MariaDB Server MDEV-26420

Buffer overflow on instant ADD/DROP of generated column

Details

Type:	Bug
Status:	CLOSED (View Workflow)
Priority:	Blocker
Resolution:	Fixed
Affects Version/s:	10.4, 10.5, 10.6, 10.7
Fix Version/s:	10.4.26, 10.5.17, 10.6.9, (4)
Component/s:	Data Definition - Alter Table, (1) Storage Engine - InnoDB
Labels:	ASAN crash regression-10.4
Environment:	Linux version 5.13.0-1-MANJARO (builduser@LEGION) (gcc (GCC) 11.1.0, GNU ld (GNU Binutils) 2.36.1) #1 SMP PREEMPT Mon Jun 7 06:16:10 UTC 2021 x86_64

Description

PoC:

```
CREATE TABLE v0 ( v1 TIME NOT NULL PRIMARY KEY ) ;
 ALTER TABLE v0 ADD COLUMN v0 INT GENERATED ALWAYS AS ( lpad ( 'x' , NULL = 32 , 'x
 SHOW LOCAL STATUS WHERE COALESCE ( 27 , 51 - 39 ) = 'x' ;
 DELETE FROM v0 WHERE 44707452.000000 ;
 ALTER TABLE v0 ADD COLUMN v0 INT GENERATED ALWAYS AS ( v1 + v1 ) , DROP COLUMN v0
 SELECT COUNT ( * ) FROM v0 WHERE v1 = -128 AND v1 = 'x' ;
```

Log and Asan report:

```
2021-08-16 14:41:38 0 [Note] InnoDB: Compressed tables use zlib 1.2.11
2021-08-16 14:41:38 0 [Note] InnoDB: Number of pools: 1
2021-08-16 14:41:38 0 [Note] InnoDB: Using crc32 + pclmulqdq instructions
2021-08-16 14:41:38 0 [Note] mysqld: O_TMPFILE is not supported on /tmp (disabl
2021-08-16 14:41:38 0 [Note] InnoDB: Using liburing
2021-08-16 14:41:38 0 [Note] InnoDB: Initializing buffer pool, total size = 134
2021-08-16 14:41:38 0 [Note] InnoDB: Completed initialization of buffer pool
2021-08-16 14:41:38 0 [Note] InnoDB: 128 rollback segments are active.
2021-08-16 14:41:38 0 [Note] InnoDB: Creating shared tablespace for temporary t
2021-08-16 14:41:38 0 [Note] InnoDB: Setting file './ibtmp1' size to 12 MB. Phy
2021-08-16 14:41:38 0 [Note] InnoDB: File './ibtmp1' size is now 12 MB.
```

```
2021-08-16 14:41:38 0 [Note] InnoDB: 10.7.0 started; log sequence number 42161;
2021-08-16 14:41:38 0 [Note] InnoDB: Loading buffer pool(s) from /home/fuboat/m
2021-08-16 14:41:38 0 [Note] Plugin 'FEEDBACK' is disabled.
2021-08-16 14:41:38 0 [Note] InnoDB: Buffer pool(s) load completed at 210816 14
2021-08-16 14:41:38 0 [Note] Server socket created on IP: '0.0.0.0'.
2021-08-16 14:41:38 0 [Note] Server socket created on IP: '::'.
2021-08-16 14:41:38 0 [Note] /usr/local/mysql/bin//mysqld: ready for connection
Version: '10.7.0-MariaDB'  socket: '/tmp/0.socket'  port: 3306  Source distribu
```

Issue Links

is caused by

MDEV-15562 Instant DROP COLUMN or changing the order of columns — CLOSED

links to

CVE-2022-32081

Activity

Alice Sherepa added a comment - 2021-08-26 16:08

Thank you! Reproducible on 10.4-10.6:

```
--source include/have_innodb.inc

CREATE TABLE t1 (i int AS ('x') stored, j int) engine=innodb;
ALTER TABLE t1 ADD COLUMN i INT GENERATED ALWAYS AS ('a'), DROP COLUMN i ;
```

10.4 dc6bc85cd29586631d

```
Version: '10.4.22-MariaDB-debug-log'
================================================================
==620710==ERROR: AddressSanitizer: use-after-poison on address 0x6190000f3
WRITE of size 56 at 0x6190000f3998 thread T27
    #0 0x5623146114a0 in prepare_inplace_add_virtual /10.4/src/storage/inn



    #1 0x5623146199c0 in prepare_inplace_alter_table_dict /10.4/src/storag
    #2 0x56231462eafc in ha_innobase::prepare_inplace_alter_table(TABLE*,
    #3 0x562313fc4575 in handler::ha_prepare_inplace_alter_table(TABLE*, A
    #4 0x562313a01860 in mysql_inplace_alter_table /10.4/src/sql/sql_table
    #5 0x562313a136d2 in mysql_alter_table(THD*, st_mysql_const_lex_string
    #6 0x562313b97587 in Sql_cmd_alter_table::execute(THD*) /10.4/src/sql/
```

10.4 dc6bc85cd29586631d

```
#7 0x562313788a52 in mysql_execute_command(THD*) /10.4/src/sql/sql_par
#8 0x5623137b22d6 in mysql_parse(THD*, char*, unsigned int, Parser_sta
#9 0x562313788a52 in dispatch_command(enum_server_command, THD*, char*
#10 0x5623137854cf in do_command(THD*) /10.4/src/sql/sql_parse.cc:1373
#11 0x562313b7eb38 in do_handle_one_connection(CONNECT*) /10.4/src/sql
```

Marko Mäkelä added a comment - 2022-07-26 13:19

For Alice Sherepa's test case, the memory had been poisoned here:

10.4 8911823f65a6557ce66ea5f8aecd55b115a85606

```
#3  0x0000562765feda32 in mem_heap_create_func (size=<optimized out>,
    size@entry=1024,
    file_name=file_name@entry=0x562766ef84c0 "/mariadb/10.4/storage/innobase/h
    at /mariadb/10.4/storage/innobase/include/mem0mem.inl:375
#4  0x000056276600a835 in ha_innobase::prepare_inplace_alter_table (
    this=0x61d000247ca8, altered_table=<optimized out>,
    ha_alter_info=<optimized out>)
    at /mariadb/10.4/storage/innobase/handler/handler0alter.cc:7964
#5  0x000056276589c741 in handler::ha_prepare_inplace_alter_table (
    this=0x61d000247ca8, altered_table=altered_table@entry=0x7f6ba9085510,
    ha_alter_info=ha_alter_info@entry=0x7f6ba90837a0)
    at /mariadb/10.4/sql/handler.cc:4637
```

So, it looks like an out-of-bounds access.

Marko Mäkelä added a comment - 2022-07-26 13:34

The out-of-bounds access occurs here:

```
                new (&ctx->add_vcol[j]) dict_v_col_t();
```

The `j=0` is out-of-bounds because `ctx->num_to_add_vcol=0` (instead of the correct value 1). The math goes wrong in `prepare_inplace_add_virtual()`:

```
static
bool
prepare_inplace_add_virtual(
        Alter_inplace_info*     ha_alter_info,
        const TABLE*            altered_table,
        const TABLE*            table)
{
        ha_innobase_inplace_ctx*        ctx;
```

```
ulint                                   i = 0;
ulint                                   j = 0;

ctx = static_cast<ha_innobase_inplace_ctx*>
        (ha_alter_info->handler_ctx);

ctx->num_to_add_vcol = altered_table->s->virtual_fields
        + ctx->num_to_drop_vcol - table->s->virtual_fields;
```

Here, we have both `altered_table->s->virtual_fields` and `table->s->virtual_fields` equal to 1. InnoDB really cares about generated columns that are not stored in the clustered index, so it looks like we must iterate through all columns to see for which ones `Field::stored_in_db()` holds.

The bug does not affect 10.3. For newer servers a work-around (to disable ~~MDEV-15562~~) is `innodb_instant_alter_column_allowed=add_last`.

Marko Mäkelä added a comment - 2022-07-26 13:41

The simple fix is to over-estimate the size of the buffer as `altered_table->s->virtual_fields + ctx->num_to_drop_vcol` (not subtract any pre-existing generated columns), and to assign `j` at the end of the function to `ctx->num_to_add_vcol`.

People

Assignee:

Marko Mäkelä

Reporter:

Zhiyong Wu

Votes:

0 Vote for this issue

Watchers:

4 Start watching this issue

Dates

Created:

2021-08-19 03:09

Updated:

2022-07-26 14:21

Resolved:

2022-07-26 14:21

Git Integration

Error rendering 'com.xiplink.jira.git.jira_git_plugin:git-issue-webpanel'. Please contact your Jira administrators.