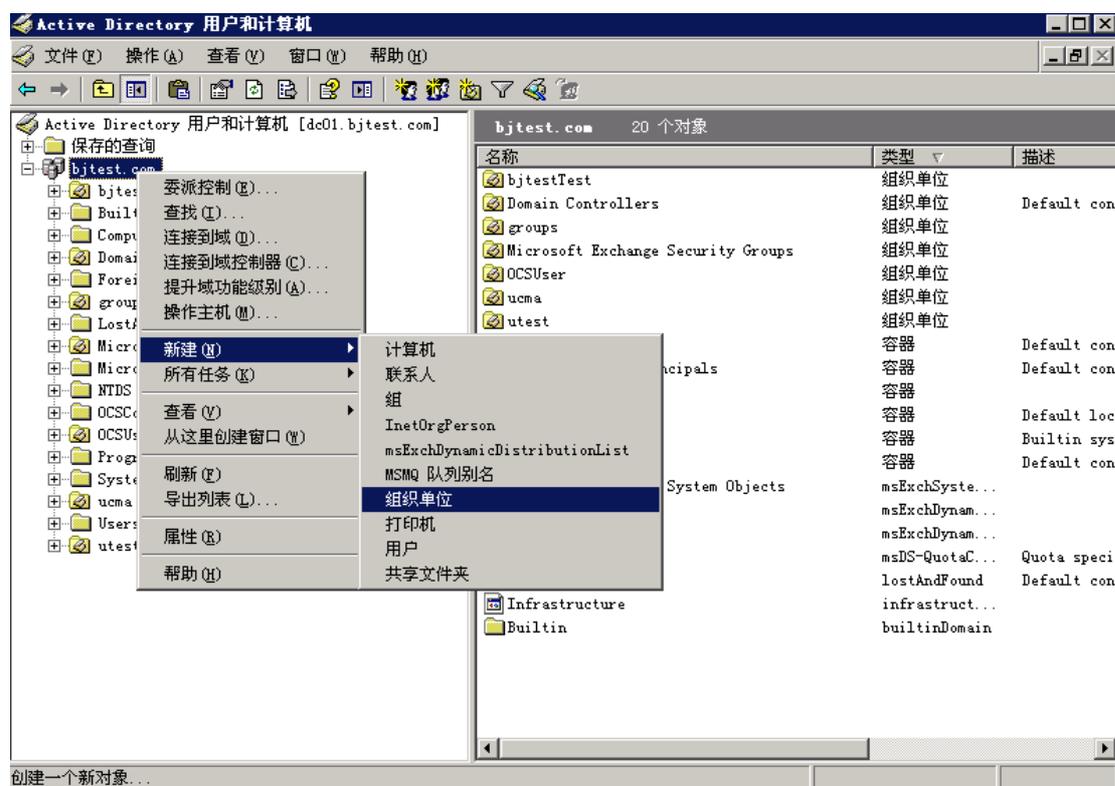


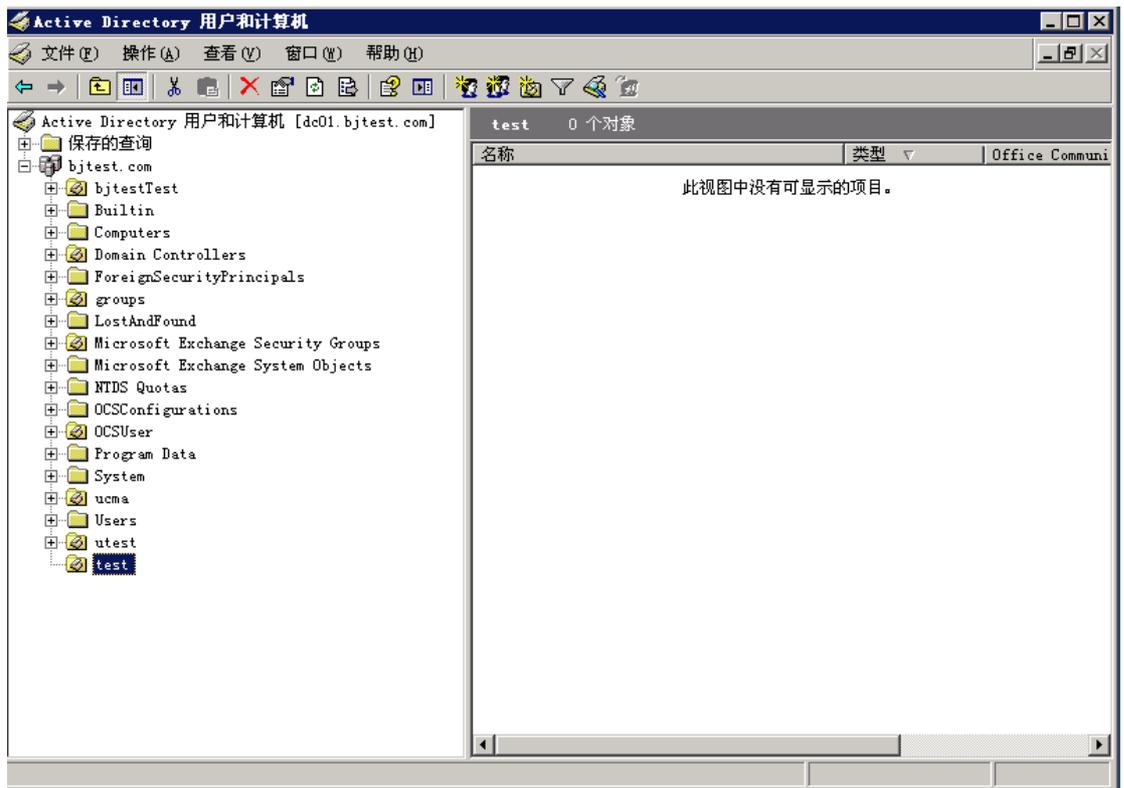
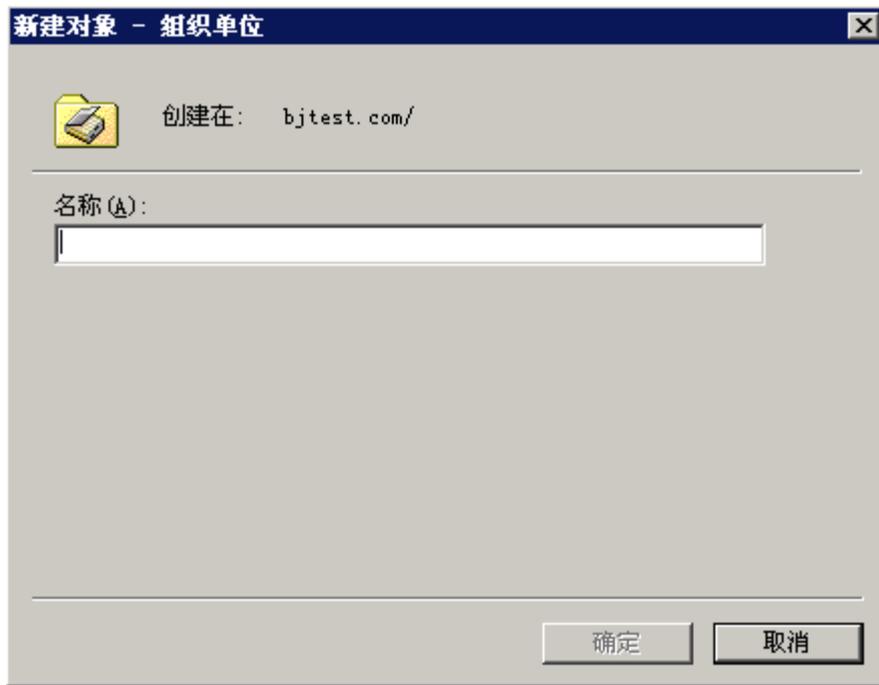
AD 用户、组织单元的导入和导出

AD 用户的导入和导出工具有很多种，这里只解释微软官方提供的工具

第三方工具，通过 API、ADSI 二次开发的工具不在讨论之列。在下接触 AD 不就，文中难免疏漏，还请各位指正，先行谢过。

入门演示





在 c:\下面保存此文件为 test.csv 路径和文件名可以使用其它



```
test.csv - 记事本
文件(F) 编辑(E) 格式(O) 查看(V) 帮助(H)
DN,objectClass,sAMAccountName
"CN=test1111,OU=test,DC=bjtest,DC=COM",user,test1111
"CN=test2222,OU=test,DC=bjtest,DC=COM",user,test2222
"CN=test33333,OU=test,DC=bjtest,DC=COM",user,test33333
```

注意别留多余的空格自找麻烦

其实也就是这样一个表

DN	objectClass	sAMAccountName
"CN=test1111,OU=test,DC=bjtest,DC=COM"	user	test1111
"CN=test2222,OU=test,DC=bjtest,DC=COM"	user	test2222
"CN=test33333,OU=test,DC=bjtest,DC=COM"	user	test33333

用逗号分割的形式写，最后一项后面不写逗号

然后再命令行下输入下面指令

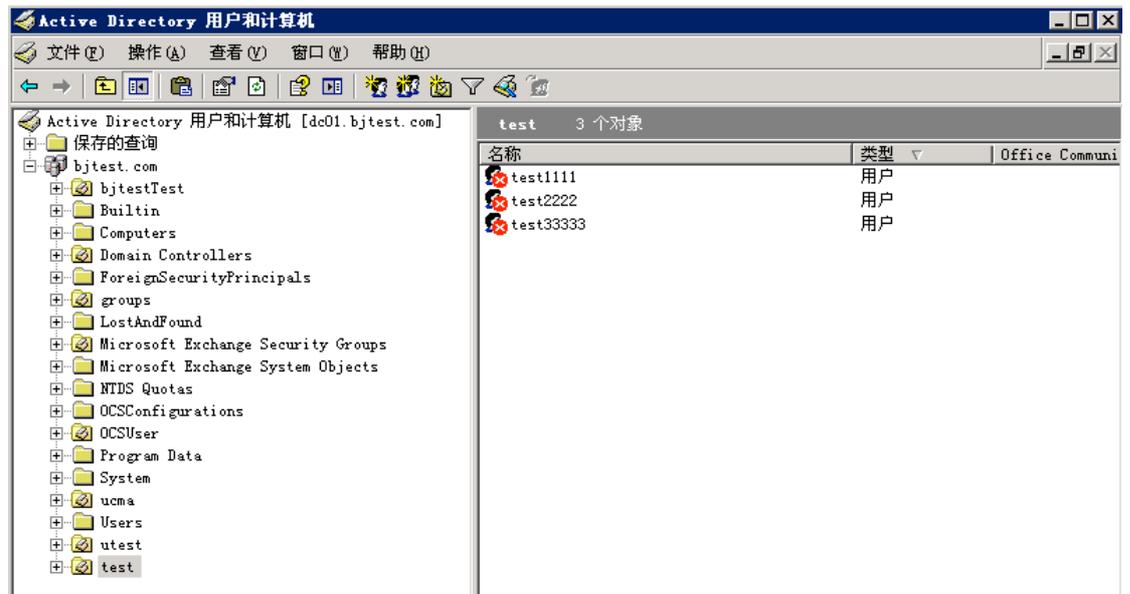
第一条 type test.csv 是为了显示文本中的内容

第二条是用来导入用户的命令

```
C:\WINDOWS\system32\cmd.exe
C:\>type test.csv
DN,objectClass,sAMAccountName
"CN=test1111,OU=test,DC=bjtest,DC=COM",user,test1111
"CN=test2222,OU=test,DC=bjtest,DC=COM",user,test2222
"CN=test33333,OU=test,DC=bjtest,DC=COM",user,test33333
C:\>csude -i -f test.csv
连接到 "<null>"
用 SSPI 作为当前用户登录
从 "test.csv" 文件导入目录
加载项目....
3 个项目修改成功。

命令成功完成

C:\>
```



用户已经导入了，但用户是禁用的。如果要该用户能登陆域要把用户启用。

DN	objectClass	sAMAccountName
----	-------------	----------------

这三个属性是必须的，也是最少的导入属性。

大批量导入用户

这样导入只是为了有个初步的认识，下面介绍一些实际操作中的问题。

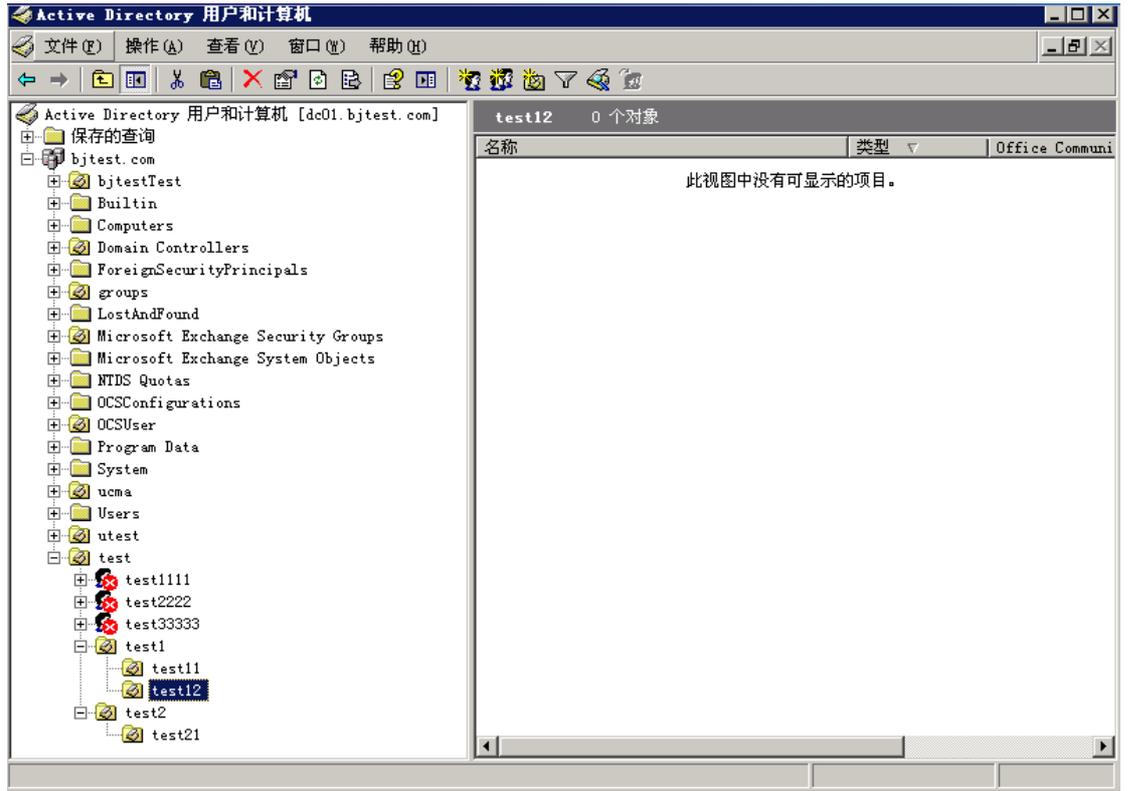
比如一个测试 AD，为了模拟生产 AD 的环境，需要把用户和组织单元全部导入，事实上我遇到了一个上百个组织单元，接近 7000 个用户信息的导入操作，涉及真实信息需要保密，此处只还是做一个模拟的例子。

既然需要导入这么多用户，肯定要有相应的信息了，于是我得到了两个 csv 文件。

一个是组织单元的导出文件，一个是用户的导出文件。

文件的内容不便透露，这里还是模拟一下。

下面再 test 下面建几个组织单元（OU），如图。



然后建几个用户分别放进对应的 ou 里，格式从前向后依次是最小的 ou，如下格式

DN	objectClass	sAMAccountName
"CN=testAA,OU=test11,OU=test1,OU=test,DC=bjtest,DC=com"	user	test1111
"CN=testbb,OU=test12,OU=test1,OU=test,DC=bjtest,DC=com"	user	test2222
"CN=testcc,OU=test2,OU=test,DC=bjtest,DC=com"	user	test3333

文本内容这样写

DN,objectClass,sAMAccountName

"CN=testAA,OU=test11,OU=test1,OU=test,DC=bjtest,DC=com",user,testAA

"CN=testbb,OU=test12,OU=test1,OU=test,DC=bjtest,DC=com",user,testbb

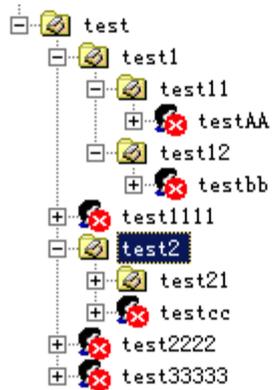
"CN=testcc,OU=test2,OU=test,DC=bjtest,DC=com",user,testcc



```
C:\>csude -i -f test.csv
连接到“<null>”
用 SSPI 作为当前用户登录
从“test.csv”文件导入目录
加载项目...
3 个项目修改成功。

命令成功完成

C:\>_
```



用户已经导入，注意这种最简的导入方式用户默认是禁用的，并且是没有密码的，后面将讲解怎么批量添加密码。

如果希望导入用户是启用状态的，请使用 `userAccountControl`，将其值设定为 512 或 66080 等值，具体数值的含义请参考 <http://support.microsoft.com/kb/305144>

同样可以写入电话，Email，办公部门等等其他属性，关于批量修改特定属性将在后面介绍。

使用 CSVDE 导入的方法要求有两点

- 1 用户导入所使用的文本文件，格式必须严格符合。
- 2 放置导入用户的组织单元需要事先建立好，才能导入。

针对问题 1，比如文本格式有误，会提示

```
C:\>csude -i -f test.csv
连接到“<null>”
用 SSPI 作为当前用户登录
从“test.csv”文件导入目录
加载项目。
在 2: 其他 行添加出错
服务器边的错误为“提供的名称不是正确格式的帐户名。”
0 个项目修改成功。
在程序中出现一个错误
没有写入日志文件。要生成日志文件，请
通过 -j 选项来指定日志文件路径。
```

可以使用命令

```
C:\>csvde -i -f test.csv -j c:\
连接到“<null>”
用 SSPI 作为当前用户登录
从“test.csv”文件导入目录
加载项目。
在 2: 已存在 行添加出错
服务器边的错误为“指定的用户已存在。”
0 个项目修改成功。
在程序中出现一个错误
```

并会在指定路径输出一个错误日志，更多关于 CSVDE 的用法请参考微软的文档
<http://support.microsoft.com/kb/327620/zh-cn>

[http://technet.microsoft.com/en-us/library/cc772704\(WS.10\).aspx](http://technet.microsoft.com/en-us/library/cc772704(WS.10).aspx)。

针对要点 2 下面介绍：

如何批量导入组织单元

```
常规参数
=====
-i          打开导入模式<默认为导出>
-f filename 输入或输出文件名
-s servername 要绑定到的服务器<默认为计算机域的 DC>
-v          打开详细模式
-c FromDN ToDN 发生的从 FromDN 到 ToDN 的取代
-j path     日志文件的位置
-t port     端口号<默认为 389>
-u          使用 Unicode 格式
-?         帮助
```

在讲解导入之前，我们先了解下 csvde 的导出，之后就可以明白导入文本的写法。

在命令行下输入

```
C:\>csvde -f users.csv -d "ou=test,dc=bjtest,dc=com" -r
"(&(objectcategory=person)(objectclass=user))" -l DN,objectClass,samaccountname -j c:\
```

```
C:\>csvde -f users.csv -d "ou=test,dc=bjtest,dc=com" -r "(&(objectcategory=person)
(objectclass=user))" -l DN,objectClass,samaccountname -j c:\
连接到“<null>”
用 SSPI 作为当前用户登录
将目录导出到文件 users.csv
搜索项目...
写出项目
.....
导出完毕。后续处理正在进行...
导出了 6 个项目
命令成功完成
```

打开 users.csv 文件

```
users.csv - 记事本
文件(F) 编辑(E) 格式(O) 查看(V) 帮助(H)
DN,objectClass,sAMAccountName
"CN=test1111,OU=test,DC=bjtest,DC=com",user,test1111
"CN=test2222,OU=test,DC=bjtest,DC=com",user,test2222
"CN=test33333,OU=test,DC=bjtest,DC=com",user,test33333
"CN=testAA,OU=test11,OU=test1,OU=test,DC=bjtest,DC=com",user,testAA
"CN=testbb,OU=test12,OU=test1,OU=test,DC=bjtest,DC=com",user,testbb
"CN=testcc,OU=test2,OU=test,DC=bjtest,DC=com",user,testcc
```

下面解释这个命令的含义

-d "ou=test,dc=bjtest,dc=com" 搜索 bjtest.com 下的 ou 为 test 下的内容

-r "(&(objectcategory=person)(objectclass=user))" 搜索 objectcategory 属性为 person 并且 objectclass=user 的对象 (&是“与”的意思)

-l DN,objectClass,samaccountname 是导出哪些属性

```
常规参数
=====
-i 打开导入模式<默认为导出>
-f filename 输入或输出文件名
-s servername 要绑定到的服务器<默认为计算机域的 DC>
-v 打开详细模式
-c FromDN ToDN 发生的从 FromDN 到 ToDN 的取代
-j path 日志文件的位置
-t port 端口号<默认为 389>
-u 使用 Unicode 格式
-? 帮助
```

在讲解导入之前，我们先了解下 csvde 的导出，之后就可以明白导入文本的写法。

在命令行下输入

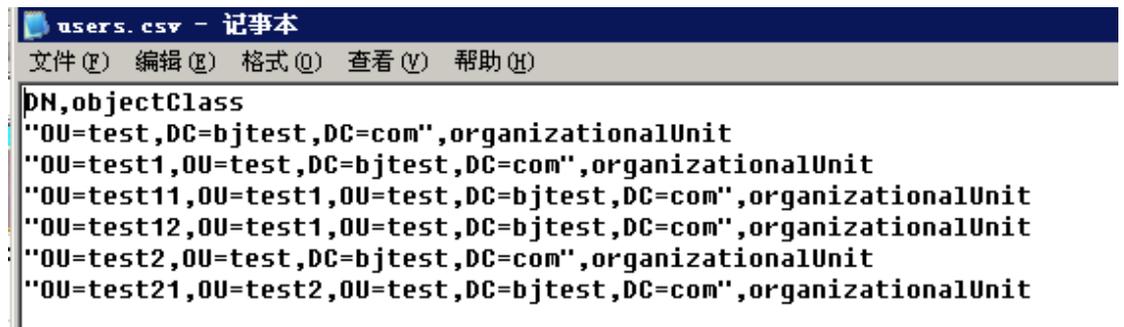
```
C:\>csvde -f users.csv -d "ou=test,dc=bjtest,dc=com" -r
"(&(objectcategory=person)(objectclass=user))" -l DN,objectClass,samaccountname -j c:\
```

把这个命令修改一下

```
C:\>csvde -f users.csv -d "ou=test,dc=bjtest,dc=com" -r
"(&(objectcategory=OrganizationalUnit))" -l DN,objectClass,samaccountname -j c:\
```

```
C:\>csvde -f users.csv -d "ou=test,dc=bjtest,dc=com" -r "(objectclass=organizationalunit)" -l DN,objectClass,samaccountname -j c:\
连接到 "(null)"
用 SSPI 作为当前用户登录
将目录导出到文件 users.csv
搜索项目...
写出项目
...
导出完毕。后续处理正在进行...
导出了 6 个项目
命令成功完成
C:\>
```

查看导出的文本文件



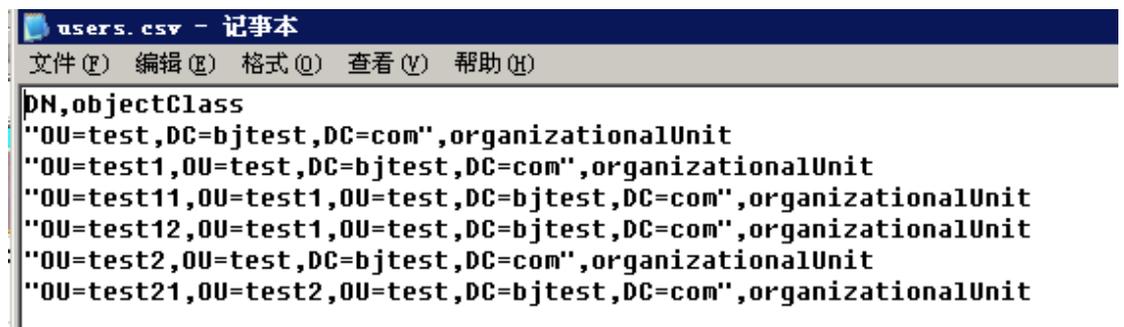
```
users.csv - 记事本
文件(F) 编辑(E) 格式(O) 查看(V) 帮助(H)
DN,objectClass
"OU=test,DC=bjtest,DC=com",organizationalUnit
"OU=test1,OU=test,DC=bjtest,DC=com",organizationalUnit
"OU=test11,OU=test1,OU=test,DC=bjtest,DC=com",organizationalUnit
"OU=test12,OU=test1,OU=test,DC=bjtest,DC=com",organizationalUnit
"OU=test2,OU=test,DC=bjtest,DC=com",organizationalUnit
"OU=test21,OU=test2,OU=test,DC=bjtest,DC=com",organizationalUnit
```

就只含有组织单元了。

这样，也就是按这个格式写入你的组织结构就可以了。但是编辑也是有技巧的，下面介绍两个技巧，用到两个软件，一个是我们熟悉的 office 系列的 Excel 表格制作工具，一个是 UtralEdit

用 Excel 编辑文本的技巧

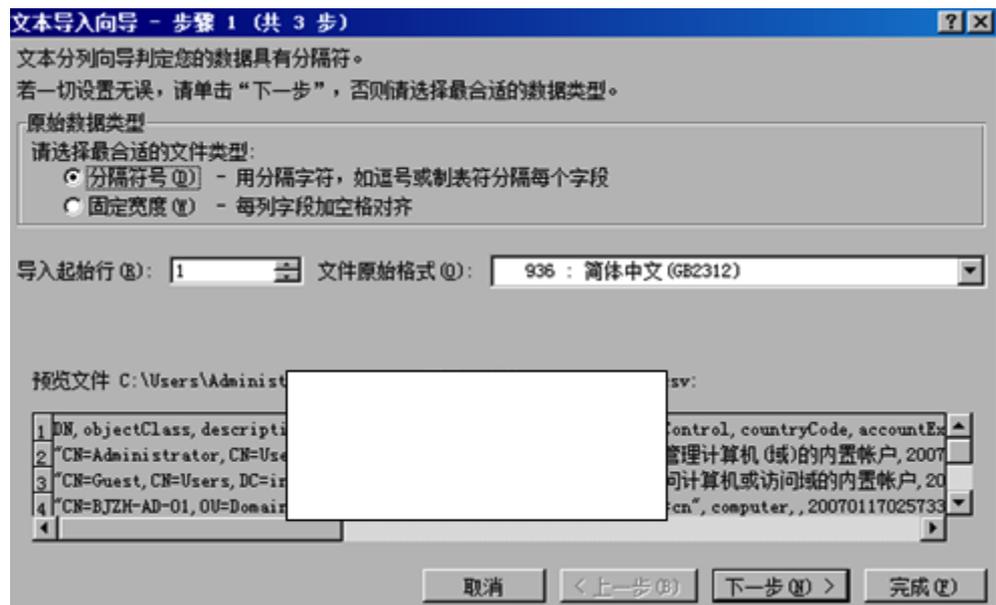
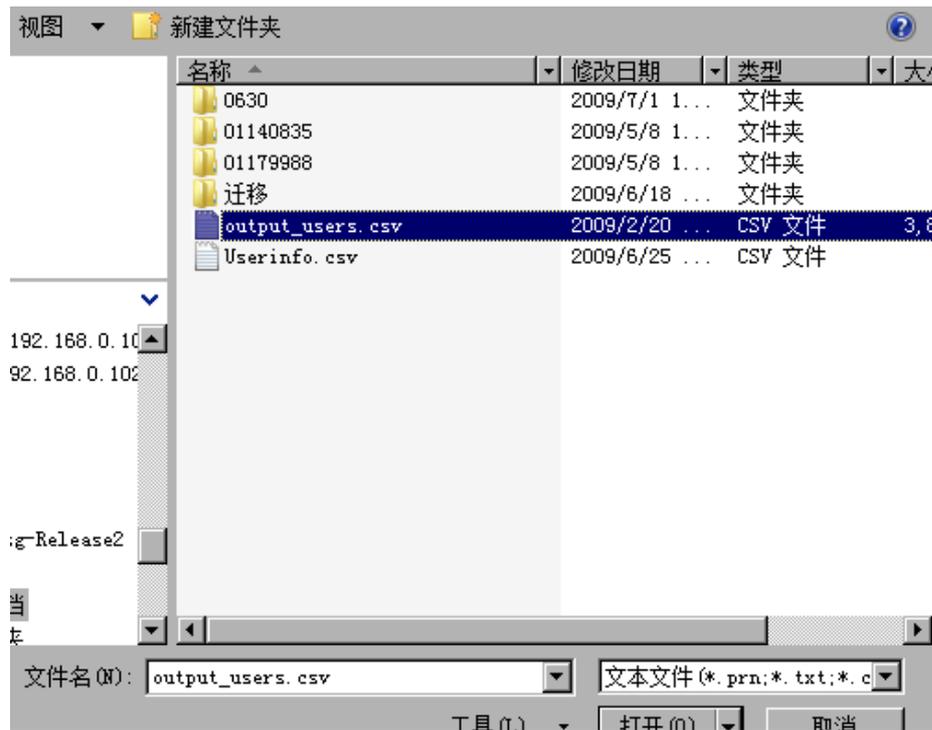
使用 Excel 主要是用来编辑逗号分割的文件 (*.csv)，原因是我得到的那个逗号分割的文件太大了.. 实际演示下就明白了。打开 Excel



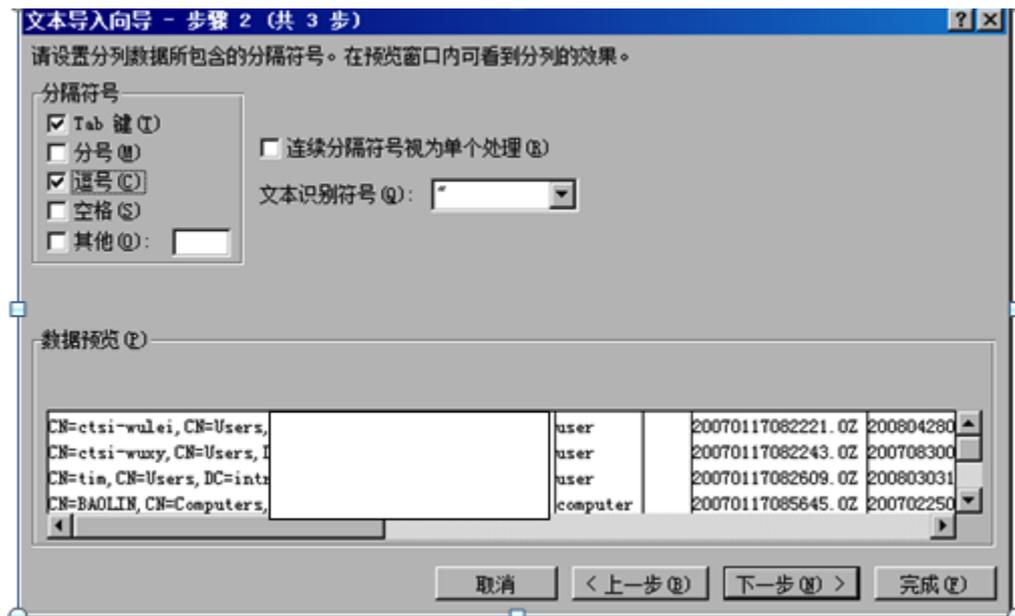
```
users.csv - 记事本
文件(F) 编辑(E) 格式(O) 查看(V) 帮助(H)
DN,objectClass
"OU=test,DC=bjtest,DC=com",organizationalUnit
"OU=test1,OU=test,DC=bjtest,DC=com",organizationalUnit
"OU=test11,OU=test1,OU=test,DC=bjtest,DC=com",organizationalUnit
"OU=test12,OU=test1,OU=test,DC=bjtest,DC=com",organizationalUnit
"OU=test2,OU=test,DC=bjtest,DC=com",organizationalUnit
"OU=test21,OU=test2,OU=test,DC=bjtest,DC=com",organizationalUnit
```

我的是 2007，2003 的一样操作。

右下角需选中 csv，打开你的目标文件



此处选中逗号



如果文本识别符号选择无的话，可以把引号内的内容拆开，这样更容易编辑了。



具体编辑的技巧就是 Excel 的使用技巧了，不再多说，避免跑题太远。总之利用 Excel 可以方便的编辑导出的文件。编辑的目的，我这次操作是更换域名，也就是批量的把 DC=*,DC=* 替换成 DC=bjtest,DC=com，还算方便。

用 UltraEdit 编辑文本的技巧

这部分待续，UE 卸载掉了，关键词：列操作，替换

使用 **ldifde** 命令导入导出 **AD** 对象

LDIFDE 的使用与 CSVDE 的操作相似，由于使用不多，了解更多信息请访问以下两个链接了解。

<http://support.microsoft.com/kb/237677/zh-cn>

<http://www.microsoft.com/china/technet/prodtechnol/windowsserver2003/technologies/directory/active-directory/stepbystep/adbulk.aspx>

简单对这个命令进行介绍

查看 LDIFDE 常规参数

单击**开始**，指向**程序**，指向**附件**，然后单击“**命令提示符**”。

在命令提示符下，键入：**ldifde ?**。然后，屏幕上会显示内置的帮助文件，如下所示。

常规参数

```
-i                打开 Import 模式 (默认为 Export)
-f filename      输入或输出文件名
-s servername    要绑定到的服务器名称(默认为登录域的 DC)
-c FromDN ToDN  从 FromDN 到 ToDN 发生的取代
-v              打开 Verbose 模式
-j              日志文件的位置
-t              端口号(默认为 389)
-u              使用 Unicode 格式
-?              帮助
```

使用上面的

```
C:\>csvde -f users.csv -d "ou=test,dc=bjtest,dc=com" -r
"(&(objectcategory=person)(objectclass=user))" -l DN,objectClass,samaccountname -j c:\
```

把 csvde 换成 ldifde 可以得到下面的结果，可以看出只是导出的格式不同而以。

```
C:\>ldifde -f users.csv -d "ou=test,dc=bjtest,dc=com" -r "<&(objectcategory=OrganizationalUnit)>" -l DN,objectClass,samaccountname -j c:\
连接到 "dc01.bjtest.com"
用 SSPI 作为当前用户登录
将目录导出到文件 users.csv
搜索项目...
写出项目.....
导出了 6 个项目

命令成功完成

C:\>
```

```
users.csv - 记事本
文件(F) 编辑(E) 格式(O) 查看(V) 帮助(H)

dn: OU=test,DC=bjtest,DC=com
changetype: add
objectClass: top
objectClass: organizationalUnit

dn: OU=test1,OU=test,DC=bjtest,DC=com
changetype: add
objectClass: top
objectClass: organizationalUnit

dn: OU=test11,OU=test1,OU=test,DC=bjtest,DC=com
changetype: add
objectClass: top
objectClass: organizationalUnit

dn: OU=test12,OU=test1,OU=test,DC=bjtest,DC=com
changetype: add
objectClass: top
objectClass: organizationalUnit

dn: OU=test2,OU=test,DC=bjtest,DC=com
changetype: add
objectClass: top
objectClass: organizationalUnit

dn: OU=test21,OU=test2,OU=test,DC=bjtest,DC=com
changetype: add
objectClass: top
objectClass: organizationalUnit
```

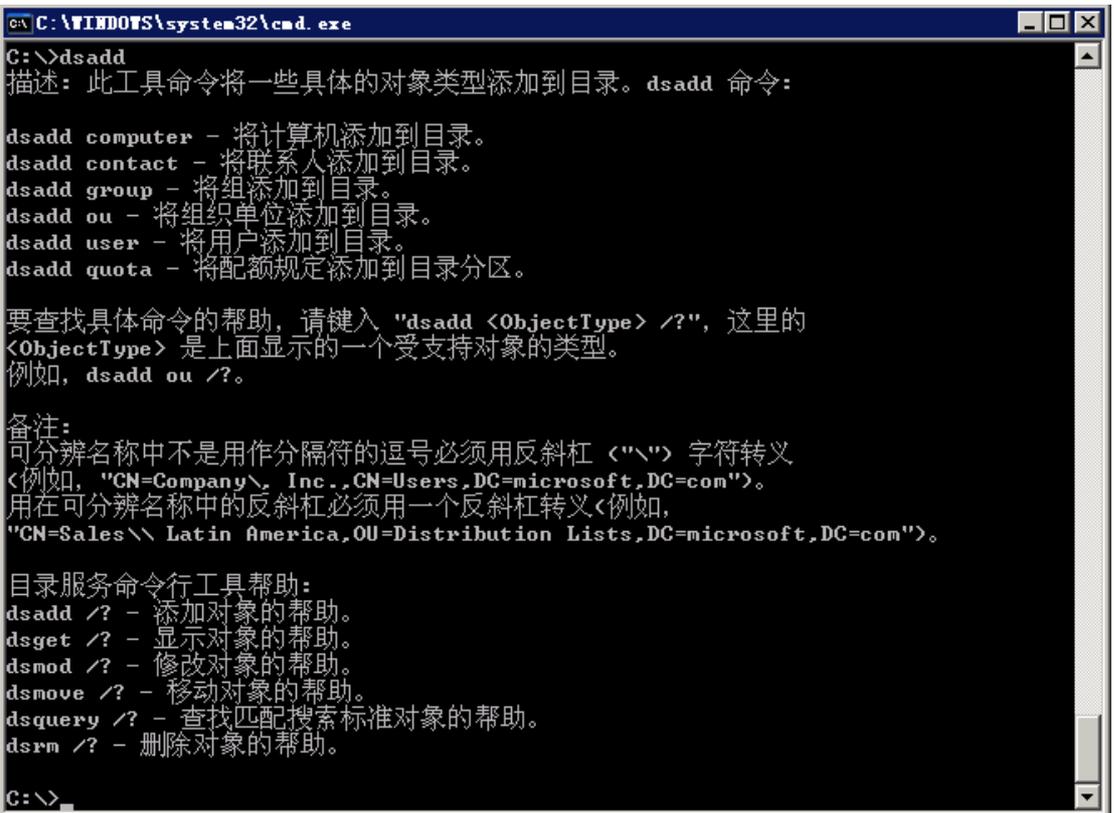
和 CSVDE 同样 Ldifde 也不能导入或导出密码。此处不更多介绍

既然不能导入导出密码，就要另外想办法。7000 个用户逐个设置密码显然是不现实的。

于是需要使用另一个命令 DSADD 系列。

使用 **DSADD**、**DSMOD** 等操作 **AD** 对象

直接在根目录键入 dsadd 可以看到命令的说明



```
C:\WINDOWS\system32\cmd.exe
C:\>dsadd
描述: 此工具命令将一些具体的对象类型添加到目录。dsadd 命令:

dsadd computer - 将计算机添加到目录。
dsadd contact - 将联系人添加到目录。
dsadd group - 将组添加到目录。
dsadd ou - 将组织单位添加到目录。
dsadd user - 将用户添加到目录。
dsadd quota - 将配额规定添加到目录分区。

要查找具体命令的帮助, 请键入 "dsadd <ObjectType> /?", 这里的
<ObjectType> 是上面显示的一个受支持对象的类型。
例如, dsadd ou /?。

备注:
可分辨名称中不是用作分隔符的逗号必须用反斜杠 (<"\"> 字符转义
<例如, "CN=Company\, Inc.,CN=Users,DC=microsoft,DC=com">。
用在可分辨名称中的反斜杠必须用一个反斜杠转义<例如,
"CN=Sales\\ Latin America,OU=Distribution Lists,DC=microsoft,DC=com">。

目录服务命令行工具帮助:
dsadd /? - 添加对象的帮助。
dsget /? - 显示对象的帮助。
dsmod /? - 修改对象的帮助。
dsmove /? - 移动对象的帮助。
dsquery /? - 查找匹配搜索标准对象的帮助。
dsrm /? - 删除对象的帮助。

C:\>
```

输入 dsadd user /? 可以得到关于 user 的相关内容

内容太长, 我把文本输出了

另请参阅：

dsadd computer /? - 将计算机添加到目录的帮助。

dsadd contact /? - 将联系人添加到目录的帮助。

dsadd group /? - 将组添加到目录的帮助。

dsadd ou /? - 将组织单位添加到目录的帮助。

dsadd user /? - 将用户添加到目录的帮助。

dsadd quota /? - 将配额添加的目录的帮助

目录服务命令行工具帮助：

dsadd /? - 添加对象的帮助。

dsget /? - 显示对象的帮助。

dsmod /? - 修改对象的帮助。

dsmove /? - 移动对象的帮助。

dsquery /? - 查找与搜索标准匹配对象的帮助。

dsrm /? - 删除对象的帮助。

描述： 把用户添加到目录。

语法： dsadd user <UserDN> [-samid <SAMName>] [-upn <UPN>] [-fn <FirstName>]
[-mi <Initial>] [-ln <LastName>] [-display <DisplayName>]
[-empid <EmployeeID>] [-pwd {<Password> | *}] [-desc <Description>]
[-memberof <Group ...>] [-office <Office>] [-tel <Phone#>]
[-email <Email>] [-hometel <HomePhone#>] [-pager <Pager#>]
[-mobile <CellPhone#>] [-fax <Fax#>] [-iptel <IPPhone#>]
[-webpg <WebPage>] [-title <Title>] [-dept <Department>]
[-company <Company>] [-mgr <Manager>] [-hmdir <HomeDir>]
[-hmdrv <DriveLtr:>] [-profile <ProfilePath>] [-loscr <ScriptPath>]
[-mustchpwd {yes | no}] [-canchpwd {yes | no}]
[-reversiblepwd {yes | no}] [-pwdneverexpires {yes | no}]
[-acctexpires <NumDays>] [-disabled {yes | no}]
[{-s <Server> | -d <Domain>}] [-u <UserName>]
[-p {<Password> | *}] [-q] [{-uc | -uco | -uci}]

参数：

值	描述
<UserDN>	需要项。要添加的用户可分辨名称(DN)。如果目标对象被省略, 将从标准输入(stdin)中读取。
-samid <SAMName>	设置用户的 SAM 帐户名为 <SAMName>。如果没有指定, dsadd 会使用 <UserDN> 公用名(CN)里的前 20 个字符创建 SAM 帐户名。
-upn <UPN>	设置 upn 值为 <UPN>。
-fn <FirstName>	设置用户名为 <FirstName>。
-mi <Initial>	设置用户中间名首字母为 <Initial>。
-ln <LastName>	设置用户姓为 <LastName>。
-display <DisplayName>	设置用户显示名为 <DisplayName>。
-empid <EmployeeID>	设置用户雇员 ID 为 <EmployeeID>。
-pwd {<Password> *}	设置用户密码为 <Password>。如果是 *, 会提示您输入密码。
-desc <Description>	设置用户描述为 <Description>。
-memberof <Group ...>	把用户设置成一个或多个组成员 <Group ...>
-office <Office>	设置用户办公室位置为 <Office>。
-tel <Phone#>	设置用户电话号码为 <Phone#>。
-email <Email>	设置用户电子邮件地址为 <Email>。
-hometel <HomePhone#>	设置用户家庭电话号码为 <HomePhone#>。
-pager <Pager#>	设置用户寻呼机号码为 <Pager#>。
-mobile <CellPhone#>	设置用户移动电话号码为 <CellPhone#>。
-fax <Fax#>	设置用户传真号码为 <Fax#>。
-iptel <IPPhone#>	设置用户的 IP 电话号码为 <IPPhone#>。
-webpg <WebPage>	设置用户的网页 URL 为 <WebPage>。
-title <Title>	设置用户的职务为 <Title>。
-dept <Department>	设置用户的部门为 <Department>。
-company <Company>	设置用户的公司信息为 <Company>。
-mgr <Manager>	设置用户的经理为 <Manager> (格式是 DN)。
-hmdir <HomeDir>	设置用户主目录为 <HomeDir>。如果它是 UNC 路径, 一个将映射到此路径的驱动器号必须也要用 -hmdrv 指定。
-hmdrv <DriveLtr:>	设置用户主驱动器号为 <DriveLtr:>
-profile <ProfilePath>	设置用户的配置文件路径为 <ProfilePath>。
-loscr <ScriptPath>	设置用户的登录脚本路径为 <ScriptPath>。

-mustchpwd {yes | no} 用户在下次登录时是否更改密码。默认值: no。

-canchpwd {yes | no} 用户是否可以更改密码。如果 -mustchpwd 是 "yes", 它应该是 "yes"。默认值: yes。

-reversiblepwd {yes | no} 是否使用可逆的加密保存密码。默认值: no。

-pwdneverexpires {yes | no} 用户密码是否永远不过期。默认值: no。

-acctexpires <NumDays> 设置用户帐户从今天起在 <NumDays> 天内过期。0 值意味着今天结束后帐户就过期; 正数值意味着帐户在未来过期; 负数意味着该帐户已经过期并将过期日期设置在过去; 字符串值 "never" 意味着该帐户永远不过期。

-disabled {yes | no} 用户帐户是否禁用。默认值: no。

{-s <Server> | -d <Domain>}
 -s <Server> 使用名称 <Server> 连接到域控制器 (DC)。
 -d <Domain> 在域 <Domain> 连接一个 DC。
 默认值: 在登录域的 DC。

-u <UserName> 作为 <UserName> 连接。默认值: 该登录的用户。用户名可以是: 用户名, 域\用户名或用户主体名称 (UPN)。

-p {<Password> | *} 用户的密码 <UserName>。如果已输入 *, 会提示您输入密码。

-q 安静模式: 不在标准输出显示任何输出。

{-uc | -uco | -uci} -uc 指定从管道的输入或至管道输出用 Unicode 格式。
 -uco 指定至管道或文件的输出用 Unicode 格式。
 -uci 指定从管道或文件的输入用 Unicode 格式。

备注:

如果您在命令提示没有提供目标对象, 会从标准输入 (stdin) 获取目标对象。可以从键盘、重定向文件或从另一个命令的管道输出接受 Stdin 数据。要从键盘或在重定向文件中标记 stdin 数据的结束, 请使用 Control+Z 表示文件结束 (EOF)。

如果您提供的值包含空格, 请在文字两边使用引号 (例如, "CN=John Smith, CN=Users, DC=microsoft, DC=com")。如果您输入了多个值, 这些值必须用空格隔开 (例如, 一个可分辨名称列表)。

特殊令牌 `$username$` (不区分大小写)可以用来把 SAM 帐户名放在一个参数的值。
例如, 如果目标用户 DN 是 `CN=Jane Doe, CN=users, CN=microsoft, CN=com`,
SAM 帐户名属性是 `"janed"`, `-hmdir` 参数能有以下替换值:

```
-hmdir \users\$username$\home
```

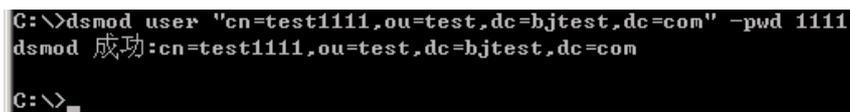
`-hmdir` 参数的值被修改为以下值:

```
-hmdir \users\janed\home
```

描述已经很详细了, 比如我需要给用户 `test1111` 设置密码, 则输入以下命令:

```
dsmod user "cn=test1111,ou=test,dc=bjtest,dc=com" -pwd 1111
```

含义就是修改 `bjtest.com` 下 `test` 组织单元内的 `test1111` 用户的密码为 `1111`



```
C:\>dsmod user "cn=test1111,ou=test,dc=bjtest,dc=com" -pwd 1111
dsmod 成功:cn=test1111,ou=test,dc=bjtest,dc=com
C:\>
```

成功。修改, 新增, 查询用户都可以这样操作。这个命令相比 `csvde` 的优点可以通过下面的例子来体现:

比如我现在拿到一份名单

张三, 电话 12345

李四, 电话 23456

王五, 电话 34567

需要在 AD 中建立 3 个用户, 写入电话, 并把密码初始值设置为和自己电话一样的数值, 注意也就是每个人的密码还是不同的, 并要写 AD 的名字和显示名 (`displayname` 这个属性, 做过 OCS 的应该知道这个名字将会是 OC 用户的显示名), 将 3 个用户放入 `test` 组织单元中。

那么下面把文本修改成如下的内容, 下面文本还是用逗号分割:

zhangsan, 张三, 12345

lisi, 李四, 23456

wangwu, 王五, 34567

可以运行下面命令

```
dsadd user "cn=zhangsan,ou=test,dc=bjtest,dc=com" -display 张三 -pwd 12345 -tel 12345
```

```
C:\>dsmod user "cn=test1111,ou=test,dc=bjtest,dc=com" -pwd 1111
dsmod 成功:cn=test1111,ou=test,dc=bjtest,dc=com

C:\>dsadd user "cn=zhangsan,ou=test,dc=bjtest,dc=com" -display 张三 -pwd 12345 -tel 12345
dsadd 成功:cn=zhangsan,ou=test,dc=bjtest,dc=com

C:\>
```

同样操作李四，王五，如果用户多了，可以用 Excel 编辑这个命令序列，存成.bat 文件执行，但还有其他的办法，更适合大量用户时进行操作。



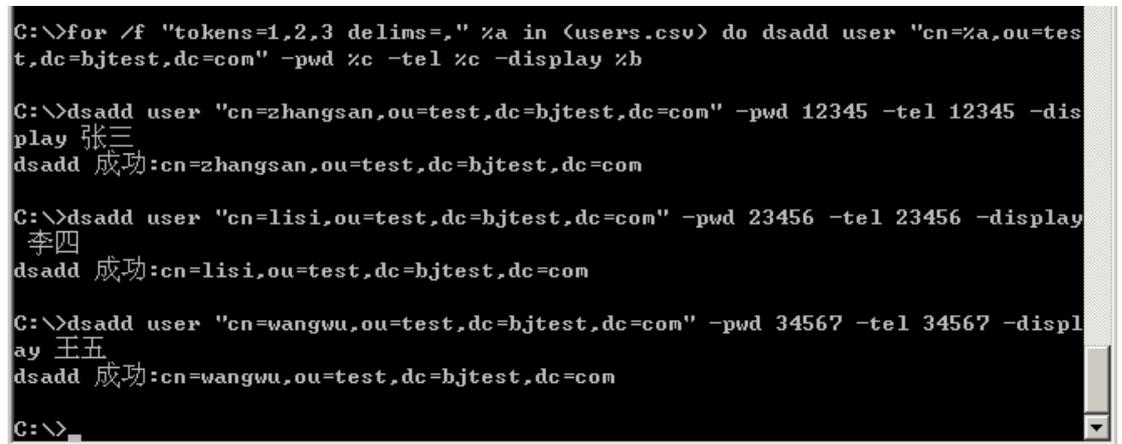
```
users.csv - 记事本
文件(F) 编辑(E) 格式(O) 查看(V) 帮...
zhangsan,张三,12345
lisi,李四,23456
wangwu,王五,34567
```

文本内容如上面所示

输入下面命令（为了显示在一行内换了小字体，应用请复制）

```
for /f "tokens=1,2,3 delims=," %a in (users.csv) do dsadd user "cn=%a,ou=test,dc=bjtest,dc=com" -pwd %c -tel %c -display %b
```

结果如图



```
C:\>for /f "tokens=1,2,3 delims=," %a in (users.csv) do dsadd user "cn=%a,ou=test,dc=bjtest,dc=com" -pwd %c -tel %c -display %b

C:\>dsadd user "cn=zhangsan,ou=test,dc=bjtest,dc=com" -pwd 12345 -tel 12345 -display 张三
dsadd 成功:cn=zhangsan,ou=test,dc=bjtest,dc=com

C:\>dsadd user "cn=lisi,ou=test,dc=bjtest,dc=com" -pwd 23456 -tel 23456 -display 李四
dsadd 成功:cn=lisi,ou=test,dc=bjtest,dc=com

C:\>dsadd user "cn>wangwu,ou=test,dc=bjtest,dc=com" -pwd 34567 -tel 34567 -display 王五
dsadd 成功:cn>wangwu,ou=test,dc=bjtest,dc=com

C:\>
```

下面将讲解的是

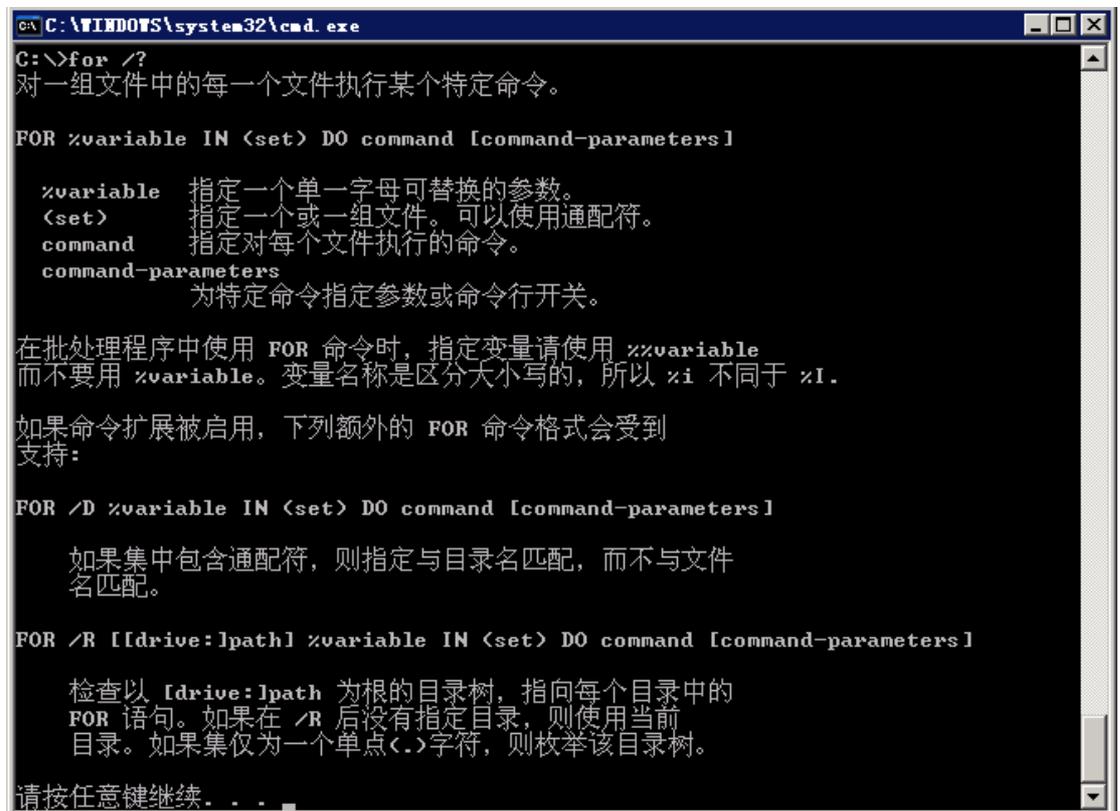
利用 **for** 循环语句实现批量操作

解释该命令

token=1, 2, 3 每行取第 1, 2, 3

FOR 语句是个标准的批处理命令

在命令行下运行 FOR /? 同样会得到很多信息，又耐心的可以逐页看完



```
C:\WINDOWS\system32\cmd.exe
C:\>for /?
对一组文件中的每一个文件执行某个特定命令。

FOR %variable IN (set) DO command [command-parameters]

%variable  指定一个单一字母可替换的参数。
(set)      指定一个或一组文件。可以使用通配符。
command    指定对每个文件执行的命令。
command-parameters
            为特定命令指定参数或命令行开关。

在批处理程序中使用 FOR 命令时，指定变量请使用 %%variable
而不要用 %variable。变量名称是区分大小写的，所以 %i 不同于 %I。

如果命令扩展被启用，下列额外的 FOR 命令格式会受到支持：

FOR /D %variable IN (set) DO command [command-parameters]

    如果集中包含通配符，则指定与目录名匹配，而不与文件名匹配。

FOR /R [[drive:]lpath] %variable IN (set) DO command [command-parameters]

    检查以 [drive:]lpath 为根的目录树，指向每个目录中的
    FOR 语句。如果在 /R 后没有指定目录，则使用当前
    目录。如果集仅为一个单点(<.)字符，则枚举该目录树。

请按任意键继续. . .
```

这里仅介绍配合 DSADD 的应用

FOR 配合文本文件使用可以这样理解

比如一个句子

我让 () 去学校，拿一本 () 书回来，送给 () 老师

而文本中，就是

张三，历史，李

李四，政治，何

王五，生物，张

于是就填出 3 个句子

我让（张三）去学校，拿一本（历史）书回来，送给（李）老师

我让（李四）去学校，拿一本（政治）书回来，送给（何）老师

我让（王五）去学校，拿一本（生物）书回来，送给（张）老师

写法记住一个固定的就行了，忘记了去 **google** 一下也很多。这里就根据前面举的例子来讲解

```
for /f "tokens=1,2,3 delims=," %a in (users.csv) do dsadd user "cn=%a,ou=test,dc=bjtest,dc=com" -pwd %c -tel %c -display %b
```

tokens=1, 2, 3 就是每行读取地 1, 2, 3 个值

delims=, 就是认为逗号是分开每个值的标识

%a 就是从 **a** 开始命名，第一个是**%a**,第二个值是**%b**，第三个值是**%c**

依次类推，最多就 **a-z 26** 个，好像也可以用**%%a**，依次类推。

括号内的 **users.csv** 就是对应读取的文本文件了，

do 后面就是要填入的“句型”了

也就是说，把文本的内容每行以逗号分开读取 3 个值

```
重复执行 dsadd user "cn=第一个值,ou=test,dc=bjtest,dc=com" -pwd 第三个值 -te  
l 第三个值-display 第二值
```

恩，就这样，更多内容请自行搜索吧，不给链接了。可以搜索关键词：

for delims 或者 for /f

使用脚本添加某一特定属性

声明：此部分内容属于分享内容，代码为论坛网友 **ghjconan**（如果记错了名字抱歉）提供，在此再次表示感谢。

这里讲这个内容也是遇到一个实际情况，需要批量填写一批 OCS 的线路 URI，于是查到一个类似的脚本在开发人员指导下修改后操作成功。

下面分享这个案例，希望有所借鉴

需求是像域用户填写某个属性的内容，经查询这个属性的名称叫

msRTCSIP-Line

于是写了一个文本，内容大概如下

```
test1, tel: +861012345678
```

```
test2, tel: +861012345678
```

test3, tel: +861012345678

将上述内容保存为文本文件 Userinfo.csv

将下面脚本保存成 A.vbs 并和 Userinfo.csv 放在同一路径下，在 DC 上运行。

```
Dim objInfo
Const ADS_SCOPE_SUBTREE = 2

Set fso = CreateObject("Scripting.FileSystemObject")
Set objFile = fso.OpenTextFile("Userinfo.csv")

Set objConnection = CreateObject("ADODB.Connection")
Set objCommand = CreateObject("ADODB.Command")
objConnection.Provider = "ADsDSOObject"
objConnection.Open "Active Directory Provider"
Set objCommand.ActiveConnection = objConnection
objCommand.Properties("Page Size") = 1000
objCommand.Properties("Searchscope") = ADS_SCOPE_SUBTREE

Do
On Error Resume Next

    objInfo = Split(objfile.ReadLine,",")
    strDN = FindUserDN(objInfo(0))
    Set objUser = GetObject("LDAP://" & strDN)
    objUser.Put "msRTCSIP-Line",objInfo(1)
    objUser.SetInfo
```

```

If Err.Number <> 0 Then
    WScript.Echo "Error:" & objInfo(0) & "," & objInfo(1)
End If
strDN = ""
Loop Until objfile.AtEndOfStream = True

objfile.Close

WScript.Echo "Done!"

Function FindUserDN(strUserDisplayName)

    objCommand.CommandText = _
        "SELECT distinguishedName FROM 'LDAP://OU=OCSuser,DC=
C=test,DC=com,DC=cn' WHERE objectCategory='user' " & _
        "AND displayname = " & strUserDisplayName & ""

    Set objRecordSet = objCommand.Execute

    objRecordSet.MoveFirst

    Do Until objRecordSet.EOF

        arrDN = objRecordSet.Fields("distinguishedName").Value

```

```
If IsArray(arrDN) = True Then
```

```
    WScript.Echo strUserDisplayName
```

```
    FindUserDN = arrDN(0)
```

```
Else
```

```
    FindUserDN = arrDN
```

```
End if
```

```
objRecordSet.MoveNext
```

```
Loop
```

```
If Err.Number <> 0 Then
```

```
    WScript.Echo Err.Description & ":" & strUserDisplayNam
```

```
e
```

```
    Err.Clear
```

```
End If
```

```
End Function
```

说明：

msRTCSIP-Line : 是要修改的属性，即线路 URI

LDAP://OU=OCSuser,DC=uc,DC=cdb,DC=com,DC=cn: 是要查询用户的
AD 路径

displayname : 是判断用户的标识，也就是查询到 displayname 是 test1
的用户，填写后面的值进入线路 URL

Userinfo.csv: 是导入的文件

此脚本须和 **Userinfo.csv** 在同一路径下，双击此脚本保存的.vbs 文件即可。

也可以在命令行模式下输入 `cscript a.vbs`

未提示错误，只提示 **done!** 就是修改成功了。