Trac Permissions

Error: Macro TracGuideToc(None) failed

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

Trac uses a simple, case sensitive, permission system to control what users can and can't access.

Permission privileges are managed using the <u>trac-admin</u> tool or the *General / Permissions* panel in the *Admin* web interface.

In addition to the default permission policy described in this page, it is possible to activate additional permission policies by enabling plugins and listing them in the `[trac] permission_policies` configuration entry in the <u>TracIni</u>. See <u>TracFineGrainedPermissions</u> for more details.

Non-authenticated users accessing the system are assigned the name "anonymous". Assign permissions to the "anonymous" user to set privileges for anonymous/guest users. The parts of Trac that a user does not have the privileges for will not be displayed in the navigation. In addition to these privileges, users can be granted additional individual rights in effect when authenticated and logged into the system. All logged in users belong to the virtual group "authenticated", which inherits permissions from "anonymous".

Available Privileges

To enable all privileges for a user, use the `TRAC_ADMIN` permission. Having `TRAC_ADMIN` is like being `root` on a *NIX system: it will allow you to perform any operation.

Otherwise, individual privileges can be assigned to users for the various different functional areas of Trac (**note that the privilege names are case-sensitive**):

Repository Browser

`BROWSER_VIEW`	View directory listings in the <u>repository browser</u>
`LOG_VIEW`	View revision logs of files and directories in the <u>repository browser</u>
`FILE_VIEW`	View files in the <u>repository browser</u>
`CHANGESET_VIEW`	View <u>repository check-ins</u>

Ticket System

`TICKET_VIEW`	View existing tickets and perform ticket queries
`TICKET_CREATE`	Create new tickets
`TICKET_APPEND`	Add comments or attachments to tickets
`TICKET_CHGPROP`	Modify ticket properties (priority, assignment, keywords, etc.) except description field, cc field add/remove when logged in or set email to pref
`TICKET_MODIFY`	Includes both `TICKET_APPEND` and `TICKET_CHGPROP`, and in addition allows resolving tickets
`TICKET_EDIT_CC`	Full modify cc field
`TICKET_EDIT_DESCRIPTION`	Modify description field
`TICKET_ADMIN`	All `TICKET_*` permissions, plus the deletion of ticket attachments and modification of the description field

Attention: the "view tickets" button appears with the `REPORT_VIEW` permission.

Roadmap

`MILESTONE_VIEW`	View a milestone
`MILESTONE_CREATE`	Create a new milestone
`MILESTONE_MODIFY`	Modify existing milestones
`MILESTONE_DELETE`	Delete milestones
`MILESTONE_ADMIN`	All `MILESTONE_*` permissions
`ROADMAP_VIEW`	View the roadmap page, is not (yet) the same as MILESTONE_VIEW, see #4292
`ROADMAP_ADMIN`	to be removed with #3022, replaced by MILESTONE_ADMIN

Reports

`REPORT_VIEW`	View reports, i.e. the "view tickets" link.
`REPORT_SQL_VIEW`	View the underlying SQL query of a report
`REPORT_CREATE`	Create new reports
`REPORT_MODIFY`	Modify existing reports
`REPORT_DELETE`	Delete reports
`REPORT_ADMIN`	All `REPORT_*` permissions

Wiki System

WIKI_VIEW   View existing <u>wiki</u> pages

WIKI_CREATE   Create new <u>wiki</u> pages

WIKI_MODIFY   Change <u>wiki</u> pages

WIKI_DELETE   Delete <u>wiki</u> pages and attachments

WIKI_ADMIN   All WIKI_* permissions, plus the management of *readonly* pages.

Permissions

PERMISSION_GRANT   add/grant a permission

PERMISSION_REVOKE   remove/revoke a permission

PERMISSION_ADMIN   All PERMISSION_* permissions

Others

TIMELINE_VIEW   View the <u>timeline</u> page

SEARCH_VIEW   View and execute <u>search</u> queries

CONFIG_VIEW   Enables additional pages on *About Trac* that show the current configuration or the list of installed plugins

EMAIL_VIEW   Shows email addresses even if `trac show_email_addresses` configuration option is `false`?

Granting Privileges

You grant privileges to users using <u>trac-admin</u>. The current set of privileges can be listed with the following command:

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

This command will allow the user *bob* to delete reports:

```
$ trac-admin /path/to/projenv permission add bob REPORT_DELETE
```

The permission add command also accepts multiple privilege names:

```
$ trac-admin /path/to/projenv permission add bob REPORT_DELETE WIKI_CREATE
```

Or add all privileges:

```
$ trac-admin /path/to/projenv permission add bob TRAC_ADMIN
```

Permission Groups

There are two built-in groups, "authenticated" and "anonymous".
Any user who has not logged in is automatically in the "anonymous" group.
Any user who has logged in is also in the "authenticated" group.
The "authenticated" group inherits permissions from the "anonymous" group.
eg. if the "anonymous" group has permission WIKI_MODIFY, it's not necessary to add the WIKI_MODIFY permisison to the "authenticated" group as well.

Custom groups may be defined that inherit permissions from the two built-in groups.

Permissions can be grouped together to form roles such as *developer*, *admin*, etc.

```
$ trac-admin /path/to/projenv permission add developer WIKI_ADMIN
$ trac-admin /path/to/projenv permission add developer REPORT_ADMIN
$ trac-admin /path/to/projenv permission add developer TICKET_MODIFY
$ trac-admin /path/to/projenv permission add bob developer
$ trac-admin /path/to/projenv permission add john developer
```

Group membership can be checked by doing a `permission list` with no further arguments; the resulting output will include group memberships. **Use lowercase for group names, as uppercase is reserved for permissions**.

Adding a New Group and Permissions

Permission groups can be created by assigning a user to a group you wish to create, then assign permissions to that group.

The following will add *bob* to the new group called *beta_testers* and then will assign WIKI_ADMIN permissions to that group. (Thus, *bob* will inherit the WIKI_ADMIN permission)

```
$ trac-admin /path/to/projenv permission add bob beta_testers
$ trac-admin /path/to/projenv permission add beta_testers WIKI_ADMIN
```

Removing Permissions

Permissions can be removed using the 'remove' command. For example:

This command will prevent the user *bob* from deleting reports:

```
$ trac-admin /path/to/projenv permission remove bob REPORT_DELETE
```

Just like `permission add`, this command accepts multiple privilege names.

You can also remove all privileges for a specific user:

```
$ trac-admin /path/to/projenv permission remove bob '*'
```

Or one privilege for all users:

```
$ trac-admin /path/to/projenv permission remove '*' REPORT_ADMIN
```

Default Permissions

By default on a new Trac installation, the `anonymous` user will have *view* access to everything in Trac, but will not be able to create or modify anything. On the other hand, the `authenticated` users will have the permissions to *create and modify tickets and wiki pages*.

anonymous
> BROWSER_VIEW CHANGESET_VIEW FILE_VIEW LOG_VIEW MILESTONE_VIEW REPORT_SQL_VIEW REPORT_VIEW ROADMAP_VIEW SEARCH_VIEW TICKET_VIEW TIMELINE_VIEW WIKI_VIEW

authenticated
> TICKET_CREATE TICKET_MODIFY WIKI_CREATE WIKI_MODIFY

See also: TracAdmin, TracGuide and TracFineGrainedPermissions