How to add cron jobs from command line or bash scripts / Add crontab jobs in a script

Author : admin

I'm currently writting a script which is supposed to be adding new crontab jobs and do a bunch of other mambo jambo.

By so far I've been aware of only one way to add a cronjob non-interactively like so:

> linux:~# echo '*/5 * * * * /root/myscript.sh' | crontab -

Though using the **| crontab -** would work it has one major pitfall, I did completely forgot / *crontab -* **OVERWRITES CURRENT CRONTAB!** with the crontab passed by with the echo command.
One must be extremely careful if he decides to use the above example as you might loose your crontab definitions permanently!

Thanksfully it seems there is another way to add crontabs non interactively via a script, as I couldn't find any good blog which explained something different from the classical example with pipe to *crontab -*, I dropped by in the good old irc.freenode.net to consult the bash gurus there ;)

So I entered irc and asked the question **how can I add a crontab via bash shell script without overwritting my old existing crontab definitions** less than a minute later one guy with a nickname **geirha** was kind enough to explain me how to get around the annoying overwridding.

The solution to the ovewrite was expected, first you use crontab to dump current crontab lines to a file and then you append the new cron job as a new record in the file and finally you ask the crontab program to read and insert the crontab definitions from the newly created files.
So here is the exact code one could run inside a script to include new crontab jobs, next to the already present ones:

linux:~# crontab -l > file; echo '*/5 * * * * /root/myscript.sh >/dev/null 2>&1' >> file; crontab file

The above definition as you could read would make the new record of ***/5 * * * * /root/myscript.sh >/dev/null** be added next to the existing crontab scheduled jobs.

Now I'll continue with my scripting, in the mean time I hope this will be of use to someone out there ;)