xctf攻防世界 MISC高手进阶区 Hidden-Message

原创

l8947943 于 2022-01-15 17:58:30 发布 5574 收藏

分类专栏： 攻防世界misc之路 文章标签： 安全 web安全 misc

本文链接：https://blog.csdn.net/l8947943/article/details/122513430

版权

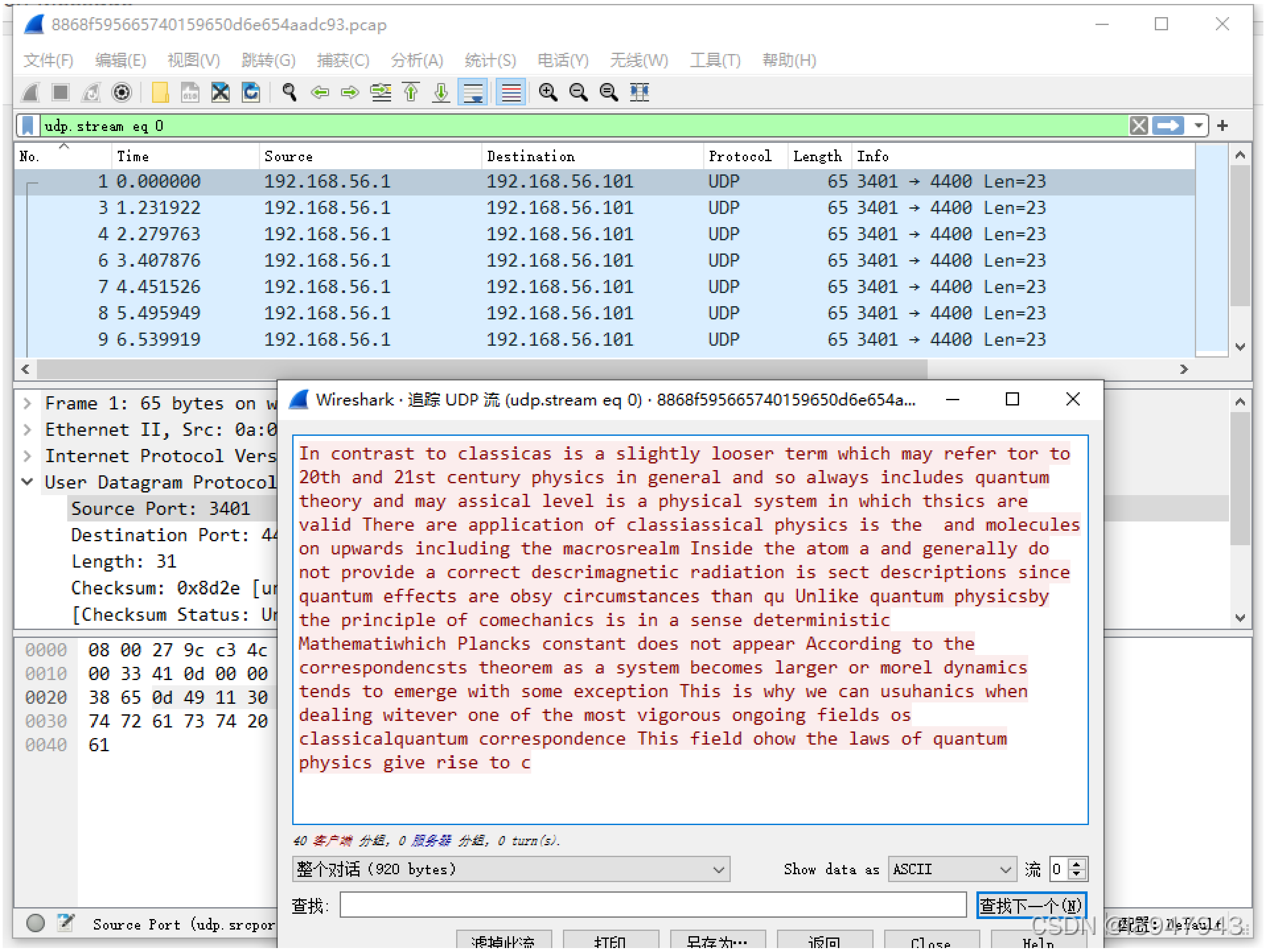
攻防世界misc之路 专栏收录该内容

68 篇文章 2 订阅

订阅专栏

1. 进入环境，下载附件

题目给的是pcap文件，果断用wireshark打开，如图：



8868f595665740159650d6e654aadc93.pcap

文件(F) 编辑(E) 视图(V) 跳转(G) 捕获(C) 分析(A) 统计(S) 电话(Y) 无线(W) 工具(T) 帮助(H)

udp.stream eq 0

```
No.   Time       Source          Destination       Protocol  Length  Info
    1 0.000000   192.168.56.1    192.168.56.101    UDP           65  3401 → 4400 Len=23
    3 1.231922   192.168.56.1    192.168.56.101    UDP           65  3401 → 4400 Len=23
    4 2.279763   192.168.56.1    192.168.56.101    UDP           65  3401 → 4400 Len=23
    6 3.407876   192.168.56.1    192.168.56.101    UDP           65  3401 → 4400 Len=23
    7 4.451526   192.168.56.1    192.168.56.101    UDP           65  3401 → 4400 Len=23
    8 5.495949   192.168.56.1    192.168.56.101    UDP           65  3401 → 4400 Len=23
    9 6.539919   192.168.56.1    192.168.56.101    UDP           65  3401 → 4400 Len=23
```

```
> Frame 1: 65 bytes on w
> Ethernet II, Src: 0a:0
> Internet Protocol Vers
v User Datagram Protocol
    Source Port: 3401
    Destination Port: 44
    Length: 31
    Checksum: 0x8d2e [ur
    [Checksum Status: Ur
```

```
0000  08 00 27 9c c3 4c
0010  00 33 41 0d 00 00
0020  38 65 0d 49 11 30
0030  74 72 61 73 74 20
0040  61
```

Wireshark · 追踪 UDP 流 (udp.stream eq 0) · 8868f595665740159650d6e654a...

```
In contrast to classicas is a slightly looser term which may refer tor to
20th and 21st century physics in general and so always includes quantum
theory and may assical level is a physical system in which thsics are
valid There are application of classiassical physics is the  and molecules
on upwards including the macrosrealm Inside the atom a and generally do
not provide a correct descrimagnetic radiation is sect descriptions since
quantum effects are obsy circumstances than qu Unlike quantum physicsby
the principle of comechanics is in a sense deterministic
Mathematiwhich Plancks constant does not appear According to the
correspondencsts theorem as a system becomes larger or morel dynamics
tends to emerge with some exception This is why we can usuhanics when
dealing witever one of the most vigorous ongoing fields os
classicalquantum correspondence This field ohow the laws of quantum
physics give rise to c
```

40 客户端 分组, 0 服务器 分组, 0 turn(s).

整个对话 (920 bytes)　Show data as ASCII　流 0

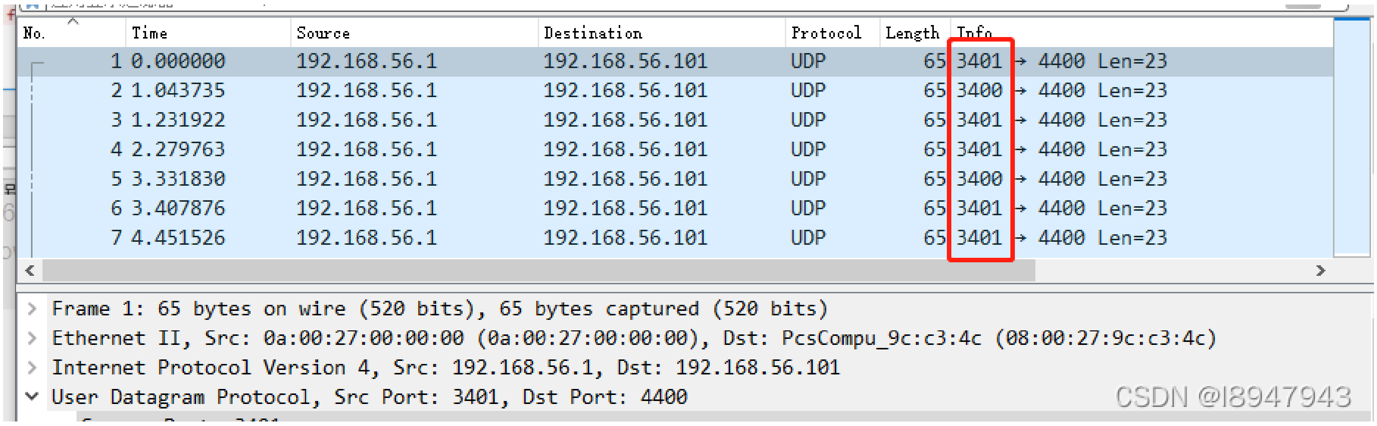
查找：

追踪流也没什么隐含的信息。

2. 问题分析

观察抓包数据

Source、Destination、Protocol、Length都是一样的，Time列看不出什么规律，就生了Info了，srcport不断反复横跳，desport一成不变，Len都是23，如图：



No.	Time	Source	Destination	Protocol	Length	Info
1	0.000000	192.168.56.1	192.168.56.101	UDP	65	3401 → 4400 Len=23
2	1.043735	192.168.56.1	192.168.56.101	UDP	65	3400 → 4400 Len=23
3	1.231922	192.168.56.1	192.168.56.101	UDP	65	3401 → 4400 Len=23
4	2.279763	192.168.56.1	192.168.56.101	UDP	65	3401 → 4400 Len=23
5	3.331830	192.168.56.1	192.168.56.101	UDP	65	3400 → 4400 Len=23
6	3.407876	192.168.56.1	192.168.56.101	UDP	65	3401 → 4400 Len=23
7	4.451526	192.168.56.1	192.168.56.101	UDP	65	3401 → 4400 Len=23

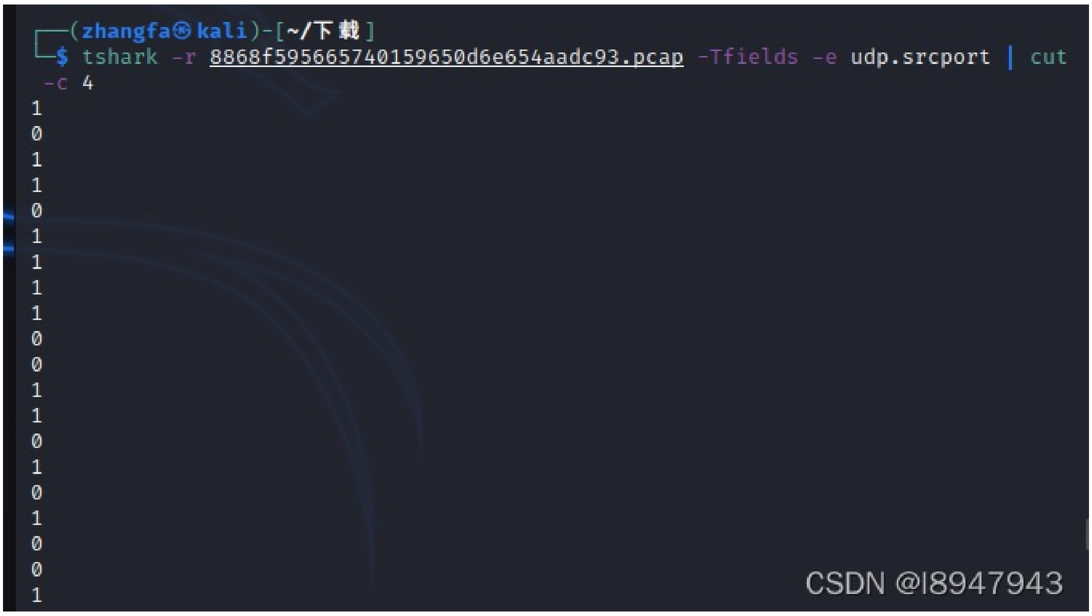
```
> Frame 1: 65 bytes on wire (520 bits), 65 bytes captured (520 bits)
> Ethernet II, Src: 0a:00:27:00:00:00 (0a:00:27:00:00:00), Dst: PcsCompu_9c:c3:4c (08:00:27:9c:c3:4c)
> Internet Protocol Version 4, Src: 192.168.56.1, Dst: 192.168.56.101
v User Datagram Protocol, Src Port: 3401, Dst Port: 4400
```

CSDN @l8947943

猜想1->0->1的变化可能代表某种信息。

使用kali提取信息

```
# 使用tshark过滤出源端口，使用cut裁取端口的最后一位
tshark -r 8868f595665740159650d6e654aadc93.pcap -Tfields -e udp.srcport | cut -c 4
```



```
┌──(zhangfa㉿kali)-[~/下载]
└─$ tshark -r 8868f595665740159650d6e654aadc93.pcap -Tfields -e udp.srcport | cut
 -c 4
1
0
1
1
0
1
1
1
1
0
0
1
1
0
1
0
1
0
0
1
```

CSDN @l8947943

得到最终的数据为：

> 10110111100110101001011010001100100110101001000110011101100110101000110110011000

3. 字符转ASCII

```python
s = "10110111100110101001011010001100100110101001000110011101100110101000110110011000"

flag = ''
for i in range(len(s)):
    if s[i] == '0':
        flag += '1'
    else:
        flag += '0'

print(flag)

# 原始字符串翻译
print(''.join(chr(int(s[i : i + 8], 2)) for i in range(0, len(s), 8)))
# 取反码字符串翻译
print(''.join(chr(int(flag[i : i + 8], 2)) for i in range(0, len(flag), 8)))
```

得到最终的答案：`Heisenberg`