SimWidget Engine - Windows Services Guide

Version: v1.0.0
Last updated: 2025-01-09

Overview

SimWidget Engine can run as Windows services for automatic startup and background operation. This guide covers installation, management, and troubleshooting.

Services Architecture

Service	Port	Purpose
SimWidget Main Server	8080	Core WebSocket server, SimConnect integration
SimWidget Agent	8585	Claude chat assistant for development
SimWidget Remote Support	8590	Remote command API

Quick Start

Manual Start (Development)

```batch
C:\DevClaude\SimWidget_Engine\start-all-servers.bat
```

Service Installation (Production)

```powershell
# Run as Administrator
cd C:\DevClaude\SimWidget_Engine\Admin\remote-support
node install-service.js

cd C:\DevClaude\SimWidget_Engine\Admin\agent
node install-service.js
```

Service Management Commands

PowerShell Commands (Run as Administrator)

```powershell
powershell

# List SimWidget services
Get-Service | Where-Object {$_.Name -like "*SimWidget*"}

# Start a service
Start-Service -Name "SimWidgetMainServer"
Start-Service -Name "SimWidgetAgent"
Start-Service -Name "SimWidgetRemoteSupport"

# Stop a service
Stop-Service -Name "SimWidgetMainServer"
Stop-Service -Name "SimWidgetAgent"
Stop-Service -Name "SimWidgetRemoteSupport"

# Restart a service
Restart-Service -Name "SimWidgetMainServer"

# Check service status
Get-Service -Name "SimWidgetMainServer" | Select-Object Status, StartType

# Set service to auto-start
Set-Service -Name "SimWidgetMainServer" -StartupType Automatic

# Set service to manual start
Set-Service -Name "SimWidgetMainServer" -StartupType Manual

# Disable a service
Set-Service -Name "SimWidgetMainServer" -StartupType Disabled
```

SC Command (Command Prompt as Administrator)

```batch
batch
```

```batch
:: Query service status
sc query SimWidgetMainServer

:: Start service
sc start SimWidgetMainServer

:: Stop service
sc stop SimWidgetMainServer

:: Delete service (uninstall)
sc delete SimWidgetMainServer

:: Create service manually
sc create SimWidgetMainServer binPath= "C:\DevClaude\SimWidget_Engine\service.exe" start= auto
```

NET Commands

```batch
:: Start service
net start SimWidgetMainServer

:: Stop service
net stop SimWidgetMainServer
```

Using Node-Windows for Service Installation

SimWidget uses node-windows package for service management.

Install Service Script Template

```javascript
```

```javascript
// install-service.js
const Service = require('node-windows').Service;
const path = require('path');

const svc = new Service({
  name: 'SimWidget Main Server',
  description: 'SimWidget Engine WebSocket server for MSFS 2024',
  script: path.join(__dirname, 'server.js'),
  nodeOptions: [],
  workingDirectory: __dirname,
  allowServiceLogon: true
});

svc.on('install', () => {
  console.log('Service installed successfully');
  svc.start();
});

svc.on('alreadyinstalled', () => {
  console.log('Service already installed');
});

svc.on('error', (err) => {
  console.error('Service error:', err);
});

svc.install();
```

Uninstall Service Script Template

```
javascript
```

```javascript
// uninstall-service.js
const Service = require('node-windows').Service;
const path = require('path');

const svc = new Service({
  name: 'SimWidget Main Server',
  script: path.join(__dirname, 'server.js')
});

svc.on('uninstall', () => {
  console.log('Service uninstalled successfully');
});

svc.uninstall();
```

Services Manager GUI

Open Windows Services Manager:

```
services.msc
```

Or via Run dialog: [Win + R] → type [services.msc] → Enter

Event Viewer Logs

Check service logs in Event Viewer:

1. Open Event Viewer: [Win + R] → [eventvwr.msc]

2. Navigate to: Windows Logs → Application

3. Filter by Source: "SimWidget" or "node"

PowerShell Log Query

```
powershell
```

```powershell
# Get recent SimWidget events
Get-EventLog -LogName Application -Source "SimWidget*" -Newest 20

# Get all Node.js related events
Get-EventLog -LogName Application | Where-Object {$_.Message -like "*SimWidget*"} | Select-Object -First 10
```

Troubleshooting

Service Won't Start

1. **Check port availability:**

```powershell
powershell

netstat -ano | findstr :8080
netstat -ano | findstr :8585
netstat -ano | findstr :8590
```

2. **Kill process using port:**

```powershell
powershell

# Find PID using port
$pid = (Get-NetTCPConnection -LocalPort 8080 -ErrorAction SilentlyContinue).OwningProcess
if ($pid) { Stop-Process -Id $pid -Force }
```

3. **Check Node.js path:**

```powershell
powershell

where node
node --version
```

4. **Verify dependencies:**

```batch
batch
```

```
cd C:\DevClaude\SimWidget_Engine\simwidget-hybrid
npm install
```

Service Crashes on Startup

1. **Check logs:**

```powershell
Get-Content "C:\DevClaude\SimWidget_Engine\simwidget-hybrid\backend\logs\server.log" -Tail 50
```

2. **Run manually to see errors:**

```batch
cd C:\DevClaude\SimWidget_Engine\simwidget-hybrid\backend
node server.js
```

3. **Check environment variables:**

```powershell
[System.Environment]::GetEnvironmentVariable("NODE_ENV", "Machine")
```

Permission Issues

1. **Run as Administrator** when installing/uninstalling services
2. **Check service account:**

```powershell
Get-WmiObject Win32_Service | Where-Object {$_.Name -like "*SimWidget*"} | Select-Object Name, StartName
```

3. **Grant Log on as Service right:**
    - Run `secpol.msc`
    - Navigate to: Local Policies → User Rights Assignment
    - Add user to "Log on as a service"

Firewall Configuration

Allow SimWidget Ports

```powershell
# Run as Administrator
New-NetFirewallRule -DisplayName "SimWidget Main Server" -Direction Inbound -Port 8080 -Protocol TCP -Action Allow
New-NetFirewallRule -DisplayName "SimWidget Agent" -Direction Inbound -Port 8585 -Protocol TCP -Action Allow
New-NetFirewallRule -DisplayName "SimWidget Remote Support" -Direction Inbound -Port 8590 -Protocol TCP -Action Al
```

Remove Firewall Rules

```powershell
Remove-NetFirewallRule -DisplayName "SimWidget Main Server"
Remove-NetFirewallRule -DisplayName "SimWidget Agent"
Remove-NetFirewallRule -DisplayName "SimWidget Remote Support"
```

Service Recovery Options

Configure automatic restart on failure:

```powershell
# Set recovery options (restart after 1 minute)
sc failure SimWidgetMainServer reset= 86400 actions= restart/60000/restart/60000/restart/60000
```

Or via Services GUI:

1. Open `services.msc`

2. Right-click service → Properties

3. Recovery tab → Set failure actions

Startup Dependencies

If SimWidget depends on other services (e.g., FSUIPC):

```powershell
powershell
# Add dependency
sc config SimWidgetMainServer depend= "FSUIPC7"

# Remove dependency
sc config SimWidgetMainServer depend= ""
```

Quick Reference Card

Task	Command
Start all servers	start-all-servers.bat
List services	Get-Service *SimWidget*
Start service	Start-Service SimWidgetMainServer
Stop service	Stop-Service SimWidgetMainServer
Restart service	Restart-Service SimWidgetMainServer
Check ports	netstat -ano | findstr :8080
View logs	eventvwr.msc
Open Services	services.msc

File Locations

Item	Path
Main Server	C:\DevClaude\SimWidget_Engine\simwidget-hybrid\backend\server.js
Agent Server	C:\DevClaude\SimWidget_Engine\Admin\agent\agent-server.js
Remote Support	C:\DevClaude\SimWidget_Engine\Admin\remote-support\service.js

Item	Path
Start Script	C:\DevClaude\SimWidget_Engine\start-all-servers.bat
Service Logs	C:\DevClaude\SimWidget_Engine\logs\

See Also

- CLAUDE.md - Project overview
- ARCHITECTURE.md - System architecture
- GETTING-STARTED.md - Setup guide