**Setting automatic Crontab backups for all Users in Linux Environment**

There are loads of ways to take crontab backup as part of Disaster Recovery Management action, but among all, below one is one of the easiest steps I have ever discovered.

Instead of running a crontab backup script individually for each user, we can run the below-attached code as a ‘root’ user which will create crontab backups for all the users at a single go.

While editing the crontab, all the changes in crontab files get stored at a specific file directory, eg: ‘/var/spool/cron/’ directory. We can directly use those files present inside ‘/var/spool/cron/’ directory as a backup file by copying them to a certain location periodically. The same approach stated above has been implemented in the attached code 'automation linux shell script'.

The above crontab backup script can run every day to record everyday changes. The backup frequency depends on the business to business and server space. Based on my analysis for a fully loaded test lab contains more than 40+ users, the total size of my backup on my Linux box would not cross more than 200 KB for each day (approx. 11.7 MB for all 60 days), which is pretty small. Also, the above script can purge/delete the backups older than 60 days.

The script can store backups at ‘/tmp/’ directory but it can modify it to some other directory based on the requirement. Replace ‘crondir’ variable in the above script with the directory where you want your backups to get placed.

**Note:** Sometimes ‘/tmp’ directory gets cleaned up at some fixed intervals, hence it is not advisable to keep backups in ‘/tmp’ directory.

According to the attached script, all the backups will get stored in the ‘tmp’ directory. Within the directory, there will be a sub-directory with named timestamp “YYYYMMDD\_HHMMSS” for every day. Under the timestamp directory, all the backups will be stored. Find the below demo.

**Backup files:**

**#> cd /tmp**

**#> ls -lrt**

drwxrwsrwx 2 root root 4096 Aug 17 04:35 20180617\_065899

**#> ls -lrt 20180817\_043126**

-rw------- 1 user1 user1 253 Aug 17 04:31 user1

-rw------- 1 user2 user2 385 Aug 17 04:31 user2

-rw------- 1 user3 user3 1292 Aug 17 04:31 user3

<.. more files ..>

**Log files:**

All the log files will be placed at ‘/tmp/logs’ directory with current date suffix in ‘YYYYMMDD’ format as mentioned below.

**#> cat /tmp/logs /backup\_crontab\_alluser\_20180817.log**

This directory has been created...

Crontab backed up successfully...

**Advantages:**

* We don’t need to set up a crontab backup script manually for new upcoming user/use cases.
* This approach will save a significant amount of time by automating crontab backups.
* CPU utilization will also get improved by running just 1 script for all the use-cases instead of running 1 script for each user/use cases.