

OpenLiteSpeed Tab

Member OpenLiteSpeed

Tab: OpenLiteSpeed.
Sub Tabs: Overview, Recovery Wizard

Use the **OpenLiteSpeed** tab to manage standalone OpenLiteSpeed servers in a hosting pool. It gives you one place to check the web server, create recovery points, restore sites, and move sites to another compatible OpenLiteSpeed server.

This tab is for servers running OpenLiteSpeed directly. cPanel servers use the cPanel tools instead.

What is included

Area	What you use it for
Overview	Check server health, open the WebAdmin panel, reload or restart OpenLiteSpeed, test configuration, update PHP packages, and create a server config backup.
Backups	Create site, web-file, database, and full recovery points for OpenLiteSpeed applications.
Transfer Wizard	Move an OpenLiteSpeed site to another compatible OpenLiteSpeed member.

Community and Pro

Feature	Community	Pro
View detected OpenLiteSpeed status	Included	Included
Open WebAdmin link and view basic server details	Included	Included
Refresh detected applications	Included	Included
View linked databases when discovered	Included	Included
Create OpenLiteSpeed backups	Upgrade required	Included

Feature	Community	Pro
Restore OpenLiteSpeed backups	Upgrade required	Included
Transfer sites between OpenLiteSpeed servers	Upgrade required	Included
Automatic DNS cutover after restore or transfer	Upgrade required	Included

Pro turns this tab from a read-only view into a recovery and migration workspace. It is designed for operators who need to move sites quickly, keep usable recovery points, and reduce manual rebuild work.

Overview

Open **Pool**, choose the OpenLiteSpeed member, then select **OpenLiteSpeed**.

The **Overview** area shows:

- WebAdmin access for the selected member.
- Current service state and key server details.
- Actions for reload, restart, configuration test, backup, upgrade, and PHP package updates.
- Recent log output when available.

Use **Config test** before restarting a busy server. If the test fails, fix the configuration first so the web server does not reload into a broken state.

Backups

Open **Backups** to work with detected OpenLiteSpeed applications.

Each application card shows the site name, document root, PHP and TLS information when available, linked databases when discovered, and the latest recovery point.

To create a recovery point:

1. Select the application.
2. Choose **Backup now**.
3. Pick the backup type.
4. Confirm the storage estimate and start the backup.

Common backup types:

Backup type	Use it when
Full site backup	You want the safest recovery point for a real application.

Backup type	Use it when
Files only	You only changed web files, themes, plugins, uploads, or static content.
Databases only	You only need a database snapshot for a linked schema.

Recovery points can be restored, downloaded, inspected, or deleted from the same workspace.

OpenLiteSpeed Transfer Wizard

Use the OLS **Transfer Wizard** when you want to move an OpenLiteSpeed site to another compatible OpenLiteSpeed member.

Transfer a Site (OpenLiteSpeed to OpenLiteSpeed)

To transfer a site:

1. Open **Transfer Wizard**.
2. Select the source application.
3. Choose a compatible destination member.
4. Confirm whether DNS will be updated manually or automatically.
5. Start the transfer and watch the progress panel.

Destination servers must be compatible. cPanel servers are not valid OpenLiteSpeed transfer targets, even if they also run LiteSpeed or expose similar services.

DNS cutover

After a restore or transfer, DNS still decides where visitors go.

Option	Availability	What happens
Manual DNS update	Community and Pro	The site is moved or restored, then you update the A record yourself.
Automatic DNS update	Pro	The platform updates the saved DNS provider record after the operation completes.

Automatic DNS is useful when you want a cleaner migration flow and already have a DNS key saved for the domain or account.

Good operating habits

- Run a full backup before major changes.
- Use configuration test before restarting OpenLiteSpeed.
- Keep enough storage for at least one known-good recovery point per important site.
- Confirm DNS after any restore or transfer.

- Use Pro for production migration and recovery workflows where speed matters.

Direct Transfer security model

Direct Transfer is not an open file receiver and it is not general-purpose SSH access. The API acts as the control plane for a single copy operation:

1. **The authenticated operator starts a transfer.** The dashboard calls the Worker with the selected source member, destination member, cPanel username, route mode, and DNS handling choice. The Worker checks plan access, pool ownership, WHM/cPanel compatibility, account state, agent versions, destination disk, and route availability before creating the run.
2. **API creates a one-use session.** The Worker mints a random session ID, records the exact source node, destination node, account username/domain, expected byte count, destination address, source IP filter, status, and expiry. Session APIs reject unknown, expired, terminal, wrong-role, or wrong-status sessions.
3. **Only enrolled agents can advance it.** Source and destination agents call back using their normal enrolled heartbeat credentials. The Worker verifies that the calling node is the recorded source or destination for that session before accepting prepare, public-key registration, progress, transfer-complete, or restore-complete updates.
4. **The source generates an ephemeral SSH key.** The source agent creates a session-scoped keypair for this transfer. The private key stays on the source server. The public key is sent to the Worker, which queues an authorised job for the destination agent.
5. **Destination installs a restricted receiver key.** The destination agent writes the public key to root `authorized_keys` with a forced command for chunk receive only, plus `from=<source-ip>`, `no-port-forwarding`, `no-X11-forwarding`, `no-agent-forwarding`, `no-pty`, and `no-user-rc`. A matching `session.json` must exist before any chunk is accepted.
6. **Chunks are individually checked.** The source agent streams fixed-size chunks over SSH. The forced receiver accepts only `sessionId`, chunk index, and chunk SHA-256, reads the chunk from stdin, verifies the chunk hash, and writes it under that session's temporary receive directory.
7. **Archive integrity is checked before restore.** After transfer, the destination agent assembles the archive and verifies the final SHA-256 recorded on the session/run before calling `restorepkg`.
8. **Session cleanup is automatic.** On success, failure, expiry, or revoke, the destination removes the matching `authorized_keys` line and deletes the temporary session directory. The Worker will also queue `cpanel_account_sync_revoke` if the run fails before normal cleanup.

The session lifetime is time-limited. It has a 30-minute minimum and scales with package size using a conservative transfer-rate estimate, so a large package has enough time to complete without leaving a permanent receiver open.