

Authentik Docs

Docs copied from authentik

- [Welcome to authentik](#)
- [Docker Compose installation](#)
- [Configuration](#)

Welcome to authentik

What is authentik?

authentik is an open-source Identity Provider, focused on flexibility and versatility. With authentik, site administrators, application developers, and security engineers a dependable and secure solution for authentication in almost any type of environment. There are robust recovery actions available for the users and applications, including user profile and password management. You can quickly edit, deactivate, or even impersonate a user profile, and set a new password for new users or reset an existing password.

You can use authentik in an existing environment to add support for new protocols, so introducing authentik to your current tech stack doesn't present re-architecting challenges. We already support all of the major providers, such as OAuth2, SAML, LDAP, and SCIM, so you can pick the protocol that you need for each application.

The authentik product provides the following consoles:

- **Admin interface:** a visual tool for the creation and management of users and groups, tokens and credentials, application integrations, events, and the Flows that define standard and customizable login and authentication processes. Easy-to-read visual dashboards display system status, recent logins and authentication events, and application usage.
- **User interface:** this console view in authentik displays all of the applications and integrations in which you have implemented authentik. Click on the app that you want to access to open it, or drill down to edit its configuration in the admin interface
- **Flows:** [Flows](#) are the steps by which the various *Stages* of a login and authentication process occurs. A stage represents a single verification or logic step in the sign-on process. authentik allows for the customization and exact definition of these flows.

Installation

Refer to the installation steps in either [Docker-compose](#) or [Kubernetes](#).

For more information about configuration, Beta versions, and additional installation options, see our main [Installation](#) section.

Screenshots

authentik can use Light or Dark mode for the Admin interface, User interface and the flow interface.

screen_flow_dark.jpg
screen_flow_light.jpg

screen_apps_dark.jpg
screen_apps_light.jpg

screen_admin_dark.jpg
screen_admin_light.jpg

Docker Compose installation

This installation method is for test-setups and small-scale production setups.

Requirements

- A host with at least 2 CPU cores and 2 GB of RAM
- Docker
- Docker Compose

Preparation

To download the latest `docker-compose.yml` open your terminal and navigate to the directory of your choice.

Run the following command:

```
wget https://goauthentik.io/docker-compose.yml
```

If this is a fresh authentik installation, you need to generate a password and a secret key. If you don't already have a password generator installed, you can run this command to install **pwgen**, a popular generator:

```
# You can also use openssl instead: `openssl rand -base64 36`  
sudo apt-get install -y pwgen
```

Next, run the following commands to generate a password and secret key and write them to your `.env` file:

```
echo "PG_PASS=$(pwgen -s 40 1)" >> .env
echo "AUTHENTIK_SECRET_KEY=$(pwgen -s 50 1)" >> .env
# Because of a PostgreSQL limitation, only passwords up to 99 chars are
supported
# See https://www.postgresql.org/message-id/09512C4F-8CB9-4021-B455-
EF4C4F0D55A0@amazon.com
```

To enable error reporting, run the following command:

```
echo "AUTHENTIK_ERROR_REPORTING__ENABLED=true" >> .env
```

Email configuration (optional but recommended)

It is also recommended to configure global email credentials. These are used by authentik to notify you about alerts and configuration issues. They can also be used by [Email stages](#) to send verification/recovery emails.

To configure email credentials, append this block to your `.env` file

```
# SMTP Host Emails are sent to
AUTHENTIK_EMAIL__HOST=localhost
AUTHENTIK_EMAIL__PORT=25
# Optionally authenticate (don't add quotation marks to your password)
AUTHENTIK_EMAIL__USERNAME=
AUTHENTIK_EMAIL__PASSWORD=
# Use StartTLS
AUTHENTIK_EMAIL__USE_TLS=false
# Use SSL
AUTHENTIK_EMAIL__USE_SSL=false
AUTHENTIK_EMAIL__TIMEOUT=10
# Email address authentik will send from, should have a correct @domain
AUTHENTIK_EMAIL__FROM=authentik@localhost
```

Configure for port 80/443

By default, authentik listens internally on port 9000 for HTTP and 9443 for HTTPS. To change the exposed ports to 80 and 443, you can set the following variables in `.env`:

```
COMPOSE_PORT_HTTP=80
COMPOSE_PORT_HTTPS=443
```

See [Configuration](#) to change the internal ports. Be sure to run `docker-compose up -d` to rebuild with the new port numbers.

Startup

DANGER

The server assumes to have local timezone as UTC. All internals are handled in UTC; whenever a time is displayed to the user in UI, the time shown is localized. Do not update or mount `/etc/timezone` or `/etc/localtime` in the authentik containers. This will not give any advantages. It will cause problems with OAuth and SAML authentication, e.g. [see this GitHub issue](#).

Afterwards, run these commands to finish:

```
docker-compose pull
docker-compose up -d
```

The `docker-compose.yml` file statically references the latest version available at the time of downloading the compose file. Each time you upgrade to a newer version of authentik, you download a new `docker-compose.yml` file, which points to the latest available version. For more information, refer to the **Upgrading** section in the [Release Notes](#).

To start the initial setup, navigate to `http://<your server's IP or hostname>:9000/if/flow/initial-setup/`.

There you are prompted to set a password for the akadmin user (the default user).

An explanation about what each service in the docker compose file does, see [Architecture](#).

Configuration

These are all the configuration options you can set via environment variables.

Append any of the following keys to your `.env` file, and run `docker-compose up -d` to apply them.

Info

The double-underscores are intentional, as all these settings are translated to yaml internally, a double-underscore indicates the next level.

All of these variables can be set to values, but you can also use a URI-like format to load values from other places:

- `env://<name>` Loads the value from the environment variable `<name>`. Fallback can be optionally set like `env://<name>?<default>`
- `file://<name>` Loads the value from the file `<name>`. Fallback can be optionally set like `file://<name>?<default>`

Checking settings

To check if your config has been applied correctly, you can run the following command to output the full config:

```
docker-compose run --rm worker dump_config
# Or for kubernetes
kubectl exec -it deployment/authentik-worker -c authentik -- ak dump_config
```

PostgreSQL Settings

- `AUTHENTIK_POSTGRES__HOST`: Hostname of your PostgreSQL Server
- `AUTHENTIK_POSTGRES__NAME`: Database name

- `AUTHENTIK_POSTGRES__USER`: Database user
- `AUTHENTIK_POSTGRES__PORT`: Database port, defaults to 5432
- `AUTHENTIK_POSTGRES__PASSWORD`: Database password, defaults to the environment variable `POSTGRES_PASSWORD`
- `AUTHENTIK_POSTGRES__USE_PGBOUNCER`: Adjust configuration to support connection to PgBouncer
- `AUTHENTIK_POSTGRES__SSLMODE`: Strictness of ssl verification. Defaults to `verify-ca`
- `AUTHENTIK_POSTGRES__SSLROOTCERT`: CA root for server ssl verification
- `AUTHENTIK_POSTGRES__SSLCERT`: Path to x509 client certificate to authenticate to server
- `AUTHENTIK_POSTGRES__SSLKEY`: Path to private key of `SSLCERT` certificate

Redis Settings

- `AUTHENTIK_REDIS__HOST`: Hostname of your Redis Server
- `AUTHENTIK_REDIS__PORT`: Redis port, defaults to 6379
- `AUTHENTIK_REDIS__PASSWORD`: Password for your Redis Server
- `AUTHENTIK_REDIS__TLS`: Use TLS to connect to Redis, defaults to false
- `AUTHENTIK_REDIS__TLS_REQS`: Redis TLS requirements, defaults to "none"
- `AUTHENTIK_REDIS__DB`: Database, defaults to 0
- `AUTHENTIK_REDIS__CACHE_TIMEOUT`: Timeout for cached data until it expires in seconds, defaults to 300
- `AUTHENTIK_REDIS__CACHE_TIMEOUT_FLOWS`: Timeout for cached flow plans until they expire in seconds, defaults to 300
- `AUTHENTIK_REDIS__CACHE_TIMEOUT_POLICIES`: Timeout for cached policies until they expire in seconds, defaults to 300
- `AUTHENTIK_REDIS__CACHE_TIMEOUT_REPUTATION`: Timeout for cached reputation until they expire in seconds, defaults to 300

Listen Setting

- `AUTHENTIK_LISTEN__HTTP`: Listening address:port (e.g. `0.0.0.0:9000`) for HTTP (Server and Proxy outpost)
- `AUTHENTIK_LISTEN__HTTPS`: Listening address:port (e.g. `0.0.0.0:9443`) for HTTPS (Server and Proxy outpost)
- `AUTHENTIK_LISTEN__LDAP`: Listening address:port (e.g. `0.0.0.0:3389`) for LDAP (LDAP outpost)
- `AUTHENTIK_LISTEN__LDAPS`: Listening address:port (e.g. `0.0.0.0:6636`) for LDAPS (LDAP outpost)

- `AUTHENTIK_LISTEN__METRICS`: Listening address:port (e.g. `0.0.0.0:9300`) for Prometheus metrics (All)
- `AUTHENTIK_LISTEN__DEBUG`: Listening address:port (e.g. `0.0.0.0:9900`) for Go Debugging metrics (All)
- `AUTHENTIK_LISTEN__TRUSTED_PROXY_CIDRS`: List of CIDRs that proxy headers should be accepted from (Server)

Defaults to `127.0.0.0/8`, `10.0.0.0/8`, `172.16.0.0/12`, `192.168.0.0/16`, `fe80::/10`, `:::1/128`.

Requests directly coming from one an address within a CIDR specified here are able to set proxy headers, such as `X-Forwarded-For`. Requests coming from other addresses will not be able to set these headers.

authentik Settings

AUTHENTIK_SECRET_KEY

Secret key used for cookie signing and unique user IDs, don't change this after the first install.

AUTHENTIK_LOG_LEVEL

Log level for the server and worker containers. Possible values: debug, info, warning, error

Starting with 2021.12.3, you can also set the log level to `trace`. This has no affect on the core authentik server, but shows additional messages for the embedded outpost.

DANGER

Setting the log level to `trace` will include sensitive details in logs, so it shouldn't be used in most cases.

Logs generated with `trace` should be treated with care as they can give others access to your instance, and can potentially include things like session cookies to authentik **and other pages**.

Defaults to `info`.

AUTHENTIK_COOKIE_DOMAIN

Which domain the session cookie should be set to. By default, the cookie is set to the domain authentik is accessed under.

AUTHENTIK_GEOIP

Path to the GeoIP database. Defaults to `/geoup/GeoLite2-City.mmdb`. If the file is not found, authentik will skip GeoIP support.

AUTHENTIK_DISABLE_UPDATE_CHECK

Disable the inbuilt update-checker. Defaults to `false`.

AUTHENTIK_ERROR_REPORTING

- `AUTHENTIK_ERROR_REPORTING_ENABLED`

Enable error reporting. Defaults to `false`.

Error reports are sent to <https://sentry.io>, and are used for debugging and general feedback.

Anonymous performance data is also sent.

- `AUTHENTIK_ERROR_REPORTING_SENTRY_DSN`

Sets the DSN for the Sentry API endpoint.

When error reporting is enabled, the default Sentry DSN will allow the authentik developers to receive error reports and anonymous performance data, which is used for general feedback about authentik, and in some cases, may be used for debugging purposes.

Users can create their own hosted Sentry account (or self-host Sentry) and opt to collect this data themselves.

- `AUTHENTIK_ERROR_REPORTING_ENVIRONMENT`

The environment tag associated with all data sent to Sentry. Defaults to `customer`.

When error reporting has been enabled to aid in debugging issues, this should be set to a unique value, such as an e-mail address.

- `AUTHENTIK_ERROR_REPORTING_SEND_PII`

Whether or not to send personal data, like usernames. Defaults to `false`.

AUTHENTIK_EMAIL

- `AUTHENTIK_EMAIL_HOST`

Default: `localhost`

- `AUTHENTIK_EMAIL__PORT`

Default: `25`

- `AUTHENTIK_EMAIL__USERNAME`

Default: `` (Don't add quotation marks)

- `AUTHENTIK_EMAIL__PASSWORD`

Default: `` (Don't add quotation marks)

- `AUTHENTIK_EMAIL__USE_TLS`

Default: `false`

- `AUTHENTIK_EMAIL__USE_SSL`

Default: `false`

- `AUTHENTIK_EMAIL__TIMEOUT`

Default: `10`

- `AUTHENTIK_EMAIL__FROM`

Default: `authentik@localhost`

Email address authentik will send from, should have a correct @domain

To change the sender's display name, use a format like `Name <account@domain>`.

AUTHENTIK_OUTPOSTS

- `AUTHENTIK_OUTPOSTS__CONTAINER_IMAGE_BASE`

Placeholders:

- `%(type)s`: Outpost type; proxy, ldap, etc
- `%(version)s`: Current version; 2021.4.1
- `%(build_hash)s`: Build hash if you're running a beta version

Placeholder for outpost docker images. Default: `ghcr.io/goauthentik/%(type)s:%(version)s`.

- `AUTHENTIK_OUTPOSTS__DISCOVER`

Configure the automatic discovery of integrations. Defaults to `true`.

By default, the following is discovered:

- Kubernetes in-cluster config
- Kubeconfig
- Existence of a docker socket

AUTHENTIK_AVATARS

Configure how authentik should show avatars for users. Following values can be set:

Default: `gravatar,initials`

- `none`: Disables per-user avatars and just shows a 1x1 pixel transparent picture
- `gravatar`: Uses gravatar with the user's email address
- `initials`: Generated avatars based on the user's name
- Any URL: If you want to use images hosted on another server, you can set any URL.

Additionally, these placeholders can be used:

- `%(username)s`: The user's username
- `%(mail_hash)s`: The email address, md5 hashed
- `%(upn)s`: The user's UPN, if set (otherwise an empty string)

Starting with authentik 2022.8, you can also use an attribute path like `attributes.something.avatar`, which can be used in combination with the file field to allow users to upload custom avatars for themselves.

Starting with authentik 2023.2, multiple modes can be set, and authentik will fallback to the next mode when no avatar could be found. For example, setting this to `gravatar,initials` will attempt to get an avatar from Gravatar, and if the user has not configured on there, it will fallback to a generated avatar.

`AUTHENTIK_DEFAULT_USER_CHANGE_NAME` [?](#)

INFO

Requires authentik 2021.12.5

Enable the ability for users to change their name, defaults to `true`.

`AUTHENTIK_DEFAULT_USER_CHANGE_EMAIL`

INFO

Requires authentik 2021.12.1

Enable the ability for users to change their Email address, defaults to `false`.

`AUTHENTIK_DEFAULT_USER_CHANGE_USERNAME`

Info

Requires authentik 2021.12.1

Enable the ability for users to change their Usernames, defaults to `false`.

AUTHENTIK_GDPR_COMPLIANCE

Info

Requires authentik 2021.12.1

When enabled, all the events caused by a user will be deleted upon the user's deletion. Defaults to `true`.

AUTHENTIK_DEFAULT_TOKEN_LENGTH

Info

Requires authentik 2022.4.1

Configure the length of generated tokens. Defaults to 60.

AUTHENTIK_IMPERSONATION

Info

Requires authentik 2022.4.2

Globally enable/disable impersonation. Defaults to `true`.

AUTHENTIK_FOOTER_LINKS

Info

Requires authentik 2021.12.1

This option configures the footer links on the flow executor pages.

The setting can be used as follows:

```
AUTHENTIK_FOOTER_LINKS=' [{"name": "Link  
Name", "href": "https://goauthentik.io"} ]'
```

AUTHENTIK_LDAP_TASK_TIMEOUT_HOURS

INFO

Requires authentik 2023.1

Timeout in hours for LDAP synchronization tasks.

Defaults to `2`.

AUTHENTIK_LDAP_PAGE_SIZE

INFO

Requires authentik 2023.6.1

Page size for LDAP synchronization. Controls the number of objects created in a single task.

Defaults to `50`.

AUTHENTIK_LDAP_TLS_CIPHERS

INFO

Requires authentik 2022.7

Allows configuration of TLS Ciphers for LDAP connections used by LDAP sources. Setting applies to all sources.

Defaults to `null`.

[AUTHENTIK_WEB_WORKERS ?](#)

INFO

Requires authentik 2022.9

Configure how many gunicorn worker processes should be started (see <https://docs.gunicorn.org/en/stable/design.html>).

If running in Kubernetes, the default value is set to 2 and should in most cases not be changed, as scaling can be done with multiple pods running the web server. Otherwise, authentik will use 1 worker for each 4 CPU cores + 1 as a value below 2 workers is not recommended.

AUTHENTIK_WEB_THREADS

INFO

Requires authentik 2022.9

Configure how many gunicorn threads a worker processes should have (see <https://docs.gunicorn.org/en/stable/design.html>).

Defaults to 4.

Custom python settings

To modify additional settings further than the options above allow, you can create a custom python file and mount it to `/data/user_settings.py`. This file will be loaded on startup by both the server and the worker. All default settings are [here](#)

CAUTION

Using these custom settings is not supported and can prevent your authentik instance from starting. Use with caution.