Astra Wine

Руководство по установке

2026-06-11

Оглавление

Системные требования . . . 2
Подготовка . . . 5
- Генерация ключей SSH . . . 5
- Настройка устройств пользователей . . . 6
- Загрузка установщика . . . 11

Развёртывание на одном узле . . . 12
- Установка . . . 12
- Обновление . . . 15
  - Подготовка к обновлению . . . 15
  - Обновление . . . 16
- Удаление платформы . . . 17

Развёртывание в кластере Kubernetes . . . 20
- Установка . . . 20

Справочник . . . 23
- Параметры развёртывания . . . 23
  - Развёртывание на одном узле . . . 23
  - Развёртывание в кластере Kubernetes . . . 34

Системные требования

Общие требования

Сетевые порты

Для корректной работы Astra Wine в параметрах сети разрешите прохождение TCP-трафика через указанные порты.

Таблица 1. Порты сервера

<table>
<tr><th rowspan="2">Сервис</th><th colspan="2">Порт</th></tr>
<tr><th>Развёртывание на одном узле</th><th>Развёртывание в кластере Kubernetes</th></tr>
<tr><td>Веб-сервер NGINX</td><td>443</td><td rowspan="8">443</td></tr>
<tr><td>Панель мониторинга Grafana</td><td>3000</td></tr>
<tr><td>Сервер сообщений Centrifugo</td><td>8000</td></tr>
<tr><td>Сервер авторизации Keycloak</td><td>8443</td></tr>
<tr><td>Semaphore</td><td>8081</td></tr>
<tr><td rowspan="2">S3-совместимое хранилище MinIO</td><td>9000</td></tr>
<tr><td>9001</td></tr>
<tr><td>Служба мониторинга Prometheus</td><td>9090</td></tr>
</table>

Таблица 2. Порты рабочих станций

Сервис	Порт	Направление трафика
Сервер SSH	22	Входящий
Подключение к серверу	443	Исходящий

Рабочие станции

Оборудование рабочих станций должно иметь характеристики не ниже указанных.

Таблица 3. Минимальные технические характеристики рабочей станции

Параметр	Значение, не ниже
Количество ядер CPU	10
Тактовая частота CPU, ГГц	2,1[1]
Оперативная память (DDR4 или новее), ГБ	16
Свободное дисковое пространство в корневом разделе, ГБ	50

На рабочие станции пользователей должна быть установлена одна из версий ОС, указанных в таблице.

Таблица 4. Требования к ОС рабочих станций

Релиз ОС	Версия, не ниже	Режим защищенности
Astra Linux Special Edition 1.8.x	1.8.1	Базовый («Орёл») или Усиленный («Воронеж»)
Astra Linux Special Edition 1.7.x	1.7.3	Базовый («Орёл») или Усиленный («Воронеж»)

Также должны быть подключены основной (main)[2] и расширенный (extended) репозитории используемой версии Astra Linux Special Edition или их зеркала. Например, для Astra Linux Special Edition 1.8.1.UU2 содержимое файла `/etc/apt/sources.list`, использующего репозитории ПАО Группа Астра, должно иметь следующий вид:

```
deb https://download.astralinux.ru/astra/frozen/1.8_x86-64/1.8.1/uu/2/repository-main
1.8_x86-64 main contrib non-free non-free-firmware
deb https://download.astralinux.ru/astra/frozen/1.8_x86-64/1.8.1/uu/2/repository-
extended 1.8_x86-64 main contrib non-free non-free-firmware
```

Также необходимо обеспечить возможность загрузки из Интернета или с собственного сервера дистрибутивов Windows-приложений, которые требуются для корректной работы префиксов WINE.

Рабочие места администратора и пользователей

Для работы с пользовательским интерфейсом необходимо наличие на рабочих станциях администратора и пользователей одного из следующих веб-браузеров:

Таблица 5. Рекомендуемые веб-браузеры

Браузер	Версия, не ниже
Chromium	108.0.5359.71
Mozilla Firefox	102
Яндекс Браузер	23.7.2.767

Требования для развёртывания на одном узле

Для корректной работы сервера Astra Wine рекомендуется использовать для развёртывания узел с характеристиками не ниже указанных.

Таблица 6. Рекомендуемые технические характеристики узла

Параметр	Значение, не ниже
Количество ядер CPU	12

Параметр	Значение, не ниже
Тактовая частота CPU, ГГц	2
Оперативная память, ГБ	16
Свободное дисковое пространство, ГБ	1000
Тип дискового хранилища	SSD
Количество сетевых адаптеров	2
Скорость сетевого подключения, ГБит/с	10

Для работы сервера необходим процессор Intel или AMD с архитектурой x86_64, имеющий поддержку набора инструкций SSE4.2. Процессоры с архитектурой ARM не поддерживаются.

Таблица 7. Требования к ОС узла

Релиз ОС	Версия, не ниже
Astra Linux Special Edition 1.8.x	1.8.1
Astra Linux Special Edition 1.7.x	1.7.3

Требования для развёртывания в кластере Kubernetes

Эта часть документации находится в стадии разработки.

[1] Рекомендуется использовать процессор Intel Core i7 11, 12 или 13 поколения.

[2] В Astra Linux Special Edition 1.7.x — базовый (base).

Подготовка

Для выполнения некоторых задач платформа Astra Wine использует возможности системы автоматизации Ansible.

В Ansible *управляющим* называется узел, на котором запускаются задания автоматизации, а настраиваемые с их помощью узлы — *управляемыми*. В Astra Wine роль управляющего узла играет сервер платформы, а в роли управляемых узлов выступают устройства пользователей.

Ansible позволяет подключаться к управляемым узлам различными способами. В Astra Wine для подключения используется протокол SSH с авторизацией по ключу.

Исходя из этого для подготовки инфраструктуры к установке Astra Wine выполните следующие действия:

1. На сервере создайте основную пару ключей SSH, которая будет использоваться для доступа к управляемым узлам.
2. Настройте устройства пользователей (управляемые узлы):
    - Создайте учётную запись для Ansible.
    - Настройте использование `sudo` без ввода пароля.
    - Настройте сервер OpenSSH.
    - Разместите публичный ключ SSH.
    - Отключите блокировку `ptrace`.
3. Загрузите архив с установщиком Astra Wine.

Генерация ключей SSH

Для создания пары ключей SSH используйте утилиту `ssh-keygen`, например:

```
ssh-keygen -t ed25519 -f ~/.ssh/astra-wine -N "" -C "Astra Wine Key"
```

Здесь:

- `-t ed25519` — алгоритм шифрования, на основе которого генерируется ключ.
- `-f ~/.ssh/astra-wine` — имя файла приватного ключа. Публичный ключ будет создан в том же каталоге под именем `astra-wine.pub`.
- `-N ""` — создание приватного ключа без парольной защиты.
- `-C "Astra Wine Key"` — необязательный комментарий к ключу.

Настройка устройств пользователей

Для успешной автоматизации управления устройствами с помощью Ansible требуется настройка специальных учётных записей, прав доступа, SSH-подключения и других параметров системы. В разделах ниже представлены руководства для настройки пользовательских устройств для работы с платформой Astra Wine.

Учётная запись для Ansible

Для управления устройствами с помощью Ansible необходима служебная учётная запись с определёнными правами доступа.

Создание учётной записи Ansible

Для создания служебной учётной записи Ansible выполните следующие действия:

1. Создайте учётную запись пользователя:

   ```
   sudo useradd -m <ansible_user>
   ```

   Название учётной записи должно быть одним и тем же на всех устройствах.

2. Задайте пароль:

   ```
   sudo passwd <ansible_user>
   ```

3. Установите высокий уровень целостности:

   ```
   sudo pdpl-user -i 63 <username>
   ```

   Чтобы проверить установленный уровень целостности выполните команду:

   ```
   pdpl-user <username>
   ```

Права доступа

Для выполнения служебных операций необходимо обеспечить доступ служебной учётной записи Ansible к системным файлам и каталогам, указанным в таблице ниже.

Каталог	Права	Назначение
`/opt/`	rw	Создание и использование каталогов для установки Windows-приложений.
`/lib/systemd/system/`	rw	Используется механизмом автообновления агента.

Каталог	Права	Назначение
Корневой каталог домашних каталогов пользователей (обычно `/home/`)	rw	Получение списка пользователей, установка и удаление Windows-приложений в пользовательских окружениях.
`/tmp/`	rw	Загрузка временных файлов и префиксов WINE, используемых при установке и обновлении.
`/home/<ansible_user>/agent/`	rwx	Рабочий каталог агента. Используется для хранения файлов агента и выполнения операций.
`/etc/wineagent/`	rwx	Каталог с конфигурационными файлами агента.
`/usr/local/bin/`	rwx	Размещение и обновление бинарного файла агента.

Использование `sudo` без ввода пароля

Чтобы выключить запрос пароля при использовании `sudo`, внесите соответствующие изменения в файл `/etc/sudoers` с использованием `visudo`.

`visudo` — безопасный способ правки файла `/etc/sudoers`, поскольку:

- проверяет синтаксис перед сохранением;
- предотвращает одновременное редактирование;
- не допускает сохранения ошибочных конфигураций, которые могут лишить пользователей доступа к `sudo`.

1. Запустите `visudo` с повышенными привилегиями:

   ```
   sudo visudo
   ```

   Система запросит ваш пользовательский пароль.

2. Добавьте в конец файла следующую строку:

   ```
   <ansible_user> ALL=(ALL:ALL) NOPASSWD: ALL
   ```

   Здесь <ansible_user> — название созданной служебной учётной записи Ansible.

3. Сохраните изменения и закройте файл.

Установка и настройка сервера OpenSSH

Установите и запустите сервер OpenSSH:

1. Убедитесь, что конфигурационный файл `/etc/apt/sources.list` содержит ссылку на основной (main) репозиторий используемой версии Astra Linux Special Edition или его

зеркало.

Например, для Astra Linux Special Edition 1.8.1.UU2 содержимое файла `/etc/apt/sources.list`, использующего репозиторий ПАО Группа Астра, должно иметь следующий вид:

```
deb http://download.astralinux.ru/astra/frozen/1.8_x86-64/1.8.1/uu/2/repository-
main     1.8_x86-64 main contrib non-free non-free-firmware
```

2. Обновите кеш APT:

```
sudo apt-get update
```

3. Установите пакеты сервера OpenSSH:

```
sudo apt-get install openssh-server --yes
```

4. Включите автоматический запуск службы сервера OpenSSH при загрузке ОС:

```
sudo systemctl enable ssh.service
```

5. Запустите службу сервера OpenSSH:

```
sudo systemctl start ssh.service
```

Размещение публичных ключей SSH

Чтобы к узлу можно было подключиться по SSH, используя ключ:

1. Авторизуйтесь на узле от имени созданного ранее пользователя.
2. Если в домашнем каталоге пользователя нет подкаталога `.ssh/`, создайте его:

```
mkdir ~/.ssh/
```

3. Если в каталоге `~/.ssh/` не существует файл `authorized_keys`, создайте его:

```
touch ~/.ssh/authorized_keys
```

4. Добавьте в файл `~/.ssh/authorized_keys` содержимое созданного ранее файла публичного ключа SSH, например, с помощью команды:

```
cat /path/to/key.pub | tee -a ~/.ssh/authorized_keys
```

Здесь `/path/to/key.pub` — путь к файлу публичного ключа SSH.

5. Установите на файл `~/.ssh/authorized_keys` режим доступа, разрешающий чтение и запись только пользователю-владельцу и запрещающий доступ всем остальным пользователям и группам:

```
chmod 0600 ~/.ssh/authorized_keys
```

Если узлы разрешают подключение по SSH с использованием пароля, публичную часть ключа можно разместить на устройствах пользователей с помощью утилиты `ssh-copy-id`:

```
ssh-copy-id -i /path/to/key.pub <user>@<host>
```

Здесь:

- `/path/to/key.pub` — путь к файлу публичного ключа;
- `<user>` — название учётной записи пользователя-администратора узла;
- `<host>` — IP-адрес или FQDN узла.

Отключение блокировки `ptrace`

Дистрибутивы некоторых Windows-приложений при настройке префикса WINE используют возможности системной утилиты `ptrace`. Если в Astra Linux Special Edition включена блокировка `ptrace`, некоторые Windows-приложения и их дистрибутивы не смогут быть корректно настроены для последующей работы.

За блокировку возможностей `ptrace` отвечает модуль ядра `yama`. Этот модуль не поддерживает настройку исключений, поэтому для корректной работы платформы Astra Wine необходимо выключить блокировку `ptrace` на уровне ОС.

Чтобы узнать текущий статус блокировки, выполните команду:

```
astra-ptrace-lock status
```

Значение `АКТИВНА` говорит о том, что блокировка `ptrace` включена.

Выключить блокировку `ptrace` можно несколькими способами:

- С помощью утилиты `astra-ptrace-lock` (рекомендуется).
- Изменение параметров ядра.

- Настройка файла ответов.

Используйте этот способ, если установка ОС на устройства пользователей выполняется в автоматическом режиме.

Использование утилиты `astra-ptrace-lock`

1. Выполните команду:

```
sudo astra-ptrace-lock disable
```

2. Перезагрузите устройство.

Изменение параметров ядра

1. В конфигурационном файле `/etc/sysctl.d/999-astra.conf` для параметра `kernel.yaml.ptrace_scope` задайте значение `0`:

```
kernel.yama.ptrace_scope=0
```

2. Отключите автоматический запуск службы `astra-ptrace-lock.service` при загрузке ОС:

```
sudo systemctl astra-ptrace-lock.service disable
```

3. Перезагрузите устройство.

Настройка файла ответов

Astra Linux Special Edition 1.7 использует для автоматической установки утилиту `debian-installer`. Файл ответов называется `preseed.cfg`. Порядок его создания описан в статье Справочного центра Автоматическая установка Astra Linux Special Edition 1.7.

Пример фрагмента файла ответов с отключением блокировки `ptrace`:

`preseed.cfg`

```
d-i astra-additional-setup/additional-settings-smolensk multiselect Disable ptrace
capability
```

Astra Linux Special Edition 1.8 использует для автоматической установки утилиту `astra-installer`. Файл ответов называется `astra-installer-preseed.yaml`. Порядок его создания описан в статье Справочного центра Автоматическая установка Astra Linux Special Edition x.8. Там же приводятся инструкции по преобразованию файла `preseed.cfg` в `astra-installer-preseed.yaml`.

Убедитесь, что в `astra-installer-preseed.yaml` закомментированы или удалены строки,

отвечающие за включение блокировки `ptrace`:

`astra-installer-preseed.yaml`

```
---
# ...
features:
  - astra-ptrace-lock
```

Загрузка установщика

1. Авторизуйтесь в Личном кабинете.
2. Загрузите архив с установщиком Astra Wine в соответствии с выбранным вариантом развёртывания: на одном узле или в кластере Kubernetes.
3. Загрузите каталог Windows-приложений.

   Для загрузки доступны только те приложения и версии, которые входят в приобретённую лицензию.

Файлы доступны для загрузки при наличии действующей лицензии на продукт.

Развёртывание на одном узле

Развёртывание Astra Wine на одном узле (standalone) представляет собой конфигурацию, при которой все компоненты платформы устанавливаются и работают на одном физическом или виртуальном сервере.

Эта архитектура отличается простотой настройки и минимальными требованиями к инфраструктуре. Она подходит для случаев, когда необходима простая и быстрая установка без сложной инфраструктуры кластера.

Развёртывание Astra Wine на одном узле предоставляет следующие преимущества:

- простота установки и управления;
- минимальные требования к ресурсам;
- быстрое развёртывание системы;
- отсутствие необходимости в поддержке инфраструктуры Kubernetes.

Установка

Сервер Astra Wine может быть развёрнут на одном узле. Такая конфигурация не обеспечивает отказоустойчивости, но проста в настройке и не требовательна к ресурсам.

Развёртывание

Для развёртывания сервера Astra Wine:

1. Убедитесь, что конфигурационный файл `/etc/apt/sources.list` содержит ссылки на основной (main)[1] и расширенный (extended) репозитории используемой версии Astra Linux Special Edition или их зеркала.

   Например, для Astra Linux Special Edition 1.8.1.UU2 содержимое файла `/etc/apt/sources.list`, использующего репозитории ПАО Группа Астра, должно иметь следующий вид:

   ```
   deb https://download.astralinux.ru/astra/frozen/1.8_x86-64/1.8.1/uu/2/repository-main     1.8_x86-64 main contrib non-free non-free-firmware
   deb https://download.astralinux.ru/astra/frozen/1.8_x86-64/1.8.1/uu/2/repository-extended 1.8_x86-64 main contrib non-free non-free-firmware
   ```

2. Обновите кеш APT:

   ```
   sudo apt update
   ```

3. Установите необходимые пакеты:

```
sudo apt install docker docker.io docker-compose-v2 --yes
```

4. Перейдите в каталог, в который загрузили архив с установщиком Astra Wine, например:

```
cd ~/Загрузки/
```

5. Распакуйте архив с установщиком во временный каталог:

```
unzip astrawine_standalone_<version>.zip -d /tmp/
```

6. Подготовьте и добавьте в каталог `/tmp/astrawine_standalone/self-hosted/certs` собственные SSL-сертификат и ключ со следующими названиями:

   - `private.crt` — сертификат сервера;
   - `public.key` — закрытый ключ сервера.

   Убедитесь, что сертификаты доверены на следующих устройствах:

   - с которых будет осуществляться подключение к веб-интерфейсу сервера Astra Wine;
   - на которых будет установлен агент.

   Если при развёртывании платформы указанные файлы отсутствуют, они будут сгенерированы автоматически. При этом используется самоподписанный SSL-сертификат, который не доверен по умолчанию.

7. Перейдите в каталог установщика:

```
cd /tmp/astrawine_standalone/self-hosted/
```

8. В конфигурационном файле `environ/.env` в значении параметра `ALLOWED_HOSTS` укажите одной строкой через запятую список IP-адресов, с которых должно быть разрешено подключение к веб-интерфейсу сервера Astra Wine.

   При значении `0.0.0.0` будет разрешено подключение с любого IP-адреса.

9. Запустите сценарий установки:

```
sudo bash ./install.sh
```

Установщик выведет в терминал текст лицензионного соглашения.

1. Прочтите лицензионное соглашение.

Для прокрутки его текста используйте клавиши со стрелками и пробел. Когда вы дочитаете соглашение до конца, установщик выведет запрос:

```
Принимаете лицензионное соглашение? (y/N)
```

1. Чтобы принять лицензионное соглашение, нажмите `y`.

Установка будет запущена. Дождитесь загрузки образов и вывода запроса на ввод названия учётной записи администратора:

```
Введите username администратора:
```

1. Введите название учётной записи администратора и нажмите `Enter`.

Требования к названию учётной записи администратора

- Можно использовать буквы латинского алфавита, цифры, символы `@`, `.`, `-` и `_`.
- Нельзя использовать название `root`.

Если оставить поле пустым, будет использовано значение по умолчанию — `admin`.

Если название учётной записи указано верно, в терминал будет выведен запрос на ввод пароля администратора:

```
Введите password администратора:
```

1. Введите пароль администратора и нажмите `Enter`.

   Если оставить поле пустым, будет использовано значение по умолчанию — `admin`.

2. Дождитесь возврата терминала к приглашению ко вводу команд.

Проверка корректности развёртывания

Чтобы проверить корректность развёртывания и работоспособности сервера Astra Wine, запустите браузер и перейдите по ссылкам, предоставляющим доступ к веб-интерфейсам служб. Если в конфигурационном файле `environ/.env` вы указали для настройки `ALLOWED_HOSTS` собственное значение, в таблице ниже вместо `127.0.0.1` подставьте собственный IP-адрес.

Таблица 8. Ссылки, службы и учётные данные по умолчанию

Ссылка	Служба	Учётная запись	Пароль
127.0.0.1	Пользовательская часть веб-интерфейса платформы	`admin`	`admin`

Ссылка	Служба	Учётная запись	Пароль
127.0.0.1/admin/	Административная часть веб-интерфейса платформы	`root`	`root`
127.0.0.1:8443	Keycloak	`admin`	`admin`
127.0.0.1:9000	S3-совместимое хранилище MinIO	`minioadmin`	`minioadmin`

Если страницы открываются, а авторизация проходит успешно, развёртывание платформы Astra Wine также считается успешным.

Обновление

Обновление платформы Astra Wine состоит из двух основных этапов:

1. Подготовка среды.

   Создание резервной копии данных и размещение новых образов контейнеров в каталоге `dockerimages/`. Этот этап гарантирует, что при возникновении ошибок будет возможен возврат системы к прежнему состоянию, а обновление будет использовать корректные версии образов.

2. Развёртывание обновления.

   Этот этап включает остановку текущих контейнеров, очистку старых и загрузку новых образов и применение миграций базы данных на существующих томах. Такой подход позволяет обновить систему без потери пользовательских данных.

Подготовка к обновлению

Этот этап обеспечивает сохранность данных и готовит файловую структуру к использованию новых контейнеров.

Для подготовки системы к обновлению выполните следующие шаги:

1. Загрузите обновленную версию установщика Astra Wine.
2. Перейдите в каталог, в котором развёрнута платформа, например:

   ```
   cd ~/astrawine_standalone/self-hosted/
   ```

3. Переименуйте каталог с текущими образами:

   ```
   mv dockerimages dockerimages_old
   ```

   Это позволит сохранить старую версию образов на случай необходимости отката.

4. В каталоге `wine_data/` создайте подкаталог `backup/`, если он ещё не существует:

   ```
   mkdir -p wine_data/backup
   ```

5. Выполните резервное копирование базы данных:

   ```
   sudo bash ./backup_db.sh
   ```

   Результатом будет формирование дампа БД, который сохранится в каталоге `wine_data/backup/`.

6. Создайте каталог `dockerimages` для копирования в него новой версии образов:

   ```
   mkdir dockerimages
   ```

7. Скопируйте содержимое каталога `dockerimages` новой версии платформы Astra Wine в созданный каталог:

   ```
   cp ~/Загрузки/astrawine_standalone/self-hosted/dockerimages/* dockerimages/
   ```

   Эти образы будут использованы при обновлении контейнеров.

Обновление

На этом этапе происходит остановка текущих контейнеров, удаление старых образов и запуск обновлённой версии платформы Astra Wine с сохранением данных.

Для обновления платформы Astra Wine выполните следующие шаги:

1. Остановите все контейнеры платформы:

   ```
   docker compose -p astrawine down
   ```

2. Удалите все образы Docker, не связанные с контейнерами:

   ```
   docker image prune -fa
   ```

   Если в системе есть другие образы кроме Astra Wine, удалите последние вручную:

   ```
   docker image rm <IMAGE_ID>
   ```

   Здесь IMAGE_ID — идентификатор контейнера Astra Wine.

3. Загрузите новые контейнеры из каталога `dockerimages/`:

```
sudo bash ./load_container.sh
```

4. Запустите обновлённые контейнеры, используя существующие тома:

```
docker compose -p astrawine up -d
```

5. Подготовьте миграции БД:

```
docker compose -p astrawine exec app python3 manage.py makemigrations
```

6. Выполните миграции:

```
docker compose -p astrawine exec app python3 manage.py migrate
```

После успешного выполнения миграций и корректного запуска контейнеров обновление платформы считается завершённым.

Удаление платформы

Для полного удаления платформы Astra Wine с сервера выполните последовательные шаги, обеспечивающие корректное удаление всех компонентов: префиксов, агентов и самой платформы.

1. Удаление префиксов WINE

Удалите все префиксы, установленные через Astra Wine. Для этого выполните следующие действия:

1. Перейдите в графический интерфейс Astra Wine.
2. На панели навигации выберите **Задачи** › **Список задач**.
3. На панели инструментов нажмите кнопку **[+]**.
4. Заполните форму:
   - **Тип задачи** — выберите **Удалить приложение на устройстве**;
   - **Версия приложения** — выберите префикс WINE, который необходимо удалить;
   - **Агент** — выберите агент, с устройства которого необходимо удалить префикс WINE;
   - **Запустить сразу** — активируйте этот параметр, чтобы запустить задачу сразу после её создания.
5. Нажмите кнопку **[Сохранить]**.

Задача на удаление префикса WINE будет создана и добавлена в очередь. Для отслеживания статуса её выполнения перейдите в раздел Список задач.

6. Убедитесь, что задача завершилась успешно и все пакеты удалены.
7. Повторите пункты 1-6 для всех пользователей и установленных им префиксов WINE.

Для ускорения удаления префиксов WINE можно использовать создание задачи на группу пользователей

2. Удаление агентов

Удалите все агенты. Для этого выполните следующие действия:

1. Перейдите в графический интерфейс Astra Wine.
2. На панели навигации выберите **Задачи** › **Список задач**.
3. На панели инструментов нажмите кнопку **[+]**.
4. Заполните форму:
   - **Тип задачи** — выберите **Удалить агента**;
   - **Тип агента** — выберите тип агента, который необходимо удалить;
   - **Пользователь** — выберите пользователя, с устройства которого необходимо удалить агента;
   - **Агент** — выберите агент, который необходимо удалить с устройства пользователя;
   - **Запустить сразу** — активируйте этот параметр, чтобы запустить задачу сразу после её создания.
5. Нажмите кнопку **[Сохранить]**.

   Задача на удаление агента будет создана и добавлена в очередь. Для отслеживания статуса её выполнения перейдите в раздел Список задач.
6. Убедитесь, что задача завершилась успешно и агент удален.
7. Повторите пункты 1-6 для всех пользователей и установленных им агентов.

Для ускорения удаления агентов можно использовать создание задачи на группу пользователей.

3. Удаление платформы

Для удаления платформы Astra Wine с сервера выполните следующие действия:

1. Остановите контейнеры:

```
docker compose -p astrawine down -v
```

2. Удалите Astra Wine:

```
docker system prune -af
```

Эти команды:

1. Останавливает и удаляет все контейнеры платформы.
2. Удаляет связанные тома данных.
3. Полностью очищает окружение Astra Wine с сервера.

После выполнения указанных шагов платформа Astra Wine будет полностью удалена из системы.

[1] В Astra Linux Special Edition 1.7.x — базовый (base).

Развёртывание в кластере Kubernetes

Развёртывание Astra Wine в кластере Kubernetes представляет собой конфигурацию, при которой компоненты платформы устанавливаются и работают в составе кластера Kubernetes.

Данная архитектура ориентирована на промышленную эксплуатацию и позволяет использовать возможности Kubernetes для обеспечения масштабируемости, отказоустойчивости и гибкого управления ресурсами. Компоненты платформы развёртываются в виде контейнеров и управляются средствами оркестратора.

Развёртывание Astra Wine в Kubernetes предоставляет следующие преимущества:

- масштабируемость компонентов платформы;
- отказоустойчивость на уровне инфраструктуры;
- гибкое управление ресурсами и изоляция компонентов;
- удобство обновления и сопровождения;
- интеграция с существующей инфраструктурой Kubernetes.

Установка

Платформа Astra Wine может быть развёрнута в кластере Kubernetes. Данный вариант развёртывания предназначен для промышленной эксплуатации и обеспечивает масштабируемость и отказоустойчивость на уровне инфраструктуры.

Развёртывание платформы не включает установку и настройку самого кластера Kubernetes. Необходимо самостоятельно подготовить кластер и обеспечить его соответствие системным требованиям.

Развёртывание

Развёртывание Astra Wine в Kubernetes выполняется с использованием Helm.

Перед началом установки убедитесь, что:

- кластер Kubernetes доступен и настроен;
- утилита `kubectl` настроена на работу с целевым кластером;
- утилита `helm` установлена на рабочей станции администратора;
- имеется архив с Helm-чартом платформы Astra Wine.

При использовании закрытого контура необходимо заранее загрузить образы контейнеров платформы Astra Wine в собственный контейнерный реестр и указать его адрес в конфигурации Helm-чарта.

Для развёртывания сервера Astra Wine в кластере Kubernetes:

1. Создайте каталог для распаковки Helm-чарта.
2. Распакуйте архив с Helm-чартом платформы в созданный каталог:

```
tar -xvf astrawine_helm_<version>.tar.gz -C /path/to/directory/
```

3. Скорректируйте параметры развёртывания в файлах `values.yaml` и `secrets.example.yaml` под особенности вашей инфраструктуры.

Описания переменных приведены в справочнике.

4. Выполните команду для установки платформы:

```
helm install astrawine ./ \
  --namespace astrawine \
  --create-namespace \
  -f values.yaml \
  -f secrets.example.yaml
```

5. Дождитесь возврата терминала к приглашению ко вводу команд.

   В среднем процесс установки занимает 5 минут.

Проверка корректности развёртывания

Чтобы убедиться в успешной установке платформы:

1. Проверьте статус Helm-релиза:

```
helm list -n astrawine
```

2. Убедитесь, что все поды находятся в состоянии `Running` или `Completed`:

```
kubectl get pods -n astrawine
```

3. Проверьте доступность веб-интерфейсов платформы по доменным именам, указанным в параметре `baseDomain`, например:

   Таблица 9. Ссылки, службы и учётные данные по умолчанию

Ссылка	Служба	Учётная запись	Пароль
<baseDomain>	Пользовательская часть веб-интерфейса платформы	admin	admin

Ссылка	Служба	Учётная запись	Пароль
<baseDomain>/admin/	Административная часть веб-интерфейса платформы	`root`	`root`
keycloack.<baseDomain>	Keycloak	`admin`	`admin`
<baseDomain>/s3/	S3-совместимое хранилище MinIO	`minioadmin`	`minioadmin`

Если Helm-релиз имеет статус `DEPLOYED`, а все поды успешно запущены, развёртывание платформы Astra Wine считается успешным.

Справочник

Параметры развёртывания

Параметры развёртывания задаются в конфигурационных файлах, хранящихся в подкаталоге `environ/` установщика.

Значения логических параметров задаются в числовом или строковом виде:

Таблица 10. Значения логических параметров

Значение параметра	Число	Строка
Выключен	`0`	`false`
Включен	`1`	`true`

Развёртывание на одном узле

`.env`

Файл `.env` содержит основные параметры развёртывания платформы и её подключения к различным сервисам.

DEBUG

Отладочный режим работы сервера Astra Wine.

ALLOWED_HOSTS

Список IP-адресов, с которых разрешена обработка запросов на создание, изменение, удаление и обновление данных платформы Astra Wine.

ALLOWED_PORT

Номер порта, при подключении по которому разрешена обработка запросов на создание, изменение, удаление и обновление данных платформы Astra Wine.

PROTOCOL

Протокол, используемый для доступа к API платформы Astra Wine.

Возможные значения:

- `http`;
- `https`.

CELERY_BROKER_URL

URI для доступа к БД Redis, в которой хранятся данные брокера Celery.

CELERY_RESULT_BACKEND

URI для доступа к БД Redis, в которой хранятся данные бэкенда Celery.

WINE_GROUP_ADMIN

Название группы, предоставляющей пользователям платформы Astra Wine привилегии роли администратора.

WINE_GROUP_ROOT

Название группы, предоставляющей пользователям платформы Astra Wine привилегии служебной роли.

WINE_GROUP_USER

Название группы, предоставляющей пользователям платформы Astra Wine привилегии роли пользователя.

MAX_TASK_RETRY

Максимальное количество попыток выполнения задач.

WINE_SYSTEM_ACCOUNT_LOGIN

Название системной учётной записи, используемой для выполнения задач без участия пользователя.

WINE_SYSTEM_ACCOUNT_PASSWORD

Пароль системной учётной записи, используемой для выполнения задач без участия пользователя.

S3_ENDPOINT

URI для доступа к API S3-совместимого хранилища.

S3_ACCESS_KEY

Ключ доступа к S3-совместимому хранилищу.

S3_SECRET_KEY

Ключ для доступа к API S3-совместимого хранилища.

S3_STORAGE_BUCKET_NAME

Название бакета в S3-совместимом хранилище.

S3_REGION_NAME

Название географического региона, в котором размещается S3-совместимое хранилище.

S3_CUSTOM_DOMAIN

URI для доступа к S3-совместимому хранилищу.

S3_URL_PROTOCOL

Протокол для доступа к S3-совместимому хранилищу.

Возможные значения:

- `http:`;
- `https:`.

AWS_S3_USE_SSL

Использование защищённого соединения (SSL/TLS) при подключении к S3-совместимому хранилищу.

Возможные значения:

- `0` — незащищённое соединение;
- `1` — защищённое соединение.

AWS_S3_VERIFY

Проверка TLS-сертификата сервера S3-совместимого хранилища.

Возможные значения:

- `0` — сертификат сервера не проверяется;
- `1` — сертификат сервера должен быть доверенным и корректно подписан.

SEMAPHORE_ANSIBLE_USER_LOGIN

Название служебной учётной записи Ansible.

Указанная учётная запись должна существовать на рабочей станции пользователя, разрешать подключение по SSH с использованием пароля и позволять выполнять команды с привилегиями суперадминистратора.

SEMAPHORE_ANSIBLE_USER_PASSWORD

Пароль служебной учётной записи Ansible.

CENTRIFUGO_API_KEY

Идентификатор, используемый для доступа к API Centrifugo.

CENTRIFUGO_API_URL

URI для доступа к API Centrifugo.

CENTRIFUGO_SECRET_KEY

Секретный ключ, используемый для доступа к API Centrifugo.

POSTGRES_USER

Название учётной записи, используемой для подключения к БД с данными платформы Astra Wine.

POSTGRES_DB

Название БД для хранения данных платформы Astra Wine.

POSTGRES_PASSWORD

Пароль для доступа к БД платформы Astra Wine.

PGDATA

Путь к каталогу для хранения данных СУБД PostgreSQL.

POSTGRES_HOST

IP-адрес или FQDN сервера PostgreSQL, хранящего данные платформы Astra Wine.

POSTGRES_PORT

Порт для подключения к серверу PostgreSQL с БД платформы Astra Wine.

REDIS_HOST

IP-адрес или FQDN сервера Redis.

REDIS_PORT

Порт сервера Redis.

MINIO_ROOT_USER

Название учётной записи администратора S3-совместимого хранилища MinIO.

MINIO_ROOT_PASSWORD

Пароль администратора S3-совместимого хранилища MinIO.

MINIO_IDENTITY_OPENID_CONFIG_URL

URL конфигурации OpenID Connect-провайдера, используемый MinIO для получения параметров аутентификации.

MINIO_IDENTITY_OPENID_CLAIM_USERINFO

Источник данных OpenID Connect, используемый MinIO для получения атрибутов пользователя (response UserInfo).

MINIO_IDENTITY_OPENID_CLIENT_ID

Идентификатор клиента OpenID Connect, используемого MinIO для подключения к провайдеру аутентификации.

MINIO_IDENTITY_OPENID_CLIENT_SECRET

Секретный ключ клиента OpenID Connect, используемого MinIO для аутентификации у провайдера.

MINIO_IDENTITY_OPENID_DISPLAY_NAME

Отображаемое название OpenID Connect-провайдера в интерфейсе аутентификации MinIO.

MINIO_IDENTITY_OPENID_REDIRECT_URI

URI перенаправления, используемый MinIO при завершении процесса OpenID Connect-аутентификации.

MINIO_BROWSER_REDIRECT_URL

URL перенаправления веб-интерфейса MinIO после завершения аутентификации пользователя.

MINIO_IDENTITY_OPENID_CLAIM_NAME

Название утверждения (claim), используемого MinIO для идентификации и сопоставления пользователя при OpenID Connect-аутентификации.

MINIO_IDENTITY_OPENID_SCOPES

Список областей доступа (scopes) OpenID Connect, запрашиваемых MinIO у провайдера аутентификации.

KC_BOOTSTRAP_ADMIN_USERNAME

Название учётной записи администратора Keycloak.

KC_BOOTSTRAP_ADMIN_PASSWORD

Пароль администратора Keycloak.

KC_DB

Название БД с данными Keycloak.

KC_DB_URL

URI для доступа к БД Keycloak.

KC_DB_USERNAME

Название учётной записи для доступа к БД Keycloak.

KC_DB_PASSWORD

Пароль для доступа к БД Keycloak.

KC_HOSTNAME_DJANGO

IP-адрес или FQDN сервера Keycloak, используемого платформой Astra Wine.

KC_HOSTNAME

IP-адрес или FQDN узла для развёртывания на нём сервера Keycloak.

KC_HOSTNAME_PORT

Номер порта, используемый для подключения к серверу Keycloak.

KC_HOSTNAME_STRICT

Строгая проверка доменного имени сервера Keycloak.

KC_HOSTNAME_STRICT_HTTPS

Строгая проверка TLS-сертификата сервера Keycloak.

KC_LOG_LEVEL

Уровень критичности сведений, попадающих в журнал работы сервера Keycloak. Сообщения с уровнем ниже указанного в журнала попадать не будут.

KC_METRICS_ENABLED

Предоставление доступа к конечной точке API, содержащей метрики сервера Keycloak.

KC_HEALTH_ENABLED

Предоставление доступа к конечной точке API, содержащей данные о работоспособности сервера Keycloak.

KC_REALM

Название домена, обслуживаемого сервером Keycloak.

KC_CLIENT_ID

Идентификатор платформы Astra Wine на сервере Keycloak.

KC_CLIENT_SECRET

Секретный ключ, используемый для доступа к серверу Keycloak.

KC_HTTPS_CERTIFICATE_FILE

Путь к файлу TLS-сертификата, используемого сервером Keycloak для обработки HTTPS-соединений.

KC_HTTPS_CERTIFICATE_KEY_FILE

Путь к файлу приватного ключа TLS-сертификата, используемого сервером Keycloak.

PROMETHEUS_URL

URI для доступа к API сервера Prometheus.

PROMETHEUS_TIMEOUT

Временной интервал в секундах между опросами состояния сервисов со стороны системы мониторинга Prometheus.

MINIO_PROMETHEUS_AUTH_TYPE

Тип аутентификации сервера Prometheus при доступе к API S3-совместимого хранилища MinIO.

Возможные значения:

- `"jwt"` — токен JWT;
- `"public"` — аутентификация не требуется.

DATA_SOURCE_URI

URI для подключения Prometheus к источнику данных с метриками в СУБД.

DATA_SOURCE_USER

Название учётной записи для подключения Prometheus к источнику данных в СУБД.

DATA_SOURCE_PASS

Пароль учётной записи, используемой Prometheus для подключения к источнику данных в СУБД.

GF_SECURITY_ADMIN_PASSWORD

Пароль учётной записи администратора Grafana.

GF_SECURITY_ADMIN_USER

Название учётной записи администратора Grafana.

GF_AUTH_GENERIC_OAUTH_ENABLED

Использование OAuth-аутентификации в Grafana.

GF_AUTH_GENERIC_OAUTH_NAME

Отображаемое название OAuth-провайдера в интерфейсе Grafana.

GF_AUTH_GENERIC_OAUTH_ALLOW_SIGN_UP

Автоматическое создание учётных записей пользователей Grafana при аутентификации через OAuth.

GF_AUTH_GENERIC_OAUTH_CLIENT_ID

Идентификатор OAuth-клиента Grafana, зарегистрированного у провайдера аутентификации.

GF_AUTH_GENERIC_OAUTH_CLIENT_SECRET

Секретный ключ OAuth-клиента Grafana, используемый для аутентификации у провайдера.

GF_AUTH_GENERIC_OAUTH_SCOPES

Список OAuth-областей доступа (scopes), запрашиваемых Grafana при аутентификации пользователей.

GF_AUTH_GENERIC_OAUTH_AUTH_URL

URL конечной точки авторизации OAuth-провайдера, используемой Grafana.

GF_AUTH_GENERIC_OAUTH_TOKEN_URL

URL конечной точки получения токенов OAuth-провайдера, используемой Grafana.

GF_AUTH_GENERIC_OAUTH_API_URL

URL OAuth API провайдера для получения информации о пользователе, используемый Grafana.

GF_AUTH_GENERIC_OAUTH_ROLE_ATTRIBUTE_PATH

Выражение для определения роли пользователя в Grafana на основании атрибутов, полученных от OAuth-провайдера.

GF_AUTH_GENERIC_OAUTH_ROLE_ATTRIBUTE_STRICT

Соответствие роли пользователя Grafana данным, полученным от OAuth-провайдера.

GF_AUTH_GENERIC_OAUTH_TLS_SKIP_VERIFY_INSECURE

Проверка TLS-сертификата OAuth-провайдера при аутентификации Grafana.

Возможные значения:

- `false` — проверка отключена;
- `true` — проверка включена.

GF_SESSION_COOKIE_SECURE

Передача Cookie сессии Grafana только по защищённому HTTPS-соединению.

GF_SESSION_COOKIE_SAMESITE

Политика SameSite для Cookie сессии Grafana.

GF_SERVER_ROOT_URL

Корневой URL, по которому осуществляется доступ к веб-интерфейсу Grafana.

GF_SERVER_SERVE_FROM_SUB_PATH

Публикация веб-интерфейса Grafana из подкаталога URL.

GF_SERVER_PROTOCOL

Протокол, используемый сервером Grafana для обработки HTTP-запросов.

GF_SERVER_DOMAIN

Доменное имя или IP-адрес сервера Grafana.

GF_SERVER_HTTP_ADDR

IP-адрес, на котором сервер Grafana принимает входящие соединения.

GF_SERVER_HTTP_PORT

Порт, на котором сервер Grafana принимает HTTP-соединения.

SEMAPHORE_DB_USER

Название учётной записи для подключения к БД Semaphore.

SEMAPHORE_DB_PASS

Пароль для доступа к БД Semaphore.

SEMAPHORE_DB_HOST

IP-адрес или FQDN сервера БД Semaphore.

SEMAPHORE_DB_PORT

Порт сервера БД Semaphore.

SEMAPHORE_DB_NAME

Название БД Semaphore.

SEMAPHORE_DB_DIALECT

Тип СУБД, используемой для хранения данных Semaphore.

Возможные значения:

- `bolt` — BoltDB;
- `mysql` — MySQL;
- `postgres` — PostgreSQL.

SEMAPHORE_ADMIN_PASSWORD

Пароль администратора Semaphore.

SEMAPHORE_ADMIN_NAME

Имя администратора Semaphore.

SEMAPHORE_ADMIN_EMAIL

Адрес электронной почты администратора Semaphore.

SEMAPHORE_ADMIN

Название учётной записи администратора Semaphore.

SEMAPHORE_COOKIE_HASH

Хеш Cookie, используемых для доступа к Semaphore.

SEMAPHORE_COOKIE_ENCRYPTION

Пароль для шифрования Cookie сервера Semaphore.

SEMAPHORE_ACCESS_KEY_ENCRYPTION

Ключ шифрования Semaphore.

SEMAPHORE_ENCRYPTION_KEY

Ключ шифрования для защиты данных сервера Semaphore.

SEMAPHORE_URL

URI для доступа к API сервера Semaphore.

SEMAPHORE_OIDC_PROVIDERS

Конфигурация провайдеров OIDC для Semaphore.

SEMAPHORE_WEB_ROOT

URL, по которому доступен веб-интерфейс Semaphore.

ANSIBLE_PYTHON_INTERPRETER

Путь к интерпретатору Python управляемых узлах.

REDIS_ADDR

URI сервера Redis.

LDAP_USER

Название учётной записи LDAP, используемой для аутентификации запросов.

LDAP_PASSWORD

Пароль учётной записи LDAP, используемой для выполнения запросов к каталогу LDAP.

LDAP_URL

URI сервера LDAP.

GUACAMOLE_CRYPT_KEY

Ключ шифрования, используемый Apache Guacamole для защиты чувствительных данных.

SSO_KEYCLOAK_AUTH

Использование Keycloak в качестве сервиса единой аутентификации (SSO) платформы Astra Wine. При этом локальная аутентификация не отключается.

`system_account.env`

Файл `system_account.env` содержит учётные данные, используемые Semaphore для подключения к рабочим станциям пользователей при выполнении заданий Ansible.

SEMAPHORE_ANSIBLE_USER_LOGIN

Название служебной учётной записи Ansible.

Указанная учётная запись должна существовать на рабочей станции пользователя, разрешать подключение по SSH с использованием пароля и позволять выполнять команды с привилегиями суперадминистратора.

SEMAPHORE_ANSIBLE_USER_PASSWORD

Пароль служебной учётной записи Ansible.

`ip_for_deploy.env`

Файл `ip_for_deploy.env` содержит параметры доступа к платформе Astra Wine.

IP_FOR_DEPLOY`

IP-адрес для развёртывания платформы Astra Wine.

PORT_FOR_DEPLOY

Порт для доступа к платформе Astra Wine.

Развёртывание в кластере Kubernetes

`secrets.example.yaml`

ALLOWED_HOSTS

Список IP-адресов, с которых разрешена обработка запросов на создание, изменение, удаление и обновление данных платформы Astra Wine.

ALLOWED_PORT

Номер порта, при подключении по которому разрешена обработка запросов на создание, изменение, удаление и обновление данных платформы Astra Wine.

ANSIBLE_PYTHON_INTERPRETER

Путь к интерпретатору Python управляемых узлах.

CELERY_BROKER_URL

URI для доступа к БД Redis, в которой хранятся данные брокера Celery.

CELERY_RESULT_BACKEND

URI для доступа к БД Redis, в которой хранятся данные бэкенда Celery.

CENTRIFUGO_API_KEY

Идентификатор, используемый для доступа к API Centrifugo.

CENTRIFUGO_API_URL

URI для доступа к API Centrifugo.

CENTRIFUGO_SECRET_KEY

Секретный ключ, используемый для доступа к API Centrifugo.

DEBUG

Отладочный режим работы сервера Astra Wine.

DATA_SOURCE_PASS

Пароль учётной записи, используемой Prometheus для подключения к источнику данных в СУБД.

DATA_SOURCE_URI

URI для подключения Prometheus к источнику данных с метриками в СУБД.

DATA_SOURCE_USER

Название учётной записи для подключения Prometheus к источнику данных в СУБД.

GF_AUTH_GENERIC_OAUTH_API_URL

URL OAuth API провайдера для получения информации о пользователе, используемый Grafana.

GF_AUTH_GENERIC_OAUTH_AUTH_URL

URL конечной точки авторизации OAuth-провайдера, используемой Grafana.

GF_AUTH_GENERIC_OAUTH_CLIENT_ID

Идентификатор OAuth-клиента Grafana, зарегистрированного у провайдера аутентификации.

GF_AUTH_GENERIC_OAUTH_CLIENT_SECRET

Секретный ключ OAuth-клиента Grafana, используемый для аутентификации у провайдера.

GF_AUTH_GENERIC_OAUTH_ENABLED

Использование OAuth-аутентификации в Grafana.

GF_AUTH_GENERIC_OAUTH_NAME

Отображаемое название OAuth-провайдера в интерфейсе Grafana.

GF_AUTH_GENERIC_OAUTH_ROLE_ATTRIBUTE_PATH

Выражение для определения роли пользователя в Grafana на основании атрибутов, полученных от OAuth-провайдера.

GF_AUTH_GENERIC_OAUTH_ROLE_ATTRIBUTE_STRICT

Соответствие роли пользователя Grafana данным, полученным от OAuth-провайдера.

GF_AUTH_GENERIC_OAUTH_SCOPES

Список OAuth-областей доступа (scopes), запрашиваемых Grafana при аутентификации пользователей.

GF_AUTH_GENERIC_OAUTH_TOKEN_URL

URL конечной точки получения токенов OAuth-провайдера, используемой Grafana.

GF_AUTH_GENERIC_OAUTH_TLS_SKIP_VERIFY_INSECURE

Проверка TLS-сертификата OAuth-провайдера при аутентификации Grafana.

Возможные значения:

- `false` — проверка отключена;
- `true` — проверка включена.

GF_AUTH_GENERIC_OAUTH_ALLOW_SIGN_UP

Автоматическое создание учётных записей пользователей Grafana при аутентификации через OAuth.

GF_SECURITY_ADMIN_PASSWORD

Пароль учётной записи администратора Grafana.

GF_SECURITY_ADMIN_USER

Название учётной записи администратора Grafana.

GF_SERVER_DOMAIN

Доменное имя или IP-адрес сервера Grafana.

GF_SERVER_HTTP_ADDR

IP-адрес, на котором сервер Grafana принимает входящие соединения.

GF_SERVER_HTTP_PORT

Порт, на котором сервер Grafana принимает HTTP-соединения.

GF_SERVER_PROTOCOL

Протокол, используемый сервером Grafana для обработки HTTP-запросов.

GF_SERVER_ROOT_URL

Корневой URL, по которому осуществляется доступ к веб-интерфейсу Grafana.

GF_SERVER_SERVE_FROM_SUB_PATH

Публикация веб-интерфейса Grafana из подкаталога URL.

GF_SESSION_COOKIE_SAMESITE

Политика SameSite для Cookie сессии Grafana.

GF_SESSION_COOKIE_SECURE

Передача Cookie сессии Grafana только по защищённому HTTPS-соединению.

GUACAMOLE_CRYPT_KEY

Ключ шифрования, используемый Apache Guacamole для защиты чувствительных данных.

KC_BOOTSTRAP_ADMIN_PASSWORD

Пароль администратора Keycloak.

KC_BOOTSTRAP_ADMIN_USERNAME

Название учётной записи администратора Keycloak.

KEYCLOAK_ADMIN_USER

Название учётной записи администратора Keycloak.

KEYCLOAK_ADMIN_PASSWORD

Пароль администратора Keycloak.

KC_CLIENT_ID

Идентификатор платформы Astra Wine на сервере Keycloak.

KC_CLIENT_SECRET

Секретный ключ, используемый для доступа к серверу Keycloak.

KC_DB

Название БД с данными Keycloak.

KC_DB_PASSWORD

Пароль для доступа к БД Keycloak.

KC_DB_URL

URI для доступа к БД Keycloak.

KC_DB_USERNAME

Название учётной записи для доступа к БД Keycloak.

KC_HEALTH_ENABLED

Предоставление доступа к конечной точке API, содержащей данные о работоспособности сервера Keycloak.

KC_HOSTNAME

IP-адрес или FQDN узла для развёртывания на нём сервера Keycloak.

KC_HOSTNAME_ADMIN_URL

Внешний URL панели администратора сервера Keycloak.

KC_HOSTNAME_DJANGO

IP-адрес или FQDN сервера Keycloak, используемого платформой Astra Wine.

KC_HOSTNAME_PATH

Путь публикации сервера Keycloak относительно корневого URL.

KC_HOSTNAME_PORT

Номер порта, используемый для подключения к серверу Keycloak.

KC_HOSTNAME_STRICT

Строгая проверка доменного имени сервера Keycloak.

KC_HOSTNAME_STRICT_HTTPS

Строгая проверка TLS-сертификата сервера Keycloak.

KC_HOSTNAME_URL

URI для доступа к серверу Keycloak.

KC_HTTP_ENABLED

Использование HTTP-протокола для подключения к серверу Keycloak.

KC_LOG_LEVEL

Уровень критичности сведений, попадающих в журнал работы сервера Keycloak. Сообщения с уровнем ниже указанного в журнала попадать не будут.

KC_METRICS_ENABLED

Предоставление доступа к конечной точке API, содержащей метрики сервера Keycloak.

KC_PROXY

Использование сервера прокси для подключения к серверу Keycloak.

KC_PROXY_ADDRESS_FORWARDING

Использование заголовков `X-Forwarded-*` для определения адреса и протокола клиентского соединения.

KC_REALM

Название домена, обслуживаемого сервером Keycloak.

LICENSE_AGENT_LIMIT

Максимально допустимое количество агентов в рамках лицензии платформы.

MAX_TASK_RETRY

Максимальное количество попыток выполнения задач.

MINIO_BROWSER_REDIRECT_URL

URL перенаправления веб-интерфейса MinIO после завершения аутентификации пользователя.

MINIO_IDENTITY_OPENID_CLIENT_ID

Идентификатор клиента OpenID Connect, используемого MinIO для подключения к провайдеру аутентификации.

MINIO_IDENTITY_OPENID_CLIENT_SECRET

Секретный ключ клиента OpenID Connect, используемого MinIO для аутентификации у провайдера.

MINIO_IDENTITY_OPENID_CLAIM_NAME

Название утверждения (claim), используемого MinIO для идентификации и сопоставления пользователя при OpenID Connect-аутентификации.

MINIO_IDENTITY_OPENID_CLAIM_USERINFO

Источник данных OpenID Connect, используемый MinIO для получения атрибутов пользователя (response UserInfo).

MINIO_IDENTITY_OPENID_CONFIG_URL

URL конфигурации OpenID Connect-провайдера, используемый MinIO для получения параметров аутентификации.

MINIO_IDENTITY_OPENID_DISPLAY_NAME

Отображаемое название OpenID Connect-провайдера в интерфейсе аутентификации MinIO.

MINIO_IDENTITY_OPENID_REDIRECT_URI

URI перенаправления, используемый MinIO при завершении процесса OpenID Connect-аутентификации.

MINIO_IDENTITY_OPENID_SCOPES

Список областей доступа (scopes) OpenID Connect, запрашиваемых MinIO у провайдера аутентификации.

MINIO_PROMETHEUS_AUTH_TYPE

Тип аутентификации сервера Prometheus при доступе к API S3-совместимого хранилища MinIO.

Возможные значения:

- `jwt` — токен JWT;
- `public` — аутентификация не требуется.

MINIO_ROOT_PASSWORD

Пароль администратора S3-совместимого хранилища MinIO.

MINIO_ROOT_USER

Название учётной записи администратора S3-совместимого хранилища MinIO.

PROMETHEUS_TIMEOUT

Временной интервал в секундах между опросами состояния сервисов со стороны системы мониторинга Prometheus.

PROMETHEUS_URL

URI для доступа к API сервера Prometheus.

POSTGRES_DB

Название БД для хранения данных платформы Astra Wine.

POSTGRES_HOST

IP-адрес или FQDN сервера PostgreSQL, хранящего данные платформы Astra Wine.

POSTGRES_PASSWORD

Пароль для доступа к БД платформы Astra Wine.

POSTGRES_PORT

Порт для подключения к серверу PostgreSQL с БД платформы Astra Wine.

POSTGRES_USER

Название учётной записи, используемой для подключения к БД с данными платформы Astra Wine.

PROTOCOL

Протокол, используемый для доступа к API платформы Astra Wine.

Возможные значения:

- `http`;
- `https`.

PORT_FOR_DEPLOY

Порт для доступа к платформе Astra Wine.

PGDATA

Путь к каталогу для хранения данных СУБД PostgreSQL.

REDIS_ADDR

URI сервера Redis.

REDIS_HOST

IP-адрес или FQDN сервера Redis.

REDIS_PORT

Порт сервера Redis.

S3_ACCESS_KEY

Ключ доступа к S3-совместимому хранилищу.

S3_CUSTOM_DOMAIN

URI для доступа к S3-совместимому хранилищу.

S3_ENDPOINT

URI для доступа к API S3-совместимого хранилища.

S3_SECRET_KEY

Ключ для доступа к API S3-совместимого хранилища.

S3_STORAGE_BUCKET_NAME

Название бакета в S3-совместимом хранилище.

S3_URL_PROTOCOL

Протокол для доступа к S3-совместимому хранилищу.

Возможные значения:

- `http:`;
- `https:`.

SEMAPHORE_ACCESS_KEY_ENCRYPTION

Ключ шифрования Semaphore.

SEMAPHORE_ADMIN

Название учётной записи администратора Semaphore.

SEMAPHORE_ADMIN_EMAIL

Адрес электронной почты администратора Semaphore.

SEMAPHORE_ADMIN_NAME

Имя администратора Semaphore.

SEMAPHORE_ADMIN_PASSWORD

Пароль администратора Semaphore.

SEMAPHORE_COOKIE_ENCRYPTION

Пароль для шифрования Cookie сервера Semaphore.

SEMAPHORE_COOKIE_HASH

Хеш Cookie, используемых для доступа к Semaphore.

SEMAPHORE_DB_DIALECT

Тип СУБД, используемой для хранения данных Semaphore.

Возможные значения:

- `bolt` — BoltDB;
- `mysql` — MySQL;
- `postgres` — PostgreSQL.

SEMAPHORE_DB_HOST

IP-адрес или FQDN сервера БД Semaphore.

SEMAPHORE_DB_NAME

Название БД Semaphore.

SEMAPHORE_DB_PASS

Пароль для доступа к БД Semaphore.

SEMAPHORE_DB_PORT

Порт сервера БД Semaphore.

SEMAPHORE_DB_USER

Название учётной записи для подключения к БД Semaphore.

SEMAPHORE_ENCRYPTION_KEY

Ключ шифрования для защиты данных сервера Semaphore.

SEMAPHORE_OIDC_PROVIDERS

Конфигурация провайдеров OIDC для Semaphore.

SEMAPHORE_URL

URI для доступа к API сервера Semaphore.

SEMAPHORE_WEB_ROOT

URL, по которому доступен веб-интерфейс Semaphore.

SEMAPHORE_PORT

Порт, на котором сервер Semaphore принимает входящие HTTP-соединения.

SEMAPHORE_ANSIBLE_USER_LOGIN

Название служебной учётной записи Ansible.

Указанная учётная запись должна существовать на рабочей станции пользователя, разрешать подключение по SSH с использованием пароля и позволять выполнять команды с привилегиями суперадминистратора.

SEMAPHORE_ANSIBLE_USER_PASSWORD

Пароль служебной учётной записи Ansible.

SSO_KEYCLOAK_AUTH

Использование Keycloak в качестве сервиса единой аутентификации (SSO) платформы Astra Wine. При этом локальная аутентификация не отключается.

WINE_GROUP_ADMIN

Название группы, предоставляющей пользователям платформы Astra Wine привилегии роли администратора.

WINE_GROUP_ROOT

Название группы, предоставляющей пользователям платформы Astra Wine привилегии служебной роли.

WINE_GROUP_USER

Название группы, предоставляющей пользователям платформы Astra Wine привилегии роли пользователя.

WINE_SYSTEM_ACCOUNT_LOGIN

Название системной учётной записи, используемой для выполнения задач без участия пользователя.

WINE_SYSTEM_ACCOUNT_PASSWORD

Пароль системной учётной записи, используемой для выполнения задач без участия пользователя.