

CS312 Project #1 Answer Key

February 8, 2016

Answers

1. Setup an OpenStack virtual machine and do the following:
 - (a) Configure `rsyslog` to send all `local2` facility messages with a warning severity level to `/var/log/cs312.log`
 - (b) Use the `logger` utility and send a message chosen by you with the same facility and severity level as described above. Show the command you used.
 - (c) Show the output of the of the `/var/log/cs312.log` and the last 5 lines of `/var/log/messages`.

```
# /etc/rsyslog.d/cs312.conf
local2.warning /var/log/cs312.log

$ systemctl restart rsyslog
$ logger -p local2.warning 'this is a message'

$ cat /var/log/cs312.log
Feb  4 17:32:36 proj1 centos: this is a message

$ tail -n 5 /var/log/messages
Feb  8 17:54:13 proj1 systemd: Stopping System Logging
    => Service...
Feb  8 17:54:13 proj1 systemd: Starting System Logging
    => Service...
Feb  8 17:54:13 proj1 systemd: Started System Logging
    => Service.
Feb  8 17:54:23 proj1 centos: this is a message
```

2. Using the `ks.cfg` example from the slides as a base, modify it to do the following. Assume all changes are done via kickstart options and not done using pre or post installation scripts. Please include your full `ks.cfg` config file for the answer and the contents of `/var/log/anaconda/anaconda.log` in a separate file. Please test your kickstart file using virtualbox or vmware to ensure it works properly:
 - (a) Setup the volumes to be: `/boot` (512M), `swap` (1G), `/` (rest of the disk). Have `/boot` be a primary partition, while `swap` and the `rootfs` be logical volumes with the volume group named `vg_cs312`.
 - (b) Add the EPEL repository using the following as the URL:
`http://epel.osuosl.org/7/x86-64/`. This repository is only needed during the kickstart installation phase.
 - (c) Add the base CentOS repository using the following as the URL:
`http://centos.osuosl.org/7/os/x86-64/`
 - (d) Set to our current timezone
 - (e) Install `bash-completion` package
 - (f) Install the Mariadb (Mysql) Server package
 - (g) Enable the Mariadb (Mysql) service on boot

```
install
cdrom
lang en_US.UTF-8
keyboard us
network --bootproto=dhcp
rootpw cs312
firewall --disabled
selinux --permissive
unsupported_hardware
bootloader --location=mbr
text
skipx
zerombr
clearpart --all --initlabel
part /boot --fstype=ext4 --size=512
part pv.01 --grow --size=100
volgroup vg_cs312 pv.01
logvol swap --vgname=vg_cs312 --name=swap --fstype=swap --
    ↪ size=1024
```

```

logvol / --vgname=vg_cs312 --name=root --fstype=ext4 --grow
    ↪ --size=100
repo --name=epel --baseurl=http://epel.osuosl.org/7/x86_64/
repo --name=base --baseurl=http://centos.osuosl.org/7/os/
    ↪ x86_64/
timezone US/Pacific
# OR
# timezone America/Los_Angeles
services --enabled=mariadb
auth --enablshadow --passalgo=sha512 --kickstart
firstboot --disabled
poweroff
user --name=cs312 --plaintext --password cs312

%packages --nobase
bash-completion
mariadb-server
%end

```

- Using `dd` and `losetup`, create enough 100MB loopback devices for a RAID5 array. How many loopback devices do you need?

```

for i in 1 2 3 ; do
    dd if=/dev/zero of=disk${i} bs=1M count=100
    losetup /dev/loop${i} disk${i}
done

```

3 loopback devices

- Create a RAID5 software raid array using the loop devices in the previous question as a device named `/dev/md0`. Please show the command, the output it shows, the contents of `/proc/mdstat` and the output of `mdadm -D /dev/md0`.

```

$ mdadm -C /dev/md0 --level=5 --raid-devices=3 \
  /dev/loop1 /dev/loop2 /dev/loop3
mdadm: Defaulting to version 1.2 metadata
mdadm: array /dev/md0 started.

$ cat /proc/mdstat
Personalities : [raid6] [raid5] [raid4]
md0 : active raid5 loop3[3] loop2[1] loop1[0]
      202752 blocks super 1.2 level 5, 512k chunk, algorithm
      ↪ 2 [3/3] [UUU]

```

unused devices: <none>

```
$ mdadm -D /dev/md0
/dev/md0:
    Version : 1.2
  Creation Time : Thu Feb  4 18:49:22 2016
    Raid Level : raid5
    Array Size : 202752 (198.03 MiB 207.62 MB)
  Used Dev Size : 101376 (99.02 MiB 103.81 MB)
    Raid Devices : 3
  Total Devices : 3
  Persistence : Superblock is persistent

    Update Time : Thu Feb  4 18:49:23 2016
      State : clean
  Active Devices : 3
 Working Devices : 3
 Failed Devices : 0
 Spare Devices : 0

    Layout : left-symmetric
  Chunk Size : 512K

     Name : proj1.novalocal:0 (local to host proj1.
        ↪ novalocal)
    UUID : fd5a0218:cb1f4095:7f0855c2:a5b3294a
  Events : 18
```

Number	Major	Minor	RaidDevice	State	
0	7	1	0	active sync	/dev/
	↪ loop1				
1	7	2	1	active sync	/dev/
	↪ loop2				
3	7	3	2	active sync	/dev/
	↪ loop3				

5. Using the previous RAID5 array, fail one of the drives, remove the disk. Show the commands, their output, the contents of /proc/mdstat and the output of mdadm -D /dev/md0.

```
$ mdadm /dev/md0 -f /dev/loop1
mdadm: set /dev/loop1 faulty in /dev/md0
```

```
$ mdadm /dev/md0 -r /dev/loop1
mdadm: hot removed /dev/loop1 from /dev/md0
```

```

$ mdadm -D /dev/md0
/dev/md0:
    Version : 1.2
  Creation Time : Thu Feb  4 18:49:22 2016
    Raid Level : raid5
    Array Size : 202752 (198.03 MiB 207.62 MB)
  Used Dev Size : 101376 (99.02 MiB 103.81 MB)
    Raid Devices : 3
  Total Devices : 2
  Persistence : Superblock is persistent

    Update Time : Thu Feb  4 18:52:42 2016
      State : clean, degraded
  Active Devices : 2
 Working Devices : 2
 Failed Devices : 0
  Spare Devices : 0

    Layout : left-symmetric
  Chunk Size : 512K

    Name : proj1.novalocal:0 (local to host proj1.
      ↪ novalocal)
   UUID : fd5a0218:cb1f4095:7f0855c2:a5b3294a
  Events : 21

    Number   Major   Minor   RaidDevice State
     0         0       0         0   removed
     1         7       2         1   active sync  /dev/
      ↪ loop2
     3         7       3         2   active sync  /dev/
      ↪ loop3

```

6. Write out a cronjob definition that does the following: Run the command `date` every 3 hours at 22 minutes past the hour but only on Fridays, Sundays and Wednesdays. Also configure the email to be sent to `foo@example.org`.

```

MAILTO=foo@example.org
22 */3 * * 0,3,5 /bin/date
# OR
22 */3 * * sun,wed,fri /bin/date

```

7. Write a systemd service unit file that does the following:

- (a) Is parameterized
- (b) Is considered active after all processes are exited
- (c) Runs the command `/bin/echo <msg>` where `<msg>` is an arbitrary message chosen by you and includes the parameterized variable

```
[Unit]
Description=CS312 Echo Service
```

```
[Service]
Type=oneshot
RemainAfterExit=yes
ExecStart=-/bin/echo '%I'
```

```
[Install]
WantedBy=multi-user.target
```

8. Suppose this unit file is named `foo@.service` and located in `/etc/systemd/system`. Provide the commands to do the following:
- (a) Enable the service without using `systemctl` (Hint: this command should create a symlink)
 - (b) Start a parameterized instance of the service with the parameter set to `bar`. (You can use `systemctl` on this part)

```
# Create parameterized service
$ ln -s /etc/systemd/system/foo@.service \
  /etc/systemd/system/foo@bar.service
# Enable parameterized service
$ ln -s /etc/systemd/system/foo@bar.service \
  /etc/systemd/system/default.target.wants/foo@bar.service
# Reload systemd
$ systemctl daemon-reload
# Start the service
$ systemctl start foo@bar.service
```