1. I had trouble finding a way to check users on a server. I eventually came across the w command. This command retrieves info on logged in users and displays what they are doing. It also reports a remote user's machine address, command they are using at the time, and the time they logged in. Below is the script I came up with.

now=$(date +%Y-%m-%d-%H:%M)

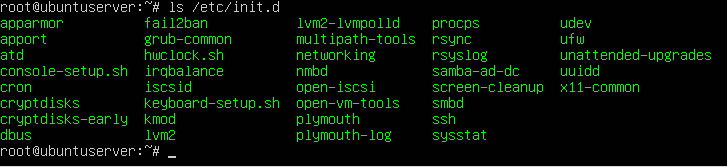
w > networkusers$now.txt

Running the script on a test machine gave the following output.

Untitled:securitytest.png

This script will run the command and output it to a file that contains the date and time the script was run. In order for this to be effective, a cron job should be used to run it every hour. Using the command 'crontab -e' and create a new job that will run the script. At the start of the line for this new job, you should have the string '0 \* \* \* \*'. This will schedule the job to be done every hour. You will need to follow this string with the exact path to your script. Cron jobs are used to perform repetitive tasks that need to be done at specified times. They could be used to track activity or starts scripts automatically at times when servers are less active.

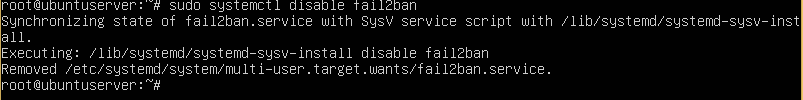
2. To view the daemons that run on boot, use the command 'ls /etc/init.d/'.



To enable a daemon, you need its name. Use the command 'sudo systemctl enable' and then the name of the daemon. To disable the daemon, you use the same command, only replace 'enable' with 'disable'.

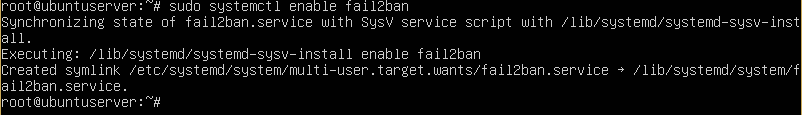
Let's try it.

Here we enter the command 'sudo systemctl disable fail2ban'.



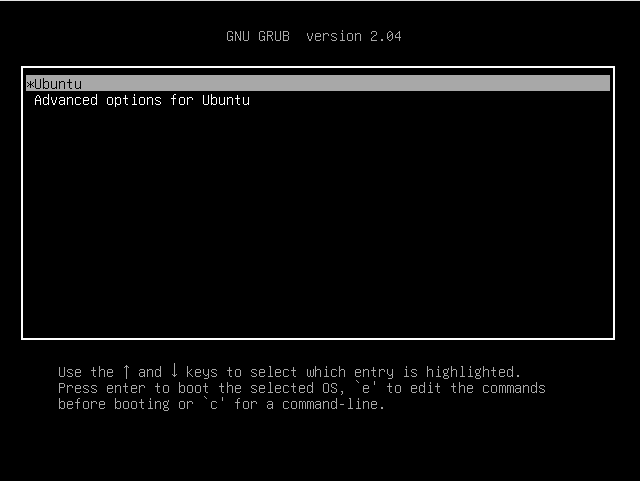
Now, fail2ban has been disabled.

To enable it again, we use the command 'sudo systemctl enable fail2ban'.



Now fail2ban is enabled again.

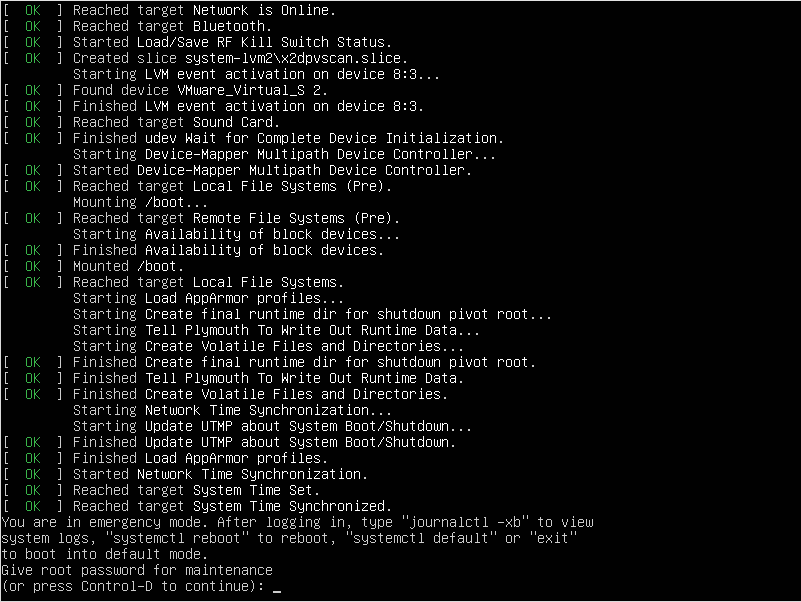
3. On an Ubuntu system, booting into emergency mode is pretty simple. Turn on the system and after the BIOS logo disappears hit the ESC key. This will display the Grub menu.



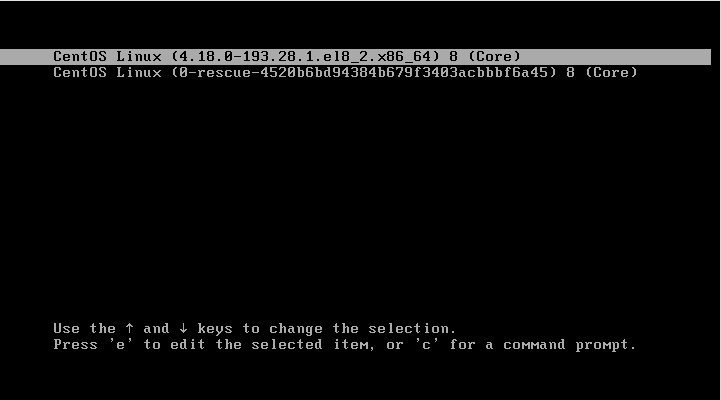
Select the Ubuntu option and then press 'e' to edit. Find the line that begins with 'linux' and add to the end of that line 'systemd.unit=emergency.target'.



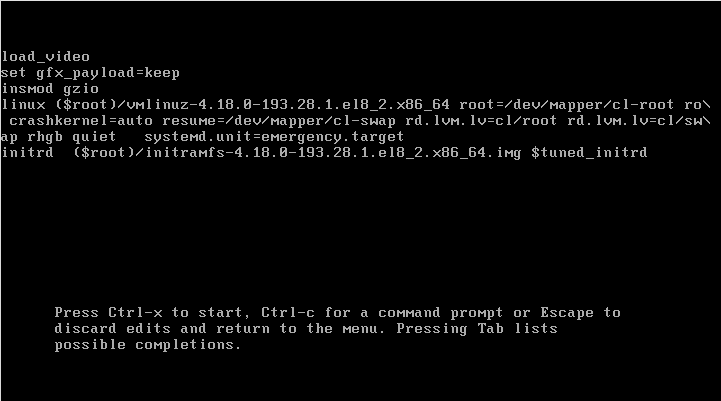
Once you have done that, press either Ctrl-x or F10 and you will enter emergency mode.



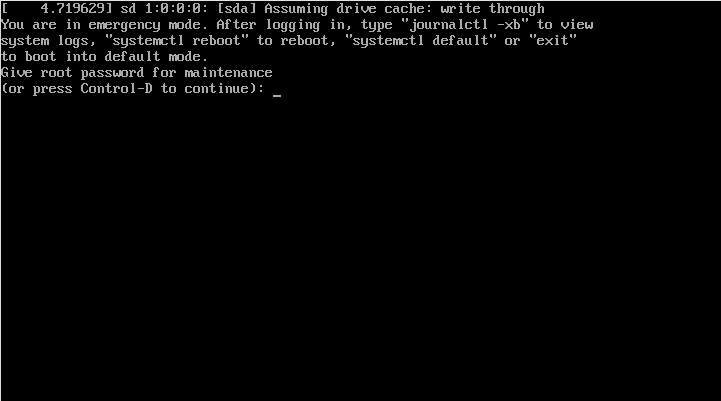
On a CentOS machine, to boot into emergency mode, power up the machine and when the boot menu appears press any key but enter.



Now, press 'e' to modify the first entry. Find the line that starts with 'linux' and add 'systemd.unit=emergency.target' to the end of the line.



Once everything is done, hit Ctrl-x to boot in emergency mode.



Emergency mode is similar to rescue mode in a few ways. But the biggest difference is that emergency mode mounts the root filesystem only instead of all local filesystems. On top of that, only the most basic services are started. If init files are corrupted, booting into rescue mode might not be an option. Since init files are not loaded in emergency mode, you are able to mount other filesystems and potentially recover data that could be lost on reinstall.

sources

www.thegeekstuff.com/2009/03/4-ways-to-identify-who-is-logged-in-on-your-linux-system/

likegeeks.com/listing-users-in-linux/

www.hostinger.com/tutorials/cron-job

bash.cyberciti.biz/guide/Daemons

ostechnix.com/how-to-boot-into-rescue-mode-or-emergency-mode-in-ubuntu-18-04/