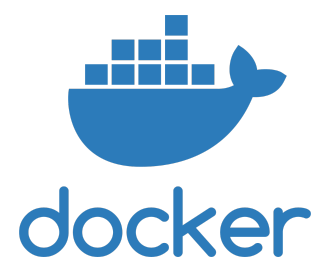


# Docker Certified Associate Study Guide

Version 1.0  
September 2017



## Introduction

This examination is based upon the most critical job activities a *Docker Certified Associate* performs. The skills and knowledge certified by this examination represent a level of expertise where a certified Docker Associate can:

- Run containerized applications from pre-existing images stored in a centralized registry
- Deploy images across the cluster
- Install, maintain, and operate the Docker platform
- Triage issue reports from stakeholders and resolve
- Standup new Docker environments and perform general maintenance and configuration
- Migrate traditional applications to containers
- Configure and troubleshoot Docker engine

The knowledge and skills required at this level should include all of the following objective components:

- 6-months experience with Docker
- Exposure to Docker Enterprise Edition
- Experience with container security
- Experience with at least 1 cloud provider
- Understanding of Docker Best Practices
- Experience with configuration management tools
- Experience with Linux and/or Windows Server

These training courses or equivalency will assist in exam preparation:

- [Docker Fundamentals](#)
- [Docker for Enterprise Developers](#)
- [Docker Security Course](#)

The skills and knowledge measured by this examination are derived from an understanding of the jobs of current Docker users. A team of highly qualified Docker experts defined the test content and wrote the test items.

**Note: This examination blueprint includes weighting, test objectives, and example content. Example topics and concepts are included to clarify the test objectives; they should not be construed as a comprehensive listing of all the content of this examination.**

The following table lists the domains measured by this examination and the extent to which they are represented.

Name of Domain	% of Exam
Orchestration	25%
Image Creation, Management, and Registry	20%
Installation and Configuration	15%
Networking	15%
Security	15%
Storage and Volumes	10%

## Response Limits

The examinee selects, from four (4) or more response options, the option(s) that best completes the statement or answers the question. Distracters or wrong answers are response options that examinees with incomplete knowledge or skill would likely choose, but they are generally plausible responses fitting into the content area defined by the test objective.

Test item formats used in this examination are as follows:

- **Multiple-choice:** The examinee selects one option that best answers the question or completes a statement. The option can be embedded in a graphic where the examinee “points and clicks” on a selection choice to complete the test item.
- **Multiple-response:** The examinee selects more than one option that best answers the question or completes a statement.
- **Sample Directions:** Read the statement or question, and, from the response options, select only the option(s) that represent the most correct or best answer(s) given the information.

## Content Limits

### Domain 1: Orchestration (25% of exam)

Content may include the following:

- Complete the setup of a swarm mode cluster, with managers and worker nodes
- State the differences between running a container vs running a service
- Demonstrate steps to lock a swarm cluster
- Extend the instructions to run individual containers into running services under swarm
- Interpret the output of "docker inspect" commands
- Convert an application deployment into a stack file using a YAML compose file with "docker stack deploy"
- Manipulate a running stack of services
- Increase # of replicas
- Add networks, publish ports
- Mount volumes
- Illustrate running a replicated vs global service
- Identify the steps needed to troubleshoot a service not deploying
- Apply node labels to demonstrate placement of tasks
- Sketch how a Dockerized application communicates with legacy systems
- Paraphrase the importance of quorum in a swarm cluster
- Demonstrate the usage of templates with "docker service create"

### Domain 2: Image Creation, Management, and Registry (20% of exam)

Content may include the following:

- Describe Dockerfile options [add, copy, volumes, expose, entrypoint, etc)
- Show the main parts of a Dockerfile
- Give examples on how to create an efficient image via a Dockerfile
- Use CLI commands such as list, delete, prune, rmi, etc to manage images
- Inspect images and report specific attributes using filter and format
- Demonstrate tagging an image
- Utilize a registry to store an image
- Display layers of a Docker image
- Apply a file to create a Docker image

- Modify an image to a single layer
- Describe how image layers work
- Deploy a registry (not architect)
- Configure a registry
- Log into a registry
- Utilize search in a registry
- Tag an image
- Push an image to a registry
- Sign an image in a registry
- Pull an image from a registry
- Describe how image deletion works
- Delete an image from a registry

### **Domain 3: Installation and Configuration (15% of exam)**

Content may include the following:

- Demonstrate the ability to upgrade the Docker engine
- Complete setup of repo, select a storage driver, and complete installation of Docker engine on multiple platforms
- Configure logging drivers (splunk, journald, etc)
- Setup swarm, configure managers, add nodes, and setup backup schedule
- Create and manager user and teams
- Interpret errors to troubleshoot installation issues without assistance
- Outline the sizing requirements prior to installation
- Understand namespaces, cgroups, and configuration of certificates
- Use certificate-based client-server authentication to ensure a Docker daemon has the rights to access images on a registry
- Consistently repeat steps to deploy Docker engine, UCP, and DTR on AWS and on premises in an HA config
- Complete configuration of backups for UCP and DTR
- Configure the Docker daemon to start on boot

#### Domain 4: Networking (15% of exam)

Content may include the following:

- Create a Docker bridge network for a developer to use for their containers
- Troubleshoot container and engine logs to understand a connectivity issue between containers
- Publish a port so that an application is accessible externally
- Identify which IP and port a container is externally accessible on
- Describe the different types and use cases for the built-in network drivers
- Understand the Container Network Model and how it interfaces with the Docker engine and network and IPAM drivers
- Configure Docker to use external DNS
- Use Docker to load balance HTTP/HTTPs traffic to an application (Configure L7 load balancing with Docker EE)
- Understand and describe the types of traffic that flow between the Docker engine, registry, and UCP controllers
- Deploy a service on a Docker overlay network
- Describe the difference between "host" and "ingress" port publishing mode

#### Domain 5: Security (15% of exam)

Content may include the following:

- Describe the process of signing an image
- Demonstrate that an image passes a security scan
- Enable Docker Content Trust
- Configure RBAC in UCP
- Integrate UCP with LDAP/AD
- Demonstrate creation of UCP client bundles
- Describe default engine security
- Describe swarm default security
- Describe MTLS
- Identity roles
- Describe the difference between UCP workers and managers
- Describe process to use external certificates with UCP and DTR

## Domain 6: Storage and Volumes (10% of exam)

Content may include the following:

- State which graph driver should be used on which OS
- Demonstrate how to configure devicemapper
- Compare object storage to block storage, and explain which one is preferable when available
- Summarize how an application is composed of layers and where those layers reside on the filesystem
- Describe how volumes are used with Docker for persistent storage
- Identify the steps you would take to clean up unused images on a filesystem, also on DTR
- Demonstrate how storage can be used across cluster nodes

## Sample Test Questions

The answers to the sample test questions are at the bottom of the document.

Q1. Which command is used to place an image into a registry?

- A. docker commit
- B. docker tag
- C. docker push
- D. docker images
- E. docker pull

Q2. Which network allows Docker Trusted Registry components running on different nodes to communicate and replicate Docker Trusted Registry data?

- A. dtr-ol
- B. dtr-hosts
- C. dtr-br
- D. dtr-vlan

Q3. Which of the following is not an endpoint exposed by Docker Trusted Registry that can be used to assess the health of a Docker Trusted Registry replica?

- A. /health
- B. /nginx\_status
- C. /api/v0/meta/cluster\_status
- D. /replica\_status

Q4. Which of the following endpoints exposed by Docker Trusted Registry can be used to assess the health of a Docker Trusted Registry replica?

- A. /health
- B. /api/health
- C. /replica\_status
- D. /nginx/health

Q5. One of your developers is trying to push an image to the registry (dtr.example.com). The push fails with the error “denied: requested access to the resource is denied”. What should you verify the user has completed?

- A. `docker login -u <username> -p <password> dtr.example.com`
- B. `docker registry login -u username -p <password> dtr.example.com`
- C. `docker push <username>/<image:tag> dtr.example.com`
- D. `docker images login -u <username> -p <password> dtr.example.com`

Q6. You have been asked to backup the swarm state on a Linux installation. By default, where do Docker manager nodes store the swarm state and manager logs?

- A. `/var/run/docker/swarm`
- B. `/var/lib/docker/swarm`
- C. `/etc/docker/swarm`
- D. `/run/docker/swarm`

Q7. Which of the following will put the Docker engine into debug mode?

- A. `echo '{"debug": true}' > /var/lib/docker/daemon.json ; sudo kill -HUP <pid of dockerd>`
- B. `echo '{"debug": true}' > /etc/docker/config.json ; sudo kill -HUP <pid of dockerd>`
- C. `echo '{"debug": true}' > /var/lib/docker/config.json ; sudo kill -HUP <pid of dockerd>`
- D. `echo '{"debug": true}' > /etc/docker/daemon.json ; sudo kill -HUP <pid of dockerd>`



Q8. How do you deploy 4 new instances of nginx with a single command?

- A. `docker service create --replicas 4 --name myservice nginx`
- B. `docker service create --instances 4 --name myservice nginx`
- C. `docker service scale myservice=4 nginx`
- D. `docker service scale --replicas 4 --name myservice nginx`

Q9. You are using self-signed UCP certs and have a second DNS name that points to your internal controllers. When installing UCP, which flag should you use to add this additional name?

- A. `--internal-server-cert`
- B. `--dns`
- C. `--san`
- D. `--external-server-cert`

## Answers

Q1 = C

Q2 = A

Q3 = D

Q4 = A

Q5 = A

Q6 = B

Q7 = D

Q8 = A

Q9 = C



[success.docker.com/certification](https://success.docker.com/certification)

Questions? Email [certification@docker.com](mailto:certification@docker.com)

Copyright © 2017 Docker, Inc. All Rights Reserved. Docker and the Docker logo are trademarks or registered trademarks of Docker in the United States and other countries. All brand names, product names, or trademarks belong to their respective holders

