**SUMMARY**

A concurrency bug in MySQL-5.0.22

**DETAILS**

Some details can also be found at: <http://bugs.mysql.com/bug.php?id=19938>

This bug is due to a data race.

Thread1이 handle\_slave\_sql()에서 mi를 사용하는 동안 thread2가 end\_slave에서 mi를 null로 설정가능.

|  |  |
| --- | --- |
| Thread1 (sql/slave.cpp) | Thread2 (sql/slave.cpp) |
| pthread\_handler\_t  handle\_slave\_sql(void \*arg)  {  …  RELAY\_LOG\_INFO\* rli = &((MASTER\_INFO\*)arg)->rli;  ..  // tell the world we are done  pthread\_mutex\_unlock(&rli->run\_lock);  #ifndef DBUG\_OFF  // TODO: reconsider the code below  if (abort\_slave\_event\_count && !rli->events\_till\_abort)  goto slave\_begin;  #endif  my\_thread\_end();  pthread\_exit(0);  DBUG\_RETURN(0); // Can't return anything here  } | void end\_slave()  {  pthread\_mutex\_lock(&LOCK\_active\_mi);  if (active\_mi)  {  .  .  delete active\_mi;  active\_mi= 0;  }  pthread\_mutex\_unlock(&LOCK\_active\_mi);  } |
| 0-lock | 1-lock |