Red Hat Storage 3
Technical Notes

Detailed notes on the changes implemented in Red Hat Storage 3

Anjana Suparna Sriram Shalaka Harne Pavithra Srinivasan
Bhavana Mohan

Red Hat Storage 3 Technical Notes

Detailed notes on the changes implemented in Red Hat Storage 3

Anjana Suparna Sriram
Red Hat Engineering Content Services
asriram@redhat.com

Shalaka Harne
Red Hat Engineering Content Services
sharne@redhat.com

Pavithra Srinivasan
Red Hat Engineering Content Services
psriniva@redhat.com

Bhavana Mohan
Red Hat Engineering Content Services
bmohanra@redhat.com

Legal Notice

Abstract

The Red Hat Storage 3.0 Technical Notes list and documents the changes made to the Red Hat Storage 3.0.

Table of Contents

Chapter 1. RHBA-2015:0038

The bugs contained in this chapter are addressed by advisory RHBA-2015:0038. Further information about this advisory is available at https://rhn.redhat.com/errata/RHBA-2015-0038.html.

build

BZ#1164721

The gstatus utility is added to the Red Hat Storage Server to provide an easy-to-use, high-level view of the health of a trusted storage pool with a single command. It gathers status/health information of the Red Hat Storage nodes, volumes, and bricks by executing the GlusterFS commands.

gluster-afr

BZ#1122492

Previously, executing **`gluster volume heal`** ***`volname`*** **`info`** command repeatedly caused excessive logging of split-brain messages and resulted in a large log file. With this fix, these split-brain messages in log-file are suppressed.

BZ#1055489

Previously, executing **`volume heal info`** command flooded glfsheal log file with entrylk failure messages. With this fix, the log levels of these log messages are lowered to appropriate levels.

BZ#1114999

Previously, executing **`gluster volume heal`** ***`vol-name`*** **`info`** command when user serviceable snapshot was enabled caused the command to fail with *Volume vol-name is not of type replicate* message. With this fix, executing the command lists the files that need healing.

BZ#969355

Previously, when a brick was replaced and the data was yet to be synchronized, all the operations on the brick, which was just replaced would fail and the failures were logged even when the files/directories did not exist. With this fix, messages are not logged when the files do not exist.

BZ#1054712

Previously, executing **`gluster volume heal`** ***`VOLNAME`*** **`info`** command used to print random characters for some files when stale entries were present in indices/xattrop folder. With this fix, no junk characters are printed.

gluster-dht

BZ#1154836

Previously, the rebalance operation failed to migrate files if the volume had both quota and **`features.quota-deem-statfs`** option enabled. This was due to an incorrect free space calculation. With this fix, the free space calculation issue is resolved and the rebalance operation successfully migrates the files.

BZ#1142087

Previously, a warning message asking the user to restore data from the removed bricks was displayed even when the **remove-brick** command was executed with the **force** option. With this fix, this warning message is no longer displayed.

BZ#1122886

Previously, if a mkdir sees EEXIST [as a result of lookup and mkdir race] on a non-hashed subvolume, it reports I/O error to the application. With this fix, if the mkdir is successful on the hashed subvolume, then no error is propagated to the client.

BZ#1140517

Previously, executing the rebalance status command displayed incorrect values for the number of skipped and failed file migrations. With this fix, the command displays the correct values for the number of skipped and failed file migrations.

gluster-nfs

BZ#1102647

Previously, even though **nfs.rpc-auth-reject** option was reset, hosts/addresses which were rejected before, were still unable to access the volume over NFS. With this fix, the issue is resolved and hosts/addresses that were rejected are now allowed to access the volume over NFS.

BZ#1147460

Previously, as a consequence of using ACLs over NFS, the memory leaked and caused the NFS-server process to be terminated by the Linux kernel OOM-killer. With this fix, the issue is resolved.

BZ#1118359

Support for mounting a subdirectory over UDP is added. Users can now mount a subdirectory of a volume over NFS with the MOUNT protocol over UDP.

BZ#1142283

Previously, the help text of **nfs.mount-rmtab** displayed incorrect filename for the rmtab cache. With this fix, the correct the filename of the rmtab cache is displayed in the help text.

BZ#991300

Previously, Gluster-NFS did not resolve symbolic links into directory handle and mount failed. With this fix, if a symbolic link is consistent throughout the volume, then the subdirectory mounts for symbolic link works.

BZ#1101438

Previously, when **root-squash** was enabled or even when no permissions were given to a file, NFS threw permission errors. With this fix, these permission errors are not displayed.

gluster-quota

BZ#1146830

Previously, enabling Quota on Red Hat Storage 3.0 did not create pgfid extended attributes on existing data. The pgfid extended attributes are used to construct the ancestry path (from the file to the volume root) for nameless lookups on files. As NFS relies heavily on nameless lookups, quota enforcement through NFS would be inconsistent if quota were to be enabled on a volume with existing data. With this fix, the pgfid xattrs in the lookup on the existing data are healed.

gluster-smb

BZ#1127658

Previously, when the gluster volume was accessed through libgfapi, xattrs were being set on parent of the brick directories. This led to add-brick failures if new bricks were to be under the same parent directory. With this fix, xattrs are not set on the parent directory. However, existing xattrs on parent directory would remain and users must manually remove it if any add-brick failures are encountered.

BZ#1175088

Previously, creating a new file over the SMB protocol, took a long time if the parent directory had many files in it. This was due to a bug in an optimization made to help Samba to ignore case comparison of requested file name to every entry in the directory. With this fix, the time taken to create a new file over the SMB protocol takes lesser time than before, even if the parent directory had many files in it.

BZ#1107606

Previously, setting either the **`user.cifs`** or **`user.smb`** option to disable did not stop the sharing of SMB shares when the SMB share is already available. With this fix, setting either **`user.cifs`** or **`user.smb`** to disable ensures that the SMB share is immediately stopped.

gluster-snapshot

BZ#1157334

Active snapshots consume similar resources as a regular volume. Therefore, to reduce the resource consumption, newly created snapshots will be in deactivated state, by default. New snapshot configuration option **`activate-on-create`** has been added to configure the default option. You must explicitly activate new snapshots manually for accessing that snapshot.

gluster-swift

BZ#1180463

The Object Expiration feature is now supported in Object Storage. This feature allows you to schedule deletion of objects that are stored in the Red Hat Storage volume. You can use the Object expiration feature to specify a lifetime for objects in the volume. When the lifetime of an object expires, it automatically stops serving that object and shortly thereafter removes the object from the Red Hat Storage volume.

glusterfs

BZ#1111566

Previously, executing rebalance status command displayed *Another transaction is in progress* message after rebalance process is started which indicates that the cluster wide lock is not released for certain reasons and further CLI commands were not allowed. With this fix, all possible error cases in the glusterd op state machine are handled and the cluster wide lock is released.

BZ#1138547

Previously, peer probe failed during rebalance as the global peerinfo structure was modified while a transaction was in progress. The peer was rejected and could not be added into the trusted cluster. With this fix, local peer list is maintained in gluster op state machine on a per transaction basis such that peer probe and rebalance can go on independently. Now, probing a peer during rebalance operation will be successful.

BZ#1061585

Previously, if the setuid bit of a file was set and if the file was migrated after a *remove-brick* operation, after the file migration, the setuid bit did not exist. With this fix, changes are made to ensure that the file permissions retain the setuid bit even after file migration.

BZ#1162468

Previously, no error message was displayed if a CLI command was timed out. With this fix, code is added to display error message if the CLI command is timed out.

glusterfs-geo-replication

BZ#1146369

Previously, in geo-replication, RENAME was processed as UNLINK in slave if renamed file is deleted in Master. Due to this, rename does not succeeded in Slave and if a file created with the same name in Master will not be propogated to Slave. Hence, Slave will have file with old GFID. With this fix, Slave will not have file with corrupt GFID as RENAME is handled as RENAME instead of delete in slave.

BZ#1152990

Previously, the list of slave hosts were fetched only one time during geo-replication start and geo-replication workers used that list to connect to slave nodes. Due to this, when a slave node goes down, geo-replication worker always tries to connect to same node instead of switching to other slave node and geo-replication worker goes to faulty state. Hence, the data synchronizing to slave was delayed. With this fix, on a slave node failure, the list of slave nodes are fetched again and chooses different node to connect.

BZ#1152992

Previously, when glusterd process was stopped, the other processes like glusterfsd, gsyncd were not stopped. With this fix, a new script is provided to stop all gluster processes.

BZ#1144428

Previously, while geo-replication synchronizes directory renames, File's blob was sent for directory entry creation to gfid-access translator resulting *Invalid blob length* marked as ENOMEM and geo-replication went faulty with *Cannot allocate memory* backtrace. With this fix, during renames, if source is not present on slaves, direct entry creation on slave is done only for files and not for directories and geo-replication can successfully synchronizes rename of directories to slave without ENOMEM backtrace.

BZ#1104061

Previously, geo-replication failed to synchronize ownership of empty files or files copied from other location. Hence, files in slave had different ownership and permissions. This was due to GID not being propogated to slave and changelog being missed recording SETATTR in master due to issue in changelog slicing. With this fix, files in both master and slave will have the same ownership and permission.

BZ#1139156

Previously, Geo-replication missed synchronizing a few files to slave when I/O happened during geo-replication start. With this fix, slave does not miss any files if I/O happens during geo-replication start.

BZ#1142960

Previously, when geo-replication was paused and the node was rebooted, geo-replication status remained at *Stable(paused)* state even after session was resumed. The further geo-replication pause displayed *Geo-rep already paused* message. With this fix, there is no mismatch between status file and actual status of geo-replication processes and the geo-replication status in rebooted node remains intact after session is resumed.

BZ#1102594

Previously, Geo-replication was not logging the list of files which failed to synchronize to slave. With this fix, geo-replication logs the gfids of skipped files when files fail to synchronize after maximum number of retries of changelog.

glusterfs-rdma

BZ#1169764

Previously, for socket writev, all the buffers are aggregated and received at the remote end as one payload. So there is only one buffer needed to hold the data. But for RDMA, the remote endpoint will read the data from client buffer as one by one. So there was no place for holding the data starting from second buffer.

glusterfs-server

BZ#1113965

Previously, if AFR self-heal involves healing of renamed directories, the gfid handle of the renamed directories was removed from the sink brick. In a distributed replicate volume, performing *readdir* of the directories resulted in duplicate listing for . and .. entries and for files having dht *link.to* attribute because of this issue. With this fix, the gfid-handle of the renamed directory is not removed.

BZ#1152900

Previously, there was 100% CPU utilization and continuous memory allocation which made the glusterFS process unusable and caused a very high load on the Red Hat Storage Server and possibly rendering it unresponsive to other requests. This was due to the parsing of a Remote Procedure Call (RPC) packet containing a continuation RPC-record, causing an infinite loop in the receiving glusterFS process. With this fix, such RPC-records are handled appropriately and do not lead to service disruptions.

BZ#1130158

Previously, executing rebalance status command displayed *Another transaction is in progress* message after rebalance process is started which indicates that the cluster-wide lock is not released. Hence, further CLI commands were not allowed. With this fix, all error cases in the glusterd op state machine are handled properly, cluster wide lock is released, and further CLI commands are allowed.

BZ#1123732

Previously, the rebalance state of a volume was not being saved on peers where rebalance was not started, that is, peers which do not contain bricks belonging to the volume. Hence, if glusterd processes were restarted on these peers, running a volume status command lead to the occurrence of error logs in the glusterd log files. With this fix, these error logs no longer appear in glusterd logs.

BZ#1109742

Previously, when a glusterd process with operating version lower than that of the trusted storage pool connected to the cluster, it brought down the operating version of the trusted storage pool. This happens even if the peer was not part of the storage pool. With this fix, the operating version of the trusted storage pool will not be lowered.

Chapter 2. RHBA-2015:0039

The bugs contained in this chapter are addressed by advisory RHBA-2015:0039. Further information about this advisory is available at https://rhn.redhat.com/errata/RHBA-2015-0039.html

gluster-nagios-addons

BZ#1136205

Previously, the Nagios plug-in sent the volume status request to the Red Hat Storage node without converting the Nagios host name to the respective IP Address. When the **glusterd** service was stopped on one of the nodes in a Red Hat Storage Trusted Storage Pool, the volume status displayed a warning and the status information was empty. With this fix, the error scenarios are handled properly and the system ensures that the **glusterd** service starts before it sends such a request to a Red Hat Storage node.

BZ#1109727

Previously, when one of the bricks in a replica pair was down in a replicate volume type, the status of the Geo-replication session was set to **FAULTY**. This resulted in the status of the Nagios plug-in to be set to **CRITICAL**. With this fix, changes are made to ensure that if only one of bricks in a replica pair is down, the status of the Geo-replication session is set to **PARTIAL FAULTY** as the Geo-replication session is active on another Red Hat Storage node, in such a scenario.

BZ#1109752

Previously, the Geo-replication status plug-in displayed a **Warning** state when the Red Hat Storage volume was locked due to another volume operation. With this fix, when a volume is locked, the command is executed again after a wait time. If the error message persists, the status plug-in displays the state as **unknown**.

BZ#1141171

Previously, the status of the quorum service displayed an incorrect status. With this fix, a buffering issue is fixed and the quorum service displays the appropriate status.

BZ#1143995

Previously, when a brick was created from a thin-provisioned volume, the brick utilization would not display the actual brick utilization of the thin pool. With this fix, bricks with thin-logical volume display both the thin-logical volume utilization and the actual thin pool utilization.

BZ#1109702

Previously, even after a volume was deleted, the volume information continued to appear in the output of the **Cluster-quorum** service plug-in. The plug-in retains the information of the volume which lost the quorum and updates it only when the quorum is either lost or regained. With this fix, the stale information in the output is removed and the plug-in output is displayed appropriately. As a result, the information about deleted volumes is not present in plug-in output.

BZ#1120832

Previously, when the value for the **hostname_in_nagios** parameter was not configured in the **/etc/nagios/nagios_server.conf** file, the corresponding log message that was recorded, was unclear. With this fix, a clear message is displayed.

BZ#1105568

Previously, the status message for CTDB, NFS, Quota, SMB, and Self Heal services were not clearly defined in the Nagios Remote Plug-in Executor. With this fix, the plug-in for these services return the correct error message and when the **`glusterd`** service is offline, clear values are displayed for **`Status`** and **`Status Information`** fields.

BZ#1109723

Previously, the **`Auto-config`** service would not work if the **`glusterd`** service was offline in any of the nodes in the Red Hat Storage trusted storage pool. With this fix, the Auto-config service works even if the **`glusterd`** service is down in some of the nodes in the trusted storage pool provided that the **`glusterd`** service is running in the node which is used as sync host in the auto-config service.

nagios-server-addons

BZ#1128007

Previously, when all the nodes in a Red Hat Storage trusted storage pool were offline, all the volumes were moved to an **`UNKNOWN`** state and the cluster status was displayed as UP with message **`OK:None of the volumes are in critical state`**. With this fix, changes are made to consider all the status of volumes while computing the status of the Red Hat Storage trusted storage pool.

BZ#1109843

Previously, if the host that is used for discovery was detached from the Red Hat Storage trusted storage pool, then all the hosts would get removed from the Nagios configuration when an auto-discovery was performed. With this fix, the **`auto-config`** service does not remove any configuration detail if the host used for discovery is detached from the Red Hat Storage trusted storage pool.

BZ#1119233

Previously, the graph for cluster utilization did not display values in percentage on the Y-axis. This happened because the plug-in used the default template where the scale value of the graph was not fixed. With this fix, a specific template is implemented for the Nagios plug-in.

BZ#1139228

Previously, if the host that was used for discovery was detached from the Red Hat Storage Trusted Storage Pool, then all the hosts would get removed from the Nagios configuration when auto-discovery was performed. With this fix, the **`auto config`** service does not remove the configurations and it works as expected.

BZ#1138943

Previously, the **`auto-config`** service tried to restart the Nagios service though there was a configuration error. As a result, auto-config service reported a message: **`restarted nagios successfully`**, though the Nagios service was not running. With this fix, changes are made to check the configuration before restarting Nagios service.

rhsc

BZ#1112183

Previously, users could select a starting date later than end date in the **Trends** tab of the Red Hat Storage Console. With this fix, a validation is performed and an appropriate alert message is displayed.

BZ#1152877

Previously, when a host had multiple network addresses, the system failed to identify the brick correctly from the output of **gluster volume status** command. As a result, the brick status appeared to be offline after a node restart, though the bricks were online. With this fix, changes are made to ensure that the brick statuses are displayed appropriately.

BZ#1138143

Previously, users could view only a few of the utilization graphs in the **Trends** tab of the Red Hat Storage Console. To view service based information, users had to navigate to the Nagios Web UI and there was no such link provided on the Red Hat Storage Console. With this release, a link is added to help the user navigate to the Nagios web UI from the **Trends** tab when monitoring is enabled.

BZ#1138108

Previously, the **glusterpmd** service needed to be manually started in the Red Hat Storage node after adding the node to the Red Hat Storage Console. With this fix, the **glusterpmd** service works as expected. To fix this issue, after updating Red Hat Storage Console and the Red Hat Storage nodes to version 3.0.3, you must **reinstall** the Red Hat Storage nodes that were previously added to the Red Hat Storage Console.

BZ#1111087

Previously, there was no mechanism to enable the monitoring feature after disabling it. With this fix, the user can enable monitoring by executing **rhsc-monitoring enable** command from the command line interface.

BZ#1111079

Previously, the Red Hat Storage Console installed Nagios and enabled monitoring by default. After the installation, if the user disabled the monitoring feature, the Nagios server would not stop running on the Red Hat Storage Console node. With this fix, to disable the monitoring feature, execute the **rhsc-monitoring disable** command on the command line interface. This would stop the Nagios Server and Nagios Service Check Acceptor (NSCA) server.

BZ#1106459

Previously, an error was displayed when moving a Red Hat Storage node from one Red Hat Storage Trusted Storage Pool to another. With this fix, checks that inhibits such movements are removed.

BZ#1057574

Previously, the add host operation using the SSH public key by following the Guide Me link failed. This happened due to an incorrect authentication method being set. With this fix, hosts can be added successfully using the SSH public key.

Chapter 3. RHBA-2014:1820

The bugs contained in this chapter are addressed by advisory RHBA-2014:1820. Further information about this advisory is available at https://rhn.redhat.com/errata/rhs-3.0-errata.html.

BZ#1154752, BZ#1154753, BZ#1154754

Higher versions of 'samba', 'glusterfs', and 'augeas-libs' packages was released in Red Hat Enterprise Linux 6.6. This caused package dependency conflicts with the same packages in Red Hat Storage 3 which is based on Red Hat Enterprise Linux 6.5. Hence, this resulted in update failure for the currently installed Red Hat Storage 3 and layered installation failure of freshly installed Red Hat Storage 3. With this update, these package dependency conflicts are resolved and the layered installation of Red Hat Storage 3 on Red Hat Enterprise Linux 6.6 is successful.

BZ#1159273

With this update, Red Hat Enterprise Linux 6.6 product certificate is provided with the 'redhat-storage-server' package.

Chapter 4. RHBA-2014:1819

The bugs contained in this chapter are addressed by advisory RHBA-2014:1819. Further information about this advisory is available at https://rhn.redhat.com/errata/RHBA-2014-1819.html.

nagios-server-addons

BZ#1154306

Previously, updating the system to Red Hat Enterprise Linux 6.6 failed, as the updated version of **`rrdtool-perl`** was not available. With this fix, the updated packages of **`rrdtool`** and **`rrdtool-perl`** is added to the *Red Hat Storage 3 Nagios Server* channel and the system update to Red Hat Enterprise Linux 6.6 is successful. Now, Red Hat Storage Console 3.0 supports updates on Red Hat Enterprise Linux 6.6.

rhsc-docs

BZ#1157946

Removed the step **`yum install subscription-manager-migration-data`** from the Section 3.7 of the Installation Guide.

BZ#1158337

Updated the Installation Guide that Red Hat Storage Console is now supported with Red Hat Enterprise Linux 6.6 server.

Chapter 5. RHEA-2014:1278

The bugs contained in this chapter are addressed by advisory RHEA-2014:1278. Further information about this advisory is available at https://rhn.redhat.com/errata/RHEA-2014-1278.html.

gluster-afr

BZ#1097581

Previously, data loss was observed when one of the bricks in a replica pair goes offline, and a new file is created in the interim before the other brick is back online. When the first brick is available again before a self heal process happens on that directory of the brick and consequently if the second brick goes offline again and new files are created on the first brick, and it crashes at a certain point leaving the directory in a stale state although it has new data. When both the bricks in the replica pair are back online, the newly created data on the first brick is deleted leading to data loss. With this fix, the data loss is not observed.

BZ#1055707

Previously, **`glusterfs`** stored symlinks to each of the directories present on the bricks in **`brick-directory/.glusterfs`** to access them via glusterfs file ID i.e gfid. Some cases were observed where the symlink went missing for a particular directory and from then on directories were created instead of symlinks for the directories with missing symlinks. With this fix, symlinks is created even in these cases.

BZ#1120245

Previously, the metadata **`self-heal`** did not deallocate the memory it allocates and this led to high memory usage of the **`self-heal`** daemon. With this fix, deallocation of memory works as expected, hence metadata self-heal of numerous files does not lead to high memory usage of the **`self-heal`** daemon.

BZ#986317

An enhancement has been made to the **`gluster volume heal`** ***`volname`*** **`info`** command. With this fix, this command lists only the files or directories that need self-heal.

gluster-dht

BZ#1116137

Previously, even if the inode times (**`mtime`**, **`ctime`** etc) have been reset to past values using the **`setattr`** command, the values were not reflected in the subsequent metadata (stat) information. With this fix, the inode timestamp values are set with the **`force`** option in the inode context with the **`setattr`** command and the inode timestamps are reflected appropriately.

BZ#1090986

Previously, the directory entries were read only from the subvolume which has been up for the longest time. If a newly created directory was not yet created on the longest up subvolume when a snapshot was taken, the restored snapshot mount point did not list the newly created directory. With this fix, the directory entries are filtered from their corresponding hashed subvolumes. Only in case of a hashed subvolume having a NULL value (either due to a layout anomaly or a hashed volume being offline), the entry is filtered from the subvolume that has been up the longest.

BZ#1117283

If a file is not found on its cached subvolume, a lookup operation for the file is sent to all subvolumes. Previously, this operation would identify linkto files as regular files and proceed with file operations on it. With this fix, the linkto file is not identified as a regular file and if it is stale, it will not be linked.

BZ#1125958

Previously, some operations would fail if the directory in which they were performed was missing on some bricks in the volume (this could happen if the directory was created when those bricks were down). If a caller bypasses lookup and calls access due to saved/cached inode information (like the NFS server does) then, dht_access fails the operation if an ENOENT error is returned. With this fix, if the directory is not found in one sub-volume, then the information is fetched from the next sub-volume.

BZ#1121099

Previously, when the cluster topology changed due to add-brick, all subvolumes of DHT did not contain the directories till a rebalance was completed. With this fix, the problem has been resolved in dht_access thereby preventing DHT from misrepresenting a directory as a file in the case presented above.

gluster-nfs

BZ#1018894

In gluster volume set, values for keys **`nfs.rpc-auth-allow`** and **`nfs.rpc-auth-reject`** now support wildcard characters and IPv4 subnetwork pattern using CIDR format. However, wildcard character and subnetwork pattern must not be mixed.

BZ#1098862

Previously, the glusterFS NFS server did not validate the unsupported RPC procedure and segmentation faults. With this fix, the system validates the RPC procedures for glusterFS NFS ACL program as a result, a system crash is averted.

BZ#1116992

Previously, mounting a volume over NFS (TCP) with **`MOUNT`** over UDP failed due to a strict verification of memory allocations. Enabling the **`nfs.mount-udp`** did not support NFS Server mount exports over UDP (**`MOUNT`** protocol only, NFS will always use TCP). As a result, when the users tried to use the MOUNT service over UDP, connections timed out and the mount operation failed. With this release, the **`MOUNT`** service works over UDP as expected and supports mounting of complete volumes. However, it does not support sub-directory exports (for example, **`server:/volume/subdir`**).

gluster-quota

BZ#1092429

Previously, the **`quoatd`** process started blocking the epoll thread when glusterd was started. This led to glusterd being deadlocked during startup. As a result, the daemon processes could not start correctly. As a result, two instances of the daemon processes were observed. With this fix, **`Quoatd`** is started separately leaving the epoll thread free to serve other requests. All the daemon processes start properly and display only a single instance of each process.

BZ#1103688

Previously, the quota limits could not be set or configured as the **root squash** feature blacklisted the glusterd client used to configure the quota limits on a brick. With this fix, the **glusterd** client is added to a **white-list** of the **root-squash** exception list. With this fix, quota limit is set without any issue.

BZ#1030432

Previously, even if the quota limit was not set, quota used to send the **quota-deem-statfs** key to the dictionary resulting in incorrect calculations. With this fix, the value of the **size** field for the mount point is cumulative of all the bricks and does not lead to incorrect calculation.

BZ#1095267

Previously, while trying to enable quota again, the system tried to access a NULL transport object leading to a crash. With this fix, a new transport connection is created every time quota is enabled.

BZ#1111468

Previously, a dictionary leak was observed while updating the quota cache and this resulted in high memory consumption leading to an out of memory condition when quota was enabled. With this fix, the quota memory consumption is reduced and a leakage is not observed.

BZ#1020333

Previously, extended attributes namely **trusted.glusterfs.quota.limit-set** and **trusted.glusterfs.volume-id** are visible from any FUSE mount point on the client machine. With this fix, quota related extended attributes is not visible on FUSE mount on client machine. Hence, a client will not be able to read or write to the extended attributes.

BZ#1026227

Previously, stopping a volume displayed **Transport end point not connected state** message in the quota auxiliary mount. With this fix, quota auxiliary mount is unmounted after the volume stop command is executed.

gluster-smb

BZ#1086827

Previously, entries in **/etc/fstab** directory for glusterFS mounts did not have the **_netdev** option. This led to a few systems becoming unresponsive. With this fix, the hook scripts have the **_netdev** option defined for glusterFS mounts in **/etc/fstab** directory and mount operation is successful.

BZ#1111029

Previously, when **chgrp** is performed, **glfs_chown**fails to change the group as the UID is invalid. Hence, **chgrp** operation on any files in CIFS mount fails with **Permission denied** error. With this fix, the libgfapi code has been modified to set GID and **chgrp** does not fail on a CIFS mount, if the user and group has the required permission to perform the operation.

BZ#1104574

Previously, disabling the **user.smb** or **user.cifs** options would start the SMB process. With this fix, a SIGHUP signal is sent to reload the configurations if the SMB process is running, else no action is taken.

BZ#1056012

Previously, when a volume sub directory was exported using Samba in a CTDB setup, the **`log.ctdb`** file would display **`ERROR: samba directory sub-dir not available`** message even if the users were able to access the share. With this fix, the sub directory of a volume is accessible using windows/Linux clients through CTDB and the errors are not seen in the log file.

gluster-snapshot

BZ#1124583

Previously, snapshot bricks are mounted with **`rw`**, **`nouuid`** mount options. With this fix, the mount options used in the original brick is used.

BZ#1132058

Previously, if the brick mount options contained **`=`**, then anything after **`=`** was omitted. For example, mount option **`rw,noatime,allocsize=1MiB,noattr2`** was parsed as **`rw,noatime,allocsize`**.With this fix, this option works as expected.

BZ#1134316

Previously, the default value of **`open fd limit`** was 1024. This was not sufficient and only ~500 bricks could connect to **`glusterd`** with two socket connections for each brick. With this fix, the limit is increased to 65536 and **`glusterd`** connects up to 32768 bricks.

gluster-swift

BZ#1039569

Previously, headers **`X-Delete-At`** and **`X-Delete-After`** were accepted although object expiration feature was not fully implemented, thus leading to confusion. With this fix, the **`X-Delete-At`** and **`X-Delete-After`** headers are not accepted.

glusterfs

BZ#1098971

Previously, rebalance was triggered even if the file was deleted and a directory with the same name was created during the interval between readdir and file-mirgation. Since file migration was attempted using a directory inode, this led to the rebalance process to crash. With this fix, it is ensured that file migration is not attempted, if the file obtained during readdir no longer exists. This is done by looking up for the gfid associated with the name of the file. If a different file/directory is created with the same name, it would get a new gfid and hence the lookup would fail. When the lookup fails, migration of the file is skipped.

BZ#1044646

Previously, if an user running an application belonged to more than approximately 93 groups, the authentication header in the RPC packets sent from the client to the server exceeded the maximum size. This led to an I/O error and the glusterFS client failed to create the RPC packet and did not send anything to the glusterFS bricks. With this fix, users who belong to more than approximately 93 groups can use Red Hat Storage volumes. When the **`server.manage-gids`** option is enabled, the glusterFS Native client is not restricted to 32 groups and the group-ownership permissions based on files/directories is handled more transparently as server side ACL checks are applied to all the groups of a user.

BZ#1018383

The brick processes and QEMU (live migration) use the same range of TCP ports for listening. When live migration fails, retries causes an other port to be used. This caused conflicts and prevented several attempts of live migration to fail. With this fix, a new option **`base-port`** is introduced in **`/etc/glusterd/glusterd.vol`** file and live migration works and does not need to be retried in order to find a free port.

BZ#1110651

Previously, the Distributed Hash (DHT) Table Translator expected the individual sub-volumes to return their local space consumption and availability during file creation as part of **`min-free-disk`** calculation. When the **`quota-deem-statfs`** option is enabled on a volume, the quota translators on each bricks returned the volume-wide space consumption and availability of disk space. This caused DHT to eventually always route all file creations to its first sub-volume, resulting in the incorrect input values it received for **`min-free-disk`** calculation. With this fix, the load of the file creation operation is balanced correctly based on the **`min-free-disk`** criterion.

BZ#1110311

This issue is hit when two or more rebalance processes are acting on same file. After add-brick, if a file hashes to newly added brick, lookup will fail as the file wouldn't be present. In such cases lookup is performed on all the nodes and if a linkto file is found, it gets deleted assuming it to be a stale one (since the previous lookup on hashed-subvolume failed). If rebalance-1 creates a linkto-file on newly added brick as a part of file migration, this linkto-file will be deleted by rebalance-2 which considers it to be stale. Now, since this file was under migration being copied into hashed-subvolume, we would loose the file. The fix is to add careful checks for determining what is considered as a stale linkto file.

BZ#1108570

Previously, when the peer that is probed for was offline and the **`peer-probe`** or **`peer-detach`** commands were executed in quick succession, the **`glusterd`** management service would become unresponsive. With this fix, the **`peer-probe`** and **`peer-detach`** commands work as expected.

BZ#1094716

Previously, **`glusterd`** was not backward compliant with Red Hat Storage 2.1. This lead to peer probe not completing successfully, when probed from a Red Hat Storage 2.1 peer, and lead to **`glusterd`** crashing when peer detach was attempted.With this fix, **`glusterd`** has been fixed to make it backward compliant and peer probes is successful and hence **`glusterd`** does not crash.

BZ#1061580

Previously, when all the bricks in replica group go down while writes are in progress on that replica group, the mount used to hang some times due to stale structures that were not removed from the list. With this fix, removing of stale structures from the list is added to fix the issue.

BZ#1046908

Previously, the **`glusterd`** management service would not maintain the status of rebalance. As a result, after a node reboot, rebalance processes that were complete would also restart. With this fix, after a node reboot the completed rebalance processes do not restart.

BZ#1098691

Previously, earlier releases of **nfs-ganesha** forced the administrator to restart the **nfs-ganesha** server, if an export was added or removed while nfs-ganesha was already started. With this release, you can add and remove exports without restarting the server.

BZ#1057540

Previously, when reading network traces that included WRITE procedures, the details were confusing. A WRITE procedure always had a size of 0 bytes. With this fix, the size of the data for a WRITE procedure is set and Wireshark can be used to display the size of the data.

BZ#1085254

Previously, warning messages were not logged when quota soft limit was met. With this fix, setting the quota **soft-timeout** and **hard-timeout** values to zero ensures logging of warning messages.

BZ#1024459

Previously, creating a hard link where the source and destination files were in the same directory failed in the first attempt. With this fix, hard link creation is successful in the first attempt.

BZ#1091986

A new cluster option, **cluster.op-version** has been introduced which can be used to bump the cluster operating version. The cluster operating version can be bumped using the command **# gluster volume set all cluster.op-version *OP-VERSION***.

The **op-version** will be bumped only if:

- all the peers in the cluster support it, and
- the new **op-version** is greater than the current cluster **op-version**

This set operation will not do any other changes other than changing and saving the cluster **op-version** in the **glusterd.info** file.This feature is only useful for gluster storage pools that have been upgraded from Red Hat Storage 2.1 to Red Hat Storage 3.0. In such a cluster, the only valid value to the key is 3, the **op-version** of RHS-3.0. Hence, setting the option **cluster.op-version** on all volumes will bump up the cluster operating version and allow newer features to be used.

BZ#1108018

Previously, the glusterFS management service was not backward compatible with the Red Hat Storage 2.1 version. As a result, the peers entered the peer reject state during the rolling upgrade from Red Hat Storage 2.1. With this fix, the glusterFS management service is made backward compatible and the peers no longer enter a **peer reject** state.

BZ#1006809

Earlier, **mkdir** failures returned **ENOENT** for the failures due to parents not being present. **DHT-selfheal** considers a brick which returned **ENOENT** during lookup, as part of layout assuming that the lookup might be racing with a **mkdir**. Hence, the newly added brick would be considered as part of directory layout. However, the directory creation itself might have failed because of parents not being present on new brick. Subsequently when a file that is about to be created within that directory hashes to the new brick, it would fail as the parent directory is not present. With this fix, treating parent being absent on a sub-volume (in this case because the directory hierarchy is yet to be constructed on the newly added brick) as **`ESTALE`** error (as opposed to **`ENOENT`**) and as a result, the newly added brick is not considered as part of the layout of a directory and no new files will be hashed to the newly added brick.

BZ#1080245

Previously, the directory structure **`/quota_limit_dir/subdir`** and **`quota_limit_dir`** is set with some limit. When **`quota-deem-statfs`** is enabled, the output of **`df /quota_limit_dir`** would display quota modified values with respect to the **`quota_limit_dir`** where as **`df /quota_limit_dir/subdir`** would display the quota modified values with respect to volume root (/).With this fix, any subdirectory within a quota_limit_dir would show the modified values as in the **`/quota_limit_dir`**. It searches for the nearest parent that has quota limit set and modifies the statvfs with respect to the parent's limit value.

BZ#976902

Previously, peer detach force failed if the peer (to be detached) has bricks as part of a distributed volume. However if the peer holds all of the bricks of that volume and if that peer holds no other bricks, peer detach is successful.

BZ#1003914

Previously, when **`remove-brick commit`** is executed **`remove-brick start`** no warning was displayed and it removes the brick with data loss. With this fix, if **`remove-brick commit`** is executed r**`remove-brick start`**, an error is displayed, **`Removing brick(s) can result in data loss. Do you want to Continue? (y/n) y volume remove-brick commit: failed: Brick 10.70.35.172:/brick0 is not decommissioned. Use start or force option.`**

BZ#970686

Previously, a file could not be unlinked if the hashed subvolume was offline and cached subvolume was online. With this fix, upon unlinking the file, the file on the cached subvolume is deleted and the stale link file on the hashed subvolume is deleted upon lookup with the same name.

BZ#951488

Previously, **`rebalance-status`** command would display the status even if rebalance operation was not running on the volume. This is observed only when remove-brick operation is running on the same volume. With this fix, **`rebalance-status`** would display status only if a rebalance operation is running on the volume..

BZ#921528

Previously, the end of hard link migration, the fop used to return **`ENOTSUP`** for all the cases. Hence, this added to the failure count and the **`remove-brick`** status shows failure for all the files. With this fix, this has now been resolved.

BZ#1058405

Previously, performance/write-behind xlator did not track changes to the size of the file correctly when "extending writes" beyond a "hole at end of the file" are done. Normal reading from the area which was sparsefied (aka hole), hit the server without write-behind flushing the write with offset after the hole, returned an error (since read was done beyond EOF of the file on server). This region was memory mapped and errors during reading through a memory mapped area would trigger a SIGBUS signal. Applications do not normally handle this signal and crash or exit prematurely. With this fix, the performance/write-behind xlator is improved to track the size of the file. With this it can identify writes beyond a hole at the end of file. If a read is done in the hole, it will flush the write before sending read to server. Since, this write has already extended the file on the server, subsequent read wouldn't fail. Hence, applications do not receive an unexpected error or SIGBUS and function the same on glusterfs-fuse as on other filesystems.

BZ#1043566

Previously, on upgrade of *glusterfs-server* package, existing rpmsave files of hook scripts in **`/var/lib/glusterd/hooks/1/`** directory would get re-saved with a **`.rpmsave`** suffix appended resulting in multiple rpmsave files. With this fix, the hook scripts are treated as config files of the package *glusterfs-server* and are saved in a RPM standard way.

BZ#842788

Previously, order of the volume list changed when **`glusterd`** is restarted. With this fix, volumes will be listed in the ascending order always.

glusterfs-fuse

BZ#1086421

Previously, **`mount.glusterfs`** did not return standard error codes. Hence, applications mounting Red Hat Storage volumes over the gluster native protocol, expected to receive well known and documented standard error return values. Returning incorrect/non-standard errors causes confusion to the applications mounting the volumes, in case an error occurred. With this fix, applications do not need spacial error handling for mounting Red Hat Storage volumes, the standard error values get recognized and handled correctly.

glusterfs-geo-replication

BZ#1049014

Previously, when configuration georep_session_working_dir is added in the geo-replication, when upgraded geo-rep session the config file was not updated so geo-rep was unable to get the value of georep_session_working_dir. This led to Geoo-rep worker to crash. With this fix, Geo-rep upgrade is handled in the code, while running geo-replication if it finds georep_session_working_dir is missing then it upgrades the config file and no worker crashes are observed.

BZ#1044420

Previously, when geo-rep worker crashed while geo-rep was trying to handle the signal from a worker thread and due to limitation in python, signals can be handled only in main thread. Hence, geo-rep monitor crashed and syncing does not happen from that node. With this fix, geo-rep worker crash is handled gracefully in the code and if geo-rep worker crashes, geo-rep monitor will crash.

BZ#1095314

In Geo replication, working directories for changelog consumption were stored under /var/run/gluster/master/slave-url/brick-hash and now at /var/run/gluster*. Reason: /var/run/gluster* is not picked by sos-report and on reboot content of that Directory might wiped out. Result (if any): Change in location of changelog consumption logs and working directory for Geo-rep changelog consumption.

BZ#1105323

Previously, ping was used to check the connectivity of slave, even though ping enable is not required in slave to start geo-rep session. Hence, Geo-rep create failed if ping is disabled in slave. Now with this fix, Geo-rep now checks only ssh connectivity to slave and Geo-rep create does not fail even though ping is disabled by firewall.

BZ#1101910

Previously, if the user is created without primary group in mount-broker setup, geo-rep fails to set proper ownership of .ssh and authorized keys. Hence, the mount-broker setup failed and the right permissions for .ssh and authorized keys were set manually. With this fix, this issue has been resolved.

BZ#1064597

Previously, when using replicate volume in geo-replication all the bricks participated in syncing data to slave. If bricks are replica pair, one will become active and other one will be passive. If a node goes down, passive brick may become active and vice versa. The switching interval was 60 sec. So even if a node goes down, it was not switching immediately. Hence, this led to delay in syncing data to slave. With this fix, switching time is reduced to 1 sec, so that a passive node immediately becomes active when other node goes down and the delay in syncing is reduced.

BZ#1030052

During a Geo-replication session, the gsyncd process restarts when you set use-tarssh, a Geo-replication configuration option to true even if it is already being set.

BZ#1030393

Previously, when tar+ssh is used as the sync engine, due to an fd leak, the open descriptor count will cross the max allowed limit and cause the gsync daemon to crash. This led to fix file descriptor leak. With this fix, no geo-rep worker crash is observed.

BZ#1111577

Previously, Geo-replication synchronizes files through hybrid crawl after it completes full file system crawl and did not use changelogs during that time. Due to this, deletes and renames happened during that window is not propagated to slave. Hence, slave will have additional files compared to Master.

BZ#1038994

Previously, when a Passive node becomes active it collects the old changelogs to process, geo-rep identifies and removes respective changelog file from the list if it is already processed. If list is empty geo-rep worker was crashing since it was unable to process empty list. This led to Geo-rep worker crash. With this fix, Geo-rep handles the empty list of changelog files and no Geo-rep worker crash is observed.

BZ#1113471

The Geo-replication does not use xsync crawl for the first time but uses history crawl even when change detector set to xsync.

BZ#1110672

Previously while establishing a geo-replication session, the master volume and slave volume sizes were not computed properly and as a result, the geo-replication sessions could not be created. With this fix, the calculation errors are fixed and geo-replication session creation succeeds.

BZ#1098053

With this fix, a support for non-root privileged slave volume is added by tweaking the current geo-rep setup process and scripts, without affectiong regular (root-privileged) master-slave sessions.

BZ#1111587

When force recursive deletes (rm -rf) command is run on master, the directories were not deleted in all distribute nodes in the backend for slave because of order of entrylocks was leading to deadlock and the slave mounts were hanging. Fixed the ordering issue so that all mounts take the lock in same order to fix the deadlock thus this issue.

BZ#1058999

Previously, when the gsyncd.conf for a particular geo-rep session had a missing state-file or pid-file entry, glusterd did not leverage the default template where the information is present.This led to geo-rep status becoming defunct. With this fix, if entries such as **`state_file`** or **`pid-file`** are missing in the **`gsyncd.conf`** or if the **`gsyncd.conf`** is also missing, **`glusterd`** looks for the missing configs in the **`gsyncd_template.conf`**.

BZ#1104121

Previously, while setting up mount-broker geo-replication if the entire slave url is not provided, the status shows "Config Corrupted". With this fix, you must provide the entire slave url while setting up mount-broker geo-replication.

glusterfs-server

BZ#1095686

Previously, the server quorum framework in glusterd would perform the quorum action (start or stop bricks) unconditionally on a quorum event, even if the new event did not cause the quorum status to change. This could cause bricks which were taken down for maintenance to be started in the middle of maintenance. With this fix, the current and previous quorum status are checked before attempting to start or stop bricks. Bricks are only started or stopped if the quorum status changed. Bricks brought down for maintenance will no longer be started on spurious quorum events.

BZ#1065862

Previously, when one or more nodes in the cluster is off line, gluster CLI commands may be hung. In this release, with the introduction of ping-timer for glusterd peer connections, commands would fail if one or more nodes are off line, after ping-timeout seconds. By default, the ping-timeout is configured as 30 secs for glusterd connections.

BZ#1029444

Previously we are able to get/set the "trusted.glusterfs.volume-id" extended-attribute from the mountpoint. After the fix xattr 'trusted.glusterfs.volume-id' not show on the mount point and throws permission error when tried to set this xattr.

BZ#1096614

An enhancemenet has been added to the eaddir-ahead translator. This is enabled by default on newly created volumes in Red Hat Storage 3.0 which improves the readdir performance for the new volumes.

> **Note**
>
> readdir-ahead is not compatible with RHS-2.1, so new volumes created with RHS-3.0 cannot be used with RHS-2.1 clients until readdir-ahead is disabled.

BZ#1108505

Previously, the way **quotad** was being started on the new peer when peer probed, lead to **glusterd** being deadlocked. Hence, the peer probe command failed. With this fix, **quotad** is now started in a non-blocking way during peer probe which no longer blocks **quotad** and peer probe is successfully.

BZ#1109150

Previously, when multiple snapshot operations were performed simultaneously from different nodes in a cluster, the **glusterd** daemon peers gets disconnected by ping-timer. Now with this fix, you must disable the ping-timer by setting the **ping-timeout** to **0** in **/etc/glusterfs/glusterd.vol** file and restart gluster daemon service and the peers do not get disconnected by ping-timer.

BZ#1035042

Previously, entries in **/etc/fstab for glusterfs** mounts did not have **_netdev** option. This led to some systems becoming unresponsive. With this fix, the hook scripts have **_netdev** option defined for glusterFS mounts in the **/etc/fstab** and the mount operation is successful.

BZ#891352

Red Hat Storage Snapshot is a new feature which has been included in this release. This feature enables you to take snapshot of an online (started) Red Hat Storage volume. This is a crash consistent snapshot of the specified Red Hat Storage volume. During snapshot some of the entry fops is blocked to achieve crash consistency. Snapshot feature is based from thinly provisioned LVM snapshot. Therefore to take a snapshot, all the Red Hat Storage volume bricks must be on an independent thinly provisioned LVM. The resultant snapshot is a read-only Red Hat Storage volume, which can be only mounted via FUSE.

BZ#1048749

Previously, a subdirectory mount request was successful even though the host was configured with the **nfs.rpc-auth-reject** option. With this fix, the clients requesting the mount are validated against the **nfs.rpc-auth-reject** irrespective of type of mount (either the volume mount or subdirectory mount). As a result, if the host is configured with **nfs.rpc-auth-reject**, the mount request from the same host would fail for any type of mount requests.

BZ#1046284

Previously, while executing **gluster volume remove-brick** without any option, it defaults to force commit which resulted in data loss. With this fix, remove-brick cannot be executed without an explicit option. You must provide the option in the command line **volume remove-brick *VOLNAME* [replica *COUNT*] BRICK ... start|stop|status|commit|force**, else the command displays an error.

BZ#969993

Previously, **gluster volume set help** did not display the configuration options for **white-behind** performance translator namely:

write-behind performance translator namely:

- performance.nfs.flush-behind
- performance.nfs.write-behind-window-size
- performance.nfs.strict-o-direct
- performance.nfs.strict-write-ordering

With this fix, the options are displayed with description.

BZ#1006772

Previously, if NFS server did not access the NLM port number of the NFS client, then server log displayed **Unable to get NLM port of the client. Is the firewall running on client? OR Are RPC services running (rpcinfo -p)?** instead of **Unable to get NLM port of the client. Is the firewall running on client?**. With this fix, this issue has been resolved.

BZ#1043915

In this release, two new volume tuning options are introduced in the **gluster volume set volname** command namely **server.anonuid** and **server.anongid**. These options make it possible to define a UID and GID that is used for anonymous access. These options are defined per volume and the **server.root-squash** option must be enabled with these options.

BZ#1071377

Previously, if length of the volume name, sub folders is more than 256 characters in the brick path, and brick vol file length is more than 256 characters, error messages were displayed. Now with this fix, more than 256 characters is not allowed.

BZ#1109795

Previously, a deadlock in the changelog translator caused the I/O operations to stall and resulted in the file system becoming unresponsive. With this fix, no deadlocks are observed during interruptions in the locked regions.

nfs-ganesha

BZ#1091921

Two new commands, **gluster vol set volname nfs-ganesha.host IP** and **gluster vol set volname nfs-ganesha.enable ON**are introduced with this fix which enable you to use glusterfs volume set options to export/unexport volumes through nfs-ganesha.

BZ#1104016

With this release a new option, **Disable_ACL**, is added to nfs-ganesha. This option helps in enabling or disabling ACL. Setting this option to **true** disables ACLs and setting this option to **false** enables ACLs.

Chapter 6. RHEA-2014:1277

The bugs contained in this chapter are addressed by advisory RHEA-2014:1277. Further information about this advisory is available at https:/rhn.redhat.com/errata/RHEA-2014-1277.html.

redhat-access-plugin-rhsc

BZ#1054034

Previously, if the **`Cancel`** button was clicked on the Red Hat Access Login window, it would not allow you to retry logging in to Red Hat Access again by clicking on the **`Log in`** button. With this release, the **`Login`** button works as expected.

rhsc

BZ#1089067

Previously, there was no error handling capability for the command: **`rhsc-setup --generate-answer=<answer-file>`**. If an invalid answer file was provided, the Red Hat Storage Console setup script would fail with an error. With this release, the error is handled while writing the answer file. If an invalid path is provided, the setup reports the error as a warning and continues to function as expected.

BZ#1044847

Previously, the host column could not be sorted on the **Services** tab of Clusters when the **Show All** view was clicked. The order of the rows would get interchanged with every refresh task. With this enhancement, the **Host** column entries are sorted before they are displayed on the Console.

BZ#1061725

Previously if the **Status** dialog box was open and simultaneously a remove-brick operation was stopped from the CLI, the task was displayed as **`Commit Pending`** because the status dialog box would return the status as **`Completed`**. This resulted in an incorrect status message on the Console. With this fix, the **Status** Dialog box displays the correct status for a stop remove-brick operation.

BZ#1063923

Previously, administrators of Red Hat Storage deployments had no easy mechanism to track the health of a server. A poll-based mechanism used the existing **`glusterFS`** CLI to identify the volume status and node status. A five minute polling interval displayed stale data. In this release, with the Nagios plugin integration, the Red Hat Storage Console has monitoring capabilities such as:

- Monitoring of critical entities such as servers, networking, volumes, clusters and services.
- Alerting when critical infrastructure components fail and recover, providing administrators with notice of important events. Alerts can be delivered via email and SNMP.
- Reports providing a historical record of outages, events and notifications for later review.
- Trending and capacity planning graphs and reports that allow for infrastructure upgrades before failures.

BZ#1064712

Previously the **Skipped File Count** field always displayed zero on the **Remove Brick Status** dialog box. In this release, the **Skipped File Count** field is removed.

BZ#1084891

Previously, the Red Hat Storage Console did not display performance metrics and lacked monitoring capability. With this release, a new monitoring feature is introduced to display graphs and utilization trends for clusters, volumes, and bricks. It also displays host network utilization, memory utilization, CPU utilization, swap space and disk utilization.

BZ#1064295

Previously while performing a remove brick operation, clicking the **Remove** button before the pop-up closed on **Remove Brick** window led to a remove brick operation failure, and the remove brick icon was not displayed in the **Activities** column. With this fix, the **Remove-brick** icon appears in the volume activities column, the tasks in the task pane are updated as expected, and an appropriate message is displayed if the remove brick icon is clicked when a task is already in progress.

BZ#1065227

Previously, the glusterFS task list information would consume a considerable amount of time to synchronize with other nodes to provide consistent information about the newly created tasks. If the glusterFS task list did not return the information about a task, the task was marked as **Unknown**. Although the task is active, the Console would fail to monitor it. With this fix, a minimum wait time of 10 minutes is introduced before a task is cleared. As a result, the task information is displayed correctly on the Red Hat Storage Console.

BZ#998928

Previously, there were no errors reported when you start the **ovirt-engine-notifier** and there was no notification that the **ovirt-engine-notifier** started successfully. With this fix, the error message **No transport is enabled, nothing to do** is displayed when starting the **ovirt-engine-notifier** when **MAIL_SERVER** option in the configuration file is not defined.

BZ#1032533

Previously, after logging in to the Red Hat Storage Console, an additional HTTP authentication dialog box was displayed with the user name and password prompt. With this fix, the additional dialog box is not displayed.

BZ#1044598

Previously, when the start **remove-brick** operation failed, a few localization constants were displayed instead of a comprehensible error message. With this fix, the localization constants are properly mapped to appropriate messages.

Revision History

Revision 3-14 **Thu Jan 15 2015** **Shalaka Harne**

Updated Technical Notes for Red Hat Storage 3.0.3 GA, RHBA-2015:0038 and RHBA-2015:0039.

Revision 3-11 **Mon Jan 12 2015** **Pavithra Srinivasan**

Updated Technical Notes for Red Hat Storage Console 3.0.3 GA, RHBA-2015:0038 and RHBA-2015:0039.

Revision 3-10 **Mon Nov 10 2014** **Bhavana Mohan**

Updated Technical Notes for Red Hat Storage 3.0.2 GA, RHBA-2014:1820 and RHBA-2014:1819.

Revision 3-8 **Fri Nov 07 2014** **Shalaka Harne**

Updated Technical Notes for Red Hat Storage Console 3.0.2 GA, RHBA-2014:1820 and RHBA-2014:1819.

Revision 3-7 **Mon Sep 22 2014** **Anjana Suparna Sriram**

Version for 3.0 GA release, RHEA-2014:1278 and RHEA-2014:1277.