Oracle Linux 7

Managing Software

F32599-05
August 2022

Oracle Linux 7 Managing Software,

F32599-05

Contents

Preface

Conventions v
Documentation Accessibility v
Access to Oracle Support for Accessibility v
Diversity and Inclusion v

1 Managing Software With Yum

About Yum 1-1
About ULN 1-1

2 Yum Configuration

Yum Configuration Directives 2-1
Configuring Use of a Proxy Server 2-2
Yum Repository Configuration 2-3
Downloading the Oracle Linux Yum Server Repository Files 2-3
Using Yum on Oracle Cloud Infrastructure Systems 2-5
Using Yum Utilities to Manage Configuration 2-5

3 Using Yum from the Command Line

4 Yum Groups

5 Using the Yum Security Plugin

6 Switching CentOS or Scientific Linux Systems to Use the Oracle Linux Yum Server

7 Creating a Local Yum Repository Using an ISO Image

8 Setting up a Local Yum Server Using an ISO Image

9 Using the yum-cron Interface to Automatically Keep Your System Up To Date

10 For More Information About Yum

Preface

Oracle Linux 7: Managing Software describes how to install, upgrade, and manage software on Oracle Linux 7 systems by using the Yum utility.

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/.

For information about the accessibility of the Oracle Help Center, see the Oracle Accessibility Conformance Report at https://www.oracle.com/corporate/accessibility/templates/t2-11535.html.

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 Managing Software With Yum

This chapter provides an overview of Yum, and includes details about Yum configuration and the `yum` that you use to install and upgrade software packages on your Oracle Linux systems.

About Yum

Oracle Linux provides the `yum` utility which you can use to install or upgrade RPM packages. The main benefit of using yum is that it also installs or upgrades any package dependencies. `yum` downloads the packages from repositories such as those that are available on the Oracle Linux yum server, but you can also set up your own repositories on systems that do not have Internet access.

The Oracle Linux yum server is a convenient way to install Oracle Linux and Oracle VM packages, including bug fixes, security fixes and enhancements, rather than installing them from installation media. You can access the server at https://yum.oracle.com/.

You can also subscribe to the Oracle Linux and Oracle VM errata mailing lists to be notified when new packages are released. You can access the mailing lists at https://oss.oracle.com/mailman/listinfo/el-errata and https://oss.oracle.com/mailman/listinfo/oraclevm-errata.

About ULN

The repositories that are available on the Oracle Linux yum server are aligned with the channels that are available on the Unbreakable Linux Network (ULN), with the exception of ULN channels that are limited to Oracle Linux Premier Support customers. These include channels for products such as Ksplice.

On Oracle Cloud Infrastructure, the regional yum servers that are made available via the service gateways are configured to also include additional repositories to provide access to content typically available to Oracle Linux Premier Support customers on ULN.

ULN is tightly integrated with yum. If you have registered your system with ULN, you can use `yum` commands with ULN channels to maintain the software on your system.

For more detailed information about working with ULN, including how to create and use a Yum server that acts as a local mirror of the ULN channels, see Oracle Linux: Unbreakable Linux Network User's Guide for Oracle Linux 6 and Oracle Linux 7.

2

Yum Configuration

The main configuration file for yum is /etc/yum.conf. This chapter describes how to configure directives in the configuration file.

Yum Configuration Directives

The global definitions for yum are located under the [main] section heading of the yum configuration file. The following table lists the important directives.

Directive	Description
cachedir	Directory used to store downloaded packages.
debuglevel	Logging level, from 0 (none) to 10 (all).
exactarch	If set to 1, only update packages for the correct architecture.
exclude	A space separated list of packages to exclude from installs or updates, for example: exclude=VirtualBox-4.? kernel*.
gpgcheck	If set to 1, verify the authenticity of the packages by checking the GPG signatures. You might need to set gpgcheck to 0 if a package is unsigned, but you should be wary that the package could have been maliciously altered.
gpgkey	Pathname of the GPG public key file.
installonly_limit	Maximum number of versions that can be installed of any one package.
keepcache	If set to 0, remove packages after installation.
logfile	Pathname of the yum log file.
obsoletes	If set to 1, replace obsolete packages during upgrades.
plugins	If set to 1, enable plugins that extend the functionality of yum.
proxy	URL of a proxy server including the port number. See Configuring Use of a Proxy Server.
proxy_password	Password for authentication with a proxy server.
proxy_username	User name for authentication with a proxy server.
reposdir	Directories where yum should look for repository files with a .repo extension. The default directory is /etc/yum.repos.d.

See the yum.conf(5) manual page for more information.

The following listing shows an example [main] section from the yum configuration file.

```
[main]
cachedir=/var/cache/yum
keepcache=0
debuglevel=2
logfile=/var/log/yum.log
exactarch=1
obsoletes=1
gpgkey=file://media/RPM-GPG-KEY
gpgcheck=1
pligins=1
installonly_limit=3
```

It is possible to define repositories below the [main] section in /etc/yum.conf or in separate repository configuration files. By default, yum expects any repository configuration files to be located in the /etc/yum.repos.d directory unless you use the reposdir directive to define alternate directories.

Configuring Use of a Proxy Server

If your organization uses a proxy server as an intermediary for Internet access, specify the proxy setting in /etc/yum.conf as shown in the following example.

```
proxy=http://proxysvr.example.com:3128
```

If the proxy server requires authentication, additionally specify the proxy_username, and proxy_password settings.

```
proxy=http://proxysvr.example.com:3128
proxy_username=yumacc
proxy_password=clydenw
```

If you use the yum plugin (yum-rhn-plugin) to access the ULN, specify the enableProxy and httpProxy settings in /etc/sysconfig/rhn/up2date as shown in this example.

```
enableProxy=1
httpProxy=http://proxysvr.example.com:3128
```

If the proxy server requires authentication, additionally specify the enableProxyAuth, proxyUser, and proxyPassword settings.

```
enableProxy=1
httpProxy=http://proxysvr.example.com:3128
enableProxyAuth=1
proxyUser=yumacc
proxyPassword=clydenw
```

Caution:

All yum users require read access to /etc/yum.conf or /etc/sysconfig/rhn/up2date. If these files must be world-readable, do not use a proxy password that is the same as any user's login password, and especially not root's password.

Yum Repository Configuration

The yum configuration file or yum repository configuration files can contain one or more sections that define repositories.

The following table lists the basic directives for a repository.

Directive	Description
baseurl	Location of the repository channel (expressed as a file://, ftp://, http://, or https:// address). This directive must be specified.
enabled	If set to 1, permit yum to use the channel.
name	Descriptive name for the repository channel. This directive must be specified.

Any other directive that appears in this section overrides the corresponding global definition in [main] section of the yum configuration file. See the yum.conf(5) manual page for more information.

The following listing shows an example repository section from a configuration file.

```
[ol6_u2_base]
name=Oracle Linux 6 U2 - $basearch - base
baseurl=https://yum.oracle.com/repo/OracleLinux/OL6/2/base/$basearch
gpgkey=file:///etc/pki/rpm-gpg/RPM-GPG-KEY
gpgcheck=1
enabled=1
```

In this example, the values of gpgkey and gpgcheck override any global setting. yum substitutes the name of the current system's architecture for the variable *$basearch*.

yum automatically searches the /etc/yum.repos.d directory for files with the suffix .repo and appends these to the configuration when it is processing. Use this directory to define repository files for repositories that you want to make available.

Downloading the Oracle Linux Yum Server Repository Files

The Oracle Linux yum server provides a direct mapping of all of the Unbreakable Linux Network (ULN) channels that are available to the public without any specific support agreement. The repository labels used for each repository on the Oracle Linux yum server map directly onto the channel names on ULN. See Oracle Linux: Unbreakable Linux Network User's Guide for Oracle Linux 6 and Oracle Linux 7 at https://docs.oracle.com/en/operating-systems/oracle-linux/uln-user/ol_channels_uln.html for more information about the channel names and common suffixes used for channels and repositories.

Prior to January 2019, Oracle shipped a single yum repository configuration file for each Oracle Linux release. This configuration file is copied into /etc/yum.repos.d/public-yum-ol7.repo at installation, but can also be downloaded from the Oracle Linux yum server directly to obtain updates.

The original configuration file is deprecated in favor of modular repository files that are managed and updated automatically via yum in the form of RPM packages that are more targeted in scope. For example, core repository configuration files required for Oracle Linux 7 are available in the oraclelinux-release-el7 package. This package includes all of the repository configuration required to install base packages for the release, including packages from the ol7_latest, ol7_addons repositories and all of the supported repositories for UEK.

The modular yum repository configuration files released as packages that can be maintained via yum can help to simplify repository management and also ensure that your yum repository definitions are kept up to date automatically, whenever you update your system.

A list of all available RPM files to manage all of the possible yum repository configurations for your release can be obtained by running:

```
sudo yum list *release-el7*
```

To install the yum repository configuration for a particular set of software that you wish to use, use yum to install the corresponding package. For example, to install the yum repository configuration for the Oracle Linux Software Collection Library, run:

```
sudo yum install oracle-softwarecollection-release-el7
```

If your system is still configured to use the original single yum repository configuration file at /etc/yum.repos.d/public-yum-ol7.repo, you should update your system to transition to the current approach to handling yum repository configuration. To do this, ensure that your system is up to date and then run the /usr/bin/ol_yum_configure.sh script:

```
sudo yum update
sudo /usr/bin/ol_yum_configure.sh
```

The /usr/bin/ol_yum_configure.sh script checks the /etc/yum.repos.d/public-yum-ol7.repo file to determine which repositories are already enabled and installs the appropriate corresponding packages before renaming the original configuration file to /etc/yum.repos.d/public-yum-ol7.repo.sav to disable it in favor of the more recent modular repository configuration files.

If, for some reason, you manage to remove all configuration to access the Oracle Linux yum server repositories, you should create a temporary yum repository configuration file at /etc/yum.repos.d/ol7-temp.repo with the following as the minimum required content:

```
[ol7_latest]
name=Oracle Linux $releasever Latest ($basearch)
baseurl=https://yum.oracle.com/repo/OracleLinux/OL7/latest/$basearch/
gpgkey=file:///etc/pki/rpm-gpg/RPM-GPG-KEY-oracle
gpgcheck=1
enabled=1
```

Then reinstall the oraclelinux-release-el7 package to restore the default yum configuration:

```
sudo yum reinstall oraclelinux-release-el7
sudo rm /etc/yum.repos.d/ol7-temp.repo
```

For more information on manually setting up Oracle Linux yum server repository configuration files, see https://yum.oracle.com/getting-started.html.

You can enable or disable repositories in each repository configuration file by setting the value of the enabled directive to 1 or 0 for each repository listed in the file, as required. The preferred method of enabling or disabling repositories under Oracle Linux 7 is to use the `yum-config-manager` command as described in Using Yum Utilities to Manage Configuration.

Using Yum on Oracle Cloud Infrastructure Systems

Compute instances in Oracle Cloud Infrastructure have access to regional yum servers via the service gateway. Regional yum servers on Oracle Cloud Infrastructure differ from the Oracle Linux yum server in that they also mirror content available on restricted ULN channels.

Yum repository configuration in Oracle Linux typically makes use of a yum variable in the baseurl for managing appropriate yum server access. For example, the baseurl to the `_latest` repository for Oracle Linux 7 is:

```
baseurl=https://yum$ociregion.oracle.com/repo/OracleLinux/OL7/latest/$basearch/
```

The `$ociregion` variable can be set by populating content in `/etc/yum/vars/ociregion`. If this file does not exist, or the file is empty, the baseurl is expanded to point to the publicly accessible Oracle Linux yum server. In the case of a typical Oracle Cloud Infrastructure compute instance, the value of variable is set when the instance is created so that the baseurl is expanded to point to the closest regional yum server on the Oracle Cloud Infrastructure service network. For example, if `$ociregion` is set to `-phx`, the baseurl expands to point to the regional yum server located in Phoenix.

By using variables, configuration can remain relatively standard across Oracle Linux deployments but provide access to the additional resources available to Oracle Cloud Infrastructure customers.

For more information on Oracle Cloud Infrastructure and yum repositories, see See Oracle Linux: Unbreakable Linux Network User's Guide for Oracle Linux 6 and Oracle Linux 7.

Using Yum Utilities to Manage Configuration

The `yum-utils` package includes several utilities that can help you to manage configuration and apply updates safely to your existing configuration. Most significant of these is `yum-config-manager`.

To install the `yum-utils` package:

```
sudo yum install -y yum-utils
```

You can use `yum-config-manager` to add repositories either at a specified URL, or within a specified repository file. For example, to add the legacy repository configuration file for Oracle Linux 7 from the Oracle Linux yum server:

```
sudo yum-config-manager --add-repo https://yum.oracle.com/public-yum-ol7.repo
```

> **Note:**
>
> The legacy repository configuration file is unmaintained and deprecated. The information in this file may not be current and newer repositories may not be listed.

You can use the same command to automatically generate a repository configuration file for a valid yum repository, by pointing to the URL where the repository is hosted. For example, to create a new configuration file in `/etc/repos.d` for the Unbreakable Enterprise Kernel Release 5 repository. run:

```
sudo yum-config-manager --add-repo https://yum.oracle.com/repo/OracleLinux/OL7/
UEKR5/x86_64
```

To enable a repository using `yum-config-manager`, use the `--enable` option. For example, to enable the `ol7_addons` repository, run:

```
# yum-config-manager --enable ol7_addons
```

You can use the `--disable` option in a similar way to disable a repository.

The `yum-config-manager` tool can also be used to set other configuration options using the `--setopt` and `--save` options. See the `yum-config-manager(1)` manual page for more information.

For a list of the tools included in the yum-utils package and a description of what these tools can do, see the `yum-utils(1)` manual page for more information.

3 Using Yum from the Command Line

The following table shows some examples of common tasks that you can perform using `yum`.

Command	Description
`yum repolist`	Lists all enabled repositories.
`yum list`	Lists all packages that are available in all enabled repositories and all packages that are installed on your system.
`yum list installed`	Lists all packages that are installed on your system.
`yum list available`	Lists all packages that are available to be installed in all enabled repositories.
`yum search` *string*	Searches the package descriptions for the specified string.
`yum provides` *feature*	Finds the name of the package to which the specified file or feature belongs. For example: `yum provides /etc/sysconfig/atd`
`yum info` *package*	Displays detailed information about a package. For example: `yum info bind`
`yum install` *package*	Installs the specified package, including packages on which it depends. For example: `yum install ocfs2-tools`
`yum check-update`	Checks whether updates exist for packages that are already installed on your system.
`yum update` *package*	Updates the specified package, including packages on which it depends. For example: `yum upgrade nfs-utils`
`yum update`	Updates all packages, including packages on which they depend.
`yum remove` *package*	Removes the specified package. For example: `yum erase nfs-utils`
`yum erase` *package*	Removes the specified package. This command has the same effect as the `yum remove` command.
`yum update`	Updates all packages, including packages on which they depend.
`yum clean all`	Removes all cached package downloads and cached headers that contain information about remote packages. Running this command can help to clear problems that can result from unfinished transactions or out-of-date headers.

Command	Description
`yum help`	Displays help about `yum` usage.
`yum help` *command*	Displays help about the specified `yum` command. For example: `yum help upgrade`
`yum shell`	Runs the `yum` interactive shell.

See the `yum(8)` manual page for more information.

To list the files in a package, use the `repoquery` utility, which is included in the `yum-utils` package. For example, the following command lists the files that the `btrfs-progs` package provides.

```
sudo repoquery -l btrfs-progs

/sbin/btrfs
/sbin/btrfs-convert
/sbin/btrfs-debug-tree
.
.
.
```

Note:

`yum` makes no distinction between installing and upgrading a kernel package. `yum` always installs a new kernel regardless of whether you specify `update` or `install`.

4
Yum Groups

A set of packages can themselves be organized as a *yum group*. Examples include the groups for Eclipse, fonts, and system administration tools. The following table shows the `yum` commands that you can use to manage these groups.

Command	Description
`yum grouplist`	Lists installed groups and groups that are available for installation.
`yum groupinfo` *groupname*	Displays detailed information about a group.
`yum groupinstall` *groupname*	Installs all the packages in a group.
`yum groupupdate` *groupname*	Updates all the packages in a group.
`yum groupremove` *groupname*	Removes all the packages in a group.

5

Using the Yum Security Plugin

The security plugin is integrated with `yum` in Oracle Linux 7 and allows you to obtain a list of all of the errata that are available for your system, including security updates. You can also use Oracle Enterprise Manager 12c Cloud Control or management tools such as Katello, Pulp, Red Hat Satellite, Spacewalk, and SUSE Manager to extract and display information about errata.

To list the errata that are available for your system, enter:

```
sudo yum updateinfo list

ELBA-2018-2000 bugfix         NetworkManager-1:1.10.2-16.el7_5.x86_64
ELBA-2018-2000 bugfix         NetworkManager-adsl-1:1.10.2-16.el7_5.x86_64
ELBA-2018-2000 bugfix         NetworkManager-config-server-1:1.10.2-16.el7_5.noarch
ELBA-2018-2000 bugfix         NetworkManager-glib-1:1.10.2-16.el7_5.x86_64
ELBA-2018-2000 bugfix         NetworkManager-libnm-1:1.10.2-16.el7_5.x86_64
ELBA-2018-2000 bugfix         NetworkManager-ppp-1:1.10.2-16.el7_5.x86_64
ELBA-2018-2000 bugfix         NetworkManager-team-1:1.10.2-16.el7_5.x86_64
ELBA-2018-2000 bugfix         NetworkManager-tui-1:1.10.2-16.el7_5.x86_64
ELBA-2018-2000 bugfix         NetworkManager-wifi-1:1.10.2-16.el7_5.x86_64
ELBA-2018-1994 bugfix         binutils-2.27-28.base.el7_5.1.x86_64
ELBA-2018-1980 bugfix         control-center-1:3.26.2-9.el7_5.x86_64
ELBA-2018-1980 bugfix         control-center-filesystem-1:3.26.2-9.el7_5.x86_64
ELBA-2018-4142 bugfix         dracut-033-535.0.2.el7.x86_64
ELBA-2018-4142 bugfix         dracut-config-rescue-033-535.0.2.el7.x86_64
ELBA-2018-4142 bugfix         dracut-network-033-535.0.2.el7.x86_64
ELEA-2018-0838 enhancement    filesystem-3.2-25.el7.x86_64
ELSA-2018-2113 Critical/Sec.  firefox-60.1.0-4.0.1.el7_5.x86_64
...
```

The output from the command sorts the available errata in order of their IDs, and it also specifies whether each erratum is a security patch (*severity* `/Sec.`), a bug fix (`bugfix`), or a feature enhancement (`enhancement`). Security patches are listed by their severity: `Important`, `Moderate`, or `Low`.

You can use the `--sec-severity` option to filter the security errata by severity, for example:

```
sudo yum updateinfo list --sec-severity=Moderate

ELSA-2018-1852 Moderate/Sec. kernel-3.10.0-862.3.3.el7.x86_64
ELSA-2018-1852 Moderate/Sec. kernel-devel-3.10.0-862.3.3.el7.x86_64
ELSA-2018-1852 Moderate/Sec. kernel-headers-3.10.0-862.3.3.el7.x86_64
ELSA-2018-1852 Moderate/Sec. kernel-tools-3.10.0-862.3.3.el7.x86_64
ELSA-2018-1852 Moderate/Sec. kernel-tools-libs-3.10.0-862.3.3.el7.x86_64
ELSA-2018-2123 Moderate/Sec. python-2.7.5-69.0.1.el7_5.x86_64
ELSA-2018-2123 Moderate/Sec. python-libs-2.7.5-69.0.1.el7_5.x86_64
ELSA-2018-1852 Moderate/Sec. python-perf-3.10.0-862.3.3.el7.x86_64
...
```

To list the security errata by their Common Vulnerabilities and Exposures (CVE) IDs instead of their errata IDs, specify the keyword `cves` as an argument:

```
sudo yum updateinfo list cves

 CVE-2017-7762    Critical/Sec.  firefox-60.1.0-4.0.1.el7_5.x86_64
 CVE-2018-12359   Critical/Sec.  firefox-60.1.0-4.0.1.el7_5.x86_64
 CVE-2018-12363   Critical/Sec.  firefox-60.1.0-4.0.1.el7_5.x86_64
 CVE-2018-12364   Critical/Sec.  firefox-60.1.0-4.0.1.el7_5.x86_64
 CVE-2018-12366   Critical/Sec.  firefox-60.1.0-4.0.1.el7_5.x86_64
 CVE-2018-5156    Critical/Sec.  firefox-60.1.0-4.0.1.el7_5.x86_64
 CVE-2018-5188    Critical/Sec.  firefox-60.1.0-4.0.1.el7_5.x86_64
 CVE-2018-6126    Critical/Sec.  firefox-60.1.0-4.0.1.el7_5.x86_64
 CVE-2018-12360   Critical/Sec.  firefox-60.1.0-4.0.1.el7_5.x86_64
 CVE-2018-12362   Critical/Sec.  firefox-60.1.0-4.0.1.el7_5.x86_64
 CVE-2018-12365   Critical/Sec.  firefox-60.1.0-4.0.1.el7_5.x86_64
 CVE-2018-12020   Important/Sec. gnupg2-2.0.22-5.el7_5.x86_64
 CVE-2018-3665    Moderate/Sec.  kernel-3.10.0-862.3.3.el7.x86_64
 CVE-2018-3639    Important/Sec. kernel-3.10.0-862.6.3.el7.x86_64
 CVE-2017-11600   Important/Sec. kernel-3.10.0-862.6.3.el7.x86_64
...
```

Similarly, the keywords `bugfix`, `enhancement`, and `security` filter the list for all bug fixes, enhancements, and security errata.

You can use the `--cve` option to display the errata that correspond to a specified CVE, for example:

```
sudo yum updateinfo list --cve CVE-2018-3665

ELSA-2018-1852 Moderate/Sec.  kernel-3.10.0-862.3.3.el7.x86_64
ELSA-2018-1852 Moderate/Sec.  kernel-devel-3.10.0-862.3.3.el7.x86_64
ELSA-2018-1852 Moderate/Sec.  kernel-headers-3.10.0-862.3.3.el7.x86_64
ELSA-2018-1852 Moderate/Sec.  kernel-tools-3.10.0-862.3.3.el7.x86_64
ELSA-2018-1852 Moderate/Sec.  kernel-tools-libs-3.10.0-862.3.3.el7.x86_64
ELSA-2018-4144 Important/Sec. kernel-uek-firmware-4.1.12-124.16.4.el7uek.noarch
ELSA-2018-1852 Moderate/Sec.  python-perf-3.10.0-862.3.3.el7.x86_64
updateinfo list done
```

To display more information, specify `info` instead of `list`, for example:

```
sudo yum updateinfo info --cve CVE-2018-3665

===============================================================================
   kernel security update
===============================================================================
  Update ID : ELSA-2018-1852
    Release : Oracle Linux 7
       Type : security
     Status : final
     Issued : 2018-06-14
       CVEs : CVE-2018-3665
Description : [3.10.0-862.3.3.OL7]
            : - Oracle Linux certificates (Alexey Petrenko)
            : - Oracle Linux RHCK Module Signing Key was
            :   compiled into kernel
            :   (olkmod_signing_key.x509)(alexey.petrenko@oracle.com)
            : - Update x509.genkey [bug 24817676]
            :
            : [3.10.0-862.3.3]
            : - [x86] always enable eager FPU by default on
            :   non-AMD processors (Paolo Bonzini) [1589051
            :   1589048] {CVE-2018-3665}
   Severity : Moderate
```

```
===============================================================================
  Unbreakable Enterprise kernel security update
===============================================================================
  Update ID : ELSA-2018-4144
    Release : Oracle Linux 7
       Type : security
     Status : final
     Issued : 2018-06-15
       CVEs : CVE-2018-3665
Description : [4.1.12-124.16.4]
            : - x86/fpu: Make eager FPU default (Mihai Carabas)
            :   [Orabug: 28135099]  {CVE-2018-3665}
   Severity : Important
updateinfo info done
```

To update all packages for which security-related errata are available to the latest versions of the packages, even if those packages include bug fixes or new features but not security errata, enter:

```
sudo yum --security update
```

To update all packages to the latest versions that contain security errata, ignoring any newer packages that do not contain security errata, enter:

```
sudo yum --security update-minimal
```

To update all kernel packages to the latest versions that contain security errata, enter:

```
sudo yum --security update-minimal kernel*
```

You can also update only those packages that correspond to a CVE or erratum, for example:

```
sudo yum update --cve CVE-2018-3665
sudo yum update --advisory ELSA-2018-4144
```

> **Note:**
>
> Some updates might require you to reboot the system. By default, the boot manager will automatically enable the most recent kernel version.

For more information, see the `yum-security(8)` manual page.

6

Switching CentOS or Scientific Linux Systems to Use the Oracle Linux Yum Server

You can use the `centos2ol.sh` script to convert CentOS 5 and 6 or Scientific Linux 5 and 6 systems to Oracle Linux. The script configures `yum` to use the Oracle Linux yum server and installs a few additional packages that are required. There is no need to reboot the system.

To perform the switch to Oracle Linux, run the following commands as `root`:

```
curl -O https://linux.oracle.com/switch/centos2ol.sh
sudo sh centos2ol.sh
```

For more information, see https://linux.oracle.com/switch/centos/.

7 Creating a Local Yum Repository Using an ISO Image

> **Note:**
>
> The system must have sufficient storage space to host a full Oracle Linux Media Pack DVD image (approximately 4.1 GB for Oracle Linux 7.5).

To create a local yum repository (for example, if a system does not have Internet access):

1. On a system with Internet access, download a full Oracle Linux DVD image from the Oracle Software Delivery Cloud at https://edelivery.oracle.com/linux onto removable storage (such as a USB memory stick). For example, `V975367-01.iso` contains the Oracle Linux 7.5 Media Pack for x86 (64 bit).

   > **Note:**
   >
   > You can verify that the ISO was copied correctly by comparing its checksum with the digest value that is listed on `edelivery.oracle.com`, for example:
   >
   > ```
   > sha1sum V975367-01.iso
   >
   > 4B08559B6176F701563C2E49553C89491C0F8F4E V975367-01.iso
   > ```

2. Transfer the removable storage to the system on which you want to create a local yum repository, and copy the DVD image to a directory in a local file system.

   ```
   cp /media/USB_stick/V975367-01.iso /ISOs
   ```

3. Create a suitable mount point, for example `/var/OSimage/OL7.5_x86_64`, and mount the DVD image on it.

   ```
   mkdir -p /var/OSimage/OL7.5_x86_64
   sudo mount -o loop,ro /ISOs/V975367-01.iso /var/OSimage/
   OL7.5_x86_64
   ```

   > **Note:**
   >
   > Include the read-only mount option (`ro`) to avoid changing the contents of the ISO by mistake.

4. Create an entry in `/etc/fstab` so that the system always mounts the DVD image after a reboot.

   ```
   /ISOs/V975367-01.iso /var/OSimage/OL7.5_x86_64 iso9660 loop,ro 0 0
   ```

5. Disable all existing yum repositories.

   In the /etc/yum.repos.d directory, edit any existing repository files and disable all entries by setting enabled=0. If you have the yum-utils package installed, as described in Using Yum Utilities to Manage Configuration, you can disable all repositories by running:

   ```
   sudo yum-config-manager --disable \*
   ```

6. Create the following entries in a new repository file (for example, /etc/yum.repos.d/OL75.repo).

   ```
   [OL75]
   name=Oracle Linux 7.5 x86_64
   baseurl=file:///var/OSimage/OL7.5_x86_64
   gpgkey=file:///etc/pki/rpm-gpg/RPM-GPG-KEY
   gpgcheck=1
   enabled=1
   ```

   Note that the correct GPG key file must exist at the path specified for the gpgkey parameter. You can download the GPG keys used to sign all of the Oracle Linux release packages from the Oracle Linux yum server. See https://yum.oracle.com/faq.html#a10 for more information.

7. Clean up the yum cache.

   ```
   sudo yum clean all
   ```

8. Test that you can use yum to access the repository.

   ```
   sudo yum repolist

   Loaded plugins: refresh-packagekit, security
   ...
   repo id                          repo name
   status
   OL75                             Oracle Linux 7.5 x86_64
   5,070
   repolist: 5,070
   ```

8 Setting up a Local Yum Server Using an ISO Image

To set up a local yum server (for example, if you have a network of systems that do not have Internet access):

1. Choose one of the systems to be the yum server, and create a local yum repository on it as described in Creating a Local Yum Repository Using an ISO Image.
2. Install the Apache HTTP server from the local yum repository.

   ```
   sudo yum install httpd
   ```

3. If SELinux is enabled in enforcing mode on your system:
   a. Use the `semanage` command to define the default file type of the repository root directory hierarchy as `httpd_sys_content_t`:

      ```
      sudo /usr/sbin/semanage fcontext -a -t httpd_sys_content_t "/var/OSimage(/.*)?"
      ```

   b. Use the `restorecon` command to apply the file type to the entire repository.

      ```
      sudo /sbin/restorecon -R -v /var/OSimage
      ```

   > **Note:**
   >
   > The `semanage` and `restorecon` commands are provided by the `policycoreutils-python` and `policycoreutils` packages.

4. Create a symbolic link in `/var/www/html` that points to the repository:

   ```
   sudo ln -s /var/OSimage /var/www/html/OSimage
   ```

5. Edit the HTTP server configuration file, `/etc/httpd/conf/httpd.conf`, as follows:
   a. Specify the resolvable domain name of the server in the argument to `ServerName`.

      ```
      ServerName server_addr:80
      ```

      If the server does not have a resolvable domain name, enter its IP address instead.

   b. Verify that the setting of the `Options` directive in the `<Directory "/var/www/html">` section specifies `Indexes` and `FollowSymLinks` to allow you to browse the directory hierarchy, for example:

      ```
      Options Indexes FollowSymLinks
      ```

   c. Save your changes to the file.
6. Start the Apache HTTP server, and configure it to start after a reboot.

   ```
   sudo systemctl start httpd
   sudo systemctl enable httpd
   ```

7. If you have enabled a firewall on your system, configure it to allow incoming HTTP connection requests on TCP port 80, for example:

   ```
   sudo firewall-cmd --zone=zone --add-port=80/tcp
   sudo firewall-cmd --permanent --zone=zone --add-port=80/tcp
   ```

8. Disable all existing yum repositories on the server and each client system.

   In the `/etc/yum.repos.d` directory, edit any existing repository files and disable all entries by setting `enabled=0`. If you have the `yum-utils` package installed, as described in Using Yum Utilities to Manage Configuration, you can disable all repositories by running:

   ```
   sudo yum-config-manager --disable \*
   ```

9. Edit the repository file on the server (for example, `/etc/yum.repos.d/OL75.repo`):

   ```
   [OL75]
   name=Oracle Linux 7.5 x86_64
   baseurl=http://server_addr/OSimage/OL7.5_x86_64
   gpgkey=file:///etc/pki/rpm-gpg/RPM-GPG-KEY
   gpgcheck=1
   enabled=1
   ```

   Replace *server_addr* with the IP address or resolvable host name of the local yum server.

10. On each client, copy the repository file from the server to the `/etc/yum.repos.d` directory.

11. On the server and each client, test that you can use `yum` to access the repository.

    ```
    sudo yum repolist

    Loaded plugins: refresh-packagekit, security
    ...
    repo id                          repo name
    status
    OL75                             Oracle Linux 7.5 x86_64
    5,070
    repolist: 5,070
    ```

9

Using the yum-cron Interface to Automatically Keep Your System Up To Date

As an alternative to manually running the `yum update` command to keep your system up to date with the latest security patches and bug fixes, you can use the `yum-cron` interface, which is provided as an additional package in Oracle Linux. The interface is used to provide automatic notifications of updates and also to download updates, which can then be installed automatically by using `crontab`.

To get started, install the `yum-cron` package from the `ol7_latest` repository. Then, enable and start the service so that it checks for updates daily:

```
sudo yum install yum-cron
sudo systemctl enable --now yum-cron.service
```

To customize the behavior of the `yum-cron` interface, edit `/etc/yum/yum-cron.conf`. You can create alternate configuration files that can be invoked on different schedules, depending on your requirements. Oracle Linux also includes an alternate configuration file at `/etc/yum/yum-cron-hourly.conf`. By default, when `yum-cron` runs, it loads its configuration from `/etc/yum/yum-cron.conf` unless an alternate configuration is offered as an argument when it is run.

Oracle Linux creates two crontab entries, by default, when `yum-cron` is installed. These are located in `/etc/cron.daily/0yum-daily.cron` and `/etc/cron.hourly/0yum-hourly.cron`. The daily crontab entry runs `yum-cron` using the default configuration in `/etc/yum/yum-cron.conf`. The hourly crontab entry invokes `yum-cron` to run with the alternate configuration at `/etc/yum/yum-cron-hourly.conf`. By creating multiple configuration entries and crontab entries, you can control the frequency that different yum update operations are performed. For instance, you may wish to update yum repository metadata on an hourly schedule, while applying critical security updates on a daily schedule. You could also create a configuration to apply all remaining updates on a weekly or monthly schedule.

Configuration files allow you to configure different parameters to control the kinds of updates that are managed by `yum-cron`; whether updates are applied or only downloaded; and how to provide notification of updates. The default configuration files are commented to provide explanations for each setting. The following settings are notable:

- **`update_cmd`**

  This option controls the types of updates that are either downloaded or applied by `yum-cron`. The following options are available:

  default
  Fetch all updates.

  security
  Fetch only security updates.

security-severity:Critical
Fetch only security updates marked with "Critical" severity.

minimal
Fetch package errata version updates only.

minimal-security
Fetch package errata versions marked as security updates only.

minimal-security-severity:Critical
Fetch package errata versions marked as security updates with "Critical" severity.

For mission-critical systems that require a scheduled downtime to apply updates, you can direct `yum-cron` to only install minimal security updates by specifying the `Critical` severity, for example:

```
update_cmd = minimal-security-severity:Critical
```

Consider also using Oracle Ksplice if you wish to minimize scheduled downtime for kernel patches and updates. See About Ksplice in Oracle Linux: Ksplice User's Guide for more information.

- **`apply_updates`**

  By default, the `yum-cron` interface downloads updates when the `download_updates` setting is specified. However, it does not apply them. You can change this behaviour by modifying the `apply_updates` setting as follows:

  ```
  apply_updates = yes
  ```

- **`exclude`**

  To avoid updating specific packages, use a wildcard with the `exclude` setting. For example, to exclude kernel updates set this value as follows:

  ```
  exclude = kernel*
  ```

- **`update_messages`**

  To control whether you receive notifications of new updates, you can set this value as follows:

  ```
  update_messages = yes
  ```

- **`emit_via`**

  By default, `yum-cron` is configured to output notifications to `stdio`, which means that messages are printed into `/var/log/cron`. You can disable any notifications, by setting this value to `None` or by leaving it empty. If you wish to receive email notifications, you can set this value to `email`, but you must also configure the `[email]` settings later in the configuration.

- **`[email]`**

  `yum-cron` can provide email notifications for software updates using SMTP. You can enable this by setting the `emit_via` configuration value. The configuration includes a section where you can specify email parameters including the From and To addresses used in the email notification, and the SMTP server that `yum-cron` should use to sent the mail. For example, you could configure this as follows:

```
[email]
email_from = server@example.com
email_to = admin@example.com
email_host = smtp.example.com
```

For more information about `yum-cron`, refer to the `yum-cron(8)` manual page and to the comments within the default configuration file.

10 For More Information About Yum

For more information about yum, see http://yum.baseurl.org/.

Frequently asked questions about the Oracle Linux yum server are answered at https://yum.oracle.com/faq.html.

For more information about how to download the latest packages from the Unbreakable Linux Network and make the packages available through a local yum server, see https://www.oracle.com/technical-resources/articles/it-infrastructure/unbreakable-linux-network.html.