



ФИЛОСОФИЯ.ИТ  
РОСАТОМ

Общество с ограниченной ответственностью «Философия.ИТ»

Россия, 107023, г. Москва, ул. Измайловский Вал, д. 30  
Тел.: +7 (495) 988-37-38, факс: +7 (495) 988-37-38,  
e-mail: customer@fil-it.ru

ИНН 7713728490  
КПП 771901001  
ОГРН 1117746379145

# **ПОШАГОВАЯ ИНСТРУКЦИЯ ПО РАЗВЕРТЫВАНИЮ ЭКЗЕМПЛЯРА ПО**

**ПЛАТФОРМА ИСКУССТВЕННОГО ИНТЕЛЛЕКТА «КОГНИТРОН»**

г. Москва

## **Оглавление**

<b>1.</b>	<b>Общие положения.....</b>	<b>3</b>
<b>1.1.</b>	<b>Полное наименование программы для ЭВМ, обозначение .....</b>	<b>3</b>
<b>1.2.</b>	<b>Назначение системы .....</b>	<b>3</b>
<b>1.3.</b>	<b>Разработчик системы .....</b>	<b>3</b>
<b>1.4.</b>	<b>Назначение документа .....</b>	<b>3</b>
<b>2.</b>	<b>Требования к аппаратному обеспечению.....</b>	<b>4</b>
<b>3.</b>	<b>Требования к программному обеспечению.....</b>	<b>5</b>
<b>4.</b>	<b>Установка ПО .....</b>	<b>6</b>
<b>4.1.</b>	<b>Установка зависимого ПО.....</b>	<b>6</b>
<b>4.2.</b>	<b>Размещение дистрибутива и разворачивание файлов .....</b>	<b>7</b>
<b>4.3.</b>	<b>Импорт образов контейнеров.....</b>	<b>7</b>
<b>4.4.</b>	<b>Настройка окружения (.env).....</b>	<b>8</b>
<b>4.5.</b>	<b>Установка сертификатов TLS .....</b>	<b>8</b>
<b>4.6.</b>	<b>Запуск и остановка системы .....</b>	<b>9</b>
<b>5.</b>	<b>Настройка аутентификации.....</b>	<b>9</b>
<b>6.</b>	<b>Доступ к системе.....</b>	<b>14</b>
<b>7.</b>	<b>Контактная информация.....</b>	<b>14</b>

## **1. Общие положения**

### **1.1. Полное наименование программы для ЭВМ, обозначение**

Полное наименование Программы для ЭВМ: Платформа искусственного интеллекта «Когнитрон» – далее по тексту Система.

### **1.2. Назначение системы**

«Когнитрон» - комплексное программное решение, объединяющее инструменты обучения и развертывания моделей искусственного интеллекта и создания ИИ-помощников и ИИ-агентов. Платформа поддерживает работу с большими языковыми моделями (LLM) модальности любого типа: текст, изображение, аудио, видео. Решение выступает в роли среды, обеспечивающей коммуникацию и координацию ИИ-агентов между собой, и средства интеграции с внешними ИТ-системами.

### **1.3. Разработчик системы**

Полное наименование: Общество с ограниченной ответственностью «Философия.ИТ».

Сокращенное наименование: ООО «Философия.ИТ».

### **1.4. Назначение документа**

Настоящий документ входит в комплект эксплуатационной документации по Платформе искусственного интеллекта «Когнитрон» и содержит пошаговую инструкцию для развертывания Системы.

## **2. Требования к аппаратному обеспечению**

Для разворачивания комплекса необходимо подготовить сервера, который должен отвечать следующим характеристикам:

*a) Сервер LLM*

- CPU: 16 ядер;
- Оперативная память (MEM): 64 ГБ;
- Накопитель (SSD): 100 ГБ;
- GPU: NVIDIA A100 / H100, 80 ГБ.

*б) Бэкэнд-сервер*

- CPU: 16 ядер;
- Оперативная память (MEM): 32 ГБ;
- Накопитель (SSD): 300 ГБ.

### **3. Требования к программному обеспечению**

Для разворачивания комплекса на серверах необходимо установить следующие компоненты:

*a) Сервер LLM*

- ОС: РЭДОС 8, Astra Linux 1.8 и более поздние выпуски;
- ПО: Docker Engine, Docker compose, NVIDIA Container Toolkit, NVIDIA GPU driver.

*б) Бэкэнд-сервер*

- ОС: РЭДОС 8, Astra Linux 1.8 и более поздние выпуски
- ПО: Docker Engine, Docker compose.

## 4. Установка ПО

Развертывание системы выполняется путем импортирования образов контейнеров и запуска контейнеров с помощью файла конфигурации `docker compose`.

Для выполнения установки необходим доступ к операционной системе с правами администратора (root). Необходимо ознакомится с документацией на необходимые компоненты по ссылкам:

- <https://developer.nvidia.com/cuda-downloads/>
- <https://docs.docker.com/engine/install/ubuntu/>
- <https://docs.nvidia.com/datacenter/cloud-native/container-toolkit/latest/install-guide.html>
- <https://catalog.ngc.nvidia.com/orgs/nvidia/containers/pytorch>

Подключитесь к серверу с помощью команды `ssh admin@сервер.домен.`

### 4.1. Установка зависимого ПО

#### Отключение автоматического обновления

Отключите автообновления командами:

```
#!/bin/bash
sudo systemctl stop unattended-upgrades
sudo systemctl disable unattended-upgrades
sudo systemctl mask unattended-upgrades
```

#### Установка драйвера NVIDIA

Установите драйвера NVIDIA следующими командами.

```
#!/bin/bash
wget https://developer.download.nvidia.com/compute/cuda/repos/ubuntu2204/x86_64/cuda-keyring_1.1-1_all.deb
sudo dpkg -i cuda-keyring_1.1-1_all.deb
sudo apt update
sudo apt install -y nvidia-driver-570-server-open
sudo reboot
```

Проверьте: `nvidia-smi`

#### Установка Docker и Docker Compose

Для установки Docker и Docker Compose выполните следующие команды.

```
#!/bin/bash
sudo apt update
sudo apt install -y ca-certificates curl
sudo install -m 0755 -d /etc/apt/keyrings
sudo curl -fsSL https://download.docker.com/linux/ubuntu/gpg -o /etc/apt/keyrings/docker.asc
sudo chmod a+r /etc/apt/keyrings/docker.asc
```

```
echo "deb [arch=$(dpkg --print-architecture) signed-by=/etc/apt/keyrings/docker.asc] https://download.docker.com/linux/ubuntu $(. /etc/os-release && echo "${UBUNTU_CODENAME:-$VERSION_CODENAME}") stable" | sudo tee /etc/apt/sources.list.d/docker.list > /dev/null
sudo apt update
sudo apt install -y docker-ce docker-ce-cli containerd.io docker-buildx-plugin docker-compose-plugin
sudo systemctl enable docker
sudo systemctl start docker
sudo usermod -aG docker $USER
```

Проверьте: `docker --version && docker compose version`

### Установка NVIDIA Container Toolkit

```
#!/bin/bash

curl -fsSL https://nvidia.github.io/libnvidia-container/gpgkey | sudo gpg --dearmor -o /usr/share/keyrings/nvidia-container-toolkit-keyring.gpg
curl -s -L https://nvidia.github.io/libnvidia-container/stable/deb/nvidia-container-toolkit.list | sed 's#deb https://#deb [signed-by=/usr/share/keyrings/nvidia-container-toolkit-keyring.gpg] https://#g' | sudo tee /etc/apt/sources.list.d/nvidia-container-toolkit.list
sudo apt update
sudo apt install -y nvidia-container-toolkit
sudo nvidia-ctk runtime configure --runtime=docker
sudo systemctl restart docker
```

Проверьте: `docker run --gpus all -it --rm nvcr.io/nvidia/cuda:12.3.1-base-ubuntu22.04 nvidia-smi`

## 4.2. Размещение дистрибутива и разворачивание файлов

Скопируйте дистрибутив на сервер: `scp cognitron-*.tgz admin@сервер:/opt/cognitron-dist`.

На сервере:

```
#!/bin/bash
mkdir -p /opt/cognitron
tar -xzvf /opt/cognitron-dist/cognitron-config.tgz -C /opt/cognitron
tar -xzvf /opt/cognitron-dist/cognitron-vllm.tgz -C /opt/cognitron/vllm/cache
tar -xzvf /opt/cognitron-dist/cognitron-infinity.tgz -C /opt/cognitron/infinity/cache
```

## 4.3. Импорт образов контейнеров

Образы контейнеров поставляются в виде файла архива `container_images.tgz`

Для импорта образов контейнеров выполните команду:

```
#!/bin/bash
docker load -i /opt/cognitron-dist/cognitron-images.tgz
```

## 4.4. Настройка окружения (.env)

Отредактируйте файл `/opt/cognitron/.env`

Перечень переменных окружения

Имя переменной	Значение	Примечание
COGNITRON_URL_HOST	cognitron.domain.local	Указать полное DNS имя сервера, по которому будет осуществляться доступ к системе
COGNITRON_DB_DATABASE	cognitron	Имя базы данных
COGNITRON_DB_USERNAME	cognitron	Имя пользователя доступа к базе данных
COGNITRON_DB_PASSWORD	**password**	Пароль доступа к базе данных
COGNITRON_OS_USERNAME	cognitron	Имя пользователя доступа к OpenSearch
COGNITRON_OS_PASSWORD	**password**	Пароль доступа к OpenSearch
COGNITRON_KEYCLOAK_PREFIX	/auth	Префикс доступа к консоли Keycloak
COGNITRON_KEYCLOAK_ADMIN	keycloak	Имя администратора Keycloak
COGNITRON_KEYCLOAK_ADMIN_PASSWORD	**password**	Пароль администратора Keycloak
COGNITRON_KEYCLOAK_DB_DATABASE	keycloak	Имя базы данных Keycloak
COGNITRON_KEYCLOAK_DB_USERNAME	keycloak	Имя пользователя доступа к базе данных Keycloak
COGNITRON_KEYCLOAK_DB_PASSWORD	**password**	Пароль доступа к базе данных Keycloak
COGNITRON_PGADMIN_PREFIX	/pgadmin	Префикс доступа к консоли PGadmin
COGNITRON_PGADMIN_DEFAULT_EMAIL	admin@pgadmin.com	Имя администратора PGadmin
COGNITRON_PGADMIN_DEFAULT_PASSWORD	**password**	Пароль администратора PGadmin
COGNITRON_REDIS_PASSWORD	**password**	Пароль Redis
COGNITRON_VLLM_API_KEY	**api_key_64_characters**	Ключ доступа сервера vLLM
COGNITRON_INFINITY_API_KEY	**api_key_64_characters**	Ключ доступа сервера Infibity
COGNITRON_OPENSEARCH_PREFIX	/osds	Префикс доступа к консоли OpenSearch
COGNITRON_OPENSEARCH_PASSWORD	**password**	Пароль администратора OpenSearch

## 4.5. Установка сертификатов TLS

Разместите в файлах на сервере

Закрытый ключ в файле `/opt/cognitron/traefik/cert.key`

Цепочка сертификатов в файле `/opt/cognitron/traefik/cert.cer`

## 4.6. Запуск и остановка системы

Запуск и остановка системы выполняется командой `docker compose` в директории `/opt/cognitron`

Для запуска системы выполните команду:

```
cd /opt/cognitron; docker compose up -d
```

В результате выполнения команды будут созданы контейнеры в целевой конфигурации и будет запущена система с автоматически запуском при загрузке операционной системы.

Для остановки системы и удаления контейнеров выполните команду:

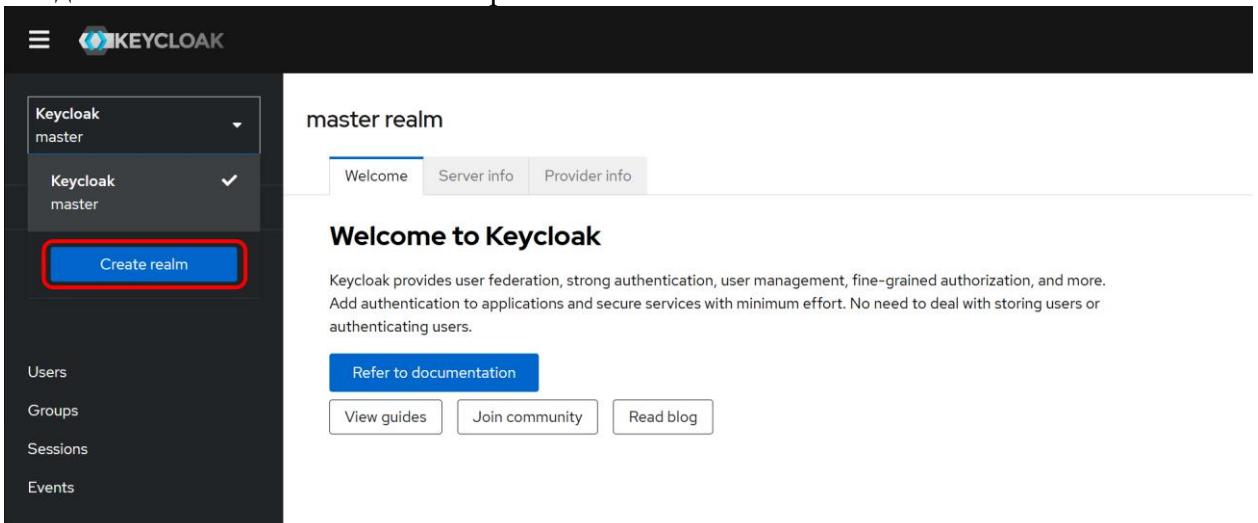
```
cd /opt/cognitron; docker compose down
```

В результате система будет остановлена, контейнеры будут удалены, автоматического запуска системы при загрузке операционной системы не будет выполняться.

## 5. Настройка аутентификации

Откройте консоль управления Keycloak по ссылке <https://cognitron.domain.local/auth> (используйте значения переменных `COGNITRON_URL_HOST` и `COGNITRON_KEYCLOAK_PREFIX`)

В качестве имени пользователя и пароля используйте значения из переменных `COGNITRON_KEYCLOAK_ADMIN` и `COGNITRON_KEYCLOAK_ADMIN_PASSWORD`.  
Создайте новый realm в меню выбора realm



The screenshot shows the Keycloak administration interface. On the left, a sidebar menu has a dropdown set to 'Keycloak master'. Underneath it, another dropdown is also set to 'Keycloak master'. At the bottom of this sidebar, a blue button labeled 'Create realm' is highlighted with a red box. The main content area is titled 'master realm' and contains the 'Welcome to Keycloak' message. Below this, there is a brief description of Keycloak's features and links to 'Refer to documentation', 'View guides', 'Join community', and 'Read blog'.

Установите имя нового realm `cognitron`

Create realm

A realm manages a set of users, credentials, roles, and groups. A user belongs to and logs into a realm. Realms are isolated from one another and can only manage and...

Resource file

Drag a file here or browse to upload

Browse... Clear

Upload a JSON file

Realm name \*

cognitron

Enabled

On

Create Cancel

Перейдите в пункт Clients и выберите пункт Import client

Clients

Clients are applications and services that can request authentication of a user. [Learn more](#)

Clients list Initial access token Client registration

Search for client Create client Import client Refresh

Client ID	Name	Type	Description
account	client_account	OpenID Connect	-
account-console	client_account-console	OpenID Connect	-
admin-cli	client_admin-cli	OpenID Connect	-
broker	client_broker	OpenID Connect	-

Загрузите JSON файл следующего содержимого.

```
{
  "clientId": "cognitron-frontend",
  "name": "",
  "description": "",
  "rootUrl": "",
  "adminUrl": "",
  "baseUrl": "",
  "surrogateAuthRequired": false,
  "enabled": true,
  "alwaysDisplayInConsole": false,
  "clientAuthenticatorType": "client-secret",
  "redirectUris": [
    "/*"
  ],
  "webOrigins": [
    "*"
  ],
  "notBefore": 0,
  "bearerOnly": false,
  "consentRequired": false,
  "standardFlowEnabled": true,
  "implicitFlowEnabled": false,
  "directAccessGrantsEnabled": true,
  "serviceAccountsEnabled": false,
  "publicClient": true,
  "frontchannelLogout": true,
  "protocol": "openid-connect",
  "attributes": {
    "client.secret.creation.time": "1729624683",
    "client.introspection.response.allow.jwt.claim.enabled": "false",
  }
}
```

```
"post.logout.redirect.uris": "+",
"oauth2.device.authorization.grant.enabled": "false",
"use.jwks.url": "false",
"backchannel.logout.revoke.offline.tokens": "false",
"use.refresh.tokens": "true",
"exclude.session.state.from.auth.response": "false",
"realm_client": "false",
"oidc.ciba.grant.enabled": "false",
"client.use.lightweight.access.token.enabled": "false",
"exclude.issuer.from.auth.response": "true",
"backchannel.logout.session.required": "true",
"client_credentials.use_refresh_token": "false",
"acr.loa.map": "{}",
"require.pushed.authorization.requests": "false",
"tls.client.certificate.bound.access.tokens": "false",
"display.on.consent.screen": "false",
"token.response.type.bearer.lower-case": "false"
},
"authenticationFlowBindingOverrides": {},
"fullScopeAllowed": true,
"nodeReRegistrationTimeout": -1,
"defaultClientScopes": [
  "web-origins",
  "acr",
  "profile",
  "roles",
  "basic",
  "email"
],
"optionalClientScopes": [
  "address",
  "phone",
  "organization",
  "offline_access",
  "micropause-jwt"
],
"access": {
  "view": true,
  "configure": true,
  "manage": true
}
}
```

The screenshot shows the Keycloak interface for importing a client. On the left, a sidebar menu is visible with options like Manage, Clients, Client scopes, Realm roles, Users, Groups, Sessions, Events, Configure, Realm settings, Authentication, Identity providers, and User federation. The 'Clients' option is selected. The main content area is titled 'Import client' and contains the sub-tutorial 'Clients are applications and services that can request authentication of a user.' A red box highlights a code editor where a JSON configuration file is being uploaded. The file content is as follows:

```
1  {
2   "clientId": "cognitron-frontend",
3   "name": "",
4   "description": "",
5   "rootUrl": "",
6   "adminUrl": "",
7   "baseUrl": ""
```

Below the code editor, there are input fields for 'Client ID' (set to 'cognitron-frontend'), 'Name' (empty), 'Description' (empty), and a toggle switch for 'Always display in UI' (set to 'Off'). The 'Type' is set to 'openid-connect'. Under 'Client authentication', the toggle switch is set to 'Off'. Under 'Authorization', the toggle switch is set to 'Off'. In the 'Authentication flow' section, 'Standard flow' and 'Direct access grants' are checked, while 'Implicit flow', 'Service accounts roles', 'OAuth 2.0 Device Authorization Grant', and 'OIDC CIBA Grant' are unchecked. At the bottom are 'Save' and 'Cancel' buttons, with 'Save' also highlighted by a red box.

В разделе Realms roles создайте роль с именем cognitron-system-admin

The screenshot shows the Keycloak interface for creating a new role. The sidebar menu is identical to the previous screenshot, with 'Realm roles' selected. The main content area is titled 'Create role' and shows the 'Role name' field containing 'cognitron-system-admin', which is also highlighted by a red box. Below it is a 'Description' field with an empty text area. At the bottom are 'Save' and 'Cancel' buttons, with 'Save' highlighted by a red box.

В разделе Users создайте пользователя с произвольным именем, установите значение Email verified

Required user actions Select action

Email verified On

General

Username \* administrator

Email Email

First name First name

Last name Last name

Groups Join Groups

На закладке Credentials задайте пользователю пароль, снимите значение Temporary

Enabled

Details Credentials Role mapping Groups Consents Identity provider links Sessions

Set password for administrator

Password \* .....

Password confirmation \* .....

Temporary Off

Save Cancel

На закладке Role mapping присвойте пользователю роль cognitron-system-admin, для чего выбрать фильтр ролей realms и выбрать роль cognitron-system-admin.

The screenshot shows the Keycloak interface for managing users. A specific user named 'administrator' is selected. The 'Role mapping' tab is currently active. A modal window titled 'Assign roles to administrator' is open, listing available realm roles. The role 'cognitron-system-admin' is checked and highlighted with a red box. The 'Assign' button at the bottom left of the modal is also highlighted with a red box.

## 6. Доступ к системе

В браузере перейдите по адресу <https://cognitron.domain.local/auth> (используйте значения переменной COGNITRON\_URL\_HOST)

Для входа используйте учетные данные пользователя с ролью cognitron-system-admin, созданные на этапе настройки аутентификации.

## 7. Контактная информация

В случае возникновения вопросов по настоящей инструкции просим обратиться к следующим специалистам:

Белоусов Олег Викторович,  
Директор по искусственному интеллекту  
+7 (916) 904 30 88,  
[obelousov@fil-it.ru](mailto:obelousov@fil-it.ru)

Русанов Николай Константинович,  
Системный архитектор  
+7(916) 603 76 15,  
[nrusanov@fil-it.ru](mailto:nrusanov@fil-it.ru)