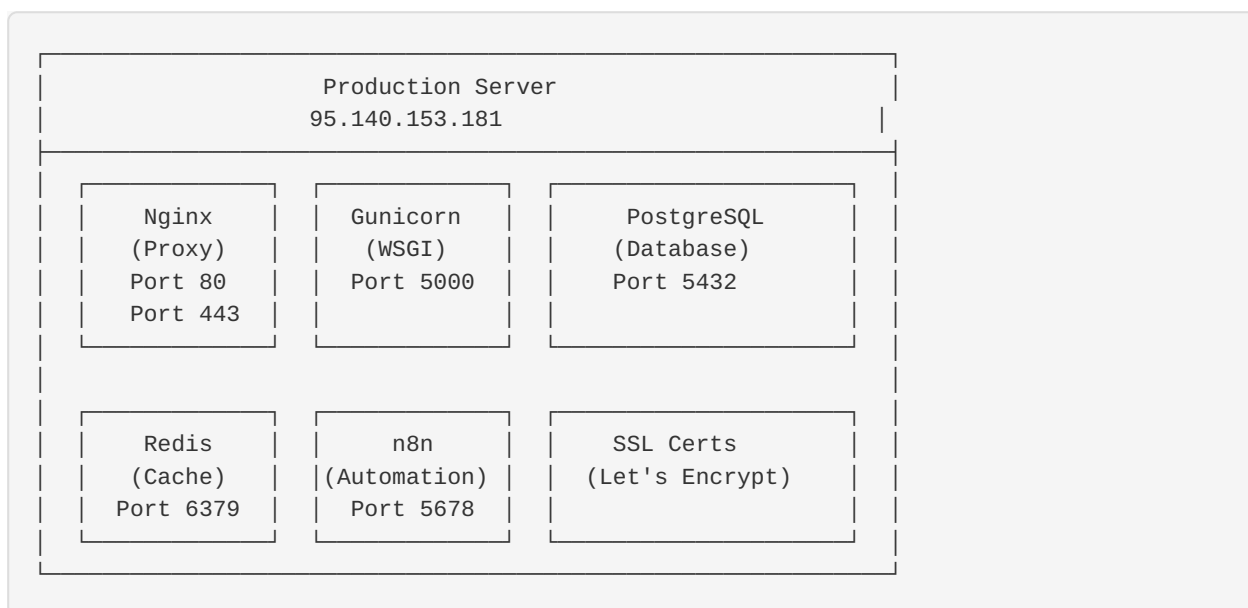


# Руководство по развертыванию BaiMuras Platform

## Обзор

Данное руководство содержит подробные инструкции по развертыванию BaiMuras Platform в продакшн среде, включая основной сайт, админ-панель и систему автоматизации.

## Архитектура развертывания



## Требования к серверу

### Минимальные требования

- **OS:** Ubuntu 20.04 LTS или новее
- **CPU:** 2 vCPU
- **RAM:** 4 GB
- **Storage:** 50 GB SSD
- **Network:** 100 Mbps

### Рекомендуемые требования

- **OS:** Ubuntu 22.04 LTS
- **CPU:** 4 vCPU
- **RAM:** 8 GB
- **Storage:** 100 GB SSD
- **Network:** 1 Gbps

# Подготовка сервера

## 1. Обновление системы

```
sudo apt update && sudo apt upgrade -y
sudo apt install -y curl wget git vim htop
```

## 2. Установка Python 3.11

```
sudo apt install -y software-properties-common
sudo add-apt-repository ppa:deadsnakes/ppa
sudo apt update
sudo apt install -y python3.11 python3.11-venv python3.11-dev
sudo apt install -y python3-pip
```

## 3. Установка PostgreSQL

```
sudo apt install -y postgresql postgresql-contrib
sudo systemctl start postgresql
sudo systemctl enable postgresql

# Создание пользователя и базы данных
sudo -u postgres psql
CREATE DATABASE baimuras_db;
CREATE USER baimuras_user WITH PASSWORD 'BaiMuras2025!@#';
GRANT ALL PRIVILEGES ON DATABASE baimuras_db TO baimuras_user;
\q
```

## 4. Установка Redis

```
sudo apt install -y redis-server
sudo systemctl start redis-server
sudo systemctl enable redis-server
```

## 5. Установка Nginx

```
sudo apt install -y nginx
sudo systemctl start nginx
sudo systemctl enable nginx
```

## 6. Установка Node.js (для n8n)

```
curl -fsSL https://deb.nodesource.com/setup_18.x | sudo -E bash -
sudo apt install -y nodejs
```

# Развертывание приложения

## 1. Клонирование репозитория

```
cd /var/www
sudo git clone https://github.com/ardakchapaev/baimuras.space.git
sudo chown -R www-data:www-data baimuras.space
cd baimuras.space
```

## 2. Создание виртуального окружения

```
sudo -u www-data python3.11 -m venv venv
sudo -u www-data ./venv/bin/pip install --upgrade pip
sudo -u www-data ./venv/bin/pip install -r requirements.txt
```

## 3. Настройка переменных окружения

```
sudo -u www-data cp .env.example .env
sudo -u www-data vim .env
```

Содержимое `.env` :

```
# Основные настройки
FLASK_ENV=production
SECRET_KEY=your-super-secret-key-here
DATABASE_URL=postgresql://baimuras_user:BaiMuras2025!@@localhost/baimuras_db

# Настройки почты
MAIL_SERVER=smtp.gmail.com
MAIL_PORT=587
MAIL_USE_TLS=True
MAIL_USERNAME=admin@baimuras.space
MAIL_PASSWORD=your-app-password

# Настройки Redis
REDIS_URL=redis://localhost:6379/0

# Настройки файлов
UPLOAD_FOLDER=/var/www/baimuras.space/uploads
MAX_CONTENT_LENGTH=16777216

# Настройки безопасности
WTF_CSRF_ENABLED=True
SESSION_COOKIE_SECURE=True
SESSION_COOKIE_HTTPONLY=True
```

## 4. Инициализация базы данных

```
sudo -u www-data ./venv/bin/flask db init
sudo -u www-data ./venv/bin/flask db migrate -m "Initial migration"
sudo -u www-data ./venv/bin/flask db upgrade
```

## 5. Создание директорий

```
sudo mkdir -p /var/www/baimuras.space/uploads
sudo mkdir -p /var/www/baimuras.space/logs
sudo chown -R www-data:www-data /var/www/baimuras.space/uploads
sudo chown -R www-data:www-data /var/www/baimuras.space/logs
```

## Настройка Gunicorn

### 1. Создание systemd сервиса

```
sudo vim /etc/systemd/system/baimuras.service
```

Содержимое файла:

```
[Unit]
Description=Gunicorn instance to serve BaiMuras
After=network.target

[Service]
User=www-data
Group=www-data
WorkingDirectory=/var/www/baimuras.space
Environment="PATH=/var/www/baimuras.space/venv/bin"
ExecStart=/var/www/baimuras.space/venv/bin/gunicorn --config gunicorn.conf.py
wsgi:application
ExecReload=/bin/kill -s HUP $MAINPID
Restart=always

[Install]
WantedBy=multi-user.target
```

### 2. Запуск сервиса

```
sudo systemctl daemon-reload
sudo systemctl start baimuras
sudo systemctl enable baimuras
sudo systemctl status baimuras
```

## Настройка Nginx

### 1. Создание конфигурации сайта

```
sudo vim /etc/nginx/sites-available/baimuras.space
```

Содержимое файла:

```

server {
    listen 80;
    server_name baimuras.space www.baimuras.space;
    return 301 https://$server_name$request_uri;
}

server {
    listen 443 ssl http2;
    server_name baimuras.space www.baimuras.space;

    ssl_certificate /etc/letsencrypt/live/baimuras.space/fullchain.pem;
    ssl_certificate_key /etc/letsencrypt/live/baimuras.space/privkey.pem;

    ssl_protocols TLSv1.2 TLSv1.3;
    ssl_ciphers ECDHE-RSA-AES256-GCM-SHA512:DHE-RSA-AES256-GCM-SHA512:ECDHE-RSA-AES256-
GCM-SHA384:DHE-RSA-AES256-GCM-SHA384;
    ssl_prefer_server_ciphers off;
    ssl_session_cache shared:SSL:10m;

    client_max_body_size 16M;

    location / {
        proxy_pass http://127.0.0.1:5000;
        proxy_set_header Host $host;
        proxy_set_header X-Real-IP $remote_addr;
        proxy_set_header X-Forwarded-For $proxy_add_x_forwarded_for;
        proxy_set_header X-Forwarded-Proto $scheme;
    }

    location /static {
        alias /var/www/baimuras.space/src/static;
        expires 1y;
        add_header Cache-Control "public, immutable";
    }

    location /uploads {
        alias /var/www/baimuras.space/uploads;
        expires 1y;
        add_header Cache-Control "public";
    }
}

```

## 2. Активация сайта

```

sudo ln -s /etc/nginx/sites-available/baimuras.space /etc/nginx/sites-enabled/
sudo nginx -t
sudo systemctl reload nginx

```

# Настройка SSL сертификатов

## 1. Установка Certbot

```

sudo apt install -y certbot python3-certbot-nginx

```

## 2. Получение сертификата

```
sudo certbot --nginx -d baimuras.space -d www.baimuras.space
```

## 3. Автоматическое обновление

```
sudo crontab -e
# Добавить строку:
0 12 * * * /usr/bin/certbot renew --quiet
```

# Развертывание админ-панели

## 1. Клонирование репозитория админ-панели

```
cd /var/www
sudo git clone https://github.com/ardakchapaev/baimuras-admin-hub.git
sudo chown -R www-data:www-data baimuras-admin-hub
```

## 2. Настройка виртуального окружения

```
cd baimuras-admin-hub
sudo -u www-data python3.11 -m venv venv
sudo -u www-data ./venv/bin/pip install -r requirements.txt
```

## 3. Создание systemd сервиса для админ-панели

```
sudo vim /etc/systemd/system/baimuras-admin.service
```

Содержимое:

```
[Unit]
Description=Gunicorn instance to serve BaiMuras Admin
After=network.target

[Service]
User=www-data
Group=www-data
WorkingDirectory=/var/www/baimuras-admin-hub
Environment="PATH=/var/www/baimuras-admin-hub/venv/bin"
ExecStart=/var/www/baimuras-admin-hub/venv/bin/gunicorn --bind 127.0.0.1:5001 --workers
3 app:app
ExecReload=/bin/kill -s HUP $MAINPID
Restart=always

[Install]
WantedBy=multi-user.target
```

## 4. Настройка Nginx для админ-панели

```
sudo vim /etc/nginx/sites-available/hub.baimuras.space
```

Содержимое:

```
server {
    listen 80;
    server_name hub.baimuras.space;
    return 301 https://$server_name$request_uri;
}

server {
    listen 443 ssl http2;
    server_name hub.baimuras.space;

    ssl_certificate /etc/letsencrypt/live/hub.baimuras.space/fullchain.pem;
    ssl_certificate_key /etc/letsencrypt/live/hub.baimuras.space/privkey.pem;

    location / {
        proxy_pass http://127.0.0.1:5001;
        proxy_set_header Host $host;
        proxy_set_header X-Real-IP $remote_addr;
        proxy_set_header X-Forwarded-For $proxy_add_x_forwarded_for;
        proxy_set_header X-Forwarded-Proto $scheme;
    }
}
```

## 5. Получение SSL сертификата для админ-панели

```
sudo certbot --nginx -d hub.baimuras.space
```

## Настройка n8n автоматизации

### 1. Установка n8n

```
sudo npm install -g n8n
```

### 2. Создание пользователя для n8n

```
sudo useradd -m -s /bin/bash n8n
sudo su - n8n
```

### 3. Настройка переменных окружения для n8n

```
vim ~/.bashrc
# Добавить:
export N8N_BASIC_AUTH_ACTIVE=true
export N8N_BASIC_AUTH_USER=admin@baimuras.space
export N8N_BASIC_AUTH_PASSWORD=Admin123!
export N8N_HOST=0.0.0.0
export N8N_PORT=5678
export N8N_PROTOCOL=http
```

### 4. Создание systemd сервиса для n8n

```
sudo vim /etc/systemd/system/n8n.service
```

Содержимое:

```
[Unit]
Description=n8n workflow automation
After=network.target

[Service]
Type=simple
User=n8n
ExecStart=/usr/bin/n8n start
Restart=always
Environment=N8N_BASIC_AUTH_ACTIVE=true
Environment=N8N_BASIC_AUTH_USER=admin@baimuras.space
Environment=N8N_BASIC_AUTH_PASSWORD=Admin123!
Environment=N8N_HOST=0.0.0.0
Environment=N8N_PORT=5678

[Install]
WantedBy=multi-user.target
```

### 5. Запуск n8n

```
sudo systemctl daemon-reload
sudo systemctl start n8n
sudo systemctl enable n8n
```

## Процедуры обновления

### 1. Обновление основного приложения

```
cd /var/www/baimuras.space
sudo -u www-data git pull origin main
sudo -u www-data ./venv/bin/pip install -r requirements.txt
sudo -u www-data ./venv/bin/flask db upgrade
sudo systemctl restart baimuras
```



## 2. Обновление админ-панели

```
cd /var/www/baimuras-admin-hub
sudo -u www-data git pull origin main
sudo -u www-data ./venv/bin/pip install -r requirements.txt
sudo systemctl restart baimuras-admin
```

## Мониторинг и логирование

### 1. Настройка логирования

```
sudo mkdir -p /var/log/baimuras
sudo chown www-data:www-data /var/log/baimuras
```

### 2. Ротация логов

```
sudo vim /etc/logrotate.d/baimuras
```

Содержимое:

```
/var/log/baimuras/*.log {
    daily
    missingok
    rotate 52
    compress
    delaycompress
    notifempty
    create 644 www-data www-data
    postrotate
        systemctl reload baimuras
    endscript
}
```

### 3. Мониторинг сервисов

```
# Проверка статуса всех сервисов
sudo systemctl status baimuras
sudo systemctl status baimuras-admin
sudo systemctl status n8n
sudo systemctl status nginx
sudo systemctl status postgresql
sudo systemctl status redis-server
```

## Резервное копирование

### 1. Скрипт резервного копирования базы данных

```
sudo vim /usr/local/bin/backup-baimuras.sh
```

Содержимое:

```
#!/bin/bash
BACKUP_DIR="/var/backups/baimuras"
DATE=$(date +%Y%m%d_%H%M%S)

mkdir -p $BACKUP_DIR

# Backup database
pg_dump -h localhost -U baimuras_user baimuras_db > $BACKUP_DIR/db_backup_$DATE.sql

# Backup uploads
tar -czf $BACKUP_DIR/uploads_backup_$DATE.tar.gz /var/www/baimuras.space/uploads

# Remove old backups (keep 30 days)
find $BACKUP_DIR -name "*.sql" -mtime +30 -delete
find $BACKUP_DIR -name "*.tar.gz" -mtime +30 -delete
```

## 2. Настройка cron для автоматического бэкапа

```
sudo chmod +x /usr/local/bin/backup-baimuras.sh
sudo crontab -e
# Добавить:
0 2 * * * /usr/local/bin/backup-baimuras.sh
```

## Устранение неполадок

### Общие проблемы

#### 1. Приложение не запускается

```
# Проверить логи
sudo journalctl -u baimuras -f
sudo tail -f /var/log/baimuras/error.log
```

#### 2. База данных недоступна

```
# Проверить статус PostgreSQL
sudo systemctl status postgresql
# Проверить подключение
sudo -u postgres psql -c "SELECT version();"

```

#### 3. Nginx ошибки

```
# Проверить конфигурацию
sudo nginx -t
# Проверить логи
sudo tail -f /var/log/nginx/error.log
```

## 4. SSL сертификаты

```
# Проверить статус сертификатов
sudo certbot certificates
# Обновить сертификаты
sudo certbot renew --dry-run
```

## Команды диагностики

```
# Проверка портов
sudo netstat -tlnp | grep :80
sudo netstat -tlnp | grep :443
sudo netstat -tlnp | grep :5000

# Проверка дискового пространства
df -h

# Проверка памяти
free -h

# Проверка процессов
ps aux | grep gunicorn
ps aux | grep nginx
```

## Поддержка

При возникновении проблем с развертыванием:

1. Проверьте логи сервисов
2. Убедитесь в правильности конфигурации
3. Проверьте доступность портов и сервисов
4. Обратитесь к документации или создайте Issue в репозитории

---

### Контакты для поддержки:

- Email: [admin@baimuras.space](mailto:admin@baimuras.space)
- GitHub Issues: <https://github.com/ardakchapaev/baimuras.space/issues>