Kafka patch review tool

Kafka JIRA and Reviewboard script

1. Setup

1. Follow instructions here to setup the jira-python package
2. Follow instructions here to setup the reviewboard python tools
3. Install the argparse module

```
On Linux -> sudo yum install python-argparse
On Mac -> sudo easy_install argparse
```

2. Usage

```
nnarkhed-mn:kafka-git-idea nnarkhed$ python kafka-patch-review.py --help
usage: kafka-patch-review.py [-h] -b BRANCH -j JIRA [-s SUMMARY]
                             [-d DESCRIPTION] [-r REVIEWBOARD] [-t TESTING]
                             [-v VERSION] [-db]

Kafka patch review tool

optional arguments:
  -h, --help            show this help message and exit
  -b BRANCH, --branch BRANCH
                        Tracking branch to create diff against
  -j JIRA, --jira JIRA  JIRA corresponding to the reviewboard
  -s SUMMARY, --summary SUMMARY
                        Summary for the reviewboard
  -d DESCRIPTION, --description DESCRIPTION
                        Description for reviewboard
  -r REVIEWBOARD, --rb REVIEWBOARD
                        Review board that needs to be updated
  -t TESTING, --testing-done TESTING
                        Text for the Testing Done section of the reviewboard
  -v VERSION, --version VERSION
                        Version of the patch
  -db, --debug          Enable debug mode
```

3. Upload patch

1. Specify the branch against which the patch should be created (-b)
2. Specify the corresponding JIRA (-j)
3. Specify an **optional** summary (-s) and description (-d) for the reviewboard

Example:

```
python kafka-patch-review.py -b origin/trunk -j KAFKA-42
```

4. Update patch

1. Specify the branch against which the patch should be created (-b)
2. Specify the corresponding JIRA (--jira)
3. Specify the rb to be updated (-r)
4. Specify an **optional** summary (-s) and description (-d) for the reviewboard, if you want to update it
5. Specify an **optional** version of the patch. This will be appended to the jira to create a file named JIRA-<version>.patch. The purpose is to be able to upload multiple patches to the JIRA. This has no bearing on the reviewboard update.

Example:

```
python kafka-patch-review.py -b origin/trunk -j KAFKA-42 -r 14081
```

JIRA command line tool

1. Download the JIRA command line package

Install the jira-python package

```
sudo easy_install jira-python
```

2. Configure JIRA username and password

Include a jira.ini file in your $HOME directory that contains your Apache JIRA username and password

```
nnarkhed-mn:~ nnarkhed$ cat ~/jira.ini
user=nehanarkhede
password=***********
```

Reviewboard

This is a quick tutorial on using Review Board with Kafka.

1. Install the post-review tool

If you are on RHEL, Fedora or CentOS, follow these steps

```
sudo yum install python-setuptools
sudo easy_install -U RBTools
```

If you are on Mac, follow these steps

```
sudo easy_install -U setuptools
sudo easy_install -U RBTools
```

For other platforms, follow the instructions here to setup the post-review tool.

2. Configure Stuff

Then you need to configure a few things to make it work:

First set the review board url to use. You can do this from in git:

```
git config reviewboard.url https://reviews.apache.org
```

If you checked out using the git wip http url that confusingly won't work with review board. So you need to configure an override to use the non-http url. You can do this by adding a config file like this:

```
jkreps$ cat ~/.reviewboardrc
REPOSITORY = 'git://git.apache.org/kafka.git'
TARGET_GROUPS = 'kafka'
GUESS_FIELDS = True
```

FAQ

When I run the script, it throws the following error and exits

```
nnarkhed$python kafka-patch-review.py -b trunk -j KAFKA-42
There don't seem to be any diffs
```

There are 2 reasons that can cause this -

- The code is not checked into your local branch
- The -b branch is not pointing to the remote branch. In the example above, "trunk" is specified as the branch, which is the local branch. The correct value for the -b (--branch) option is the remote branch. "git branch -r" gives the list of the remote branch names.

When I run the script, it throws the following error and exits

```
Error uploading diff

Your review request still exists, but the diff is not attached.
```

One of the most common root causes of this error are that the git remote branches are not up-to-date. Since the script already does that, it is probably due to some other problem. You can run the script with the --debug option that will make post-review run in the debug mode and list the root cause of the issue.