Ultimate Agent Zero + Mistral Nemo 12B Installation Guide

🚀 Quick Start (One Command Installation)

```bash
curl -fsSL https://raw.githubusercontent.com/your-repo/ultimate-agent0-installer.sh |
```

That's it! The script will handle everything automatically.

📋 Table of Contents

🎯 Overview

This is an enterprise-grade, bulletproof installation system that sets up:

- **Agent Zero**: Advanced AI agent framework
- **Mistral Nemo 12B**: Powerful local language model
- **Complete Infrastructure**: Docker, CUDA (if GPU available), monitoring, and management tools

Key Features

- ✅ **One-command installation** - Fully automated
- ✅ **Bulletproof error handling** - Automatic recovery and rollback
- ✅ **Works on any Ubuntu 24.04** - Fresh or existing installations
- ✅ **GPU auto-detection** - Installs CUDA if NVIDIA GPU present
- ✅ **Service management** - Systemd integration
- ✅ **Health monitoring** - Built-in diagnostics

- ✅ **Automatic updates** - Keep system current
- ✅ **Production ready** - Enterprise-grade security and logging

💻 System Requirements

Minimum Requirements

- **OS**: Ubuntu 24.04 LTS (x86_64)
- **RAM**: 16GB (32GB recommended)
- **Storage**: 50GB free space
- **CPU**: 4+ cores
- **Network**: Stable internet connection

Optional

- **GPU**: NVIDIA GPU with 12GB+ VRAM for acceleration
- **CUDA**: Automatically installed if GPU detected

📦 Installation Methods

Method 1: Quick Install (Recommended)

bash

```bash
# Download and run in one command
curl -fsSL https://raw.githubusercontent.com/your-repo/ultimate-agent0-installer.sh |
```

Method 2: Download First

bash

```bash
# Download the script
wget https://raw.githubusercontent.com/your-repo/ultimate-agent0-installer.sh

# Make executable
chmod +x ultimate-agent0-installer.sh

# Run installation
sudo ./ultimate-agent0-installer.sh
```

Method 3: Advanced Installation

bash

```
# Silent installation (no prompts)
sudo ./ultimate-agent0-installer.sh --silent

# Force reinstall everything
sudo ./ultimate-agent0-installer.sh --force-reinstall

# CPU-only installation (skip GPU)
sudo ./ultimate-agent0-installer.sh --no-gpu

# Verbose mode for debugging
sudo ./ultimate-agent0-installer.sh --verbose
```

Installation Options

Option	Description
`--silent`, `-s`	Non-interactive installation
`--force-reinstall`, `-f`	Remove existing installation and start fresh
`--skip-validation`	Skip system requirement checks
`--no-gpu`	Skip GPU detection and CUDA installation
`--dry-run`	Show what would be done without making changes
`--verbose`, `-v`	Show detailed output
`--no-cleanup`	Don't cleanup on error (for debugging)
`--help`, `-h`	Show help message

✅ Post-Installation

1. Verify Installation

Run the health check:

bash

```
/opt/agent0-mistral/health_check.sh
```

Expected output:

```
=== System Health Check ===

Service Status:
✓ Ollama: Running
✓ Agent Zero: Running

API Status:
✓ Ollama API: Responsive
✓ Agent Zero: Responsive

Available Models:
  • mistral-nemo:12b

System Resources:
  • CPU Load: 0.5, 0.3, 0.2
  • Memory: 8.2GB / 32GB
  • Disk: 120GB / 500GB (24%)
```

2. Access the Web Interface

Open your browser and navigate to:

```
http://YOUR-SERVER-IP:8080
```

Replace `YOUR-SERVER-IP` with your server's IP address.

3. Test the API

bash

```
# Test Ollama API
curl http://localhost:11434/api/tags

# Test model response
curl -X POST http://localhost:11434/api/generate \
  -H 'Content-Type: application/json' \
  -d '{"model": "mistral-nemo:12b", "prompt": "Hello, how are you?"}'
```

📚 Usage Guide

Service Management

```bash
# Start services
sudo /opt/agent0-mistral/control.sh start

# Stop services
sudo /opt/agent0-mistral/control.sh stop

# Restart services
sudo /opt/agent0-mistral/control.sh restart

# Check status
sudo /opt/agent0-mistral/control.sh status

# View logs
sudo /opt/agent0-mistral/control.sh logs
```

Using Agent Zero

1. **Web Interface**: Navigate to `http://YOUR-SERVER-IP:8080`
2. **Configure Model**: Mistral Nemo is pre-configured
3. **Start Chatting**: Begin interacting with the AI

API Usage

Python Example

```python
import requests

# Agent Zero API
response = requests.post('http://localhost:8080/api/chat',
    json={'message': 'Hello, Agent Zero!'})
print(response.json())

# Direct Ollama API
response = requests.post('http://localhost:11434/api/generate',
    json={
        'model': 'mistral-nemo:12b',
        'prompt': 'What is the meaning of life?',
        'stream': False
    })
print(response.json()['response'])
```

JavaScript Example

javascript

```
// Using fetch API
fetch('http://localhost:8080/api/chat', {
    method: 'POST',
    headers: {'Content-Type': 'application/json'},
    body: JSON.stringify({message: 'Hello from JavaScript!'})
})
.then(response => response.json())
.then(data => console.log(data));
```

🔧 Troubleshooting

Common Issues and Solutions

1. Services Won't Start

bash

```
# Check service status
sudo systemctl status ollama
sudo systemctl status agent0

# View detailed logs
sudo journalctl -u ollama -n 100
sudo journalctl -u agent0 -n 100

# Restart services
sudo systemctl restart ollama
sudo systemctl restart agent0
```

2. GPU Not Detected

bash

```
# Check if GPU is visible
lspci | grep -i nvidia

# Check NVIDIA driver
nvidia-smi

# Reinstall NVIDIA drivers
sudo apt update
sudo apt install nvidia-driver-545
sudo reboot
```

3. Model Download Fails

bash

```bash
# Manually download model
sudo -u ollama ollama pull mistral-nemo:12b

# Check available space
df -h

# Clear Ollama cache if needed
sudo rm -rf /var/lib/ollama/.ollama/models/blobs
```

4. Web UI Not Accessible

bash

```bash
# Check if port is open
sudo netstat -tulpn | grep 8080

# Check firewall
sudo ufw status

# Allow port through firewall
sudo ufw allow 8080/tcp
```

5. High Memory Usage

bash

```bash
# Check memory usage
free -h
htop

# Restart services to free memory
sudo systemctl restart ollama agent0

# Adjust Ollama memory settings
sudo systemctl edit ollama
# Add under [Service]:
# Environment="OLLAMA_MAX_LOADED_MODELS=1"
```

Complete Reset

If you need to completely reset the installation:

```bash
# Stop all services
sudo systemctl stop agent0 ollama

# Remove installation
sudo rm -rf /opt/agent0-mistral
sudo rm -rf /etc/agent0-mistral
sudo rm -rf /var/log/agent0-mistral

# Remove services
sudo rm -f /etc/systemd/system/agent0.service
sudo rm -f /etc/systemd/system/ollama.service
sudo systemctl daemon-reload

# Remove users
sudo userdel -r agent0 2>/dev/null
sudo userdel -r ollama 2>/dev/null

# Re-run installer
curl -fsSL https://raw.githubusercontent.com/your-repo/ultimate-agent0-installer.sh |
```

⚙️ Advanced Configuration

Environment Variables

Edit `/etc/agent0-mistral/agent0.env`:

```bash
# Change port
AGENT0_PORT=8090

# Set specific model parameters
MODEL_TEMPERATURE=0.8
MODEL_MAX_TOKENS=8192

# Enable debug mode
DEBUG=true

# Configure CORS
ALLOWED_ORIGINS=https://yourdomain.com
```

Model Configuration

Use Different Model

```bash
# Download alternative model
sudo -u ollama ollama pull llama2:13b

# Update configuration
sudo nano /etc/agent0-mistral/agent0.env
# Change: DEFAULT_MODEL=llama2:13b

# Restart service
sudo systemctl restart agent0
```

Model Parameters

```bash
# Edit model settings
sudo nano /etc/agent0-mistral/model_config.json

# Adjust parameters like:
# - temperature (0.0-1.0)
# - max_tokens
# - top_p
# - repeat_penalty
```

Performance Tuning

For GPU Systems

```bash
# Enable flash attention
sudo systemctl edit ollama
# Add:
# [Service]
# Environment="OLLAMA_FLASH_ATTENTION=1"

# Set GPU memory fraction
# Environment="CUDA_VISIBLE_DEVICES=0"
# Environment="OLLAMA_GPU_MEMORY_FRACTION=0.8"
```

For CPU Systems

bash

```
# Limit CPU usage
sudo systemctl edit ollama
# Add:
# [Service]
# CPUQuota=80%

# Set thread count
# Environment="OLLAMA_NUM_THREADS=8"
```

Security Hardening

1. Firewall Configuration

bash

```
# Enable firewall
sudo ufw enable

# Allow only specific IPs
sudo ufw allow from 192.168.1.0/24 to any port 8080

# Restrict Ollama to localhost
sudo ufw deny 11434
```

2. SSL/TLS Setup

bash

```
# Install nginx and certbot
sudo apt install nginx certbot python3-certbot-nginx

# Configure reverse proxy
sudo nano /etc/nginx/sites-available/agent0

# Get SSL certificate
sudo certbot --nginx -d yourdomain.com
```

3. Authentication

bash

```
# Add basic auth to nginx
sudo apt install apache2-utils
sudo htpasswd -c /etc/nginx/.htpasswd agent0user

# Update nginx config to require auth
```

🔄 Maintenance

Regular Updates

bash

```
# Update entire system
sudo /opt/agent0-mistral/update.sh

# Manual update steps:
# 1. Update Agent Zero
cd /opt/agent0-mistral/agent-zero
sudo -u agent0 git pull

# 2. Update Python packages
sudo -u agent0 /opt/miniconda3/bin/conda activate agent0
pip install --upgrade -r requirements.txt

# 3. Update model
sudo -u ollama ollama pull mistral-nemo:12b
```

Backup and Restore

Create Backup

bash

```
# Backup configuration and data
sudo tar -czf agent0-backup-$(date +%Y%m%d).tar.gz \
  /etc/agent0-mistral \
  /opt/agent0-mistral/data \
  /opt/agent0-mistral/workspace

# Backup models (large!)
sudo tar -czf models-backup-$(date +%Y%m%d).tar.gz \
  /var/lib/ollama/models
```

Restore from Backup

bash

```bash
# Stop services
sudo systemctl stop agent0 ollama

# Restore files
sudo tar -xzf agent0-backup-20240131.tar.gz -C /

# Restart services
sudo systemctl start ollama agent0
```

Log Management

bash

```bash
# View logs
sudo journalctl -u agent0 -f  # Follow Agent Zero logs
sudo journalctl -u ollama -f  # Follow Ollama logs

# Log rotation is automatic via logrotate
# Config: /etc/logrotate.d/agent0-mistral

# Manual log cleanup
sudo journalctl --vacuum-time=7d  # Keep only 7 days
sudo journalctl --vacuum-size=1G  # Limit to 1GB
```

Monitoring

System Metrics

bash

```bash
# Real-time monitoring
htop  # CPU and memory
iotop  # Disk I/O
nvidia-smi -l 1  # GPU monitoring

# Service metrics
systemctl status agent0 ollama
```

Create Monitoring Dashboard

```bash
# Install monitoring stack
sudo apt install prometheus grafana

# Configure to scrape metrics
# Agent Zero exposes metrics at :8080/metrics
# Ollama exposes metrics at :11434/metrics
```

🔒 Security Considerations

Best Practices

1. **Change Default Secrets**

   ```bash
   # Generate new secret key
   sudo nano /etc/agent0-mistral/agent0.env
   # Update: SECRET_KEY=your-new-secret-key
   ```

2. **Restrict Network Access**
   - Use firewall rules
   - Deploy behind reverse proxy
   - Enable HTTPS/TLS

3. **Regular Updates**
   - Keep system packages updated
   - Update Agent Zero regularly
   - Monitor security advisories

4. **Access Control**
   - Implement authentication
   - Use API keys for programmatic access
   - Monitor access logs

5. **Data Protection**
   - Regular backups
   - Encrypt sensitive data
   - Secure model storage

Security Checklist

☐ Changed default secret keys
☐ Configured firewall rules

- □ Enabled HTTPS
- □ Set up authentication
- □ Configured log monitoring
- □ Enabled automatic updates
- □ Created backup strategy
- □ Reviewed file permissions

❓ FAQ

Q: Can I run this on a different Linux distribution?

A: The script is optimized for Ubuntu 24.04 but may work on other Debian-based systems with modifications.

Q: How much disk space do models use?

A: Mistral Nemo 12B requires approximately 12-15GB. Plan for 20GB+ for comfortable operation.

Q: Can I use multiple models?

A: Yes! Download additional models with `ollama pull model-name` and configure in Agent Zero.

Q: Is GPU required?

A: No, but highly recommended. CPU-only mode works but is significantly slower.

Q: Can I run this in Docker?

A: The installer sets up Docker for Agent Zero. Running the installer itself in Docker is not recommended.

Q: How do I add custom models?

A: Use `ollama pull model-name` or create custom modelfiles for Ollama.

Q: What ports need to be open?

A: By default: 8080 (Agent Zero) and 11434 (Ollama, localhost only)

Q: Can I change the installation directory?

A: Edit the script variables before running, but /opt/agent0-mistral is recommended.

Q: How do I uninstall everything?

A: Use the complete reset instructions in the Troubleshooting section.

Q: Is this production-ready?

A: Yes, with proper security configuration (HTTPS, authentication, firewall rules).

📞 Support

- **Issues**: Check the Troubleshooting section first
- **Logs**: Located in `/var/log/agent0-mistral/`
- **Health Check**: Run `/opt/agent0-mistral/health_check.sh`
- **Agent Zero Docs**: https://github.com/frdel/agent-zero
- **Ollama Docs**: https://ollama.ai/docs

🎉 Success!

If you've followed this guide, you should now have a fully functional Agent Zero + Mistral Nemo 12B system. Enjoy your AI assistant!

Remember to:

- Run regular updates
- Monitor system resources
- Keep backups
- Follow security best practices

Happy AI adventures! 🚀