TracAdmin

Error: Macro TracGuideToc(None) failed

```
'NoneType' object has no attribute 'find'
```

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 those operations can also be performed via the *Admin* web interface, an updated version of the WebAdmin plugin now integrated within Trac (since version 0.11).

Usage

For nearly every `trac-admin` command, you'll need to specify the path to the TracEnvironment that you want to administer as the first argument, for example:

```
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'll only get a very succinct list of commands (`help` and `initenv`), the same list you'd get when invoking `trac-admin` alone. Also, `trac-admin --version` will tell you about the Trac version (e.g. 0.12) corresponding to the program.

If you want to get a comprehensive list of the available commands and sub-commands, you need to specify an existing environment:

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

Some commands have a 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>
```

`trac-admin <targetdir> initenv`

This subcommand is very important as it's the one used to create a TracEnvironment in the specified `<targetdir>`. That directory must not exists 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.

    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.
```

It supports an extra `--inherit` option, which can be used to specify a global configuration file which can be used share 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, but the advantage of specifying the inherited configuration file at environment creation time is that only the options *not* already specified in the global configuration file will be written in the created environment's `conf/trac.ini` file. See TracIni#GlobalConfiguration.

Note that in version 0.11 of Trac, `initenv` lost an extra last argument `<templatepath>`, which was used in previous versions to point to the `templates` folder. If you are using the one-liner '`trac-admin /path/to/trac/ initenv <projectname> <db> <repostype> <repospath>`' in the above and getting an error that reads **'`Wrong number of arguments to initenv: 4`'**, then this is because you're using a `trac-admin` script from an **older** version of Trac.

Interactive Mode

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

Once you're 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'll 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 can 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.

    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.

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>

    Make a hot backup copy of an environment

milestone add <name> [due]

    Add milestone

milestone completed <name> <completed>

    Set milestone complete date

    The <completed> date must be specified in the "YYYY-MM-DD" 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 "YYYY-MM-DD" 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 list [user]

    List permission rules

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

    Remove a permission rule

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,
    type, url

repository sync <repos> [rev]

    Resume synchronization of repositories

    Similar to `resync`, but doesn't clear the already synchronized changesets.
    Useful for resuming an interrupted `resync`.

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

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> [name] [email]

    Create a session for the given sid

    Populates the name and email attributes and sets the session as
    authenticated.

session delete <sid> [...]

    Delete the session of the specified sid

    Specifying the sid 'anonymous' will delete all anonymous sessions. Using
    '*' will delete all sessions, regardless.

session list <sid> [...]

    List the name and email for the sids specified

    Specifying the sid 'anonymous' will list all of the unauthenticated
    sessions. '*' may be used to list all sessions regardless of authenticated
    status.

session purge <age>

    Purge all anonymous sessions older than the given age

    Age may be specified as a relative time like "90 days ago", or in YYYYMMDD
    format.

session set <name|email> <sid> <value>

    Set the name or email attribute of the given sid

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

    Upgrade database to current version

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: TracGuide, TracBackup, TracPermissions, TracEnvironment, TracIni, TracMigrate