

# [YARN-5428] Allow for specifying the docker client configuration directory

[\(1\) Introduction](#)

[\(2\) Issue](#)

[\(3\) Docker Client Configuration Consumers](#)

[\(4\) Proposed Solution](#)

[\(4.1\) Docker Client Configuration Directory YARN Property for Administrators.](#)

[Overview](#)

[Pros](#)

[Cons](#)

[Implementation](#)

[\(4.2\) Distribute Application Owner Supplied Docker Client Configuration with the Application.](#)

[Overview](#)

[Pros](#)

[Cons](#)

[Implementation](#)

[\(5\) Proposed Solution](#)

## (1) Introduction

Docker provides a client server model for the management of containers. Docker allows the user to specify a [docker client configuration directory](#) as part of the docker command line interface. It is common to populate this configuration file with client side configuration properties, such as; registry login credentials, HTTP proxy settings, formatting rules, and user defined aliases. The default value used by the docker client is \$HOME/.docker, config.json is implicit.

## (2) Issue

The client configuration file is most commonly created when a user successfully executes the “docker login” command, so that this interactive command is not necessary for every docker operation. This mechanism works well when interacting with docker on a user’s desktop, but does not scale to meet the needs associated with container orchestration.

Several challenges exist with the interactive nature of docker login in the context of clusters of machines.

- 1) The user may not have OS login access to all cluster members.
- 2) Supplying credentials interactively is not feasible given the various ways that applications can be submitted to YARN.
- 3) The potential for credentials to be stored may warrant additional security to ensure that users cannot read credentials they do not have access to.
- 4) Failure to populate the client configuration can result in application failures if the image cannot be obtained, introducing image management complexity and/or difficult to diagnose failures.

## (3) Docker Client Configuration Consumers

There are at least two classes of consumers for the docker client config; administrators and application owners. Administrators are those users that manage the cluster as a whole. Application owners submit applications that run tasks in docker containers. Each of these user groups are expected to need control over the docker client configuration.

Administrators would be responsible for setting the HTTP proxy settings and global registry credentials.

Application owners would be responsible for setting user and team specific registry credentials. They may also need to set HTTP proxy settings, but this is likely to be rare.

## (4) Proposed Solution

### (4.1) Docker Client Configuration Directory YARN Property for Administrators.

#### Overview

Introduce a new YARN property that allows the administrator to specify the docker client configuration directory.

#### Pros

- 1) Allows administrators to define the docker client configuration in a single location.
- 2) Shifts responsibility of managing the docker client configuration to the administrator versus YARN.

## Cons

1. Centralized management of the docker client configuration does not meet all use cases, including those requiring role base registry access.
2. Adds a dependency for provisioning additional cluster members, which may negatively impact auto scale solutions.

## Implementation

Provide a final configuration property to allow administrators to define the docker client config directory. Disallow users being able to specify the docker client configuration directory. This allows for the administrator case, but not the case where users have their own private registry. This solution puts the responsibility on the administrator and warnings would need to be documented regarding shared credentials. Note this approach follows the [kubernetes](#) model.

## (4.2) Distribute Application Owner Supplied Docker Client Configuration with the Application.

### Overview

Allow users to include an archive that contains the client.json file that can be submitted as part of the application and localized for use with docker.

### Pros

- 1) Meets the use case of allowing application owners to provide docker client credentials.

### Cons

- 1) Puts the burden on all frameworks that leverage YARN to supply the client configuration. Each framework that wants to leverage docker on YARN must now supply this archive or client configuration directory.

## Implementation

Identify the existence of “docker.tar.gz” in the application submission and localize to the user cache. Validate that only .docker/config.json exists, is a file, and can be read. Supply the directory to the docker client configuration directory argument. Note this approach follows the [Marathon](#) model.

## (5) Proposed Solution

Initially, to meet the administrator need, solution 4.1 will be implemented. This allows the administrator to deploy the docker client configuration for use by YARN and eliminate some of the current burden associated with image management. The proposed solutions above are complimentary and can be adopted independently.