РУКОВОДСТВО АДМИНИСТРАТОРА

Версия 1.0

Листов 125

2022

АННОТАЦИЯ

Настоящий документ является руководством администратора Прикладного программного обеспечения «Аврора Центр» релиз 3.1.1 (далее — ППО).

Настоящий документ содержит общую информацию о ППО, описание установки, обновления, удаления и резервного копирования ППО, а также описывает управление сервисами и их настройками, кроме того, настоящий документ содержит информацию о конфигурационных файлах ППО.

СОДЕРЖАНИЕ

1. ОБЩАЯ ИНФОРМАЦИЯ

1.1. Состав и назначение ППО

ППО предназначено для управления мобильными устройствами (МУ), функционирующими под управлением операционной системы (ОС) Аврора и ОС Android, а также для управления жизненным циклом мобильных приложений (МП) и обновлением ОС Аврора.

ПРИМЕЧАНИЕ. Под обновлением ОС Аврора понимается получение из доверенного хранилища пакетов с изменениями ОС (образа ОС) и их установка. При этом указанные процессы выполняются штатными средствами самой ОС, а ППО участвует лишь в их инициализации в ОС и не гарантирует их успешного завершения.

ППО является прикладным программным обеспечением со встроенными механизмами защиты информации от несанкционированного доступа.

ППО состоит из следующих подсистем:

- подсистема безопасности (ПБ);
- подсистема «Маркет» (ПМ);
- подсистема Платформа управления (ПУ);
- подсистема управления тенантами (ПУТ)[1];
- подсистема обновления ОС (ПООС).

Взаимодействие между подсистемами и компонентами подсистем осуществляется с использованием протокола HTTP стандарт RFC 2616, при этом обмен данными осуществляется в формате RFC 8259 (JSON).

[1] Наличие ПУТ в составе ППО зависит от варианта поставки (подраздел 1.5).

В качестве сервера базы данных (БД) используется сервер с установленной системой управления базами данных (СУБД) Postgres Pro или PostgreSQL, в которой хранятся данные ППО, для чего при развертывании создается специальная БД. Для хранения информации о сессиях используется СУБД Redis.

Подсистемы, входящие в состав ППО, позволяют выполнять логирование информационных сообщений, сообщений об ошибках, предупреждений и отладочной информации в системный журнал ОС (systemd-journald).

1.1.1. Подсистема безопасности

ПБ состоит из следующих компонентов:

- Консоль входа пользователей;
- Консоль администратора ПБ;
- Сервер приложений ПБ.

Консоль входа пользователей позволяет пользователям ППО осуществлять ввод идентификационной и аутентификационной информации.

Консоль администратора ПБ позволяет управлять учетными записями пользователей и работать с журналом регистрации событий.

Сервер приложений ПБ представляет собой совокупность веб-приложений, реализующих функции безопасности, а также позволяющих хранить в БД и предоставлять пользователям ППО доступ к данным об учетных записях и журналу регистрации событий.

ПБ предназначена для реализации следующих функций безопасности ППО:

- идентификации и аутентификации пользователей и МУ;
- управления идентификаторами пользователей и МУ;
- управления средствами аутентификации;
- управления учетными записями пользователей и МУ;
- ролевого управления доступом субъектов доступа к объектам доступа;
- регистрации событий безопасности;

– предоставления пользователям доступа к интерфейсу ПБ.

1.1.2. Подсистема «Маркет»

ПМ состоит из следующих компонентов:

– Консоль администратора ПМ;

– Консоль разработчика ПМ;

– МП «Аврора Маркет»;

– Сервер приложений ПМ.

Консоль администратора ПМ позволяет осуществлять взаимодействие Администратора Аврора Маркета с ПМ в части работы с МП.

Консоль разработчика ПМ позволяет добавлять новые и обновлять ранее загруженные МП, а также получать доступ к данным МП.

МП «Аврора Маркет» выполняется на МУ под управлением ОС и служит для отображения данных о МП, а также для их загрузки, установки, обновления и удаления.

Сервер приложений ПМ представляет собой совокупность веб-приложений, позволяющих хранить в БД и предоставлять пользователям ППО информацию о МП, при этом сами МП, их значки и скриншоты хранятся в БД.

ПМ предназначена для обеспечения:

– управления жизненным циклом МП (загрузка, согласование, удаление и публикация);

– управления дистрибуцией опубликованных МП (скачивание, установка, обновление и удаление);

– предоставления пользователям доступа к интерфейсу ПМ.

1.1.3. Подсистема Платформа управления

ПУ состоит из следующих компонентов:

– Консоль администратора ПУ;

– МП «Аврора Центр»;

– Сервер приложений ПУ.

Консоль администратора ПУ позволяет осуществлять взаимодействие Администратора Платформы управления с ПУ.

МП «Аврора Центр» выполняется на МУ под управлением ОС и позволяет осуществлять взаимодействие ПУ с МУ, а также в зависимости от управляющего сообщения или назначенного офлайн-сценария, полученного от Сервера приложений ПУ, имеет возможность управлять различными функциями МУ.

Сервер приложений ПУ представляет собой совокупность веб-приложений, позволяющих хранить в БД и предоставлять пользователям ППО данные о настройках и конфигурации ОС, а также формировать управляющие сообщения и офлайн-сценарии для МП «Аврора Центр».

ПУ предназначена для обеспечения:

– управления отдельными МУ (оперативное управление) и группами МУ;

– управления политиками, офлайн-сценариями;

– управления записями о МУ и пользователях МУ;

– управления МП на МУ;

– контроля состояния МУ;

– контроля применения политик на МУ;

– мониторинг событий и предоставление отчетности;

– предоставление пользователям доступа к интерфейсу ПУ.

1.1.4. Подсистема управления тенантами

ПУТ состоит из следующих компонентов:

– Консоль администратора ПУТ;

– Сервер приложений ПУТ.

Консоль администратора ПУТ позволяет осуществлять взаимодействие Администратора тенантов с ПУТ.

Сервер приложений ПУТ представляет собой совокупность веб-приложений, позволяющих хранить в БД и предоставлять пользователям ППО данные о тенантах, а также осуществлять управление тенантами.

ПУТ предназначена для обеспечения:

– управления жизненным циклом тенантов (создание, редактирование и удаление тенантов);

– управление организациями;

– управление контактными лицами организаций.

1.1.5. Подсистема обновления ОС

ПООС состоит из следующего компонента:

– Сервер приложений ПООС.

Сервер приложений ПООС представляет собой совокупность веб-приложений, позволяющих хранить в БД и предоставлять информацию и адреса хранения пакетов загрузочного модуля ОС.

Для хранения и дистрибуции пакетов ОС применяется файловый сервер, развернутый с использованием Nginx.

ПООС предназначена для обеспечения:

– предоставления информации о пакетах ОС;

– управления дистрибуцией пакетов ОС.

1.2. Субъекты доступа и права на доступ к интерфейсам ППО

1.2.1. Субъекты доступа (роли) ППО

Субъектами доступа являются пользователи ППО и МП «Аврора Центр» (процесс МП «Аврора Центр»), при этом субъектам доступа может быть назначена одна или несколько ролей, позволяющих выполнять следующие действия:

– Администратор учетных записей – управлять учетными записями пользователей;

– Оператор аудита – работать с журналом регистрации событий;

– Администратор Аврора Маркета – управлять ПМ с помощью интерфейса ППО;

– Разработчик – добавлять новые, обновлять ранее загруженные МП и получать информацию о них;

– Редактор приложений – обновлять и получать информацию о ранее загруженных МП;

– Пользователь Аврора Маркета – загружать МП и получать информацию о них;

– Администратор Платформы управления – управлять ПУ через интерфейс ППО;

– Администратор тенантов – управлять ПУТ через интерфейс ППО;

– МП «Аврора Центр» – назначается учетным записям МП «Аврора Центр» (процесс без участия пользователей, управляющих МУ).

ПРИМЕЧАНИЕ. В ППО обязательно наличие учетной записи пользователя с ролью Администратор учетных записей.

1.2.2. Права на доступ к интерфейсам ППО

Права на доступ к соответствующим разделам интерфейса ППО приведены в таблице (Таблица 1).

Таблица 1

Интерфейс ППО		Права на доступ	
Раздел	Подраздел	Подсистема	Консоль/Субъект доступа
Мультитенант	Тенанты	ПУТ	Консоль администратора ПУТ Администратор тенантов
	Организации	ПУТ	Консоль администратора ПУТ Администратор тенантов
Мониторинг	Индикаторы	ПУ	Консоль администратора ПУ Администратор Платформы управления

Интерфейс ППО		Права на доступ	
Раздел	Подраздел	Подсистема	Консоль/Субъект доступа
	Аудит	ПБ	Консоль администратора ПБ Оператор аудита
Управление	Устройства	ПУ	Консоль администратора ПУ Администратор Платформы управления
	Пользователи	ПУ	Консоль администратора ПУ Администратор Платформы управления
	Политики	ПУ	Консоль администратора ПУ Администратор Платформы управления
	Сценарии	ПУ	Консоль администратора ПУ Администратор Платформы управления
	Приложения	ПМ	Консоль администратора ПМ Администратор Аврора Маркета
	Витрины	ПМ	Консоль администратора ПМ Администратор Аврора Маркета
	Связки ключей	ПМ	Консоль администратора ПМ Администратор Аврора Маркета
Администрирование	Учетные записи	ПБ	Консоль администратора ПБ Администратор учетных записей
	Настройки	ПУ	Консоль администратора ПУ Администратор Платформы управления
	Орг. структура	ПУ	Консоль администратора ПУ Администратор Платформы управления
	Версии ОС	ПМ	Консоль администратора ПМ Администратор Аврора Маркета
Консоль разработчика ПМ		ПМ	Консоль разработчика ПМ Разработчик, Редактор приложений
МП «Аврора Маркет»		ПМ	Пользователь Аврора Маркета
МП «Аврора Центр»		ПУ	Процесс МП «Аврора Центр»

1.3. Описание принципов безопасной работы средства

1.3.1. Общая информация

ППО реализует следующие функции безопасности:

– идентификация и аутентификация субъектов доступа и объектов доступа;

– управление доступом субъектов доступа к объектам доступа;

– регистрация событий безопасности.

При использовании ППО необходимо выполнение следующих мер по защите информации от несанкционированного доступа:

– соблюдение парольной политики;

– соблюдение требования, согласно которому пароль не должен включать в себя легко вычисляемые сочетания символов;

– отсутствие у пользователя права передачи личного пароля третьим лицам;

– обязанность пользователя при вводе пароля исключить возможность его перехвата третьими лицами и техническими средствами.

При эксплуатации ППО запрещается:

– оставлять без контроля незаблокированные программные средства и/или ППО;

– разглашать пароли, выводить пароли на дисплей, принтер или иные средства отображения информации.

1.3.2. Компрометация паролей

Под компрометацией паролей необходимо понимать следующее:

– физическую утерю носителя с парольной информацией;

– передачу идентификационной информации по открытым каналам связи;

– перехват пароля при распределении идентификаторов;

– сознательную передачу информации третьему лицу.

ПРИМЕЧАНИЕ. При компрометации пароля пользователь обязан незамедлительно оповестить Администратора учетных записей.

1.3.3. Описание параметров (настроек) безопасности средства, доступных каждой роли пользователей, и их безопасные значения

Настройки параметров безопасности ППО доступны только пользователям с ролью Администратор учетных записей и заключаются в возможности управления ролями пользователей ППО.

Пользователям ППО должны назначаться минимальные права и привилегии, необходимые для выполнения ими своих должностных обязанностей (функций).

1.4. Условия выполнения

ПРИМЕЧАНИЕ. Для работы пользователей с интерфейсом ППО необходимо выполнение следующих условий:

– веб-браузер должен поддерживать технологии: TLS, CSS3, HTML5, ECMAScript 5 и Cookie. Рекомендуется использовать веб-браузер Chrome версии 90 или выше;

– разрешение экрана монитора должно быть не менее 1280x960 px.

Для функционирования ППО необходимы описанные в настоящем подразделе программно-технические средства.

В таблице (Таблица 2) приведены аппаратные характеристики серверов приложений ППО.

Таблица 2

Параметр	Количество МУ				
	10 000	50 000	100 000	275 000	550 000
Процессор	2 ядра	3 ядра	4 ядра	10 ядер	10 ядер
Объем оперативной памяти	4 ГБ	5 ГБ	6 ГБ	8 ГБ	8 ГБ
Объем жесткого диска	HDD 60 ГБ	HDD 80 ГБ	HDD 110 ГБ	HDD 130 ГБ	HDD 160 ГБ

Параметр	Количество МУ				
	10 000	50 000	100 000	275 000	550 000
Количество серверов	3	3	3	3	6

В таблице (Таблица 3) приведены аппаратные характеристики серверов БД[2].

Таблица 3

Параметр	Количество МУ						
	10 000 ПБ, ПМ, ПУ	50 000 ПБ, ПМ, ПУ	100 000 ПБ, ПМ, ПУ	275 000		550 000	
				ПБ	ПМ, ПУ	ПБ	ПМ, ПУ
Процессор	2 ядра	5 ядер	6 ядер	3 ядра	8 ядер	4 ядра	16 ядер
Объем оперативной памяти	4 ГБ	6 ГБ	8 ГБ	12 ГБ	12 ГБ	24 ГБ	24 ГБ
Объем жесткого диска	SSD 700 ГБ	SSD 3,5 ТБ	SSD 7 ТБ	SSD 11.6 ТБ	SSD 7.7 ТБ	SSD 23.2 ТБ	SSD 15.4 ТБ
Количество серверов	1	1	1	1	1	1	1

ПРИМЕЧАНИЕ. Для обеспечения отказоустойчивости необходимо добавить минимум один сервер с репликой БД (резервный сервер БД).

В таблице (Таблица 4) приведены программные характеристики серверов приложений ППО.

Таблица 4

Параметр	Значение
Операционная система	Одна из следующих ОС: – CentOS версии 7 или выше; – Альт 8 СП; – РЕД ОС 7.3; – РЕД ОС 7.3.1
Балансировщик микросервисов	Nginx Web Server версии 1.20.1 или выше
Система обнаружения сервисов	Consul версии 1.12 или выше
Средство управления конфигурациями микросервисов	Consul Template версии 0.25.1 или выше
Сервис гарантированной доставки сообщений	Nats Streaming Server версии 0.24.6 или выше

[2] В таблице не учитываются резервные серверы БД, необходимые для создания отказоустойчивой конфигурации.

Параметр	Значение
Прикладное программное обеспечение	ППО «Аврора Центр»

В таблице (Таблица 5) приведены программные характеристики серверов БД.

Таблица 5

Параметр	Значение
Операционная система	Одна из следующих ОС: – CentOS версии 7 или выше; – Альт 8 СП; – РЕД ОС 7.3; – РЕД ОС 7.3.1
СУБД	Одна из следующих СУБД: – Postgres Pro Certified 14.2.1; – Postgres Pro Enterprise Certified 13.6.1; – Postgres Pro Standard 12.11.2; – Postgres Pro Standard 13.7.2; – Postgres Pro Standard 14.4.1; – PostgreSQL 11.14 или выше; – PostgreSQL 12.9 или выше; – PostgreSQL 13.5 или выше; – PostgreSQL 14.1 или выше.
СУБД для хранения сессий	Redis 6.2.6 или выше
Расширение СУБД PostgreSQL для партиционирования таблиц БД	`pg_partman` 4 или выше
Расширение СУБД PostgreSQL поддерживающее быстрый поиск схожих строк	`pg_trgm`

В таблице (Таблица 6) приведены программные характеристики МУ.

Таблица 6

Параметр	Значение
Операционная система	ОС Аврора, ОС Android
Прикладное программное обеспечение	– МП «Аврора Центр»; – МП «Аврора Маркет»

Варианты конфигурации среды функционирования, в которых проводилось тестирование ППО, приведены в таблице (Таблица 7).

Таблица 7

ОС	СУБД	СЗИ НСД
CentOS-7.6	PostgreSQL 11.15	
CentOS-7.6	PostgreSQL 11.15	Специальное программное обеспечение (СПО) СЗИ НСД «Аккорд-X К»
CentOS-7.6	PostgreSQL 12.10	СПО СЗИ НСД «Аккорд-X К»
CentOS-7.6	PostgreSQL 13.5	СЗИ НСД «Dallas Lock Linux»
CentOS-7.7, kernel: 3.10.0-1062.9.1.el7.x86_64	PostgreSQL 14.1	СЗИ «Secret Net LSP» версия 1.10.1
CentOS-7.6	Postgres Pro Standard 12.11.2	
CentOS-7.6	Postgres Pro Standard 14.4.1	
Альт 8 СП (версии 8.0)	PostgreSQL 11.14	
Альт 8 СП (версии 8.4)	PostgreSQL 11 (из репозитория ОС)	
Альт 8 СП (версии 8.4)	Postgres Pro Standard 12.6	
Альт 8 СП (версии 8.4)	Postgres Pro Standard 13.7.2	
Альт 8 СП (версии 8.4)	Postgres Pro Certified (версия ядра postgres: 14.2.1)	
Альт 8 СП (версии 8.4)	Postgres Pro Enterprise Certified (версия ядра postgres: 13.6.1)	
РЕД ОС 7.3	Postgres Pro Standard 13.7.2	
РЕД ОС 7.3	Postgres Pro Standard 14.4.1	
РЕД ОС 7.3.1	PostgreSQL 12.9	
РЕД ОС 7.3.1	PostgreSQL 13.5	
РЕД ОС 7.3.1	PostgreSQL 14.1	

1.5. Варианты поставки

ППО может иметь несколько различных конфигураций, при этом вариант поставки определяется в соответствующем Лицензионном договоре:

– **Вариант № 1:** Прикладное программное обеспечение «Аврора Центр», состоящий из: ПБ, ПМ, ПУ, ПООС;

– **Вариант № 2:** Прикладное программное обеспечение «Аврора Центр» (Мультитенант), состоящий из: ПБ, ПМ, ПУ, ПУТ, ПООС.

Состав Изделия в зависимости от вариантов поставки приведен в таблице (Таблица 8).

Таблица 8

Состав и наименование	Прикладное программное обеспечение «Аврора Центр»	Прикладное программное обеспечение «Аврора Центр» (Мультитенант)
	Вариант № 1	Вариант № 2
«Руководство пользователя. Часть 1. Подсистема безопасности»	+	+
«Руководство пользователя. Часть 2. Подсистема «Маркет»»	+	+
«Руководство пользователя. Часть 3. Подсистема Платформа управления»	+	+
«Руководство пользователя. Часть 4. Подсистема управления тенантами»	Не входит в комплект поставки	+
«Руководство пользователя. Часть 5. Мобильное приложение «Аврора Маркет» для операционной системы Аврора»	+	+
«Руководство пользователя. Часть 6. Мобильное приложение «Аврора Центр» для операционной системы Аврора»	+	+
«Руководство пользователя. Часть 7. Мобильное приложение «Аврора Маркет» для операционной системы Android»	+	+
«Руководство пользователя. Часть 8. Мобильное приложение «Аврора Центр» для операционной системы Android»	+	+
«Руководство администратора»	+	+

1.6. Указания по эксплуатации

При эксплуатации ППО необходимо соблюдать следующие рекомендации:

– правом доступа к серверам с установленным ППО должны обладать лица, обеспечивающие функционирование ИС, прошедшие соответствующую подготовку, ознакомившиеся с эксплуатационной документацией (ЭД) на ППО и не рассматривающиеся в качестве нарушителей ИБ;

– должны быть предусмотрены меры, исключающие возможность несанкционированного изменения аппаратной части технических средств, на которых установлено ППО;

– должна обеспечиваться периодическая (не реже одного раза в месяц) проверка целостности программных компонентов ППО;

– на МУ и серверах с установленным ППО должны быть реализованы меры, исключающие возможность использования средств разработки и отладчиков для редактирования кода и оперативной памяти, используемой ППО;

– должны быть обеспечены меры, исключающие возможность несанкционированной модификации программных и информационных компонентов ППО;

– физический доступ к серверам с установленной СУБД должен предоставляться лицам, включенным эксплуатирующей организацией (оператором/потребителем) ИС в перечень лиц, которые не рассматриваются в качестве нарушителя ИБ;

– каналы связи, расположенные в пределах контролируемой зоны, должны быть защищены организационно-техническими мерами;

– каналы связи, расположенные за пределами контролируемой зоны, должны быть защищены с использованием средств криптографической защиты информации;

– должны быть предприняты меры по межсетевому экранированию.

При эксплуатации ППО рекомендуется применять следующие дополнительные организационные и технические меры:

– в процессе эксплуатации ППО должно быть обеспечено регулярное отслеживание наличия и осуществление установки обновлений безопасности ПО, входящего в среду функционирования ППО в соответствии с разделом 9 настоящего документа;

– ежемесячно должен производиться поиск актуальных уязвимостей и сведений об уязвимостях ППО и среды функционирования, анализ идентифицированных уязвимостей на предмет возможности их использования для нарушения безопасности;

– в среде функционирования ППО (общесистемное ПО, указанное в разделе 9 настоящего документа) должны быть установлены все имеющиеся обновления безопасности и «патчи» для ликвидации известных уязвимостей;

– в случае обнаружения уязвимостей в ПО ППО должно производиться их устранение в соответствии с методами и процедурами, установленными предприятием-разработчиком;

– оператор с периодичностью один раз в неделю должен получать информацию о выходе обновлений ППО через службу технической поддержки предприятия-изготовителя.

2. УСТАНОВКА ППО

ВНИМАНИЕ! Администратору/разработчику при копировании команд из настоящего документа в формате .pdf необходимо проявлять внимательность и дополнительно проверять результаты выполнения соответствующих команд на экране.

2.1. Общая информация

Установка ППО и компонентов среды функционирования ППО осуществляется с помощью сценариев установки ППО, выполняемых на управляющей ЭВМ и написанных с использованием декларативного языка разметки для описания конфигураций Ansible. Сценарии установки ППО позволяют выполнить установку как локально (все компоненты на одной ЭВМ), так и с удаленной ЭВМ (управляющей ЭВМ) (Рисунок 1).

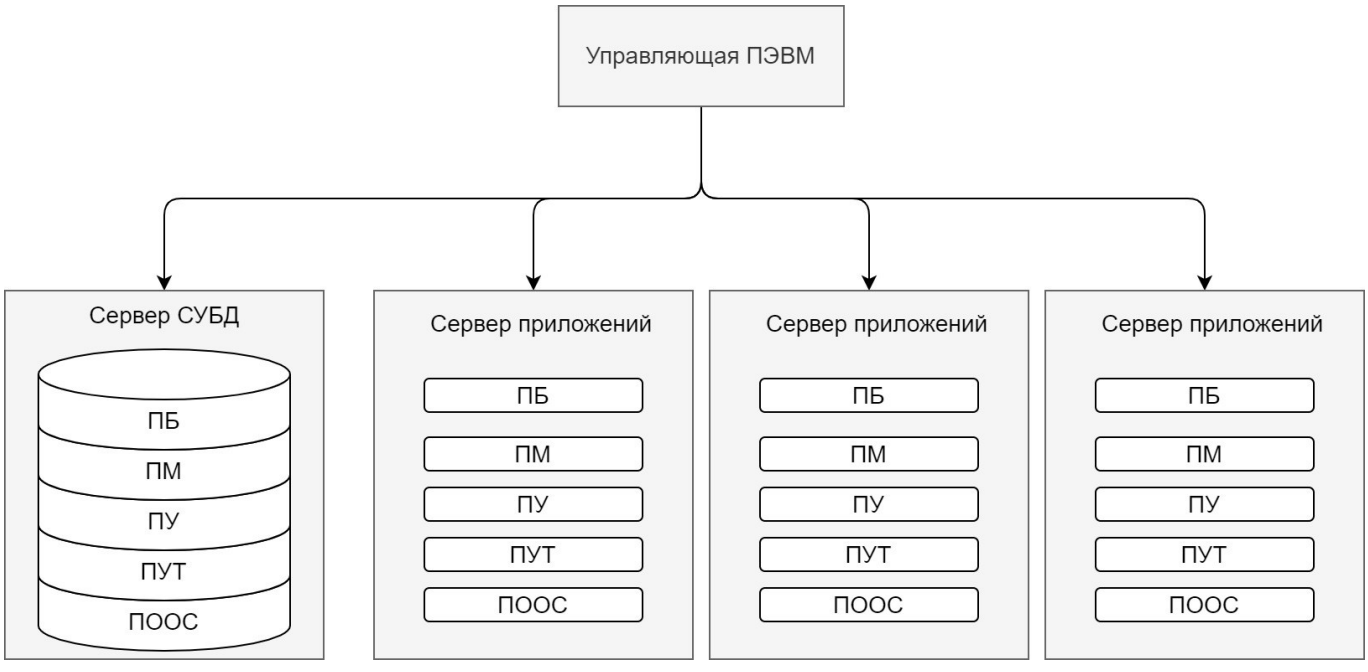


Рисунок 1

ПРИМЕЧАНИЕ. Управляющая ЭВМ необходима только для установки ППО.

Чтобы развернуть ППО необходимо выполнить следующие действия:

1) Убедиться, что все системные требования соблюдены;

2) Установить и настроить ОС на сервера приложений и сервера БД (подраздел 2.2);

3) Развернуть и настроить управляющую ЭВМ (подраздел 2.3);

4) Настроить компоненты среды функционирования ППО (п. 2.4.1);

5) Настроить ППО (п. 2.4.2);

6) Установить компоненты среды функционирования ППО (п. 2.5.1);

7) Установить ППО (п. 2.5.2);

8) Настроить подсистемы ППО (подраздел 2.8);

9) При необходимости выполнить дополнительные настройки ППО и его среды функционирования (подраздел 2.9);

10) Проверить корректность установки и функционирования ППО (подраздел 2.11).

2.2. Порядок установки и настройки ОС на серверах приложений и серверах БД

Для установки и настройки ОС на серверах приложений и серверах БД необходимо выполнить действия, описанные ниже.

2.2.1. Установить ОС CentOS версии 7, ОС РЕД ОС версии 7.3 или ОС Альт 8 СП на серверы приложений и серверы БД.

ВНИМАНИЕ! Перед установкой указанной выше ОС необходимо ознакомиться с требованиями, приведенными в документации на СЗИ НСД.

ОС должна быть установлена в минимальной конфигурации без графического интерфейса. Для установки ОС CentOS версии 7 необходимо использовать iso-образ, в названии которого содержится «Minimal». Например, CentOS-7-x86_64-Minimal-1810.iso.

Необходимо, чтобы настройки сети ОС соответствовали следующим требованиям:

1) Для основного сетевого интерфейса должен присутствовать конфигурационный `ifcfg` файл:

– ОС CentOS: `/etc/sysconfig/network-scripts/ifcfg-<имя интерфейса>`

2) Сетевой интерфейс должен автоматически запускаться при загрузке ОС.

Для этого параметр `ONBOOT` в конфигурационном файле `/etc/sysconfig/network-scripts/ifcfg-<имя интерфейса>` ОС CentOS должен иметь значение «`yes`»:

```
ONBOOT=yes
```

3) Приоритеты в конфигурационном файле `/etc/nsswitch.conf` файле должны выглядеть следующим образом (при использовании `dnsmasq`):

```
hosts: files dns ...
```

где `...` - остальные опции, если они используются.

4) На сетевых интерфейсах серверов приложений и БД должны быть настроены статические ip-адреса (использование динамических адресов, выдаваемых по DHCP, не допускается).

2.2.2. Перейти в учетную запись суперпользователя с помощью команды:

– ОС CentOS версии 7 и ОС РЕД ОС версии 7.3:

```
su -
```

– ОС Альт 8 СП:

```
su-
```

2.2.3. Установить пакет `sudo` (в случае использования ОС Альт 8 СП)

Для этого необходимо выполнить следующие действия:

2.2.3.1. Исключить CD-ROM из списка доступных репозиториев с помощью команды:

```
apt-repo rm all cdroms
```

2.2.3.2. Обновить ОС и ядро ОС с помощью команд:

```
apt-get update
apt-get dist-upgrade
apt-get update
apt-get dist-upgrade
update-kernel
integalert fix
```

2.2.3.3. Установить пакет с помощью `sudo` команды:

```
apt-get install sudo
```

2.2.4. Назначить пользователям ОС права на выполнение команд от имени суперпользователя.

Для этого необходимо добавить пользователя в группу `wheel`, выполнив команду:

```
usermod -aG wheel "имя_пользователя"
```

2.2.5. Разрешить пользователям, входящим в группу `wheel`, выполнять команды без ввода пароля. Для этого необходимо в файле `/etc/sudoers` раскомментировать строку:

```
# %wheel    ALL=(ALL)     NOPASSWD: ALL
```

Открыть файл `/etc/sudoers` необходимо от имени суперпользователя с помощью команды:

```
visudo
```

ВНИМАНИЕ! Права на выполнение команд от имени суперпользователя должны быть назначены всем пользователям (на управляющей ЭВМ, серверах приложений и серверах БД), которыми осуществляется установка компонентов среды функционирования, СУБД и ППО. В противном случае, в процессе установки возникнут ошибки.

2.2.6. Установить кодировку UTF-8 с помощью команды:

– ОС CentOS версии 7 и ОС РЕД ОС версии 7.3:

```
localectl set-locale LANG=en_US.UTF-8
```

– ОС Альт 8 СП:

В конфигурационном файле `/etc/sysconfig/i18n` задать следующее значение параметра `LANG`:

```
LANG=en_US.UTF-8
```

2.2.7. Задать имя хоста с помощью команды:

```
hostnamectl set-hostname "имя_хоста.имя_домена"
```

ВНИМАНИЕ! При задании имени хоста обязательно должно быть задано имя домена, которое отделяется точкой. Например:

```
hostnamectl set-hostname ocs-app.local
```

2.2.8. В настройках DNS-сервера или файлах `/etc/hosts` указать имена хостов (`hostname`) и полные имена доменов (`FQDN`) всех серверов кластера:

```
"ip-адрес" "имя_хоста.имя_домена"
```

Например (в файле `/etc/hosts`):

```
192.168.0.108 ocs-app.local
```

2.2.9. В файле `/etc/resolv.conf` указать адреса DNS-серверов:

```
nameserver "ip-адрес"
```

Например:

```
nameserver 192.168.0.1
```

2.2.10. Настроить маршрут по умолчанию (`default gateway`) через `lan` интерфейс в соответствии с документацией на ОС.

2.2.11. Задать текущие дату и время с помощью команды:

```
date -s 'YYYY-MM-DD HH:MI:SS'
```

Например:

```
date -s '2021-03-31 12:34:56'
```

2.2.12. Перезагрузить управляющую ЭВМ и серверы с помощью команды:

```
reboot
```

ВНИМАНИЕ! В процессе установки ППО службы SELinux и Firewalld будут отключены.

2.3. Порядок развертывания и настройки управляющей ЭВМ

ВНИМАНИЕ! Перед развертыванием и настройкой управляющей ЭВМ необходимо произвести установку и настройку ОС на серверах приложений и серверах БД в соответствии с подразделом 2.2 настоящего документа.

Для развертывания и настройки управляющей ЭВМ необходимо выполнить действия, описанные ниже.

2.3.1. Установить на управляющую ЭВМ одну из следующих ОС: ОС CentOS версии 7.9.2009, ОС Альт 8 СП или ОС Ubuntu 20.04.

Управляющая ЭВМ может быть развернута как на отдельной ЭВМ, так и на сервере приложений.

2.3.2. Настроить сетевое взаимодействие управляющей ЭВМ с серверами приложений и серверами БД.

Настройка сети на управляющей ЭВМ осуществляется в соответствии с ЭД на ОС.

2.3.3. С помощью последовательного выполнения команд установить на управляющей ЭВМ следующие пакеты:

– ОС CentOS версии 7.9.2009 (управляющая ЭВМ):

```
sudo yum -y install epel-release
sudo yum -y install jq
sudo yum -y install sed
sudo yum -y install coreutils
sudo yum -y install rsync
sudo yum -y install python2-pip
sudo python -m pip install --upgrade "pip < 21.0"
sudo python -m pip install wheel
sudo python -m pip install ansible==2.9.27
sudo python -m pip install yamllint
```

– ОС Альт 8 СП (управляющая ЭВМ):

```
sudo apt-get -y install jq
sudo apt-get -y install coreutils
sudo apt-get -y install rsync
sudo apt-get -y install python-module-pip
sudo python -m pip install --upgrade "pip < 21.0"
sudo python -m pip install --upgrade "setuptools < 45"
```

```
sudo python -m pip install --upgrade "wheel < 1.0"
sudo python -m pip install PyYAML==5.4.1
sudo python -m pip install pycparser==2.21
sudo python -m pip install MarkupSafe==1.1.1
sudo python -m pip install jinja2==2.11.3
sudo python -m pip install enum34==1.1.10
sudo python -m pip install six==1.16.0
sudo python -m pip install ipaddress~=1.0.0
sudo python -m pip install cffi==1.15.0
sudo python -m pip install cryptography==3.3.2
sudo python -m pip install ansible==2.9.27
sudo python -m pip install yamllint
```

– ОС РЕД ОС (управляющая ЭВМ):

```
sudo yum -y install expect
sudo yum -y install gcc
sudo yum -y install python3-devel
sudo yum -y install libffi-devel
sudo yum -y install rust
sudo yum -y install cargo
sudo yum -y install openssl-devel
sudo yum -y install python3-pip
sudo python3 -m pip install pycparser==2.21
sudo python3 -m pip install cffi==1.15.0
sudo python3 -m pip install cryptography==36.0.2
sudo python3 -m pip install MarkupSafe==2.1.1
sudo python3 -m pip install Jinja2==3.1.1
sudo python3 -m pip install PyYAML==6.0
sudo python3 -m pip install ansible==2.9.27
sudo python3 -m pip install yamllint
```

– ОС Ubuntu 20.04 (управляющая ЭВМ):

```
sudo apt update
sudo apt -y install jq
sudo apt -y install sed
sudo apt -y install coreutils
sudo apt -y install rsync
sudo apt -y install python3-pip
sudo apt -y install python3-yaml
sudo python3 -m pip install wheel
sudo python3 -m pip install MarkupSafe==2.1.1
sudo python3 -m pip install jinja2==3.1.1
sudo python3 -m pip install PyYAML==6.0
sudo python3 -m pip install cryptography==36.0.2
sudo python3 -m pip install cffi==1.15.0
sudo python3 -m pip install pycparser==2.21
sudo python3 -m pip install ansible==2.9.27
sudo python3 -m pip install yamllint
sudo apt install python-is-python3
```

2.3.4. Настроить SSH доступ управляющей ЭВМ к серверам приложений и серверам БД (даже в случае, когда управляющая ЭВМ и серверы установлены на одной ЭВМ):

– сформировать ключевую пару на управляющем сервере:

```
ssh-keygen -t rsa -b 4096
```

– скопировать открытый ключ на сервер приложений и БД:

```
ssh-copy-id <имя пользователя>@<сервер приложений>
ssh-copy-id <имя пользователя>@<сервер БД>
```

– проверить доступ с управляющей машины на серверы приложений и БД по SSH ключу (при выполнении команд ниже ввод пароля не должен требоваться):

```
ssh <имя пользователя>@<сервер приложения>
ssh <имя пользователя>@<сервер БД>
```

ПРИМЕЧАНИЕ. Управляющие команды, формируемые сценариями установки ППО, передаются с использованием протокола SSH.

2.3.5. Создать на управляющей ЭВМ отдельный каталог и скопировать в него каталог /server, находящийся на DVD с Изделием.

2.3.6. Перейти в каталог /server с помощью команды:

```
cd <путь к каталогу server>
```

2.3.7. Запустить installer-ac.sh (или installer-ac-mt.sh[3]) с помощью команды:

```
bash installer-ac.sh
или
bash installer-ac-mt.sh
```

2.3.8. Ознакомиться с «Лицензионным соглашением» и принять его.

Для того чтобы принять «Лицензионное соглашение» (Рисунок 2), необходимо после вопроса «Вы принимаете условия лицензии (y/n)?» ввести «y».

В результате в каталоге с файлом installer-ac.sh будет создан каталог install-<версия ППО>. Например, /install-release-v3.1.1.

[3] Название файла зависит от варианта поставки ППО.

```
[omp@ocs-app ~]$ ./installer_ac.sh

ЛИЦЕНЗИОННОЕ СОГЛАШЕНИЕ С КОНЕЧНЫМ ПОЛЬЗОВАТЕЛЕМ

ВАЖНО! ПЕРЕД ИСПОЛЬЗОВАНИЕМ ПРОГРАММНОГО ОБЕСПЕЧЕНИЯ, К КОТОРОМУ ПРИЛАГАЕТСЯ ДАННОЕ ЛИЦЕНЗИОННОЕ СОГЛАШЕ
НИЕ С КОНЕЧНЫМ ПОЛЬЗОВАТЕЛЕМ (ДАЛЕЕ ПО ТЕКСТУ – «ЛИЦЕНЗИОННОЕ СОГЛАШЕНИЕ»), ПОЖАЛУЙСТА, ВНИМАТЕЛЬНО ПРОЧ
ИТАЙТЕ НИЖЕСЛЕДУЮЩИЕ УСЛОВИЯ. ЕСЛИ ВЫ НЕ СОГЛАШАЕТЕСЬ С УСЛОВИЯМИ НАСТОЯЩЕГО ЛИЦЕНЗИОННОГО СОГЛАШЕНИЯ, Т
О ВЫ НЕ ИМЕЕТЕ ПРАВА ИСПОЛЬЗОВАТЬ ПРОГРАММНОЕ ОБЕСПЕЧЕНИЕ В КАКИХ-ЛИБО ЦЕЛЯХ.

1. ОПРЕДЕЛЕНИЯ
«Правообладатель» – общество с ограниченной ответственностью «Открытая мобильная платформа» (ООО «Открыт
ая мобильная платформа»), 420500, Республика Татарстан, Верхнеуслонский район, г. Иннополис, ул. Универс
итетская, д. 7, офис 59, ОГРН 1161690087020.
«ПО» – прикладное программное обеспечение «Аврора Центр» (ППО «Аврора Центр»), состоящее из следующих по
дсистем: прикладного программного обеспечения «Аврора Центр: Платформа управления» (ППО «Аврора Центр: П
латформа управления»), прикладного программного обеспечения «Аврора Центр: Маркет» (ППО «Аврора Центр: М
аркет») и Сервиса уведомлений Аврора, подробное описание функциональных возможностей которого содержатся
 в Документации. Данное Лицензионное соглашение применяется как к ППО «Аврора Центр», включающему в себя
 все перечисленные выше подсистемы, так и к каждой подсистеме в отдельности вне зависимости от комплектн
ости.
«Документация» – относящиеся к ПО сопроводительные материалы, в том числе Руководство по установке и нас
тройке, Руководство Пользователя, Руководство Администратора, которые принадлежат Правообладателю.
«Устройство» – это аппаратная система (физическая или виртуальная) со встроенным запоминающим устройство
м, на которых может быть запущено ПО.
«Права на интеллектуальную собственность» – все права на интеллектуальную и промышленную собственность,
включая права на изобретения, открытия и патенты на изобретения, включая заявки на выдачу патентов и пер
еизданные патенты, повторные заявки или заявки в продолжение и частичные продолжения; авторские права; о
бразцы и промышленные образцы; товарные знаки, знаки обслуживания, оформление товара и права на аналогич
ные объекты; секреты производства (ноу-хау), коммерческую тайну и конфиденциальную информацию; права на
топологии интегральных микросхем и права на фотошаблоны; и другие исключительные права.
«Лицензионное соглашение» – предоставляемое Вам Правообладателем ограниченное право на использование фун
кциональности ПО на условиях простой (неисключительной) лицензии в соответствии с условиями настоящего Л
ицензионного соглашения.
«Конечный пользователь» – любое юридическое лицо (организация), которое приобрело ПО для собственного ис
пользования и не для продажи.
«Пользователь» – физическое лицо, непосредственно осуществляющее эксплуатацию ПО в целях и порядке, опре
деляемом Конечным пользователем.
Настоящее Лицензионное соглашение является юридическим соглашением между Вами (далее по тексту – Конечны
й пользователь) и Правообладателем.
```

Рисунок 2

Также возможно автоматическое принятие лицензионного соглашения (без блокировки процесса установки). В данном случае необходимо ознакомиться с лицензионным соглашением (файл «Лицензионное соглашение.pdf» находится в архиве docs/docs-ac.zip или docs/docs-ac-mt.zip) и согласиться с ним. Для автоматического принятия лицензионного соглашения необходимо использовать флаг `--accept-license`:

```
bash installer-ac.sh --accept-license
или
bash installer-ac-mt.sh --accept-license
```

ВНИМАНИЕ! В случае несогласия с лицензионным соглашением использование ППО запрещается.

2.4. Порядок настройки компонентов среды функционирования ППО и ППО

ПРИМЕЧАНИЕ. Сценарии установки позволяют выполнить настройку и установку ППО, а также компонентов среды функционирования ППО для нескольких различных окружений. Порядок конфигурирования и установки ППО для нескольких окружений описан в п. 2.9.13.

2.4.1. Настройка компонентов среды функционирования

ПРИМЕЧАНИЕ. При задании паролей, секретов, токенов компонентов среды функционирования ППО допустимо использовать следующие специальные символы:

```
~$&*()-=_;.
```

Для настройки компонентов среды функционирования необходимо выполнить действия, описанные ниже.

2.4.1.1. Перейти в каталог со сценариями установки с помощью команды:

```
cd install-<версия ППО>/install-ac/
или
cd install-<версия ППО>/install-ac-mt/
```

Например:

```
cd install-release-v3.1.1/install-ac/
```

ПРИМЕЧАНИЕ. Дальнейшие действия по установке и настройке компонентов среды функционирования ППО, а также ППО, необходимо выполнять из данного каталога.

2.4.1.2. В конфигурационном файле `inventories/hosts.yml` задать адреса серверов (имена хостов), на которые будут установлены компоненты среды функционирования ППО.

Описание порядка задания адресов в конфигурационном файле `inventories/hosts.yml` приведено в п. 2.9.11 настоящего документа.

Для отображения адреса ЭВМ необходимо выполнить команду:

```
hostname
```

Примеры файлов `hosts.yml` для однонодовой и кластерной конфигурации приведены в каталоге `samples/ac/inventories/` (`samples/ac-mt/inventories/`).

Описание параметров конфигурационного файла `inventories/hosts.yml` приведено в п. 7.1.1 настоящего документа.

2.4.1.3. В конфигурационном файле `config/vars/_vars.yml` необходимо задать либо поменять предустановленные значения следующих параметров:

– параметры подключения подсистем ППО к БД:

```
postgresql:
  port: 5432
```

При использовании балансировщика БД необходимо задать адрес хоста балансировщика, например:

```
postgresql:
  host: "10.189.221.57"
```

– пароль суперпользователя `"postgres"` СУБД Postgresql, если установка СУБД осуществляется с помощью сценариев установки:

```
postgres_password: "postgres"
```

– версию СУБД:

```
pg_version: 12
```

Перечень допустимых значений параметра приведен в таблице (Таблица 9).

Таблица 9

Значение параметра	Версия СУБД
`11`	PostgreSQL 11
`12`	PostgreSQL 12
`13`	PostgreSQL 13
`14`	PostgreSQL 14
`11-pro`	Postgres Pro Standard 11
`12-pro`	Postgres Pro Standard 12
`13-pro`	Postgres Pro Standard 13
`14-pro`	Postgres Pro Standard 14
`13-entcert`	Postgres Pro Enterprise Certified 13
`14-stdcert`	Postgres Pro Certified 14

– имя и пароль пользователя СУБД Postgresql с ролью «replication»:

```
pg_replication_user:
```

```
  name: replication
  password: 123FD5648ert**h
```

– имя и пароль суперпользователя СУБД Postgresql от имени которого будет осуществляться установка ППО:

```
pg_custom_superuser:
  username: ocs_superuser
  password: ClacVob*Twes0Ls6
```

При самостоятельной установке СУБД, суперпользователя необходимо создать с помощью скрипта `samples/create_superuser.sql` выполнив команду:

```
psql -U <пользователь, от имени которого выполняется команда> -h
<адрес хоста СУБД> -f create_superuser.sql -v login='<имя
суперпользователя>' -v pass='<пароль суперпользователя>' -v
expr='<срок действия учетной записи суперпользователя>'
```

Например:

```
psql -U postgres -h 192.168.137.15 -f create_superuser.sql -v
login='ompdbuser' -v pass='Admin123!' -v expr='10 years'
```

ПРИМЕЧАНИЕ. Описание параметров конфигурационных файлов сценариев установки среды функционирования ППО, сценариев установки ППО и ППО приведены в самих конфигурационных файлах в виде комментариев.

2.4.1.4. В конфигурационном файле `config/secret.yml` задать пароли, секреты и токены:

– пароль доступа к БД:

```
  database:
    password: ocs
```

– пароль доступа к СУБД Redis в параметре `redis_password`:

```
  redis:
    password: "@rTT9089087fslk"
```

– секретный ключ для аутентификации запросов к сервисам ППО:

```
  hmac:
    key: "DEFAULT-F1IWp0t5dY5lYJrm7H-DEFAULT"
```

– токен доступа к сервису гарантированной доставки сообщений `Nats Streaming Server`:

```
  transport:
    nats:
```

```
    authToken: "FF12fddgdhFLL"
```

– токен доступа к системе обнаружения сервисов `Consul`:

```
consul:
  token: "ae9f5abb-6b8f-9252-59c5-53bcb651f182"
```

– секретный ключ клиентов (сервисов):

```
defaultOidcClientSecret: "HWfwehfoIOHwfe233WEfvwewe"
```

– ключ шифрования секретов, хранящихся в БД:

```
encrypt:
  keys:
    - "master-key-example"
```

– пароли, используемые для защиты критичной информации (например, cookie сессии):

```
oidcpSecrets:
  system:
    - kdj%93cxk+57nMa4
  cookie:
    - 9v_wer8*&r=_hY8u
```

ВНИМАНИЕ! Длина пароля должна быть не менее 16 символов. При обновлении ППО удалять старые пароли запрещается. Новые пароли необходимо добавлять в начало списка.

2.4.1.5. Настроить параметры взаимодействия клиентов СУБД (при необходимости).

ПРИМЕЧАНИЕ. По умолчанию сценарии установки автоматически задают параметры взаимодействия клиентов (например, серверов приложений ППО) СУБД.

Настройки взаимодействия клиентов с СУБД задаются в секции `pg_hba_settings` конфигурационного файла `config/vars/_vars.yml`.

Данная секция содержит следующие параметры:

– `type` - тип подключения к СУБД;

– `name` - имена пользователей СУБД, правила доступа для которых определяет данная запись;

– `database` - имена баз данных, доступ к которым описывает данная запись;

– address - IP-адрес подключения или IP-адрес подсети;

– method - метод аутентификации.

2.4.2. Настройка ППО (подсистем ППО)

ВНИМАНИЕ! Перед выполнением настроек необходимо изучить порядок работы с конфигурационными файлами, приведенный в п. 8.2.9 настоящего документа.

ПРИМЕЧАНИЕ. При задании паролей, секретов, токенов компонентов среды функционирования ППО допустимо использовать следующие специальные символы:

```
~$&*()-=_;.
```

Для настройки ППО необходимо выполнить следующие действия:

2.4.2.1. Перейти в созданный каталог (каталог: /install-<версия ППО>/install-ac/ или /install-<версия ППО>/install-ac-mt/).

2.4.2.2. Выполнить настройку url-адресов ППО

Url-адреса ППО задаются в секции publicUris в конфигурационном файле config/config.yml.j2.

Url-адрес ППО можно не задавать, если ППО устанавливается в однонодовой конфигурации и используется незащищенное соединение (протокол HTTP) с сервером приложений. В данном случае будет использоваться адрес, заданный в группе app конфигурационного файла inventories/hosts.yml.

Возможны следующие варианты настройки:

2.4.2.2.1 Url-адреса ППО используют один домен

В данном случае в параметре commonAddress необходимо задать имя домена и порт, например:

```
    ac:
      commonAddress: "http://acenter.example.ru:8009"
```

2.4.2.2.2 Url-адреса ППО разделены на внешний и внутренний домены

Подобное разделение, например, требуется, когда необходимо ограничить доступ к Консолям администраторов из внешней сети.

В данном случае в параметре `commonAddress` необходимо задать имя домена и порт для внешних адресов, а в параметре `adminAddress` задать имя домена и порт для внутренних адресов, например:

```
    ac:
      commonAddress: "http://acenter.example.ru:8009"
      adminAddress: "http://admin.example:8009
```

2.4.2.2.3 Url-адреса ППО используют разные домены

В данном случае для каждой подсистемы ППО задается свой собственный домен, например:

```
    auth:
      adminCrossTenantAddress: "http://authadmin.example:8009"
      publicAddress: "http://authpublic.acenter.example.ru:8009"
    aps:
      adminAddress: "http://apsadmin.acenter.example.ru:8009"
      devAddress: "http://apsdev.acenter.example.ru:8009"
      marketAddress:
"http://apsmarket.acenter.example.ru:8009"
    push:
      adminAddress: "http://pushadmin.example:8009"
      publicAddress: "http://pushpublic.acenter.example.ru:8009"
    mt:
      adminAddress: "http://mt.example:8009"
    pkgrepo:
      adminAddress: "http://pkgrepoadmin.example:8009"
      mobileAddress: "http://pkgrepomobile.acenter.example.ru:8009"
      repoAddress: "http://pkgrepo.acenter.example.ru:8009"
```

2.4.2.2.4 Настройка url-адресов ППО при использовании защищенного соединения

Незащищенное (протокол HTTP) соединение с сервером приложений допустимо использовать в пилотных проектах, где отсутствует обработка конфиденциальной информации. В остальных случаях должна обеспечиваться защита каналов связи. При использования защищенного соединения (протокол HTTPS) значения параметров будут иметь следующий вид:

```
    ac:
      commonAddress: "https://acenter.example.ru"
```

2.4.2.3. Отредактировать конфигурационный файл `config/config.yml.j2`

В данном конфигурационном файле необходимо задать либо поменять предустановленные значения:

– домен учетных записей пользователей для тенанта "default":

```
defaultIdentityDomain: "omprussia.ru"
```

– уровень детализации сообщений логирования (рекомендуется задать "info" при тестовой эксплуатации и "warn" при промышленной эксплуатации):

```
logger:
  level: "info"
```

2.4.2.4. Выполнить настройки безопасности ППО, другие дополнительные настройки ППО и настройки подсистем ППО (при необходимости).

ВНИМАНИЕ! Перед установкой ППО необходимо выполнить настройки безопасности ППО, дополнительные настройки ППО и настройки подсистем ППО (при необходимости).

Перечень и описание дополнительных настроек ППО приведен в подразделе 2.8 настоящего документа.

2.5. Порядок установки компонентов среды функционирования ППО и ППО

2.5.1. Установка компонентов среды функционирования ППО

Для установки компонентов среды функционирования ППО необходимо на управляющей ЭВМ выполнить действия, описанные ниже.

2.5.1.1. Обеспечить синхронизацию времени между нодами кластера

При эксплуатации ППО в кластерной конфигурации необходимо обеспечить синхронизацию времени между нодами кластера (например, с помощью утилиты chrony).

Для проверки синхронизации времени необходимо запустить bash-скрипт, предварительно задав адреса хостов в переменной HOSTS:

```
#!/bin/bash
HOSTS="
acenterdb01
```

```
acenterdb02
acenterapp01
acenterapp02
acenterapp03
"
# Prepare ssh sessions to increase speed time checking.
for i in $HOSTS
do
  ssh $i "echo $i session created" > /dev/null
done
for i in $HOSTS
do
  ssh $i "echo $i `date`" &
done
sleep 10
```

2.5.1.2. Установить на серверы приложений и серверы БД необходимые пакеты с помощью команд:

– серверы приложений:

```
ansible-playbook -i inventories/hosts.yml play-managed-node-
prerequisites.yml -vv -u <имя пользователя> --extra-vars
"node_type=app" --limit app
```

– серверы БД:

```
ansible-playbook -i inventories/hosts.yml play-managed-node-
prerequisites.yml -vv -u <имя пользователя> --extra-vars
"node_type=db" --limit postgresql
```

Например:

– серверы приложений:

```
ansible-playbook -i inventories/hosts.yml play-managed-node-
prerequisites.yml -vv -u omp --extra-vars "node_type=app" --limit app
```

– серверы БД:

```
ansible-playbook -i inventories/hosts.yml play-managed-node-
prerequisites.yml -vv -u omp --extra-vars "node_type=db" --limit
postgresql
```

Для установки всех пакетов на все серверы (на все серверы приложений и серверы БД независимо от их типа) необходимо выполнить команду:

```
ansible-playbook -i inventories/hosts.yml play-managed-node-
prerequisites.yml -vv -u <имя пользователя>
```

Порядок действий для самостоятельной установки пакетов приведен в пп. 2.9.6.3.2 настоящего документа.

2.5.1.3. Установить компоненты среды функционирования ППО с помощью команды:

```
ANSIBLE_USER="<имя пользователя>" ./deploy-infra.sh
```

Например:

```
ANSIBLE_USER="omp" ./deploy-infra.sh
```

В случае, если требуется репликация БД, команду установки компонентов среды функционирования необходимо запустить с параметром `--extra-vars "pg_slave_recreate=true"`:

```
ANSIBLE_USER="<имя пользователя>" ./deploy-infra.sh --extra-vars
"pg_slave_recreate=true"
```

Описание параметров запуска скрипта `deploy-infra.sh` и их возможные значения приведены в подразделе 3.1 настоящего документа.

ВНИМАНИЕ! Скрипт `deploy-infra.sh` позволяет устанавливать только СУБД PostgreSQL 11/12/13/14. СУБД Postgres Pro необходимо устанавливать самостоятельно.

При использовании СУБД Postgres Pro либо, если установку СУБД PostgreSQL 11/12/13/14 необходимо выполнить самостоятельно (без использования сценариев установки компонентов среды функционирования ППО), то команда установки компонентов среды функционирования имеет следующий вид:

```
ANSIBLE_USER="<имя пользователя>" ./deploy-infra.sh --skip-database
```

Описание установки и настройки СУБД Postgres Pro, а также требования к самостоятельной установке СУБД приведены в подразделе 2.7 настоящего документа.

Также предусмотрена возможность установки компонентов среды функционирования по отдельности с помощью следующих команд:

```
ANSIBLE_USER="<имя пользователя>" ./deploy-infra.sh -c dnsmasq
ANSIBLE_USER="<имя пользователя>" ./deploy-infra.sh -c nginx
ANSIBLE_USER="<имя пользователя>" ./deploy-infra.sh -c consul
ANSIBLE_USER="<имя пользователя>" ./deploy-infra.sh -c consul-template
ANSIBLE_USER="<имя пользователя>" ./deploy-infra.sh -c nats-streaming-
server
ANSIBLE_USER="<имя пользователя>" ./deploy-infra.sh -c redis
```

```
ANSIBLE_USER="<имя пользователя>" ./deploy-infra.sh -c ocs-user
ANSIBLE_USER="<имя пользователя>" ./deploy-infra.sh -c db
```

2.5.2. Установка ППО

Для установки ППО необходимо выполнить команду:

```
ANSIBLE_USER=<имя пользователя> ./deploy-ac.sh
```

Описание параметров запуска скрипта `deploy-ac.sh` и их возможные значения приведены в подразделе 3.2 настоящего документа.

Например:

```
ANSIBLE_USER=omp ./deploy-ac.sh
```

Для установки подсистем по отдельности необходимо в параметре --subsystems задать имя подсистемы. Установка подсистем ППО должна осуществляться строго в следующей последовательности: ПБ, ПМ, ПООС, ПУ, ПУТ. Пример установки подсистем по отдельности:

```
ANSIBLE_USER=omp ./deploy-ac.sh --subsystems auth
ANSIBLE_USER=omp ./deploy-ac.sh --subsystems appstore
ANSIBLE_USER=omp ./deploy-ac.sh --subsystems pkgrepo
ANSIBLE_USER=omp ./deploy-ac.sh --subsystems emm
ANSIBLE_USER=omp ./deploy-ac.sh --subsystems mt
```

2.5.3. Выполнить настройки подсистем ППО

Настройка подсистем ППО осуществляется в соответствии с подразделом 2.8 настоящего документа.

2.5.4. Выполнение ограничения по применению

ВНИМАНИЕ! После установки и настройки ППО необходимо выполнить ограничения по применению, произвести настройки безопасности компонентов среды функционирования и настроить СЗИ. Необходимая информация приведена в п. 2.9.6 настоящего документа.

2.5.5. Проверка корректности установки и функционирования ППО

Проверка осуществляется в соответствии с подразделом 2.11 настоящего документа.

2.6. Адреса веб-консолей

Первоначальный вход в ППО осуществляется с помощью Консоли администратора ПБ и предустановленной учетной записи с ролью Администратор учетных записей:

- логин: admin@omprussia.ru;
- пароль: Admin123!

ПРИМЕЧАНИЕ. При первом входе в ППО необходимо сменить пароль.

В таблице (Таблица 10) приведены адреса веб-консолей.

Таблица 10

Веб-консоль	URL-адрес веб-консоли
Консоль администратора ПБ	http://<сервер приложения>:8009/auth/admin/
Консоль администратора ПМ	http://<сервер приложения>:8009/appstore/admin/
Консоль разработчика ПМ	http://<сервер приложения>:8009/appstore/dev/
Консоль администратора ПУ	http://<сервер приложения>:8009/emm/admin/
Консоль администратора ПУТ	http://<сервер приложения>:8009/mt/admin/

2.7. Самостоятельная установка и настройка СУБД

2.7.1. Порядок установки и настройки СУБД Postgres Pro

Для установки и настройки СУБД Postgres Pro необходимо выполнить действия, описанные ниже.

2.7.1.1. Установить на серверы БД необходимые пакеты согласно п. 2.9.7.

2.7.1.2. Установить и инициализировать СУБД Postgres Pro

При инициализации СУБД необходимо установить следующие значения параметров:

```
LC_COLLATE 'en_US.UTF-8'
LC_CTYPE 'en_US.UTF-8'
ENCODING UTF8
```

Установка и инициализация СУБД Postgres Pro осуществляется в соответствии с ЭД на СУБД. После установки СУБД необходимо назначить пароль для пользователя `postgres` и создать на время установки символьную ссылку (`symlink`) для сокета PostgreSQL PRO с помощью следующих команд:

```
psql -U postgres
ALTER USER postgres with PASSWORD 'пароль';
exit
```

2.7.1.3. В конфигурационных файлах СУБД `pg_hba.conf и postgresql.conf` задать следующие параметры:

– тип соединения, диапазон IP-адресов клиентов БД;

– имя БД, имя пользователя;

– способ аутентификации клиентов;

– пароль пользователя СУБД в параметре `pg_superuser_password`.

2.7.1.4. Установить расширения `pg_partman` и `pg_cron` с помощью команд:

– ОС CentOS версии 7 и СУБД Postgres Pro 12:

```
sudo rpm -ivh pg_partman_12pro-std-4.6.0-1.el7.x86_64.rpm
sudo rpm -ivh pg_cron_12pro-std-1.4.1-1.el7.x86_64.rpm
```

– ОС CentOS версии 7 и СУБД Postgres Pro 14:

```
sudo rpm -ivh pg_partman_14pro-std-4.6.0-1.el7.x86_64.rpm
sudo rpm -ivh pg_cron_14pro-std-1.4.1-1.el7.x86_64.rpm
```

– ОС Альт 8 СП и СУБД Postgres Pro 12:

```
sudo rpm -ivh pg_partman_12pro-std-4.6.0-1.alt.x86_64.rpm
sudo rpm -ivh pg_cron_12pro-std-1.4.1-1.alt.x86_64.rpm
```

– ОС Альт 8 СП и СУБД Postgres Pro 13:

```
sudo rpm -ivh pg_partman_13pro-std-4.6.0-alt1.x86_64.rpm
sudo rpm -ivh pg_cron_13pro-std-1.4.1-alt1.x86_64.rpm
```

– ОС Альт 8 СП и СУБД Postgres Pro 14:

```
sudo rpm -ivh pg_partman_14pro-std-4.6.0-alt1.x86_64.rpm
sudo rpm -ivh pg_cron_14pro-std-1.4.1-alt1.x86_64.rpm
```

– ОС Альт 8 СП и СУБД Postgres Pro Enterprise 13:

```
sudo rpm -ivh pg_partman_13pro-ent-4.6.0-alt1.x86_64.rpm
sudo rpm -ivh pg_cron_13pro-ent-1.4.1-alt1.x86_64.rpm
```

– ОС РЕД ОС версии 7.3 и СУБД Postgres Pro 13:

```
sudo rpm -ivh pg_partman_13pro-std-4.6.0-1.redos7.x86_64.rpm
sudo rpm -ivh pg_cron_13pro-std-1.4.1-1.redos7.x86_64.rpm
```

– ОС РЕД ОС версии 7.3 и СУБД Postgres Pro 14:

```
sudo rpm -ivh pg_partman_14pro-std-4.6.0-1.redos7.x86_64.rpm
sudo rpm -ivh pg_cron_14pro-std-1.4.1-1.redos7.x86_64.rpm
```

Указанные rpm-пакеты находятся на DVD с загрузочными модулями ППО в архиве `/server/install-infra.tar.gz/install-infra/binary/postgresql/`.

2.7.1.5. Перезапустить сервис СУБД Postgres Pro с помощью команды:

```
sudo systemctl restart <имя сервиса СУБД>
```

Например:

```
sudo systemctl restart postgrespro-std-11
```

2.7.2. Порядок установки и настройки СУБД PostgreSQL 11/12/13/14

2.7.2.1. Установить на серверы БД необходимые пакеты согласно п. 2.9.7.

2.7.2.2. Выполнить установку и настройку СУБД PostgreSQL 11/12/13/14 в соответствии с документацией на СУБД.

При инициализации СУБД должны быть установлены следующие значения параметров:

```
LC_COLLATE 'en_US.UTF-8'
LC_CTYPE 'en_US.UTF-8'
ENCODING UTF8
```

2.7.2.3. Установить расширения `pg_partman` и `pg_cron` с помощью команд:

– ОС CentOS версии 7 и СУБД PostgreSQL 11:

```
sudo rpm -ivh pg_partman_11-4.6.0-1.rhel7.x86_64.rpm
sudo rpm -ivh pg_cron_11-1.4.1-1.rhel7.x86_64.rpm
```

– ОС CentOS версии 7 и СУБД PostgreSQL 12:

```
sudo rpm -ivh pg_partman_12-4.6.0-1.rhel7.x86_64.rpm
sudo rpm -ivh pg_cron_12-1.4.1-1.rhel7.x86_64.rpm
```

– ОС CentOS версии 7 и СУБД PostgreSQL 13:

```
sudo rpm -ivh pg_partman_13-4.6.0-1.rhel7.x86_64.rpm
sudo rpm -ivh pg_cron_13-1.4.1-1.rhel7.x86_64.rpm
```

– ОС CentOS версии 7 и СУБД PostgreSQL 14:

```
sudo rpm -ivh pg_partman_14-4.6.0-1.rhel7.x86_64.rpm
sudo rpm -ivh pg_cron_14-1.4.1-1.rhel7.x86_64.rpm
```

– ОС Альт 8 СП и СУБД PostgreSQL 11:

```
sudo rpm -ivh pg_partman_11-4.6.0-1.alt.x86_64.rpm
sudo rpm -ivh pg_cron_11-1.4.1-1.alt.x86_64.rpm
```

– ОС РЕД ОС и СУБД PostgreSQL 12:

```
sudo rpm -ivh pg_partman_12-4.6.0-1.redos7.x86_64.rpm
sudo rpm -ivh pg_cron_12-1.4.1-1.redos7.x86_64.rpm
```

– ОС РЕД ОС и СУБД PostgreSQL 13:

```
sudo rpm -ivh pg_partman_13-4.6.0-1.redos7.x86_64.rpm
sudo rpm -ivh pg_cron_13-1.4.1-1.redos7.x86_64.rpm
```

– ОС РЕД ОС и СУБД PostgreSQL 14:

```
sudo rpm -ivh pg_partman_14-4.6.0-1.redos7.x86_64.rpm
sudo rpm -ivh pg_cron_14-1.4.1-1.redos7.x86_64.rpm
```

Указанные rpm-пакеты находятся на DVD с загрузочными модулями ППО в каталоге `/server/install-infra.tar.gz/install-infra/binary/postgresql/pg_partman/`.

2.7.2.4. Перезапустить сервис СУБД PostgreSQL в соответствии с документацией на СУБД.

2.8. Описание настройки подсистем ППО

2.8.1. Описание настройки ПМ

Настройка ПМ заключается в настройке файлового хранилища ПМ.

ПРИМЕЧАНИЕ. Настройку файлового хранилища необходимо выполнять после установки ППО.

Для настройки файлового хранилища ПМ необходимо выполнить следующие действия:

– на Сервере приложений ПМ создать каталог, в котором будут храниться файлы МП (иконки, скриншоты, rpm-пакеты), загружаемые разработчиками, либо каталог, к которому будет монтироваться единое файловое хранилище;

– в случае если файлы МП будут храниться в каталоге на Сервере приложений ПМ, необходимо создать каталог в соответствии с параметром `filestoragePath` конфигурационного файла `config/subsystems/appstore/config.yml`, в созданном каталоге потребуется создать каталог `applications-api` и назначить его владельцем пользователя ocs, под которым работают сервисы ПМ:

```
sudo mkdir -p /ocs/appstore/applications-api
sudo chown ocs:ocs /ocs/appstore/applications-api
```

Параметр `filestoragePath` конфигурационного файла `config/subsystems/appstore/config.yml` может иметь следующий вид:

```
filestoragePath: "/ocs/appstore"
```

– при использовании единого файлового хранилища для хранения файлов МП на Сервере приложений ПМ необходимо создать каталог `/ocs` и назначить его владельцем пользователя ocs, под которым работают сервисы ПМ:

```
sudo mkdir -p /ocs
sudo chown ocs:ocs /ocs
```

ВНИМАНИЕ! При эксплуатации ППО в кластерной конфигурации все ноды Сервера приложений ПМ должны иметь доступ к единому файловому хранилищу. Соответственно, все ноды Сервера приложений ПМ должны быть настроены на работу с данным файловым хранилищем.

Пример настройки единого файлового хранилища и работы Сервера приложений ПМ (ноды Сервера приложений ПМ) приведены в п. 2.9.3 настоящего документа.

2.8.2. Описание настройки ПУ

2.8.2.1. Настройка подключения ПУ к серверу LDAP

Для настройки взаимодействия ППО с сервером LDAP необходимо в секции `ldapServer` конфигурационного файла `config/subsystems/emm/config.yml` задать требуемые значения:

– `address` — адрес расположения сервера LDAP;

– `parentGroup` — группа, с которой будет производиться экспорт данных из сервера LDAP;

– `userCN` — логин технической учетной записи сервера LDAP;

– `password` — пароль от технической учетной записи сервера LDAP;

– `pageSize` — количество элементов, которое будет импортировано за одну итерацию.

Ниже приведен пример настройки подключения к серверу LDAP в `_vars.yml`:

```
ldapServer:
  address:  "ldap://dc01.omptest.example"
  parentGroup: "ou=Test,DC=omptest,DC=local"
  userCN: "OMPTEST\\\\Admin"
  password: "&UJM,ki8"
  pageSize: 1000
```

Описание требований к данным, содержащимся в Active Directory, приведено в таблице (Таблица 11).

ВНИМАНИЕ! Параметры E-mail, First_name, Last_name являются обязательными.

Таблица 11

Параметр	Описание	Примечание
Group	Название группы пользователей. Формат: от 3 до 64 символов	Если группа пользователей МУ уже существует, выполняется привязка группы к пользователям МУ

Параметр	Описание	Примечание
E-mail (обязательный)	Рабочая почта пользователя МУ. Формат: <логин>@<доменное_имя>, от 2 до 256 символов	Рабочая почта пользователя МУ должна быть уникальной
First_name (обязательный)	Имя пользователя МУ. Формат: от 2 до 64 символов. Символы: а-я; a-z; А-Я; A-Z; -; пробел	
Last_name (обязательный)	Фамилия. Формат: от 2 до 64 символов. Символы: а-я; a-z; А-Я; A-Z; -; пробел	
Job_title	Должность. Формат: от 2 до 256 символов. Символы: а-я; a-z; А-Я; A-Z; -; пробел	
Phone_number	Номер телефона. Формат: от 2 до 64 символов. Только цифры	

2.8.2.2. Настройка взаимодействия Сервера приложений ПУ с Сервисом уведомлений Аврора версии 1.1.2

В случае необходимости взаимодействия Сервера приложений ПУ с СУА потребуется выполнить следующие настройки:

2.8.2.2.1 Синхронизировать время между Сервером приложений ПУ и СУА.

2.8.2.2.2 Зарегистрировать в СУА проект и получить конфигурационные файлы с настройками: `push-mobile-app.yml и push-server.yml.`

2.8.2.2.3 Задать публичный адрес СУА (значение параметра должно соответствовать значению параметра `push_public_address` в конфигурационном файле `push-server.yml`). Для этого в конфигурационном файле `config/config.yml.j2` задать параметр `config.publicUris.push.publicAddress`, например:

```
publicUris:
    push:
      publicAddress: "http://ocs-app.local:8009"
```

2.8.2.2.4 Переустановить ПУ в соответствии с п. 2.5.2 настоящего документа.

2.8.2.2.5 В Консоли администратора ПУ («Администрирование» - «Настройки» - «Интеграция» - «Сервер Push») задать параметры взаимодействия Сервера приложений ПУ и СУА.

Описания назначения параметров и порядок настройки приведены в документе «Руководство пользователя. Часть 3. Подсистема Платформа управления».

2.8.3. Описание настройки ПООС

Для загрузки пакетов ОС в файловое хранилище ПООС необходимо выполнить следующие действия:

– скопировать в произвольный каталог файлового хранилища ПООС архив с пакетами ОС и распаковать его в каталог, заданный в параметре `root` секции `location /pkgrepo/mobile` конфигурационного файла `/etc/nginx/conf.d/locations-external/ocs-pkgrepo-nginx-static.location` (по умолчанию каталог: `/ocs/pkgrepo/repos`), либо в параметре `repos_root` конфигурационного файла `install-<версия ППО>/install-ac/config/subsystems/pkgrepo/vars/ocs-pkgrepo-nginx-static.yml` сценариев установки ППО:

```
tar -xf <имя файла с архивом> -C /ocs/pkgrepo/repos
rm <имя файла с архивом>
```

– зарегистрировать переданный релиз (версию), добавив в файл `/ocs/pkgrepo/meta/main.json` описание из переданного вместе с архивом meta-файла. Путь к файлу `main.json` задается в параметре `alias` секции `location /pkgrepo/mobile/meta` конфигурационного файла `/etc/nginx/conf.d/locations-external/ocs-pkgrepo-nginx-static.location` (по умолчанию каталог: `/ocs/pkgrepo/meta`) либо в параметре `meta_root` конфигурационного файла `install-<версия ППО>/install-ac/config/subsystems/pkgrepo/vars/ocs-pkgrepo-nginx-static.yml` сценариев установки ППО. Пример файла `main.json`:

```
{
  "brand": "OMPCert",
  "releases": [
    {
      "deviceModel": "p4903",
      "latest": "3.0.2.23",
      "versions": [
        {
          "version": "3.0.2.22",
      "from": []
        },
        {
          "version": "3.0.2.23",
          "from": [
              "3.0.2.22"
      ]
    }
      ]
    },
    {
      "deviceModel": "l801em",
      "latest": "3.0.2.23",
      "versions": [
        {
          "version": "3.0.2.22",
      "from": []
        },
        {
          "version": "3.0.2.23",
          "from": [
              "3.0.2.22"
      ]
    }
      ]
    }
  ]
}
```

– перезапустить сервис `ocs-pkgrepo-pkg-repo-api` с помощью команды:

```
ANSIBLE_USER=<имя пользователя> ./deploy-ac.sh --apps ocs-pkgrepo-pkg-repo-api --action restart
```

– для проверки корректности настройки необходимо войти в Консоль администратора ПУ, далее перейти в подраздел «Настройки» раздела «Администрирование», в раскрывающемся поле «Интеграция» выбрать вкладку «Обновление ОС». Убедиться, что отображаются имя сервера, модели МУ и доступные версии ОС (Рисунок 3).

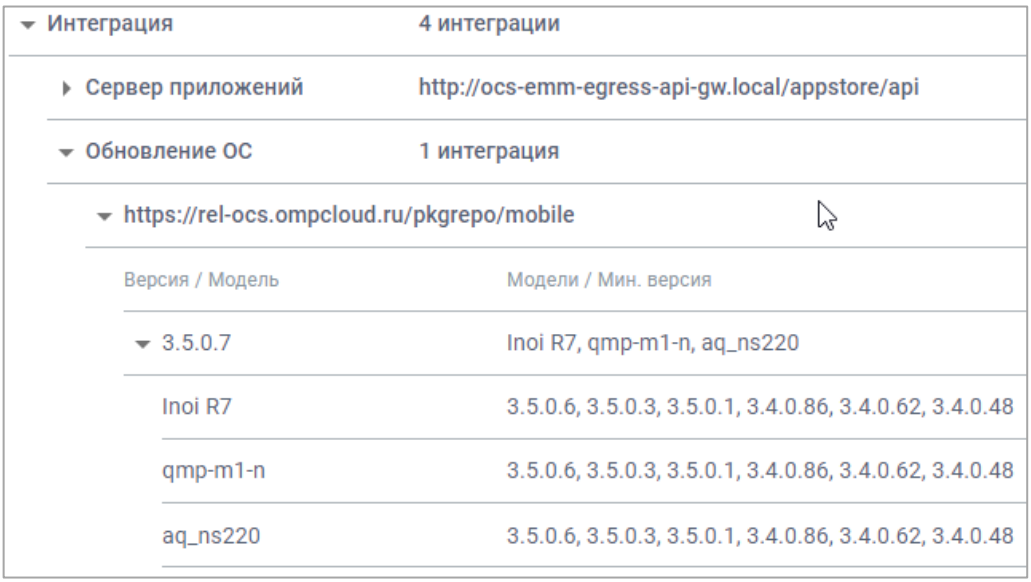


Рисунок 3

2.9. Дополнительные настройки ППО и среды функционирования ППО

2.9.1. Настройка взаимодействия сервера приложений ПУ с SMTP-сервером

Для настройки взаимодействия ППО с SMTP-сервером необходимо в секции `smtp` конфигурационного файла подсистемы ПУ `config.yml` (`config/subsystems/emm/config.yml`) задать требуемые значения:

– адрес эл. почты, с которого отправляются письма (параметр: `from`);

– адрес сервера эл. почты (параметр: `address`);

– тип аутентификации (параметр: `authType`);

– параметры для заданного типа аутентификации (`usrername`, `password` и др.).

ПРИМЕЧАНИЕ. Значения параметров from и username должны быть идентичны, в противном случае почтовый сервер будет отклонять сообщения.

В ПУ поддерживаются `PLAIN`, `CRAM-MD5`, `LOGIN` типы аутентификации `SMTP`. В зависимости от используемого типа аутентификации необходимо задать следующие параметры (остальные параметры оставить без изменений):

– PLAIN:

```
smtp:
  from: "user@example.com"
  address: "smtp.example.com:1025"
  tls: true
  authType: "PLAIN"
  host: "example.com"
  username: "test_username"
  password: "test_password"
  identity: "identity"
```

– CRAM-MD5:

```
smtp:
  from: "user@example.com"
  address: "smtp.example.com:1025"
  tls: true
  authType: "CRAM-MD5"
  username: "test_username"
  secret: "test_secret"
```

– LOGIN:

```
smtp:
  from: "user@example.com"
  address: "smtp.example.com:1025"
  tls: true
  authType: "LOGIN"
  username: "test_username"
  password: "test_password"
```

– без аутентификации:

```
smtp:
  from: "user@example.com"
  address: "smtp.example.com:1025"
  tls: true
```

После изменения настроек необходимо перезапустить сервисы с помощью команды:

```
ANSIBLE_USER=<имя пользователя> ./deploy-ac.sh --apps ocs-emm-
dispatcher-api,ocs-emm-enrollments-api --action restart
```

2.9.2. Настройка разделения трафика

ППО позволяет разделять входящий трафик (url-запросы) следующими способами:

– по basepath - каждая Консоль администратора/разработчика (либо API для взаимодействия с МП) привязана к определенному basepath. Basepath заданы в секции config.publicUris конфигурационного файла internal.yml;

– по доменам (субдоменам) – каждая Консоль администратора/разработчика и API для взаимодействия с МП (либо группа консолей и API) опционально может быть привязана к определенному домену. Рекомендуется публичные консоли и API привязывать к домену, который имеет доступ из сети Интернет, а внутренние консоли (Консоли администраторов) привязывать к домену, не имеющему доступ из сети Интернет. Разделение трафика по доменам описано в пп. 2.4.2.2 настоящего документа.

2.9.3. Пример настройки единого файлового хранилища

Необходимо установить NFS сервер в соответствии с официальной документацией на ОС RedHat, приведенной на странице: https://access.redhat.com/documentation/en-us/red_hat_enterprise_linux/7/html/storage_administration_guide/nfs-serverconfig.

Далее необходимо выполнить монтирование файловой системы NFS к каталогу /ocs с помощью команды:

```
mount example.com:/export/ocsfs /ocs
```

где,

– example.com – имя узла файлового сервера NFS;

– /export/ocsfs – каталог, который экспортирует example.com;

– /ocs – каталог, к которому осуществляется монтирование.

Для проверки корректности монтирования необходимо выполнить команду:

```
ls /ocs
```

и убедиться, что полученный список файлов соответствует списку файлов в каталоге /export/ocsfs на компьютере example.com.

Монтирование файловой системы NFS также может быть выполнено посредством редактирования файла /etc/fstab. Для этого в данный файл необходимо добавить запись следующего вида:

```
example.com:/export/ocsfs /ocs nfs defaults 0 0
```

Редактирование файла /etc/fstab должно осуществляться суперпользователем.

В файловом хранилище необходимо в соответствии с параметром filestoragePath конфигурационного файла config/subsystems/appstore/config.yml создать каталог, где будут храниться файлы МП (иконки, скриншоты, rpm-пакеты), загружаемые разработчиками. В созданном каталоге потребуется создать каталог applications-api:

```
mkdir -p /ocs/appstore/applications-api
```

Параметр filestoragePath конфигурационного файла config/subsystems/appstore/config.yml может иметь следующий вид:

```
filestoragePath: "/ocs/appstore"
```

В файловом хранилище необходимо создать каталог, в котором будут храниться пакеты ОС ПООС:

```
mkdir -p /ocs/pkgrepo
```

Создаваемый каталог должен соответствовать параметрам, заданным в конфигурационном файле /etc/nginx/conf.d/ocs-pkgrepo-nginx-static.conf веб-сервера Nginx Web Server.

2.9.4. Настройка кэширования ответов сервисов

Для увеличения производительности ППО применяется кэширование ответов сервисов с помощью Nginx. При этом доступ к закэшированным данным осуществляется через шлюзы доступа ППО.

Настройки кэширования задаются в следующих конфигурационных файлах сценариев установки среды функционирования ППО:

1) В конфигурационном файле `shared_roles/nginx/defaults/main.yml` задаются:

– `cache_enabled` - включение/выключение кэширования;

– `cache_path` - каталог хранения кэша;

– `keys_zone` - имя зоны в разделяемой памяти, где будет храниться кэш;

– `keys_zone_size` - размер зоны в разделяемой памяти;

– `cache_max_size` - максимальный размер выделяемой под кэш памяти (когда место заканчивается, nginx удаляет устаревшие данные);

– `cache_inactive` - время, после которого кэш будет автоматически очищаться.

Например:

```
cache_enabled: true
cache_path: "/var/cache/nginx"
keys_zone: "proxy_cache"
keys_zone_size: "50m"
cache_max_size: "10G"
cache_inactive: "30m"
```

ПРИМЕЧАНИЕ. Максимальный размер выделяемой под кэш памяти должен быть не менее 10 ГБ.

2) В конфигурационных файлах `config/subsystems/<название подсистемы>/vars/services.yml` задаются API функции (endpoint-ы) ППО для которых необходимо выполнять кэширование, а также параметры кэширования для каждой API функции:

– `proxy_cache` - включение кэширования для API функции;

– `proxy_cache_valid` - время кэширования ответа (возможно задать время кэширования для определенных статусов ответа);

– `proxy_cache_lock` - параметр определяет возможность прохождения нескольких запросов на бэкенд (к сервисам ППО). При значении «on» запрещается прохождение нескольких запросов к сервису ППО, все повторные запросы будут ожидать появления ответа в кэше, либо таймаут блокировки запроса к странице;

– `proxy_cache_use_stale` - параметр определяет, в каких случаях можно использовать устаревший закэшированный ответ;

– `add_header: "X-Cache-Status  $upstream_cache_status"` - директива добавляет http-заголовок, содержащий статус кэширования.

Например:

```
...
      nginx_location_dashboard:
        path: "~ /v1/dashboards/[^/]+$"
        proxy_cache: "proxy_cache"
        proxy_cache_valid: "200 {{ cache_interval_dynamic }}"
        proxy_cache_lock: "on"
        proxy_cache_use_stale: "updating"
        proxy_cache_background_update: "on"
        add_header: "X-Cache-Status $upstream_cache_status"
```

2.9.5. Действия по безопасной установке и настройке средства

ПРИМЕЧАНИЕ. Установка, настройка и эксплуатация ППО должна осуществляться в соответствии с ЭД на ППО.

При использовании ППО в ГИС (ИСПДн, АСУ, КИИ), не содержащих информации составляющей государственной тайны, в зависимости от класса защищенности, должны быть установлены значения параметров, приведенные в таблице (Таблица 12).

Таблица 12

Параметр	Значение (для ГИС 4-го класса)	Значение (для ГИС 3-го класса)	Значение (для ГИС 2-го класса)	Значение (для ГИС 1-го класса)
Конфигурационный файл ПБ (сценария установки ПБ): `/var/ocs/config/subsystems/auth/config.yml` (`config/subsystems/auth/config.yml`)				
Период времени неиспользования идентификатора (учетной записи) пользователя, через	Устанавливается на усмотрение оператора ИС, например:	Не более 90 дней, например: `maxAccountInactivityPeriod: 2160h`	Не более 90 дней, например: `maxAccountInactivityPeriod: 2160h`	Не более 45 дней, например: `maxAccountInactivityPeriod: 1080h`

Параметр	Значение (для ГИС 4-го класса)	Значение (для ГИС 3-го класса)	Значение (для ГИС 2-го класса)	Значение (для ГИС 1-го класса)
которое происходит его блокирование: `config.maxAccountInactivityPeriod`	`maxAccountInactivityPeriod: 2160h`			
Минимальная длина пароля: `config.passwordSettings.minLength`	Не менее 6 символов, например: `config.passwordSettings.minLength: 6`	Не менее 6 символов, например: `config.passwordSettings.minLength: 6`	Не менее 6 символов, например: `config.passwordSettings.minLength: 6`	Не менее 8 символов, например: `config.passwordSettings.minLength: 8`
Алфавит пароля для учетных записей пользователей не настраивается. Пароли учетных записей пользователей должны содержать буквы верхнего и нижнего регистров, цифры и специальные символы (это контролируется ППО). Алфавит пароля для учетных записей МУ: • минимальное число цифр в пароле: `config.passwordSettings.minDigits` • минимальное число букв верхнего регистра в пароле: `config.passwordSettings.minUpperLetters`	Не менее 30 символов, например: `minDigits: 1` `minUpperLetters: 0` `minLowerLetters: 1` `minSpecialChars: 0`	Не менее 60 символов, например: `minDigits: 1` `minUpperLetters: 1` `minLowerLetters: 1` `minSpecialChars: 0`	Не менее 70 символов, например: `minDigits: 1` `minUpperLetters: 1` `minLowerLetters: 1` `minSpecialChars: 1`	Не менее 70 символов, например: `minDigits: 1` `minUpperLetters: 1` `minLowerLetters: 1` `minSpecialChars: 1`

Параметр	Значение (для ГИС 4-го класса)	Значение (для ГИС 3-го класса)	Значение (для ГИС 2-го класса)	Значение (для ГИС 1-го класса)
• минимальное число букв нижнего регистра в пароле: `config.passwordSettings.minUpperLetters` • минимальное число спецсимволов в пароле: `config.passwordSettings.minSpecialChars`				
Максимальное время действия пароля: `config.passwordExpirationTime`	Не более 180 дней, например: `passwordExpirationTime: "4320h"`	Не более 120 дней, например: `passwordExpirationTime: "2880h"`	Не более 90 дней, например: `passwordExpirationTime: "2160h"`	Не более 60 дней, например: `passwordExpirationTime: "1440h"`
Число последних использованных паролей, которые запрещено использовать пользователями при создании новых паролей: `config.passwordHistoryDepth`	Устанавливается на усмотрение оператора ИС, например: `passwordHistoryDepth: 3`	Устанавливается на усмотрение оператора ИС, например: `passwordHistoryDepth: 3`	Устанавливается на усмотрение оператора ИС, например: `passwordHistoryDepth: 3`	Устанавливается на усмотрение оператора ИС, например: `passwordHistoryDepth: 3`
Максимальное количество неуспешных попыток аутентификации (ввода неправильного пароля) до блокировки: `config.failedLoginTries`	От 3 до 10 попыток, например: `failedLoginTries: 10`	От 3 до 10 попыток, например: `failedLoginTries: 10`	От 3 до 8 попыток, например: `failedLoginTries: 8`	От 3 до 4 попыток, например: `failedLoginTries: 4`

Параметр	Значение (для ГИС 4-го класса)	Значение (для ГИС 3-го класса)	Значение (для ГИС 2-го класса)	Значение (для ГИС 1-го класса)
Время блокировки учетной записи пользователя в случае достижения установленного максимального количества неуспешных попыток аутентификации: `config.failedLoginBlockTime`	От 3 до 15 минут, например: `failedLoginBlockTime: "3m"`	От 5 до 30 минут, например: `failedLoginBlockTime: "5m"`	От 10 до 30 минут, например: `failedLoginBlockTime: "10m"`	От 15 до 60 минут, например: `failedLoginBlockTime: "15m"`
Количество одновременных сессий для привилегированных учетных записей: `config.privilegedSessionsLimit`	Устанавливается на усмотрение оператора ИС, например: `privilegedSessionsLimit: 10`	Устанавливается на усмотрение оператора ИС, например: `privilegedSessionsLimit: 10`	Устанавливается на усмотрение оператора ИС, например: `privilegedSessionsLimit: 10`	Не более 2-х, например: `privilegedSessionsLimit: 2`
Количество одновременных сессий для непривилегированных учетных записей: `config.unprivilegedSessionsLimit`	Устанавливается на усмотрение оператора ИС, например: `unprivilegedSessionsLimit: 10`	Устанавливается на усмотрение оператора ИС, например: `unprivilegedSessionsLimit: 10`	Устанавливается на усмотрение оператора ИС, например: `unprivilegedSessionsLimit: 10`	Устанавливается на усмотрение оператора ИС, например: `unprivilegedSessionsLimit: 10`
Общий конфигурационный файл ППО (шаблон общего конфигурационного файла ППО): `/var/ocs/config/config.yml (config/config.yml.j2)`				
Время бездействия (неактивности) пользователя, через которое осуществляется завершение сеанса пользователя:	Устанавливается на усмотрение оператора ИС, например: `rememberFor: 30m`	Устанавливается на усмотрение оператора ИС, например: `rememberFor: 30m`	Не более 15 минут, например: `rememberFor: 15m`	Не более 5 минут, например: `rememberFor: 5m`

Параметр	Значение (для ГИС 4-го класса)	Значение (для ГИС 3-го класса)	Значение (для ГИС 2-го класса)	Значение (для ГИС 1-го класса)
config.session.rememberFor				

2.9.6. Действия по реализации функций безопасности среды функционирования ППО

2.9.6.1. Установка, настройка и эксплуатация СЗИ НСД

Эксплуатация ППО и СУБД должна осуществляться в одной из следующих ОС:

– CentOS версии 7 с установленными СЗИ НСД «Dallas Lock Linux», СЗИ «Secret Net LSP» или СЗИ НСД «Аккорд-Х К»;

– Альт 8 СП.

Установка СЗИ НСД должна осуществляться после установки ППО. После установки СЗИ НСД необходимо повторно назначить пользователям ОС права на выполнение команд от имени суперпользователя в соответствии с подразделом 2.2 настоящего документа.

Установка, настройка и эксплуатация СЗИ НСД и ОС Альт 8 СП должна осуществляться в соответствии с ЭД на СЗИ (ОС).

2.9.6.2. Требования к межсетевому экранированию

Необходимо, чтобы защита периметра (физических и (или) логических границ) ИС осуществлялась с использованием межсетевого экрана требуемого класса защиты.

Межсетевой экран должен пропускать трафик только на внешние порты ППО, при этом остальной трафик должен быть запрещен. Перечень внешних портов ППО в зависимости от варианта настройки приведен в таблице (Таблица 13).

Таблица 13

Номер порта (протокол)	Описание	Конфигурационный файл, в котором задается порт	Тип порта[4]
Сервисы ППО «Аврора Центр»			
10000 - 10500 (tcp)	Порты сервисов ППО	`shared_roles/systemd-deploy/templates/systemd-supplicant.sh.j2 /usr/bin/systemd-supplicant.sh`	внутренний
Nginx			
80 (tcp)	Служит для взаимодействия сервисов ППО друг с другом и используется в том случае, когда осуществляется разделение трафика по basepath (url-адресам)	`shared_roles/consul-template/defaults/main.yml`	внутренний
8009 (tcp)	Балансировщик микросервисов (Nginx Web Server). Используется в том случае, когда осуществляется разделение трафика по basepath (url-адресам)	`config/vars/_vars.yml config/config.yml.j2 /etc/nginx/conf.d/ocs.conf`	внешний
СУБД PostgreSQL			
5432 (tcp)	СУБД PostgreSQL	`shared_roles/postgresql/defaults/main.yml`	внутренний
СУБД Redis			

[4] Типы портов:

1. Внешние - доступ к данному типу портов осуществляется из-за пределов контролируемой зоны. Например, запросы от пользователей с ролью Пользователь Аврора Маркет. Доступ к данным портам имеет нарушитель;
2. Внутренние - доступ к данному типу портов может осуществляться только из контролируемой зоны. Данные порты используются для взаимодействия: между сервисами ППО, сервисов ППО с компонентами среды функционирования ППО, компонентами среды функционирования ППО, привилегированных пользователей с ППО.

Номер порта (протокол)	Описание	Конфигурационный файл, в котором задается порт	Тип порта[4]
6379 (tcp)	redis-server	shared_roles/redis/defaults/main.yml	внутренний
26379 (tcp)	redis-sentinel	shared_roles/redis/defaults/main.yml	внутренний
Consul			
8300 (tcp)	https://www.consul.io/docs/install/ports		внутреннее
8301 (tcp/udp)	https://www.consul.io/docs/install/ports		внутренний
8302 (tcp/udp)	https://www.consul.io/docs/install/ports		внутренний
8600 (tcp/udp)	https://www.consul.io/docs/install/ports	shared_roles/consul/defaults/main.yml	внутренний
8500 (tcp)	https://www.consul.io/docs/install/ports	shared_roles/consul/defaults/main.yml	внутренний
Nats Streaming Server			
4222 (tcp)	nats_port	shared_roles/nats-streaming-server/defaults/main.yml	внутренний
6222 (tcp)	nats_cluster_port	shared_roles/nats-streaming-server/defaults/main.yml	внутренний
8222 (tcp)	nats_monitoring_port	shared_roles/nats-streaming-server/defaults/main.yml	внутренний
Dnsmasq			
53	dnsmasq		внутренний
Операционная система			
22	Порт SSH. Используется для развертывания и администрирования ППО. **ВНИМАНИЕ!** Возможность использования		внутренний

Номер порта (протокол)	Описание	Конфигурационный файл, в котором задается порт	Тип порта[4]
	данного порта определяется документацией СЗИ от НСД		

ПРИМЕЧАНИЕ. Рекомендуется запретить доступ к ППО привилегированных пользователей из-за пределов контролируемой зоны, запретив доступ к Консоли администратора ПБ. Также при необходимости можно запретить доступ к остальным веб-консолям. Для этого необходимо разрешить трафик только по требуемым url-адресам в соответствии с п. 2.9.2 настоящего документа.

2.9.6.3. Настройка ОС CentOS

2.9.6.3.1 Для затруднения возможностей сбора информации о системе, необходимо исключить метки времени из заголовков TCP пакетов, выполнив описанные ниже действия.

2.9.6.3.1.1 В конфигурационный файл `/etc/sysctl.conf` добавить строку:

```
net.ipv4.tcp_timestamps = 0
```

2.9.6.3.1.2 Применить конфигурацию, выполнив команду:

```
sysctl -p /etc/sysctl.conf
```

2.9.6.3.1.3 Проверить корректность конфигурации, выполнив команду:

```
sysctl -a | grep net.ipv4.tcp_timestamps
```

Если настройки заданы правильно, должно быть выведено значение:

```
net.ipv4.tcp_timestamps = 0
```

2.9.6.3.2 Настройка запрета `ssh` доступа к серверам приложений по логину и паролю

Для выполнения настройки необходимо выполнить следующие действия:

2.9.6.3.2.1 В конфигурационном файле `/etc/ssh/sshd_config` задать следующие значения параметров:

```
PasswordAuthentication no
AuthenticationMethods publickey
```

2.9.6.3.2.2 Перезапустить службу sshd с помощью команды:

```
sudo service sshd reload
```

2.9.6.3.3 Настройка минимальной сложности пароля

Настройка сложности пароля осуществляется в конфигурационном файле `/etc/security/pwquality.conf`. Рекомендуется задать следующие значения параметров:

– минимальная длина пароля:

```
minlen = 8
```

– алфавит пароля (минимальное количество используемых классов символов):

```
minclass = 4
```

– максимальная длина последовательности символов (abcd, 12345 и т.п.):

```
maxsequence = 3
```

– максимальное число идущих подряд одинаковых символов:

```
maxrepeat = 3
```

2.9.6.4. Защита Nginx от Slowloris Attack

Slowloris Attack - это тип атаки «отказ от обслуживания» на веб-сервер. В рамках данной атаки осуществляется попытка открытия множества соединений с целевым веб-сервером и удержания их открытыми как можно дольше. Это достигается путем открытия соединений с целевым веб-сервером и отправки частичного запроса. Пораженный сервер будет поддерживать соединения открытыми, заполняя, таким образом, свой максимальный пул параллельных соединений, что в конечном итоге не позволит создавать новые соединения.

Для защиты Nginx от данной атаки необходимо в зависимости от условий эксплуатации установить значения следующих параметров как можно меньше:

– `client_body_timeout` - интервал времени, на протяжении которого сервер будет ждать тело запроса;

– `client_header_timeout` - интервал времени, на протяжении которого сервер будет ждать заголовок запроса;

– `keepalive_timeout` - лимит времени ожидания keep-alive соединения.

Рекомендуется установить следующие значения параметров:

```
client_body_timeout 10;
client_header_timeout 10;
keepalive_timeout 10;
```

2.9.7. Самостоятельная установка необходимых пакетов на серверы приложений и серверы БД

Для установки необходимых пакетов на серверы приложений и серверы БД необходимо выполнить действия, описанные ниже.

2.9.7.1. Получить список необходимых пакетов

Перечень необходимых пакетов, которые должны быть установлены на серверы приложений и серверы БД, задан в файле `play-managed-node-prerequisites.yml`, находящемся в каталоге со сценариями установки ППО. Данный файл имеет следующую структуру:

```
...
    - name: install requirements to <операционная система>
...
        - name: install os packages on db node
          loop:
            <перечень пакетов сервера БД>
        - name: install os packages on app node
          loop:
            <перечень пакетов сервера приложений>
```

В секции `name: install requirements to <операционная система>` задается перечень пакетов для указанной ОС. Данная секция содержит две подсекции, в которых задается перечень пакетов для сервера приложений и сервера БД.

В подсекции `name: install os packages on db node` задается перечень пакетов для сервера БД.

В подсекции `name: install os packages on app node` задается перечень пакетов для сервера приложений.

Далее представлен пример перечня пакетов для сервера приложений и сервера БД, функционирующих под управлением ОС CentOS 7:

```
...
  tasks:
    - name: install requirements to CentOS7
      block:
        - debug:
            msg: install requirements to CentOS7

        - name: install os packages on db node
          package:
            name: "{{ item }}"
            state: present
          loop:
            - epel-release
            - jq
            - unzip
            - perl-libs
            - libxslt
            - postgresql-libs
            - libicu
          when: node_type == "db" or node_type == "all"

        - name: install os packages on app node
          package:
            name: "{{ item }}"
            state: present
          loop:
            - net-tools
            - epel-release
            - jq
            - unzip
            - perl-libs
            - libxslt
            - postgresql-libs
            - libicu
            - dnsmasq
            - bind-utils
          when: node_type == "app" or node_type == "all"
      when: ansible_distribution == "CentOS" and
ansible_distribution_major_version == "7"
```

2.9.7.2. Установить пакеты

Установка пакетов осуществляется в соответствии с документацией на ОС.

2.9.8. Требования к установке и настройке внешнего балансировщика (на примере Nginx)

Установки и настройка внешнего балансировщика Nginx осуществляется пользователями (системными администраторами) ППО самостоятельно. Внешний балансировщик должен поддерживать проксирование tcp-соединений.

Для проверки возможности проксирования tcp-соединений необходимо выполнить проверку корректности конфигурации Nginx с помощью команды:

```
sudo nginx -t
```

При появлении сообщения `unknown directive "stream"` требуется добавить поддержку модуля ngx_stream_module.so. Для этого необходимо:

– в конфигурационном файле Nginx (файл: `/etc/nginx/nginx.conf`) добавить строку:

```
load_module '/usr/lib64/nginx/modules/ngx_stream_module.so';
```

– перезапустить Nginx с помощью команды:

```
sudo systemctl reload nginx
```

2.9.8.1. Настройка балансировщика для поддержки мультитенантной конфигурации

Так как для каждого тенанта используется отдельный поддомен, то необходимо настроить обработку поддоменов на внешнем балансировщике. Для этого необходимо:

– выпустить сертификаты вида `*.acenter.example.ru` и `acenter.example.ru`;

– добавить dns-записи `*.acenter.example.ru` и `acenter.example.ru`;

– в конфигурационном файле внешнего балансировщика добавить обработку dns-записей `*.acenter.example.ru` (примеры конфигурационных файлов приведены в `samples/ac-mt/nginx_external-balancer/conf.d/one-node.conf`).

2.9.9. Активация (разблокировка) учетной записи пользователя с помощью sql-запроса к БД

Разблокировка учетных записей пользователей ППО осуществляется Администратором учетных записей с помощью Консоли администратора ПБ. Однако учетная запись Администратора учетных записей также может быть заблокирована (например, при длительной неактивности Администратора учетных записей).

В этом случае для разблокировки учетной записи необходимо выполнить следующие действия:

2.9.9.1. Подключится к БД ПБ (auth) с помощью следующей команды:

```
psql -U auth -h <ip-адрес сервера БД> -d auth
```

Например:

```
psql -U auth -h 192.168.0.107 -d auth
```

2.9.9.2. Разблокировать учетную запись пользователя с помощью с sql-запроса:

```
update accounts_users.accounts set is_active=true,
last_activity_at=now() where login='<email пользователя>';
```

Например:

```
update accounts_users.accounts set is_active=true,
last_activity_at=now() where login='admin@omprussia.ru';
```

2.9.10. Действия после сброса МУ к заводским настройкам

Сброс МУ возвращает его к заводским настройкам. После сброса МУ, в зависимости от того, каким образом были первоначально установлены МП ППО (МП «Аврора Центр» и МП «Аврора Маркет»), могут отсутствовать либо быть сброшены до первоначальной версии.

После сброса МУ необходимо выполнить следующие действия:

1) Установить МП ППО, если после сброса МУ они отсутствуют;

2) Активировать МУ в ПУ в соответствии с документом «Руководство пользователя. Часть 3. Подсистема Платформа управления»;

3) Обновить МП ППО в соответствии с подразделом 5.3 настоящего документа.

2.9.11. Порядок задания адресов (доменных имен) в конфигурационном файле inventories/hosts.yml

Задание адресов (доменных имен) осуществляется посредством их добавления в секцию `hosts`, например:

```
...
        app:
          hosts:
            acenterapp01:
            acenterapp02:
            acenterapp03:
```

Допускается добавление адресов при помощи добавления хостов в группы и дальнейшего переиспользования групп. Например, для Nginx будут заданы адреса из группы `app`, которая заполнена выше:

```
...
    ocs:
      children:
        app:
          hosts:
            acenterapp01:
            acenterapp02:
            acenterapp03:
        nginx:
          children:
            app:
```

Также допускается смешанное задание адресов посредством их добавления в секцию `hosts`, а также посредством добавления хостов в группы и дальнейшего переиспользования групп. Например, для Nginx будут заданы адреса из группы `app`, которая заполнена выше и адреса из секции `hosts`:

```
...
    ocs:
      children:
        app:
```

```
          hosts:
            acenterapp01:
            acenterapp02:
            acenterapp03:
        nginx:
          children:
            app:
          hosts:
            acenterapp04:
            acenterapp05:
```

При необходимости установки на хост только определенных подсистем ППО, необходимо после адреса хоста добавить параметр `subsystems` с перечнем подсистем, например:

```
...
        app:
          hosts:
            acenterapp01:
              subsystems: auth
            acenterapp02:
              subsystems: emm
            acenterapp03:
              subsystems: appstore,pkgrepo
```

Конфигурационный файл сценария установки среды функционирования ППО на одной ЭВМ с доменным именем `ocs-app.local` имеет следующий вид:

```
all:
  children:
    ocs:
      children:
        app:
          hosts:
            ocs-app.local:
        postgresql:
          children:
            postgresql_masters:
              hosts:
                ocs-app.local:
            postgresql_slaves:
              hosts:
        nginx:
          children:
            app:
          hosts:
        consul:
          children:
```

```
            consul_servers:
              children:
                app:
              hosts:
            consul_agents:
        consul_template:
          children:
            app:
        nats_streaming_server:
          children:
                app:
          hosts:
        redis:
          children:
            redis_masters:
              children:
                app:
              hosts:
            sentinel:
              children:
                app:
              hosts:
```

Примеры файлов `hosts.yml` для однонодовой и кластерной конфигураций приведены в каталоге `samples/ac/inventories/` (или в каталоге `samples/ac-mt/inventories/`).

Описание параметров конфигурационного файла `inventories/hosts.yml` приведено в п. 7.1.1 настоящего документа.

2.9.12. Порядок настройки срока хранения событий безопасности

Срок хранения событий безопасности задается в поле `retention` таблицы `partman.part_config events` БД ПБ (`auth`).

Для просмотра и изменения срока хранения необходимо выполнить следующую последовательность действий:

2.9.12.1. Подключиться к БД ПБ (`auth`) с помощью следующей команды:

```
psql -U auth -h <ip-адрес сервера БД> -d auth
```

Например:

```
psql -U auth -h 192.168.0.107 -d auth
```

2.9.12.2. Просмотреть текущее значение срока хранения с помощью скрипта:

```
select retention from partman.part_config where parent_table =
'audit.audit_events';
```

2.9.12.3. Изменить срок хранения с помощью скрипта:

```
UPDATE partman.part_config SET retention = '<количество дней> days'
where parent_table = 'audit.audit_events';
```

Например:

```
UPDATE partman.part_config SET retention = '90 days' where
parent_table = 'audit.audit_events';
```

2.9.13. Порядок настройки ППО для его установки на различные окружения

Сценарии установки позволяют выполнить настройку и установку ППО, а также компонентов среды функционирования ППО для нескольких различных окружений. Для того чтобы выполнить настройку и установку ППО для заданного окружения необходимо выполнить следующие действия:

2.9.13.1. Перейти в каталог (каталог: `/install-<версия ППО>/install-ac/` или `/install-<версия ППО>/install-ac-mt/`).

2.9.13.2. Создать инвентарный файл `hosts.yml` по следующему пути: `inventories/<название окружения>/hosts.yml`

Описание параметров конфигурационного файла `hosts.yml` приведено в п. 7.1.1 настоящего документа.

2.9.13.3. Создать каталог `config/environments/<название окружения>/`, создать в данном каталоге требуемые конфигурационные файлы с учетом их расположения в каталоге `config` и задать требуемые значения параметров

Более подробная информация по работе с конфигурационными файлами окружения приведена в п. 8.2.8.

2.9.13.4. Выполнить установку компонентов среды функционирования ППО и ППО в соответствии с подразделом 2.5 для заданного окружения, указав в командах установки путь к инвентарному файлу и имя окружения

Примеры команд:

– команда установки всех пакетов на все серверы (на все серверы приложений и серверы БД независимо от их типа):

```
ansible-playbook -i inventories/<название окружения>/hosts.yml play-managed-node-prerequisites.yml -vv -u <имя пользователя>
```

– команда установки компонентов среды функционирования ППО:

```
ANSIBLE_USER="<имя пользователя>" ./deploy-infra.sh --env "<название окружения>"
```

– команда установки ППО:

```
ANSIBLE_USER=<имя пользователя> ./deploy-ac.sh --env "<название окружения>"
```

2.9.14. Удаление персональных данных из учетной записи пользователя и персональных данных контактного лица организации

2.9.14.1. Для удаления персональных данных из учетной записи пользователя необходимо выполнить следующие действия:

– подключиться к БД ПБ (`auth`) с помощью следующей команды:

```
psql -U auth -h <ip-адрес сервера БД> -d auth
```

Например:

```
psql -U auth -h 192.168.0.107 -d auth
```

– выполнить sql-запрос:

```
update accounts_users.accounts set login='', last_name='', first_name='', patronymic='' where login='<email пользователя>';
```

Например:

```
update accounts_users.accounts set login='', last_name='', first_name='', patronymic='' where login='ivanov@omprussia.ru';
```

2.9.14.2. Для удаления персональных данных контактного лица организации необходимо выполнить следующие действия:

– подключиться к БД ПУТ (`mt`) с помощью следующей команды:

```
psql -U mt -h <ip-адрес сервера БД> -d mt
```

Например:

```
psql -U mt -h 192.168.0.107 -d mt
```

– выполнить sql-запрос:

```
update organizations.contact_persons set first_name='', last_name='',
patronymic='', email='' where email='<email контактного лица>';
```

Например:

```
update organizations.contact_persons set first_name='', last_name='',
patronymic='', email='' where email='ivanov@omprussia.ru';
```

2.9.15. Сброс пароля учетной записи

В случае утери пароля от учетной записи с ролью Администратор учетных записей и не возможности его восстановления штатным способом (например, если в ППО была только одна учетная запись с указанной ролью), необходимо выполнить следующие действия для сброса пароля учетной записи:

– подключиться к БД ПБ (`auth`) с помощью следующей команды:

```
psql -U auth -h <ip-адрес сервера БД> -d auth
```

Например:

```
psql -U auth -h 192.168.0.107 -d auth
```

– в файле `samples/sql/activate_user_account.sql`, находящимся в каталоге со сценариями установки ППО, задать логин учетной записи в параметре `accountLogin`, например:

```
accountLogin text := 'admin@omprussia.ru'
```

– скопировать содержимое файла `activate_user_account.sql` в консоль и выполнить скрипт, нажав клавишу «Enter».

После выполнения указанных действия пароль будет иметь значение «admin».

2.9.16. Восстановление учетной записи пользователя тенанта в случае ее удаления

Для восстановления учетной записи пользователя тенанта в случае ее удаления необходимо выполнить следующие действия:

– подключиться к БД ПБ (`auth`) с помощью следующей команды:

```
psql -U auth -h <ip-адрес сервера БД> -d auth
```

Например:

```
psql -U auth -h 192.168.0.107 -d auth
```

– в файле `samples/sql/create_tenant_default_user_account.sql`, находящимся в каталоге со сценариями установки ППО, задать логин учетной записи (параметр: `accountLogin`) и код тенанта (параметр: `accountTenantCode`), например:

```
accountLogin text := 'admin@omprussia.ru';
accountTenantCode text := 'default';
```

ПРИМЕЧАНИЕ. Код тенанта можно посмотреть в карточке тенанта.

– скопировать содержимое файла `create_tenant_default_user_account.sql` в консоль и выполнить скрипт, нажав клавишу «Enter».

2.10. Установка МП

2.10.1. Установка МП на МУ с помощью приложения «Терминал»

Для установки МП на МУ с помощью приложения «Терминал» необходимо выполнить следующие действия:

1) подключить МУ к ЭВМ с помощью USB-кабеля;

2) на МУ переключиться в режим «Протокол передачи мультимедиа (MTP)», в результате в ОС отобразится внешний носитель «INOI R7» (Рисунок 4);

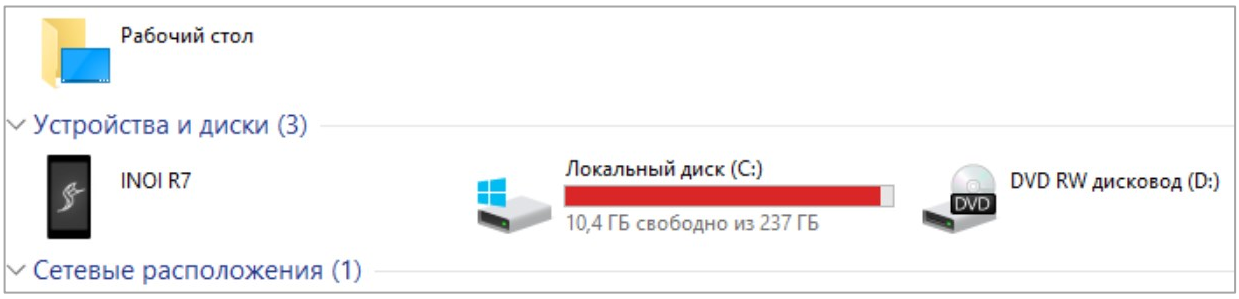


Рисунок 4

3) перейти в каталог Downloads и скопировать в него загрузочный модуль МП (RPM-пакеты);

Загрузочный модуль, в зависимости от версии ОС, находится на DVD с загрузочным модулем в каталоге:

– МП «Аврора Центр»:

`/mobile-apps/aurora_center/<версия ОС>/`

Например:

`/mobile-apps/aurora_center/3.2.1/` (для ОС Аврора версии 3.2.1).

– МП «Аврора Маркет»:

`/mobile-apps/aurora_market/<версия ОС>/`

Например:

`/mobile-apps/aurora_market/3.2.1/` (для ОС Аврора версии 3.2.1);

4) используя МП «Терминал», перейти в каталог с RPM-пакетами с помощью команды:

```
cd /home/nemo/Downloads/
```

Предварительно необходимо задать пароль для МП «Терминал», выполнив следующие действия:

– провести по Экрану приложений снизу вверх и коснуться значка  . Отобразится меню настроек;

– в меню настроек перейти к разделу «Настройка защиты»;

– выбрать пункт «Доступ к терминалу» и сгенерировать пароль (либо задать пароль вручную);

5) установить пакеты с помощью команды:

```
devel-su pkcon install-local *.rpm
```

2.10.2. Установка МП на МУ с помощью ПУ

Установка МП «Аврора Маркет» также может осуществляться с помощью ПУ. Подробное описание данного процесса приведено в документе «Руководство пользователя. Часть 3. Подсистема Платформа управления».

2.11. Проверка корректности установки и функционирования ППО

2.11.1. Общие сведения

Для проверки корректности установки и функционирования ППО, а также среды функционирования ППО в состав сценариев установки включена утилита для формирования диагностического отчета.

Для формирования диагностического отчета необходимо перейти в каталог со сценариями установки (каталог: `install-<версия ППО>/install-ac/` или `install-<версия ППО>/install-ac-mt/`) и выполнить команду:

```
ansible-playbook play-diagnostic-report.yml -i inventories/hosts.yml -
vv --user <имя пользователя>
```

В результате в каталоге report будет сформирован файл в report.html.

Диагностический отчет формируется в виде файла в формате HTML и содержит следующие разделы:

– общая информация о статусе сервисов ППО;

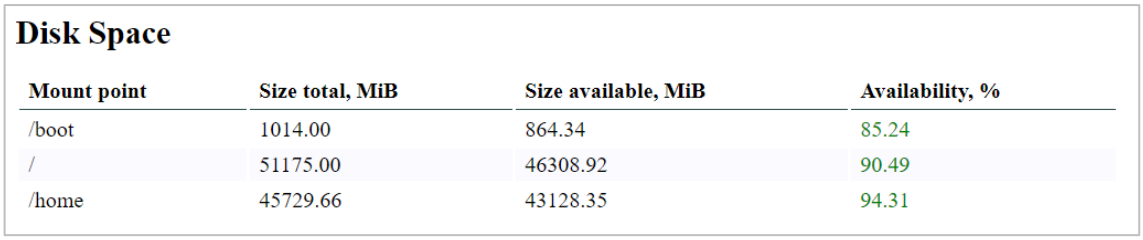
– общая информация о статусе компонентов среды функционирования;

– разделы, содержащие детальную информацию об отдельных сервисах ППО и компонентах среды функционирования.

2.11.2. Описание параметров диагностического отчета

2.11.2.1. Раздел «Disk Space»

Раздел содержит информацию о полном и доступном объеме дискового пространства для ППО (Рисунок 5).



Disk Space

Mount point	Size total, MiB	Size available, MiB	Availability, %
/boot	1014.00	864.34	85.24
/	51175.00	46308.92	90.49
/home	45729.66	43128.35	94.31

Рисунок 5

Описание назначения столбцов таблицы, а также информация о возможных значениях приведены в таблице (Таблица 14).

Таблица 14

Название столбца	Описание	Возможные значения (примеры значений)
Mount points	Каталог, к которому монтируется файловое хранилище (точка монтирования)	Путь к каталогу, например: `/home`.
Size total, MiB	Размер файлового хранилища, примонтированного к заданному каталогу	Объем физической памяти в Мб, например: 45729,66
Size available, MiB	Объем свободного места в файловом хранилище, примонтированного к заданному каталогу	Объем физической памяти в Мб, например: 43128,35
Availability, %	Объем свободного места в файловом хранилище в процентном соотношении (к полному объему)	От 0 до 100, например: 94.31

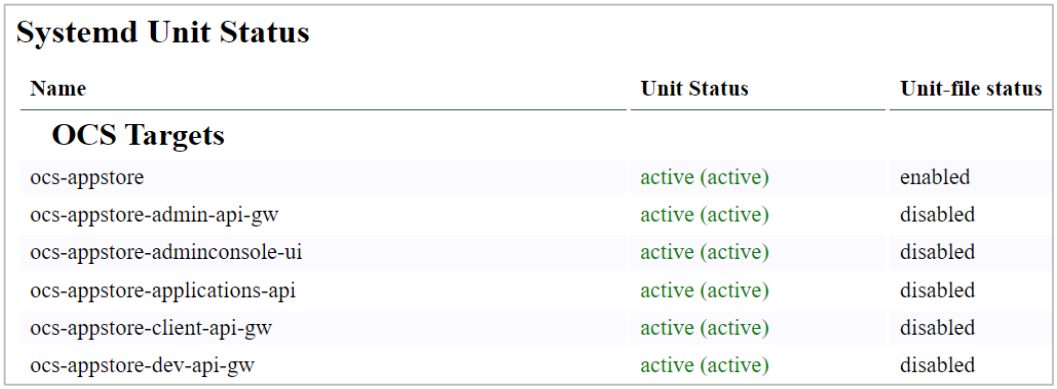
ПРИМЕЧАНИЕ. В случае если объем свободного места менее 15%, поле закрашивается цветом.

2.11.2.2. Раздел «Systemd Unit Status»

В данном разделе приведена общая информация о статусе сервисов ППО и компонентов среды функционирования и состоит из следующих подразделов:

2.11.2.2.1 OCS Targets

Подраздел содержит информацию о статусе конфигураций групп сервисов ППО (Рисунок 6).



Systemd Unit Status

Name	Unit Status	Unit-file status
OCS Targets		
ocs-appstore	active (active)	enabled
ocs-appstore-admin-api-gw	active (active)	disabled
ocs-appstore-adminconsole-ui	active (active)	disabled
ocs-appstore-applications-api	active (active)	disabled
ocs-appstore-client-api-gw	active (active)	disabled
ocs-appstore-dev-api-gw	active (active)	disabled

Рисунок 6

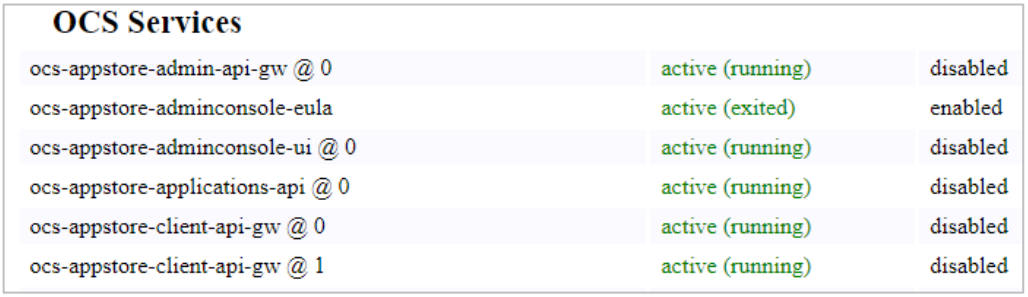
Описание назначения столбцов таблицы, а также информация о возможных значениях приведены в таблице (Таблица 15).

Таблица 15

Название столбца	Описание	Возможные значения (примеры значений)
Name	Имя конфигурации группы сервисов	Возможные значения определяются перечнем конфигураций групп сервисов ППО
Unit Status	Информация о статусе группы сервисов	`active` - группа сервисов запущена и выполняется; `activating` - группа сервисов запускается; `deactivating` - группа сервисов выключается; `inactive` - группа сервисов выключена; `failed` – при запуске группы сервисов произошла ошибка; `missed` - компонент отсутствует
Unit-file status	Информация о присутствии конфигурационного файла запуска группы сервисов в автозапуске	`enabled` - присутствует в автозапуске `disabled` - отсутствует в автозапуске

2.11.2.2.2 OCS Services

Подраздел содержит информацию о статусе сервисов ППО (Рисунок 7).



OCS Services

ocs-appstore-admin-api-gw @ 0	active (running)	disabled
ocs-appstore-adminconsole-eula	active (exited)	enabled
ocs-appstore-adminconsole-ui @ 0	active (running)	disabled
ocs-appstore-applications-api @ 0	active (running)	disabled
ocs-appstore-client-api-gw @ 0	active (running)	disabled
ocs-appstore-client-api-gw @ 1	active (running)	disabled

Рисунок 7

Описание назначения столбцов таблицы, а также информация о возможных значениях приведены в таблице (Таблица 16).

Таблица 16

Название столбца	Описание	Возможные значения (примеры значений)
Name	Название сервиса ППО	Возможные значения определяются перечнем сервисов ППО и имеют следующий формат `<имя группы сервисов>@<номер экземпляра сервиса в группе>.service`, например: `ocs-appstore-admin-api-gw@0.service`.
Unit Status	Информация о статусе сервиса	`active` - сервис запущен и выполняется; `activating` - сервис запускается; `deactivating` - сервис выключается; `inactive` – сервис выключен; `failed` - при запуске сервиса произошла ошибка
Unit-file status	Информация о присутствии конфигурационного файла запуска сервиса в автозапуске	`enabled` - присутствует в автозапуске `disabled` - отсутствует в автозапуске

2.11.2.2.3 Mandatory services

Подраздел содержит информацию о статусе сервисов компонентов среды функционирования (Рисунок 8).

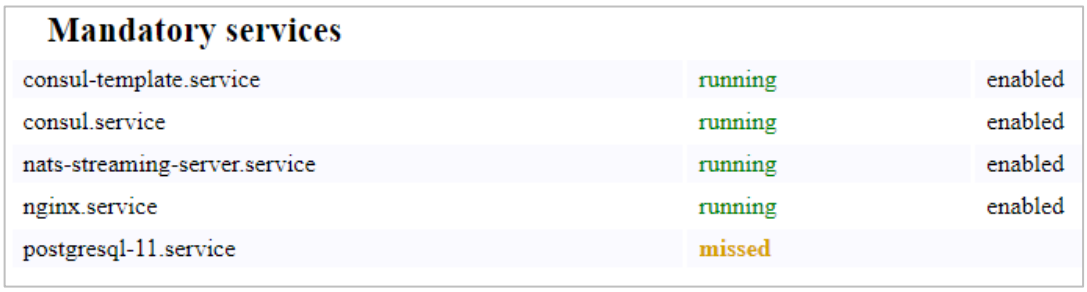


Рисунок 8

Описание назначения столбцов таблицы, а также информация о возможных значениях приведены в таблице (Таблица 17).

Таблица 17

Название столбца	Описание	Возможные значения (примеры значений)

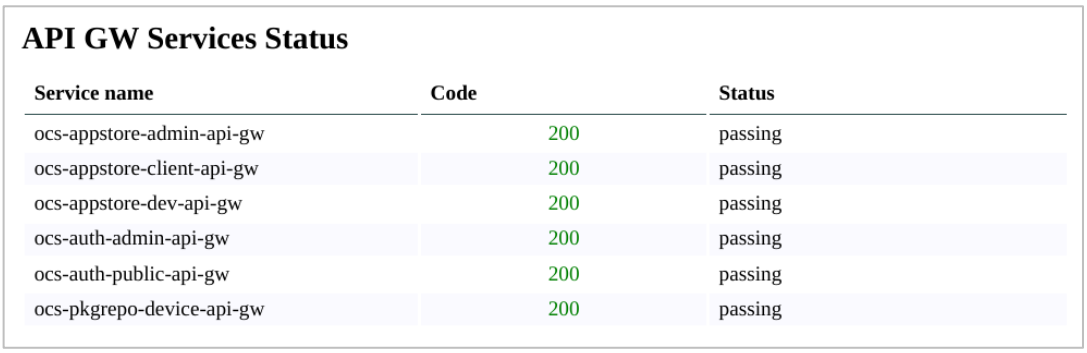
Name	Название сервиса компонента среды функционирования	Возможные значения имеют следующий формат `<имя сервиса>.service` и определяются Разработчиком. Перечень возможных значений:
		`consul-template.service`
		`consul.service`
		`nats-streaming-serever.service`
		`nginx.service`
		`postgresql-11.service`
		`postgresql.service`

Unit Status	Информация о статусе сервиса	`active` - сервис запущен и выполняется; `activating` - сервис запускается; `deactivating` - сервис выключается; `inactive` - сервис выключен; `failed` – при запуске сервиса произошла ошибка; `enabled` - сервис присутствует в автозапуске; `disabled` - сервис отсутствует в автозапуске; `missed` - компонент отсутствует
Unit-file status	Информация о присутствии конфигурационного файла запуска сервиса в автозапуске	`enabled` - присутствует в автозапуске `disabled` - отсутствует в автозапуске

2.11.2.3. Раздел «API GW Service Status»

Раздел содержит информацию о статусе регистрации сервисов в системе

обнаружения сервисов (Consul). На рисунке (Рисунок 9) приведен пример статуса регистрации сервисов в системе обнаружения сервисов.



API GW Services Status

Service name	Code	Status
ocs-appstore-admin-api-gw	200	passing
ocs-appstore-client-api-gw	200	passing
ocs-appstore-dev-api-gw	200	passing
ocs-auth-admin-api-gw	200	passing
ocs-auth-public-api-gw	200	passing
ocs-pkgrepo-device-api-gw	200	passing

Рисунок 9

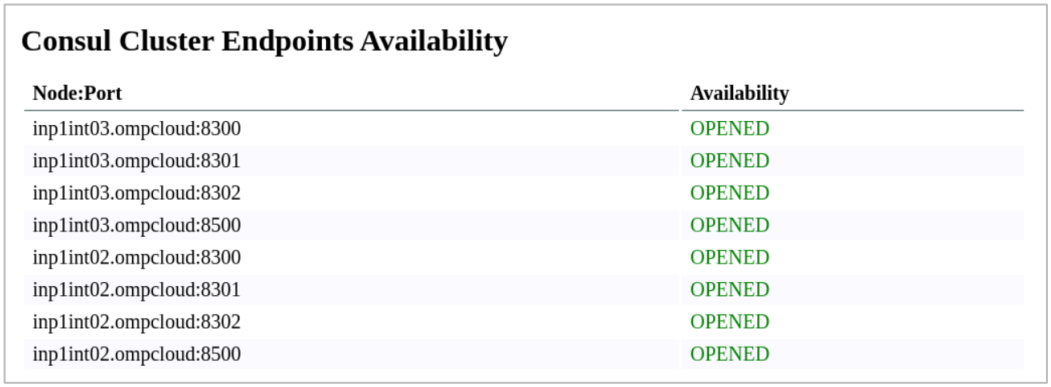
Описание назначения столбцов таблицы, а также информация о возможных значениях приведены в таблице (Таблица 18).

Таблица 18

Название столбца	Описание	Возможные значения (примеры значений)
Service name	Название сервиса ППО	Возможные значения определяются перечнем сервисов ППО
Code	Код http-ответа	Возможные значения определяются протоколом HTTP
Status	Информация о статусе регистрации сервиса в системе обнаружения сервисов (Consul)	Возможные значения определяются Consul. Статус «passing» означает, что проверка пройдена успешно

2.11.2.4. Раздел «Consul Cluster Endpoints Availability»

Раздел содержит информацию о проверке доступности интерфейсных функций системы обнаружения сервисов (Consul). На рисунке (Рисунок 10) приведен пример отображения информации о доступности интерфейсных функций системы обнаружения сервисов.



Consul Cluster Endpoints Availability

Node:Port	Availability
inp1int03.ompcloud:8300	OPENED
inp1int03.ompcloud:8301	OPENED
inp1int03.ompcloud:8302	OPENED
inp1int03.ompcloud:8500	OPENED
inp1int02.ompcloud:8300	OPENED
inp1int02.ompcloud:8301	OPENED
inp1int02.ompcloud:8302	OPENED
inp1int02.ompcloud:8500	OPENED

Рисунок 10

Перечень интерфейсных функций Consul приведен в документации на Consul (`https://www.consul.io/docs/install/ports`). Информация о доступности интерфейсных функций Consul предоставляется только в случае кластерной (многонодовой) конфигурации.

Описание назначения столбцов таблицы, а также информация о возможных значениях приведены в таблице (Таблица 19).

Таблица 19

Название столбца	Описание	Возможные значения (примеры значений)
Node:Port	Адрес функции	Адрес функции представлен в следующем формате: `<имя хоста>:<порт>`. Проверка выполняется только для функций, доступных на следующих портах: 8300, 8301, 8302, 8500. Например: `acenter.example:8300`
Availability	Статус доступности функции	В случае доступности функции принимает значение «OPENED». В ином случае выводится код ошибки и сообщение, определяемое Consul

2.11.2.5. Раздел «Consul Service Health Check»

Раздел содержит информацию о статусе регистрации сервисов ППО в системе обнаружения сервисов Consul (Рисунок 11).

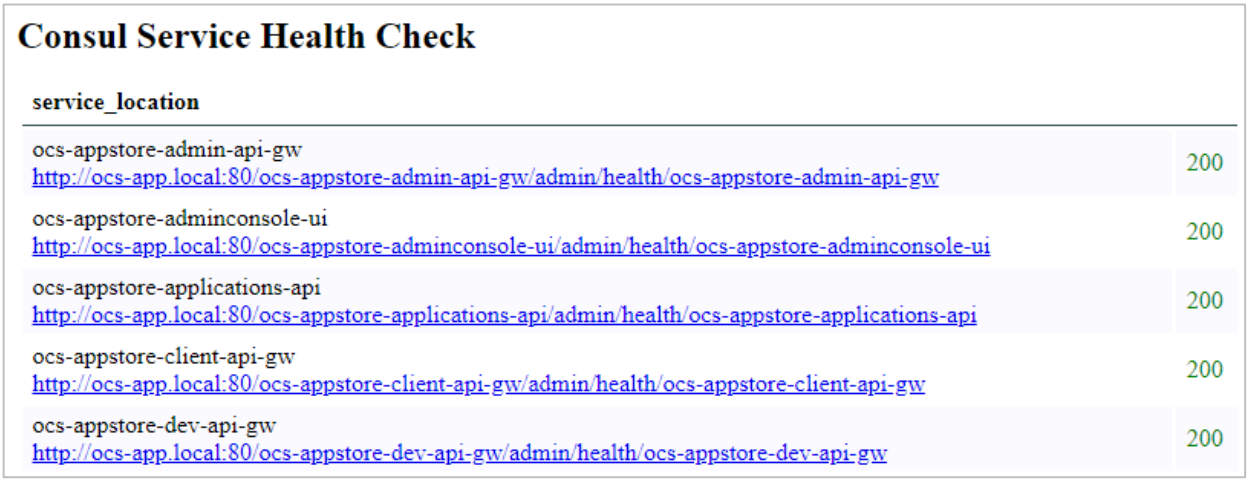


Рисунок 11

Описание назначения столбцов таблицы, а также информация о возможных значениях приведены в таблице (Таблица 20).

Таблица 20

Название столбца	Описание	Возможные значения (примеры значений)
Первый столбец	Название сервиса ППО и url-адрес функции (endpoint) сервиса «healthcheck»	Возможные значения определяются перечнем сервисов ППО
Второй столбец	Код http-ответа	Возможные значения определяются протоколом HTTP

Перечисленные заголовки `"service_location"`, `"expose_location"`, `"service_vhost"`, `"expose_port"`, `"static"` – это режимы работы `consul-template`.

2.11.2.6. Раздел «Cluster Nodes Reachability»

Раздел содержит информацию о результатах проверки доступности серверов (нод) кластера (Рисунок 12).

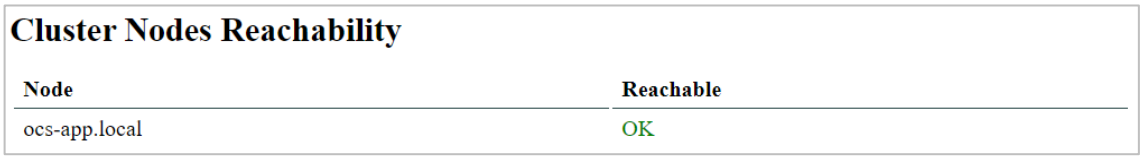


Рисунок 12

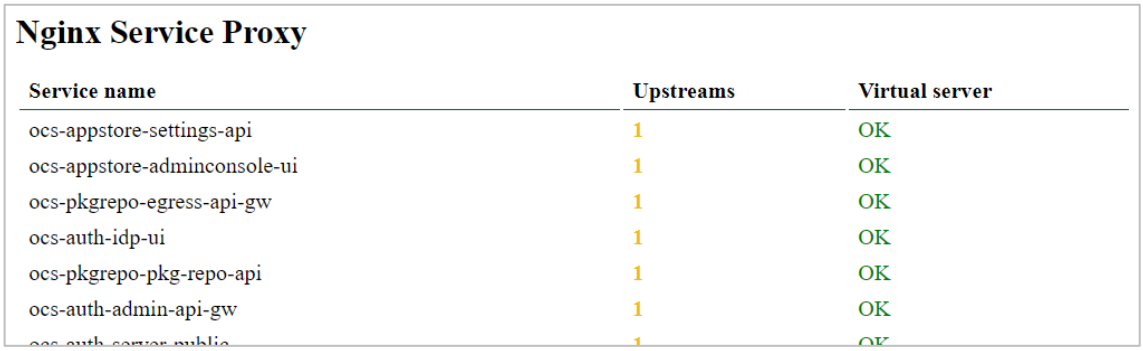
Описание назначения столбцов таблицы, а также информация о возможных значениях приведены в таблице (Таблица 21).

Таблица 21

Название столбца	Описание	Возможные значения (примеры значений)
Node	Адрес сервера (хоста)	Определяется доменными именами хостов
Reachable	Информация о доступности сервера	Может принимать значения: «OK» (в случае доступности) или содержать сообщение об ошибке, которое вернет утилита ping

2.11.2.7. Раздел «Nginx Service Proxy»

Раздел содержит информацию о проверке конфигурации балансировщика микросервисов Nginx Web Server для каждого сервиса ППО (Рисунок 13).



Nginx Service Proxy

Service name	Upstreams	Virtual server
ocs-appstore-settings-api	1	OK
ocs-appstore-adminconsole-ui	1	OK
ocs-pkgrepo-egress-api-gw	1	OK
ocs-auth-idp-ui	1	OK
ocs-pkgrepo-pkg-repo-api	1	OK
ocs-auth-admin-api-gw	1	OK
ocs-auth-server-public	1	OK

Рисунок 13

Описание назначения столбцов таблицы, а также информация о возможных значениях приведены в таблице (Таблица 22).

Таблица 22

Название столбца	Описание	Возможные значения (примеры значений)
Service name	Название сервиса ППО	Возможные значения определяются перечнем сервисов ППО
Upstreams	Количество экземпляров сервиса, заданных в конфигурационном файле Nginx	Целочисленные значения от 1 до n

Название столбца	Описание	Возможные значения (примеры значений)
Virtual server	Информация о наличие секции «server» для указанного сервиса ППО в конфигурационном файле Nginx. В данной секции заданы настройки «виртуального» сервиса ППО, который осуществляет перенаправление (проксирование) http-запросов на «реальные» экземпляры сервиса	«ОК» - секция `server` присутствует «`No server block found!`» - секция отсутствует

2.11.2.8. Раздел «Filestorage Configuration»

Раздел содержит информацию о конфигурации файловых хранилищ ПМ и ПООС (Рисунок 14).

Рисунок 14

Настройка «Filestorage location» содержит путь к каталогу и его статус.

В настройке «Configuration file» указан конфигурационный файл, в котором задан путь к файловому хранилищу.

3. УПРАВЛЕНИЕ КОМПОНЕНТАМИ СРЕДЫ ФУНКЦИОНИРОВАНИЯ, СЕРВИСАМИ, НАСТРОЙКАМИ СЕРВИСОВ И ПОДСИСТЕМ

3.1. Управление компонентами среды функционирования ППО

Управление сервисами ППО заключается в их установке, обновлении и удалении и осуществляется с помощью скрипта `deploy-infra.sh` из каталога `install-<версия ППО>`, созданного на этапе развертывания управляющей ЭВМ (подраздел 2.3).

Формат команды управления сервисами имеет следующий вид:

```
ANSIBLE_USER=<имя пользователя> ./deploy-infra.sh <параметры>
```

Описание параметров команды управления:

1) `<имя пользователя>`

В параметре указывается имя привилегированного sudo-пользователя, под которым настроен SSH-доступ к серверам приложений и БД.

2) `-A, --action`

Данный параметр задает действие, которое необходимо выполнить, и может принимать следующие значения:

– параметр отсутствует - будет выполнена установка или обновление компонента (компонентов);

– `flush_all` - будет выполнено удаление компонента (компонентов).

3) `-c, --components`

Данный параметр задает компонент среды функционирования, для которого будет выполнена команда управления, и может принимать следующие значения:

– `dnsmasq`;

– `nginx`;

– `consul`;

– `consul-template`;

– `nats-streaming-server`;

- redis;
- ocs-user;
- db.

В данном параметре может задаваться список подсистем, например:

```
--components dnsmasq, nginx
```

По умолчанию (если параметр на задан) команда управления будет применена ко всем компонентам.

4) `-d, --database`

Перечень допустимых значений параметра приведен в таблице (Таблица 9).

По умолчанию (если параметр не задан) будет использоваться значение, заданное в параметре `pg_version` конфигурационного файла `config/vars/_vars.yml`.

При отсутствии СУБД в перечне компонентов (параметр --components) значение данного параметра будет игнорироваться.

5) `--skip-database`

При наличии данного параметра СУБД не устанавливается.

6) `-l, --limit`

Данный параметр задает перечень хостов, для которых будет выполнена команда управления, например:

```
--limit  example01.omp,example02.omp
```

По умолчанию (если параметр на задан) команда управления будет применена ко всем хостам согласно инвентарному файлу `inventories/hosts.yml`.

7) `-e, --extra-vars`

В данном параметре передаются внешние переменные для скриптов развертывания. В ППО используются следующие внешние переменные:

- `pg_slave_recreate=true` - служит для инициализации реплики БД.

8) `--force-infra-install`

Флаг служит для управления принудительной повторной установки компонентов среды функционирования, в случаях, когда версия компонентов среды

функционирования не изменилась и может принимать следующие значения:

– `false` - повторная установка компонентов той же версии выполнена не будет;

– `true` - будет выполнена повторная установка компонентов не зависимо от того изменилась версия или нет.

По умолчанию (если флаг не задан) флаг имеет значение `true`.

9) `--help`

Вывод справочной информации.

Примеры команд управления:

1) Установка или обновление всех компонентов:

```
ANSIBLE_USER="omp" ./deploy-infra.sh --database 12
```

2) Установка или обновление Nginx на хосте `ocs-app.local`:

```
ANSIBLE_USER="omp" ./deploy-infra.sh --components nginx --limit ocs-app.local
```

3) Удаление Nginx:

```
ANSIBLE_USER="omp" ./deploy-infra.sh --components nginx --action flush_all
```

4) Получение справочной информации:

```
./deploy-infra.sh --help
```

3.2. Управление сервисами ППО

Управление сервисами ППО заключается в их установке, запуске, остановке, перезапуске, изменении настроек и осуществляется с помощью скрипта `deploy-ac.sh из каталога install-<версия ППО>`, созданного на этапе развертывания управляющей ЭВМ (п. 2.3.8).

Формат команды управления сервисами имеет следующий вид:

```
ANSIBLE_USER=<имя пользователя> ./deploy-ac.sh <параметры>
```

Описание параметров команды управления:

1) `<имя пользователя>`

В параметре указывается имя привилегированного sudo-пользователя, под которым настроен SSH-доступ к серверам приложений и БД.

2) `-s, --subsystems`

Данный параметр задает подсистему, для которой будет выполнена команда управления, и может принимать следующие значения:

- `auth` (для ПБ);
- `appstore` (для ПМ);
- `emm` (для ПУ);
- `mt` (для ПУТ);
- `pkgrepo` (для ПООС).

В данном параметре может задаваться список подсистем, например:

```
--subsystems auth,appstore,pkgrepo,emm,mt
```

По умолчанию параметр (если иное значение не задано) имеет значение:

```
--subsystems auth,appstore,pkgrepo,emm,mt
```

3) `-a, --apps`

Данный параметр задает перечень сервисов, для которых будет выполнена команда управления. Например:

```
--apps ocs-auth-adminconsole-ui,ocs-appstore-adminconsole-ui
```

Если необходимо выполнить команду сразу для всех сервисов подсистемы, потребуется перечислить через запятую все сервисы подсистемы либо задать значение параметра:

```
--apps all
```

По умолчанию параметр (если иное значение не задано) имеет значение «all».

В случае если заданные в параметре «`--apps`» сервисы не соответствуют заданным в параметре «`--subsystems`» подсистемам, управляющая команда к таким сервисам применена не будет. При этом управление шлюзами доступа (сервисами шлюзов доступа) осуществляется в рамках той подсистемы, для которой они предназначены. Состав подсистем приведен в таблице (Таблица 23).

Таблица 23

Значение параметра «--subsystems»	Сервисы (значение параметра «--apps»)
ПБ	
auth	ocs-auth-admin-api-gw
	ocs-auth-public-api-gw
	ocs-auth-admin-cross-tenant-api-gw
	ocs-auth-server-public-proxy
	ocs-auth-idp-api
	ocs-auth-accounts-devices-api
	ocs-auth-accounts-users-api
	ocs-auth-server-admin
	ocs-auth-server-public
	ocs-auth-audit-api
	ocs-auth-subsystems-api
	ocs-auth-config-api
	ocs-auth-adminconsole-ui
	ocs-auth-idp-ui
ПМ	
appstore	ocs-appstore-applications-api
	ocs-appstore-settings-api
	ocs-appstore-adminconsole-ui
	ocs-appstore-devconsole-ui
	ocs-appstore-admin-api-gw
	ocs-appstore-client-api-gw
	ocs-appstore-dev-api-gw
	ocs-appstore-egress-api-gw
ПУ	
emm	ocs-emm-applications-api
	ocs-emm-dispatcher-api
	ocs-emm-devices-api
	ocs-emm-state-manager-api
	ocs-emm-enrollments-api
	ocs-emm-policies-api
	ocs-emm-reports-api
	ocs-emm-users-api
	ocs-emm-journal-api
	ocs-emm-jobs-api
	ocs-emm-admin-api-gw
	ocs-emm-device-api-gw
	ocs-emm-egress-api-gw
ПУТ	
mt	ocs-mt-tenants-api
	ocs-mt-organizations-api
	ocs-mt-admin-api-gw
	ocs-mt-egress-api-gw
ПООС	

Значение параметра «--subsystems»	Сервисы (значение параметра «--apps»)
pkgrepo	ocs-pkgrepo-pkg-repo-api
	ocs-pkgrepo-device-api-gw
	ocs-pkgrepo-admin-api-gw
	ocs-pkgrepo-egress-api-gw

4) `-A, --action`

Данный параметр задает действие, которое необходимо выполнить. Перечень допустимых действий и соответствующие им значения параметра приведены в таблице (Таблица 24).

Таблица 24

Значение параметра «--action»	Действие
deploy	Установка
start	Запуск
stop	Остановка
restart	Перезапуск
config	Изменение настроек (переустановка конфигурационного файла)
flush_all	Удаление

По умолчанию параметр (если не задано иное значение) имеет значение `deploy`.

ВНИМАНИЕ! Установка подсистем ППО должна осуществляться строго в следующей последовательности: ПБ, ПМ, ПООС, ПУ, ПУТ.

5) `-C, --clients`

Данный параметр задает OIDC клиентов, для которых будет выполнена команда управления. Например:

```
--clients auth-admin-console, aps-admin-console
```

При необходимости выполнить команду сразу для всех OIDC клиентов потребуется перечислить через запятую все OIDC клиенты либо задать значение параметра:

```
--clients all
```

По умолчанию параметр (если не задано иное значение) имеет значение `all`.

6) `-d, --database`

Данный параметр задает СУБД, которая установлена на сервере БД. Перечень допустимых значений параметра приведен в таблице (Таблица 9).

Например:

```
--database 12
```

По умолчанию (если параметр не задан) будет использоваться значение, заданное в параметре `pg_version` конфигурационного файла `config/vars/_vars.yml`.

7) `--help`

Вывод справочной информации.

Примеры команд управления:

1) Остановка всех сервисов ПМ:

```
ANSIBLE_USER=omp ./deploy-ac.sh --action stop
```

2) Запуск сервисов `ocs-appstore-applications-api` и `ocs-appstore-adminconsole` ПМ:

```
ANSIBLE_USER=omp ./deploy-ac.sh --apps ocs-appstore-applications-
api,ocs-appstore-adminconsole --action start
```

3) Получение справочной информации:

```
./deploy-ac.sh --help
```

3.3. Управление настройками сервисов и подсистем ППО

Управление настройками сервисов и подсистем ППО может осуществляться двумя способами.

3.3.1. Способ 1 (рекомендуемый)

Для изменения настроек сервисов и подсистем ППО данным способом необходимо выполнить следующие действия:

3.3.1.1. Задать требуемые значения параметров в конфигурационных файлах сценариев установки ППО и подсистем ППО.

3.3.1.2. Переустановить конфигурационные файлы с помощью следующей команды:

– ПБ:

```
ANSIBLE_USER=<имя пользователя> ./deploy-ac.sh --subsystems auth --
action config
```

– ПМ:

```
ANSIBLE_USER=<имя пользователя> ./deploy-ac.sh --subsystems appstore -
-action config
```

– ПООС:

```
ANSIBLE_USER=<имя пользователя> ./deploy-ac.sh --subsystems pkgrepo --
action config
```

– ПУ:

```
ANSIBLE_USER=<имя пользователя> ./deploy-ac.sh --subsystems emm --
action config
```

– ПУТ:

```
ANSIBLE_USER=<имя пользователя> ./deploy-ac.sh --subsystems mt --
action config
```

3.3.2. Способ 2

Для изменения настроек сервисов и подсистем ППО данным способом необходимо выполнить следующие действия:

3.3.2.1. Задать требуемые значения параметров в конфигурационных файлах сервисов и подсистем ППО. Описание параметров конфигурационных файлов сценариев установки подсистем ППО приведено в разделе 8 настоящего документа.

3.3.2.2. Перезапустить требуемые сервисы с помощью следующей команды:

– ПБ:

```
ANSIBLE_USER=<имя пользователя> ./deploy-ac.sh --subsystems auth --
apps <перечень сервисов ПБ> --action restart
```

– ПМ:

```
ANSIBLE_USER=<имя пользователя> ./deploy-ac.sh --subsystems appstore -
-apps <перечень сервисов ПМ> --action restart
```

– ПООС:

```
ANSIBLE_USER=<имя пользователя> ./deploy-ac.sh --subsystems pkgrepo --
apps <перечень сервисов ПООС> --action restart
```

– ПУ:

```
ANSIBLE_USER=<имя пользователя> ./deploy-ac.sh --subsystems emm --apps
<перечень сервисов ПУ> --action restart
```

– ПУТ:

```
ANSIBLE_USER=<имя пользователя> ./deploy-ac.sh --subsystems mt --apps
<перечень сервисов ПУТ> --action restart
```

4. РЕЗЕРВНОЕ КОПИРОВАНИЕ

ВНИМАНИЕ! Приведенные ниже имена файлов и каталогов характерны для типового варианта установки ППО и среды функционирования ППО.

4.1. Резервное копирование после установки (обновления) ППО

После успешной установки (обновления) ППО необходимо создать резервную копию каталога `install-<версия ППО>/install-ac/` (`install-<версия ППО>/install-ac-mt/`).

4.2. Периодическое резервное копирование и резервное копирование перед установкой обновлений

Периодичность резервного копирования определяется регламентами эксплуатирующей организации.

Периодическое резервное копирование и резервное копирование перед установкой обновлений выполняется в приведенной ниже последовательности.

4.2.1. Резервное копирование данных

4.2.1.1. Создать резервные копии баз данных ПБ (auth), ПМ (appstore), ПУ (emm), ПУТ (mt) и ПООС (pkgrepo).

Резервная копия БД выполняется в соответствии с ЭД на используемую СУБД либо в соответствии с регламентами эксплуатирующей организации.

4.2.1.2. Создать резервную копию каталога с файлами МП «Аврора Маркет».

Для этого необходимо создать резервную копию каталога файлами МП «Аврора Маркет» на Сервере приложений ПМ либо в едином файловом хранилище в зависимости от того, какой вариант хранения файлов используется. Информация о нахождении каталога с файлами МП «Аврора Маркет» приведена в п. 2.8.1.

4.2.1.3. Создать резервную копию каталога с файлами пакетов ОС ПООС.

Для этого необходимо создать резервную копию каталога с файлами пакетов ОС ПООС на Сервере приложений ПООС либо в едином файловом хранилище в зависимости от того, какой вариант хранения файлов используется. Информация о нахождении каталога с файлами пакетов ОС ПООС приведена в п. 2.8.3.

4.2.2. Резервное копирование ППО

4.2.2.1. Создать резервную копию каталога с конфигурационными файлами подсистем и сервисов ППО (каталог: `/var/ocs/`).

4.2.2.2. Создать резервную копию сервисов ППО.

Для этого необходимо создать резервную копию файлов `*.target` и `*.service` по маске `ocs-*`, находящихся в каталоге `/etc/systemd/system/`, а также бинарных файлов сервисов по маске `ocs-*`, находящихся в каталоге `/usr/bin/`.

4.2.3. Резервное копирование компонентов среды функционирования

4.2.3.1. Создать резервную копию Nginx Web Server (каталог: `/etc/nginx/`).

4.2.3.2. Создать резервную копию Consul (каталог: `/opt/consul/`).

4.2.3.3. Создать резервную копию Consul Template (каталог: `/etc/consul-template/`).

4.2.3.4. Создать резервную копию Nats Streaming Server (каталог: `/data/nats/`).

4.2.3.5. Создать резервную копию конфигурационных файлов Redis (файлы: `/etc/redis.conf/etc/redis-sentinel.conf`).

4.2.3.6. Создать резервную копию конфигурационных файлов СУБД

– при использовании СУБД PostgreSQL выполнить резервную копию файлов: `/var/lib/pgsql/<версия СУБД PostgreSQL>/data/postgresql.conf`, `/var/lib/pgsql/<версия СУБД PostgreSQL>/data/pg_hba.conf` для ОС CentOS и `/var/lib/pgsql/data/postgresql.conf`, `/var/lib/pgsql/data/pg_hba.conf` для ОС Альт 8 СП;

– при использовании СУБД PostgresPro выполнить резервную копию файлов: /opt/pgpro/std-<версия СУБД PostgresPro>/data/postgresql.conf, /opt/pgpro/std-<версия СУБД PostgresPro>/data/pg_hba.conf).

4.2.3.7. Создать резервную копию сетевых настроек.

Для этого необходимо создать резервные копии следующих файлов:

– /etc/hosts;

– /etc/hostname;

– конфигурационные файлы DNS-сервера.

5. ОБНОВЛЕНИЕ ППО И ОС АВРОРА

5.1. Порядок обновления

В рамках поддержки жизненного цикла ППО предприятие-изготовитель вносит в него изменения, направленные на улучшение эксплуатационных характеристик и устранение недостатков.

Доведение информации о выпуске обновлений ППО до каждого потребителя ППО осуществляется посредством отправки сообщений на электронные адреса потребителей.

Предусмотрены следующие способы предоставления обновлений потребителям:

– отправка новой версии ППО с сопроводительным письмом;

– публикация ISO образа загрузочного модуля новой версии ППО на официальном сайте предприятия-разработчика.

Потребитель также имеет возможность получить информацию о выходе обновлений через службу технической поддержки предприятия-разработчика по тел.: +7 (495) 269-09-80 либо по электронной почте: support@omp.ru.

Установка обновлений ППО должна осуществляться в соответствии с требованиями, приведенными в документе «Руководство администратора».

Если потребитель ППО не может реализовать компенсирующие меры по защите информации или ограничения по применению ППО рекомендуется прекратить его применение.

Выполнять обновление ППО до требуемой версии допустимо только с последней сертифицированной версии ППО, либо более поздней версии ППО. Для обновления ППО с более ранних версий (версий, предшествующих последней сертифицированной версии ППО) необходимо сначала выполнить последовательное обновление до предшествующих версий ППО, прошедших сертификационные

испытания. На текущий момент сертификационные испытания проводились для следующих версий ППО: 2.1.3, 2.2.1, 2.2.2, 2.5.1.

Например, при обновлении ППО с версии 2.2.1 до версии 3.1.1, необходимо выполнить следующую цепочку обновлений:

– обновление с версии 2.2.1 до версии 2.2.2;
– обновление с версии 2.2.2 до версии 2.5.1;
– обновление с версии 2.5.1 до версии 3.1.1.

После версии 2.5.1 были выпущены версии: 3.0.0, 3.0.1, 3.1.0. С данных версий до версии 3.1.1 можно обновиться «напрямую», минуя обновления до промежуточных версий.

5.2. Обновление сервера приложений ППО

ВНИМАНИЕ! Для установки обновления ППО количество свободного места на жестком диске сервера БД ПБ должно быть не меньше, чем размер самой БД ПБ. При недостаточном количестве свободного места на жестком диске его необходимо увеличить. Продолжительность процесса обновления ППО зависит от размера БД и может занять длительное время.

Для обновления сервера приложений ППО необходимо выполнить описанные ниже действия.

5.2.1. Создать резервную копию данных, ППО и компонентов среды функционирования в соответствии с разделом 3 настоящего документа.

5.2.2. Скопировать на управляющую ЭВМ архив с новой версией ППО и распаковать его в соответствии с п. 2.3.5 - 2.3.8 настоящего документа.

5.2.3. Обновить на управляющей ЭВМ пакеты в соответствии с п. 2.3.3 настоящего документа.

5.2.4. Настроить компоненты среды функционирования ППО и ППО в соответствии с подразделом 2.4 настоящего документа.

5.2.5. Выполнить upgrade скрипты

Для этого необходимо перейти в каталог со сценариями установки новой версии ППО (каталог: `install-<новая версия ППО>/install-ac/`) и выполнить все скрипты `play-upgrade_to_release_<версия ППО>.yml`, версии которых лежат в диапазоне между установленной версий ППО (**не включительно**) и новой версией ППО (**включительно**). Для некоторых версий ППО скрипты могут отсутствовать. Запуск скриптов осуществляется с помощью команды:

```
ansible-playbook -i inventories/hosts.yml -u <имя пользователя>
release_upgrade/play-upgrade_to_release_<версия ППО>.yml -vv --diff
```

Например, для обновления ППО с версии 2.2.2 до версии 2.5.1 необходимо выполнить следующие команды:

```
ansible-playbook -i inventories/hosts.yml -u <имя пользователя>
release_upgrade/play-upgrade_to_release_2.5.0.yml -vv --diff
ansible-playbook -i inventories/hosts.yml -u <имя пользователя>
release_upgrade/play-upgrade_to_release_2.5.1.yml -vv --diff
```

5.2.6. При обновлении СУБД PostgreSQL 11/12/13/14 до новой старшей версии[5] (`major version`) необходимо удалить СУБД (без удаления данных), выполнив команду:

```
ANSIBLE_USER=<имя пользователя> ./deploy-infra.sh --components db --
action flush_all
```

5.2.7. Установить компоненты среды функционирования в соответствии с п. 2.5.1 настоящего документа.

5.2.8. Установить ППО в соответствии с п. 2.5.2 настоящего документа.

5.2.9. Выполнить post_upgrade скрипты

Для этого необходимо перейти в каталог со сценариями установки новой версии ППО (каталог: `install-<новая версия ППО>/install-ac/`) и выполнить все скрипты `play-post_upgrade_to_release_<версия ППО>.yml`, версии которых лежат в диапазоне между старой версий ППО (**не включительно**) и новой версией

[5] Согласно спецификации SemVer 2.0.0.

ППО (**включительно**). Для некоторых версий ППО скрипты могут отсутствовать. Запуск скриптов осуществляется с помощью команды:

– скрипт `play-post_upgrade_to_release_3.0.0.yml`:

```
ansible-playbook -i inventories/hosts.yml -u <имя пользователя>
release_upgrade/play-post_upgrade_to_release_3.0.0.yml  -vv --diff --
limit <хост с ПМ>
```

В параметре `limit` необходимо указать имя одного из хостов с установленной ПМ (например, `ocs-app.local`).

– скрипт `play-post_upgrade_to_release_3.1.0.yml`:

```
ansible-playbook -i inventories/hosts.yml -u <имя пользователя>
release_upgrade/play-post_upgrade_to_release_3.1.0.yml -vv --diff
```

5.2.10. Перезапустить сервис `ocs-pkgrepo-pkg-repo-api` с помощью команды:

```
ANSIBLE_USER=<имя пользователя> ./deploy-ac.sh --subsystems pkgrepo --
apps ocs-pkgrepo-pkg-repo-api --action restart
```

5.2.11. Оповестить пользователей ППО о необходимости очистить кэш и cookies веб-браузера. Иначе при открытии интерфейса ППО будет ошибка HTTP ERROR 400.

5.3. Обновление мобильных приложений ППО

ВНИМАНИЕ! Перед обновлением МП ППО необходимо обновить сервер приложений ППО в соответствии с подразделом 5.1 настоящего документа, а также ОС Аврора до требуемой версии. Допускается обновление МП до более новой версии (downgrade МП не допускается).

Для обновления МП ППО необходимо последовательно выполнить действия, описанные ниже.

5.3.1. Запросить в технической поддержке предприятия-изготовителя пакет `omp-ocs-updater`

Пакет `omp-ocs-updater` представляет собой rpm-пакет, включающий пакеты МП из состава загрузочного модуля ППО, а также bash-скрипт для их правильной установки.

5.3.2. Загрузить в ПМ пакет `omp-ocs-updater` для тех версий ОС Аврора, под которыми происходит эксплуатация МП. Порядок действий по загрузке МП в ПМ описан в документе «Руководство пользователя. Часть 2. Подсистема «Маркет».

5.3.3. С помощью политики «Приложения/Установка приложений на устройство» (функционал ПУ) установить новую версию МП на тестовую группу МУ с целью проверки корректности обновления. Порядок работы с политиками и группами МУ описан в документе «Руководство пользователя. Часть 3. Подсистема Платформа управления».

ВНИМАНИЕ! В данной группе МУ должны находиться МУ под управлением одной и той же трехзначной версии ОС Аврора (например, версии 3.2.3).

5.3.4. Загрузить в ПМ пакет `omp-ocs-updater` для тех версий ОС Аврора, под которыми происходит эксплуатация МП. Порядок действий по загрузке МП в ПМ приведен в документе «Руководство пользователя. Часть 2. Подсистема «Маркет».

5.3.5. Убедиться, что в карточке МУ (в Консоли администратора ПУ) отображается требуемая версия МП.

5.3.6. После успешного обновления МП на тестовой группе МУ, выполнить обновление аналогичным образом для остальных МУ.

5.4. Обновление ОС Аврора с помощью ПУ

Обновление ОС Аврора выполняется в следующей последовательности:

5.4.1. Обновить сервер приложений ППО в соответствии с подразделом 5.1 настоящего документа.

5.4.2. Обновить ОС Аврора до требуемой версии на тестовой группе МУ с целью проверки корректности обновления с помощью политики «Приложения/Установить версию ОС». Порядок работы с политиками и группами МУ приведен в документе «Руководство пользователя. Часть 3. Подсистема Платформа управления».

5.4.3. Убедиться, что после окончания заданного в правиле временного интервала обновления в карточке каждого МУ из тестовой группы отображается требуемая версия ОС Аврора.

5.4.4. Обновить МП ППО на тестовой группе МУ в соответствии с подразделом 5.3 настоящего документа.

5.4.5. Выполнить обновление аналогичным образом для остальных МУ после успешного обновления ОС Аврора и МП на тестовой группе МУ.

6. УДАЛЕНИЕ ППО

Для удаления сервисов ППО необходимо выполнить следующие действия:

6.1. Перейти в каталог со сценариями установки ППО.

6.2. Удалить сервисы ППО с помощью следующих команд:

– ПБ:

```
ANSIBLE_USER=<имя пользователя> ./deploy-ac.sh --subsystems auth --
action flush_all
```

– ПМ:

```
ANSIBLE_USER=<имя пользователя> ./deploy-ac.sh --subsystems appstore -
-action flush_all
```

– ПУ:

```
ANSIBLE_USER=<имя пользователя> ./deploy-ac.sh --subsystems emm --
action flush_all
```

– ПООС:

```
ANSIBLE_USER=<имя пользователя> ./deploy-ac.sh --subsystems pkgrepo --
action flush_all
```

– ПУТ:

```
ANSIBLE_USER=<имя пользователя> ./deploy-ac.sh --subsystems mt --
action flush_all
```

6.3. Удалить компоненты среды функционирования ППО с помощью следующих команд:

– dnsmasq:

```
-ANSIBLE_USER=<имя пользователя> ./deploy-infra.sh --components
dnsmasq --action flush_all
```

– nginx:

```
ANSIBLE_USER=<имя пользователя> ./deploy-infra.sh --components nginx -
-action flush_all
```

– consul:

```
ANSIBLE_USER=<имя пользователя> ./deploy-infra.sh --components consul
--action flush_all
```

— consul template:

```
–ANSIBLE_USER=<имя пользователя> ./deploy-infra.sh --components
consul-template --action flush_all
```

— nats streaming server:

```
ANSIBLE_USER=<имя пользователя> ./deploy-infra.sh --components nats-
streaming-server --action flush_all
```

— redis:

```
ANSIBLE_USER=<имя пользователя> ./deploy-infra.sh --components redis -
-action flush_all
```

— postgresql (без удаления данных):

```
ANSIBLE_USER=<имя пользователя> ./deploy-infra.sh  --components db --
action flush_all
```

— postgresql (с удалением данных):

```
ANSIBLE_USER=<имя пользователя> ./deploy-infra.sh  --components db --
action flush_all --extra-vars "pg_uninstall_delete_data=true"
```

7. КОНФИГУРАЦИОННЫЕ ФАЙЛЫ СЦЕНАРИЕВ УСТАНОВКИ СРЕДЫ ФУНКЦИОНИРОВАНИЯ

7.1. Конфигурационные файлы сценариев установки среды функционирования

7.1.1. Инвентарный файл inventories/hosts.yml

В инвентарном файле `inventories/hosts.yml` задаются адреса серверов приложений и серверов БД, на которые будут установлены компоненты среды функционирования. Описание секций инвентарного файла `inventories/hosts.yml` приведено в таблице (Таблица 25).

Таблица 25

Секция конфигурационного файла	Описание
`all.children.ocs.children.app`	Сервера приложений ППО
`all.children.ocs.children.postgresql.children.postgresql_masters`	СУБД Postgres (главный сервер БД)
`all.children.ocs.children.postgresql.children.postgresql_slaves`	Реплика СУБД Postgres (резервный сервер БД)
`all.children.ocs.children.nginx`	Балансировщик микросервисов «Nginx Web Server»
`all.children.ocs.children.consul`	Система обнаружения сервисов «Consul»
`all.children.ocs.children.consul-template`	Средство управления конфигурациями микросервисов «Consul Template»
`all.children.ocs.children.nats_streaming_server`	Сервис гарантированной доставки сообщений «Nats Streaming Server»
`all.children.ocs.children.redis.children.redis_masters`	СУБД Redis для хранения сессий
`all.children.ocs.children.redis.children.sentinel`	Redis Sentinel обеспечивает высокую доступность СУБД Redis

Файл сценария установки для установки среды функционирования ППО на одном сервере с доменным именем `ocs-app.local` имеет следующий вид:

```
all:
  children:
    ocs:
      children:
        app:
          hosts:
            ocs-app.local:
        postgresql:
          children:
            postgresql_masters:
              hosts:
                ocs-app.local:
            postgresql_slaves:
              hosts:
        nginx:
          children:
            app:
          hosts:
        consul:
          children:
            consul_servers:
              children:
                app:
              hosts:
            consul_agents:
        consul_template:
          children:
            app:
        nats_streaming_server:
          children:
                app:
          hosts:
        redis:
          children:
            redis_masters:
              children:
                app:
              hosts:
            sentinel:
              children:
                app:
              hosts:
```

7.1.2. Настройки сценариев установки среды функционирования ППО в конфигурационных файлах config/vars/_vars.yml и config/subsystems/<название подсистемы>/vars/_vars.yml

В данных конфигурационных файлах задаются настройки следующих компонентов среды функционирования ППО: Nats Streaming Server, Consul, СУБД Redis и СУБД PostgreSQL. Конфигурационные файлы `_vars.yml` используются только в процессе установки.

Описание параметров конфигурационных файлов `_vars.yml` приведены в самих конфигурационных файлах в виде комментариев.

7.1.3. Настройки паролей и секретов компонентов среды функционирования в конфигурационных файлах config/secret.yml и config/subsystems/<название подсистемы>/secret.yml

В данных конфигурационных файлах задаются пароли и секреты следующих компонентов среды функционирования ППО: Nats Streaming Server, Consul, СУБД Redis и СУБД PostgreSQL.

8. КОНФИГУРАЦИОННЫЕ ФАЙЛЫ ППО (СЦЕНАРИЕВ УСТАНОВКИ ППО)

8.1. Общая информация о конфигурационных файлах ППО

ПРИМЕЧАНИЕ. Описание параметров конфигурационных файлов сценариев установки ППО и ППО приведены в самих конфигурационных файлах в виде комментариев.

Структура конфигурационных файлов ППО в общем виде приведена на рисунке (Рисунок 15). Жирным выделены файлы, подлежащие редактированию. Редактирование параметров в остальных файлах не предполагается.

```
/var/ocs/config/
├── environments
│   └── <название окружения>
│       ├── vars
│       │   └── _vars.yml
│       └── config.yml
├── subsystems
│   └── <название подсистемы>
│       ├── applications
│       │   ├── <название сервиса>
│       │   └── vars
│       │       ├── services.yml
│       │       └── _vars.yml
│       ├── config.yml
│       ├── internal.yml
│       └── scopes.yml
├── config.yml
├── internal.yml
└── secret.yml
```

Рисунок 15

ППО содержит следующие типы конфигурационных файлов:

– конфигурационный файл ППО (`/vars/ocs/config/config.yml`);

– конфигурационные файлы подсистем ППО (`/vars/ocs/config/subsystems/<название подсистемы>/config.yml`);

– конфигурационные файлы с паролями и токенами компонентов среды функционирования ППО (`/vars/ocs/config/secret.yml`);

– конфигурационные файлы сервисов (модулей) ППО (`/vars/ocs/config/subsystems/<название подсистемы>/applications/<название сервиса>/`).

В конфигурационном файле ППО содержатся настройки ППО.

В конфигурационных файлах подсистем содержатся настройки подсистем ППО. Также в конфигурационные файлы подсистем вынесены (могут быть вынесены) отдельные настройки сервисов ППО, которые может изменять администратор ППО. В данном случае в конфигурационном файле содержится секция с именем сервиса. Например, секция для сервиса `ocs-auth-accounts-users-api` выглядит следующим образом:

```
#---------------------------------------------------------------------
---------
# Parameters for user accounts
#---------------------------------------------------------------------
---------
ocs-auth-accounts-users-api:

  ##
  # The number of recently used passwords,
  # which system will store for forbidding use it for new password
creating.
  ##
  passwordHistoryDepth: 3

  ##
  # Maximum inactivity period 45 days.
  # If account not use system during this time, account will be
blocked.
  # Must be greater or equal to OIDC refresh token lifetime.
  ##
  maxAccountInactivityPeriod: "1080h"
```

ВНИМАНИЕ! Редактирование конфигурационных файлов сервисов не предполагается.

8.2. Общая информация о конфигурационных файлах сценариев установки ППО

Структура конфигурационных файлов сценариев установке ППО в общем виде приведена на рисунке (Рисунок 16). Жирным выделены файлы, подлежащие редактированию. Редактирование параметров в остальных файлах не предполагается.

```
install-ac-mt
├── config
│   ├── environments
│   │   └── <название окружения>
│   ├── subsystems
│   │   └── <название подсистемы>
│   │       ├── applications
│   │       │   └── <название сервиса>
│   │       ├── vars
│   │       │   ├── services.yml
│   │       │   └── _vars.yml
│   │       ├── config.yml(или config.yml.j2)
│   │       ├── internal.yml
│   │       └── scopes.yml
│   ├── vars
│   │   └── _vars.yml
│   ├── config.yml.j2
│   ├── internal.yml
│   ├── yamllint.release.yml
│   └── secret.yml
└── inventories
    ├── <название окружения>
    └── hosts.yml
```

Рисунок 16

Сценарии установки ППО содержат следующие типы конфигурационных файлов:

– конфигурационный файл `inventories/hosts.yml`;

– конфигурационный файл сценария установки ППО `config/vars/_vars.yml`;

– конфигурационные файлы сценариев установки подсистем ППО `config/subsystems/<название подсистемы>/vars/_vars.yml`;

– шаблон конфигурационного файла ППО (`config/config.yml.j2`);

– конфигурационные файлы подсистем ППО (config/subsystems/<название подсистемы>/config.yml);

– шаблоны конфигурационных файлов подсистем ППО (config/subsystems/<название подсистемы>/config.yml.j2);

– конфигурационные файлы сервисов (модулей) ППО (config/subsystems/<название подсистемы>/applications/<название сервиса>/);

– конфигурационный файл с паролями и токенами компонентов среды функционирования ППО config/secret.yml.

8.2.1. Конфигурационный файл inventories/hosts.yml

Конфигурационный файл inventories/hosts.yml содержит адреса серверов (имена хостов), на которые установлены (будут установлены) компоненты среды функционирования ППО и подсистемы ППО.

Описание параметров конфигурационного файла inventories/hosts.yml приведено в п. 7.1.1 настоящего документа.

8.2.2. Общий конфигурационный файл сценариев установки config/vars/_vars.yml

Конфигурационный файл config/vars/_vars.yml является общим для всех подсистем и модулей ППО и содержит полный перечень общих параметров, относящихся к подсистемам и модулям ППО.

Конфигурационные файлы _vars.yml используются только в процессе установки, при том как конфигурационные файлы config.yml (config.yml.j2) используются как в процессе установки, так и в процессе эксплуатации ППО.

8.2.3. Конфигурационные файлы сценариев установки для подсистем ППО (файлы: config/subsystems/<название подсистемы>/vars/_vars.yml)

Конфигурационные файлы `_vars.yml` подсистем содержат параметры, относящиеся к конкретной подсистеме. Также данные файлы могут быть дополнены параметрами из общего конфигурационного файла, значения которых необходимо переопределить для заданной подсистемы.

Конфигурационные файлы `_vars.yml` в основном содержат настройки взаимодействия подсистем с компонентами среды функционирования. Располагаются данные конфигурационные файлы в каталоге со сценариями установки по следующему пути:

```
config/subsystems/<название подсистемы>/vars/_vars.yml
```

Например, конфигурационный файл vars.yml для ПБ:

```
config/subsystems/auth/vars/_vars.yml
```

8.2.4. Шаблоны конфигурационных файлов ППО и подсистем ППО

На основе данных файлов в процессе установки ППО формируются конфигурационные файлы ППО и подсистем ППО. Значения параметров в шаблонах конфигурационных файлах подсистем ППО задаются администратором, а также сценариями установки на основе значений, заданных администратором в конфигурационных файлах `_vars.yml`.

Располагаются данные конфигурационные файлы по следующему пути:

```
config/config.yml.j2
config/subsystems/<название подсистемы>/config/services/config.yml.j2
```

Например, шаблон конфигурационного файла ПБ:

```
config/subsystems/auth/config/services/config.yml.j2
```

8.2.5. Конфигурационный файл с паролями и токенами компонентов среды функционирования ППО config/secret.yml

В данном конфигурационном файле задаются пароли и токены Nats Streaming Server, Consul, СУБД Redis, СУБД PostgreSQL, а также секретный ключ клиентов (сервисов) и ключ шифрования секретов, хранящихся в БД . При установке ППО данные конфигурационные файлы копируются на серверы приложений.

8.2.6. Конфигурационные файлы подсистем ППО

В данных конфигурационных файлах задаются значения параметров подсистем ППО. В отличии от шаблонов конфигурационных файлов подсистем ППО, значения параметров задаются только администратором.

Располагаются данные конфигурационные файлы по следующему пути:

```
config/subsystems/<название подсистемы>/config/services/config.yml
```

Например, шаблон конфигурационного файла ПМ:

```
config/subsystems/appstore/config/services/config.yml.j2
```

8.2.7. Конфигурационных файлов сервисов ППО

Конфигурационных файлов сервисов располагаются в каталоге со сценариями установки по следующему пути:

```
config/subsystems/<название подсистемы>/applications/<название сервиса>/
```

Например, конфигурационные файлы сервиса `ocs-auth-adminconsole-ui` ПБ:

```
config/subsystems/auth/applications/ocs-auth-adminconsole-ui/
```

Описание параметров конфигурационных файлов сервисов приведено в самих конфигурационных файлах в виде комментариев.

ВНИМАНИЕ! Редактировать конфигурационных файлов сервисов ППО не предполагается.

8.2.8. Конфигурационные файлы окружений

В конфигурационных файлах окружения переопределяются параметры конфигурационных файлов, описанных в п. 8.2.1 - 8.2.8 для заданного окружения. Располагаются данные конфигурационные в каталогах `config/environments/<название окружения>/` и `inventories/<название окружения>/`.

Для переопределения параметра необходимо выполнить следующие действия:

– создать в каталоге `inventories/<название окружения>/` конфигурационный файл `hosts.yml` по аналогии с файлом `inventories/hosts.yml` и задать в созданном файле требуемые значения параметров;

– создать в каталоге `config/environments/<название окружения>/` требуемый конфигурационный файл с учетом его расположения в каталоге `config`

Например, для переопределения параметров конфигурационного файла `config/vars/_vars.yml` для окружения release, должен быть создан следующий конфигурационный файл: `config/environments/release/config/vars/_vars.yml`.

– скопировать требуемый параметр (включая секцию, в которую входит параметр) из общего конфигурационного файла сценариев установки ППО или конфигурационного файла сценариев установки подсистем ППО;

– вставить скопированное значение в аналогичный конфигурационный файл для заданного окружения;

– задать требуемое значение параметра.

8.2.9. Порядок работы с конфигурационными файлами сценариев установки ППО

Параметры конфигурационных файлов сценариев установки применяются согласно приоритетам, заданным в таблице (Таблица 26).

Таблица 26

Типы конфигурационных файлов	Каталог (имя файла)	Порядок применения параметров (приоритет параметров)

Общие (для всех подсистем и модулей ППО) конфигурационные файлы (шаблоны конфигурационных файлов) сценариев установки ППО	`config/vars/_vars.yml`
	
	`config/config.yml.j2`
		1 (самый низкий приоритет)

Конфигурационные файлы сценариев установки подсистем ППО	`config/subsystems/<название подсистемы>/vars/` Например, `config/subsystems/auth/vars/`	2

Общие (для всех подсистем и модулей ППО) конфигурационные файлы сценариев установки ППО для заданного окружения	`config/environments/<название окружения>/vars/_vars.yml`
	
	`config/environments/<название окружения>/config.yml`
		3

Конфигурационные файлы сценариев установки подсистем ППО для заданного окружения	`config/environments/<окружение>/<название подсистемы>/vars/`	4 (самый высокий приоритет)

При установке ППО параметры конфигурационных файлов применяются согласно порядку, приведенному в таблице (Таблица 26). Т.е. сценарий установки обрабатывает сначала конфигурационные файлы в каталоге `config/vars/`, затем в каталоге `config/subsystems/<название подсистемы>/vars/` и т.д. Если, например, какой-либо параметр одновременно задан и в `config/vars/` и `config/subsystems/<название подсистемы>/vars/`, ППО будет установлено со значением параметра, заданным в `config/subsystems/<название подсистемы>/vars/`.

Ниже описаны правила обработки сценариями установки ППО параметров, массивов и списков, если они одновременно заданы в нескольких конфигурационных файлах.

Правило обработки параметров: значение параметра в конфигурационном файле с более высоким приоритетом переопределяет значение параметра в конфигурационном файле с более низким приоритетом.

Пример параметра:

```
redis_password: "example_redis_password"
```

Правило обработки массивов: массив в конфигурационном файле с более высоким приоритетом переопределяет массив в конфигурационном файле с более низким приоритетом.

Пример массива:

```
pg_hba_settings:
  - type: local  # Unix-socket access
    name: all
    database: all
    method: trust
  - type: host  # Localhost IPv4 access
    name: all
    database: all
    address: 127.0.0.1/32
    method: trust
  - type: host  # Localhost IPv6 access
    name: all
    database: all
    address: ::1/128
    method: trust
  - type: host  # Gitlab CI vbox-testing
    name: all
    database: all
    address: 172.17.0.0/16
    method: md5
```

Правило обработки списков: если список в конфигурационном файле с более низким приоритетом содержит новые элементы (которых не было в конфигурационном файле с более высоким приоритетом), они добавляются к исходному списку. Значение параметра в списке, содержащемся в конфигурационном файле с более высоким приоритетом, переопределяет значение параметра из списка содержащемся в конфигурационном файле с более низким приоритетом.

Пример списка:

```
postgresql:
  dbname: example_db_name # database name
  port: 5432              # port
  user: example_user      # user
  password: ocs           # password
  extensions: ["pg_partman_bgw", "pg_trgm", "pg_stat_statements",
"pgcrypto"] # necessary extensions
```

9. ГАРАНТИЙНЫЕ ОБЯЗАТЕЛЬСТВА

Предприятие-изготовитель гарантирует работоспособность ППО в соответствии с заявленными характеристиками, предусмотренными настоящим документом, при соблюдении потребителем требований ЭД.

Предприятие-изготовитель проводит мониторинг общедоступных источников информации, публикующих сведения об уязвимостях, на предмет появления в них сведений об уязвимостях в компонентах ППО и принимает меры, направленные на устранение выявленных уязвимостей или исключающие возможность использования нарушителями выявленных уязвимостей.

Предприятие-изготовитель обеспечивает устранение уязвимостей посредством предоставления потребителям описания необходимых организационно-технических процедур, направленных на устранение выявленной уязвимости. Также предприятие-изготовитель, в рамках проведения работ по устранению выявленных уязвимостей, разрабатывает обновления ПО.

Предприятие-изготовитель не предоставляет гарантий или условий (явных или подразумеваемых законодательством Российской Федерации) относительно гарантий товарной пригодности, интегрируемости, годности к использованию для выполнения конкретных задач потребителя, отсутствия ошибок, возможности функционирования при использовании совместно с любым программным или аппаратным обеспечением.

В случае выявления в ППО ошибок и дефектов, свидетельствующих о несоответствии ППО ЭД и не являющихся уязвимостями ППО, предприятие-изготовитель по факту получения рекламации потребителя обязуется устранить ошибки и/или дефекты при выпуске обновленных версий ППО и уведомить об этом потребителей ППО.

Рекламации потребителя принимаются при условии, что дефект в ППО не вызван допущенными со стороны потребителя нарушениями в эксплуатации, хранении и транспортировке ППО.

Адрес предприятия-изготовителя для направления рекламаций: 420500, Республика Татарстан, Верхнеуслонский район, г. Иннополис, ул. Университетская, д. 7, офис 59, ОГРН 1161690087020.

ПЕРЕЧЕНЬ ТЕРМИНОВ И СОКРАЩЕНИЙ

Используемые в настоящем документе термины и сокращения приведены в таблице (Таблица 27).

Таблица 27

Термин/ Сокращение	Расшифровка
АСУ	Автоматизированная система управления
БД	База данных
Воркер	Потоки, принадлежащие браузеру, которые можно использовать для выполнения JS-кода без блокировки цикла события
ГИС	Государственная информационная система
ИС	Информационные системы
ИСПДн	Информационная система персональных данных
КИИ	Критическая информационная инфраструктура
Механизм CORS	CORS (Cross-origin resource sharing) – технология современных браузеров, которая позволяет предоставить веб-странице доступ к ресурсам другого домена
МП	Мобильное приложение
МУ	Мобильное устройство
ОС	Операционная система
ОТК	Отдел технического контроля
ПБ	Подсистема безопасности
ПМ	Подсистема «Маркет»
ПООС	Подсистема обновления ОС
ПУТ	Подсистема управления тенантами
ППО	Прикладное программное обеспечение «Аврора Центр»
Предприятие-изготовитель	Общество с ограниченной ответственностью «Открытая мобильная платформа» (ООО «Открытая мобильная платформа»)
ПУ	Подсистема Платформа управления
СЗИ	Средство защиты информации

Термин/ Сокращение	Расшифровка
СЗИ НСД	Средство защиты информации от несанкционированного доступа
СПО	Специальное программное обеспечение
СУА	Сервис уведомлений Аврора
СУБД	Система управления базами данных
Токен (маркер)	Токен - аутентификационные данные, которые выдаются пользователю после успешной авторизации и являются ключом для доступа к службам
ЭВМ	Электронно-вычислительная машина
ЭД	Эксплуатационная документация
API	Application Programming Interface – описание способов (набор классов, процедур, функций, структур или констант), которыми одна компьютерная программа может взаимодействовать с другой программой
Cookie	Небольшой фрагмент данных, отправленный веб-сервером и хранимый на ЭВМ пользователя. Веб-клиент (обычно веб-браузер) всякий раз при попытке открыть страницу соответствующего сайта пересылает этот фрагмент данных веб-серверу в составе http-запроса
CSS3	Cascading Style Sheets 3 – спецификация CSS. Представляет собой формальный язык, реализованный с помощью языка разметки
ECMAScript 5	Встраиваемый расширяемый не имеющий средств ввода-вывода язык программирования, используемый в качестве основы для построения других скриптовых языков
HEALTH-запрос	Запрос проверки доступности API
HTML5	HyperText Markup Language, version 5 – язык для структурирования и представления содержимого веб-страницы
HTTP	HyperText Transfer Protocol – протокол прикладного уровня передачи данных. Основой HTTP является технология «клиент-сервер», то есть предполагается существование потребителей (клиентов), которые инициируют соединение и посылают запрос, и поставщиков (серверов), ожидают соединения для получения запроса, производят необходимые действия и возвращают обратно сообщение с результатом

Термин/ Сокращение	Расшифровка
HTTPS	Hypertext Transfer Protocol Secure - расширение протокола HTTP для поддержки шифрования в целях повышения безопасности. Данные в протоколе HTTPS передаются поверх криптографических протоколов SSL или TLS
IMEI	International Mobile Equipment Identity – уникальный номер мобильного устройства, состоящий из 15 цифр
JSON	JavaScript Object Notation – текстовый формат обмена данными, основанный на JavaScript
MAC	Media Access Control – уникальный идентификатор, присваиваемый каждой единице оборудования компьютерных сетей
Nginx	Веб-сервер и почтовый прокси-сервер, работающий на Unix-подобных ОС
OIDC	OpenID Connect – уровень аутентификации OAuth 2.0, инфраструктуры авторизации. Контролируется OpenID Foundation
QR-код	Quick Response Code – код быстрого реагирования, предоставляющий информацию для быстрого ее распознавания с помощью камеры на мобильном устройстве
SSH	Secure SHell – сетевой протокол прикладного уровня, позволяющий производить удаленное управление ОС и туннелирование TCP-соединений (например, для передачи файлов)
TLS	Transport Layer Security – криптографический протокол, обеспечивающий защищенную передачу данных между узлами в сети Интернет
URL	Uniform Resource Locator – единообразный локатор (определитель местонахождения) ресурса

ЛИСТ РЕГИСТРАЦИИ ИЗМЕНЕНИЙ

№ версии документа	Содержание изменения	ФИО инициатора	Дата
1.0	Начальная версия	Фомин П.	22.08.2022 г.