

UNIVERSIDAD TÉCNICA DEL NORTE



**FACULTAD DE INGENIERÍA EN CIENCIAS APLICADAS
CARRERA DE INGENIERÍA EN SISTEMAS COMPUTACIONALES**

ARTÍCULO CIENTÍFICO

TEMA:

**ESTUDIO DEL CONTENEDOR CLOUD DOCKER Y PROPUESTA
DE IMPLEMENTACIÓN PARA LA PLATAFORMA CLOUD FICA.**

AUTOR:

MILTON ANDRÉS SIMBAÑA ALARCÓN

DIRECTOR:

ING. XAVIER MAURICIO REA PEÑAFIEL, MSC.

Ibarra-Ecuador

2016

ESTUDIO DEL CONTENEDOR CLOUD DOCKER Y PROPUESTA DE IMPLEMENTACIÓN PARA LA PLATAFORMA CLOUD FICA

Milton Simbaña
Facultad de Ingeniería en Ciencias Aplicadas
Universidad Técnica del Norte
Ibarra, Ecuador
masimbaniaa@utn.edu.ec

Palabras clave. - Contenedores Cloud, Cloud Computing, Servidores Privados, Virtualización con Docker, Docker, Bitnami.

I. INTRODUCCIÓN

Resumen. - Docker es un proyecto de código abierto que permite crear contenedores que se denominan máquinas virtuales ligeras que serán menos exigentes en cuanto a recursos de hardware y software, una de las características de estos contenedores es brindar portabilidad, ligereza y autosuficiencia a las aplicaciones que se desplieguen utilizando contenedores Cloud.

La característica principal de docker es la de brindar la opción de crear infraestructuras virtuales, esto depende de las aplicaciones que se vaya a desplegar. Bitnami es una herramienta que proporciona a docker instalaciones de software manejables, configurables ofreciéndole soporte a docker, por ello docker y Bitnami trabajan conjuntamente para establecer mejores ambientes de desarrollo permitiendo crear infraestructuras virtuales de forma estructurada.

Esta herramienta tecnológica tiende a extenderse debido a que las organizaciones poseen problemas de aislamiento de aplicaciones en los servidores, esto ocasiona que el servidor solo contenga instalaciones de software no administrables y por ende no permite la optimización y fluidez en un servidor.

Adicionalmente al crear una infraestructura virtual en un servidor real es probable que si se desea añadir una infraestructura diferente a la que se encuentra instalada cause problemas de comunicación entre instalaciones y no funcionen bien.

Este problema se distingue en muchas organizaciones que se dedican a construir infraestructuras virtuales normales, pero el problema se encuentra en que usan distintos servidores aumentando sus costos en equipos y no

tienen una administración de instalaciones.
(RedesZone, 2016)

La solución a este problema que tienden a tener las organizaciones es el uso de la herramienta docker debido a que brinda al usuario la capacidad de administrar una diversidad de infraestructuras virtuales y adicionalmente optimizar la fluidez de las aplicaciones alojadas.



*Figura 1.*Infraestructuras virtuales docker.

Docker no contiene todo un sistema completo, sino únicamente aquellas librerías, archivos y configuraciones necesarias para desplegar las funcionalidades que contenga. Asimismo, docker se encarga de gestionar el contenedor, las aplicaciones e instalaciones realizadas que contenga. (L. T. Semmens, 2016)

Características y funcionalidades de Docker.

- Autogestión de los contenedores.
- Fiabilidad
- Aplicaciones libres de las dependencias instaladas en el sistema anfitrión.
- Capacidad para desplegar varios contenedores.
- Contenedores muy livianos que facilitan su almacenaje, transporte y despliegue.
- Capacidad para desplegar una amplia gama de aplicaciones.
- Compatibilidad Multi-Sistema es decir que podremos desplegar nuestros contenedores en múltiples plataformas.
- Se puede compartir los contenedores a través del repositorio Docker Hub. (Docker, 2016)

Requisitos para la instalación de Docker.

Docker es una herramienta multiplataforma es decir se adapta a varios sistemas operativos ya sea en Linux, Windows o MAC además puede instalarse bajo cualquier plataforma Cloud como OpenStack, Open Nebula, Eucalyptus entre otros. (Docker, 2016)

Tabla 1 *Requisitos mínimos de instalación.*

Windows	Linux	Mac OS
- Para instalación de Docker se necesita contar con Windows 7 x64 en adelante. - Adicionalmente dirigirse a la página oficial de Docker y descargaremos los componentes necesarios para su instalación.	- Para instalar Docker en Linux se ha tomado tres S.O usados frecuentemente como son Centos, Ubuntu, Fedora. - En Centos debe contar con la versión 7x64 en adelante. - En Ubuntu se debe contar con la versión 14.04 x64 en adelante. - En Fedora se debe contar con la versión 22 x64 en adelante.	- Para instalar Docker se debe contar con OS X 10.8 "Mountain Lion". - Adicionalmente dirigirse a la página oficial de Docker y descargaremos componentes necesarios para su instalación.
Requisitos mínimos de hardware		
- Memoria RAM de 1GB. - Espacio en disco de 20 GB.		

Docker también presenta instrucciones personalizadas propias de la herramienta de tal manera que su estructura es comprensible y se adapta de forma rápida a exigencias que tenga el desarrollador o el administrador que se encuentren construyendo infraestructuras virtuales.

Por ello se muestra los comandos más usados por los administradores de la herramienta permitiendo manipular la herramienta según las necesidades del administrador o desarrollador. (Docker, 2016)

Tabla 2 *Comando básicos de Docker.*

Comando básicos de Docker	
docker version	Muestra la versión de docker instalada.
docker info	Muestra las características de docker.
docker images	Muestra imágenes de docker instaladas.
docker ps	Muestra los contenedores en ejecución.
docker pull [imagen]:[Version]	Realiza un petición de descarga de imágenes.
docker run [imagen]:[Version]	Ejecuta una imagen especifica instalada.

Ventajas y desventajas de la herramienta

Docker

Ventajas:

- Capacidad de aislar infraestructuras virtuales con sus respectivas aplicaciones alojadas
- Muestra una gran ventaja en los tamaños de instalacion y la herramienta misma presentando asi ligereza y autosuficiencia en un servidor.
- Bitnami despliega una diversidad de instalaciones compatibles con docker.
- Instrucciones faciles de ejecutar y recordar.
- Actualizaciones rápidas y facil acceso.

Desventajas

Esta herramienta presenta dependencia de conexiones a la red ya sea de internet o intranet en caso de ser un servidor privado siendo asi una desventaja de la herrmienta Docker.

II. MATERIALES Y MÉTODOS



Figura 2. Ejecución de la herramienta Docker.

Después de mencionar las características de la herramienta también se realizó un caso práctico para demostrar que la herramienta funciona y que además brinda las ventajas mencionadas anteriormente.

También se debe mencionar que junto a Docker se usó a Bitnami como complemento para las instalaciones necesarias, con el objetivo de crear una infraestructura virtual, esto permitió que el uso de Docker sea más compacto y evito el desarrollo de un dockerfile que contenga las instrucciones para instalar las herramientas requeridas por la infraestructura a desarrollarse.

La infraestructura virtual a desarrollarse se estructuro de forma que admita el alojamiento de

una aplicación JSF con una base de datos PostgreSQL.

Para ello necesitaremos crear o buscar las instalaciones que permitan el alojamiento mencionado, en este caso se estableció como un servidor web a Tomcat en su versión 8.0.36 (Versión reciente) y finalmente a PostgreSQL en su versión 9.6 (Versión reciente).

Herramienta Bitnami

Bitnami contiene una diversa biblioteca de instalaciones y librerías compatibles con Docker por ello se realizó la instrucción de Git pull para obtener estas imágenes Docker desde Bitnami.

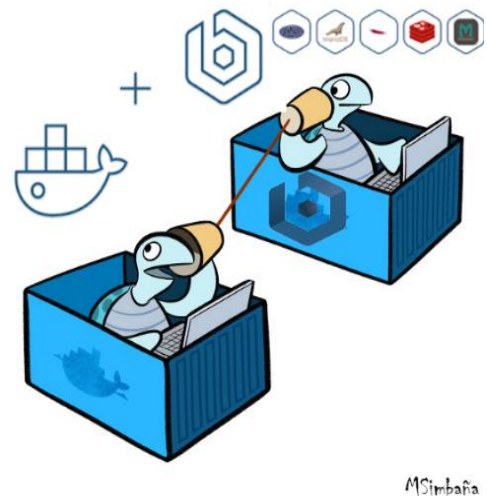


Figura 3. Enlace Docker y Bitnami.

Tabla 3 Tamaño de la herramienta Docker y componentes a usarse.

Nombre de la imagen Virtual Docker	Tamaño del archivo	Versión
Tomcat	426.8 MB	8.0.36
Postgres	259.8 MB	9.2
Docker	2.849 MB	1.11.0

Se debe mencionar que esta infraestructura virtual de prueba se instaló usando la versión de Linux Ubuntu 14.04(x64).



Figura 4. Instalaciones de Docker en Ubuntu.

Después de haber estructurado todos los materiales de construcción de servidor y de infraestructura virtual se puede obtener las imágenes en un servidor local docker usando instrucciones de Git para obtenerlas.

```
docker pull bitnami/tomcat:latest
docker pull bitnami/postgresql:latest
```

Figura 5. Obtención de las imágenes Bitnami para Docker.

Después de haber obtenido las imágenes procedemos a ejecutarlas con ciertos parámetros de

credenciales y puertos por el cual se brindará el servicio ejecutado.

```
docker run -d -p 9090:5432 --name db1 -e POSTGRESQL_USERNAME=masimbaniaa
-e POSTGRESQL_PASSWORD=1003862792 -e POSTGRESQL_DATABASE=juegos_db
bitnami/postgresql:latest
```

Figura 6. Instrucción para ejecutar la imagen de PostgreSQL.

```
docker run -d -p 8090:8080 --name app1 -e TOMCAT_USERNAME=admin
-e TOMCAT_PASSWORD=admin --link db1 bitnami/tomcat:latest
```

Figura 7. Instrucción para ejecutar la imagen de Tomcat.

Al finalizar con las instrucciones, se encontrarán en ejecución Tomcat y postgres de forma que se comuniquen entre ellos permitiendo de esta forma alojar aplicaciones.

III. ANALISIS DE RESULTADOS

Esta herramienta es muy ligera y fácil de instalar ya que solo bastan tres instrucciones para lograrlo, también dentro de sí se tiene imágenes docker que contienen instalaciones independientes con los que se podrá generar los contenedores Docker.

Bitnami es una herramienta que proporcionó las imágenes docker de una forma fácil y segura, además esta herramienta brinda soporte total a todas las imágenes docker que se obtuvieron a través de

Git Hub, es por eso que la herramienta Bitnami en un elemento fundamental para el desarrollo de infraestructuras virtuales.

No solo se realizó la instalación de la herramienta Docker sino también se ha verificado su funcionalidad con el objetivo de proponer el uso de esta herramienta en la plataforma FICA, adicionalmente tras desarrollar la aplicación de prueba para alojarla en una infraestructura virtual, se determinó que esta herramienta es compatible con varios Framework's y bases de datos con lo cual se posibilita la construcción de un centro de datos usando esta tecnología.

El desarrollo de la aplicación permitió demostrar que esta tecnología funciona correctamente y que puede ser usada en grandes entidades ayudando a gestionar diversas aplicaciones mejorando su rendimiento.

Para determinar lo mencionado anteriormente se realizó un test de ejecución usando la herramienta JMeter el cual permitió comparar los tiempos de ejecución entre las instalaciones de docker y las instalaciones tradicionales y se determinó que la

mejor opción a usar es Docker según los resultados obtenidos.

Tabla 4 *Comparación de ejecución de herramientas.*

Herramienta	Tiempo de ejecución Docker	Tiempo de ejecución Tradicional
Tomcat + Aplicación	1minuto :40 segundos	4 minutos :30 segundos
PostgresSQL + BDD	1minuto:10 segundos	2 minuto:50 segundos

IV. CONCLUSIONES

- Docker es una tendencia tecnológica que podrían adoptar muchas entidades con el fin de mejorar el rendimiento en sus aplicaciones y disminuir costes en hardware y software.
- Para este estudio la implementación de docker no presentó limitaciones en su software libre en cuanto a su funcionalidad, de este modo permitió cumplir totalmente con los objetivos planteados.
- Debido a las características que presenta docker, se logró crear una infraestructura virtual que posteriormente permitió el alojamiento de aplicaciones web.
- Usar esta tecnología es sencilla y permite publicar aplicaciones de forma rápida

debido a que docker usa máquinas virtuales ligeras en Linux, además usa librerías y archivos de configuración solamente si son estrictamente necesarios.

- La aplicación de prueba se la pudo realizar sin mayor problema ya que efectivamente Bitnami fue clave para el desarrollo de la infraestructura virtual que requería la aplicación de prueba desarrollada en JSF.
- Esta herramienta es sumamente ligera para instalar y su funcionamiento es realmente óptimo ya que permite realizar instancias de las imágenes docker para crear contenedores de este modo podremos alojar varias aplicaciones con su propia infraestructura virtual.

Methodology, P. t. (25 de 05 de 2016). *Systems2win.com*.
Obtenido de *Systems2win.com*:
<http://www.systems2win.com/LK/lean/PDCA.htm>

RedesZone. (10 de 10 de 2016). *Docker, qué es y cómo funciona la virtualización de contenedores*. Obtenido de <http://www.redeszone.net/2016/02/24/docker-funciona-la-virtualizacion-contenedores/>

Autor

Milton Simbaña. Estudiante de la carrera de Ingeniería en Sistemas Computacionales de la Universidad Técnica del Norte 2016.

Intereses de investigación. Ingeniería de software, desarrollo de aplicaciones, plataformas Cloud, Contenedores Cloud, Normas ISO, Herramientas CRM.



V. REFERENCIAS

Docker. (12 de 10 de 2016). *Welcome to the Docs*. Obtenido de Docker: <https://docs.docker.com/>

Garzás, J. (10 de 10 de 2016). *¿Qué es Docker? ¿Para qué se utiliza? Explicado de forma sencilla*. Obtenido de <http://www.javiergarzas.com/2015/07/que-es-docker-sencillo.html>

L. T. Semmens, R. B. (15 de 10 de 2016). *Integrated Structured Analysis and Formal Specification Techniques*. Obtenido de [http://comjnl.oxfordjournals.org/content/35/6/600.sh](http://comjnl.oxfordjournals.org/content/35/6/600.short)
ort