


**ПРОГРАММНОЕ ОБЕСПЕЧЕНИЕ
«ИСТИМА-А»**

Инструкция по установке

2024 г.



СОДЕРЖАНИЕ

1	Требования к конфигурации сервера	3
2	Состав пакета установки.....	3
3	Подготовка рабочей директории	4
4	Установка программных компонентов	5
4.1	Установка драйвера NVIDIA.....	5
4.2	Установка Docker	5
4.3	Установка PostgreSQL	5
5	Настройка программных компонентов.....	5
5.1	Настройка PostgreSQL.....	5
5.2	Настройка Docker.....	6

1 Требования к конфигурации сервера

Минимальные аппаратные требования к обеспечению работы ИСТИМА-А приведены в таблице 1.

Таблица 1

Компонент	Минимальные требования
Количество потоков центрального процессора, шт	8
Производительность процессора	2.9 ГГц
Объем оперативной памяти	32 Гбайт DDR4
Производительность тензорных ядер GPU	60 Тфлопс
Объем оперативной памяти GPU	6 Гбайт DDR6
Сетевой интерфейс	Fast Ethernet, Gigabit Ethernet
Тип разъема сетевого интерфейса	RJ-45
Потребляемая мощность	500 Вт

На момент написания инструкции сервер функционирует на базе операционной системы Ubuntu 20.04. Сервер должен быть подключен к локальной сети и иметь доступ в сеть Интернет. Инсталляция программного обеспечения требует наличия минимум 30 ГБ свободного места на диске.

2 Состав пакета установки

В качестве комплекта для инсталляции передаётся файл лицензии **lic**, установочные файлы **ista-keeper-install-build.sh** и **ista-keeper-install-engine.sh**, а также архив **distrib.tar**. Описание назначения файлов приведено в таблице 2.

Таблица 2

Установочный файл	Назначение
ista-keeper-install-build.sh	Создает основную папку проекта ista-keeper, в которой содержатся все необходимые установочные скрипты, а также различные файлы, необходимые для функционирования программы (исполняемые, двоичные, конфигурационные и др.)
ista-keeper-install-engine.sh	Добавляет в папку проекта необходимые для работы файлы
distrib.tar	Содержит драйвер NVIDIA и tar-файлы с Docker-образами
lic	Файл лицензии

3 Подготовка рабочей директории

Поместить установочные файлы в директорию home/user.

Выполнить:

```
sudo bash ista-keeper-install-build.sh  
sudo bash ista-keeper-install-engine.sh
```

Далее, находясь в директории home/user, распаковать tar-архив и переместить распакованную папку в основную папку проекта с помощью команд:

```
tar -xf distrib.tar  
sudo mv distrib /etc/ista-keeper/
```

Переместить файл лицензии в основную папку проекта:

```
sudo mv lic /etc/ista-keeper/
```

4 Установка программных компонентов

4.1 Установка драйвера NVIDIA

Выполнить скрипт установки NVIDIA драйвера:

```
bash scripts/install/nvidia_install.sh
```

Подтверждением успешной установки является вывод `nvidia-smi`(вызывается автоматически в конце установки).

4.2 Установка Docker

Выполнить скрипт установки Docker:

```
bash scripts/install/docker_install.sh
```

4.3 Установка PostgreSQL

Выполнить скрипт установки PostgreSQL 10:

```
bash scripts/install/postgresql_install.sh
```

5 Настройка программных компонентов

5.1 Настройка PostgreSQL

Существующий скрипт обеспечивает автоматическую настройку базы данных PostgreSQL 10. В терминале выполнить:

```
bash scripts/install/configure_db.sh
```

5.2 Настройка Docker

Распаковка и установка Docker-образов, ранее помещенных в папку distrib:

```
bash scripts/install/get_containers.sh
```

После установки перейти в директорию `ista-keeper/docker-compose/` и **ВЫПОЛНИТЬ**:

```
cat > daemon.json << EOF
{
  "bridge": "none"
}
EOF
sudo mv daemon.json /etc/docker/
newgrp docker
```

Перезагрузите сервер.

Теперь необходимо удостовериться в успешности выполненной установки. Для запуска Docker из директории `/etc/ista-keeper/docker-compose/` выполнить:

```
docker compose up
```

Если все сделано верно, в терминале отобразятся строки с логами, циклично информирующие об успешной проверке лицензии и отсутствии входных данных с камер (рисунок 1).

```
docker-compose-main-1 | [2023-01-18 18:01:44,913] {(unknown file):0} DEBUG: Reading settings.ini...
docker-compose-main-1 | [2023-01-18 18:01:44,914] {(unknown file):0} INFO: Main started
docker-compose-main-1 | [2023-01-18 18:01:44,930] {(unknown file):0} INFO: License checked successfully
docker-compose-main-1 | [2023-01-18 18:01:44,931] {(unknown file):0} DEBUG: building streams META ...
docker-compose-main-1 | [2023-01-18 18:01:44,935] {(unknown file):0} DEBUG: building streams META done.
docker-compose-main-1 | [2023-01-18 18:01:44,935] {(unknown file):0} INFO: No alive connections are configured! Exiting...
docker-compose-main-1 exited with code 0
docker-compose-oss-alerts-1 | [x] Working -Ready to Send Done
docker-compose-oss-alerts-1 | [x] Working -Ready to Send Done
```

Рисунок 1 – Вид логов программы после успешной настройки

Нажмите Ctrl + C для остановки программы.