

# How do you change the directory JobScheduler uses for temporary files

- [JobScheduler Master](#)
  - [Instructions for Unix](#)
  - [Instructions for Windows](#)
- [JobScheduler Agent](#)
  - [Instructions for Unix](#)
  - [Instructions for Windows](#)

The JobScheduler Master and Agent create temporary files. The JobScheduler retrieves from the operation system the information where to store such files.

- For Windows and Unix JobScheduler considers the setting of the environment variable `%TMP%` respectively `$TMP` for the location of temporary files.
- For Windows the directory for temporary files is usually associated with the account:
  - User accounts are located below the `%UserProfile%` directory structure.
  - System accounts are usually located below the `%WinDir%` directory structure.
- For Unix JobScheduler creates a subdirectory from it's account to store temporary files, e.g. in `/tmp/jobscheduler_user/`

## JobScheduler Master

You can set the folder for temporary files by adding the variable `TMP` to the JobScheduler environment, see [How to add environment variables for all jobs](#). Consider that jobs will use the folder for temporary files as well.

### Instructions for Unix

- rename `./user_bin/jobscheduler_environment_variables.sh-example` to `./user_bin/jobscheduler_environment_variables.sh`
- edit `./user_bin/jobscheduler_environment_variables.sh`:

#### Sample for Unix

```
TMP=/path/to/tmp/folder
export TMP
```

- make sure that the JobScheduler account has read, write and execute permissions for `/path/to/tmp/folder`
- restart the JobScheduler

### Instructions for Windows

- rename `.\user_bin\jobscheduler_environment_variables.cmd-example` to `.\user_bin\jobscheduler_environment_variables.cmd`
- edit `.\user_bin\jobscheduler_environment_variables.cmd`:

#### Sample for Windows

```
set SCHEDULER_START_PARAMS=%SCHEDULER_START_PARAMS% -env="TMP=\path\to\tmp\folder"
```

- make sure that the JobScheduler Madster account has read, write and execute permissions for `/path/to/tmp/folder`
- execute the following sequence of commands
  - `.\bin\jobscheduler.cmd stop`
  - `.\bin\jobscheduler.cmd remove`
  - `.\bin\jobscheduler.cmd install`
    - After installation optionally assign an account to the Windows service and modify the start type, e.g. to automated start.
  - `.\bin\jobscheduler.cmd start_service`

## JobScheduler Agent

You can set the folder for temporary files by adding the variable `TMP` to the JobScheduler environment. Consider that jobs will use the folder for temporary files as well.

### Instructions for Unix

- rename `./bin/jobscheduler_agent_instance.sh-example` to `./bin/jobscheduler_agent_<port>.sh` where *port* is the numeric port that the Agent is listening for.
- edit the `JAVA_OPTIONS` in `./bin/jobscheduler_agent_<port>.sh`:

#### Sample for Unix

```
JAVA_OPTIONS="-Djava.io.tmpdir=/path/to/tmp/folder"
```

- make sure that the JobScheduler Agent account has read, write and execute permissions for `/path/to/tmp/folder`
- execute the following sequence of commands for an Agent that is listening for port 4445:
  - `./bin/jobscheduler_agent_4445.sh stop`
  - `./bin/jobscheduler_agent_4445.sh start`

## Instructions for Windows

- rename `.\bin\jobscheduler_agent_instance.cmd-example` to `.\bin\jobscheduler_agent_<port>.cmd` where *port* is the numeric port that the Agent is listening for
- edit `.\bin\jobscheduler_agent_<port>.cmd`:

#### Sample for Windows

```
JAVA_OPTIONS=-Djava.io.tmpdir=\path\to\tmp\folder
```

- make sure that the JobScheduler Agent account has read, write and execute permissions for `\path\to\tmp\folder`
- execute the following sequence of commands for an Agent that is listening for port 4445:
  - `.\bin\jobscheduler_agent_4445.cmd stop`
  - `.\bin\jobscheduler_agent_4445.cmd start_service`