

Ghost Update Fix — The Exact Playbook

Permissions + Ownership + Ghost-CLI mismatch (Ubuntu 24.04, Node v22)

Outcome: Ghost 6.18.1 □ Ghost-CLI 1.28.4 □ Service healthy □

1) What broke (symptoms)

- `ghost update` refused to run due to file permissions + content ownership errors
- CLI warned: "You are running an outdated version of Ghost-CLI"
- `npm list -g` showed the new CLI, but `ghost --version` still showed the old one

2) Golden rule (safe ownership model)

- Keep the install folder owned by your admin user (ex: `adam`)
- Make only `content/` owned by `ghost:ghost` (Ghost must write uploads & generated files)
- Don't blindly chown the whole install to ghost — it can break Ghost-CLI checks

Quick backup first (stored under /home/adam/):

```
sudo mkdir -p /home/adam/backups/ghost/myconversationswithai/$(date +%F-%H%M%S) && \
BKP="/home/adam/backups/ghost/myconversationswithai/$(date +%F-%H%M%S)" && \
```

3) Fix what Ghost-CLI complained about

Run inside the Ghost install folder:

- Fix file permissions (exclude versions/)
- Set content ownership to ghost:ghost

```
sudo find ./ ! -path "./versions/*" -type f -exec chmod 664 {} \;
sudo chown -R ghost:ghost ./content
```

4) CLI mismatch diagnosis (PATH + npm root)

Key idea: you can have the right package installed but the wrong `ghost` binary being executed.

Check where `ghost` is coming from and where npm installs globals.

```
which -a ghost
type -a ghost
npm list -g --depth=0 | grep ghost-cli
npm root -g
npm prefix -g
```

5) The fix that made versions match

Because `ghost` was in /usr/bin, the correct move was to point it at the global Ghost-CLI under your active Node install (npm root -g).

```
sudo ln -sf "$(npm root -g)/ghost-cli/bin/ghost" /usr/bin/ghost
hash -r
/usr/bin/ghost --version
```

Tip: If `ghost --version` lies, trust `npm root -g` + the symlink target — not the printed banner.

Generated for: myconversationswithai • Feb 11, 2026

6) Verify: Is the sudo rule that prevents pain...