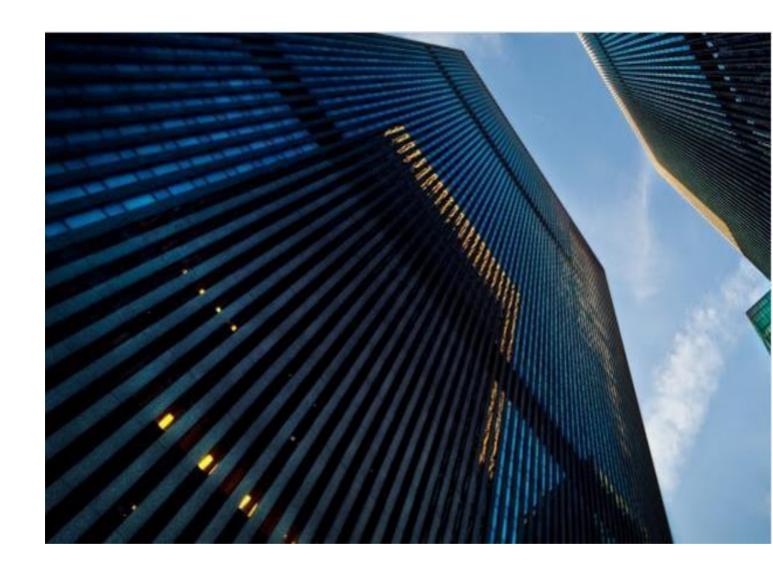
# Advanced Information Security

# Corporation



10/07/2015

Advanced Information Security Corporation Security Advisory Report

## MySQL Database 5.6.x (LATEST) Security Report

## Software Security Notification

**Software: Oracle's MySQL v.5.6.24 (LATEST)** 

#### **Vulnerability:**

(11) Buffer Overflow Vulnerabilities / ~ Deprecated & Insecure Function use (Missing Bounds-checks)

#### Software Overview

MySQL is an open source relational database management system (RDBMS), and the world's most popular Open Source database. MySQL is a popular choice of database for use in web applications, and is a central component of the widely used LAMP open source web application software stack. MySQL is used in many high profile large-websites worldwide. MySQL was created by a Swedish company, MySQL AB founded by David Axmark, Allan Larsson and Michael "Monty" Widenius. The first version of the software appeared on 23 of May, 1995. Oracle Corporation acquired Sun Microsystems in April, 2009 and are now the owners of MySQL Copyright and Trademark.

## Summary

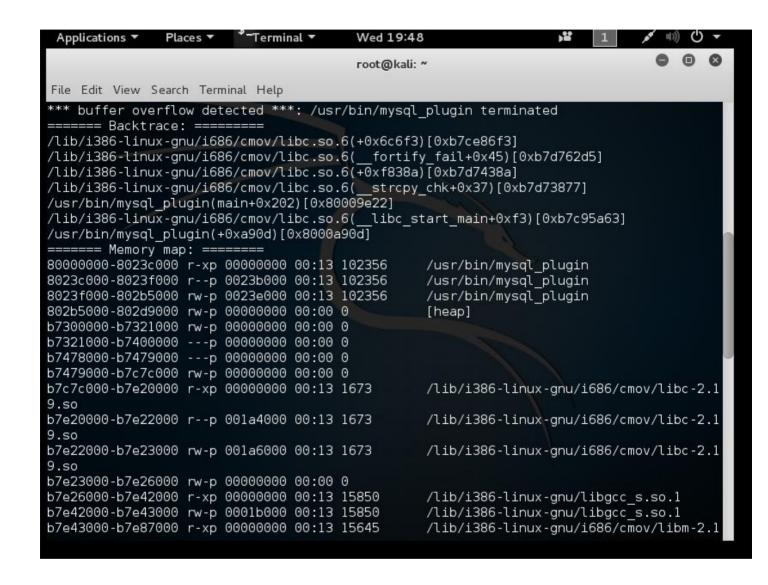
During a manual source-code audit of Oracle's MySQL v 5.6.24 database; conducted internally by the Advanced Information Security Group, instances of insecure function calls were observed in the software. The issues stem from the lack of any manual control metrics, which would prevent data from being overwritten into other sensitive locations.

```
793
          /* read the plugin config file and check for match against argument */
794
          else
795
          {
796
            strcpy(plugin name, argv[i]);
797
            strcpy(config file, argv[i]);
798
            strcat(config file, ".ini");
799
          }
800
801
```

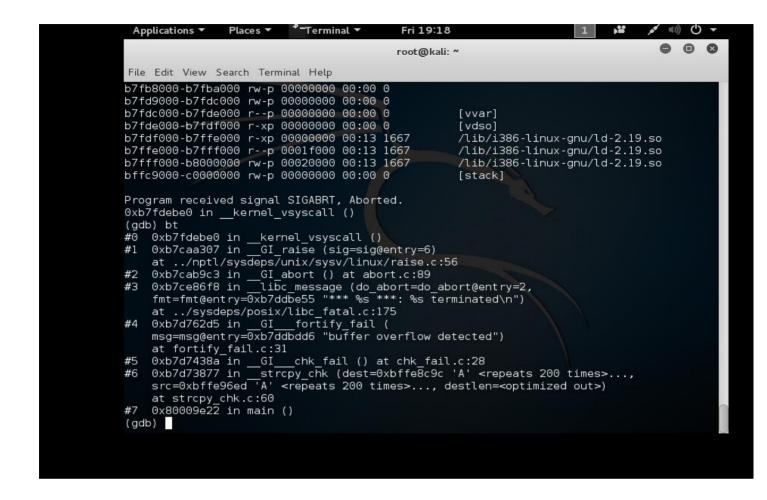
## Description

Unsafe use of the **strcpy**() function, has been triggered resulting in a buffer overflow condition. Therefore, in the aforementioned experiment input is copied from the command-line, to a fixed length destination buffer. No bounds checks are provided to ensure that the source does not exceed in size, and therefore would not overwrite the destination buffer.

#### **Technical Details**



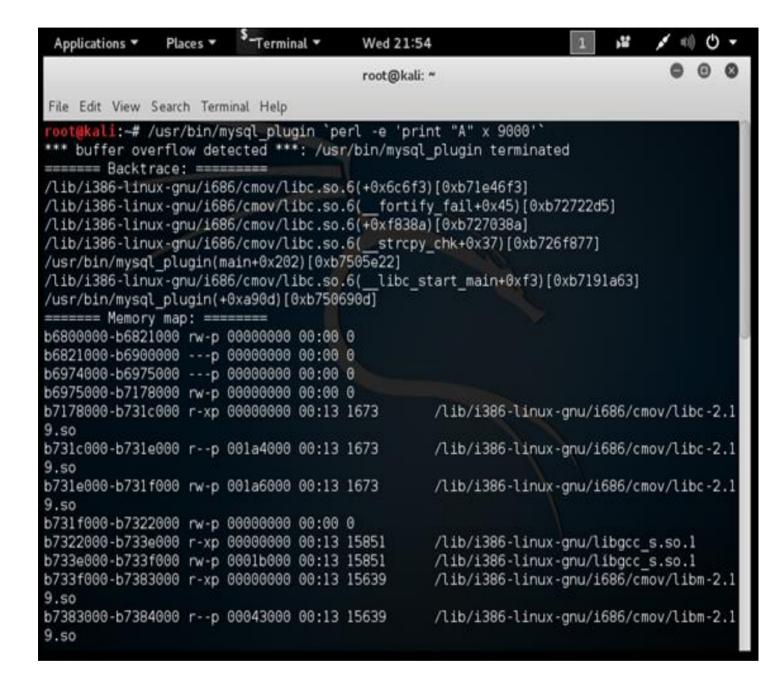
#### **Technical Details**



#### **Technical Details**

```
(gdb) disas
Dump of assembler code for function
                                     kernel vsyscall:
   0xb7fdebd0 <+0>:
                                %ecx
                         push
   0xb7fdebd1 <+1>:
                                %edx
                         push
                                %ebp
   0xb7fdebd2 <+2>:
                         push
   0xb7fdebd3 <+3>:
                                %esp,%ebp
                         mov
   0xb7fdebd5 <+5>:
                         sysenter
   0xb7fdebd7 <+7>:
                         nop
   0xb7fdebd8 <+8>:
                         nop
   0xb7fdebd9 <+9>:
                         nop
   0xb7fdebda <+10>:
                         nop
   0xb7fdebdb <+11>:
                         nop
   0xb7fdebdc <+12>:
                         nop
   0xb7fdebdd <+13>:
                         nop
   0xb7fdebde <+14>:
                                 $0x80
                         int
=> 0xb7fdebe0 <+16>:
                                %ebp
                         pop
   0xb7fdebe1 <+17>:
                                %edx
                         pop
   0xb7fdebe2 <+18>:
                                %ecx
                         pop
   0xb7fdebe3 <+19>:
                         ret
End of assembler dump.
(gdb)
```

### Proof of Concept Exploit - MySQL v5.6.24



## **Technical Synopsis**

The cause of the issue is an unsafe strcpy call; which can lead to a buffer overflow condition. A user-supplied string from the command-line is copied to a fixed length destination buffer, and the vulnerable function call is demonstrated below in red.

```
/lib/i386-linux-gnu/i686/cmov/libc.so.6(+0x6c6f3)[0xb7le46f3]
/lib/i386-linux-gnu/i686/cmov/libc.so.6(__fortify_fail+0x45)[0xb72722d5]
/lib/i386-linux-gnu/i686/cmov/libc.so.6(+0xf838a)[0xb727038a]
/lib/i386-linux-gnu/i686/cmov/libc.so.6(__strcpy_chk+0x37)[0xb726f877]
/usr/bin/mysql_plugin(main+0x202)[0xb7505e22]
/lib/i386-linux-gnu/i686/cmov/libc.so.6(__libc_start_main+0xf3)[0xb7191a63]
/usr/bin/mysql_plugin(+0xa90d)[0xb750690d]
```

Source Code at Line: 796
File: (../mysql/mysql-5.6.24/client/mysql plugin.c)

## MYSQL v5.6.24 Vulnerability List

I. Main.c (../mysql/mysql-5.6.24/regex/main.c:577)

```
char *name;
572
573
574
              static char efbuf[100];
575
              my_regex_t re;
576
577
              sprintf(efbuf, "MY REG %s", name);
              assert(strlen(efbuf) < sizeof(efbuf));
578
579
              re.re endp = efbuf;
580
              (void) my regerror (MY REG ATOI, &re, efbuf, sizeof (efbuf));
581
              return(atoi(efbuf));
582
      }
```

**Description:** A char\* type is copied to a fixed length destination buffer. This could lead to a buffer overflow.

# 2. Code Snippet – mysql\_plugin.c (../mysql/mysql-5.6.24/client/mysql\_plugin.c:796)

**Description:** Unsafe use of strcpy could lead to an overflow condition. A user-supplied string from the command-line is copied to a fixed length destination buffer.

3. Code Snippet – mysql\_plugin.c (../mysql/mysql-5.6.24/client/mysql\_plugin.c:797)

**Description:** Unsafe Use of strepy could lead to an overflow condition. A user-supplied string from the command-line is copied to a fixed length destination buffer. This could lead to a buffer overflow.

4. Code Snippet – main.c (../mysql/mysql-5.6.24/regex/main.c:544)

**Description:** A char\* type is being copied to a fixed length destination buffer. This could lead to a buffer overflow.

**Description:** Insecure sprintf. A char\* type is being copied to a fixed length destination buffer. This could lead to a buffer overflow.

6. Code Snippet – reader.cpp (../mysql/mysql-5.6.24/storage/ndb/src/kernel/blocks/dblqh/redoLogReader/reader.cpp:413)

```
406  void readArguments(int argc, const char** argv)
407  {
408   if(argc < 2 || argc > 9) {
409       usage(argv[0]);
410       doExit();
411   }
412
413   strcpy(fileName, argv[1]);
```

**Description:** Unsafe use of strcpy could lead to an overflow condition. A user-supplied string from the command-line is written to a fixed length destination buffer. This could lead to a buffer overflow if the input provided, is of greater size than the destination buffer.

7. Code Snippet – main.c (../mysql/mysql-5.6.24/regex/main.c:531)

**Description:** Unsafe use of strcpy could lead to an overflow condition. A char\* type is being copied to a fixed length destination buffer. This could lead to a buffer overflow.

8. Code Snippet – mysqlshow.c (../mysql/mysql-5.6.24/client/mysqlshow.c:710)

```
if (mysql select db(mysql,db))
702
703
          fprintf(stderr, "%s: Cannot connect to db: %s: %s\n", my progname, db,
704
                  mysql error(mysql));
705
          return 1:
706
        1
707
708
        if (opt count)
709
710
          sprintf(query, "select count(*) from "%s", table);
711
          if (mysql query(mysql,query) || !(result=mysql store result(mysql)))
712
713
            fprintf(stderr, "%s: Cannot get record count for db: %s, table: %s: %s\n",
714
                    my progname, db, table, mysql error(mysql));
715
            return 1:
716
717
          row= mysql fetch row(result);
718
          rows= (ulong) strtoull(row[0], (char**) 0, 10);
719
          mysql free result (result);
720
```

**Description:** Insecure use of sprintf. A char\* type is being copied to a fixed length destination buffer. This could lead to a buffer overflow.

9. Code Snippet – conf\_to\_src.c (../mysql/mysql-5.6.24/libmysql/conf to src.c:121

```
116
     void
117
     print arrays for (char *set)
118
119
       FILE *f:
120
121
       sprintf(buf, "%s.conf", set);
122
123
       if ((f = fopen(buf, "r")) == NULL) {
124
         fprintf(stderr, "%s: can't read conf file for charset %s\n", prog, set);
125
         exit(EXIT FAILURE);
126
127
```

**Description:** Insecure use of sprintf. A char\* type is being copied to a fixed length destination buffer. This could lead to a buffer overflow.

 Code Snippet – PosixAsyncFile.ccp (../mysql/mysql-5.6.24/storage/ndb/src/kernel/blocks/ndbfs/PosixAsyncFile.cpp:784)

```
V01d
      PosixAsyncFile::rmrfReg(Request *request, const char * src, bool removePath)
774
775
       if(!request->par.rmrf.directory)
776
777
          // Remove file
778
         if(unlink(src) != 0 && errno != ENOENT)
779
           request->error = errno;
780
          return;
781
782
783
        char path[PATH MAX];
784
        strcpy (path, src);
        strcat (path, "/");
```

**Description:** Unsafe Use of strepy could lead to an overflow condition. A char\* type is being copied to a fixed length destination buffer. This could lead to a buffer overflow.

II. Code Snippet - Win32AsyncFile.ccp (../mysql/mysql-5.6.24/storage/ndb/src/kernel/blocks/ndbfs/Win32AsyncFile.cpp:377)

```
362
363
     Win32AsyncFile::rmrfReq(Request * request, const char * src, bool removePath) {
364
       if (!request->par.rmrf.directory)
365
          // Remove file
366
367
         if (!DeleteFile(src))
368
369
           DWORD dwError = GetLastError();
           if (dwError != ERROR_FILE NOT FOUND)
370
371
             request->error = dwError;
372
373
         return;
374
375
376
       char path[PATH MAX];
       strcpy(path, src);
       strcat(path, "\\*");
378
379
380
       WIN32 FIND DATA ffd;
       HANDLE hFindFile;
381
```

<<< Size of PATH is PATH\_MAX 256

**Description:** Unsafe Use of strepy could lead to an overflow condition. A char\* type is being copied to a fixed length destination buffer. This, could lead to an overflow.

## Acknowledgements

Sincere Thanks to Oracle Corporation for their excellent cooperation.

## References

[1] Security Focus Website (2015). Advanced Information Security Corporation, Security Advisory - MySQL v5.6.24 Buffer Overflows. [Online] Available at: http://www.securityfocus.com/archive/1/536634 [Accessed 7 Oct. 2015].