



# RadZen DCMQ

---

*DICOM Queue Network Device*

Version 3.1

## Installation Guide

---

# 1 Table of Contents

---

1	Table of Contents .....	i
2	Introduction .....	1
2.1	What is RadZen DCMQ? .....	1
2.2	Where Does It Fit? .....	1
2.3	Platforms .....	1
2.4	Basic Concepts .....	2
2.4.1	Inbound Server .....	2
2.4.2	Outbound Queue .....	2
2.4.3	Routing Rules .....	2
2.4.4	Persistence of Entries .....	2
3	Installation .....	3
3.1	Before You Begin .....	3
3.1.1	Docker Compose .....	3
3.2	Installation .....	3
3.2.1	Windows, Mac and Linux .....	3
3.2.2	NAS Devices .....	4
3.2.3	Basic Configuration .....	4
3.2.3.1	Ports .....	4
3.2.3.2	Container Name .....	4
3.2.3.3	Persistent Data .....	4
3.2.3.4	CORS Origins (Optional) .....	4
3.2.3.5	Time Zone .....	4
3.2.4	(Optional) Launching DCMQ with Docker Compose .....	5
4	The Administration Console .....	6
4.1	Accessing the Console .....	6
4.1.1	System Administrator Login .....	6
5	Support .....	7
5.1	Support Contact .....	7
5.2	Warranty .....	7

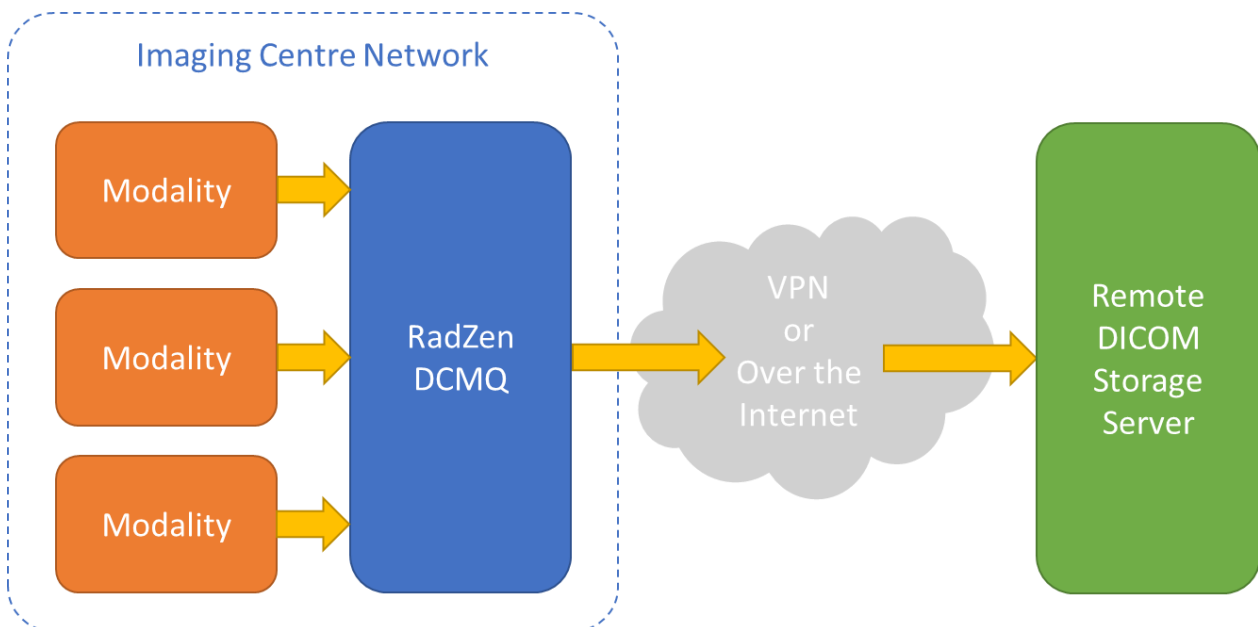
## 2 Introduction

### 2.1 What is RadZen DCMQ?

RadZen DCMQ is a simple, low-cost solution to route and send DICOM data across a network. It is virtual Docker appliance that has the sole function of receiving multiple DICOM instances concurrently, storing them and sending them on to another DICOM server one at a time.

### 2.2 Where Does It Fit?

RadZen DCMQ can be used where DICOM data needs to be sent securely over an Internet connection or VPN, or where the network bandwidth is limited and the sending software is not capable of queuing the DICOM data for transmission.



### 2.3 Platforms

RadZen DCMQ is packaged as a Docker image that can be run on any AMD64 or ARM32 platform that is running Docker.

A description of the Community and Enterprise Editions, and the relevant installation instructions for Docker can be found at:

<https://docs.docker.com/install/overview/>

Some NAS platforms such as Synology and QNAP have built-in Docker support. Make sure the device has at least 2GB of memory installed.

## 2.4 Basic Concepts

The DCMQ is comprised of two main parts: the inbound server and the outbound queue.

### 2.4.1 Inbound Server

The Inbound server “listens” on the network for DICOM transmissions addressed to it by network address and Application Entity (AE) Title.

Each transmission usually corresponds to a single DICOM image being sent from a modality. DICOM images or objects are received by the DCMQ, registered as entries on the queue and then stored on disk.

### 2.4.2 Outbound Queue

The outbound queue is responsible for transmitting the instances on the queue to the configured destination, one at a time. The destination is specified by network address and AE Title.

The outbound queue “wakes up” at regular intervals as specified in the configuration to process entries on the queue. It will first check that the destination is available to receive transmissions before processing the queue.

### 2.4.3 Routing Rules

Routing rules defined by the administrator allow instances that have been received to be routed to one or more Outbound Queues.

### 2.4.4 Persistence of Entries

All queue entries, the internal database and security certificates are stored on Docker volumes. This means that if there were instances on a queue when the Docker container was stopped, they will still be there when the DCMQ container is restarted, and processing will resume.

## 3 Installation

---

### 3.1 Before You Begin

For Windows, Mac and Linux computers, a Docker host must be installed on the computer that DCMQ is to be installed on.

A description of the Community and Enterprise Editions, and the relevant installation instructions can be found at:

<https://docs.docker.com/install/overview/>

If DCMQ is being installed on a NAS (Network Attached Storage) device that supports Docker containers (such as a QNAP TS-x31P2), make sure the device has at least 2GB of memory. Remember to also identify the type of processor running in the device, e.g. ARM, Intel Atom or Intel Celeron, as this will determine whether a variant of the DCMQ docker image will need to be installed.

#### 3.1.1 Docker Compose

Docker Compose provides a convenient way of starting up one or more Docker containers using a docker-compose.yml configuration file.

Docker Compose is installed by default on Community Edition for Mac and Windows but will need to be installed on Linux computers, and for Enterprise Edition. Instructions can be found here:

<https://docs.docker.com/compose/install/>

NAS devices usually come with their own container launch and management apps so Docker Compose is not required in these use cases.

## 3.2 Installation

### 3.2.1 Windows, Mac and Linux

Installation of a Docker container must be performed from a command-line or terminal. The following command will start a basic instance of DCMQ:

```
$ docker run -d -p 9080:5000 -p 11112:11112 --restart=always --name dcmq-1 \
-v dcmq-data:/app/data \
-v dcmq-logs:/app/logs \
-v dcmq-certs:/app/certs \
-e "TZ=Australia/Sydney" \
docker-registry.futurezen.com.au/radzen/dcmq:3.1-amd64
```

This will pull the latest AMD64 platform image for DCMQ and run it in a new container

- detached (-d)
- mapped to internal port 11112 (-p)
- name the container dcmq-1 (--name)
- always restart the container if it fails or if the host restarts (--restart=always)

- create and mount persistent data volumes (`-v`) for data, logs and certs

### 3.2.2 NAS Devices

Refer to the NAS device user manual for launching a Docker container. For devices that have an ARM processor, the ARM 32-bit DCMQ image may be required:

```
docker-registry.futurezen.com.au/radzen/dcmq:3.1-arm32
```

### 3.2.3 Basic Configuration

Refer to the Docker manuals for advanced container configuration, such as attaching multiple virtual network adapters or more advanced use of volumes and bindings.

#### 3.2.3.1 Ports

The DCMQ console runs on port 5000 by default. This can be configured by exposing a different port and mapping it to 5000 using the `-p` option. For example, `-p 18080:5000` exposes port 18080 and maps network traffic to port 5000.

The DCMQ engine runs on port 11112 by default and as with the DCMQ console, a different port may be exposed instead and mapped to port 11112.

#### 3.2.3.2 Container Name

Use the `--name` option to give the container a meaningful name. This name makes it easier to manage the container later.

#### 3.2.3.3 Persistent Data

Docker volumes allow data to be stored between container lifecycles. Using the `-v` option specifies the volume and mount point in the container. The mount points required are as follows:

- `/app/data` : DCMQ database and DICOM instance data
- `/app/logs` : Log files for the web console and DCMQ engine
- `/app/certs` : SSL certificate files uploaded to enable SSL/TLS encryption

#### 3.2.3.4 CORS Origins (Optional)

CORS is a security mechanism to ensure that a web request is from a trusted origin. The web console employs CORS to prevent hacking attacks.

To enable CORS, this must be set to all possible URLs that the web console can be accessed with. Separate each URL with semicolon (;).

For example, if the DCMQ was installed on a docker host with IP address 192.168.100.30 running on port 9080 and had a domain name of "mydcmq.local", the CORS Origins setting would be:

```
-e "CORS_ORIGINS=http://192.168.100.30:9080;http://mydcmq.local:9080"
```

#### 3.2.3.5 Time Zone

By default, the DCMQ container time zone is set to Universal Time Coordinates (UTC) which happens to be the same as Greenwich Mean Time (GMT) + 0.

To create the container with a different time zone, pass the following option on the docker run command:

```
-e "TZ=MyTimeZone"
```

Replace MyTimeZone with the appropriate time zone as defined in the IANA Time Zone Database, e.g. Australia/Sydney. A full list can be found on Wikipedia at:

[https://en.wikipedia.org/wiki/List\\_of\\_tz\\_database\\_time\\_zones](https://en.wikipedia.org/wiki/List_of_tz_database_time_zones)

### 3.2.4 (Optional) Launching DCMQ with Docker Compose

Docker Compose provides a convenient way of launching the DCMQ container with a configuration file.

The docker command in section 3.2.1 can be defined as a docker-compose.yaml file as follows:

```
version: "3"
services:
  dcmq:
    image: docker-registry.futurezen.com.au/radzen/dcmq:3.1-amd64
    ports:
      - "9080:5000"
      - "11112:11112"
    network_mode: "bridge"
    volumes:
      - dcmq-data:/app/data
      - dcmq-logs:/app/logs
      - dcmq-certs:/app/certs
    environment:
      - TZ=Australia/Sydney
volumes:
  dcmq-data:
  dcmq-logs:
  dcmq-certs:
```

To start the DCMQ container, run the command from the same directory as the docker-compose.yaml file:

```
$ docker-compose up -d
```

The advantages of using Docker Compose are:

- no need to script the launch of the container
- the configuration is captured in the docker-compose.yaml file
- multiple DCMQ containers can be launched at once
- Docker Compose can manage container configuration if the docker-compose.yaml configuration is changed

Refer to the Docker Compose documentation for more information:

<https://docs.docker.com/compose/overview/>

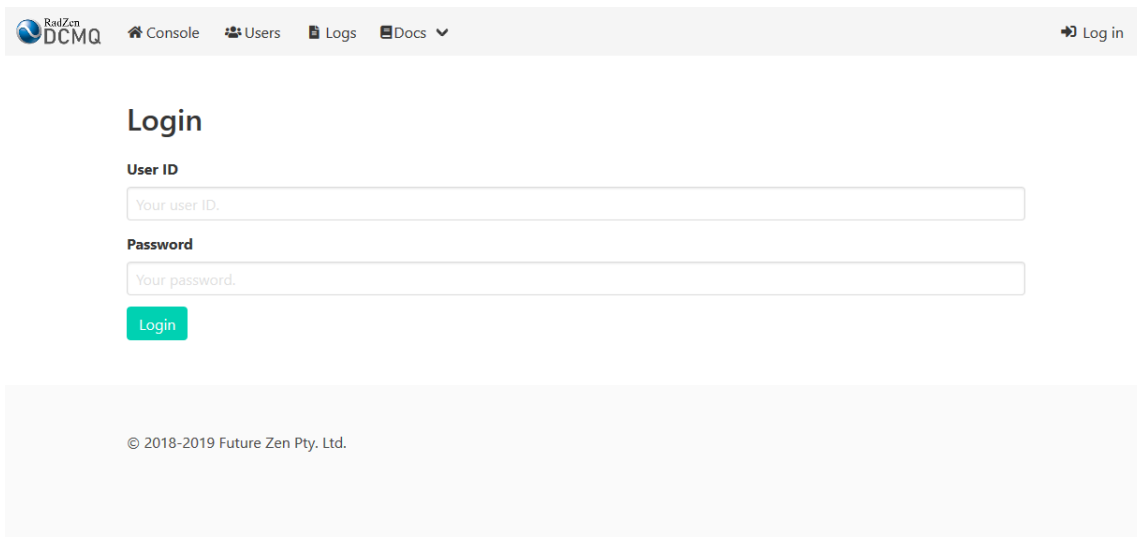
## 4 The Administration Console

### 4.1 Accessing the Console

To access the Administration Console, open a web-browser and visit:

**http://<DCMQ\_address>:<DCMQ\_port>/**

The login page will be displayed, prompting for a username and password.



#### 4.1.1 System Administrator Login

This user profile is created by default. It provides full access to the DCMQ configuration and management of queues. Keep these credentials safe.

<b>Username</b>	admin
<b>Default Password</b>	radzen

**IMPORTANT!** For security, make sure you log in and change your password immediately (see User Manual Section 4.3 Changing Your Password).



## 5 Support

---

### 5.1 Support Contact

Please direct all support requests via our support form at:

<https://www.futurezen.com.au/support>

A response should be received within 1-2 working days.

### 5.2 Warranty

Software defects identified within 12 months of purchasing a license will be rectified free of charge.

All other support requests will be charged according to the fee schedule on the Future Zen website.