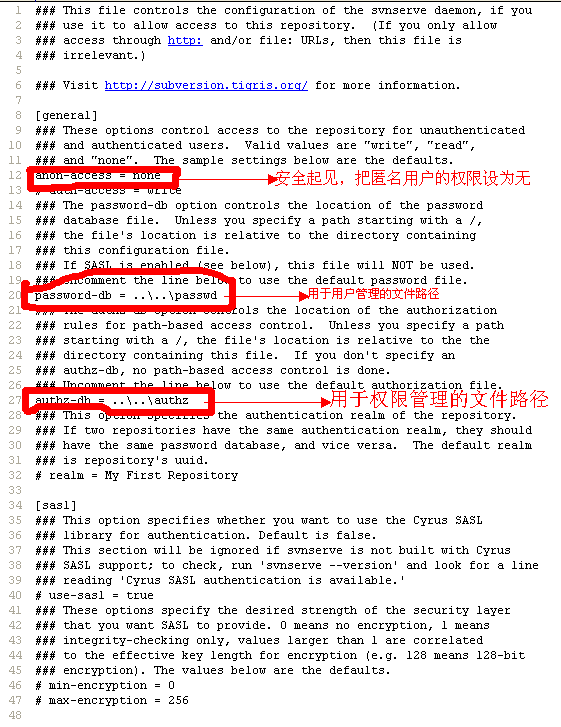
# 多仓库配置

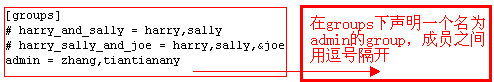
以前使用svn的时候会只建一个仓库，然后把所有的项目都放在一个仓库里面。虽然这样也可以实现版本控制，但是随着项目的增多就会发现一个问题，因为整个仓库是有一连串的版本号的，然后项目又是交替进行的，这样就出现了单个仓库版本号断层的问题。比如有一个仓库repository，有一个项目tiantian,有一个项目tianyi，刚开始天天操作并提交了可能版本为10，然后tianyi这个项目操作并提交了，版本为11了，tiantian再操作并提交版本就为12了，断层了。虽然说这也没有很大的问题，但是当查看历史版本的时候那种感觉不好！所以现在就是一个项目一个仓库了。  
假设仓库都统一放在E:\svnserver目录下，那么先执行命令svnadmin create E:\svnserver/tiantian建立仓库tiantian，同样建立仓库tianyi。把tiantian下面的conf下面的passwd和authz拷贝到E:\svnserver下进行统一的用户管理和权限管理。然后分别在tiantian和tianyi下面的conf/svnserve.conf文件中把用户管理和权限管理的相关文件设为E:\svnserver下面的passwd和authz，注意在配置文件的每一行的开始处不能有空格

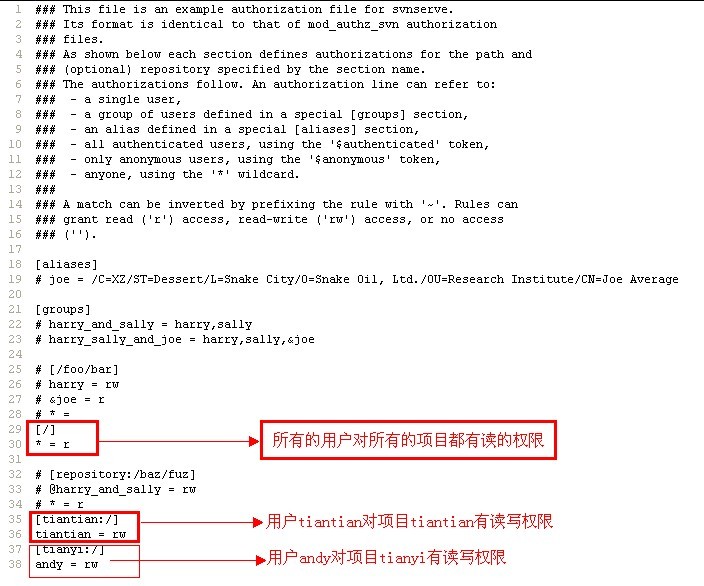


这样就可以在E:\svnserver下的passwd和authz文件中进行统一的配置进行管理了。

passwd中的用户名和密码是键值对形式，如tiantian = tiantian，则表示用户tiantian的密码为tiantian

在authz中进行权限管理时可单独对用户进行权限配置，也可以为一个组进行配置，声明一个组时，可以在authz文件中的groups下进行声明，组成员之间用逗号隔开。



在进行授权的时候，可以直接对某一目录授权，也可以是对某一个项目授权。授权一般是有授权目标和用户权限组成的，都是在授权目标后跟着用户权限。如：[/]tiantian = rw，则表示给用户tiantian所有的权限，其中的"/"表示根目录，这是直接对目录授权的形式。  
直接对项目授权：[tiantian:/]andy = rw。表示给用户andy对项目tiantian的所有内容的读写权限  
  


给用户组授权，在给用户组授权的时候需要在用户组前面加上"@"标记，用以表明该用户是一个用户组

http://dl.iteye.com/upload/attachment/553509/392ea195-a175-3274-9098-26baef752f69.png

添加服务到自启动服务中：



在CMD中执行上述命令就可以将目录E:\svnserver下的所有版本库都加入到系统服务中