

Инструкция по установке программного продукта «Astra Migration»

Листов 12

Содержание

1. Назначение ПО.....	3
2. Состав комплекта поставки.....	3
3. Способы установки серверной части.....	4
4. Подготовка конфигурации перед установкой.....	4
4.1. Установка Docker.....	4
4.2. Настройка RabbitMQ.....	4
4.3. Настройка PostgreSQL.....	5
4.4. Настройка MinIO.....	7
5. Автоматическая установка серверной части.....	7
6. Ручная установка серверной части.....	8
6.1. Импорт образов AMS.....	8
6.2. Импорт образов сервисов обеспечения.....	8
6.3. Запуск контейнеров.....	8
7. Проверка работоспособности серверной части.....	9
8. Установка клиентской части.....	9
8.1. Установка АМА.....	9
8.2. Настройка АМА.....	10
9. Особенности использования внешних сервисов.....	12
10. Результат установки.....	12
Перечень сокращений.....	13

1. Назначение ПО

Программный продукт «Astra Migration» — клиент-серверное ПО, предназначенное для миграции рабочих станций под управлением ОС Windows на ОС Astra Linux.

В состав ПО входят:

- AMS — серверная часть Astra Migration Server, обеспечивающая управление процессом миграции;
- АМА — агент миграции Astra Migration Agent, устанавливаемый на исходные рабочие станции под управлением ОС Windows;
- АМЛ — установочные файлы Astra Migration Linux для целевой операционной системы;
- компоненты `firstboot`, применяемые при первоначальной настройке ОС Astra Linux.

2. Состав комплекта поставки

Комплект поставки включает следующие каталоги и файлы:

Каталог/файл	Назначение
<code>ams</code>	образы микросервисов серверной части AMS. Имена образов начинаются с префикса <code>amt-*</code> . Данные образы обязательны к использованию
<code>redistributable</code>	<code>docker</code> -образы сервисов обеспечения функционирования AMS, в том числе PostgreSQL, MinIO и RabbitMQ. Данные образы не являются частью ПО Astra Migration и при необходимости могут быть заменены соответствующими сервисами, уже развернутыми в инфраструктуре
<code>components</code>	установочные файлы Astra Migration Agent (АМА) и Astra Migration Linux (АМЛ), а также компоненты <code>firstboot</code> , используемые при установке целевой ОС Astra Linux
<code>setup.sh</code>	shell-скрипт автоматической установки серверной части
<code>docker-compose.yml</code>	файл конфигурации для запуска контейнеров AMS
<code>docker-compose-minio.yml</code>	файл конфигурации для запуска контейнера MinIO

3. Способы установки серверной части

Установка серверной части AMS может быть выполнена одним из способов:

- автоматически, с использованием скрипта `setup.sh`;
- вручную, поэтапно.

Перед запуском установки необходимо настроить параметры брокера сообщений RabbitMQ и СУБД PostgreSQL в файле `docker-compose.yml`, параметры доступа к БД в файле `docker-entrypoint-initdb.d/01-init.sql`, а также параметры S3-хранилища MinIO в файле `docker-compose-minio.yml` (см. Раздел 4).

4. Подготовка конфигурации перед установкой

4.1. Установка Docker

Перед выполнением установки серверной части необходимо установить Docker Compose. Предварительно подключить зафиксированную ветку интернет-репозитория Astra Linux. В файле `/etc/apt/sources.list` добавить строки:

```
# Основной репозиторий
deb https://dl.astralinux.ru/astra/frozen/1.8_x86-64/1.8.5/main-repository/ 1.8_x86-64 contrib main non-free non-free-firmware

# Расширенный репозиторий
deb https://dl.astralinux.ru/astra/frozen/1.8_x86-64/1.8.5/extended-repository/ 1.8_x86-64 contrib main non-free non-free-firmware
```

После чего выполнить команду:

```
sudo apt install docker-compose-v2
```

4.2. Настройка RabbitMQ

RabbitMQ используется серверной частью Astra Migration для обмена сообщениями между микросервисами.

В файле `docker-compose.yml` необходимо в параметре `RABBITMQ_DEFAULT_PASS` задать пароль пользователя RabbitMQ, оставив остальные параметры со значениями по умолчанию:

```
services:
  rabbitmq:
```

```
environment:
  RABBITMQ_DEFAULT_USER: migration-server
  RABBITMQ_DEFAULT_PASS: CHANGEME
  RABBITMQ_DEFAULT_VHOST: my_vhost
ports:
  - 5672:5672
```

Также в параметре `RMQ_PASSWORD` необходимо задать пароль для подключения микросервисов AMS:

```
x-common-env:
  RMQ_PASSWORD: CHANGEME
```

4.3. Настройка PostgreSQL

СУБД PostgreSQL используется в качестве базы данных серверной части Astra Migration.

Настройка СУБД PostgreSQL включает:

- создание пользователей БД и служебных БД;
- задание параметров доступа к экземпляру БД;
- настройку параметров подключения микросервисов Astra Migration к служебным БД.

Для создания пользователей БД и служебных БД необходимо в конфигурационном файле `docker-entrypoint-initdb.d/01-init.sql` выполнить следующие настройки:

- создать пользователя `migrations_wave_service_user`, от имени которого сервис управления волнами миграции будет обращаться к своей служебной базе данных, и задать для данного пользователя пароль:

```
CREATE USER "migrations_wave_service_user" WITH PASSWORD $amt$CHANGEME$amt$;
```

- создать служебную базу данных `migrations_wave_service_db` и назначить владельцем пользователя `migrations_wave_service_user`:

```
CREATE DATABASE "migrations_wave_service_db" WITH OWNER "migrations_wave_service_user";
```

- создать пользователя `source_os_user`, от имени которого сервис, связанный с исходными ОС, будет обращаться к своей служебной базе данных и задать для данного пользователя пароль:

```
CREATE USER "source_os_user" WITH PASSWORD $amt$CHANGEME$amt$;
```

- создать служебную базу данных `migrations_wave_service_db` и назначить владельцем пользователя `migrations_wave_service_user`:

```
CREATE DATABASE "source_os_db" WITH OWNER "source_os_user";
```

Для задания параметров доступа к экземпляру БД необходимо в конфигурационном файле `docker-compose.yml` в параметре `POSTGRES_PASSWORD` установить пароль для доступа, оставив остальные параметры со значениями по умолчанию:

```
services:
  postgres:
    environment:
      POSTGRES_DB: migration-server
      POSTGRES_USER: migration-server
      POSTGRES_PASSWORD: CHANGEME
```

Для настройки параметров подключения микросервисов Astra Migration к служебным базам данных необходимо в файле `docker-compose.yml` задать пароли пользователей, от имени которых данные микросервисы к ним подключаются.

В файле `docker-compose.yml` в конфигурации контейнеров используются следующие переменные:

```
DB_LOGIN
DB_PASSWORD
```

Значение переменной `DB_LOGIN` определяет имя пользователя базы данных, а значение переменной `DB_PASSWORD` должно соответствовать паролю этого пользователя, заданному в файле `docker-entrypoint-initdb.d/01-init.sql`.

Если в конфигурации контейнера указано значение:

```
DB_LOGIN: source_os_user
```

то в переменной `DB_PASSWORD` необходимо указать пароль пользователя `source_os_user`, заданный в файле `docker-entrypoint-initdb.d/01-init.sql`.

Если в конфигурации контейнера указано значение:

```
DB_LOGIN: migrations_wave_service_user
```

то в переменной `DB_PASSWORD` необходимо указать пароль пользователя `migrations_wave_service_user`, заданный в файле `docker-entrypoint-initdb.d/01-init.sql`.

4.4. Настройка MinIO

MinIO используется ПО Astra Migration в качестве S3-совместимого хранилища.

В файле `docker-compose-minio.yml` необходимо задать имя пользователя, пароль и логический контейнер в хранилище MinIO:

```
services:
  minio:
    environment:
      S3_ADMIN_USER= adminminio
      S3_ADMIN_PASSWORD= CHANGEME
      S3_BUCKET= amt
```

Необходимо скопировать файлы AML и First Boot из папки `components` в S3 хранилище, в соответствующий настройкам S3 бакет (настройка `S3_BUCKET_NAME`). Название образа AML будет использовано при создании сценария миграции (см. пункт 6.1 Руководства Администратора). Название First Boot должно быть `firstboot.deb` без версии.

5. Автоматическая установка серверной части

Для автоматической установки серверной части следует перейти в каталог с комплектом поставки и выполнить команду:

```
bash ./setup.sh
```

Скрипт `setup.sh` выполняет следующие действия:

- импорт `docker`-образов из каталога `ams` в локальное хранилище `Docker`;
- импорт `docker`-образов из каталога `redistributable` в локальное хранилище `Docker` в случае отсутствия сервисов обеспечения в инфраструктуре;
- запуск контейнеров AMS и сервисов обеспечения с использованием `Docker Compose`.

После завершения работы скрипта `setup.sh` рекомендуется проверить состояние контейнеров (см. Раздел 7).

6. Ручная установка серверной части

6.1. Импорт образов AMS

Если образы серверной части AMS поставляются в виде tar-архивов, то их загрузка в локальное хранилище Docker выполняется командой:

```
docker load -i ams/<имя_файла_образа>.tar
```

Указанную команду следует выполнить для каждого образа из каталога ams.

Если вместо локального хранилища используется сервис хранения контейнерных образов Docker Registry, то после загрузки необходимо присвоить образу тег с указанием адреса registry, после чего отправить образ в registry:

```
docker tag <имя_образа> <адрес_registry>/<имя_образа>  
docker push <адрес_registry>/<имя_образа>
```

6.2. Импорт образов сервисов обеспечения

В инфраструктуре могут отсутствовать рабочие экземпляры сервисов обеспечения PostgreSQL, MinIO и RabbitMQ. В таком случае их образы необходимо импортировать из каталога redistributable.

Если образы поставляются в виде tar-архивов, то выполнить команду:

```
docker load -i redistributable/<имя_файла_образа>.tar
```

Данную команду необходимо выполнить для каждого образа из каталога redistributable.

Если в инфраструктуре уже используются экземпляры сервисов обеспечения PostgreSQL, MinIO и RabbitMQ, то в конфигурационных файлах следует указать параметры подключения к ним (см. Раздел 9).

6.3. Запуск контейнеров

После настройки конфигурационных файлов и загрузки образов запуск контейнеров должен быть выполнен из каталога с файлами поставки.

Если используется плагин Docker Compose, то необходимо последовательно выполнить команды:


```
docker compose -f docker-compose.yml -f docker-compose-minio.yml up -d
docker compose up -d
```

Если используется утилита `docker-compose`, то необходимо последовательно выполнить команды:

```
docker-compose -f docker-compose.yml -f docker-compose-minio.yml up -d
docker compose up -d
```

7. Проверка работоспособности серверной части

После запуска контейнеров необходимо проверить их состояние. При использовании плагина Docker Compose выполнить команду:

```
docker compose -f docker-compose.yml -f docker-compose-minio.yml ps
```

При использовании утилиты `docker-compose` выполнить команду:

```
docker-compose -f docker-compose.yml -f docker-compose-minio.yml ps
```

Контейнеры должны находиться в состоянии «Up».

При необходимости анализа журналов регистрации событий контейнера выполнить команду:

```
docker logs <имя_контейнера>
```

Если в ходе установки использовались внешние экземпляры СУБД PostgreSQL, S3-хранилища MinIO и брокера сообщений RabbitMQ, то следует дополнительно проверить доступность указанных сервисов обеспечения с узла, на котором развернут AMS.

8. Установка клиентской части

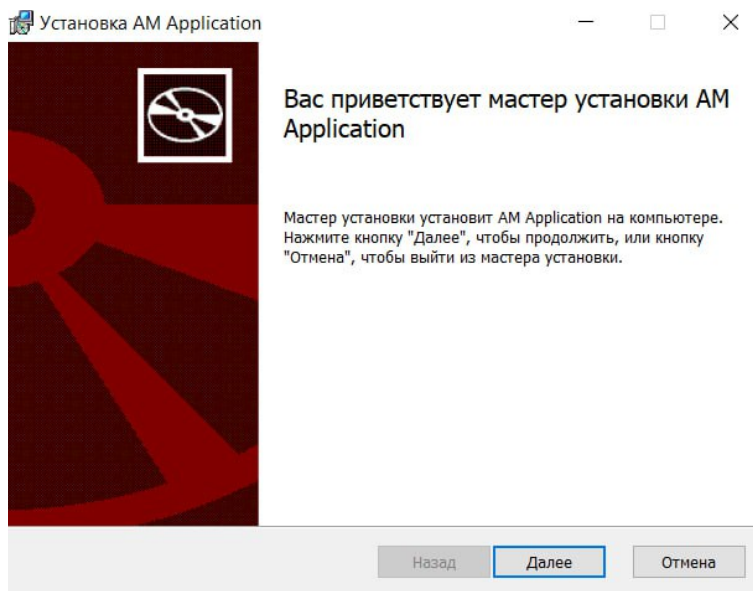
8.1. Установка АМА

Для установки приложения АМА необходимо скопировать `msi`-пакет из каталога `components` на рабочую станцию с ОС Windows, после чего запустить

данный пакет и выполнить действия с использованием мастера установки.

В ходе установки следует:

- принять лицензионное соглашение;
- выбрать путь установки (по умолчанию — C:\Program Files\AM Application);
- подтвердить установку.



Интерфейс установщика АМА

Для проверки статуса работы агента АМА необходимо открыть окно **Службы** одним из способов:

- **Панель управления → Система и безопасность → Инструменты Windows → Службы;**
- выполнить команду: `services.msc`

В появившемся интерфейсе в списке служб найти **АМА Service** и убедиться, что служба работает. При необходимости перезапустить службу **АМА Service**.

8.2. Настройка АМА

Для настройки приложения АМА необходимо создать конфигурационный файл `%PROGRAMDATA%\ama\etc\config.toml`.

В корневой части конфигурационного файла необходимо задать параметры запуска и указать имя службы для приложения:

```
debug = false
service_name = "amt-migration-application-service"
```

В секции `[artifacts_settings]` в параметре `storage_url` задать адрес S3-хранилища серверной части Astra Migration:

```
[artifacts_settings]
storage_url = "https://amt-s3.devos.astralinux.ru/amt"
```

В секции [mq_settings] задать параметры подключения к брокеру сообщений RabbitMQ, который используется на серверной части Astra Migration:

```
[mq_settings]
login = "admin"
password = "password"
host = "localhost"
port = 5672
vhost = "my_vhost"
```

Параметры в секции [mq_settings] должны соответствовать значениям, заданным в конфигурационном файле docker-compose.yml при настройке брокера сообщений RabbitMQ:

- login — значение RABBITMQ_DEFAULT_USER;
- password — значение RABBITMQ_DEFAULT_PASS;
- vhost — значение RABBITMQ_DEFAULT_VHOST;
- host — адрес хоста, на котором доступен сервис RabbitMQ;
- port — порт RabbitMQ, по умолчанию 5672.

ВНИМАНИЕ! При задании пароля допускается использовать:

- латинские буквы верхнего и нижнего регистра (A-Z, a-z);
- цифры (0-9);
- специальные символы, за исключением: обратного слэша «\», прямого слэша «/» и вертикальной черты «|».

В секции [event_settings] задать параметры повторной обработки событий и имя брокера сообщений RabbitMQ:

```
[events_settings]
retry_attempts = 3
retry_wait = 10
amt_exchange = "amt.events"
```

После задания всех параметров следует сохранить конфигурационный файл %PROGRAMDATA%\ama\etc\config.toml и перезагрузить компьютер с ОС Windows для завершения настройки.

9. Особенности использования внешних сервисов

Если вместо образов из каталога `redistributable` используются внешние экземпляры сервисов СУБД PostgreSQL, S3-хранилища MinIO и брокера сообщений RabbitMQ, то в конфигурации AMS необходимо указать параметры подключения к данным сервисам, обеспечив следующее:

- сетевой доступ AMS к PostgreSQL, MinIO и RabbitMQ;
- корректность учетных данных;
- соответствие портов и параметров подключения настройкам используемой инфраструктуры.

10. Результат установки

Установка ПО Astra Migration считается успешной при выполнении следующих условий:

- контейнеры AMS запущены;
- контейнеры сервисов обеспечения запущены либо настроено подключение к внешним сервисам;
- приложение АМА установлено на компьютер с ОС Windows;
- в файле `%PROGRAMDATA%\ama\etc\config.toml` заданы параметры подключения к RabbitMQ.

Перечень сокращений

- AMS — Astra Migration Server
- АМА — Astra Migration Agent
- АМЛ — Astra Migration Linux
- S3 — Simple Storage Service
- БД — база данных
- ОС — операционная система
- ПО — программное обеспечение
- СУБД — система управления базами данных