

GARUDAHOKI WhatsApp Bot - Deployment Guide untuk Namecheap Hosting

Panduan Lengkap Deployment

Overview

Panduan ini akan membantu Anda men-deploy bot WhatsApp GARUDAHOKI ke Namecheap hosting dengan langkah-langkah detail dan troubleshooting.

Langkah 1: Persiapan Hosting & Domain

A. Pembelian Domain & Hosting

1. Login ke Namecheap

- Kunjungi <https://namecheap.com>
- Login ke akun Anda

2. Beli Domain

Contoh domain yang disarankan:

- botgarudahoki.com
- garudahoki-bot.com
- api-garudahoki.com

3. Pilih Hosting Plan

Minimum Requirements:

- Shared Hosting: Stellar Plus atau Business
- VPS: Pulsar atau Quasar (recommended)
- Specs minimal: 2GB RAM, 2 CPU cores, 50GB storage

B. Setup DNS & Domain

1. Konfigurasi DNS

...

2. Di Namecheap dashboard, pilih domain Anda

3. Klik "Manage" > "Advanced DNS"

4. Tambahkan A Record:

- Host: @
- Value: IP hosting Anda
- TTL: Automatic

5. Tambahkan CNAME untuk subdomain:

- Host: api
- Value: yourdomain.com
- TTL: Automatic

...

6. Verifikasi SSL Certificate

...

7. Di cPanel, cek "SSL/TLS Status"

8. Pastikan SSL aktif untuk domain utama dan subdomain

9. Jika belum, aktifkan Let's Encrypt SSL

...

Langkah 2: Upload Project ke Hosting

A. Persiapan File Project

1. Compress Project

```
```bash
cd /home/ubuntu/garudahoki_whatsapp_bot

Exclude node_modules dan file tidak perlu
tar -czf garudahoki-bot.tar.gz \
-exclude=node_modules \
-exclude=.git \
-exclude=logs \
-exclude=.env \
.
```
```

1. Buat Environment File

```
bash
# Buat .env file terpisah untuk production
cp .env.example .env.production
```

B. Upload via cPanel

1. Login ke cPanel

```
URL: https://yourdomain.com:2083
atau dari Namecheap dashboard > Hosting > Manage
```

2. Setup Directory Structure

...

3. Buka "File Manager"

4. Navigate ke /public_html

5. Buat folder struktur:

```
/public_html/
├─ api/ (untuk backend)
├─ admin/ (untuk admin panel)
└─ webhook/ (untuk webhook endpoint)
```

...

6. Upload & Extract Files

...

7. Upload garudahoki-bot.tar.gz ke /public_html/api/

8. Right-click file > “Extract”
9. Delete tar.gz file setelah extract
- ...

C. Setup File Permissions

```
# Via SSH atau File Manager
chmod 755 /public_html/api
chmod 644 /public_html/api/.env
chmod 755 /public_html/api/bin/www
```

Langkah 3: Setup Database PostgreSQL

A. Buat Database

1. **Via cPanel**
 - ...
2. Buka “PostgreSQL Databases”
3. Create Database:
 - Database Name: garudahoki_bot
4. Create User:
 - Username: garudahoki_user
 - Password: [generate strong password]
5. Add User to Database:
 - User: garudahoki_user
 - Database: garudahoki_bot
 - Privileges: ALL
 - ...

6. Catat Database Info

```
Host: localhost
Port: 5432
Database: cpanel_username_garudahoki_bot
Username: cpanel_username_garudahoki_user
Password: [your_password]
```

B. Import Database Schema

1. **Via phpPgAdmin**
 - ...
2. Buka “phpPgAdmin” di cPanel
3. Login dengan database credentials
4. Select database “garudahoki_bot”
5. Klik “SQL” tab
6. Copy-paste isi file database/schema.sql
7. Execute query
 - ...

8. Via SSH (jika tersedia)

```
bash
psql -h localhost -U cpanel_username_garudahoki_user -d cpanel_username_garudahoki_bot <
database/schema.sql
```

Langkah 4: Konfigurasi Environment Variables

A. Setup .env File

1. Edit .env di File Manager

```
bash
# Navigate ke /public_html/api/
# Edit .env file dengan konfigurasi production
```

2. Production Environment Configuration

```
```.env
Server Configuration
NODE_ENV=production
PORT=3000
HOST=0.0.0.0
```

# Database Configuration (sesuaikan dengan info dari cPanel)

```
DATABASE_URL=postgresql://cpanel_username_garudahoki_user:your_password@localhost:5432/cpanel_username_garudahoki_bot
```

# WhatsApp API Configuration

```
WHATSAPP_PROVIDER=sleekflow
WHATSAPP_PHONE_NUMBER_ID=your_phone_number_id
WHATSAPP_ACCESS_TOKEN=your_access_token
WHATSAPP_WEBHOOK_VERIFY_TOKEN=your_verify_token
WHATSAPP_WEBHOOK_SECRET=your_webhook_secret
```

# LiveChatInc Integration

```
LIVECHAT_LICENSE_ID=your_license_id
LIVECHAT_CLIENT_ID=your_client_id
LIVECHAT_CLIENT_SECRET=your_client_secret
LIVECHAT_REDIRECT_URI=https://yourdomain.com/auth/livechat/callback
```

# Admin Panel

```
ADMIN_USERNAME=admin
ADMIN_PASSWORD=your_secure_password
JWT_SECRET=your_jwt_secret_key_min_32_chars
```

# Bot Configuration

```
BOT_TIMEZONE=Asia/Jakarta
BOT_LANGUAGE=id
```

# Security

```
CORS_ORIGIN=https://yourdomain.com
RATE_LIMIT_WINDOW_MS=900000
RATE_LIMIT_MAX_REQUESTS=100
```

```
Logging
LOG_LEVEL=info
LOG_FILE=logs/bot.log
...
```

## B. Secure Environment File

```
Set proper permissions
chmod 600 .env
```



## Langkah 5: Install Dependencies & Start Application

### A. Install Node.js Dependencies

#### 1. Via SSH Terminal

```
bash
cd /public_html/api
npm install --production
```

#### 2. Jika SSH tidak tersedia, gunakan Node.js App di cPanel

```
...
```

#### 3. Buka “Node.js App” di cPanel

#### 4. Create App:

- Node.js version: 18.x atau terbaru
- Application mode: Production
- Application root: api
- Application URL: yourdomain.com/api
- Application startup file: server.js

#### 5. Klik “Create”

#### 6. Install dependencies via interface

```
...
```

### B. Setup Process Manager

#### 1. Menggunakan PM2 (jika SSH tersedia)

```
```bash
# Install PM2 globally
npm install -g pm2

# Start application
pm2 start ecosystem.config.js --env production

# Save PM2 configuration
pm2 save
```

```
# Setup auto-restart on reboot
pm2 startup
...
```

1. Menggunakan cPanel Node.js App

- ...
- 2. Di Node.js App interface
- 3. Klik "Start App"
- 4. Monitor status di dashboard
- ...

Langkah 6: Setup Web Server Configuration

A. Setup .htaccess untuk Apache

1. Buat .htaccess di /public_html/

```
``apache
# Redirect API calls to Node.js app
RewriteEngine On
RewriteCond %{REQUEST_URI} ^/api/
RewriteRule ^api/(.*)$ http://localhost:3000/$1 [P,L]

# Redirect admin panel
RewriteCond %{REQUEST_URI} ^/admin
RewriteRule ^admin/(.*)$ http://localhost:3000/admin$1 [P,L]

# Redirect webhook
RewriteCond %{REQUEST_URI} ^/webhook/
RewriteRule ^webhook/(.*)$ http://localhost:3000/webhook/$1 [P,L]

# Enable HTTPS redirect
RewriteCond %{HTTPS} off
RewriteRule ^(.*)$ https://%{HTTP_HOST}%{REQUEST_URI} [L,R=301]
...
```

B. Setup Nginx (jika menggunakan VPS dengan Nginx)

1. Buat Nginx config

```
``nginx
server {
    listen 80;
    server_name yourdomain.com www.yourdomain.com;
    return 301 https://$server_name$request_uri;
}

server {
    listen 443 ssl http2;
    server_name yourdomain.com www.yourdomain.com;
```

```

ssl_certificate /path/to/ssl/cert.pem;
ssl_certificate_key /path/to/ssl/private.key;

location /api/ {
    proxy_pass http://localhost:3000/;
    proxy_http_version 1.1;
    proxy_set_header Upgrade $http_upgrade;
    proxy_set_header Connection 'upgrade';
    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;
    proxy_cache_bypass $http_upgrade;
}

location /admin {
    proxy_pass http://localhost:3000/admin;
    proxy_http_version 1.1;
    proxy_set_header Upgrade $http_upgrade;
    proxy_set_header Connection 'upgrade';
    proxy_set_header Host $host;
    proxy_cache_bypass $http_upgrade;
}

location /webhook/ {
    proxy_pass http://localhost:3000/webhook/;
    proxy_http_version 1.1;
    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;
}

}
...

```

Langkah 7: Konfigurasi Webhook WhatsApp

A. Setup Webhook URL

1. Di Provider Dashboard (SleekFlow/Zixflow)

Webhook URL: `https://yourdomain.com/webhook/whatsapp`

Verify Token: [sesuai dengan WHATSAPP_WEBHOOK_VERIFY_TOKEN di `.env`]

2. Test Webhook

```

``bash
# Test webhook verification
curl -X GET "https://yourdomain.com/webhook/whatsapp?
hub.mode=subscribe&hub.challenge=test&hub.verify_token=your_verify_token"

```

Should return the challenge value

```

...

```

B. Verify Webhook Status

1. Check di Provider Dashboard

- Status harus "Connected" atau "Verified"
- Test dengan mengirim pesan ke bot

Langkah 8: Testing Deployment

A. Test Application Status

1. Check Application Health

```
```bash
Test API endpoint
curl https://yourdomain.com/api/health

Should return: {"status": "OK", "timestamp": "..."}
```
```

1. Test Admin Panel

- 2. Buka <https://yourdomain.com/admin>
- 3. Login dengan ADMIN_USERNAME dan ADMIN_PASSWORD
- 4. Verify dashboard loads properly

B. Test Bot Functionality

1. Test Welcome Message

- 2. Kirim pesan "Hi" ke nomor WhatsApp bot
- 3. Verify menerima sapaan dengan branding GARUDAHOKI
- 4. Check time-based greeting sesuai waktu WIB

5. Test Menu System

- 6. Reply dengan "a" untuk Informasi Umum
- 7. Verify submenu Bonus dan Pola muncul
- 8. Test "b" untuk Bicara dengan Agen

9. Test Content Filter

- 10. Kirim kata kasar (testing)
- 11. Verify warning message
- 12. Test auto-block setelah 3 pelanggaran



Langkah 9: Monitoring & Maintenance

A. Setup Monitoring

1. Application Logs

```

```bash
Via SSH
tail -f /public_html/api/logs/bot.log

Via PM2
pm2 logs garudahoki-bot
```

```

1. Database Monitoring

```

```sql
- Check active sessions
SELECT COUNT(*) FROM chat_sessions WHERE status = 'active';

- Check blocked numbers
SELECT COUNT(*) FROM blocked_numbers;

- Check violations
SELECT COUNT(*) FROM user_violations WHERE created_at > NOW() - INTERVAL '24 hours';
```

```

B. Backup Strategy

1. Database Backup

```

bash
# Daily backup script
pg_dump -h localhost -U cpanel_username_garudahoki_user cpanel_username_garudahoki_bot >
backup_$(date +%Y%m%d).sql

```

2. File Backup

```

bash
# Backup application files
tar -czf app_backup_$(date +%Y%m%d).tar.gz /public_html/api

```



Troubleshooting Common Issues

Issue 1: Application Won't Start

Symptoms:

- 502 Bad Gateway error
- Application not responding

Solutions:

```
# Check application status
pm2 status

# Check logs for errors
pm2 logs garudahoki-bot

# Restart application
pm2 restart garudahoki-bot

# Check environment variables
cat .env | grep -v PASSWORD
```

Issue 2: Database Connection Error

Symptoms:

- "Database connection failed" in logs
- Admin panel shows loading forever

Solutions:

```
# Test database connection
psql -h localhost -U cpanel_username_garudahoki_user -d cpanel_username_garudahoki_bot
-c "SELECT 1;"

# Check DATABASE_URL format
echo $DATABASE_URL

# Verify database exists
psql -h localhost -U cpanel_username_garudahoki_user -l
```

Issue 3: Webhook Not Receiving Messages

Symptoms:

- Bot tidak merespon pesan WhatsApp
- Webhook verification failed

Solutions:

```
# Test webhook endpoint
curl -X POST https://yourdomain.com/webhook/whatsapp \
  -H "Content-Type: application/json" \
  -d '{"test": "message"}'

# Check SSL certificate
curl -I https://yourdomain.com

# Verify webhook URL di provider dashboard
# Check firewall settings
```

Issue 4: Admin Panel Not Loading

Symptoms:

- Login page shows but dashboard doesn't load
- "Connection error" messages

Solutions:

```
# Check JWT_SECRET is set
grep JWT_SECRET .env

# Verify admin credentials
grep ADMIN_ .env

# Check browser console for errors
# Clear browser cache and cookies
```

Support & Maintenance

Regular Maintenance Tasks

1. **Weekly:**
 - Check application logs
 - Monitor database size
 - Verify webhook status
2. **Monthly:**
 - Update dependencies
 - Database cleanup
 - Performance optimization
3. **Quarterly:**
 - Security audit
 - Backup verification
 - Provider API updates

Getting Help

1. **Check logs first:**

```
bash
pm2 logs garudahoki-bot --lines 100
```
2. **Common log locations:**
 - Application: /public_html/api/logs/bot.log
 - PM2: ~/.pm2/logs/
 - System: /var/log/
3. **Contact support jika diperlukan:**
 - Namecheap Support untuk hosting issues
 - WhatsApp Provider Support untuk API issues
 - LiveChatInc Support untuk integration issues

 **Selamat! Bot WhatsApp GARUDAHOKI Anda sudah berhasil di-deploy dan siap digunakan!**