

vFuel On Demand Use Cloud Execution Environment

OPERATING INSTRUCTION

Copyright

© Ericsson AB 2016. All rights reserved. No part of this document may be reproduced in any form without the written permission of the copyright owner.

Disclaimer

The contents of this document are subject to revision without notice due to continued progress in methodology, design and manufacturing. Ericsson shall have no liability for any error or damage of any kind resulting from the use of this document.

Trademark List

All trademarks mentioned herein are the property of their respective owners. These are shown in the document Trademark Information.



Contents

1	Introduction	1
1.1	Description	1
1.2	Prerequisites	1
2	Procedure	3
2.1	Disable vFuel	3
2.2	Enable vFuel	4





1 Introduction

This document summarizes the procedure for disabling/enabling vFuel on a Single Server environment.

1.1 Description

vFuel manages the CEE infrastructure life cycle and runs in a separate virtual machine on CentOS Linux.

vFuel is responsible for the following:

- Installation of the CEE software
- Updates of the CEE software
- Adding and removing hardware resources

To save resources on a single server environment, vFuel is not migrated to the production environment but it is switched off on the installation host or laptop. vFuel can be reactivated when needed, for example: to add new packages on vCIC or Compute nodes, or to do an update.

1.2 Prerequisites

The Cloud Execution Environment (CEE) is installed by using CEE Installation, as described in *CEE Installation*.





2 Procedure

vCIC and Compute nodes run several services or use configurations related to vFuel, as follows:

- Mcollective service
- Nailgun agent
- DNS configuration

Disabling vFuel stops these services and removes all related configurations. Before any update or modification of the environment, vFuel needs to be enabled again.

2.1 Disable vFuel

To disable vFuel, proceed with the following steps on vFuel as `root` user:

1. Execute the following commands:

```
cd /usr/share/ericsson-orchestration/playbooks  
  
openstack-ansible fuel-disable.yml
```

The expected output shows only `ok` and `changed` statuses and no `failed` at `PLAY RECAP`.

Example:



```
[root@fuel playbooks]# openstack-ansible fuel-disable.yml
Variable files: "-e @/etc/openstack_deploy/user_secrets.yml -e @/etc/openstack_deploy/user_variables.yml "
```

```
PLAY [disable fuel related services] *****

TASK: [fail ] *****
skipping: [cic-1]
skipping: [compute-0-1]

TASK: [stop mcollective service] *****
changed: [compute-0-1]
changed: [cic-1]

TASK: [remove fuel and upstream dns entries] *****
changed: [cic-1] => (item=['/etc/resolv.conf', u'192.168.0.11'])
changed: [compute-0-1] => (item=['/etc/resolv.conf', u'192.168.0.11'])
changed: [cic-1] => (item=['/etc/resolv.conf', u'10.51.40.100'])
changed: [compute-0-1] => (item=['/etc/resolv.conf', u'10.51.40.100'])
changed: [cic-1] => (item=['/etc/resolvconf/resolv.conf.d/original', u'192.168.0.11'])
changed: [compute-0-1] => (item=['/etc/resolvconf/resolv.conf.d/original', u'192.168.0.11'])
changed: [compute-0-1] => (item=['/etc/resolvconf/resolv.conf.d/original', u'10.51.40.100'])
changed: [cic-1] => (item=['/etc/resolvconf/resolv.conf.d/original', u'10.51.40.100'])

TASK: [comment out dnsmasq resolv entries] *****
skipping: [compute-0-1] => (item={'dest': '/etc/resolv.dnsmasq.conf', 'regexp': '^([^#].*)'})
skipping: [compute-0-1] => (item={'dest': '/etc/dnsmasq.d/dns.conf', 'regexp': '^(^server=.)'})
changed: [cic-1] => (item={'dest': '/etc/resolv.dnsmasq.conf', 'regexp': '^([^#].*)'})
changed: [cic-1] => (item={'dest': '/etc/dnsmasq.d/dns.conf', 'regexp': '^(^server=.)'})

TASK: [disable nailgun-agent in cron] *****
changed: [compute-0-1]
changed: [cic-1]

NOTIFIED: [restart dnsmasq] *****
changed: [cic-1]

PLAY RECAP *****
cic-1                : ok=5    changed=5    unreachable=0    failed=0
compute-0-1         : ok=4    changed=3    unreachable=0    failed=0

[root@fuel playbooks]#
```

2. Switch off the vFuel Virtual Machine (VM) by executing the following command:

```
shutdown -h now
```

Using Linux `libvirt` the VM is listed as shut off:

```
root@compute-0-1:~$ virsh list --all
 Id      Name                               State
-----
 -      fuel_master                       shut off
```

vFuel is now disabled. Enabling again is described in Section 2.2 on page 4.

2.2 Enable vFuel

To enable vFuel again, proceed with the following steps. All commands entered on vFuel should be run as the `root` user.

1. Switch on the vFuel VM on the Linux or Windows installation laptop by executing the following commands:



- a. Using Linux `libvirt`, get the `<vm name>` by executing the following command:

```
virsh list --inactive
```

- b. Start the vFuel VM by executing the following command:

```
virsh start <vm name>
```

2. Log on to the restarted vFuel and ensure that the Fuel services have started successfully by executing the following command:

```
fuel-utis check_all
```

Note: Following a system startup vFuel needs time to initialize services.

3. Some more minutes are needed for vFuel to start operating.

To check if vFuel is operational, execute the following command:

```
fuel node
```

Note: It can be necessary to execute the command several times, until it is successful.

The expected output contains a properly formatted table with the node list.

Note: The nodes `online` value will remain `False` until vFuel is re-enabled.

```
[root@fuel tasks]# fuel node
id | status | name | cluster | ip | mac | roles | pending_roles | online | group_id
---|-----|-----|-----|---|----|-----|-----|-----|-----
1 | ready | compute-0-1 | 1 | 192.168.0.20 | 90:55:ae:39:f7:26 | compute, virt | | False | 1
2 | ready | cic-1 | 1 | 192.168.0.22 | 22:03:a8:b3:06:49 | controller | | False | 1
[root@fuel tasks]#
```

Figure 1 Node List Example

4. On vFuel, execute the following commands to enable it:

```
cd /usr/share/ericsson-orchestration/playbooks
```

```
openstack-ansible fuel-enable.yml
```

vFuel is enabled again.

The expected output shows only `ok` and `changed` statuses and no `failed` at `PLAY RECAP`.

Example:



```
[root@fuel playbooks]# openstack-ansible fuel-enable.yml
Variable files: "-e @/etc/openstack_deploy/user_secrets.yml -e @/etc/openstack_deploy/user_variables.yml "
```

PLAY [enable fuel related services] *****

TASK: [fail] *****

skipping: [cic-1]

skipping: [compute-0-1]

TASK: [start mcollective service] *****

changed: [compute-0-1]

changed: [cic-1]

TASK: [add fuel and upstream dns entries] *****

changed: [cic-1] => (item=[/etc/resolv.conf, u'192.168.0.11'])

changed: [compute-0-1] => (item=[/etc/resolv.conf, u'192.168.0.11'])

changed: [cic-1] => (item=[/etc/resolv.conf, u'10.51.40.100'])

changed: [compute-0-1] => (item=[/etc/resolv.conf, u'10.51.40.100'])

changed: [compute-0-1] => (item=[/etc/resolvconf/resolv.conf.d/original, u'192.168.0.11'])

changed: [cic-1] => (item=[/etc/resolvconf/resolv.conf.d/original, u'192.168.0.11'])

changed: [compute-0-1] => (item=[/etc/resolvconf/resolv.conf.d/original, u'10.51.40.100'])

changed: [cic-1] => (item=[/etc/resolvconf/resolv.conf.d/original, u'10.51.40.100'])

TASK: [remove comment from dnsmasq resolv entries] *****

skipping: [compute-0-1] => (item={dest: '/etc/resolv.dnsmasq.conf', 'regexp': '^#(.*?)'})

skipping: [compute-0-1] => (item={dest: '/etc/dnsmasq.d/dns.conf', 'regexp': '^#(server=.*?)'})

changed: [cic-1] => (item={dest: '/etc/resolv.dnsmasq.conf', 'regexp': '^#(.*?)'})

changed: [cic-1] => (item={dest: '/etc/dnsmasq.d/dns.conf', 'regexp': '^#(server=.*?)'})

TASK: [enable nailgun-agent in cron] *****

changed: [compute-0-1]

changed: [cic-1]

NOTIFIED: [restart dnsmasq] *****

changed: [cic-1]

PLAY RECAP *****

cic-1	: ok=5	changed=5	unreachable=0	failed=0
compute-0-1	: ok=4	changed=3	unreachable=0	failed=0

```
[root@fuel playbooks]# █
```