RHEL: Multipathing basics

Article Number: 128 | Rating: 5/5 from 1 votes | Last Updated: Sat, Jun 2, 2018 8:48 AM

RHEL: Multipathing basics

```
# Tested on RHEL 5 & 6

# DM-Multipath is a feature of Red Hat from RHEL 5 on and can be used
to provide:
#
# Redundancy: DM-Multipath can provide failover in an active/passive
configuration. In an
#   active/passive configuration, only half the paths are used at any
time for I/O. If any
#   element of an I/O path (the cable, switch, or controller) fails,
DM-Multipath switches
#   to an alternate path.
#   In an active/active configuration all the paths are used in a
round-robin fashion.
#
# Improved Performance: DM-Multipath can be configured in
active/active mode, where I/O is
#   spread over the paths in a round-robin fashion. In some
configurations, DM-Multipath
#   can detect loading on the I/O paths and dynamically re-balance
the load.

# By default, DM-Multipath includes support for the most common
storage arrays that support
# multipathing. The supported devices can be found in the
multipath.conf.defaults file. If
# your storage array supports DM-Multipath and is not configured by
default in this file,
# you may need to add it to the config file.
```

```
# DM-Multipathing components
# ------------------------------------------------------------------------
-----------------------

# - dm-multipath kernel module: Reroutes I/O and supports failover
for paths and path
#   groups.
# - multipathd daemon: Monitors paths; as paths fail and come back,
it may initiate path
#   group switches. Provides for interactive changes to multipath
devices. This must be
#   restarted for any changes to the /etc/multipath.conf file.
# - multipath command: Lists and configures multipath devices.
Normally started up with
#   /etc/rc.sysinit, it can also be started up by a udev program
whenever a block device
#   is added or it can be run by the initramfs file system.
# - kpartx command: Creates device mapper devices for the partitions
on a device It is
#   necessary to use this command for DOS-based partitions with DM-
MP. The 'kpartx' is
#   provided in its own package, but the device-mapper-multipath
package depends on it.


# DM-Multipathing config files
# ------------------------------------------------------------------------
-----------------------

# - /etc/multipath.conf: Main configuration file.
# -
/usr/share/doc/device-mapper-multipath-X.X.X/multipath.conf.defaults:
Lists support
#   storage arrays, if your array is not listed it still may be
possible to configure it
#   in the multipath.conf file .
# - /var/lib/multipath/bindings: This file is automatically
```

```
maintained by the multipath
#   program. It relates user-friendly names and device WWIDs.

# Each multipath device has a World Wide Identifier (WWID), which is
guaranteed to be
# unique and unchanging. By default the name of multipath device is
set to its WWID but
# there is an option in /etc/multipath.conf, "user_friendly_names"
which sets the alias
# to a node-unique name of the form of mpathX:

   ## Use user friendly names, instead of using WWIDs as names.
   defaults {
      user_friendly_names yes
      bindings_file /etc/multipath_bindings
   }


# DM-Multipathing devices
# ------------------------------------------------------------------
----------------------

# multipath creates three different ways to access the device:
# /dev/mapper/mpathX: These are create early in the boot sequence,
thus these are ideal
#    for logical volumes, boot devices
# /dev/mpathX: Are provided as a convenience so that all multipathed
devices can be seen
#    in one directory. These devices are created by the udev device
manager and may not
#   be available during startup.
# /dev/dm-X: These are for external use only.

# There is a fourth option that consists of setting an alias in the
/etc/multipath.conf file.


# DM-Multipath setup
```

```
# ------------------------------------------------------------------
----------------------

# - Install the device-mapper-multipath rpm
# - Edit the /etc/multipath.conf configuration file:
#      - comment out the default blacklist or create you own exclude
blacklist
#      - change any of the default (if required)
# - Start the multipath daemons
# - Create the multipath device with the multipath command


# Basic multipath.conf file
# ------------------------------------------------------------------
----------------------

# We can create the initial configuration file by running following
command:

mpathconf --enable

#   - the default section configures the multipath to use friendly
names, there are a
#      number of other options that can be used.
#   - the blacklist section excludes specific disks from being
multipathed, notice the
#      exclusion of all wwid disks
#   - the blacklist exceptions section includes the devices with a
specific wwid to be
#      included
#   - the multipaths section creates aliases that match a specific
disk to a alias using
#      the wwid


   multipath.conf (basic)     defaults {
      user_friendly_names yes
      path_group_policy failover
```

```
   }

   blacklist {
      devnode "^(ram|raw|loop|fd|md|dm-|sr|scd|st)[0-9]*"
      devnode "^(hd|xvd|vd)[a-z]*"
      wwid "*"
   }

   # Make sure our multipath devices are enabled.

   blacklist_exceptions {
      wwid "20017580006c00034"
      wwid "20017580006c00035"
      wwid "20017580006c00036"
      wwid "20017580006c00037"
   }

   multipaths {
     multipath {
         wwid "20017580006c00034"
         alias mpath0
     }
     multipath {
         wwid "20017580006c00035"
         alias mpath1
     }
     multipath {
         wwid "20017580006c00036"
         alias mpath2
     }
     multipath {
         wwid "20017580006c00037"
         alias mpath3
     }
   }


# Once multipath.conf configured, perform following steps to start
```

```
multipathd:

modprobe dm-multipath

service multipathd start

multipath -d
# This will perform a dry to make sure everything is ok. Fix anything
that
# appears as a problem.

multipath -v2
# Commits the configuration

multipath -ll

chkconfig multipathd on
# Make devices to be configured after a reboot



# Now, we should see something similar to the output below, each
device is active and ready.

multipath -ll |grep mpath
   mpath2 (20017580006c00036) dm-7 IBM,2810XIV
   mpath1 (20017580006c00035) dm-6 IBM,2810XIV
   mpath0 (20017580006c00034) dm-5 IBM,2810XIV
   mpath3 (20017580006c00037) dm-8 IBM,2810XIV



# Following example shows connections to a HP EVA

multipath -ll
   mpath2 (360060e800571100000007110000005405) dm-8 HP,OPEN-V
   [size=408G][features=1 queue_if_no_path][hwhandler=0][rw]
   _ round-robin 0 [prio=2][active]
    _ 2:0:1:0 sdc 8:32  [active][ready]
    _ 3:0:2:0 sdn 8:208 [active][ready]
```

```
mpath1 (360060e80057110000000711000000810a) dm-7 HP,OPEN-V
[size=408G][features=1 queue_if_no_path][hwhandler=0][rw]
_ round-robin 0 [prio=2][active]
 _ 2:0:0:1 sdb 8:16  [active][ready]
 _ 3:0:0:1 sdl 8:176 [active][ready]
mpath0 (360060e80057110000000711000002206) dm-6 HP,OPEN-V
[size=408G][features=1 queue_if_no_path][hwhandler=0][rw]
_ round-robin 0 [prio=2][active]
 _ 2:0:0:0 sda 8:0   [active][ready]
 _ 3:0:0:0 sdk 8:160 [active][ready]
mpath9 (360060e80057110000000711000005306) dm-15 HP,OPEN-V
[size=408G][features=1 queue_if_no_path][hwhandler=0][rw]
_ round-robin 0 [prio=2][active]
 _ 2:0:7:0 sdj 8:144 [active][ready]
 _ 3:0:4:0 sdp 8:240 [active][ready]
mpath8 (360060e80057110000000711000008305) dm-14 HP,OPEN-V
[size=408G][features=1 queue_if_no_path][hwhandler=0][rw]
_ round-robin 0 [prio=2][active]
 _ 2:0:6:1 sdi 8:128 [active][ready]
 _ 3:0:5:1 sdr 65:16 [active][ready]
mpath7 (360060e80057110000000711000002506) dm-13 HP,OPEN-V
[size=408G][features=1 queue_if_no_path][hwhandler=0][rw]
_ round-robin 0 [prio=2][active]
 _ 2:0:6:0 sdh 8:112 [active][ready]
 _ 3:0:5:0 sdq 65:0  [active][ready]
mpath6 (360060e80057110000000711000007408) dm-12 HP,OPEN-V
[size=408G][features=1 queue_if_no_path][hwhandler=0][rw]
_ round-robin 0 [prio=2][active]
 _ 2:0:5:0 sdg 8:96  [active][ready]
 _ 3:0:6:0 sds 65:32 [active][ready]
mpath5 (360060e80057110000000711000002305) dm-11 HP,OPEN-V
[size=408G][features=1 queue_if_no_path][hwhandler=0][rw]
_ round-robin 0 [prio=2][active]
 _ 2:0:4:0 sdf 8:80  [active][ready]
 _ 3:0:7:0 sdt 65:48 [active][ready]
mpath4 (360060e80057110000000711000006207) dm-10 HP,OPEN-V
[size=408G][features=1 queue_if_no_path][hwhandler=0][rw]
_ round-robin 0 [prio=2][active]
```

```
    _ 2:0:3:0 sde 8:64  [active][ready]
    _ 3:0:3:0 sdo 8:224 [active][ready]
   mpath3 (360060e800571100000007110000000409) dm-9 HP,OPEN-V
   [size=408G][features=1 queue_if_no_path][hwhandler=0][rw]
   _ round-robin 0 [prio=2][active]
    _ 2:0:2:0 sdd 8:48  [active][ready]
    _ 3:0:1:0 sdm 8:192 [active][ready]


# If you have made a mistake in the multipath.conf file use following
steps to correct it:
vi /etc/multipath.conf
service multipathd reload
multipath -F
multipath -d
multipath -v2

# It may be that the array we have is not in the
multipath.conf.defaults file. We can add a
# device section (check manufacture's documentation). Below is an
example of a HP OPEN-V
# series array.

    device {
        vendor "HP"
        product "OPEN-.*"
        getuid_callout "/sbin/scsi_id -g -u -s /block/%n"
        hardware_handler "0"
        path_selector "round-robin 0"
        path_grouping_policy multibus
        failback immediate
        rr_weight uniform
        no_path_retry 12
        rr_min_io 1000
        path_checker tur
    }
```

```
# Advanced multipath.conf file
# ------------------------------------------------------------------
----------------------

# The configuration file is divided into the following sections:

#   - defaults: general setup parameters
#   - blacklist: lists specific devices to exclude from multipathing
#   - blacklist exceptions: lists devices that would otherwise be
excluded
#   - multipaths: settings for the characteristics of individual
multipath devices
#   - devices: settings for non-default storage arrays

# We can blacklist any device but we need to tell multipath what to
exclude. Some examples:

# wwid
# --------------------------------

# Specific wwid
   blacklist {
      wwid "20017580006c00034"
   }

# All wwid
   blacklist {
      wwid "*"
   }

# device name
# --------------------------------

# All sd devices From "a" to "z"
   blacklist {
      devnode "^sd[a-z]"
   }
```

```
# A more advanced example
   blacklist {
      devnode "^(ram|raw|loop|fd|md|dm-|sr|scd|st)[0-9]*"
      devnode "^(hd|xvd|vd)[a-z]*"
   }

# device type
# ---------------------------------

# Blacklist HP devices
   blacklist {
      device {
         vendor "HP"
         product "*"
      }
   }


# To exclude from the blacklist we create an exception list

# wwid
# ---------------------------------

# Exclude a specific wwid
   blacklist_exceptions {
      wwid "20017580006c00034"
   }

# Exclude all wwid
   blacklist_exceptions {
      wwid "*"
   }

# device name
# ---------------------------------

# All sd devices x through z
   blacklist_exceptions {
```

```
      devnode "^sd[x-z]"
   }


# device type
# ---------------------------------

# Exclude HP devices
   blacklist_exceptions {
      device {
         vendor "HP"
         product "*"
      }
   }


# The default section has a number of parameters which can be changed

# Parameter                  Default Value              Description
# ----------                 --------------
-------------------------------------------------
#
# udev_dir                   /udev                      Specifies the
directory where udev device nodes are created.
#
# verbosity                  2                          (RHEL 5.3 and
later) Specifies the verbosity level
#                                                       of the
command. It can be overridden by the -v command line option.
#
# polling_interval           5                          Specifies the
interval between two path checks in seconds.
#
# selector                   round-robin 0              Specifies the
default algorithm to use in determining what
#                                                       path to use
for the next I/O
#                                                       operation.
```

```
#
# path_grouping_policy       failover                    Specifies the
default path grouping policy to apply to
#                                                      unspecified
multipaths.
#
#                                                      Possible
values include:
#                                                        failover =
1 path per priority group
#                                                        multibus =
all valid paths in 1 priority group
#
group_by_serial = 1 priority group per detected serial number
#
group_by_prio = 1 priority group per path priority value
#
group_by_node_name = 1 priority group per target node name
#
# getuid_callout           /sbin/scsi_id -g -u -s          Specifies the
default program and arguments to call out to obtain a
#                                                      unique path
identifier. An absolute path is required.
#
# prio_callout                                          Specifies the
the default program and arguments to call out to
#                                                      obtain a path
weight. Weights are summed for each path group to
#                                                      determine the
next path group to use in case of failue. "none" is a
#                                                      valid value.
#
# path_checker               readsector0                Specifies the
default method used to determine the state of the
#                                                      paths.
Possible values include: readsector0, rdac, tur, cciss_tur,
#                                                      hp_tur (RHEL
5.5 and later), emc_clariion, hp_sw, and directio.
```

```
#
# features                                                The extra
features of multipath devices. The only existing feature
#                                                       is
queue_if_no_path, which is the same as setting no_path_retry to
#                                                       queue.
#
# rr_min_io                 1000                        Specifies the
number of I/O requests to route to a path before
#                                                       switching to
the next path in the current path group.
#
# max_fds                                               (RHEL 5.2 and
later) Sets the maximum number of open file descriptors
#                                                       for the
multipathd process. In RHEL 5.3, this option allows a value of
#                                                       max, which
sets the number of open file descriptors to the system maximum.
#
# rr_weight                 uniform                     If set to
priorities, then instead of sending rr_min_io requests to a
#                                                       path before
calling selector to choose the next path, the number of
#                                                       requests to
send is determined by rr_min_io times the path's priority,
#                                                       as determined
by the prio_callout program. Currently, there are priority
#                                                       callouts only
for devices that use the group_by_prio path grouping policy,
#                                                       which means
that all the paths in a path group will always have the same
#                                                       priority.
#                                                       If set to
uniform, all path weights are equal.
#
# failback                  manual                      Specifies
path group failback. A value of 0 or immediate specifies that
#                                                       as soon as
```

```
there is a path group with a higher priority than the current
#                                                       path group
the system switches to that path group. A numeric value greater
#                                                       than zero
specifies deferred failback, expressed in seconds. A value of
#                                                       manual
specifies that failback can happen only with operator intervention.
#
# no_path_retry             null                        A numeric
value for this attribute specifies the number of times the
#                                                       system should
attempt to use a failed path before disabling queueing
#                                                       A value of
fail indicates immediate failure, without queuing.
#                                                       A value of
queue indicates that queuing should not stop until the path
#                                                       is fixed.
#
# flush_on_last_del         no                          (RHEL 5.3 and
later) If set to yes, the multipathd daemon will disable
#                                                       queueing when
the last path to a device has been deleted.
#
# queue_without_daemon      yes                         (RHEL 5.3 and
later) If set to no, the multipathd daemon will disable
#                                                       queueing for
all devices when it is shut down.
#
# user_friendly_names       no                          If set to
yes, specifies that the system should using the bindings file
#                                                       to assign a
persistent and unique alias to the multipath, in the form of
#                                                       mpathn. The
default location of the bindings file is /var/lib/multipath/bindings,
#                                                       but this can
be changed with the bindings_file option. If set to no, specifies
#                                                       that the
system should use use the WWID as the alias for the multipath. In
```

```
#                                                       either case,
what is specified here will be overriden by any device-specific
#                                                       aliases you
specify in the multipaths section of the configuration file.
#
# bindings_file   /var/lib/multipath/bindings           (RHEL 5.2 and
later) The location of the bindings file that is used with the
#
user_friend_names option.
#
# mode              The default value is                (RHEL 5.3 and
later) The mode to use for the multipath device nodes, in octal.
#                   determined by the process.
#
# uid               The default value is                (RHEL 5.3 and
later) The user ID to use for the multipath device nodes. You
#                   determined by the process.          must use the
numeric user ID.
#
# gid               The default value is                (RHEL 5.3 and
later) The group ID to use for the multipath device nodes. You
#                   determined by the process.          must use the
numeric group ID.
#
# checker_timeout   The default value is taken from     (RHEL 5.5 and
later) The timeout value to use for path checkers that issue
#                   sys/block/sdx/device/timeout.       SCSI commands
with an explicit timeout, in seconds.



# The multipaths section parameters are as follows
#
# Parameter                 Description
# ----------                ---------------------------------------
#
# wwid                      Specifies the WWID of the multipath
device to which the multipath attributes apply.
```

```
#
# alias                     Specifies the symbolic name for the
multipath device to which the multipath attributes apply.
#
# path_group_policy
# prio_callout
# path_selector
# failback
# rr_weight                 The same as the defaults table
# no_path_retry
# flush_on_last_del
# rr_min_io
# mode
# uid_gid



# The devices section parameters are as follows
#
# Parameter                 Description
# ----------                ---------------------------------------
# vendor                    Specifies the vendor name of the storage
device to which the device attributes apply, for example
#                           COMPAQ.
#
# product                   Specifies the product name of the
storage device to which the device attributes apply, for example
#                           HSV110 (C)COMPAQ.
#
# path_checker              Specifies the default method used to
determine the state of the paths. Possible values include
#                           readsector0, rdac, tur, cciss_tur,
hp_tur, emc_clariion, hp_sw, and directio.
#
# features                  The extra features of multipath devices.
The only existing feature is queue_if_no_path, which is
#                           the same as setting no_path_retry to
queue.
```

```
#
# hardware_handler            Specifies a module that will be used to
perform hardware specific actions when switching path groups
#                             or handling I/O errors. Possible values
include 0, 1 emc, and 1 rdac. The default value is 0.
#
# product_blacklist           Specifies a regular expression used to
blacklist devices by product.
#
# path_group_policy
# getuid_callout
# prio_callout
# path_selector
# failback                    The same as defaults table
# rr_weight
# no_path_retry
# flush_on_last_del
# rr_min_io
```