Table of Contents

Red Hat Enterprise Linux 8

Break-fix Practice 1

<div style="background-color:red; color:white">

BREAK-FIX TRAINING
This training requires solving a problem that's simulated in a hands-on lab environment.

</div>

Course: CEE-CB-001

Version: February 2020

Description: This break-fix training simulates a problem booting Red Hat Enterprise Linux 8 (RHEL 8) when systemd daemons are reported as dead. You'll investigate and solve the problem in a hands-on lab environment to complete the training.

How to use this module:

- Look for gray < and > marks on either the bottom or the left and right sides of this pane, depending on the size of the window. Click those to navigate to the previous or next page, respectively.
- Jump to a specific page using the navigation links at the left.
- Play audio for a page using the player at the top of that page. Audio often provides more complete information than the text and graphics alone. A transcript is available from a link on the same page.

In this break-fix:

Type of issue: Booting RHEL 8 when *systemd* daemons are reported as dead.

What you should already know:
This training assumes that you are a Red Hat Certified System Administrator (RHCSA) or have equivalent experience administrating RHEL 7 or RHEL 8.

How break-fix training works:

1. Challenge yourself in the **Break-fix Activity**. It describes a problem scenario that's simulated in the lab environment where you can investigate and solve the problem.

2. Follow up with the **Guided Solution**:

   - If you couldn't solve the problem on your own, read and follow the instructions in the Guided Solution to complete this training.

   - If you were successful solving the problem, read the solution for useful tips you can apply when solving similar problems.

Lab Environment

Successful completion for this training includes hands-on lab activities hosted in a cloud-based lab environment.

PROVISIONING

(1) Log in to the OpenTLC lab portal.

(2) On the far left, mouse over **Services** and select **Catalogs** from the pop-up menu.

(3) Select to expand **All Services** and **Support Labs**.

(4) Select **cee-cb-001** under that list.

(5) Select **Order**.

(6) Complete the application request: read the **Runtime Warning**, check the box to confirm the runtime and expiration dates, and select **Submit**.

IMPORTANT: Expect **up to 20 minutes** to provision your lab environment.

(7) Look for information on how to access your lab environment from one of two places:

- **Information email**
  Look for an email from *Red Hat OPENTLC <noreply@opentlc.com>* with the Subject similar to: *Your Red Hat OPENTLC service provision request for OTLC-LAB_COMPLETED has completed.* This email may arrive before the environment is ready to use. If you don't receive this email within 15 minutes, you can generate a new one from OpenTLC: **Services > Active Services > OTLC-LAB-NAME* > App Control > Status > Submit**

- **The OpenTLC UI**
  Look in the *Custom Attributes* section on the right in OpenTLC: **Services > Active Services > OTLC-LAB-*NAME***

SYSTEM INFO
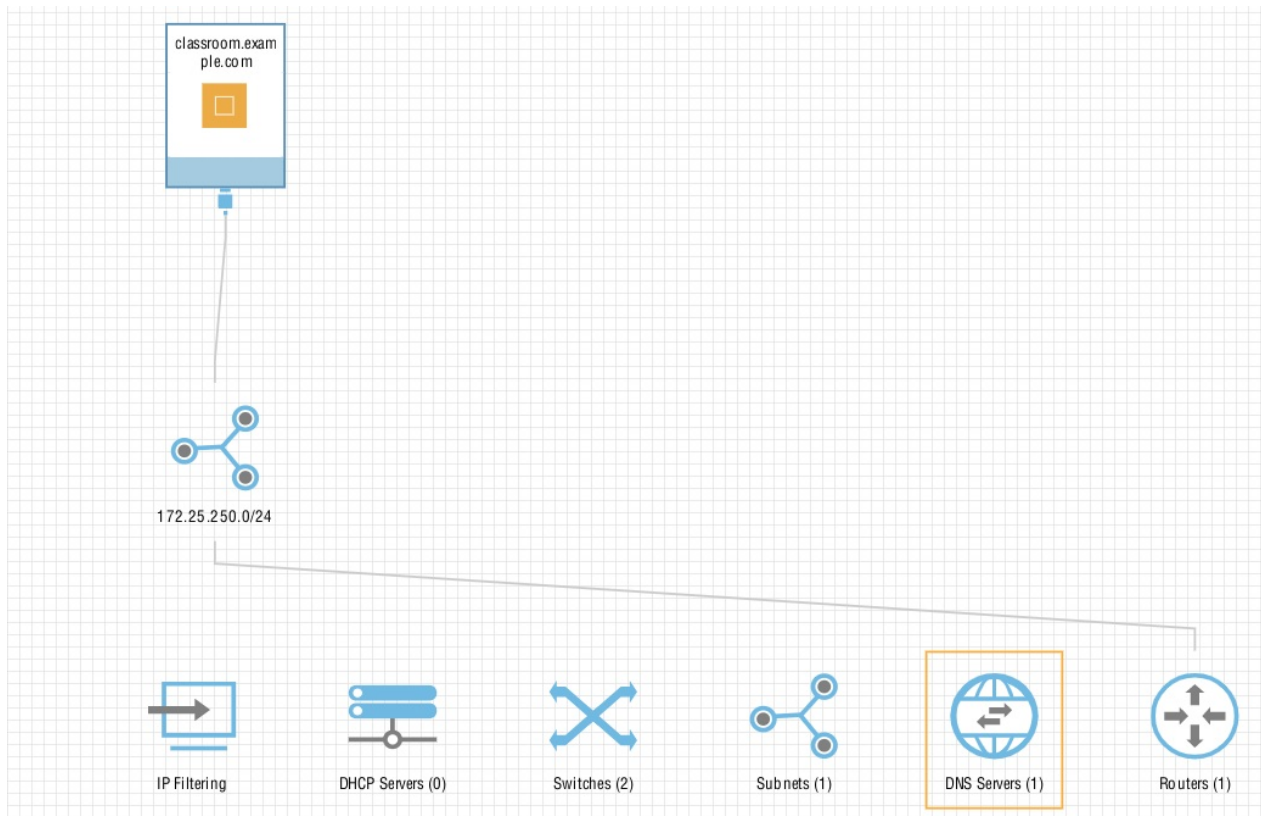
System	IP	Credentials	Description
classroom	172.25.250.254	student/redhat, root/shamrock masher positron tweet	Central SSH access point, provides DNS and DHCP services for other systems

SSH ACCESS

(1) Use the SSH command shown here to access your environment, modifying the command based on the information you received by email:

```
$ ssh flastname-redhat.com@classroom-guid.red.osp.opentlc.com
```

(2) When prompted, log in to your lab environment using one of these options:

- A password set by OpenTLC and provided in the information email.

- An SSH key pair configured as described here: http://www.opentlc.com/ssh.html

```
$ ssh flastname-redhat.com@classroom-guid.red.osp.opentlc.com
The authenticity of host 'classroom-guid.red.osp.opentlc.com (169.47.191.199)' can't be established.
ECDSA key fingerprint is SHA256:v01n4XWXr0lphfGpBiSSvbasmrlQZul2ntS8g0Kbmdk.
Are you sure you want to continue connecting (yes/no)? yes

flastname-redhat.com@classroom-guid.red.osp.opentlc.com's password: <PASSWORD>

[flastname-redhat.com@classroom-guid ~]$ sudo su -
Last login: Thu Oct 24 14:19:41 EDT 2019 from 61.0.147.106 on pts/0
[root@classroom-guid ~]#
```

CONSOLE ACCESS

If you need console access to any of the machines in this environment, follow these steps:

(1) Retrieve the **Master Console** URL from the information email you received on provisioning your lab environment. Look for a line that's similar to this one:

```
Master Console: https://console-redvnc.apps.shared.na.openshift.opentlc.com
```

(2) Open this console URL in your web browser, and select **Log in with OpenShift**.

(3) Enter your OpenTLC username and password at the OpenShift login prompt.

(4) If a dialog appears requiring you to *Authorize Access* for a service account, choose to allow the selected permissions to continue.

(5) Select **Access Console** for a given virtual machine to open a VNC console session with that system.

LOCAL WEB BROWSER ACCESS (HOSTED WEB UI)

(1) Use the same *ssh* command from your local system as for command line access, but add the argument **-CfND 8080**

```
[user1@laptop ~]$ ssh flastname-redhat.com@classroom-guid.red.osp.opentlc.com -CfnND 8080
```

(2) Configure your local web browser to send all web traffic through **localhost:8080**.

```
[user1@laptop ~]$ google-chrome --proxy-server="socks5://127.0.0.1:8080" --host-resolver-rules="MAP * 0.0.0.0 , EXCLUDE localhost" &
```

▼ Show transcript

Successful completion for this training includes hands-on lab activities. Use the information on this page to launch you cloud-based lab environment, locate the URLs and credentials to access that environment, familiarize yourself with the network setup, and use SSH or a local web browser to access lab systems.

Break-fix Activity

This section presents a scenario based on a real customer issue.

1. Read the scenario and success criteria.
2. Follow any instructions provided for setting up your lab environment.
3. Resolve the issue as simulated in your lab environment.
4. After resolving the issue, run the grading script and submit the completion code as prompted.

You must successfully complete this section to receive a passing grade for this module.

If you exhaust your experience and resources, and you're unable to solve the issue on your own, use the Guided Solution to step through one possible solution, with tips for addressing similar issues.
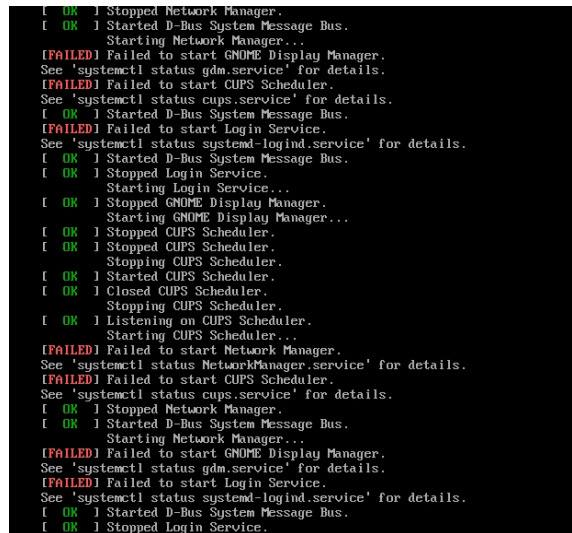
Scenario & success criteria

SCENARIO

Your team handles a Red Hat Enterprise Linux 7 server. Your team has recently upgraded this server from RHEL 7 to RHEL 8.

After the upgrade, the system is no longer booting. You noticed that the system is stuck at *systemd-logind*, and there are a couple of other daemons reporting as failed.

The system boot is hanging with error logs as shown in this screen capture (select image to view at full size):

To address the issue, you've decided to boot the system in rescue mode and investigate further.

SUCCESS CRITERIA

After you investigate and solve the issue, the system should boot and run without errors at the *graphical.target* runlevel, and the root user should be able to connect to the system using SSH.

Setup instructions

After you launch your lab environment, follow these setup instructions for the break-fix scenario:

(1) Get VNC console access to the **classroom** system using the *Console Access* instructions on the Lab Environment page.

(2) Confirm that the system automatically boots to the RHEL 8 DVD that's attached to the system, and that the screen looks like the image here:
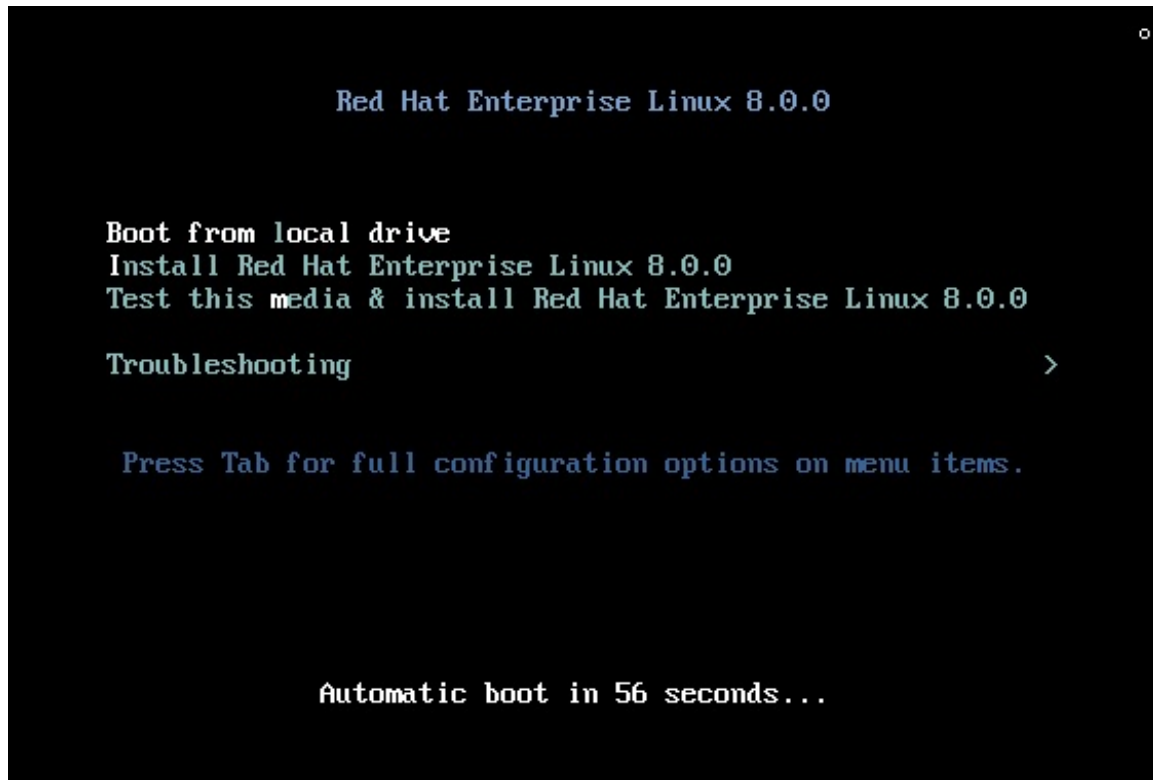
(3) Allow the system to *Boot from local drive* automatically after a few seconds.

(4) Watch the system boot, then confirm that the boot fails to complete.

(5) Confirm that you see the same errors reported in the scenario (time stamps and error messages may differ).

NOTE: This scenario is set in the month of February, so focus your investigation on all existing logs for February.

(7) With these logs available to work with, proceed with investigating and solving the issue to meet the given success criteria.

Check your work

After you have resolved the issue and met the success criteria, run the grading script as shown here to receive a completion code:

```
[root@classroom ~]# cee-cb-001 grade
Initiating cee-cb-001 with option grade
Grading.  Please wait.
Success.
COMPLETION CODE: <check your output for this value>
Completed cee-cb-001 with option grade successfully
```

Submit that completion code here to receive a grade for this training:

COMPLETION CODE provided by *cee-cb-001 grade*: _____

ans: OMIT

Guided Solution

This section presents a guided solution to the Break-fix Activity.

Use this section to:

- Learn one possible path to resolving the customer issue.

- Get some tips for investigating and resolving similar issues.

If you're still stuck after exhausting your expertise and resources on the Break-fix Activity:

- Use this solution to complete the activity.
- Check your work when you finish.

Solution (1 of 4)

(1) Confirm that some daemons are reporting as failed as shown in the screen capture from the scenario. Here's a look at that screen capture, though note some values on your lab screen will be different:

TIP: Allow the system to report as many daemons failed as possible to get first-hand information on what all is broken, and to capture those logs in */var/log.*

Solution (2 of 4)

(2) Boot the system to rescue mode, and search */var/log/messages* for a cause of the boot issue.

For your reference: How to boot a system into rescue mode

(3) In rescue mode, use *chroot* to change to the root partition of your file system:

```
# chroot /mnt/sysimage
```

(4) Study the most recent */var/log/messages* file for clues. Start by analyzing all failed error messages. Here is a sample error log:

```
# grep -i failed /var/log/messages
...
Feb  5 04:53:00 classroom rngd[917]: Failed to init entropy source hwrng
Feb  5 04:53:00 classroom rngd[917]: Failed to init entropy source rdrand
Feb  5 04:53:01 classroom dbus-daemon[911]: Failed to start message bus: Could not get UID and GID for username "dbus"
Feb  5 04:54:30 classroom systemd[1]: Failed to get initial list of names: Connection timed out
Feb  5 04:54:30 classroom systemd[1]: dbus.service: Failed with result 'exit-code'.
Feb  5 04:54:30 classroom systemd[1]: polkit.service: Failed with result 'timeout'.
Feb  5 04:54:30 classroom systemd[1]: Failed to start Authorization Manager.
Feb  5 04:54:30 classroom systemd[1]: Dependency failed for Dynamic System Tuning Daemon.
Feb  5 04:54:30 classroom systemd[1]: tuned.service: Job tuned.service/start failed with result 'dependency'.
Feb  5 04:54:30 classroom dbus-daemon[961]: Failed to start message bus: Could not get UID and GID for username "dbus"
Feb  5 04:56:00 classroom systemd[1]: Failed to get initial list of names: Connection timed out
Feb  5 04:56:00 classroom systemd[1]: udisks2.service: Failed with result 'timeout'.
...
```

(5) In these error logs, notice a reported issue with the *dbus* user indicating that the *dbus* user ID (UID) and group ID (GID) are missing. The logged error should look like this one:

```
...
# grep -i "uid and gid" /var/log/messages
Feb  5 04:53:01 classroom dbus-daemon[911]: Failed to start message bus: Could not get UID and GID for username "dbus"
Feb  5 04:54:30 classroom dbus-daemon[961]: Failed to start message bus: Could not get UID and GID for username "dbus"
Feb  5 04:56:01 classroom dbus-daemon[984]: Failed to start message bus: Could not get UID and GID for username "dbus"
Feb  5 04:56:01 classroom dbus-daemon[991]: Failed to start message bus: Could not get UID and GID for username "dbus"
Feb  5 04:57:31 classroom dbus-daemon[1022]: Failed to start message bus: Could not get UID and GID for username "dbus"
...
```

(6) Given this error message, use *getent* to check whether the *dbus* user and group exists:

```
# getent passwd dbus
# getent group dbus
```

Based on those command outputs, confirm that the *dbus* user and group are missing.

TIP: Most system users like *dbus* are installed in RHEL by pre- and post-install scripts in RPM packages.

Solution (3 of 4)

(7) Use the command shown here to search for all *dbus* related packages on the system, which may have installed the dbus user and group:

```
# rpm -qa | grep dbus
dbus-1.12.8-7.el8.x86_64
python3-dbus-1.2.4-14.el8.x86_64
dbus-libs-1.12.8-7.el8.x86_64
dbus-x11-1.12.8-7.el8.x86_64
python3-pydbus-0.6.0-5.el8.noarch
dbus-common-1.12.8-7.el8.noarch
dleyna-connector-dbus-0.3.0-2.el8.x86_64
dbus-tools-1.12.8-7.el8.x86_64
dbus-daemon-1.12.8-7.el8.x86_64
python3-slip-dbus-0.6.4-11.el8.noarch
dbus-glib-0.110-2.el8.x86_64
```

Out of all those listed packages, what package might have installed dbus user?

(8) Run this script to display all RPM scripts executed by the packages in your "dbus" search:

```
$ for i in `rpm -qa | grep dbus`; do echo $i ; echo '--------------------------------' ;rpm -q --scripts $i ; done | less
```

(9) In the script output, identify which package was responsible for adding the *dbus* user and group.

Confirm that **dbus-daemon** is the package you're looking for based on this output from the script:

```
...
dbus-daemon-1.12.8-7.el8.x86_64
-------------------------------
preinstall scriptlet (using /bin/sh):
# Add the "dbus" user and group
/usr/sbin/groupadd -r -g 81 dbus 2>/dev/null || :
/usr/sbin/useradd -c 'System message bus' -u 81 -g 81 \
    -s /sbin/nologin -r -d '/' dbus 2> /dev/null || :
...
```

Solution (4 of 4)

(10) Recover the missing user and group by directly running those same *useradd* and *groupadd* commands in rescue mode:

```
$ /usr/sbin/groupadd -r -g 81 dbus 2>/dev/null
$ /usr/sbin/useradd -c 'System message bus' -u 81 -g 81 -s /sbin/nologin -r -d '/' dbus 2> /dev/null
```

(11) Exit from rescue mode, and use *Boot with local disk* from the boot disk.

(12) **Check whether you've met the success criteria** to confirm that no other action is required to resolve this issue:

- The system appears to boot normally to the *graphical.target* runlevel.
- The root user can sign in using SSH.

(13) Return to Check your work, and follow the instructions there to run the grading script and submit the completion code.

Resources

Display more verbose boot-related messages during system startup

Boot a system into rescue mode

Feedback

Thank you for taking time to provide your feedback on this training using the form below.

How likely are you to recommend this training module to other associates?

Not at all Likely ○0 ○1 ○2 ○3 ○4 ○5 ○6 ○7 ○8 ○9 ○10 Extremely likely

Based on your learning experience, this module:

☐ Did not cover the stated objectives

☐ Had bug(s) in the learning environment

☐ Had bug(s) in the written content

☐ Was too long

☐ Was too short

☐ Had engaging multimedia interactions

☐ Had no blocking issues

Enter additional comments here...

Submit FeedbackReset