



Secure Clustered HA docker Registries with a Binary Repository Manager

White Paper



Introduction

Container technology is not new. It has been around since the year 2000 with [freeBSD jails](#) that had access to the operating system kernel and a few other system resources. More recently, [Docker](#) has emerged as the “King of Containers” with more and more enterprises adopting Docker technology to run applications in data centers, on IT infrastructure and developer laptops alike. But just as component-based development spawned by the open source revolution created challenges in managing components within the enterprise, the ever-increasing number of Docker images used by an organization presents similar challenges:

- › How do you distribute and share images within your organization?
- › How do you manage who can access an image?
- › How do you make it easy to find images?
- › How do you support a variety of policies for managing images?
- › How do you ensure your images are always available for use?

The answer to all of these questions is Artifactory, a Binary Repository Manager that functions as a single access point organizing all of your Docker images. With full support for the Docker Registry API, Artifactory is transparent to the Docker client and can therefore boost your organization’s productivity by removing the overhead connected to managing your applications and images that are developed to run in Docker containers.



Distribute and share images across your organization

To maximize reuse of your codebase, and to ensure your developers work under the same environment, you want an easy way to share images within your team and across your organization.

Using **local repositories**, Artifactory gives you a central location to deploy and store your images - effectively, a private Docker registry that can replace the Docker Hub. When all teams know that any image can be accessed from a single URL, managing access to images between the different teams becomes very easy. But what if you want to share your images with colleagues who are in geographically remote sites of your organization?

To allow sharing between different sites, you can replicate your repositories to another instance of Artifactory which is outside of your local network. Replicated repositories are automatically synchronized with their source periodically, so that your images can be made available to different teams wherever they may be located around the world.

Local Repositories

Local repositories are physical, locally managed repositories that are typically used to deploy internal and external releases as well as development builds. By storing all your images in local repositories, they can be made available from a single access point across your organization from one common URL.

[Learn more >](#)



Security and Access Control

Every organization needs to implement security policies so that developers can only store images in authorized locations, and access images that they are authorized to use.

Artifactory uses a set of permissions you can configure to provide security and access control for Docker images. By defining appropriate users and groups, you can control where developers can access or deploy images, whether they can annotate metadata, or delete images and more. Artifactory also provides control over access to your servers through full integration with the most common access protocols such as LDAP, SAML, Crowd and others. The comprehensive security and access control capabilities in Artifactory help you manage your

development process by ensuring that developers can only access repositories for which they are authorized to. For example, you can ensure that developers can deploy a release candidate to a local QA repository, but only authorized QA staff, who have ensured that the candidate has met the required standard, can move it to the “releases” repository from which production images are pulled.

This level of security and access control that Artifactory provides for local repositories is far greater than that currently offered by the Docker Hub. Artifactory effectively replaces the Docker Hub to let you manage any number of secure and private Docker registries within your organization.



Smart search for images using custom properties

Given the multitude of images in your system, finding something specific can sometimes get quite complex.

Artifactory provides you with flexible search capabilities both through the UI, and using the extensive REST API. You can find images based on any combination of inherent attributes such as name, version, timestamp, **checksum** and more. Artifactory also provides some common built-in searches. For example, you can ask Artifactory for the “latest” version of any image without having to specify a particular build number. Artifactory knows how to compare all the different versions of an image in any of its repositories and provide the latest one available.

But the full power of smart search comes with the flexibility that Artifactory provides you with custom properties that you can assign to your images, and then use in your searches. For example, you could define a property to classify the status of images indicating if they have completed QA or not. Then, when deciding which images

to upload to production, you could make sure that your search only provides images that have been approved by your QA team. Using these capabilities, Artifactory’s flexible smart search lets you search for images using virtually any set of rules relevant to your workflow.

Checksum-based search

Searching for an image by its checksum is a powerful feature supported by Artifactory thanks to its unique checksum-based storage. Even if an image has been renamed, moved or deployed outside of your organization, you can trace it back to the original version. Simply run the image through a checksum tool (both MD5 and SHA1 are supported) and run a “Checksum” search in Artifactory to retrieve the original version.

[Learn more >](#)

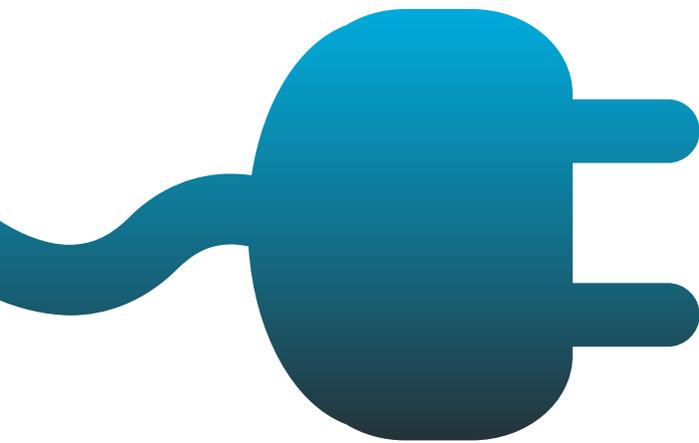
04

User Plugins

While Artifactory provides an extensive set of features to manage images, it's impossible to accommodate all the requirements that different organizations may have. Enter user plugins.

User plugins present a long list of entry points which effectively extend the Artifactory REST API providing a simple way to implement complex behavior. This gives you enormous freedom to support virtually any custom

requirement in your workflow including scheduling tasks, managing security and authentication, deployment, maintenance and cleanup and more. To keep things simple, user plugins are written as Groovy scripts and have a simple DSL to wrap them as closures within the extension points. The plugins can be changed and redeployed on-the-fly, and can even be debugged - all from within your favorite IDE.





Remote Docker Registries

Artifactory is an intermediary between developers and Docker Hub, and handles it as a remote repository. Once an image has been downloaded, Artifactory stores it in a local cache. Upon receiving subsequent requests, Artifactory performs a smart checksum search for the requested image, and if it has already been downloaded, then the locally cached copy is provided. Therefore, images are only downloaded once and is then locally available to all other developers in the organization (thus reducing network traffic). Naturally, this is all transparent to the individual developer. Once Docker Hub is configured to access images through Artifactory, the developer can get on with what she does best and leave it to Artifactory to manage the images.

Docker Hub is an invaluable resource that you need on a regular basis. But since Docker Hub is a remote resource, what do you do if it goes down or if there is an issue with the network. And what happens if “you” are actually a build server? Artifactory fully implements the Docker Registry API specification which allows it to fully proxy any public

or private Docker repository such as Bintray, Docker Hub or other private Docker registries. Artifactory can then treat Docker Hub like any other remote repository, so when an image is requested, Artifactory can provide it from the local cache effectively screening you from any issues with Docker Hub or the network. You will always have access to the images in your system, and your builds won't be held up by issues with the network or Docker Hub.

If we look at network traffic from the point of view of a build server, the benefits are clear. A typical project may need tens if not hundreds of images from Docker Hub. For the server to build these projects, all the images must be downloaded and made available to the server environment and since each image can be 10s of MBs or much more, it will add strain to your network. Downloading all these required images takes a significant amount of time which delays the build process. By caching the images locally, the build process is much quicker and incurs much less networking.



System stability and reliability with Artifactory High Availability

Playing such a central role in the management of images, your Binary Repository Manager can become a mission-critical component of your organization. Any downtime can have severe consequences to your productivity, and you need to ensure developers can access your Docker registries at all times.

Artifactory supports a **High Availability** network configuration with a cluster of 2 or more Artifactory servers on the same Local Area Network. A redundant network architecture means that there is no single-point-of-failure, and your system can continue to operate as long as at least one of the Artifactory nodes is operational. This maximizes your uptime and can take it to levels of up to “five nines” availability. Moreover, your system can accommodate larger load bursts with no compromise to performance. With horizontal server scalability, you can easily increase your capacity to meet any load requirements as your organization grows. Finally, by using an architecture with multiple servers, Artifactory HA

lets you perform most maintenance tasks with no system downtime.

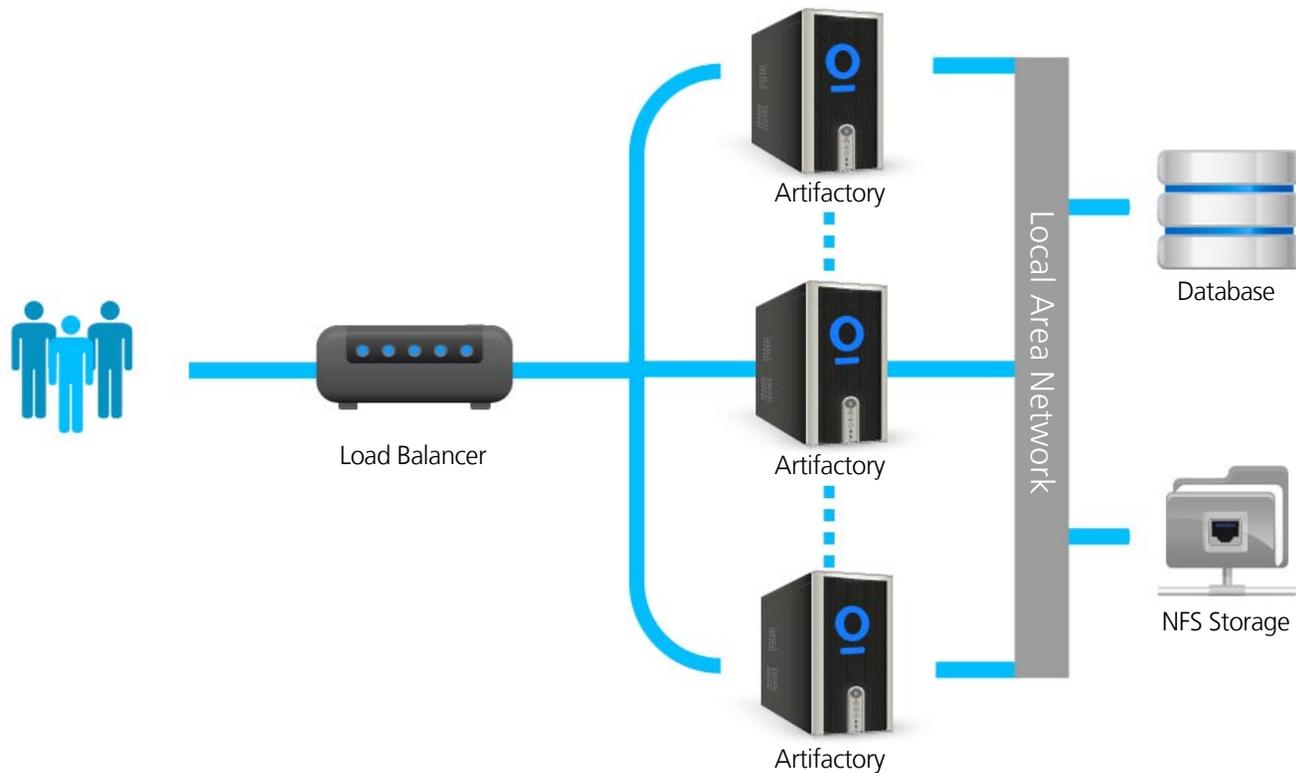
High Availability Systems

Systems that are considered mission-critical to an organization can be deployed in a **High Availability** configuration to increase stability and reliability. This is done by replicating nodes in the system and deploying them as a redundant cluster to remove the complete reliability on any single node. In a High Availability configuration there is no single-point-of-failure. If any specific node goes down, the system continues to operate seamlessly and transparently to its users through the remaining, redundant nodes with no down time or degradation of system performance as a whole.

[Learn more >](#)



System stability and reliability with Artifactory High Availability



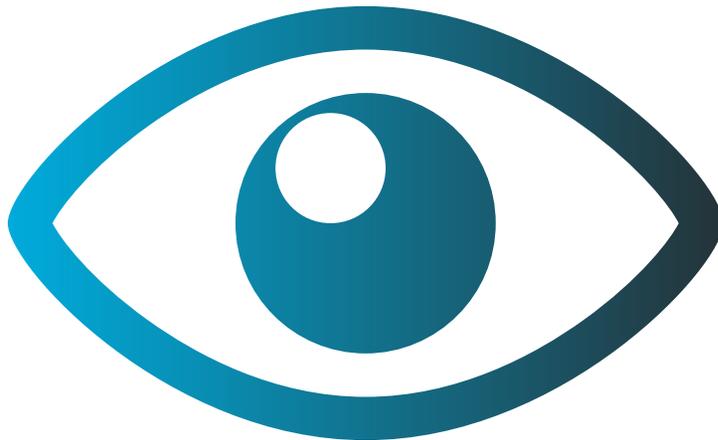


Maintenance and Monitoring

With the growth of automated software delivery systems, the number of images you generate can grow very quickly. Without proper management, your systems can quickly get clogged with old and irrelevant images.

Artifactory keeps your system organized and free of clutter with automatic, timed cleanup processes. With a few

simple settings, you can schedule tasks to clean up old builds and unused images. You can set restrictions on and monitor disk space usage, or define “watches” to receive an alert whenever there is a change to your most critical images. And with an extensive REST API, Artifactory can support virtually any rule-based cleanup protocol you would want to implement in your organization’s scripts.



Summary

This paper has shown how a Binary Repository Manager such as Artifactory can boost productivity of your organization's development and DevOps teams by managing and optimizing access to Docker images. Through local repositories with security and access control, Artifactory can replace Docker Hub and provide secure and private Docker registries where organizations can manage their Docker images. And since Artifactory is agnostic to the binary types that it manages, it can work with Linux packages, RubyGems, NuGet Gallery and virtually any binary package format to support the different platforms used by your organization, and provide central control over all binaries, both when uploading builds and when downloading artifacts.

For more information on how Artifactory can boost your organization's performance, please contact us at info@jfrog.com.