**シェルプログラミングでGitlab管理**

**(2015/11/30 – 2015/12/1)**

**Phần 1: Cài đặt Gitlab Community Edition (CE) trên CentOS**

Làm theo hướng dẫn trong trang chủ:

Link: <https://about.gitlab.com/downloads/#centos7>

Tham khảo: <http://to-developer.com/blog/?p=1726>

++ B1: Cài đặt và cấu hình các phụ thuộc cần thiết

Nếu bạn cài đặt Postfix để gửi email xin vui lòng chọn 'Internet Site "trong khi cài đặt. Thay vì sử dụng Postfix bạn cũng có thể sử dụng Sendmail hoặc cấu hình một máy chủ tùy chỉnh SMTP.Nếu bạn muốn sử dụng Exim, hãy cấu hình nó như một máy chủ SMTP.Trên Centos 6 và 7, các lệnh dưới đây cũng sẽ mở HTTP và truy cập SSH trong hệ thống tường lửa

>sudo yum install curl openssh-server

>sudo systemctl enable sshd

>sudo systemctl start sshd

>sudo yum install postfix

>sudo systemctl enable postfix

>sudo systemctl start postfix

>sudo firewall-cmd --permanent --add-service=http

>sudo systemctl reload firewalld

++ B2: Cài đặt gitlab package server

>curl https://packages.gitlab.com/install/repositories/gitlab/gitlab-ce/script.rpm.sh | sudo bash

>sudo yum install gitlab-ce

++ B3: Config and start gitlab

>sudo gitlab-ctl reconfigure

++ B4: Sửa file config

>sudo vim /etc/gitlab/gitlab.rb

+++++++++++++++++++++++++++++++++++++++++++++

external\_url 'http://localhost:8000'

gitlab\_rails['gravatar\_enabled'] = false

+++++++++++++++++++++++++++++++++++++++++++++

++ B5: Reconfig lại 1 lần nữa

sudo gitlab-ctl reconfigure

++ B6: Khởi động gitlab server

>sudo gitlab-ctl start # Khởi động

>sudo gitlab-ctl stop # Dừng lại

>sudo gitlab-ctl status # Check trạng thái

++ B7: Sau khi đã hoàn thành 6 bước thì ta dùng brower đăng nhập vào đường dẫn:

http://localhost:8000/

Và nhập vào usesername and password là:

+++ username: root

+++ password: 5iveL!fe

Phần 2: Các thao tác với Gitlab

+ Sao chép 1 kho chứa đã tồn tại

Nếu bạn muốn có 1 bản sao của 1 khó chứa Git có sẵn thì thực thi lệnh:

>git clone url\_ssh\_mã\_nguồn thư\_mục\_chứa\_mã\_nguồn

Nếu không định nghĩa tên thì nó sẽ tự động tạo thư mục theo tên đường dẫn

có trong link url

+ Tiếp theo ta sẽ tạo 1 project với gitlab

Create project with gitlab: vien\_test\_first\_project

ssh: git@localhost:root/vien\_test\_first\_project.git

+ Tạo ssh key trên centos7 như sau:

>ssh-keygen -t rsa

Lúc này nó sẽ tự động sinh trong thư mục /home/vagrant/.ssh/ 2 file đó là

id\_rsa và id\_rsa.pub(Đây là file chứa public key)

Ta sẽ mở file id\_rsa.pub bằng vim

>sudo vim /home/vagrant/.ssh/id\_rsa.pub

Coppy key và add vào trong gitlab làm public key

++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++

+ Task1: Như vậy ta sẽ cần tạo ra 1 chương trình shell script nhập ssh url vào

++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++

và nó sẽ tự động clone code về máy

git clone $ssh\_url $dest\_folder

Nếu không có $dest\_folder thì nó sẽ tự động sinh ra 1 folder tương ứng với tên chương trình

VD:

git@localhost:root/vien\_test\_first\_project.git -> vien\_test\_first\_project sẽ tự động được tạo ra

++ Phương án 1: Fix link cứng(テスト版)

#!/bin/bash

#clone git with url exists

respository="git@localhost:root/vien\_test\_first\_project.git"

git\_source="/home/vagrant\_workspace/git\_source/vien\_test\_first\_project"

git clone "$respository" "$git\_source"

++ Phương án 2: Chuyển $url và $dest thành biến và sẽ được nhập từ bàn phím(本番)

#!/bin/bash

echo "Please input project SSH link:"

read ssh\_url\_project

while [ $ssh\_url\_project == "" ]; do

echo "Khong duoc de trong ! Yeu cau nhap lai 1 lan nua"

read ssh\_url\_project

done

echo "Please input destional folder(co the de trong):"

read dest\_folder\_path

if [ "$dest\_folder\_path" != "" ]; then

git clone "$ssh\_url\_project" "$dest\_folder\_path"

else

git clone "$ssh\_url\_project"

fi

echo "......Done......"

+++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++

+ Task2: Create Repository

+++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++

#!/bin/bash

echo "新しいリポジトリ名を入力する："

read repository\_name

while [ "$repository\_name" == "" ]; do

echo "何にも記録していないので、もう一度入力してください!"

read repository\_name

done

echo "新しいリポジトリ名を確認しました！"

echo "早々お待ちくださいませ...."

git init $repository\_name

echo "完成しました!"