Unbreakable Enterprise Kernel

Release Notes for Unbreakable Enterprise Kernel Release 4 Update 6

E92390-13
October 2024

ORACLE®

Unbreakable Enterprise Kernel Release Notes for Unbreakable Enterprise Kernel Release 4 Update 6,

E92390-13

Contents

1   New Features and Changes

2   Fixed Issues

3   Known Issues

4   Installation and Availability

Preface

> **⚠ Important:**
>
> The software described in this documentation is either in Extended Support or Sustaining Support. See Oracle Open Source Support Policies for more information.
>
> We recommend that you upgrade the software described by this documentation as soon as possible.

Unbreakable Enterprise Kernel: Release Notes for Unbreakable Enterprise Kernel Release 4 Update 6 (4.1.12-112) provides a summary of the new features, changes, and known issues in the Unbreakable Enterprise Kernel Release 4 Update 6.

Conventions

The following text conventions are used in this document:

Convention	Meaning
boldface	Boldface type indicates graphical user interface elements associated with an action, or terms defined in text or the glossary.
italic	Italic type indicates book titles, emphasis, or placeholder variables for which you supply particular values.
`monospace`	Monospace type indicates commands within a paragraph, URLs, code in examples, text that appears on the screen, or text that you enter.

Documentation Accessibility

For information about Oracle's commitment to accessibility, visit the Oracle Accessibility Program website at https://www.oracle.com/corporate/accessibility/.

Access to Oracle Support for Accessibility

Oracle customers that have purchased support have access to electronic support through My Oracle Support. For information, visit https://www.oracle.com/corporate/accessibility/learning-support.html#support-tab.

Diversity and Inclusion

Oracle is fully committed to diversity and inclusion. Oracle respects and values having a diverse workforce that increases thought leadership and innovation. As part of our initiative to build a more inclusive culture that positively impacts our employees, customers, and partners, we are working to remove insensitive terms from our products and documentation. We are also mindful of the necessity to maintain compatibility with our customers' existing technologies and the need to ensure continuity of service as Oracle's offerings and industry standards evolve. Because of these technical constraints, our effort to remove insensitive terms is ongoing and will take time and external cooperation.

1

New Features and Changes

> **⚠ Important:**
>
> The software described in this documentation is either in Extended Support or Sustaining Support. See Oracle Open Source Support Policies for more information.
>
> We recommend that you upgrade the software described by this documentation as soon as possible.

The Unbreakable Enterprise Kernel Release 4 (UEK R4) is Oracle's fourth major release of its heavily tested and optimized operating system kernel for Oracle Linux 6 Update 7 or later, and Oracle Linux 7 Update 1 or later, on the x86-64 architecture. It is based on the mainline Linux kernel version 4.1.12.

UEK R4 update 6 uses the 4.1.12-112.14.1 version and build of the UEK R4 kernel, which includes security and bug fixes, as well as driver updates. This kernel has been tested within environments running the latest available Oracle Linux releases: Oracle Linux 6 Update 8, Oracle Linux 6 Update 9, Oracle Linux 7 Update 3, and Oracle Linux 7 Update 4.

Oracle actively monitors upstream check-ins and applies critical bug and security fixes to UEK R4.

UEK R4 uses the same versioning model as the mainline Linux kernel version. It is possible that some applications might not understand the 4.1 versioning scheme. However, regular Linux applications are usually neither aware of nor affected by Linux kernel version numbers.

Notable Changes

The following notable changes and features are included in this update:

- **Automatic NUMA balancing disabled by default**

  In this update, the auto-enabling of automatic Non-uniform memory access (NUMA) balancing has been disabled. This change addresses several issues that were encountered on systems with multiple NUMA nodes, where automatic NUMA balancing was enabled. Symptoms that were reported included high `iowait` times and numerous processes (like Oracle DB processes) observed in the `D` state, that were sitting on the `wait_on_page_bit()` function in the process stack.

  You can manually enable automatic NUMA balancing by setting the `kernel.numa_balancing` kernel parameter to `1`.

  To set the `kernel.numa_balancing` kernel parameter at runtime, use the `sysctl` command, as follows:

  ```
  # sysctl -w kernel.numa_balancing=1
  ```

  To make the enabling of automatic NUMA balancing persistent across a reboot, add the `kernel.numa_balancing=1` parameter to the `/etc/sysctl.conf` file.

To verify whether the `kernel.numa_balancing` kernel parameter is set, run the following command:

```
# sysctl kernel.numa_balancing
```

- **Deferred compaction feature backported and set to `madvise` by default**

  The deferred compaction feature for Transparent Huge Pages (THP) has been backported from the mainline kernel, and the default `defrag` behavior has been set to `madvise`. This fix resolves an issue where THP could cause application stalls in the event that a huge page allocation from THP took a long time due to memory fragmentation.

- **FRWR support added in RDS**

  This update adds support for MR registration through work requests in RDS, which is commonly referred to as FRWR/fastreg/FRMR. With this change, RDS can choose either FMR or FRWR as the registration method. The choice is based on the preference that is specified in the `prefer_frwr` module parameter, as well as the support that is offered by the underlying device.

- **`ibacm` available in the Oracle OFED release**

  Starting with the Oracle OFED 1.0.0-33 release, `ibacm` includes Netlink support. The `ibacm` service processes path record queries and also caches path records. Because `ibacm` provides an easy path record cache for kernel components, this implementation greatly improves performance on large fabric systems.

- **`libnvdimm` subsystem added to kernel**

  This update adds the `libnvdimm` kernel subsystem, which is responsible for the detection, configuration, and management of Non-Volatile Dual Inline Memory Modules (NVDIMMs). As a result, if NVDIMMs are present in the system, they are exposed through the `/dev/pmem*` device nodes and can be configured by using the `ndctl` utility.

- **`panic_on_rcu_stall` parameter added to `sysctl` to debug RCU stalls on a `vmcore`**

  The `panic_on_rcu_stall` interface is useful for defining the root cause of RCU stalls when using a `vmcore`. You can set the parameter's value to `1` to call the `panic()` function whenever an RCU stall is detected.

- **Target Core Module in Userspace (TCMU) support for in-kernel iSCSI target**

  This feature provides support for userspace pass-through, which enables userspace modules to be specified as iSCSI targets. Typically, built-in modules are used as backstores or storage engines for SCSI devices and are implemented entirely as kernel code. Backstores cover common use cases, but some target solutions can support alternate backstores and act as translators for initiators that might need to store data on these non-traditional storage systems. By supporting TCMU, a small adapter module can be released for each backstore. This functionality is similar to Filesystem in Userspace (FUSE), but at the SCSI layer rather than the file system layer.

- **`uio_hv_generic` driver enabled**

  The `uio_hv-generic` driver is a generic driver that can be bound to any Hyper-V VMBus device, providing there is userspace access to networking and adequate storage is available. The addition of this driver benefits both virtio and Xen drivers.

Crypto API Improvements

The following Crypto API improvements have been made in this update:

- **Bug fixes**

  Several bug fixes have been applied to provide greater security.

- `ccp` **driver added**

  The `ccp` driver provides support for the AMD Cryptographic Coprocessor (CCP). The AMD CCP provides hardware encryption, hashing, and other related operations. The driver version that is provided is 1.1.0. In addition, several upstream patches have been applied to resolve initial bugs that were found with this driver.

- **Hash handling fixes**

  Fixes have been applied to various components within the Crypto API to better handle NULL hashes. In particular, a fix has been applied to the `algif_hash` API to fix a crash that was caused in the digest operation when a NULL hash was used with the `shash` algorithm.

  A fix has also been applied to fix an `ahash` EINPROGRESS notification callback loop that could cause a denial of service.

- **Jitter Entropy RNG added**

  The Jitter Entropy Random Number Generator (RNG) collects entropy through CPU timing differences to the Linux kernel. This feature is available by default through the `algif_rng` interface. Generated numbers can be added back to the kernel through the `/dev/random` file, which then makes these numbers available to other `/dev/random` users. The result is that the operating system now has more sources of entropy available.

- `testmgr` **bug fix for out-of-bounds access**

  The `testmgr` code that is used to test software implementations was generating an out-of-bounds access error when running some `esp aead` tests. This issue was due to the limited IV size of the `aes_gcm_enc`/`dec test` templates and the way in which the `testmgr` code handled the input from `memcpy`. To prevent this error from occurring, the array was increased to an appropriate size.

DTrace Improvements

A number of bug fixes and enhancements including module and utility updates are included for DTrace on UEK R4u6, bringing the current version to 0.6.1.

The following are other notable changes that are included:

- **Bug fixes**

  Numerous bug fixes have been applied to provide greater stability and better performance.

- **DTrace lockstat provider added**

  A patch that adds a definition for the lockstat DTrace provider has been added in this update. DTrace lockstat support enables the dynamic tracing of kernel locking events, for example, which locks are most frequently used, which locks exhibit the most contention, and which locks are held the longest.

- **Dynamically sized** `tracemem()` **action added**

  An optional third argument form has been added to the `tracemem()` action to support variable-sized structures. This feature was previously added to Oracle Solaris DTrace.

- **FBT improvements**

  Several patches and bug fixes were applied to improve performance and expand the capability of the Function Boundary Tracing (FBT) module.

- **I/O provider added for NFS**

  In this update, DTrace I/O provider `start` and `done` probes have been added for NFS and DTrace I/O provider probes have been added for NFS `read` and `write` requests.

- **`llquantize()` aggregation added**

  The linear-log quantize aggregating action, `llquantize()`, provides log/linear results and enables you to collect data in linear-step buckets. This feature was previously added to Oracle Solaris.

- **Lock-free algorithm implemented for IPI function calls**

  The implementation of the lock-free algorithm reduces the amount of IPI (interprocessor interrupts) function calls that are observed on the system. Instead of relying on IPI traps that can cause a barrier, the lock-free algorithm is more lightweight,thereby reducing stress to the kernel and CPU, and improving system performance.

- **Structure member names can be the same as D keywords**

  A bug fix has been applied to fix a problem where syntax errors occurred if structure member names clashed with D keywords, for example, `struct` and `group.self`. You can now refer to structure members by using the same names as D keywords.

- **USDT support for 32-bit applications on 64-bit hosts**

  A bug fix has been applied to enable user-level, statically defined tracing (USDT) support for 32-bit applications on 64-bit hosts.

Xen Hypervisor Improvements

Several patches for the Xen Hypervisor that were backported from changes in the Linux 4.11 and 4.12 kernel releases are included in this update. Many of these patches improve performance, error handling, and also provide general bug fixes.

The following are the notable fixes and improvements that have been made in this update:

- A patch was applied to the Xen memory management code to resolve an issue where overlapping memory allocation could have resulted in a kernel panic on a guest virtual machine.

- A patch was applied to prevent the printing of the following unnecessary error message, which appears in the `dmesg` output when booting a PV guest:

  ```
  Unable to read sysrq code in control/sysrq
  ```

- Patches have been applied in this update to improve security for Xen.

- This update includes improved support for NVMe devices by Xen guests.

- Patches were applied and updated in the Xen `blkfront` module to optimize the management of grants in the `blkfront` ring by ensuring that persistent grants are always allocated first. This change is also implemented to avoid an I/O hang when there are very few, but enough, grants available.

- A patch has been applied to fix a race condition that could result in an I/O hang in the `blkfront` driver.

- Logging improvements that help with debugging grant management in Xen block drivers have been included in this update.

File System Improvements

The following file systems improvements have been made:

btrfs Updates

- A patch was applied to include the name and ID of a subvolume or snapshot in `/proc/mounts`. This change makes it easier to identify whether `/` is mounted as a subvolume or as a snapshot, where the `root` partition is formatted using `btrfs`.

- Several patches were applied to fix issues surrounding deduping of inodes and to resolve an issue where `ctime` and `mtime` timestamps were updated during a dedupe operation . Another issue with unaligned block sizes, or when deduping past EOFs that could result in an error while deduping, was also resolved.

- Several patches were applied to bring `btrfs` in line with userspace tools that support options requiring a new free space `btree` mount feature in the kernel.

- Several patches were applied to support an enhanced structure that carries the device ID as a user argument when deleting a device by ID. This change brings `btrfs` in line with userspace tools that support this feature.

NFS Updates

- New static NFS trace points have been added in this update to provide more efficiency in debugging. These trace points are in addition to those that already exist in the Linux kernel NFS client.

  The following six trace points are provided specifically for capturing NFS I/O operations `nfs:nfs_initiate_read`, `nfs:nfs_readpage_done`, `nfs:nfs_initiate_write`, `nfs:nfs_writeback_done`, `nfs:nfs_initiate_commit`, and `nfs:nfs_commit_done`.

  You have the option of enabling the trace points individually or simultaneously. Each generated event includes timestamp information, as well as information about the CPU core and process that requested the operation.

  You use the `trace-cmd` tool to control NFS trace points, in the same way that you control other static trace points in the kernel. Use the `trace-cmd list | grep nfs` command to discover trace points that are related to the kernel NFS client.

- An issue that caused the NFS4v1 client to discard its lease and apply for a new lease during transparent state migration has been fixed. This issue significantly reduced any benefit of transparent state migration. A second issue for transparent state migration was also identified when a client lost its sequence ID for a lease that was migrated. A fix was applied to cause the client to save and use the contrived slot sequence value that is returned by the destination server.

- Security fixes were backported to be included in this release, including a check for oversized NFSv2 and NFSv3 arguments that could lead to Denial of Service (DoS); and a set of patches to resolve an issue that caused an NFSv4 server to be vulnerable to a remote DoS attack due to improper validation of layout type while processing the pNFS LAYOUTGET operand.

OCFS2 Updates

- Several patches were backported to fix an issue that caused recursive cluster locking resulting in deadlocks that could hang nodes in the cluster.

- A patch was included to fix a deadlock issue that could result during inode creation using POSIX ACLs. The code did not account for cluster-wide inode locks, which could cause a deadlock if there is a blocked remote lock request waiting for the lock to be downconverted.

XFS Updates

- Several patches were applied to fix numerous bugs pertaining to XFS error handling and behavior. Other changes include enhancements that provide improved configurable error support for XFS.

  You can tweak XFS error handling through the `sysfs` interface, as shown in this example:

  ```
  # ls -l /sys/fs/xfs/dm-1/error/metadata/
  total 0
  drwxr-xr-x. 2 root root 0 Oct 24 16:54 default
  drwxr-xr-x. 2 root root 0 Oct 24 16:54 EIO
  drwxr-xr-x. 2 root root 0 Oct 24 16:54 ENODEV
  drwxr-xr-x. 2 root root 0 Oct 24 16:54 ENOSPC

  # ls -l /sys/fs/xfs/dm-1/error/metadata/default/
  total 0
  -rw-r--r--. 1 root root 4096 Oct 24 16:54 max_retries
  -rw-r--r--. 1 root root 4096 Oct 24 16:54 retry_timeout_seconds

  # ls -l /sys/fs/xfs/dm-1/error/metadata/ENOSPC/
  total 0
  -rw-r--r--. 1 root root 4096 Oct 24 16:54 max_retries
  -rw-r--r--. 1 root root 4096 Oct 24 16:54 retry_timeout_seconds
  ```

- Patches were applied to resolve general bugs and provide some performance enhancements. Fixes included: a backported patch to fix a deadlock issue that resulted when work queue log processes hung while waiting for each others to complete; a couple of patches to resolve an issue around read-only remounts that resulted in the file system reporting failed orphan inode recovery on remount; and a patch to resolve a race condition that could cause file corruption during a file-extending direct I/O write.

Driver Updates

The Unbreakable Enterprise Kernel supports a wide range of hardware and devices. In close cooperation with hardware and storage vendors, several device drivers have been updated or added by Oracle, per the information in the following table:

Driver	Version	Description
aacraid	1.2-1[41010]-ms + Patches	Adaptec Advanced Raid Products
be2net	11.4.0.0	Broadcom/Emulex OneConnect 10Gbps NIC Driver
bnx2x	1.713.10 + Patches	QLogic BCM57710/57711/57711E/ 57712/57712_MF/57800/57800_MF/ 57810/57810_MF/57840/57840_MF Driver
bnxt_en	1.8.0 + Patches	Broadcom/Emulex BCM573xx NIC Driver
ccp	1.1.0	AMD Cryptographic Coprocessor Driver
ccp-crypto	1.0.0	AMD Cryptographic Coprocessor Crypto API Support

Driver	Version	Description
e1000	7.3.21-k8-NAPI + Patches	Intel(R) PRO/1000 Network Driver
e1000e	3.2.6-k + Patches	Intel(R) PRO/1000 Network Driver
fnic	1.6.0.34	Cisco FCoE HBA Driver
i40e	2.1.14-k	Intel® Ethernet Connection XL710 Network Driver
i40evf	3.0.0-k	Intel® XL710 X710 Virtual Function Network Driver
igb	5.4.0-k	Intel® Gigabit Ethernet Network Driver
igbvf	2.4.0-k	Intel® Gigabit Virtual Function Driver
ixgbe	5.1.0-k	Intel® 10 Gigabit PCI Express Network Driver
ixgbevf	4.1.0-k + Patches	Intel® 82599 Virtual Function Driver
lpfc	0:11.4.0.2	Broadcom/Emulex LightPulse Fibre Channel SCSI Driver
megaraid_sas	07.701.17.00-rc1	Avago MegaRAID SAS Driver
mgag200	Patches	Matrox GP200e4 Device Driver
mpt3sas	16.100.00.00	LSI MPT Fusion SAS 3.0 Device Driver
nvme nvme-core	1.0 + Patches	NVMe Block Device Driver and Core Support
qed qede	8.10.11.21 8.10.10.21	QLogic FastLinQ 4xxxx Ethernet Driver and Core Module
qedi	8.10.4.0	QLogic Storage Driver
qla2xxx	9.00.00.00.40.0-k	QLogic Fibre Channel HBA Driver
smartpqi	1.1.2-126	Microsemi Smart Family Controller Driver
uio_hv_generic	0.02.0	Userspace I/O Driver for VMBus
xen-blkback xen-blkfront xen-netback	Patches	Xen Virtual Block Device and Xen Virtual Network Device drivers
xscore xsvhba xsvnic xve	6.0.r8044	Oracle Virtual Network Driver Modules for Core Support, VHBA, XSVNIC and Virtual Ethernet.

Additional Notes For Driver Updates

The following notes are included at the request of a vendor for the listed driver:

- lpfc: Locked optics support is enabled for LPE32000 HBAs and all variant HBAs of this architecture. With this capability, these HBAs will detect and enable both Avago or Emulex certified SFP and QSFP optics.

For driver rev 11.0.0.13 and higher unqualified optics will be disabled, the link will not come up, an error message is written to the log file and the lpfc driver will display this message:

```
3176 Port Name [wwpn] Unqualified optics - Replace with Avago optics for Warranty
and Technical support
```

- `mpt3sas`: As of UEK R4 update 4, the `mpt2sas` driver has been merged with the `mpt3sas` driver to provide a single driver module that supports both SAS 2.0 and SAS 3.0 HBAs. Changes have been applied to `dracut` to correctly handle the module aliases for the migration to a single driver module.

Technology Preview

The following features that are included in the Unbreakable Enterprise Kernel Release 4 are still under development, but are made available for testing and evaluation purposes:

- **DCTCP (Data Center TCP)**

  DCTCP enhances congestion control by making use of the Explicit Congestion Notification (ECN) feature of state-of-the-art network switches. DCTCP reduces buffer occupancy and improves throughput by allowing a system to react more intelligently to congestion than is possible using TCP.

- **DRBD (Distributed Replicated Block Device)**

  A shared-nothing, synchronously replicated block device (*RAID1 over network*), designed to serve as a building block for high availability (HA) clusters. It requires a cluster manager (for example, pacemaker) for automatic failover.

- **Kernel module signing facility**

  Applies cryptographic signature checking to modules on module load, checking the signature against a ring of public keys compiled into the kernel. GPG is used to do the cryptographic work and determines the format of the signature and key data.

- **Server-side parallel NFS**

  Server-side parallel NFS (pNFS) improves the scalability and performance of an NFS server by making file metadata and data available on separate paths.

Compatibility

Oracle Linux maintains user-space compatibility with Red Hat Enterprise Linux (RHEL), which is independent of the kernel version running underneath the operating system. Existing applications in user space will continue to run unmodified on the Unbreakable Enterprise Kernel Release 4 and no re-certifications are needed for RHEL certified applications.

To minimize impact on interoperability during releases, the Oracle Linux team works closely with third-party vendors whose hardware and software have dependencies on kernel modules. The kernel ABI for UEK R4 will remain unchanged in all subsequent updates to the initial release. In this release, there are changes to the kernel ABI relative to UEK R3 that require recompilation of third-party kernel modules on the system. Before installing UEK R4, verify its support status with your application vendor.

Header Packages for Development

As of UEK-3.8-QU2, the `kernel-uek-headers` package is no longer built and distributed. There are three kernel packages that might be useful for development purposes. The `kernel-headers` package forms part of the API for user space programs. The `kernel-devel` package is

used for standard RHCK development and module compilation. The `kernel-uek-devel` package is used for UEK development and module compilation. Neither the `kernel-uek-headers`, nor the `kernel-headers` packages, are needed for kernel development.

The `kernel-headers` package provides the C header files that specify the interface between user-space binaries or libraries and UEK or RHCK. These header files define the structures and constants that you need to build most standard programs or to rebuild the `glibc` package.

The `kernel-devel` and `kernel-uek-devel` packages provide the kernel headers and makefiles that you need to build modules against UEK and RHCK.

To install the packages required to build modules against UEK and the C header files for both UEK and RHCK:

```
# yum install kernel-uek-devel-`uname -r` kernel-headers
```

2
Fixed Issues

> **⊘ Important:**
>
> The software described in this documentation is either in Extended Support or Sustaining Support. See Oracle Open Source Support Policies for more information.
>
> We recommend that you upgrade the software described by this documentation as soon as possible.

This chapter describes fixed issues for the Unbreakable Enterprise Kernel Release 4.

> **⊘ Important:**
>
> Run the `yum update` command regularly to ensure that the latest bug fixes and security errata are installed on your system.

- **Docker issues encountered when the `overlay2` storage driver and SELinux was enabled and set to Enforcing mode fixed**

  The issues that were encountered with Docker containers when the Docker engine was configured to run with the `overlay2` storage driver and SELinux enabled and set to Enforcing mode has been fixed. (Bug ID 25684456)

- **ext4: System hang issue when processing corrupted orphaned inode list fixed**

  An issue with a corrupted orphaned inode list that resulted in the inode being processed repeatedly, and therefore hanging the system, has been fixed in this update. (Bug ID 24433290)

- **XFS: File system corruption occurs after direct I/O writes fixed**

  A race condition that resulted in post-eof blocks being used for direct I/O writes causing corruption in the file system has been fixed. (Bug ID 26128822)

- **ifup-ib: Permission denied error is fixed**

  An issue that resulted in a 'Permission denied' error when bringing up an InfiniBand interface has been resolved with the release of rdma-3.10-3.0.37 that is included for use with UEK R4u6. (Bug ID 26197105)

3

Known Issues

> **⚠ Important:**
>
> The software described in this documentation is either in Extended Support or Sustaining Support. See Oracle Open Source Support Policies for more information.
>
> We recommend that you upgrade the software described by this documentation as soon as possible.

This chapter describes the known issues in this update.

KVM guests with less than 4 GB of memory might fail to auto reserve crash kernel memory

KVM guests that are running UEK R4 might fail to auto reserve memory for a crash kernel if the `crashkernel=auto` setting is used on a guest with less than 4 GB of physical memory.

To prevent this problem from occurring on KVM guests with less than 4 GB of physical memory, you can reserve memory for a crash kernel by explicitly requesting a reservation size, for example, `crashkernel=128M`. (Bug ID 26933217)

btrfs, ext4 and xfs: Kernel panic when freeze and unfreeze operations are performed in multiple threads

Freeze and unfreeze operations that are performed across multiple threads on any supported file system can cause the system to hang and the kernel to panic. This problem is the result of a race condition that occurs when the unfreeze operation is triggered before it is actually frozen. The resulting unlock operation attempts a write operation on a non-existent lock, resulting in the kernel panic. (Bug ID 25321899)

btrfs Issues

The following are known `btrfs` issues:

- **Send operation causes soft lockup on large deduped file**

  Using `btrfs send` on a large deduped file results in a soft lockup or out-of-memory issue. This problem occurs because the `btrfs send` operation cannot handle a large deduped file containing file extents that are all pointing to one extent, as these types of file structures create tremendous pressure for the `btrfs send` operation.

  To prevent this issue from occurring, do not use `btrfs send` on systems with less than 4 GB of memory. (Bug ID 25306023)

- **Kernel oops when unmounting during a quota rescan or disable**

Operations that trigger a quota rescan or to disable the quota on a mounted file system cause a kernel oops message when attempting to unmount the file system. This can cause the system to hang. (Bug ID 22377928)

- **Kernel oops when removing shared extents using qgroup accounting**

    The removal of shared extents where quota group (qgroup) accounting is used can result in a kernel oops message. This relates to an issue where inaccurate results are obtained during a back reference walk, due to missing records when adding delayed references. (Bug ID 21554517)

- **No warning when balancing file system on RAID**

    The `btrfs filesystem balance` command does not warn that the RAID level can be changed under certain circumstances, and does not provide the choice of cancelling the operation. (Bug ID 16472824)

- **Double count of overwritten space in qgroup show**

    When you overwrite data in a file, starting somewhere in the middle of the file, the overwritten space is counted twice in the space usage numbers that `btrfs qgroup show` displays. Using the `btrfs quota rescan` does not help fix this issue either. (Bug ID 16609467)

- **Location of `btrfs-progs` and `btrfs-progs-devel` packages**

    The `btrfs-progs` and `btrfs-progs-devel` packages for use with UEK R4 are made available in the `ol6_x86_64_UEKR4` and `ol7_x86_64_UEKR4` ULN channels and the `ol6_UEKR4` and `ol7_UEKR4` channels on the Oracle Linux yum server. In UEK R3, these packages were made available in the `ol6_x86_64_latest` and `ol7_x86_64_latest` ULN channels and the `ol6_latest` and `ol7_latest` channels on the Oracle Linux yum server.

ext4 Issues

The following are known `ext4` issues:

- **System hangs on unmount after an append to a file with negative `i_size`**

    While it is invalid for a file system to load an inode with a negative `i_size`, it is possible to create a file like this and append to it. However, doing so causes an integer overflow in the routine's underlying writeback, resulting in the kernel locking up. (Bug ID 25565527)

- **Hang occurs during dynamic expansion of inode size**

    A hang occurs with the `ext4` file system during the dynamic expansion of inode size when using the inode's `i_extra_size` field. (Bug ID 25718971)

xfs Issues

The following are known `xfs` issues:

- **Invalid corrupted file system error resulting from a problem with log recovery on v5 superblocks**

    A problem with log recovery on v5 superblocks that causes the metadata LSN not to update for buffers that it writes out, can result in a corruption error similar to the following:

```
[1044224.901444] XFS (sdc1): Metadata corruption detected at
xfs_dir3_block_write_verify+0xfd/0x110 [xfs], block 0x1004e90
[1044224.901446] XFS (sdc1): Unmount and run xfs_repair
...
```

```
[1044224.901460] XFS (sdc1): xfs_do_force_shutdown(0x8) called from line 1249
of file fs/xfs/xfs_buf.c.  Return address = 0xffffffffa07a8910
[1044224.901462] XFS (sdc1): Corruption of in-memory data detected.  Shutting
down filesystem
[1044224.901463] XFS (sdc1): Please umount the filesystem and rectify the
problem(s)
[1044224.904207] XFS (sdc1): log mount/recovery failed: error -117
[1044224.904456] XFS (sdc1): log mount failed"
```

This problem is encountered because the log attempts to replay a buffer update that is no longer valid due to subsequent replayed updates. The result is a corruption error, when in fact, the file system is fine. (Bug ID 25380003)

- **System hangs on unmount after a buffered append to a file with negative `i_size`**

  While it is invalid for a file system to load an inode with a negative `i_size`, it is possible to create a file like this, and in the case where a buffer appends to it, an integer overflow in the routine's underlying writeback results in the kernel locking up. A direct append does not cause this behavior. (Bug ID 25565490)

- **System hangs during `xfs_fsr` on two-extent files with speculative preallocation**

  During an `xfs_fsr` process on extents that are generated by speculative preallocation, the code that determines whether all of the extents fit inline miscalculates because the `di_nextents` call that is used does not account for these extents. This results in corruption of the in-memory inode, and ultimately the code attempts to move memory structures using incorrectly calculated ranges. This causes a kernel panic. (Bug ID 25333211)

- **XFS quotas are disabled after a read-only remount on Oracle Linux 6**

  Quotas are disabled on XFS if the file system is remounted with read-only permissions on Oracle Linux 6. (Bug ID 22908906)

- **Overlay file system is unable to mount on XFS where there is no `d_type` support**

  Overlay file systems rely on a feature known as `d_type` support. This feature is a field within a data structure that provides some metadata about files in a directory entry within the base file system. Overlay file systems use this field to track many file operations such as file ownership changes and whiteouts. `d_type` support can be enabled in XFS when the file system is created, by using the `-n ftype=1` option. When `d_type` support is not enabled, an overlay file system might become corrupt and behave in unexpected ways. For this reason, this update release of UEK R4 prevents the mounting of an overlay file system on an XFS base, where `d_type` support is not enabled.

  The `root` partition on Oracle Linux is automatically formatted with `-n ftype=0`, where XFS is selected as the file system. Thus, for backward compatibility reasons, if you have overlay file systems in place already and these are not hosted on alternate storage, you must migrate them to a file system that is formatted with `d_type` support enabled.

  To check that the XFS file system is formatted correctly:

  ```
  # xfs_info /dev/sdb1 |grep ftype
  ```

  Replace */dev/sdb1* with the path to the correct storage device. If the information returned by this command includes `ftype=0`, you must migrate the overlay data held in this directory to storage that is formatted correctly.

  To correctly format a new block device with the XFS file system with support for overlay file systems, do:

  ```
  # mkfs -t xfs -n ftype=1 /dev/sdb1
  ```

Replace */dev/sdb1* with the path to the correct storage device. It is essential that you use the `-n ftype=1` option when you create the file system.

If you do not have additional block storage available, it is possible to create an XFS file system image and loopback that can be mounted. For example, to create a 5 GB image file in the `root` directory, you could use the following command:

```
# mkfs.xfs -d file=1,name=/OverlayStorage,size=5g -n ftype=1
```

To temporarily mount this file, you can enter:

```
# mount -o loop -t xfs /OverlayStorage /mnt
```

Adding an entry in `/etc/fstab` to make a permanent mount for this storage, might look similar to the following:

```
/OverlayStorage    /mnt       xfs     loop          0 0
```

This configuration can help as a temporary solution to solve upgrade issues. However, using a loopback mounted file system image as a form of permanent storage is not recommended for production environments. (Bug ID 26165630)

DIF/DIX is not supported for ext file systems

The Data Integrity Field (DIF) and Data Integrity Extension (DIX) features that have been added to the SCSI standard are dependent on a file system that is capable of correctly handling attempts by the memory management system to change data in the buffer while it is queued for a write.

The ext2, ext3 and ext4 file system drivers do not prevent pages from being modified during I/O which can cause checksum failures and a "Logical block guard check failed" error. Other file systems such as XFS are supported. (Bug ID 24361968)

Console appears to hang when booting

When booting Oracle Linux 6 on hardware with an ASPEED graphics controller, the console might appear to hang during the boot process after starting `udev`. However, the system does boot properly and is accessible. The workaround is to add `nomodeset` as a kernel boot parameter in `/etc/grub.conf`. (Bug ID 22389972)

Docker Issues

The following are known Docker issues:

- **Running `yum install` within a container on an overlayfs file system can fail with the following error:**

  ```
  Rpmdb checksum is invalid: dCDPT(pkg checksums): package_name
  ```

  This error can break Dockerfile builds but is expected behavior from the kernel and is a known issue upstream (see https://github.com/docker/docker/issues/10180.)

  The workaround is to run `touch /var/lib/rpm/*` before installing the package.

  Note that this issue is fixed in any Oracle Linux images available on the Docker Hub or Oracle Container Registry, but the issue could still be encountered when running any container based on a third-party image. (Bug ID 21804564)

- **Docker can fail where it uses the `overlay2` storage driver on XFS-formatted storage**

  A kernel patch has been applied to prevent overlay mounts on XFS if the `ftype` is not set to 1. This fix resolves an issue where XFS did not properly support the whiteout features of an overlay filesystem if `d_type` support was not enabled. If the Docker Engine is already using XFS-formatted storage with the `overlay2` storage driver, an upgrade of the kernel can cause Docker to fail if the underlying XFS file system is not created with the `-n ftype=1` option enabled. The root partition on Oracle Linux 7 is automatically formatted with `-n ftype=0` where XFS is selected as the file system. Therefore, if you intend to use the `overlay2` storage driver in this environment, you must format a separate device for this purpose. (Bug ID 25995797)

DTrace Issues

The following are known DTrace issues:

- Argument declarations with USDT probe definitions cannot be declared with derived types such as `enum`, `struct`, or `union`.

- The following compiler warning can be ignored for USDT probe definition arguments of type `string` (which is a D type but not a C type):

  provider_def.h:*line*#: warning: parameter names (without types) in function
  declaration

-

- Multi-threaded processes under `ustack()`, `usym()`, `uaddr()` and `umod()`, which perform `dlopen()` in threads other than the first thread might not have accurate symbol resolution for symbols introduced by `dlopen()`. (Bug ID 20045149)

Error, some other host already uses address *xxx.xxx.xxx.xxx*

The following error message might be triggered in certain instances:

Error, some other host already uses address   *xxx.xxx.xxx.xxx*

The following are the two instances in which this error message might be triggered:

- When active-bonding is enabled, and you run the `ifup` *ib-interface* command.

- When you run the `service rdma start` command.

You can ignore this message, as in both cases, the InfiniBand interface is brought up successfully. (Bug IDs 21052903, 26639723)

Increased `dom0` memory requirement when using Mellanox HCAs on Oracle VM Server

Oracle VM Servers running UEKR4u2 and upward in `dom0` require at least 400MB more memory to use the Mellanox® drivers. This memory requirement is a result of the default size of the SRQ count being increased from 64K to 256K in later versions of the kernel and the `scale_profile` option is now enabled by default in the `mlx_core` module.

In the case where out-of-memory errors are observed in `dom0`, the maximum `dom0` memory size should be increased. Alternative workarounds might involve manually setting the module parameters for the `mlx4_core` driver. To set these parameters, edit `/etc/modprobe.d/`

`mlx4_core.conf` and set `scale_profile` to `0`. Alternately, set `log_num_srq` to `16`. The preferred resolution to this issue is to increase the memory allocated to `dom0` on an Oracle VM Server. (Bug ID 23581534)

LXC Issues

The following are known LXC issues:

- **The `lxc-net` service does not always start immediately after installation on Oracle Linux 6**

  The `lxc-net` service does not always start immediately after installation on Oracle Linux 6, even though this action is specified as part of the RPM post-installation script. This can prevent the `lxcbr0` interface from coming up. If this interface is not up after installation, you can manually start it by running `service lxc-net start`. (Bug ID 23177405)

- **LXC read-only `ip_local_port_range` parameter**

  With `lxc-1.1` or later and UEK R4, `ip_local_port_range` is a read-writable parameter under `/proc/sys/net/ipv4` in an Oracle Linux container rather than being read-only. (Bug ID 21880467)

NVMe devices not found under the `/dev` directory after PCI rescan

After removing the PCI bus of NVM Express (NVMe) adapter card devices and running a rescan of the PCI bus, no NVMe adapter card devices are found under the `/dev` directory.

The workaround for this issue is to also remove the PCI slot that the NVMe adapter card device is plugged into before running a rescan of the PCI bus. (Bug ID 26610285)

OFED iSER target login fails from an initiator on Oracle Linux 6

An Oracle Linux 6 system with the `oracle-ofed-release` packages installed and an iSER (iSCSI Extensions for RDMA) target configured, fails to login to the iSER target as an initiator. On the Oracle Linux 6 initiator machine, the following behavior is typical:

```
# iscsiadm -m node -T iqn.iser-target.t1 -p 10.196.100.134 --login
Logging in to [iface: default, target: iqn.iser-target.t1, portal:
10.196.100.134,3260] (multiple)
iscsiadm: Could not login to [iface: default, target: iqn.iser-target.t1,
portal: 10.196.100.134,3260].
iscsiadm: initiator reported error (8 - connection timed out)
iscsiadm: Could not log into all portals
```

This is expected behavior resulting from an errata fix for CVE-2016-4564, to protect against a write from an invalid context.

(Bug ID 23615903)

Open File Description (OFD) locks are not supported on NFSv4 mounts

NFS is not designed to handle OFD locking. (Bug ID 22948696).

Oracle VM Server MSI-X interrupt allocation failure for 16 GB QLogic FC HBA

The Intel `ixgbe`/`ixgbevf` and QLogic `qla2xxx` drivers compete for MSI-X resources when using a 16 GB QLogic Fibre Channel HBA on systems that are running Oracle VM Server 3.4. As a result, if both drivers are used in a system, and an attempt is made to create the maximum number of Virtual Function (VF) devices that are allowed for the `ixgbe`/`ixgbevf` driver, an interrupt allocation failure occurs during the creation of the last VF device.

This issue is fully resolved by using the latest Oracle-supported firmware for the QLogic 16GB card (FC Firmware v 8.07.71 or later). (Bug IDs 25952728, 26916827)

Possible kernel crash during manual unloading of QLogic FC HBA driver module

A kernel crash might occur while manually unloading the QLogic Fibre Channel (FC) Host Bus Adapter (HBA) driver module for the Oracle 7101674 16 GB HBA model on an Oracle Sun Server X4-2 that is running UEK R4 update 6. (Bug ID 27248515)

RDMA service is not set to start at boot time

Because the `ibacm` service starts at boot time, but the `rdma` service does not start, an error indicating the `ibacm` service failed to start is displayed when the system boots. This error is also logged in `/var/log/ibacm.log` immediately after the system boots.

The workaround for this issue is to manually start the `ibacm` service after every boot by running the `service ibacm start` command.

(Bug IDs 26883485 and 27043535)

SDP performance degradation

The Sockets Direct Protocol (SDP), which was designed to provide an RDMA alternative to TCP over InfiniBand networks, is known to suffer from performance degradation on more recent kernels such as UEK R4u2 and later. There is no active development on this protocol.

Although the library for this protocol is still available for this kernel, support is limited. You should consider using TCP on top of IP over InfiniBand as a more stable alternative. (Bug ID 22354885)

Shared Receive Queue (SRQ) is an experimental feature for RDS and is disabled by default

The SRQ function that optimizes resource usage within the `rds_rdma` module is experimental and is disabled by default. A warning message is displayed when you enable this feature by setting the `rds_ib_srq_enabled` flag. (Bug ID 23523586).

Unloading or removing the `rds_rdma` module is unsupported

Once the `rds_rdma` module has been loaded, you cannot remove the module using either `rmmod` or `modprobe -r`. Unloading of the `rds_rdma` module is unsupported and can trigger a kernel panic. Do not set the `module_unload_allowed` flag for this module. (Bug ID 23580850).

4

Installation and Availability

> **⚠ Important:**
>
> The software described in this documentation is either in Extended Support or Sustaining Support. See Oracle Open Source Support Policies for more information.
>
> We recommend that you upgrade the software described by this documentation as soon as possible.

You can install the Unbreakable Enterprise Kernel Release 4 on Oracle Linux 6 Update 7 or later, or Oracle Linux 7 Update 1 or later, running either the Red Hat compatible kernel or a previous version of the Unbreakable Enterprise Kernel. If you are still running an older version of Oracle Linux, first update your system to the latest available update release.

The Unbreakable Enterprise Kernel Release 4 is supported on the x86-64 architecture, but not on x86.

Installation Overview

If you have a subscription to Oracle Unbreakable Linux support, you can obtain the packages for Unbreakable Enterprise Kernel Release 4 by registering your system with the Unbreakable Linux Network (ULN) and subscribing it to additional channels. See Subscribing to ULN Channels.

If your system is not registered with ULN, you can obtain most of the packages from the Oracle Linux yum server. See Enabling Access to Oracle Yum Channels.

Having subscribed your system to the appropriate channels on ULN or the Oracle Linux yum server, upgrade your system. See Upgrading Your System.

After upgrading to UEK R4, you can replace any existing OFED packages with the Oracle-supported OFED packages, see Installing the Oracle-Supported OFED Packages.

Subscribing to ULN Channels

The kernel image and user-space packages are available on the following ULN channels for Oracle Linux 6:

- `ol6_x86_64_latest` (latest user-space packages for Oracle Linux 6 other than DTrace, OFED, and DRBD packages)
- `ol6_x86_64_UEKR4` (`kernel-uek*`, `dtrace-modules-*`, and `libdtrace-*`)
- `ol6_x86_64_UEKR4_DTrace_userspace` (`dtrace-utils*`)
- `ol6_x86_64_UEKR4_OFED` (latest OFED tools packages)
- `ol6_x86_64_mysql-ha-utils` (`drbd84-utils`)

The kernel image and user-space packages are available on the following ULN channels for Oracle Linux 7:

- `ol7_x86_64_latest` (all of the latest user-space packages for Oracle Linux 7 other than DTrace, OFED, and DRBD packages)

- `ol7_x86_64_latest_optional` (the latest optional user-space packages for Oracle Linux 7 other than DTrace, OFED, and DRBD packages)

- `ol7_x86_64_UEKR4` (`kernel-uek*`, `dtrace-modules-*`, and `libdtrace-*`)

- `ol7_x86_64_UEKR4_DTrace_userspace` (`dtrace-utils*`)

- `ol7_x86_64_UEKR4_OFED` (latest OFED tools packages)

- `ol7_x86_64_mysql-ha-utils` (`drbd84-utils`)

The following procedure assumes that you have already registered your system with ULN.

To subscribe your system to a channel on ULN:

1. Log in to https://linux.oracle.com with your ULN user name and password.

2. On the Systems tab, click the link named for the system in the list of registered machines.

3. On the System Details page, click **Manage Subscriptions**.

4. On the System Summary page, select each required channel from the list of available channels and click the right arrow to move the channel to the list of subscribed channels.

   For Oracle Linux 6, subscribe the system to the `ol6_x86_64_latest` and `ol6_x86_64_UEKR4` channels. If required, you can also add the channels for the DTrace, OFED, and DRBD packages. You do not need to subscribe the system to the `ol6_x86_64_UEK_latest` or `ol6_x86_64_UEKR3_latest` channels.

   For Oracle Linux 7, subscribe the system to the `ol7_x86_64_latest` and `ol7_x86_64_UEKR4` channels. If required, you can also add the channels for the DTrace, OFED, and DRBD packages. You do not need to subscribe the system to the `ol7_x86_64_UEKR3` channel.

5. Click **Save Subscriptions**.

For information about using ULN, see Oracle Linux: Unbreakable Linux Network User's Guide for Oracle Linux 6 and Oracle Linux 7.

Enabling Access to Oracle Yum Channels

On the Oracle Linux yum server at https://yum.oracle.com/, the kernel image and userspace packages are available on the following channels.

For Oracle Linux 6:

- `ol6_latest` (latest user-space packages for Oracle Linux 6 other than the OFED tool packages)

- `ol6_UEKR4` (`kernel-uek*`, `dtrace-modules-*`, and `libdtrace-*`)

- `ol6_UEKR4_OFED` (latest OFED tools packages)

For Oracle Linux 7:

- `ol7_latest` (latest user-space packages for Oracle Linux 7 other than the OFED tool packages)

- `ol7_UEKR4` (`kernel-uek*`, `dtrace-modules-*`, and `libdtrace-*`)

- `ol7_UEKR4_OFED` (latest OFED tools packages)

> **Note:**
>
> To be able to install UEK R4, enable the appropriate `ol6_UEKR4` or `ol7_UEKR4` channel and disable the `ol6_UEKR3_latest` or `ol7_UEKR3` channel.
>
> The DRBD (Distributed Replicated Block Device) packages are not available on the Oracle Linux yum server.

To enable access to the Oracle Linux 6 repositories on the Oracle Linux yum server, use `yum-config-manager`. For example, to enable access to the `ol6_latest` and `ol6_UEKR4` repositories, run the following:

```
# yum-config-manager --enable ol6_latest,ol6_UEKR4
```

To enable access to the Oracle Linux 7 repositories on the Oracle Linux yum server, use `yum-config-manager`. For example, to enable access to the `ol7_latest` and `ol7_UEKR5` repositories, run the following:

```
# yum-config-manager --enable ol7_latest,ol7_UEKR5
```

> **Note:**
>
> You can only use `yum-config-manager` to enable or disable repositories where you already have a configuration file for the specified repository. Repository configurations are typically stored in `/etc/yum.repos.d`. The repository configurations required to install UEK on Oracle Linux 7 are included in the `oraclelinux-release-el7` package. The repository configurations required to install UEK on Oracle Linux 6 are included in the `oraclelinux-release-el6` package. If you do not have the `yum-config-manager` command available you may need to install the `yum-utils` package.

See https://docs.oracle.com/en/operating-systems/oracle-linux/6/admin/ol_yum.html in Oracle® Linux 6: Administrator's Guide or the *Yum* section in Oracle Linux 7: Administrator's Guide for more information.

Upgrading Your System

To upgrade your system to UEK R4:

1. After enabling access to the appropriate channels, including `ol6_UEKR4` or `ol7_UEKR4`, on the Oracle Linux yum server or `ol6_x86_64_UEKR4` or `ol7_x86_64_UEKR4` on ULN, run the following command:

   ```
   # yum update
   ```

2. After upgrading the system, reboot it, selecting the UEK R4 kernel (version 4.1.12) if this is not the default boot kernel.

   See *Boot and Service Configuration* chapter in Oracle Linux 7: Administrator's Guide for more information on updating the default boot kernel on Oracle Linux 7.

See https://docs.oracle.com/en/operating-systems/oracle-linux/6/admin/ol_bootconf.html for more information on updating the default boot kernel on Oracle Linux 6.

For instructions on how to install the Oracle-supported OFED packages after upgrading to UEK R4, see Installing the Oracle-Supported OFED Packages.

If you are upgrading from Oracle Linux 7 Update 3 to Oracle Linux 7 Update 4 and you already have Oracle-supported OFED packages for UEK R4 installed on your system, follow the upgrade procedures that are described in Oracle® Linux 7: Release Notes for Oracle Linux 7.4.

See https://docs.oracle.com/en/operating-systems/oracle-linux/6/admin/ol_yum.html in Oracle® Linux 6: Administrator's Guide or the *Yum* chapter in in Oracle Linux 7: Administrator's Guide for more information.

The kernel's source code is available via a public git source code repository at https://oss.oracle.com/git/?p=linux-uek.git;a=summary.

Installing the Oracle-Supported OFED Packages

The following procedure describes how to install the OFED packages that are provided by Oracle, including how to remove any existing OFED packages.

> **✎ Note:**
>
> For any additional preparation that is required prior to installing or upgrading OFED packages, refer to the release notes for the Oracle Linux release that you are running.

To install the OFED packages that are provided by Oracle:

1. If your system is registered with ULN, subscribe the system to the `ol6_x86_64_UEKR4_OFED` or `ol7_x86_64_UEKR4_OFED` channel on ULN as appropriate.

   By default, the `ol7_x86_64_UEKR4` and `ol7_x86_64_latest` channels are enabled when you register an Oracle Linux 7 system with ULN; and the `ol6_x86_64_UEKR4` and `ol6_x86_64_latest` channels are enabled when you register an Oracle Linux 6 system with ULN. Check that these channels are still enabled before you begin installing the OFED packages provided by Oracle. If enabled, disable the `ol6_x86_64_optional_latest` or `ol7_x86_64_latest` channel, or you may encounter dependency issues.

   If your system uses the Oracle Linux yum server ensure that your system is up to date and that you have transitioned to use the modular yum repository configuration by installing the `oraclelinux-release-el6` or `oraclelinux-release-el7` package and running the `/usr/bin/ol_yum_configure.sh` script. For example, run the following as root:

   ```
   # yum install oraclelinux-release-el7
   ```

   ```
   # /usr/bin/ol_yum_configure.sh
   ```

   Then use `yum-config-manager` to enable the `ol7_UEKR4_OFED` repository for Oracle Linux 7 or the `ol6_UEKR4_OFED` repository for Oracle Linux 6. By default, `ol7_latest` and `ol7_UEKR4` are already enabled on Oracle Linux 7 and the `ol6_latest` and `ol6_UEKR4` are already enabled on Oracle Linux 6. If these repositories are not enabled on your system, you should enable these as well. For example as root, run:

   ```
   # yum-config-manager --enable ol7_latest ol7_UEKR4 ol7_UEKR4_OFED
   ```

Use `yum-config-manager` to disable the `ol7_optional_latest` or `ol6_optional_latest` repository or you may encounter dependency issues if this repository is enabled. For exampe:

```
# yum-config-manager --disable ol7_optional_latest
```

2. If you are running Oracle Linux 7, stop and disable the `rdma.service` service.

```
# systemctl stop rdma.service
# systemctl disable rdma.service
```

3. Remove any existing OFED packages:

```
# yum remove 'ibacm*'
# yum remove 'ib-bonding*'
# yum remove 'ibutils*'
# yum remove 'infiniband-diags*'
# yum remove 'libibacl*'
# yum remove 'libibcm*'
# yum remove 'libibmad*'
# yum remove 'libibumad*'
# yum remove 'libibverbs*'
# yum remove 'libmlx4*'
# yum remove 'librdmacm*'
# yum remove 'libsdp*'
# yum remove 'mstflint*'
# yum remove 'ofed-docs*'
# yum remove 'ofed-scripts*'
# yum remove 'opensm*'
# yum remove 'perftest*'
# yum remove 'qperf*'
# yum remove 'sdpnetstat*'
# yum remove 'rdma*'
# yum remove 'rds-tools*'
```

4. Clean all `yum` cached files from all enabled repositories:

```
# yum clean all
```

5. Run one of the following commands, based on server type:

   - For a bare metal server, install the OFED packages for UEK R4 as follows:

     ```
     # yum install oracle-ofed-release
     ```

   - For a server that will function as a guest, install the OFED packages for UEK R4 as follows:

     ```
     # yum install oracle-ofed-release-guest
     ```

6. Enable the RDMA service by entering the following command:

   ```
   # chkconfig rdma on
   ```

Each UEK release requires a different set of OFED packages. If you change the kernel on your system to a UEK release earlier than UEK R4, remove the existing UEK R4-based OFED packages before installing the correct packages for the new kernel by running the following command:

```
# yum remove --setopt=clean_requirements_on_remove=1 oracle-ofed-release
```

> ⚠️ **Caution:**
>
> Downgrading UEK versions is not advisable, except for testing purposes.

To update OFED packages that are already installed for UEK R4, run this command:

```
# yum update oracle-ofed-release
```

To update the OFED packages that are already installed on the guest, run this command on the guest:

```
# yum update oracle-ofed-release-guest
```