全国信安竞赛半决赛（华北赛区）部分题目题解

原创

buchiyexiao. 于 2019-06-09 17:38:17 发布 399 收藏 1

分类专栏： ctf 文章标签： writeup

本文链接：https://blog.csdn.net/qq_43647462/article/details/91355120

由于能力有限，本次题解仅包括第二个比赛日的WEB1和WEB2以及WEB2的修补思路

WEB1

首先我们直接可以发现我们的目的是在ctf表中查询flag这个字符串，也可以通过id=1和id=0发现可能是一个布尔注入，但是经过测试我们可以发现两个问题：

第一个，你的sql语句，大多数关键词union by and 都被加到了黑名单里面，如何构造sql语句是一个大问题。

第二个，你如何回显你所需要的内容，他回显的内容是固定的，因此如何得到flag也是一个问题

后面的思路是龙哥和风鹰侠的，我只是负责帮忙爆破

解决问题的关键在于黑名单中并没有sleep也并没有substr或者mid这种字符串操作的命令关键词，因此基于时间的BOOL类型SQL注入时可行的，因为flag的长度有限，因此我们可以首先去爆破flag的长度，也就是判断在第几位时flag对应的字符Ascii码大于0不成立，语句中$$为intruder爆破时的爆破字节

id=1^(if((ascii(substr((select(flag)from(ctf)),1,1))>0),sleep(5),1))

然后借助intruder得到flag的长度是16，其实自己手动爆破可能更快，通过看这个sleep的时间得到该位数的flag字符是否存在

然后我们借助二分法对flag的每一个字符进行爆破，此时手动爆破即可

id=1^(if((ascii(substr((select(flag)from(ctf)),x,1))>y),sleep(2),1))

为了方便观察速度把sleep改成了2，每次我们需要更改x的值为6-15，y为Ascii码，可以借助二分法进行爆破，如果认为可能会出现问题，把1-5和16爆破得出flag{和}可以确认其为flag，最终flag为flag{a0af676352}，虽然我们做出来这个题的时候已经关闭flag提交窗口八分钟了，但是借助Xshell连接上服务器得到flag与正确答案无误

WEB2

WEB2是很典型的代码审计，我们直接查看源码时发现注释里标有src，传入?src=1可以得到源码，进行代码审计

源码共有五个判断（忽略判断src那个，无影响），第一个判断每一个request传入的key的value，不能出现a-z和A-Z，但是以下我们可以知道，我们又不得不传入字母，因此这是我们需要绕过的第一步；第二个判断是正则匹配不能出现cyber，flag和ciscn，但是我们需要传入这三个，因此这是我们需要绕过的第二步；第三个是cyber的MD5判断，这是我们需要绕过的第三步；第四个是正则匹配ciscn中含有ciscnsec且ciscn不等于ciscnsec；第五个是flag调用file_get_contents读取的文件内容为security即可获得flag

第一个，我们post，get，cookie传的值全都变成了键值对，值不能有字母，但是该方法存在覆盖机制，也就是post传入的是数字然后get再传入相同名称的值是字母时，判断时会判断post请求。

第二个，借助url编码可以绕过

第三个，md5()和sha()的判断都可以借助数组进行绕过

第四个，虽然限制了开头结尾，但是没有/d可以通过ciscnsec%0a绕过

第五个，flag借助file_get_contents读取协议绕过

因此我们构造出payload：
cybe%72[]=null&c%69scn=c%69scnsec%0a&%66lag=data:text/plain;charset=unicode,security，然后POST传入三个参数均为数字即可得到flag

修补方法

在我进行修的时候首先考虑的是，我要不能让攻击者通过post可以绕过我的字母判断，即我通过获取是否存在post传入的cyber，flag和ciscn进行判断，将其加入至黑名单内，然后我考虑的是再借助正则判断一下security的存在（治标不治本，只是不知道写啥了凑数的），最后我还很缺德的为了防止MD5判断的绕过，禁用了[]符号即其Ascii码的判断

PS：当然，现场我并没有实现这么多，因为确实紧张加上第一次参加，我只是卡了一手post然后判断了security的存在，但是看分数的话，结果好像并无补成，如果有更好的方法，可以找我一起交流哈~