THE RESEARCH AND IMPLEMENTATION OF CLOUD COMPUTING PLATFORM BASED ON DOCKER

DI LIU[1], LIBIN ZHAO[1]

[1]International Center for Wavelet Analysis and Its Applications, School of Information and Software Engineering, University of Electronic Science and Technology of China, Chengdu 611731
E-MAIL: liudi@uestc.edu.cn,libinzhao224@gmail.com

Abstract:

Cloud computing is now leading-edge Internet technology and virtualization technology is one of the important part of cloud computing system .Virtualization, in computing, refers to the act of creating a virtual (rather than actual) version of something, including but not limited to a virtual computer hardware platform, operating system (OS), storage device, or computer network resources. Docker is a new type of virtualization technology. In this paper, through an application instance, we will describe docker's applications and advantages in the cloud Computing.

Keywords:

Docker; Linux Container; Cloud Computing; Virtualization

1. Introduction

Cloud Computer is a rising computing model that evolved from Distributed computing, parallel processing and grid computing. Cloud computing is computing in which large groups of remote servers are networked to allow the centralized data storage, and online access to computer services or resources. In other words, the "cloud" are the Internet resources which exist on the server clusters, including hardware resources (servers, storage, CPU, etc.) and software resources (e.g. application software, integrated development environment, etc.).Local computer only needs to send a request through Internet, the remote will provide the resources you needed and the results will return to the local computer. So, local computer almost need to do nothing, all the processing will execute in the server clusters. Virtualization technology is the key technology of cloud computing technology, through it, deployment of cloud computing system can be implemented. It make cloud computing be a real service.

Virtualizing a computing system's physical resources to achieve improved sharing and utilization has been well established for decades [1].Virtualization began in 1960s mainframe computers as a method of logically dividing the mainframes' resources for different applications. Since then, the meaning of the term has broadened. Now there are a lot of virtualization technology, for example, the RedHat KVM, VmWare ESX, compared with these virtualization technologies, Docker has a lot of their own advantages. Docker is an open-source project that automates the deployment of applications inside software containers, by providing an additional layer of abstraction and automation of operating system – level virtualization on Linux. Through the implementation of a cloud-based analytical platform, this article describes the application of docker in cloud computing and its own advantages.

2. Docker System

Docker is an open platform for developers and sysadmins to build, ship, and run distributed applications. Consisting of Docker Engine, a portable, lightweight runtime and packaging tool, and Docker Hub, a cloud service for sharing applications and automating workflows, Docker enables apps to be quickly assembled from components and eliminates the friction between development, QA, and production environments. As a result, IT can ship faster and run the same app, unchanged, on laptops, data center VMs, and any cloud.

Docker is an open-source implementation of the deployment engine which powers dotCloud, a popular Platform-as-a-Service. It benefits directly from the experience accumulated over several years of large-scale operation and support of hundreds of thousands of applications and databases. It relies on a different sandboxing method known as containerization. Most modern operating system kernels now support the primitives necessary for containerization, including Linux with openvz, vserver and recently LXC. LXC(Linux Container) is a userspace interface for the Linux kernel containment features. Through a powerful API and simple tools, it lets Linux users easily create and manage system or

978-1-4799-7208-1/14/$31.00 ©2014 IEEE

application containers. In that sense, we found that all container-based systems have a near-native performance of CPU, memory, disk and network [2].

Compared with the traditional virtualization technology, docker can fast start, and occupy less resources, as the basis of instance environment here [3].

3. Core technology and Architecture Advantages

3.1. Docker's Architecture Advantages

Docker was superior to the traditional virtualization technology, lies in the difference between their architecture. Fig.1 shows the differents between container and VM. For the ordinary Virtual Machines，each virtualized application includes not only the application - which may be only 10s of MB - and the necessary binaries and libraries, but also an entire guest operating system - which may weigh 10s of GB, but for the docker，the Docker Engine container comprises just the application and its dependencies. It runs as an isolated process in userspace on the host operating system, sharing the kernel with other containers. Thus, it enjoys the resource isolation and allocation benefits of VMs but is much more portable and efficient.

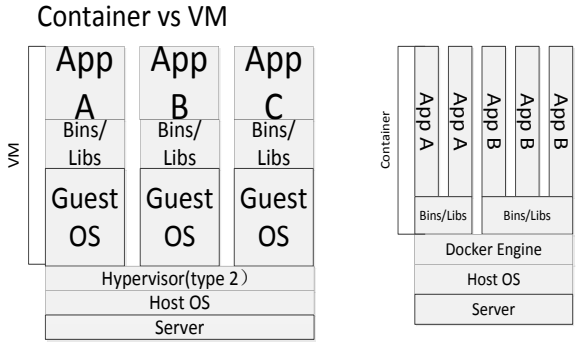

Fig.1 Different of VM and Container

3.2. Core technology

Docker is written in Go and makes use of several Linux kernel features to deliver the functionality we've seen. The core technology of it are Namespaces, Cotrol group and UFS.

Docker takes advantage of a technology called namespaces to provide the isolated workspace we call the container. When you run a container, Docker creates a set of namespaces for that container. This provides a layer of isolation: each aspect of a container runs in its own namespace and does not have access outside it. Some of the namespaces that Docker uses are: The pid namespace(PID: Process ID),The net namespace(NET: Networking),The ipc namespace (IPC: InterProcess Communication),The mnt namespace (MNT: Mount),The uts namespace(UTS: Unix Timesharing System).

Docker also makes use of another technology called cgroups or control groups. A key to running applications in isolation is to have them only use the resources you want. This ensures containers are good multi-tenant citizens on a host. Control groups allow Docker to share available hardware resources to containers and, if required, set up limits and constraints. For example, limiting the memory available to a specific container.

Union file systems, or UnionFS, are file systems that operate by creating layers, making them very lightweight and fast. Docker uses union file systems to provide the building blocks for containers. Docker can make use of several union file system variants including: AUFS, btrfs, vfs, and DeviceMapper.

Docker combines these components into a wrapper we call a container format. The default container format is called libcontainer. Docker also supports traditional Linux containers using LXC. In the future, Docker may support other container formats, for example, by integrating with BSD Jails or Solaris Zones.

4. Docker Instance of Cloud Computing

4.1. The useing of Docker

Docker can make us freely use various resources as us ual, and it's not necessary for us to develop the application under the restricted qualification like using the general PaaS. To use the Docker, we needs to install Docker Engine on the machines. As can be seen from Fig.2, Docker can build images automatically by reading the instructions from a Dockerfile. A Dockerfile is a text document that contains all the commands you would normally execute manually in order to build a Docker image. Then under Docker Engine, the Host 1 OS run the Container that the Dockerfile builded. And Host 1 can push the image of Container A online, that the other hosts can search it and pull it. So the image can be used by other hosts to build Container B,Container C,etc.

So Docker is a lightweight virtualization tool that can package an application and its dependencies in a virtual container that can run on any Linux server. A containerized application can run on premise, on public cloud, on private cloud oron bare metal. And these containers can be transferred from one Docker host to another providing therefore transferability out of the box [4].

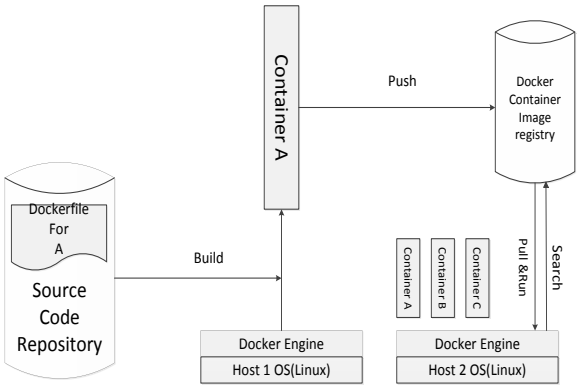

Fig.2 The deployment of container

4.2. Cloud platform architecture

The instance implement a multitask parsing Cloud platform, it is a private PaaS system.

We utilize the Redis to generate the database of our pa rser platform. Redis is an open source, BSD licensed, advanced key-value cache and store. It is often referred to as a data structure server since keys can contain strings, hashes, lists, sets, sorted sets, bitmaps and hyperloglogs.

In the actual production environment, the data that the customers need to parse may be very different. According to the data, the system assigns the different computing tasks to different platforms. As we can see from the Fig.3, the platforms are the container managed by the Docker Engine. The parsing modules and dependent files are configured in the separate Container. Here the front-end application apply regular expressions to match the different kinds of logs, such as XML format.

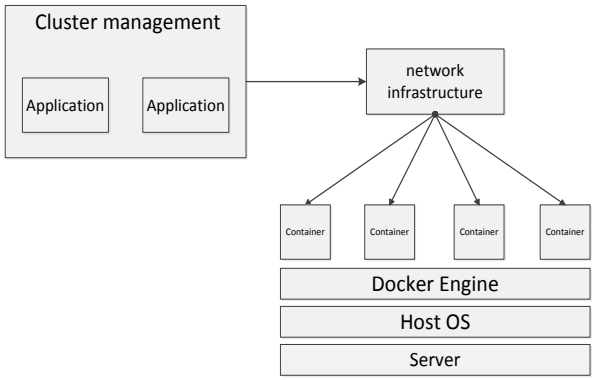

Fig.3 The architecture of platform

4.3. Container Application

The message transfer part between the containers and the host is the pub/sub mechanism in Redis. We add a data volume to a container with the Docker run command. A data volume is a specially-designated directory within one or more containers that bypasses the Union File System to provide several useful features for persistent or shared data. So the log files can be shared between the containers and the host. The front-end application will classify the log files by the forms. As a publisher, the reader will publish the logs to the fixed channels, then the parsing module acquires the logs through subscribing the channel. Finally the logs will be divided into meaningful portions by the parsing module. The core of the module are dynamic libraries builded by the Flex&Bison. The structure of equalizer using CMA algorithm shown in Fig.4.

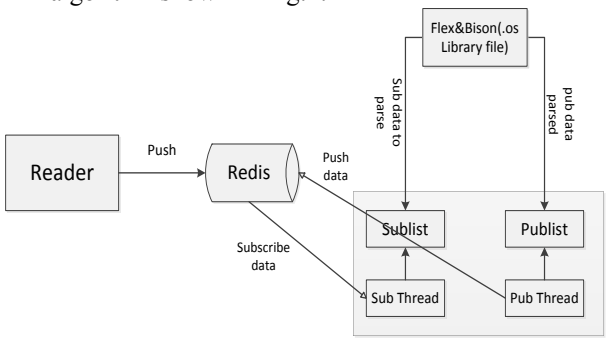

Fig.4 The flow chart of parsing module

4.4 PaaS Platform deployment

The instance adopts two high-performance HP server as the host machines of the Docker containers. In order to better use the Docker, we use Linux as the operating system, every machine deploy six containers. The containers can parser different logs, and the front-end application is running on the host machines.

The data can be shared by the container and host machines. When that container was created, the -P flag was used to automatically map any network ports inside it to a random high port from the range 49153 to 65535 on our Docker host. Network port mappings are not the only way Docker containers can connect to one another. Docker also has a linking system that allows you to link multiple containers together and send connection information from one to another. When containers are linked, information about a source container can be sent to a recipient container. This allows the recipient to see selected data describing aspects of the source container. So we can trasmit the information through the different containers and the host machines. At the same time, we can use scripting tools for the dynamic monitoring of platform.

5. Conclusion

This instance is a cloud parsing platform based on Docker. The platform can solve various parsing tasks according to the requirement of customer. And the containers have many advantages that other virtual products don't have, such as rapid application deployment, portability across machines ,lightweight footprint and minimal overhead, etc.

Consisting of Docker Engine, a portable, lightweight runtime and packaging tool, and Docker Hub, a cloud service for sharing applications and automating workflows, Docker enables apps to be quickly assembled from components and eliminates the friction between development, QA, and production environments. As a result, it can ship faster and run the same app, unchanged, on laptops, data center VMs, and any cloud. This paper expounded the application of Docker in Cloud Computing, especially in the private PaaS platform.

Acknowledgements

This paper is supported by International Center for Wavelet Analysis and Its Applications, School of Information and Software Engineering, University of Electronic Science and Technology of China, Chengdu.

References

[1] R.P. Goldberg, "Survey of Virtual Machine Research," Computer, June 1974, pp.34-45.

[2] Xavier M G, Neves M V, Rossi F D,et al. Performance evaluation of container-based virtualization for high performance computing environments[C]//Parallel, Distributed and Network-Based Processing (PDP), 2013 21st Euromicro International Conference on. IEEE, 2013: 233-240

[3] A Vouk M. Cloud computing – issues, research and implementations [J]. CIT. Journal of Computing and Information Technology, 2008, 16(4): 235-246.

[4] Kratzke N. Lightweight Virtualization Cluster How to Overcome Cloud Vendor Lock-In [J]. Journal of Computer and Communications, 2014, 2(12): 1.