

## NAME

**yum.conf** – Configuration file for **yum(8)**.

## DESCRIPTION

Yum uses a configuration file at **/etc/yum.conf**.

Additional configuration files are also read from the directories set by the **reposdir** option (default is **/etc/yum.repos.d**). See the **reposdir** option below for further details.

## PARAMETERS

There are two types of sections in the yum configuration file(s): main and repository. Main defines all global configuration options. There should be only one main section. The repository section(s) define the configuration for each repository/server. There should be one or more repository sections.

### [main] OPTIONS

The [main] section must exist for yum to do anything. It consists of the following options:

**cachedir** Directory where yum should store its cache and db files. The default is **/var/cache/yum**.

**persistdir** Directory where yum should store information that should persist over multiple runs. The default is **/var/lib/yum**.

**keepcache** Either **'1'** or **'0'**. Determines whether or not yum keeps the cache of headers and packages after successful installation. Default is **'1'** (keep files)

**reposdir** A list of directories where yum should look for .repo files which define repositories to use. Default is **/etc/yum.repos.d**. Each file in this directory should contain one or more repository sections as documented in **[repository] options** below. These will be merged with the repositories defined in **/etc/yum.conf** to form the complete set of repositories that yum will use.

**debuglevel** Debug message output level. Practical range is 0–10. Default is **'2'**.

**errorlevel** Error message output level. Practical range is 0–10. Default is **'2'**.

**rpmverbosity** Debug scriptlet output level. **'info'** is the default, other options are: **'critical'**, **'emergency'**, **'error'**, **'warn'** and **'debug'**.

**protected\_packages** This is a list of packages that yum should never completely remove. They are protected via Obsoletes as well as user/plugin removals.

The default is: **yum glob:/etc/yum/protected.d/\*.conf** So any packages which should be protected can do so by including a file in **/etc/yum/protected.d** with their package name in it.

Also if this configuration is set to anything, then yum will protect the package corresponding to the running version of the kernel.

**protected\_multilib** Either **'1'** or **'0'**. This tells yum whether or not it should perform a check to make sure that multilib packages are the same version. For example, if this option is off (rpm behavior) then in some cases it might be possible for **pkgA-1.x86\_64** and **pkgA-2.i386** to be installed at the same time. However this is very rarely desired. Install only packages, like the kernel, are exempt from this check. The default is **'1'**.

**logfile** Full directory and file name for where yum should write its log file.

**gpgcheck** Either '1' or '0'. This tells yum whether or not it should perform a GPG signature check on packages. When this is set in the [main] section it sets the default for all repositories. The default is '0'.

**localpkg\_gpgcheck** Either '1' or '0'. This tells yum whether or not it should perform a GPG signature check on local packages (packages in a file, not in a repository). The default is '0'.

**repo\_gpgcheck** Either '1' or '0'. This tells yum whether or not it should perform a GPG signature check on the repodata. When this is set in the [main] section it sets the default for all repositories. The default is '0'.

**skip\_broken** Either '1' or '0'. Resolve depsolve problems by removing packages that are causing problems from the transaction.

**assumeyes** Either '1' or '0'. Determines whether or not yum prompts for confirmation of critical actions. Default is '0' (do prompt).

Command-line option: **-y --assumeyes**

**assumeno** Either '1' or '0'. If yum would prompt for confirmation of critical actions, assume the user chose no. This is basically the same as doing "echo | yum ..." but is a bit more usable. This option overrides **assumeyes**, but is still subject to **alwaysprompt**. Default is '0' (do prompt).

Command-line option: **--assumeno**

**alwaysprompt** Either '1' or '0'. Without this option, yum will not prompt for confirmation when the list of packages to be installed exactly matches those given on the command line. Unless **assumeyes** is enabled, it will still prompt for package removal, or when additional packages need to be installed to fulfill dependencies. Default is '1'.

**tolerant** Either '1' or '0'. If enabled, then yum will be tolerant of errors on the command line with regard to packages. For example: if you request to install foo, bar and baz and baz is installed; yum won't error out complaining that baz is already installed. Default to '0' (not tolerant). Note: This option currently does nothing.

Command-line option: **-t**

**exclude** List of packages to exclude from updates or installs. This should be a space separated list. Shell globs using wildcards (eg. \* and ?) are allowed.

**+IP +query\_install\_excludes** +This applies the command line --exclude option (only, not the configuration +exclude above) to installed packages being shown in some query commands +(currently: list/info/search/provides).

**exactarch** Either '1' or '0'. Set to '1' to make yum update only update the architectures of packages that you have installed. ie: with this enabled yum will not install an i686 package to update an i386 package. Default is '1'.

**installonlypkgs** List of package provides that should only ever be installed, never updated. Kernels in particular fall into this category. Defaults to kernel, kernel-bigmem, kernel-enterprise, kernel-smp, kernel-debug, kernel-unsupported, kernel-source, kernel-devel, kernel-PAE, kernel-PAE-

debug.

Note that because these are provides, and not just package names, kernel-devel will also apply to kernel-debug-devel, etc.

Note that "kernel-modules" is **not** in this list, in RHEL-6, and so anything providing that is updated like any other package.

**installonly\_limit** Number of packages listed in installonlypkgs to keep installed at the same time. Setting to 0 disables this feature. Default is '3'. Note that this functionality used to be in the "installonlyn" plugin, where this option was altered via tokeep. Note that as of version 3.2.24, yum will now look in the yumdb for a installonly attribute on installed packages. If that attribute is "keep", then they will never be removed.

**kernelpkgnames** List of package names that are kernels. This is really only here for the updating of kernel packages and should be removed out in the yum 2.1 series.

**showduplicatesfromrepos** Either '0' or '1'. Set to '1' if you wish to show any duplicate packages from any repository, from package listings like the info or list commands. Set to '0' if you want only to see the newest packages from any repository. Default is '0'.

**obsoletes** This option only has affect during an **update**. It enables yum's obsoletes processing logic. Useful when doing distribution level upgrades. See also the yum **upgrade** command documentation for more details (yum(8)). Default is 'true'.

Command-line option: **--obsoletes**

**overwrite\_groups** Either '0' or '1'. Used to determine yum's behaviour if two or more repositories offer the package groups with the same name. If **overwrite\_groups** is '1' then the group packages of the last matching repository will be used. If **overwrite\_groups** is '0' then the groups from all matching repositories will be merged together as one large group.

**groupremove\_leaf\_only** Either '0' or '1'. Used to determine yum's behaviour when the groupremove command is run. If **groupremove\_leaf\_only** is '0' (default) then all packages in the group will be removed. If **groupremove\_leaf\_only** is '1' then only those packages in the group that aren't required by another package will be removed.

**enable\_group\_conditionals** Either '0' or '1'. Determines whether yum will allow the use of conditionals packages. Default is '1' (package conditionals are allowed).

**group\_package\_types** List of the following: optional, default, mandatory. Tells yum which type of packages in groups will be installed when 'groupinstall' is called. Default is: default, mandatory

**installroot** Specifies an alternative installroot, relative to which all packages will be installed.

Command-line option: **--installroot**

**distroverpkg** The package used by yum to determine the "version" of the distribution. This can be any installed package. Default is 'redhat-release'. You can see what provides this manually by using: "yum whatprovides redhat-release".

**diskspacecheck** Either '0' or '1'. Set this to '0' to disable the checking for sufficient disk space before a RPM transaction is run. Default is '1' (perform the check).

**tsflags** Comma or space separated list of transaction flags to pass to the rpm transaction set. These include 'noscripts', 'notriggers', 'nodocs', 'test', 'justdb' and 'nocontexts'. 'repackage' is also available but that does nothing with newer rpm versions. You can set all/any of them. However, if you don't know what these do in the context of an rpm transaction set you're best leaving it alone. Default is an empty list.

**recent** Number of days back to look for 'recent' packages added to a repository. Used by the **list recent** command. Default is '7'.

**retries** Set the number of times any attempt to retrieve a file should retry before returning an error. Setting this to '0' makes yum try forever. Default is '10'.

**keepalive** Either '0' or '1'. Set whether HTTP keepalive should be used for HTTP/1.1 servers that support it. This can improve transfer speeds by using one connection when downloading multiple files from a repository. Default is '1'.

**timeout** Number of seconds to wait for a connection before timing out. Defaults to 30 seconds. This may be too short of a time for extremely overloaded sites.

**http\_caching** Determines how upstream HTTP caches are instructed to handle any HTTP downloads that Yum does. This option can take the following values:

'all' means that all HTTP downloads should be cached.

'packages' means that only RPM package downloads should be cached (but not repository meta-data downloads).

'none' means that no HTTP downloads should be cached.

The default is 'all'. This is recommended unless you are experiencing caching related issues. Try to at least use 'packages' to minimize load on repository servers.

**throttle** Enable bandwidth throttling for downloads. This option can be expressed as a absolute data rate in bytes/sec. An SI prefix (k, M or G) may be appended to the bandwidth value (eg. '5.5k' is 5.5 kilobytes/sec, '2M' is 2 Megabytes/sec).

Alternatively, this option can specify the percentage of total bandwidth to use (eg. '60%'). In this case the **bandwidth** option should be used to specify the maximum available bandwidth.

Set to '0' to disable bandwidth throttling. This is the default.

**bandwidth** Use to specify the maximum available network bandwidth in bytes/second. Used with the **throttle** option (above). If **throttle** is a percentage and **bandwidth** is '0' then bandwidth throttling will be disabled. If **throttle** is expressed as a data rate (bytes/sec) then this option is ignored. Default is '0' (no bandwidth throttling).

**sslcacert** Path to the directory containing the databases of the certificate authorities yum should use to verify SSL certificates. Defaults to none - uses system default

**sslverify** Boolean - should yum verify SSL certificates/hosts at all. Defaults to True.

Note that the plugin yum-rhn-plugin will force this value to true, and may alter other ssl settings (like hostname checking), even if the machine is not registered.

**sslclientcert** Path to the SSL client certificate yum should use to connect to repos/remote sites Defaults to none.

Note that if you are using curl compiled against NSS (default in Fedora/RHEL), curl treats sslclientcert values with the same basename as `_identical_`. This version of yum will check that this isn't true and output an error when the repositories "foo" and "bar" violate this, like so:

sslclientcert basename shared between foo and bar

**sslclientkey** Path to the SSL client key yum should use to connect to repos/remote sites Defaults to none.

**ssl\_check\_cert\_permissions** Boolean - Whether yum should check the permissions on the paths for the certificates on the repository (both remote and local). If we can't read any of the files then yum will force `skip_if_unavailable` to be true. This is most useful for non-root processes which use yum on repos. that have client cert files which are readable only by root. Defaults to True.

**history\_record** Boolean - should yum record history entries for transactions. This takes some disk space, and some extra time in the transactions. But it allows how to know a lot of information about what has happened before, and display it to the user with the `history info/list/summary` commands. yum also provides the `history undo/redo` commands. Defaults to True.

Note that if history is recorded, yum uses that information to see if any modifications to the rpmdb have been done outside of yum. These are always bad, from yum's point of view, and so yum will issue a warning and automatically run some of "yum check" to try and find some of the worst problems altering the rpmdb might have caused.

This means that turning this option off will stop yum from being able to detect when the rpmdb has changed and thus, it will never warn you or automatically run "yum check". The problems will likely still be there, and yumdb etc. will still be wrong but yum will not warn you about it.

**history\_record\_packages** This is a list of package names that should be recorded as having helped the transaction. yum plugins have an API to add themselves to this, so it should not normally be necessary to add packages here. Not that this is also used for the packages to look for in `--version`. Defaults to rpm, yum, yum-metadata-parser.

**history\_list\_view** Which column of information to display in the "yum history list" command. There are currently three options: users, cmds (or commands), single-user-commands.

Older versions of yum acted like "users", which always outputs the user who initiated the yum transaction. You can now specify "commands" which will instead always output the command line of the transaction. You can also specify "single-user-commands" which will display the users if there are more than one, otherwise it will display the command line.

You can also specify "default" which currently selects "users".

**commands** List of functional commands to run if no functional commands are specified on the command line (eg. "update foo bar baz quux"). None of the short options (eg. -y, -e, -d) are accepted for this option.

**syslog\_ident** Identification (program name) for syslog messages.

**syslog\_facility** Facility name for syslog messages, see syslog(3). Default is 'LOG\_USER'.

**syslog\_device** Where to log syslog messages. Can be a local device (path) or a host:port string to use a remote syslog. If empty or points to a nonexistent device, syslog logging is disabled. Default is '/dev/log'.

**proxy** URL to the proxy server that yum should use.

**proxy\_username** username to use for proxy

**proxy\_password** password for this proxy

**username** username to use for basic authentication to a repo or really any url.

**password** password to use with the username for basic authentication.

**plugins** Either '0' or '1'. Global switch to enable or disable yum plugins. Default is '0' (plugins disabled). See the **PLUGINS** section of the **yum(8)** man for more information on installing yum plugins.

**pluginpath** A list of directories where yum should look for plugin modules. Default is '/usr/share/yum-plugins' and '/usr/lib/yum-plugins'.

**pluginconfpath** A list of directories where yum should look for plugin configuration files. Default is '/etc/yum/pluginconf.d'.

**metadata\_expire** Time (in seconds) after which the metadata will expire. So that if the current metadata downloaded is less than this many seconds old then yum will not update the metadata against the repository. If you find that yum is not downloading information on updates as often as you would like lower the value of this option. You can also change from the default of using seconds to using days, hours or minutes by appending a d, h or m respectively. The default is 6 hours, to compliment yum-updatesd running once an hour. It's also possible to use the word "never", meaning that the metadata will never expire. Note that when using a metalink file the metalink must always be newer than the metadata for the repository, due to the validation, so this timeout also applies to the metalink file.

**mirrorlist\_expire** Time (in seconds) after which the mirrorlist locally cached will expire. If the current mirrorlist is less than this many seconds old then yum will not download another copy of the mirrorlist, it has the same extra format as metadata\_expire. If you find that yum is not downloading the mirrorlists as often as you would like lower the value of this option.

**mdpolicy** You can select from different metadata download policies depending on how much data you want to download with the main repository metadata index. The advantages of downloading

more metadata with the index is that you can't get into situations where you need to use that metadata later and the versions available aren't compatible (or the user lacks privileges) and that if the metadata is corrupt in any way yum will revert to the previous metadata.

'instant' - Just download the new metadata index, this is roughly what yum always did, however it now does some checking on the index and reverts if it classifies it as bad.

'group:primary' - Download the primary metadata with the index. This contains most of the package information and so is almost always required anyway. This is the default.

'group:small' - With the primary also download the updateinfo metadata, this is required for yum-security operations and it also used in the graphical clients. This file also tends to be significantly smaller than most others.

'group:main' - With the primary and updateinfo download the filelists metadata and the group metadata. The filelists data is required for operations like "yum install /bin/bash", and also some dependency resolutions require it. The group data is used in some graphical clients and for group operations like "yum grouplist Base".

'group:all' - Download all metadata listed in the index, currently the only one not listed above is the other metadata, which contains the changelog information which is used by yum-changelog. This is what "yum makecache" uses.

**multilib\_policy** Can be set to 'all' or 'best'. All means install all possible arches for any package you want to install. Therefore yum install foo will install foo.i386 and foo.x86\_64 on x86\_64, if it is available. Best means install the best arch for this platform, only.

**bugtracker\_url** URL where bugs should be filed for yum. Configurable for local versions or distro-specific bugtrackers.

**color** Display colorized output automatically, depending on the output terminal, always (using ANSI codes) or never. Command-line option: **—color**

**color\_list\_installed\_older** The colorization/highlighting for packages in list/info installed which are older than the latest available package with the same name and arch. Default is 'bold'. Possible values are a comma separated list containing: bold, blink, dim, reverse, underline, fg:black, fg:red, fg:green, fg:yellow, fg:blue, fg:magenta, fg:cyan, fg:white, bg:black, bg:red, bg:green, bg:yellow, bg:blue, bg:magenta, bg:cyan, bg:white.

**color\_list\_installed\_newer** The colorization/highlighting for packages in list/info installed which are newer than the latest available package with the same name and arch. Default is 'bold,yellow'. See color\_list\_installed\_older for possible values.

**color\_list\_installed\_reinstall** The colorization/highlighting for packages in list/info installed which is the same version as the latest available package with the same name and arch. Default is 'normal'. See color\_list\_installed\_older for possible values.

**color\_list\_installed\_extra** The colorization/highlighting for packages in list/info installed which has no available package with the same name and arch. Default is 'bold,red'. See color\_list\_installed\_older for possible values.

**color\_list\_available\_upgrade** The colorization/highlighting for packages in list/info available which is an upgrade for the latest installed package with the same name and arch. Default is 'bold,blue'. See color\_list\_installed\_older for possible values.

**color\_list\_available\_downgrade** The colorization/highlighting for packages in list/info available which is a downgrade for the latest installed package with the same name and arch. Default is 'dim,cyan'. See color\_list\_installed\_older for possible values.

**color\_list\_available\_install** The colorization/highlighting for packages in list/info available which has no installed package with the same name and arch. Default is 'normal'. See color\_list\_installed\_older for possible values.

**color\_list\_available\_reinstall** The colorization/highlighting for packages in list/info available which is the same version as the installed package with the same name and arch. Default is 'bold,underline,green'. See color\_list\_installed\_older for possible values.

**color\_search\_match** The colorization/highlighting for text matches in search. Default is 'bold'. See color\_list\_installed\_older for possible values.

**color\_update\_installed** The colorization/highlighting for packages in the "updates list" which are installed. The updates list is what is printed when you run "yum update", "yum list updates", "yum list obsoletes" and "yum check-update". Default is 'normal'. See color\_list\_installed\_older for possible values.

**color\_update\_local** The colorization/highlighting for packages in the "updates list" which are already downloaded. The updates list is what is printed when you run "yum update", "yum list updates", "yum list obsoletes" and "yum check-update". Default is 'bold'. See color\_list\_installed\_older for possible values.

**color\_update\_remote** The colorization/highlighting for packages in the "updates list" which need to be downloaded. The updates list is what is printed when you run "yum update", "yum list updates", "yum list obsoletes" and "yum check-update". Default is 'normal'. See color\_list\_installed\_older for possible values.

**clean\_requirements\_on\_remove** When removing packages (by removal, update or obsoletion) go through each package's dependencies. If any of them are no longer required by any other package then also mark them to be removed. Boolean (1, 0, True, False, yes,no) Defaults to False

**reset\_nice** If set to true then yum will try to reset the nice value to zero, before running an rpm transaction. Defaults to True.

**depsolve\_loop\_limit** Set the number of times any attempt to depsolve before we just give up. This shouldn't be needed as yum should always solve or fail, however it has been observed that it can loop forever with very large system upgrades. Setting this to '0' (or "<forever>") makes yum try forever. Default is '100'.



## [repository] OPTIONS

The repository section(s) take the following form:

**Example:** [repositoryid]  
name=Some name for this repository  
baseurl=url://path/to/repository/

**repositoryid** Must be a unique name for each repository, one word.

**name** A human readable string describing the repository.

**baseurl** Must be a URL to the directory where the yum repository's 'repodata' directory lives. Can be an http://, ftp:// or file:// URL. You can specify multiple URLs in one baseurl statement. The best way to do this is like this:

```
[repositoryid]
name=Some name for this repository
baseurl=url://server1/path/to/repository/
        url://server2/path/to/repository/
        url://server3/path/to/repository/
```

If you list more than one baseurl= statement in a repository you will find yum will ignore the earlier ones and probably act bizarrely. Don't do this, you've been warned.

You can use HTTP basic auth by prepending "user:password@" to the server name in the baseurl line. For example: "baseurl=http://user:passwd@example.com/".

**metalink** Specifies a URL to a metalink file for the repomd.xml, a list of mirrors for the entire repository are generated by converting the mirrors for the repomd.xml file to a baseurl. The metalink file also contains the latest timestamp from the data in the repomd.xml, the length of the repomd.xml and checksum data. This data is checked against any downloaded repomd.xml file and all of the information from the metalink file must match. This can be used instead of or with the **baseurl** option. Substitution variables, described below, can be used with this option. This option disables the mirrorlist option. As a special hack is the mirrorlist URL contains the word "metalink" then the value of mirrorlist is copied to metalink (if metalink is not set).

**mirrorlist** Specifies a URL to a file containing a list of baseurls. This can be used instead of or with the **baseurl** option. Substitution variables, described below, can be used with this option. As a special hack is the mirrorlist URL contains the word "metalink" then the value of mirrorlist is copied to metalink (if metalink is not set).

**enabled** Either '1' or '0'. This tells yum whether or not use this repository.

**gpgcheck** Either '1' or '0'. This tells yum whether or not it should perform a GPG signature check on the packages gotten from this repository.

**repo\_gpgcheck** Either '1' or '0'. This tells yum whether or not it should perform a GPG signature check on the repodata from this repository.

**gpgkey** A URL pointing to the ASCII-armored GPG key file for the repository. This option is used if yum needs a public key to verify a package and the required key hasn't been imported into the

RPM database. If this option is set, yum will automatically import the key from the specified URL. You will be prompted before the key is installed unless the **assumeyes** option is set.

Multiple URLs may be specified here in the same manner as the **baseurl** option (above). If a GPG key is required to install a package from a repository, all keys specified for that repository will be installed.

**gpakey** A URL pointing to the ASCII-armored CA key file for the repository. This is a normal gpg public key - but this key will be used to validate detached signatures of all other keys. The idea is you are asked to confirm import for this key. After that any other gpg key needed for package or repository verification, if it has a detached signature which matches this key will be automatically imported without user confirmation.

**exclude** Same as the [main] **exclude** option but only for this repository. Substitution variables, described below, are honored here.

**includepkgs** Inverse of exclude. This is a list of packages you want to use from a repository. If this option lists only one package then that is all yum will ever see from the repository. Defaults to an empty list. Substitution variables, described below, are honored here.

**enablegroups** Either '0' or '1'. Determines whether yum will allow the use of package groups for this repository. Default is '1' (package groups are allowed).

**failovermethod** Either 'roundrobin' or 'priority'.

'roundrobin' randomly selects a URL out of the list of URLs to start with and proceeds through each of them as it encounters a failure contacting the host.

'priority' starts from the first baseurl listed and reads through them sequentially.

**failovermethod** defaults to 'roundrobin' if not specified.

**keepalive** Either '1' or '0'. This tells yum whether or not HTTP/1.1 keepalive should be used with this repository. See the global option in the [main] section above for more information.

**timeout** Overrides the **timeout** option from the [main] section for this repository.

**http\_caching** Overrides the **http\_caching** option from the [main] section for this repository.

**retries** Overrides the **retries** option from the [main] section for this repository.

**throttle** Overrides the **throttle** option from the [main] section for this repository.

**bandwidth** Overrides the **bandwidth** option from the [main] section for this repository.

**sslcert** Overrides the **sslcert** option from the [main] section for this repository.

**sslverify** Overrides the **sslverify** option from the [main] section for this repository.

**sslclientcert** Overrides the **sslclientcert** option from the [main] section for this repository.

**sslclientkey** Overrides the **sslclientkey** option from the [main] section for this repository.

**ssl\_check\_cert\_permissions** Overrides the **ssl\_check\_cert\_permissions** option from the [main] section for this repository.

**metadata\_expire** Overrides the **metadata\_expire** option from the [main] section for this repository.

**mirrorlist\_expire** Overrides the **mirrorlist\_expire** option from the [main] section for this repository.

**proxy** URL to the proxy server for this repository. Set to `'_none_'` to disable the global proxy setting for this repository. If this is unset it inherits it from the global setting

**proxy\_username** username to use for proxy. If this is unset it inherits it from the global setting

**proxy\_password** password for this proxy. If this is unset it inherits it from the global setting

**username** username to use for basic authentication to a repo or really any url. If this is unset it inherits it from the global setting

**password** password to use with the username for basic authentication. If this is unset it inherits it from the global setting

**cost** relative cost of accessing this repository. Useful for weighing one repo's packages as greater/less than any other. defaults to 1000

**skip\_if\_unavailable** If set to True yum will continue running if this repository cannot be contacted for any reason. This should be set carefully as all repos are consulted for any given command. Defaults to False.

## URL INCLUDE SYNTAX

The inclusion of external configuration files is supported for `/etc/yum.conf` and the `.repo` files in the `/etc/yum.repos.d` directory. To include a URL, use a line of the following format:

```
include=url://to/some/location
```

The configuration file will be inserted at the position of the `"include="` line. Included files may contain further include lines. Yum will abort with an error if an inclusion loop is detected.

## GLOB: FOR LIST OPTIONS

Any of the configurations options which are a list of items can be specified using the glob syntax: **glob:/etc/path/somewhere.d/\*.conf**. This will read in all files matching that glob and include all lines in

each file (excluding comments and blank lines) as items in the list.

## VARIABLES

There are a number of variables you can use to ease maintenance of yum's configuration files. They are available in the values of several options including **name**, **baseurl** and **commands**.

**\$releasever** This will be replaced with the value of the version of the package listed in **distro-verpkg**. This defaults to the version of 'redhat-release' package.

**\$arch** This will be replaced with the architecture of your system as detected by yum.

**\$basearch** This will be replaced with your base architecture in yum. For example, if your \$arch is i686 your \$basearch will be i386.

**\$uuid** This will be replaced with a unique but persistent uuid for this machine. The value that is first generated will be stored in /var/lib/yum/uuid and reused until this file is deleted.

**\$YUM0-\$YUM9** These will be replaced with the value of the shell environment variable of the same name. If the shell environment variable does not exist then the configuration file variable will not be replaced.

As of 3.2.28, any properly named file in /etc/yum/vars is turned into a variable named after the filename (or overrides any of the above variables). Filenames may contain only alphanumeric characters and underscores and be in lowercase.

Note that no warnings/errors are given if the files are unreadable, so creating files that only root can read may be confusing for users.

Also note that only the first line will be read and all new line characters are removed, as a convenience. However, no other checking is performed on the data. This means it is possible to have bad character data in any value.

## FILES

/etc/yum.conf  
/etc/yum.repos.d/  
/etc/yum/pluginconf.d/  
/etc/yum/protected.d  
/etc/yum/vars

## SEE ALSO

yum(8)