


Configuration Reference

This page contains the list of all the available Airflow configurations that you can set in `airflow.cfg` file or using environment variables.

 Note

For more information on setting the configuration, see [Setting Configuration Options](#)

Sections:

- [\[core\]](#)
- [\[logging\]](#)
- [\[metrics\]](#)
- [\[secrets\]](#)
- [\[cli\]](#)
- [\[debug\]](#)
- [\[api\]](#)
- [\[lineage\]](#)
- [\[atlas\]](#)
- [\[operators\]](#)
- [\[hive\]](#)
- [\[webserver\]](#)
- [\[email\]](#)
- [\[smtp\]](#)
- [\[sentry\]](#)
- [\[celery_kubernetes_executor\]](#)
- [\[celery\]](#)
- [\[celery_broker_transport_options\]](#)
- [\[dask\]](#)
- [\[scheduler\]](#)
- [\[kerberos\]](#)
- [\[github_enterprise\]](#)
- [\[admin\]](#)
- [\[elasticsearch\]](#)
- [\[elasticsearch_configs\]](#)
- [\[kubernetes\]](#)
- [\[smart_sensor\]](#)

[core]

dags_folder

The folder where your airflow pipelines live, most likely a subfolder in a code repository. This path must be absolute.

| Type |
|---|
| string |
| Default |
| <code>{AIRFLOW_HOME}/dags</code> |
| Environment Variable |
| <code>AIRFLOW__CORE__DAGS_FOLDER</code> |

hostname_callable

Hostname by providing a path to a callable, which will resolve the hostname. The format is "package.function".

For example, default value “socket.getfqdn” means that result from getfqdn() of “socket” package will be used as hostname.

No argument should be required in the function specified. If using IP address as hostname is preferred, use value `airflow.utils.net.get_host_ip_address`

| Type |
|---|
| string |
| Default |
| <code>socket.getfqdn</code> |
| Environment Variable |
| <code>AIRFLOW__CORE__HOSTNAME_CALLABLE</code> |

default_timezone

Default timezone in case supplied date times are naive can be utc (default), system, or any IANA timezone string (e.g. Europe/Amsterdam)

| Type |
|--|
| string |
| Default |
| <code>utc</code> |
| Environment Variable |
| <code>AIRFLOW__CORE__DEFAULT_TIMEZONE</code> |

executor

The executor class that airflow should use. Choices include `SequentialExecutor` , `LocalExecutor` , `CeleryExecutor` , `DaskExecutor` , `KubernetesExecutor` , `CeleryKubernetesExecutor` or the full import path to the class when using a custom executor.

| Type |
|--------------------------------------|
| string |
| Default |
| <code>SequentialExecutor</code> |
| Environment Variable |
| <code>AIRFLOW__CORE__EXECUTOR</code> |

sql_alchemy_conn

The SQLAlchemy connection string to the metadata database. SQLAlchemy supports many different database engine, more information their website

| Type |
|---|
| string |
| Default |
| <code>sqlite:/// {AIRFLOW_HOME} /airflow.db</code> |
| Environment Variables |
| <code>AIRFLOW__CORE__SQL_ALCHEMY_CONN</code> |
| <code>AIRFLOW__CORE__SQL_ALCHEMY_CONN_CMD</code> |
| <code>AIRFLOW__CORE__SQL_ALCHEMY_CONN_SECRET</code> |

sql_engine_encoding

New in version 1.10.1.

The encoding for the databases

| Type |
|------------------------------------|
| string |
| Default |
| utf-8 |
| Environment Variable |
| AIRFLOW__CORE__SQL_ENGINE_ENCODING |

sql_engine_collation_for_ids

New in version 2.0.0.

Collation for dag_id , task_id , key columns in case they have different encoding. This is particularly useful in case of mysql with utf8mb4 encoding because primary keys for XCom table has too big size and sql_engine_collation_for_ids should be set to utf8mb3_general_ci .

| Type |
|---|
| string |
| Default |
| None |
| Environment Variable |
| AIRFLOW__CORE__SQL_ENGINE_COLLATION_FOR_IDS |

sql_alchemy_pool_enabled

If SqlAlchemy should pool database connections.

| Type |
|---|
| string |
| Default |
| True |
| Environment Variable |
| AIRFLOW__CORE__SQL_ALCHEMY_POOL_ENABLED |

sql_alchemy_pool_size

The SqlAlchemy pool size is the maximum number of database connections in the pool. 0 indicates no limit.

| Type |
|--------------------------------------|
| string |
| Default |
| 5 |
| Environment Variable |
| AIRFLOW__CORE__SQL_ALCHEMY_POOL_SIZE |

sql_alchemy_max_overflow

New in version 1.10.4.

The maximum overflow size of the pool. When the number of checked-out connections reaches the size set in pool_size, additional connections will be returned up to this limit. When those additional connections are returned to the pool, they are disconnected and discarded. It follows then that the total number of simultaneous connections the pool will allow is pool_size + max_overflow, and the total number of “sleeping” connections the pool will allow is pool_size. max_overflow can be set to **-1** to indicate no overflow limit; no limit will be placed on the total number of concurrent connections. Defaults to **10**.

| Type |
|--|
| string |
| Default |
| 10 |
| Environment Variable |
| AIRFLOW__CORE__SQL_ALCHEMY_MAX_OVERFLOW |

sql_alchemy_pool_recycle

The SQLAlchemy pool recycle is the number of seconds a connection can be idle in the pool before it is invalidated. This config does not apply to sqlite. If the number of DB connections is ever exceeded, a lower config value will allow the system to recover faster.

| Type |
|--|
| string |
| Default |
| 1800 |
| Environment Variable |
| AIRFLOW__CORE__SQL_ALCHEMY_POOL_RECYCLE |

sql_alchemy_pool_pre_ping

New in version 1.10.6.

Check connection at the start of each connection pool checkout. Typically, this is a simple statement like “SELECT 1”. More information here: <https://docs.sqlalchemy.org/en/13/core/pooling.html#disconnect-handling-pessimistic>

| Type |
|---|
| string |
| Default |
| True |
| Environment Variable |
| AIRFLOW__CORE__SQL_ALCHEMY_POOL_PRE_PING |

sql_alchemy_schema

New in version 1.10.3.

The schema to use for the metadata database. SQLAlchemy supports databases with the concept of multiple schemas.

| Type |
|--|
| string |
| Default |
| '' |
| Environment Variable |
| AIRFLOW__CORE__SQL_ALCHEMY_SCHEMA |

sql_alchemy_connect_args

New in version 1.10.11.

Import path for connect args in SqlAlchemy. Defaults to an empty dict. This is useful when you want to configure db engine args that SqlAlchemy won't parse in connection string. See https://docs.sqlalchemy.org/en/13/core/engines.html#sqlalchemy.create_engine.params.connect_args

| Type |
|---|
| string |
| Default |
| None |
| Environment Variable |
| AIRFLOW__CORE__SQL_ALCHEMY_CONNECT_ARGS |

parallelism

The amount of parallelism as a setting to the executor. This defines the max number of task instances that should run simultaneously on this airflow installation

| Type |
|----------------------------|
| string |
| Default |
| 32 |
| Environment Variable |
| AIRFLOW__CORE__PARALLELISM |

dag_concurrency

The number of task instances allowed to run concurrently by the scheduler in one DAG. Can be overridden by `concurrency` on DAG level.

| Type |
|--------------------------------|
| string |
| Default |
| 16 |
| Environment Variable |
| AIRFLOW__CORE__DAG_CONCURRENCY |

dags_are_paused_at_creation

Are DAGs paused by default at creation

| Type |
|--|
| string |
| Default |
| True |
| Environment Variable |
| AIRFLOW__CORE__DAGS_ARE_PAUSED_AT_CREATION |

max_active_runs_per_dag

The maximum number of active DAG runs per DAG

| Type |
|--|
| string |
| Default |
| 16 |
| Environment Variable |
| AIRFLOW__CORE__MAX_ACTIVE_RUNS_PER_DAG |

load_examples

Whether to load the DAG examples that ship with Airflow. It's good to get started, but you probably want to set this to `False` in a production environment

| Type |
|------------------------------|
| string |
| Default |
| True |
| Environment Variable |
| AIRFLOW__CORE__LOAD_EXAMPLES |

load_default_connections

New in version 1.10.10.

Whether to load the default connections that ship with Airflow. It's good to get started, but you probably want to set this to `False` in a production environment

| Type |
|---|
| string |
| Default |
| True |
| Environment Variable |
| AIRFLOW__CORE__LOAD_DEFAULT_CONNECTIONS |

plugins_folder

Path to the folder containing Airflow plugins

| Type |
|-------------------------------|
| string |
| Default |
| {AIRFLOW_HOME}/plugins |
| Environment Variable |
| AIRFLOW__CORE__PLUGINS_FOLDER |

execute_tasks_new_python_interpreter

New in version 2.0.0.

Should tasks be executed via forking of the parent process (“False”, the speedier option) or by spawning a new python process (“True” slow, but means plugin changes picked up by tasks straight away)

When are plugins (re)loaded?

Type

boolean

Default

False

Environment Variable

AIRFLOW__CORE__EXECUTE_TASKS_NEW_PYTHON_INTERPRETER

fernet_key

Secret key to save connection passwords in the db

Type

string

Default

{FERNET_KEY}

Environment Variables

AIRFLOW__CORE__FERNET_KEY

AIRFLOW__CORE__FERNET_KEY_CMD

AIRFLOW__CORE__FERNET_KEY_SECRET

donot_pickle

Whether to disable pickling dags

Type

string

Default

True

Environment Variable

AIRFLOW__CORE__DONOT_PICKLE

dagbag_import_timeout

How long before timing out a python file import

Type

float

Default

30.0

Environment Variable

AIRFLOW__CORE__DAGBAG_IMPORT_TIMEOUT

dagbag_import_error_tracebacks

New in version 2.0.0.

Should a traceback be shown in the UI for dagbag import errors, instead of just the exception message

Type

boolean

Default

True

Environment Variable

AIRFLOW__CORE__DAGBAG_IMPORT_ERROR_TRACEBACKS

dagbag_import_error_traceback_depth

New in version 2.0.0.

If tracebacks are shown, how many entries from the traceback should be shown

Type

integer

Default

2

Environment Variable

AIRFLOW__CORE__DAGBAG_IMPORT_ERROR_TRACEBACK_DEPTH

dag_file_processor_timeout

New in version 1.10.6.

How long before timing out a DagFileProcessor, which processes a dag file

Type

string

Default

50

Environment Variable

AIRFLOW__CORE__DAG_FILE_PROCESSOR_TIMEOUT

task_runner

The class to use for running task instances in a subprocess. Choices include StandardTaskRunner, CgroupTaskRunner or the full import path to the class when using a custom task runner.

Type

string

Default

StandardTaskRunner

Environment Variable

AIRFLOW__CORE__TASK_RUNNER

default_impersonation

If set, tasks without a `run_as_user` argument will be run with this user Can be used to de-elevate a sudo user running Airflow when executing tasks

Type

string

Default

''

Environment Variable

AIRFLOW__CORE__DEFAULT_IMPERSONATION

security

What security module to use (for example kerberos)

Type

string

Default

''

Environment Variable

AIRFLOW__CORE__SECURITY

unit_test_mode

Turn unit test mode on (overwrites many configuration options with test values at runtime)

Type

string

Default


False

Environment Variable

AIRFLOW__CORE__UNIT_TEST_MODE

enable_xcom_pickling

Whether to enable pickling for xcom (note that this is insecure and allows for RCE exploits).

 See also

<https://docs.python.org/3/library/pickle.html#comparison-with-json>

Type

string

Default

False

Environment Variable

AIRFLOW__CORE__ENABLE_XCOM_PICKLING

killed_task_cleanup_time

When a task is killed forcefully, this is the amount of time in seconds that it has to cleanup after it is sent a SIGTERM, before it is SIGKILLED

Type

string

Default

60

Environment Variable

`AIRFLOW__CORE__KILLED_TASK_CLEANUP_TIME`

dag_run_conf_overrides_params

Whether to override params with dag_run.conf. If you pass some key-value pairs through `airflow dags backfill -c` or `airflow dags trigger -c`, the key-value pairs will override the existing ones in params.

Type

string

Default

`True`

Environment Variable

`AIRFLOW__CORE__DAG_RUN_CONF_OVERRIDES_PARAMS`

dag_discovery_safe_mode

New in version 1.10.3.

When discovering DAGs, ignore any files that don't contain the strings `DAG` and `airflow`.

Type

string

Default

`True`

Environment Variable

`AIRFLOW__CORE__DAG_DISCOVERY_SAFE_MODE`

default_task_retries

New in version 1.10.6.

The number of retries each task is going to have by default. Can be overridden at dag or task level.

Type

string

Default

`0`

Environment Variable

`AIRFLOW__CORE__DEFAULT_TASK_RETRIES`

min_serialized_dag_update_interval

New in version 1.10.7.

Updating serialized DAG can not be faster than a minimum interval to reduce database write rate.

Type

string

Default

`30`

Environment Variable

AIRFLOW__CORE__MIN_SERIALIZED_DAG_UPDATE_INTERVAL

min_serialized_dag_fetch_interval

New in version 1.10.12.

Fetching serialized DAG can not be faster than a minimum interval to reduce database read rate. This config controls when your DAGs are updated in the Webserver

Type

string

Default

10

Environment Variable

AIRFLOW__CORE__MIN_SERIALIZED_DAG_FETCH_INTERVAL

store_dag_code

New in version 1.10.10.

Whether to persist DAG files code in DB. If set to True, Webserver reads file contents from DB instead of trying to access files in a DAG folder.

Type

string

Default

None

Environment Variable

AIRFLOW__CORE__STORE_DAG_CODE

Example

False

max_num_rendered_ti_fields_per_task

New in version 2.0.0.

Maximum number of Rendered Task Instance Fields (Template Fields) per task to store in the Database. All the template_fields for each of Task Instance are stored in the Database. Keeping this number small may cause an error when you try to view **Rendered** tab in TaskInstance view for older tasks.

Type

integer

Default

30

Environment Variable

AIRFLOW__CORE__MAX_NUM_RENDERED_TI_FIELDS_PER_TASK

check_slas

New in version 1.10.8.

On each dagrun check against defined SLAs

Type

| string |
|----------------------------------|
| Default |
| True |
| Environment Variable |
| AIRFLOW__CORE__CHECK_SLAS |

xcom_backend

New in version 1.10.12.

Path to custom XCom class that will be used to store and resolve operators results

| Type |
|-------------------------------------|
| string |
| Default |
| airflow.models.xcom.BaseXCom |
| Environment Variable |
| AIRFLOW__CORE__XCOM_BACKEND |
| Example |
| path.to.CustomXCom |

lazy_load_plugins

New in version 2.0.0.

By default Airflow plugins are lazily-loaded (only loaded when required). Set it to **False** , if you want to load plugins whenever ‘airflow’ is invoked via cli or loaded from module.

| Type |
|---|
| boolean |
| Default |
| True |
| Environment Variable |
| AIRFLOW__CORE__LAZY_LOAD_PLUGINS |

lazy_discover_providers

New in version 2.0.0.

By default Airflow providers are lazily-discovered (discovery and imports happen only when required). Set it to False, if you want to discover providers whenever ‘airflow’ is invoked via cli or loaded from module.

| Type |
|---|
| boolean |
| Default |
| True |
| Environment Variable |
| AIRFLOW__CORE__LAZY_DISCOVER_PROVIDERS |

max_db_retries

Number of times the code should be retried in case of DB Operational Errors. Not all transactions will be retried as it can cause undesired state. Currently it is only used in `DagFileProcessor.process_file` to retry `dagbag.sync_to_db`.

| Type |
|-------------------------------|
| integer |
| Default |
| 3 |
| Environment Variable |
| AIRFLOW__CORE__MAX_DB_RETRIES |

worker_precheck (Deprecated)

Deprecated since version 2.0.0: The option has been moved to [celery.worker_precheck](#)

base_log_folder (Deprecated)

Deprecated since version 2.0.0: The option has been moved to [logging.base_log_folder](#)

remote_logging (Deprecated)

Deprecated since version 2.0.0: The option has been moved to [logging.remote_logging](#)

remote_log_conn_id (Deprecated)

Deprecated since version 2.0.0: The option has been moved to [logging.remote_log_conn_id](#)

remote_base_log_folder (Deprecated)

Deprecated since version 2.0.0: The option has been moved to [logging.remote_base_log_folder](#)

encrypt_s3_logs (Deprecated)

Deprecated since version 2.0.0: The option has been moved to [logging.encrypt_s3_logs](#)

logging_level (Deprecated)

Deprecated since version 2.0.0: The option has been moved to [logging.logging_level](#)

fab_logging_level (Deprecated)

Deprecated since version 2.0.0: The option has been moved to [logging.fab_logging_level](#)

logging_config_class (Deprecated)

Deprecated since version 2.0.0: The option has been moved to [logging.logging_config_class](#)

colored_console_log (Deprecated)

Deprecated since version 2.0.0: The option has been moved to [logging.colored_console_log](#)

colored_log_format (Deprecated)

Deprecated since version 2.0.0: The option has been moved to [logging.colored_log_format](#)

colored_formatter_class (Deprecated)

Deprecated since version 2.0.0: The option has been moved to [logging.colored_formatter_class](#)

log_format (Deprecated)

Deprecated since version 2.0.0: The option has been moved to [logging.log_format](#)

simple_log_format (Deprecated)

Deprecated since version 2.0.0: The option has been moved to [logging.simple_log_format](#)

task_log_prefix_template (Deprecated)

Deprecated since version 2.0.0: The option has been moved to [logging.task_log_prefix_template](#)

log_filename_template (Deprecated)

Deprecated since version 2.0.0: The option has been moved to [logging.log_filename_template](#)

log_processor_filename_template (Deprecated)

Deprecated since version 2.0.0: The option has been moved to [logging.log_processor_filename_template](#)

dag_processor_manager_log_location (Deprecated)

Deprecated since version 2.0.0: The option has been moved to [logging.dag_processor_manager_log_location](#)

task_log_reader (Deprecated)

Deprecated since version 2.0.0: The option has been moved to [logging.task_log_reader](#)

[logging]

base_log_folder

The folder where airflow should store its log files This path must be absolute

| Type |
|--|
| string |
| Default |
| <code>{AIRFLOW_HOME}/logs</code> |
| Environment Variable |
| <code>AIRFLOW__LOGGING__BASE_LOG_FOLDER</code> |

remote_logging

Airflow can store logs remotely in AWS S3, Google Cloud Storage or Elastic Search. Set this to True if you want to enable remote logging.

| Type |
|---|
| string |
| Default |
| <code>False</code> |
| Environment Variable |
| <code>AIRFLOW__LOGGING__REMOTE_LOGGING</code> |

remote_log_conn_id

Users must supply an Airflow connection id that provides access to the storage location.

| Type |
|---|
| string |
| Default |
| <code>''</code> |
| Environment Variable |
| <code>AIRFLOW__LOGGING__REMOTE_LOG_CONN_ID</code> |

google_key_path

Path to Google Credential JSON file. If omitted, authorization based on [the Application Default Credentials](#) will be used.

| Type |
|--|
| string |
| Default |
| <code>''</code> |
| Environment Variable |
| <code>AIRFLOW__LOGGING__GOOGLE_KEY_PATH</code> |

remote_base_log_folder

Storage bucket URL for remote logging S3 buckets should start with "s3://" Cloudwatch log groups should start with "cloudwatch://" GCS buckets should start with "gs://" WASB buckets should start with "wasb" just to help Airflow select correct handler Stackdriver logs should start with "stackdriver://"

| Type |
|---|
| string |
| Default |
| <code>''</code> |
| Environment Variable |
| <code>AIRFLOW__LOGGING__REMOTE_BASE_LOG_FOLDER</code> |

encrypt_s3_logs

Use server-side encryption for logs stored in S3

| Type |
|--|
| string |
| Default |
| <code>False</code> |
| Environment Variable |
| <code>AIRFLOW__LOGGING__ENCRYPT_S3_LOGS</code> |

logging_level

Logging level

| Type |
|--------|
| string |

Default

INFO

Environment Variable

AIRFLOW__LOGGING__LOGGING_LEVEL

fab_logging_level

Logging level for Flask-appbuilder UI

Type

string

Default

WARN

Environment Variable

AIRFLOW__LOGGING__FAB_LOGGING_LEVEL

logging_config_class

Logging class Specify the class that will specify the logging configuration This class has to be on the python classpath

Type

string

Default

''

Environment Variable

AIRFLOW__LOGGING__LOGGING_CONFIG_CLASS

Example

my.path.default_local_settings.LOGGING_CONFIG

colored_console_log

New in version 1.10.4.

Flag to enable/disable Colored logs in Console Colour the logs when the controlling terminal is a TTY.

Type

string

Default

True

Environment Variable

AIRFLOW__LOGGING__COLORED_CONSOLE_LOG

colored_log_format

New in version 1.10.4.

Log format for when Colored logs is enabled

Type

string

Default

```
[%(blue)s%(asctime)s%(reset)s] {{%(blue)s%(filename)s:%(reset)s%(lineno)d}} %(log_color)s%(levelname)s%(reset)s - %(log_color)s%(message)s%(reset)s
```

Environment Variable

```
AIRFLOW__LOGGING__COLORED_LOG_FORMAT
```

colored_formatter_class

New in version 1.10.4.

Type

string

Default

```
airflow.utils.log.colored_log.CustomTTYColoredFormatter
```

Environment Variable

```
AIRFLOW__LOGGING__COLORED_FORMATTER_CLASS
```

log_format

Format of Log line

Type

string

Default

```
[%%(asctime)s] {{%(filename)s:%(lineno)d}} %(levelname)s - %(message)s
```

Environment Variable

```
AIRFLOW__LOGGING__LOG_FORMAT
```

simple_log_format

Type

string

Default

```
%(asctime)s %(levelname)s - %(message)s
```

Environment Variable

```
AIRFLOW__LOGGING__SIMPLE_LOG_FORMAT
```

task_log_prefix_template

Specify prefix pattern like mentioned below with stream handler TaskHandlerWithCustomFormatter

Type

string

Default

```
{} 
```

Environment Variable

```
AIRFLOW__LOGGING__TASK_LOG_PREFIX_TEMPLATE
```

Example

```
{{ti.dag_id}}-{{ti.task_id}}-{{execution_date}}-{{try_number}}
```

log_filename_template

Formatting for how airflow generates file names/paths for each task run.

Type

string

Default

```
{{{ ti.dag_id }}}/{{{ ti.task_id }}}/{{{ ts }}}/{{{ try_number }}}.log
```

Environment Variable

```
AIRFLOW__LOGGING__LOG_FILENAME_TEMPLATE
```

log_processor_filename_template

Formatting for how airflow generates file names for log

Type

string

Default

```
{{{ filename }}}.log
```

Environment Variable

```
AIRFLOW__LOGGING__LOG_PROCESSOR_FILENAME_TEMPLATE
```

dag_processor_manager_log_location

New in version 1.10.2.

full path of dag_processor_manager logfile

Type

string

Default

```
{AIRFLOW_HOME}/logs/dag_processor_manager/dag_processor_manager.log
```

Environment Variable

```
AIRFLOW__LOGGING__DAG_PROCESSOR_MANAGER_LOG_LOCATION
```

task_log_reader

Name of handler to read task instance logs. Defaults to use `task` handler.

Type

string

Default

```
task
```

Environment Variable

```
AIRFLOW__LOGGING__TASK_LOG_READER
```

extra_loggers

A comma-separated list of third-party logger names that will be configured to print messages to consoles.

Type

string

Default

''

Environment Variable

AIRFLOW__LOGGING__EXTRA_LOGGERS

Example

connexion,sqlalchemy

[metrics]

StatsD (<https://github.com/etsy/statsd>) integration settings.

statsd_on

Enables sending metrics to StatsD.

Type

string

Default

False

Environment Variable

AIRFLOW__METRICS__STATSD_ON

statsd_host

Type

string

Default

localhost

Environment Variable

AIRFLOW__METRICS__STATSD_HOST

statsd_port

Type

string

Default

8125

Environment Variable

AIRFLOW__METRICS__STATSD_PORT

statsd_prefix

Type

string

Default

airflow

Environment Variable

AIRFLOW__METRICS__STATSD_PREFIX

statsd_allow_list

New in version 1.10.6.

If you want to avoid sending all the available metrics to StatsD, you can configure an allow list of prefixes (comma separated) to send only the metrics that start with the elements of the list (e.g: "scheduler,executor,dagrun")

Type

string

Default

''

Environment Variable

AIRFLOW__METRICS__STATSD_ALLOW_LIST

stat_name_handler

A function that validate the statsd stat name, apply changes to the stat name if necessary and return the transformed stat name.

The function should have the following signature: def func_name(stat_name: str) -> str:

Type

string

Default

''

Environment Variable

AIRFLOW__METRICS__STAT_NAME_HANDLER

statsd_datadog_enabled

To enable datadog integration to send airflow metrics.

Type

string

Default

False

Environment Variable

AIRFLOW__METRICS__STATSD_DATADOG_ENABLED

statsd_datadog_tags

List of datadog tags attached to all metrics(e.g: key1:value1,key2:value2)

Type

string

Default

''

| Environment Variable |
|--|
| <code>AIRFLOW__METRICS__STATSD_DATADOG_TAGS</code> |
| |
| statsd_custom_client_path |
| If you want to utilise your own custom Statsd client set the relevant module path below. Note: The module path must exist on your PYTHONPATH for Airflow to pick it up |
| |
| Type |
| string |
| Default |
| None |
| Environment Variable |
| <code>AIRFLOW__METRICS__STATSD_CUSTOM_CLIENT_PATH</code> |

[secrets]

backend

New in version 1.10.10.

Full class name of secrets backend to enable (will precede env vars and metastore in search path)

| Type |
|---|
| string |
| Default |
| <code>''</code> |
| Environment Variable |
| <code>AIRFLOW__SECRETS__BACKEND</code> |
| Example |
| <code>airflow.providers.amazon.aws.secrets.systems_manager.SystemsManagerParameterStoreBackend</code> |

backend_kwargs

New in version 1.10.10.

The backend_kwargs param is loaded into a dictionary and passed to __init__ of secrets backend class. See documentation for the secrets backend you are using. JSON is expected. Example for AWS Systems Manager ParameterStore: `{"connections_prefix": "/airflow/connections", "profile_name": "default"}}`

| Type |
|---|
| string |
| Default |
| <code>''</code> |
| Environment Variable |
| <code>AIRFLOW__SECRETS__BACKEND_KWARGS</code> |

[cli]

api_client

In what way should the cli access the API. The LocalClient will use the database directly, while the json_client will use the api running on the webserver

| Type |
|--|
| string |
| Default |
| <code>airflow.api.client.local_client</code> |
| Environment Variable |
| <code>AIRFLOW__CLI__API_CLIENT</code> |

endpoint_url

If you set web_server_url_prefix, do NOT forget to append it here, ex: `endpoint_url = http://localhost:8080/myroot` So api will look like: `http://localhost:8080/myroot/api/experimental/...`

| Type |
|---|
| string |
| Default |
| <code>http://localhost:8080</code> |
| Environment Variable |
| <code>AIRFLOW__CLI__ENDPOINT_URL</code> |

[debug]

fail_fast

New in version 1.10.8.

Used only with `DebugExecutor` . If set to `True` DAG will fail with first failed task. Helpful for debugging purposes.

| Type |
|--|
| string |
| Default |
| <code>False</code> |
| Environment Variable |
| <code>AIRFLOW__DEBUG__FAIL_FAST</code> |

[api]

enable_experimental_api

New in version 2.0.0.

Enables the deprecated experimental API. Please note that these APIs do not have access control. The authenticated user has full access.

Warning

This [Experimental REST API](#) is deprecated since version 2.0. Please consider using [the Stable REST API](#). For more information on migration, see [UPDATING.md](#)

| Type |
|--------------------|
| boolean |
| Default |
| <code>False</code> |

| Environment Variable |
|--|
| <code>AIRFLOW__API__ENABLE_EXPERIMENTAL_API</code> |

auth_backend

How to authenticate users of the API. See <https://airflow.apache.org/docs/stable/security.html> for possible values. ("airflow.api.auth.backend.default" allows all requests for historic reasons)

| Type |
|--|
| string |
| Default |
| <code>airflow.api.auth.backend.deny_all</code> |
| Environment Variable |
| <code>AIRFLOW__API__AUTH_BACKEND</code> |

maximum_page_limit

Used to set the maximum page limit for API requests

| Type |
|---|
| integer |
| Default |
| <code>100</code> |
| Environment Variable |
| <code>AIRFLOW__API__MAXIMUM_PAGE_LIMIT</code> |

fallback_page_limit

Used to set the default page limit when limit is zero. A default limit of 100 is set on OpenApi spec. However, this particular default limit only work when limit is set equal to zero(0) from API requests. If no limit is supplied, the OpenApi spec default is used.

| Type |
|--|
| integer |
| Default |
| <code>100</code> |
| Environment Variable |
| <code>AIRFLOW__API__FALLBACK_PAGE_LIMIT</code> |

google_oauth2_audience

The intended audience for JWT token credentials used for authorization. This value must match on the client and server sides. If empty, audience will not be tested.

| Type |
|---|
| string |
| Default |
| <code>''</code> |
| Environment Variable |
| <code>AIRFLOW__API__GOOGLE_OAUTH2_AUDIENCE</code> |
| Example |

project-id-random-value.apps.googleusercontent.com

google_key_path

Path to Google Cloud Service Account key file (JSON). If omitted, authorization based on [the Application Default Credentials](#) will be used.

| Type |
|-------------------------------|
| string |
| Default |
| '' |
| Environment Variable |
| AIRFLOW__API__GOOGLE_KEY_PATH |
| Example |
| /files/service-account-json |

[lineage]

backend

what lineage backend to use

| Type |
|---------------------------|
| string |
| Default |
| '' |
| Environment Variable |
| AIRFLOW__LINEAGE__BACKEND |

[atlas]

sasl_enabled

| Type |
|------------------------------|
| string |
| Default |
| False |
| Environment Variable |
| AIRFLOW__ATLAS__SASL_ENABLED |

host

| Type |
|----------------------|
| string |
| Default |
| '' |
| Environment Variable |
| AIRFLOW__ATLAS__HOST |

port

| Type |
|----------------------|
| string |
| Default |
| 21000 |
| Environment Variable |
| AIRFLOW__ATLAS__PORT |

username

| Type |
|--------------------------|
| string |
| Default |
| |
| Environment Variable |
| AIRFLOW__ATLAS__USERNAME |

password

| Type |
|---------------------------------|
| string |
| Default |
| |
| Environment Variables |
| AIRFLOW__ATLAS__PASSWORD |
| AIRFLOW__ATLAS__PASSWORD_CMD |
| AIRFLOW__ATLAS__PASSWORD_SECRET |

[operators]

default_owner

The default owner assigned to each new operator, unless provided explicitly or passed via `default_args`

| Type |
|-----------------------------------|
| string |
| Default |
| airflow |
| Environment Variable |
| AIRFLOW__OPERATORS__DEFAULT_OWNER |

default_cpus

| Type |
|---------|
| string |
| Default |
| 1 |

Environment Variable

AIRFLOW__OPERATORS__DEFAULT_CPUS

default_ram

Type

string

Default

512

Environment Variable

AIRFLOW__OPERATORS__DEFAULT_RAM

default_disk

Type

string

Default

512

Environment Variable

AIRFLOW__OPERATORS__DEFAULT_DISK

default_gpus

Type

string

Default

0

Environment Variable

AIRFLOW__OPERATORS__DEFAULT_GPUS

allow_illegal_arguments

Is allowed to pass additional/unused arguments (args, kwargs) to the BaseOperator operator. If set to False, an exception will be thrown, otherwise only the console message will be displayed.

Type

string

Default

False

Environment Variable

AIRFLOW__OPERATORS__ALLOW_ILLEGAL_ARGUMENTS

[hive]

default_hive_mapred_queue

Default mapreduce queue for HiveOperator tasks

Type

string

Default

''

Environment Variable

AIRFLOW__HIVE__DEFAULT_HIVE_MAPRED_QUEUE

mapred_job_name_template

Template for mapred_job_name in HiveOperator, supports the following named parameters hostname, dag_id, task_id, execution_date

Type

string

Default

None

Environment Variable

AIRFLOW__HIVE__MAPRED_JOB_NAME_TEMPLATE

[webserver]

base_url

The base url of your website as airflow cannot guess what domain or cname you are using. This is used in automated emails that airflow sends to point links to the right web server

Type

string

Default

http://localhost:8080

Environment Variable

AIRFLOW__WEBSERVER__BASE_URL

default_ui_timezone

New in version 1.10.10.

Default timezone to display all dates in the UI, can be UTC, system, or any IANA timezone string (e.g. Europe/Amsterdam). If left empty the default value of core/default_timezone will be used

Type

string

Default

UTC

Environment Variable

AIRFLOW__WEBSERVER__DEFAULT_UI_TIMEZONE

Example

America/New_York

web_server_host

The ip specified when starting the web server

| Type |
|-------------------------------------|
| string |
| Default |
| 0.0.0.0 |
| Environment Variable |
| AIRFLOW__WEBSERVER__WEB_SERVER_HOST |

web_server_port

The port on which to run the web server

| Type |
|-------------------------------------|
| string |
| Default |
| 8080 |
| Environment Variable |
| AIRFLOW__WEBSERVER__WEB_SERVER_PORT |

web_server_ssl_cert

Paths to the SSL certificate and key for the web server. When both are provided SSL will be enabled. This does not change the web server port.

| Type |
|---|
| string |
| Default |
| '' |
| Environment Variable |
| AIRFLOW__WEBSERVER__WEB_SERVER_SSL_CERT |

web_server_ssl_key

Paths to the SSL certificate and key for the web server. When both are provided SSL will be enabled. This does not change the web server port.

| Type |
|--|
| string |
| Default |
| '' |
| Environment Variable |
| AIRFLOW__WEBSERVER__WEB_SERVER_SSL_KEY |

web_server_master_timeout

Number of seconds the webserver waits before killing gunicorn master that doesn't respond

| Type |
|----------------------|
| string |
| Default |
| 120 |
| Environment Variable |

AIRFLOW__WEBSERVER__WEB_SERVER_MASTER_TIMEOUT

web_server_worker_timeout

Number of seconds the gunicorn webserver waits before timing out on a worker

Type

string

Default

120

Environment Variable

AIRFLOW__WEBSERVER__WEB_SERVER_WORKER_TIMEOUT

worker_refresh_batch_size

Number of workers to refresh at a time. When set to 0, worker refresh is disabled. When nonzero, airflow periodically refreshes webserver workers by bringing up new ones and killing old ones.

Type

string

Default

1

Environment Variable

AIRFLOW__WEBSERVER__WORKER_REFRESH_BATCH_SIZE

worker_refresh_interval

Number of seconds to wait before refreshing a batch of workers.

Type

string

Default

30

Environment Variable

AIRFLOW__WEBSERVER__WORKER_REFRESH_INTERVAL

reload_on_plugin_change

New in version 1.10.11.

If set to True, Airflow will track files in plugins_folder directory. When it detects changes, then reload the gunicorn.

Type

boolean

Default

False

Environment Variable

AIRFLOW__WEBSERVER__RELOAD_ON_PLUGIN_CHANGE

secret_key

Secret key used to run your flask app It should be as random as possible

| Type |
|---------------------------------------|
| string |
| Default |
| {SECRET_KEY} |
| Environment Variables |
| AIRFLOW__WEBSERVER__SECRET_KEY |
| AIRFLOW__WEBSERVER__SECRET_KEY_CMD |
| AIRFLOW__WEBSERVER__SECRET_KEY_SECRET |

workers

Number of workers to run the Gunicorn web server

| Type |
|-----------------------------|
| string |
| Default |
| 4 |
| Environment Variable |
| AIRFLOW__WEBSERVER__WORKERS |

worker_class

The worker class gunicorn should use. Choices include sync (default), eventlet, gevent

| Type |
|----------------------------------|
| string |
| Default |
| sync |
| Environment Variable |
| AIRFLOW__WEBSERVER__WORKER_CLASS |

access_logfile

Log files for the gunicorn webserver. '-' means log to stderr.

| Type |
|------------------------------------|
| string |
| Default |
| - |
| Environment Variable |
| AIRFLOW__WEBSERVER__ACCESS_LOGFILE |

error_logfile

Log files for the gunicorn webserver. '-' means log to stderr.

| Type |
|--------|
| string |

Default

-

Environment Variable

AIRFLOW__WEBSERVER__ERROR_LOGFILE

access_logformat

Access log format for gunicorn webserver. default format is `%(h)s %(l)s %(u)s %(t)s "%(r)s" %(s)s %(b)s "%(f)s" "%(a)s"` documentation - <https://docs.gunicorn.org/en/stable/settings.html#access-log-format>

Type

string

Default

''

Environment Variable

AIRFLOW__WEBSERVER__ACCESS_LOGFORMAT

expose_config

Expose the configuration file in the web server

Type

string

Default

False

Environment Variable

AIRFLOW__WEBSERVER__EXPOSE_CONFIG

expose_hostname

New in version 1.10.8.

Expose hostname in the web server

Type

string

Default

True

Environment Variable

AIRFLOW__WEBSERVER__EXPOSE_HOSTNAME

expose_stacktrace

New in version 1.10.8.

Expose stacktrace in the web server

Type

string

Default

True

Environment Variable

AIRFLOW__WEBSERVER__EXPOSE_STACKTRACE

dag_default_view

Default DAG view. Valid values are: `tree`, `graph`, `duration`, `gantt`, `landing_times`

Type

string

Default

tree

Environment Variable

AIRFLOW__WEBSERVER__DAG_DEFAULT_VIEW

dag_orientation

Default DAG orientation. Valid values are: `LR` (Left->Right), `TB` (Top->Bottom), `RL` (Right->Left), `BT` (Bottom->Top)

Type

string

Default

LR

Environment Variable

AIRFLOW__WEBSERVER__DAG_ORIENTATION

demo_mode

Puts the webserver in demonstration mode; blurs the names of Operators for privacy.

Type

string

Default

False

Environment Variable

AIRFLOW__WEBSERVER__DEMO_MODE

log_fetch_timeout_sec

The amount of time (in secs) webserver will wait for initial handshake while fetching logs from other worker machine

Type

string

Default

5

Environment Variable

AIRFLOW__WEBSERVER__LOG_FETCH_TIMEOUT_SEC

log_fetch_delay_sec

New in version 1.10.8.

Time interval (in secs) to wait before next log fetching.

| Type |
|---|
| integer |
| Default |
| 2 |
| Environment Variable |
| AIRFLOW__WEBSERVER__LOG_FETCH_DELAY_SEC |

log_auto_tailing_offset

New in version 1.10.8.

Distance away from page bottom to enable auto tailing.

| Type |
|---|
| integer |
| Default |
| 30 |
| Environment Variable |
| AIRFLOW__WEBSERVER__LOG_AUTO_TAILING_OFFSET |

log_animation_speed

New in version 1.10.8.

Animation speed for auto tailing log display.

| Type |
|---|
| integer |
| Default |
| 1000 |
| Environment Variable |
| AIRFLOW__WEBSERVER__LOG_ANIMATION_SPEED |

hide_paused_dags_by_default

By default, the webserver shows paused DAGs. Flip this to hide paused DAGs by default

| Type |
|---|
| string |
| Default |
| False |
| Environment Variable |
| AIRFLOW__WEBSERVER__HIDE_PAUSED_DAGS_BY_DEFAULT |

page_size

Consistent page size across all listing views in the UI

| Type |
|-------------------------------|
| string |
| Default |
| 100 |
| Environment Variable |
| AIRFLOW__WEBSERVER__PAGE_SIZE |

navbar_color

Define the color of navigation bar

| Type |
|----------------------------------|
| string |
| Default |
| #fff |
| Environment Variable |
| AIRFLOW__WEBSERVER__NAVBAR_COLOR |

default_dag_run_display_number

Default dagrun to show in UI

| Type |
|--|
| string |
| Default |
| 25 |
| Environment Variable |
| AIRFLOW__WEBSERVER__DEFAULT_DAG_RUN_DISPLAY_NUMBER |

enable_proxy_fix

New in version 1.10.1.

Enable werkzeug [ProxyFix](#) middleware for reverse proxy

| Type |
|--------------------------------------|
| boolean |
| Default |
| False |
| Environment Variable |
| AIRFLOW__WEBSERVER__ENABLE_PROXY_FIX |

proxy_fix_x_for

New in version 1.10.7.

Number of values to trust for [X-Forwarded-For](#) . More info: https://werkzeug.palletsprojects.com/en/0.16.x/middleware/proxy_fix/

| Type |
|---------|
| integer |

Default

1

Environment Variable

AIRFLOW__WEBSERVER__PROXY_FIX_X_FOR

proxy_fix_x_proto

New in version 1.10.7.

Number of values to trust for X-Forwarded-Proto

Type

integer

Default

1

Environment Variable

AIRFLOW__WEBSERVER__PROXY_FIX_X_PROTO

proxy_fix_x_host

New in version 1.10.7.

Number of values to trust for X-Forwarded-Host

Type

integer

Default

1

Environment Variable

AIRFLOW__WEBSERVER__PROXY_FIX_X_HOST

proxy_fix_x_port

New in version 1.10.7.

Number of values to trust for X-Forwarded-Port

Type

integer

Default

1

Environment Variable

AIRFLOW__WEBSERVER__PROXY_FIX_X_PORT

proxy_fix_x_prefix

New in version 1.10.7.

Number of values to trust for X-Forwarded-Prefix

Type

| |
|--|
| integer |
| Default |
| 1 |
| Environment Variable |
| AIRFLOW__WEBSERVER__PROXY_FIX_X_PREFIX |

cookie_secure

New in version 1.10.3.

Set secure flag on session cookie

| |
|-----------------------------------|
| Type |
| string |
| Default |
| False |
| Environment Variable |
| AIRFLOW__WEBSERVER__COOKIE_SECURE |

cookie_samesite

New in version 1.10.3.

Set samesite policy on session cookie

| |
|-------------------------------------|
| Type |
| string |
| Default |
| Lax |
| Environment Variable |
| AIRFLOW__WEBSERVER__COOKIE_SAMESITE |

default_wrap

New in version 1.10.4.

Default setting for wrap toggle on DAG code and TI log views.

| |
|----------------------------------|
| Type |
| boolean |
| Default |
| False |
| Environment Variable |
| AIRFLOW__WEBSERVER__DEFAULT_WRAP |

x_frame_enabled

New in version 1.10.8.

Allow the UI to be rendered in a frame

| |
|--|
| |
|--|

| Type |
|--|
| boolean |
| Default |
| True |
| Environment Variable |
| AIRFLOW__WEBSERVER__X_FRAME_ENABLED |

analytics_tool

Send anonymous user activity to your analytics tool choose from google_analytics, segment, or metarouter

| Type |
|---|
| string |
| Default |
| None |
| Environment Variable |
| AIRFLOW__WEBSERVER__ANALYTICS_TOOL |

analytics_id

New in version 1.10.5.

Unique ID of your account in the analytics tool

| Type |
|---|
| string |
| Default |
| None |
| Environment Variable |
| AIRFLOW__WEBSERVER__ANALYTICS_ID |

show_recent_stats_for_completed_runs

‘Recent Tasks’ stats will show for old DagRuns if set

| Type |
|---|
| boolean |
| Default |
| True |
| Environment Variable |
| AIRFLOW__WEBSERVER__SHOW_RECENT_STATS_FOR_COMPLETED_RUNS |

update_fab_perms

New in version 1.10.7.

Update FAB permissions and sync security manager roles on webserver startup

| Type |
|--------|
| string |
| |

| Default |
|--------------------------------------|
| True |
| Environment Variable |
| AIRFLOW__WEBSERVER__UPDATE_FAB_PERMS |

session_lifetime_minutes

New in version 1.10.13.

The UI cookie lifetime in minutes. User will be logged out from UI after `session_lifetime_minutes` of non-activity

| Type |
|--|
| integer |
| Default |
| 43200 |
| Environment Variable |
| AIRFLOW__WEBSERVER__SESSION_LIFETIME_MINUTES |

[email]

Configuration email backend and whether to send email alerts on retry or failure

email_backend

Email backend to use

| Type |
|-------------------------------------|
| string |
| Default |
| airflow.utils.email.send_email_smtp |
| Environment Variable |
| AIRFLOW__EMAIL__EMAIL_BACKEND |

default_email_on_retry

Whether email alerts should be sent when a task is retried

| Type |
|--|
| boolean |
| Default |
| True |
| Environment Variable |
| AIRFLOW__EMAIL__DEFAULT_EMAIL_ON_RETRY |

default_email_on_failure

Whether email alerts should be sent when a task failed

| Type |
|---------|
| boolean |
| Default |

True

Environment Variable

AIRFLOW__EMAIL__DEFAULT_EMAIL_ON_FAILURE

subject_template

File that will be used as the template for Email subject (which will be rendered using Jinja2). If not set, Airflow uses a base template.

See also

Email Configuration

Type

string

Default

None

Environment Variable

AIRFLOW__EMAIL__SUBJECT_TEMPLATE

Example

/path/to/my_subject_template_file

html_content_template

File that will be used as the template for Email content (which will be rendered using Jinja2). If not set, Airflow uses a base template.

See also

Email Configuration

Type

string

Default

None

Environment Variable

AIRFLOW__EMAIL__HTML_CONTENT_TEMPLATE

Example

/path/to/my_html_content_template_file

[smtp]

If you want airflow to send emails on retries, failure, and you want to use the `airflow.utils.email.send_email_smtp` function, you have to configure an smtp server here

smtp_host

Type

string

Default

localhost

Environment Variable

AIRFLOW__SMTP__SMTP_HOST

smtp_starttls

| Type |
|------------------------------|
| string |
| Default |
| True |
| Environment Variable |
| AIRFLOW__SMTP__SMTP_STARTTLS |

smtp_ssl

| Type |
|-------------------------|
| string |
| Default |
| False |
| Environment Variable |
| AIRFLOW__SMTP__SMTP_SSL |

smtp_user

| Type |
|--------------------------|
| string |
| Default |
| None |
| Environment Variable |
| AIRFLOW__SMTP__SMTP_USER |
| Example |
| airflow |

smtp_password

| Type |
|-------------------------------------|
| string |
| Default |
| None |
| Environment Variables |
| AIRFLOW__SMTP__SMTP_PASSWORD |
| AIRFLOW__SMTP__SMTP_PASSWORD_CMD |
| AIRFLOW__SMTP__SMTP_PASSWORD_SECRET |
| Example |
| airflow |

smtp_port

| Type |
|---------|
| string |
| Default |

| Environment Variable |
|---------------------------------------|
| <code>AIRFLOW__SMTP__SMTP_PORT</code> |

smtp_mail_from

| Type |
|--|
| string |
| Default |
| <code>airflow@example.com</code> |
| Environment Variable |
| <code>AIRFLOW__SMTP__SMTP_MAIL_FROM</code> |

smtp_timeout

| Type |
|--|
| integer |
| Default |
| <code>30</code> |
| Environment Variable |
| <code>AIRFLOW__SMTP__SMTP_TIMEOUT</code> |

smtp_retry_limit

| Type |
|--|
| integer |
| Default |
| <code>5</code> |
| Environment Variable |
| <code>AIRFLOW__SMTP__SMTP_RETRY_LIMIT</code> |

[sentry]

Sentry (<https://docs.sentry.io>) integration. Here you can supply additional configuration options based on the Python platform. See: <https://docs.sentry.io/error-reporting/configuration/?platform=python>. Unsupported options: `integrations`, `in_app_include`, `in_app_exclude`, `ignore_errors`, `before_breadcrumb`, `before_send`, `transport`.

sentry_on

Enable error reporting to Sentry

| Type |
|---|
| string |
| Default |
| <code>false</code> |
| Environment Variable |
| <code>AIRFLOW__SENTRY__SENTRY_ON</code> |

sentry_dsn

New in version 1.10.6.

| Type |
|--|
| string |
| Default |
| <code>''</code> |
| Environment Variable |
| <code>AIRFLOW__SENTRY__SENTRY_DSN</code> |

[celery_kubernetes_executor]

This section only applies if you are using the `CeleryKubernetesExecutor` in `[core]` section above

kubernetes_queue

New in version 2.0.0.

Define when to send a task to `KubernetesExecutor` when using `CeleryKubernetesExecutor` . When the queue of a task is the value of `kubernetes_queue` (default `kubernetes`), the task is executed via `KubernetesExecutor` , otherwise via `CeleryExecutor`

| Type |
|--|
| string |
| Default |
| <code>kubernetes</code> |
| Environment Variable |
| <code>AIRFLOW__CELERY_KUBERNETES_EXECUTOR__KUBERNETES_QUEUE</code> |

[celery]

This section only applies if you are using the `CeleryExecutor` in `[core]` section above

celery_app_name

The app name that will be used by celery

| Type |
|--|
| string |
| Default |
| <code>airflow.executors.celery_executor</code> |
| Environment Variable |
| <code>AIRFLOW__CELERY__CELERY_APP_NAME</code> |

worker_concurrency

The concurrency that will be used when starting workers with the `airflow celery worker` command. This defines the number of task instances that a worker will take, so size up your workers based on the resources on your worker box and the nature of your tasks

| Type |
|-----------------|
| string |
| Default |
| <code>16</code> |

| Environment Variable |
|--|
| AIRFLOW__CELERY__WORKER_CONCURRENCY |

worker_autoscale

The maximum and minimum concurrency that will be used when starting workers with the `airflow celery worker` command (always keep minimum processes, but grow to maximum if necessary). Note the value should be `max_concurrency,min_concurrency` Pick these numbers based on resources on worker box and the nature of the task. If autoscale option is available, `worker_concurrency` will be ignored.
<http://docs.celeryproject.org/en/latest/reference/celery.bin.worker.html#cmdoption-celery-worker-autoscale>

| Type |
|--------|
| string |

| Default |
|-------------|
| None |

| Environment Variable |
|--|
| AIRFLOW__CELERY__WORKER_AUTOSCALE |

| Example |
|--------------|
| 16,12 |

worker_prefetch_multiplier

Used to increase the number of tasks that a worker prefetches which can improve performance. The number of processes multiplied by `worker_prefetch_multiplier` is the number of tasks that are prefetched by a worker. A value greater than 1 can result in tasks being unnecessarily blocked if there are multiple workers and one worker prefetches tasks that sit behind long running tasks while another worker has unutilized processes that are unable to process the already claimed blocked tasks. <https://docs.celeryproject.org/en/stable/userguide/optimizing.html#prefetch-limits>

| Type |
|---------|
| integer |

| Default |
|-------------|
| None |

| Environment Variable |
|--|
| AIRFLOW__CELERY__WORKER_PREFETCH_MULTIPLIER |

| Example |
|----------|
| 1 |

worker_log_server_port

When you start an airflow worker, airflow starts a tiny web server subprocess to serve the workers local log files to the airflow main web server, who then builds pages and sends them to users. This defines the port on which the logs are served. It needs to be unused, and open visible from the main web server to connect into the workers.

| Type |
|--------|
| string |

| Default |
|-------------|
| 8793 |

| Environment Variable |
|--|
| AIRFLOW__CELERY__WORKER_LOG_SERVER_PORT |

worker_umask

Umask that will be used when starting workers with the `airflow celery worker` in daemon mode. This control the file-creation mode mask which determines the initial value of file permission bits for newly created files.

| Type |
|--|
| string |
| Default |
| <code>0o077</code> |
| Environment Variable |
| <code>AIRFLOW__CELERY__WORKER_UMASK</code> |

broker_url

The Celery broker URL. Celery supports RabbitMQ, Redis and experimentally a sqlalchemy database. Refer to the Celery documentation for more information.

| Type |
|---|
| string |
| Default |
| <code>redis://redis:6379/0</code> |
| Environment Variables |
| <code>AIRFLOW__CELERY__BROKER_URL</code> |
| <code>AIRFLOW__CELERY__BROKER_URL_CMD</code> |
| <code>AIRFLOW__CELERY__BROKER_URL_SECRET</code> |

result_backend

The Celery result_backend. When a job finishes, it needs to update the metadata of the job. Therefore it will post a message on a message bus, or insert it into a database (depending of the backend) This status is used by the scheduler to update the state of the task The use of a database is highly recommended <http://docs.celeryproject.org/en/latest/userguide/configuration.html#task-result-backend-settings>

| Type |
|--|
| string |
| Default |
| <code>db+postgresql://postgres:airflow@postgres/airflow</code> |
| Environment Variables |
| <code>AIRFLOW__CELERY__RESULT_BACKEND</code> |
| <code>AIRFLOW__CELERY__RESULT_BACKEND_CMD</code> |
| <code>AIRFLOW__CELERY__RESULT_BACKEND_SECRET</code> |

flower_host

Celery Flower is a sweet UI for Celery. Airflow has a shortcut to start it `airflow celery flower` . This defines the IP that Celery Flower runs on

| Type |
|---|
| string |
| Default |
| <code>0.0.0.0</code> |
| Environment Variable |
| <code>AIRFLOW__CELERY__FLOWER_HOST</code> |

flower_url_prefix

The root URL for Flower

| Type |
|---|
| string |
| Default |
| <code>''</code> |
| Environment Variable |
| <code>AIRFLOW__CELERY__FLOWER_URL_PREFIX</code> |
| Example |
| <code>/flower</code> |

flower_port

This defines the port that Celery Flower runs on

| Type |
|---|
| string |
| Default |
| <code>5555</code> |
| Environment Variable |
| <code>AIRFLOW__CELERY__FLOWER_PORT</code> |

flower_basic_auth

New in version 1.10.2.

Securing Flower with Basic Authentication Accepts user:password pairs separated by a comma

| Type |
|--|
| string |
| Default |
| <code>''</code> |
| Environment Variables |
| <code>AIRFLOW__CELERY__FLOWER_BASIC_AUTH</code> |
| <code>AIRFLOW__CELERY__FLOWER_BASIC_AUTH_CMD</code> |
| <code>AIRFLOW__CELERY__FLOWER_BASIC_AUTH_SECRET</code> |
| Example |
| <code>user1:password1,user2:password2</code> |

default_queue

Default queue that tasks get assigned to and that worker listen on.

| Type |
|---|
| string |
| Default |
| <code>default</code> |
| Environment Variable |
| <code>AIRFLOW__CELERY__DEFAULT_QUEUE</code> |

sync_parallelism

New in version 1.10.3.

How many processes CeleryExecutor uses to sync task state. 0 means to use max(1, number of cores - 1) processes.

| Type |
|-----------------------------------|
| string |
| Default |
| 0 |
| Environment Variable |
| AIRFLOW__CELERY__SYNC_PARALLELISM |

celery_config_options

Import path for celery configuration options

| Type |
|---|
| string |
| Default |
| airflow.config_templates.default_celery.DEFAULT_CELERY_CONFIG |
| Environment Variable |
| AIRFLOW__CELERY__CELERY_CONFIG_OPTIONS |

ssl_active

| Type |
|-----------------------------|
| string |
| Default |
| False |
| Environment Variable |
| AIRFLOW__CELERY__SSL_ACTIVE |

ssl_key

| Type |
|--------------------------|
| string |
| Default |
| |
| Environment Variable |
| AIRFLOW__CELERY__SSL_KEY |

ssl_cert

| Type |
|----------------------|
| string |
| Default |
| |
| Environment Variable |

AIRFLOW__CELERY__SSL_CERT

ssl_cacert

| Type |
|-----------------------------|
| string |
| Default |
| <code>''</code> |
| Environment Variable |
| AIRFLOW__CELERY__SSL_CACERT |

pool

New in version 1.10.4.

Celery Pool implementation. Choices include: `prefork` (default), `eventlet`, `gevent` or `solo`. See: <https://docs.celeryproject.org/en/latest/userguide/workers.html#concurrency> <https://docs.celeryproject.org/en/latest/userguide/concurrency/eventlet.html>

| Type |
|-----------------------|
| string |
| Default |
| <code>prefork</code> |
| Environment Variable |
| AIRFLOW__CELERY__POOL |

operation_timeout

New in version 1.10.8.

The number of seconds to wait before timing out `send_task_to_executor` or `fetch_celery_task_state` operations.

| Type |
|------------------------------------|
| float |
| Default |
| <code>1.0</code> |
| Environment Variable |
| AIRFLOW__CELERY__OPERATION_TIMEOUT |

task_track_started

New in version 2.0.0.

Celery task will report its status as ‘started’ when the task is executed by a worker. This is used in Airflow to keep track of the running tasks and if a Scheduler is restarted or run in HA mode, it can adopt the orphan tasks launched by previous SchedulerJob.

| Type |
|-------------------------------------|
| boolean |
| Default |
| <code>True</code> |
| Environment Variable |
| AIRFLOW__CELERY__TASK_TRACK_STARTED |

task_adoption_timeout

New in version 2.0.0.

Time in seconds after which Adopted tasks are cleared by CeleryExecutor. This is helpful to clear stalled tasks.

| Type |
|--|
| integer |
| Default |
| 600 |
| Environment Variable |
| AIRFLOW__CELERY__TASK_ADOPTION_TIMEOUT |

task_publish_max_retries

New in version 2.0.0.

The Maximum number of retries for publishing task messages to the broker when failing due to `AirflowTaskTimeout` error before giving up and marking Task as failed.

| Type |
|---|
| integer |
| Default |
| 3 |
| Environment Variable |
| AIRFLOW__CELERY__TASK_PUBLISH_MAX_RETRIES |

worker_precheck

New in version 1.10.1.

Worker initialisation check to validate Metadata Database connection

| Type |
|----------------------------------|
| string |
| Default |
| False |
| Environment Variable |
| AIRFLOW__CELERY__WORKER_PRECHECK |

[celery_broker_transport_options]

This section is for specifying options which can be passed to the underlying celery broker transport. See: http://docs.celeryproject.org/en/latest/userguide/configuration.html#std:setting-broker_transport_options

visibility_timeout

The visibility timeout defines the number of seconds to wait for the worker to acknowledge the task before the message is redelivered to another worker. Make sure to increase the visibility timeout to match the time of the longest ETA you're planning to use. `visibility_timeout` is only supported for Redis and SQS celery brokers. See: http://docs.celeryproject.org/en/master/userguide/configuration.html#std:setting-broker_transport_options

| Type |
|--------|
| string |

| Default |
|--|
| None |
| Environment Variable |
| AIRFLOW__CELERY_BROKER_TRANSPORT_OPTIONS__VISIBILITY_TIMEOUT |
| Example |
| 21600 |

[dask]

This section only applies if you are using the DaskExecutor in [core] section above

cluster_address

The IP address and port of the Dask cluster’s scheduler.

| Type |
|--------------------------------|
| string |
| Default |
| 127.0.0.1:8786 |
| Environment Variable |
| AIRFLOW__DASK__CLUSTER_ADDRESS |

tls_ca

TLS/ SSL settings to access a secured Dask scheduler.

| Type |
|-----------------------|
| string |
| Default |
| '' |
| Environment Variable |
| AIRFLOW__DASK__TLS_CA |

tls_cert

| Type |
|-------------------------|
| string |
| Default |
| '' |
| Environment Variable |
| AIRFLOW__DASK__TLS_CERT |

tls_key

| Type |
|----------------------|
| string |
| Default |
| '' |
| Environment Variable |

AIRFLOW__DASK__TLS_KEY

[scheduler]

job_heartbeat_sec

Task instances listen for external kill signal (when you clear tasks from the CLI or the UI), this defines the frequency at which they should listen (in seconds).

| Type |
|---------------------------------------|
| string |
| Default |
| 5 |
| Environment Variable |
| AIRFLOW__SCHEDULER__JOB_HEARTBEAT_SEC |

clean_tis_without_dagrun_interval

New in version 2.0.0.

How often (in seconds) to check and tidy up 'running' TaskInstances that no longer have a matching DagRun

| Type |
|---|
| float |
| Default |
| 15.0 |
| Environment Variable |
| AIRFLOW__SCHEDULER__CLEAN_TIS_WITHOUT_DAGRUN_INTERVAL |

scheduler_heartbeat_sec

The scheduler constantly tries to trigger new tasks (look at the scheduler section in the docs for more information). This defines how often the scheduler should run (in seconds).

| Type |
|---|
| string |
| Default |
| 5 |
| Environment Variable |
| AIRFLOW__SCHEDULER__SCHEDULER_HEARTBEAT_SEC |

num_runs

New in version 1.10.6.

The number of times to try to schedule each DAG file -1 indicates unlimited number

| Type |
|---------|
| string |
| Default |
| -1 |
| |

Environment Variable

AIRFLOW__SCHEDULER__NUM_RUNS

processor_poll_interval

New in version 1.10.6.

The number of seconds to wait between consecutive DAG file processing

Type

string

Default

1

Environment Variable

AIRFLOW__SCHEDULER__PROCESSOR_POLL_INTERVAL

min_file_process_interval

Number of seconds after which a DAG file is parsed. The DAG file is parsed every min_file_process_interval number of seconds. Updates to DAGs are reflected after this interval. Keeping this number low will increase CPU usage.

Type

string

Default

30

Environment Variable

AIRFLOW__SCHEDULER__MIN_FILE_PROCESS_INTERVAL

dag_dir_list_interval

How often (in seconds) to scan the DAGs directory for new files. Default to 5 minutes.

Type

string

Default

300

Environment Variable

AIRFLOW__SCHEDULER__DAG_DIR_LIST_INTERVAL

print_stats_interval

How often should stats be printed to the logs. Setting to 0 will disable printing stats

Type

string

Default

30

Environment Variable

AIRFLOW__SCHEDULER__PRINT_STATS_INTERVAL

pool_metrics_interval

New in version 2.0.0.

How often (in seconds) should pool usage stats be sent to statsd (if statsd_on is enabled)

| Type |
|---|
| float |
| Default |
| 5.0 |
| Environment Variable |
| AIRFLOW__SCHEDULER__POOL_METRICS_INTERVAL |

scheduler_health_check_threshold

New in version 1.10.2.

If the last scheduler heartbeat happened more than scheduler_health_check_threshold ago (in seconds), scheduler is considered unhealthy. This is used by the health check in the “/health” endpoint

| Type |
|--|
| string |
| Default |
| 30 |
| Environment Variable |
| AIRFLOW__SCHEDULER__SCHEDULER_HEALTH_CHECK_THRESHOLD |

orphaned_tasks_check_interval

New in version 2.0.0.

How often (in seconds) should the scheduler check for orphaned tasks and SchedulerJobs

| Type |
|---|
| float |
| Default |
| 300.0 |
| Environment Variable |
| AIRFLOW__SCHEDULER__ORPHANED_TASKS_CHECK_INTERVAL |

child_process_log_directory

| Type |
|---|
| string |
| Default |
| {AIRFLOW_HOME}/logs/scheduler |
| Environment Variable |
| AIRFLOW__SCHEDULER__CHILD_PROCESS_LOG_DIRECTORY |

scheduler_zombie_task_threshold

Local task jobs periodically heartbeat to the DB. If the job has not heartbeat in this many seconds, the scheduler will mark the associated task instance as failed and will re-schedule the task.

| Type |
|---|
| string |
| Default |
| 300 |
| Environment Variable |
| AIRFLOW__SCHEDULER__SCHEDULER_ZOMBIE_TASK_THRESHOLD |

catchup_by_default

Turn off scheduler catchup by setting this to `False`. Default behavior is unchanged and Command Line Backfills still work, but the scheduler will not do scheduler catchup if this is `False`, however it can be set on a per DAG basis in the DAG definition (catchup)

| Type |
|--|
| string |
| Default |
| True |
| Environment Variable |
| AIRFLOW__SCHEDULER__CATCHUP_BY_DEFAULT |

max_tis_per_query

This changes the batch size of queries in the scheduling main loop. If this is too high, SQL query performance may be impacted by one or more of the following: - reversion to full table scan - complexity of query predicate - excessive locking Additionally, you may hit the maximum allowable query length for your db. Set this to 0 for no limit (not advised)

| Type |
|---------------------------------------|
| string |
| Default |
| 512 |
| Environment Variable |
| AIRFLOW__SCHEDULER__MAX_TIS_PER_QUERY |

use_row_level_locking

New in version 2.0.0.

Should the scheduler issue `SELECT ... FOR UPDATE` in relevant queries. If this is set to `False` then you should not run more than a single scheduler at once

| Type |
|---|
| boolean |
| Default |
| True |
| Environment Variable |
| AIRFLOW__SCHEDULER__USE_ROW_LEVEL_LOCKING |

max_dagruns_to_create_per_loop

New in version 2.0.0.

Max number of DAGs to create DagRuns for per scheduler loop

Default: 10

See also

Scheduler Tuneables

Type

string

Default

None

Environment Variable

AIRFLOW__SCHEDULER__MAX_DAGRUNS_TO_CREATE_PER_LOOP

max_dagruns_per_loop_to_schedule

New in version 2.0.0.

How many DagRuns should a scheduler examine (and lock) when scheduling and queuing tasks.

Default: 20

See also

Scheduler Tuneables

Type

string

Default

None

Environment Variable

AIRFLOW__SCHEDULER__MAX_DAGRUNS_PER_LOOP_TO_SCHEDULE

schedule_after_task_execution

New in version 2.0.0.

Should the Task supervisor process perform a “mini scheduler” to attempt to schedule more tasks of the same DAG. Leaving this on will mean tasks in the same DAG execute quicker, but might starve out other dags in some circumstances

Default: True

Type

boolean

Default

None

Environment Variable

AIRFLOW__SCHEDULER__SCHEDULE_AFTER_TASK_EXECUTION

parsing_processes

New in version 1.10.14.

The scheduler can run multiple processes in parallel to parse dags. This defines how many processes will run.

| Type |
|---------------------------------------|
| string |
| Default |
| 2 |
| Environment Variable |
| AIRFLOW__SCHEDULER__PARSING_PROCESSES |

use_job_schedule

New in version 1.10.2.

Turn off scheduler use of cron intervals by setting this to False. DAGs submitted manually in the web UI or with trigger_dag will still run.

| Type |
|--------------------------------------|
| string |
| Default |
| True |
| Environment Variable |
| AIRFLOW__SCHEDULER__USE_JOB_SCHEDULE |

allow_trigger_in_future

New in version 1.10.8.

Allow externally triggered DagRuns for Execution Dates in the future Only has effect if schedule_interval is set to None in DAG

| Type |
|---|
| string |
| Default |
| False |
| Environment Variable |
| AIRFLOW__SCHEDULER__ALLOW_TRIGGER_IN_FUTURE |

statsd_on (Deprecated)

Deprecated since version 2.0.0: The option has been moved to [metrics.statsd_on](#)

statsd_host (Deprecated)

Deprecated since version 2.0.0: The option has been moved to [metrics.statsd_host](#)

statsd_port (Deprecated)

Deprecated since version 2.0.0: The option has been moved to [metrics.statsd_port](#)

statsd_prefix (Deprecated)

Deprecated since version 2.0.0: The option has been moved to [metrics.statsd_prefix](#)

statsd_allow_list (Deprecated)

Deprecated since version 2.0.0: The option has been moved to [metrics.statsd_allow_list](#)

stat_name_handler (Deprecated)

Deprecated since version 2.0.0: The option has been moved to [metrics.stat_name_handler](#)

statsd_datadog_enabled (Deprecated)

Deprecated since version 2.0.0: The option has been moved to [metrics.statsd_datadog_enabled](#)

statsd_datadog_tags (Deprecated)

Deprecated since version 2.0.0: The option has been moved to [metrics.statsd_datadog_tags](#)

statsd_custom_client_path (Deprecated)

Deprecated since version 2.0.0: The option has been moved to [metrics.statsd_custom_client_path](#)

max_threads (Deprecated)

Deprecated since version 1.10.14: The option has been moved to [scheduler.parsing_processes](#)

[kerberos]

ccache

| Type |
|--|
| string |
| Default |
| <code>/tmp/airflow_krb5_ccache</code> |
| Environment Variable |
| <code>AIRFLOW__KERBEROS__CCACHE</code> |

principal

gets augmented with fqdn

| Type |
|---|
| string |
| Default |
| <code>airflow</code> |
| Environment Variable |
| <code>AIRFLOW__KERBEROS__PRINCIPAL</code> |

reinit_frequency

| Type |
|--|
| string |
| Default |
| <code>3600</code> |
| Environment Variable |
| <code>AIRFLOW__KERBEROS__REINIT_FREQUENCY</code> |

kinit_path

| Type |
|--------------------------------------|
| string |
| Default |
| kinit |
| Environment Variable |
| AIRFLOW__KERBEROS__KINIT_PATH |

keytab

| Type |
|----------------------------------|
| string |
| Default |
| airflow.keytab |
| Environment Variable |
| AIRFLOW__KERBEROS__KEYTAB |

[github_enterprise]

api_rev

| Type |
|--|
| string |
| Default |
| v3 |
| Environment Variable |
| AIRFLOW__GITHUB_ENTERPRISE__API_REV |

[admin]

hide_sensitive_variable_fields

UI to hide sensitive variable fields when set to True

| Type |
|---|
| string |
| Default |
| True |
| Environment Variable |
| AIRFLOW__ADMIN__HIDE_SENSITIVE_VARIABLE_FIELDS |

sensitive_variable_fields

A comma-separated list of sensitive keywords to look for in variables names.

| Type |
|-----------|
| string |
| Default |
| '' |
| |

Environment Variable

AIRFLOW__ADMIN__SENSITIVE_VARIABLE_FIELDS

[elasticsearch]

host

New in version 1.10.4.

Elasticsearch host

Type

string

Default

''

Environment Variable

AIRFLOW__ELASTICSEARCH__HOST

log_id_template

New in version 1.10.4.

Format of the log_id, which is used to query for a given tasks logs

Type

string

Default

{{dag_id}}-{{task_id}}-{{execution_date}}-{{try_number}}

Environment Variable

AIRFLOW__ELASTICSEARCH__LOG_ID_TEMPLATE

end_of_log_mark

New in version 1.10.4.

Used to mark the end of a log stream for a task

Type

string

Default

end_of_log

Environment Variable

AIRFLOW__ELASTICSEARCH__END_OF_LOG_MARK

frontend

New in version 1.10.4.

Qualified URL for an elasticsearch frontend (like Kibana) with a template argument for log_id Code will construct log_id using the log_id template from the argument above. NOTE: The code will prefix the <https://> automatically, don't include that here.

Type

string

Default

True

Environment Variable

AIRFLOW__ELASTICSEARCH__FRONTEND

write_stdout

New in version 1.10.4.

Write the task logs to the stdout of the worker, rather than the default files

Type

string

Default

False

Environment Variable

AIRFLOW__ELASTICSEARCH__WRITE_STDOUT

json_format

New in version 1.10.4.

Instead of the default log formatter, write the log lines as JSON

Type

string

Default

False

Environment Variable

AIRFLOW__ELASTICSEARCH__JSON_FORMAT

json_fields

New in version 1.10.4.

Log fields to also attach to the json output, if enabled

Type

string

Default

asctime, filename, lineno, levelname, message

Environment Variable

AIRFLOW__ELASTICSEARCH__JSON_FIELDS

[elasticsearch_configs]

use_ssl

New in version 1.10.5.

| Type |
|---|
| string |
| Default |
| False |
| Environment Variable |
| AIRFLOW__ELASTICSEARCH_CONFIGS__USE_SSL |

verify_certs

New in version 1.10.5.

| Type |
|--|
| string |
| Default |
| True |
| Environment Variable |
| AIRFLOW__ELASTICSEARCH_CONFIGS__VERIFY_CERTS |

[kubernetes]

pod_template_file

New in version 1.10.11.

Path to the YAML pod file. If set, all other kubernetes-related fields are ignored.

| Type |
|--|
| string |
| Default |
| '' |
| Environment Variable |
| AIRFLOW__KUBERNETES__POD_TEMPLATE_FILE |

worker_container_repository

The repository of the Kubernetes Image for the Worker to Run

| Type |
|--|
| string |
| Default |
| '' |
| Environment Variable |
| AIRFLOW__KUBERNETES__WORKER_CONTAINER_REPOSITORY |

worker_container_tag

The tag of the Kubernetes Image for the Worker to Run

| Type |
|--------|
| string |

Default

''

Environment Variable

AIRFLOW__KUBERNETES__WORKER_CONTAINER_TAG

namespace

The Kubernetes namespace where airflow workers should be created. Defaults to `default`

Type

string

Default

`default`

Environment Variable

AIRFLOW__KUBERNETES__NAMESPACE

delete_worker_pods

If True, all worker pods will be deleted upon termination

Type

string

Default

`True`

Environment Variable

AIRFLOW__KUBERNETES__DELETE_WORKER_PODS

delete_worker_pods_on_failure

New in version 1.10.11.

If False (and delete_worker_pods is True), failed worker pods will not be deleted so users can investigate them.

Type

string

Default

`False`

Environment Variable

AIRFLOW__KUBERNETES__DELETE_WORKER_PODS_ON_FAILURE

worker_pods_creation_batch_size

New in version 1.10.3.

Number of Kubernetes Worker Pod creation calls per scheduler loop. Note that the current default of "1" will only launch a single pod per-heartbeat. It is HIGHLY recommended that users increase this number to match the tolerance of their kubernetes cluster for better performance.

Type

string

Default

1

Environment Variable

AIRFLOW__KUBERNETES__WORKER_PODS_CREATION_BATCH_SIZE

multi_namespace_mode

New in version 1.10.12.

Allows users to launch pods in multiple namespaces. Will require creating a cluster-role for the scheduler

Type

boolean

Default

False

Environment Variable

AIRFLOW__KUBERNETES__MULTI_NAMESPACE_MODE

in_cluster

Use the service account kubernetes gives to pods to connect to kubernetes cluster. It's intended for clients that expect to be running inside a pod running on kubernetes. It will raise an exception if called from a process not running in a kubernetes environment.

Type

string

Default

True

Environment Variable

AIRFLOW__KUBERNETES__IN_CLUSTER

cluster_context

New in version 1.10.3.

When running with in_cluster=False change the default cluster_context or config_file options to Kubernetes client. Leave blank these to use default behaviour like `kubectl` has.

Type

string

Default

None

Environment Variable

AIRFLOW__KUBERNETES__CLUSTER_CONTEXT

config_file

New in version 1.10.3.

Path to the kubernetes configfile to be used when `in_cluster` is set to False

Type

string

Default

None

Environment Variable

AIRFLOW__KUBERNETES__CONFIG_FILE

kube_client_request_args

New in version 1.10.4.

Keyword parameters to pass while calling a kubernetes client `core_v1_api` methods from Kubernetes Executor provided as a single line formatted JSON dictionary string. List of supported params are similar for all `core_v1_apis`, hence a single config variable for all apis. See:

https://raw.githubusercontent.com/kubernetes-client/python/41f11a09995efcd0142e25946adc7591431bfb2f/kubernetes/client/api/core_v1_api.py

Type

string

Default

''

Environment Variable

AIRFLOW__KUBERNETES__KUBE_CLIENT_REQUEST_ARGS

delete_option_kwargs

New in version 1.10.12.

Optional keyword arguments to pass to the `delete_namespaced_pod` kubernetes client `core_v1_api` method when using the Kubernetes Executor. This should be an object and can contain any of the options listed in the `v1DeleteOptions` class defined here: https://github.com/kubernetes-client/python/blob/41f11a09995efcd0142e25946adc7591431bfb2f/kubernetes/client/models/v1_delete_options.py#L19

Type

string

Default

''

Environment Variable

AIRFLOW__KUBERNETES__DELETE_OPTION_KWARGS

Example

```
{"grace_period_seconds": 10}
```

enable_tcp_keepalive

Enables TCP keepalive mechanism. This prevents Kubernetes API requests to hang indefinitely when idle connection is time-outed on services like cloud load balancers or firewalls.

Type

boolean

Default

False

Environment Variable

AIRFLOW__KUBERNETES__ENABLE_TCP_KEEPALIVE

tcp_keep_idle

When the `enable_tcp_keepalive` option is enabled, TCP probes a connection that has been idle for `tcp_keep_idle` seconds.

| Type |
|------------------------------------|
| integer |
| Default |
| 120 |
| Environment Variable |
| AIRFLOW__KUBERNETES__TCP_KEEP_IDLE |

tcp_keep_intvl

When the *enable_tcp_keepalive* option is enabled, if Kubernetes API does not respond to a keepalive probe, TCP retransmits the probe after *tcp_keep_intvl* seconds.

| Type |
|-------------------------------------|
| integer |
| Default |
| 30 |
| Environment Variable |
| AIRFLOW__KUBERNETES__TCP_KEEP_INTVL |

tcp_keep_cnt

When the *enable_tcp_keepalive* option is enabled, if Kubernetes API does not respond to a keepalive probe, TCP retransmits the probe *tcp_keep_cnt number* of times before a connection is considered to be broken.

| Type |
|-----------------------------------|
| integer |
| Default |
| 6 |
| Environment Variable |
| AIRFLOW__KUBERNETES__TCP_KEEP_CNT |

[smart_sensor]

use_smart_sensor

New in version 2.0.0.

When *use_smart_sensor* is True, Airflow redirects multiple qualified sensor tasks to smart sensor task.

| Type |
|---|
| boolean |
| Default |
| False |
| Environment Variable |
| AIRFLOW__SMART_SENSOR__USE_SMART_SENSOR |

shard_code_upper_limit

New in version 2.0.0.

`shard_code_upper_limit` is the upper limit of `shard_code` value. The `shard_code` is generated by `hashcode % shard_code_upper_limit`.

| Type |
|---|
| integer |
| Default |
| 10000 |
| Environment Variable |
| AIRFLOW__SMART_SENSOR__SHARD_CODE_UPPER_LIMIT |

shards

The number of running smart sensor processes for each service.

| Type |
|-------------------------------|
| integer |
| Default |
| 5 |
| Environment Variable |
| AIRFLOW__SMART_SENSOR__SHARDS |

sensors_enabled

New in version 2.0.0.

comma separated sensor classes support in smart_sensor.

| Type |
|--|
| string |
| Default |
| NamedHivePartitionSensor |
| Environment Variable |
| AIRFLOW__SMART_SENSOR__SENSORS_ENABLED |

Was this entry helpful?



Want to be a part of Apache Airflow?[Join community](#)

name brands are trademarks of their respective holders, including The Apache Software Foundation.