

## [TracAdmin](#)

### Table of Contents

<b>TracAdmin</b>	<b>1</b>
Usage	2
<code>trac-admin &lt;targetdir&gt; initenv</code>	2
Interactive Mode	2
Full Command Reference	3

Trac is distributed with a powerful command-line configuration tool. This tool can be used to configure and customize your Trac installation to better fit your needs.

Some of the configurations can also be performed via the web administration module.

## Usage

For nearly every `trac-admin` command, the first argument must be the path to the [TracEnvironment](#):

```
$ trac-admin /path/to/projenv wiki list
```

The only exception is for the `help` command, but even in this case, if you omit the environment you will only get a very succinct list of commands (`help` and `initenv`), the same list you would get when invoking `trac-admin` alone. `trac-admin --version` will tell you about the Trac version (e.g. 1.2) corresponding to the program.

To get a comprehensive list of the available command and sub-command, specify the path to an environment:

```
$ trac-admin /path/to/projenv help
```

Some commands have more detailed help, which you can access by specifying the command's name as a subcommand for `help`:

```
$ trac-admin /path/to/projenv help <command>
```

When an error occurs in a `trac-admin` command the return value will be 2 and error messages are printed to `stderr`.

**`trac-admin <targetdir> initenv`**

This subcommand is very important as is the one used to create a [TracEnvironment](#) in the specified `<targetdir>`. That directory must not exist prior to the call.

```
initenv [<projectname> <db> [<repostype> <repospath>]]

Create and initialize a new environment

If no arguments are given, then the required parameters are requested
interactively unless the optional argument `--config` is specified.

One or more optional arguments --inherit=PATH can be used to specify the
"[inherit] file" option at environment creation time, so that only the
options not already specified in one of the global configuration files are
written to the conf/trac.ini file of the newly created environment.
Relative paths are resolved relative to the "conf" directory of the new
environment.

The optional argument --config=PATH can be used to specify a configuration
file that is used to populate the environment configuration. The arguments
<projectname>, <db> and any other arguments passed in the invocation are
optional, but if specified will override values in the configuration file.
```

It supports the `--inherit` option, which specifies a global configuration file for sharing settings between several environments. You can also inherit from a shared configuration afterwards, by setting the `[inherit] file` option in the `conf/trac.ini` file in your newly created environment. The advantage of specifying the inherited configuration file at environment creation time is that only the options *not* specified in the global configuration file will be written to the environment's `conf/trac.ini` file. See [TracIni#GlobalConfiguration](#).

## Interactive Mode

`trac-admin` starts in interactive mode when the environment path is the only argument. Commands can then be executed on the selected environment using the prompt, which offers tab-completion for non-Windows environments, and when the Python `readline` module is available. It also offers automatic repetition of the last command issued.

Once you are in interactive mode, you can also get help on specific commands or subsets of commands:

For example, to get an explanation of the `resync` command, run:

```
$ help resync
```

To get help on all the Wiki-related commands, run:

```
$ help wiki
```

## Full Command Reference

You will find below the detailed help for all the commands available by default in `trac-admin`. Note that this may not match the list given by `trac-admin <yourenv> help`, as the commands pertaining to components disabled in that environment won't be available and conversely some plugins activated in the environment may add their own commands.

```
help

    Show documentation

initenv [<projectname> <db> [<repostype> <repospath>]]

    Create and initialize a new environment

    If no arguments are given, then the required parameters are requested
    interactively unless the optional argument `--config` is specified.

    One or more optional arguments --inherit=PATH can be used to specify the
    "[inherit] file" option at environment creation time, so that only the
    options not already specified in one of the global configuration files are
    written to the conf/trac.ini file of the newly created environment.
    Relative paths are resolved relative to the "conf" directory of the new
    environment.

    The optional argument --config=PATH can be used to specify a configuration
    file that is used to populate the environment configuration. The arguments
    <projectname>, <db> and any other arguments passed in the invocation are
    optional, but if specified will override values in the configuration file.

attachment add <realm:id> <path> [author] [description]

    Attach a file to a resource

    The resource is identified by its realm and identifier. The attachment will
    be named according to the base name of the file.

attachment export <realm:id> <name> [destination]

    Export an attachment from a resource to a file or stdout

    The resource is identified by its realm and identifier. If no destination
    is specified, the attachment is output to stdout.

attachment list <realm:id>

    List attachments of a resource

    The resource is identified by its realm and identifier.

attachment remove <realm:id> <name>

    Remove an attachment from a resource
```

The resource is identified by its realm and identifier.

```
changeset added <repos> <rev> [rev] [...]
```

Notify trac about changesets added to a repository

This command should be called from a post-commit hook. It will trigger a cache update and notify components about the addition.

```
changeset modified <repos> <rev> [rev] [...]
```

Notify trac about changesets modified in a repository

This command should be called from a post-revprop hook after revision properties like the commit message, author or date have been changed. It will trigger a cache update for the given revisions and notify components about the change.

```
component add <name> [owner]
```

Add a new component

```
component chown <name> <owner>
```

Change component ownership

```
component list
```

Show available components

```
component remove <name>
```

Remove/uninstall a component

```
component rename <name> <newname>
```

Rename a component

```
config get <section> <option>
```

Get the value of the given option in "trac.ini"

```
config remove <section> <option>
```

Remove the specified option from "trac.ini"

```
config set <section> <option> <value>
```

Set the value for the given option in "trac.ini"

```
deploy <directory>
```

Extract static resources from Trac and all plugins

```
hotcopy <backupdir> [--no-database]
```

Make a hot backup copy of an environment

The database is backed up to the 'db' directory of the destination, unless the --no-database option is specified.

```
milestone add <name> [due]
```

Add milestone

```
milestone completed <name> <completed>
```

Set milestone complete date

The <completed> date must be specified in the "d MMM yyyy HH:mm:ss" or "YYYY-MM-DDThh:mm:ss±hh:mm" (ISO 8601) format. Alternatively, "now" can be used to set the completion date to the current time. To remove the completion date from a milestone, specify an empty string ("").

```
milestone due <name> <due>
```

Set milestone due date

The <due> date must be specified in the "d MMM yyyy HH:mm:ss" or "YYYY-MM-DDThh:mm:ss±hh:mm" (ISO 8601) format. Alternatively, "now" can be used to set the due date to the current time. To remove the due date from a milestone, specify an empty string ("").

```
milestone list
```

Show milestones

```
milestone remove <name>
```

Remove milestone

```
milestone rename <name> <newname>
```

Rename milestone

```
permission add <user> <action> [action] [...]
```

Add a new permission rule

```
permission export [file]
```

Export permission rules to a file or stdout as CSV

```
permission import [file]
```

Import permission rules from a file or stdin as CSV

```
permission list [user]
```

List permission rules

```
permission remove <user> <action> [action] [...]
```

Remove a permission rule

```
priority add <value>
```

Add a priority value option

```
priority change <value> <newvalue>
```

Change a priority value

priority list

Show possible ticket priorities

priority order <value> up|down

Move a priority value up or down in the list

priority remove <value>

Remove a priority value

repository add <repos> <dir> [type]

Add a source repository

repository alias <name> <target>

Create an alias for a repository

repository list

List source repositories

repository remove <repos>

Remove a source repository

repository resync <repos> [rev]

Re-synchronize trac with repositories

When [rev] is specified, only that revision is synchronized. Otherwise, the complete revision history is synchronized. Note that this operation can take a long time to complete. If synchronization gets interrupted, it can be resumed later using the `sync` command.

To synchronize all repositories, specify "\*" as the repository.

repository set <repos> <key> <value>

Set an attribute of a repository

The following keys are supported: alias, description, dir, hidden, name, sync\_per\_request, type, url

repository sync <repos> [rev]

Resume synchronization of repositories

It works like `resync`, except that it doesn't clear the already synchronized changesets, so it's a better way to resume an interrupted `resync`.

See `resync` help for detailed usage.

resolution add <value>

Add a resolution value option

resolution change <value> <newvalue>

Change a resolution value

resolution list

Show possible ticket resolutions

resolution order <value> up|down

Move a resolution value up or down in the list

resolution remove <value>

Remove a resolution value

session add <sid[:0|1]> [name] [email]

Create a session for the given sid

Populates the name and email attributes for the given session. Adding a suffix ':0' to the sid makes the session unauthenticated, and a suffix ':1' makes it authenticated (the default if no suffix is specified).

session delete <sid[:0|1]> [...]

Delete the session of the specified sid

An sid suffix ':0' operates on an unauthenticated session with the given sid, and a suffix ':1' on an authenticated session (the default). Specifying the sid 'anonymous' will delete all anonymous sessions.

session list [sid[:0|1]] [...]

List the name and email for the given sids

Specifying the sid 'anonymous' lists all unauthenticated sessions, and 'authenticated' all authenticated sessions. '\*' lists all sessions, and is the default if no sids are given.

An sid suffix ':0' operates on an unauthenticated session with the given sid, and a suffix ':1' on an authenticated session (the default).

session purge <age>

Purge anonymous sessions older than the given age or date

Age may be specified as a relative time like "90 days ago", or as a date in the "d MMM yyyy HH:mm:ss" or "YYYY-MM-DDThh:mm:ss±hh:mm" (ISO 8601) format.

session set <name|email|default\_handler> <sid[:0|1]> <value>

Set the name or email attribute of the given sid

An sid suffix ':0' operates on an unauthenticated session with the given sid, and a suffix ':1' on an authenticated session (the default).

severity add <value>

Add a severity value option

```
severity change <value> <newvalue>
```

Change a severity value

```
severity list
```

Show possible ticket severities

```
severity order <value> up|down
```

Move a severity value up or down in the list

```
severity remove <value>
```

Remove a severity value

```
ticket remove <number>
```

Remove ticket

```
ticket_type add <value>
```

Add a ticket type

```
ticket_type change <value> <newvalue>
```

Change a ticket type

```
ticket_type list
```

Show possible ticket types

```
ticket_type order <value> up|down
```

Move a ticket type up or down in the list

```
ticket_type remove <value>
```

Remove a ticket type

```
upgrade [--no-backup]
```

Upgrade database to current version

The database is backed up to the directory specified by [trac] backup\_dir (the default is 'db'), unless the --no-backup option is specified. The shorthand alias -b can also be used to specify --no-backup.

```
version add <name> [time]
```

Add version

```
version list
```

Show versions

```
version remove <name>
```

Remove version

```
version rename <name> <newname>
```



## Rename version

```
version time <name> <time>
```

### Set version date

The <time> must be specified in the "d MMM yyyy HH:mm:ss" or "YYYY-MM-DDThh:mm:ss±hh:mm" (ISO 8601) format. Alternatively, "now" can be used to set the version date to the current time. To remove the date from a version, specify an empty string ("").

```
wiki dump <directory> [page] [...]
```

### Export wiki pages to files named by title

Individual wiki page names can be specified after the directory. A name ending with a \* means that all wiki pages starting with that prefix should be dumped. If no name is specified, all wiki pages are dumped.

```
wiki export <page> [file]
```

### Export wiki page to file or stdout

```
wiki import <page> [file]
```

### Import wiki page from file or stdin

```
wiki list
```

### List wiki pages

```
wiki load <path> [...]
```

### Import wiki pages from files

If a given path is a file, it is imported as a page with the name of the file. If a path is a directory, all files in that directory are imported.

```
wiki remove <page>
```

### Remove wiki page

```
wiki rename <page> <new_name>
```

### Rename wiki page

```
wiki replace <path> [...]
```

### Replace the content of wiki pages from files (DANGEROUS!)

This command replaces the content of the last version of one or more wiki pages with new content. The previous content is lost, and no new entry is created in the page history. The metadata of the page (time, author) is not changed either.

If a given path is a file, it is imported as a page with the name of the file. If a path is a directory, all files in that directory are imported.

**WARNING:** This operation results in the loss of the previous content and cannot be undone. It may be advisable to backup the current content using

```
"wiki dump" beforehand.
```

```
wiki upgrade
```

```
Upgrade default wiki pages to current version
```

---

See also: [TracBackup](#), [TracEnvironment](#), [TracIni](#), [TracMigrate](#)