Gitlab备份与恢复

1.1.8 备份 Gitlab 创建备份使用Gitlab一键安装包安装Gitlab非常简单, 同样的备份恢复与迁移也非常简单. 使用一条命令即可创建完整的Gitlab备份:  
gitlab-rake gitlab: backup:create  
使用以上命令会在 /var/opt/gitlab/backups 目录下创建一个名称类似为 1393513186\_gitlab\_backup.tar 的压缩包, 这个压缩包就是Gitlab整个的完整部分, 其中开头的 1393513186 是备份创建的日期.

Gitlab 修改备份文件默认目录你也可以通过修改 /etc/gitlab/gitlab.rb 来修改默认存放备份文件的目录:  
gitlab\_rails['backup\_path'] = '/mnt/backups'/mnt/backups 修改为你想存放备份的目录即可, 修改完成之后使用 gitlab-ctl reconfigure 命令重载配置文件即可.

Gitlab 自动备份也可以通过crontab使用备份命令实现自动备份:  
sudo su -  
crontab -e  
加入以下, 实现每天凌晨2点进行一次自动备份:  
0 2 \* \* \* /opt/gitlab/bin/gitlab-rake gitlab:backup:create

Gitlab 恢复同样, Gitlab的从备份恢复也非常简单:

# 停止相关数据连接服务

gitlab-ctl stop unicorn

gitlab-ctl stop sidekiq

# 从1393513186编号备份中恢复

gitlab-rake gitlab:backup:restore BACKUP=1393513186

# 启动Gitlab

sudo gitlab-ctl start Gitlab

迁移迁移如同备份与恢复的步骤一样, 只需要将老服务器/var/opt/gitlab/backups目录下的备份文件拷贝到新服务器上的/var/opt/gitlab/backups即可(如果你没修改过默认备份目录的话). 但是需要注意的是新服务器上的Gitlab的版本必须与创建备份时的Gitlab版 本号相同. 比如新服务器安装的是最新的7.60版本的Gitlab, 那么迁移之前, 最好将老服务器的Gitlab 升级为7.60在进行备份.  
其他最新版本的Gitlab已经修复了HTTPS设备的BUG, 现在使用 官方HTTPS配置 即可轻松启用HTTPS.