

# Ubuntu ECC error checking

3 steps are needed for this.

## Additional reading:

[http://buttersideup.com/edacwiki/Main\\_Page](http://buttersideup.com/edacwiki/Main_Page)

<http://www-947.ibm.com/support/entry/portal/docdisplay?lnocid=MIGR-5074651> (interesting warning - blacklist memory checks if you have chipkill ram in classy IBM servers and get spurious EDAC errors)

## Install the right software:

```
oot@waxh0015:~# apt-get install edac-utils
Reading package lists... Done
Building dependency tree
Reading state information... Done
The following extra packages will be installed:
  libedac1 libsysfs2
The following NEW packages will be installed:
  edac-utils libedac1 libsysfs2
0 upgraded, 3 newly installed, 0 to remove and 0 not upgraded.
Need to get 54.9 kB of archives.
After this operation, 348 kB of additional disk space will be used.
Do you want to continue [Y/n]? y
Get:1 http://de.archive.ubuntu.com/ubuntu/ oneiric/main libsysfs2 amd64
2.1.0+repack-1 [23.9 kB]
Get:2 http://de.archive.ubuntu.com/ubuntu/ oneiric/universe libedac1 amd64
0.16-1 [10.8 kB]
Get:3 http://de.archive.ubuntu.com/ubuntu/ oneiric/universe edac-utils
amd64 0.16-1 [20.3 kB]
Fetched 54.9 kB in 0s (220 kB/s)
Selecting previously deselected package libsysfs2.
(Reading database ... 57822 files and directories currently installed.)
Unpacking libsysfs2 (from .../libsysfs2_2.1.0+repack-1_amd64.deb) ...
Selecting previously deselected package libedac1.
Unpacking libedac1 (from .../libedac1_0.16-1_amd64.deb) ...
Selecting previously deselected package edac-utils.
Unpacking edac-utils (from .../edac-utils_0.16-1_amd64.deb) ...
Processing triggers for man-db ...
Processing triggers for ureadahead ...
ureadahead will be reprofiled on next reboot
Setting up libsysfs2 (2.1.0+repack-1) ...
Setting up libedac1 (0.16-1) ...
Setting up edac-utils (0.16-1) ...
* Not enabling Memory Error Detection and Correction since EDAC_DRIVER is
not set [ OK ]
* Loading DIMM labels for Memory Error Detection and Correction: edac
[ OK ]
Processing triggers for libc-bin ...
ldconfig deferred processing now taking place
```

Note the error message above, it's not clear how to set the EDAC\_DRIVER and why the script returns "OK" if it is NOT enabling it's main purpose. Hilarious.  
On the other hand, if we test on we'll see that things might in fact, work.

### Verify the driver is working

```
root@waxh0015:~# edac-ctl --status  
edac-ctl: drivers are loaded.
```

### Check you're getting a EDAC/ECC report

```
root@waxh0015:~# edac-util -v
mc0: 0 Uncorrected Errors with no DIMM info
mc0: 0 Corrected Errors with no DIMM info
mc0: csrow0: 0 Uncorrected Errors
mc0: csrow0: ch0: 0 Corrected Errors
mc0: csrow0: ch1: 0 Corrected Errors
mc0: csrow1: 0 Uncorrected Errors
mc0: csrow1: ch0: 0 Corrected Errors
mc0: csrow1: ch1: 0 Corrected Errors
mc0: csrow2: 0 Uncorrected Errors
mc0: csrow2: ch0: 0 Corrected Errors
mc0: csrow2: ch1: 0 Corrected Errors
mc0: csrow3: 0 Uncorrected Errors
mc0: csrow3: ch0: 0 Corrected Errors
mc0: csrow3: ch1: 0 Corrected Errors
mc1: 0 Uncorrected Errors with no DIMM info
mc1: 0 Corrected Errors with no DIMM info
mc1: csrow0: 0 Uncorrected Errors
mc1: csrow0: ch0: 0 Corrected Errors
mc1: csrow0: ch1: 0 Corrected Errors
mc1: csrow1: 0 Uncorrected Errors
mc1: csrow1: ch0: 0 Corrected Errors
mc1: csrow1: ch1: 0 Corrected Errors
mc1: csrow2: 0 Uncorrected Errors
mc1: csrow2: ch0: 0 Corrected Errors
mc1: csrow2: ch1: 0 Corrected Errors
mc1: csrow3: 0 Uncorrected Errors
mc1: csrow3: ch0: 0 Corrected Errors
mc1: csrow3: ch1: 0 Corrected Errors
mc2: 0 Uncorrected Errors with no DIMM info
mc2: 0 Corrected Errors with no DIMM info
mc2: csrow2: 0 Uncorrected Errors
mc2: csrow2: ch0: 0 Corrected Errors
mc2: csrow2: ch1: 0 Corrected Errors
mc2: csrow3: 0 Uncorrected Errors
mc2: csrow3: ch0: 0 Corrected Errors
mc2: csrow3: ch1: 0 Corrected Errors
mc3: 0 Uncorrected Errors with no DIMM info
mc3: 0 Corrected Errors with no DIMM info
mc3: csrow2: 0 Uncorrected Errors
mc3: csrow2: ch0: 0 Corrected Errors
mc3: csrow2: ch1: 0 Corrected Errors
mc3: csrow3: 0 Uncorrected Errors
mc3: csrow3: ch0: 0 Corrected Errors
mc3: csrow3: ch1: 0 Corrected Errors
```

Nice to know: Linux does not do PCI checksum checking by default. Yet another reason why it's "faster" than real UNIX systems that bother to do all these error checks