

# VIRTUALIZACIÓN Y CONTENEDORES

Yofer Nain Catari Cabrera

*Universidad Privada de Tacna \Facultad de Ingenieria \Escuela Profesional de Ingenieria de Sistemas*

## Resumen

En este trabajo se realizará un estudio de las tecnologías de virtualización de contenedores con el fin de implementar y poner en marcha un sistema que permita orquestar el despliegue de aplicaciones sobre un entorno empresarial. Para ello, se realizará en primera instancia un análisis de los sistemas de virtualización más habituales para continuar introduciendo los conceptos y sistemas de virtualización de contenedores. Una vez introducida la parte teórica se analizarán distintas herramientas de virtualización de contenedores centrándonos en la herramienta Docker para la cual se detalla su arquitectura, funcionamiento y proceso de instalación para finalizar con un par de ejemplos prácticos de despliegue de servicios. A continuación, una vez que ya hemos implementado y analizado un sistema de virtualización de contenedores como tecnología necesaria de base, pasamos a estudiar distintas soluciones del mercado para implementar un sistema de orquestación basado en microservicios para el despliegue de aplicaciones de carácter corporativo. Finalizamos con la implantación, instalación y puesta en marcha del sistema estudiado acompañado de unos ejemplos de orquestación usando dos aplicaciones de código abierto que se ven bastante habitualmente en los entornos corporativos actuales para dar soporte a distintas soluciones. .

**Palabras clave:** Virtualización y Contenedores .

## Abstract

In this job we do a study of container virtualization technologies will be carried out in order to implement a system that allows orchestrating the deployment of applications and services in enterprise environments. This will be done in the first instance an analysis of the most common virtualization systems to continue introducing concepts and systems of containerization virtualization. Once the theoretical part is introduced, different container virtualization tools are analyzed, focusing on the Docker tool, which details its architecture, operation and installation process to finish with a couple of practical examples of service deployment. Then, once we have already implemented and analyzed a container virtualization system and all necessary background technology, we started to study different solutions in the market to implement a micro-services based orchestration systems for the deployment of corporate applications. Then finish with implementation, installation and first steps of the studied system accompanied by some examples of orchestration using two open source applications that are quite commonly used in the current corporate environments to support different solutions.

**Keywords:** Virtualization and Containers

## I. INTRODUCCIÓN

La tecnología que ha llegado para complementar y completar la virtualización de servidores es la virtualización de contenedores de aplicaciones. Esta tecnología va un paso mas allá en el paradigma de la virtualización, permitiendo no sólo el salto de virtualizar servidores sino también de virtualizar directamente un contenedor donde se ejecuta una aplicación, permitiendo de este modo una mayor abstracción aislando la componente "lógica de la aplicación" del componente "sistema operativo".

## II. OBJETIVOS

### A. General:

### B. Específicos:

## III. MARCO TEÓRICO

### A. Máquinas Virtuales

### B. Contenedores

### C. Diferencias

## IV. TIPOS DE VIRTUALIZACION

### A. ¿Cuales son las Ventajas de la Virtualización

## B. Contenedores

-

## V. VIRTUALIZACION DE CONTENEDORES

La virtualización basada en contenedores es una aproximación a la virtualización en la cual se ejecuta la capa de virtualización como una aplicación aislada dentro del sistema operativo del equipo anfitrión. En este tipo de sistemas sólo se ejecuta un único núcleo del sistema operativo o Kernel<sup>19</sup>, que es el del sistema operativo anfitrión, y este ayudado por el software de virtualización de contenedores específico es el que se encarga de crear nuevos entornos de ejecución (que podríamos comparar con las máquinas virtuales) y que reciben el nombre de contenedores. Un contenedor es sencillamente un proceso para el sistema operativo que internamente contiene la aplicación que queremos ejecutar y todas las dependencias derivadas de la misma. Empaquetamos una aplicación en una unidad estandarizada para desempeñar un servicio que contiene lo necesario para funcionar como un todo (código, librerías, software, etc), empaquetado bajo la analogía de un contenedor.

### A. Diferencia Entre Virtualizacion Clasica y Virtualizacion de Contenedores

#### B. Tecnologia a Utilizar

#### C. Presedentes a Utilizar

#### D. Ventajas y Desventajas

## VI. CONTENEDORES DOCKER

Además de la propia herramienta principal de Docker para el control y gestión de los contenedores, existe un importante ecosistema de aplicaciones y utilidades integrado en el propia Docker que permite dotar a esta solución de un gran numero de herramientas y funcionalidades adicionales que aportan distintas soluciones de valor. Docker Machine, Docker Swarm y Docker Compose son tres de las más salientables y que combinadas, permiten que los contenedores sean más portables y escalables de tal forma que pueden ser más fácilmente desplegados y administrados en conjunto por lo que combinadas permiten la orquestación de sistemas a gran escala. Así, todo el ecosistema de aplicaciones que aporta la solución Docker van orientadas a dotar de un mayor numero de funcionalidades y herramientas de gestión y automatización para un sistema de aplicaciones embebidas en un contenedor virtual. Podemos considerar que con Docker podemos realizar empaquetados de aplicaciones como si se tratase de una unidad estandarizada y

estanca (contenedor) que incluya todo lo necesario para ejecutar esa aplicación (Código, librerías, bibliotecas del sistema, herramientas, etc) y que puede ser replicado de forma rápida, fiable y sistemática.

### A. Orquesta de Aplicacion

### B. Docker y Otros Container : Mas alla de la virtualizacion

-

### C. Contenedor Docker la Tecnologia de Contenedores a mano

### D. Ventajas de la Tecnologia de Contenedores

### E. ¿Son Seguros los Contenedores?

## VII. HYPERVISORES BASE METAL

### A. Tipos de Hypervisores

### B. Ventajas y Desventajas

### C. Productos Comerciales

## VIII. CONCLUSIONES

- Como hemos podido comprobar a lo largo de es proyecto, partimos de los requisitos base de una empresa que tenia un centro de datos con servidores propios, con personal para mantenerlo y que pretendía encontrar y validar un sistema de código abierto que les permitiese implantar un sistema de orquestación de aplicaciones. Como ha quedado patente, se han conseguido todos los objetivos iniciales que se planteaban al inicio del mismo. Para ello, en primer lugar se ha realizado un análisis de las distintas tecnologías de virtualización clásica y de virtualización de contenedores, comparándolas y analizando cual era la solución por la que debíamos optar para la orquestación de aplicaciones. Una vez visto que la solución pasaba por el uso de la virtualización de contenedores, se estudio la tecnología subyacente que la hacia posible, para poder entender su funcionamiento y posibles limitaciones. A continuación se analizaron varias alternativas de aplicaciones de virtualización y orquestación, con la premisa inicial del uso de software de código abierto, que nos permitieran acometer una solución, hasta decantarnos finalmente por la herramienta Docker y su ecosistema de aplicaciones como solución para conseguir los

objetivos planteados. Al mismo tiempo que se selecciono la aplicación, se planteo una arquitectura complementaria para el despliegue de la solución Docker. Y llegados a este punto, se realizo un análisis exhaustivo del funcionamiento de todo el ecosistema de aplicaciones de Docker, analizando y estudiando su funcionamiento y documentándolo con ejemplos prácticos. Considerando que existen dos tipos de orquestaciones, la orquestación estática y la orquestación dinámica, analizamos en profundidad dos de las herramientas del ecosistema Docker que permiten dar solución a los objetivos que nos planteábamos: Comprobamos como Docker Compose nos permitía orquestar servicios de manera estática y como Docker Swarm nos permitía lo mismo pero esta vez para orques-

tados dinámicamente con lo cual, la solución propuesta cumple todas la condiciones iniciales que nos marcamos. Finalmente se realizaron dos ejemplos prácticos completos de cada una de las distintas orquestaciones, cada una con su herramienta particular (Docker Compose y Docker Swarm) pudiendo verificar que la elección de la solución Docker nos permitió dar una solución completa que incluso dotaba a nuestro proyecto de posibilidades de escalado y orquestación con una sencillez más allá de las esperadas personalmente al comienzo del mismo.

## IX. WEBGRAFIA