## 创建用户

建立一个linux用户，discenter，密码也是discenter，其余环境参数参照cbpay.

命令如下：

useradd –g pay –d /ztkx/discenter –m discenter

注意：最好手动敲，如果直接黏贴可能出现问题（由于win7和linux中空格等字符编码不一致）。

set -o vi

使用vi方式编辑命令，对于习惯vi的人来说可以加快编辑命令的效率。

export PS1=`hostname`':$PWD>'

用来设置显示的样式，

在未设置的情况下进入终端后显示[test33@/ztkx/discenter]，

设置完成后则显示为test33:/ztkx/discenter>。

## Coredump日志

在默认情况下coredump功能是关闭的，

进行判断时在控制台中输入：ulimit –c ，如果返回为“0”，则表示文件大小限制为0，也就是不生成coredump文件 ，

但是在开发过程中有时为了调试问题，还是需要在特定的用户环境下打开 core 文件产生的设置。

在用户的 ~/.bash\_profile 里加上 ulimit -c unlimited 来让特定的用户可以产生 core 文件。

，unlimited表示大小没有限制，也可改为 1024等来限制coredump文件的大小。

修改core目录的命令(注意

echo "./core\_%s\_%p\_%u\_%g\_%t\_%e.log" > /proc/sys/kernel/core\_pattern

* 文件名的格式为core\_致产生 core 的信号\_进程ID\_用户ID\_实际组ID\_时间 (由1970年1月1日计起的秒数)

  %p - insert pid into filename 添加 pid

    %u - insert current uid into filename 添加当前 uid

    %g - insert current gid into filename 添加当前 gid

    %s - insert signal that caused the coredump into the filename 添加导致产生 core 的信号

    %t - insert UNIX time that the coredump occurred into filename 添加 core 文件生成时的 unix 时间

    %h - insert hostname where the coredump happened into filename 添加主机名

    %e - insert coredumping executable name into filename 添加命令名

备份：|/usr/libexec/abrt-hook-ccpp %s %c %p %u %g %t e

%c的含义未查到。

测试core日志功能：kill -s SIGSEGV $$