Full credit is given to the above companies including the OS that this PDF file was generated!

Red Hat Enterprise Linux Release 9.2 Manual Pages on 'systemd-sysupdate.service.8' command

$ man systemd-sysupdate.service.8

SYSTEMD-SYSUPDATE(8) systemd-sysupdate SYSTEMD-SYSUPDATE(8)

NAME

systemd-sysupdate, systemd-sysupdate.service, systemd-sysupdate.timer, systemd-sysupdate-reboot.service, systemd-sysupdate-reboot.timer - Automatically Update OS or Other Resources

SYNOPSIS

systemd-sysupdate [OPTIONS...]

systemd-sysupdate.service

DESCRIPTION

systemd-sysupdate atomically updates the host OS, container images, portable service images or other sources, based on the transfer configuration files described in sysupdate.d(5).

This tool implements file, directory, or partition based update schemes, supporting multiple parallel installed versions of specific resources in an A/B (or even: A/B/C, A/B/C/D/, ...) style. A/B updating means that when one version of a resource is currently being used, the next version can be downloaded, unpacked, and prepared in an entirely separate location, independently of the first, and ? once complete ? be activated, swapping the roles so that it becomes the used one and the previously used one becomes the one that is replaced by the next update, and so on. The resources to update are defined in transfer files, one for each resource to be updated. For example, resources that may be updated with this tool could be: a root file system partition, a

matching Verity partition plus one kernel image. The combination of the three would be considered a complete OS update.

The tool updates partitions, files or directory trees always in whole, and operates with at least two versions of each of these resources: the current version, plus the next version: the one that is being updated to, and which is initially incomplete as the downloaded data is written to it; plus optionally more versions. Once the download of a newer version is complete it becomes the current version, releasing the version previously considered current for deletion/replacement/updating.

When installing new versions the tool will directly download, decompress, unpack and write the new version into the destination. This is done in a robust fashion so that an incomplete download can be recognized on next invocation, and flushed out before a new attempt is initiated.

Note that when writing updates to a partition, the partition has to exist already, as systemd-sysupdate will not automatically create new partitions. Use a tool such as systemd-repart(8) to automatically create additional partitions to be used with systemd-sysupdate on boot.

The tool can both be used on the running OS, to update the OS in "online" state from within itself, and on "offline" disk images, to update them from the outside based on transfer files embedded in the disk images. For the latter, see --image= below. The latter is particularly interesting to update container images or portable service images.

The systemd-sysupdate.service system service will automatically update the host OS based on the installed transfer files. It is triggered in regular intervals via systemd-sysupdate.timer. The systemd-sysupdate-reboot.service will automatically reboot the system after a new version is installed. It is triggered via systemd-sysupdate-reboot.timer. The two services are separate from each other as it is typically advisable to download updates regularly while the system is up, but delay reboots until the appropriate time (i.e.

typically at night). The two sets of service/timer units may be enabled separately.

For details about transfer files and examples see sysupdate.d(5).

COMMAND

The following commands are understood:

list [VERSION]

If invoked without an argument, enumerates downloadable and installed versions, and shows a summarizing table with the discovered versions and their properties, including whether there's a newer candidate version to update to. If a version argument is specified, shows details about the specific version, including the individual files that need to be transferred to acquire the version.

If no command is explicitly specified this command is implied.

check-new

Checks if there's a new version available. This internally enumerates downloadable and installed versions and returns exit status 0 if there's a new version to update to, non-zero otherwise. If there is a new version to update to, its version identifier is written to standard output.

update [VERSION]

Installs (updates to) the specified version, or if none is specified to the newest version available. If the version is already installed or no newer version available, no operation is executed.

If a new version to install/update to is found, old installed versions are deleted until at least one new version can be installed, as configured via InstanceMax= in sysupdate.d(5), or via the available partition slots of the right type. This implicit operation can also be invoked explicitly via the vacuum command described below.

vacuum

Deletes old installed versions until the limits configured via

InstanceMax= in sysupdate.d(5) are met again. Normally, it should not be necessary to invoke this command explicitly, since it is implicitly invoked whenever a new update is initiated.

pending

Checks whether a newer version of the OS is installed than the one currently running. Returns zero if so, non-zero otherwise. This compares the newest installed version's identifier with the OS image version as reported by the IMAGE_VERSION= field in /etc/os-release. If the former is newer than the latter, an update was apparently completed but not activated (i.e. rebooted into) yet.

reboot

Similar to the pending command but immediately reboots in case a newer version of the OS has been installed than the one currently running. This operation can be done implicitly together with the update command, after a completed update via the --reboot switch, see below. This command will execute no operation (and return success) if no update has been installed, and thus the system was not rebooted.

components

Lists components that can be updated. This enumerates the /etc/sysupdate.*.d/, /run/sysupdate.*.d/ and /usr/lib/sysupdate.*.d/ directories that contain transfer files. This command is useful to list possible parameters for --component= (see below).

-h, --help

Print a short help text and exit.

--version

Print a short version string and exit.

OPTIONS

The following options are understood:

--component=, -C

Selects the component to update. Takes a component name as

argument. This has the effect of slightly altering the search logic for transfer files. If this switch is not used, the transfer files are loaded from /etc/sysupdate.d/*.conf, /run/sysupdate.d/*.conf and /usr/lib/sysupdate.d/*.conf. If this switch is used, the specified component name is used to alter the directories to look in to be /etc/sysupdate.component.d/*.conf, /run/sysupdate.component.d/*.conf and /usr/lib/sysupdate.component.d/*.conf, each time with the component string replaced with the specified component name.

Use the components command to list available components to update. This enumerates the directories matching this naming rule.

Components may be used to define a separate set of transfer files for different components of the OS that shall be updated separately. Do not use this concept for resources that shall always be updated together in a synchronous fashion. Simply define multiple transfer files within the same sysupdate.d/ directory for these cases.

This option may not be combined with --definitions=.

--definitions=

A path to a directory. If specified, the transfer *.conf files are read from this directory instead of /usr/lib/sysupdate.d/*.conf, /etc/sysupdate.d/*.conf, and /run/sysupdate.d/*.conf.

This option may not be combined with --component=.

--root=

Takes a path to a directory to use as root file system when searching for sysupdate.d/*.conf files.

--image=

Takes a path to a disk image file or device to mount and use in a similar fashion to --root=, see above. If this is used and partition resources are updated this is done inside the specified disk image.

--instances-max=, -m

Takes a decimal integer greater than or equal to 2. Controls how

many versions to keep at any time. This option may also be configured inside the transfer files, via the InstancesMax= setting, see sysupdate.d(5) for details.

--sync=

Takes a boolean argument, defaults to yes. This may be used to specify whether the newly updated resource versions shall be synchronized to disk when appropriate (i.e. after the download is complete, before it is finalized, and again after finalization). This should not be turned off, except to improve runtime performance in testing environments.

--verify=

Takes a boolean argument, defaults to yes. Controls whether to cryptographically verify downloads. Do not turn this off, except in testing environments.

--reboot

When used in combination with the update command and a new version is installed, automatically reboots the system immediately afterwards.

--no-pager

Do not pipe output into a pager.

--no-legend

Do not print the legend, i.e. column headers and the footer with hints.

--json=MODE

Shows output formatted as JSON. Expects one of "short" (for the shortest possible output without any redundant whitespace or line breaks), "pretty" (for a pretty version of the same, with indentation and line breaks) or "off" (to turn off JSON output, the default).

EXIT STATUS

On success, 0 is returned, a non-zero failure code otherwise.

SEE ALSO

systemd(1), sysupdate.d(5), systemd-repart(8)