

Gerrit的用户使用文档

Gerrit服务器地址：172.17.140.140:8081
用户登录：ldap账户(@sysenew前的名字)

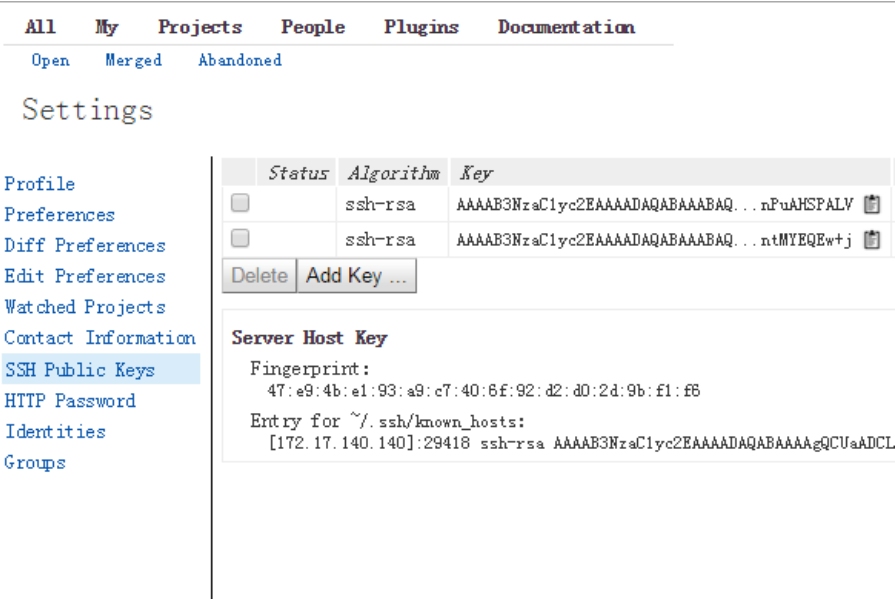
安装：git-review, git

```
yum isntall git-review git -y
```

普通用户

权限：只有review的权限，没有submit的权限，即没有push的权限。

- 1. 登录Gerrit 利用ldap账号登录；
- 2. 上传ssh key
本地：ssh-keygen,默认
gerrit用户设置->ssh public keys->add key



添加~/.ssh/id_rsa.pub中的key

项目开发

Projects->General->点击Clone-with-commit-msg-hook->点击ssh，拷贝克隆命令， 结果如下：

Project glance

Clone | Clone with commit-msg hook | anonymous http | http | ssh |
git clone ssh://wenzhang1@172.17.140.140:29418/glance && scp -p -P 29418 wenzhang1@172.17.140.140:hooks/commit-msg glance/.git/hoo

Description

Project Options

State:

Submit Type:

Allow content merges:

Create a new change for every commit not in the target branch:

Require Change-Id in commit message:

Maximum Git object size limit:

Contributor Agreements

Require Signed-off-by in commit message:

Project Commands

Commands:

```
git clone ssh://wenzhang1@172.17.140.140:29418/glance && scp -p -P 29418 wenzhang1@172.17.140.140:hooks/commit-msg glance/.git/hoo
cd glance
git checkout -tb ${workbranch} origin/${workbranch} #切换到对应的分支
{working...Such as: touch readme}
git add *
git commit -m "Add readme file"
#提交review,将工作的分支提交至gerriit平台等待review
git review -r origin ${workbranch}
```

Search term

Change 48 - Needs Code-Review

Add test_file which is useless

Change-Id: Ic98d43c41ca6896679cc300f4025e4f910d37f06

Owner: wenzhang1

Reviewers:

Project: glance

Branch: easystack

Topic:

Strategy: Merge if Necessary

Uploaded: in the future

Code-Review

Author: wenzhang1 <wenzhang1@sysnew.com> Oct 9, 2016 3:11 AM
Committer: wenzhang1 <wenzhang1@sysnew.com> Oct 9, 2016 3:11 AM
Commit: 745427f22619d212c0df65df9b78404efe8f91ba
Parent(s): 16f77306d7a3d3f783672d387d96ed6a08571db3
Change-Id: Ic98d43c41ca6896679cc300f4025e4f910d37f06

Files	Open All	Diff against: Base	Edit	Comments	Size
Commit Message					0
A test_file					+0, -0

History

wenzhang1

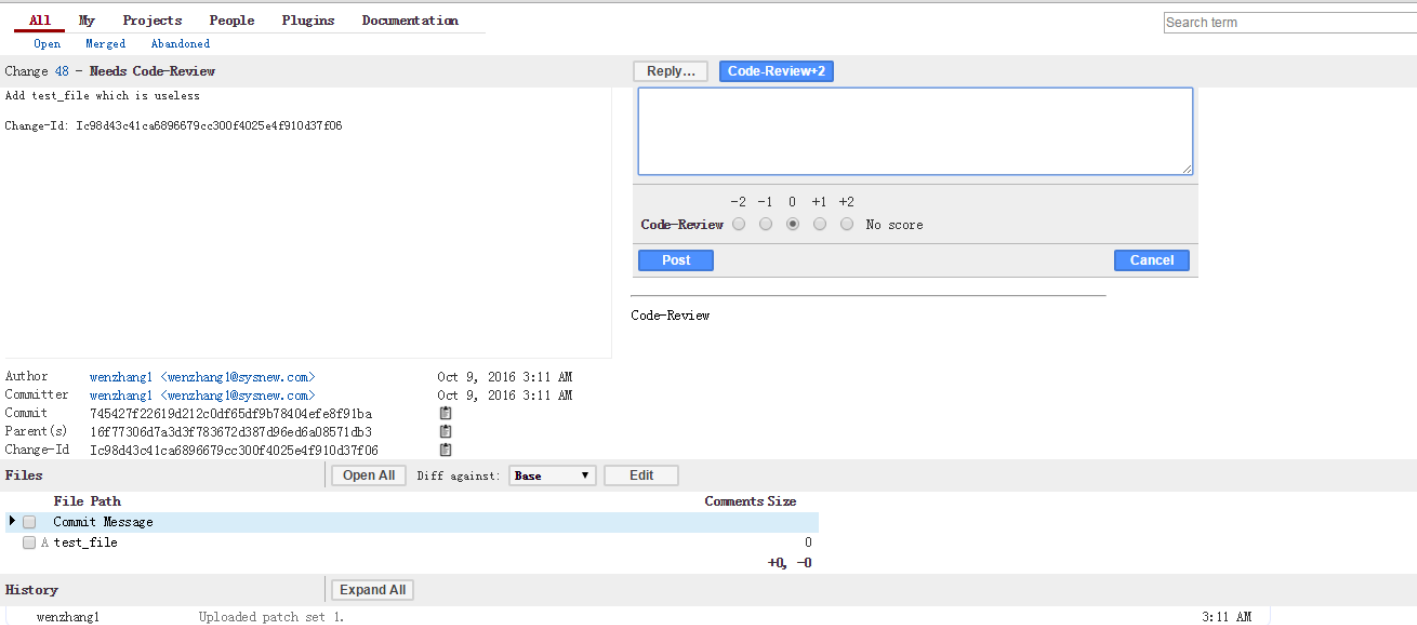
Uploaded patch set 1.

Oct 9 3:11 AM

上图即为此次提交变更的状态，可以邀请reviewers(project group-core member)进行review。当reviewer认同此次变更，并且点击code review+2时，点击submit，此次变更就被并入版本库中，并被同步至gitlab。

项目核心人员

权限：对项目具有:Read、Abandon、Label Code-review、以及submit权限。
当有变更提交时，登录gerrit。进入all->open，进入提交的change，点击reply，如图：



可以写下review，评分。-2表示不能合并，+2表示可以合并。也可以直接点击code-review，表示可以合并。点击submit以merge。

管理员

权限：拥有所有权限，项目的创建、删除，用户组的创建和删除、用户组的权限分配。

创建项目：

```
#CLI的方式
ssh -p 29418 wenzhang1@172.17.140.140 gerrit create-project glance #wenzhang1需要有管理员权限

#dashboard
登录gerrit->Project->Create new project
```

删除项目：

删除项目可以通过界面操作，如下图：

project nova

Clone | Clone with commit-msg hook | anonymous http | http | ssh |
git clone ssh://wenzhang1@172.17.140.140:29418/nova

Description

Project Options

State: Active

Submit Type: Merge if Necessary

Allow content merges: INHERIT (true)

Create a new change for every commit not in the target branch: INHERIT (false)

Require Change-Id in commit message: INHERIT (true)

Maximum Git object size limit:

Contributor Agreements

Require Signed-off-by in commit message: INHERIT (false)

Save Changes

Project Commands

Commands: Delete... Run GC Create Change Edit Config

Are you really sure you want to delete the project: "nova"?
☒ Delete project even if open changes exist?
☐ Preserve GIT Repository?
Delete

若要保留本地库，勾选第二个。

添加组以及分配权限：

1. Dashboard操作：

创建group：

All My Projects **People** Plugins Documentation
List Groups Create New Group

Create Group

Create New Group

Create Group

people->create group，输入group name。

分配权限：

进入项目->access->edit，如下：

Project glance

Rights Inherit From:

Reference: refs/heads/*			
Read	<input type="button" value="ALLOW"/> glance	<input type="checkbox"/> Exclusive	X
	<input type="button" value="ALLOW"/> glance-core		X
Group Name:	<input type="text"/>	<input type="button" value="Add"/>	X
Abandon	<input type="button" value="ALLOW"/> glance-core	<input type="checkbox"/> Exclusive	X
Add Group			
Push Signed Tag	<input type="button" value="ALLOW"/> glance-core	<input type="checkbox"/> Exclusive	X
Add Group			
Label Code-Review	<input type="button" value="-2"/> <input type="button" value="+2"/> glance-core	<input type="checkbox"/> Exclusive	X
Add Group			
Submit	<input type="button" value="ALLOW"/> glance-core	<input type="checkbox"/> Exclusive	X
Add Group			
<input type="button" value="Add Permission ..."/>			

[Add Reference](#)

Commit Message (optional):

在add permission选项中选择要分配的权限。如添加read权限，点击Add Group，添加group，使得此group具有此权限。如上图，glance-core具有read、abandon、push tag、label code review、submit的权限。

若要删除某个组的权限，只需要在上图中指定权限下删除此group即可。

2. 脚本建组以及分配权限

脚本：

```

#创建组
ssh -p 29418 wenzhang1@172.17.140.140 gerit create-group ${group-name}

#批量创建
# setup-gerit-access.sh
#!/bin/bash
set -x
#ssh -p 29418 qianli1@172.17.140.140 gerit create-project $1
ssh -p 29418 wenzhang1@172.17.140.140 gerit create-group $1
ssh -p 29418 wenzhang1@172.17.140.140 gerit create-group $1-core
git init $1 ; cd $1
git remote add origin ssh://wenzhang1@172.17.140.140:29418/$1
git pull origin refs/meta/config
cat << EOF > project.config
[access "refs/heads/*"]
    abandon = group $1-core
    label-Code-Review = -2..+2 group $1-core
    submit = group $1-core
[access "refs/heads/*"]
    read = group $1-core
    read = group $1
EOF
cat << EOF > groups

```

```
`ssh -p 29418 wenzhang1@172.17.140.140 gerrit ls-groups -v | awk '{print $2 "\t" $1}' | grep -e '$'\t'$1`
EOF
git add --all
git commit -a -m 'Updated permissions'
git push origin HEAD:refs/meta/config
cd ..
```

运行./setup-gerrit-access.sh glance，则创建了group glance，以及group glance-core并分配在项目glance中的权限。

3. 删除组

组的删除需要通过删除数据库才能完成。

删除组的脚本如下：

```
#如删除glance组
mysql -u root
use gerritdb;
delete from account_group_names where name='glacne';
delete from account_groups where name='glance';
```

Gerrit与Gitlab项目同步

代码迁移：

```
#clone gitlab代码库中的代码
git config --global user.name "wenzhang1"
git config --global user.email "wenzhang1@sysnew.com"
git clone git@172.17.249.122:easystack-openstack/nova.git
cd nova

#编辑.gitreview
vim .gitreview
[gitreview]
host=172.17.140.140 # gerrit 服务器地址
port=29418 # gerrit 端口
project=nova.git #项目名称

# 添加.gitreview到版本库
git add .gitreview
git commit .gitreview -m 'Admin add .gitreview file.'

#在gerrit上创建项目
ssh -p 29418 wenzhang1@172.17.140.140 gerrit create-project nova

#添加gerrit server，推送gitlab的tags 到gerrit.
git remote add gerrit ssh://wenzhang1@172.17.140.140:29418/nova
git push --tags gerrit refs/remotes/origin/*:refs/heads/*
```

同步设置：

完成gerrit中项目的初始化设置，当开发者向gerrit中提交变更，并被接收并入gerrit本地库，此时需要将变更同步至gitlab库，配置同步，/etc/gerrit/gerrit_site/etc/replication.config:

```
[remote "nova"]
projects = nova
url=git@172.17.249.122:easystack-openstack/nova.git
push = +refs/heads/*:refs/heads/*
push = +refs/tags/*:refs/tags/*
push = +refs/changes/*:refs/changes/*
thread = 3
```

将本地root用户的id_rsa.pub添加至gitlab。并且编辑~/.ssh/config文件：

```
vi ~/.ssh/config
Host 172.17.140.140
IdentityFile ~/.ssh/id_rsa
#更改config权限
chmod 600 ~/.ssh/config
```

```
#设置host-key
echo "StrictHostKeyChecking no" >> /etc/ssh/ssh_config #默认不添加;
或:
ssh-keyscan -t rsa 172.17.249.122 > ~/.ssh/known_hosts
ssh-keygen -H -f ~/.ssh/known_hosts
```