

Акционерное общество «Инжиниринговая компания «АЭМ-технологии» (АО «АЭМ-технологии»)

**АВТОМАТИЗИРОВАННАЯ СИСТЕМА ВИБРАЦИОННОГО МОНИТОРИНГА И
ПРОГНОЗИРОВАНИЯ ТЕХНИЧЕСКОГО СОСТОЯНИЯ И ДИАГНОСТИКИ
ДЕФЕКТОВ ДИНАМИЧЕСКОГО ОБОРУДОВАНИЯ**

**Инструкция по установке экземпляра программного обеспечения, предоставленного
для проведения экспертной проверки.**

На 8 листах

Санкт-Петербург
2026

Аннотация

В настоящем документе приведена инструкция по установке экземпляра программного обеспечения автоматизированной системы вибрационного мониторинга и прогнозирования технического состояния и диагностики дефектов динамического оборудования для проведения экспертной проверки.

Оглавление

ОГЛАВЛЕНИЕ.....	3
1. ОБЩИЕ СВЕДЕНИЯ	4
2. ТЕХНИЧЕСКИЕ ТРЕБОВАНИЯ К СЕРВЕРУ ДЛЯ РАЗВЕРТЫВАНИЯ СИСТЕМЫ	5
3. ИНСТРУКЦИЯ ПО РАЗВЕРТЫВАНИЮ СИСТЕМЫ.....	6
3.1 Развертывание серверной части.....	6
3.2 Установка и настройка баз данных.....	6
3.3 Обновление СВД «ВЗОР»	7
3.4 Настройка резервного копирования	7
3.5 АРІ-РУКОВОДСТВО	7

1. ОБЩИЕ СВЕДЕНИЯ

Полное наименование: автоматизированная система вибрационного мониторинга и прогнозирования технического состояния и диагностики дефектов динамического оборудования».

Краткие наименования: СВД «ВЗОР», Система.

Владелец: Акционерное общество «Инжиниринговая компания «АЭМ-технологии» (АО «АЭМ-технологии»)

2. ТЕХНИЧЕСКИЕ ТРЕБОВАНИЯ К СЕРВЕРУ ДЛЯ РАЗВЕРТЫВАНИЯ СИСТЕМЫ

СВД «ВЗОР» разворачивается на сервере со следующими характеристиками:

- процессор: AMD EPYC 7313 (2 шт);
- оперативная память: 32 ГБ DDR4 3200 МГц (2 шт);
- накопитель: SSD 960 ГБ, NVMe/PCIe Gen4;
- сетевая карта: Intel x550, 2*RJ45 10G NIC;
- платформа: серверная платформа DEPO.

3. ИНСТРУКЦИЯ ПО РАЗВЕРТЫВАНИЮ СИСТЕМЫ.

3.1 РАЗВЕРТЫВАНИЕ СЕРВЕРНОЙ ЧАСТИ

Перед установкой необходимо убедиться, что сервер соответствует минимальным требованиям. На сервере с установленной Astra Linux должны присутствовать модули Docker и Docker Compose.

Для установки выполните команды: `sudo apt update && sudo apt upgrade -y`

`sudo apt install -y docker.io docker-compose-plugin`

Проверить корректность установки можно с помощью команд: `docker --version` `docker compose version`

После подготовки среды выполните разворачивание СВД «ВЗОР»:

1. Скопируйте файлы СВД «ВЗОР» на сервер и распакуйте архив `LimanISU.tar.xz` в рабочую директорию.
2. Запустите установочный скрипт: `chmod +x run.sh ./run.sh`
3. Дождитесь завершения установки и убедитесь, что все контейнеры работают корректно с помощью команды `docker ps`.
4. После успешного разворачивания откройте браузер и перейдите по адресу <http://server-ip:3000>.

3.2 Установка и настройка баз данных

Для работы СВД «ВЗОР» используются три базы данных: PostgreSQL, ClickHouse и MinIO. Они разворачиваются в контейнерах Docker с использованием `docker-compose` и автоматически настраиваются с помощью конфигурационных файлов.

Перед установкой баз данных убедитесь, что сервер соответствует минимальным требованиям и на нем установлены Docker и Docker Compose.

Для разворачивания баз данных используется конфигурационный архив, содержащий `docker-compose.yml` и необходимые настройки.

Запустите базы данных с помощью `docker-compose`.

3.3 ОБНОВЛЕНИЕ СВД «ВЗОР»

Для загрузки актуальной версии выполните:

```
cd ~/liman-isu
```

```
git pull origin main
```

```
./update.sh
```

Обновление СВД «ВЗОР» осуществляется путем замены исходного кода и повторного применения процесса обновления.

3.4 НАСТРОЙКА РЕЗЕРВНОГО КОПИРОВАНИЯ

Настроить автоматическое резервное копирование базы данных можно через crontab:

```
crontab -e
```

Создайте дампы базы данных PostgreSQL и сохраните его в каталоге /backup/ с датой в имени файла при помощи команды:

```
0 2 * * * /usr/bin/docker exec postgres pg_dump -U admin db >
/backup/db_$(date +%F).sql
```

Иницируйте процесс резервного копирования ClickHouse с использованием clickhouse-backup командой:

3.5 API-РУКОВОДСТВО

СВД «ВЗОР» предоставляет **RESTful API** для работы с данными и интеграции с внешними системами. Все запросы и ответы осуществляются в формате **JSON**.

Для работы с API необходима аутентификация. Для этого используется **API-ключ**, который передается в заголовках запросов.

Пример заголовка для авторизации:

Authorization: Bearer

API поддерживает формат данных **JSON** для обмена информацией между клиентом и сервером. Все запросы и ответы обрабатываются в этом формате.

Пример запроса с форматированием JSON:

```
{"id": 1,
```

"name": "admin",

"email": "admin@localhost.ru",

"roles": ["USER", "ADMIN"]}

API поддерживает следующие HTTP-методы:

- GET: получение данных (например, получения информации о пользователе или оборудования).
- POST: создание новых записей.
- PUT: обновление существующих данных.
- DELETE: удаление данных.