



## ***Rocky Enterprise Linux 9.2 Manual Pages on command 'bundle-outdated2.7.1'***

**C:\>man bundle-outdated2.7.1**

BUNDLE-OUTDATED(1)

BUNDLE-OUTDATED(1)

### NAME

bundle-outdated - List installed gems with newer versions available

### SYNOPSIS

```
bundle outdated [GEM] [--local] [--pre] [--source] [--strict] [--parseable |  
--porcelain] [--group=GROUP] [--groups] [--update-strict] [--patch|--minor|--major]  
[--filter-major] [--filter-minor] [--filter-patch] [--only-explicit]
```

### DESCRIPTION

Outdated lists the names and versions of gems that have a newer version available in the given source. Calling outdated with [GEM [GEM]] will only check for newer versions of the given gems. Prerelease gems are ignored by default. If your gems are up to date, Bundler will exit with a status of 0. Otherwise, it will exit 1.

### OPTIONS

--local

Do not attempt to fetch gems remotely and use the gem cache instead.

--pre Check for newer pre-release gems.

--source

Check against a specific source.

--strict

Only list newer versions allowed by your Gemfile requirements.

--parseable, --porcelain

Use minimal formatting for more parseable output.

--group

List gems from a specific group.

--groups

List gems organized by groups.

--update-strict

Strict conservative resolution, do not allow any gem to be updated past latest --patch | --minor | --major.

--minor

Prefer updating only to next minor version.

--major

Prefer updating to next major version (default).

--patch

Prefer updating only to next patch version.

--filter-major

Only list major newer versions.

--filter-minor

Only list minor newer versions.

--filter-patch

Only list patch newer versions.

--only-explicit

Only list gems specified in your Gemfile, not their dependencies.

## PATCH LEVEL OPTIONS

See [bundle update\(1\) bundle-update.1.html](#) for details.

One difference between the patch level options in bundle update and here is the --strict option. --strict was already an option on outdated before the patch level options were added. --strict wasn't altered, and the --update-strict option on outdated reflects what --strict does on bundle update.

## FILTERING OUTPUT

The 3 filtering options do not affect the resolution of versions, merely what versions are shown in the output.

If the regular output shows the following:

- \* faker (newest 1.6.6, installed 1.6.5, requested ~> 1.4) in groups "development, test"
- \* hashie (newest 3.4.6, installed 1.2.0, requested = 1.2.0) in groups "default"
- \* headless (newest 2.3.1, installed 2.2.3) in groups "test"

--filter-major would only show:

- \* hashie (newest 3.4.6, installed 1.2.0, requested = 1.2.0) in groups "default"

--filter-minor would only show:

- \* headless (newest 2.3.1, installed 2.2.3) in groups "test"

--filter-patch would only show:

\* faker (newest 1.6.6, installed 1.6.5, requested ~> 1.4) in groups "development, test"

Filter options can be combined. --filter-minor and --filter-patch would show:

\* faker (newest 1.6.6, installed 1.6.5, requested ~> 1.4) in groups "development, test"

\* headless (newest 2.3.1, installed 2.2.3) in groups "test"

Combining all three filter options would be the same result as providing none of them.

December 2019

BUNDLE-OUTDATED(1)