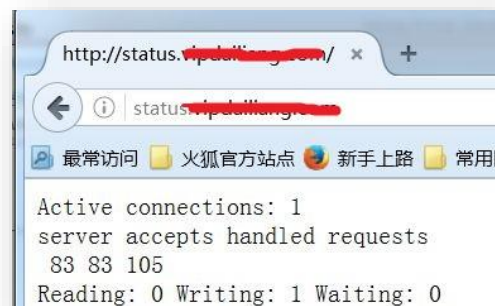


## 1. 克隆 4 台**新虚拟机**，IP 配置如下：

虚拟机 7	eth2:201.1.1.100
虚拟机 8	eth2:201.1.1.5
	eth3:201.1.2.5
虚拟机 9	eth3:201.1.2.100
虚拟机 10	eth3:201.1.2.200

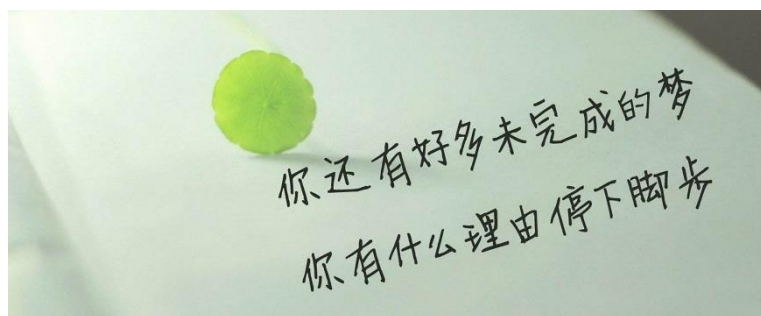


- 在虚拟机 8 上源码安装 Nginx 软件，要求如下：
- 要求安装 ssl 模块以及 stub\_status 模块
- 安装前创建用户 nginx，安装时指定安装用户为 nginx
- 该 nginx 配置为后端两台虚拟机的调度器



## 2. 在虚拟机 9 和 10 上部署 LNMP 环境：

- 部署 lnmp 环境（支持动静分离）
- 部署测试性页面（参考 lnmp\_soft/php\_scripts/php-memcached-demo.tar.gz）
- 测试登陆页面的效果
- 查看/etc/php-fpm.d/www.conf 配置文件，确认 session 会话信息保存位置
- 到/var/lib/php/session 目录下查看 session 文件



## 3. 使用 memcached 实现会话信息共享

- 虚拟机 8 安装部署 memcached 缓存数据库
- 修改虚拟机 9 和 10 的配置，让 PHP 保存会话信息到 memcached
- 默认 PHP 无法连接 memcached，需要安装对应的扩展模块

- d) 完成后再次通过集群调度器访问登陆页面，测试登陆效果（为了查看效果，可用将后台页面的背景颜色适当调整）

#### 4. 配置 Tomcat 服务器，要求如下：

- a) 在虚拟机 9 和 10 上安装部署 tomcat
- b) 在虚拟机 8 上配置调度器，调度后台 2 个 tomcat 站点
- c) 测试查看负载均衡效果



#### 5. 使用版本控制系统：

- a) 使用 subversion 创建版本仓库
- b) 将自己之前写的 shell 脚本导入到仓库中
- c) 配置 SVN 服务器，让所有客户端都可用访问 svn 服务器，下载上传代码到服务器
- d) 模拟多人协作修改 shell 脚本（修改不同文件，修改相同文件不同和，修改相同文件相同行）
- e) 登陆 github 网站，创建 note 仓库，将自己之前的笔记上传 github 仓库（再也不用担心笔记丢失）
- f)

#### 6. 将 nginx 源码包制作成 RPM 软件包

#### 7. 预习，提前搜索 vpn 相关知识