

# Guia de Instalacion Completa

## Requisitos Previos

### Hardware Minimo

- CPU: 2 cores
- RAM: 4GB (2GB minimo)
- Disco: 20GB libres
- Red: IP publica o tunel (para WhatsApp)

### Software Requerido

- Docker 20.10+
- Docker Compose 2.0+
- Git
- Conexion a Internet estable

### Cuentas Necesarias

- Cuenta OpenAI o Anthropic
  - Servidor Evolution API (propio o servicio)
  - Email con IMAP habilitado (Gmail, Outlook, etc.)
- 

## Instalacion Paso a Paso

### 1. Clonar Repositorio

```
bash
git clone https://github.com/tuusuario/n8n-email-agent.git
cd n8n-email-agent
```

### 2. Configurar Variables de Entorno

```
bash
cp .env.example .env
nano .env
```

**Variables obligatorias:**

```
env

# IA (elige una)
OPENAI_API_KEY=sk-xxxxxxxxxxxxxxxxxx

# o
ANTHROPIC_API_KEY=sk-ant-xxxxxxxxxxxxxxxxxx


# WhatsApp
EVOLUTION_API_URL=https://tu-servidor.com
EVOLUTION_API_KEY=tu_clave_aqui
EVOLUTION_INSTANCE_NAME=familia-bot


# Email
EMAIL_HOST=imap.gmail.com
EMAIL_PORT=993
EMAIL_USER=tu_email@gmail.com
EMAIL_PASSWORD=tu_app_password


# N8N
N8N_ENCRYPTION_KEY=genera_con_openssl
```

### 3. Generar Claves de Seguridad

```
bash

# Generar encryption key para N8N
openssl rand -hex 32


# Anadela a tu .env
echo "N8N_ENCRYPTION_KEY=tu_clave_generada" >> .env
```

### 4. Configurar Docker Compose

Archivo docker-compose.yml incluido:

yaml

version: '3.8'

services:

n8n:

image: n8nio/n8n:latest

ports:

- "5678:5678"

environment:

- N8N\_ENCRYPTION\_KEY=\${N8N\_ENCRYPTION\_KEY}

volumes:

- n8n\_data:/home/node/.n8n

restart: unless-stopped

redis:

image: redis:7-alpine

ports:

- "6379:6379"

volumes:

- redis\_data:/data

restart: unless-stopped

qdrant:

image: qdrant/qdrant:latest

ports:

- "6333:6333"

volumes:

- qdrant\_data:/qdrant/storage

restart: unless-stopped

volumes:

n8n\_data:

redis\_data:

qdrant\_data:

## 5. Levantar Servicios

bash

*# Iniciar todos los contenedores*

**docker-compose** up -d

*# Verificar que esten corriendo*

**docker-compose** ps

## Salida esperada:

NAME	STATUS	PORTS
n8n	Up 10 seconds	0.0.0.0:5678->5678/tcp
redis	Up 10 seconds	0.0.0.0:6379->6379/tcp
qdrant	Up 10 seconds	0.0.0.0:6333->6333/tcp

## 6. Acceder a N8N

1. Abre navegador: <http://localhost:5678>
2. Crea cuenta inicial (usuario/contrasena)
3. Completa configuracion inicial

## 7. Importar Workflows

### Desde la interfaz N8N:

1. Menu -> Workflows
2. Click en "Import from File"
3. Selecciona: workflows/Email-AI-Agent-Main.json
4. Repite para: workflows/Context-Manager.json

### Desde CLI (alternativa):

```
bash
docker exec n8n n8n import:workflow --input=/data/workflows/
```

## 8. Configurar Credenciales en N8N

Para cada workflow, configura:

### OpenAI/Claude:

- Settings -> Credentials -> Add Credential
- Tipo: OpenAI / Anthropic
- Pega tu API Key

### Evolution API:

- Add Credential -> HTTP Request
- URL: tu\_evolution\_url
- Header: apikey con tu clave

### IMAP Email:

- Add Credential -> Email (IMAP)
- Host: `imap.gmail.com`
- User: `tu_email@gmail.com`
- Password: `tu_app_password`
- Port: 993
- SSL: Activado

## 9. Conectar WhatsApp (Evolution API)

Si usas Evolution API propio:

```
bash

# Generar QR para vincular WhatsApp
curl -X POST ${EVOLUTION_API_URL}/instance/connect/${EVOLUTION_INSTANCE_NAME} \
-H "apikey: ${EVOLUTION_API_KEY}"
```

Escanea el QR con tu WhatsApp.

**Verificar conexion:**

```
bash

curl ${EVOLUTION_API_URL}/instance/connectionState/${EVOLUTION_INSTANCE_NAME} \
-H "apikey: ${EVOLUTION_API_KEY}"
```

## 10. Activar Workflows

En N8N:

1. Abre workflow Email-AI-Agent-Main
2. Click en "Active" (switch arriba a la derecha)
3. Repite para Context-Manager

---

## Verificacion de Instalacion

**Test 1: Servicios Activos**

```
bash
```

```
# N8N
```

```
curl http://localhost:5678/healthz
```

```
# Redis
```

```
redis-cli ping
```

```
# Respuesta esperada: PONG
```

```
# Qdrant
```

```
curl http://localhost:6333/healthz
```

## Test 2: Email a WhatsApp

1. Envía un email a tu dirección configurada
2. Asunto: "Test Bot"
3. Espera 5 minutos (polling interval)
4. Verifica WhatsApp

## Test 3: Logs N8N

```
bash
```

```
docker logs n8n -f --tail 100
```

Busca: Workflow "Email-AI-Agent-Main" started

---

## Solucion de Problemas

### Error: N8N no inicia

```
bash
```

```
# Verificar logs
```

```
docker logs n8n
```

```
# Recrear contenedor
```

```
docker-compose down
```

```
docker-compose up -d
```

### Error: No se conecta a Redis

```
bash
```

```
# Verificar Redis activo
```

```
docker ps | grep redis
```

```
# Test conexion
```

```
docker exec redis redis-cli ping
```

## Error: Emails no se procesan

### Verificaciones:

#### 1. IMAP configurado correctamente:

```
bash
```

```
telnet imap.gmail.com 993
```

#### 2. Gmail requiere "App Password":

- Google Account -> Security
- 2-Step Verification -> App Passwords
- Genera contraseña para "Mail"

#### 3. Verificar firewall:

```
bash
```

```
sudo ufw allow 993/tcp
```

## Error: WhatsApp desconectado

```
bash
```

```
# Reconectar instancia
```

```
curl -X POST ${EVOLUTION_API_URL}/instance/restart/${EVOLUTION_INSTANCE_NAME} \
-H "apikey: ${EVOLUTION_API_KEY}"
```

---

## Seguridad Post-Instalacion

### 1. Cambiar Puerto N8N (produccion)

```
yaml
```

```
# docker-compose.yml
```

```
ports:
```

```
- "8080:5678"
```

## 2. Habilitar HTTPS

```
bash
```

```
# Instalar Caddy o Nginx
```

```
apt install caddy
```

```
# Configurar reverse proxy
```

```
caddy reverse-proxy --from tu-dominio.com --to localhost:5678
```

## 3. Firewall

```
bash
```

```
# Permitir solo puertos necesarios
```

```
ufw allow 22/tcp # SSH
```

```
ufw allow 443/tcp # HTTPS
```

```
ufw deny 5678/tcp # Bloquear N8N directo
```

```
ufw enable
```

---

## Siguiente Paso

Una vez instalado, consulta:

- TECHNICAL.md - Arquitectura detallada
- QUICKSTART.md - Uso basico

## Soporte

Problemas? Abre un issue en GitHub