



Using tar and ssh to efficiently copy files preserving permissions Have you had situations where disk-space is sparse, so making full tars (although compressed) is impossible? Here is an ssh trick that could help you copy over files without using too much disk space. This trick will tar a directory from a computer, but the file that it would normally create, is standard out, so it is redirected back to the script on the computer you are working on. The computer you are working on extracts the information directly, so there is no location where (redundant) files are stored.

ssh user@machine-where-precious-data-is "tar czpf - /some/important/data" | tar xzpf - -C /new/root/directory You are now directly copying data from the "machine-where-precious-data-is" to the machine you are working on, using the benefits of tar (preserving permissions, links, etc) but not being hindered by the difficulties of tar. (making these possibly large files and so on.) I used this trick to copy users directories from one machine to the other. An alternative command, reverse and not crossing filesystem boundaries:

tar cpf - /some/important/data | ssh user@destination-machine "tar xpf - -C /some/directory/" • Articles • Robert de Bock • Add new comment

Comments Tue, 08/12/2014 - 17:02 — Kai (not verified) Also, if you wish to create a remote backup tar, tar into an ssh-pipe which remotely cats stdin into your target file: tar czf - -C /path/to/source files-and-folders | ssh user@target-host "cat - > /path/to/target/backup.tar.gz" • reply Wed, 08/06/2014 - 12:26 — Amyvc10 (not verified) I have always had issues with I have always had issues with managing disk-space. I was looking for a perfect solution to copy files efficiently without using too much disk space. I have never used tar and ssh for copying files and I am impressed with this method. snore stop • reply Sat, 06/07/2014 - 10:25 — Limdi (not verified) Easy way to monitor transfer Easy way to monitor transfer speed: pv Just put it between two pipes. a | pv | b pipes a to pv to b. apt-get install pv Example: ssh user@machine-where-precious-data-is "tar czpf - /some/important/data" | pv | tar xzpf - -C /new/root/directory • reply Thu, 01/23/2014 - 11:12 — Angelo Schneider (not verified) Some people call this command Some people call this command sequene "wondertar" or "wonder tar", so I write it here to make it easier to google for :D • reply Sun, 11/24/2013 - 13:31 — Alex (not verified) Is there any way to display Is there any way to display progress and transfer speed? • reply Thu, 07/18/2013 - 22:26 — Dean (not verified) Any idea what to do if I'm Any idea what to do if I'm getting a broken pipe message? • reply Mon, 12/17/2012 - 15:43 — KrK (not verified) Mistake is in the 2nd Mistake is in the 2nd example. "tar cXpf" - 'X' option is for excluding files and exclude file is not specified. Also you may consider to use '-C' option for ssh instead of using 'z' as tar option in the 1st example. On some UNIXes, like HP-UX or Solaris 'z' is not supported by the system default tar. • reply Wed, 12/19/2012 - 09:16 — Robert de Bock You are correct, removed the You are correct, removed the "X". Well spotted, thanks! • reply Sun, 12/23/2012 - 01:23 — KrK (not verified) NP. Btw, thx for the article. NP. Btw, thx for the article. I work with different UNIXes for almost 9 years and usually i do not memorize all commands and their options. MAN helps of course, but i prefer to google for articles like this one. I knew what i wanted to do, i knew which commands will do this, i even used them exactly this way like year ago. But brain do not keep rarely used information and your article just gave me an example of command line which will do the stuff, helped me to recall. So such articles are useful for UNIX newbies of course, but also may save time for ppl with experience. Good luck. :) • reply Tue, 08/10/2010 - 08:53 — Stefan (not verified) Perfect: ssh Perfect: ssh user@machine-where-precious-data-is "tar czpf - /some/important/data" | tar xzpf - -C /new/root/directory Thanks for a wicked useful trick!