16.04 upgrade broke mysql-server

Asked 3 years, 10 months ago   Active 16 days ago   Viewed 195k times

My overall upgrade went well, but I am left with a critically lingering issue of mysql-server not being
able to install itself, and nothing I am trying gets it to work.

129

This is the error I see when trying to install/reinstall it:

45

```
Setting up mysql-server-5.7 (5.7.11-0ubuntu6) ...
Job for mysql.service failed because the control process exited with error code.
See "systemctl status mysql.service" and "journalctl -xe" for details.
invoke-rc.d: initscript mysql, action "start" failed.
dpkg: error processing package mysql-server-5.7 (--configure):
 subprocess installed post-installation script returned error exit status 1
dpkg: dependency problems prevent configuration of mysql-server:
 mysql-server depends on mysql-server-5.7; however:
   Package mysql-server-5.7 is not configured yet.

dpkg: error processing package mysql-server (--configure):
 dependency problems - leaving unconfigured
No apport report written because the error message indicates its a followup
error from a previous failure.
                   Processing triggers for systemd (229-4ubuntu4) ...
Processing triggers for ureadahead (0.100.0-19) ...
Errors were encountered while processing:
 mysql-server-5.7
 mysql-server
E: Sub-process /usr/bin/dpkg returned an error code (1)
```

EDIT: Looks like I'm not the only one: https://bugs.launchpad.net/ubuntu/+source/mysql-5.7/+bug/1573279

upgrade mysql 16.04

edited Apr 22 '16 at 18:23   asked Apr 22 '16 at 17:53

TheGremlyn
2,674   3   11   15

I have same problem here. – KernelPanic Apr 23 '16 at 7:46

1   Had to uninstall phpmyadmin, which was causing the dependency problems somehow. After reinstalling, everything is working fine again. – Hinrich Apr 23 '17 at 17:39

1   do-release-upgrade from Ubuntu 14 to 16 performs an unsupported upgrade from MySQL 5.5 to 5.7, so it is expected MySQL is broken after, as reported here: bugs.launchpad.net/ubuntu/+source/ubuntu-release-upgrader/+bug/... – Marco Marsala Oct 24 '17 at 5:24 ✎

1   @MarcoMarsala I don't know if we can say it is *expected* for MySQL to be broken after the Ubuntu 14 to 16 upgrade, though I suppose it depends on your perspective. To the average user, it is certainly not expected to have an LTS upgrade break something like MySQL. It is surprising that was not caught during testing, though. – TheGremlyn Oct 24 '17 at 12:51

apt install phpmyadmin --reinstall fixed the above with no need to touch mysql after... go figure – Andy Apr 9 '19 at 14:44

10 Answers

▲

124

▼

✓

↺

The instructions @andrew-beerman posted are on the right track, though they aren't quite clear to me and seem to recommend more than is necessary. I pieced together the answer from the above and a helpful post in the bug thread.

These are the steps I took to correct this:

1. Back up your `my.cnf file` in `/etc/mysql` and remove or rename it

       sudo mv /etc/mysql/my.cnf /etc/mysql/my.cnf.bak

2. Remove the folder `/etc/mysql/mysql.conf.d/` using

       sudo rm -r /etc/mysql/mysql.conf.d/

3. Verify you don't have a `my.cnf` file stashed somewhere else (I did in my home dir!) or in `/etc/alternatives/my.cnf` use

       sudo find / -name my.cnf

4. Backup and remove `/etc/mysql/debian.cnf` files (not sure if needed, but just in case)

✕

5. In case your syslog shows an error like "mysqld: Can't read dir of '/etc/mysql/conf.d/'" create a symbolic link:

```
sudo ln -s /etc/mysql/mysql.conf.d /etc/mysql/conf.d
```

Then the service should be able to start with `sudo service mysql start` .

That got it working!

edited Jul 6 '17 at 6:11	answered Apr 25 '16 at 13:38
muru **1**	TheGremlyn **2,674** 3 11 15

19	after all these operations (I did it before) I still have same error issues... — Serge P Apr 29 '16 at 14:05
5	You have to remove every my.cnf.* from the /etc/mysql directory. Look for my.cnf.backup, my.cnf.fallback and my.cnf.migrated - all of these files have to be removed too (backup first!) — hitzi May 3 '16 at 8:51
8	@SergiiP `sudo find / -name "my.cnf"` might come in handy. — starbeamrainbowlabs May 14 '16 at 11:31
1	mysql-server-core- should be mysql-server-core-5.7. Otherwise worked like a charm. Thank you! — David Tay May 20 '16 at 12:56
1	+1. You've made my day! Also morning and evening! I was stuck. — Max Yudin Jun 7 '16 at 17:44

▲

58

▼

↺

Today I got the same problem, after trying many solutions i found that the problem was the command `sudo systemctl disable mysql.service` that i used to disable MySQL auto starting, so to get it working i re-enabled again MySQL server using the command `sudo systemctl enable mysql.service` and run again the upgrade process and it terminated perfectly.

answered Apr 25 '16 at 22:37
Naruto Biju Mode **681** 4 2

5	I ran in to this same problem and the same fix worked for me. — Allen Apr 26 '16 at 15:13
5	Worked for me. Did every solution here, mysql worked after this one. — Błażej Michalik Apr 29 '16 at 14:51
6	Same here - this should be top post. — a1phanumeric Sep 15 '16 at 14:32
4	Worked for me too! Thank you so much. Just to be explicit, run: `sudo systemctl enable mysql.service` and after that run `sudo apt install -f` . — Fernando Paladini Jan 25 '17 at 0:03 ✎
	This is all I had to do. I had disabled autostart like described by @naruto. I enabled autostart and run "sudo apt upgrade". Now fixed. Should be top answer. This is a much simpler, more standard and less trouble prone solution. — Jack Holt Aug 7 '17 at 16:00 ✎

20

```
subprocess installed post-installation script returned error exit status 1
```

▼

🕐

However, this `installed post-installation script` is not mentioned by name. After much tinkering, I found out that its name is (in my case) `/var/lib/dpkg/info/mysql-server-5.7.postinst` .

Open this file with `sudo vi /var/lib/dpkg/info/mysql-server-5.7.postinst` , or your preferred editor.

At the top, change line 3 (or so): `set -e` to `set -x` , save the file. (option `-e` is "exit on errors", `-x` means "explicitly show command executed", presumably)

Run `sudo dpkg --configure -a --log /tmp/dpkg.log` (the --log option is optional). You can also simply run `apt upgrade` if you know it'll be the only package that will be upgraded.

Now you get verbose output of the `mysql-server-5.7.postinst` bash script, and you can figure out what's wrong.

In my case it unsuccessfully tried to (re-)run `mysql_upgrade` , but that was not needed for my customized mysql installation. I was *sure* I've run it manually before, successfully, and all was well.

So I commmented out line 321 (for older mysqld releases try line 281),

```
#mysql_upgrade --defaults-file=/etc/mysql/debian.cnf || result=$?
```

and the command that has failed before, `sudo apt upgrade` (run it again), finished successfully, and dpkg removed the error status for this package.

Now you can set back the `set -x` to `set -e` (mentioned above). And optionally uncomment the mysql-upgrade line.

Extra work might be required if you have moved your mysql data partition to a nonstandard location. I moved mine from `/var/lib/mysql/data` to a different drive via symlink. Then you might have to remove the symbolic link temporarily, before the `postinst` script manipulation. Then re-create it after running the package upgrade.

After the next minor version upgrade of the mysqld debian package, this problem with the `/var/lib/dpkg/info/mysql-server-5.7.postinst` script can show up again.

edited Jan 23 '18 at 8:10   answered Jul 1 '16 at 12:03

                                                        knb
                                                        3,029   2   24   33

I ended up purging everything and then downloading the community mysql-server version and manually installing from here: dev.mysql.com/downloads/mysql – RyanNerd Sep 1 '16 at 19:34

Exactly the same thing happened to me, and your steps solved it to. Bur for me my mysql_upgrade call was on line 320. Can you explain why it return a non-zero value when called from postinst script? – emiliopedrollo Sep 24 '16 at 18:28 ✏

@emiliopedrollo No I can't explain here. But I think the line number is now 320 because recently the package maintainers have augmented the postinstall script, I have observed the same thing the other day, during the last run of the software updater (which included a new mysql deb package). – knb Sep 24 '16 at 19:25 ✏

ran `dpkg` again and it worked! – Allen Hamilton Apr 24 '17 at 5:17 ✏️

The instructions here fixed it on my server: https://bugs.mysql.com/bug.php?id=72722

17

I can understand the pain of having your system in inconsistent state but lets not worry about the whole situation and take it step by step to get the system clean.

First lets see the current state of all the mysql packages on machine using: dpkg -l | grep mysql (Please paste the output excluding last column)

The first column denotes the current status of the package. Here are the possible options:

ii) Installed rc) Removed config-files kept (This should be the state of all the packages you have removed with 'apt-get remove' that does not remove config-files under /etc)

For this to work, you will need to run 'apt-get purge <pkg-name>' till you do not see any packages in the above list.

Please remember that some non-mysql-server packages like python-mysql.connector and python-mysqldb, if installed, need not be removed as they do not have any affect on this situation but if removed might cause trouble to applications using them.

We will definitely try to re-visit our docs to see how can we safeguard users from getting into this trouble. Thanks for sharing your feedback in detail with us.

edited Jul 27 '19 at 14:54   answered Apr 22 '16 at 22:05
Toskan   Andrew Beeman
211   4   14   **171**   4

---

2   Thanks for `dpkg -l | grep mysql` . It helped to understand the direction. – Max Yudin Jun 7 '16 at 17:47

command was wrong for purge it must be `sudo apt-get purge <pkg-name>` I fixed it with an edit but: I used unicode characters for the `<` because otherwise it wouldnt display. Be warned copy pasting this command – Toskan Jul 27 '19 at 6:37

In my case, with strace, I saw that /var/run/mysqld/ didn't exist and mysqld can't create the file mysqld.sock.

3

These commands solved my problem:

```
mkdir /var/run/mysqld
chown mysql.mysql /var/run/mysqld
chmod 700 /var/run/mysqld
```

Now:

And mysql works again :)

answered Apr 27 '16 at 8:14

Leonardo Catalinas
31 1

I suggest you add these lines to /usr/share/mysql/mysql-systemd-start starting at line 25, then you won't have to recreate this directory after every reboot (sorry the line returns don't work in this comment): if [! -d /run/msyqld]; then mkdir -p -m0755 /run/mysqld || { echo "Unable to create /run/mysqld"; exit 1; } chown mysql:mysql /run/mysqld || { echo "Unable to chown /run/mysqld"; exit 1; } fi – scoobydoo Jun 30 '16 at 14:44

✎

3

In my case I could solv the problem by adding

```
# Allow log file access
/home/system/var/log/mysql.err rw,
/home/system/var/log/mysql.log rw,
/home/system/var/log/mysql/ r,
/home/system/var/log/mysql/** rw,
```

to `/etc/apparmor.d/local/usr.sbin.mysqld`

For more details take a look at my answer (by ChristophS) at stackoverflow.

edited May 23 '17 at 12:39

Community ♦
1

answered Apr 29 '16 at 12:30

ChristophS
200 2 6

0

I had the same issue. I tried to re-install mysql several times, but had no success.

I figured out that the problem for me was, that another mysql process was already running.

In details:

After I read carefully logs in `/var/log/mysql/error.log` , and found:

[ERROR] Can't start server: Bind on TCP/IP port: Address already in use

[ERROR] Do you already have another mysqld server running on port: 3306 ?

Looks like another application was using already the port.

I checked it using `ps -aux | grep 3306` :

```
$ ps -aux | grep 3306
milkovs+  6802  0.0  0.0  16336  1084 pts/19   S+   21:39   0:00 grep --
color=auto 3306
```

✕

```
error=/var/log/mysql/error.log --pid-file=/var/run/mysqld/mysqld.pid --
socket=/var/run/mysqld/mysqld.sock --port=3306 --log-syslog=1 --log-syslog-
facility=daemon --log-syslog-tag=
```

And I killed the running process `sudo kill -15 14706`

Then I started mysql: `/etc/init.d/mysql start`

Finally mysql works for me! I hope it helps somebody.

answered Aug 20 '16 at 20:02

milkovsky
155   1   10

None of the answers on this page worked for me.

0

I ended up going to the [Oracle downloads page](#), downloading `mysql-apt-config_0.8.8-1_all.deb` , and installing MySQL from Oracle repo:

```
sudo dpkg -i mysql-apt-config_0.8.8-1_all.deb
sudo apt update
sudo apt install mysql-server
```

edited Jul 11 '18 at 18:10   answered Sep 25 '17 at 17:19
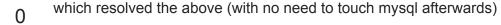
ostrokach
624   5   9

2   `isntall -> install` typo. Stupid SO does not allow me to correct one character. – Csaba Toth Jul 11 '18 at 16:54

Why did you go to the oracle to download mysql – Sinscary Aug 7 '19 at 8:40

@Sinscary This answer is from awhile back, but it was probably because they had a newer version than the system package manager? – ostrokach Aug 7 '19 at 15:42 ✎

I have had the issue on a few servers now The fix was to run apt install phpmyadmin --reinstall

0

which resolved the above (with no need to touch mysql afterwards)

answered Apr 9 '19 at 14:46

Andy
111   2

For WSI Ubuntu 16.04 version I found this helpful article [here](#)

✕

2. Look for this line `mysql_upgrade --defaults-file=/etc/mysql/debian.cnf || result=$?`

3. Add this `service mysql restart` directly above line in number 2.

4. Save and run this `sudo dpkg --configure -a`

it should look like

```
service mysql restart
mysql_upgrade --defaults-file=/etc/mysql/debian.cnf || result=$?
```

Hope this help

answered Feb 7 at 18:23

Fil
123   1   1   7