安全测试着重测试功能实现问题。重点则在于测试前后端实现方式，逻辑，交互过程。测试用例是特殊符号，程序语言函数。

[http://127.0.0.1:60080/login.php](http://124.250.31.122:60080/login.php)

admin

talsec

## 暴力破解

场景：系统登录功能

原理：登录接口的登录次数没有限制，可以一直测试直到爆破出账号密码。

测试方法：用burp爆破。社工库。

防御：登录次数限制。

## 命令注入

127.0.0.1

127.0.0.1;ifconfig. && |

场景：一个调用系统命令实现的功能。

原理：功能实现调用了系统命令，参数没有限制与处理，当传入参数为系统/编程语言命令的时候，程序将命令执行。

测试方法：输入系统命令或功能函数

Reboot 命令会怎样？？？

防御：发现异常字符直接返回异常，不进行处理。

'&'    
';'    
'| '   
'-'    
'$'    
'('    
')'    
'`'    
'||'

## 跨站请求伪造

Low: http://47.52.233.199/csrflow.html

漏洞位置：用户改密，添加关注等功能。

原理：请求参数是固定的。

攻击场景：恶意链接或页面放到博客论坛等让你访问。

访问地址的原因： 社工。

测试方法：观察功能实现参数是否固定

防御：二次校验，不固定变量。

## 文件包含

本地文件包含：/etc/passwd 相对路径，绝对路径。../fi/file1.php

远程文件包含：<http://47.52.233.199/info.txt>。包含一个内网地址。

http://47.52.233.199/write.txt， 将一个木马文件写入服务器。

漏洞场景：系统有加载文件（上传，下载，读取，加载）的功能。本地或远程都可以。参数没有限制与处理，且用户可控。

原理： php中的include函数，会将include包含的任意文件当作php文件执行。所以可以用来执行恶意脚本。

测试方法：包含其他文件

防御：校验page字段的值。用白名单过滤。

## 文件上传

漏洞场景：上传功能

原理： 上传没有限制文件类型，可以上传脚本文件

测试方法：上传脚本文件

防御：服务器后端限制文件类型。

Low：直接上传

## SQL注入

漏洞场景：多见于数据库操作，即增删改查任意数据库中的信息的时候.

查学生课表，订单，身份信息等等

原理： 将用户输入当做sql语句执行

测试方法： 输入sql语句，尝试SQL注入

Payload: 1 ' or '1'='1'#

防御：过滤用户输入。

SELECT first\_name, last\_name FROM users WHERE user\_id = '输入内容';

SELECT first\_name, last\_name FROM users WHERE user\_id = '1' or '1'='1’#';

## XSS漏洞

漏洞场景：反馈功能，评论功能

原理： 用户输入被当成了JS代码执行

测试方法：XSSPAYLOAD

<script>alert(1);</script>

防御：过滤用户输入。

[http://47.9189:28666/xssplatform/index.php?do=login](http://47.91.156.89:28666/xssplatform/index.php?do=login" \t "_blank)