编写自动化测试脚本心得

--------入门篇

本文中将不会讲解 ISEE 的测试原理、不说明 Python 的常用语法、不介绍 OTP 测试平台的架构，自动化测试组的牛人们已经为我们编写了很多这些方面的资料，而且我也怕学艺不精说的不对，因为……我还是一只小小的菜鸟。写这篇文档分享我的一点点小心得，只是为了让后面更多的菜鸟们在编写第一个脚本的时候少一些困惑、多一点自信。

1、现在大家使用的 ISEE 工具，分为安装版和拷贝版。两者在使用上一个很大的区别是，拷贝版本不能新建测试用例、测试文件夹。使用拷贝版的同事，在已有测试用例中新建测试脚本，脚本的执行效果是一样的。

2、测试脚本的结构。常用测试脚本的结构基本相同，分为三大部分：

1）引用测试用例需要的类、库等文件

-----这部分的改动很容易

2）定义测试实现类 A，这个类通常有两个函数 def

Block1：测试用例初始化。

def InitTest(self):

-----这里主要是初始化 TA，大多数情况下不需要修改

Block2：测试用例主体

def Testing(self):

------这部分是我们的重点了，所有的脚本功能都要在这里定义完成

3）实例化 A，脚本执行定义动作的入口

-----这部分基本不需要改动，直接复用借用前辈们的代码就 OK 啦

3、脚本的第一行都会有这样一段，注意哦，这个不是注释，不能删除的。有了这句才能在脚本里写中文。

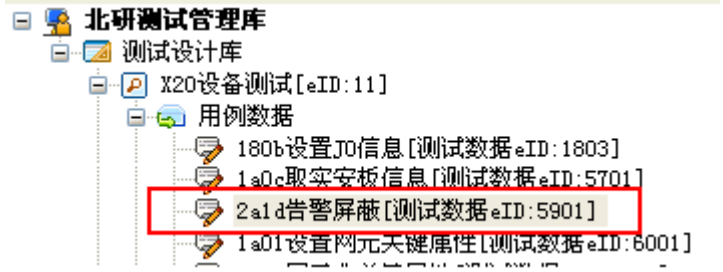
#coding:utf-8

4、脚本里需要发送的消息除了在脚本中需要构造输入参数之外，还要保证在 ISEE 中有对应命令码的用例数据。举例如下：

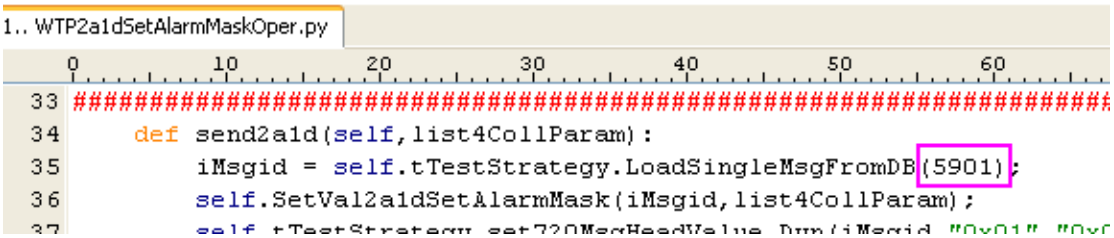
脚本中有如下代码，需要发送 0x2a1d 命令

```
self.tTestTaskManager.writeLog("设置板不在位告警屏蔽", 0)
list4CollParam = [[masterSNP, "0x00", "0x4", [["0x0721000000000ffffffffffffffffffffffff
SetAlarmMaskOperObj = WTP2a1dSetAlarmMaskOper("M800-1",self)
self.tTestTaskManager.clearMsg();
SetAlarmMaskOperObj.send2a1d(list4CollParam)
```

此时需要确认用例数据中有 0x2a1d 命令数据。如果没有需要新建，只要构造报文头部分就可以了，其他的内容我们强大的自动化平台全部在后台搞定。



到这里还没有结束，还需要将刚刚新建测试数据的 eID 值填写到对应的底层函数中。



5、输入参数的填写

发送函数的入参填写，是菜鸟们很困惑的问题。应该以什么形式、什么结构填写？答案只有一个：按照底层 XXXOper.py 文件中 sendXX 函数的注释格式，所有参数大家习惯上都是以 16 进制字符串格式填写。例如：

```
# 输入参数: dic4CollParam
#           = [SetType,
#              (
#               "板地址1":[应安类型1,应安硬件版本1,应安软件版本1,[应安板附加信息1,应安板附加信息2,......],[应安板描述信息1,应安板描述信息2......],[外部连线信息1,外部连线信息2,.......]],
#               "板地址2":[应安类型2,应安硬件版本2,应安软件版本2,[应安板附加信息1,应安板附加信息2,......],[应安板描述信息1,应安板描述信息2......],[外部连线信息1,外部连线信息2,.......]]
#               ......
#              )
#               ]

dic4CollParam = ["0x0",{"0x104":["0xd8", "0x3140000", "0x31e0000", ["0x101", "0x0"], ["0x101", "0x0"], ["0x101", "0x0"]]}]

SetReqBoardInfoObj = WTP1a0bSetReqBoardInfoOper("M800-1",self)
self.tTestTaskManager.clearMsg();
SetReqBoardInfoObj.send1a0b(dic4CollParam)
```

6、我们的脚本大多数时候都是在模拟 U3 网管对设备下发一些操作指令，通过设备的 reply 消息或者设备状态变化判断设备的功能情况。在脚本接收设备报文这部分，需要脚本设计者完成一个特定的清空消息队列、load 消息的操作。

```
self.tTestTaskManager.clearMsg();
GetActBoardInfoObj.send1a0c([strMasterSNP])
sleep(2)

self.tTestTaskManager.loadAllMsg("clear")
getBoardInfo = GetActBoardInfoObj.recv1a0c(self.tTestTaskManager.tDict4Msg)
```

至于这个约定的来由和原理，我就不转述了，直接推荐您阅读看下面这篇文档。

7、很多时候我们是要接收设备的应答，分析判断应答的具体字段内容。这部分我个人认为是个难点，需要 Python 语法的功力支持。对于这部分我只是想说，大多数消息的应答都是以数据字典和列表的形式返回给上层脚本的。所以建议大家提前学习一下数据字典和列表的基本操作，应该就可以搞定了。

上面就是我在完成了为数不多的几个脚本之后总结的一点小经验，希望可以帮助大家在初写脚本的时候更快掌握脚本的写作方法和技巧。