Subject: OpenVZ, Bind and stalling TCP connections.
Posted by agl241 on Wed, 24 Sep 2008 16:10:53 GMT
View Forum Message <> Reply to Message

Hi

After migration of our DNS servers to OpenVZ we have noticed that on one of our DNS server zone transfer stops working
after few our after starting. New TCP connections are created
but after initial handshake no traffic is sent by server to client (e.g. client command: host -l domain.pl) - so zone transfer doesn't work anymore. The one solution is to restart the server. After tcpdumping the stalling connections I've noticed that client AXFR requests reaches the server, but the server doesn't ack received packets, it continously sends SYN,ACK packets. Isn't it the symptom of overruning aplication TCP receive buffer? All other connections doesn't stop to run.

0:22:06.784637 IP CLIENT.56555 > SERVER.53: S 259415162:259415162(0) win 5840 <mss 1460,sackOK,timestamp 2309875204 0,nop,wscale 7>
10:22:06.784694 IP SERVER.53 > CLIENT.56555: S 3981433138:3981433138(0) ack 259415163 win 5792 <mss 1460,sackOK,timestamp 72583240 2309875204,nop,wscale 4>
10:22:06.784876 IP CLIENT.56555 > SERVER.53: . ack 1 win 46 <nop,nop,timestamp 2309875204 72583240>
10:22:06.784879 IP CLIENT.56555 > SERVER.53: P 1:29(28) ack 1 win 46 <nop,nop,timestamp 2309875204 72583240> 21415 AXFR? atman.pl. (26)
10:22:06.986488 IP CLIENT.56555 > SERVER.53: P 1:29(28) ack 1 win 46 <nop,nop,timestamp 2309875405 72583240> 21415 AXFR? atman.pl. (26)
10:22:07.389585 IP CLIENT.56555 > SERVER.53: P 1:29(28) ack 1 win 46 <nop,nop,timestamp 2309875808 72583240> 21415 AXFR? atman.pl. (26)
10:22:08.193482 IP CLIENT.56555 > SERVER.53: P 1:29(28) ack 1 win 46 <nop,nop,timestamp 2309876612 72583240> 21415 AXFR? atman.pl. (26)
10:22:09.801451 IP CLIENT.56555 > SERVER.53: P 1:29(28) ack 1 win 46 <nop,nop,timestamp 2309878220 72583240> 21415 AXFR? atman.pl. (26)
10:22:10.311252 IP SERVER.53 > CLIENT.56555: S 3981433138:3981433138(0) ack 259415163 win 5792 <mss 1460,sackOK,timestamp 72586767 2309878220,nop,wscale 4>
10:22:10.311502 IP CLIENT.56555 > SERVER.53: . ack 1 win 46 <nop,nop,timestamp 2309878730 72586767,nop,nop,sack 1 {0:1}>

The system is Debian Etch , Bind version: 9.3.4-2etch3 (standard package).
The same problem appears both with newest kernel from
 openvz.org debian-systs repository (v. 028stab053.dso2) and vanilla linux kernel patched with 2.6.18-028stab056 openvz.

Looking forward for your advice.

--
Andrzej Lemieszek

Subject: Re: OpenVZ, Bind and stalling TCP connections.
Posted by maratrus on Thu, 25 Sep 2008 05:51:56 GMT
View Forum Message <> Reply to Message

Hello,

anything in "/proc/user_beancounters"?

Subject: Re: OpenVZ, Bind and stalling TCP connections.
Posted by agl241 on Thu, 25 Sep 2008 10:43:25 GMT
View Forum Message <> Reply to Message

Nothing unusual - limits are not reached.

Here is my UBC:

```
uid  resource          held    maxheld    barrier      limit    failcnt
  2037:  kmemsize      10245132   14460186  184857600  184857600        0
         lockedpages         0          0        256        256         0
         privvmpages    219769     224006    1048576    1114112         0
         shmpages         1280       2576      21504      21504         0
         dummy               0          0          0          0         0
         numproc            21         37        240        240         0
         physspages     205265     206941          0 2147483647         0
         vmguarpages         0          0     311209 2147483647         0
         oomguarpages   205265     206941    1671168 2147483647         0
         numtcpsock         14         85        360        360         0
         numflock            1          3        188        206         0
         numpty              4          6         16         16         0
         numsiginfo          0          3        256        256         0
         tcpsndbuf      136152     495872    1720320    2703360         0
         tcprcvbuf      229376     873920    6881280   10813440         0
         othersockbuf   426312    2237016   72069120 1254338560         0
         dgramrcvbuf         0     510816    8388608    8388608         0
         numothersock      193       1004     461824     461824         0
         dcachesize          0          0  177484800  177484800         0
         numfile         39294      58799     462200     462200         0
         dummy               0          0          0          0         0
         dummy               0          0          0          0         0
         dummy               0          0          0          0         0
         numiptent          10         10        128        128         0
```

Subject: Re: OpenVZ, Bind and stalling TCP connections.

Posted by maratrus on Fri, 26 Sep 2008 11:03:40 GMT
View Forum Message <> Reply to Message

Hi,

here is the note I was suggested to pay your attention to:

your server sent the SYN,ACK packet with the same ISN twice.
Quote:

10:22:06.784637 IP CLIENT.56555 > SERVER.53: S 259415162:259415162(0) win 5840 <mss 1460,sackOK,timestamp 2309875204 0,nop,wscale 7>

HERE >>>>>10:22:06.784694 IP SERVER.53 > CLIENT.56555: S 3981433138:3981433138(0) ack 259415163 win 5792 <mss 1460,sackOK,timestamp 72583240 2309875204,nop,wscale 4>

10:22:06.784876 IP CLIENT.56555 > SERVER.53: . ack 1 win 46 <nop,nop,timestamp 2309875204 72583240>
.....
HERE >>>>>10:22:10.311252 IP SERVER.53 > CLIENT.56555: S 3981433138:3981433138(0) ack 259415163 win 5792 <mss 1460,sackOK,timestamp 72586767 2309878220,nop,wscale 4>
10:22:10.311502 IP CLIENT.56555 > SERVER.53: . ack 1 win 46 <nop,nop,timestamp 2309878730 72586767,nop,nop,sack 1 {0:1}>

So, we can deduced that the following packet didn't receive your server.
Quote:

10:22:06.784876 IP CLIENT.56555 > SERVER.53: . ack 1 win 46 <nop,nop,timestamp 2309875204 72583240>

So, could you please tcpdump connections at four places simultaneously:

- on client side (common network device)
- on the HN (common network device)
- on the HN (appropriate veth/venet device)
- indide the VE (appropriate net device)

And also please check your netfilters

By the way, was your tcpdump output from the first post done from inside the VE or from where?

Subject: Re: OpenVZ, Bind and stalling TCP connections.
Posted by agl241 on Fri, 26 Sep 2008 15:25:03 GMT

Thanks for your response.

My previous tcpdump output was produced from inside of the container (tcpdumps on the hardware node itself produce similar results).

After further investigation and trying to trace Bind I've found that indeed named process listening on the TCP port does not receive any notification about incoming connections when it is locked.

At the same time I've found that number of listen (accept) queue overflows inside the VE is very high (for the 1 day uptime the appriprate numbers in /proc/net/netstat was: ListenOverflows - 95584, ListenDrops - 95584) - this counter is increased each time incoming ACK packet (last from the 3-way handshake) is received, but accept queue for the listening socket is full, so connection is silenly dropped.

My hypothesis is that for some unknown reason accept queue of the
listening Bind socket is filled up and later it can't recover itself. No further incoming connections to this port can reach ESTABLISHED state , and on way to recover named is to restart it, so it allocates new listening socket. I think that locking wasn't the result of the sync flooding, because single syncs without later acknowledgment don't reach this fragment of code (tcp_v4_syn_recv_sock).

At the moment I can't reproduce error, so it makes further debugging difficult - but I'm sure that it will appear soon again.

Regards.

Andrzej Lemieszek

Below are little formated contents of my "/proc/net/netstat":

1: SyncookiesSent - 0
2: SyncookiesRecv - 0
3: SyncookiesFailed - 0
4: EmbryonicRsts - 60553
5: PruneCalled - 0
6: RcvPruned - 0
7: OfoPruned - 0
8: OutOfWindowIcmps - 1
9: LockDroppedIcmps - 0
10: ArpFilter - 0
11: TW - 0
12: TWRecycled - 0
13: TWKilled - 0
14: PAWSPassive - 0
15: PAWSActive - 0
16: PAWSEstab - 2
17: DelayedACKs - 22483

18: DelayedACKLocked - 306
19: DelayedACKLost - 125
20: ListenOverflows - 95584
21: ListenDrops - 95584
22: TCPPrequeued - 89
23: TCPDirectCopyFromBacklog - 0
24: TCPDirectCopyFromPrequeue - 40
25: TCPPrequeueDropped - 0
26: TCPHPHits - 290407
27: TCPHPHitsToUser - 0
28: TCPPureAcks - 9559422
29: TCPHPAcks - 6528630
30: TCPRenoRecovery - 1
31: TCPSackRecovery - 387
32: TCPSACKReneging - 0
33: TCPFACKReorder - 2
34: TCPSACKReorder - 5
35: TCPRenoReorder - 1
36: TCPTSReorder - 46
37: TCPFullUndo - 35
38: TCPPartialUndo - 1491
39: TCPDSACKUndo - 0
40: TCPLossUndo - 251
41: TCPLoss - 396
42: TCPLostRetransmit - 0
43: TCPRenoFailures - 0
44: TCPSackFailures - 5559
45: TCPLossFailures - 2396
46: TCPFastRetrans - 648
47: TCPForwardRetrans - 64
48: TCPSlowStartRetrans - 19419
49: TCPTimeouts - 14825
50: TCPRenoRecoveryFail - 0
51: TCPSackRecoveryFail - 82
52: TCPSchedulerFailed - 0
53: TCPRcvCollapsed - 0
54: TCPDSACKOldSent - 124
55: TCPDSACKOfoSent - 0
56: TCPDSACKRecv - 383
57: TCPDSACKOfoRecv - 0
58: TCPAbortOnSyn - 0
59: TCPAbortOnData - 44
60: TCPAbortOnClose - 32
61: TCPAbortOnMemory - 0
62: TCPAbortOnTimeout - 6259
63: TCPAbortOnLinger - 0
64: TCPAbortFailed - 0
65: TCPMemoryPressures - 0