МойОфис

Комплект Средств Разработки (SDK)

Руководство по установке

СЕРВЕР СОВМЕСТНОГО РЕДАКТИРОВАНИЯ

ООО «НОВЫЕ ОБЛАЧНЫЕ ТЕХНОЛОГИИ»

ПРОГРАММНОЕ ОБЕСПЕЧЕНИЕ «МОЙОФИС КОМПЛЕКТ СРЕДСТВ РАЗРАБОТКИ (SDK)» «СЕРВЕР СОВМЕСТНОГО РЕДАКТИРОВАНИЯ» 3.0

РУКОВОДСТВО ПО УСТАНОВКЕ

Версия 2

На 47 листах

Дата публикации: 10.07.24

Москва
2024

Все упомянутые в этом документе названия продуктов, логотипы, торговые марки и товарные знаки принадлежат их владельцам.

Товарные знаки «МойОфис» и «MyOffice» принадлежат ООО «НОВЫЕ ОБЛАЧНЫЕ ТЕХНОЛОГИИ».

Ни при каких обстоятельствах нельзя истолковывать любое содержимое настоящего документа как прямое или косвенное предоставление лицензии или права на использование товарных знаков, логотипов или знаков обслуживания, приведенных в нем.

Любое несанкционированное использование этих товарных знаков, логотипов или знаков обслуживания без письменного разрешения их правообладателя строго запрещено.

СОДЕРЖАНИЕ

ПЕРЕЧЕНЬ СОКРАЩЕНИЙ, ТЕРМИНОВ И ОПРЕДЕЛЕНИЙ

В настоящем документе применяют следующие сокращения с соответствующими расшифровками (см. Таблицу 1).

Таблица 1 — Сокращения и расшифровки

Сокращение, термин	Расшифровка и определение
AD	Microsoft Active Directory, Активный каталог
API	Application Programming Interface, интерфейс программирования приложений
Auth SSO	Single Sign-On, подсистема единого входа (аутентификации и авторизации)
CA	Certificate Authority, удостоверяющий центр для подтверждения подлинности ключей шифрования
CDN	Content Delivery Network, сеть доставки содержимого (в данном случае одна из ролей Auth SSO сервера)
CU	Converter Unit, сервис конвертирования разных форматов файлов
DCS	Document Collaboration Service, сервис редактирования и коллаборации документов на базе кода Core
DNS	Domain Name System, система доменных имен
DU	Document Unit, синоним DCS
EFK	Стек ПО для централизованного сбора и визуализации журналов событий, ElasticSearch + Fluentd + Kibana
ESIA	ЕСИА, Единая Система Идентификации и Аутентификации, информационная система в РФ, обеспечивающая санкционированный доступ для информационного взаимодействия (граждан-заявителей и должностных лиц органов исполнительной власти) к информации, содержащейся в государственных и иных информационных системах
ETCD	Распределенная система хранения конфигурации
FCM	Firebase Cloud Messaging, сервис уведомлений мобильных приложений Google, ранее назывался GCM
FQDN	Fully Qualified Domain Name, полностью определенное имя домена
GCM	Google Cloud Messaging, сервис нотификаций мобильных приложений Google, заменен сервисом FCM
HMS	Huawei Mobile Services, сервис нотификаций мобильных приложений Huawei
Inventory	Файл для настройки Ansible с перечислением ролей и их IP-адресов
IPVS	IP Virtual Server
JKS	Java Key Store, хранилище ключей и сертификатов, доступных в виртуальном сервере Java
LDAP	Lightweight Directory Access Protocol, облегченный протокол доступа к каталогам
LO	LibreOffice, фильтры которого используются для импортирования устаревших бинарных форматов документов
SMTP	Simple Mail Transfer Protocol, протокол передачи почтовых сообщений

Сокращение, термин	Расшифровка и определение
SSH	Secure Shell, «безопасная оболочка»
UI	User Interface, пользовательский интерфейс
URL	Uniform Resource Locator, единый указатель ресурса
UX	User Experience, «опыт пользователя»
ДУ	Директория установки
ООО	Общество с ограниченной ответственностью
ОС	Операционная система
ПК	Персональный компьютер
ПО	Программное обеспечение
УЦ	Удостоверяющий центр

1 ОБЩИЕ СВЕДЕНИЯ

1.1 Назначение

«Сервер совместного редактирования (ССР)» — интегрируемая серверная система и клиентские веб-приложения, предназначенные для просмотра и совместного редактирования текстовых и табличных документов в прикладных ИТ-системах. Сервер встраивается в хранилища сторонних производителей, поддерживающих возможность взаимодействия с внешними клиентами по протоколу WOPI.

Данное решение предоставляет возможность открыть документ из внешнего хранилища документов на просмотр или редактирование в iframe и при необходимости сохранять редактируемый документ обратно в хранилище. В качестве примера интеграции было использовано хранилище NextCloud (гарантируется работоспособность на версии 26) с включенным расширением OfficeOnline (см. Рисунок 1).

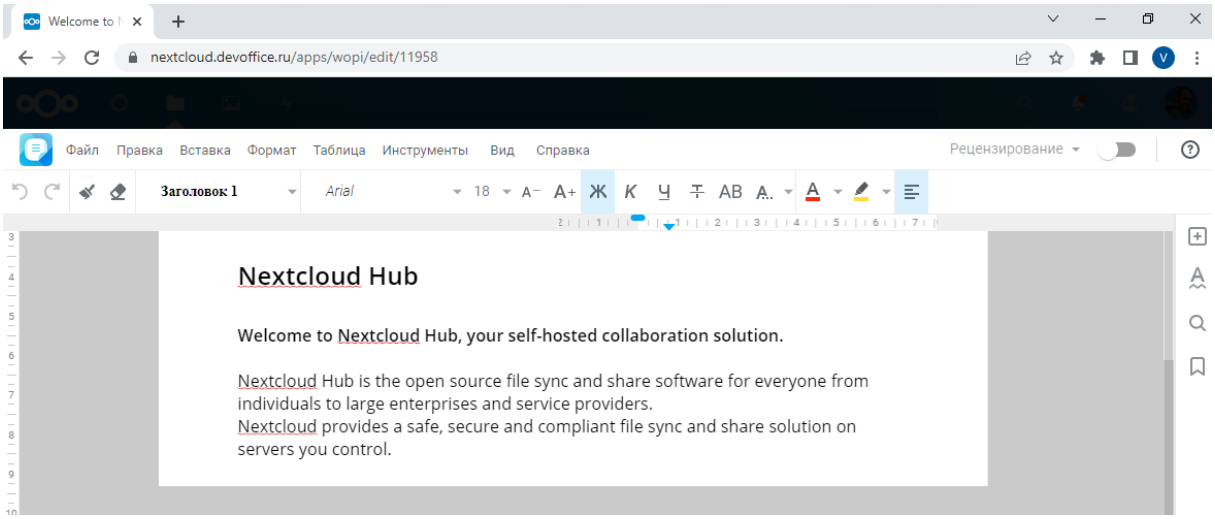

Рисунок 1 — Открытие документа в хранилище NextCloud

1.2 Описание архитектуры

Общая архитектурная схема для Сервера совместного редактирования (далее — ССР) приведена на рисунке 2.

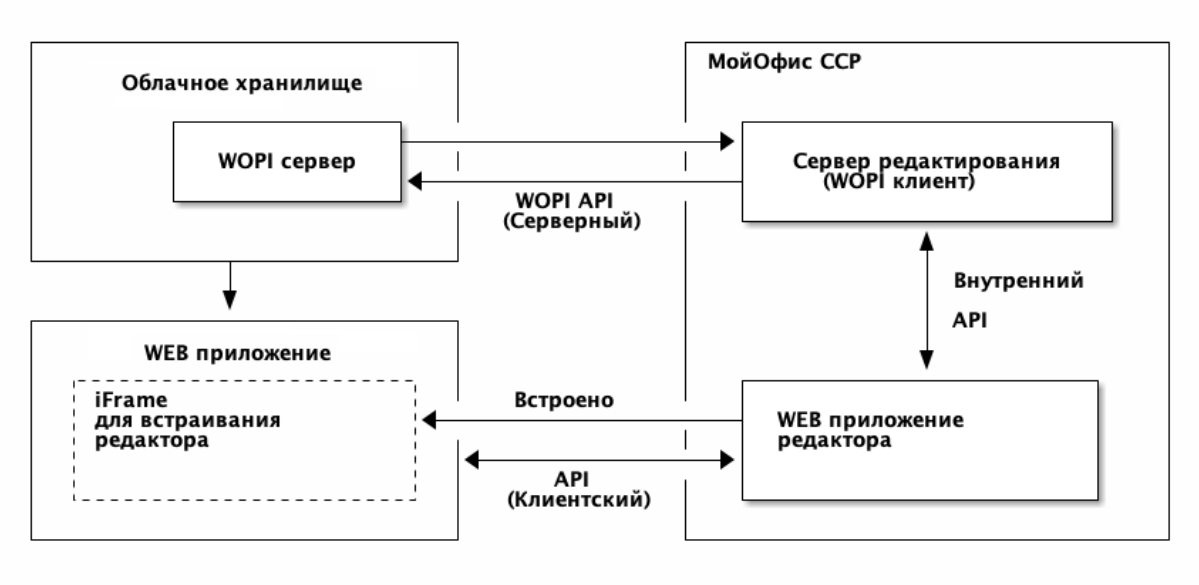

Рисунок 2 — Общая архитектурная схема ССР

1.3 Требования к персоналу

Для работы с ПО Администратору необходимо обладать релевантным опытом по следующим направлениям:

1. Основы сетевого администрирования:

– сетевая модель OSI и стек протоколов TCP/IP;

– IP-адресация и маски подсети;

– маршрутизация: статическая и динамическая;

– протокол обеспечения отказоустойчивости шлюза (VRRP).

2. Работа с подсистемой виртуализации на уровне эксперта:

– установка Docker;

– запуск/остановка/перезапуск контейнеров;

– работа с реестром контейнеров;

– работа с VMware vSphere ESXi 6.5 и выше;

– получение параметров контейнеров;

– сеть в Docker, взаимодействие приложений в контейнерах.

3. Работа с командной строкой ОС Linux:

– знания в объеме курсов Red Hat RH124, RH134, RH254;

– знания в объеме, достаточном для сдачи сертификационного экзамена Red Hat EX300.

4. Работа со службой доменных имен DNS:

– знание основных терминов (DNS, IP-адрес);

– понимание принципов работы DNS (корневые серверы, TLD-серверы, серверы имен доменов, разрешающий сервер имен);

– знание типов записи и запросов DNS.

5. Знание видов архитектуры, а также основных компонентов инфраструктуры открытых ключей (PKI):

– закрытый и открытый ключи;

– сертификат открытого ключа;

– регистрационный центр (RA);

– сертификационный центр (CA);

– хранилище сертификатов (CR).

6. Практический опыт администрирования на уровне эксперта:

– Etcd;

– Elasticsearch;

– Prometheus;

– RabbitMQ;

– Redis.

7. Работа с системой автоматизации развертывания Ansible.

1.4 Состав дистрибутива

Комплект поставки ПО предназначен для подготовки инфраструктуры сервера с ролью `operator` и дальнейшей установки ССР. Комплект включает в себя:

– исполняемый файл `co_ansible_bin_3.0-ces.run`, предназначенный для установки подсистемы управления конфигурациями;
– исполняемый файл `co_infra_3.0-ces.run`, предназначенный для установки хранилища образов Docker.

1.5 Программные и аппаратные требования

Программные и аппаратные требования к текущей версии ПО указаны в документе «"МойОфис Комплект Средств Разработки (SDK)". Сервер совместного редактирования. Системные требования».

1.6 Типовые схемы установки

Структура сервиса может быть представлена двумя типами установки:

– standalone (на один виртуальный сервер или на несколько виртуальных серверов в рамках одного физического сервера);
– кластерная (все роли устанавливаются на разные виртуальные или физические серверы).

1.6.1 Standalone

Конфигурация без отказоустойчивости используется для разработки или демонстрации возможностей продукта (virtual appliance).

Установка в минимальной конфигурации использует три сервера:

– сервер с ролью `operator` для управления процессом установки;
– сервер с ролью `cosa` для установки редакторов и дополнительного ПО;
– сервер NextCloud для размещения и хранения базовых библиотек и файлов.

1.6.2 Кластерная установка

Отказоустойчивая конфигурация, используемая для типовой установки продукта.

Для сохранения уровня отказоустойчивости не рекомендуется совмещать серверные роли между собой. Совмещение допускается в отдельных случаях для экономии ресурсов.

2 ПОДГОТОВКА К УСТАНОВКЕ

2.1 Конфигурирование ОС Astra

2.1.1 Установка на Astra SE 1.7 в защищенных вариантах

Основные отличия между вариантами защищенности Astra SE 1.7 приведены в таблице 2.

Таблица 2 — Уровни защищенности ОС Astra

Функция безопасности	Уровень защиты «Базовый»*	Уровень защиты «Усиленный»*	Уровень защиты «Максимальный»*
Замкнутая программная среда	Не доступна	Доступна (по умолчанию выключена)	Доступна (по умолчанию выключена)
Очистка освобождаемой внешней памяти	Не доступна	Доступна (по умолчанию выключена)	Доступна (по умолчанию выключена)
Мандатный контроль целостности	Не доступна	Доступна (по умолчанию включена)	Доступна (по умолчанию включена)
Мандатное управление доступом	Не доступна	Не доступна	Доступна (по умолчанию включена)

* — наименование ОС Астра в соответствии с уровнем защиты:

– Базовый уровень — Астра 1.7 «Орел»;

– Усиленный уровень — Астра 1.7 «Воронеж»;

– Максимальный уровень — Астра 1.7 «Смоленск».

Текущий уровень защищенности проверяется с помощью команды:

```
root@voronezh:~# astra-modeswitch list
0   base(orel)
1   advanced(voronezh)
2   maximum(smolensk)
root@voronezh:~# astra-modeswitch get
```

1Текущий статус замкнутой программной среды проверяется с помощью команды:

```
root@voronezh:~# astra-digsig-control status
ACTIVE
```

Текущий статус очистки освобождаемой внешней памяти (очистка разделов подкачки и гарантированное удаление файлов) проверяется с помощью команды:

```
root@voronezh:~# astra-swapwiper-control status
ACTIVE
root@voronezh:~# astra-secdel-control status
ACTIVE
```

Текущий статус мандатного контроля целостности проверяется с помощью команды:

```
root@voronezh:~# astra-mic-control status
ACTIVE
```

Текущий статус мандатного управления доступом проверяется с помощью команды:

```
root@voronezh:~# astra-mac-control status
INACTIVE
```

Текущий статус запрета включения бита выполнения проверяется с помощью команды:

```
root@voronezh:~# astra-nochmodx-lock status
ACTIVE
```

2.1.2 Установка на усиленном уровне защищенности («Воронеж»)

Установка осуществляется Ansible от имени пользователя astra, для которого должна быть настроена возможность выполнять sudo без пароля.

1. Пользователю astra необходимо установить максимальный уровень целостности 63 (соответствует администратору ОС). Проверить уровень целостности пользователя возможно с помощью команды:

```
root@voronezh:~# pdp-id -i
63
```

2. Установка Ansible и работа (версии 3.0)СЗР (версии 3.0) невозможна при включенном запрете бита исполнения. Перед началом установки на всех серверах необходимо выполнить команду:

```
astra@voronezh:~$ sudo astra-nochmodx-lock disable
astra@voronezh:~$ sudo astra-nochmodx-lock status
INACTIVE
```

3. Установка Ansible и работа (версии 3.0)СЗР (версии 3.0) невозможна при включенном режиме замкнутой программной среды. Необходимо проверить статус режима с помощью команды:

```
astra@voronezh:~$ sudo astra-digsig-control status
INACTIVE
```

4. При статусе `ACTIVE` перед началом установки на всех серверах необходимо выполнить команду:

```
astra@voronezh:~$ sudo astra-digsig-control disable
astra@voronezh:~$ sudo reboot
astra@voronezh:~$ sudo astra-digsig-control status
INACTIVE
```

5. Необходимо проверить статусы параметров безопасности, значения которых должны соответствовать таблице 3.

Таблица 3 — Параметры безопасности по умолчанию

Наименование команды	Статус
astra-bash-lock status	INACTIVE
astra-commands-lock status	INACTIVE
astra-docker-isolation status	INACTIVE

Наименование команды	Статус
astra-hardened-control status	INACTIVE
astra-interpreters-lock status	ACTIVE
astra-lkrg-control status	INACTIVE
astra-macros-lock status	INACTIVE
astra-modban-lock status	INACTIVE
astra-overlay status	INACTIVE
astra-ptrace-lock status	ACTIVE
astra-sumac-lock status	INACTIVE
astra-shutdown-lock status	INACTIVE
astra-ufw-control status	INACTIVE
astra-ulimits-control status	INACTIVE

6. Для проверки доступности репозиториев необходимо выполнить команду:

```
apt-get update
```

Команда должна завершаться без ошибки.

При наличии сбойного зеркала репозитория (например, http://mirror.yandex.ru/astra/stable/orel/repository orel InRelease), его необходимо удалить из директории `/etc/apt/sources.list`.

2.2 Конфигурирование CentOS

При использовании CentOS 7 на серверах необходимо выполнить следующие команды:

```
sed -i s/mirror.centos.org/vault.centos.org/g /etc/yum.repos.d/*.repo
sed -i s/^#.*baseurl=http/baseurl=http/g /etc/yum.repos.d/*.repo
sed -i s/^mirrorlist=http/#mirrorlist=http/g /etc/yum.repos.d/*.repo
```

Из-за прекращения поддержки CentOS 7 со стороны компании RedHat следует отключить обновление ядра в соответствии с разделом «Порядок обновления ядра Linux».

2.3 Настройка сетевых соединений

Настройка сетевого соединения должна быть выполнена на всех серверах, предназначенных для установки системы, в том числе на сервере с ролью `operator`.

Для работы серверов в локальной сети необходимо задать следующие параметры:

– IP-адрес;

– Маска подсети;

– Основной шлюз;

– DNS-сервер.

Пример настройки сетевого соединения с помощью командной строки в ОС Astra представлен в приложении В.

2.4 Подготовка сервера с ролью operator

2.4.1 Установка в сети без выхода в интернет

Для установки ССР в локальной сети, без прямого выхода в интернет, необходимо обеспечить доступность дополнительных пакетов ПО. Перечень необходимого ПО приведен в документе «"МойОфис Комплект Средств Разработки (SDK)". Сервер совместного редактирования. Системные требования».

Для обеспечения доступности следует выполнить следующие действия:

– скачать файлы со стандартного репозитория с помощью прямого выхода в интернет;

– разместить файлы на локальном сервере;

– обеспечить доступ к серверу по локальной сети;

– установить ПО и настроить локальный репозиторий (см. Приложение А);

– выполнить замену стандартного репозитория на локальный (см. Приложение Б).

Замена стандартного репозитория на локальный выполняется на сервере с ролью `operator`.

2.4.2 Установка подсистемы управления конфигурациями

Установка выполняется на сервере с ролью `operator`. Порядок действий при установке:

1. Скопировать файл `co_ansible_bin_3.0-ces.run` в корневую директорию пользователя (где `3.0-ces` — имя версии).

2. Запустить скрипт установки:

```
bash co_ansible_bin_3.0-ces.run
```

3. Дать согласие на продолжение установки, нажав на клавишу «Y». Пример запроса:

```
Do you want to continue?[y/N] y
```

4. После завершения установки на экране пользователя будет отображен список выполненных операций и сообщения. Необходимо убедиться, что список содержит сообщения `[ OK ]` или `[CHANGE]` — это свидетельствует об успешной установке.

При получении сообщения `[FAIL]` необходимо обратиться в техническую поддержку.

После выполнения скрипта установки будет создана директория `~/install_co`.

2.4.3 Установка дополнительного ПО

В соответствии с разделом «Программные требования» на сервере с ролью `operator` необходимо установить пакеты дополнительного ПО.

Рекомендуется использовать «чистую» ОС для предотвращения появления ошибок, связанных с использованием конфигурационных файлов.

Конфигурационные файлы, установленные по умолчанию (например: `/etc/ansible/ansible.cfg`), необходимо удалить или заменить файлами из комплекта поставляемого ПО.

Для установки пакетов необходимо обеспечить серверу с ролью `operator` выход в интернет.

Запрещается установка последних версий дополнительного ПО, доступных в репозитории. Перед установкой следует ознакомиться с требованиями к версиям ansible-core и модулям Python.

2.4.4 Автоматическая установка дополнительного ПО

Установка дополнительного ПО может быть выполнена автоматически с помощью скрипта установки `venv_setup.sh`, расположенного в директории `~/install_co/contrib`.

Для запуска автоматической установки необходимо выполнить команду:

```
bash ~/install_co/contrib/venv_setup.sh
```

После выполнения скрипта будет создана директория `~/venv`. Для использования директории следует выполнить команду:

```
source ~/venv/bin/activate
```

Все последующие операции, связанные с ПО Python и Ansible, необходимо выполнять с включенной директорией `~/venv`.

2.4.5 Установка хранилища образов Docker

Установка выполняется на сервере с ролью `operator`. Порядок действий при установке:

1. Скопировать файл `co_infra_3.0-ces.run` на сервер с ролью `operator` (где `3.0` — имя версии).
2. Запустить скрипт установки:

```
bash co_infra_3.0-ces.run
```

3. Дождаться проверки целостности файла и его распаковки.

Пример вывода:

```
Verifying archive integrity...100%       MD5 checksums are OK. All good.
Uncompressing Co Infrastructure Node Package [RELEASE]100%
```

4. Дать согласие на продолжение установки, нажать на клавишу «Y».

```
Do you want to continue?  [y/N] y
```

5. После завершения работы исполняемого файла на экране пользователя будет отображен список выполненных операций. Необходимо убедиться, что список содержит сообщения `[ OK ]` или `[CHANGE]` — это свидетельствует об успешной установке.

При получении сообщения `[FAIL]` необходимо обратиться в техническую поддержку.

Для использования других систем контейнеризации необходимо обратиться в техническую поддержку.

2.4.6 Настройка зависимостей Python

На сервере с ролью `operator` зависимости Pyhton указаны в файле `~/install_co/contrib/ces/requirements.txt`.

Для использования зависимостей из файла необходимо выполнить команду:

```
pip3 install -r ~/install_co/contrib/ces/requirements.txt
```

При установке модулей Pyhton с помощью скрипта `venv_setup.sh` настройка зависимостей выполняется автоматически.

2.5 Подготовка конфигурационных файлов

Все операции с конфигурационными файлами выполняются на сервере с ролью `operator`.

2.5.1 Интеграция по протоколу WOPI

В файле переменных `~/install_co/group_vars/co/main.yml` необходимо указать домен сервиса Nextcloud в переменной `csp_allowed_frame_ancestors`, например:

```
csp_allowed_frame_ancestors:
  - "nextcloud.example.net"
```

Примечания:

1. Интеграция с мессенджерами в режиме WOPI недоступна.

2. Интеграция с WOPI тестировалась только для хранилища Nextcloud с плагином https://apps.nextcloud.com/apps/officeonline.

В конфигурационном файле Nextcloud `config/config.php` необходимо добавить переменную `'allow_local_remote_servers'=> true`.

На сервере Nextcloud в режиме администратора следует настроить интеграцию с Office Online. Указать адрес сформированный в соответствии с разделом «Внешние DNS-записи»:

```
https://docs[-<domain_env>.]<domain_name>
```

Для устранения ошибки, при которой копирование выделенного текста не работает в режиме просмотра документа, необходимо предоставить `iframe` доступ к Clipboard API следующим образом:

```
<iframe src="https://docs[-<domain_env>.]<domain_name>" \
allow="clipboard-read; clipboard-write"></iframe>
```

2.5.2 Порядок размещения и заполнения файлов конфигурации

Директория установки содержит предзаполненные файлы конфигураций, подготовленные для упрощения настройки системы. Директория `~/install_co/contrib/ces` содержит два каталога с файлами конфигурации: для standalone и кластерной установки.

При обновлении системы допускается использование скопированных и заполненных файлов конфигурации предыдущей версии. Для актуализации значений переменных и параметров установки необходимо ознакомиться со списком изменений в приложении Д.

В примере показан порядок размещения и настройки файлов конфигурации для кластерной установки:

1. Перейти в каталог `~/install_co/` с помощью команды:

```
cd ~/install_co
```

2. Скопировать файл `~/install_co/contrib/ces/ansible.cfg` в корневой раздел директории установки с помощью команды:

```
cp ~/install_co/contrib/ces/ansible.cfg ansible.cfg
```

3. Подготовить файл `hosts.yml`

Примеры заполненных файлов можно найти в каталоге `~/install_co/contrib/ces/`.

Внутри директории `~/install_co/contrib/ces` находятся два каталога: `cluster` и `standalone`.

В зависимости от типа установки (см. раздел «Типовые схемы установки») необходимо выбрать соответствующую директорию и скопировать файл `hosts.yml` с помощью команды:

```
cp ~/install_co/contrib/ces/cluster/hosts.yml hosts.yml
```

В примере указан путь для кластерной установки.

4. Заполнить файл `hosts.yml` в соответствии с решением об используемой архитектуре.
5. Скопировать SSL-ключи для внешнего домена в каталог `certificates`. Подробнее см. в разделе «Размещение SSL-сертификатов для шифрования».

6. Создать в директории групповых переменных `~/install_co/group_vars` каталог для серверов с именем группы установки из файла `hosts.yml`. По умолчанию при установке в указанной директории создается каталог `co_setup`.
7. Скопировать в директорию групповых переменных `group_vars` каталог с переменными для заполнения:

```
cp -r ~/install_co/contrib/ces/cluster/group_vars/co_setup/* group_vars/co_setup
```

8. Открыть файл `main.yml` из каталога размещения и отредактировать значения параметров в соответствии с разделом «Конфигурирование файла main.yml».

2.5.3 Конфигурирование файла hosts.yml

В файл `hosts.yml` вносятся только полностью определенные имена доменов (FQDN). Эти домены будут использоваться во время и после установки системы для обращения к внутренним сервисам.

Преднастроенный файл `hosts.yml` (скопированный в соответствии с п. 3 раздела «Порядок размещения и заполнения файлов конфигурации») содержит примеры заполнения в следующем формате: `co-etcd-1.installation.example.net:`

где:

– `co-etcd-1` — имя сервера для подгруппы `co-etcd`;

– `installation.example.net` — имя домена установки.

Запись в файле `hosts.yml` при использовании группы серверов отличается записью имени сервера: `co-etcd-[1:3].installation.example.net:`

где: `co-etcd-[1:3]` — группа серверов `co-etcd`.

В кластерной конфигурации используется один или несколько серверов для одной роли.

Пример заполнения файла `hosts.yml`:

```
all:
  children:

    co:
      children:                                          # Перечень групп
        co_audit:                                        # Подгруппа co_audit
          hosts:
            co-audit-[1:2].installation.example.net:   # Имя DNS-сервера

        co_etcd:
          hosts:
            co-etcd-[1:3].installation.example.net:
```

В конфигурации standalone для всех ролей используется один и тот же сервер.

Пример заполнения файла `hosts.yml`:

```
all:
  children:

    co:
      children:
        co_audit:
          hosts:
            co-infra-1.installation.example.net:

        co_etcd:
          hosts:
            co-infra-1.installation.example.net:
```

Объединение ролей может применяться в кластерной установке, если ресурсы организации ограничены.

В соответствии с выполненными DNS-записями и принятым решением об архитектуре устанавливаемой системы необходимо заполнить файл `hosts.yml`.

2.5.4 Конфигурирование файла main.yml

Для первичной установки системы необходимо скопировать предзаполненный файл конфигурации из директориии `~/install_co/contrib/ces/`. Порядок подготовки файла `main.yml` определен в разделе «Порядок размещения и заполнения файлов конфигурации».

При повторной установке необходимо открыть с помощью текстового редактора файл расположенный в директории `~/install_co/group_vars/co_setup/main.yml` и изменить значения для обязательных переменных, перечисленных в таблице 5.

Для коректной работы системы необходимо вручную добавить в конфигурационный файл `main.yml` переменные и значения, указанные в таблице 4.

Таблица 4 — Дополнительные переменные конфигурации

Наименование переменной	Значение
chatbot_messenger	"none"
chatbot_squadus_login	""
chatbot_squadus_password	""
chatbot_squadus_server	""

Описание переменных из конфигурационного файла `main.yml` представлено в таблице 5.

Таблица 5 — Основные переменные

Наименование переменной	Заполнение обязательно	Описание
Конфигурация Ansible		
ansible_user	-	Имя пользователя, с которым Ansible подключается к хостам по ssh
co_domain_module	-	Строка-шаблон формирования полного доменного имени
co_external_domain	-	Основной домен, на котором будет работать система
domain_env	-	Домен зоны устанавливается в соответствии с разделом «Внешние DNS-записи»
domain_name	+	Имя домена, указывается в соответствии с доменом установки
Конфигурация CA (Центра сертификации)		
ca_main_config.auth_keys.services.key	+	Сгенерировать ключ для доступа к CFSSL API с помощью команды: `"openssl rand -hex 16"`
Конфигурация Docker		
docker_daemon_parameters: insecure-registries	+	Установка реестра образов. Заменить на IP-адрес или FQDN имя сервера с ролью `operator` и порт 5000 (например `["10.1.2.3:5000"]`)
bip	-	Адрес сетевого интерфейса (моста) Docker
dns	-	Внутренние DNS-серверы (если не используется unbound)
mtu	-	Размер сетевого пакета сети Docker (может изменяться в виртуальных сетях OpenStack)
docker_image_registry	+	Установка реестра контейнеров. Заменить на IP-адрес или FQDN имя сервера с ролью `operator` и порт 5000 (например `10.1.2.3:5000`)

Наименование переменной	Заполнение обязательно	Описание
cu_pool_size	-	Количество conversion units (оставить без изменения)
pregen_pool_size	-	Количество pregen units (оставить без изменения)
du_pool_size	-	Количество document units (оставить без изменения)
Конфигурация ETCD		
etcd_browser_password	+	Имя пользователя для веб-доступа к etcd
etcd_browser_username	-	Имя пользователя для веб-доступа к etcd
Конфигурация Grafana		
grafana_admin_password	+	Пароль администратора grafana
Конфигурация ELK		
elasticsearch_admin_password	+	Пароль администратора elasticsearch
elasticsearch_admin_password_hash	+	Хеш пароля администратора elasticsearch
elasticsearch_kibana_password_hash	+	Хеш пароля пользователя elasticsearch Kibana
kibana_elasticsearch_password	+	Пароль пользователя elasticsearch Kibana
Конфигурация KEEPALIVED		
keepalived_redis_password	+	Пароль авторизации в keepalived для конфигурации redis
keepalived_redis_vip_address	+	IPv4 адрес в подсети серверов кластерной установки
Конфигурация RabbitMQ		
rabbitmq_federation_enabled	-	Включение федерации RabbitMQ (значение: `true` или `false`)
rabbitmq_users.root.password	+	Пароль для root пользователя RabbitMQ
rabbitmq_users.couser.password	+	Пароль для `couser` пользователя RabbitMQ
Конфигурация REDIS		
redis_password	+	Пароль для Redis команды AUTH
Конфигурация TLS		
tls_ca_filename	-	Имя файла с промежуточными доверенными сертификатами

Наименование переменной	Заполнение обязательно	Описание
tls_cert_filename	-	Имя файла сертификата на домены, указанные в разделе «Настройка DNS»
tls_key_filename	-	Имя файла приватного ключа от доменного сертификата
Конфигурация Openresty		
openresty_api_username	-	Имя пользователя для доступа к CO Manage API
openresty_api_password	+	Пароль пользователя для доступа к CO Manage API

Для генерации паролей и salt рекомендуется использовать утилиту pwgen. Безопасный пароль необходимо генерировать с помощью команды:

```
pwgen <длина пароля> 1
```

где `<длина пароля>` — должна быть не меньше 20 символов.

Для генерации хешей паролей необходимо использовать утилиту htpasswd. Хеш генерируется с помощью команды:

```
htpasswd -bnBC 12 "" <пароль> | tr -d ':\n'
```

Дополнительные переменные перечислены в таблице 6. Для изменения значения необходимо открыть с помощью текстового редактора файл `extra_vars.yml`, расположенный в директории: `~/install_co/group_vars/co_setup`.

Таблица 6 — Дополнительные переменные

Наименование роли	Заполнение обязательно	Описание
unbound_forward_addresses	-	Список внешних или внутренних DNS, на которые будут отсылаться запросы из unbound

2.6 Создание и размещение сертификатов

2.6.1 Создание SSL-сертификатов

Для обеспечения защищенного соединения между пользователем и сервером CCP используется проверка SSL-сертификата. Организации необходимо установить SSL-сертификат на свой сервер, чтобы поддерживать безопасную сессию с браузерами пользователей.

SSL-сертификаты выпускаются доверенным центром сертификации. Браузеры, ОС и мобильные устройства поддерживают список корневых сертификатов доверенных центров сертификации.

В отдельных случаях (например для демонстрации продукта) допускается использование самоподписанного сертификата. Порядок создания самоподписанных сертификатов описан в приложении Г.

Для упрощения настройки файл переменных `~/install_co/group_vars/co_setup/main.yml` (подготовленный в соответствии с требованиями раздела «Порядок размещения и заполнения файлов конфигурации») содержит имена сертификатов по умолчанию (секция `TLS cert and key filenames`).

Необходимо использовать сертификаты, выданные центром сертификации для вашей организации, или создать группу новых самоподписанных сертификатов.

2.6.2 Размещение SSL-сертификатов для шифрования

Порядок размещения сертификатов:

1. Разместить сертификат внешнего домена:

```
~/install_co/certificates/server.crt
```

2. Разместить ключ внешнего домена:

```
~/install_co/certificates/server.nopass.key
```

3. Разместить цепочку сертификатов промежуточных центров сертификации (CA) внешнего домена:

```
~/install_co/certificates/ca.pem
```

2.7 Настройка DNS

2.7.1 Внутренние DNS-записи

Внутренние DNS-записи предназначены для установки системы на серверы кластера.

Для всех серверов, перечисленных в файле `hosts.yml` в соответствии с разделом «Конфигурирование файла hosts.yml» необходимо создать DNS-записи. Для создания записей необходимо использовать DNS-сервер вашей организации.

При отсутствии DNS-сервера организации необходимо создать записи на сервере с ролью `operator` в файле `/etc/hosts`.

Пример содержимого файла `/etc/hosts` для установки типа standalone:

```
192.168.1.100 co-infra-1.installation.example.net
```

Пример содержимого файла `/etc/hosts` для кластерной установки:

```
192.168.1.100 co-etcd-1.installation.example.net
192.168.1.101 co-etcd-2.installation.example.net
192.168.1.102 co-etcd-3.installation.example.net
192.168.1.103 co-imc-mq-1.installation.example.net
192.168.1.104 co-imc-mq-2.installation.example.net
192.168.1.105 co-imc-mq-3.installation.example.net
```

Количество записей соответствует количеству используемых физических или виртуальных серверов.

DNS-сервер организации должен содержать аналогичные записи в соответствии с требованиями собственной настройки.

2.7.2 Внешние DNS-записи

Внешние DNS-записи предназначены для подключения пользователей к сервисам.

На DNS-сервере вашей организации необходимо создать записи в соответствии с таблицей 7 или 8.

При отсутствии DNS-сервера организации необходимо создать записи на сервере с ролью `operator` в файле `/etc/hosts` (см. раздел «Внутренние DNS-записи»).

Запрещается использовать в качестве домена зону *.local

Таблица 7 сформирована для параметра `co_domain_module` со значением `{service}.{domain}` (т.е. формирование ссылок через точку к указанному домену).

При формировании записи `{service}.{domain}` переменная `<domain_env>` не используется. В файле `~/install_co/group_vars/co_setup/main.yml` значение переменной `domain_env` должно быть пустым:

```
domain_env: ""
```

Таблица 7 — Внешние DNS-записи со значением `{service}.{domain}`

Имя записи	Тип записи	Значение	Комментарий
auth.<domain_name>	A	IP-адрес сервера, указанного в группе co_lb_core_wopi	Адрес приложения авторизации и целевой страницы Auth SSO. Количество A записей должно соответствовать количеству серверов
cdn.<domain_name>	CNAME	auth.<domain_name>	Адрес CDN
coapi.<domain_name>	CNAME	auth.<domain_name>	Адрес COAPI
docs.<domain_name>	CNAME	auth.<domain_name>	Адрес приложения редакторов
files.<domain_name>	CNAME	auth.<domain_name>	Адрес приложения файлового менеджера
links.<domain_name>	CNAME	auth.<domain_name>	Адрес ссылок на документы
_https._tcp.<domain_name>	SRV	auth.<domain_name>	Опционально, для мобильных клиентов

Таблица 8 сформирована для параметра `co_domain_module` со значением `{service}-{domain}` (т.е. формирование ссылок через тире к указанному домену).

Таблица 8 — Внешние DNS-записи со значением `{service}-{domain}`

Имя записи	Тип записи	Значение	Комментарии
auth-<domain_env> .<domain_name>	A	IP-адрес сервера, указанного в группе co_lb_core_wopi	Адрес приложения авторизации и целевой страницы Auth SSO. Количество A записей должно соответствовать количеству серверов
cdn-<domain_env> .<domain_name>	CNAME	auth-<domain_env> .<domain_name>	Адрес CDN
coapi-<domain_env> .<domain_name>	CNAME	auth-<domain_env> .<domain_name>	Адрес COAPI
docs-<domain_env> .<domain_name>	CNAME	auth-<domain_env> .<domain_name>	Адрес приложения редакторов
files-<domain_env> .<domain_name>	CNAME	auth-<domain_env> .<domain_name>	Адрес приложения файлового менеджера
links-<domain_env> .<domain_name>	CNAME	auth-<domain_env> .<domain_name>	Адрес ссылок на документы
_https._tcp-<domain_env> .<domain_name>	SRV	auth-<domain_env> .<domain_name>	Опционально, для мобильных клиентов

2.7.3 Настройка внутренних DNS-записей

Во время установки производится настройка и запуск локального кеширующего DNS-сервера (Unbound) на серверах группы `co_etcd`. Сервер служит для обработки запросов внутри установки и предназначен для работы контейнеров и серверов через соответствующие параметры групповых переменных.

По умолчанию серверы будут перенастроены на работу через Unbound и не будут принимать параметры DNS-серверов по DHCP.

При необходимости Unbound может быть сконфигурирован для работы с внутренними DNS-серверами. По умолчанию Unbound настроен на маршрутизацию запросов на адреса 8.8.8.8 и 8.8.4.4.

DNS-записи, используемые для работы внутри установки, формируются через «.» (точку) относительно вписанного в файл inventory имени сервера. DNS-записи создаются в Unbound автоматически на основе переменных Ansible.

Этот параметр можно переопределить двумя способами:

1. Заполнить все адреса вручную на основе примеров в файле групповых переменных, расположенного в следующей директории:

```
~/install_co/group_vars/co_setup/extra_vars.yml
```

2. Заполнить все необходимые записи на внешнем DNS-сервере без использования Ansible. При подобном варианте необходимо создать «A» — записи для каждого сервера, вписанного в файл `~/install_co/contrib/ces/cluster/hosts.yml`, а также CNAME адреса на все поддомены «*» к каждому серверу, вписанному в `hosts.yml`.

Пример заполнения таких записей приведен в таблице 9.

Таблица 9 — Пример заполнения

Имя записи	Тип записи	Значение
co-infra-1	A	10.10.1.110
*.co-infra-1	CNAME	co-infra-1

После настройки Unbound должен быть недоступен из внешней сети.

При использовании `/etc/hosts` для создания DNS-записей необходимо добавить в файл `~/install_co/group_vars/co_setup/extra_vars.yml` все записи, перечисленные в `/etc/hosts`. Например:

```
unbound_local_zones:
-  type: "transparent"
   zone: "installation.example.net"
   local_data:
        - domain: "co-etcd-1.installation.example.net"
          type: "A"
         ip: "10.1.2.3"
```

2.7.4 Проверка работы DNS на сервере с ролью operator

После настройки необходимо проверить доступность DNS на сервере с ролью `operator`.

При использовании внешнего DNS-сервера необходимо открыть файл `~/install_co/group_vars/co_setup/extra_vars.yml` с помощью текстового редактора и добавить адрес DNS-сервера, изменив IP-адрес:

```
# DNS settings in /etc/resolv.conf
unbound_forward_addresses:
- "127.0.0.1"
- "8.8.8.8"
```

Для проверки соответствия доменного имени IP-адресу сервера необходимо:

1. Установить ПО с помощью команды:

```
apt install dnsuntils
```

или

```
yum install bind-utils
```

Выбор команды зависит от типа ОС.

2. После установки ПО выполнить следующую команду:

```
> dig A co-infra-1.installation.example.net
```

Пример ответа:

```
; <<>> DiG 9.18.1-1ubuntu1.2-Ubuntu <<>> A co-infra-
1.installation.example.net
;; global options: +cmd
;; Got answer:
;; opcode: QUERY, status: NOERROR, id: 45369
;; flags: qr rd ra; QUERY: 1, ANSWER: 2, AUTHORITY: 0, ADDITIONAL: 1
;; OPT PSEUDOSECTION:
; EDNS: version: 0, flags:; udp: 65494;;
QUESTION SECTION:
;co-infra-1.installation.example.net. IN A ;;
ANSWER SECTION:
*.co-infra-1.installation.example.net. 900 IN CNAME co-infra-
1.installation.example.net.
co-infra-1.installation.example.net. 900 IN A 192.168.0.1
;; Query time: 23 msec
;; SERVER: 127.0.0.53#53(127.0.0.53) (UDP) ;; WHEN: Tue Jan 10 15:56:32
MSK 2023
;; MSG SIZE rcvd: 95
```

В ответе необходимо найти секцию `ANSWER SECTION` и проверить, что доменное имя соответствует IP-адресу.

```
*.co-infra-1.installation.example.net. 900 IN CNAME co-infra-
1.installation.example.net.
co-infra-1.installation.example.net. 900 IN A 192.168.0.1
```

3 ДОПОЛНИТЕЛЬНЫЕ ПАРАМЕТРЫ УСТАНОВКИ

В разделе представлены дополнительные параметры установки системы. Настройка перечисленных функций не обязательна.

Если специализированные требования к установке отсутствуют, необходимо перейти в раздел «Запуск установки».

3.1 Порядок обновления ядра Linux

При установке ОС на серверы кластера ядро может быть автоматически обновлено до минимальной требуемой версии. По умолчанию ядро обновляется на kernel-lt (LTS) в ОС Redhat-based (CentOS, РЕД ОС). В ОС Debian-based (Ubuntu, Astra) по умолчанию ядро не обновляется. Поддержка других ядер не гарантируется, обратитесь в техническую поддержку за более подробной информацией.

Для отключения обновления в ОС Redhat-based (CentOS, РЕД ОС) при запуске установки необходимо использовать следующую команду:

```
-e kernel_ml_enabled=false
```

Для обновления ядра до kernel-lt (LTS) в ОС Debian-based (Ubuntu, Astra) при запуске установки необходимо использовать следующую команду:

```
-e kernel_ml_deb_enabled=true
```

В ОС Altlinux автоматическое обновление ядра не поддерживается.

3.2 Настройка дополнительных серверов для аудита

Настройка дополнительных Fluentd серверов для сбора событий выполняется с помощью текстового редактора в файле `~/install_co/group_vars/co_setup/main.yml`. Необходимо добавить в файл перечисленные команды, изменив IP-адреса и порты:

```
# LOG servers for the environment
fluentd_server_upstream_log_servers:
  - ip: "10.10.10.10"
    port: 24225
  - ip: "11.11.11.11"
    port: 24225
```

Данная настройка применяется только при использовании в установке роли `log`. Включение функции задается с помощью переменной, указанной в таблице 10.

Таблица 10 — Подключение серверов аудита

Расположение переменной	Наименование переменной	Тип переменной	Значение
group_vars/co_setup/ main.yml	common_fluent_ logging_enabled	boolean	true / false (по умолчанию)

3.3 Остановка и запуск системы с помощью консольных команд

Для работы с консолью ПО МойОфис администратору системы необходимо обеспечить ssh-доступ к серверу в контуре установки.

Сервисы ССР управляются с помощью Docker.

Просмотр списка сервисов на сервере подсистемы:

```
docker ps
```

Для остановки сервиса `<service_name>` из списка сервисов необходимо выполнить следующую команду:

```
docker stop <service_name>
```

Для перезапуска сервиса следует выполнить следующую команду:

```
docker restart <service_name>
```

Для остановки сервиса `docker` необходимо выполнить следующую команду:

```
systemctl stop docker
```

Для корректного завершения работы сервисов следует выполнить следующую команду:

```
shutdown <option>
```

Ноды сервисов рекомендуется выключать по очереди. Параметр `<option>` позволяет использовать дополнительные параметры выключения, в том числе таймер и опцию перезапуска.

Пример (немедленное выключение с остановкой сервисов):

```
shutdown -h now
```

Запуск подсистемы осуществляется при инициализации и запуске аппаратной части.

3.4 Настройка обработки журналов

Настройка обработки журналов (logrotate) в текущей версии ПО не автоматизирована и настраивается самостоятельно администратором.

3.5 Настройка ротации журналов событий в ElasticSearch

Для защиты диска от переполнения записи журнала событий старше 120 дней автоматически удаляются. Процедура использует политики удаления устаревших индексов в ElasticSearch.

Период автоматического удаления (в днях) задается при развертывании в файле `~/install_co/group_vars/co_setup/extra_vars.yml` с помощью переменной `es_index_retention_period_days`.

3.6 Настройка автоматического отключения неактивного пользователя

ССР позволяет автоматически отключать пользователей от редактируемого документа, в случае их бездействия.

Для настройки необходимо присвоить новые значения переменным, перечисленным в таблице 11.

Таблица 11 — Переменные для автоматического отключения неактивного пользователя

Наименование сервиса	Наименование переменной	Тип переменной	Размерность	Значение по умолчанию	Описание
etcd (CO)	config/dcm/ romdocuments. ttlSecs	number	секунды	360	Время хранения кешированного файла документа на диске сервиса DCM
	config/dcm/ dcm.du.edits. expireSecs	number	секунды	11000	Период работы в режиме редактирования без сохранения. При истечении времени выполняется перезапуск сервиса DU с разрывом сессии редактирования
	config/nps-du/ Du.Env. TimeInactivityMins	number	минуты	180	Время до автомати-ческого разрыва сессии редактирования при бездействии пользователя

На этапе развертывания ССР присвоить новое значение возможно только для времени до автоматического разрыва сессии редактирования при бездействии пользователя. Для изменения значения переменной при запуске скрипта установки необходимо использовать следующую команду:

```
-e DU_MAX_TIME_FOR_INACTIVE_COLLABORATOR_MINS=120
```

Порядок запуска скрипта установки описан в разделе «Установка».

3.7 Карта портов

Карта портов представлена в таблице 12.

Таблица 12 — Карта портов

Группа Ansible	Сервис	Прослушиваемый порт	Порт подключения	Куда обращается
all	node_exporter	9100/tcp	-	-
	cadvisor	9101/tcp	-	-
	fluentd_agent	5140/udp, 5160/udp, 5165/udp, 5180/tcp, 24224/tcp, 5185/udp, 2430/tcp	-	-
	docker	-	5000/tcp	docker-registry
co	haproxy	20001/tcp, 20002/tcp, 20004-20007/tcp	-	-
co_lb_core_wopi	openresty-lb-core-auth	80/tcp, 443/tcp, 8080/tcp, 8443/tcp, 8888/tcp	20001/tcp, 20002/tcp, 20004/tcp, 20005/tcp, 20007/tcp, 8404/tcp	haproxy
co_etcd	etcd	2379/tcp, 2380/tcp	-	-
	etcd_browser	8001/tcp	-	-
co_mq	rabbitmq	4369/tcp, 5672/tcp, 15672/tcp, 25672/tcp	-	-
co_cvm	cvm	9094/tcp	2379/tcp	etcd
			20005/tcp, 20006/tcp, 20007/tcp	haproxy
			5160/tcp, 5180/tcp, 24224/tcp	fluentd-agent
co_cu	sdd_cu	9097/tcp	24224/tcp	fluentd-agent
	cu	30000-65535/tcp	26379/tcp	redis_sentinel
			9097/tcp	cu
			24224/tcp	fluentd-agent
co_dcm	dcm	9095/tcp	2379/tcp	etcd
			20001/tcp, 20002/tcp, 20004/tcp, 20005/tcp, 20006/tcp,	haproxy

Группа Ansible	Сервис	Прослушиваемый порт	Порт подключения	Куда обращается
			20007/tcp	
			5160/tcp, 5180/tcp, 24224/tcp	fluentd-agent
			443/tcp	nextcloud
			26379/tcp	redis_sentinel
co_du	du	30000-65535/tcp	26379/tcp	redis_sentinel
			9098/tcp	du
			24224/tcp	fluentd-agent
	sdd_du	9098/tcp	24224/tcp	fluentd-agent
co_jod	jod	9096/tcp	2379/tcp	etcd
			20001/tcp, 20002/tcp, 20004/tcp, 20005/tcp, 20006/tcp, 20007/tcp	haproxy
			5160/tcp, 5180/tcp, 24224/tcp	fluentd-agent
			26379/tcp	redis_sentinel
co_nm	nm	9092/tcp	2379/tcp	etcd
			20001/tcp, 20002/tcp, 20004/tcp, 20005/tcp, 20006/tcp, 20007/tcp	haproxy
			5160/tcp, 5180/tcp, 24224/tcp	fluentd-agent
			26379/tcp	redis_sentinel
co_pregen	pregen	8002/tcp	24224/tcp	fluentd-agent
	lsyncd	9022/tcp	-	-
co_imc	redis	6379/tcp, 16379/tcp	-	-
	redis_sentinel	26379/tcp	6379/tcp	redis
co_infra	ca	8890/tcp	-	
	nginx	80/tcp, 81/tcp*	9090/tcp	prometheus
			3000/tcp	grafana
			9093/tcp	alertmanager
			5601/tcp	kibana
			8001/tcp	etcd_browser
	prometheus	9090/tcp	9093/tcp	alertmanager
			9115/tcp	blackbox_exporter
			9101/tcp	cadvisor
			2379/tcp	etcd

Группа Ansible	Сервис	Прослушиваемый порт	Порт подключения	Куда обращается
			9100/tcp	node_exporter
			9121/tcp	redis_exporter
	grafana	3000/tcp	-	-
	alertmanager	9093/tcp	-	-
	blackbox_exporter	9115/tcp	-	-
	redis_exporter	9121/tcp	-	-
	elasticsearch	9200/tcp, 9300/tcp	-	-
	kibana	5601/tcp	9200/tcp	elasticsearch
	fluentd_server	2430/tcp, 5140/udp, 5160/udp, 5165/udp, 5180/tcp, 5185/udp, 24224/tcp	-	-
operator	docker-registry	5000/tcp	-	-

4 УСТАНОВКА

4.1 Запуск установки

Запуск установки продукта выполняется на сервере с ролью `operator` с помощью команды:

```
ansible-playbook playbooks/main.yml --diff
```

Скорость установки зависит от выделенных вычислительных ресурсов. Для обеспечения непрерывности установки рекомендуется использовать дополнительное ПО Screen, Tmux.

В процессе выполнения команды запускаются роли, описанные в разделе «Конфигурирование файла main.yml».

4.2 Проверка корректности установки

Для проверки работоспособности установленного ПО и корректности установки необходимо запустить ПО «МойОфис Документы», выполнив следующие действия:

1. Открыть в поддерживаемом веб-браузере страницу установленного сервиса Nextcloud.

2. Войти в Nextcloud и открыть документ на редактирование.

4.3 Диагностика состояния подсистем

4.3.1 Диагностика состояния Nginx

Перечень проверок для диагностики состояния Ngnix указан в таблице 13.

Таблица 13 — Перечень проверок для диагностики Nginx

Тип проверки	Адрес	Примечание
Проверка статуса работы подсистем Auth/SSO и Core	https://<локальный-адрес-сервера>:8443/api/manage/core/status	Параметр «all» в ответе должен быть равен строке «OK»
	https://<локальный-адрес-сервера>:8443/api/manage/docs/status	
Проверка текущей конфигурации	https://<локальный-адрес-сервера>:8443/api/manage/config	
Просмотр журналов доступа и ошибок системы Auth/SSO (в случае отсутствия сервера с ролью `co_log`)	https://<локальный-адрес-сервера>:8443/api/manage/logs/error	В качестве альтернативы используется просмотр журналов событий на сервере с ролью`co_lb_core_wopi`, по умолчанию место расположения журнала событий: `/srv/docker/openresty/logs/`
	https://<локальный-адрес-сервера>:8443/api/manage/logs/access	
	https://<локальный-адрес-сервера>:8443/api/manage/logs/access_full	
Просмотр списка активных сессий и авторизованных пользователей подсистемы Auth/SSO	https://<локальный-адрес-сервера>:8443/api/manage/sessions	
	https://<локальный-адрес-сервера>:8443/api/manage/users	

Адрес сервера выбирается из указанных в группе `co_lb_core_wopi` файла `hosts.yml`.

Для обеспечения безопасности доступ к порту 8443, ограниченный на стороне Nginx, должен распространяться на локальный сервер и внутренние (частные) сети с адресами по стандарту RFC1918. Категорически запрещается открывать доступ к порту из публичных сетей.

4.3.2 Диагностика состояния Lsyncd

Диагностика состояния Lsyncd применяется только для кластерного режима установки (в standalone конфигурации lsyncd не используется).

Проверить синхронизацию необходимо в журнале событий с помощью команды:

```
docker logs --tail 10 lsyncd
```

Контейнер lsyncd должен быть запущен на всех узлах с ролью `co_lb_core_wopire_wopi`. Проверить статус его работы следует с помощью команды:

```
cat /srv/docker/lsyncd/conf/lsyncd/lsyncd.status
```

4.3.3 Диагностика состояния RabbitMQ

Проверка статуса очереди сообщений осуществляется через веб-интерфейс RabbitMQ по адресу http://<локальный-адрес-сервера>:15672. Логин и пароль для авторизации находится в переменных, используемых в текущей установке.

Адрес сервера выбирается из указанных в группе `co_mq` файла inventory. Предусмотрены возможности проверки состояния кластера RabbitMQ, создания или удаления очереди обмена или отдельных сообщений.

В качестве логина используется «root» (без кавычек), в качестве пароля значение переменной: `rabbitmq_users.root.password`.

Для обеспечения безопасности доступ к данному порту должен быть ограничен локальным сервером и внутренними (приватными) сетями с адресами по стандарту RFC1918. Категорически запрещается открывать доступ к нему из публичных сетей.

5 ИЗВЕСТНЫЕ ПРОБЛЕМЫ И СПОСОБЫ РЕШЕНИЯ

5.1 Проблема установки модуля python3-libselinux

Описание проблемы:

В некоторых случаях в процессе работы установки на ОС Centos, Redos возможно появление следующей ошибки:

```
      2023-01-01 12:00:00,001 p=28456 u=root n=ansible | fatal: [10.100.100.100]:
FAILED! => {"changed": false, "msg": "No package matching 'python3-libselinux'
found available, installed or updated",
      "rc": 126, "results": ["No package matching 'python3-libselinux' found
      available, installed or updated"]}
```

Решение:

Выполнить следующую команду и продолжить установку:

```
      sed -i 's@python3-libselinux@libselinux-python3@'\
      ./_versions/3.0/collections/ansible_collections/nct/system/roles/python3/va
rs/R{ED,edHat}.yml
```

5.2 Решение проблемы с логами

При остановке ротации (архивирования) логов сервисов Nginx или Pregen необходимо обновить политики безопасности на серверах с ролью `openresty-lb-core-wopi` и ролью `pregen`.

Обновления политики безопасности выполняются с помощью команды:

```
      restorecon -R /srv/docker
```

После обновления политики необходимо проверить ротацию логов через 48 часов.

Например:

```
[root@jenny ~]# cd /srv/docker/openresty/logs/
[root@jenny logs]# ls
access_full.log access_full.log-20231224-1703378461.gz access.log-20231222-
1703205421.gz  error.log error.log-20231224-1703378461.gz
access_full.log-20231221-1703118661.gz  access_full.log-20231225-1703464201
access.log-20231223-1703290201.gz  error.log-20231221-1703118661.gz
error.log-20231225-1703464201  access_full.log-20231222-1703205421.gz
access.log  access.log-20231224-1703378461.gz  error.log-20231222-
1703205421.gz  nginx.pid
access_full.log-20231223-1703290201.gz  access.log-20231221-1703118661.gz
access.log-20231225-1703464201 error.log-20231223-1703290201.gz
```

5.3 Переполнение диска данными мониторинга

Описание проблемы:

Быстрое заполнение диска при установке standalone или для кластерной установки, на узле кластера с ролью `co_infra`.

Решение:

Быстрое заполнение диска может происходить при поступлении большого количества данных мониторинга или логирования, из-за неправильно настроенных политик их хранения.

По умолчанию данные мониторинга располагаются в директории `/srv/docker/prometheus/data`. Время хранения данных задается при установке с помощью переменной `prometheus_storage_tsdb_retention_time` (по умолчанию "21d", то есть 21 день).

При переполнении диска данными мониторинга база данных Prometheus может быть повреждена. Для восстановления работоспособности необходимо удалить директорию `/srv/docker/prometheus/data`. После удаления директории следует переустановить роль, ограничив ее опцией `-limit`, только для роли `co_infra` и указав сценарий `playbooks/infra.yml`. Пример команды:

```
ansible-playbook -i playbooks/infra.yml --tags prometheus --limit co_infra
```

Объем данных журнала событий зависит от количества узлов кластера, количества их контейнеров и уровня протоколирования различных сервисов (настраиваются с помощью Etcd). По умолчанию данные журнала событий располагаются в директории `/srv/docker/elasticsearch/data`. Время хранения данных задается при установке с помощью переменной `es_index_retention_period_days` в файле `~/install_co/group_vars/co_setup/extra_vars.yml`. Значение по умолчанию `"120"`, что означает — 120 дней. В случае переполнения диска данными журнала событий, предусмотрено удаление более старые индексов вручную (структуры хранения и поиска данных в объеме 1 дня). Для этого на узле с ролью `co_infra` необходимо выполнить следующие команды:

```
# пароль вводить из переменной elasticsearch_opendistro_admin_password
curl -k --user admin https://localhost:9200/_cat/indices
# выбрать индексы, подлежащие удалению, начинающиеся с "co-"
curl -X DELETE -k --user admin https://localhost:9200/co-<YYYY.MM.DD>
```

Для уменьшения уровня логирования необходимо изменить значения переменных, приведенных в таблице 14.

Таблица 14 — Перечень переменных журнала мониторинга

Наименование переменной	Значение по умолчанию	Значение для уменьшения глубины лога
common_co_log_level	info	warn/error

Наименование переменной	Значение по умолчанию	Значение для уменьшения глубины лога
chatbot_log_level	info	warn/error
cvm_cu_log_level	info	warn/error
cvm_log_level	info	warn/error
dcm_du_log_level	info	warn/error
dcm_log_level	info	warn/error
du_log_level	info	warn/error
du_nps_log_level	info	warn/error
sdd_log_level	info	warn/error

Приложение А

Порядок установки и настройки локального репозитория

1. Создать каталог для размещения репозитория с помощью команды:

```
sudo mkdir -p /srv/repo/alse/main
```

2. Примонтировать образ установочного диска (если на компьютере нет каталога /media/cdrom — то создать каталог /media/cdrom) с помощью команды:

```
[ -d /media/cdrom ] || sudo mkdir /media/cdrom
sudo mount /путь_к_ISO-образу /media/cdrom
```

3. Скопировать файлы из образа в каталог репозитория с помощью команды:

```
sudo cp -a /media/cdrom/* /srv/repo/alse/main
```

4. Отмонтировать ISO-образ диска с помощью команды:

```
sudo umount /media/cdrom
```

4.1 Если требуется, выполнить аналогичные действия для базового репозитория (диска со средствами разработки) с помощью команды:

```
sudo mkdir -p /srv/repo/alse/base
[ -d /media/cdrom ] || sudo mkdir /media/cdrom
sudo mount /путь_к_ISO-образу /media/cdrom
sudo cp -a /media/cdrom/* /srv/repo/alse/base
sudo umount /media/cdrom
```

5. Для обновления основного репозитория (основного диска) с помощью команды:

```
sudo mkdir -p /srv/repo/alse/update-main
[ -d /media/cdrom ] || sudo mkdir /media/cdrom
sudo mount /путь_к_ISO-образу /media/cdrom
sudo cp -a /media/cdrom/* /srv/repo/alse/update-main
sudo umount /media/cdrom
```

6. Для обновления базового репозитория (диска с обновлением средств разработки) с помощью команды:

```
sudo mkdir -p /srv/repo/alse/update-base
[ -d /media/cdrom ] || sudo mkdir /media/cdrom
sudo mount /путь_к_ISO-образу /media/cdrom
sudo cp -a /media/cdrom/* /srv/repo/alse/update-base
sudo umount /media/cdrom
```

Приложение Б

Замена стандартного репозитория на локальный

Замена стандартного репозитория на локальный выполняется на сервере с ролью `operator`. Перечисленный порядок действий используется в ОС Astra. Для замены репозитория необходимо:

1. Отключить внешние репозитории, запустив команду:

```
sed -i "s/^/#/" /etc/apt/sources.list
```

2. Добавить локальный внешний репозиторий, запустив команду:

```
tee -a /etc/apt/sources.list << EOF
deb http://$IP_ADDRESS:8081/repository/astra/ 1.7_x86-64 \
main contrib non-free
deb http://$IP_ADDRESS:8081/repository/astra-ext/ 1.7_x86-64 \
main contrib non-free
EOF
```

где `$IP_ADDRESS` — IP-адрес локального сервера для хранения файлов.

3. Обновить индекс репозитория, запустив команду:

```
apt update
```

4. Проверить доступность репозитория (произвести поиск произвольного пакета), запустив команду:

```
apt search pwgen
```

5. Убедиться, что в выводе команды присутствует название пакета pwgen. Вывод команды:

```
root@operator:~# apt search pwgen
Sorting... Done
Full Text Search... Done
pwgen/stable 2.08-1 amd64
Automatic Password generation
root@operator:~#
```

6. Настроить менеджер модулей (pip) на использование локального репозитория, запустив команду:

```
tee /etc/pip.conf << EOF
[global]
trusted-host = $IP_ADDRESS
index = http://$IP_ADDRESS:8081/repository/pypi-proxy/pypi
index-url = http://$IP_ADRESS:8081/repository/pypi-proxy/simple
EOF
```

где `$IP_ADDRESS` — IP-адрес локального сервера для хранения файлов.

Приложение В

Настройка сетевых соединений

Пример настройки сетевого соединения с помощью командной строки в ОС Astra.

1. Для проверки необходимо открыть файл с сетевыми настройками с помощью команды:

```
nano /etc/network/interfaces
```

В открывшемся окне редактора проверить наличие следующей строки:

```
# This file describes the network interfaces available on your system
# and how to activate them. For more information, see interfaces(5).

source /etc/network/interfaces.d/*
```

1.1 Закрыть окно и вернуться к строке терминала.

1.2 Создать новое соединение с помощью команды:

```
sudo nano /etc/network/interfaces.d/eth0
```

Примечание: если на вашем сервере установлены другие редакторы (vim, vi) замените в команде `nano` на другой редактор.

2. В открывшемся окне редактора в зависимости от типа используемого для настроек ввести команду из пункта 2.1 или 2.2.

2.1 При использовании статического IP-адреса необходимо ввести:

```
echo "auto eth0
iface eth0 inet static
address 192.168.1.100
netmask 255.255.255.0
gateway 192.168.1.1" > /etc/network/interfaces.d/eth0
```

В примере используются произвольные настройки сетевого соединения. Необходимо заменить предложенные настройки (192.168.1.100, 255.255.255.0, 192.168.1.1) на настройки сетевого окружения созданных серверов.

2.2 При использовании DHCP в окне редактора необходимо ввести:

```
echo "auto eth0
iface eth0 inet dhcp" > /etc/network/interfaces.d/eth0
```

Для корректной работы необходимо закрепить IP-адреса за серверами с помощью настроек DHCP-сервера вашего шлюза (коммутатора).

3. После ввода переменных файл сохранить. Повторно открыть файл командой из пункта 1 для проверки.

4. Задать DNS-сервер

```
echo "nameserver 8.8.4.4" > /etc/resolv.conf
```

Адрес DNS-сервера 8.8.4.4 указан произвольно, если в локальной сети существует внутренний DNS-сервер, необходимо изменить адрес 8.8.4.4.

5. Применить настройки сетевого соединения

```
sudo systemctl restart networking
```

Повторить выполнение действия для каждого сервера, используемого для установки.

Приложение Г

Порядок создания самоподписанного сертификата

По умолчанию браузеры не доверяют самоподписанным сертификатам, рекомендуется использовать его только для внутренних целей или в целях тестирования.

1. Проверка или установка OpenSSL.

OpenSSL доступен по умолчанию во всех основных дистрибутивах Linux.

Для поиска установленного ПО OpenSSL и проверки версии необходимо выполнить команду:

```
$ openssl version
```

Если вывод с информацией о версии OpenSSL отсутствует — программа не установлена.

Для установки OpenSSL выполните следующую команду:

```
$ sudo dnf install openssl
```

или

```
$ sudo yum install openssl
```

Выбор команды зависит от типа ОС.

2. Создание SSL-сертификата.

Для создания самоподписанного сертификата SSL необходимо использовать следующую команду:

```
$ openssl req -x509 -nodes -days 365 -newkey rsa:4096 -keyout
server.nopass.key -out server.crt
```

С помощью команды будет создан самоподписанный сертификат, который будет действителен в течение 365 дней.

По умолчанию сертификат и файл ключа будут созданы в текущем каталоге (в каталоге, из которого выполняется команда).

Описание флагов использованных в команде приведено в таблице 15.

Таблица 15 — Значения флагов команды

Флаг	Описание
req	Выполнить запрос на подпись сертификата
-newkey rsa:4096	Создать ключ RSA длиной 4096 бит. Если не указано иное, по умолчанию будет создан ключ длиной 2048 бит
-keyout	Указать имя файла для хранения закрытого ключа
-out	Указать имя файла для хранения нового сертификата
-nodes	Пропустить шаг по созданию сертификата с парольной фразой
-x509	Создать сертификат формата X.509
-days	Указать время действия сертификата в днях

Описание полей при создании сертификата приведено в таблице 16.

Таблица 16 — Значения полей CSR

Поле	Описание
C =	Название страны (двухбуквенный код)
ST =	Название штата или провинции
L =	Название населенного пункта
O =	Полное название вашей организации
OU =	Название организационной единицы
CN =	Полное доменное имя

3. Создание закрытого ключа.

Закрытый ключ необходим для подписи вашего SSL-сертификата. Для создания и сохрания закрытого ключа необходимо выполнить команду:

```
$ openssl genrsa -out server.nopass.key
```

Значения флагов команды:

– `genrsa` — создать закрытый ключ RSA;

– `-out` — выходной файл.

По умолчанию закрытый ключ будет храниться в текущем каталоге (в каталоге, из которого выполняется команда).

4. Создание запроса на подпись сертификата (CSR).

CSR — информация, отправляемая в удостоверяющий центр. Для создания CSR необходимо выполнить следующую команду:

```
$ openssl req -new -key server.nopass.key -out server.csr
```

Описание флагов использованных в команде приведено в таблице 17.

Таблица 17 — Значения флагов команды

Флаг	Описание
req	Запрос на подпись сертификата
-new	Новый запрос
-key	Путь, где хранится ваш файл закрытого ключа
-out	Имя выходного файла

После запуска команды, представленной ниже, будет создан самоподписаный сертификат, который будет действителен в течение 365 дней.

```
$ openssl x509 -req -days 365 -in server.csr -signkey server.nopass.key \
-out server.crt
```

5. Проверка деталей сертификата выполняется с помощью команды:

```
$ openssl x509 -text -noout -in server.crt
```

Приложение Д

Перечень изменений в документе

В данном приложении представлен перечень изменений относительно даты публикации и версии документа.

18.06.2024 Подготовлен документ версии 1.

10.07.2024 Подготовлен документ версии 2 со следующими изменениями:

– добавлен раздел «Конфигурация CentOS»;
– в раздел «Конфигурирование файла main.yml» добавлена таблица «Дополнительные переменные конфигурации».