
pwclient

Sep 06, 2019

Contents

| | | |
|----------|---------------------------|-----------|
| 1 | pwclient | 1 |
| 1.1 | Installation | 1 |
| 1.2 | Getting Started | 1 |
| 1.3 | Development | 2 |
| 2 | Configuration | 3 |
| 2.1 | Format | 3 |
| 2.2 | Example | 4 |
| 2.3 | Legacy Format | 4 |
| 3 | Usage | 5 |
| 3.1 | pwclient | 5 |
| 4 | Release Notes | 11 |
| 4.1 | 1.0.0 | 11 |
| | Index | 13 |

pwclient is a VCS-agnostic tool for interacting with [Patchwork](#), the web-based patch tracking system.

1.1 Installation

The easiest way to install *pwclient* and its dependencies is using `pip`. To do so, run:

```
$ pip install pwclient
```

You can also install *pwclient* manually. First, install the required dependencies. On Fedora, run:

```
$ sudo dnf install python-pbr
```

On Ubuntu, run:

```
$ sudo apt-get install python-pbr
```

Once dependencies are installed, clone this repo and run `setup.py`:

```
$ git clone https://github.com/getpatchwork/pwclient
$ cd pwclient
$ pip install --user . # or 'sudo python setup.py install'
```

1.2 Getting Started

To use *pwclient*, you will need a `.pwclientrc` file, located in your home directory (`$HOME` or `~`). Patchwork itself provides sample `.pwclientrc` files for projects at `/project/{projectName}/pwclientrc/`. For example, [here](#) is the `.pwclientrc` file for Patchwork itself.

1.3 Development

If you're interested in contributing to *pwclient*, first clone the repo:

```
$ git clone https://github.com/getpatchwork/pwclient
$ cd pwclient
```

Create a *virtualenv*, then install the package in *editable* mode:

```
$ virtualenv .venv
$ source .venv/bin/activate
$ pip install --editable .
```

pwclient reads configuration from the `.pwclientrc` file, located in your home directory (`$HOME` or `~`). Patchwork itself provides sample `.pwclientrc` files for projects at:

`/project/{projectName}/pwclientrc/`

For example, [here](#) is the `.pwclientrc` file for Patchwork itself.

2.1 Format

The `.pwclientrc` file is an [INI-style](#) config file, **containing** an `options` section along with a section for each project.

The `options` section provides the following configuration options:

default The default project to use. Must be configured if not specifying a project via the command line.

signoff Add a `Signed-Off-By:` line to commit messages when applying patches using the `git-am` command. Defaults to `False`.

3way Enable three-way merge when applying patches using the `git-am` command. Defaults to `False`.

The names of the project sections must correspond to the project names in Patchwork, as reflected in the project's URL in Patchwork. Multiple projects can be defined, but no two projects can share the same name. Project sections require the following configuration options:

url The URL of the XMLRPC endpoint for the Patchwork instance that the project is available on. Typically `$PATCHWORK_URL/xmlrpc`. For example:

`https://patchwork.ozlabs.org/xmlrpc`

In addition, the following options are optional:

username Your Patchwork username.

password Your Patchwork password.

Note: Patchwork credentials are only needed for certain operations, such as updating the state of a patch. You will also require admin privileges on the instance in question.

2.2 Example

```
[base]
default = patchwork

[patchwork]
url = http://patchwork.ozlabs.org/xmlrpc/
username = johndoe
password = password
```

2.3 Legacy Format

Older Patchwork instances may provide a legacy version of the `.pwclientrc` file that did not support multiple projects. *pwclient* will automatically convert this version of the file to the latest version.

3.1 pwclient

```
usage: pwclient [-h]
                {apply,git-am,get,info,projects,check-list,check-info,check-create,
↪states,view,update,list,search}
                ...
```

-h, --help
show this help message and exit

Use 'pwclient <command> -help' for more info.

To avoid unicode encode/decode errors, you should export the LANG or LC_ALL environment variables according to the configured locales on your system. If these variables are already set, make sure that they point to valid and installed locales.

3.1.1 pwclient apply

```
usage: pwclient apply [--help] [-h] [-p PROJECT] PATCH_ID [PATCH_ID ...]
```

patch_id
patch ID

--help
show this help message and exit

-h
lookup by patch hash

-p <project>
lookup patch in project

3.1.2 pwclient check-create

```
usage: pwclient check-create [--help] [-h] [-p PROJECT] [-c CONTEXT]
                               [-s {pending,success,warning,fail}]
                               [-u TARGET_URL] [-d DESCRIPTION]
                               PATCH_ID [PATCH_ID ...]
```

patch_id
patch ID

--help
show this help message and exit

-h
lookup by patch hash

-p <project>
lookup patch in project

-c <context>

-s {pending, success, warning, fail}

-u <target_url>

-d <description>

3.1.3 pwclient check-info

```
usage: pwclient check-info ID
```

id
check ID

3.1.4 pwclient check-list

```
usage: pwclient check-list
```

3.1.5 pwclient get

```
usage: pwclient get [--help] [-h] [-p PROJECT] PATCH_ID [PATCH_ID ...]
```

patch_id
patch ID

--help
show this help message and exit

-h
lookup by patch hash

-p <project>
lookup patch in project

3.1.6 pwclient git-am

```
usage: pwclient git-am [--help] [-h] [-p PROJECT] [-s] [-3]
      PATCH_ID [PATCH_ID ...]
```

- patch_id**
patch ID
- help**
show this help message and exit
- h**
lookup by patch hash
- p** <project>
lookup patch in project
- s, --signoff**
pass ‘-signoff’ to ‘git-am’
- 3, --3way**
pass ‘-3way’ to ‘git-am’

3.1.7 pwclient info

```
usage: pwclient info [--help] [-h] [-p PROJECT] PATCH_ID [PATCH_ID ...]
```

- patch_id**
patch ID
- help**
show this help message and exit
- h**
lookup by patch hash
- p** <project>
lookup patch in project

3.1.8 pwclient list

```
usage: pwclient list [-h] [-s STATE] [-a {yes,no}] [-p PROJECT] [-w WHO]
      [-d WHO] [-n MAX#] [-N MAX#] [-m MESSAGEID] [-f FORMAT]
      [STR]
```

- str**
substring to search for patches by name
- h, --help**
show this help message and exit
- s** <state>
filter by patch state (e.g., ‘New’, ‘Accepted’, etc.)
- a** {yes,no}
filter by patch archived state

- p** <project>
filter by project name (see 'projects' for list)
- w** <who>
filter by submitter (name, e-mail substring search)
- d** <who>
filter by delegate (name, e-mail substring search)
- n** <max#>
limit results to first n
- N** <max#>
limit results to last N
- m** <messageid>
filter by Message-Id
- f** <format>
print output in the given format. You can use tags matching fields, e.g. `%{id}`, `%{state}`, or `%{msgid}`.

3.1.9 pwclient projects

```
usage: pwclient projects [-h]
```

- h, --help**
show this help message and exit

3.1.10 pwclient search

```
usage: pwclient search [-h] [-s STATE] [-a {yes,no}] [-p PROJECT] [-w WHO]
                        [-d WHO] [-n MAX#] [-N MAX#] [-m MESSAGEID] [-f FORMAT]
                        [STR]
```

- str**
substring to search for patches by name
- h, --help**
show this help message and exit
- s** <state>
filter by patch state (e.g., 'New', 'Accepted', etc.)
- a** {yes,no}
filter by patch archived state
- p** <project>
filter by project name (see 'projects' for list)
- w** <who>
filter by submitter (name, e-mail substring search)
- d** <who>
filter by delegate (name, e-mail substring search)
- n** <max#>
limit results to first n

- N** <max#>
limit results to last N
- m** <messageid>
filter by Message-Id
- f** <format>
print output in the given format. You can use tags matching fields, e.g. `%{id}`, `%{state}`, or `%{msgid}`.

3.1.11 pwclient states

```
usage: pwclient states [-h]
```

- h, --help**
show this help message and exit

3.1.12 pwclient update

```
usage: pwclient update [--help] [-h] [-p PROJECT] [-c COMMIT-REF] [-s STATE]
      [-a {yes,no}]
      PATCH_ID [PATCH_ID ...]
```

patch_id
patch ID

- help**
show this help message and exit

- h**
lookup by patch hash
- p** <project>
lookup patch in project
- c** <commit-ref>
commit reference hash
- s** <state>
set patch state (e.g., 'Accepted', 'Superseded' etc.)
- a** {yes,no}
set patch archived state

using a COMMIT-REF allows for only one ID to be specified

3.1.13 pwclient view

```
usage: pwclient view [--help] [-h] [-p PROJECT] PATCH_ID [PATCH_ID ...]
```

patch_id
patch ID

- help**
show this help message and exit

pwclient

- h** lookup by patch hash
- p** <project>
 lookup patch in project

CHAPTER 4

Release Notes

4.1 1.0.0

4.1.1 Prelude

Initial release of *pwclient* package.

Symbols

- help
 - pwclient-apply command line option, 5
 - pwclient-check-create command line option, 6
 - pwclient-get command line option, 6
 - pwclient-git-am command line option, 7
 - pwclient-info command line option, 7
 - pwclient-update command line option, 9
 - pwclient-view command line option, 9
- 3, -3way
 - pwclient-git-am command line option, 7
- N <max#>
 - pwclient-list command line option, 8
 - pwclient-search command line option, 8
- a {yes,no}
 - pwclient-list command line option, 7
 - pwclient-search command line option, 8
 - pwclient-update command line option, 9
- c <commit-ref>
 - pwclient-update command line option, 9
- c <context>
 - pwclient-check-create command line option, 6
- d <description>
 - pwclient-check-create command line option, 6
- d <who>
 - pwclient-list command line option, 8
 - pwclient-search command line option, 8
- f <format>
 - pwclient-list command line option, 8
 - pwclient-search command line option, 9
- h
 - pwclient-apply command line option, 5
 - pwclient-check-create command line option, 6
 - pwclient-get command line option, 6
 - pwclient-git-am command line option, 7
 - pwclient-info command line option, 7
 - pwclient-update command line option, 9
 - pwclient-view command line option, 9
- h, -help
 - pwclient command line option, 5
 - pwclient-list command line option, 7
 - pwclient-projects command line option, 8
 - pwclient-search command line option, 8
 - pwclient-states command line option, 9
- m <messageid>
 - pwclient-list command line option, 8
 - pwclient-search command line option, 9
- n <max#>
 - pwclient-list command line option, 8
 - pwclient-search command line option, 8
- P <project>
 - pwclient-apply command line option, 5
 - pwclient-check-create command line option, 6
 - pwclient-get command line option, 6
 - pwclient-git-am command line

option,7
 pwclient-info command line option,7
 pwclient-list command line option,7
 pwclient-search command line option,8
 pwclient-update command line option,9
 pwclient-view command line option,10
 -s {pending,success,warning,fail}
 pwclient-check-create command line option,6
 -s <state>
 pwclient-list command line option,7
 pwclient-search command line option,8
 pwclient-update command line option,9
 -s, -signoff
 pwclient-git-am command line option,7
 -u <target_url>
 pwclient-check-create command line option,6
 -w <who>
 pwclient-list command line option,8
 pwclient-search command line option,8

I

id
 pwclient-check-info command line option,6

P

patch_id
 pwclient-apply command line option,5
 pwclient-check-create command line option,6
 pwclient-get command line option,6
 pwclient-git-am command line option,7
 pwclient-info command line option,7
 pwclient-update command line option,9
 pwclient-view command line option,9
 pwclient command line option
 -h, -help,5
 pwclient-apply command line option
 -help,5
 -h,5
 -p <project>,5
 patch_id,5

pwclient-check-create command line option
 -help,6
 -c <context>,6
 -d <description>,6
 -h,6
 -p <project>,6
 -s {pending,success,warning,fail},6
 -u <target_url>,6
 patch_id,6
 pwclient-check-info command line option
 id,6
 pwclient-get command line option
 -help,6
 -h,6
 -p <project>,6
 patch_id,6
 pwclient-git-am command line option
 -help,7
 -3, -3way,7
 -h,7
 -p <project>,7
 -s, -signoff,7
 patch_id,7
 pwclient-info command line option
 -help,7
 -h,7
 -p <project>,7
 patch_id,7
 pwclient-list command line option
 -N <max#>,8
 -a {yes,no},7
 -d <who>,8
 -f <format>,8
 -h, -help,7
 -m <messageid>,8
 -n <max#>,8
 -p <project>,7
 -s <state>,7
 -w <who>,8
 str,7
 pwclient-projects command line option
 -h, -help,8
 pwclient-search command line option
 -N <max#>,8
 -a {yes,no},8
 -d <who>,8
 -f <format>,9
 -h, -help,8
 -m <messageid>,9
 -n <max#>,8
 -p <project>,8
 -s <state>,8

-w <who>,8
str,8
pwclient-states command line option
-h, -help,9
pwclient-update command line option
-help,9
-a {yes,no},9
-c <commit-ref>,9
-h,9
-p <project>,9
-s <state>,9
patch_id,9
pwclient-view command line option
-help,9
-h,9
-p <project>,10
patch_id,9

S

str
pwclient-list command line option,7
pwclient-search command line
option,8