# [V8] FAQ: How to run a Data Integrity Check on Linux (CLI) machine?

**Article ID**: 8008
**Reviewed**: 2021-03-03

**Product Version**:
AhsayOBM: 8.1 to 8.x
OS: Linux / FreeBSD

# Description

This article outlines the steps to perform a Data Integrity Check on a Linux / FreeBSD computer with Command Line Interface (CLI) for backup data stored in the backup destination (e.g. local destination, AhsayCBS or other cloud storage).

The functions of the Data Integrity Check (DIC) is to:

1. Identify and remove the files and/or folders in the backup destination(s) which do not appear in the index.
2. Identify and remove the files and/or folders which appear in the index but do not actually exist in the backup destination(s).
3. Identify and remove corrupted files from the backup destination(s) when the Run Cyclic Redundancy Check (CRC) During Data Integrity Check setting is enabled.
4. Identify and remove partially uploaded (orphan) files from the backup destination(s) to free up storage space.
5. Identify and rebuild malfunction file index (option available for version 8.5.0.63 or above).
6. Update the storage statistics for the backup set(s).

Important:
The Data Integrity Check <u>cannot</u> fix or repair files that are already corrupted. It will only identify and remove any corrupted files from the backup destination(s), so these files can be re-uploaded again on the next backup job if they still exist on the backup source. However, any corrupted files removed from the Retention Area will be not be re-uploaded and will no longer be restorable.

Also, the Rebuild index function mentioned in point 5 above is only available for AhsayOBM version 8.5.0.63 or above.

# Steps

Data integrity check can be performed by running the RunDataIntegrityCheck.sh script file located in the installation bin folder.

By default, the Data Integrity Check script will:

- Perform check for all backup sets
- Perform check for all storage destinations

- Perform check with Cyclic Redundancy Check (CRC) disabled

Assumption:
AhsayOBM is installed in the following path /usr/local/obm

1. Login to the Linux machine with root.

2. Edit the the RunDataIntegrityCheck.sh script with a text editor:

```
# cd /usr/local/obm/bin
# vi RunDataIntegrityCheck.sh
```

Modify the settings in the file according to your requirement:

```
####################################### RunDataIntegrityCheck.sh ######################################
# You can use this shell script to run any of your backup sets from the
# command line. Just customize the "User Defined Section" below with your
# values for your backup action.
######################################################################################################

################################### START: User Defined Section ###################################
# ---------------------------------------- SETTING_HOME (Optional) ----------------------------------------
# | Directory to your setting home.
# | Default to ${HOME}/.obm when not set.
# | e.g. SETTING_HOME="${HOME}/.obm"
# ----------------------------------------------------------------------------------------------------
SETTING_HOME=""

# ---------------------------------------------- BACKUP_SET ----------------------------------------------
# | The name or ID of the backup set that you want to run.
# | If backup set name is not in English, please use ID instead.
# | e.g. BACKUP_SET="1119083740107"
# | or BACKUP_SET="FileBackupSet-1"
# | You can use "ALL" to run data integrity check for all backup sets.
# | i.e. BACKUP_SET="ALL"
# |
# | You can leave this parameter blank if you have only 1 backup set.
# ----------------------------------------------------------------------------------------------------
BACKUP_SET="ALL"

# ---------------------------------------------- BACKUP_DEST ----------------------------------------------
# | The name or ID of the backup destination that you want to run.
# | If backup destination name is not in English, please use ID instead.
# | e.g. BACKUP_DEST="1740107119083"
# | or BACKUP_DEST="Destination-1"
# | You can use "ALL" to run data integrity check for all destinations.
# | i.e. BACKUP_DEST="ALL"
# |
# | You can leave this parameter blank if you have only 1 destination.
# | Remark: This option is ignored if BACKUP_SET="ALL"
# ----------------------------------------------------------------------------------------------------
BACKUP_DEST="ALL"

# ---------------------------------------------- CRC_MODE ----------------------------------------------
# | You can run Cyclic Redundancy Check (CRC) during data integrity check
# | Options available: ENABLE-CRC/DISABLE-CRC
# | i.e. CRC_MODE="ENABLE-CRC"
# | or CRC_MODE="DISABLE-CRC"
# ----------------------------------------------------------------------------------------------------
CRC_MODE="ENABLE-CRC"

# ---------------------------------------------- REBUILD_MODE ----------------------------------------------
# | You can run rebuild index
# | Options available: ENABLE-REBUILD/DISABLE-REBUILD
# | i.e. REBUILD_MODE="ENABLE-REBUILD"
# | or REBUILD_MODE="DISABLE-REBUILD"
# ----------------------------------------------------------------------------------------------------
REBUILD_MODE="DISABLE_REBUILD"

################################### END: User Defined Section ###################################
```

Important:
The REBUILD_MODE option is only available for version 8.5.0.63 or above.

3. Start the data integrity check by running the RunDataIntegrityCheck.sh script file.

**Note**:

The following output is an example of a integrity check performed for a file backup set on DropBox storage destination:

```
# sh RunDataIntegrityCheck.sh
-
Using APP_HOME : /usr/local/obm
Using SETTING_HOME :
Using JAVA_HOME : /usr/local/obm/jvm
Using JAVA_EXE : /usr/local/obm/jvm/bin/java
Using JAVA_OPTS : -Xrs -Xms128m -Xmx768m -client -Dsun.nio.PageAlignDirectMemory=true
Using JNI_PATH : -Djava.library.path=.
Using CLASSPATH : .:./cb.jar
-
Running data integrity check for backup set - 'ALL', destination - 'ALL' ...
log4j:WARN No appenders could be found for logger (org.apache.http.impl.conn.PoolingClientConnectionManager).
log4j:WARN Please initialize the log4j system properly.
log4j:WARN See http://logging.apache.org/log4j/1.2/faq.html#noconfig for more info.
[doStart] Start data integrity check on backup set "backupset_name(backupset_id)" all destination, crc enabled
[doDetail] Start processing data integrity check on backup set= "backupset_name" destination= "Dropbox4Business1TB"
[doLogProgress] Browsing "backupset_id/blocks/YYYY-MM-DD-hh-mm-ss"
[doLogProgress] Processing Job "Current", "/bin"
[doLogProgress] Verifying CRC of file "/usr/bin/a2p"
.
.
[doLogProgress] Processing Job "Current", "/usr/sbin"
[doLogProgress] Checking dangling backup file index entries...
[doInfo] Existing statistics of backup set= "backupset_name" destination= "Dropbox4Business1TB": Data area compressed
size: 57.48MB, Data area uncompressed size: 127.8MB, Data area file count: 1497, Retention area compressed size: 0B,
Retention area uncompressed size: 0B, Retention area file count: 0
[doInfo] Recalculated statistics of backup set= "backupset_name" destination= "Dropbox4Business1TB": Data area
compressed size: 57.48MB, Data area uncompressed size: 127.8MB, Data area file count: 1497, Retention area compressed
size: 0B, Retention area uncompressed size: 0B, Retention area file count: 0
[doInfo] The statistics of backup set= "backupset_name" destination= "Dropbox4Business1TB" is correct.
[doInfo] Saving encrypted backup file index to backupset_id/blocks at destination Dropbox4Business1TB...
[doDetail] Data integrity check on backup set= "backupset_name" destination= "Dropbox4Business1TB" is completed
[doEnd][INFO] Finished data integrity check on backup set "backupset_name(backupset_id)" all destination, crc enabled
[doInfo] Completed data integrity check on backup set "backupset_name(backupset_id)" all destination, crc enabled
```

**Note**:

The warning messages displayed can be safely ignored. The data integrity check is not affected by the warning.

4. Integrity check logs can be found in the following path /root/.obm/system/IntegrityCheck:

```
# cd /root/.obm/system/IntegrityCheck
# ls -la
total 36
drwxr-xr-x. 2 root root 4096 Jun 17 15:12 .
drwxr-xr-x. 5 root root 59 Jun 17 14:29 ..
-rw-r--r--. 1 root root 1600 Jun 17 14:29 2016-06-17-14-29-02.log
-rw-r--r--. 1 root root 271 Jun 17 14:56 2016-06-17-14-56-14.log
-rw-r--r--. 1 root root 1600 Jun 17 15:10 2016-06-17-15-10-26.log
```

# Keywords

CRC, Linux, Unix, data integrity check, CLI