**Loadrunner使用web\_custom\_request函数自定义请求**

# 一、删除指定机票

**1、准备工作**

本文以loadrunner自带的飞机订票系统实现自定义请求删除机票操作

首先用loadrunner将Itinerary中的机票信息获取出来：



至于怎么操作在这里就不讲述。

脚本内容：

*web\_url("welcome.pl",*

*"URL=http://127.0.0.1:1080/WebTours/welcome.pl?page=itinerary",*

*"TargetFrame=",*

*"Resource=0",*

*"RecContentType=text/html",*

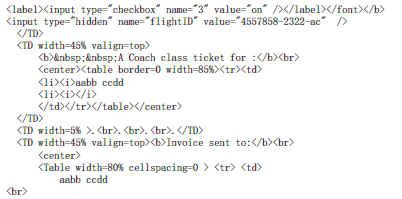
*"Referer=http://127.0.0.1:1080/WebTours/nav.pl?page=menu&in=home",*

*"Snapshot=t3.inf",*

*"Mode=HTML",*

*LAST);*

从服务器返回的response中可以查看到返回的机票信息：



**2、自定义删除前两张机票**

通过将删除操作的请求数据的截断，可以看到请求的body中内容如下：1=on&flightID=7787-789-ac&flightID=128007430-1587-ac&flightID=4557158-2329-ac&flightID=4557858-3091-ac&flightID=128026019-3886-ac&removeFlights.x=61&removeFlights.y=6&.cgifields=4&.cgifields=1&.cgifields=3&.cgifields=2&.cgifields=5

其中：

1=on表示：第一个checkbox是被选中；flightID=7787-789-ac表示：第一条记录对应的flightid。

因此要删除前两张机票，则需要将前两条记录的flightID关联出来。

//关联第一条记录的flightID

*web\_reg\_save\_param("fID1",*

*"LB=INPUT TYPE=\"hidden\" NAME=\"flightID\" VALUE=\"",*

*"RB=\"",*

*"ORD=1",*

*"SEARCH=BODY",*

*LAST)*

//关联第二条记录的flightID

*web\_reg\_save\_param("fID2",*

*"LB=INPUT TYPE=\"hidden\" NAME=\"flightID\" VALUE=\"",*

*"RB=\"",*

*"ORD=2",*

*"SEARCH=BODY",*

*LAST)*

**3、源代码：**

定义一个字符串保存body的内容：*char test\_body[1000]*

接下来组合body内容(主要运用C的拼接命令：strcat)：

*strcpy(creq, "Body=1=on&flightID=");*

*strcat(creq, lr\_eval\_string("{fID1}"));*

*strcat(creq, "&2=on&flightID=");*

*strcat(creq, lr\_eval\_string("{fID2}"));*

*strcat(creq, "&.cgifields=1&.cgifields=2");*

*strcat(creq, "&removeFlights.x=116&removeFlights.y=8");*

请求的body内容已经组合完毕，下面编写请求：

*web\_custom\_request("itinerary.pl",*

*"Method=POST",*

*"URL=http://localhost/MercuryWebTours/itinerary.pl",*

*"RecContentType=text/xml",*

*creq,*

*"Snapshot=t4.inf",*

*LAST)*

**完整源代码为：**

*char creq[500];*

*web\_reg\_save\_param("fID1","LB=INPUT TYPE=\"hidden\" NAME=\"flightID\" VALUE=\"", "RB=\"", "ORD=1","SEARCH=BODY", LAST);*

*web\_reg\_save\_param("fID2","LB=INPUT TYPE=\"hidden\" NAME=\"flightID\" VALUE=\"", "RB=\"","ORD=2", "SEARCH=BODY", LAST);*

*web\_url("welcome.pl",*

*"URL=http://localhost/MercuryWebTours/welcome.pl?page=itinerary",*

*"Resource=0",*

*"RecContentType=text/html", "Referer=http://localhost/MercuryWebTours/nav.pl?page=menu&in=home",*

*"Snapshot=t3.inf",*

*"Mode=HTML",*

*EXTRARES,*

*"URL=images/in\_itinerary.gif", "Referer=http://localhost/MercuryWebTours/nav.pl?page=menu&in=itinerary", ENDITEM, "URL=images/home.gif","Referer=http://localhost/MercuryWebTours/nav.pl?page=menu&in=itinerary", ENDITEM,LAST);*

*strcpy(creq, "Body=1=on&flightID=");*

*strcat(creq, lr\_eval\_string("{fID1}"));*

*strcat(creq, "&2=on&flightID=");*

*strcat(creq, lr\_eval\_string("{fID2}"));*

*strcat(creq, "&.cgifields=1&.cgifields=2");*

*strcat(creq, "&removeFlights.x=116&removeFlights.y=8");*

*lr\_output\_message(creq);*

*web\_custom\_request("itinerary.pl",*

*"Method=POST",*

*"URL=http://localhost/MercuryWebTours/itinerary.pl",*

*"RecContentType=text/xml",*

*creq,*

*"Snapshot=t4.inf",*

*LAST);*

# 二、随机删除一张机票

删除机票的原理在上篇中已经表述过了，接下来是要实现更加复杂的，随机删除操作。

主要内容就是获取机票总数、产生要删除的机票号、根据要删除的机票号获取机票ID。

//获取机票总数

*int web\_reg\_save\_param("flight\_count",*

*"LB= <b>A total of ",*

*"RB= scheduled flights."*

*LAST);*

//产生要删除的随机数

*int n=rand()%n+1;（1~n范围内的随机数）*

//根据机票号获取关联的机票ID

*web\_reg\_save\_param("fID1",*

*"LB=INPUT TYPE=\"hidden\" NAME=\"flightID\" VALUE=\"",*

*"RB=\"", "ORD=ALL",*

*"SEARCH=BODY",*

*LAST);*

1、分析过程

通过抓包查看删除机票的请求内容

1=on&flightID=128007430-818-ac&flightID=4557158-1560-ac&3=on&flightID=4557858-2322-ac&flightID=128026019-3116-ac&removeFlights.x=57&removeFlights.y=7&.cgifields=4&.cgifields=1&.cgifields=3&.cgifields=2

发现有几点需要注意。当要删除的机票为第一张时，开始的内容应为：1=on；当删除的不为第一张时为：&n=on；

如果要删除的为第一张机票，紧跟1=on后的字符串为&flightID=xxxxx；当要删除的不为第一张机票时，开头的内容应为：flightID=xxxxxx。

获取相应机票的机票ID使用sprintf函数:

*sprintf(arrayParamName, "{flightID\_%d}", x)*

*注：先取得机票的序号，然后根据序号获取对应机票的ID，拼接后保存到arrayParamName数组中。*

*完整源代码：*

*Action()*

*{*

*char flight\_id[50]; //存储机票列表的记录值*

*char cgifields[50]; //存储机票的cgifields值*

*int flight\_count; //存储机票列表的记录数*

*int x=0; //序号，用于循环操作*

*char \*strreq; //请求串*

*int n=0; //存储生成的随机数*

*char temp[3]; //存储随机数的字符串格式*

*// 保存机票ID*

*web\_reg\_save\_param("flightID",*

*"LB=name=\"flightID\" value=\"",*

*"RB=\" />",*

*"Ord=ALL",*

*"Search=All",*

*LAST);*

*//保存cgifields*

*web\_reg\_save\_param("cgifields",*

*"LB=name=\".cgifields\" value=\"",*

*"RB=\" />",*

*"Ord=ALL",*

*"Search=All",*

*LAST);*

*//保存机票总数*

*web\_reg\_save\_param\_param("flightID\_count",*

*"LB=A total of ",*

*"RB= scheduled flights",*

*LAST);*

*//拼接body的开头 BODY=*

*strcat(streeq,"BODY=");*

*//将获取到的机票数字字符转化成整数*

*flight\_count=atoi(lr\_evalstrint("{flightID\_count}"));*

*//根据机票总数产生要删除的机票序号:1~flight\_count*

*n=rand()%flight\_count+1;*

*//对机票进行循环*

*for(x=1;x<=flight\_count;x++){*

*//如果是要删除的是第一张机票，则开头应为1=on，将内容写到strreq中*

*if(x==1&&n==1){*

*itoa(n,temp,10);*

*strcat(strreq,temp);*

*strcat(strreq,"=on");*

*}*

*//如果要删除的不是第一张机票，则要删除的机票处内容为&n=on*

*else{*

*if(x==n){*

*itoa(n,temp,10);*

*strcat(strreq,"&");*

*strcat(strreq,temp);*

*strcat(strreq,"=on");*

*}*

*}*

*//如果要删除的不是第一张机票，则请求开头内容为 flightID=xxxxxxxx*

*if(x==1&n!=1){*

*strcat(strreq,"flightID=");*

*}*

*//如果要删除的为第一张机票，则紧跟着n=on后面的内容为&flightID=xxxxxxxx*

*else{*

*strcat(strreq,"&flightID=");*

*}*

*//将flightID字符串存储到flight\_id中*

*sprintf(flight\_id,"{flightID\_%d}",x);*

*//获取相应位置的flight\_id*

*strcat(strreq,lr\_eval\_string(flight\_id));*

*}*

*strcat(strreq,"&removeFlights.x=57&removeFlights.y=7");*

*//拼接&.cgifields内容，有多少张机票就有多少个值*

*for(x=1;x<=flight\_count;x++){*

*strcat(strreq,"&.cgifields=");*

*sprintf(cgifields，"{cgifields\_%d}",x);*

*strcat(strreq,lr\_eval\_string(cgifields));*

*}*

*//采用web\_custom\_request函数发送请求*

*web\_custom\_request("del\_flights",*

*"URL=URL=http://127.0.0.1:1080/WebTours/itinerary.pl",*

*"Method=POST";*

*strreq,*

*LAST);*

*return 0;*

*}*