

LNApi Install

As of August 26, 2026

Documentation note: LNApi capabilities, fields, validation rules, and examples may change as releases are finalized. Confirm current behavior with Benedict Group before relying on details for production.

This page is for the server or IT person hosting LNApi on IIS. If you are only calling LNApi from an integration, start with [Getting started](#) instead.

1. Confirm the target .NET version

Use the release notes or deployment package from Benedict Group to confirm the .NET version this LNApi build targets.

Install the **.NET Hosting Bundle** for the app's target .NET version before configuring the IIS site. The Hosting Bundle includes both the .NET Runtime and the ASP.NET Core Module that IIS uses to start and proxy the LNApi application.

If IIS was already installed when you ran the Hosting Bundle installer, restart IIS after the install completes.

2. Prepare IIS

On the server that will host LNApi:

- Enable the **IIS Web Server** role.
- Install the **.NET Hosting Bundle** for the LNApi build you are deploying.
- Bind the site to an HTTPS hostname and certificate trusted by the clients that will call LNApi.
- Do not rely on plain HTTP. LNApi rejects plain-HTTP requests; production traffic should use only `https://`.

3. Deploy the application files

Place the published LNApi files in a server-owned folder, for example:

```
C:\inetpub\LNApi
```

Keep configuration files, connection strings, and environment-specific values under your normal server change-control process. Test and live instances should have separate folders, separate IIS sites or applications, and separate configuration.

Grant the IIS application pool identity read access to the application folder. Grant write access only where LNApi is configured to write logs or temporary files.

4. Configure the IIS site

Create an IIS site or application that points to the LNApi publish folder.

Use these application pool settings unless Benedict Group gives you site-specific instructions:

Setting	Value
.NET CLR version	No Managed Code

Setting	Value
Managed pipeline mode	Integrated
Start mode	AlwaysRunning , if your IIS version supports it

Bind the site to the LNApi hostname your callers will use, for example:

```
https://YOUR-LNAPI-HOST
```

The base URL you publish here is the same host your integration uses in requests such as:

```
https://YOUR-LNAPI-HOST/v1/insurance-renewal
```

5. Verify the installation

After deployment:

- Recycle the LNApi application pool.
- Confirm the HTTPS binding and certificate are valid from a client machine.
- Send a dry-run request against the test instance first. See Getting started for the sample `curl` request.
- Check Windows Event Viewer, IIS logs, and the LNApi logs if the application does not start or returns a server error.

A **401** from LNApi usually means the application is running but the API key is missing, wrong, revoked, or for the wrong environment. A **500** means the application reached server-side code but something failed; check the server logs.

6. Upgrades

For an LNApi upgrade:

- Back up the current application folder and configuration.
- Check whether the new build targets a different .NET version.
- Install the matching .NET Hosting Bundle if the target .NET version changed.
- Replace the published application files.
- Recycle the application pool.
- Verify with a dry-run request before sending live updates.

Troubleshooting quick checks

Symptom	What to check
IIS returns 500.31 or 500.32	The required .NET runtime is missing or incompatible. Install the Hosting Bundle for the app's target .NET version.
IIS returns 500.30	The app failed during startup. Check Event Viewer, LNApi logs, configuration, and database connectivity.
Browser or client cannot connect over HTTPS	Check the IIS binding, certificate, firewall, DNS, and whether the certificate is trusted by the caller.
LNApi returns 401	Check <code>X-API-Key</code> , the key value, and whether the key belongs to the test or live instance you are calling.
LNApi returns 415	Send <code>Content-Type: application/json</code> .