Cron Job is a Linux utility that sets a schedule command or script on your server so that some tasks run automatically at a specified time and date**. Crontab** command is used to create, modify and view cron jobs.

* 1: Minute (0-59)
* 2: Hours (0-23)
* 3: Day (0-31)
* 4: Month (0-12 [12 == December])
* 5: Day of the week(0-7 [7 or 0 == sunday])
* /path/to/command – Script or command name to schedule

Syntax: 1 2 3 4 5 /root/backup.sh

**Operators**

**The asterisk (\*)**: This operator specifies all possible values for a field.

**The comma (,)** : This operator specifies a list of values, for example: “1,5,10,15,20, 25”.

**The dash (-)** : This operator specifies a range of values, for example: “5-15” days

**The separator (/)** : This operator specifies a step value, for example: “0-23/” can be used in the hours field to specify command execution every other hour.

1. Write a Script to detect ip addresses trying to gain access, examples of things to pay attention to include all use between midnight and 6, all logins for a specific user, anything else you consider behavior that should send up a red flag.

**last** : command that shows the recent logins

**last username :** command that show the logins of a particular user

**last -s** : (Since) option to restrict the output to only show login event that took place since a specific data

**last -t** : (until) option to specific an end date.

#!/bin/bash

last -s 00:00 -t 06:00

last robert

The above script will show the logs of users that login between midnight and 6 as well as logins from a user named robert.

Additional commands to expand our user behavior script

**sudo cat /var/log/btmp** : Display all failed login attempts.

**sudo cat var/log/utmp** : Display the current login user.

**sudo cat /var/log/wtmp** : Display the record of each login/logout.

**sudo cat /var/log/lastlog** : Display every user’s last login.

**sudo cat /var/log/secure** : Display authentication and security related messages and error. This is helpful to keeps tracks of users auhthentication requests. Debian/Ubuntu information is stored in **/var/log/auth.log**, while Redhat/CentOS is stored in **/var/log/secure**.

**sudo cat var/log/maillog** : Display the mails logs sent from the server.

**iostat** : Command to monitor system input/output . Make sure to install this command line with **sudo apt-get install sysstat -y** .

Another Method

#!/bin/bash

sudo cat /var/log/btmp >> Filename

sudo cat var/log/utmp >> Filename

sudo cat /var/log/wtmp >> Filename

sudo cat /var/log/lastlog >> Filename

sudo cat /var/log/secure >> Filename

sudo cat var/log/maillog >> Filename

iostat >> Filename

Note: On this case we have to use cron. We can set cron to run the script automatically at a later time.

**crontab -e** : command to edit or create our own crontab file.

Within the editor type the following: **0 0-6 \* \* \* /home/user/script.sh**

This will run the script every day from 12 a.m. to 6 a.m. without the need of a user input.

2. Write script to detect changes to a specific directory.  Such as changes to /var/log or /etc/ think about using a diff here, or a hash.

What is **Auditd** ? A Linux command utility that provides a way to create a log rule for every action on a server. It is very useful to know what process was executed when, what I delete / modify / read, etc.

**-w** is the path and files to watch.

**-p** is the permissions to monitor. **r**– for read access, **w** – for write access,**x** – for execute access and a – for change of file or director attribute.

**-k** is the key name for the rule.

To create a rule, type the following in the command line :

**sudo vi /etc/audit/rules.d/audit.rules**

At the bottom of that file, add the following lines

**-w /home/user -p wa -k var\_log**

**-w /etc/shadow -p wa -k shadow**

**-w /etc/passwd -p wa -k psswd**

The above script will detect for any changes to the directory logs and user account information.

The built-in auditd search command allows us to view for any entries that contains the key name in the auditd log

**ausearch -k**

**var\_log ausearch -k**

**shadow ausearch -k psswd**

3. Monitor hidden files, root executables, and see if changes are made, who made them, and when they were changed.

**mkdir** : command to create a directory

**mv** : command to move files

**mv** /home/user/.\* /home/user/new\_dir

Note: Create a directory and move all hidden files to the new directory.

To monitor hidden files

**-w /home/user/new\_directory**

The following script allows to monitor root executables and then will create a file to see if changes were made.

**-w /var/log -p wa -k var\_log**

References

Gite, Vivek. “How To Add Jobs To Cron Under Linux or UNIX.” *NixCraft*, 30 Sept. 2019, www.cyberciti.biz/faq/how-do-i-add-jobs-to-cron-under-linux-or-unix-oses/.

Kili, Aaron, et al. “Aaron Kili.” *Tecmint*, 22 Sept. 2017, www.tecmint.com/linux-system-auditing-with-auditd-tool-on-centos-rhel/.

McKay, Dave. “How to Use the Last Command on Linux.” *How*, How-To Geek, 31 May 2019, www.howtogeek.com/416023/how-to-use-the-last-command-on-linux/.

Wallen, Jack. “How to Monitor Events on Your Linux Data Center Servers with Auditd.” *TechRepublic*, TechRepublic, 5 Mar. 2019, www.techrepublic.com/article/how-to-monitor-events-on-your-linux-data-center-servers-with-auditd/.