

报文模拟测试工具用户手册

目 录

目 录.....	2
工具描述.....	4
功能特点.....	5
适用范围.....	6
文件目录结构说明.....	6
case 目录.....	7
file 目录.....	7
ini 目录.....	7
log 目录.....	7
report 目录.....	7
主界面.....	8
使用说明.....	8
增加、修改删除案例.....	8
密钥配置界面.....	9
通讯参数配置界面.....	9
修改案例界面.....	11
文本模式修改界面.....	12
设置案例集界面.....	12
报文属性设置.....	13
常量设置界面.....	14
发送案例.....	15
发送次数.....	15
清空.....	15
终止发送.....	15
清空日志.....	15
文件格式说明.....	16
config.ini 例子.....	16
Case.ini 例子.....	16

iso.ini 例子.....	17
交易文件的配置.....	17
正交易配置.....	19
反交易配置.....	20
可选配置.....	21
服务应答配置.....	22
案例设置.....	22
非 8583 配置.....	22
启动服务.....	25
返回应答.....	26
8583 配置.....	28
启动服务.....	28
返回应答.....	29
支持函数列表.....	30
String.....	30
time.....	30
Req.....	30
Rev.....	31
Def.....	31
Tlv.....	31
Tlvasc.....	31
服务端函数列表.....	32
Request.....	32
Iso.....	32
Sep.....	32
常见问题.....	33
更新计划.....	33
修改记录.....	33

工具描述

系统具有友好界面，方便直观便于配置，相关案例数据存储采用文本文件进行保存，便于开发测试人员修改、共享测试案例。

本软件可以模拟不同类型的交易报文，可以对交易测试案例进行统一管理，并可以进行简单时间统计和成功率统计。

使用本软件可以减轻传统测试过程中的修改-编译-测试-的循环等待时间，在测试过程中可以根据需要随时更改报文内容。

本软件支持任意格式的报文，可以模拟不同格式的报文，如定长，变长，XML,8583 等报文。每个域的内容可以是常量，也可以支持约定的表达式。

本软件可以根据需要设置对应答相关域进行合法性检查，可以校验应答报文和请求报文的匹配关系，可以校验域的长度，校验域的内容等。

本软件支持 MAC 的生成、校验以及 PIN 加密处理，同时可以根据需要调整是否需要进行 MAC 和 PIN 加密。

不软件也可以支持服务端功能，模拟服务端应答处理，根据根据不同的报文场景，返回不同的应答报文。

支持自定义扩展函数。

支持动态库调用，根据需要进行自行扩展。

本软件运行程序无需安装，只需将相关程序和测试案例文件拷贝到相应的文件夹下即可执行。

本系统目前局限性如下：

目前仅对 8583 格式报文进行了解包处理，显示应答报文的各个域和内容；而对于其他非 8583 的报文仅仅列出发送报文和接收报文的实际内容。

目前通讯协议仅支持短连接协议；因此目前对于多条发送只能逐条进行发送，暂不支持并发。模拟客户端和服务端均只支持 TCP 短连接。

功能特点

本软件主要功能实现全部基于脚本编写，改造方便，可自由扩充功能。

脚本支持常用语法 if/elseif/else if/end if , while/end while , for/end for 等。

脚本支持常用字符串功能 string.len(),string.format()等。

脚本支持常用时间处理功能 time.format(),time.clock()等。

脚本支持扩展动态库。

脚本支持用户自定义函数。

脚本范例参见”case\测试脚本”目录下的脚本文件。

需要更多功能,请联系作者: nj_zhujun@163.com

暂只提供经编译后的脚本(*.e 文件)

脚本示例:

```
print ( "time()=",time.time())
print ( "clock()=",time.clock())
print ( "date()=",time.date())
print ( "date()=%Y%m%d")
print ( "strftime()=%Y%m%d")
print ( "strftime()=Today:%Y/%m/%d")
print ( "strftime()=Time:%Y/%m/%d %H:%M:%S")
tm=time.localtime()
print ("tm->day=",tm.day)
print ("tm->day=",tm.month)
print ("tm->day=",tm.year)
```

```
a=1
print("start")
while (a<=10)
    print (a)
    a=a+1
end while
print("end while when a=",a)
```

适用范围

可以模拟各类报文;

- ✓ 针对 8583 规范, 可以模拟不同的 8583 规范;。
如支持: 银联规范报文中的各个规范接口 (终端 1.0, 终端 2.0, 商户 1.0, 商户 2.0, 渠道 1.0, 渠道 2.0, CUPS 发起等发起方的报文规范)。
- ✓ 针对定长, XML, 变长等报文, 目前仅显示报文内容。
- ✓ 模拟服务端, 根据不同的请求交易类型模拟对应的应答报文。

文件目录结构说明

程序目录结构如下所示:



case 目录



测试8583



测试xml



测试脚本

每个目录下配置了对应的 iso 格式标准，案例配置文件和案例文件
如 xml 目录下：

包含 case.ini ,iso.ini,*.txt

其中 case.ini 指定了案例的文件，在选择案例的时候根据输入内容查找到对应的案例文件。

其中 iso.ini 为相应的 8583 配置接口。

其中*.txt 为相应的案例配置文件。

file 目录

存放每笔交易报文信息，用于匹配原交易报文内容。

ini 目录

const.ini
配置设置
2 KBerrormsg.ini
配置设置
2 KBkey.ini
配置设置
1 KBseq.ini
配置设置
1 KB

Const.ini 常量配置文件，配置相关常量信息

Errormsg.ini 错误代码配置文件，配置代码和提示信息（可以添加应答码错误对照内容）

Seq.ini 流水号配置文件，存放交易流水号（系统自动更新，勿手工改动）

Key.ini 密钥配置文件，存放交易密钥信息（可在界面中修改）

log 目录

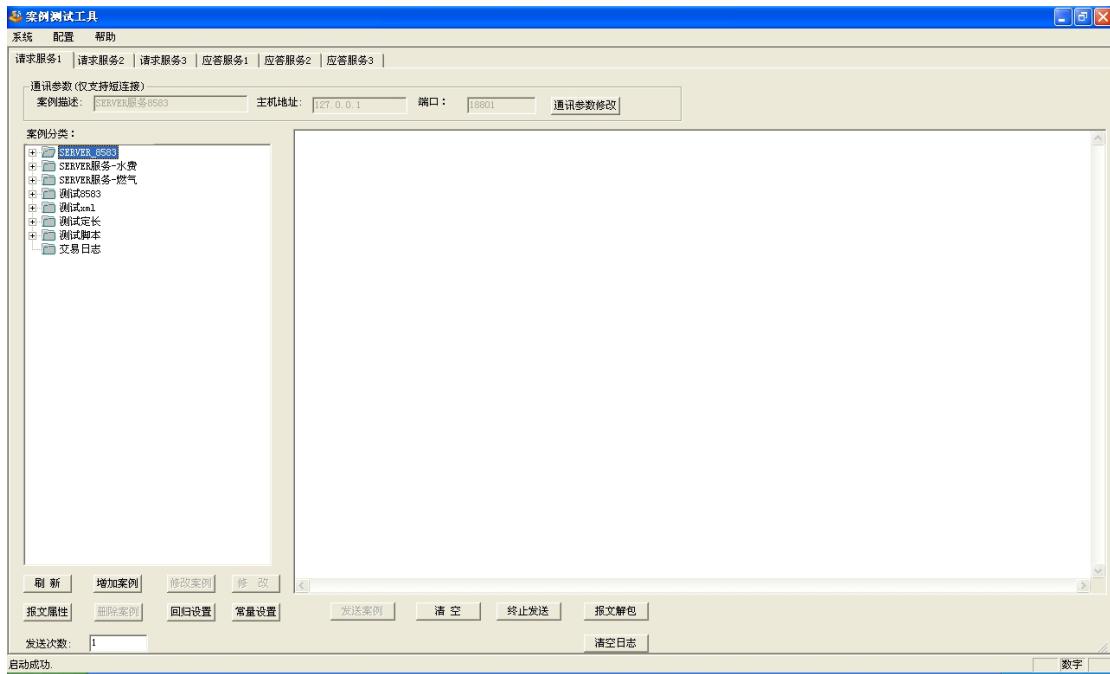
存放日志信息，日志信息统一按照日期进行存放，便于查看。

report 目录

存放案例回归测试日志信息和统计信息。

主界面

本软件启动后，主界面显示如下所示：



刷新: 刷新案例文件目录。

增加案例: 新建交易案例。

修改案例: 对案例进行修改。

修改: 对指定的文件内容进行修改。

报文属性: 设置报文的属性。

删除案例: 删除交易案例。

回归设置: 对案例集进行回归配置。

常量设置: 对常量内容进行配置。

使用说明

增加、修改删除案例

点击案例分类对应的案例，按右键弹出菜单。



密钥配置界面

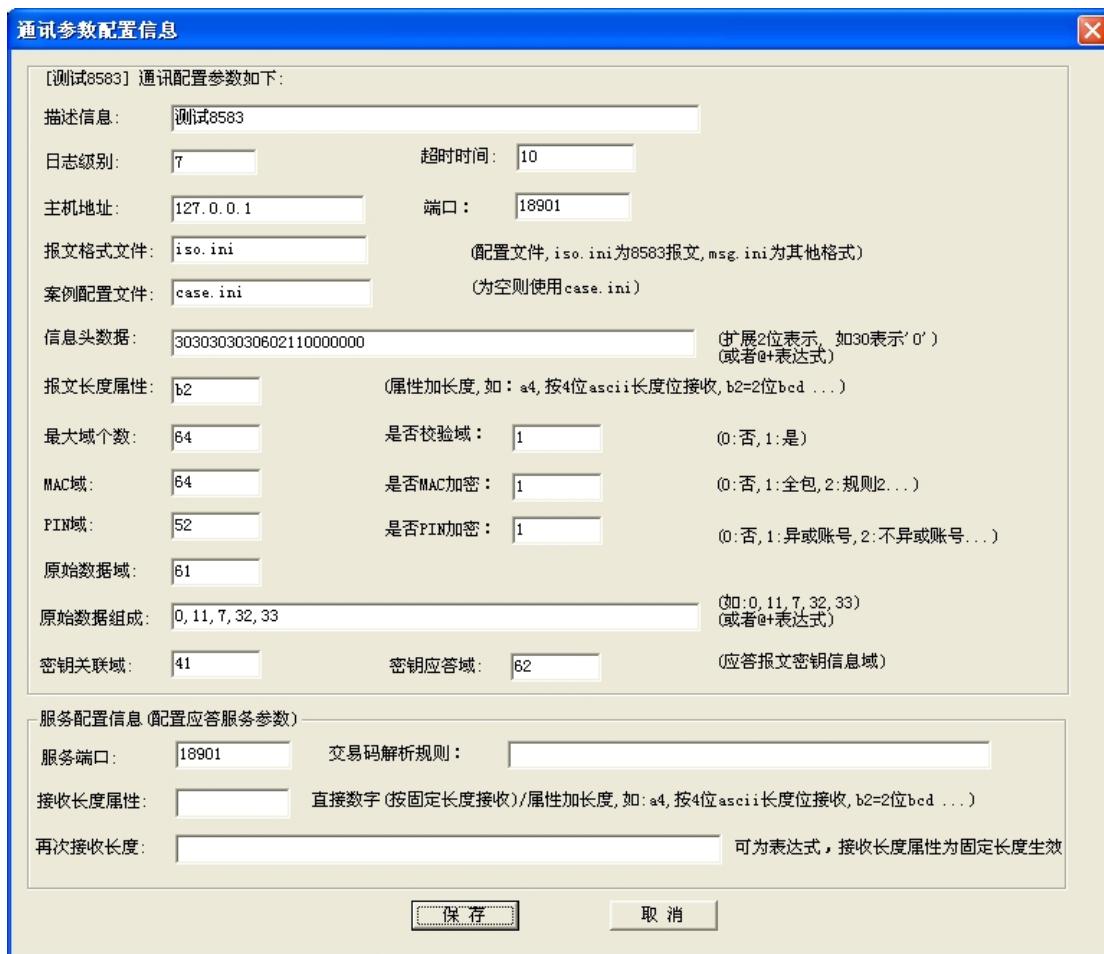
通过该界面可以手工设置对应的终端(机构)的主密钥, PINKEY 和 MACKEY 的相关信息。



PIKLEN (设定 PINKEY 的长度, 单倍长度为 16, 双倍长度为 32)

通讯参数配置界面

设置某类交易规范的通讯信息和其他配置信息。



通讯设置：

日志级别: 设置日志显示的级别，从 0-9，级别越高，显示的提示信息越多。

主机地址: 设置交易发送的主机地址。

端口: 设置交易发送的主机端口。

信息头数据: 设置交易报文的信息头报文内容。

报文长度属性: 设置交易报文的长度属性（属性+长度,a4: 4 位 ASCII, b2: 2 位 BCD)。

最大域个数: 设置交易报文的最大域个数。

MAC 域: 设置 MAC 所对应的域。

PIN 域: 设置 PIN 所对应的域。

是否 MAC 加密: 设置是否需要 MAC 校验。

是否 PIN 加密: 设置是否需要 PIN 校验。

密钥关联域: 设置密钥的取值关联域。

原子数据域: 设置密钥的原始数据关联域。

原始数据组成: 设置原始数据域的组成内容。

密钥应答域: 设置密钥的应答域，即签到交易从该域中取的密钥的信息，进行更新密钥。

服务设置：

服务端口: 作为服务端，绑定的 TCP 侦听端口。

交易码解析规则: 作为服务端，收到报文后，根据设定的交易码规则，解析出交易码。

接收长度属性: 设置交易报文的长度（直接数字）或者属性（属性+长度,a4: 4 位 ASCII,

b2: 2 位 BCD)。

再次接收长度: 对于长度不固定的报文, 可以先设置“接收长度”为固定值, 再根据此表达式设置“再次接收的长度”, 按此长度进行二次接收报文。

修改案例界面

修改案例配置

查询余额																																																																																														
交易描述:	支付类-查询余额	冲正原交易文件:		(非冲正交易不填)																																																																																										
是否生成MAC:	<input type="checkbox"/>	(无MAC域)																																																																																												
报文配置																																																																																														
<table border="1"> <thead> <tr> <th>域</th> <th>子项</th> <th>域描述</th> <th>生成方式</th> <th>实际内容</th> </tr> </thead> <tbody> <tr><td>0</td><td></td><td></td><td>指定值</td><td>0200</td></tr> <tr><td>2</td><td></td><td></td><td>自动生成</td><td></td></tr> <tr><td>3</td><td></td><td></td><td>指定值</td><td>310000</td></tr> <tr><td>11</td><td></td><td></td><td>自动生成</td><td></td></tr> <tr><td>22</td><td></td><td></td><td>指定值</td><td>021</td></tr> <tr><td>25</td><td></td><td></td><td>指定值</td><td>00</td></tr> <tr><td>26</td><td></td><td></td><td>指定值</td><td>06</td></tr> <tr><td>35</td><td></td><td></td><td>自动生成</td><td></td></tr> <tr><td>36</td><td></td><td></td><td>自动生成</td><td></td></tr> <tr><td>41</td><td></td><td></td><td>自动生成</td><td></td></tr> <tr><td>42</td><td></td><td></td><td>自动生成</td><td></td></tr> <tr><td>49</td><td></td><td></td><td>自动生成</td><td></td></tr> <tr><td>52</td><td></td><td></td><td>自动生成</td><td></td></tr> <tr><td>53</td><td></td><td></td><td>指定值</td><td>1000000000000000</td></tr> <tr><td>60</td><td></td><td></td><td>指定值</td><td>0100000165</td></tr> <tr><td>62</td><td>1</td><td></td><td>指定值</td><td>1</td></tr> <tr><td>62</td><td>2</td><td></td><td>指定值</td><td>2</td></tr> </tbody> </table>					域	子项	域描述	生成方式	实际内容	0			指定值	0200	2			自动生成		3			指定值	310000	11			自动生成		22			指定值	021	25			指定值	00	26			指定值	06	35			自动生成		36			自动生成		41			自动生成		42			自动生成		49			自动生成		52			自动生成		53			指定值	1000000000000000	60			指定值	0100000165	62	1		指定值	1	62	2		指定值	2
域	子项	域描述	生成方式	实际内容																																																																																										
0			指定值	0200																																																																																										
2			自动生成																																																																																											
3			指定值	310000																																																																																										
11			自动生成																																																																																											
22			指定值	021																																																																																										
25			指定值	00																																																																																										
26			指定值	06																																																																																										
35			自动生成																																																																																											
36			自动生成																																																																																											
41			自动生成																																																																																											
42			自动生成																																																																																											
49			自动生成																																																																																											
52			自动生成																																																																																											
53			指定值	1000000000000000																																																																																										
60			指定值	0100000165																																																																																										
62	1		指定值	1																																																																																										
62	2		指定值	2																																																																																										
合法性配置 (C:可选, M: 必选, X:必须无, _:原样返回) <table border="1"> <thead> <tr> <th>域</th> <th>域描述</th> <th>检查标志</th> </tr> </thead> <tbody> <tr><td>3</td><td></td><td>M, _</td></tr> <tr><td>39</td><td></td><td>X, M</td></tr> <tr><td>41</td><td></td><td>M, _</td></tr> <tr><td>42</td><td></td><td>M, _</td></tr> <tr><td>49</td><td></td><td>M, _</td></tr> <tr><td>52</td><td></td><td>C, X</td></tr> </tbody> </table>					域	域描述	检查标志	3		M, _	39		X, M	41		M, _	42		M, _	49		M, _	52		C, X																																																																					
域	域描述	检查标志																																																																																												
3		M, _																																																																																												
39		X, M																																																																																												
41		M, _																																																																																												
42		M, _																																																																																												
49		M, _																																																																																												
52		C, X																																																																																												
<input type="button" value="保 存"/> <input type="button" value="取 消"/>																																																																																														

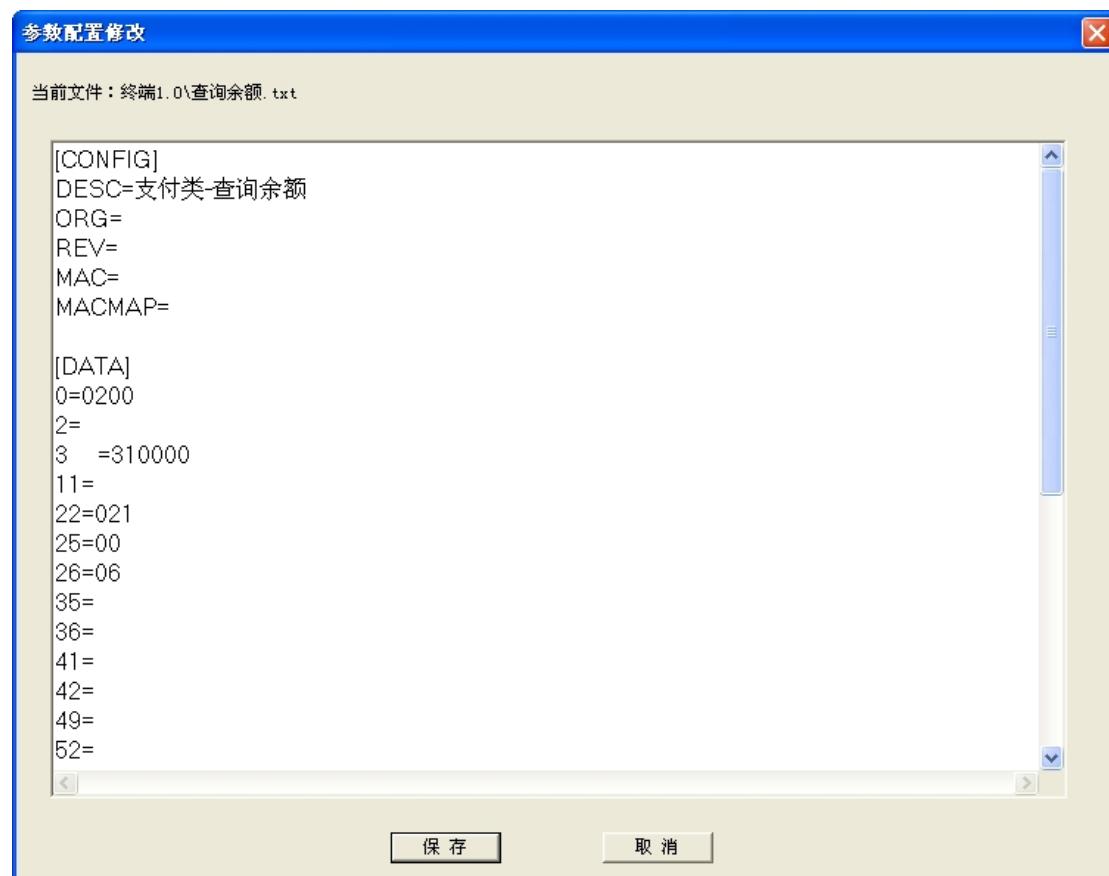
通过界面的模式对交易案例进行修改, 修改后保存即可。

域: 待设置的域, 并赋值。

子项: 设置域的子项, 并指定值。

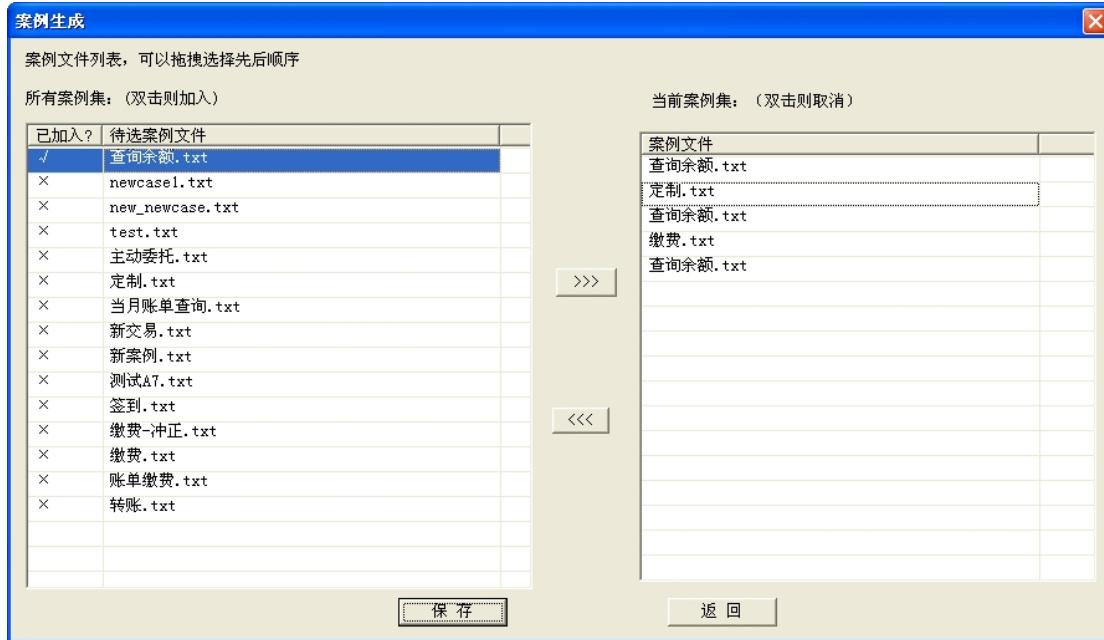
检查标志: 设定请求, 应答 X,_M,C 等类别 (X: 必须无, M: 必须有, _: 应答必须同请求,C:可选)

文本模式修改界面



直接对案例配置文件进行修改，修改完毕后点击保存。

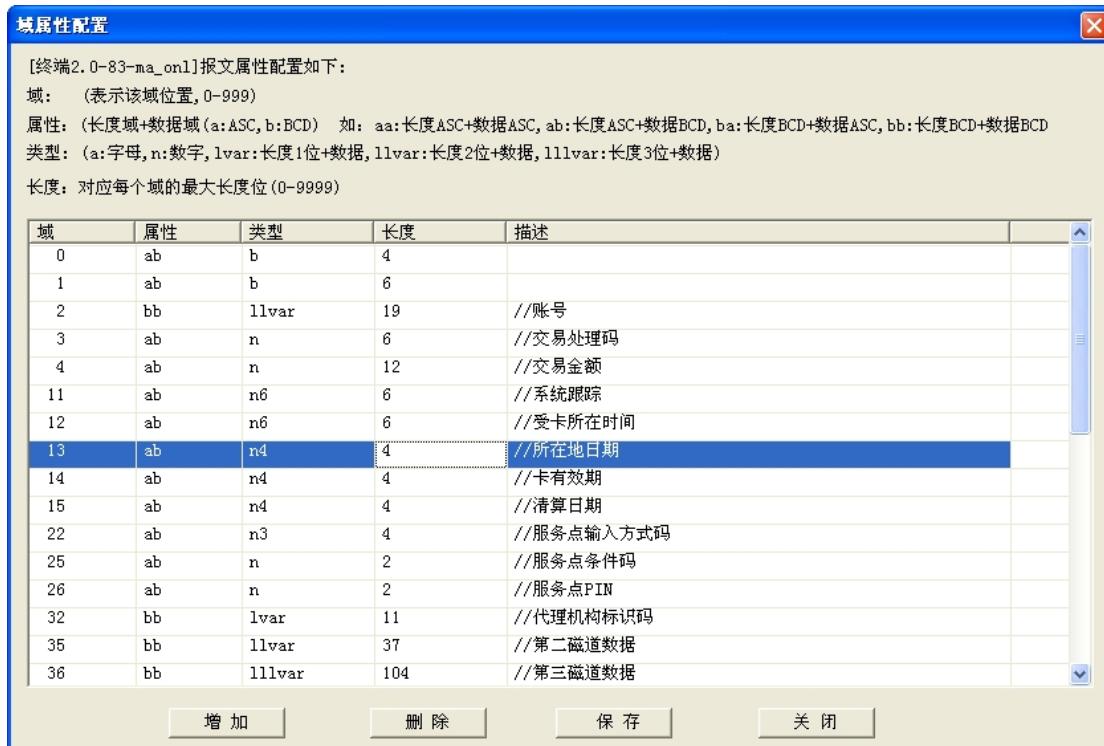
设置案例集界面



设置案例集，左边显示的是目前已经配置好的案例，右边是待发送的回归案例集。
通过拖拽的方式可以实现案例的发送顺序。

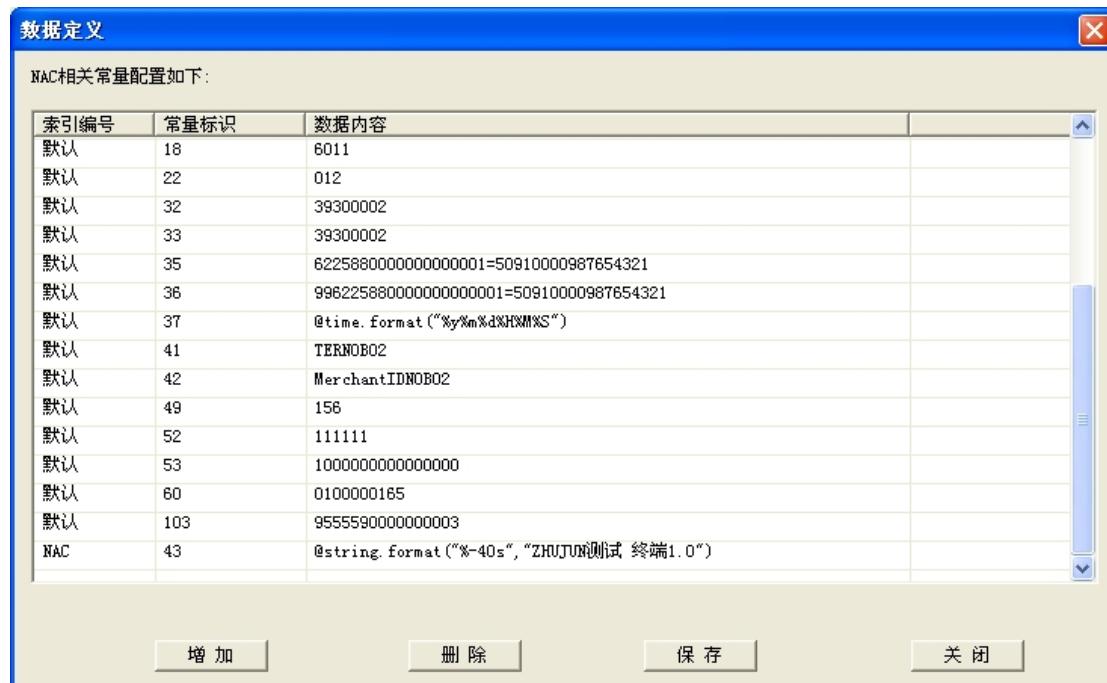
报文属性设置

配置每个域的属性，类型，长度已经描述信息。



常量设置界面

配置交易的常量信息



交易发送时从配置的常量信息表中获取相关的结果进行组包发送。

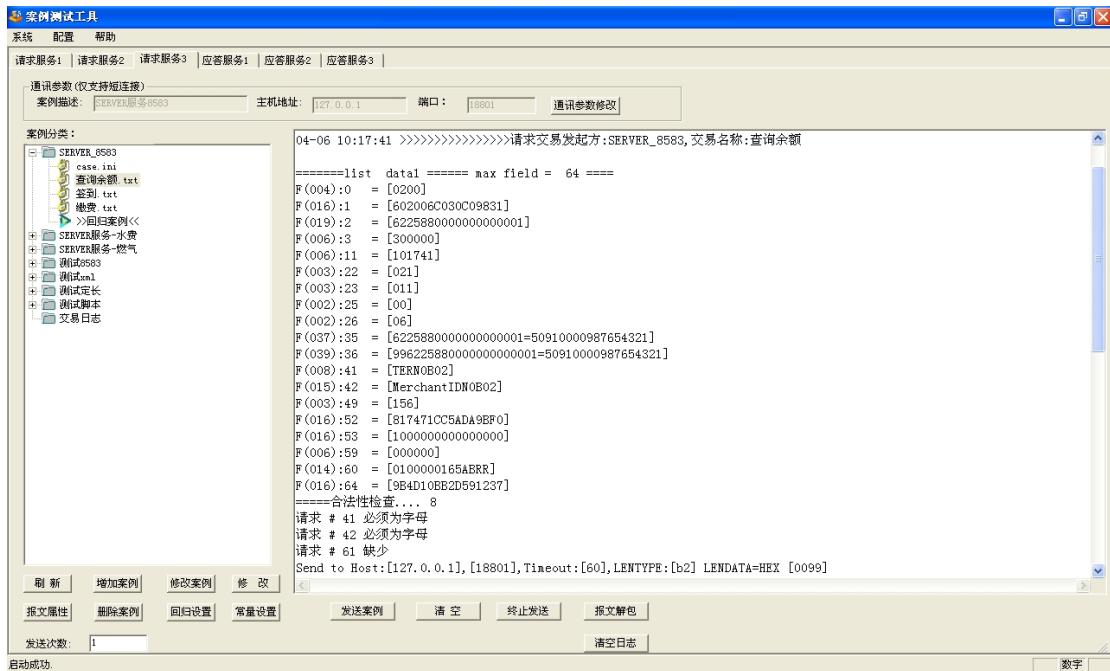
“索引编号”为“默认”表示全局有效，否则仅对单独的报文集有效。

其中

@string.format 为字符串格式化

@time.format 为时间串格式化

发送案例



发送次数

设置发送的次数，进行循环发送

清空

清空界面显示的提示信息

终止发送

终止目前的交易

清空日志

清理多余的交易日志

文件格式说明

config.ini 例子

```
[DEBUG]
LEVEL=5

[CONFIG]
HOST=172.17.252.82
#HOST=127.0.0.1
PORT=17205
CFGFILE=iso.ini
MAXFLD=128
MACFLD=128
MAC=1
ORGFLD=0,11,7,32,33
CASEFILE=case.ini
CASEPATH=mer/
ORGPATH=file/
#密钥域
KEYFLD=32
```

Case.ini 例子

```
#输入选项, 文件名  
1,签到.txt  
2,当月账单查询.txt,  
3,查询余额.txt,  
4,购买电子产品.txt
```

iso.ini 例子

```
#域 打包类型 数据类型 长度 描述  
2 aa llvar 19 //账号 n..19(LLVAR) 2 个字节的长度值+最大 19  
个字节(数字字符)的主账号  
3 aa n 6 //交易处理码 n6, 6 位定长数字字符
```

类型为 LLVAR: 长度按 2 位打包
类型为 LLLVAR: 长度按 3 位打包
类型为其他: 不进行长度位打包
aa: 长度 asc, 数据 asc
bb: 长度 bcd, 数据 bcd
ab: 长度 asc, 数据 bcd
ba: 长度 bcd, 数据 asc

交易文件的配置

交易文件以 txt 格式存放, 分为 CONFIG,DATA,CHECK,SCRIPT 描述。
具体格式如下:

[CONFIG]段说明:

DESC:案例描述信息

REV:指定冲正文件

MAC:是否校验 MAC(N:不校验)

UPDATEKEY:是否更新 KEY(N:不更新)

[DATA]段说明:

每行格式:域[.子域]=内容(空则取常量配置,自动生成)

例:0=0200

例:60.1=123

例:60.2=456

[CHECK]段说明:

每行格式:域=请求标志,应答标志(M:必填,O:可选,C:条件可选,X:必无,-:必须和请求相同)

即 M,M-请求必送/应答必回;C,M:请求可选/应答必回

例:0=M,M

例:39=X,M

[SCRIPT]段说明:

//!!判断脚本!!如: (if (resp(3) != "310000") print("应答 3 不为 310000") end if)

正交易配置

```
[CONFIG]
#交易名称描述
DESC=账单费用查询-托管商户
#原始数据信息
ORG=
#原交易名
REV=
#MAC 标志
MAC=
#MAC 域
MACMAP=

[DATA]
0=0200
2=
3=310000
7=
11=
25=92
35=@string.format("%s=05091000098765432", req(2))
41=
42=
48.1=0000
48.2=DF440005ABCDEF
48.3=DF4F01220060808
48.10=#
49=
60=07123456001
[CHECK]
[SCRIPT]
```

配置格式为: 域(.子域)=内容
若内容为空则取常量中的设置

反交易配置

冲正（撤销，退货）交易配置文件例子：

```
[CONFIG]
#交易名称描述
DESC=撤销
#原始数据信息
ORG=
#原交易名
REV=缴费.txt
#MAC 标志
MAC=
```

```
[DATA]
0=0200
3=&
7=
11=
12=
13=
25=&
32=
33=
41=
42=
48=&
49=&
60=&
61=&resp
[CHECK]
[SCRIPT]
```

&指引用原交易域，必须指定 REV，设置原交易的文件名

&req 指仅引用原交易请求域

&resp 指仅引用原交易应答域，必须有应答报文后存在该值。

可选配置

[CHECK] 可选，可以通过界面配置或者直接修改文件

[SCRIPT] 可选，只能修改文件

例子 1：

```
[CHECK]
0=M,M
[SCRIPT]
if(string.len(resp(0)!=4) print("0#位数不正确!") end if
```

以上例子：

检查 0#请求，应答域必须都存在，否则打印“请求 #0 缺少”等类似提示信息。

如果应答 0#长度不为 4，则打印“0# 位数不正确”的提示信息。

例子 2：

```
[SCRIPT]
/* 检查应答 3#长度是否==6*/
if (string.len(resp(3)!=6) print("应答 3#位数不正确!") end if
/* 检查 3#是否为 310000*/
if (resp(3) != "310000")  print("应答 3#不为 310000") end if
```

通过自定义 SCRIPT，可以对应答报文进行更丰富的处理。

更多功能请自行测试。

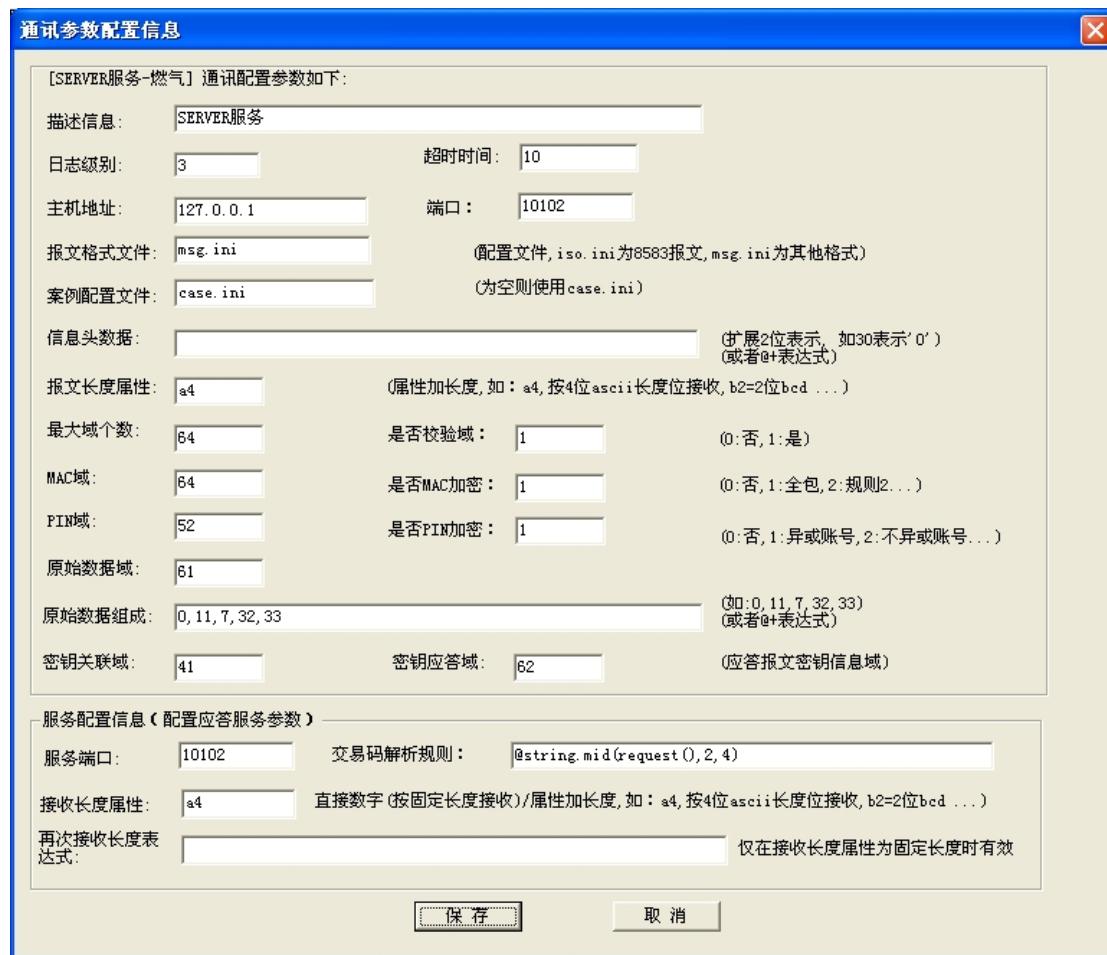
服务应答配置

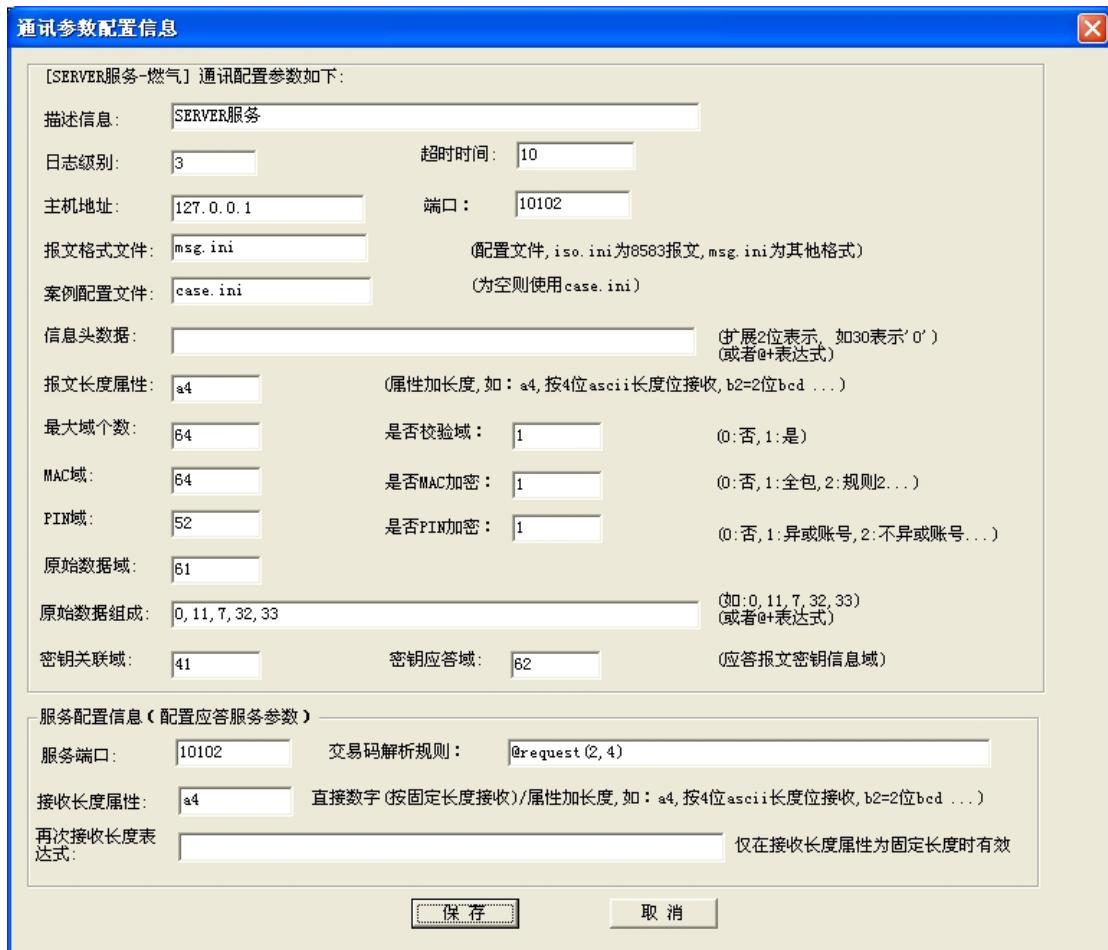
案例设置

文件以_开头，作为应答服务文件。

非 8583 配置

例子 1:





配置“交易码解析规则”，作为解析请求报文交易码规则。

如@string.mid(request(),2,4)，交易码为请求报文的第2个字节开始截取4位长度。

备注：request()为收到的实际报文内容。

@string.mid(request(),2,4)等价于@request(2,4)

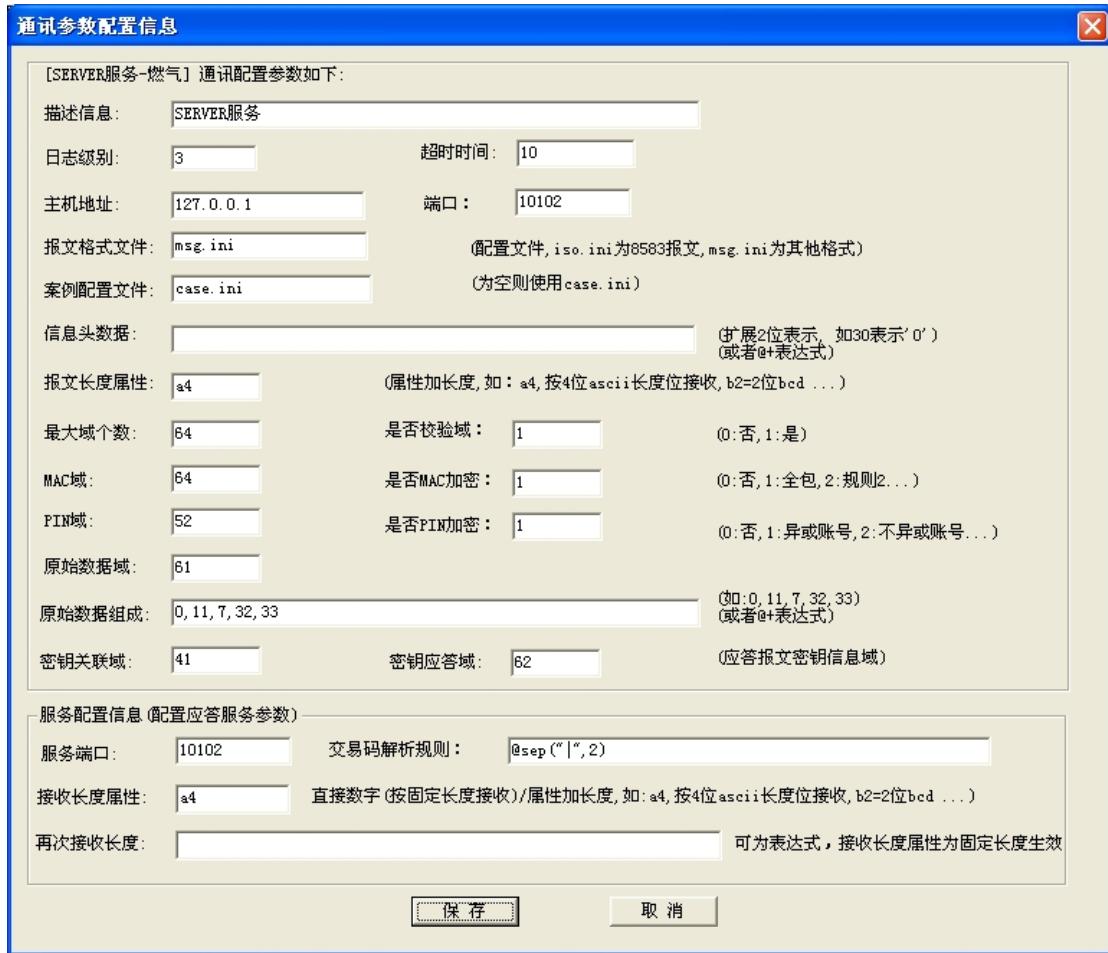
即：

接收到报文为 |2001abcdef.....

则解析交易码为‘2001’

则读取案例为_2001.txt的文件进行发送。（若无对应文件，则原包返回）

例子 2:



配置“交易码解析规则”，作为解析请求报文交易码规则。

交易码解析规则为：@sep(' ', 2)，交易码为分隔符请求报文的第 2 域。

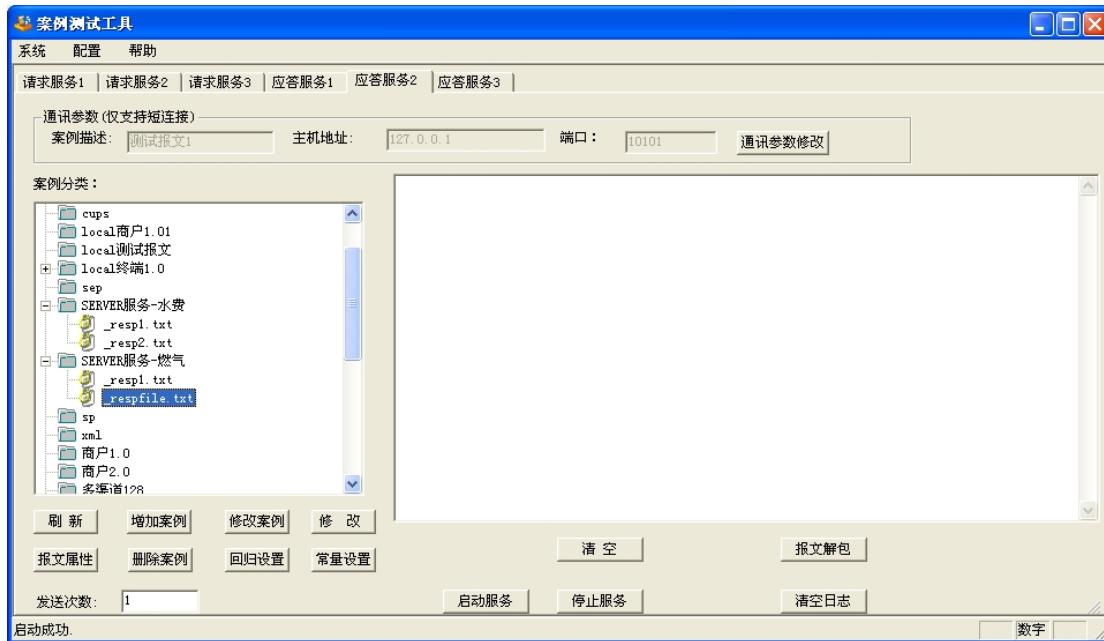
即表示：

如接收到报文为 |2001|abcdef.....

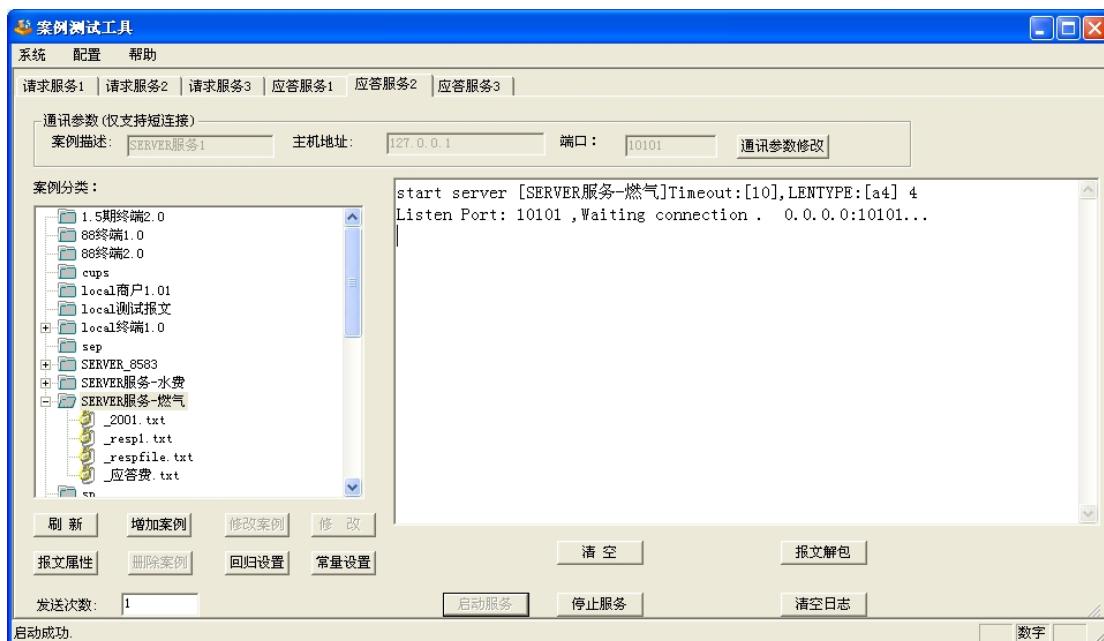
则解析交易码为 ‘2001’

则读取案例为 _2001.txt 的文件进行发送。（若无对应文件，则原包返回）

启动服务



点击对应的服务，或者点击启动服务按钮，则可以启动服务。

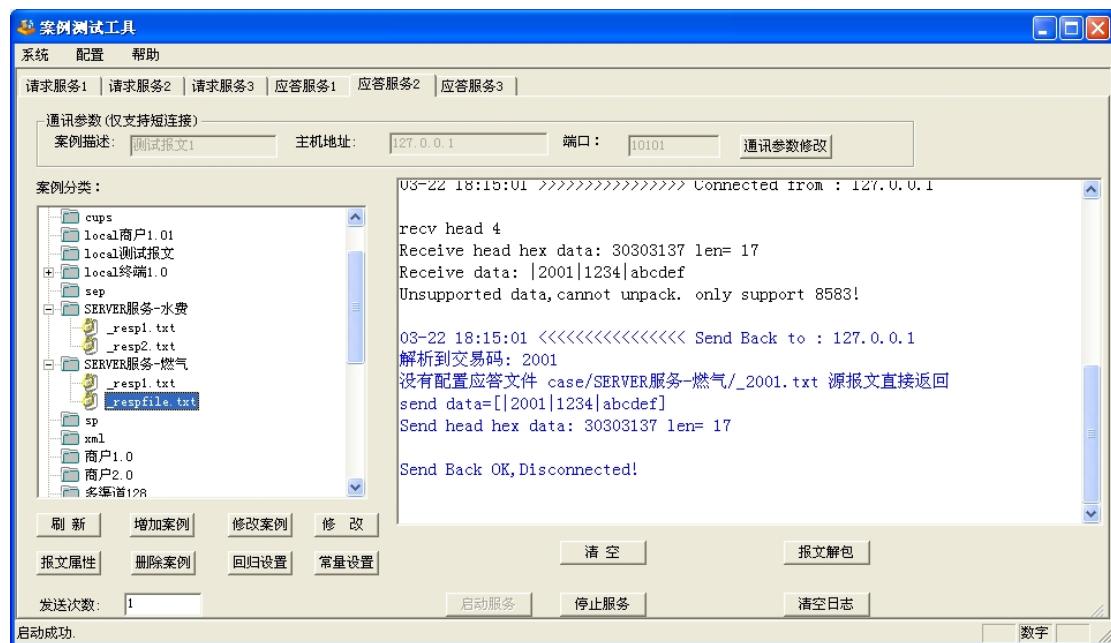


如上图所示, 启动“SERVER 服务-燃气”,
绑定 10101 端口进行侦听, 返回对应的报文。

返回应答

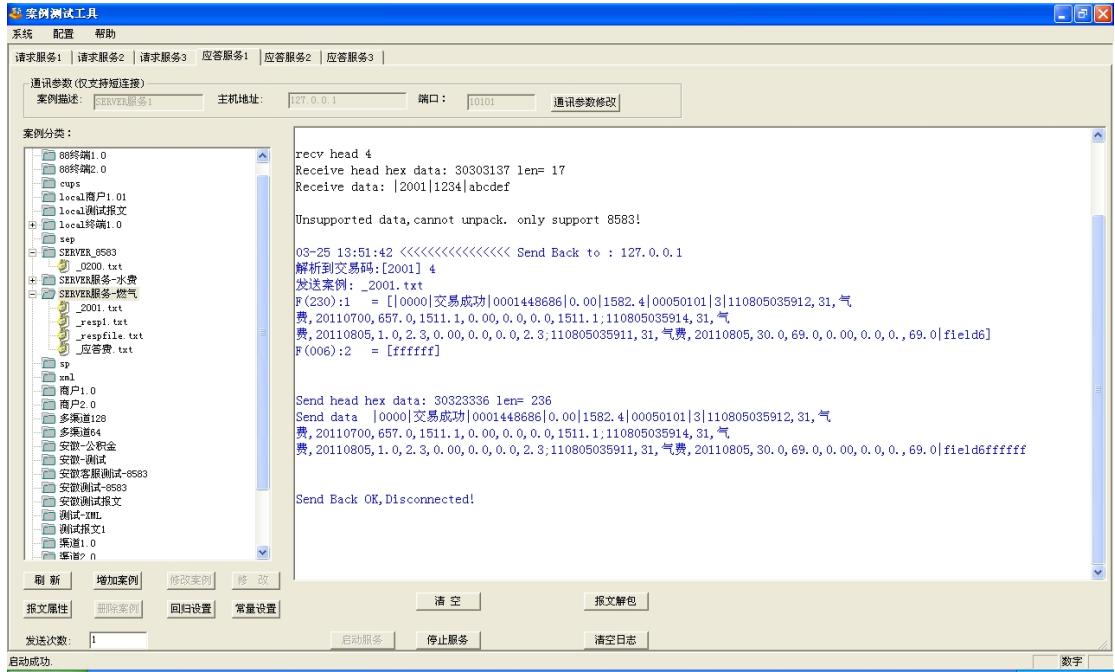
如未配置对应的应答文件, 则发送源报文, 返回。

显示如下:

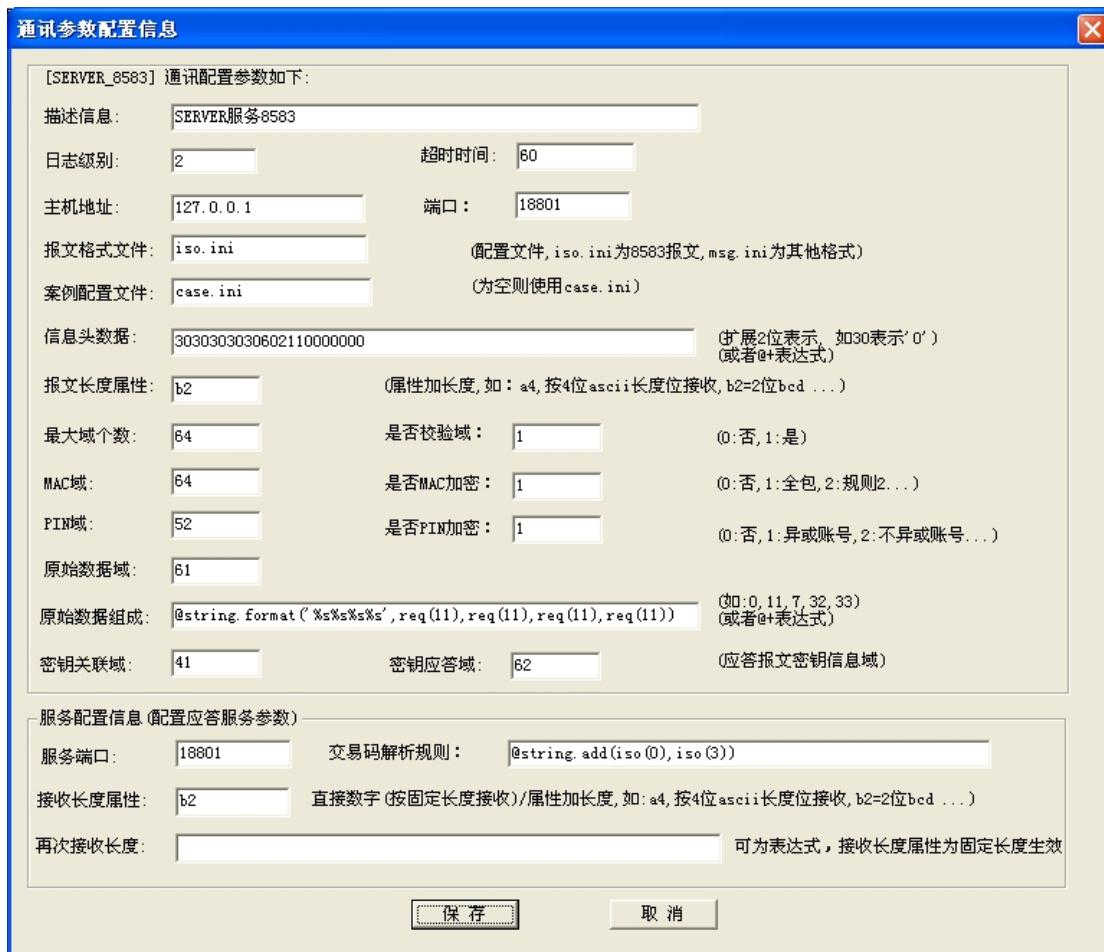


如配置了对应的应答文件, 按照文件配置的格式组织应答包。

执行结果显示如下:



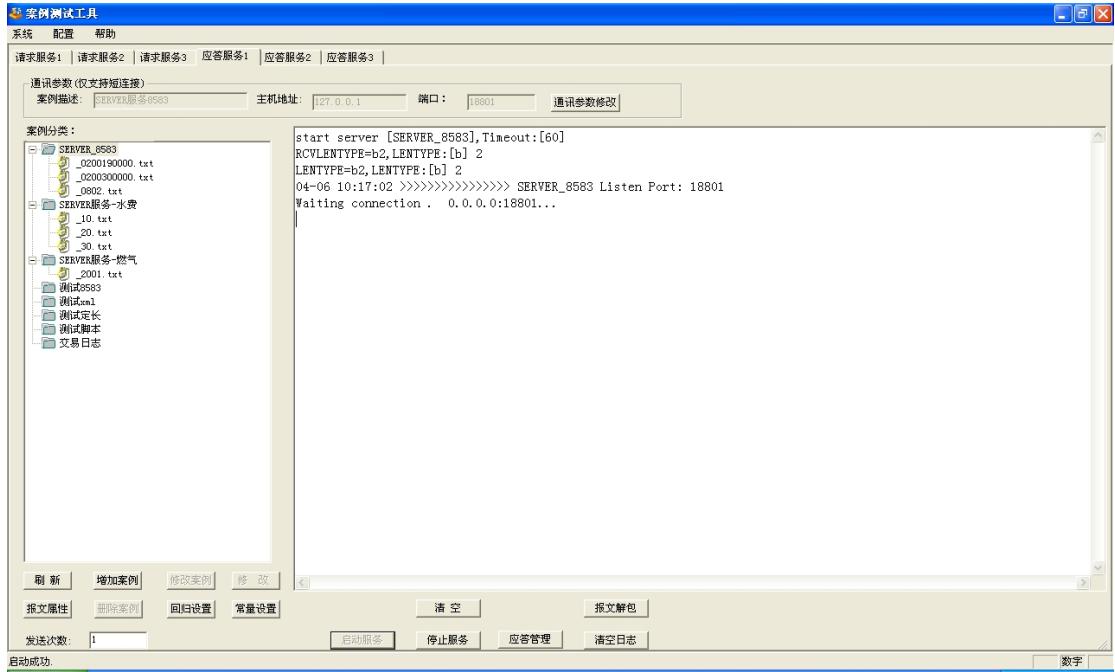
8583 配置



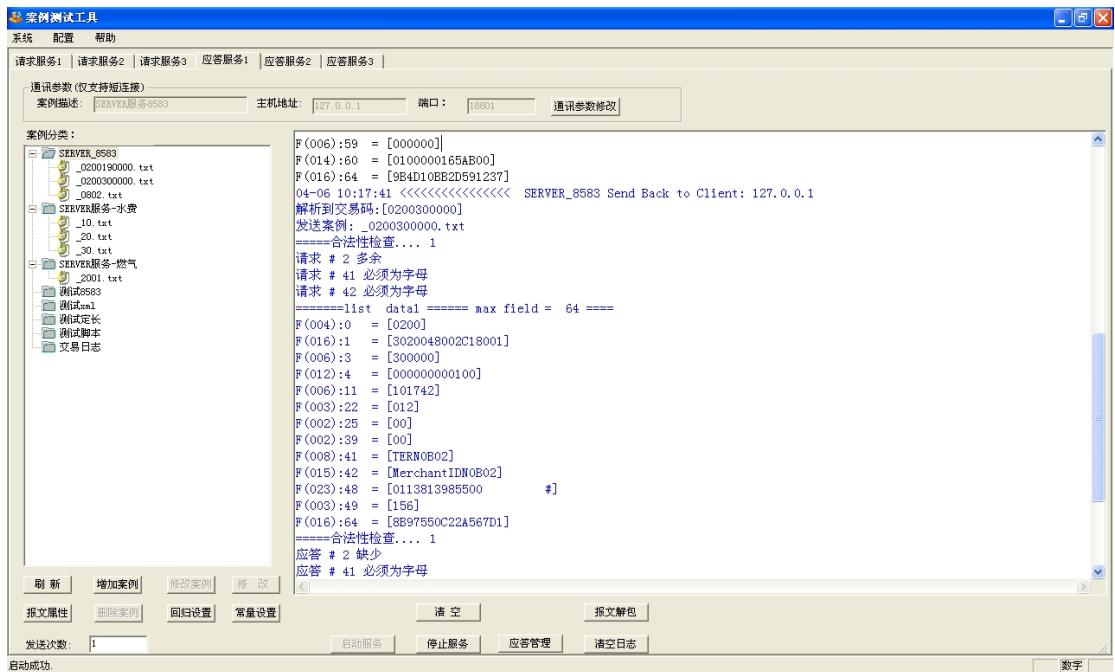
交易码解析规则为: @string.add(iso(0),iso(3)),
 表示: 交易码为请求 ISO8583 报文的第 0 域和第 3 域的组合。
 即如收到交易码为 020030000, 则按应答配置 _020030000 进行应答返回。

启动服务

选择制定的服务案例，点击“启动服务”



返回应答



应答流程为解析交易码，根据对应的案例文件发送，如果对应的案例文件不存在，则返回原请求内容。

支持函数列表

设定的数据前如果有@符号，则作为函数处理，进行表达式解析。

目前支持如下函数

String

字符串类操作函数。

string.format 类似 C 语言中的 sprintf

string.format("%s","123")	123
string.format("%s%s","123","456")	123456

string.len 类似 C 语言中的 strlen

string.len("abcd\n1234")	9
String.len("\01\02\03\04\05")	5
String.len("\00\01\02")	3

String.hex 将字符串转为十六进制

String.hex("x12\x34")	1234
String.hex("30")	3330

Stirng.asc 将十六进制转为 ASC

Stirng.asc("30")	0
String.asc("3330")	30

String.mid()

String.left()

更多请自行测试。

time

time.format 类似 C 语言中的 time 格式

time.format("%Y%m%d %H:%M:%S") // YYYYMMDD HH:MM:SS

time.time 类似 C 语言中的 time

time.clock 类似 C 语言中的 clock

Req

@req 表示发送报文的请求域

用法: @req(field)

如 req(1) 表示请求 1 域的内容

Rev

rev 表示原交易的域内容 (flag 参数可选)

用法: @rev(field [,flag])

Flag 不填先取请求域, 如无请求则再取应答域的内容

Flag=1 特指原交易请求

flag=2 特指原交易应答

如 rev(2) 表示只取请求 2 域的内容

rev(39,2) 表示只取应答 39 域的内容

Def

@def 表示取常量设置中的内容

用法: @def(text)

如:

def("cardno") 取常量中设置的 cardno 对应的内容

def(2) 取常量中设置的 2 对应的内容

Tlv

@tlv 表示设定(tag-length-value)的格式

用法: @tlv(tag,value,[flag])

flag 参数可选, 取值为 aa,ab,bb,ba 中任一, 默认 flag 为 bb

分别对应于 tag 是否压缩 (b 压缩, a 否), value 是否压缩 (b 压缩, a 否)

即 tag 压缩 (如 “9F26” -> “ 0x9f,0x26”), value 压缩 (如"abcd" -> "0xab,0xcd")

如:

tlv("9F46","abcd") 设定 9F46, 内容为 0xab,0xcd

实际报文 [9F][46][02][AB][CD]

tlv("9F47","1234","bb") 设定 9F47, 内容为 0x12,0x34

实际报文 [9F][47][02][12][34]

tlv("9F47","1234","ba") 设定 9F47, 内容为 1234

实际报文 [9F][47][04][31][32][33][34]

Tlvasc

tlvasc 表示设定(tag-length-value)的格式, 均为 asc 形式, length 为 3 位 (左补零)

用法: @tlvasc(tag,value,flag)

如:

tlvasc("9F46","1234") 设定 9F46,内容为 1234, 结果为:9F4600512345,
实际报文[39][46][34][36][30][30][34][31][32][33][34]

服务端函数列表

Request

@Request()表示作为服务端, 收到的客户端实际全部报文内容。

用在应答服务解析规则中。

用法: @request([start] [,len])

注意: 在应答服务中, 分多次接收的时候, @request()返回的内容取决于接收到的实际内容。

@request() 表示: 全部报文

@request(4) 表示: 报文第四位后的所有内容

@request(1,4) 表示: 报文第一位开始截取 4 位

Iso

@iso()表示作为服务端, 收到的客户端为 8583 报文时, 根据规则解析后, 指定的域值内容。

用在应答服务解析规则中。

用法: @iso(field)

如: @iso(0) 表示 0 域

@iso(3) 表示 3 域

@string.add(iso(0),iso(3)) 表示 0 域+3 域

Sep

@sep()表示作为服务端, 收到的客户端为分隔符报文时, 根据规则解析后, 指定的按分隔符解析后的域值内容 (从 0 开始)。

用在应答服务解析规则中。

用法: @sep(sepstr,index)

如报文为 |2001|1234|abc|def

@sep('|',1) 表示'2001'

@sep(' | ',2)

表示'1234'

常见问题

更新计划

版本持续更新中。

更新记录请参见“修改记录”。

如果您发现有什么 BUG, 或者有什么更好的建议, 请联系作者: nj_zhujun@163.com

修改记录

Version 2.1 126

修正一些 BUG。

界面部分布局调整优化。

增加了部分配置的要素。

Version 2.0 925

修正一些 BUG。

界面部分布局调整优化。

Version 1.7.0 702

增加格式化串功能。

由原先的%s,%10s,%019s 等扩展为: 格式串, 左右对齐标志, 对齐字符的格式。
如: %s,L,0 (左对齐, 补 0)

Version 1.7.0 521

修正一些 BUG, 增加服务端的设置。

Version 1.6.4.405

增加了应答服务的交易码解析，根据解析后的交易码匹配应答案例。

增加了再次接收长度，对于特殊报文，可先读取一定长度的内容，再根据此配置读取指定长度。

Version 1.6.4.317

增加了应答报文的处理，配置案例文件如: _resp.txt(以_开头)，按该文本内容格式发出报文。

增加了服务配置的接收长度属性，指明长度（如 96: 按 96 长度位固定接收），或者（a4-按 4 位长度位接收,b2-按 2 位 BCD 码长度位接收）。

Version 1.6.3.212

界面布局重新调整，适应不同分辨率。

程序优化，体积更小，速度更快。

修正一些 BUG

Version 1.6.0.108

增加报文属性中“自动化格式串”，对相关域按照指定格式进行格式化处理。

修正一些 BUG

Version 1.5

增加报文解包功能(暂只支持 8583)，输入 8583 报文按报文属性进行解包

增加报文长度属性功能 (b2: 表示 2 位 bcd 长度,a4: 表示 4 位 asc 长度)

增加显示修改案例中的“域描述”内容，同“报文属性”中每个域的“描述”

修正一些 BUG

版本更新为 1.5

Version 1.4

扩展@rev 用法，增加可选参数

增加支持多个交易请求标签

增加应答服务端标签，原包返回

增加报文域内容合法性校验(属性为 a: 校验字母，n: 校验数字，an: 校验数字和字母)

Version 1.3

修改报文属性中域属性，00-aa,01-ab,10-ba,11-bb,(a 表示 asc, b 表示 bcd)

增加 tlv 支持 (@tlv 实现 tlv 打包报文)

扩展反交易的“同原交易配置” (&表示)

增加同原请求@req,同原应答@resp,若配置了@resp, 但原交易无应答则无法发反交易

Version 1.0

开发初始版本 1.0

支持 8583 报文发送，接收报文处理。

支持 MAC 加密,PIN 转换等处理。