

STEPS TO INSTALL/CONFIGURE KATELLO

SERVER SIDE CONFIGURATION

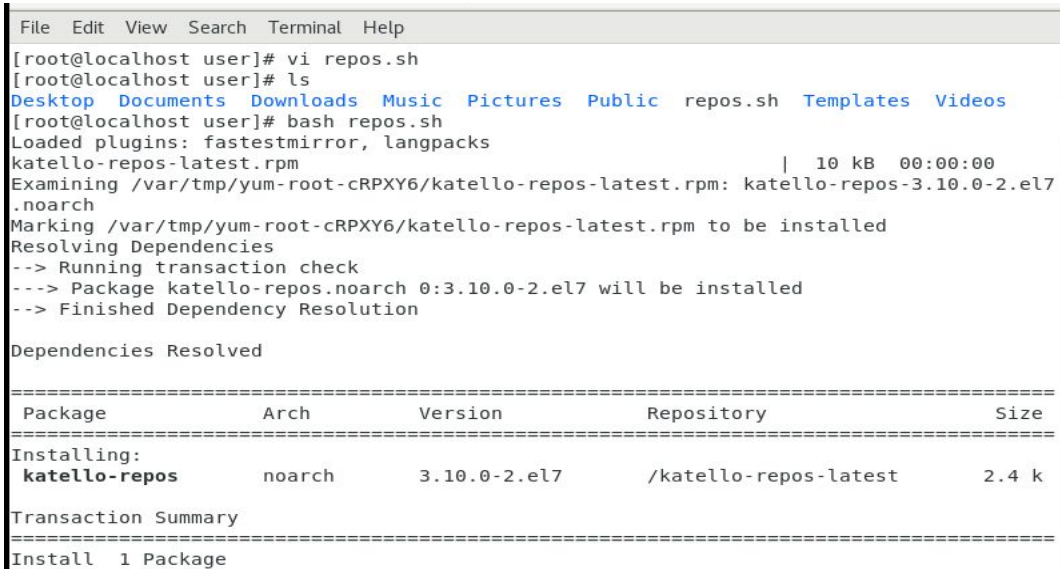
1. First we need to execute all commands as root. For that we will use “**sudo su**” to execute commands with root privileges.
2. Now create a **repos.sh** file with the following commands to include the repositories required for Katello:

```
yum -y localinstall http://dl.fedoraproject.org/pub/epel/epel-release-latest-7.noarch.rpm
yum -y localinstall
http://yum.theforeman.org/releases/1.20/el7/x86\_64/foreman-release.rpm
yum -y localinstall http://yum.puppetlabs.com/puppetlabs-release-pc1-el-7.noarch.rpm
yum -y localinstall
http://fedorapeople.org/groups/katello/releases/yum/3.10/katello/el7/x86\_64/katello-repos-latest.rpm
yum -y install foreman-release-scl python2-django
```

3. Execute the shell file using :
bash <filename.sh>

For example –

```
bash repos.sh
```



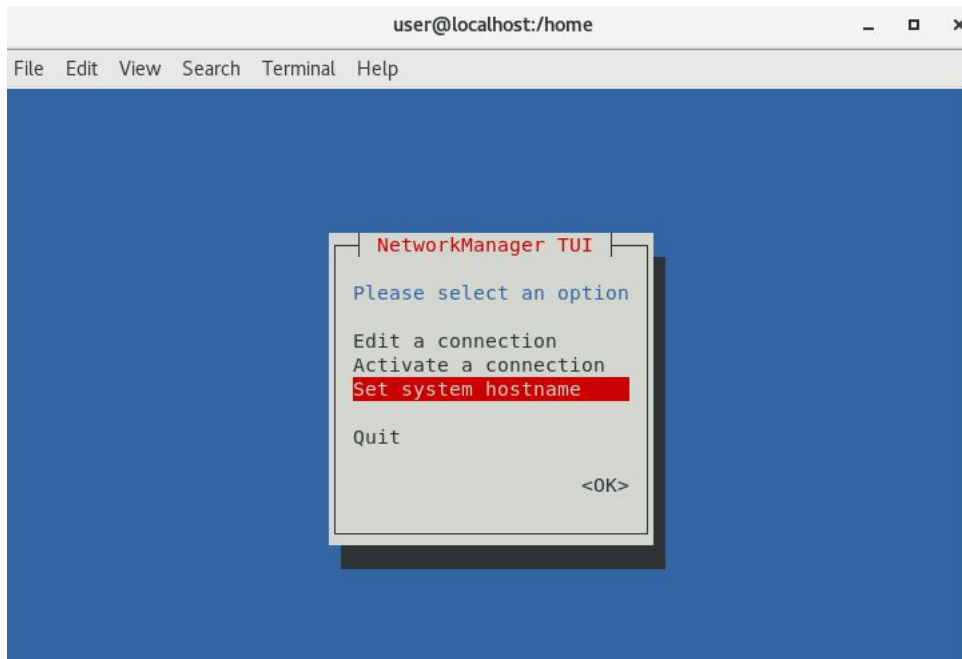
```
File Edit View Search Terminal Help
[root@localhost user]# vi repos.sh
[root@localhost user]# ls
Desktop Documents Downloads Music Pictures Public repos.sh Templates Videos
[root@localhost user]# bash repos.sh
Loaded plugins: fastestmirror, langpacks
katello-repos-latest.rpm                               | 10 kB  00:00:00
Examining /var/tmp/yum-root-cRPXY6/katello-repos-latest.rpm: katello-repos-3.10.0-2.el7
.noarch
Marking /var/tmp/yum-root-cRPXY6/katello-repos-latest.rpm to be installed
Resolving Dependencies
--> Running transaction check
--> Package katello-repos.noarch 0:3.10.0-2.el7 will be installed
--> Finished Dependency Resolution

Dependencies Resolved

=====
Package                Arch      Version      Repository      Size
=====
Installing:
katello-repos          noarch    3.10.0-2.el7 /katello-repos-latest 2.4 k
=====

Transaction Summary
=====
Install 1 Package
```

4. Now set a hostname for your host machine by typing **nmtui** and editing the hostname. Here we have given the hostname as – **katello.server.com**



Now we refresh the CLI using the command – **bash**

5. Now we need to add this hostname to our **/etc/hosts** file. To do this we will use the command:
echo "<server ip> <fqdn> <sub-domain>" >> /etc/hosts
For example:-

```
echo "192.168.9.10 katello.server.com katello" >> /etc/hosts
```

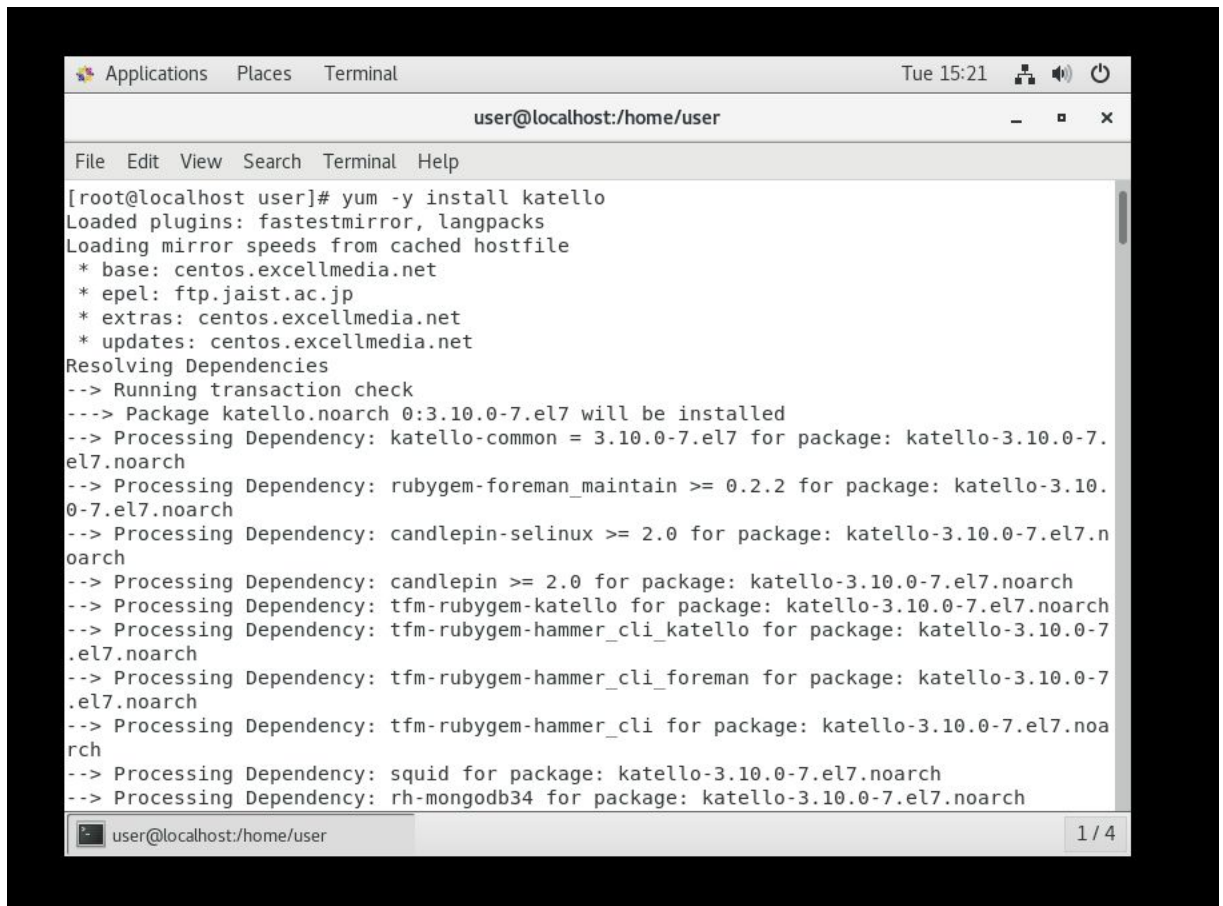
6. After setting up the appropriate repositories and changing the hostname, update your packages:

```
yum -y update
```

```
[root@localhost user]# yum -y update
Loaded plugins: fastestmirror, langpacks
Loading mirror speeds from cached hostfile
 * base: centos.excellmedia.net
 * epel: ftp.jaist.ac.jp
 * extras: centos.excellmedia.net
 * updates: centos.excellmedia.net
```

7. Then install Katello:

```
yum -y install katello
```

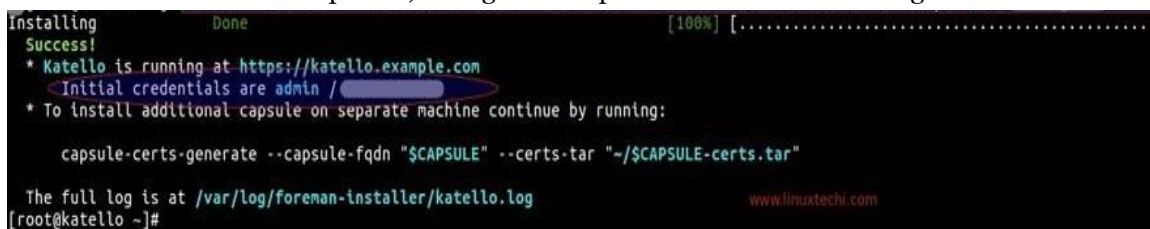
A terminal window titled 'user@localhost:/home/user' showing the command 'yum -y install katello'. The output displays the resolution of dependencies for the katello-3.10.0-7.el7.noarch package, including katello-common, rubygem-foreman_maintain, candlepin-selinux, candlepin, tfm-rubygem-katello, tfm-rubygem-hammer_cli_katello, tfm-rubygem-hammer_cli_foreman, tfm-rubygem-hammer_cli, squid, and rh-mongodb34. The window has a menu bar with 'File', 'Edit', 'View', 'Search', 'Terminal', and 'Help'. The status bar at the bottom shows 'user@localhost:/home/user' and '1 / 4'.

```
[root@localhost user]# yum -y install katello
Loaded plugins: fastestmirror, langpacks
Loading mirror speeds from cached hostfile
 * base: centos.excellmedia.net
 * epel: ftp.jaist.ac.jp
 * extras: centos.excellmedia.net
 * updates: centos.excellmedia.net
Resolving Dependencies
--> Running transaction check
--> Package katello.noarch 0:3.10.0-7.el7 will be installed
--> Processing Dependency: katello-common = 3.10.0-7.el7 for package: katello-3.10.0-7.el7.noarch
--> Processing Dependency: rubygem-foreman_maintain >= 0.2.2 for package: katello-3.10.0-7.el7.noarch
--> Processing Dependency: candlepin-selinux >= 2.0 for package: katello-3.10.0-7.el7.noarch
--> Processing Dependency: candlepin >= 2.0 for package: katello-3.10.0-7.el7.noarch
--> Processing Dependency: tfm-rubygem-katello for package: katello-3.10.0-7.el7.noarch
--> Processing Dependency: tfm-rubygem-hammer_cli_katello for package: katello-3.10.0-7.el7.noarch
--> Processing Dependency: tfm-rubygem-hammer_cli_foreman for package: katello-3.10.0-7.el7.noarch
--> Processing Dependency: tfm-rubygem-hammer_cli for package: katello-3.10.0-7.el7.noarch
--> Processing Dependency: squid for package: katello-3.10.0-7.el7.noarch
--> Processing Dependency: rh-mongodb34 for package: katello-3.10.0-7.el7.noarch
```

- Now we shall use foreman-installer to setup Katello:

foreman-installer --scenario katello

After the installation is completed , we'll get an output which'll look something like this

A terminal window showing the output of the 'foreman-installer --scenario katello' command. The output indicates that the installation is successful and provides information about the Katello instance, including the URL 'https://katello.example.com' and the initial credentials 'admin / [redacted]'. It also provides instructions for installing additional capsules and generating certificates. The window has a status bar at the bottom showing 'www.linuxtechi.com'.

```
Installing Done [100%] [.....]
Success!
 * Katello is running at https://katello.example.com
 * Initial credentials are admin / [redacted]
 * To install additional capsule on separate machine continue by running:

capsule-certs-generate --capsule-fqdn "$CAPSULE" --certs-tar "-/$CAPSULE-certs.tar"

The full log is at /var/log/foreman-installer/katello.log
[root@katello ~]#
```

- Now we need to configure certain settings in foreman for which we shall use the '-i' argument(for interactive mode) with the **foreman-installer** command.

foreman-installer -i

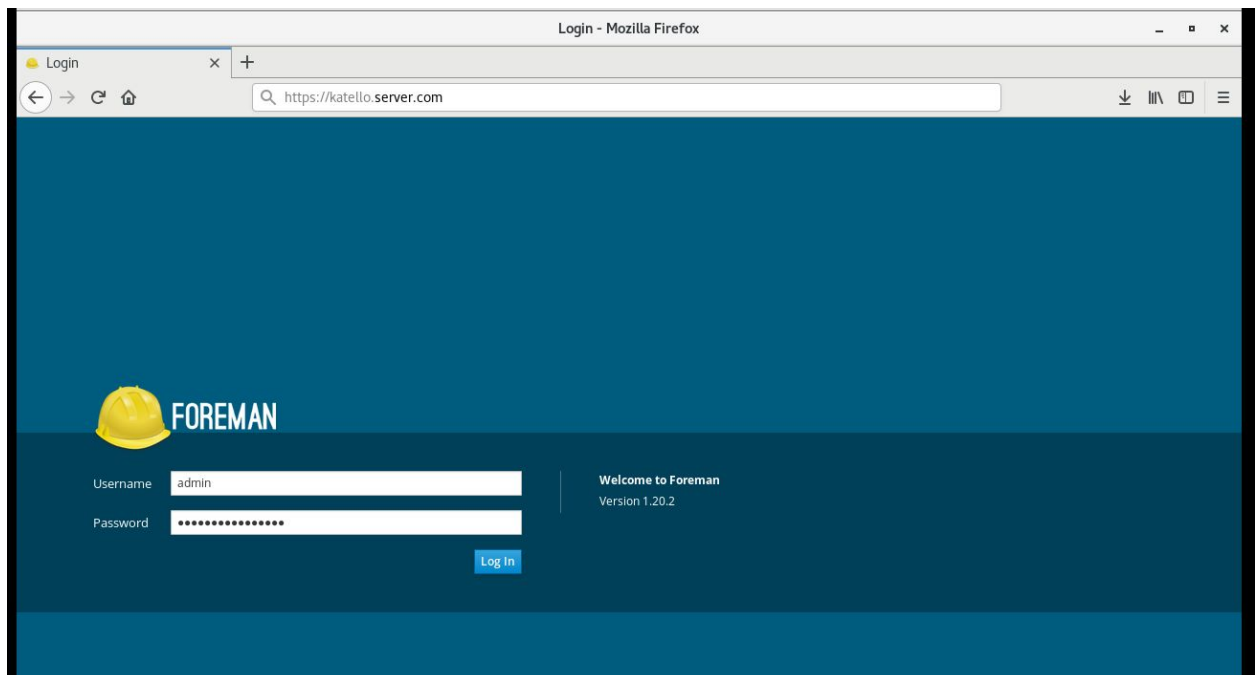
```
File Edit View Search Terminal Help
3. [✓] Configure foreman_cli
4. [x] Configure foreman_cli_discovery
5. [x] Configure foreman_cli_openscap
6. [x] Configure foreman_cli_remote_execution
7. [x] Configure foreman_cli_tasks
8. [x] Configure foreman_cli_templates
9. [x] Configure foreman_compute_ec2
10. [x] Configure foreman_compute_gce
11. [x] Configure foreman_compute_libvirt
12. [x] Configure foreman_compute_openstack
13. [x] Configure foreman_compute_ovirt
14. [x] Configure foreman_compute_rackspace
15. [x] Configure foreman_compute_vmware
16. [x] Configure foreman_plugin_ansible
17. [x] Configure foreman_plugin_bootdisk
18. [x] Configure foreman_plugin_chef
19. [x] Configure foreman_plugin_default_hostgroup
20. [x] Configure foreman_plugin_discovery
21. [x] Configure foreman_plugin_hooks
22. [x] Configure foreman_plugin_openscap
23. [x] Configure foreman_plugin_puppetdb
24. [x] Configure foreman_plugin_remote_execution
25. [x] Configure foreman_plugin_setup
26. [✓] Configure foreman_plugin_tasks

-- press enter/return to continue or q to stop --
```

user@localhost:/home/user 1 / 4

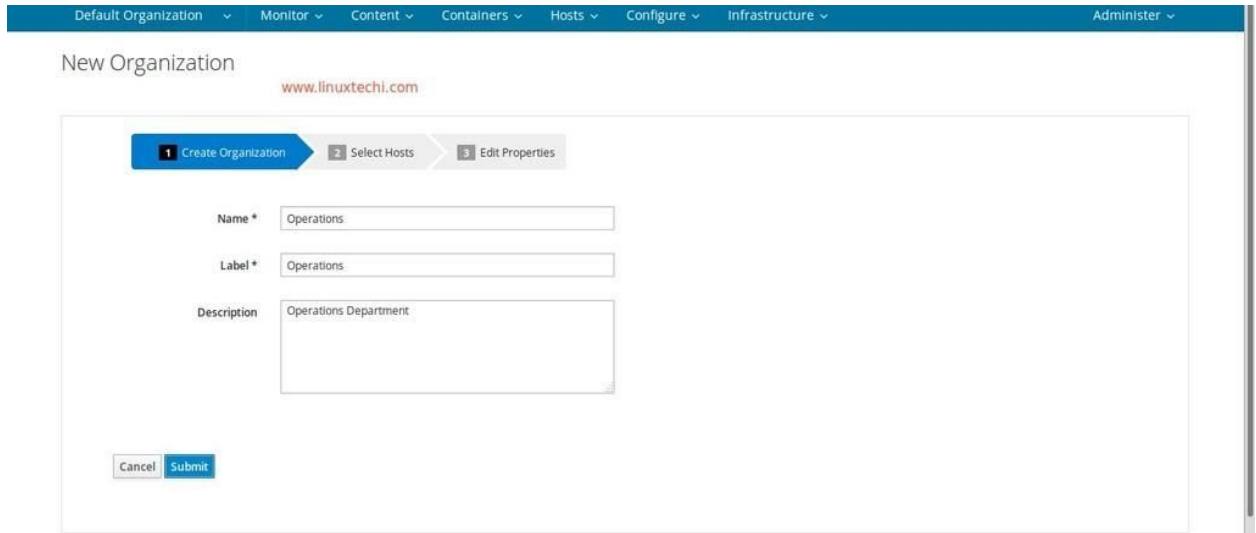
Here we need to enable certain configurations which are : **4,6,7,8,15,16,17,41**

10. Now to access the Katello dashboard, we need to open our browser and visit the url which we chose as our hostname which is: <https://katello.server.com>



Use the user/pass provided when we installed Katello using the foreman-installer.

11. Login and choose the tab called “Default Organization” and click on Manage Organizations. Once there, choose to create a new organization. Here we will name our organization as “Operations”.



Default Organization ▾ Monitor ▾ Content ▾ Containers ▾ Hosts ▾ Configure ▾ Infrastructure ▾ Administer ▾

New Organization www.linuxtechi.com

1 Create Organization 2 Select Hosts 3 Edit Properties

Name * Operations

Label * Operations

Description Operations Department

Cancel Submit

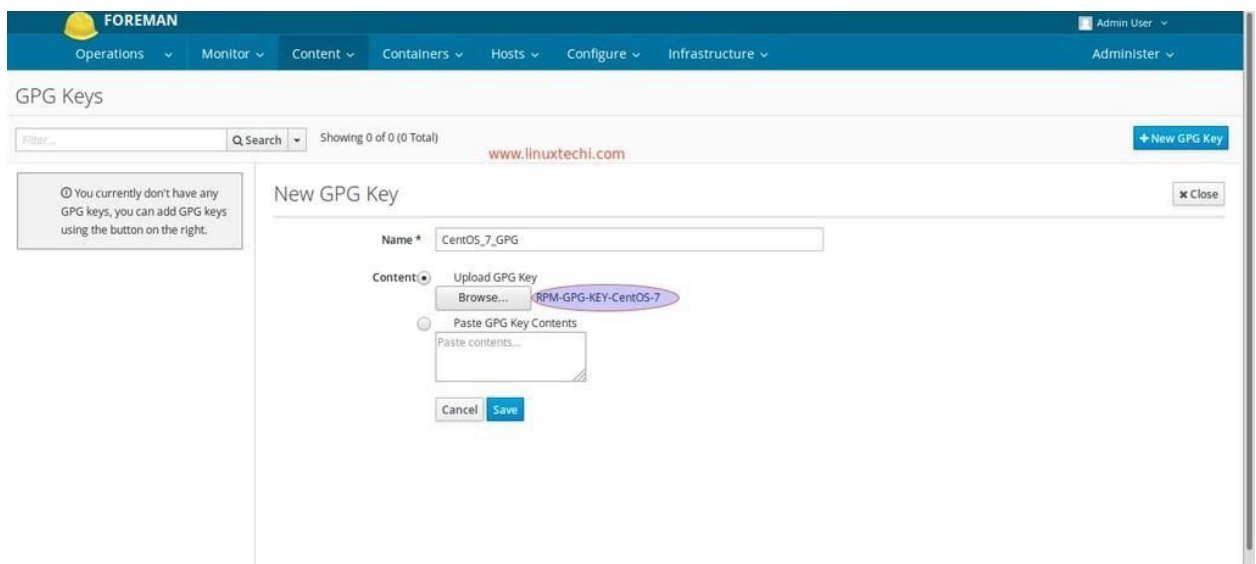
Click on Submit. In the next Window click on the ‘**Proceed to Edit**’ option as this point of time we don’t have any hosts. Finally click on **Submit** on the next window.

Now onward whatever we do in dashboard first make sure we are using ‘**Operations**’ Organization. So Go to Organization Tab and Select ‘**Operations**’

12. Let’s first create the GPG keys for CentOS 7 yum repositories. For that we need to download the key using the command:

```
wget http://mirror.centos.org/centos/RPM-GPG-KEY-CentOS-7
```

Now in the dashboard , go to **Content -> Content Credentials -> Create Content Credentials**. Specify the Key Name, in my case I’m putting it as ‘**RPM-GPG-KEY-CentOS-7**’ and then upload the above downloaded CentOS 7 RPM key.



FOREMAN Admin User ▾

Operations ▾ Monitor ▾ Content ▾ Containers ▾ Hosts ▾ Configure ▾ Infrastructure ▾ Administer ▾

GPG Keys

Filter... Search Showing 0 of 0 (0 Total) www.linuxtechi.com + New GPG Key

ⓘ You currently don't have any GPG keys, you can add GPG keys using the button on the right.

New GPG Key Close

Name * CentOS_7_GPG

Content: Upload GPG Key

Browse... RPM-GPG-KEY-CentOS-7

Paste GPG Key Contents

Paste contents...

Cancel Save

13. Now here is a step-by-step plan that we'll be performing with a single script

- o Step 1: create a product.*
- o Step 2: create RPM repositories.*
- o Step 3: sync RPM repositories.*
- o Step 4: create a content view.*
- o Step 5: add repositories to the content view.*
- o Step 6: create a lifecycle environment.*
- o Step 7: publish a content view.*
- o Step 8: promote the content view version to lifecycle environment.*
- o Step 9: create an activation key.*
- o Step 10: add subscription to the activation key.*

To do this , create a file called **configs.sh** and copy the following commands onto it:

```
#TO CREATE A PRODUCT
hammer product create --name "el7_repos" --description "Various repositories to
use with CentOS 7" --organization "Operations"
#TO ADD CENTOS 7 BASE REPO
hammer repository create --product "el7_repos" --name "base_x86_64" --label
"base_x86_64" --content-type "yum" --download-policy "on_demand" --gpg-key
"RPM-GPG-KEY-CentOS-7" --url "http://mirror.centos.org/centos/7/os/x86_64/"
--mirror-on-sync "no" --organization "Operations"
#TO ADD CENTOS 7 EXTRA REPOS
hammer repository create --product "el7_repos" --name "extras_x86_64" --label
"extras_x86_64" --content-type "yum" --download-policy "on_demand"
--gpg-key "RPM-GPG-KEY-CentOS-7" --url
"http://mirror.centos.org/centos/7/extras/x86_64/" --mirror-on-sync "no"
--organization "Operations"
#TO SYNC FIRST 2 REPOS
for i in $(seq 1 2); do hammer repository synchronize --product "el7_repos" --id "$i";
done
#CREATE A CONTENT VIEW WITH BOTH THE REPOS
hammer content-view create --name "el7_content" --description "Content view for
CentOS 7" --organization "Operations"
hammer content-view add-repository --name "el7_content" --product "el7_repos"
--organization "Operations" --repository-id "1";
hammer content-view add-repository --name "el7_content" --product "el7_repos"
--organization "Operations" --repository-id "2";
# CREATE A LIFECYCLE ENVIRONMENT CALLED "STABLE"
hammer lifecycle-environment create --name "stable" --label "stable" --prior
"Library" --organization "Operations"
# WE NEED TO PUBLISHED A CONTENT VIEW IN ORDER TO LOCK ITS
CONTENT (RPM REPOSITORIES) IN PLACE
hammer content-view publish --name "el7_content" --organization "Operations"
--description "Publishing repositories"
# PROMOTE VERSION TO LIFECYCLE ENVIRONMENT
```

```
hammer content-view version promote --content-view "el7_content" --organization
"Operations" --version "1.0" --to-lifecycle-environment "stable" --force
# CREATE AN ACTIVATION KEY CALLED "EL7-KEY"
hammer activation-key create --name "el7-key" --description "Key to use with
CentOS7" --lifecycle-environment "stable" --content-view "el7_content"
--unlimited-hosts --organization "Operations"
# ADD SUBSCRIPTION TO ACTIVATION KEY
hammer activation-key add-subscription --name "el7-key" --organization
"Operations" --quantity "1" --subscription-id "1"
#SCRIPT OVER
```

Now execute the configs.sh using: **bash configs.sh**

Now we are done with the host side configurations and will now move on to the client.

CLIENT SIDE CONFIGURATION

1. First we need to install subscription-manager on the client. For that we'll use

```
yum install subscription-manager
```

2. Now we need to add the katello server's hostname to our **/etc/hosts** file. To do this we will use the command:

```
echo "<server ip> <fqdn> <subdomain>" >> /etc/hosts
```

For example:-

```
echo "192.168.9.15 katello.server.com katello" >> /etc/hosts
```

3. Now install the katello server certificate in our client system.

```
yum -y install http://<IP>/pub/katello-ca-consumer-<fqdn>-1.0-1.noarch.rpm
```

eg:-

```
yum -y install
http://192.168.132.130/pub/katello-ca-consumer-katello.server.com-1.0-1.n
oarch.rpm
```

4. To connect to the client, we will use subscriber agent along with the activation key and organization

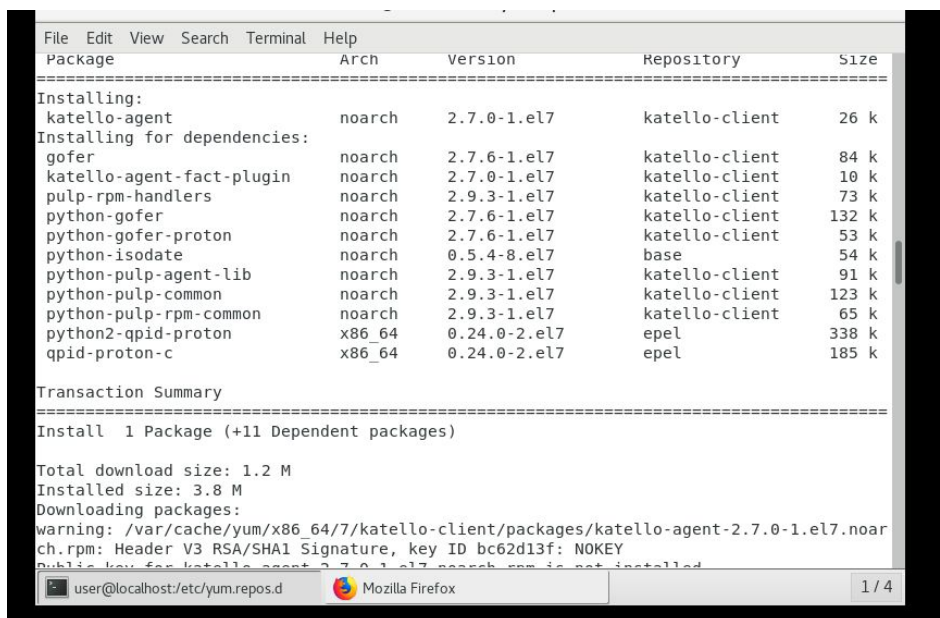
```
subscription-manager register --org="Operations"
--activationkey="el7-key"
```


After successful registration, we'll get an output similar to this:

**The system has been registered with ID:
7c0a6c2f-96f8-41b6-85e2-9765e0ec6ddf
No products installed.**

5. To install Katello agent on this client, we need to add certain repositories and then install the agent. Create a file called **katelloagent.sh** and copy the following in it and execute it:

```
yum -y install  
http://fedorapeople.org/groups/katello/releases/yum/3.2/client/el7/x86_64  
/katello-client-repos-latest.rpm  
yum -y install  
http://dl.fedoraproject.org/pub/epel/epel-release-latest-7.noarch.rpm  
yum -y install katello-agent
```



```
File Edit View Search Terminal Help
Package Arch Version Repository Size
=====
Installing:
katello-agent noarch 2.7.0-1.el7 katello-client 26 k
Installing for dependencies:
gofer noarch 2.7.6-1.el7 katello-client 84 k
katello-agent-fact-plugin noarch 2.7.0-1.el7 katello-client 10 k
pulp-rpm-handlers noarch 2.9.3-1.el7 katello-client 73 k
python-gofer noarch 2.7.6-1.el7 katello-client 132 k
python-gofer-proton noarch 2.7.6-1.el7 katello-client 53 k
python-isodate noarch 0.5.4-8.el7 base 54 k
python-pulp-agent-lib noarch 2.9.3-1.el7 katello-client 91 k
python-pulp-common noarch 2.9.3-1.el7 katello-client 123 k
python-pulp-rpm-common noarch 2.9.3-1.el7 katello-client 65 k
python2-qpid-proton x86_64 0.24.0-2.el7 epel 338 k
qpid-proton-c x86_64 0.24.0-2.el7 epel 185 k

Transaction Summary
=====
Install 1 Package (+11 Dependent packages)

Total download size: 1.2 M
Installed size: 3.8 M
Downloading packages:
warning: /var/cache/yum/x86_64/7/katello-client/packages/katello-agent-2.7.0-1.el7.noarch.rpm: Header V3 RSA/SHA1 Signature, key ID bc62d13f: NOKEY
Public key for katello-agent-2.7.0-1.el7.noarch.rpm is not installed
```

6. Start the Katello agent service and enable it to start at the system boot using the commands:

```
systemctl start goferd  
systemctl enable goferd
```

7. Remove or move default CentOS 7 and katello-agent repository to other location except redhat.repo, which are present inside **/etc/yum.repos.d/**
To do this use the following commands:


```
cd /etc/yum.repos.d/  
mkdir old  
mv CentOS-* epel* katello* old/
```

SERVER SIDE VERIFICATION

Now verify whether the Katello agent is installed on the Host from the Satellite dashboard.
Hosts -> Content Hosts -> Select Host -> check the Katello Agent parameter

To install a package, go to **Hosts -> Content Hosts -> Select Host -> Packages**

Choose Package Action as “**Package Remove**” or “**Package Install**” and then type the name of a package; then click on perform.

Thus we are done with configuring Foreman/Katello successfully.