

N8n

N8n is a powerful workflow automation platform that enables you to connect any app with any other app. With its intuitive visual editor, you can build complex automations without writing code, while still having the flexibility to add custom JavaScript when needed.

This AMI provides a production ready N8n deployment optimized for AWS, featuring automatic configuration, SSL certificate management, and persistent storage.

- [1.0.0 - ami-01255585612415786](#)

1.0.0 - ami-01255585612415786

N8n Workflow Automation

Production-ready N8n deployment for AWS with automatic SSL and persistent storage.

Requirements

Resource	Minimum	Recommended
Instance Type	t3.small	t3.medium or larger
RAM	2 GB	4 GB+
EBS Storage	20 GB	50 GB+

Getting Started

Interactive Configuration

SSH into your instance and run:

```
sudo /opt/n8n/configure-n8n.sh
```

Follow the prompts to configure your domain/IP and HTTPS settings. Once complete, open the displayed URL and create your owner account.

Automated Configuration (User Data)

Launch with JSON user data for automated setup:

```
{
  "host": "n8n.example.com",
  "use_https": true,
  "letsencrypt_email": "admin@example.com",
  "timezone": "UTC"
```

```
}
```

Field	Required	Description
host	No	Domain or IP (auto detects if not set)
use_https	No	Enable Let's Encrypt SSL (default: false)
letsencrypt_email	Yes if HTTPS	Email for SSL notifications
timezone	No	Server timezone (default: UTC)

Management Commands

```
n8n-cli status      # Check service status
n8n-cli logs        # View logs
n8n-cli logs -f     # Follow logs
sudo n8n-cli start   # Start services
sudo n8n-cli stop    # Stop services
sudo n8n-cli restart # Restart services
n8n-cli info         # View configuration
sudo n8n-cli update  # Update to latest version
sudo n8n-cli backup  # Create backup
sudo n8n-cli restore FILE # Restore from backup
n8n-cli shell        # Access container shell
```

Ports

Port	Description
22	SSH
80	HTTP (HTTPS redirect/ACME challenge)
443	HTTPS
5678	N8n Web UI (HTTP mode only)

Security Group for HTTP mode: Allow 22, 5678

Security Group for HTTPS mode: Allow 22, 80, 443

HTTPS Setup

Requirements:

- Valid domain name pointing to your instance
- Ports 80 and 443 open in security group

Certificates are automatically provisioned via Let's Encrypt and renew automatically.

Troubleshooting:

```
docker logs acme-companion      # Check certificate logs
docker logs nginx-proxy        # Check proxy logs
cd /opt/n8n && sudo docker compose restart  # Restart all services
```

Backups

```
# Create backup
sudo n8n-cli backup

# Restore backup
sudo n8n-cli restore /tmp/n8n-backup-YYYYMMDD-HHMMSS.tar.gz

# Copy to S3
aws s3 cp /tmp/n8n-backup-*.tar.gz s3://your-bucket/backups/
```

Backups include all workflows, credentials, execution history, and user accounts.

Data Locations

Path	Description
/mnt/n8n-data/n8n/	N8n application data
/mnt/n8n-data/certs/	SSL certificates
/opt/n8n/	Configuration files

Troubleshooting

Cannot access web interface:

1. Check security group allows correct ports
2. Run `n8n-cli status` to verify service is running
3. Use incognito browser to avoid cached cookies

Login screen instead of setup:

```
sudo systemctl stop n8n.service  
cd /opt/n8n && sudo docker compose down  
sudo rm -rf /mnt/n8n-data/n8n/*  
sudo rm -f /opt/n8n/.configured  
sudo /opt/n8n/configure-n8n.sh
```

HTTPS not working:

```
cd /opt/n8n && sudo docker compose restart  
docker logs acme-companion  
docker logs nginx-proxy
```

Support

- N8n Documentation: <https://docs.n8n.io>
- N8n Community: <https://community.n8n.io>