

# 攻防演练平台V1.0.2

# 自动化测试框架简介

# (Pytest+ allure+pyresttest+selenium+python)

创建时间：2019年11月19日

修改时间：2019年12月11日

**程胜**

地址：北京市朝阳区酒仙桥路6号院（电子城国际电子总部）2号楼

邮编：100015

## 一、环境准备

### 1、Python

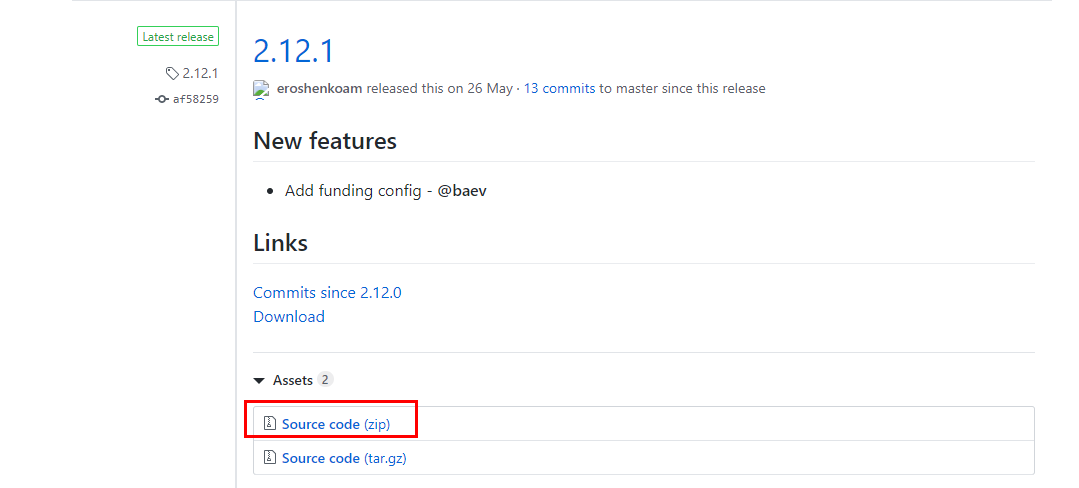
PC终端+linux虚拟机，均需安装python环境，PC终端：python3，linux虚拟机：python2.7

### 2、Allure2+Pytest+selenium（本地PC）：

在本地PC中安装allure组件和pytest模块

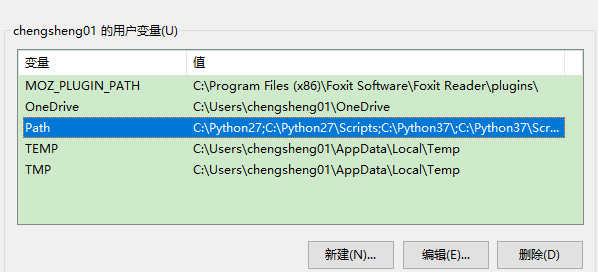
#### 2.1安装allure

<https://github.com/allure-framework/allure2/releases>

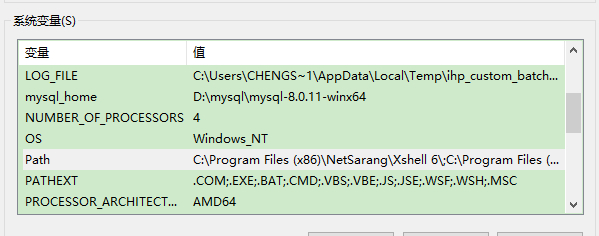


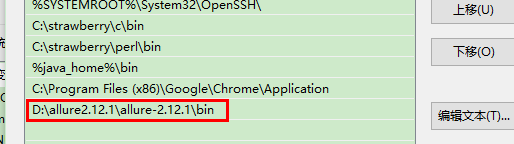


设置环境变量：

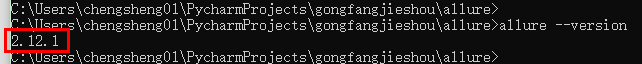






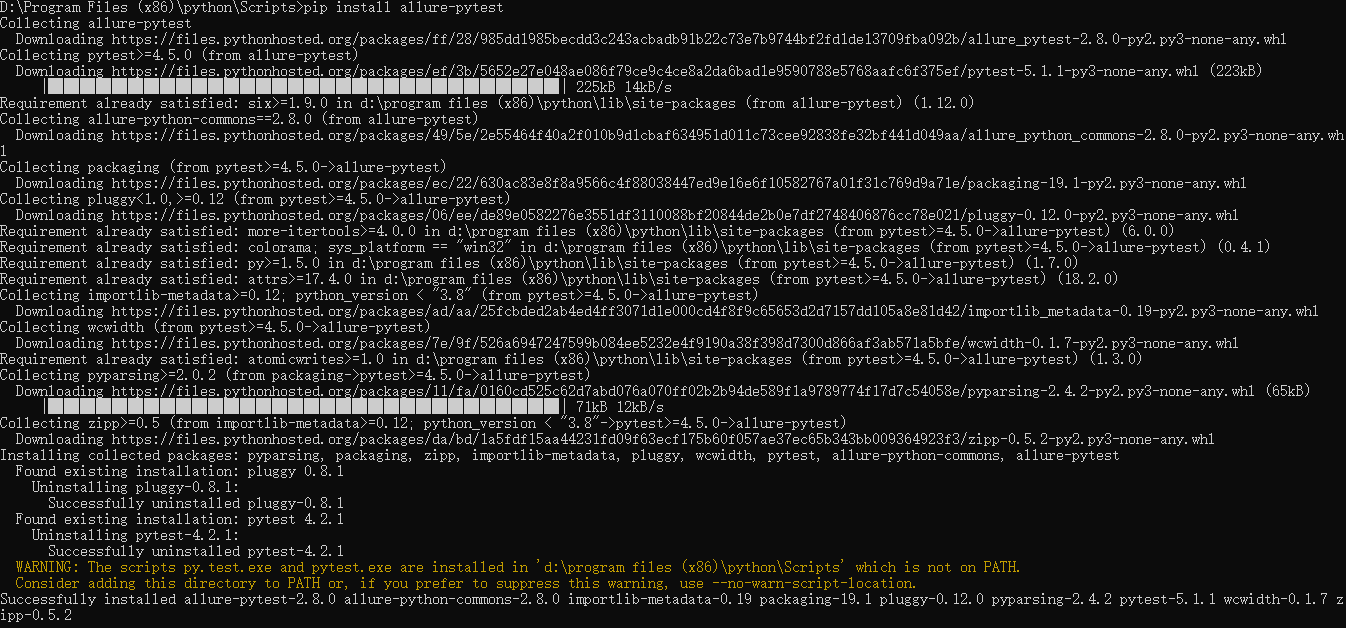


校验版本：

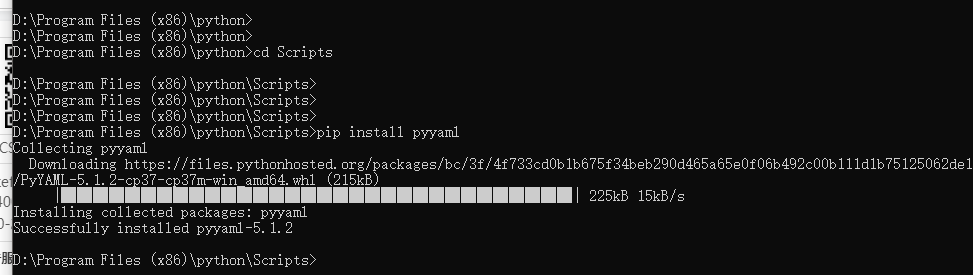


#### 2.2安装pytest：

1. pip install allure-pytest



1. pip install pyyaml



1. pip install pytest-html



#### 2.3安装selenium和webdriver：

1、pip3 install selenium

2、下载谷歌浏览器webdriver：

selenium之 chromedriver与chrome版本映射表

https://blog.csdn.net/huilan\_same/article/details/51896672

驱动下载地址

http://chromedriver.storage.googleapis.com/index.html?path=2.37/

本机谷歌浏览器安装目录

C:\Program Files (x86)\Google\Chrome\Application

将chromedriver.exe放入到chrome.exe同级目录下



（如果需要安装firefox，步骤类似，但是我操作发现需要多一个添加path路径的过程

Firefox的driver下载地址：https://github.com/mozilla/geckodriver/releases

Firefox与driver对应关系：https://blog.csdn.net/u013250071/article/details/78803230 ）

### 3、Pyresttest(linux虚拟机)

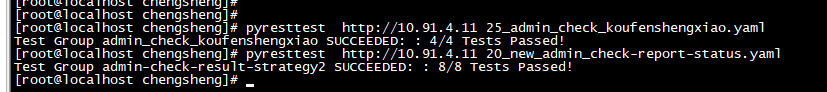
在本地linux虚拟机中安装pyresttest模块

安装包：pyresttest-1.7.1

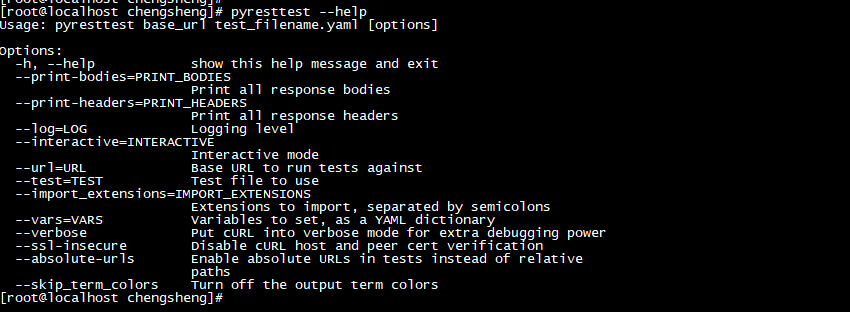


Copy到本地linux中（实际安装在的centos的虚拟机中，需要访问外网，本地虚拟机设置nat即可），执行文件夹下的安装程序：python setup.py install

安装完后，即可执行测试代码：



可以通过pyresttest http://10.91.4.11 --print-bodies=True test.yaml打印出详细的信息用于调试



## 二、框架运行逻辑

### 1、组网逻辑

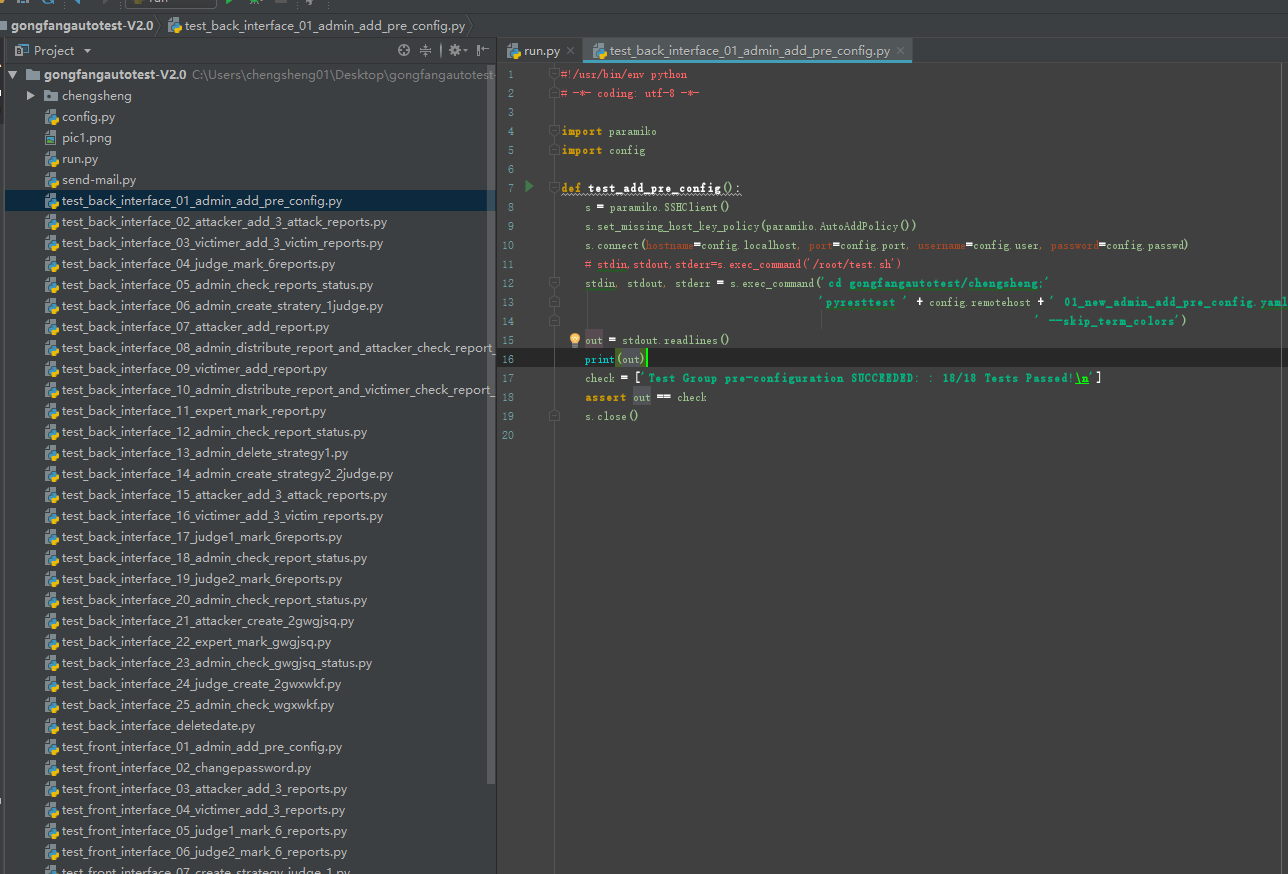
PC（allure+python3+Pytest+selinum）-------虚拟机（pyresttest+python2.7）--------被测试系统

### 2、代码运行逻辑

#### 2.1、后端自动化代码运行逻辑

PC（pytest-script）----虚拟机（pyresttest-and-python-script）-----被测试系统

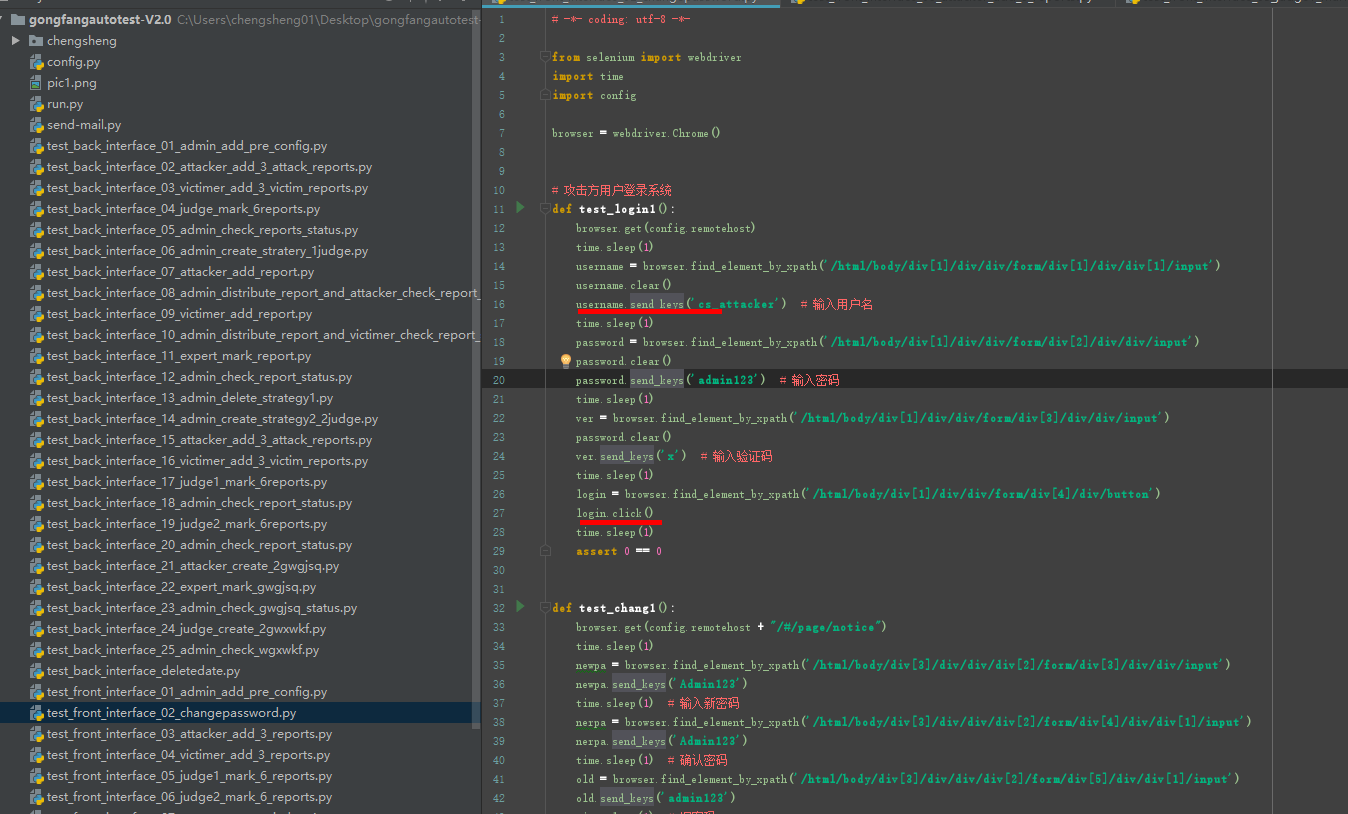
pytest脚本实现调用PC的CMD系统命令ssh链接到虚拟里,执行pyresttest 相关测试脚本，pytest用执行脚本后的回显做断言：



#### 2.2、前端自动化代码运行逻辑

PC（pytest+selenium）------被测试系统

实际代码通过selenium模块调用浏览器的webdriver，通过xpath定位页面元素，代码模拟web上的点击和输入，导入操作：

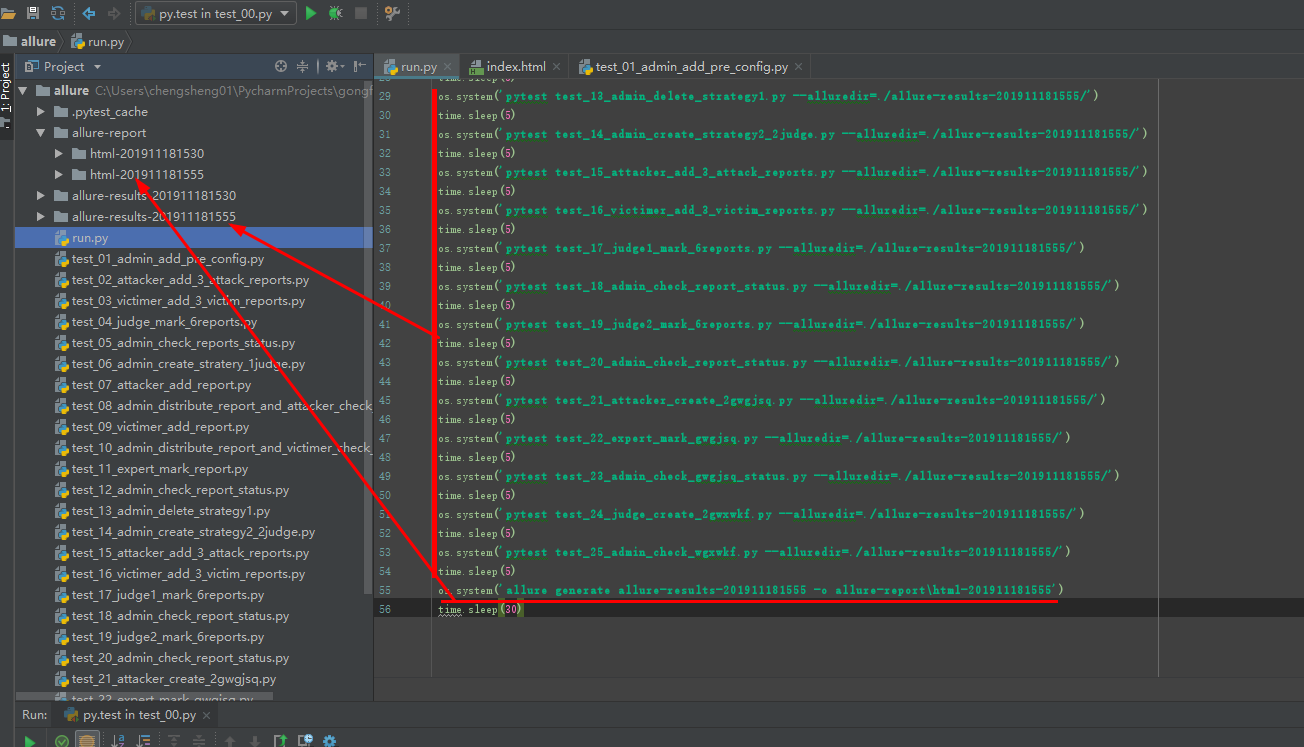


前端自动化运行效果：

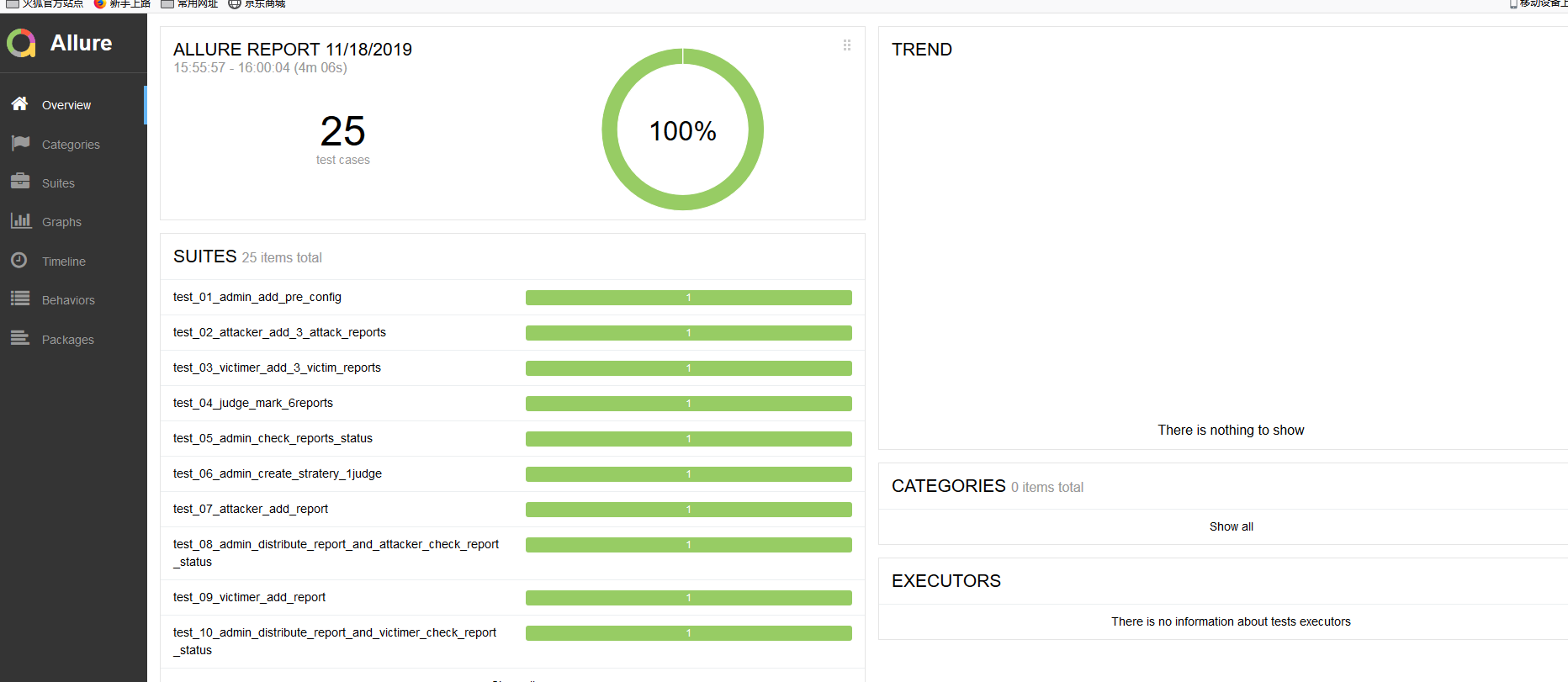


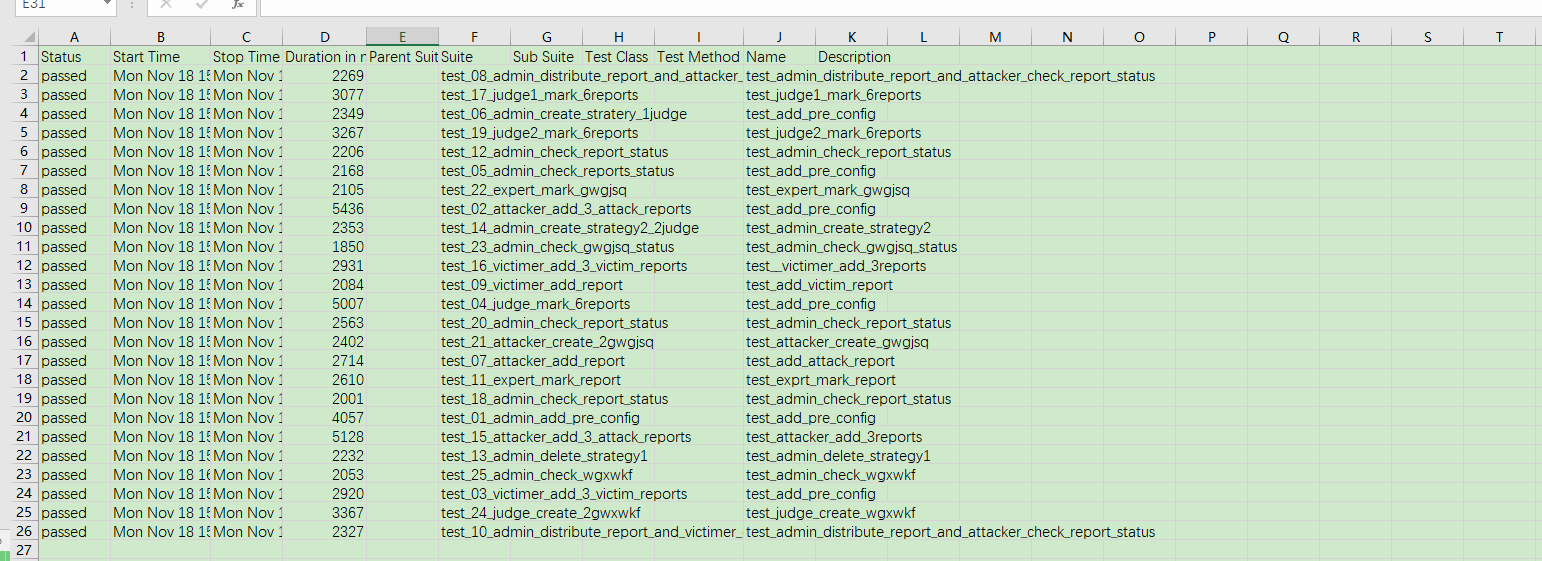
### 3、生成测试报告逻辑

PC上执行run测试脚本，run脚本中实际代码为调用windows的os系统命和alure组件在当前路径下执行pytest 测试脚本（exaple：'pytest test\_back\_interface\_01\_admin\_add\_pre\_config.py --alluredir=./allure-results-' + now），执行完了之后allure会输出结果到allure-results-now（now为一个变量，取的是实际run运行的时间）中；所有的pytest代码执行完之后会执行生成报告的命令：allure generate allure-results-' + now + ' -o allure-report-' + now + '\html-' + now：allure组建会将allure-results-now中的测试结果生成一个报告到html-now中



1. 测试报告：





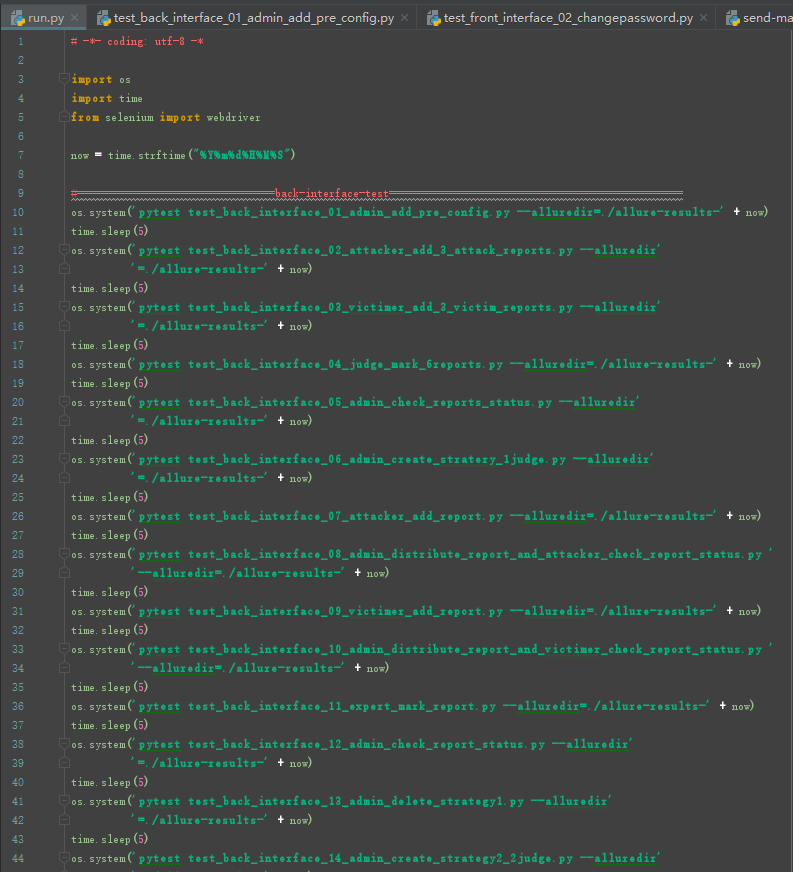
附html：

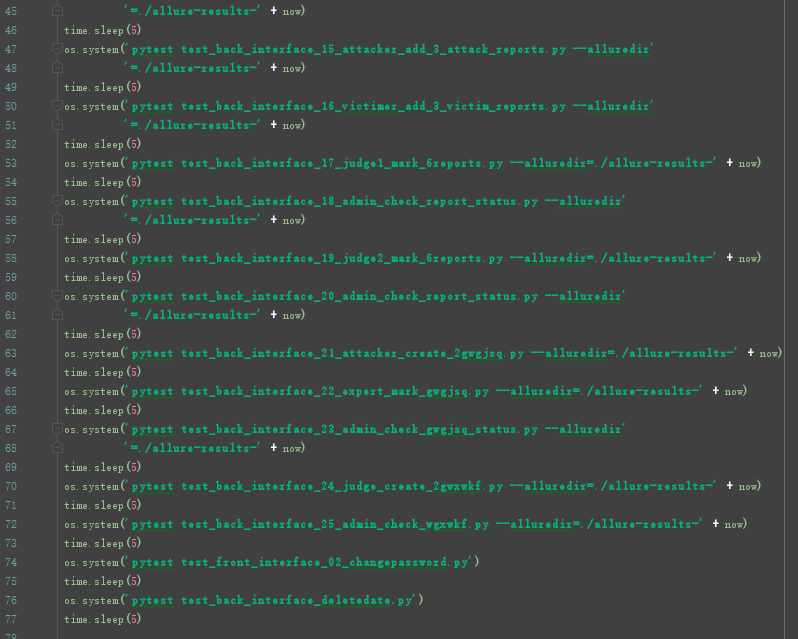


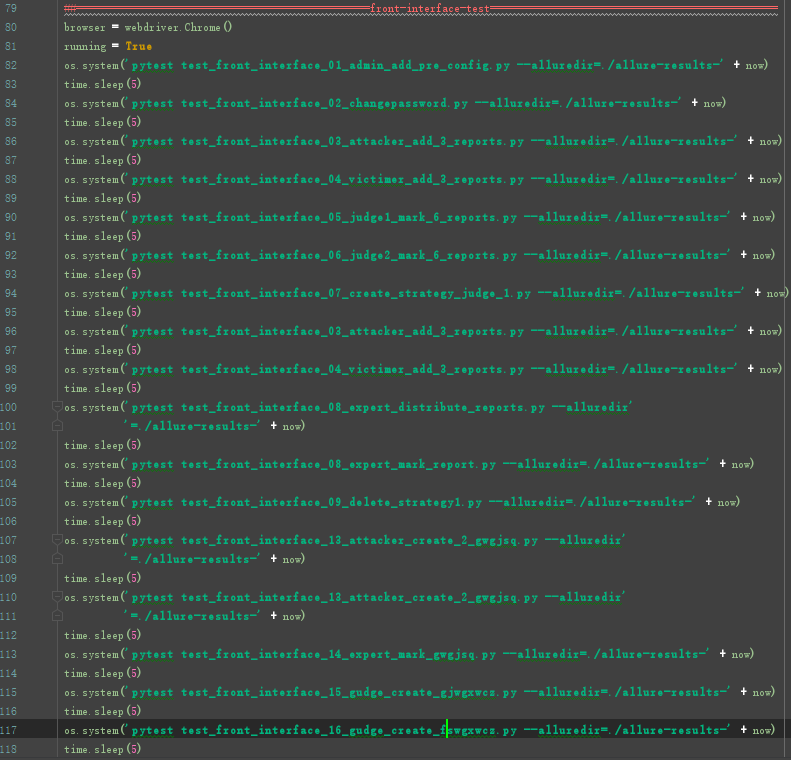
## 三、业务相关

框架实现了攻防演练平台系统的自动化测试，**其中系统核心业务流程（成绩状态流转、策略匹配等）由后端自动化代码保证，前端自动化主要保证能正常进行前端的业务正常提交**，详细见业务流程梳理。

实际运行过程中，执行逻辑为先运行后端自动化测试，然后删除数据，进行前端自动化测试，测试完成后删除测试数据，详见run脚本：







## 四、问题排查

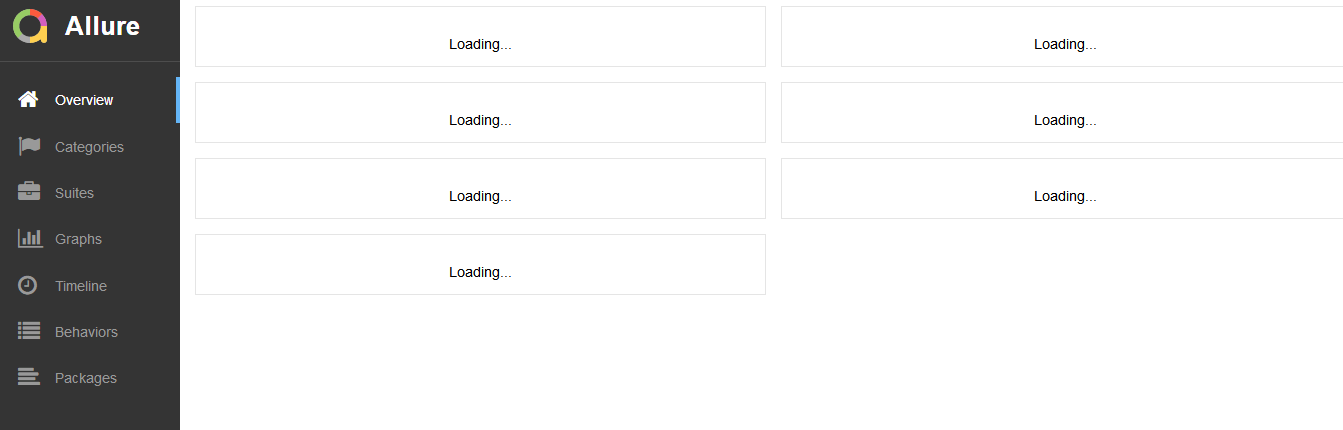
问题1：如何利用allure和pytest框架将pyresttest和python的执行结果进行可视化展示：

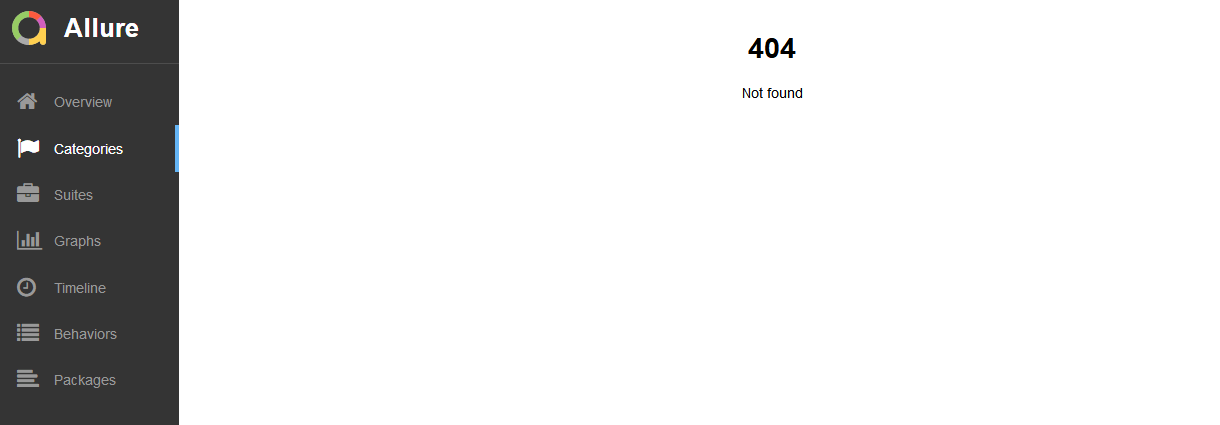
解决方案：pyresttest中执行的命令回显需要让windows获取到然后用回显进行断言。使用paramiko模块进行ssh连接虚拟机，执行命令获取回显，进行断言：



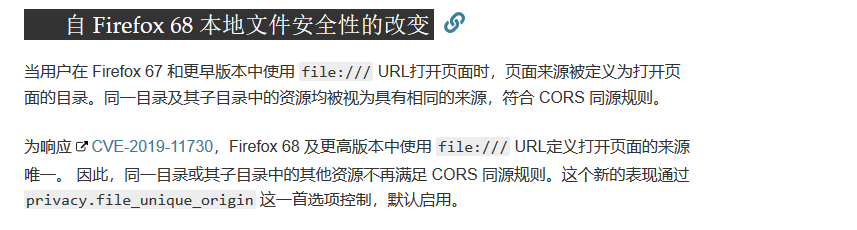
问题2：

使用火狐浏览器打开测试结果index.html，无法打开页面：





解决方案：禁用火狐浏览器同源策略，因为



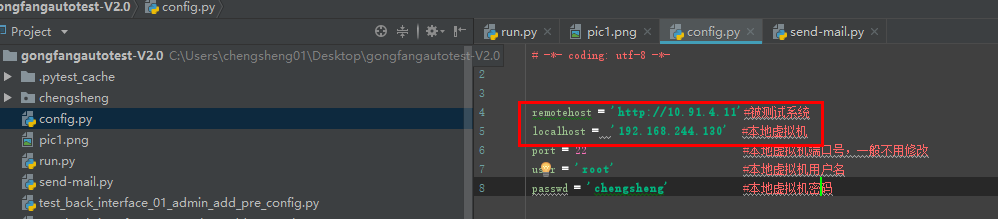
火狐浏览器-输入about:config；禁用策略

firefox: about:config 中 privacy.file\_unique\_origin属性false

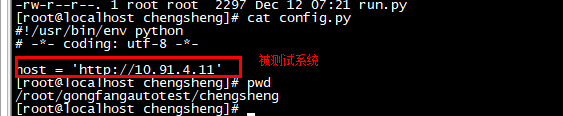
## 五、使用相关

实际使用根据不用的测试环境只需要更改config.py配置文件即可：

1、本地PC中的config.py脚本：



1. 本地虚拟机中的config.py脚本：



1. **参考链接：**

<https://www.jianshu.com/p/5ac37a6d7c05>

<https://blog.csdn.net/wust_lh/article/details/86685912>

<https://blog.csdn.net/qq_39721240/article/details/88876475>

<https://blog.csdn.net/baidu_30374343/article/details/90294518>

<https://github.com/allure-framework/allure2/releases>

<https://developer.mozilla.org/zh-CN/docs/Web/HTTP/CORS/Errors/CORSRequestNotHttp>

<http://repo.maven.apache.org/maven2/io/qameta/allure/allure-commandline/>

