

**NAME**

salsa – tool to manipulate salsa repositories and group members

**SYNOPSIS**

```
# salsa <command> <args>
salsa whoami
salsa search_project devscripts
salsa search_project qa/qa
salsa search_group js-team
salsa search_group perl-team/modules
salsa search_user yadd
salsa push_repo . --group js-team --kgb --irc devscripts --tagpending
salsa update_repo node-mongodb --group js-team --disable-kgb --desc \
    --desc-pattern "Package %p"
salsa update_repo js-team/node-mongodb --kgb --irc debian-js
salsa update_safe --all --desc --desc-pattern "Debian package %p" \
    --group js-team
salsa checkout node-mongodb --group js-team
salsa checkout js-team/node-mongodb
salsa add_user developer foobar --group-id 2665
salsa update_user maintainer foobar --group js-team
salsa del_user foobar --group js-team
```

**DESCRIPTION**

**salsa** is designed to create and configure repositories on <<https://salsa.debian.org>> and manage users of groups.

A Salsa token is required, except for search\* commands, and must be set in command line (*see below*), or in your configuration file (*~/.devscripts*):

```
SALSA_TOKEN=abcdefghi
```

or

```
SALSA_TOKEN=`cat ~/.token`
```

or

```
SALSA_TOKEN_FILE=~/.dpt.conf
```

If you choose to link another file using SALSA\_TOKEN\_FILE, it must contain a line with one of (no differences):

```
<anything>SALSA_PRIVATE_TOKEN=xxxx
<anything>SALSA_TOKEN=xxxx
```

This allows for example to use **dpt** (1) configuration file (*~/.dpt.conf*) which contains:

```
DPT_SALSA_PRIVATE_TOKEN=abcdefghi
```

**COMMANDS****Managing users and groups****join**

Request access to a group.

```
salsa join js-team
salsa join --group js-team
salsa join --group-id 1234
```

**add\_user**

Add a user to a group.

```
salsa --group js-group add_user guest foouser
salsa --group-id 1234 add_user guest foouser
salsa --group-id 1234 add_user maintainer 1245
```

First argument is the GitLab's access levels: guest, reporter, developer, maintainer, owner.

### **del\_user**

Remove a user from a group

```
salsa --group js-team del_user foouser
salsa --group-id=1234 del_user foouser
```

### **list\_groups**

List sub groups of current one if group is set, groups of current user else.

### **group**

Show group members.

```
salsa --group js-team group
salsa --group-id 1234 group
```

### **search\_group**

Search for a group using given string. Shows group id and other information.

```
salsa search_group perl-team
salsa search_group perl-team/modules
salsa search_group 2666
```

### **search\_user**

Search for a user using given string. Shows user id and other information.

```
salsa search_user yadd
```

### **update\_user**

Update user role in a group.

```
salsa --group-id 1234 update_user guest foouser
salsa --group js-team update_user maintainer 1245
```

First argument is the GitLab's access levels: guest, reporter, developer, maintainer, owner.

### **whoami**

Gives information on the token owner

```
salsa whoami
```

## **Managing repositories**

One of `--group`, `--group-id`, `--user` or `--user-id` is required to manage repositories. If both are set, salsa warns and only `--user`/`--user-id` is used. If none is given, salsa uses current user id (*token owner*).

### **check\_repo**

Verify that repo(s) are well configured. It works exactly like **update\_repo** except that it does not modify anything but just lists projects not well configured with found errors.

```
salsa --user yadd --tagpending --kgb --irc=devscripts check_repo test
salsa --group js-team check_repo --all
salsa --group js-team --rename-head check_repo test1 test2 test3
```

### **checkout or co**

Clone repo in current dir. If directory already exists, update local repo.

```
salsa --user yadd co devscripts
salsa --group js-team co node-mongodb
salsa co js-team/node-mongodb
```

You can clone more than one repository or all repositories of a group or a user:

```
salsa --user yadd co devscripts autodep8
salsa co yadd/devscripts js-team/npm
salsa --group js-team co --all           # All js-team repos
salsa co --all                           # All your repos
```

### **create\_repo**

Create public empty project. If `--group/--group-id` is set, project is created in group directory, else in user directory.

```
salsa --user yadd create_repo test
salsa --group js-team --kgb --irc-channel=devscripts create_repo test
```

### **del\_repo**

Delete a repository.

### **fork**

Forks a project in group/user repository and set “upstream” to original project. Example:

```
$ salsa fork js-team/node-mongodb --verbose
...
salsa.pl info: node-mongodb ready in node-mongodb/
$ cd node-mongodb
$ git remote --verbose show
origin          git@salsa.debian.org:me/node-mongodb (fetch)
origin          git@salsa.debian.org:me/node-mongodb (push)
upstream        git@salsa.debian.org:js-team/node-mongodb (fetch)
upstream        git@salsa.debian.org:js-team/node-mongodb (push)
```

For a group:

```
salsa fork --group js-team user/node-foo
```

### **forks**

List forks of project(s).

```
salsa forks qa/qa debian/devscripts
```

Project can be set using full path or using `--group/--group-id` or `--user/--user-id`, else it is searched in current user namespace.

### **push**

Push relevant packaging refs to origin Git remote. To be run from packaging working directory.

```
salsa push
```

It pushes the following refs to the configured remote for the debian-branch or, falling back, to the “origin” remote:

- “master” branch (or whatever is set to debian-branch in gbp.conf)
- “upstream” branch (or whatever is set to upstream-branch in gbp.conf)
- “pristine-tar” branch
- tags named “debian/\*” (or whatever is set to debian-tag in gbp.conf)
- tags named “upstream/\*” (or whatever is set to upstream-tag in gbp.conf)
- all tags, if the package’s source format is “3.0 (native)”

### **ls or list\_repos**

Shows projects owned by user or group. If second argument exists, search only matching projects

```
salsa --group js-team list_repos
salsa --user yadd list_repos foo*
```

**merge\_request, mr**

Creates a merge request.

Suppose you created a fork using **salsa fork**, modify some things in a new branch using one commit and want to propose it to original project (*branch “master”*). You just have to launch this in source directory:

```
salsa mr
```

Other example:

```
salsa mr --mr-dst-project debian/foo --mr-dst-branch debian/master
```

or simply

```
salsa mr debian/foo debian/master
```

Note that unless destination project has been set using command line, **salsa merge\_request** will search it in the following order:

using GitLab API: salsa will detect from where this project was forked

using “upstream” origin

else salsa will use source project as destination project

To force salsa to use source project as destination project, you can use “same”:

```
salsa mr --mr-dst-project same
# or
salsa mr same
```

New merge request will be created using last commit title and description.

See **--mr-\*** options for more.

**merge\_requests, mrs**

List opened merge requests for project(s)

```
salsa mrs qa/qa debian/devscripts
```

Project can be set using full path or using **--group/--group-id** or **--user/--user-id**, else it is searched in current user namespace.

**protect\_branch**

Protect/unprotect a branch.

Set protection

```
#                                project      branch merge push
salsa --group js-team protect_branch node-mongodb master m      d
```

“merge” and “push” can be one of:

**o, owner:** owner only

**m, maintainer:** o + maintainers allowed

**d, developer:** m + developers allowed

**r, reporter:** d + reporters allowed

**g, guest:** r + guest allowed

Unprotect

```
salsa --group js-team protect_branch node-mongodb master no
```

**protected\_branches**

List protected branches

```
salsa --group js-team protected_branches node-mongodb
```

**push\_repo**

Create a new project from a local Debian source directory configured with git.

**push\_repo** executes the following steps:

gets project name using debian/changelog file;  
 launches **git remote add upstream ...**;  
 launches **create\_repo**;  
 pushes local repo.

Examples:

```
salsa --user yadd push_repo ./test
salsa --group js-team --kgb --irc-channel=devscripts push_repo .
```

**rename\_branch**

Rename branch given in **--source-branch** with name given in **--dest-branch**. You can use **--no-fail** and **--all** options here.

**search, search\_project, search\_repo**

Search for a project using given string. Shows name, owner id and other information.

```
salsa search devscripts
salsa search debian/devscripts
salsa search 18475
```

**update\_repo**

Configure repo(s) using parameters given to command line. A repo name has to be given unless **--all** is set. Prefer to use **update\_safe**.

```
salsa --user yadd --tagpending --kgb --irc=devscripts update_repo test
salsa --group js-team update_repo --all
salsa --group js-team --rename-head update_repo test1 test2 test3
salsa update_repo js-team/node-mongodb --kgb --irc debian-js
```

By default when using **--all**, salsa will fail on first error. If you want to continue, set **--no-fail**. In this case, salsa will display a warning for each project that has fail but continue with next project. Then to see full errors, set **--verbose**.

**update\_safe**

Launch **check\_repo** and ask before launching **update\_repo** (unless **--yes**).

```
salsa --user yadd --tagpending --kgb --irc=devscripts update_safe test
salsa --group js-team update_safe --all
salsa --group js-team --rename-head update_safe test1 test2 test3
salsa update_safe js-team/node-mongodb --kgb --irc debian-js
```

**Other****purge\_cache**

Empty local cache.

**OPTIONS****General options****-C, --chdir**

Change directory before launching command

```
salsa -C ~/debian co debian/libapache2-mod-fcgid
```

**--cache-file**

File to store cached values. Default to **~/cache/salsa.json**. An empty value disables cache.

.devscripts value: **SALSA\_CACHE\_FILE**

**--no-cache**

Disable cache usage. Same as **--cache-file ""**

**--conf-file, --conf-file**

Add or replace default configuration files (/etc/devscripts.conf and ~/.devscripts). This can only be used as the first option given on the command-line.

replace:

```
salsa --conf-file test.conf <command>...
salsa --conf-file test.conf --conf-file test2.conf <command>...
```

add:

```
salsa --conf-file +test.conf <command>...
salsa --conf-file +test.conf --conf-file +test2.conf <command>...
```

If one **--conf-file** has no +, default configuration files are ignored.

**--no-conf, --noconf**

Don't read any configuration files. This can only be used as the first option given on the command-line.

**--debug**

Enable debugging output

**--group**

Team to use. Use `salsa search_group name` to find it.

If you want to use a subgroup, you have to set its full path:

```
salsa --group perl-team/modules/packages check_repo lemonldap-ng
```

.devscripts value: **SALSA\_GROUP**

Be careful when you use **SALSA\_GROUP** in your .devscripts file. Every **salsa** command will be executed in group space, for example if you want to propose a little change in a project using **salsa fork + salsa mr**, this “fork” will be done in group space unless you set a **--user/--user-id**. Prefer to use an alias in your .bashrc file. Example:

```
alias jsteam_admin="salsa --group js-team"
```

or

```
alias jsteam_admin="salsa --conf-file ~/.js.conf"
```

or to use both .devscripts and .js.conf:

```
alias jsteam_admin="salsa --conf-file +~/.js.conf"
```

then you can fix **SALSA\_GROUP** in ~/.js.conf

To enable bash completion for your alias, add this in your .bashrc file:

```
_completion_loader salsa
complete -F _salsa_completion jsteam_admin
```

**--group-id**

Group id to use. Use `salsa search_group name` to find it.

.devscripts value: **SALSA\_GROUP\_ID**

Be careful when you use **SALSA\_GROUP\_ID** in your .devscripts file. Every **salsa** command will be executed in group space, for example if you want to propose a little change in a project using **salsa fork + salsa mr**, this “fork” will be done in group space unless you set a **--user/--user-id**. Prefer to use an alias in your .bashrc file. Example:

```
alias jsteam_admin="salsa --group-id 2666"
```

or

```
alias jsteam_admin="salsa --conf-file ~/.js.conf
```

then you can fix **SALSA\_GROUP\_ID** in `~/.js.conf`

**--help**: displays this manpage

**-i, --info**

Prompt before sensible changes.

.devscripts value: **SALSA\_INFO** (yes/no)

**--path**

Repo path. Default to group or user path.

.devscripts value: **SALSA\_REPO\_PATH**

**--token**

Token value (see above).

**--token-file**

File to find token (see above).

**--user**

Username to use. If neither **--group**, **--group-id**, **--user** or **--user-id** is set, salsa uses current user id (corresponding to salsa private token).

**--user-id**

User id to use. Use `salsa search_user` name to find one. If neither **--group**, **--group-id**, **--user** or **--user-id** is set, salsa uses current user id (corresponding to salsa private token).

.devscripts value: **SALSA\_USER\_ID**

**--verbose**

Enable verbose output.

**--yes**

Never ask for consent.

.devscripts value: **SALSA\_YES** (yes/no)

### Update/create repo options

**--all**

When set, all project of group/user are affected by command.

**--skip**: ignore project with **--all**. Example:

```
salsa update_repo --tagpending --all --skip qa --skip devscripts
```

.devscripts value: **SALSA\_SKIP**. To set multiples values, use spaces. Example

```
SALSA_SKIP=qa devscripts
```

**--skip-file**: ignore projects in this file (1 project per line)

```
salsa update_repo --tagpending --all --skip-file ~/.skip
```

.devscripts value: **SALSA\_SKIP\_FILE**

**--ci-config-path**

Configure configuration file path of GitLab CI. Default: empty. Example:

```
salsa update_safe --ci-config-path debian/.gitlab-ci.yml debian/devscripts
```

.devscripts value: **SALSA\_CI\_CONFIG\_PATH**

**--desc --no-desc**

Configure repo description using pattern given in **desc-pattern**

.devscripts value: **SALSA\_DESC** (yes/no)

**--desc-pattern**

Repo description pattern. Default to “Debian package %p”. “%p” is replaced by repo name, while “%P” is replaced by repo name given in command (may contains full path).

.devscripts value: **SALSA\_DESC\_PATTERN**

**--email, --no-email, --disable-email**

Enable, ignore or disable email-on-push.

.devscripts value: **SALSA\_EMAIL** (yes/ignore/no, default: ignore)

**--email-recipient**

Email-on-push recipient. Can be multi valued:

```
$ salsa update_safe myrepo \
    --email-recipient foo@foobar.org \
    --email-recipient bar@foobar.org
```

If recipient value contains “%p”, it is replaced by project name.

.devscripts value: **SALSA\_EMAIL\_RECIPIENTS** (use spaces to separate multiples recipients)

**--enable-issues, --no-enable-issues, --disable-issues, --no-disable-issues**

Enable, ignore or disable issues.

.devscripts values: **SALSA\_ENABLE\_ISSUES** (yes/ignore/no, default: ignore)

**--enable-mr, --no-enable-mr, --disable-mr, --no-disable-mr**

Enable, ignore or disable merge requests.

.devscripts values: **SALSA\_ENABLE\_MR** (yes/ignore/no, default: ignore)

**--irc-channel**

IRC channel for KGB or Irker. Can be used more than one time only with **--irker**.

**Important:** channel must not include the first “#”. If salsa finds a channel starting with “#”, it will consider that the channel starts with 2 “#”!

.devscript value: **SALSA\_IRC\_CHANNEL**.

Multiple values must be space separated.

Since configuration files are read using **sh**, be careful when using “#”: you must enclose the channel with quotes, else **sh** will consider it as a comment and will ignore this value.

**--irker, --no-irker, --disable-irker**

Enable, ignore or disable Irker service

.devscripts values: **SALSA\_IRKER** (yes/ignore/no, default: ignore)

**--irker-host**

Irker host. Default: ruprecht.snow-crash.org

.devscripts value: **SALSA\_IRKER\_HOST**

**--irker-port**

Irker port. Default: empty (default value)

.devscripts value: **SALSA\_IRKER\_PORT**

**--kgb, --no-kgb, --disable-kgb**

Enable, ignore or disable KGB webhook.

.devscripts value: **SALSA\_KGB** (yes/ignore/no, default: ignore)

**--kgb-options**

List of KGB enabled options (comma separated). Default: issues\_events, merge\_requests\_events, note\_events, pipeline\_events, push\_events, tag\_push\_events, wiki\_page\_events



```
$ salsa update_safe debian/devscripts --kgb --irc-channel devscripts \
--kgb-options 'merge_requests_events,issues_events,enable_ssl_verification'
```

List of available options: confidential\_comments\_events, confidential\_issues\_events, confidential\_note\_events, enable\_ssl\_verification, issues\_events, job\_events, merge\_requests\_events, note\_events, pipeline\_events, tag\_push\_events, wiki\_page\_events

.devscripts value: **SALSA\_KGB\_OPTIONS**

#### **--no-fail**

Don't stop on error when using **update\_repo** with **--all**.

.devscripts value: **SALSA\_NO\_FAIL** (yes/no)

#### **--rename-head, --no-rename-head**

Rename HEAD branch given by **--source-branch** into **--dest-branch** and change “default branch” of project. Works only with **update\_repo**.

.devscripts value: **SALSA\_RENAME\_HEAD** (yes/no)

**--source-branch**: default “master”

.devscripts value: **SALSA\_SOURCE\_BRANCH**

**--dest-branch**: default “debian/master”

.devscripts value: **SALSA\_DEST\_BRANCH**

#### **--tagpending, --no-tagpending, --disable-tagpending**

Enable, ignore or disable “tagpending” webhook.

.devscripts value: **SALSA\_TAGPENDING** (yes/ignore/no, default: ignore)

### **Merge requests options**

#### **--mr-title**

Title for merge request. Default: last commit title.

#### **--mr-desc**

Description of new MR. Default:

empty if **--mr-title** is set

last commit description if any

#### **--mr-dst-branch** (or second command line argument)

Destination branch. Default to “master”.

#### **--mr-dst-project** (or first command line argument)

Destination project. Default: project from which the current project was forked; or, if not found, “upstream” value found using **git remote --verbose show**; or using source project.

If **--mr-dst-project** is set to **same**, salsa will use source project as destination.

#### **--mr-src-branch**

Source branch. Default: current branch.

#### **--mr-src-project**

Source project. Default: current project found using **git remote --verbose show**.

#### **--mr-allow-squash, --no-mr-allow-squash**

Allow upstream project to squash your commits, this is the default.

.devscripts value: **SALSA\_MR\_ALLOW\_SQUASH** (yes/no)

#### **--mr-remove-source-branch, --no-mr-remove-source-branch**

Remove source branch if merge request is accepted. Default: no.

.devscripts value: **SALSA\_MR\_REMOVE\_SOURCE\_BRANCH** (yes/no)

**Options to manage other Gitlab instances****--api-url**

GitLab API. Default: <https://salsa.debian.org/api/v4>.

.devscripts value: **SALSA\_API\_URL**

**--git-server-url**

Default to “git@salsa.debian.org:”

.devscripts value: **SALSA\_GIT\_SERVER\_URL**

**--irker-server-url**

Default to “ircs://irc.oftc.net:6697/”

.devscripts value: **SALSA\_IRKER\_SERVER\_URL**

**--kgb-server-url**

Default to <http://kgb.debian.net:9418/webhook/?channel=>

.devscripts value: **SALSA\_KGB\_SERVER\_URL**

**--tagpending-server-url**

Default to <https://webhook.salsa.debian.org/tagpending/>

.devscripts value: **SALSA\_TAGPENDING\_SERVER\_URL**

*Configuration file example*

Example to use salsa with <https://gitlab.ow2.org> (group “lemonldap-ng”):

```
SALSA_TOKEN=`cat ~/.ow2-gitlab-token`
SALSA_API_URL=https://gitlab.ow2.org/api/v4
SALSA_GIT_SERVER_URL=git@gitlab.ow2.org:
SALSA_GROUP_ID=34
```

Then to use it, add something like this in your .bashrc file:

```
alias llng_admin='salsa --conffile ~/.salsa-ow2.conf'
```

**SEE ALSO**

**dpt-salsa**

**AUTHOR**

Xavier Guimard <yadd@debian.org>

**COPYRIGHT AND LICENSE**

Copyright (C) 2018, Xavier Guimard <yadd@debian.org>

It contains code formerly found in dpt-salsa (*pkg-perl-tools*) copyright 2018, gregor herrmann <gregoa@debian.org>.

This library is free software; you can redistribute it and/or modify it under the terms of the GNU General Public License as published by the Free Software Foundation; either version 2, or (at your option) any later version.

This program is distributed in the hope that it will be useful, but WITHOUT ANY WARRANTY; without even the implied warranty of MERCHANTABILITY or FITNESS FOR A PARTICULAR PURPOSE. See the GNU General Public License for more details.

You should have received a copy of the GNU General Public License along with this program. If not, see <http://www.gnu.org/licenses/>.