

# **Программное обеспечение «Атлас Закупки»**

ОПИСАНИЕ ЭКЗЕМПЛЯРА  
ПРОГРАММНОГО ОБЕСПЕЧЕНИЯ  
состав, процессы, контейнеры, сети,  
расположение файлов и компонентов

Версия 1.0.0

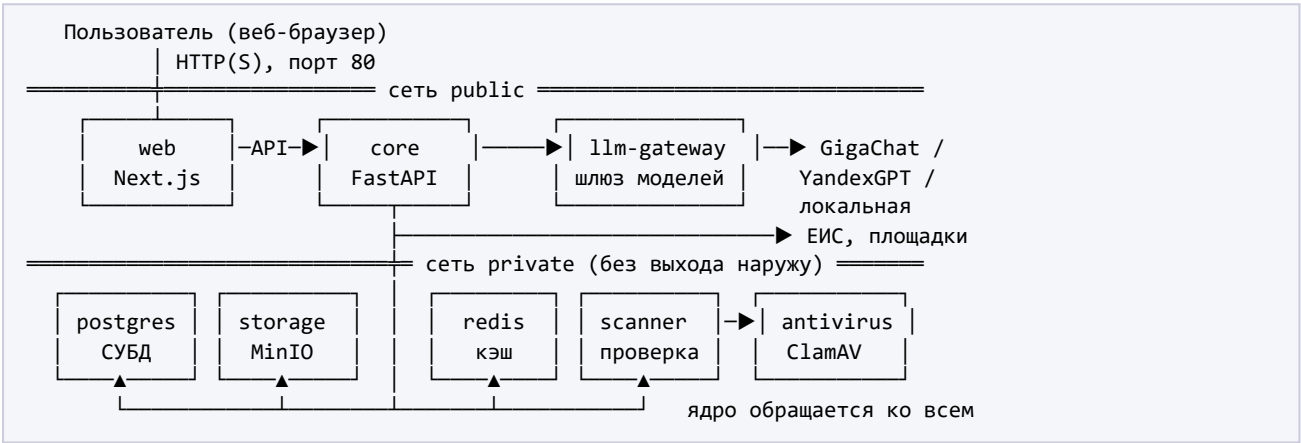
Правообладатель: ООО «СВЯЗЬ»  
ОГРН 1257800119422, ИНН 7805827854  
115487, г. Москва, ул. Нагатинская, д. 30, помещ. 42Н

Москва, 2026

# 1. АРХИТЕКТУРА ЭКЗЕМПЛЯРА

Экземпляр ПО «Атлас Закупки» 1.0.0 состоит из восьми служб, каждая работает в отдельном контейнере под управлением Docker Engine. Состав служб, сетей и томов задан файлом /opt/atlas-zakupki/docker-compose.yml; настройки — файлом /opt/atlas-zakupki/.env. Службы связаны двумя сетями:

- **private** — внутренняя сеть без выхода за пределы сервера (internal: true): база данных, хранилище документов, кэш, антивирус и прослойка антивируса. Данные организаций физически не могут быть переданы из этих контейнеров наружу;
- **public** — сеть с выходом в интернет: ядро (обращения к источникам сведений о закупках), шлюз языковых моделей (обращения к GigaChat или YandexGPT) и веб-кабинет (порт для пользователей).



## 1.1. Службы и контейнеры

| Служба      | Образ                           | Базовая система   | Порт (внутр.)                   | Сети            | Память (предел) |
|-------------|---------------------------------|-------------------|---------------------------------|-----------------|-----------------|
| core        | atlas-zakupki/core:1.0.0        | Debian 13         | 8200; на сервере 127.0.0.1:8200 | private, public | 3 ГБ            |
| web         | atlas-zakupki/web:1.0.0         | Alpine Linux 3.23 | 3300; на сервере 0.0.0.0:80     | public          | 512 МБ          |
| postgres    | postgres:16                     | Debian 13         | 5432                            | private         | 1 ГБ            |
| storage     | atlas-zakupki/storage:1.0.0     | Debian 13         | 9000                            | private         | 512 МБ          |
| antivirus   | atlas-zakupki/antivirus:1.0.0   | Alpine Linux 3.24 | 3310                            | private         | 2 ГБ            |
| scanner     | atlas-zakupki/scanner:1.0.0     | Debian 13         | 8080                            | private         | 256 МБ          |
| llm-gateway | atlas-zakupki/llm-gateway:1.0.0 | Debian 13         | 8093                            | private, public | 256 МБ          |
| redis       | redis:7.2-alpine                | Alpine Linux 3.21 | 6379                            | private         | 192 МБ          |

Имена контейнеров: <проект>-<служба>-1, по умолчанию atlas-zakupki-core-1, atlas-zakupki-web-1 и т. д. Базовые системы всех образов — свободные дистрибутивы Debian GNU/Linux и

Alpine Linux; систем с экспортными ограничениями (Red Hat, CentOS, Oracle Linux) в комплекте нет — это проверяется сценарием сборки. Служба перезапускается автоматически при сбое (restart: unless-stopped).

## 2. ПРОЦЕССЫ

| Служба      | Процессы в контейнере                                                                          | Назначение                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                         |
|-------------|------------------------------------------------------------------------------------------------|--------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| core        | uvicorn app.main:app --workers 2<br>(главный процесс и 2 рабочих процесса Python 3.12)         | Программный интерфейс ядра (FastAPI). При запуске: миграции схемы базы (alembic upgrade head), подготовка хранилища. В каждом рабочем процессе — фоновые циклы (единственность выполнения обеспечивается блокировками PostgreSQL): синхронизация с источниками закупок (каждые 30 минут, полная — раз в 6 часов); очередь фоновых задач (загрузка документации, антивирусная проверка, извлечение текста, интеллектуальный анализ, уведомления); быстрый цикл торгов (15 с); догрузка документации; перечитывание флагов. Разбор документов выполняется в отдельных дочерних процессах с пределом памяти (CAGE_*). |
| web         | next-server (Node.js 20)                                                                       | Веб-кабинет: страницы, вход, передача запросов ядру                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                |
| postgres    | postgres; checkpointer, background writer, walwriter, autovacuum launcher; процессы соединений | СУБД PostgreSQL 16; приложение подключается ролью procurement_app                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                  |
| storage     | /usr/bin/minio server /data                                                                    | S3-совместимое хранилище документов (пользователь minio, uid 1001)                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                 |
| antivirus   | /sbin/tini → /init; clamd --foreground                                                         | Антивирус ClamAV; обновление баз (freshclam) выключено: CDN ClamAV из России недоступен, база сигнатур — в образе                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                  |
| scanner     | uvicorn app:app --port 8080                                                                    | Проверка файла по запросу ядра: передаёт поток в clamd, отвечает «чисто / заражён / недоступен»                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                    |
| llm-gateway | python -u /app/gateway.py<br>(пользователь nobody)                                             | OpenAI-совместимый шлюз к GigaChat, YandexGPT, локальной модели                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                    |
| redis       | redis-server *:6379                                                                            | Кэш и счётчики занятости рабочих мест ЭТП (без сохранения на диск)                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                 |

Просмотр процессов службы: `sudo docker top atlas-zakupki-core-1`; всех служб и их состояния: `sudo atlas-zakupki status`; журнал службы: `sudo atlas-zakupki logs core`.

### 2.1. Потоки данных

1. Ядро по расписанию запрашивает сведения о закупках у источников (ЕИС и электронные площадки), приводит их к единой модели, устраняет дубли и сохраняет в PostgreSQL.
2. Документация закупки загружается ядром, передаётся на проверку службе scanner → antivirus; проверенный файл сохраняется в хранилище storage, извлечённый текст — в базе. Файл, не прошедший проверку, не выдаётся пользователям.

3. Для интеллектуального анализа ядро передаёт текст документации шлюзу `llm-gateway`, тот — выбранной модели (GigaChat, YandexGPT или локальной); результат сохраняется в базе вместе со следом вызова (модель, поставщик, время).
4. Пользователь работает в веб-кабинете; кабинет обращается к ядру по внутреннему адресу `http://core:8200`, браузер к ядру напрямую не обращается.

### 3. РАСПОЛОЖЕНИЕ ФАЙЛОВ

#### 3.1. На сервере

| Путь                                                                   | Содержимое                                                                          |
|------------------------------------------------------------------------|-------------------------------------------------------------------------------------|
| <code>/opt/atlas-zakupki/docker-compose.yml</code>                     | Состав служб, сетей, томов                                                          |
| <code>/opt/atlas-zakupki/.env</code>                                   | Настройки, пароли и ключи экземпляра (права 600)                                    |
| <code>/opt/atlas-zakupki/credentials.txt</code>                        | Адрес кабинета и выданные при установке ключи доступа (права 600)                   |
| <code>/opt/atlas-zakupki/config/profile.toml</code>                    | Профиль развёртывания (монтируется в ядро как <code>/profiles/profile.toml</code> ) |
| <code>/opt/atlas-zakupki/atlas-zakupki</code>                          | Утилита управления; ссылка <code>/usr/local/bin/atlas-zakupki</code>                |
| <code>/opt/atlas-zakupki/bootstrap.py</code>                           | Первичная настройка и выдача ключей                                                 |
| <code>/opt/atlas-zakupki/backups/</code>                               | Резервные копии ( <code>atlas-zakupki backup</code> )                               |
| <code>/opt/atlas-zakupki/docs/</code>                                  | Документация                                                                        |
| <code>/var/lib/docker/volumes/atlas-zakupki_pgdata/_data</code>        | Файлы базы данных PostgreSQL (том <code>pgdata</code> )                             |
| <code>/var/lib/docker/volumes/atlas-zakupki_storagedata/_data</code>   | Документы в хранилище (том <code>storagedata</code> )                               |
| <code>/var/lib/docker/containers/&lt;id&gt;/&lt;id&gt;-json.log</code> | Журналы служб (до 5 файлов по 20 МБ на службу)                                      |

#### 3.2. В контейнерах

| Служба   | Путь                                                                               | Содержимое                                                                                                                                                                                                                     |
|----------|------------------------------------------------------------------------------------|--------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| core     | <code>/app/app/</code>                                                             | Исходный текст ядра на Python: <code>main.py</code> (запуск), <code>modules/</code> (разделы API), <code>connectors/</code> (адаптеры источников), <code>storage/</code> , <code>notify/</code> , <code>profiles/</code> и др. |
|          | <code>/app/alembic/</code> , <code>/app/alembic.ini</code>                         | Миграции схемы базы данных                                                                                                                                                                                                     |
|          | <code>/app/entrypoint.sh</code>                                                    | Сценарий запуска: миграции, затем <code>uvicorn</code>                                                                                                                                                                         |
|          | <code>/usr/local/lib/python3.12/site-packages/</code>                              | Библиотеки Python (перечень и лицензии — <code>docs/LICENSES.md</code> )                                                                                                                                                       |
| web      | <code>/app/server.js</code> , <code>/app/.next/</code> , <code>/app/public/</code> | Собранный веб-кабинет Next.js; шрифт Inter — <code>/app/public/fonts/inter/</code>                                                                                                                                             |
| postgres | <code>/var/lib/postgresql/data</code>                                              | Каталог данных СУБД (том <code>pgdata</code> )                                                                                                                                                                                 |

|             |                                                                |                                                                     |
|-------------|----------------------------------------------------------------|---------------------------------------------------------------------|
| storage     | /usr/bin/minio; /data;<br>/usr/share/doc/minio/                | Исполняемый файл MinIO; документы (том); лицензия GNU AGPL v3       |
| antivirus   | /var/lib/clamav/;<br>/etc/clamav/clamd.conf,<br>freshclam.conf | База сигнатур (main, daily, bytecode); настройки clamd и обновления |
| scanner     | /app/app.py                                                    | Служба проверки файла                                               |
| llm-gateway | /app/gateway.py; /app/certs/russian-trusted-ca.crt             | Шлюз моделей; сертификаты Минцифры для проверки TLS GigaChat        |

Осмотр файлов внутри контейнера: `sudo docker exec -it atlas-zakupki-core-1 sh`, затем, например, `ls /app/app`, `cat /etc/os-release`.

#### 4. СЕТЕВЫЕ ОБРАЩЕНИЯ ЗА ПРЕДЕЛЫ СЕРВЕРА

| Служба      | Куда и зачем                                                                                                                                                                                                                                                                                                         | Обязательность                                            |
|-------------|----------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|-----------------------------------------------------------|
| core        | Единая информационная система в сфере закупок (zakupki.gov.ru) и электронные торговые площадки (РТС-тендер, Сбербанк-АСТ, Росэлторг, ТЭК-Торг, B2B-Center, ЗаказРФ, ЭТП ГПБ, Фабрикант, ОТС, Портал поставщиков Москвы, Электронный магазин Московской области и др.) — получение сведений о закупках и документации | Для сбора сведений; без интернета — работа с загруженными |
| core        | DaData (dadata.ru) — сведения о заказчиках из ЕГРЮЛ                                                                                                                                                                                                                                                                  | Необязательно, при заданном ключе                         |
| core        | Почтовый сервер организации (SMTP), мессенджер MAX (platform-api.max.ru) — уведомления                                                                                                                                                                                                                               | Необязательно, при заданных настройках                    |
| llm-gateway | GigaChat API (ПАО Сбербанк, gigachat.devices.sberbank.ru) или YandexGPT (ООО «Яндекс», llm.api.cloud.yandex.net), либо локальная модель в контуре заказчика                                                                                                                                                          | Для интеллектуального анализа, при заданном ключе         |

Обращений к иностранным сервисам, реестрам образов и хранилищам в работе и при установке нет. Обновление ПО выполняется только администратором экземпляра из комплекта нового выпуска (принудительного обновления и управления извне нет). Антивирус не обращается к серверам обновлений (обновление баз — с зеркала организации, по решению администратора).

#### 5. КОМПОНЕНТЫ И ИСХОДНЫЙ ТЕКСТ

| Компонент                                                                                                              | Происхождение                                                                                                                                                                                                                             |
|------------------------------------------------------------------------------------------------------------------------|-------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| Ядро (core), веб-кабинет (web), шлюз моделей (llm-gateway), служба проверки (scanner), установщик и утилита управления | Разработаны работниками правообладателя (ООО «СВЯЗЬ»); исходный текст хранится в системе управления версиями правообладателя, сборка выполняется на его сервере в г. Москве. Коммит исходного текста выпуска указан в BUILD.txt комплекта |

|                                                                              |                                                                                                                                                                                               |
|------------------------------------------------------------------------------|-----------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| PostgreSQL 16, Redis 7.2, MinIO, ClamAV, Python 3.12, Node.js 20, библиотеки | Свободное ПО третьих лиц, используется без изменений; перечень, версии и лицензии — THIRD_PARTY_NOTICES.txt и docs/LICENSES.md; MinIO собирается правообладателем из исходного текста выпуска |
|------------------------------------------------------------------------------|-----------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|

Сведения об образах выпуска (контрольные суммы идентификаторов) выводит команда

```
sudo docker image ls --digests --format '{{.Repository}}:{{.Tag}} {{.ID}}' | grep -E 'atlas-zakupki|postgres|redis'
```

Проверка отсутствия технических средств защиты авторских прав: в составе нет лицензионных ключей, активации и привязки к оборудованию; экземпляр работает без ограничения срока и числа установок.