**Loadrunner 录制登录操作**

脚本业务流：访问Discuz论坛首页——登录论坛——退出论坛。本次使用LoadRunner11版本。

一、录制脚本注意

    1、确保Discuz论坛能在服务器运行正常。

    2、录制前先试访问Discuz论坛注册一个测试用户。

    3、选择录制协议，Discuz论坛是一个B/S架构的web论坛，使用常用协议“web（HTTP/HTML）”录制。

    4、录制选项的设置，特别注意录制时的HTTP/HTML级别的选择是“基于HTML的脚本”还是“基于URL的脚本”，这里选择“基于HTML的脚本”。

    5、录制完脚本后，先保存脚本，再回放一遍。

二、优化脚本

    使用LoadRunner录制后的脚本，通常都是需要优化完善脚本。这里只展示action中的登录部分，优化完善后的脚本如下：

Action\_login()  
{

 lr\_think\_time(12);

 //添加登录时的集合点Login  
 lr\_rendezvous("login");

 //设置开始事务  
 lr\_start\_transaction("login");  
   
 web\_submit\_data("member.php",  
  "Action=http://127.0.0.1/Discuz/upload/member.php?mod=logging&action=login&loginsubmit=yes&infloat=yes&lssubmit=yes&inajax=1",  
  "Method=POST",  
  "RecContentType=text/xml",  
  "Referer=http://127.0.0.1/Discuz/upload/forum.php",  
  "Snapshot=t2.inf",  
  "Mode=HTML",  
  ITEMDATA,  
  "Name=fastloginfield", "Value=username", ENDITEM,  
  "Name=username", "Value={username}", ENDITEM,  
  "Name=password", "Value={password}", ENDITEM,  
  "Name=quickforward", "Value=yes", ENDITEM,  
  "Name=handlekey", "Value=ls", ENDITEM,  
  LAST);

 //输出参数化日志信息  
 lr\_output\_message("登录用户名：%s", lr\_eval\_string("{username}"));  
 lr\_output\_message("登录密码：%s", lr\_eval\_string("{password}"));

 web\_add\_cookie("mlMh\_2132\_checkfollow=1; DOMAIN=127.0.0.1");

 web\_add\_cookie("mlMh\_2132\_lastact=1399082672%09forum.php%09; DOMAIN=127.0.0.1");

 web\_add\_cookie("mlMh\_2132\_checkpm=1; DOMAIN=127.0.0.1");

 //设置关联 Correlation comment - Do not change!Original value='935343df' Name ='CorrelationParameter\_1'  
 web\_reg\_save\_param\_ex(  
  "ParamName=CorrelationParameter\_1",  
  "LB=formhash=",  
  "RB=\">退出",  
  SEARCH\_FILTERS,  
  "Scope=Body",  
  "RequestUrl=\*/forum.php\*",  
  LAST);

 //添加文本检查点，检查登录用户yuxiuquan是否登录  
 web\_reg\_find("Text=yuxiuquan",  
  LAST);

 web\_url("forum.php\_2",  
  "URL=http://127.0.0.1/Discuz/upload/forum.php",  
  "Resource=0",  
  "RecContentType=text/html",  
  "Referer=",  
  "Snapshot=t3.inf",  
  "Mode=HTML",  
  EXTRARES,  
  "URL=static/image/common/background.png", ENDITEM,  
  "URL=static/image/common/nv.png", ENDITEM,  
  "URL=static/image/common/search.png", ENDITEM,  
  "URL=static/image/common/arrwd.gif", ENDITEM,  
  "URL=static/image/feed/friend\_b.png", ENDITEM,  
  "URL=static/image/feed/favorite\_b.png", ENDITEM,  
  "URL=static/image/feed/thread\_b.png", ENDITEM,  
  "URL=static/image/feed/magic\_b.png", ENDITEM,  
  "URL=static/image/feed/medal\_b.png", ENDITEM,  
  "URL=static/image/common/tip\_bottom.png", ENDITEM,  
  "URL=static/image/common/style\_switch.png", ENDITEM,  
  "URL=static/image/common/user\_online.gif", ENDITEM,  
  "URL=static/image/common/qmenu.png", ENDITEM,  
  "URL=static/image/common/nv\_a.png", ENDITEM,  
  "URL=static/image/common/pt\_item.png", ENDITEM,  
  "URL=static/image/common/chart.png", ENDITEM,  
  "URL=static/image/feed/task\_b.png", ENDITEM,  
  "URL=static/image/common/scrolltop.png", ENDITEM,  
  "URL=static/image/common/titlebg.png", ENDITEM,  
  "URL=http://cp.discuz.qq.com/tips/get?rand=310&s\_id=&product\_version=X2.5&product\_release=20120413&fix\_bug=25000000&is\_founder=&s\_url=http%3A%2F%2F127.0.0.1%2FDiscuz%2Fupload%2F&last\_send\_time=&ts=1399082672&sig=&admin\_id=0&group\_id=10&open\_id=&uid=2&callback=discuzTipsCallback", ENDITEM,  
  LAST);

 //设置结束事务  
 lr\_end\_transaction("login", LR\_AUTO);

return 0;  
}

1、事务设置，开始事务和结束事务的名称必须一致，事务的设置必须成对出现，即有开始事务也要有结束事务，开始事务应该放在脚本开始的位置，结束事务放在脚本结束的位置，但不要将思考时间（lr\_think\_time函数）放在事务开始和事务结束之间。其实，设置事务，可以在“Run\_time Settings”里设置的，可以设置成“将每个操作定义为一个事务”，也可以设置为“将每个步骤定义为一个事务”，根据需要设置。我这里是自己手动在脚本设置的。

使用函数： lr\_start\_transaction("事务名称");

lr\_end\_transaction("事务名称",LR\_AUTO);

2、检查点设置，LR的检查点分为文本检查点和图像检查点，添加检查点的方法很简单，在LR的树形视图中，打开视图登录页面，选择需要检查的内容（文本或者图像），鼠标右键选择“添加文本检查点（web\_reg\_find()）”即可，若要手动添加检查点，则需要把检查点设置在返回页面的之前。最后，在脚本中添加了检查点，要在Vuser->Run-Tme Settings中更改下设置，勾选Enable Image and text check。

使用函数：web\_find() 或 web\_reg\_find() 或 web\_image\_check()，下面分别说明

（1）web\_find()

参数举例：web\_find("web\_find","RighOf=a","LeftOf=b","What=name",LAST);

参数解释："web\_find"定义该查找函数的名称；“LeftOf”和“RighOf=”用来定义查找字符的左右边界；“What=”定义查找内容。

（2）web\_reg\_find()

参数举例：web\_reg\_find("Search=Body","SaveCount=ddd","Test=aaa",LAST);

参数解释： Search用来定义查找范围，SaveCount定义查找计数变量名称，该参数可以记录在缓存中查找内容出现的次数，可以使用该值，来判断要查找的内容是否被找到。

（3）web\_image\_check()

参数说明：web\_image\_check("web\_image\_check","Alt=","Src=",LAST);;

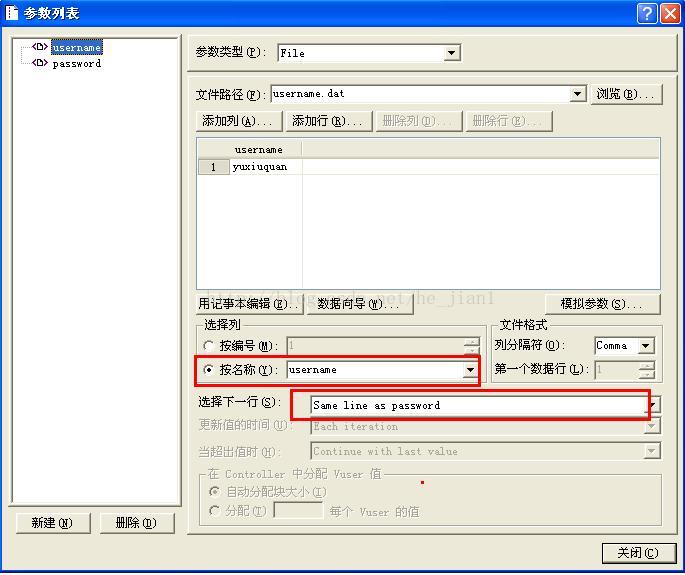
参数解释：“Alt”和“Src”的值直接取该图片在网页源代码中相应参数的值。

3、参数化设置，LR录制登录的脚本，一般需要参数化脚本中的登录用户和对应的密码，以便更加真实地模仿系统的业务操作。参数化的设置，点击工具栏http://images.cnitblog.com/i/337104/201405/031539081116739.jpg图标打开参数列表，或者点击菜单“Vuser--参数列表”，或使用快捷方式“Ctrl+L”，点击新建，定义参数名称username和password,创建表并输入内容

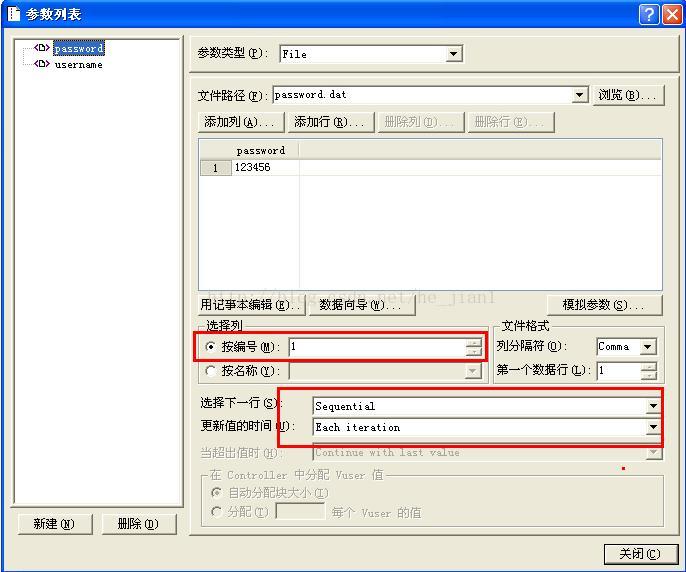
后，这里特别需要注意设置的是参数名username和password相对应，通过参数列表中的数据方式和更新值方式来设置，username参数要与password参数对应设

置正确，脚本回放时才能模拟用户登录成功，如图：

（1）、username参数名设置



（2）password参数名设置



（3）、参数化设置中，“Select next row”和“Update value on”这两个属性组合，产生九种组合方式，每种方式都不一样，参数化的结果也不一样，下面是参数取值方式说明

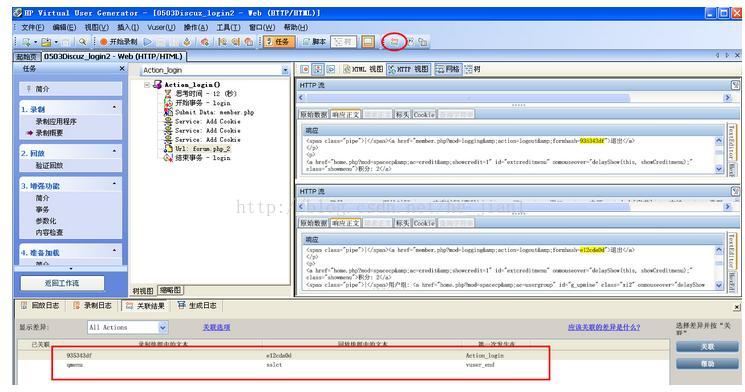
Select next row：  
Sequential：每个VU按照顺序读取。每一个虚拟用户都会按照相同的顺序读取。  
Random：每个VU随机读取一个。  
Unique：每个VU顺序取唯一的值。注意：使用该类型必须注意数据池中数据充足。  
Same Line As 某个参数（比如Name）：和前面定义的参数Name 取同行的记录。通常用在有关联性的数据上面。  
Update value on：  
Once 在所有的反复中都使用同一个值  只取一次  
each iteration 每次反复都要取新值  
each occurrence 只要发现该参数就要重新取值，也就是如果一个action中有多个该参数，每遇到一个就重新取一个值。

（4）、在脚本中，选中需要参数化的脚本，鼠标右键“使用现有参数”选择对应要参数化的参数名即可。

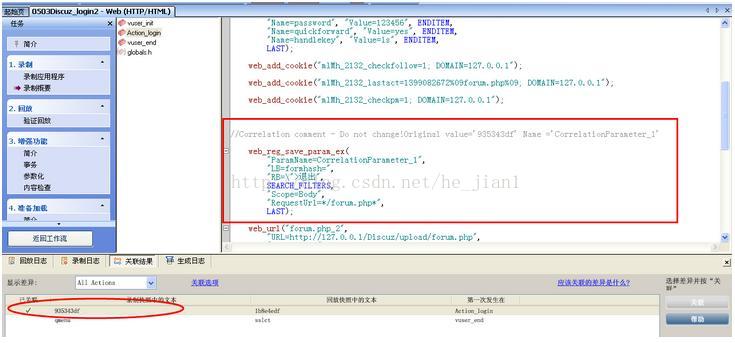
（5）、在脚本中输出参数化日志信息

//输出参数化日志信息  
lr\_output\_message("登录用户名：%s", lr\_eval\_string("{username}"));  
lr\_output\_message("登录密码：%s", lr\_eval\_string("{password}"));

4、关联设置，由于我的LR工具在录制的时设置了关联规则，在脚本中可以使用自动扫描脚本中关联值快捷键Ctrl+F8，扫描结果切换树形视图，可以看到如图



点击“关联”，返回到脚本页面，如图



再次回放脚本，在菜单“视图--测试结果”中可以看到关联函数执行成功。

5、集合点设置，集合点的设置在菜单“插入--集合”即可，集合点名称需顾名思义，看到就能理解。另外需要注意，集合点要放在开始事务的前面，思考时间（lr\_think\_time函数）的后面。

使用函数：lr\_rendezvous("集合点名称");