

Troubleshooting & Installing Beszel Agent (Windows)

Internal KB: Troubleshooting & Installing Beszel Agent (Windows)

Target Systems: Windows 10 / Windows 11

Method: WinSW (Windows Service Wrapper) via PowerShell

Primary Port: 45876

1. Overview

Windows does not natively support Linux background commands (like `nohup`) or system daemons (like `systemd`). To run the Beszel Agent reliably in the background so it survives reboots and user logouts, we use **WinSW (Windows Service Wrapper)**. This method wraps the Go binary into an official Windows Service.

2. Phase 1: The "Nuke" (Cleanup)

If you previously tried to run the agent via a `.bat` file or accidentally ran Linux commands, you must kill the orphaned processes to free up Port 45876.

Open an **Administrator PowerShell** window and execute:

```
# 1. Create the directory if it doesn't exist
New-Item -Path "C:\Beszel" -ItemType Directory -Force | Out-Null
cd C:\Beszel
```

```
# 2. Stop and uninstall any existing wrapper
.\beszel-service.exe stop 2>$null
.\beszel-service.exe uninstall 2>$null

# 3. Forcefully kill any invisible background agent processes
Stop-Process -Name "beszel-agent" -Force -ErrorAction SilentlyContinue
Stop-Process -Name "agent" -Force -ErrorAction SilentlyContinue
```

3. Phase 2: Binary Placement

Unlike Linux automated scripts, Windows requires you to manually place the agent executable.

1. Download the latest `windows_amd64` agent from the [Beszel GitHub Releases page](#).
2. Extract the archive and move the executable into your `C:\Beszel` folder.
3. Ensure the file is named exactly `beszel-agent.exe`.

4. Phase 3: The "Pave" (Installation)

With the binary in place, return to your **Administrator PowerShell** window. This script downloads the service wrapper, builds the XML configuration, opens the Windows Firewall, and starts the service.

```
$installPath = "C:\Beszel"

# 1. Download WinSW (The modern Windows Service Wrapper)
Invoke-WebRequest -Uri "https://github.com/winsw/winsw/releases/download/v2.12.0/WinSW-x64.exe" -OutFile "$installPath\bessel-service.exe"

# 2. Create the XML configuration with your Public Key
$xmlContent = @"
<service>
  <id>besselagent</id>
  <name>Beszel Agent</name>
  <description>Lightweight monitoring agent for CasaOS Hub</description>
  <executable>$installPath\bessel-agent.exe</executable>
  <env name="KEY" value="<YOUR_HUB_PUBLIC_KEY_HERE>" />
  <env name="PORT" value="45876" />
  <log mode="roll"></log>
```

```
<onfailure action="restart" delay="10 sec"/>
</service>
"@
Set-Content -Path "$installPath\beszel-service.xml" -Value $xmlContent

# 3. Open the Windows Firewall Port
New-NetFirewallRule -DisplayName "Beszel Agent" -Direction Inbound -LocalPort 45876 -Protocol
TCP -Action Allow -ErrorAction SilentlyContinue

# 4. Install and Start the background service
cd $installPath
.\beszel-service.exe install
.\beszel-service.exe start
```

5. Verification Checklist

- **Terminal Output:** The PowerShell window should report `Service 'Beszel Agent (beszelagent)' started successfully.`
- **Log Check:** Look inside `C:\Beszel`. You should now see `beszel-service.out` and `beszel-service.err` files generated by the wrapper.
- **Service Check:** Run `Get-Service beszelagent` to verify the Windows Service status reads `Running`.
- **Hub Check:** Add the system to your CasaOS Beszel dashboard using the Windows machine's IP address.

Note: Windows aggressively manages power states. If your PC goes to sleep, the agent will pause and appear offline on the Hub. Ensure your Windows Power Plan is set to keep the system awake if you require 24/7 monitoring.

Revision #3

Created 2026-03-18 04:02:44 UTC by Francis

Updated 2026-03-24 12:53:01 UTC by Francis