AI BOX

# ИНСТРУКЦИЯ ПО УСТАНОВКЕ ПРОГРАММНОГО ОБЕСПЕЧЕНИЯ

Версия 2025.3

powered by IT Expert

## СОДЕРЖАНИЕ

# АННОТАЦИЯ

Настоящая инструкция описывает порядок действий администратора по установке и настройке AI BOX.

# ТЕРМИНЫ И СОКРАЩЕНИЯ

**LLM –** large Language Model, большая языковая модель.

**АРМ –** автоматизированное рабочее место.

**ОС** – операционная система.

**ПП** – программный продукт.

**СУБД** – система управления базами данных.

**Токен –** минимальная единица текста, с которой работает LLM.

# 1. ТРЕБОВАНИЯ К ТЕХНИЧЕСКИМ И ПРОГРАММНЫМ СРЕДСТВАМ

Развертывание программного продукта AI BOX осуществляется в среде Linux с применением Docker Compose. Требования к серверным компонентам приведены в таблице (Таблица 1).

**Таблица 1. Системные требования к AI BOX**

| Характеристика | Docker Compose |
|---|---|
| **Операционная система** | Дистрибутив Linux, в котором поддерживается Docker. Рекомендуются поддерживаемые версии Ubuntu или Debian, а также «Astra Linux Special Edition», РЕД ОС релиз «Муром». |
| **Системные компоненты** | – Docker версии 28.0 и выше<br>– Docker Compose версии 2.3.0 или выше |
| **Аппаратные требования** | – Виртуальный процессор: Количество ядер — 8, частота — 2.6 ГГц и более<br>– ОЗУ: 12 ГБ и более<br>– Дисковое пространство: 100 ГБ и более, 1000 IOPS, задержка менее 20 мс<br>– Сетевой интерфейс: Количество — 2, скорость 1 Гбит/с и более<br>– Соотношение виртуальных и физических ядер: 1 виртуальное ядро на 1 физическое ядро |

# 2. УСТАНОВКА ПРОГРАММНОГО ПРОДУКТА

## 2.1 Развертывание сервисов AI BOX Hub

Установка в Docker включает шаги:

1. Установите Docker версии 28.0.0 или выше.
2. Установите Docker Compose версии 2.3.0 или выше (обновление Docker Compose описано в официальной документации).
3. Выберите любое удобное место и извлеките файлы из архива docker-images. Установите docker-образы с помощью команды в терминале: sudo bash install.sh.
4. Выберите любое удобное место в качестве рабочего каталога и извлеките файлы из архива для развертывания приложения в Docker.
5. Из корня рабочего каталога выполните в терминале скрипт: bash deploy/set_prometheus_password.sh
6. Переименуйте файл .env.example в .env и при необходимости скорректируйте значения согласно таблице (Таблица 2).
7. Создайте директорию, указанную в переменной среды NGINX_CERTS_PATH из Таблица 2 и положите в неё файлы SSL сертификатов cert.pem и key.key.
8. Создайте директорию, указанную в переменной среды EXTERNAL_LICENSE_PATH из Таблица 2 и поместите в неё файл лицензии license.json.
9. В рабочей директории проекта выполните команду: docker compose up -d --build.

**Таблица 2. Переменные среды для Docker**

| Название | Описание | Пример |
|---|---|---|
| **Конфигурация MongoDB** | | |
| MONGO_LOGIN | Логин администратора в MongoDB | login |
| MONGO_PASSWORD | Пароль администратора в MongoDB | password |
| MONGO_DB_NAME | Название базы данных MongoDB, которая будет использоваться сервисами.<br>Во время первого запуска сервиса база данных будет создана автоматически и наполнена данными.<br>После того как база данных будет проинициализирована, поменять это значение можно только после удаления всех данных в volume mongo_data | chat_bot_db |

| Название | Описание | Пример |
|---|---|---|
| **Конфигурация Mongo Express** | | |
| MONGO_EXPRESS_LOGIN | Логин для авторизации в веб-интерфейсе Mongo Express | root |
| MONGO_EXPRESS_PASSWORD | Пароль для авторизации в веб-интерфейсе Mongo Express | password |
| **Подключение BPMSoft** | | |
| BPMSOFT_URL | Базовый URL для подключения к BPMSoft | https://test-bpmsoft-url.ru |
| KEYCLOAK_CLIENT_ID | Client Id клиента из KeyCloak | api |
| KEYCLOAK_CLIENT_SECRET | Client secret клиента из KeyCloak | mbWq5cEYf0IvesrCtYctgk8ySkPVyb6d |
| **Конфигурация API** | | |
| API_DEBUG | Логгирование запросов FastAPI | False |
| API_PROJECT_NAME | Title веб-сервиса во вкладке браузера | AI BOX Hub API |
| API_VERSION | Версия API | 1.0.0 |
| API_CORS_ALLOWED_ORIGINS | Адреса с которых разрешены CORS-запросы | *; |
| SESSION_EXPIRES_IN_MIN | Время жизни сессии диалога с LLM, в минутах | 10 |
| API_ADMIN_ROLE_NAME | Роль пользователя, которая позволяет изменять данные на стороне AI BOX Hub API (CUD) | admin_aibox |
| API_SERVICE_ROLE_NAME | Роль пользователя, которая позволяет сервисам взаимодействовать с AI BOX Hub API | service |
| KEYCLOAK_URL | Адрес сервиса Keycloak | https://aibox.hub:8080 |
| **Конфигурация логирования API** | | |
| EXTERNAL_LOGS_PATH | Путь до внешней директории с логами | ../AIboxData/logs |
| LOG_CONSOLE_LEVEL | Уровень логирования для консольного вывода (DEBUG, INFO, WARNING, ERROR, CRITICAL или числовое значение) | ERROR |

| Название | Описание | Пример |
|---|---|---|
| LOG_FILE_LEVEL | Уровень логирования для файлового вывода (DEBUG, INFO, WARNING, ERROR, CRITICAL или числовое значение) | DEBUG |
| **Конфигурация локально развёрнутого сервиса эмбеддингов** | | |
| EMBEDDING_USE_OIDC | Указывает использовать ли OIDC сервис для аутентификации запросов к эмбеддеру (развёрнутому локально или в контуре). Если `True`, то `Basic` аутентификация работать не будет. | True |
| EMBEDDING_CLIENT_ID | Идентификатор клиента OIDC | bpmsoft |
| EMBEDDING_CLIENT_SECRET | Секрет клиента OIDC | Z223eWCQK7AOBJz9KKBjZ4F0s3tPglZe |
| EMBEDDING_BASE_LOGIN | Логин для базовой аутентификации в сервисе эмбеддингов | user |
| EMBEDDING_BASE_PASSWORD | Пароль для базовой аутентификации в сервисе эмбеддингов | password |
| EMBEDDING_OIDC_URL | Адрес сервиса OpenID для получения токена | http://localhost:8085 |
| EMBEDDING_OIDC_REALM | Наименование Realm в котором находится клиент для получения токена | master |
| EMBEDDING_URL | Адрес сервиса-эмбеддера для получения векторов | http://localhost:8000 |
| **Конфигурация Nginx** | | |
| SERVER_HOST | Хост сервисов | aibox.hub |
| NGINX_CERTS_PATH | Путь к сертификатам для Nginx | ./nginx/cert |
| NGINX_CLIENT_MAX_BODY_SIZE | Максимальный размер тела запроса к Nginx | 30M |
| PROMETHEUS_PORT | HTTPS-порт сервиса Prometheus | 8300 |
| KEYCLOAK_PORT | HTTPS-порт сервиса KeyCloak | 8080 |
| MONGO_EXPRESS_PORT | HTTPS-порт сервиса Mongo Express | 8081 |
| API_PORT | HTTPS-порт сервиса AIBox Hub | 8085 |
| **Конфигурация путей в файловой системе** | | |
| EXTERNAL_VECTORIZE_DATA_PATH | Путь до внешней директории, где будут доступны векторы ChromaDB | ../AIboxData/chromadb |

| Название | Описание | Пример |
|---|---|---|
| EXTERNAL_LICENSE_PATH | Путь до внешней директории, где находятся файлы лицензии | ../AIboxData/license/ |
| **Конфигурация KeyCloak** | | |
| KEYCLOAK_REALM | Используемый realm в KeyCloak | master |
| KEYCLOAK_ADMIN | Логин учётной записи администратора | admin |
| KEYCLOAK_ADMIN_PASSWORD | Пароль учётной записи администратора | Admin |
| EXTERNAL_KEYCLOAK_FILES | Путь дополнительных тем для KeyCloak | ../AIboxData/keycloaktheme/ |

## 2.2 Кастомизация KeyCloak

По умолчанию KeyCloak устанавливается с англоязычным интерфейсом и стандартной темой. При необходимости можно установить русскую локализацию и свою тему для KeyCloak.

### 2.2.1 Настройка локализации интерфейса KeyCloak

Локализация интерфейса Keycloak позволяет отображать страницы входа, восстановления пароля, регистрации пользователей и административную консоль на русском языке. Управление настройками локализации осуществляется в разделе Realm Settings административной консоли Keycloak.

Администратор может:

- включать или выключать поддержку мультиязычности;
- добавлять новые локали;
- выбирать язык интерфейса по умолчанию.

Чтобы включить локализацию интерфейса и выбрать русский язык, необходимо выполнить следующие действия:

1. Открыть административную консоль Keycloak.
2. Перейти в раздел «Realm Settings».
3. Открыть вкладку «Localization».
4. Установить параметр «Internationalization» в значение «ON».

5. В списке «Supported locales «добавить русский язык.
6. В поле «Default locale» выбрать русский язык.
7. Нажать «Save».
8. Обновить страницу браузера

После сохранения настроек ряд страниц интерфейса будут отображаться на русском языке.

Для переключения административной консоли KeyCloak на русский язык необходимо:

1. Открыть административную консоль Keycloak.
2. Нажать на имя пользователя в правом верхнем углу.
3. Выбрать пункт «Manage account».
4. Перейти в раздел «Personal info».
5. В поле «Locale» выбрать русский язык.

### 2.2.2 Персонализация интерфейса KeyCloak

Пользовательская тема оформления позволяет изменить визуальный интерфейс страниц входа, регистрации, восстановления пароля и административной консоли Keycloak в соответствии с корпоративными стандартами. Управление темами осуществляется через административную консоль и файловую систему сервера Keycloak.

Для установки своей темы требуется выполнить шаги:

1. Открыть административную консоль Keycloak.
2. Перейти в раздел «Realm Settings».
3. Открыть вкладку «Themes».
4. В полях на вкладке выбрать пользовательскую тему.
5. Нажать «Save».

После сохранения настроек страницы административной консоли KeyCloak будут использовать выбранную тему оформления.

## 2.3 Настройка Keycloak для Admin UI

**Шаг 1. Создание клиента для доступа к административной панели AI BOX.**

1. Откройте консоль Keycloak → панель Clients → Create client.

2. На вкладке «General Settings» заполните поля:
    a. Client type: OpenID Connect;
    b. Client ID: admin-ui-client.

    Нажмите «Next».
3. На вкладке «Capability config» задайте параметры:
    a. Client authentication: OFF;
    b. Authorization: OFF;
    c. Authentication flow:
        i. Standard flow: ON;
        ii. Direct access grants: OFF;
        iii. Implicit flow: OFF;
        iv. Service accounts roles: OFF;
        v. OAuth 2.0 Device Authorization Grant: OFF;
        vi. OIDC CIBA Grant: OFF.

        Нажмите «Next».
4. На вкладке «Login settings» заполните:
    a. Valid redirect URIs: [URL административной панели]/* или *;
    b. Web Origins: [URL административной панели]/* или /*.

    Нажмите «Save».

**Шаг 2. Создание ролей**

Создайте следующие роли:

- название должно соответствовать значению переменной API_ADMIN_ROLE_NAME в AI BOX Hub (Роль пользователя, которая позволяет изменять данные на стороне AI BOX Hub API (CUD). По умолчанию admin_aibox
- название должно соответствовать значению переменной API_SERVICE_ROLE_NAME в AI BOX Hub (Роль пользователя, которая позволяет взаимодействовать с AI BOX Hub API). По умолчанию service

В консоли Keycloak → панель Realm Roles. Далее для каждой из перечисленных выше ролей необходимо нажать Create Role и выполнить следующие действия.

Заполните поля:

a. Role name: название соответствующей роли;

b. Description (опционально).

Нажмите «Save».

**Шаг 3. Создание пользователя-администратора AI BOX**

1. В консоли Keycloak → панель Users → Add user.
2. Заполните поля:
   a. Username: admin_ui_aibox;
   b. Email (опционально);
   c. Required user action (опционально).

   Нажмите «Create».
3. Для задания пароля: вкладка Credentials → Set Password:
   a. Вводится пароль;
   b. Настраивается флаг «Temporary»: включено/выключено по необходимости смены пароля при первом входе.

   Пароль задан.
4. Назначение ролей пользователю:

   Необходимо выдать созданные роли на шаге 2.
   a. перейти на вкладку Role Mapping
   b. Для каждой роли нажать Assign role → выбрать соответствующую роль→ «Assign».

Пользователю назначены необходимые роли.

**Шаг 4. Переопределение browser flow**

Для того чтобы пользователи, не имеющие роли admin_aibox, не могли авторизоваться в административной панели AI BOX, необходимо создать и применить новый поток аутентификации – browser flow с проверкой роли.

*1. Открытие и копирование потока browser*

Перейти в раздел «Configure» → «Authentication» и открыть вкладку «Flows». Открыт список потоков аутентификации.

В списке выбрать поток с именем «browser», открыть меню действий (Action) и выбрать пункт «Duplicate». Открыто окно копирования потока.

Ввести имя нового потока, например, «browser-role-check», и нажать «Duplicate». Создан новый поток аутентификации на основе базового browser flow.

*2. Настройка нового потока* browser-role-check

1. Добавьте новый sub-flow для шага «browser with role check forms», нажав на кнопку «Add sub-flow» в меню.
2. Введите имя нового потока, например, «Check role after form», и нажмите кнопку «Add».
3. В Requirement у добавленного sub-flow выберите «Conditional» и нажмите кнопку «Save».
4. Для нового sub-flow выберите в меню пункт «Add condition». В открывшемся окне выберете «Condition - user role» и нажмите кнопку «Add».
5. Для добавленного condition в Requirement выберите «Required». В настройках укажите:
    а. Alias: admin-role-missed
    б. User role: administrator
    в. Negate output: ON
6. Для sub-flow «Check role after form» добавьте шаг. Для этого выберите в меню пункт «Add step». В открывшемся окне выберете «Deny access» и нажмите кнопку «Add».
7. Для добавленного шага в Requirement выберите «Required».
8. Добавьте новый sub-flow «Cookie role check», нажав на кнопку «Add sub-flow» в меню.
9. В Requirement у добавленного sub-flow выберите «Alternative» и нажмите кнопку «Save».
10. Для sub-flow «Cookie role check» добавьте sub-flow «Check role after cookies», повторив шаги 4-10, изменив имя в шаге 5.
11. Перенесите шаг «Cookie» из основного потока «browser-role-check» в созданный sub-flow «Cookie with role check».

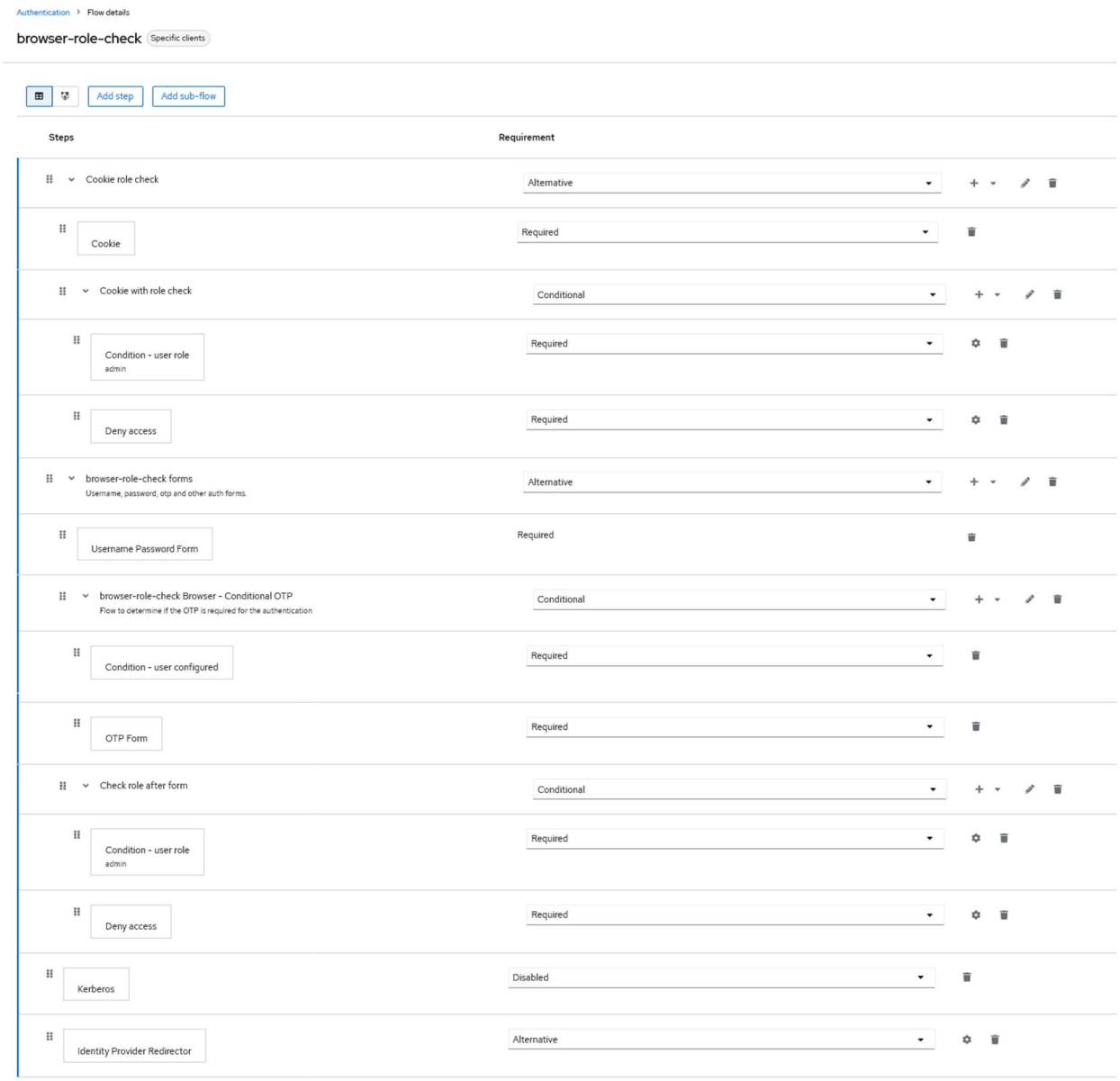


**Рисунок 1 –** пример настроенного потока аутентификации browser-role-check

*3. Настройка клиента*

1. Перейдите в список клиентов. И выберете созданного ранее клиента admin_ui_aibox.
2. Перейдите на вкладку «Advanced». В разделе «Authentication flow overrides» в настройке «Browser Flow» выберете созданный поток «browser-role-check». Нажмите кнопку «Save».

*Результат*

Создан новый поток аутентификации «browser with role check», который:

- запрещает авторизацию пользователям без роли admin_aibox;
- используется для входа в административную панель AI BOX;
- обеспечивает базовую защиту административного интерфейса от несанкционированного доступа.

## 2.4 Развёртывание сервиса Admin UI

1. Выберите любое удобное место и извлеките файл из архива client-docker-image. Установите docker-образ Admin UI с помощью команды в терминале: sudo docker load -i admin-client.tar.
2. Выберите любое удобное место в качпаестве рабочего каталога и извлеките файлы из архива для развертывания Admin UI.
3. Переименуйте файл .env.example в .env и при необходимости скорректируйте значения согласно таблице (Таблица 3).
4. Создайте директорию, указанную в параметре NGINX_CERTS_PATH из Таблица 3 и положите в неё файлы SSL сертификатов cert.pem и key.key.
5. В рабочей директории проекта выполните команду: docker compose up -d --build.

**Таблица 3. Переменные для Admin UI**

| Название | Описание | Пример |
|---|---|---|
| **Конфигурация сети** | | |
| ADMIN_PANEL_HOST | Название хоста, на котором будет доступен веб-сервис | aibox-admin-ui |
| ADMIN_PANEL_HTTPS_PORT | HTTPS-порт, на котором будет доступен веб-сервис | 443 |
| NGINX_CERTS_PATH | Путь к готовым сертификатам nginx | ./nginx/cert |
| **Бекенд AI BOX Hub** | | |
| HUB_URL | Адрес сервиса AI BOX Hub API | https://aibox.hub:8085 |
| **Авторизация** | | |

| Название | Описание | Пример |
|---|---|---|
| KEYCLOAK_URL | Адрес сервиса Keycloak | https://aibox.hub:8080 |
| KEYCLOAK_REALM | Realm для веб-сервиса | master |
| KEYCLOAK_CLIENT_ID | Идентификатор для веб-сервиса | admin-ui-client |

# 3. УПРАВЛЕНИЕ УЧЁТНЫМИ ЗАПИСЯМИ

Для того, чтобы убедиться, что AI BOX установлен корректно необходимо выполнить следующие действия (Таблица 4).

**Таблица 4. Проверка корректности установки AI BOX**

| № | Действие | Результат |
|---|---|---|
| 1. | Выполнить на сервере с развернутым дистрибутивом AI BOX команду:<br>`docker ps --format "table {{.Names}}\t{{.Status}}\t{{.Image}}"` | Выведется список сервисов AI BOX в статусе Up:<br>bot-api<br>nginx<br>mongo-express<br>mongo-db<br>chroma-db<br>ai-monitoring-prometheus<br>ai-monitoring-node-exporter<br>ai-monitoring-blackbox-exporter<br>ai-monitoring-cadvisor<br>keycloak<br>admin-client |
| 2. | Выполнить с рабочего места администратора GET-запрос:<br>https://{SERVER_HOST}:{API_PORT}/chain/health<br>С Basic авторизацией:<br>- username: {API_USER}<br>- password: {API_PASSWORD} | Код результата 200<br>{<br>"status": "ok"<br>} |
| 3. | Выполнить первый вход в панель администрирования AI BOX, используя учетную запись admin_ui_aibox | Авторизация прошла успешно. Загрузилась панель администрирования AI BOX с данными. |