

# docker

## Introduzione Teorica

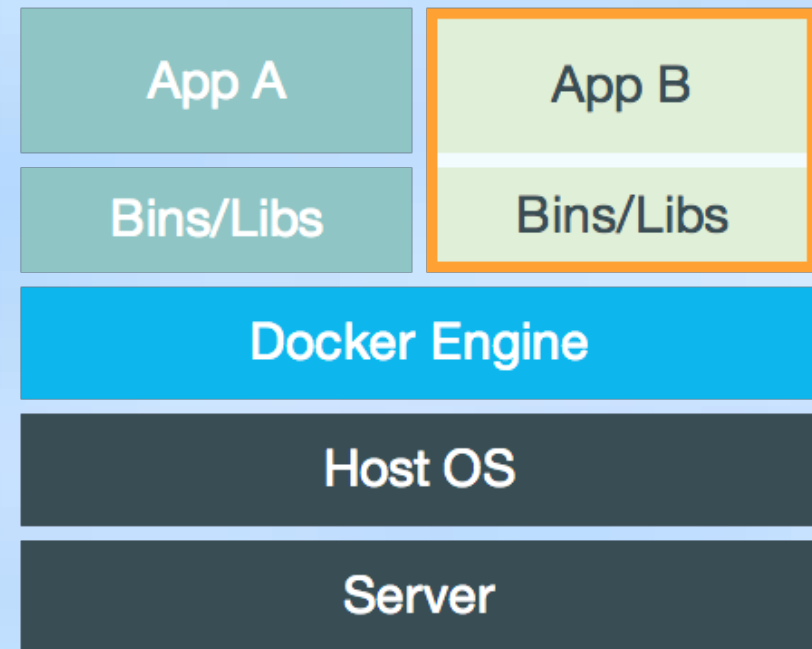
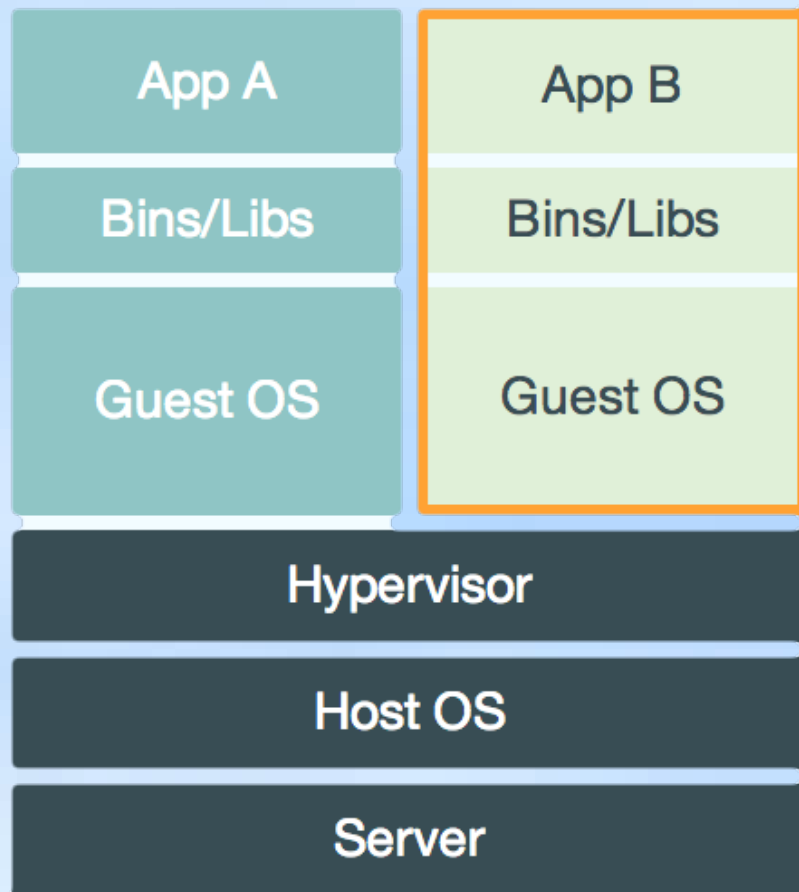
# Cos'è Docker?

Docker è un software per la gestione di virtualizzazione di sistemi basati su container, sviluppato dalla **Docker inc.** disponibile su Linux.

# Virtualizzazione basata su container?

Un container, a differenza della classica virtualizzazione, condivide il kernel della macchina Host e si differenzia solamente per le librerie e i programmi che usa.

# Differenze fra virtualizzazione classica e basata su container



# Virtualizzazione basata su container, come?

Nonostante Docker sia l'argomento di questa presentazione, il merito per il sua potenza e versatilità vanno a due componenti fondamentali del kernel Linux:

- Namespaces
- Cgroups

# Namespaces

Presenti dalla versione 2.4.19 del kernel Linux e sviluppati da Al Viro nel 2002, permettono di dividere un gruppo di processi in modo da avere delle risorse separate dal resto del sistema.

# Cgroups

Presente fin dalla versione 2.6.24 del kernel Linux e sviluppata principalmente da **Paul Menage** e **Rohit Seth** e altri sviluppatori di **Google**.

Grazie ad esso si possono gestire le risorse per un determinato gruppo di processi, limitandone se necessario, anche la potenza hardware.

# Libcontainer

Nonostante la potenza dei **Namespace** e di **Cgroup** creare e controllare container risulterebbe comunque complicato.

Per automatizzare il tutto, il team di **Docker** ha creato **libcontainer**, libreria che crea un livello di astrazione all'accoppiata **Namespace + Cgroup**. Inizialmente scritta in **Go** ma in seguito convertita in altri linguaggi più usati.

Viene ormai riconosciuta come lo standard per la virtualizzazione.



# Contenuto di un Container

- Librerie
- Demoni
- Applicazioni di sistema
- Altre applicazioni

# Docker in pratica

Docker consiste in due programmi, un demone e un client.

- **Demone:** Gestisce i processi dei container attivi
- **Client:** Permette all'utente di creare e gestire i container.

# Docker è Unico?

No, Docker non è l'unica soluzione nel campo di virtualizzazione su Linux, e soprattutto esistono soluzioni simili per altri sistemi operativi.

Ad esempio FreeBSDjail per FreeBSD o sysjail per NetBSD o OpenBSD, oppure Lxc per Linux.

# Perché Docker?

- Semplicità nella creazione, nel controllo e nella gestione dei container.
- La vasta community che condivide i container creati.

# Il Go(lang)

Il Go è un linguaggio sviluppato presso Google da un team costituito da ex sviluppatori dei **Bell Labs**.

Pensato come una rivisitazione del **C** in chiave moderna, il **Go** vanta caratteristiche come il **Garbage Collection** e tempi di compilazione brevissimi.

Docker è stato scritto totalmente in **Go**.

# Per Il Resto

- Articolo su Docker per Hacklab X
- Blog Personale – Punk Overflow
- Sito Dell'Hlcs
- Sito Ufficiale Docker
- Sito Docker Hub

# Grazie a tutti!

Questo documento è rilasciato sotto la  
licenza:  
BY-NC-SA