



BridgeLink Quick Start Guide Linux

For New Installations of BridgeLink 26.9.0

Table of Contents

1. About This Guide 3

2. Getting the Software 5

3. Installing BridgeLink 7

4. Choosing and Connecting the Database 11

5. The Service Account 13

6. Starting BridgeLink for the First Time 16

7. Installing WebAdmin 18

8. Finding the Server ID 22

9. Replacing the Certificates 24

10. The Data Pruner 28

11. Troubleshooting 31

CHAPTER 1

About This Guide

BridgeLink is Innovar Healthcare's integration platform. It moves messages between the systems in your organization and lets you build, run and watch those connections, called channels, from a web browser.

The steps here take one Linux server from download to a running BridgeLink 26.9.0 server that you manage in WebAdmin.

If you would like a hand with any part of the install, Innovar Healthcare's professional services team can help with as much or as little of it as you want.

What You Install

Piece	What it is	Runs as	Default port
BridgeLink server	The integration platform itself: your channels and the service that runs them	A systemd service under a dedicated, non-root account	8443 (HTTPS) and 8080 (HTTP)
WebAdmin	The browser-based tool you use to manage the server	The systemd service <code>bridgelinek-webadmin</code> , which runs as <code>root</code>	8444 (HTTPS)
Database	Where BridgeLink keeps channels, messages, events, users and settings	Your own PostgreSQL, MySQL, SQL Server or Oracle database	Set by your database

Open port **8443** on the BridgeLink server to other machines only if WebAdmin runs on another machine or other programs call the BridgeLink API. Channels that listen for connections, such as MLLP or HTTP listeners, need their own ports opened. Port **8080** is BridgeLink's plain-HTTP port; nothing in this guide uses it, so keep it closed to other machines.

systemd is the part of Linux that starts and watches background services. HTTPS is the encrypted form of a web connection.

Decide These First

- **Which database BridgeLink uses.** Use your own external database, not the default Derby database. See [Choosing and Connecting the Database](#). Have an empty database and a login for BridgeLink ready before you install.
- **Which account runs the BridgeLink service.** Use a dedicated system account, for example `bridgelinek`, never `root`. See [The Service Account](#).
- **Which package you install.** See [Choosing a Package](#).

Requirements

- A 64-bit Linux server with systemd
- `sudo` rights on the server
- Java 21 (recommended) or Java 17, for every package, including the RPM package, which does not install it. See **Before You Install** in **Installing BridgeLink** for the builds you can use
- An external database server with an empty database for BridgeLink
- For WebAdmin: Node.js 22, which the WebAdmin installer can install for you on RHEL 8 or later, Amazon Linux 2023, Fedora, Ubuntu and Debian
- A current browser for WebAdmin: Chrome, Edge or Firefox 120 or later, or Safari 17 or later

CHAPTER 2

Getting the Software

BridgeLink downloads are on the BridgeLink website. Customers with an Innovar Healthcare customer portal account can also download them from the portal.

1. On any computer, open <https://www.bridgelink.net/download> in a browser.
2. Select **Linux**.
3. Find the **BridgeLink Server** card and confirm it shows version **26.9.0**.
4. Choose the package you want from the table below.
5. On the server, set the download address for 26.9.0:

```
BASE=https://bridgelink-downloads.s3.us-east-2.amazonaws.com
BASE=$BASE/bridgelink-server/26.9.0/linux
```

6. Download the package you chose, with one of these commands:

```
curl -fLO "$BASE/BridgeLink_linux_26_9_0.rpm"
```

```
curl -fLO "$BASE/BridgeLink_unix_26_9_0.tar.gz"
```

```
curl -fLO "$BASE/BridgeLink_unix_26_9_0.sh"
```

These download the 26.9.0 packages directly. The download page shows only the latest version. For another version, click **All versions** on the page, copy the link of the package you want, and use it with `curl -fLO` instead.

7. Check that the file arrived:

```
ls -l
```

If the server has a desktop, you can download the file in its browser instead. If you downloaded on another computer, copy the file to the server, for example into your home folder.

You do not need a separate WebAdmin download. All three packages put the WebAdmin installer, `BridgeLink-WebAdmin-install.tar.gz`, in `/opt/BridgeLink`. Its name differs from the website's WebAdmin download, `BridgeLink-WebAdmin-26.9.0-install.tar.gz`. You run it after BridgeLink is installed.

Choosing a Package

Package	What you get	Use it when
RPM package (.rpm)	Installs to /opt/BridgeLink, creates the bridgelink system account, and installs a systemd service named blservice that runs as that account. Needs Java installed first (Java 21 recommended).	Red Hat Enterprise Linux 9, Rocky Linux 9 or AlmaLinux 9 on x86_64. Recommended there.
Archive (.tar.gz)	The BridgeLink files only. You add Java, the account and a systemd service yourself, with the steps below.	Any other distribution, or when you want full control of the layout
Installer (.sh)	A guided setup, run as root, that asks for the install folder, ports and service account, and registers a service that runs as that account	You prefer a guided setup

CHAPTER 3

Installing BridgeLink

Install Java first ([Before You Install](#)), then follow the section for the package you chose. Each one ends with BridgeLink installed in `/opt/BridgeLink` and owned by the `bridgelink` service account. A systemd service named `blservice` is set to run as that account. The archive and the installer leave it stopped; the RPM package starts it right away, so stop it before you connect the database (step 2 of [RPM Package](#)). You connect the database before the first start.

Before You Install

Install Java 21 on the server first, whichever package you chose. None of the packages installs it for you. Java 21 is recommended. Use one of these builds:

- Your distribution's OpenJDK 21 package
- Eclipse Temurin 21, from Adoptium
- Amazon Corretto 21
- Azul Zulu 21

Java 17 is also supported with an external database; the default Derby database needs Java 21. Do not install a Java newer than 21: BridgeLink 26.9 does not use one.

To install your distribution's OpenJDK 21 package, run the command for your system.

Red Hat Enterprise Linux, Rocky Linux and AlmaLinux:

```
sudo dnf install java-21-openjdk-headless
```

Amazon Linux 2023:

```
sudo dnf install java-21-amazon-corretto-headless
```

Ubuntu 22.04, Ubuntu 24.04 and Debian 13:

```
sudo apt update && sudo apt install openjdk-21-jre-headless
```

Debian 12 has no OpenJDK 21 package. On Debian 12, install Eclipse Temurin 21 from Adoptium's apt repository, or use Java 17 with an external database.

RPM Package

1. Install the package:

```
sudo dnf install ./BridgeLink_linux_26_9_0.rpm
```

2. The package starts BridgeLink right away. Stop it until the database is connected:

```
sudo systemctl stop blservice
```

The package creates the **bridgeline** account, gives it the **/opt/BridgeLink** folder, and installs the systemd service **blservice**, set to run as **bridgeline**.

Stop the service within a few seconds. If BridgeLink runs longer, it completes a first start on the default Derby database before your database is connected: it creates its keystore and **appdata/server.id**, and writes the keystore passwords into **conf/mirth.properties**. That does no harm, and you carry on with the next chapter as written. Your own database is still empty, and BridgeLink creates its tables there on the start in **Starting BridgeLink for the First Time**.

Archive (.tar.gz)

1. Create the service account:

```
sudo useradd -r -s /sbin/nologin bridgeline
```

2. Extract the archive into **/opt**. It creates **/opt/BridgeLink**:

```
sudo tar -xzf BridgeLink_unix_26_9_0.tar.gz -C /opt
```

3. Give the service account the BridgeLink folder:

```
sudo chown -R bridgeline:bridgeline /opt/BridgeLink
```

4. Open a new service file in a text editor as root. This guide uses **vi** for every file it has you edit; any editor you prefer works the same way:

```
sudo vi /etc/systemd/system/blservice.service
```

5. Press **i** to start typing, then enter this content:

```
[Unit]
Description=BridgeLink
After=network-online.target
Wants=network-online.target

[Service]
Type=simple
User=bridgelink
Group=bridgelink
NoNewPrivileges=true
WorkingDirectory=/opt/BridgeLink
ExecStart=/opt/BridgeLink/blserver
KillSignal=SIGTERM
KillMode=mixed
SuccessExitStatus=143
TimeoutStopSec=60
Restart=on-failure
RestartSec=10
LimitNOFILE=65536

[Install]
WantedBy=multi-user.target
```

6. Press **Esc**, type **:wq**, and press **Enter** to save the file and quit.
7. Tell systemd about the new service:

```
sudo systemctl daemon-reload
```

`SuccessExitStatus=143` tells systemd that the exit code BridgeLink gives when you stop it is a success. Without it, a normal `sudo systemctl stop blservice` is recorded as a failure.

The RPM package and the installer name their service `blservice` too, so the commands in this guide are the same for every package.

Installer (.sh)

On a server with no desktop, the installer runs in the terminal and asks each question there. Press **Enter** to keep the value it offers, except where a step below says to type something else.

1. Make the installer runnable:

```
chmod +x BridgeLink_unix_26_9_0.sh
```

2. Run it as root:

```
sudo ./BridgeLink_unix_26_9_0.sh
```

3. Keep the install folder `/opt/BridgeLink`. If the installer offers a different folder, change it to `/opt/BridgeLink` so the paths in this guide match.
4. On **Linux Service Account**, keep **Run the service as** set to **Dedicated service user**, and keep **User name** set to `bridgelink`. The installer creates that account if it does not exist, gives it the BridgeLink folder, and sets the `blservice` service to run as it. Do not choose **root (not recommended)**: BridgeLink refuses to run as `root` unless `server.allowRoot` is set to `true`, so a service set to root will not start. [The Service Account](#) explains why.
5. On **BridgeLink Administrator Ports**, keep **Administrator Port** at `8443` and **Web Start Port** at `8080`.
6. On **Install Plugins to BridgeLink**, press `Enter` without typing a list, to install no plugins.
7. At **Run BridgeLink?**, type `n` and press `Enter`. Pressing `Enter` alone means Yes, which would start BridgeLink on the default Derby database before you connect yours.

The installer shows the command for installing WebAdmin above the **Run BridgeLink?** prompt (a line about the service account sits between them). [Installing WebAdmin](#) covers that step.

CHAPTER 4

Choosing and Connecting the Database

Every BridgeLink server keeps its channels, messages, events, users and settings in one database. You choose which database in `/opt/BridgeLink/conf/mirth.properties`, and you make that choice before BridgeLink starts for the first time.

Use an External Database, Not Derby

BridgeLink ships set to the default Derby database. Derby needs no setup, which makes it handy for a short test. Do not use it for a server you plan to keep:

- **Requires Java 21.** BridgeLink 26.9 will not start a server on the default Derby database with any Java older than 21. It stops at once and writes this line to `logs/mirth.log`:
`embedded Derby requires Java 21+ as of 26.9; upgrade Java or switch to an external database`. No setting turns this check off.
- **Runs inside BridgeLink.** Derby lives in the same process as the server, so no other tool can open the database while BridgeLink runs, and backing it up means copying its files.
- **Lacks features.** The default Derby database does not support the JSON value type in **Lookup Manager** or regex search in **Message Browser**. SQL Server does not support regex search either.
- **Sized for evaluation.** For production message volumes, use PostgreSQL, MySQL, Microsoft SQL Server or Oracle.

An external database also fits the backup, monitoring and security routines your team already runs.

Prepare the Database

1. On your database server, create an empty database for BridgeLink, for example `bridgelinkdb`.
2. Create a login for BridgeLink that owns that database and can create tables in it.

BridgeLink creates all of its tables the first time it starts. You do not run any scripts.

Point BridgeLink at the Database

Steps 2 and 3 use the values in this table:

Database	database value	database.url pattern
PostgreSQL	postgres	<code>jdbc:postgresql://dbhost:5432/bridgelinkdb</code>
MySQL	mysql	<code>jdbc:mysql://dbhost:3306/bridgelinkdb</code>
Microsoft SQL Server	sqlserver	<code>jdbc:sqlserver://dbhost:1433;databaseName=bridgelinkdb</code>

Database	database value	database.url pattern
Oracle	oracle	jdbc:oracle:thin:@//dbhost:1521/serviceName for a service name (pluggable databases), or jdbc:oracle:thin:@dbhost:1521:SID for a SID

SQL Server and certificates. BridgeLink 26.9 connects to SQL Server with encryption on and checks the server's certificate. If your SQL Server's certificate is self-signed, comes from an internal certificate authority, or does not match the host name in the URL, BridgeLink cannot connect. Either add that certificate, or its certificate authority, to the Java trust store BridgeLink uses, or add `;trustServerCertificate=true` to the end of the URL. That keeps the connection encrypted but does not check which server answered.

Replace `dbhost`, the port and the database name (for Oracle, the service name or SID) with your own. A finished PostgreSQL example looks like this:

```
database = postgres
database.url = jdbc:postgresql://db01.example.org:5432/bridgelinkdb
database.username = bridgelink
database.password = your-password-here
```

1. Open the settings file in a text editor as root. This guide uses `vi`; any editor you prefer works the same way:

```
sudo vi /opt/BridgeLink/conf/mirth.properties
```

2. Press `i` to start editing. Find the line `database = derby` and change `derby` to the value for your database from the table above.
3. Change the `database.url` line to your database's address, using the pattern in the table.
4. Set `database.username` to the BridgeLink login.
5. Set `database.password` to that login's password.
6. Press `Esc`, type `:wq`, and press `Enter` to save the file and quit.
7. Check that the file still belongs to the service account and that others cannot read it:

```
sudo chown bridgelink:bridgelink /opt/BridgeLink/conf/mirth.properties
sudo chmod 600 /opt/BridgeLink/conf/mirth.properties
```

Protect this file. `mirth.properties` holds the database password, so only root and the service account should be able to read it. The service account must also be able to write to it: on first start, BridgeLink writes new keystore passwords into the file.

Leave the other lines as they are for now. The `https.port` and `http.port` lines hold your ports, and the `keystore` lines are managed by BridgeLink.

CHAPTER 5

The Service Account

The BridgeLink service runs in the background under one Linux account. That account decides what BridgeLink can reach on the machine and on your network.

Why Not Root

Channels can run scripts. If the service runs as **root**, any script in any channel has full control of the machine. So BridgeLink refuses to start as **root** by default, and the log shows:

```
ERROR: BridgeLink is running as root/Administrator.
```

The setting behind this is `server.allowRoot = false` in `conf/mirth.properties`. Leave it set to `false`.

The service account needs:

- Ownership of the BridgeLink folder, because BridgeLink writes its logs, its **appdata** folder and `conf/mirth.properties` there
- Read and write access to any folders your channels read from or write to
- Network access to the database and to the systems your channels connect to

Every install path in this guide sets this up for the **bridgelinek** account. The RPM package and the installer do it for you; the archive steps do it by hand.

How the Service Runs as the Account

The RPM package and the installer set the **blservice** service to run as **bridgelinek** with a small settings file, `/etc/systemd/system/blservice.service.d/10-bridgelinek-user.conf`. It sets `User=` and `Group=`, and also `NoNewPrivileges=true`, which stops channel scripts from gaining extra rights through **sudo** or similar tools.

They rewrite that file on every upgrade, so do not edit it. To change the account, or any other service setting, use `sudo systemctl edit blservice` as shown in **Changing the Account After Install**. Settings made that way take priority.

On the archive path, the service file you created sets `User=`, `Group=` and `NoNewPrivileges=true` directly.

Check Which Account BridgeLink Runs As

After the service starts, confirm which account it is running as:

```
pid="$(systemctl show -p MainPID --value blservice)"
[ "$pid" -gt 0 ] && ps -o user= -p "$pid" || echo "not running"
```

The command prints the account, which should be **bridgelink** (or the account you chose), or **not running** if the service is stopped. The account prints even while BridgeLink is failing and restarting, so also confirm that it started. The newest line that contains **Web server running at** is the one to trust:

```
sudo journalctl -u blservice | grep 'Web server running at' | tail -1
```

Changing the Account After Install

Suppose BridgeLink runs as one account and you switch it to another, for example from **bridgelink** to **svc-integration**. The files BridgeLink writes still belong to the old account. The new account cannot write its logs, its keystore or its settings, so the service fails at startup or soon after, and the log or the journal shows **Permission denied** errors.

To switch accounts and fix the permissions:

1. Create the new account, if it does not exist yet:

```
sudo useradd -r -s /sbin/nologin svc-integration
```

2. Stop the service:

```
sudo systemctl stop blservice
```

3. Open an override for the service:

```
sudo systemctl edit blservice
```

4. Set the new account and group. The installer already sets **NoNewPrivileges=true**, and your override keeps it; repeating the line in the override makes that explicit:

```
[Service]
User=svc-integration
Group=svc-integration
NoNewPrivileges=true
```

5. Save and close the editor.
6. Give the new account the BridgeLink folder:

```
sudo chown -R svc-integration:svc-integration /opt/BridgeLink
```

7. Give the new account the same access to any folders your channels use.
8. Start the service:

```
sudo systemctl start blservice
```

9. Check the account with the command in **Check Which Account BridgeLink Runs As** above.

CHAPTER 6

Starting BridgeLink for the First Time

Start the service once the database is connected and the service account is set.

1. Start BridgeLink and set it to start at boot:

```
sudo systemctl enable --now blservice
```

2. Wait about a minute.
3. Look at the state of the service:

```
systemctl status blservice
```

4. Find the newest line that contains `Web server running at`:

```
grep 'Web server running at' /opt/BridgeLink/logs/mirth.log | tail -1
```

5. Check that the time on that line is after the start you just made. That line is the sign that BridgeLink started.
6. Find the newest line that contains `Running`, to check the database name and Java version:

```
grep 'Running ' /opt/BridgeLink/logs/mirth.log | tail -1
```

7. Confirm the line names your database, not `derby`. It also names the Java version and the operating system.

An active (running) state is not enough. If BridgeLink fails to start, `systemctl status` can show the service as `failed`, or, for a service set to restart after each failure, as `active (running)` most of the time even while startup keeps failing. The `Web server running at` line is the reliable sign that BridgeLink started. The `Running` line alone is not enough either: BridgeLink still prints it when it cannot read its keystore, and port `8443` then refuses connections.

If the `Web server running at` line is missing or old, look for the reason in the log, and see [Troubleshooting](#):

```
grep ERROR /opt/BridgeLink/logs/mirth.log | tail -5
```

The service's own console output is in the journal: `journalctl -u blservice`.

On this first start, BridgeLink:

- Creates its tables in your database

- Creates a keystore holding a new self-signed certificate for port **8443**, and writes new keystore passwords into `conf/mirth.properties`
- Creates the `appdata/server.id` file that holds this server's unique ID

If an RPM install ran for more than a few seconds before you stopped it, the keystore and `server.id` already exist and BridgeLink keeps them.

Memory for the BridgeLink Server

BridgeLink starts with a 256 MB Java heap, which is enough for a personal or test install but small for production. To raise it:

1. Find the memory settings file for your package:
 - For the RPM package and the installer (.sh), whose service runs `blservice`:
`/opt/BridgeLink/blservice.voptions`
 - For the archive, whose service file in [Installing BridgeLink](#) runs `blserver`:
`/opt/BridgeLink/blserver.voptions`
2. Open that file as root. For the RPM package and the installer (.sh):

```
sudo vi /opt/BridgeLink/blservice.voptions
```

For the archive:

```
sudo vi /opt/BridgeLink/blserver.voptions
```

3. Press `i` to start editing. Change the `-Xmx` line, for example to `-Xmx1g`. Leave memory free for the operating system and the database.
4. Press `Esc`, type `:wq`, and press `Enter` to save the file and quit.
5. Restart BridgeLink:

```
sudo systemctl restart blservice
```

CHAPTER 7

Installing WebAdmin

WebAdmin is the browser-based tool you use to manage BridgeLink. It runs as its own small service, next to BridgeLink, and nobody needs to install anything on their own computer to use it. WebAdmin is a small web application run by Node.js: it serves the pages you see in the browser and passes each request on to BridgeLink. It contains no Java.

On Linux, WebAdmin is a separate step with its own installer. You run it after BridgeLink is installed, from the `BridgeLink-WebAdmin-install.tar.gz` archive that every BridgeLink package puts in `/opt/BridgeLink`.

Run the WebAdmin Installer

WebAdmin needs Node.js 22, and the installer checks. If Node.js is missing or a different version, the installer shows a **Choose an option:** menu. What happens next depends on your server:

- If Node.js is missing, option 1 adds the NodeSource and N|Solid package repositories to the server and installs Node.js 22 system-wide. That changes Node.js for every program on the server.
- If you run Ubuntu or Debian and the distribution's Node.js is older than 22, option 1 replaces it for the whole server
- On Red Hat family systems, or when the Node.js at `/usr/bin/node` is newer than 22, choosing option 1 stops the installer without changing anything and tells you what to do: remove that Node.js, or install Node.js 22 in a folder of its own. Option 2 still works once Node.js 22 is in a folder of its own.
- If other software on the server uses Node.js, install Node.js 22 in its own folder. Then choose option 2 at the menu, or start the installer with `sudo NODEJS_HOME=<folder that holds Node.js 22's bin/node> bash bridgelink-webadmin/install.sh` instead of the command in step 3.

1. Go to the BridgeLink folder:

```
cd /opt/BridgeLink
```

2. Extract the WebAdmin archive there. It creates a `bridgelink-webadmin` folder:

```
sudo tar -xzf BridgeLink-WebAdmin-install.tar.gz
```

3. Run the installer:

```
sudo bash bridgelink-webadmin/install.sh
```

4. Read the license terms it points to, `LICENSE` and `SUPPLEMENTAL-TERMS.txt`.

5. At **Do you accept the license terms? [y/N]:**, type **y**.
6. If the installer shows **Choose an option:** for Node.js, press **Enter** to choose **1) Install Node.js 22 now**. Read the list above first: this option changes Node.js for the whole server.
7. At **Enter the port for BridgeLink WebAdmin [default: 8444]:**, press **Enter**.
8. At **Select the BridgeLink deployment model:**, type **1** for **Same Host (Recommended)**.
9. At **Enter the BridgeLink engine HTTPS port [default: 8443]:**, press **Enter** unless you changed BridgeLink's HTTPS port. If you did, type that port.

```
admin@bridgelink:~/bridgelink-webadmin$ sudo bash install.sh
[install] Detected platform: linux

BridgeLink WebAdmin is licensed under the Business Source License 1.1
and the BridgeLink WebAdmin Supplemental Terms. The full documents are
included in this package:
  /home/admin/bridgelink-webadmin/LICENSE
  /home/admin/bridgelink-webadmin/SUPPLEMENTAL-TERMS.txt

Do you accept the license terms? [y/N]: y
[install] License terms accepted.
[install] Node.js: v22.23.2 at /usr/local/bin/node
[install] Copying files to /opt/bridgelink-webadmin...
[install] Files copied to /opt/bridgelink-webadmin
Enter the port for BridgeLink WebAdmin [default: 8444]:

Select the BridgeLink deployment model:
  1) Same Host (Recommended)
      I want WebAdmin to manage the BridgeLink installation on this server.
  2) Dedicated Host - Single Server
      I want WebAdmin to manage a single BridgeLink installation on another server.
  3) Dedicated Host - Multi-Server
      I want WebAdmin to manage BridgeLink installations on multiple servers.
Selection [1]: 1
```

Figure 1: A Linux terminal running the WebAdmin install.sh, showing the port prompt answered with the default 8444 and the deployment model menu with 1) Same Host (Recommended), 2) Dedicated Host – Single Server and 3) Dedicated Host – Multi-Server, with 1 entered.

While it creates the WebAdmin certificate, the installer prints a long run of **+.+.+.+.+.+** characters. Expect it: it is not an error.

The installer:

- Copies WebAdmin to **/opt/bridgelink-webadmin** and writes **/opt/bridgelink-webadmin/webadmin.conf**
- Creates a self-signed certificate for WebAdmin at **/opt/bridgelink-webadmin/cert.pem** and **key.pem**
- Installs and starts the systemd service **bridgelink-webadmin**
- Installs the **blwebadmin** command in **/usr/local/bin**

The WebAdmin service runs as **root**, and everything the installer creates belongs to **root**, so stopping or restarting WebAdmin takes **sudo**. Control the service with **systemctl**, for example **sudo systemctl restart bridgelink-webadmin**; **stop**, **start** and **status** work the same way. On Red Hat family systems, **sudo** cannot find the **blwebadmin** command, so this guide does not use it. WebAdmin writes its output and errors to the systemd journal: **sudo journalctl -u bridgelink-webadmin -n 50**.

If a firewall runs on the server, allow port **8444** so users on other computers can reach WebAdmin. With **firewalld**, which Red Hat family systems use, run:

```
sudo firewall-cmd --add-port=8444/tcp --permanent
sudo firewall-cmd --reload
```

Same Host finds BridgeLink's port. Same Host (Recommended) reads BridgeLink's HTTPS port from `/opt/BridgeLink/conf/mirth.properties` and offers it as the default in step 9. If you see a warning that it could not find the port and BridgeLink uses another one, type that port at the prompt in step 9. To change it later, open the file with **sudo vi /opt/bridgelink-webadmin/webadmin.conf**, press **i**, correct the port in **BRIDGELINK_SERVER_URL**, press **Esc**, type **:wq** and press **Enter**, then run **sudo systemctl restart bridgelink-webadmin**.

To put WebAdmin on a different machine, run the same installer there and choose **2) Dedicated Host – Single Server** with this server's address.

Sign In for the First Time

1. Open **https://your-server-name:8444** in a browser.
2. If the browser shows a certificate warning, click **Advanced**, then **Proceed to ... (unsafe)** in Chrome or **Continue to ... (unsafe)** in Edge. In Firefox, click **Advanced**, then **Accept the Risk and Continue**. Read the note after this step.

When a browser stops letting you click through. WebAdmin tells browsers to accept only a valid certificate for its server name and every name under it, on every port, for two years. This is called HSTS. A browser starts enforcing it the first time it opens WebAdmin with a certificate it trusts, for example after you install your own certificate (see [Replacing the Certificates](#)). Clicking through a warning does not start it. **localhost** and IP addresses are never affected. Once a browser enforces HSTS for a name, it also refuses BridgeLink's default certificate on port **8443** at that name.

3. In **Username**, type **admin**.
4. In **Password**, type **admin**.
5. Click **Login**.
6. On **Welcome to BridgeLink**, leave **Username** as **admin**.

7. In **New Password**, type a new password of at least 8 characters, with an uppercase letter, a lowercase letter, a number, and a special character such as **!, @, #, \$, %** or *****. A hyphen, period or space does not count as a special character, and the password cannot be **admin**. WebAdmin lists only the rules that **admin** breaks, so check your password against all of these.
8. Type it again in **Confirm Password**.
9. The profile fields on **Welcome to BridgeLink** are optional; fill them in if you want to.
10. Click **Save & Continue**.



Figure 2: The WebAdmin sign-in screen in a browser, showing the Server field filled in and read-only for a Same Host install, the Username and Password fields, and the Login button.

With **Same Host (Recommended)**, the **Server** field is filled in and cannot be changed: WebAdmin talks only to the BridgeLink server on the same machine. To point it somewhere else, change `BRIDGELINK_SERVER_URL` in `/opt/bridgelink-webadmin/webadmin.conf` and run `sudo systemctl restart bridgelink-webadmin`.

More WebAdmin Setups

For WebAdmin on another machine, one WebAdmin for several servers, your own certificate, and every setting, see the **BridgeLink WebAdmin Setup Guide** on the BridgeLink download page, <https://www.bridgelink.net/download>.

CHAPTER 8

Finding the Server ID

Every BridgeLink server has a unique server ID, created the first time it starts. Plugin licenses are issued against this ID, so Innovar Healthcare asks for it when you license a plugin.

In WebAdmin

1. Sign in to WebAdmin.
2. In the left sidebar, scroll to the bottom and, under **Help**, click **About BridgeLink**.
If **Update Available** opens instead, click **Got it**, then click **About BridgeLink** again.
3. Click the copy icon beside **Server ID**.

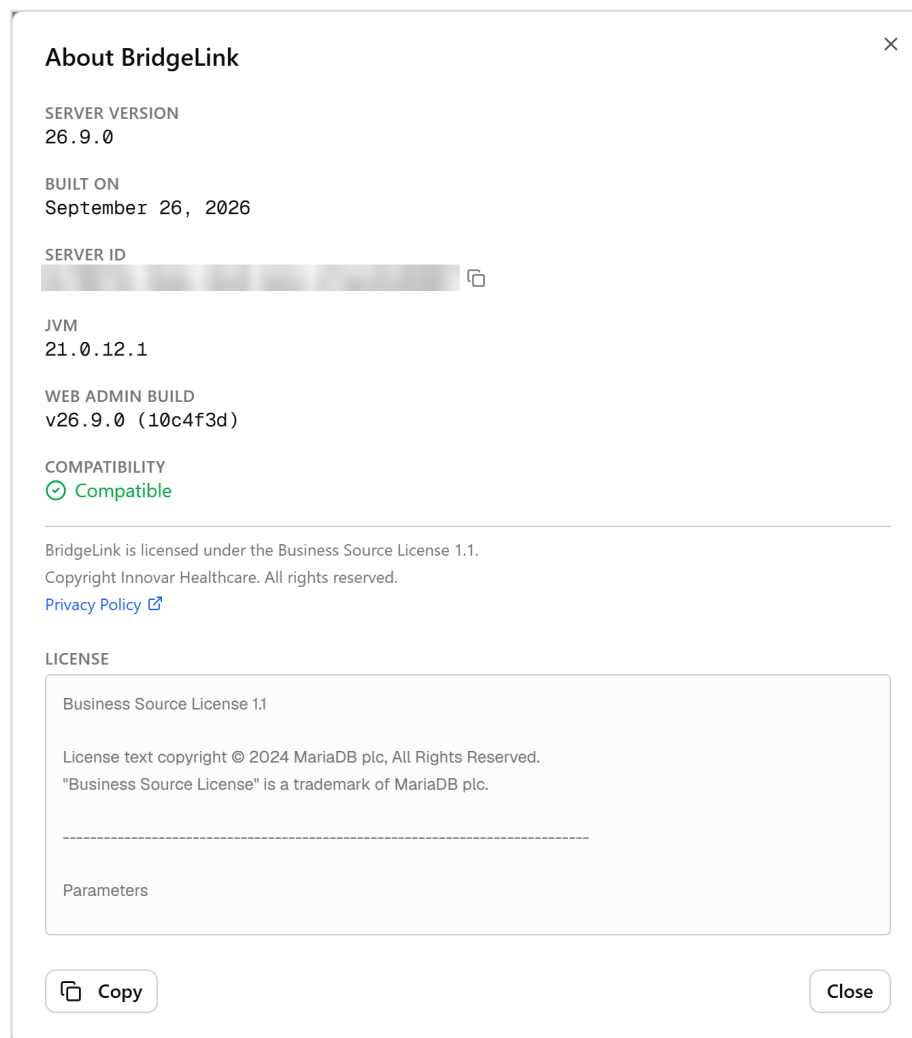


Figure 3: The About BridgeLink dialog in WebAdmin, showing the Server Version, Built On, Server ID (blurred here, with its copy icon), JVM, Web Admin Build and Compatibility rows.

On Disk

The same ID is in `/opt/BridgeLink/appdata/server.id`, on a line that reads `server.id =` followed by the ID:

```
sudo cat /opt/BridgeLink/appdata/server.id
```

Back up `/opt/BridgeLink/appdata/server.id`, `/opt/BridgeLink/appdata/keystore.jks` and `/opt/BridgeLink/conf/mirth.properties` together. The keystore holds the key BridgeLink uses to encrypt data, and `mirth.properties` holds the keystore's passwords. If you rebuild the server and put `server.id` back, the server keeps its ID and its plugin licenses. Without the file, BridgeLink creates a new ID. The RPM package starts BridgeLink as soon as it installs, so put `server.id` back with BridgeLink stopped:

1. Stop BridgeLink:

```
sudo systemctl stop blservice
```

2. Copy your saved file to `/opt/BridgeLink/appdata/server.id`.
3. Give the file to the service account:

```
sudo chown bridgelink:bridgelink /opt/BridgeLink/appdata/server.id
```

4. Start BridgeLink:

```
sudo systemctl start blservice
```

CHAPTER 9

Replacing the Certificates

A new install uses two self-signed certificates (certificates the server creates for itself, which browsers do not trust), which is why browsers warn the first time. Before you use BridgeLink in production, replace them with certificates from your organization's certificate authority (CA).

Certificate	Port	Who sees it	Where it lives
WebAdmin	8444	Every browser that opens WebAdmin	/opt/bridgelink-webadmin/cert.pem and key.pem
BridgeLink server	8443	WebAdmin and any program that calls the BridgeLink API	The mirthconnect entry in /opt/BridgeLink/appdata/keystore.jks

Most organizations replace the WebAdmin certificate first, since that is the one users see in their browsers.

Replace the WebAdmin Certificate

Before you replace WebAdmin's certificate, read **When a browser stops letting you click through** under [Sign In for the First Time](#) in [Installing WebAdmin](#).

You need the certificate and its private key as PEM files (text files that begin with -----BEGIN). Put any intermediate certificates in the certificate file after the server certificate. The private key file must not have a password.

1. Stop WebAdmin:

```
sudo systemctl stop bridgelink-webadmin
```

2. Keep copies of the current files:

```
sudo cp /opt/bridgelink-webadmin/cert.pem /opt/bridgelink-webadmin/cert.pem.bak
sudo cp /opt/bridgelink-webadmin/key.pem /opt/bridgelink-webadmin/key.pem.bak
```

3. Copy your certificate over cert.pem:

```
sudo cp your-certificate.pem /opt/bridgelink-webadmin/cert.pem
```

4. Copy your private key over key.pem:

```
sudo cp your-private-key.pem /opt/bridgelink-webadmin/key.pem
```

5. Start WebAdmin:

```
sudo systemctl start bridgelink-webadmin
```

6. Open WebAdmin in a browser and confirm there is no certificate warning.

To Undo the Swap

If browsers cannot open WebAdmin after the swap, put the old certificate back:

1. Stop WebAdmin:

```
sudo systemctl stop bridgelink-webadmin
```

2. Copy the two backup files back:

```
sudo cp /opt/bridgelink-webadmin/cert.pem.bak /opt/bridgelink-webadmin/cert.pem
sudo cp /opt/bridgelink-webadmin/key.pem.bak /opt/bridgelink-webadmin/key.pem
```

3. Start WebAdmin:

```
sudo systemctl start bridgelink-webadmin
```

Port **8444** serves the original certificate again.

Browsers that opened WebAdmin by its server name with the new certificate refuse the old self-signed one. Fix the new certificate instead, or clear the browser's HSTS entry for the server name: open `chrome://net-internals/#hsts` in Chrome or `edge://net-internals/#hsts` in Edge, and use **Delete domain security policies**. In other browsers, clear the browsing data for that server name.

Replace the BridgeLink Server Certificate

BridgeLink keeps its certificate in a keystore file under the name `mirthconnect`. The same keystore also holds the key BridgeLink uses to encrypt data, so replace only the `mirthconnect` entry. Never swap in a different keystore file.

You need the certificate and private key as a PKCS12 file (a single file, ending in `.p12` or `.pfx`, that bundles a certificate with its private key) and its password. The commands use `keytool`, which comes with Java. Each command below has placeholders in angle brackets, such as `<your keystore.storepass value>`. Replace each one, including the angle brackets, with the value described under the command. If a value contains characters the shell treats specially, put it in single quotes.

1. Stop BridgeLink:

```
sudo systemctl stop blservice
```

2. Back up the keystore:

```
sudo cp /opt/BridgeLink/appdata/keystore.jks /root/keystore.jks.bak
```

- Note the values of `keystore.storepass` and `keystore.keypass`. You use them in steps 5 and 6:

```
sudo grep '^keystore\.' /opt/BridgeLink/conf/mirth.properties
```

- List the entry name inside your PKCS12 file. Type the file's password when keytool asks; nothing shows as you type:

```
keytool -list -keystore server.p12 -storetype PKCS12
```

- Remove the old certificate entry. Do this before step 6: otherwise keytool asks whether to overwrite the existing entry, and pressing **Enter** keeps the old certificate:

```
sudo keytool -delete -alias mirthconnect \
  -keystore /opt/BridgeLink/appdata/keystore.jks -storetype JCEKS \
  -storepass <your keystore.storepass value>
```

Replace `<your keystore.storepass value>` with the `keystore.storepass` value that step 3 showed.

- Import yours as `mirthconnect`:

```
sudo keytool -importkeystore \
  -srckeystore server.p12 -srcstoretype PKCS12 \
  -srcstorepass <your .p12 file password> \
  -srcalias <the entry name from step 4> \
  -destkeystore /opt/BridgeLink/appdata/keystore.jks -deststoretype JCEKS \
  -deststorepass <your keystore.storepass value> \
  -destalias mirthconnect \
  -destkeypass <your keystore.keypass value>
```

Replace the placeholders as follows:

- `<your .p12 file password>`: the password of your PKCS12 file
 - `<the entry name from step 4>`: the entry name that step 4 listed
 - `<your keystore.storepass value>`: the `keystore.storepass` value that step 3 showed
 - `<your keystore.keypass value>`: the `keystore.keypass` value that step 3 showed
- Steps 5 and 6 print a warning that recommends migrating the keystore to PKCS12. Ignore it: BridgeLink reads the keystore as JCEKS, so do not convert it.

- Give the keystore back to the service account:

```
sudo chown bridgelink:bridgelink /opt/BridgeLink/appdata/keystore.jks
```

- Start BridgeLink:

```
sudo systemctl start blservice
```

9. Open <https://your-server-name:8443> in a browser and confirm the browser shows your certificate.

If BridgeLink does not start, or port 8443 does not answer: stop BridgeLink, run `sudo cp /root/keystore.jks.bak /opt/BridgeLink/appdata/keystore.jks`, repeat the `chown` from step 7, and start BridgeLink.

WebAdmin keeps working through the change, with no change to its own settings.

CHAPTER 10

The Data Pruner

BridgeLink keeps the messages each channel processes until the Data Pruner removes them. The Data Pruner is off on a new server, so turn it on before you send real traffic through BridgeLink; otherwise the database grows with every message.

It has two parts, and both have to be set:

- **The schedule, for the whole server:** in WebAdmin, in the left sidebar under **Administration**, click **Settings**, then the **Data Pruner** tab
- **How long each channel keeps messages:** on each channel's **Summary** tab, under **Message Pruning**. A channel whose pruning fields are blank is never pruned, even when the schedule is on.

Choose ages that match your organization's record-keeping rules.

Turn On the Schedule

1. In WebAdmin, in the left sidebar under **Administration**, click **Settings**.
2. Click the **Data Pruner** tab.
3. Under **Schedule**, set **Enable** to **Yes**.
4. Choose a **Schedule Type** and set when it runs: **Interval** runs every set number of hours or minutes, **Time** runs once a day, and **Cron** takes cron expressions. Once a day at a quiet hour suits most servers.
5. Click **Save**.

Settings

Server Administrator Tags Configuration Map Resources **Data Pruner** Message Trends Version History

Status

Current State: **Not running**

Current Process: -

Last Process: -

Next Process: Scheduled Monday, Sep 28, 1:00:00 AM

Schedule

Enable: ☒ Yes ☐ No

Schedule Type: **Time**

Time (HH : MM): **1** : **0**

Prune Settings

Block Size: **1000** (50-10000, recommended: 1000)

Prune Events: ☒ Yes ☐ No

Prune Event Age: **90** days

Archive Settings

Enable Archiving: ☐ Yes ☒ No

Figure 4: The Data Pruner tab of the Settings page, showing the Status section, the Schedule section with Enable set to Yes and Schedule Type set to Time, the Prune Settings section and the Archive Settings section.

Set How Long Each Channel Keeps Messages

1. In the left sidebar under **Build**, click **Channels**.
2. Double-click the channel to open it in the channel editor.
3. On the **Summary** tab, under **Message Pruning**, enter the number of days in **Prune metadata after** and **Prune content after**. Content cannot be kept longer than metadata.
4. Click **Save**.
5. Repeat for each channel.

Message Pruning

Prune metadata after **30** days

Prune content after **7** days

☒ Archive messages before pruning

☐ Prune errored messages

(incomplete, errored, and queued messages will not be pruned)

Figure 5: The Message Pruning section of a channel's Summary tab, showing Prune metadata after set to 30 days, Prune content after set to 7 days, Archive messages before pruning checked and Prune errored messages cleared.

Incomplete and queued messages are never pruned. Errored messages are kept too, unless you check **Prune errored messages**.

CHAPTER 11

Troubleshooting

Symptom	Likely cause	What to do
The service stops at once, and logs/mirth.log or journalctl -u blservice shows ERROR: BridgeLink is running as root/Administrator.	The service runs as root	Set User= and Group= to the service account. See The Service Account .
The service stops at once, and the log shows embedded Derby requires Java 21+ as of 26.9; upgrade Java or switch to an external database	The server is still on the default Derby database, running on Java older than 21	Connect an external database. See Point BridgeLink at the Database .
The service fails, and the log or journal shows Permission denied	The service account does not own the files BridgeLink writes, often after an account change	Give the account the BridgeLink folder. See Changing the Account After Install .
Browsers cannot reach port 8443, and the log has Could not initialize security settings	BridgeLink could not save its keystore passwords to conf/mirth.properties on first start	Give the service account ownership of conf/mirth.properties and restart the service.
The log shows aborting startup, there is no Web server running at line, and systemctl status blservice shows failed, or active (running) while it keeps restarting	The database connection is failing: the database.url, login or password is wrong, or a firewall blocks the database port. SQL Server only: its certificate is self-signed, comes from an internal certificate authority, or does not match the host name in the URL, which BridgeLink 26.9 does not accept by default.	Check the four database lines in conf/mirth.properties and test the login from the BridgeLink server. See SQL Server and certificates in Choosing and Connecting the Database . BridgeLink retries twice, 10 seconds apart, before it gives up. Once you fix the cause, run <code>sudo systemctl restart blservice</code> .
WebAdmin sign-in shows "Could not reach the BridgeLink server. Verify the server address is correct and the server is running."	BridgeLink is not running, or WebAdmin points at the wrong BridgeLink port	Start BridgeLink. Compare the port in BRIDGELINK_SERVER_URL in /opt/bridgelink-webadmin/webadmin.conf with BridgeLink's HTTPS port. If they differ, correct it and run <code>sudo systemctl restart bridgelink-webadmin</code> .

Symptom	Likely cause	What to do
WebAdmin sign-in fails while BridgeLink is running, and <code>sudo journalctl -u bridgelink-webadmin</code> shows <code>Upstream fetch failed for users/_login: TypeError: fetch failed</code>	WebAdmin is running on a Node.js version other than 22	Run the WebAdmin installer again (<code>sudo bash bridgelink-webadmin/install.sh</code> from <code>/opt/BridgeLink</code>), let it set up Node.js 22, and answer the prompts as before.
The browser warns about the certificate	WebAdmin still uses its self-signed certificate	Click through the warning, or replace the certificate. See Replace the WebAdmin Certificate . If the browser offers no way to continue, it is enforcing HSTS for that name. See When a browser stops letting you click through under Sign In for the First Time , and To Undo the Swap in Replacing the Certificates to clear it.
<code>sudo: blwebadmin: command not found</code>	On Red Hat family systems, <code>sudo</code> does not search <code>/usr/local/bin</code> , where the installer puts <code>blwebadmin</code>	Use <code>systemctl</code> instead, for example <code>sudo systemctl restart bridgelink-webadmin</code> . If <code>systemctl</code> says the unit <code>bridgelink-webadmin</code> is not found, the WebAdmin installer did not finish: run it again.

For more WebAdmin problems, see [Troubleshooting](#) in the **BridgeLink WebAdmin Setup Guide**.

Support

Customers with an active Innovar Healthcare support plan can contact support at support@innovarhealthcare.com. Include the BridgeLink version, the package you installed, your Linux distribution, what you were doing, and the most recent part of `logs/mirth.log`. `mirth.log` can contain message content, which may include patient data. Remove patient data before you send a log, or ask support for a secure way to send it.