JIP. Для этого необходимо в списке keytab-файлов нажать на иконку скачивания файла. После чего браузер выполнит скачивание и сохранит keytab-файл на локальном компьютере.

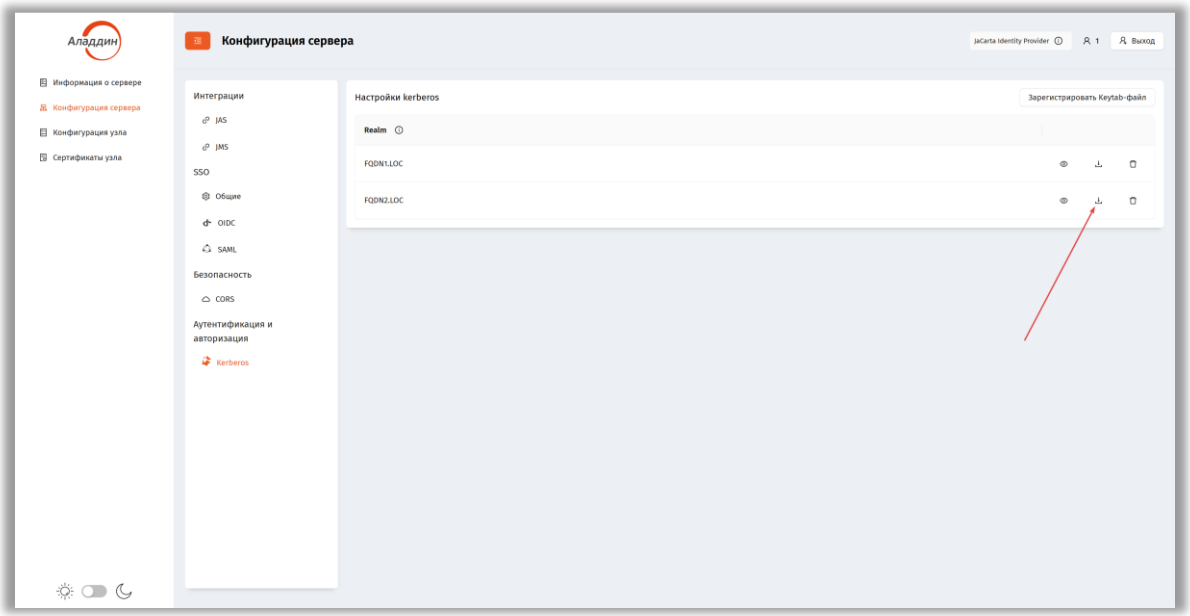

Рис. 22. Скачивание keytab-файла в web-консоли сервера JIP

В консольном агенте JIP возможность получения keytab-файла отсутствует.

18.9.8 Управление автоопределением доступности Kerberos аутентификации (shortcut)

Этим параметром можно управлять в web-консоли сервера JIP. Для этого необходимо перейти в настройки Kerberos

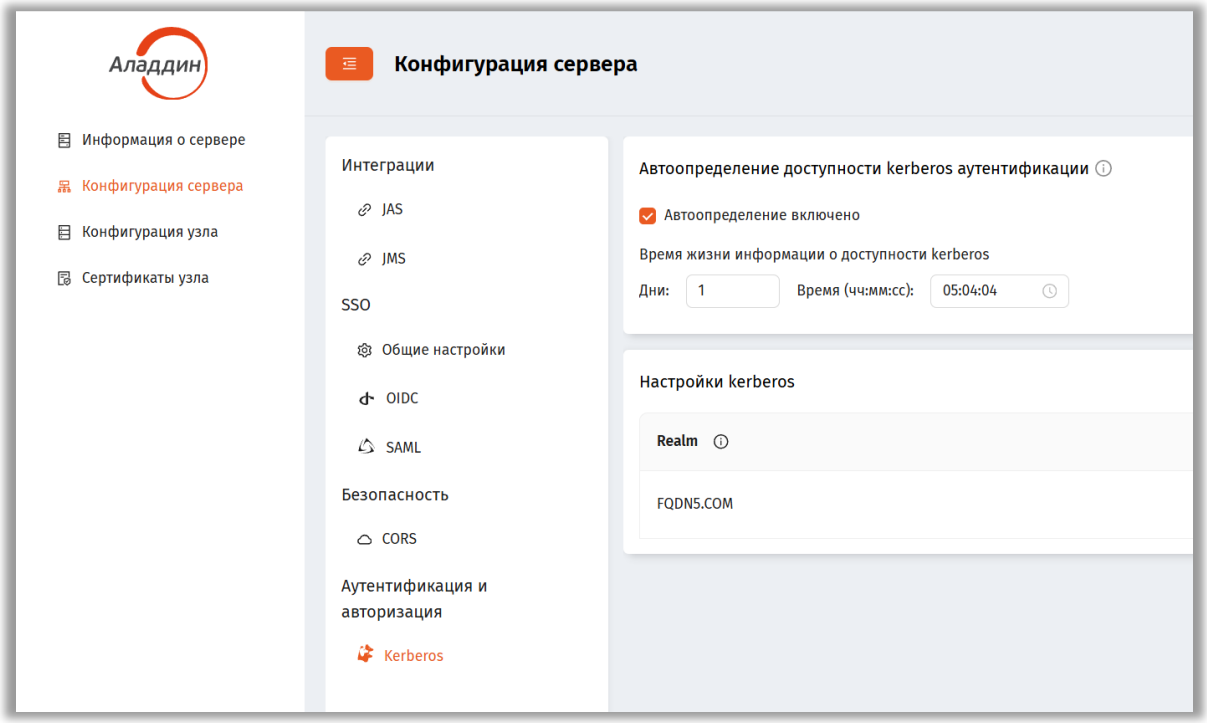

Рис. 23. Управление автоопределением доступности kerberos-аутентификации в web-консоли сервера JIP

18.10 Syslog

В этом блоке содержатся настройки отправки сообщений сервером JIP в Syslog. Параметры доступны как в консольном агенте JIP, так и в web-консоли сервера JIP.

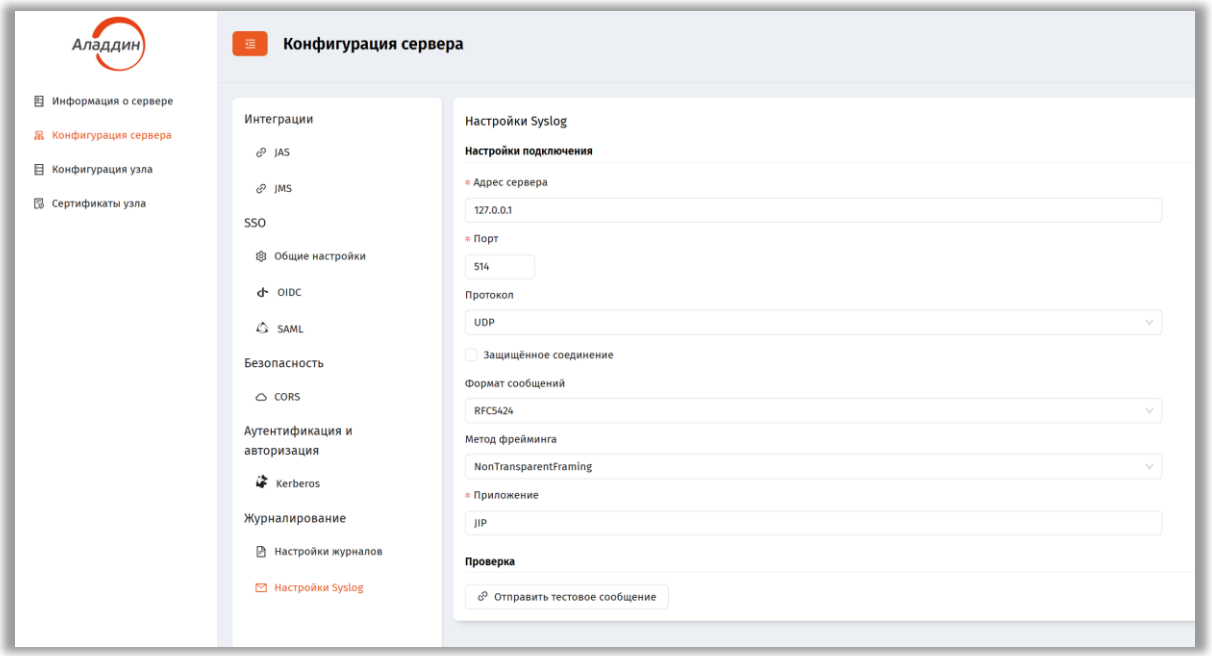

Рис. 24 – Настройка Syslog в web-консоли сервера JIP

18.10.1 Адрес сервера

Адрес Syslog сервера.

Обязательность: Обязателен
Допустимые значения: IP адрес или DNS имя Syslog сервера
Параметр для команды консольного агента: --server
Параметр при инициализации: [syslog] -> server
Обязательность при инициализации: Необязателен
Значение по умолчанию при инициализации: 127.0.0.1

18.10.2 Порт

Номер порта для подключения к Syslog серверу.

Обязательность: Обязателен
Допустимые значения: Число
Параметр для команды консольного агента: --port
Параметр при инициализации: [syslog] -> port
Обязательность при инициализации: Необязателен
Значение по умолчанию при инициализации: 514

18.10.3 Протокол

Протокол, по которому JIP будет подключаться к Syslog серверу.

Обязательность: Необязателен
Допустимые значения: TCP (1), UDP (2), Local (3) можно передавать как строковое представление так и числовое (указано в скобках)
Параметр для команды консольного агента: --protocol
Параметр при инициализации: [syslog] -> protocol
Обязательность при инициализации: Необязателен
Значение по умолчанию при инициализации: UDP

18.10.4 Защищенное соединение

Включает защищенное соединение (только если выбран протокол TCP).

Обязательность: Необязателен
Допустимые значения: true, false
Параметр для команды консольного агента: -- useEncryption
Параметр при инициализации: [syslog] -> useEncryption
Обязательность при инициализации: Необязателен
Значение по умолчанию при инициализации: false

18.10.5 Формат сообщения

Определяет в каком формате сообщение будет передано в Syslog.

Обязательность: Необязателен
Допустимые значения: RFC3164 (0), RFC5424 (1), CEF (2) можно передавать как строковое представление так и числовое (указано в скобках)
Параметр для команды консольного агента: -- rfcVersion
Параметр при инициализации: [syslog] -> rfcVersion
Обязательность при инициализации: Необязателен
Значение по умолчанию при инициализации: RFC5424

18.10.6 Метод фрейминга

Определяет метод разделения сообщений при передаче в Syslog.

Обязательность: Необязателен
Допустимые значения: OctetCounting (0), NonTransparentFraming (1) можно передавать как строковое представление так и числовое (указано в скобках)
Параметр для команды консольного агента: -- messageTransfer
Параметр при инициализации: [syslog] -> messageTransfer
Обязательность при инициализации: Необязателен
Значение по умолчанию при инициализации: NonTransparentFraming

18.10.7 Приложение

Название приложения для Syslog.

Обязательность: Необязателен
Допустимые значения: строка
Параметр для команды консольного агента: -- appName
Параметр при инициализации: [syslog] -> appName
Обязательность при инициализации: Необязателен
Значение по умолчанию при инициализации: JIP

18.10.8 Отправка тестового сообщения

Отправка тестового сообщения в Syslog может быть выполнена как с web-консоли сервера JIP, так и при помощи консольного агента JIP.

Для отправки тестового сообщения в Syslog в web-консоли сервера JIP необходимо нажать на кнопку «Отправить тестовое сообщение». Тестовое сообщение будет отправлено согласно текущим настройкам, которые отображаются на странице «Настройки Syslog». Можно менять настройки без сохранения и выполнять отправку сообщения.

В консольном агенте JIP отправка осуществляется при помощи команды:

```
sudo jip-agent syslog test
```

В отличие от web-консоли сервера JIP отправка в консольном агенте JIP выполняется на сохраненной конфигурации Syslog.

18.10.9 Пример команды консольного агента

Ниже представлена команда конфигурирования блока настроек Syslog.

```
sudo jip-agent syslog configure \
--server=192.168.2.21 \
--port=615 \
--appName=TESTJIP \
--protocol=TCP \
--rfcVersion=RFC5424 \
--messageTransfer=1 \
--useEcryption=true
```

18.11 Журналирование

В этом блоке содержатся настройки журналирования сервера JIP. Данные настройки влияют на отправку событий аудита, как части подсистемы журналирования. Параметры доступны как в консольном агенте JIP, так и в web-консоли сервера JIP.

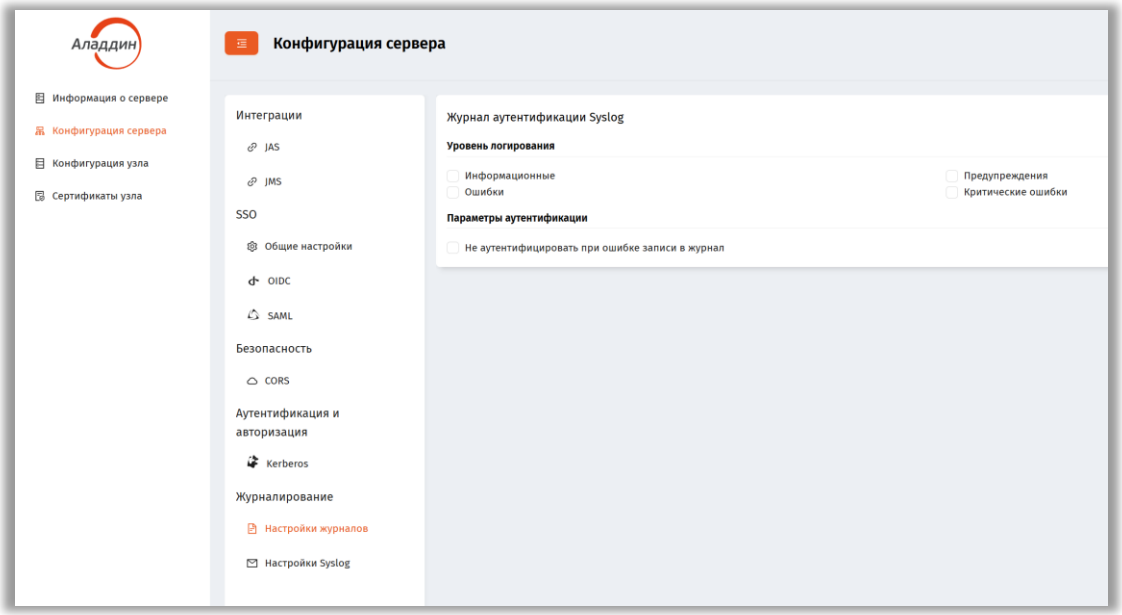

Рис. 25 – Настройка журналирования в web-консоли сервера JIP

18.11.1 Уровень логирования

Определяет какие события будут отправлены в Syslog. None – никакие, All – все события, Info – информационные, Warning – предупреждения, Error – ошибки, Fatal – фатальные ошибки.

Числовые значения можно складывать для получения нужных комбинаций, например, если в Syslog нужно отправлять только Warning (2) и Error (4), то в syslogAuthEventLogLevel нужно передать значение 6.

Обязательность: Необязателен
Допустимые значения: None (0), All (15), Info (1), Warning (2), Error (4), Fatal (8) можно передавать как строковое представление так и числовое (указано в скобках)
Параметр для команды консольного агента: -- syslogAuthEventLogLevel
Параметр при инициализации: [journaling] -> syslogAuthEventLogLevel
Обязательность при инициализации: Необязателен
Значение по умолчанию при инициализации: None

18.11.2 Параметры аутентификации

Определяет как ошибка отправки в Syslog будет влиять на прохождение бизнес-операции. Если указано значение True, то выполнение бизнес операции будет прервано с ошибкой отправки в Syslog. Если указано значение False, то в случае возникновения ошибки отправки в Syslog выполнение бизнес операции продолжится.

Обязательность: Необязателен
Допустимые значения: true, false
Параметр для команды консольного агента: -- syslogFailAuthOnError
Параметр при инициализации: [journaling] -> syslogFailAuthOnError
Обязательность при инициализации: Необязателен
Значение по умолчанию при инициализации: false

18.11.3 Пример команды консольного агента

Ниже представлена команда конфигурирования блока настроек журналирования

```
sudo jip-agent journaling configure \
--syslogAuthEventLogLevel=All \
--syslogFailAuthOnError=true
```

18.12 API проверки работоспособности (Healthcheck API)

API проверки работоспособности предоставляет единую точку доступа для мониторинга состояния сервера JIP и всех его зависимостей. API работает на отдельном порту и может использоваться системами мониторинга, балансировщиками нагрузки и оркестраторами контейнеров.

18.12.1 Адреса API проверки работоспособности

Адреса, на которых доступен Healthcheck API. Параметр: HealthcheckApiAddresses Значение по умолчанию: http://*:28105

18.12.2 Таймаут проверки

Максимальное время ожидания выполнения каждой проверки в миллисекундах. При превышении таймаута проверка считается неуспешной. Параметр: HealthcheckTimeout Значение по умолчанию: 30000 (30 секунд)

18.12.3 Мягкий таймаут проверки

Время в миллисекундах, при превышении которого проверка получает статус «Degraded» (деградация), даже если она завершилась успешно. Используется для раннего обнаружения проблем производительности. Параметр: SoftHealthcheckTimeout Значение по умолчанию: 1000 (1 секунда)

18.12.4 Эндпоинт проверки

URL: GET /healthz Коды ответа:

200 OK — все проверки успешны

503 Service Unavailable — одна или более проверок неуспешны

18.12.5 Выполняемые проверки

Имя проверки	Описание
Database	Проверка подключения к базе данных и получение версии сервера БД
JAS	Проверка доступности и валидности подключения к JAS (если настроен)
JMS	Проверка доступности и валидности подключения к JMS (если настроен)
OIDC_SigningCert	Проверка валидности сертификата подписи OIDC токенов (если OIDC включён)
SAML_SigningCert	Проверка валидности сертификата подписи SAML токенов (если SAML включён)
SSL_AdministrationService	Проверка SSL-сертификата Administration API (если HTTPS включён)
SSL_ControlService	Проверка SSL-сертификата Control API (если HTTPS включён)
SSL_HealthcheckService	Проверка SSL-сертификата Healthcheck API (если HTTPS включён)

18.12.6 Формат ответа

API возвращает ответ в формате JSON (UIHealthReport):


```
{
  "Entries": {
    "Database": {
      "Data": {},
      "Description": "Database: 1.0.0.4",
      "Duration": "00:00:00.0030752",
      "Exception": null,
      "Status": "Healthy",
      "Tags": []
    },
    "JAS": {
      "Data": {},
      "Description": null,
      "Duration": "00:00:00.2242212",
      "Exception": null,
      "Status": "Healthy",
      "Tags": []
    },
    "JMS": {
      "Data": {},
      "Description": null,
      "Duration": "00:00:00.9949582",
      "Exception": null,
      "Status": "Healthy",
      "Tags": []
    },
    "OIDC_SigningCert": {
      "Data": {
        "ExpiredOn": "2027-12-03T12:43:29+03:00"
      },
      "Description": null,
      "Duration": "00:00:00.0014521",
      "Exception": null,
      "Status": "Healthy",
      "Tags": []
    },
    "SAML  SigningCert": {
      "Data": {},
      "Description": "SAML is disabled",
      "Duration": "00:00:00.0000290",
      "Exception": null,
      "Status": "Healthy",
      "Tags": []
    }
  },
  "Status": "Healthy",
  "TotalDuration": "00:00:00.9956308"
}
```

18.12.7 Статусы проверок

Статус	Описание
Healthy	Компонент работает нормально
Degraded	Компонент работает, но с замедлением (превышен мягкий таймаут)
Unhealthy	Компонент недоступен или произошла ошибка

18.12.8 Swagger-документация

По адресу /swagger доступна интерактивная документация OpenAPI для Healthcheck API.

19.Настройки клиентов SSO в JMS

В данном разделе описаны все параметры клиентов SSO доступные к изменению после установки плагинов JIP в Консоли управления JMS в разделе “SSO”.

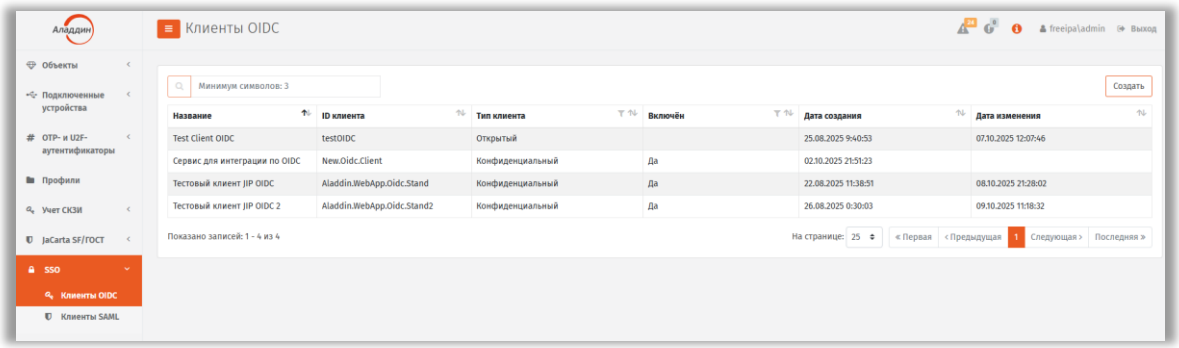

Рис. 26. Клиенты OIDC в разделе SSO Консоли управления JMS

19.1 OIDC

19.1.1 Название

Человекочитаемое название клиента.

Обязательность: Обязателен

19.1.2 Тип клиента

Определяет тип клиента. Этот выбор зависит от типа интегрируемого приложения, если приложение – это SPA-приложение, то нужно выбирать тип «Открытый», в противном случае нужно выбирать «Конфиденциальный».

Обязательность: Обязателен
Допустимые значения: Открытый, Конфиденциальный

19.1.3 ID клиента

Уникальный идентификатор клиента. Необходим для идентификации интегрируемого приложения со стороны JIP в рамках протокола OIDC.

Обязательность: Обязателен

19.1.4 Секрет

Общий секретный ключ, установленный между JIP и интегрируемым приложением. Используется для аутентификации самого интегрируемого приложения в JIP в рамках протокола OIDC.

Обязательность: Обязателен для типа "Конфиденциальный"

19.1.5 Включён

Управляет включением и выключением клиента. При отключении пользователи не смогут войти в данный клиент.

Обязательность: Обязателен
Допустимые значения: true, false

19.1.6 Introspection Feature

Позволяет интегрируемому приложению использовать функционал introspection протокола OIDC для проверки состояния токенов выдаваемых пользователю JIP.

Обязательность: Обязателен
Допустимые значения: true, false

19.1.7 Refresh Token

Позволяет интегрируемому приложению использовать функционал продления срока жизни токенов выдаваемых пользователю JIP в рамках протокола OIDC.

Обязательность: Обязателен
Допустимые значения: true, false

19.1.8 Revocation Endpoint

Позволяет интегрируемому приложению использовать функционал отзыва токенов выдаваемых пользователю JIP в рамках протокола OIDC.

Обязательность: Обязателен
Допустимые значения: true, false

19.1.9 Reference Tokens

Указывает, использовать ли вместо самодостаточных(полных) токенов ссылочные (reference) токены в рамках протокола OIDC. Выбор значения зависит от возможностей интегрируемого приложения. В случае, если явно не указано, что необходимо использование ссылочных токенов, рекомендуется их не использовать.

Обязательность: Обязателен
Допустимые значения: true, false

19.1.10 Время жизни access token (минут)

Задает продолжительность действия access-токена в рамках протокола OIDC.

Обязательность: Обязателен
Допустимые значения: целое число (в минутах)

19.1.11 Время жизни refresh token (минут)

Задает продолжительность действия refresh-токена в рамках протокола OIDC.

Обязательность: Обязателен
Допустимые значения: целое число (в минутах)

19.1.12 Адреса переадресации после входа

Список разрешённых адресов для перенаправления пользователя после успешной аутентификации.

Обязательность: Обязателен для открытых клиентов
Допустимые значения: список URL

19.1.13 Адреса переадресации после выхода

Список разрешенных адресов, на которые можно выполнить переадресацию после завершения сессии.

Обязательность: Необязателен
Допустимые значения: список URL

19.1.14 Адреса для back-channel logout

Список адресов, вызываемых сервером для уведомления клиента о завершении сессии (без участия браузера) в рамках протокола OIDC.

Обязательность: Необязателен
Допустимые значения: список URL

19.1.15 Адреса для front-channel logout

Список адресов, вызываемых для выполнения front-channel logout в рамках протокола OIDC.

Обязательность: Необязателен
Допустимые значения: список URL

19.1.16 Режим автоматического отзыва токенов

Определяет стратегию автоматического отзыва токенов пользователя при завершении сессии.

Обязательность: Обязателен
Допустимые значения: Не отзывать, Для клиента в этой сессии, Для всех клиентов в этой сессии, Для клиента во всех сессиях, Для всех клиентов и всех сессий

19.1.17 Отзывать access token

Указывает, должен ли access-токен отзываться при срабатывании механизма автоматического отзыва.

Обязательность: Обязателен
Допустимые значения: true, false

19.1.18 Отзывать refresh token

Указывает, должен ли refresh-токен отзываться при срабатывании механизма автоматического отзыва.

Обязательность: Обязателен
Допустимые значения: true, false

19.2 SAML

19.2.1 Название

Название клиента. Используется для отображения и идентификации.

Обязательность: Обязателен
Допустимые значения: Строка

19.2.2 Issuer

Идентификатор издателя (Issuer) в рамках протокола SAML.

Обязательность: Обязателен
Допустимые значения: строка (URL или уникальное имя)

19.2.3 Включён

Управляет включением и выключением клиента. При отключении пользователи не смогут войти в данный клиент.

Обязательность: Обязателен
Допустимые значения: true, false

19.2.4 Способ определения адресов клиента

Определяет, как будут задаваться адреса ACS и Logout интегрируемого приложения.

Обязательность: Обязателен
Допустимые значения: Автоматически по метаданным, Указать вручную

19.2.5 Адрес метаданных

Адрес для получения метаданных на стороне интегрируемого приложения.

Обязательность: Обязателен (при автоматическом режим)
Допустимые значения: URI

19.2.6 Адрес ACS клиента

Адрес Assertion Consumer Service (ACS) на стороне интегрируемого приложения, куда JIP будет отправлять ответы в рамках протокола SAML.

Обязательность: Обязателен (при ручном режиме)
Допустимые значения: URL

19.2.7 Адрес Logout клиента

Адрес Single Logout (SLO) на стороне интегрируемого приложения, куда будет перенаправлен пользователь после выхода из системы.

Обязательность: Обязателен (при ручном режиме)
Допустимые значения: URL

19.2.8 Использовать Artefact Binding

Определяет, будет ли использоваться механизм передачи сообщений SAML через артефакт в рамках протокола SAML.

Обязательность: Необязателен
Допустимые значения: true, false

19.2.9 Отключить проверку подписи на логауте

Отключает проверку подписи в запросах на выход от клиента. Может потребоваться, если клиент не подписывает свои logout-запросы.

Обязательность: Необязателен
Допустимые значения: true, false

19.2.10 Время жизни access token (минут)

Задает продолжительность действия access-токена, выдаваемого после успешной аутентификации.

Обязательность: Необязателен
Допустимые значения: целое число (в минутах)

19.2.11 Время жизни refresh token (минут)

Задает продолжительность действия refresh-токена для обновления access-токена.

Обязательность: Необязателен
Допустимые значения: целое число (в минутах)

20.Параметры профиля SSO в JMS

В данном разделе описаны все параметры профиля SSO доступные к изменению после установки плагинов JIP в консоли управления JMS в разделе “Профили”.

20.1.1 Имя

Название профиля. Используется для идентификации и отображения.

Обязательность: Обязателен
Допустимые значения: Строка

20.1.2 Описание

Произвольное текстовое описание профиля.

Обязательность: Необязателен
Допустимые значения: Строка

20.1.3 Клиенты SSO

Список ранее созданных клиентов SSO (SAML, OIDC), доступ к которым будет предоставлен в рамках данного профиля.

Обязательность: Обязателен
Допустимые значения: Список клиентов SAML, OIDC

20.1.4 Стратегии входа

Перечень стратегий входа и их порядок. Стратегия объединяет набор методов аутентификации. При аутентификации пользователю будет предложен выбор стратегий входа. После выбора стратегии пользователю будет необходимо пройти все перечисленные в ней методы, например, пароль и OTP.

Обязательность: Обязателен
Допустимые значения: Список стратегий

20.1.5 Требуется повторная аутентификация

Определяет, должен ли пользователь пройти повторную аутентификацию при входе в один из указанных в профиле клиентов SSO даже в случае наличия у него активной сессии.

Обязательность: Необязателен
Допустимые значения: да, нет

20.1.6 Утверждения OIDC

Настройки утверждений (claims), возвращаемых в токенах OIDC.

20.1.6.1 Утверждение

Название утверждения (claim), возвращаемого в токене OIDC.

Обязательность: Обязателен
Допустимые значения: Строка

20.1.6.2 Тип значения

Указывает, откуда брать значение утверждения.

Обязательность: Обязателен
Допустимые значения: Пользовательский атрибут, Константа

20.1.6.3 Пользовательский атрибут / Константа

Определяет значение для утверждения: либо значение атрибута пользователя, либо фиксированная константа.

Обязательность: Обязателен
Допустимые значения: Пользовательский атрибут, значение константы

20.1.7 Утверждения SAML

Настройка соответствия между утверждениями(attributes) и атрибутами пользователя, возвращаемых в ответах SAML.

20.1.7.1 Утверждение

Название утверждения (attribute), возвращаемого в ответе SAML.

Обязательность: Обязателен
Допустимые значения: Строка

20.1.7.2 Тип значения

Указывает, откуда брать значение утверждения.

Обязательность: Обязателен
Допустимые значения: Пользовательский атрибут, Константа

20.1.7.3 Пользовательский атрибут / Константа

Определяет значение для утверждения: либо значение атрибута пользователя, либо фиксированная константа.

Обязательность: Обязателен
Допустимые значения: Пользовательский атрибут, значение константы

21.Безопасность

21.1 Настройка HTTPS для API сервера JIP

В данном разделе описан процесс настройки HTTPS для административного API и API управления сервера JIP.

Для корректной работы сертификата ему требуется, чтобы в keyUsage было установлено digitalSignature и keyEncipherment, а extendedKeyUsage содержал бы в себе serverAuth (1.3.6.1.5.5.7.3.1).

21.1.1 Настройка локальной (внутренней) точки доступа без HTTPS

По умолчанию часть контроллеров API доступны по локальной (внутренней – localhost) точке доступа без поддержки SSL. Это необходимо для корректного функционирования консольного агента.

Конфигурация по умолчанию:

Табл. 8 – Настройка локальной (внутренней) точки доступа без HTTPS

Контроллер API	Точка доступа активна	Используемый порт
Интерфейс управления (ControlApi)	Да	• 28104
Административный интерфейс (AdministrationApi)	Нет	

Конфигурация локальной (внутренней) точки доступа контролируется отдельно для каждого контроллера API с помощью параметров:

- **EnableLocalhostEndpoint -** true, если требуется активировать локальную точку доступа, иначе false
- **LocalhostEndpointPort -** порт, который будет использоваться для доступа к локальной точке доступа

Примечание. При отключении локальной точки доступа могут возникнуть проблемы с работой некоторых команд консольного агента. Например, в случае, если API сервера настроено на использование HTTPS с сертификатом для определенного домена.

Для конфигурирования локальной (внутренней) точки доступа необходимо отредактировать конфигурационный файл приложения командой:

```
nano /etc/aladdin/jip-engine/appsettings.json
```

Далее нужно:

1. Найти блок настроек контроллера API [Name:]:
   - Интерфейс управления (ControlApi) – *ControlServiceWebApi*
   - Административный интерфейс (AdministrationApi) – *AdministrationServiceWebApi*
2. Создать или изменить параметр «EnableLocalhostEndpoint»
3. Создать или изменить параметр «LocalhostEndpointPort»

В результате должно получиться:

```
"EnableLocalhostEndpoint": "True",
"LocalhostEndpointPort": "28104"
```

Для применения изменений в конфигурационном файле требуется перезапуск сервиса:

```
systemctl restart aladdin-jip-engine
```

21.1.2 Генерация самоподписанного сертификата

Для тестовых целей можно использовать самоподписанный сертификат. Для генерации используется OpenSSL версии 1.1.1.

1. Нужно подготовить конфигурационный файл **openssl.conf** с таким содержанием:

```
[ req ]
default_bits        = 2048
distinguished_name = req_distinguished_name
x509_extensions     = v3_req
prompt              = no

[ req_distinguished_name ]
C  = RU
ST = Moscow
L  = Moscow
O  = MyCompany
OU = IT
CN = http://jip.local:28100/oidc

[ v3_req ]
basicConstraints = CA:FALSE
subjectAltName = @alt_names
keyUsage = digitalSignature, keyEncipherment
extendedKeyUsage = serverAuth


[ alt_names ]
DNS.1 = jip.local
DNS.2 = localhost
```

2. Заменить `jip.local` на имя своего сервера, и, в зависимости от назначения сертификата, укажите корректный CN: http://jip.local:28100/oidc для OIDC или http://jip.local:28100/saml для SAML.
3. Генерируем корневой сертификат с помощью команд выполняя их из директории где был создан файл конфигурации на первом шаге. Можете поменять данные в subj на актуальные в вашей организации:

```
openssl genrsa -out jipRootCA.key 4096
openssl req -x509 -new -nodes -key jipRootCA.key -sha256 -days 3650 -out
jipRootCA.crt -subj "/C=RU/ST=Moscow/L=Moscow/O=Aladdin-RD/OU=IT/CN=Aladdin JIP"
```

4. Генерируем сертификат на основе корневого с помощью команд выполняя их из директории где был создан файл конфигурации на первом шаге:

```
openssl req -new -nodes -out astra2.csr -keyout astra2.key -config openssl.conf

openssl x509 -req -in astra2.csr -CA jipRootCA.crt -CAkey jipRootCA.key -
CAcreateserial -out astra2.crt -days 365 -sha256 -extensions v3_req -extfile
openssl.conf

openssl pkcs12 -export -out astra2.pfx -inkey astra2.key -in astra2.crt -certfile
jipRootCA.crt
```

5. Добавляем корневой сертификат в доверенные:

5.1. в ОС Astra Linux

```
mkdir /usr/local/share/ca-certificates/my/
/bin/cp jipRootCA.crt /usr/local/share/ca-certificates/my/
update-ca-certificates
```

5.2. в RedOS/ALT Linux

```
mkdir /etc/pki/ca-trust/source/anchors/
/bin/cp jipRootCA.crt /etc/pki/ca-trust/source/anchors/
update-ca-trust
```

Последняя команда на обновление сертификатов должна вернуть `1 added, 0 removed; done.`

21.1.3 Добавление сертификата в JIP

1. Открываем web-консоль сервера JIP, в ней раздел Сертификаты узла http://localhost:5030/machineCertificates
2. Нажимаем “+ Установить”

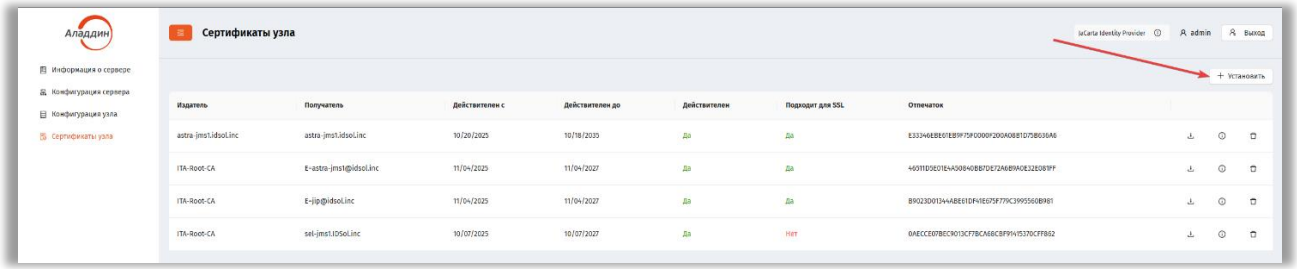

Рис. 27. Установка сертификата в web-консоли сервера JIP

3. Выбираем .pfx файл, вводим пароль, кликаем “Далее”

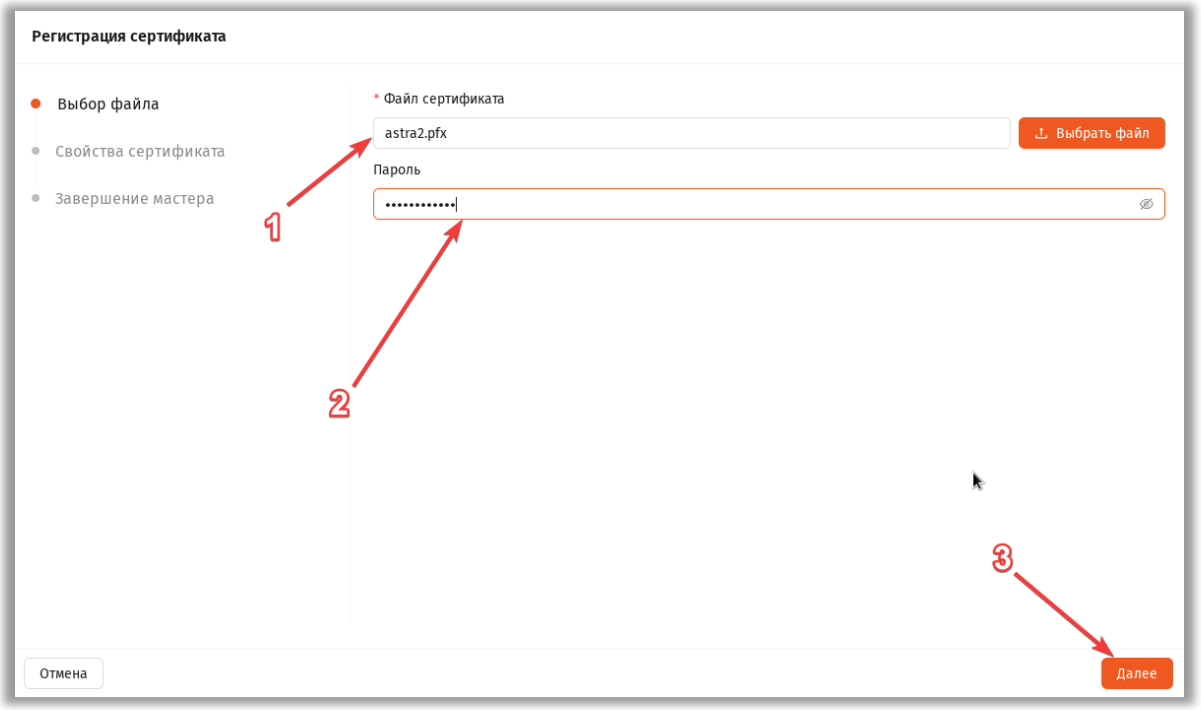

Рис. 28. Установка сертификата в web-консоли сервера JIP

4. На экране проверки сертификата кликаем “Установить”

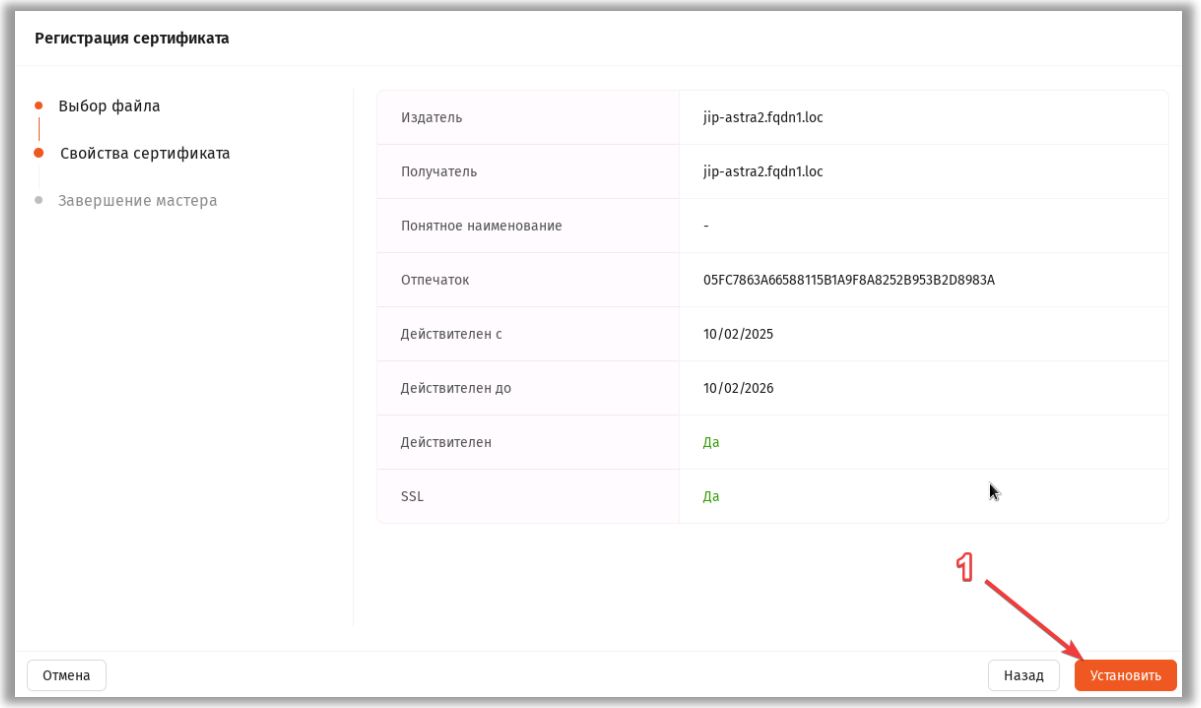

Рис. 29. Установка сертификата в web-консоли сервера JIP

21.1.4 Включение HTTPS в API управления сервера JIP

Включить HTTPS возможно через консольный агент, либо через ручную правку конфигурационного файла сервера JIP.

Для включения HTTPS необходимо указать отпечаток сертификата, ранее добавленного в JIP.

Можно скопировать его из web-консоли сервера JIP: http://localhost:5030/machineCertificates

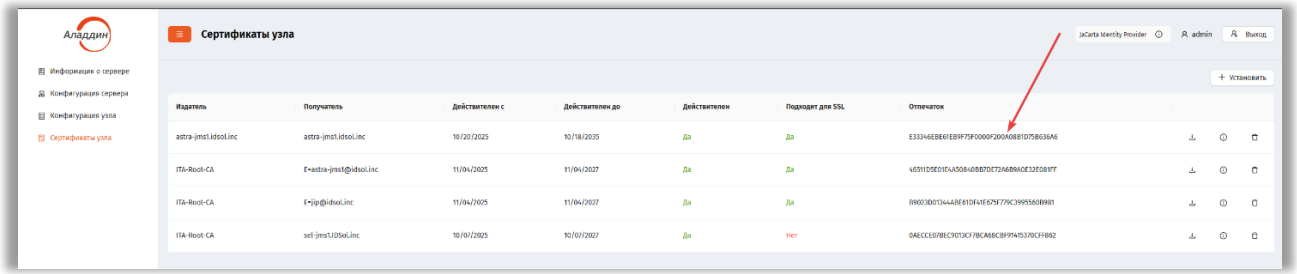

Рис. 30. Получение отпечатка сертификата в web-консоли сервера JIP

21.1.4.1 Включение через консольный агент

Включаем командой:

```
sudo jip-agent ssl enable --api=control --thumbprint="05FC7863A66588115B"
```

21.1.4.2 Включение через редактирование конфигурационного файла

1. Редактируем файл конфигурации сервера JIP:
   nano /etc/aladdin/jip-engine/appsettings.json

2. Меняем “http” на “https”, указываем отпечаток сертификата и в случае, если необходима версия протокола отличная от TLS 1.2, меняем значение параметра SslProtocol. Допустимые значения для него представлены в таблице (https://learn.microsoft.com/ru-ru/dotnet/api/system.security.authentication.sslprotocols?view=net-9.0):
   ```
   "ControlServiceWebApiAddresses": "https://*:28103",
   "Thumbprint": "05FC7863A66588115B1A9F8A8252B953B2D8983A",
   "SslProtocol": "Tls12",
   ```

3. Перезапускаем сервер JIP:
   systemctl restart aladdin-jip-engine

21.1.4.3 Настройка web-консоли сервера JIP

Web-консоль сервера JIP работает через API управления сервера JIP, поэтому если API перевели на HTTPS необходимо обновить адреса API в конфигурации web-консоли сервера JIP.

1. Редактируем конфигурационный файл:
   nano /etc/aladdin/jip-wsc/appsettings.json

   Указываем новые адреса:

   ```
   "ControlApiUrl": "https://jip-astra2.fqdn1.loc:28103",
   "HealthCheckApiUrl": "https://jip-astra2.fqdn1.loc:28105",
   ```

2. Для применения изменений в конфигурационном файле требуется рестарт сервиса:
   systemctl restart aladdin-jip-wsc

21.1.5 Включение HTTPS в административном API сервера JIP

Включить HTTPS возможно через web-консоль сервера JIP, либо через консольный агент, либо через ручную правку конфигурационного файла сервера JIP.

Для включения необходимо указать отпечаток сертификата, ранее добавленного в JIP.

Можно скопировать его из web-консоли сервера JIP: http://localhost:5030/machineCertificates

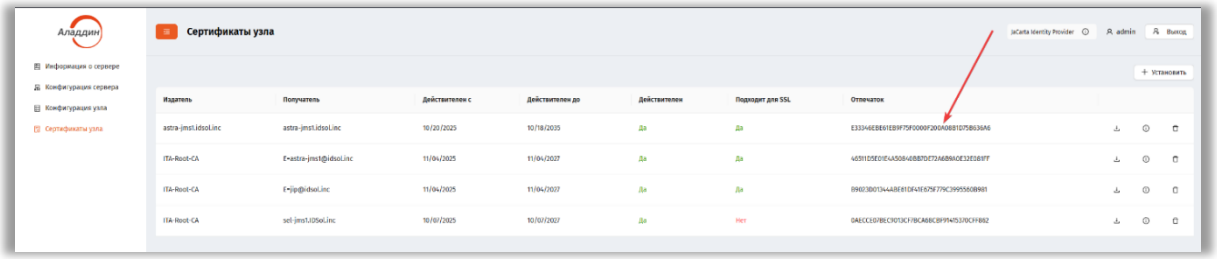

Рис. 31. Получение отпечатка сертификата в web-консоли сервера JIP

21.1.5.1 Включение через web-консоль сервера JIP

1. Открываем раздел “Конфигурация узла”: http://localhost:5030/nodeconfiguration/ssl-settings
2. Включаем https, выбираем сертификат, сохраняем

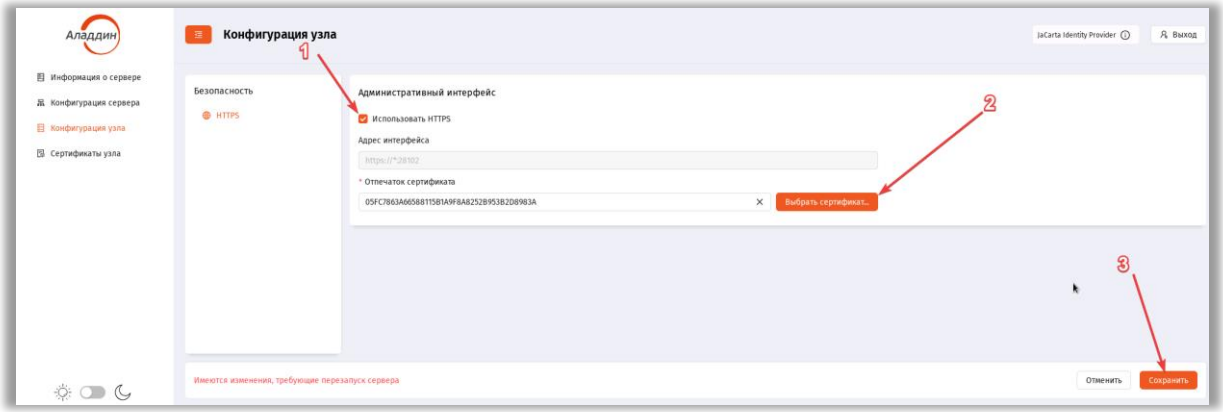

Рис. 32 — Включение HTTPS в web-консоли сервера JIP

3. Требуется ручной перезапуск сервера JIP
   `systemctl restart aladdin-jip-engine`

21.1.5.2 Включение через консольный агент

Включаем командой:

```
sudo jip-agent ssl enable --api=administration --
thumbprint="05FC7863A66588115B"
```

21.1.5.3 Включение через редактирование конфигурационного файла

1. Редактируем файл конфигурации сервера JIP:
   nano /etc/aladdin/jip-engine/appsettings.json

2. Меняем “http” на “https”, указываем отпечаток сертификата и в случае, если необходима версия протокола отличная от TLS 1.2, меняем значение параметра SslProtocol. Допустимые значения для него представлены в таблице (https://learn.microsoft.com/ru-ru/dotnet/api/system.security.authentication.sslprotocols?view=net-9.0):
   ```
   "AdministrationServiceWebApiAddresses": "https://*:28102",
   "Thumbprint": "05FC7863A66588115B1A9F8A8252B953B2D8983A",
   "SslProtocol": "Tls12",
   ```

3. Перезапускаем сервер JIP:
   systemctl restart aladdin-jip-engine.service

21.1.6 Включение HTTPS в API проверки работоспособности сервера JIP

Включить HTTPS возможно через web-консоль сервера JIP, либо через консольный агент, либо через ручную правку конфигурационного файла сервера JIP.

Для включения необходимо указать отпечаток сертификата, ранее добавленного в JIP.

Можно скопировать его из web-консоли сервера JIP: http://localhost:5030/machineCertificates

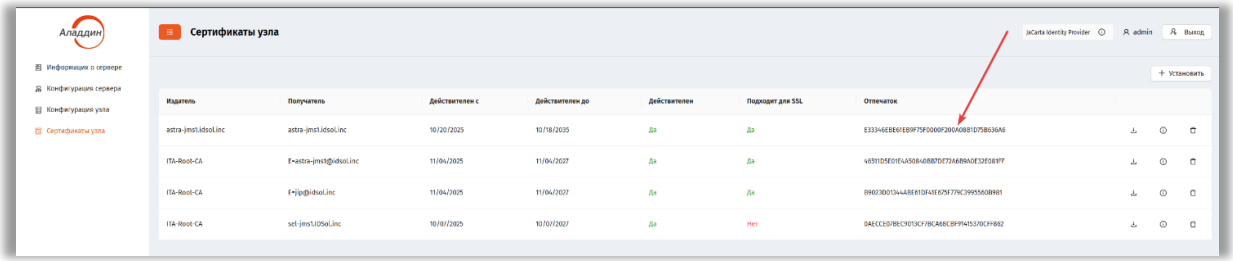

Рис. 33 – Включение HTTPS в web-консоли сервера JIP

21.1.6.1 Включение через web-консоль сервера JIP

1) Открываем раздел “Конфигурация узла”: http://localhost:5030/nodeconfiguration/ssl-settings
2) Включаем https, выбираем сертификат, сохраняем

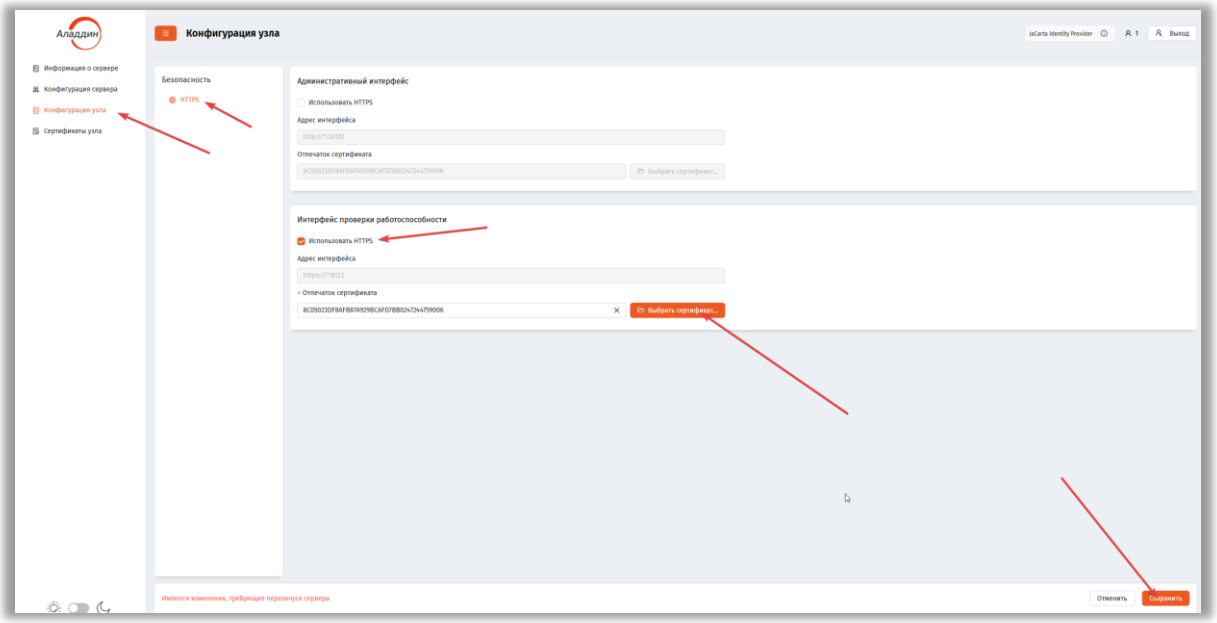

Рис. 34 – Включение HTTPS в веб-консоли управления JIP

3) Требуется ручной перезапуск сервера JIP

```
systemctl restart aladdin-jip-engine
```

21.1.6.2 Включение через консольный агент

Включаем командой:

```
sudo jip-agent ssl enable --api=healthcheck --thumbprint="05FC7863A66588115B"
```

21.1.6.3 Включение через редактирование конфигурационного файла

1) Редактируем файл конфигурации сервера JIP:

```
nano /etc/aladdin/jip-engine/appsettings.json
```

2) Меняем “http” на “https”, указываем отпечаток сертификата и в случае, если необходима версия протокола отличная от TLS 1.2, меняем значение параметра SslProtocol. Допустимые значения для него представлены в таблице (https://learn.microsoft.com/ru-ru/dotnet/api/system.security.authentication.sslprotocols?view=net-9.0):

```
"HealthcheckApiAddresses": "https://*: 8123",
"Thumbprint": "05FC7863A66588115B1A9F8A8252B953B2D8983A",
"SslProtocol": "Tls12",
```

3) Перезапускаем сервер JIP:

```
systemctl restart aladdin-jip-engine.service
```

21.2 Настройка HTTPS для web-приложения JIP

В данном разделе описан процесс настройки HTTPS для web-приложения JIP.

Для корректной работы сертификата ему требуется, чтобы в keyUsage было установлено digitalSignature и keyEncipherment, а extendedKeyUsage содержал бы в себе serverAuth (1.3.6.1.5.5.7.3.1).

21.2.1 Генерация самоподписанного сертификата

Процесс генерации самоподписанного сертификата для web- приложения идентичен процессу, описанному в 21.1.2 Генерация самоподписанного сертификата для API сервера. Только вместо имени `jip.local` в настройке **openssl.conf** нужно указать имя web-приложения. Если вами ранее уже был создан корневой сертификат, можете пропустить шаги инструкции и начать сразу с пункта 4 блока 21.1.2 Генерация самоподписанного сертификата.

21.2.2 Включение HTTPS в web-приложении JIP

Для включения HTTPS понадобится реальный или самоподписанный сертификат.

Добавляем корневой сертификат в доверенные (если он еще не был добавлен при генерации самоподписанного сертификата). Сам сертификат добавлять не нужно, достаточно только корневого сертификата:

1. в ОС Astra Linux

```
mkdir /usr/local/share/ca-certificates/my/
/bin/cp jipRootCA.crt /usr/local/share/ca-certificates/my/
update-ca-certificates
```

2. в RedOS/ALT Linux

```
mkdir /etc/pki/ca-trust/source/anchors/
/bin/cp jipRootCA.crt /etc/pki/ca-trust/source/anchors/
update-ca-trust
```

Последняя команда на обновление сертификатов должна вернуть `1 added, 0 removed; done.`

21.2.2.1 Включение при помощи сертификата из локального хранилища

1. Если subject сертификата неизвестен, то для его получения можно воспользоваться командой **`openssl x509 -in /usr/local/share/ca-certificates/my/astra2.crt –subject –noout`**

   где astra2.crt реальный или самоподписанный сертификат. Из ответа нас интересует значение блока **CN.** Для самоплдписанного сертификата мы указывали его в **openssl.conf** в 21.1.2 Генерация самоподписанного сертификата.

2. После получения subject сертификата необходимо открыть на редактирование файл настройки web-приложения
   `nano /etc/aladdin/jip-web/appsettings.json`

3. В блоке настроек [Kestrel] – [Endpoints] удалить блок [Http] и вставить новый блок [Https] следующего содержания

```
"Kestrel": {
    "Endpoints": {
```

```
      "Https": {
        "Url": "https://*:28100",
        "Certificate": {
          "Subject": "subject серитфиката",
          "Store": "My",
          "Location": "CurrentUser"
        }
      }
    }
}
```

4. Для применения изменений необходимо перезапустить web-приложение JIP командой **systemctl restart aladdin-jip-web**

21.2.2.2 Включение при помощи сертификата из файла

1. Открыть на редактирование файл настройки web-приложения **nano /etc/aladdin/jip-web/appsettings.json**

2. В блоке настроек [Kestrel] – [Endpoints] удалить блок [Http] и вставить новый блок [Https] следующего содержания

```
"Kestrel": {
    "Endpoints": {
      "Https": {
        "Url": "https://*:28100",
        "Certificate": {
          "Path": "путь_до_pfx_файла_сертификата",
          "Password": "пароль_от_сертификата"
         }
      }
    }
}
```

Примечание. Пароль сертификата можно указать в открытом виде - при первом запуске при-ложение автоматически зашифрует его в файле конфигурации.

3. Для применения изменений необходимо перезапустить web-приложение JIP командой **systemctl restart aladdin-jip-web**

21.2.2.3 Настройка OIDC

После того как изменили HTTP на HTTPS в web-приложении необходимо изменить настройки протокола OIDC в сервере JIP.

Для этого необходимо открыть web-консоль сервера JIP, выбрать раздел «Конфигурация сервер» - «SSO» - «OIDC» и изменить протокол с HTTP на HTTPS у параметра «Издатель токена» в блоке «Настройка выдаваемых пользователю токенов».

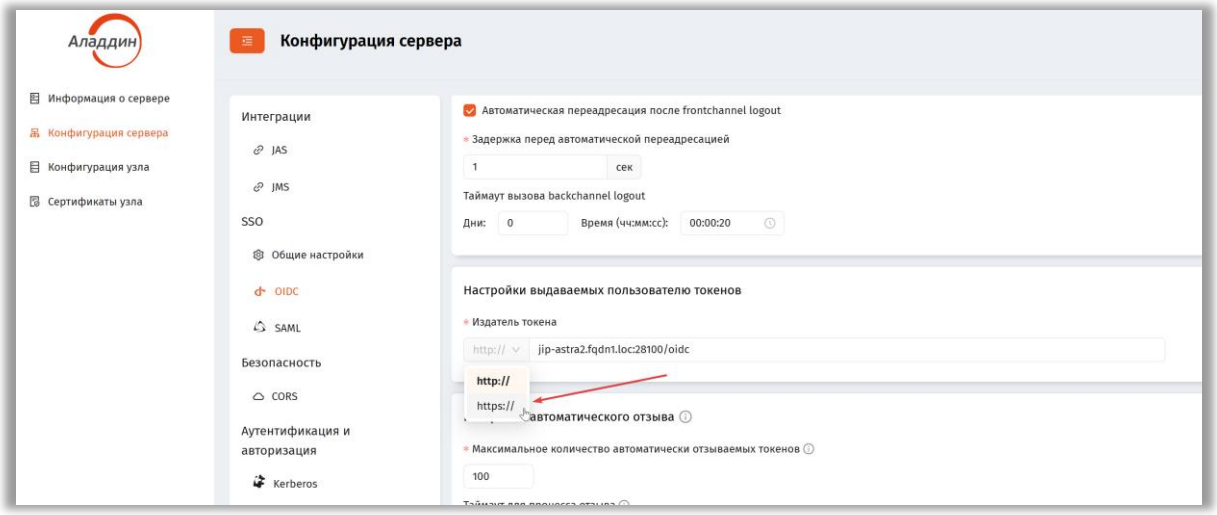

Рис. 35. Изменение протокола на HTTPS у издателя OIDC в web-консоли сервера JIP

После изменения сохранить изменения.

21.2.2.4 Настройка SAML

После того как изменили HTTP на HTTPS в web-приложении необходимо изменить настройки протокола SAML в сервере JIP.

Для этого необходимо открыть web-консоль сервера JIP, выбрать раздел «Конфигурация сервер» - «SSO» - «SAML» и изменить протокол с HTTP на HTTPS у параметров: «Издатель токена», «Адрес для artefact resolution», «Адрес single sign-on» и «Адрес single sign-out» в блоке «Настройка SAML».

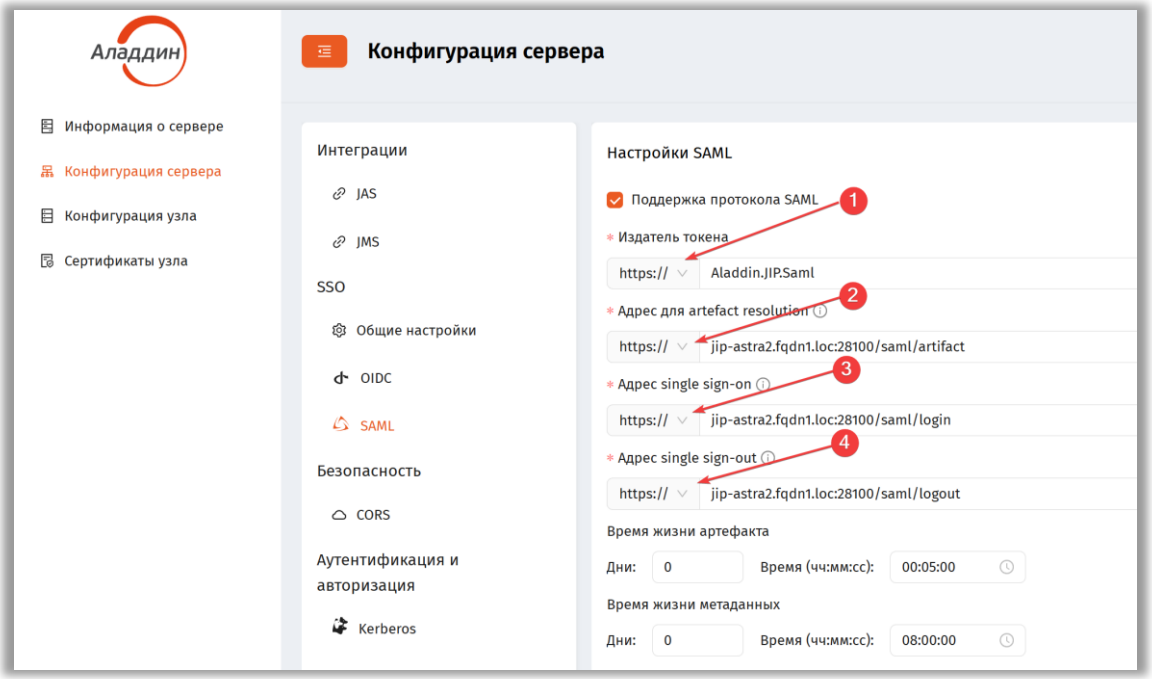

Рис. 36. Изменение протокола на HTTPS в настройках SAML в web-консоли сервера JIP

После изменения сохранить изменения.

21.2.2.5 Изменение в клиентских приложениях

Если на момент изменения с HTTP на HTTPS уже были выполнены клиентские интеграции (не важно по какому протоколу OIDC или SAML), то для обеспечения корректного взаимодействия, клиентам нужно изменить у себя адреса сервера JIP с http:// на https://.

21.3 Настройка HTTPS для web-консоли сервера JIP

21.3.1 Генерация самоподписанного сертификата

Процесс генерации самоподписанного сертификата для веб-приложения идентичен процессу, описанному в 21.1.2 Генерация самоподписанного сертификата для API сервера. Только вместо имени `jip.local` в настройке **openssl.conf** нужно указать имя web-консоли сервера JIP. Если вами ранее уже был создан корневой сертификат, можете пропустить шаги инструкции и начать сразу с пункта 4 блока 21.1.2 Генерация самоподписанного сертификата.

21.3.2 Включение HTTPS в web-приложении JIP

Для включения HTTPS понадобится реальный или самоподписанный сертификат.

Добавляем корневой сертификат в доверенные (если он еще не был добавлен при генерации самоподписанного сертификата). Сам сертификат добавлять не нужно, достаточно только корневого сертификата:

1. в ОС Astra Linux

```
mkdir /usr/local/share/ca-certificates/my/
/bin/cp jipRootCA.crt /usr/local/share/ca-certificates/my/
update-ca-certificates
```

2. в RedOS/ALT Linux

```
mkdir /etc/pki/ca-trust/source/anchors/
/bin/cp jipRootCA.crt /etc/pki/ca-trust/source/anchors/
update-ca-trust
```

Последняя команда на обновление сертификатов должна вернуть `1 added, 0 removed; done.`

21.3.2.1 Включение при помощи сертификата из локального хранилища

1. Если subject сертификата неизвестен, то для его получения можно воспользоваться командой
   `openssl x509 -in astra2.crt –subject –noout`

   где astra2.crt реальный или самоподписанный сертификат. Из ответа нас интересует значение блока **CN.** Для самоподписанного сертификата мы указывали его в **openssl.conf** в 21.1.2 Генерация самоподписанного сертификата.

2. После получения subject сертификата необходимо открыть на редактирование файл настройки web-приложения
   `nano /etc/aladdin/jip-wsc/appsettings.json`

3. В блоке настроек [Kestrel] – [Endpoints] удалить блок [Http] и вставить новый блок [Https] следующего содержания

```
"Kestrel": {
    "Endpoints": {
      "Https": {
        "Url": "https://*:5030",
        "Certificate": {
          "Subject": "subject серитфиката",
          "Store": "My",
          "Location": "CurrentUser"
        }
      }
    }
 }
```

Для применения изменений необходимо перезапустить web-приложение JIP командой **systemctl restart aladdin-jip-wsc**

21.3.2.2 Включение при помощи сертификата из файла

1. Открыть на редактирование файл настройки web-приложения **nano /etc/aladdin/jip-wsc/appsettings.json**

2. В блоке настроек [Kestrel] – [Endpoints] удалить блок [Http] и вставить новый блок [Https] следующего содержания

```
"Kestrel": {
    "Endpoints": {
      "Https": {
        "Url": "https://*:5030",
        "Certificate": {
          "Path": "путь_до_pfx_файла_сертификата",
          "Password": "пароль_от_сертификата"
         }
      }
    }
 }
```

Примечание. Пароль сертификата можно указать в открытом виде - при первом запуске при-ложение автоматически зашифрует его в файле конфигурации.

3. Для применения изменений необходимо перезапустить web-приложение JIP командой **systemctl restart aladdin-jip-wsc**

21.4 Хранение паролей

JIP использует пароли для безопасного доступа к API и СУБД. Они хранятся в конфигурационных файлах. Для безопасности эти пароли шифруются.

21.4.1 Сервер JIP

Конфигурационный файл сервера располагается по адресу /etc/aladdin/jip-engine/appsettings.json
Внутри него зашифровано три пароля:

- параметр «Password» секции «ControlServiceWebApi» - хранит пароль, который используется для доступа к API управления
- параметр «Password» секции «AdministrationServiceWebApi» - хранит пароль, который используется для доступа к API Администрирования
- часть параметра «ConnectionString» секции «DatabaseManager» - хранит пароль, который сервер использует для доступа к СУБД

21.4.2 Web-приложение JIP

Конфигурационный файл web-приложения располагается по адресу /etc/aladdin/jip-web/appsettings.json.
Внутри него зашифровано: параметр «ApiPassword» в секции «WebHost» - пароль, который web-приложение использует для доступа к API Администрирования.

21.5 Изменение паролей

Для изменения какого-либо из паролей, описанных выше, без повторной инициализации JIP вам понадобится отредактировать конфигурационный файл приложения. Остановите приложение, укажите новый пароль, вместо зашифрованного старого и запустите приложение. Посте старта приложение зашифрует новый пароль. Проверить это вы можете открыв файл конфигурации и увидев, что пароль теперь зашифрован

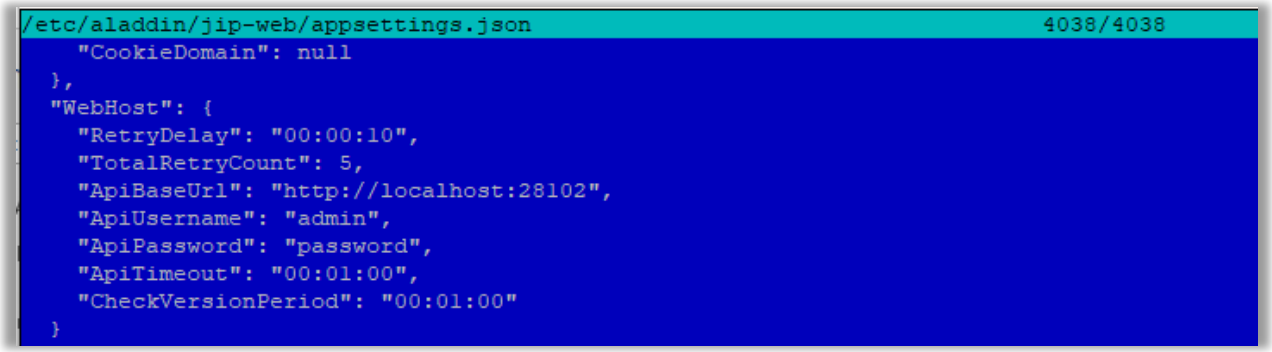

Рис. 37. Конфигурационный файл с изменённым паролем до старта приложения

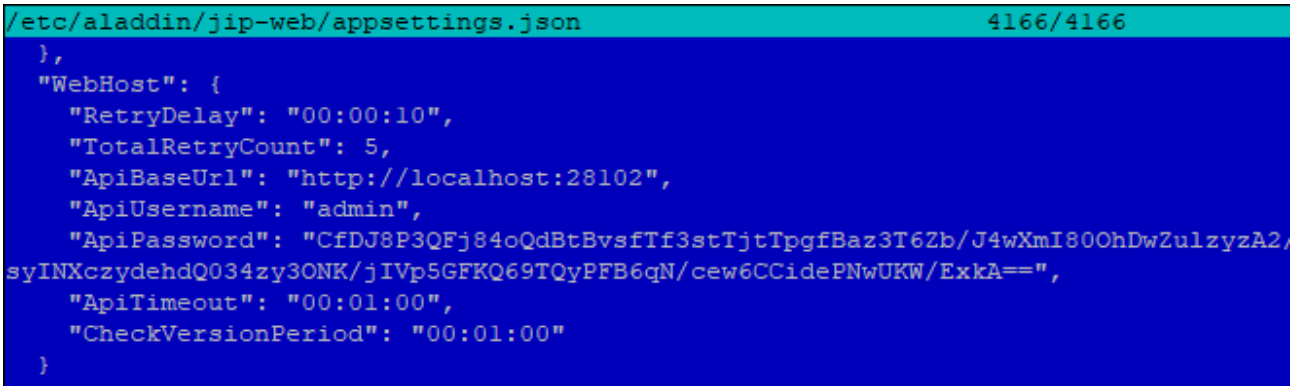

Рис. 38. Конфигурационный файл после старта приложения

22.Интеграция с внешними приложениями

В разделе описан общий порядок интеграции с внешними сервисами по протоколам OIDC и SAML.

22.1 Интеграция по OIDC

Для интеграции с JIP по протоколу OIDC внешний сервис (далее сервис) должен поддерживать такую возможность на уровне протокола взаимодействия (https://openid.net/specs/openid-connect-core-1_0.html).

В данном случае JIP выступает в качестве сервера OIDC, а сервис в качестве клиента OIDC.

JIP на данный момент поддерживает следующие flow интеграции в рамках OIDC:

- Authorization Code Flow (с PKCE и без)
- Refresh Token Flow

Для интеграции необходимо выполнить следующие действия:

1. Зарегистрировать сервис в качестве клиента OIDC
2. Создать новый или изменить существующий профиль SSO
3. Привязать новый профиль к дереву РС
4. Выполнить настройку сервиса в качестве клиента OIDC

Сервер JIP предоставляет OIDC Discovery Endpoint. Данный Endpoint доступен только по url, который был указан в качестве издателя токена (18.6.9 Издатель токена). В случае обращение по IP на OIDC Discovery Endpoint данные возвращены не будут.

22.1.1 Регистрация сервиса в JIP

Для регистрации сервиса в качестве клиента OIDC необходимо в Консоли управления JMS (далее портал) выполнить следующие действия:

Открыть список клиентов OIDC путем выбора в навигаторе пункта «SSO» -> «Клиенты OIDC». После чего нажать на кнопку «Создать» для создания нового клиента.

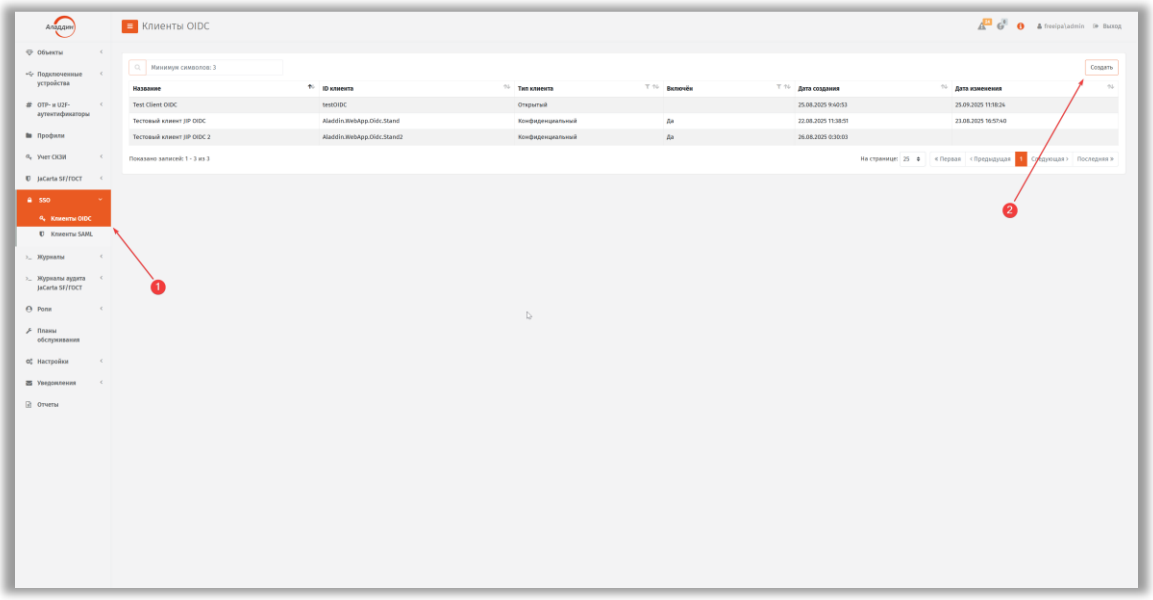

Рис. 39. Создание клиента OIDC

После нажатия на кнопку «Создать» появится форма создания нового клиента OIDC. На форме создания нового клиента OIDC необходимо указать следующие обязательные значения:

- «Название» – произвольное человекочитаемое название сервиса
- «Тип клиента» – выбрать «Открытый» или «Конфиденциальный»
- «ID клиента» – уникальный машиночитаемый идентификатор клиента. **Данное значение необходимо запомнить, т.к. оно понадобится при настройке интеграции на стороне сервиса**
- «Секрет» (для клиента типа «Конфиденциальный») – это пароль сервиса при обращении к JIP. **Данное значение необходимо запомнить, т.к. оно понадобится при настройке интеграции на стороне сервиса**
- «Включен» - на этапе настройки клиент можно отключить
- «Адрес переадресации после входа» – адрес на стороне сервиса куда пользователь будет перенаправлен после успешного входа, предоставляется сервисом
- «Адрес переадресации после выхода» – адрес на стороне сервиса куда пользователь будет перенаправлен после успешного выхода, предоставляется сервисом

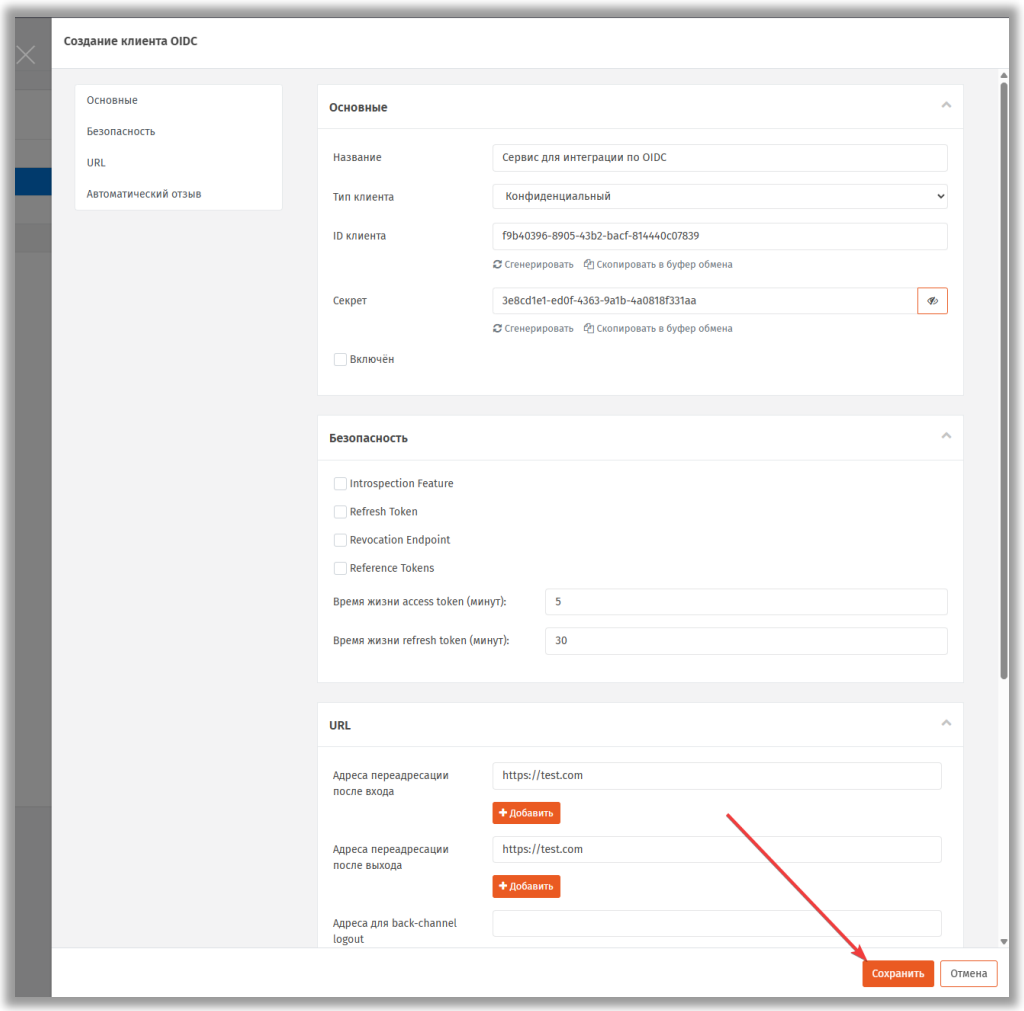

Рис. 40 – Создание клиента OIDC

После заполнения свойств клиента, необходимо нажать кнопку «Сохранить». После успешного сохранения, новый клиент появится в списке клиентов OIDC.

22.1.2 Создание профиля SSO

Для создания профиля необходимо выбрать в навигаторе пункт «Профили», далее в области списка профилей нажать на кнопку «Создать» и в выпадающем меню выбрать пункт «Профиль SSO».

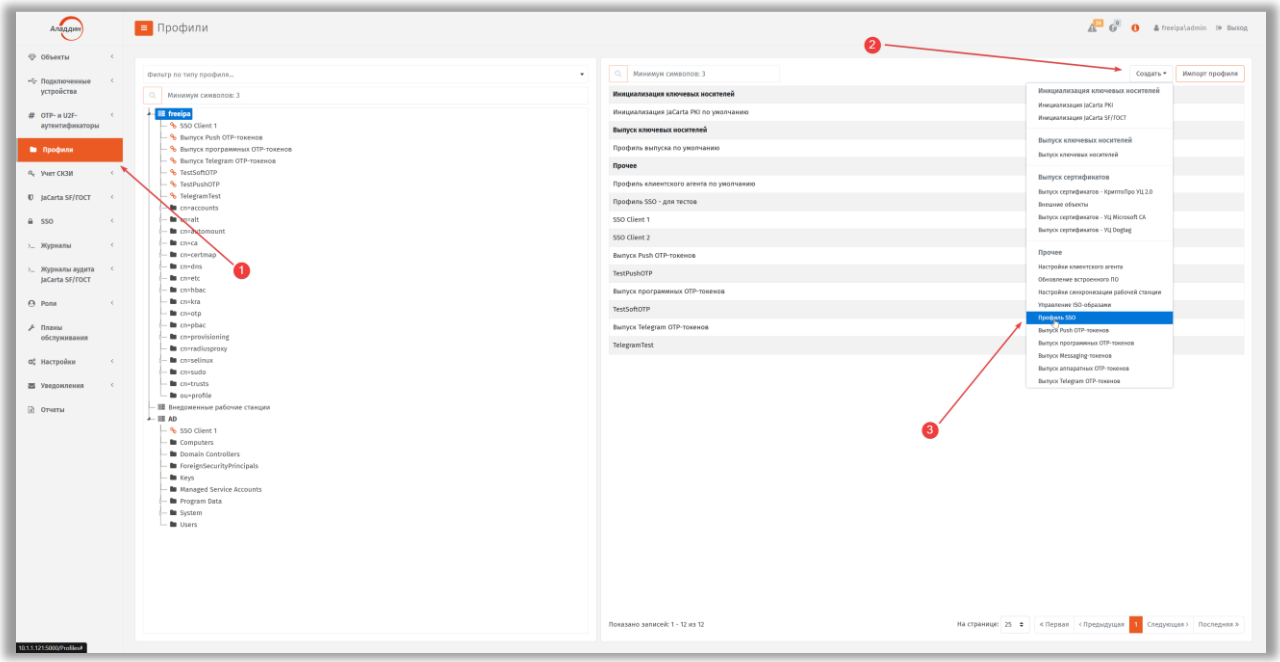

Рис. 41. Создание профиля SSO

После выполнения данных действий на экране появится форма создания нового профиля SSO. На форме необходимо заполнить следующие свойства.

- «Имя» - название профиля
- «Клиенты SSO» - список клиентов. В списке необходимо выбрать созданный клиент OIDC
- «Стратегии входа» - список стратегий, которые будут доступны пользователю при входе в один из клиентов, перечисленных в списке «Клиенты SSO»
- «Требуется ли повторная аутентификация» - необходимо выбрать значение в соответствии с политиками безопасности
- «Утверждения OIDC» - список утверждений, которые попадут в JWT токен OIDC

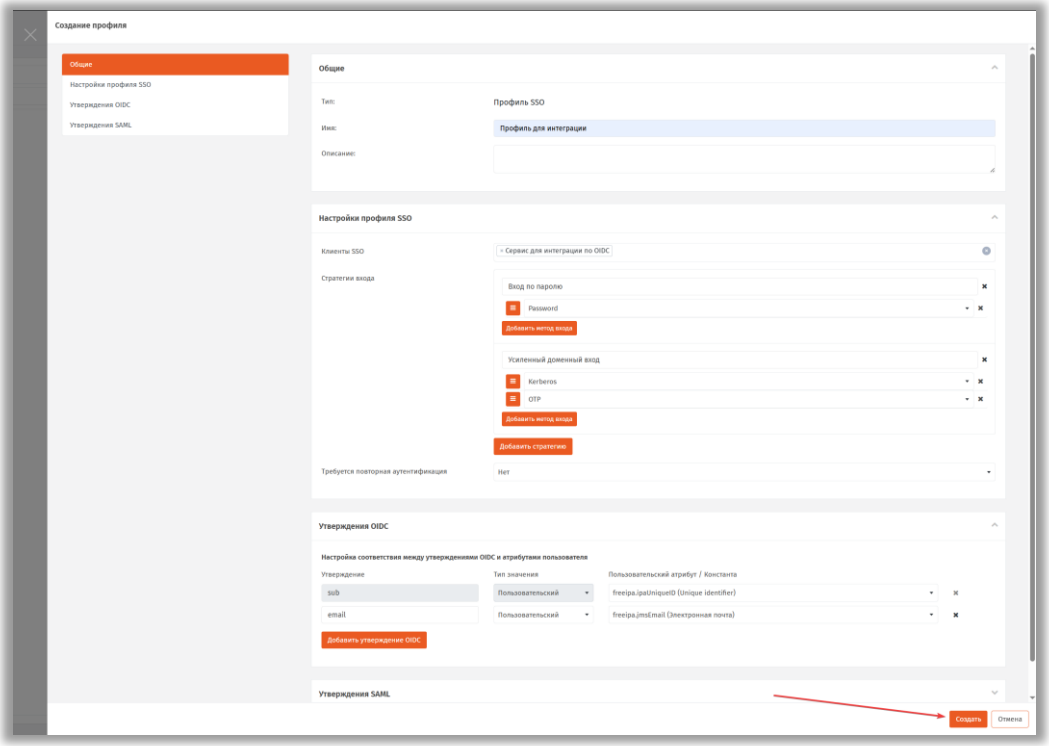

Рис. 42. Создание профиля SSO

После заполнения свойств профиля, необходимо нажать кнопку «Создать». После успешного сохранения, новый профиль появится в списке профилей.

22.1.3 Редактирование профиля SSO

Если существующий профиль SSO полностью соответствует требованиям по стратегиям входа, набору утверждений и привязке к РС, то можно не создавать новый профиль, а добавить созданный клиент OIDC в список «Клиенты SSO».

Для этого необходимо в списке профилей открыть на редактирование существующий профиль, в падающем списке «Клиенты SSO» выбрать созданный ранее клиент и нажать на кнопку «Сохранить».

22.1.4 Привязка профиля к дереву РС

Для того чтобы профиль SSO начал действовать, и сервис мог взаимодействовать с JIP по протоколу OIDC, его необходимо привязать к дереву ресурсной системы.

22.1.5 Настройка сервиса в качестве клиента OIDC

Настройка сервиса в качестве клиента OIDC должна быть выполнена согласно инструкции по интеграции с сервером OIDC для данного сервиса.

Информацию о сервере OIDC, которая может потребоваться на клиентской стороне для настройки интеграции, можно получить открыв в браузере OIDC Discovery Endpoint web-приложения JIP:

ISSUER/.well-known/openid-configuration,

где ISSUER – это url, который указан в качестве параметра (см. раздел «Издатель токена». с. 47). Данный url должен быть доступен сервису клиенту OIDC

Параметры, которые возвращаются при обращении на OIDC Discovery Endpoint соответствуют спецификации https://openid.net/specs/openid-connect-discovery-1_0.html.

Также для настройки потребуются данные, которые были указаны при создании клиента OIDC для данного сервиса, а именно:

- «ID клиента»
- «Секрет» (для клиента типа «Конфиденциальный»)

22.2 Интеграция по SAML

Для интеграции с JIP по протоколу SAML-P 2.0 внешний сервис (далее сервис) должен поддерживать такую возможность на уровне протокола взаимодействия (SAML 2.0 Core Specification, SAML 2.0 Bindings, SAML 2.0 Profiles).

В данном случае JIP выступает в качестве сервера SAML, а сервис в качестве клиента SAML.

Для интеграции необходимо выполнить следующие действия:

1. Зарегистрировать сервис в качестве клиента SAML
2. Создать новый или изменить существующий профиль SSO
3. Привязать новый профиль к дереву РС
4. Выполнить настройку сервиса в качестве клиента SAML

22.2.1 Регистрация сервиса в JIP

Для регистрации сервиса в качестве клиента SAML необходимо в Консоли управления JMS (далее портал) выполнить следующие действия:

Открыть список клиентов SAML путем выбора в навигаторе пункта «SSO» -> «Клиенты SAML». После чего нажать на кнопку «Создать» для создания нового клиента.

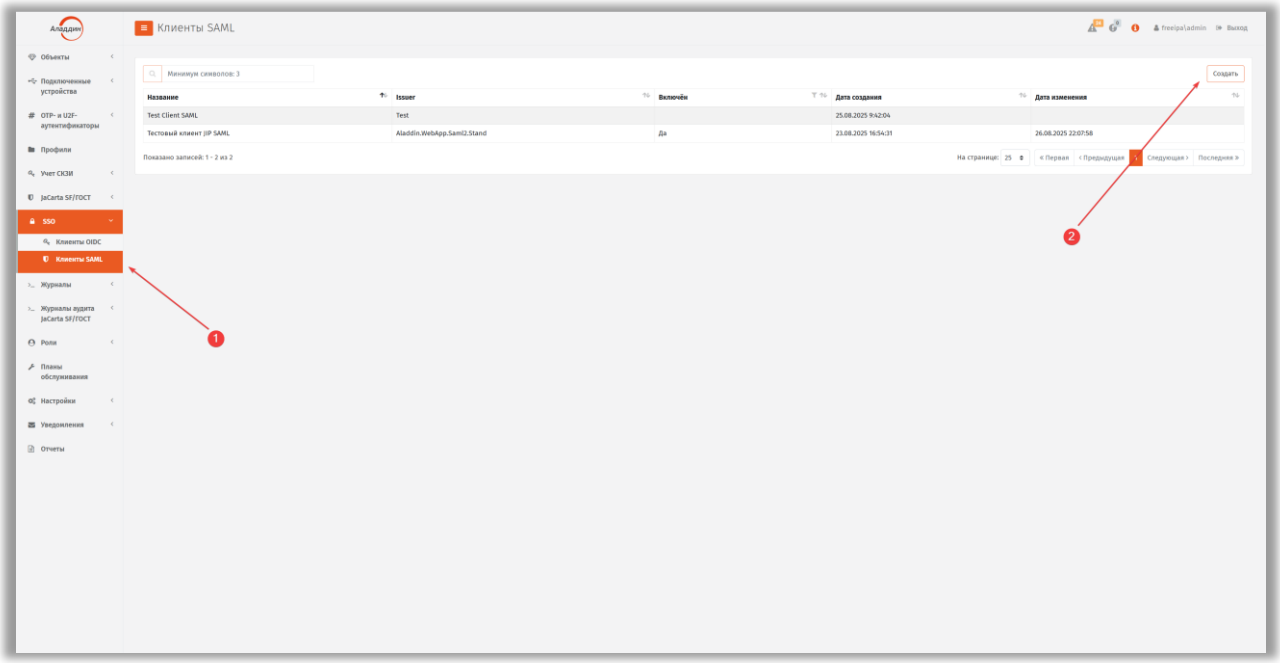

Рис. 43. Создание клиента SAML

После нажатия на кнопку «Создать» появится форма создания нового клиента SAML. На которой необходимо указать следующие обязательные значения:

- «Название» – произвольное человекочитаемое название сервиса
- «Issuer» – уникальный машиночитаемый идентификатор клиента. **Данное значение необходимо запомнить, т.к. оно понадобится при настройке интеграции на стороне сервиса.**
- «Включен» - на этапе настройки клиент можно отключить

- «Способ определения адресов клиента» – если сервис не предоставляет метаданные согласно спецификации https://docs.oasis-open.org/security/saml/v2.0/saml-metadata-2.0-os.pdf или они недоступны публично, то нужно использовать «Указать вручную», в противном случае использовать «Автоматически по метаданным»
- «Адрес ACS клиента» – адрес на стороне сервиса куда будет доставлен SAML Assertion после успешного входа, предоставляется сервисом
- «Адрес Logout клиента» – адрес на стороне сервиса куда будет перенаправлен пользователь после выхода, предоставляется сервисом

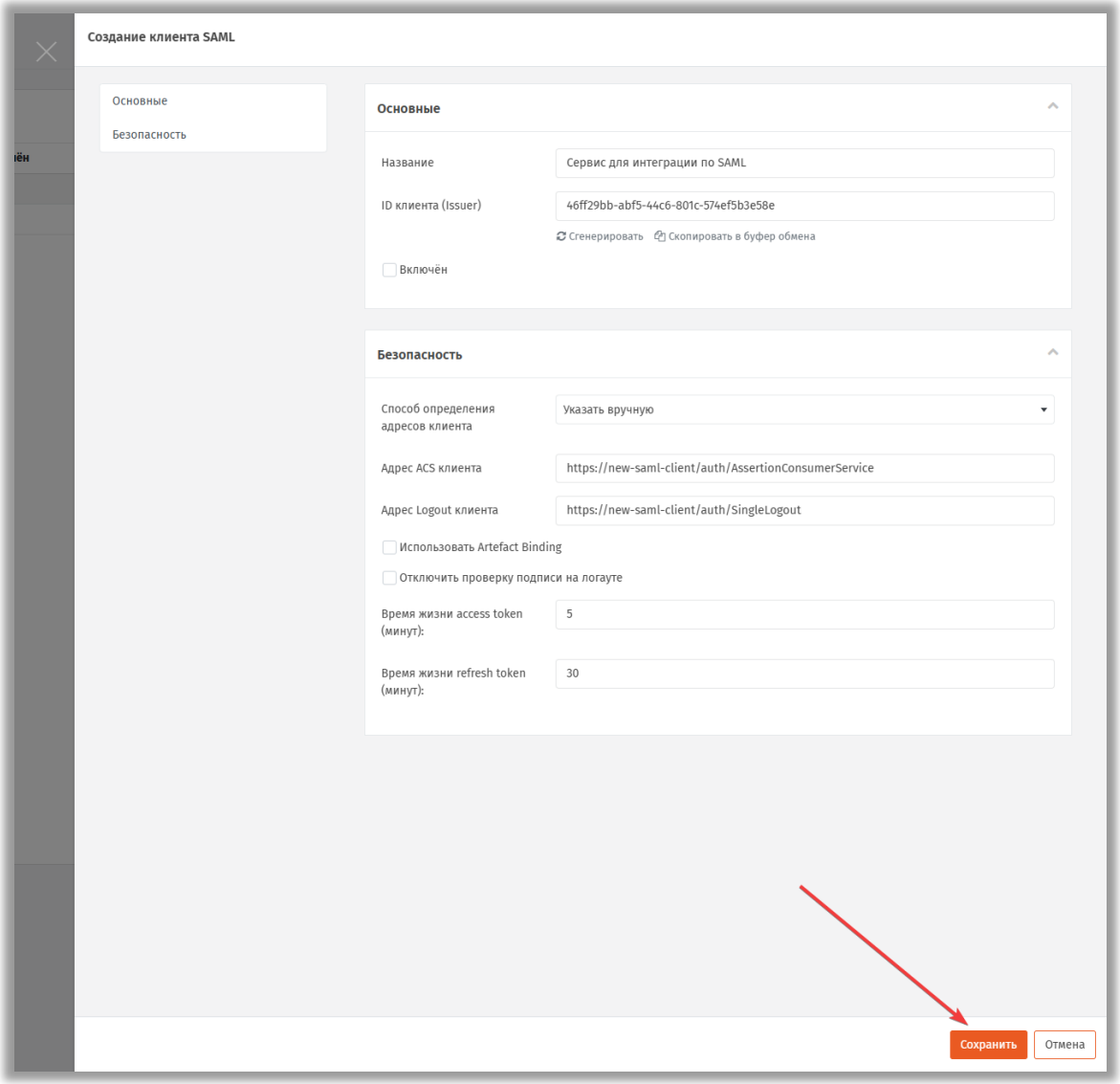

Рис. 44. Создание клиента SAML

После заполнения свойств клиента, необходимо нажать кнопку «Сохранить». После успешного сохранения, новый клиент появится в списке клиентов SAML.

22.2.2 Создание профиля SSO

Для создания профиля необходимо выбрать в навигаторе пункт «Профили», далее в области списка профилей нажать на кнопку «Создать» и в выпадающем меню выбрать пункт «Профиль SSO».

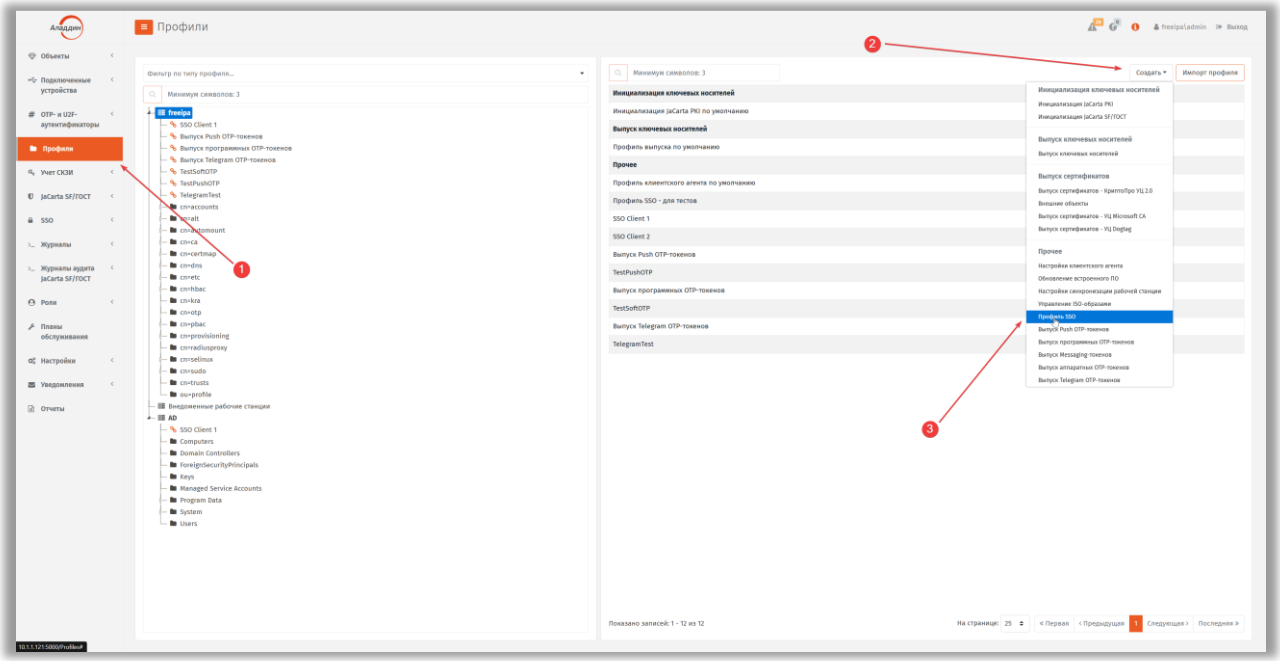

Рис. 45. Создание профиля SSO

После выполнения данных действий на экране появится форма создания нового профиля SSO. На форме необходимо заполнить следующие свойства.

- «Имя» - название профиля
- «Клиенты SSO» - список клиентов. В списке необходимо выбрать созданный клиент OIDC
- «Стратегии входа» - список стратегий, которые будут доступны пользователю при входе в один из клиентов, перечисленных в списке «Клиенты SSO»
- «Требуется ли повторная аутентификация» - необходимо выбрать значение в соответствии с политиками безопасности
- «Утверждения SAML» - список утверждений, которые попадут в SAML токен

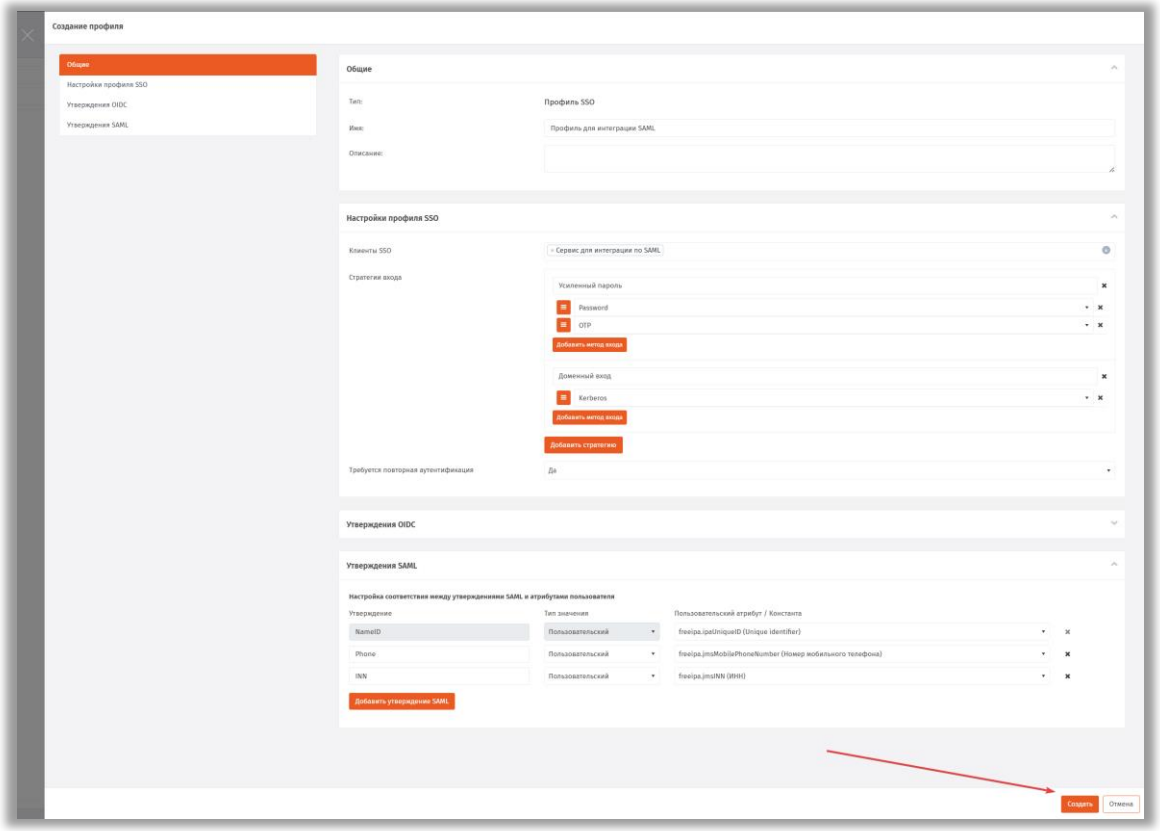

Рис. 46. Создание профиля SSO

После заполнения свойств профиля, необходимо нажать кнопку «Создать». После успешного сохранения, новый профиль появится в списке профилей.

22.2.3 Редактирование профиля SSO

Если существующий профиль SSO полностью соответствует требованиям по стратегиям входа, набору утверждений SAML и привязке к РС, то можно не создавать новый профиль, а добавить созданный клиент SAML в список «Клиенты SSO» данного профиля.

Для этого необходимо в списке профилей открыть на редактирование существующий профиль, в падающем списке «Клиенты SSO» выбрать созданный ранее клиент и нажать на кнопку «Сохранить».

22.2.4 Привязка профиля к дереву РС

Для того чтобы профиль SSO начал действовать, и сервис мог взаимодействовать с JIP по протоколу SAML, его необходимо привязать к дереву ресурсной системы.

22.2.5 Настройка сервиса в качестве клиента SAML

Настройка сервиса в качестве клиента SAML должна быть выполнена согласно инструкции по интеграции с сервером SAML для данного сервиса.

Информацию о сервере SAML, которая может потребоваться на клиентской стороне для настройки интеграции, можно получить открыв в браузере SAML Metadata Endpoint web-приложения JIP:

```
http://{HOST}/saml/metadata
```

где HOST – адрес на котором развернуто web-приложение JIP.

Параметры, которые возвращаются при обращении на JIP SAML Metadata соответствуют спецификации https://docs.oasis-open.org/security/saml/v2.0/saml-metadata-2.0-os.pdf .

Также для настройки потребуются данные, которые были указаны при создании клиента SAML для данного сервиса, а именно: **«Issuer»**

23.Настройка Kerberos

Для обеспечения работы аутентификации пользователей по Kerberos в JIP необходимо выполнить следующие действия:

- подготовить окружение для выдачи keytab-файла
- получить keytab-файл (не описывается в данном руководстве)
- зарегистрировать keytab-файл в JIP
- настроить браузер на рабочей станции пользователя

После выполнения всех действий пользователь может проходить аутентификацию в JIP по протоколу Kerberos.

Перед проверкой атуентификации на рабочей станции пользователя, после выполнения действий, рекомендуется выполнить SignOut.

23.1 Подготовка к выдаче keytab-файла

Описана подготовка на примере ресурсной системы MS AD. Подготовка к получению и получение keytab-файла для ресурсных систем других типов может отличаться.

Перед выдачей keytab-файла необходимо выполнить следующие действия:

В домене создать сервисную учетную запись, например, с именем krbuser. При создании включить опции «Запретить смену пароля пользователем» и «Срок действия пароля не ограничен».

Установить созданному пользователю в качестве SPN значение адреса web-приложения JIP в формате FQDN, например, jip-astra2.fqdn1.loc.

setspn -A HTTP/jip-astra2.fqdn1.loc krbuser

setspn -A HOST/jip-astra2.fqdn1.loc krbuser

Проверить что у пользователя в свойстве servicePrincipalName указано строго две записи HTTP/jip-astra2.fqdn1.loc и HOST/jip-astra2.fqdn1.loc, а в свойстве userPrincipalName значение krbuser@DOMAIN, где DOMAIN – это имя домена, в котором была зарегистрирована сервисная учетная запись.

23.2 Регистрация keytab-файла в JIP

Процедура регистрации keytab-файла описана в пункте 18.9.3 Регистрация keytab-файла настоящего руководства.

23.3 Настройка браузера

23.3.1 Google Chrome и Microsoft Edge

На компьютере, с которого осуществляется подключение к web-интерфейсу, в панели управления ОС Windows выберите раздел Internet options.

На закладке Security выберите зону Local intranet и нажмите на кнопку Sites.

Откроется окно Local intranet.

Нажмите на кнопку Advanced.

В открывшемся окне в поле ввода укажите полный адрес web-приложения JIP в формате FQDN (указывали при создании SPN в пункте 23.1 Подготовка к выдаче keytab-файла) и нажмите на кнопку Add, например, https://jip-astra2.fqdn1.loc.

Убедитесь, что адреса добавлены, и нажмите на кнопку Close.

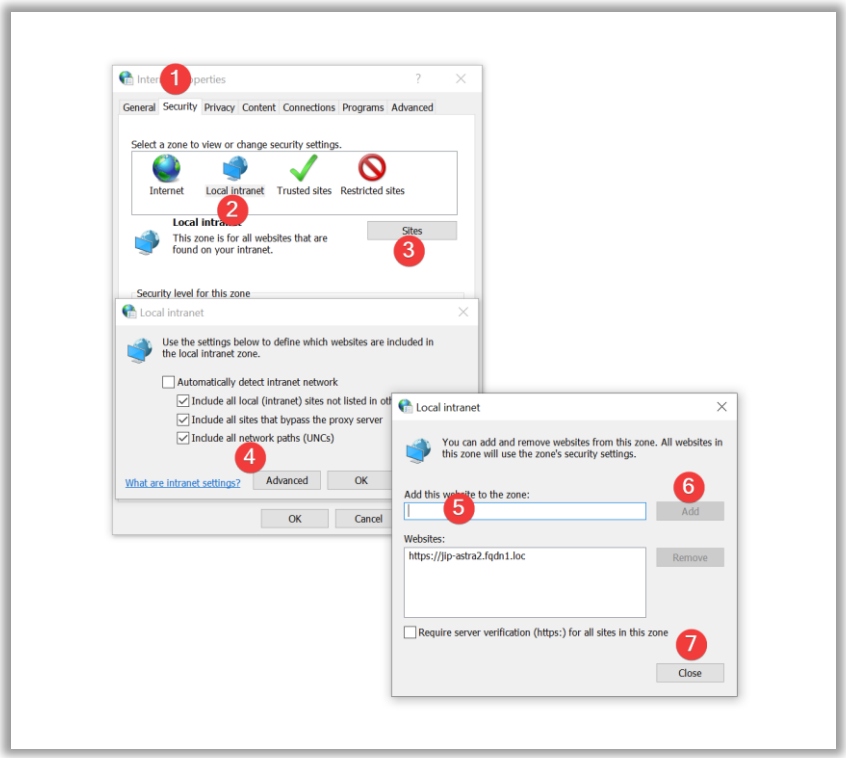

Рис. 47. Добавление адреса JIP в Local Intranet Zone

Закройте все открытые ранее окна с помощью кнопок OK.

23.3.2 Mozilla Firefox

В адресной строке браузера введите about:config и на открывшейся странице нажмите на кнопку Accept the Risk and Continue.

В строке поиска параметров введите negotiate.

В открывшемся списке параметров в полях network.negotiate-auth.delegation-uris и network.negotiate-auth.trusted-uris введите адреса web-приложения JIP в формате FQDN (указывали при создании SPN см. раздел «Подготовка к выдаче keytab-файла», с. 94), например, https://jip-astra2.fqdn1.loc.

Нажмите на значок с галочкой справа от поля, чтобы сохранить введенные адреса.

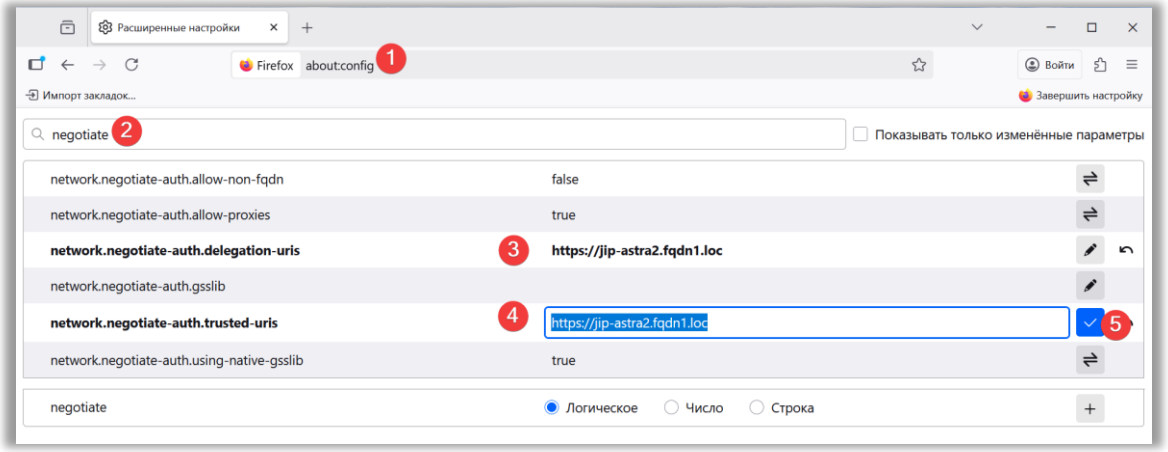

Рис. 48 – Добавление адреса JIP в разрешенные адреса для negotiate

23.4 Автоматический выбор метода Kerberos

Если в настройках включена опция проверки наличия Kerberos тикета описанная в разделе 18.9.1.3 и в 18.9.8 Управление автоопределением доступности Kerberos аутентификации (shortcut), то в процессе аутентификации пользователя будет выполнена проверка наличия Kerberos тикета и в браузере будет показана следующая страница.

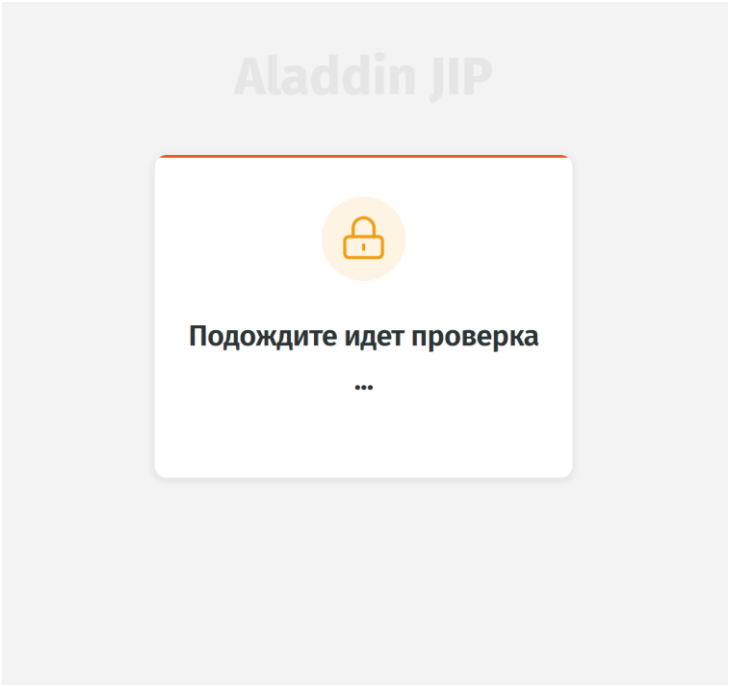

Рис. 49 – Страница проверки наличия Kerberos тикета

По итогам проверки JIP может автоматически выбрать стратегию для прохождения без показа диалога выбора стратегии. Автоматический выбор стратегии произойдет в следующих случаях:

- Если проверка показала, что тикет есть. Для клиента, в который выполняется вход, сконфигурирована строго одна стратегия с Kerberos. Будет автоматически выбрана единственная стратегия с Kerberos.

- Если проверка показала, что тикета нет. Для клиента, в который выполняется вход, сконфигурирована строго одна стратегия без Kerberos. Будет автоматически выбрана единственная стратегия без Kerberos.

24.Специфика работы в мультидоменной среде

Существует особенность в части работы с профилями SSO в мультидоменной среде.

При конфигурировании утверждений OIDC и SAML в профиле нужно следить чтобы используемые пользовательские атрибуты были из той ресурсной системы, к которой привязан данный профиль. В противном случае, можно сконфигурировать утверждения таким образом, что при входе пользователя их значения будут пустыми, и пользователь получит ошибку входа.

Например, имеем две ресурсные системы (два домена) FQDN1 и FQDN2. Есть один профиль SSO1, в котором в качестве источника для всех утверждений используются свойства пользователя из FQDN1. Данный профиль привязан к обеим ресурсным системам и действует на всех пользователей из обеих РС.

Пользователь FQDN1\test успешно проходит аутентификацию и утверждения возвращаются на клиент в виде JWT токена.

Пользователь FQDN2\auto успешно проходит аутентификацию, но при попытке получить утверждения согласно настройкам из профиля JIP получит все утверждения с пустыми значениями, т.к. пользователь FQDN2\auto не обладает свойствами из ресурсной системы FQDN1. И вход пользователя завершится ошибкой.

Для решения данной проблемы рекомендуется использовать отдельный SSO профиль на каждую ресурсную систему.

Также в настоящее время JIP не поддерживает работу со связанными ресурсными системами JMS.

25.Диагностика

Сервер JIP, web-приложение JIP и web-консоль сервера JIP ведут файлы лога. Найти их можно в директориях:

- /var/log/aladdin/jip-engine – логи сервера JIP
- /var/log/aladdin/jip-web – логи web-приложения JIP
- /var/log/aladdin/jip-wsc – логи web-консоли сервера JIP

Действия, связанные с работой плагинов JIP, фиксируются в соответствующих логах JMS.

25.1 Диагностика проблем с Kerberos

В случае возникновения проблем при аутентификации по Kerberos, нужно открыть файла лога /var/log/aladdin/jip-engine/KerberosService.log и поискать по «NTLM authentication is not supported».

Данная ошибка говорит о том, что при попытке аутентифицироваться браузер отправляет NTLM тикет вместо Kerberos тикета. Такое может происходить если неверно выполнено конфигурирование Kerberos.

Для устранения данной проблемы необходимо обратиться к разделу 23 Настройка Kerberos данного руководства и проверить все настройки, описанные в нем.

25.2 Диагностика событий аутентификации через аудит

Все события аутентификации регистрируются подсистемой аудита.

Где смотреть логи:

- Отдельный файл аудита: /var/log/aladdin/jip-web/audit/auth-audit.log (только события аудита, формат JSON)
- Общий лог приложения: /var/log/aladdin/jip-web/AladdinJIP.Web.log (все события, включая аудит)
- Консоль: при запуске приложения

Формат записи: JSON (каждая строка - отдельное событие)

25.2.1 Перечень аудируемых событий

Табл. 9 – Перечень аудируемых событий

Событие	Уровень	Сообщение	Когда записывается
LoginSuccess	Information	Выполнен вход	После успешной аутентификации и выдачи токенов
LoginFailed	Warning	Ошибка входа пользователя	При ошибке на любом этапе входа
AuthenticationMethodCompleted	Information	Пройден метод входа	После успешного прохождения отдельного метода (Password, OTP, Push, Messaging, Kerberos)
LogoutSuccess	Information	Выполнен выход	После успешного выхода пользователя
LogoutFailed	Warning	Ошибка выхода пользователя	При ошибке выхода

25.2.2 Поля аудит-записей

Основные поля JSON:

- Timestamp – дата и время события
- Level – уровень логирования
- Properties – объект с полями аудита

25.2.2.1.1 Поля аудита (объект Properties)

Табл. 10 – Поля аудита (объект Properties)

Контроллер API	Точка доступа активна	Используемый порт
Поле	Описание	Пример
Subsystem	Идентификатор подсистемы	"AUTH"
Message	Текст сообщения	"Выполнен вход"
Username	Имя пользователя	"fqdn1\\vmatveev"
ClientId	Идентификатор SSO-клиента	"Aladdin.WebApp.Saml2.Localhost"
Action	Тип операции	"Login" или "Logout"
Result	Результат операции	"Success" или "Error"
ProfileId	ID профиля SSO	GUID или null
StrategyId	ID стратегии входа	GUID или null
AuthMethod	Метод аутентификации	"Password", "OTP", "Push", "Messaging", "Kerberos" или комбинация
Error	Описание ошибки	Присутствует только в *Failed событиях

25.2.2.1.2 Пример записи успешного входа:


```
{
  "Timestamp": "2025-10-17T10:36:48.6105109+03:00",
  "Level": "Information",
  "Properties": {
    "Subsystem": "AUTH",
    "Message": "Выполнен вход",
    "Username": "fqdn1\\john",
    "ClientId": "Aladdin.WebApp.Saml2.Prod",
    "Action": "Login",
    "Result": "Success",
    "ProfileId": "77ba2f7a-2174-423e-1519-4e4e7f67235b",
    "StrategyId": "b24ca99e-eedf-4001-2b76-99e06cfdc837",
    "AuthMethod": "Password"
  }
}
```

26.Аутентификация

26.1 Идентификация пользователя

Процесс идентификации пользователя заключается в получении данных конкретного пользователя из централизованного хранилища (базы данных JIP) на основе предоставленной идентификационной информации.

26.1.1 Источники идентификационной информации

В качестве источника идентификационной информации могут выступать:

- Прямой ввод пользователем - логин в различных форматах.
- UPN (User Principal Name) из тикета Kerberos – автоматически полученные учетные данные.

26.1.2 Этапы процесса идентификации

Процесс идентификации включает следующие этапы:

- Получение идентификационной информации и определение формата.
- Поиск соответствующей записи в базе данных JIP.
- Извлечение полных данных пользователя при успешном совпадении.

Таким образом, система сопоставляет предоставленные идентификационные данные с соответствующей учетной записью в базе данных для последующей авторизации и предоставления доступа.

Данные о всех пользователях, в том числе AccountName (имя аккаунта пользователя), DomainName (имя домена ресурсной системы пользователя) и NetBIOSName (NetBIOS-имя ресурсной системы пользователя), хранятся в базе данных JIP и попадают туда в процессе периодической синхронизации из JMS.

При входе по логину и паролю пользователь вначале проходит аутентификацию в JAS и, в случае успешной аутентификации, выполняется поиск пользователя в базе данных JIP.

При входе по Kerberos обращение в JAS не выполняется, а сразу выполняется поиск пользователя по UPN из тикета Kerberos в базе данных JIP.

26.1.3 Логика идентификации пользователя

Сервер JIP, при получении идентификационных данных выполняет логику по идентификации пользователя в JIP. Вначале определяется форма идентификационных данных, после чего, в зависимости от формата, определяются значения для поиска пользователя в базе данных JIP.

Если идентификационные данные содержат символ «@», то будет выполнена проверка формата UPN. Если данные не подходят под формат UPN, идентификация завершится ошибкой.

Если идентификационные данные содержат символ «\», то будет выполнена проверка формата DOMAIN\username. Если данные не подходят под формат, то идентификация завершится ошибкой.

Если идентификационные данные не содержат символ «@» и символ «\», то будет выполнен дополнительный анализ.

26.1.3.1 Проверка формата UPN

Для определения принадлежности идентификационных данных к формату UPN применяется регулярное выражение

^([a-zA-Z0-9._%+-]+)@([a-zA-Z0-9.-]+\.[a-zA-Z]{2,})$
Значение считается UPN, если оно удовлетворяет следующим условиям:

Локальная часть (до символа «@»)

Может содержать символы: латинские буквы (a-z, A-Z), цифры (0-9), точку «.», нижнее подчёркивание «_», знак процента «%», плюс «+», дефис «-». Должна содержать хотя бы один символ из указанных

Разделитель

Обязательный символ «@» между частями.

Доменная часть (после символа «@»)

- Состоит из двух компонентов, разделённых точкой.
- Имя домена/хоста: может содержать латинские буквы, цифры, точку «.» и дефис «-»
- Обязательная точка «.» как разделитель
- Доменная зона: должна содержать только латинские буквы и состоять минимум из 2 символов.

Общие требования

- Значение должно полностью соответствовать шаблону (от начала до конца строки)
- Не допускаются символы, не указанные в шаблоне
- Русские и другие Unicode-символы не поддерживаются

Примеры

Валидные UPN: user.name@domain.com, john_doe+tag@sub-domain.org, test-user%123@company.co.uk

Невалидные UPN: user@localhost, name@domain.c, user@.com, @domain.com, русский@mail.ru.

26.1.3.2 Проверка формата DOMAIN\username

Если полученное значение идентификационных данных не было распознано как валидный UPN, то выполняется проверка на соответствие формату **DOMAIN\username.**

Выполняется разделение значения на строки с использование разделителя: обратный слеш - «\».

Если после разделения получили не две строки, то считаем, что переданное значение не подпадает под формат DOMAIN\username.

Если получили строго две непустых строки считаем вторую строку значением AccountName. По первой строке определяем имя домена в формате FQDN или NetBIOS-имя.

Для определения принадлежности к FQDN формату используется регулярное выражение **^[a-zA-Z0-9.-]+\.[a-zA-Z]{2,}$**

Данное регулярное выражение определяет следующие требования

- Состоит из двух компонентов, разделённых точкой.
- Имя домена/хоста: может содержать латинские буквы, цифры, точку «.» и дефис «-»
- Обязательная точка «.» как разделитель
- Доменная зона: должна содержать только латинские буквы и состоять минимум из 2 символов.

Примеры

Валидные FQDN: example.com, sub.domain.org, company.co.uk, my-site123.com, localhost

Невалидные FQDN: example.c, example.12, example, .com, -example.com, example-.com, русский.рф.

26.1.3.3 Анализ формата идентификационных данных

После анализа формата переданных идентификационных данных получаем следующую картину:

Если идентификационные данные распознаны как валидный UPN, то мы имеем AccountName и DomainName для поиска.

Если идентификационные данные распознаны в формате DOMAIN\username, то в зависимости от представления части DOMAIN, мы имеем либо пару AccountName и DomainName, либо пару AccountName и NetBIOSName.

Если идентификационные данные не подпадают ни под один формат, то JIP считает что передали AccountName без домена. И если не задан 18.3.6Домен по-умолчанию, то процесс идентификации завершится ошибкой. Если домен по-умолчанию задан, то он будет использован как значение DomainName для аутентификации в JAS и поиска пользователя в базе данных JIP.

26.1.3.4 Поиск в базе данных JIP

Поиск в базе данных JIP, в зависимости от наличия значений AccountName, DomainName и NetBIOSName, выполняется следующим образом:

Для пары AccountName и DomainName поиск пользователя идет по колонкам AccountName и AccountSystemDomainName в таблице Users. Поиск регистронезависимый.

Для пары AccountName и NetBIOSName поиск пользователя идет по колонкам AccountName и AccountSystemNetBiosName в таблице Users. Поиск регистронезависимый.

26.1.3.5 Завершение процесса идентификации

Если в результате поиска был найден строго один пользователь, то процесс идентификации успешно завершается, в противном случае будет ошибка поиска пользователя.

26.2 SSO (Single-Sign-On)

В случае успешной аутентификации пользователя в одно приложения web-приложение JIP выдает SSO cookie, в которой содержится следующая информация:

- Идентификатор пользователя
- Список блоков, в каждом блоке содержится список методов аутентификации, которые прошел пользователь в рамках одного процесса аутентификации
- Служебная информация.

Если пользователь переходит в браузере в другой клиент JIP при наличии SSO cookie, и в соответствии с профилем он может туда зайти, то в зависимости от установленного значения «Требуется повторная аутентификация» поведение будет следующими:

- «Требуется повторная аутентификация» = Да. Пользователь будет проходить аутентификацию заново.
- «Требуется повторная аутентификация» = Нет. Пользователь сразу войдет в приложение при условии, что он уже проходил набор методов, который требуется для входа в данное приложение.

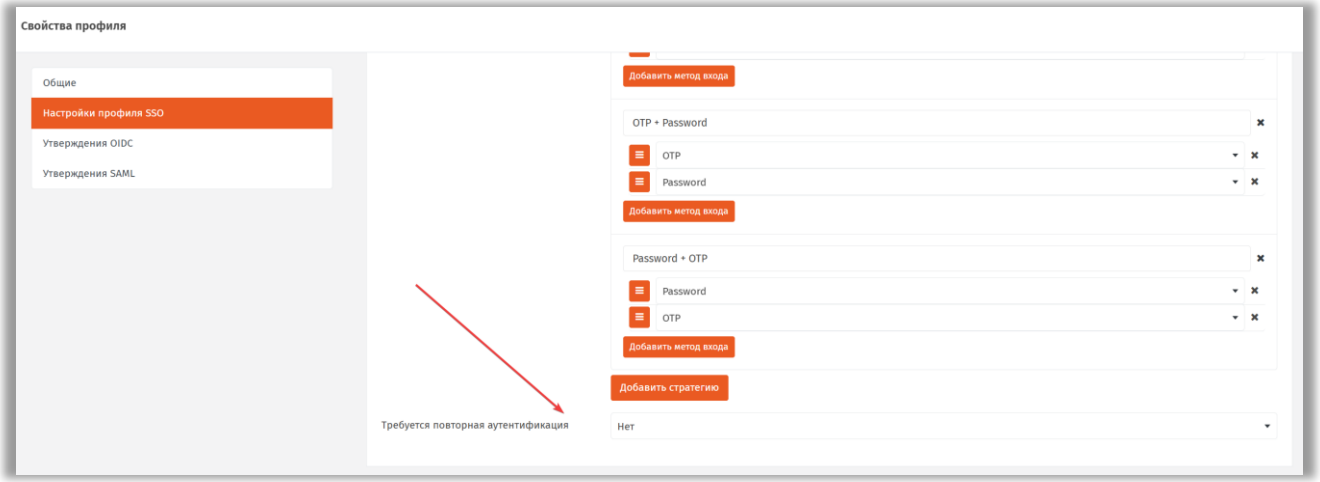

Рис. 50. Настройка «Требуется повторная аутентификация» в профиле SSO

SSO работает между клиентами OIDC, между клиентами SAML так и кросс-протокольно. Иными словами, можно зайти в клиент SAML, а потом по выданной SSO cookie войти в клиент OIDC, и наоборот, зайти в клиент OIDC, а потом по выданной SSO cookie в клиент SAML.

26.3 SLO (Single Logout)

SLO (Single Logout), или единый выход из системы, – это функция в рамках стандартов единой аутентификации (таких как SAML 2.0, OIDC), которая позволяет пользователю выйти из всех приложений и сервисов, в которые он был авторизован, одним действием. Проще говоря, это механизм "выйти везде сразу".

26.3.1 OIDC SLO

Участники процесса SLO:

- Пользователь
- Браузер
- Инициирующий Клиент (RP1) - клиент, в котором пользователь нажал "Выйти"
- Другие Клиенты (RP2, RP3...) - остальные приложения, в которые входил пользователь
- OIDC Provider - JIP

Последовательность выполнения единого выхода следующая:

Инициализация выхода (на стороне клиента)

Пользователь нажимает "Выйти" в любом из клиентов (например, RP1)

RP1 перенаправляет браузер на конечную точку выхода JIP (end_session_endpoint), передавая в параметрах:

- id_token_hint: Токен ID текущей сессии (для идентификации пользователя).
- post_logout_redirect_uri: URL, на который JIP должен перенаправить пользователя после завершения выхода.
- state: Строка для защиты от CSRF.

Валидация и подготовка к выходу (на стороне JIP)

- JIP проверяет id_token_hint, идентифицирует пользователя
- JIP определяет все клиенты, в которые входил пользователь, в рамках текущей сессии (RP1, RP2, RP3...)
- JIP отзывает все токены согласно настройками клиента
- JIP удаляет SSO cookie

Выполнение Back-Channel Logout

JIP запускает параллельные server-to-server запросы ко всем клиента (RP1, RP2, RP3...), у которых сконфигурирован backchannel_logout_uri.

Предполагается что каждый клиент:

- Проверяет подпись logout_token
- Извлекает sid (идентификатор сессии)
- Уничтожает локальную сессию и свою

Выполнение Front-Channel Logout

После выполнения всех back-channel logout, JIP:

- Генерирует HTML-страницу со скрытыми iframe
- Включает в нее запросы на frontchannel_logout_uri всех клиентов, для которых они заданы в настройках клиента
- Отправляет эту страницу в браузер пользователя
- Браузер автоматически загружает все iframe'ы, тем самым вызывая frontchannel_logout_uri адреса клиентов.

Завершение флоу

После загрузки всех iframe'ов (или таймаута) JIP перенаправляет пользователя на post_logout_redirect_uri клиента с которого был инициирован SLO.

26.3.2 SAML SLO

Полноценный SLO в данный момент не поддерживается для SAML. Выход будет выполнен только из клиента, из которого был инициирован. SSO cookie при этом будет удалена. Другие клиенты SAML и клиенты OIDC продолжат свою работу до первого обращения в JIP.

26.3.3 Кросс-протокольный SLO

В данный момент не поддерживается. Выход из клиента OIDC никак не влияет на клиентов SAML, ровно, как и выход из клиентов SAML никак не влияет на клиентов OIDC. Другие клиенты SAML и клиенты OIDC продолжат свою работу до первого обращения в JIP.

27.Поиск и устранение неисправностей

27.1 Ошибка проверки сертификатов подписи токенов OIDC/SAML. Сервер не стартует

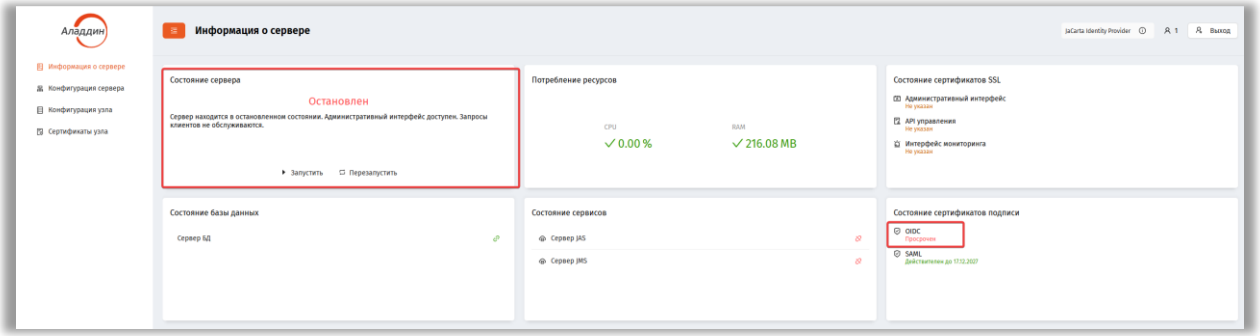

Рис. 51 – Стартовая страница web-консоли сервера JIP со статусом «Остановлен»

Необходимо заменить сертификат на рабочий или отключить поддержку протокола OIDC/SAML и после этого сделать мягкий старт сервера через Web UI WSC.

Через WSC это можно сделать следующим образом.

Меняем состояние чекбокса на "выключен" или обновляем сертификат:

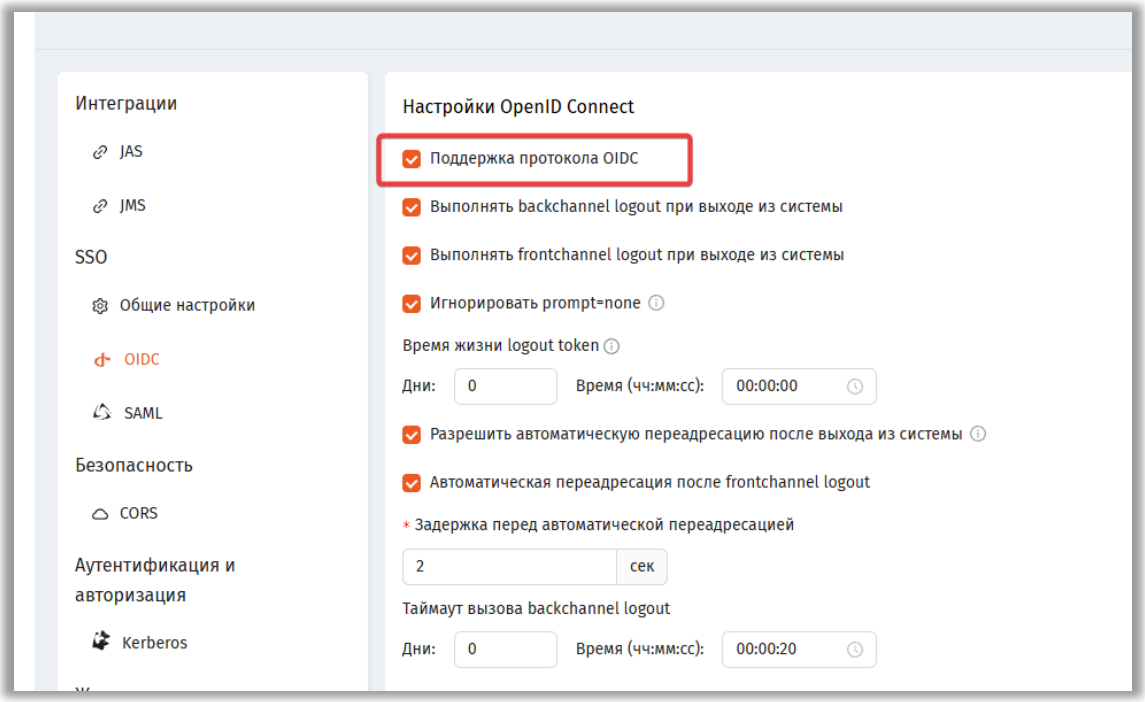

Рис. 52 – Настройки сертификата в web-консоли сервера JIP

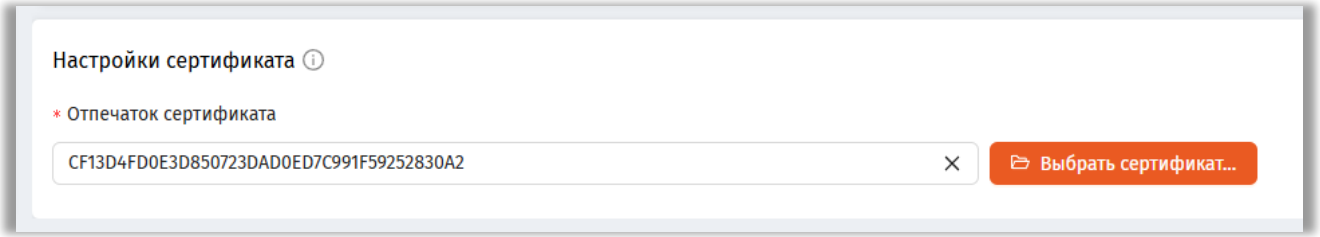

Рис. 53 – Замена сертификата в web-консоли сервера JIP

После замены сертификата убедимся, что сертификат удовлетворяет требованиям:

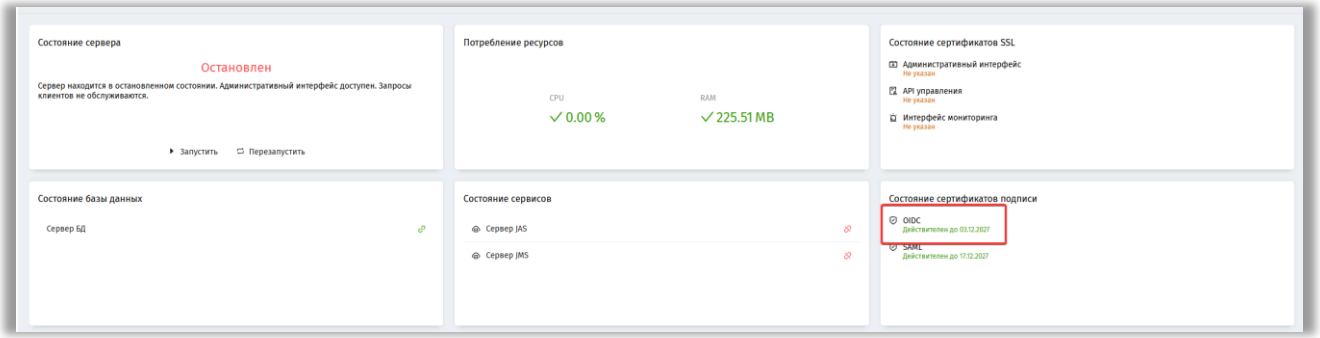

Рис. 54 – Проверка статуса сертификата в web-консоли сервера JIP

После этого можно запустить сервер:

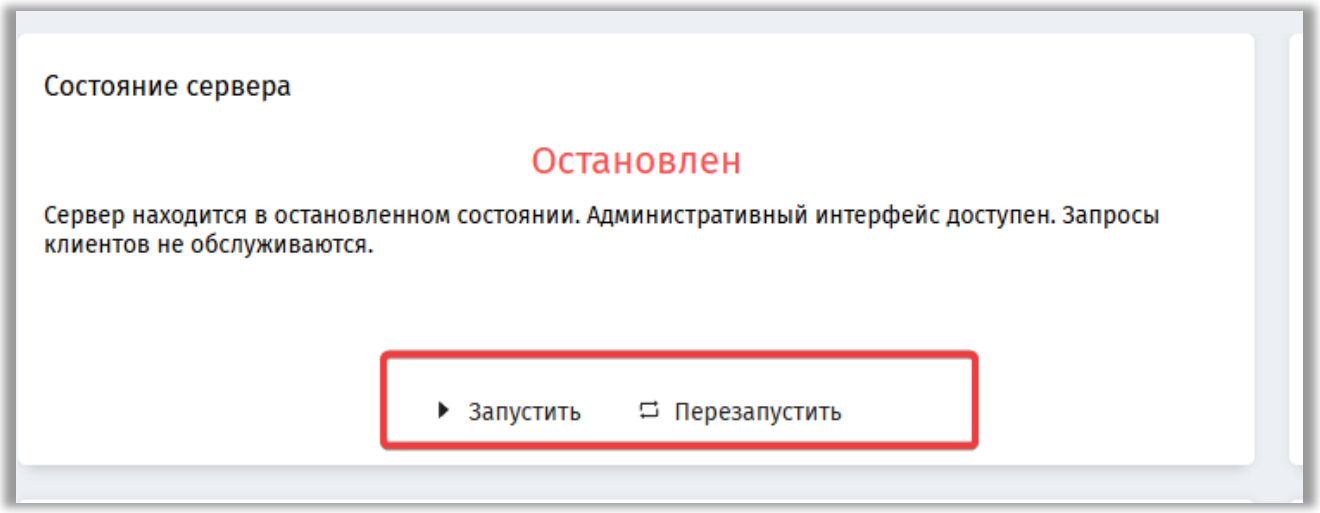

Рис. 55 – Запуск сервера через web-консоль сервера JIP

Ждем, когда сервер запустится и проверяем, что он запустился без ошибок:

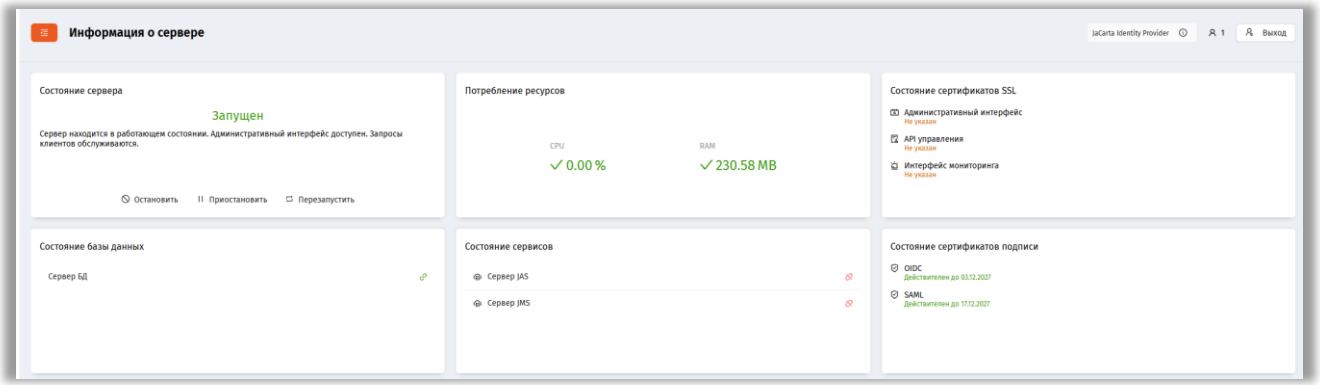

Рис. 56 – Стартовая страница web-консоли сервера JIP со статусом «Запущен»

или физически перезапустить сервер:

```
systemctl restart aladdin-jip-engine
```

28.Работа с web-консолью сервера JIP

При открытии консоли страница принимает следующий вид.

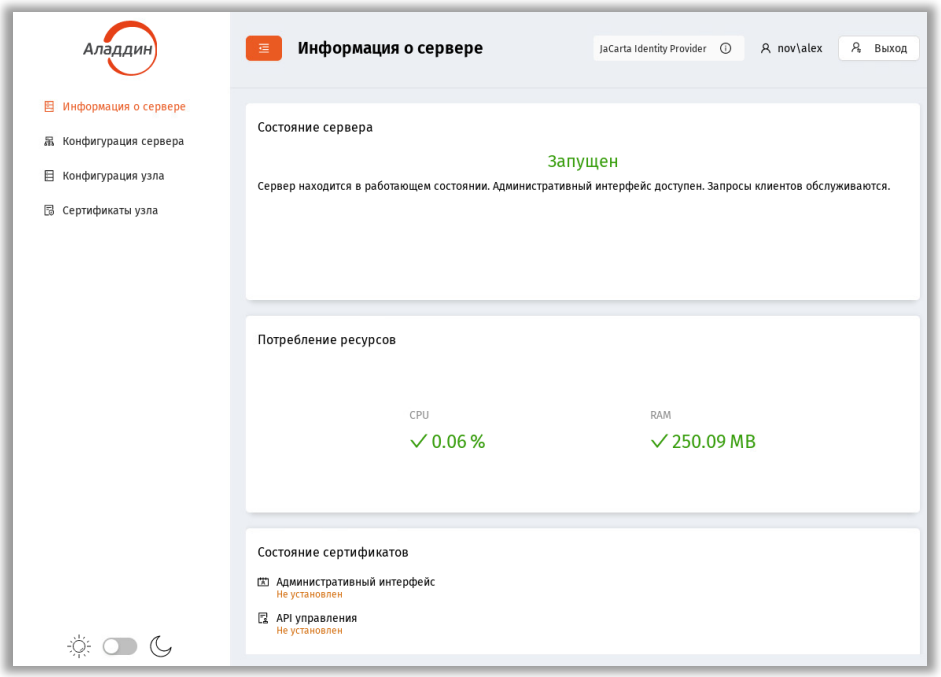

Рис. 57 – Стартовая страница web-консоли сервера JIP

Ниже описано назначение основных разделов web-консоли сервера JIP (табл. 11).

Табл. 11 – Описание разделов web-консоли сервера JIP

Раздел	Описание и ссылка на соответствующий подраздел
Информация о сервере	Раздел консоли отображает статус сервера JIP, а также отображает базовые настройки, подробнее см. «Информация о сервере», с. 108).
Конфигурация сервера	Раздел консоли содержит базовые настройки сервера JIP и параметры его интеграции с остальными компонентами ПО JMS.Подробнее см. «Конфигурация сервера», с. 109.
Конфигурация узла	Раздел консоли содержит настройки защищённого подключения к серверу JIP. Подробнее см. «Конфигурация узла», с. 121
Сертификаты узла	Пункт позволяет отобразить сертификаты, установленные для данного узла, а также установить новые сертификаты. Подробнее об установке сертификата см. раздел «Добавление сертификата в JIP», с. 74

28.1 Информация о сервере

Раздел **Информация о сервере** web-консоли выглядит следующим образом (см. рис. 58).

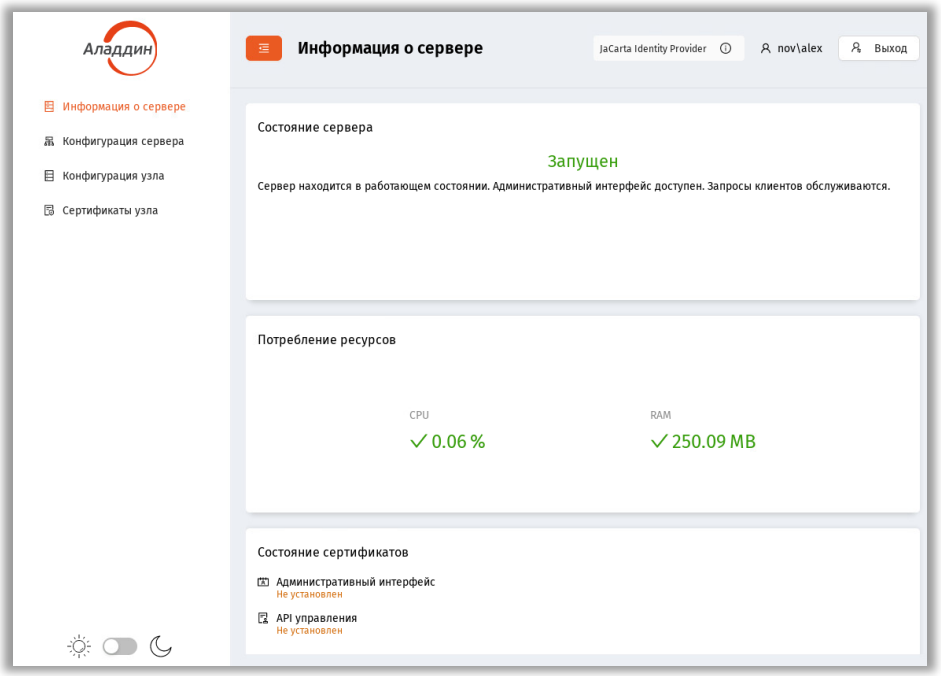

*Рис. 58 – Раздел **Информация о сервере***

Раздел **Информация о сервере** содержит следующие элементы (см. табл. 12).

Табл. 12 – Раздел Информация о сервере

Фрейм	Описание
Состояние сервера	Отображает состояние сервера JIP на текущий момент. В рабочем состоянии сервера отображается статус «Запущен».
Потребление ресурсов	Отображает занимаемую сервером оперативную память и потребление ресурсов процессора
Состояние сертификатов	Отображает статус SSL-сертификатов (установлен/не установлен) для защищённого подключения (если применяется) к различным API-интерфейсам сервера JIP.

28.2 Конфигурация сервера

Раздел **Конфигурация сервера** выглядит следующим образом.

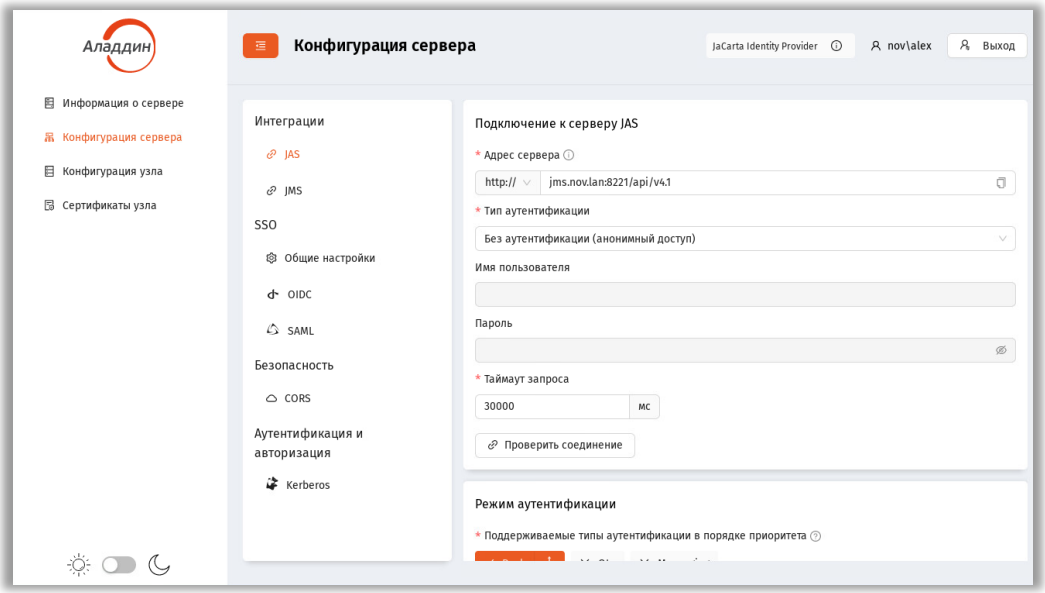

*Рис. 59 – Страница раздела **Конфигурация сервера***

Раздел содержит следующие элементы (см. табл. 13).

Табл. 13 – Элементы раздела Конфигурация сервера

Элемент интерфейса	Описание
<Секция> **Интеграции**	
JAS	Настройки интеграции с сервером JAS и режимы взаимодействия с ним, подробнее см. «Интеграция с сервером JAS», с. 110.
JMS	Настройки интеграции с сервером JMS, подробнее см. «Интеграция с сервером JMS», с. 112.
<Секция> **SSO**	
Общие настройки	Общие настройки SSO, подробнее см. «Общие настройки SSO», с. 113
OIDC	Настройки протокола OIDC, подробнее см. «Настройки OIDC», с. 114
SAML	Настройки протокола SAML, подробнее см. «Настройки SAML», с. 116
<Секция> **Безопасность**	
CORS	Настройки механизма CORS, подробнее см. «Настройки CORS», с. 118

Элемент интерфейса	Описание
<Секция> **Аутентификация и авторизация**	
Kerberos	Настройки протокола Kerberos, подробнее см. «Настройки Kerberos», с. 120

28.2.1 Интеграция с сервером JAS

Страница настройки интеграции с сервером **JAS** выглядит следующим образом.

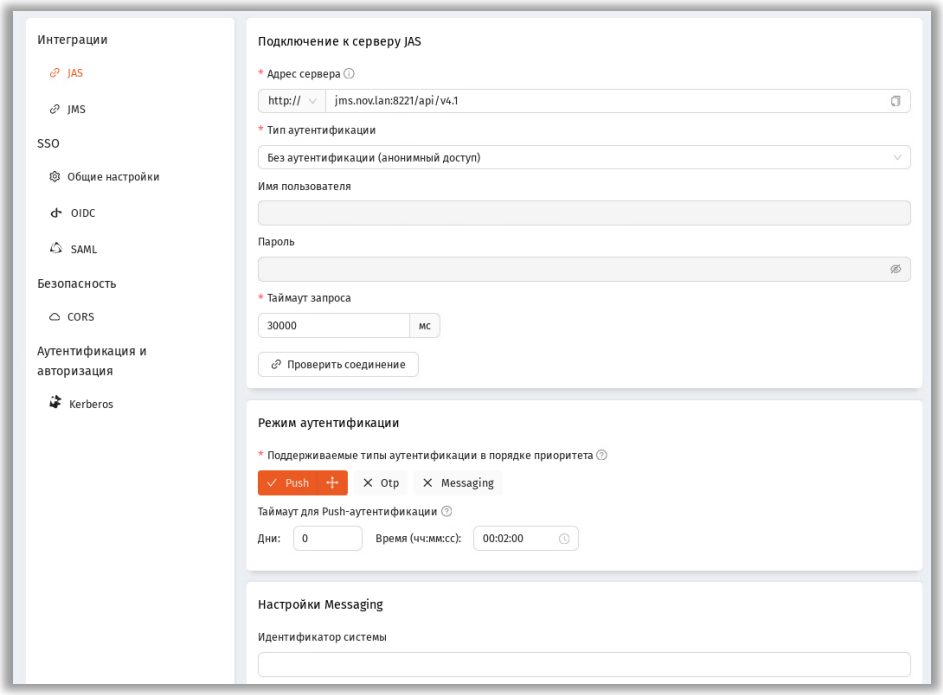

*Рис. 60 – Страница настройки интеграции с сервером **JAS***

Выполните настройку, руководствуясь Табл. 14.

Табл. 14 – Настройки интеграции с сервером JAS

Настройка	Описание
<Фрейм> **Подключение к серверу JAS**	

Адрес сервера	Укажите в данном поле адрес в следующем формате
	`https://<FQDN-имя сервера>:<порт>/<путь к API>`
	где <FQDN-имя сервера> – полное доменное имя (FQDN) сервера JAS, например, srv01.test.com
	Например:
	`"http://srv01.test.com:8221/api/v4.1"`

Настройка	Описание
Тип аутентификации	Тип аутентификации на сервере JAS. Допустимые значения: • **Без аутентификации (анонимный доступ)** – значение по умолчанию, аутентификация отключена (none); • **Basic аутентификация (логин + пароль)** базовая http-аутентификация (пароль и логин передаются в теле запроса),
Имя пользователя	Имя пользователя, от имени которого сервер JIP будет подключаться к серверу JAS (по интерфейсу **AuthenticationService**). Поле доступно только при выбранном значении **Basic** в поле **Тип аутентификации** (выше)
Пароль	В случае **Basic аутентификации (логин + пароль)** в поле указывается пароль, определенный в конфигурации сервера JAS для интерфейса **AuthenticationService** (подробнее см. раздел «Настройка сетевых программных интерфейсов сервера JAS» в руководстве по установке и настройке JAS [4])
Таймаут запроса	Таймаут ожидания ответа сервера JAS в миллисекундах
<кнопка> **Проверить соединение**	Для проверки корректности указанных параметров нажмите кнопку **Проверить соединение**. При успешном соединении отобразится уведомление «Соединение успешно установлено!»
<Фрейм> **Режим аутентификации**	
Поддерживаемые типы аутентификации в порядке приоритета	Настройка определяет набор и приоритет типов аутентификации по OTP (Messaging, Push или OTP). Используется в дальнейшем в качестве глобальной настройки при формировании **Стратегии входа** в **Профиле SSO** в *Консоли управления JMS*. Порядок настройки см. в разделе «Выбор приоритета OTP для JAS», с. 26
Таймаут для Push-аутентификации	Максимальное время ожидания реакции на Push-аутентификацию от пользователя. (Поля задания таймаута **Дни** и **Время (чч:мм:сс)**). Значение по умолчанию: **2** минуты Рекомендуется устанавливать время большее, чем задано по умолчанию, поскольку подтверждение запроса происходит на мобильном устройстве.
<Фрейм> **Настройки Messaging**	
Идентификатор системы	Идентификатор внешней системы, в которой будут искаться пользователи при аутентификации по Messaging. **Примечание.** Идентификатор должен совпадать с идентификатором в поле **Внешняя система** на вкладке **Параметры выпуска** соответствующего профиля выпуска Messaging-токенов (см. руководство по функциями управления JMS [3], раздел «Настройка профиля выпуска Messaging-токенов») См. Также параметр инициализационного файла *MessagingSystemId*

Настройка	Описание
Время жизни кода	Время жизни (в секундах) для одноразового пароля (One-Time Password), в течение которого ответ пользователя будет актуальным . См. Также параметр инициализационного файла *MessagingTtl*
Таймаут между попытками аутентификации	Таймаут (в миллисекундах) между попытками аутентификации посредством Messaging-токена. Параметр применяется непосредственно к серверу JAS, который на его основе принимает решение о возможности приёма попытки аутентификации. При попытке аутентификации, произошедшей до истечения указанного таймаута, возникает ошибка аутентификации. См. Также параметр инициализационного файла *MessagingRetryDelay*
Текст сообщения	Текст, который будет отправляться в SMS пользователю вместе с кодом аутентификации для Messaging. Например "Код аутентификации для входа в систему XYZ" Значение по умолчанию: пустая строка. См. Также параметр инициализационного файла *MessagingAdditionalInfo*

По завершении настройки нажмите **Сохранить**.

28.2.2 Интеграция с сервером JMS

Страница настройки интеграции с сервером **JMS** выглядит следующим образом.

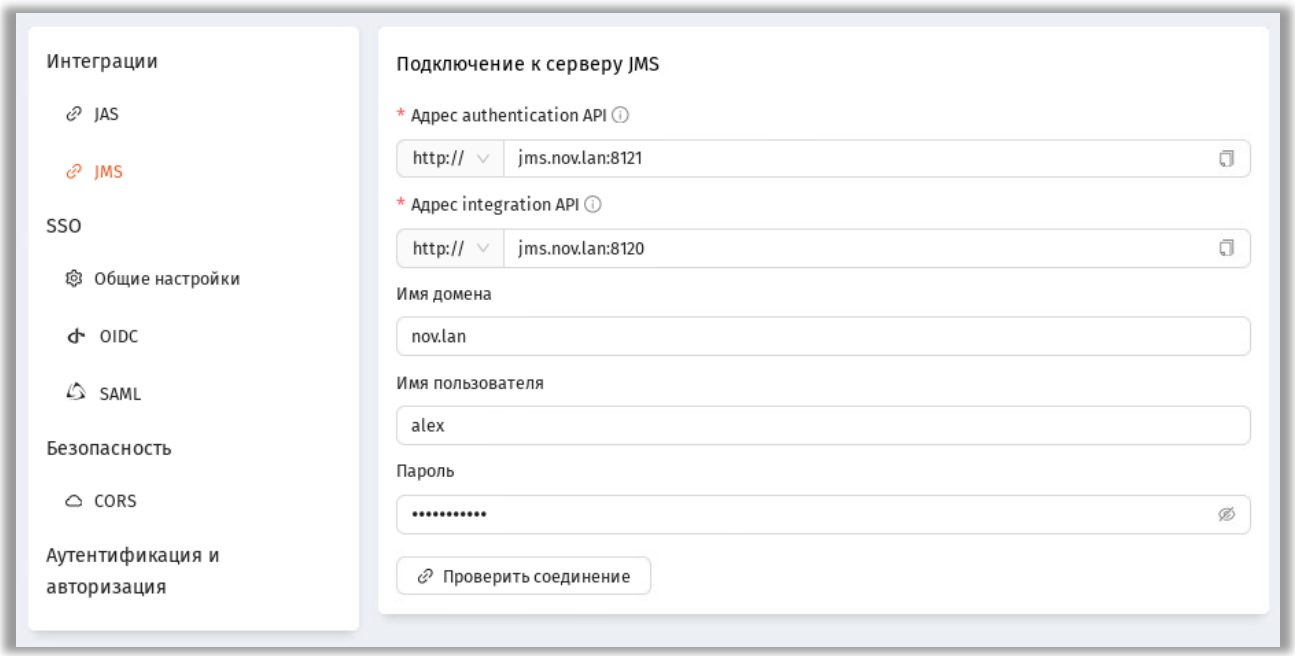

*Рис. 61 – Страница настройки интеграции с сервером **JMS***

Выполните настройку, руководствуясь Табл. 15.

Табл. 15 – Настройки интеграции с сервером JMS

Настройка	Описание
<Фрейм> **Подключение к серверу JMS**	

Адрес authentication API	Укажите адрес сервиса аутентификации сервера JMS в следующем формате
	`https://<FQDN-имя сервера>:<порт>`
	где <FQDN-имя сервера> – полное доменное имя (FQDN) сервера JMS, например, srv01.test.com
	Например:
	`"http://srv01.test.com:8121"`

Адрес integration API	Адрес интеграционного API сервера JMS (IntegrationManager API). Используется для получения данных пользователей, профилей SSO и клиентов SSO. Укажите адрес в следующем формате
	`https://<FQDN-имя сервера>:<порт>`
	где <FQDN-имя сервера> – полное доменное имя (FQDN) сервера JMS, например, srv01.test.com
	Например:
	`"http://srv01.test.com:8120"`

Имя домена	Укажите имя домена учётной записи, от имени которой будет осуществляться доступ к серверу
Имя пользователя	Имя доменного пользователя, от имени которого сервер JIP будет подключаться к серверу JMS
Пароль	Имя доменного пользователя в соответствующей ресурсной системе (FreeIPA, Active Directory и др.), от имени которого сервер JIP будет подключаться к серверу JMS
<кнопка> **Проверить соединение**	Для проверки корректности указанных параметров нажмите кнопку **Проверить соединение**. При успешном соединении отобразится уведомление «Соединение успешно установлено!»

По завершении настройки нажмите **Сохранить**.

28.2.3 Общие настройки SSO

Страница общих настроек механизма **SSO** (Single Sign-On) выглядит следующим образом.

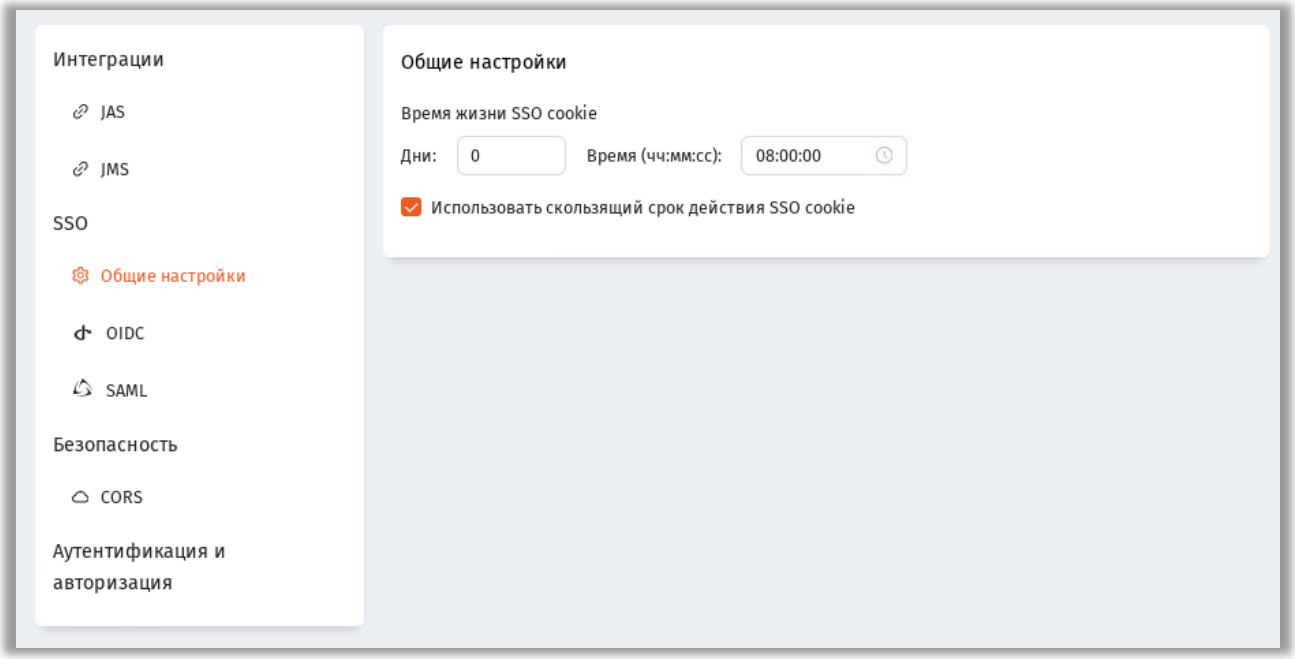

*Рис. 62 – Страница общих настроек **SSO***

Выполните настройку, руководствуясь Табл. 16.

Табл. 16 – Общие настройки SSO

Настройка	Описание
<Фрейм> **Общие настройки**	
Время жизни SSO cookie	Срок действия cookie для аутентификации. (Поля для задания параметра **Дни** и **Время (чч:мм:сс)**). Значение по умолчанию: **8** часов
Использовать скользящий срок действия SSO cookie	Флаг. Указывает на необходимость продления срока действия cookie при каждом использовании.

По завершении настройки нажмите **Сохранить**.

28.2.4 Настройки OIDC

Страница настроек протокола **OIDC** (OpenID Connect) выглядит следующим образом.

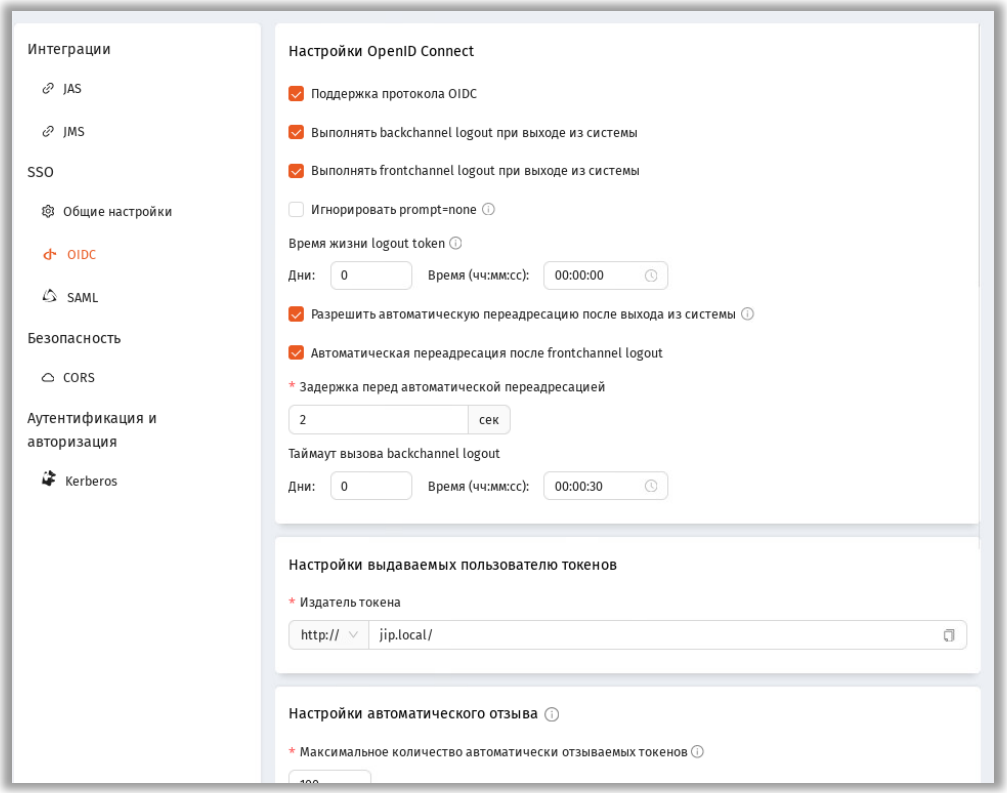

*Рис. 63 – Страница настроек **OIDC***

Выполните настройку, руководствуясь Табл. 17.

Табл. 17 – Настройки OIDC

Настройка	Описание
<Фрейм> **Настройки OpenID Connect**	
Поддержка протокола OIDC	Флаг. Отвечает за включение/выключение поддержки протокола OIDC
Выполнять backchannel logout при выходе из системы	Флаг. Разрешает выполнение backchannel logout при выходе из системы согласно настройкам клиента OIDC.
Выполнять frontchannel logout при выходе из системы	Флаг. Разрешает выполнение frontchannel logout при выходе из системы согласно настройкам клиента OIDC.
Игнорировать prompt=none	Флаг. Определяет, следует ли игнорировать требование prompt=none запроса авторизации и отображать интерфейс аутентификации при ее необходимости, то есть когда у пользователя нет активной сессии.
Время жизни logout token	Время жизни токена, который JIP отправляет клиенту при backchannel logout, чтобы уведомить его о выходе пользователя из системы (Поля для задания параметра **Дни** и **Время (чч:мм:сс)**). Значение по умолчанию: **0**
Разрешить автоматическую переадресацию после выхода из системы	Флаг. После выхода из системы пользователь будет автоматически переадресован на адрес, указанный в соответствующем клиенте OIDC.

Настройка	Описание
Автоматическая переадресация после frontchannel logout	Флаг. Определяет, выполнять ли переадресацию после frontchannel logout.
Задержка перед автоматической переадресацией	Время ожидания (в секундах) перед выполнением автоматической переадресации. Значение по умолчанию: **2**
Таймаут вызова backchannel logout	Максимальное время ожидания ответа при backchannel logout. (Поля для задания параметра **Дни** и **Время (чч:мм:сс)**). Значение по умолчанию: **30 сек**
<Фрейм> **Настройки выдаваемых пользователю токенов**	
Издатель токена	Определяет идентификатор издателя (Issuer) в рамках протокола OIDC. URL, по которому можно обратиться к web-приложению JIP извне. Может быть, например, `http://PC-DOMAIN-NAME/oidc` и будет использоваться в качестве Issuer JWT-токенов.
<Фрейм> **Настройки автоматического отзыва**	
Максимальное число автоматически отзываемых токенов	Максимальное количество токенов, которые будут отозваны у пользователя за один выход из системы, начиная с тех, что были выданы последними. Значение по умолчанию: **100**
Таймаут для процесса отзыва	Максимальная продолжительность процесса автоматического отзыва. (Поля для задания параметра **Дни** и **Время (чч:мм:сс)**). Значение по умолчанию: **15 сек**
<Фрейм> **Настройки безопасности**	
Шифрование access token	Флаг. Включает или отключает шифрование access token. (По умолчанию выключен)
<Фрейм> **Настройки сертификата**	
Отпечаток сертификата	Отпечаток сертификата, используемого в OIDC для подписи и проверки подлинности токенов. Для установки сертификата нажмите **Выбрать сертификат**

По завершении настройки нажмите **Сохранить**.

28.2.5 Настройки SAML

Страница настроек протокола **SAML** (Security Assertion Markup Language) выглядит следующим образом.

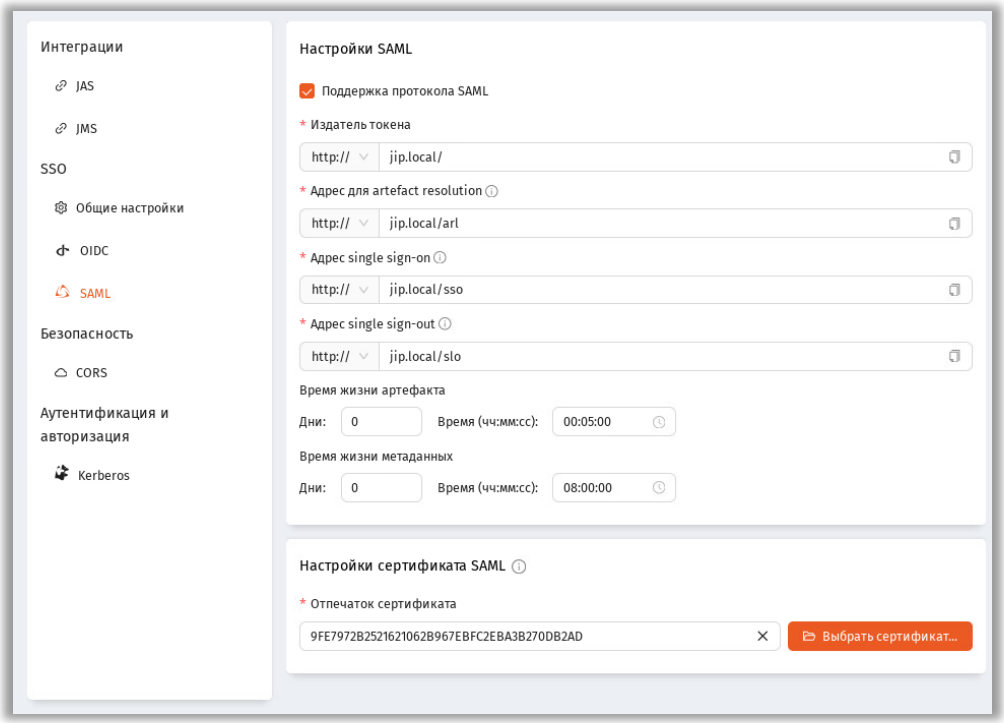

*Рис. 64 – Страница настроек **SAML***

Выполните настройку, руководствуясь Табл. 18.

Табл. 18 – Настройки SAML

Настройка	Описание
<Фрейм> **Настройки SAML**	
Поддержка протокола SAML	Флаг. Отвечает за включение/выключение поддержки протокола SAML
Издатель токена	Определяет идентификатор издателя (Issuer), который будет использоваться при выпуске утверждений SAML. URL, который будет указан в качестве идентификатора сервера и как Issuer в токене SAML. Попадает в метаданные SAML сервера, которые может читать любой клиент и каждый клиент в своей конфигурации должен будет указать этот Issuer в качестве идентификатора сервера. Главное требование к издателю – должен быть уникален в рамках федерации. По формату должен соответствовать строке URI (http//:...), но при этом не обязательно должен быть доступной ссылкой. Никак не связан с другими адресами, будь то адресом самого JIP или subject в SSL- или OIDC/SAML- сертификате.
Адрес для artefact resolution	URL, используемый для обработки запросов разрешения артефактов (Artifact Resolution Service). Указывается внешний адрес, передаваемый конечному пользователю при получении им метаданных в рамках протокола SAML. Может отличаться от адреса, по которому доступен соответствующий вызов, в случае использования обратного прокси-сервера между клиентом и JIP

Настройка	Описание
Адрес single sign-on	URL, по которому происходит вход пользователей (Single Sign-On Service). Указывается внешний адрес, передаваемый конечному пользователю при получении им метаданных в рамках протокола SAML. Может отличаться от адреса, по которому доступен соответствующий вызов, в случае использования обратного прокси-сервера между клиентом и JIP
Адрес single sign-out	URL, по которому происходит выход пользователей (Single Logout Service). Указывается внешний адрес, передаваемый конечному пользователю при получении им метаданных в рамках протокола SAML. Может отличаться от адреса, по которому доступен соответствующий вызов, в случае использования обратного прокси-сервера между клиентом и JIP
Время жизни артефакта	Указывает, как долго артефакт (SAML Artifact) будет считаться действительным. (Поля для задания параметра **Дни** и **Время (чч:мм:сс)**). Значение по умолчанию: **5 мин**
Время жизни метаданных	Определяет срок кэширования метаданных поставщика услуг (SP Metadata). (Поля для задания параметра **Дни** и **Время (чч:мм:сс)**). Значение по умолчанию: **8 ч**
<Фрейм> **Настройки сертификата SAML**	
Отпечаток сертификата	Отпечаток сертификата, используемого для подписи и проверки подлинности утверждений SAML. Для установки сертификата нажмите **Выбрать сертификат**

По завершении настройки нажмите **Сохранить**.

28.2.6 Настройки CORS

Страница настроек **CORS** (Cross-Origin Resource Sharing) выглядит следующим образом.

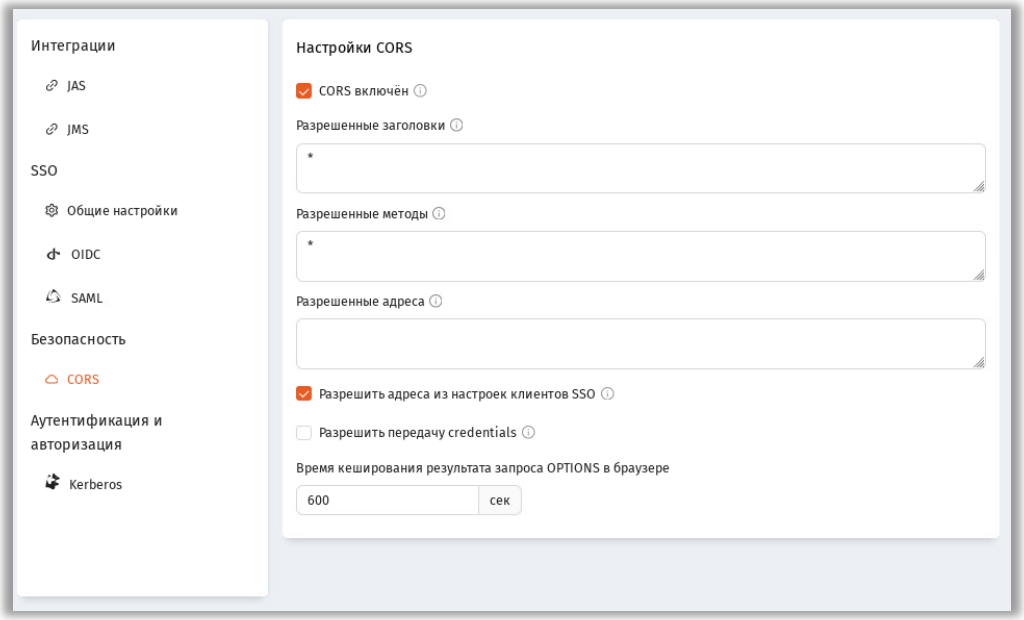

*Рис. 65 – Страница настроек **CORS***

Выполните настройку, руководствуясь Табл. 19.

Табл. 19 – Настройки CORS

Настройка	Описание
<Фрейм> **Настройки SAML**	
CORS включен	Флаг. Отвечает за включение/выключение CORS в web-приложении JIP для SAML и OIDC endpoint. Включён по умолчанию.
Разрешенные заголовки	Соответствует параметру Allowed Headers стандарта CORS. Укажите список значений через запятую. Для разрешения любых заголовков укажите звёздочку "*" (значение по умолчанию)
Разрешенные методы	Соответствует параметру Allowed Methods стандарта CORS. Укажите список значений через запятую. Для разрешения любых методов укажите звёздочку "*" (значение по умолчанию)
Разрешенные адреса	Соответствует параметру Allowed Origins стандарта CORS. Укажите список значений через запятую. Для разрешения любых методов укажите звёздочку "*"

Настройка	Описание
Разрешить адреса из настроек для клиентов SSO	Флаг. При установке автоматически добавляет адреса из настроек переадресации клиентов SSO (Login Redirect Urls, Logout Redirect Usls) в список разрешенных. Если не разрешено, то для корректной переадресации после входа или выхода из системы указанные для переадресации адреса нужно добавить вручную в список разрешенных. В противном случае переадресация будет запрещена браузером.
Разрешить передачу credentials	Флаг. Разрешает серверу включать учётные данные в кросс-доменные HTTP-запросы. К учетным данным относятся: файлы cookie, клиентские сертификаты TLS или заголовки аутентификации, содержащие имя пользователя и пароль. По умолчанию эти учётные данные не отправляются в кросс-доменных запросах, а включение этой настройки может сделать сайт уязвимым для атак с подделкой межсайтовых запросов (CSRF).
Время кеширования результата запроса OPTIONS в браузере	Определяет максимальный срок жизни (в секундах) результатов preflight (OPTIONS) запросов. Значение по умолчанию: **600**

По завершении настройки нажмите **Сохранить**.

28.2.7 Настройки Kerberos

Страница настроек **Kerberos** выглядит следующим образом.

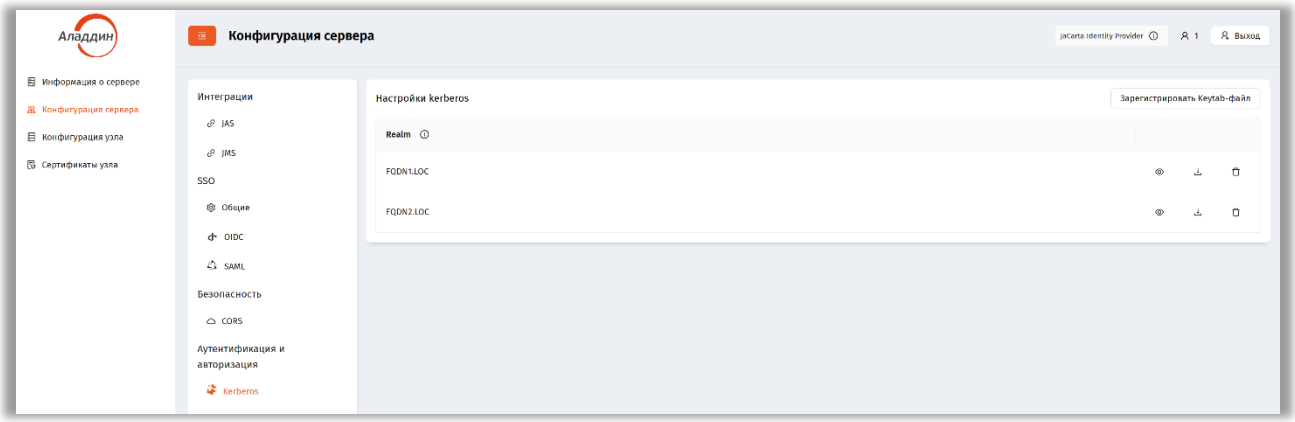

*Рис. 66 – Страница настроек **Kerberos***

Выполните настройку, руководствуясь Табл. 20.

Табл. 20 – Настройки Kerberos

Настройка	Описание
<Фрейм> **Настройки Kerberos** Фрейм содержит информацию о зарегистрированных keytab-файлах в виде их списков, сгруппированных по доменам, к которым данные keytab-файлы относятся	
<кнопка> **Зарегистрировать Keytab-файл**	Для регистрации нового keytab-файла нажмите **Зарегистрировать Keytab-файл**
<кнопки> 	Кнопки просмотра, выгрузки и удаления определены для уже загруженных Keytab-файлов

По завершении настройки нажмите **Сохранить**.

28.3 Конфигурация узла

Раздел **Конфигурация узла** выглядит следующим образом.

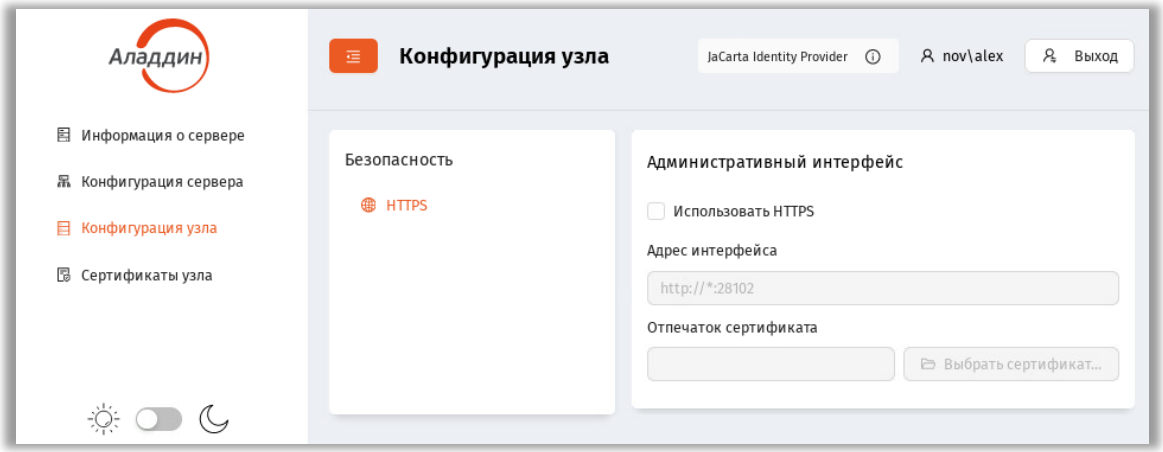

*Рис. 67 – Страница раздела **Конфигурация узла***

Для настройки узла в разделе web-консоли **Конфигурация узла** -> **Безопасность** -> **HTTPS** выполните следующие настройки руководствуясь Табл. 21.

Табл. 21 – Настройки защищённого подключения к узлу

Элемент интерфейса	Описание
<Фрейм> **Административный интерфейс** (API-интерфейс AdministrationService сервера JIP используемый для его администрирования через web-консоль или консольный агент)	
Использовать HTTPS	Установите флаг, если соединение должно осуществляться по протоколу SSL/TLS
Адрес интерфейса	Адрес административного интерфейса (считывается из файла конфигурации)
Отпечаток сертификата	Нажмите **Выбрать сертификат** и выберите сертификат, который должен использоваться в протоколах SSL/TLS для подключения к JIP (Поле доступно для редактирования только при установленном флаге **Использовать SSL-подключение**)

29.Приложения

Примеры конфигурационных файлов.

29.1 Приложение 1. Полный файл инициализации сервера JIP

Перечислены все доступные настройки.

```
[service]
; Путь до исполняемого файла (Обязателен)
execPath=\opt\aladdin\jip-engine\Aladdin.JIP.Engine
; Адреса API управления (можно несколько через ;) (Обязателен)
controlServiceUrls=http://localhost:28103
; Адреса API администрирования (можно несколько через ;) (Обязателен)
administrationServiceUrls=http://*:28102
; Язык интерфейса (en, ru) (Необязателен, по умолчанию: ru)
culture=ru

; Автоматический старт (Обязателен, по умолчанию: true)
autoStart=true

[controlPrimaryUser]
; Имя пользователя (Обязателен)
username=test
; Пароль (Обязателен)
password=test

[administrationPrimaryUser]
; Имя пользователя (Обязателен)
username=test
; Пароль (Обязателен)
password=test
```

```
[database]
; Тип СУБД (PostgreSQL, MSSQL, JatobaSQL) (Обязателен)
type=PostgreSQL
; Адрес сервера базы данных (Обязателен)
serverAddress=postgres
; Порт подключения к СУБД (Обязателен)
serverPort=5432
; Имя базы данных (Обязателен)
databaseName=JIPDB
; Режим аутентификации для мастера развертывания (password) (Обязателен)
serverLoginType=password
; Имя пользователя для создания БД (Обязателен)
serverLogin=postgres
; Пароль пользователя сервера (Обязателен)
serverPassword=password
; Режим аутентификации для подключения к БД(password) (Обязателен)
databaseLoginType=password
; Имя пользователя БД (Обязателен)
databaseLogin=postgres
; Пароль пользователя БД (Обязателен)
databasePassword=123456

[jas]
; URL сервера JAS (Обязателен)
url=http://jas.engine.local:8221/api/v4.1
; Тип аутентификации (None, Basic) (Необязателен, по умолчанию: Basic)
securityType=Basic
; Имя пользователя для доступа к API JAS (Обязателен для securityType=Basic)
username=admin
; Пароль пользователя для доступа к API JAS (Обязателен для securityType=Basic)
password=password
; Таймаут запроса (TimeSpan d.hh:mm:ss.fffffff) (Необязателен, по умолчанию: 30
секунд)
timeout=00:00:30
; Домен по-умолчанию для поиска пользователя в ресурсной системе, если пользова-тель
ввел логин без указания домена. (Необязателен)
defaultDomain=
; Поддерживаемые типы аутентификации в порядке приоритета (Otp, Push, Messaging)
(Обязателен)
authTypes=Otp,Push,Messaging
; Таймаут специально для Push-аутентификации (TimeSpan d.hh:mm:ss.fffffff). (Не-
обязателен, по умолчанию: 2 минуты).
pushTimeout=00:02:00
; Идентификатор системы для messaging (Необязателен)
messagingSystemId=
; Время жизни кода авторизации в секундах (число) (Необязателен)
messagingTTL=
; Таймаут между попытками аутентификации в миллисекундах (число) (Необязателен)
messagingRetryDelay=
; Текст сообщения для messaging (Необязателен)
messagingAdditionalInfo=

[jms]
; URL для обращения к API аутентификации JMS (Обязателен)
authenticationApiUrl=http://demo.jms.local:8121
; URL для обращения к интеграционному API JMS (Обязателен)
integrationApiUrl=http://demo.jms.local:8120
; Имя домена учетной записи (Обязателен)
authAccountSystemName=asn
; Имя пользователя для аутентификации (Обязателен)
authId=admin
; Пароль пользователя для подключения (Обязателен)
authPassword=password

[oidc]
; Включает поддержку OpenID Connect (Необязателен, по умолчанию: true)
```

```
enabled=true
; Выполнять backchannel logout (Необязателен, по умолчанию: true)
enabledBackchannelLogout=true
; Выполнять frontchannel logout (Необязателен, по умолчанию: true)
enabledFrontchannelLogout=true
; Игнорировать prompt=none (Необязателен, по умолчанию: false)
ignorePromptNone=false
; Разрешить авто-переадресацию после выхода (Необязателен, по умолчанию: true)
enabledAutoRedirectAfterLogout=true
; Авто-переадресация после frontchannel logout (Необязателен, по умолчанию: true)
autoRedirectFrontchannelLogout=true
; Задержка перед автоматической переадресацией в секундах (число) (Необязателен, по
умолчанию: 2)
autoRedirectDelay=2
; Таймаут вызова backchannel logout (TimeSpan d.hh:mm:ss.fffffff) (Необязателен, по
умолчанию: 30 секунд)
backchannelHttpClientTimeout=00:00:30

[oidcToken]
; Issuer токена (Обязателен)
tokenIssuer=http://jip.local:28100/oidc
; Максимальное число автоматически отзываемых токенов (Необязателен, по умолча-нию:
100)
revokeTokenLimit=100
; Максимальное число автоматически отзываемых авторизаций (Необязателен, по
умолчанию: 100)
authorizationsLimit=100
; Таймаут для процесса отзыва (TimeSpan d.hh:mm:ss.fffffff) (Необязателен, по
умолчанию: 15 секунд)
revokeProcessTimeout=00:00:15

[oidcSecurity]
; Включение/отключение шифрования access token (Необязателен, по умолчанию: true)
disableAccessTokenEncryption=true

[oidcCertificates]
; Отпечаток сертификата для OIDC (Обязателен)
certificateThumbprint=

[saml]
; Включение поддержки SAML (Необязателен, по умолчанию: true)
enabled=true
; Издатель токена (Issuer) (Обязателен)
endpointIssuer=http://jip.local:28100/saml
; URL для artefact resolution (Обязателен)
artifactResolutionLocation=http://jip.local:28100/saml/artifact
; URL single sign-on (Обязателен)
endpointSingleSignOnDestination=http://jip.local:28100/saml/login
; URL single sign-out (Необязателен, по умолчанию: "")
endpointSingleLogoutDestination=http://jip.local:28100/saml/logout
; Время жизни артефакта (TimeSpan d.hh:mm:ss.fffffff) (Необязателен, по умолча-нию:
5 минут)
artefactLifetime=00:05:00
; Время жизни метаданных (TimeSpan d.hh:mm:ss.fffffff) (Необязателен, по умолча-нию:
8 часов)
spMetadataCacheLifetime=08:00:00

[samlCertificates]
; Отпечаток сертификата для SAML (Обязателен)
certificateThumbprint=

[cors]
; Включение CORS (Необязателен, по умолчанию: true)
enabled=true
; Разрешенные заголовки (Необязателен, по умолчанию: *)
allowHeaders=*
```

```
; Разрешенные методы (Необязателен, по умолчанию: *)
allowMethods=*
; Разрешенные адреса (Необязателен, по умолчанию: "")
allowOrigins=
; Разрешить адреса из callback для клиентов SSO (Необязателен, по умолчанию: true)
useClientOrigins=true
; Разрешить передачу credentials (Необязателен, по умолчанию: false)
allowCredentials=false
; Время кеширования результата запроса OPTIONS в браузере (TimeSpan
d.hh:mm:ss.fffffff) (Необязателен, по умолчанию: 10 минут)
preflightMaxAge=00:10:00

[sso]
; Время жизни SSO cookie (TimeSpan d.hh:mm:ss.fffffff) (Необязателен, по умолча-нию:
8 часов)
cookieLifetime=08:00:00
; Использовать скользящий срок действия SSO cookie (Необязателен, по умолчанию:
true)
cookieSlidingExpiration=true
; Допустимое отклонение времени (TimeSpan d.hh:mm:ss.fffffff) (Необязателен, по
умолчанию: 5 минут)
clockTolerance=00:05:00
; Использовать SSO cookie постоянного хранения, сохраняется при перезапуске браузера
(Необязателен, по умолчанию: false)
isPersistent=false

[syslog]
; Адрес Syslog сервера (Необязателен, по умолчанию: 127.0.0.1)
server=127.0.0.1
; Порт Syslog сервера (Необязателен, по умолчанию: 514)
port=514
; Версия протокола (Необязателен, по умолчанию: RFC5424)
rfcVersion=RFC5424
; Использовать TLS (Необязателен, по умолчанию: false)
useEncryption=false
; Протокол (Необязателен, по умолчанию: UDP)
protocol=UDP
; Имя приложения (Необязателен, по умолчанию: JIP)
appName=JIP
; Метод разбивки на отдельные сообщения (Необязателен, по умолчанию: Non-
TrasparentFraming)
messageTransfer=NonTransparentFraming

[journaling]
; Режим логирования в Syslog (Необязателен, по умолчанию: None)
syslogAuthEventLogLevel=None
; Прерывать выполнение бизнес операции в случае возникновения ошибки при отправ-ки
сообщения в Syslog (Необязателен, по умолчанию: false)
syslogFailAuthOnError=false

[kerberos]
; Список Realms (Необязателен, по умолчанию: "")
realms=
; Список путей к keytab-файлам, которые соответствуют списку realms (Необязате-лен,
по умолчанию: "")
keyTabFilePaths=
; Включение автоматического определения наличия Kerberos тикета
shortcutEnabled=true
; Время жизни информации о наличии Kerberos тикета (TimeSpan d.hh:mm:ss.fffffff)
(Необязателен, по умолчанию: 5 минут)
shortcutLifetime=00:05:00
```

29.2 Приложение 2. Минимальный файл инициализации сервера JIP

Приведено минимальное обязательное содержимое файла инициализации. В нем требуется только указать certificateThumbprint в блоках oidcCertificates и samlCertificates, так как без сертификата эти компоненты не смогут работать.

```
[service]
; Путь до исполняемого файла (Обязателен)
execPath=\opt\aladdin\jip-engine\Aladdin.JIP.Engine
; Адреса API управления (можно несколько через ;) (Обязателен)
controlServiceUrls=http://localhost:28103
; Адреса API администрирования (можно несколько через ;) (Обязателен)
administrationServiceUrls=http://*:28102

[controlPrimaryUser]
; Имя пользователя (Обязателен)
username=test
; Пароль (Обязателен)
password=test

[administrationPrimaryUser]
; Имя пользователя (Обязателен)
username=test
; Пароль (Обязателен)
password=test

[database]
; Тип СУБД (PostgreSQL, MSSQL, JatobaSQL) (Обязателен)
type=PostgreSQL
; Адрес сервера базы данных (Обязателен)
serverAddress=postgres
; Порт подключения к СУБД (Обязателен)
serverPort=5432
; Имя базы данных (Обязателен)
databaseName=JIPDB
; Режим аутентификации для мастера развертывания (password) (Обязателен)
serverLoginType=password
; Имя пользователя для создания БД (Обязателен)
serverLogin=postgres
; Пароль пользователя сервера (Обязателен)
serverPassword=password
; Режим аутентификации для подключения к БД(password) (Обязателен)
databaseLoginType=password
; Имя пользователя БД (Обязателен)
databaseLogin=postgres
; Пароль пользователя БД (Обязателен)
databasePassword=123456

[jas]
; URL сервера JAS (Обязателен)
url=http://jas.engine.local:8221/api/v4.1
; Имя пользователя для доступа к API JAS (Обязателен)
username=admin
; Пароль пользователя для доступа к API JAS (Обязателен)
password=password
; Поддерживаемые типы аутентификации в порядке приоритета (Otp, Push, Messaging)
(Обязателен)
authTypes=Otp,Push,Messaging

[jms]
; URL для обращения к API аутентификации JMS (Обязателен)
authenticationApiUrl=http://demo.jms.local:8121
; URL для обращения к интеграционному API JMS (Обязателен)
integrationApiUrl=http://demo.jms.local:8120
```

```
; Имя домена учетной записи (Обязателен)
authAccountSystemName=asn
; Имя пользователя для аутентификации (Обязателен)
authId=admin
; Пароль пользователя для подключения (Обязателен)
authPassword=password

[oidcToken]
; Issuer токена (Обязателен)
tokenIssuer=http://jip.local:28100/oidc

[oidcCertificates]
; Отпечаток сертификата для OIDC (Обязателен)
certificateThumbprint=

[saml]
; Издатель токена (Issuer) (Обязателен)
endpointIssuer= http://jip.local:28100/saml
; URL для artefact resolution (Обязателен)
artifactResolutionLocation=http://jip.local:28100/saml/artifact
; URL single sign-on (Обязателен)
endpointSingleSignOnDestination=http://jip.local:28100/saml/login

[samlCertificates]
; Отпечаток сертификата для SAML (Обязателен)
certificateThumbprint=
```

Контакты, техническая поддержка

Офис (общие вопросы)

Адрес: 129226, Москва, ул. Докукина, д. 16, стр. 1, компания «Аладдин Р. Д.».

Телефоны: +7 (495) 223-00-01 (многоканальный), +7 (495) 988-46-40.

Факс: +7 (495) 646-08-82.

E-mail: aladdin@aladdin.ru (общий).

Web: www.aladdin.ru

Время работы: ежедневно с 10:00 до 19:00, кроме выходных и праздничных дней.

Техподдержка

Служба техподдержки принимает запросы только в письменном виде через web-сайт:

www.aladdin.ru/support/index.php

Список литературы

1 Программное обеспечение JaCarta Management System 4LX. Руководство пользователя [Текст]. — «Аладдин Р.Д.» — Файл «JMS 4LX РП.docx»

2 Программное обеспечение JaCarta Management System 4LX. Руководство администратора. Часть 1. Установка и настройка [Текст]. — «Аладдин Р.Д.». — Файл «JMS 4LX РА-1.docx»

3 Программное обеспечение JaCarta Management System 4LX. Руководство администратора. Часть 2. Функции управления [Текст]. — «Аладдин Р.Д.» — Файл «JMS 4LX РА-2.docx»

4 Программное обеспечение JaCarta Management System 4LX. Руководство администратора. Часть 3. Установка и настройка сервера аутентификации (JAS) [Текст]. — «Аладдин Р.Д.». — Файл «JMS 4LX РА-3.docx»

5 RU.АЛДЕ. 03.16.001-05 30 01-1. Формуляр [Текст]. — «Аладдин Р.Д.»

6 Единый Клиент JaCarta. Руководство администратора для операционных систем семейства Linux [Текст]. — «Аладдин Р.Д.»

7 JaCarta Management System. Подготовка и выпуск сертификатов MSCA для JMS [Текст]. — «Аладдин Р.Д.». — Файл JMS_x.x.x_Cert_Guide.docx

Полезные web-ресурсы

1 Справочный центр Astra Linux: https://wiki.astralinux.ru/

Регистрация изменений

Версия	Изменения
1.00	Исходная версия документа.

Коротко о компании

Компания «Аладдин Р. Д.» основана в апреле 1995 года и является российским разработчиком (вендором) средств защиты информации.

Компания является признанным экспертом и лидером российского рынка средств двухфакторной аутентификации пользователей, электронной подписи и защиты данных.

Основные направления

- Обеспечение безопасного доступа к информационным ресурсам предприятия, web-порталам и облачным сервисам (строгая двух- и трёхфакторная аутентификация).
- Электронная подпись (ЭП с неизвлекаемым закрытым ключом, формируемая в защищённом чипе), PKI.
- Защита персональных данных, данных на дисках компьютеров, серверов, баз данных.
- Все основные продукты имеют необходимые сертификаты ФСТЭК, ФСБ и Министерства обороны (включая работу с гостайной до уровня секретности СС).

Лицензии

- компания имеет все необходимые лицензии ФСТЭК России, ФСБ России и Министерства обороны России для проектирования, производства и поддержки СЗИ и СКЗИ, включая работу с гостайной и производство продукции в рамках гособоронзаказа.
- Система менеджмента качества продукции в компании с 2012 г. соответствует стандарту ГОСТ ISO 9001-2011 и имеет соответствующие сертификаты.
- Система проектирования, разработки, производства и поддержки продукции соответствует требованиям российского военного стандарта ГОСТ РВ 15.002-2012, необходимого для участия в реализации гособоронзаказа.

Аладдин

Лицензии ФСТЭК России № 0037 и № 0054 от 18.02.03, № 3442 от 10.11.2017
Лицензии ФСБ России № 12632 Н от 20.12.12, № 30419 от 16.08.17
Лицензия Министерства обороны РФ № 1384 от 22.08.16
Система менеджмента качества компании соответствует требованиям
ГОСТ Р ИСО 9001-2015 (ISO 9001:2015). Сертификат СМК № РОСС RU.ФК14.К00011 от 20.07.18

Тел. +7 (495) 223-00-01 Email: aladdin@aladdin.ru Web: www.aladdin.ru