

Version : **2020.01**

Last update: 2020/08/29 14:04

DOE200 - Docker: Administration

Presentation

Objectives: Master the administration of Operating-system-level virtualization with Docker.

Who can benefit: Linux Technicians and Administrators.

Prerequisites: Taken the **DOE100 - Docker: Implementation** course or possess equivalent skills.

Learning technique: Clear, theoretical course content divided into lessons and extensive LABS.

Student Progression: Student progression is monitored both in terms of effective attendance and in terms of comprehension using self-assessment tests.

Duration: 2 days (14 hours).

Prerequisites

Hardware

- A computer running MacOS, Linux, Windows™ or Solaris™ ,
- AZERTY FR or QWERTY US keyboard,
- Minimum 4 GB of RAM,
- Minimum dual-core processor,
- Headphones/Earphones,
- A microphone (optional).

Software

- If Windows™ - Putty and WinSCP,
- Chrome or Firefox web browser.

Internet

- A fast Internet connection (4G minimum) and **no** proxy,
- Unhindered access to the following domains : <https://my-short.link>, <https://ittraining.center>, <https://ittraining.io>, <https://ittraining.institute>, <https://ittraining.support>.

Curriculum

Day #1

- **DOE200 - Docker: Administration** - 1 hour.
 - Prerequisites
 - Hardware
 - Software
 - Internet
 - Using the Infrastructure
 - Connecting to the Cloud Server
 - Linux, MacOS and Windows 10 with a built-in ssh client
 - Windows 7 and Windows 10 without a built-in ssh client
 - Starting the Virtual Machine
 - Connecting to the Virtual Machine
 - Course Curriculum
- **DOE201 - Creating an image registry** - 3 hours.
 - LAB #1 - Installing a private registry
 - Installing Docker
 - Preparation
 - 1.1 - Creating a local registry,
 - 1.2 - Installing a registry on a dedicated server
- **DOE202 - Docker Compose, Docker Machine and Docker Swarm** - 3 hours.
 - LAB #1 - Docker Compose
 - 1.1 - Installation
 - 1.2 - Using docker-compose
 - LAB #2 - Docker Machine
 - 2.1 - Presentation
 - 2.2 - Preparation
 - Docker-CE
 - Mac
 - Linux
 - Windows
 - VirtualBox
 - 2.3 - Installation
 - Mac
 - Linux
 - Windows
 - 2.4 - Creating Docker virtual machines
 - 2.5 - List the Docker virtual machines
 - 2.6 - Obtaining the virtual machine IP address
 - 2.7 - Connecting to the virtual machine
 - LAB #3 - Docker Swarm
 - 3.1 - Presentation
 - 3.2 - Initialising Docker Swarm
 - 3.3 - The Leader
 - 3.4 - Joining the Swarm
 - 3.5 - Getting Swarm information
 - 3.6 - Starting a service
 - 3.7 - Scaling up and scaling down a service
 - 3.8 - Getting node information

- 3.9 - High availability
- 3.10 - Deleting a service

Jour #2

- **DOE203 - Managing a network within Swarm** - 3 hours.
 - Networking and Docker
 - LAB #1 - Preparation
 - 1.1 - Importing the virtual machines
 - 1.2 - Connecting to the virtual machines
 - LAB #2 - Managing an Overlay network
 - 2.1 - Creating an Overlay network
 - 2.2 - Creating a service
 - 2.3 - Moving a service to another Overlay network
 - 2.4 - DNS container discovery
 - 2.5 - Creating a personalised Overlay network
 - LAB #3 - Managing microservices
 - 3.1 - Using a Bridge network and links
 - 3.2 - Using an Overlay network and Docker Swarm
- **DOF204 - Managing Docker's Security** - 3 heures.
 - LAB #1 - Creating a standard user to manage the Docker daemon
 - LAB #2 - The docker-bench-security.sh script
 - LAB #3 - Securing and configuring the Docker host
 - 3.1 - [WARN] 1.2.1 - Ensure a separate partition for containers has been created
 - 3.2 - [WARN] 1.2.3 - Ensure auditing is configured for the Docker daemon
 - LAB #4 - Securing and configuring the Docker daemon
 - 4.1 - [WARN] 2.1 - Ensure network traffic is restricted between containers on the default bridge
 - 4.2 - [WARN] 2.8 - Enable user namespace support
 - 4.3 - [WARN] 2.11 - Ensure that authorization for Docker client commands is enabled
 - 4.4 - [WARN] 2.12 - Ensure centralized and remote logging is configured
 - 4.5 - [WARN] 2.14 - Ensure Userland Proxy is Disabled
 - 4.6 - [WARN] 2.17 - Ensure containers are restricted from acquiring new privileges
 - 4.7 - The /etc/docker/daemon.json file
 - LAB #5 - Securing images and image construction files
 - 5.1 - [WARN] 4.1 - Ensure a user for the container has been created
 - 5.2 - [WARN] 4.5 - Ensure Content trust for Docker is Enabled
 - 5.3 - [WARN] 4.6 - Ensure that HEALTHCHECK instructions have been added to container images
 - LAB #6 - Securing the Container Runtime
 - 6.1 - [WARN] 5.1 - Ensure AppArmor Profile is Enabled
 - 6.2 - [WARN] 5.2 - Ensure SELinux security options are set, if applicable
 - 6.3 - [WARN] 5.10 - Ensure memory usage for container is limited
 - 6.4 - [WARN] 5.11 - Ensure CPU priority is set appropriately on the container
 - 6.5 - [WARN] 5.12 - Ensure the container's root filesystem is mounted as read only
 - 6.6 - [WARN] 5.14 - Ensure 'on-failure' container restart policy is set to '5'
 - 6.7 - [WARN] 5.25 - Ensure the container is restricted from acquiring additional privileges

- 6.8 - [WARN] 5.26 - Ensure container health is checked at runtime
- 6.9 - [WARN] 5.28 - Ensure PIDs cgroup limit is used
- LAB #7 - Docker Content Trust (DCT)
 - 7.1 - The DOCKER_CONTENT_TRUST variable
 - 7.2 - DCT and the docker pull command
 - The disable-content-trust option
 - 7.3 - DCT and the docker push command
 - 7.4 - DCT and the docker build command
 - Creating a second repository
 - Deleting a signature
- LAB #8 - Securing the Docker daemon socket
 - 8.1 - Creating a Certification Authority certificate
 - 8.2 - Creating the Docker daemon's server certificate
 - 8.3 - Creating the client certificate
 - 8.4 - Starting the Docker daemon
 - 8.5 - Configuring the client
- **DOE205 - Course completion** - 1 hour.
 - What's next?
 - Training materials
 - What you need
 - Hardware
 - Software
 - Virtual Machine
 - What we covered
 - Day #1
 - Day #2
 - Resetting the course infrastructure
 - Evaluate the training session
 - Thanks

<html> <DIV ALIGN="CENTER"> Copyright © 2020 Hugh Norris

 Non-contractual document. The curriculum can be changed without notice. </div> </html>