

iMobile

Verso

Installation and Administration Guide

Document ID	IM.VERSO.INT.02
Document version	1.0
Date of issue	2026-10-06
Applies to	Verso 2026-09-25 (4bac47a)
Language	English
Publisher	iMobile

Contents

1. General information	3
2. Environment requirements	3
3. Installation	3
3.1. Package contents	3
3.2. Installation steps	4
3.3. Environment variables	5
3.4. Connecting a model	5
4. Initial configuration	6
4.1. Administrator account	6
4.2. Providers	6
4.3. Users	6
4.4. API and MCP access keys	6
4.5. Protecting the channel and access	6
5. Verifying the installation	6
5.1. Container state	6
5.2. Service response	7
5.3. Licence status	7
5.4. Reference test	7
6. Updating	7
7. Backup and restore	7
8. Removal	8
9. Troubleshooting	8
9.1. Logs	8
9.2. Common failures	8
9.3. Recovering administrator access	9
9.4. Contacting the vendor	9

About this document

This guide covers installation, initial configuration, updating, backup and troubleshooting of Verso (release of 2026-09-25) installed on the customer's own servers. It is written for an administrator who is familiar with Linux and Docker.

In the cloud service the vendor hosts, configures and updates the product; the sections on installation, updating and backup do not apply to it. Points that do apply to the cloud service are marked in the text.

1. General information

Verso is shipped as a Docker image. A single container runs the nginx front proxy (container port 80, HTTP), the web interface, the API server, the MCP server, and the document parsing and export tools. No language model is included: the application calls an external provider with an API key or a local model server.

An on-premises installation has two layers:

- the base application image `verso-production`;
- an overlay from the on-premises package: the compiled licence verification module and the setting that turns licence enforcement on.

All mutable data is in the data directory: `app_data` next to `docker-compose.yml` on the host, `/app_data` in the container.

In operation an on-premises installation does not contact the vendor's servers: the licence is verified locally, there is no update check, and no usage analytics is sent.

2. Environment requirements

- A host with Docker Engine (with BuildKit) and Docker Compose v2 that runs linux/amd64 containers. The licence verification module is built for Linux x86_64 and Python 3.11, the runtime of the base image.
- Internet access while the base image is built from source. Not needed for the build when a pre-built image is supplied.
- A free host port for the web interface (5000 in the package).
- A TLS reverse proxy if users connect over HTTPS. The container itself serves HTTP only.
- A text model source: an API key of an external provider or a local model server (Ollama, LM Studio, another server with an OpenAI-compatible API) reachable from the container over the network.
- For local models the hardware is determined by the chosen model and model server. The vendor does not state minimum CPU, memory or disk figures for the application itself.

To check the tools:

```
docker version
docker compose version
```

3. Installation

3.1. Package contents

The `verso-onprem.tar.gz` archive contains:

File or directory	Purpose
<code>docker-compose.yml</code>	Definition of the <code>verso</code> service, port 5000, data and licence directories
<code>Dockerfile.onprem</code>	Overlay on the base image: licence module and enforcement mode
<code>verso_license/</code>	Compiled licence verification module
<code>license/</code>	Directory for the licence file
<code>README.md</code>	Short installation note

The licence file `verso-license.json` and the base image or application source are supplied separately.

3.2. Installation steps

Installation takes six steps, described in the subsections below: base image, unpacking the package, licence file, configuration, start, first sign-in.

3.2.1. Base image

The package overlay is built on an image named `verso-production`. If the vendor supplied a pre-built image, load it into the local Docker image store under that name. If source was supplied, place it in a directory named `verso` and build the image:

```
cd verso
docker compose build production
docker image ls verso-production
```

Note. The image name produced by `docker compose build production` is made of the directory name and the service name, so the source directory must be named `verso`.

3.2.2. Unpacking the package

Unpack the package archive into a working directory; run the commands that follow in the directory that holds `docker-compose.yml`:

```
tar -xzf verso-onprem.tar.gz
```

3.2.3. Licence file

Put the licence file into the `license` directory under the name `verso-license.json`:

```
cp /path/to/verso-license.json license/verso-license.json
```

3.2.4. Configuration

Check the `environment` section of `docker-compose.yml` against the table in "Environment variables" and add any missing lines. If needed, write the model settings to a `.env` file next to `docker-compose.yml` (see "Connecting a model").

3.2.5. Start

```
docker compose up -d --build
docker compose ps
```

3.2.6. First sign-in

Open `http://<host-address>:5000` in a browser and create the administrator account (see "Initial configuration").

3.3. Environment variables

Variables are set as `NAME=value` lines in the `environment` section of `docker-compose.yml`. Values written as `${NAME:-}` are taken from the `.env` file.

Variable	Value	Purpose
<code>LICENSE_MODE</code>	<code>onprem</code>	Turns licence enforcement on. Required
<code>LICENSE_FILE</code>	<code>/app_data/verso-license.json</code>	Path to the licence file in the container. Required
<code>MIGRATE_DATABASE_ON_STARTUP</code>	<code>true</code>	Creates and upgrades the database schema at start-up. Required: without it the credit accounting tables are not created and the admin console overview does not open
<code>CAN_CHANGE_KEYS</code>	<code>true</code> or <code>false</code>	With <code>false</code> , provider settings are hidden in the interface and taken from environment variables only
<code>CABINET_URL</code>	<code>none</code>	Removes the links to the vendor site account from the sign-in screen. Recommended on premises
<code>DATABASE_URL</code>	database address	External PostgreSQL or MySQL instead of the built-in SQLite
<code>MAX_CONCURRENT_GENERATIONS</code>	number	Limit of simultaneous generations per account, 3 by default
<code>DISABLE_IMAGE_GENERATION</code>	<code>true</code>	Switches off image search and generation
<code>MEMO_ENABLED</code>	<code>false</code>	Switches off presentation memory
<code>LITEPARSE_TESSDATA_PATH</code>	path in the container	Directory with Tesseract text recognition data for the languages needed; the image has data for English only
<code>LOG_LEVEL</code>	<code>INFO</code> , <code>DEBUG</code> , etc.	Log verbosity, <code>INFO</code> by default

Open e-mail registration (`ALLOW_EMAIL_SIGNUP`) and credit charging (`ENFORCE_CREDITS`) are off by default. Do not enable open registration on an on-premises installation: accounts are created by the administrator.

3.4. Connecting a model

The administrator chooses the text model, image and web search providers in the interface (Settings) or sets them with environment variables. Example `.env` files:

```
# external provider
LLM=openai
OPENAI_API_KEY=<key>
```

```
# local model on an Ollama server
LLM=ollama
OLLAMA_URL=http://<model-server-address>:11434
OLLAMA_MODEL=llama3.2:3b
```

The model server address must be reachable from the container; inside the container `localhost` is the container itself. Accepted LLM values: `openai`, `anthropic`, `google`, `vertex`, `azure`, `bedrock`, `deepseek`, `openrouter`, `fireworks`, `together`, `cerebras`, `litellm`, `lmstudio`, `ollama`, `custom`, `codex`. The `codex`

value (sign-in to a ChatGPT account) needs container port 1455 to be published, which the package file does not do.

To keep all data inside the customer's network, choose a local text model, switch images off or connect a local image server (ComfyUI, Open WebUI), and leave web search off or use your own SearXNG.

4. Initial configuration

4.1. Administrator account

On first opening, the program asks you to create the administrator account: a name of at least 3 characters without spaces and a password of at least 8 characters. This becomes the primary account; it cannot be deleted or demoted from the interface.

The primary administrator can also be created unattended by adding `AUTH_USERNAME` and `AUTH_PASSWORD` to the `environment` section before the first start.

4.2. Providers

After signing in, the administrator selects the text model provider and the image provider. Later the settings are changed under Settings: Text provider, Image provider, Web search provider. The settings apply to the whole installation and are stored in the database and in `userConfig.json` in the data directory; API keys are written there in clear text.

4.3. Users

Accounts are created by the administrator: Settings, Admin, Users tab, Add user (name and password). Under Accounts the administrator resets a user's password, grants or revokes the administrator role, and deletes an account together with its presentations and files. A password reset ends all sessions of that user. Users cannot change or recover a password themselves. The admin console is in English.

4.4. API and MCP access keys

The API keys tab is used to generate, copy and revoke keys of the form `sk-presenton-...`. A key acts on behalf of the administrator who issued it; revocation takes effect immediately. A key cannot be used to sign in to the web interface or to call administration operations.

4.5. Protecting the channel and access

- Put a TLS reverse proxy in front of the container and restrict direct access to port 5000 to the proxy address.
- The Secure attribute of the session cookie is not set in the shipped configuration; the cookie has the `HttpOnly` and `SameSite=Lax` attributes.
- Files in the data directory are owned by the container's root user and are not encrypted. Restrict access to the directory on the host.
- The image contains third-party libraries whose own network activity the vendor has not audited. In an isolated network, restrict the container's outbound connections to the model server addresses.

5. Verifying the installation

5.1. Container state

```
docker compose ps
docker compose logs --tail 50
```

The `verso` service must be running; after start-up the log contains the line `Migrations run successfully`.

5.2. Service response

```
curl -s http://localhost:5000/api/v1/auth/status
```

The service returns JSON; it contains `"configured": false` before the administrator is created and `"configured": true` afterwards.

5.3. Licence status

In the interface: Settings, Dashboard, Licence block, where Status reads Valid and the tier, expiry and seats are shown. Through the API, with an access key:

```
curl -s -H "Authorization: Bearer sk-presenton-<key>" \
http://localhost:5000/api/v1/license/status
```

The reply must contain `"mode": "onprem", "valid": true` and `"reason": "ok"`.

5.4. Reference test

Create a 5-slide presentation from a short prompt, open it in the editor, and export it to PPTX and PDF. All three actions must complete without error messages.

6. Updating

The program has no automatic update and no update check.

1. Take a backup (see the next section).
2. Obtain the new base image or source from the vendor and, if it has changed, the new package.
3. When delivered as source, rebuild the base image with `docker compose build production` in the `verso` directory.
4. In the package directory, rebuild the overlay and start the container with `docker compose up -d --build`.
5. Check in the container log that the database schema was upgraded: with `MIGRATE_DATABASE_ON_STARTUP=true` the log contains the line `Migrations run successfully`.
6. Repeat the verification steps.

Licence renewal: replace `license/verso-license.json` with the new file and recreate the container:

```
docker compose up -d --force-recreate
```

7. Backup and restore

Back up the `app_data` directory (the `fastapi.db` database, `userConfig.json`, images, exported files, fonts, templates, presentation memory), the `license` directory, and the `docker-compose.yml` and `.env` files. With an external database, back it up with the database's own tools.

The program has no built-in backup. Take the copy with the container stopped:

```
docker compose stop
tar -czf verso-backup-$(date +%F).tgz app_data license docker-compose.yml .env
docker compose start
```

To restore: stop the container, put the directories and files from the archive back in place, start the container.

Note. The data directory holds provider API keys, the session signing secret and password hashes. Protect backups to the same level as the server itself.

8. Removal

```
docker compose down --rmi all
docker image rm verso-production
```

Then delete the package directory together with `app_data` and `license`. Deleting `app_data` permanently destroys all presentations and accounts.

9. Troubleshooting

9.1. Logs

- All application services write to the container's standard output. The log includes, among other things, search queries, image descriptions and paths of processed files.
- The nginx error log is kept in the container, in `/var/log/nginx/error.log`.
- The program keeps no audit log of user or administrator actions. Token usage for every model call and credit movements are recorded in the database and shown in the admin console overview.

To view the logs:

```
docker compose logs -f
docker compose exec verso tail -n 100 /var/log/nginx/error.log
```

9.2. Common failures

Symptom	Cause	Action
Generation does not start, reply 403 with <code>license_invalid</code> and reason missing, malformed, bad-signature or wrong-product	The licence file is absent, damaged or not issued for Verso	Put a valid file at <code>license/verso-license.json</code> , recreate the container
Same, reason <code>expired</code>	The licence has expired	Renew the licence and replace the file
Same, reason <code>machine-mismatch</code>	The licence is bound to another installation	Contact the vendor
Same, reason <code>module-missing</code>	The image lacks the licence verification module	Rebuild the overlay from the package
"Instance not configured" screen	No model provider selected	Configure a provider under Settings

Symptom	Cause	Action
"Could not connect to Ollama"	The model server address is not reachable from the container	Check <code>OLLAMA_URL</code> and network access
Reply 429 at sign-in	Too many failed attempts (5 in 5 minutes) or too many requests	Retry later; the limit clears within 5 minutes
Reply 429 when starting generation	The account's limit of simultaneous generations is reached	Wait for a run to finish or change <code>MAX_CONCURRENT_GENERATIONS</code>
Reply 413 on file upload	The file is larger than 100 MB	Reduce the file
Message that generation was interrupted by a server restart	The container was restarted during generation	Start generation again; finished slides were kept
Container exits with <code>Error running migrations</code>	Database schema upgrade failed	Restore the database from backup, send the log to the vendor

9.3. Recovering administrator access

The primary administrator's password is recovered through environment variables: add `AUTH_USERNAME`, `AUTH_PASSWORD` and `RESET_AUTH=true` to the `environment` section, recreate the container, and remove `RESET_AUTH` after a successful sign-in. The password must be at least 8 characters long. The session signing secret is replaced, so the sessions of all users end, and the access keys issued by that administrator are deleted.

The administrator role can be granted or revoked from the command line:

```
docker compose exec verso sh -c \
  "cd /app/servers/fastapi && python scripts/grant_admin.py --list"
docker compose exec verso sh -c \
  "cd /app/servers/fastapi && python scripts/grant_admin.py <name> [--revoke]"
```

9.4. Contacting the vendor

When contacting the vendor, state the release date, the licence edition and a description of the failure, and attach the output of `docker compose logs` after removing anything that must not be disclosed. The contact address is on the vendor site: <https://new-imobile.com/contact>.

Document history

Document version	Date	Product version	Change
1.0	2026-10-06	2026-09-25 (4bac47a)	First issue