

rsync – cheat sheet

2014-06-20

-v, --verbose	This option increases the amount of information you are given during the transfer. By default, rsync works silently. A single -v will give you information about what files are being transferred and a brief summary at the end. Two -v options will give you information on what files are being skipped and slightly more information at the end. More than two -v options should only be used if you are debugging rsync.
-I, --ignore-times	Normally rsync will skip any files that are already the same size and have the same modification timestamp. This option turns off this "quick check" behavior, causing all files to be updated.
--size-only	This modifies rsync's "quick check" algorithm for finding files that need to be transferred, changing it from the default of transferring files with either a changed size or a changed last-modified time to just looking for files that have changed in size. This is useful when starting to use rsync after using another mirroring system which may not preserve timestamps exactly.
--modify-window	When comparing two timestamps, rsync treats the timestamps as being equal if they differ by no more than the modify-window value. This is normally 0 (for an exact match), but you may find it useful to set this to a larger value in some situations. (e.g. the FAT filesystem represents times with a 2-second resolution)
-c, --checksum	This changes the way rsync checks if the files have been changed and are in need of a transfer. Without this option, rsync uses a "quick check" that (by default) checks if each file's size and time of last modification match between the sender and receiver. This option changes this to compare a 128-bit checksum for each file that has a matching size. Generating the checksums means that both sides will expend a lot of disk I/O reading all the data in the files in the transfer (and this is prior to any reading that will be done to transfer changed files), so this can slow things down significantly.
-a, --archive	This is equivalent to -rlptgoD . It is a quick way of saying you want recursion and want to preserve almost everything (with -H being a notable omission). The only exception to the above equivalence is when --files-from is specified, in which case -r is not implied. Note that -a does not preserve hardlinks , because finding multiply-linked files is expensive. You must separately specify -H .
-r, --recursive	This tells rsync to copy directories recursively. See also --dirs (-d)
-l, --links	When symlinks are encountered, recreate the symlink on the destination.
-p, --perms	This option causes the receiving rsync to set the destination permissions to be the same as the source permissions.
-t, --times	This tells rsync to transfer modification times along with the files and update them on the remote system. Note that if this option is not used, the optimization that excludes files that have not been modified cannot be effective; in other words, a missing -t or -a will cause the next transfer to behave as if it used -I , causing all files to be updated. The rsync's delta-transfer algorithm will make the update fairly efficient if the files haven't actually changed, but you're much better off using -t .
-g, --group	This option causes rsync to set the group of the destination file to be the same as the source file. If the receiving program is not running as the super-user, only groups that the invoking user on the receiving side is a member of will be preserved. Without this option, the group is set to the default group of the invoking user on the receiving side.
-o, --owner	This option causes rsync to set the owner of the destination file to be the same as the source file, but only if the receiving rsync is being run as the super-user. Without this option, the owner of new and/or transferred files are set to the invoking user on the receiving side.
-D	The -D option is equivalent to --devices --specials
--devices	This option causes rsync to transfer character and block device files to the remote system to recreate these devices. This option has no effect if the receiving rsync is not run as the super-user.

--specials	This option causes rsync to transfer special files such as named sockets and fifos.
-n, --dry-run	This makes rsync perform a trial run that doesn't make any changes (and produces mostly the same output as a real run). It is most commonly used to see what an rsync command is going to do before one actually runs it.
--delete	This tells rsync to delete extraneous files from the receiving side (ones that aren't on the sending side), but only for the directories that are being synchronized.
--exclude=PATTERN	This option is a simplified form of the --filter option that defaults to an exclude rule and does not allow the full rule-parsing syntax of normal filter rules.
-z, --compress	With this option, rsync compresses the file data as it is sent to the destination machine, which reduces the amount of data being transmitted – something that is useful over a slow connection.
--stats	This tells rsync to print a verbose set of statistics on the file transfer, allowing you to tell how effective rsync's delta-transfer algorithm is for your data.
-h, --human-readable	Output numbers in a more human-readable format. This makes big numbers output using larger units, with a K, M, or G suffix. If the option is repeated, the units are powers of 1024 instead of 1000
--progress	This option tells rsync to print information showing the progress of the transfer. This gives a bored user something to watch. Implies --verbose .
--bwlimit=KBPS	This option allows you to specify a maximum transfer rate in kilobytes per second. This option is most effective when using rsync with large files (several megabytes and up). Due to the nature of rsync transfers, blocks of data are sent, then if rsync determines the transfer was too fast, it will wait before sending the next data block. The result is an average transfer rate equaling the specified limit. A value of zero specifies no limit.

INCLUDE/EXCLUDE PATTERN RULES

- if the pattern starts with a / then it is anchored to a particular spot in the hierarchy of files, otherwise it is matched against the end of the pathname. This is similar to a leading ^ in regular expressions. Thus "/foo" would match a name of "foo" at either the "root of the transfer" (for a global rule) or in the merge-file's directory (for a per-directory rule). An unqualified "foo" would match a name of "foo" anywhere in the tree because the algorithm is applied recursively from the top down; it behaves as if each path component gets a turn at being the end of the filename. Even the unanchored "sub/foo" would match at any point in the hierarchy where a "foo" was found within a directory named "sub".
- if the pattern ends with a / then it will only match a directory, not a regular file, symlink, or device.
- rsync chooses between doing a simple string match and wildcard matching by checking if the pattern contains one of these three wildcard characters: '*', '?', and '['.
- a '*' matches any path component, but it stops at slashes.
- use '**' to match anything, including slashes.
- a '?' matches any character except a slash (/).
- a '[' introduces a character class, such as [a-z] or [[:alpha:]].
- in a wildcard pattern, a backslash can be used to escape a wildcard character, but it is matched literally when no wildcards are present.
- if the pattern contains a / (not counting a trailing /) or a "**", then it is matched against the full pathname, including any leading directories. If the pattern doesn't contain a / or a "**", then it is matched only against the final component of the filename. (Remember that the algorithm is applied recursively so "full filename" can actually be any portion of a path from the starting directory on down.)
- Here are some examples of exclude/include matching:
 - "- *.o" would exclude all names matching *.o
 - "- /foo" would exclude a file (or directory) named foo in the transfer-root directory
 - "- foo/" would exclude any directory named foo
 - "- /foo/*/bar" would exclude any file named bar which is at two levels below a directory named foo in the transfer-root directory
 - "- /foo/**/bar" would exclude any file named bar two or more levels below a directory named foo in the transfer-root directory
- "- /*." would exclude hidden files/folders in the "root" directory
- other examples: --exclude=.* --exclude=/file.dat --exclude=/folder --exclude=/folder/subfolder/file.dat