

Proyecto Final Distribuidos

Universidad ICESI

Estudiantes:

Jorge Hernandez - A00317220

Esteban Camacho - A00320168

Daniel Ocampo - A00053980

Curso: Sistemas Operativos

Docente: Daniel Barragán C.

Tema: Despliegue automático de un espejo

Correo: daniel.barragan at correo.icesi.edu.co

Repositorio: <https://github.com/LookIron/sd-project>

Objetivos

- Emplear herramientas de aprovisionamiento automático para la realización de tareas en infraestructura
- Instalar y configurar espejos de sistemas operativos en forma automática para el soporte al aprovisionamiento de máquinas virtuales en clústers de computo
- Especificar y desplegar ambientes conformados por contenedores virtuales para la realización de tareas específicas

Descripción

En ambientes conformados por cantidades considerables de nodos como es el caso de clústers de computo, se requiere contar con mecanismos eficientes para la actualización y despliegue de nueva infraestructura.

Para este proyecto deberá realizar el aprovisionamiento automático de un espejo por medio de las herramientas vistas en clase. Debe cumplir con los siguientes requerimientos:

- El espejo debe ser del sistema operativo ubuntu-xenial
- Debe especificar por medio de un arreglo los paquetes a instalar
- No es necesario que genere llaves nuevas cada vez que despliegue el espejo, puede crearlas previamente y precargarlas en la solución de aprovisionamiento automático a emplear
- Probar por medio del despliegue de uno o varios contenedores que el mirror ha sido correctamente desplegado

Herramientas utilizadas

- Aptly
- Docker
- Http
- Sistema operativo ubuntu-xenial

Procedimiento

2) Automatizacion de comandos necesarios para el aprovisionamiento:

Para realizar el aprovisionamiento de los servicios solicitados (El mirror automatizado) se realizaron algunas configuraciones al archivo de configuración de Aptly (aptly.conf) el cual es ejecutado dentro del contenedor del Mirror, al igual que su posterior instalación a traves del Dockerfile del Mirror. Dichos comandos de la instalación de Aptly y su configuración se automatizaron a traves del Dockerfile. Los comandos automatizados fueron:

```
Reemplazar la llave que viene por defecto de otra de las llaves con las que trabaja Aptly  
ADD /keys/private.asc /keys/private.asc
```

```
#Reemplazar el archivo de configuracion de Aptly      que viene preestablecido
```

```
ADD /conf/aptly.conf /etc/aptly.conf
```

```
#Importamos la llave con privada
```

```
RUN gpg --import /keys/private.asc
```

```
#Para no dejar dicha clave expuesta se elimina
```

```
RUN rm -f /keys/private.asc
```

```
#Importar keyring
```

```
RUN gpg --no-default-keyring --keyring /usr/share/keyrings/ubuntu-archive-keyring.gpg --export | gpg  
--no-default-keyring --keyring trustedkeys.gpg --import
```

```
#Instalamos aptly
```

```
RUN echo deb http://repo.aptly.info/ squeeze main > /etc/apt/sources.list
```

```
RUN apt-key adv --keyserver keys.gnupg.net --recv-keys 9E3E53F19C7DE460
```

```
RUN apt-get update
```

```
RUN apt-get install aptly -y
```

```
#Instalamos las dependencias que requiere Aptly
```

```
RUN echo deb http://co.archive.ubuntu.com/ubuntu/ xenial main restricted >> /etc/apt/sources.list
```

```
RUN apt-get update
```

```
RUN apt-get install xz-utils -y
```

```
RUN apt-get install bzip2 -y
```

```
#Agregamos el mirror a la lista del sistema
```

```
RUN echo deb http://co.archive.ubuntu.com/ubuntu/ xenial universe >> /etc/apt/sources.list
```

```
#Ejecutamos una actualizacion de librerias
```

```
RUN apt-get update
```

```
RUN apt-get install expect -y
```

```
#Seleccionar los paquetes a instalar
```

```
RUN aptly mirror create -architectures=amd64 -filter='Priority (required) | Priority (important) |  
Priority (standard) | postgresql' -filter-with-deps mirror-xenial http://mirror.upb.edu.co/ubuntu/ xenial  
main
```

```
#Actualizar el mirror
```

```
RUN aptly mirror update mirror-xenial
```

3) Configuración del archivo Dockerfile del servidor Nginx:

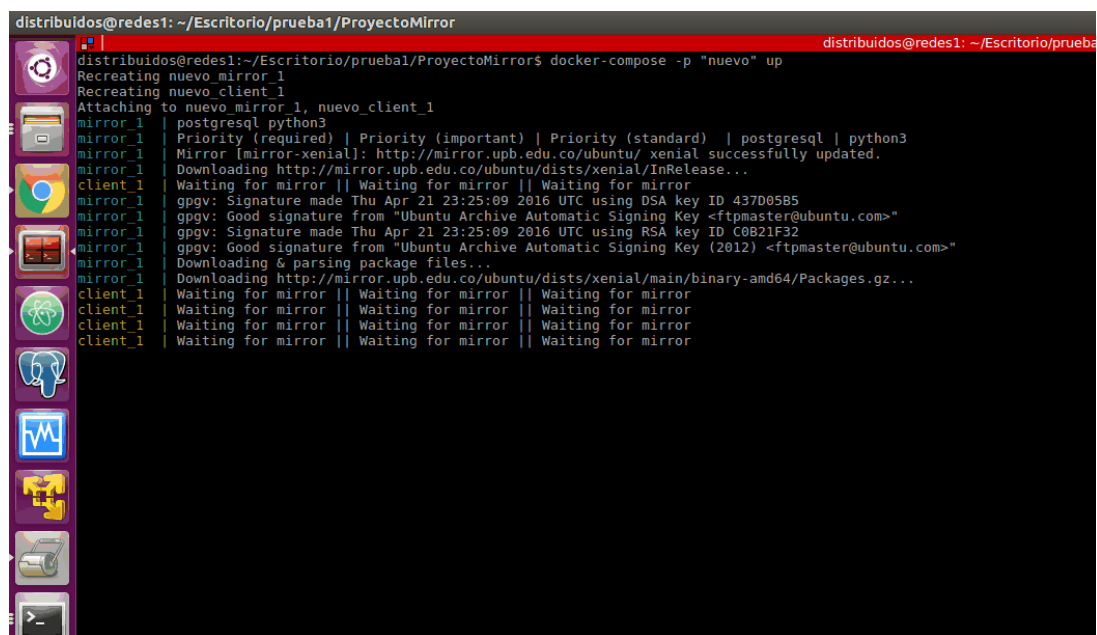
Para la realización del Mirror se utilizaron los archivos anexos en la carpeta llamada **"Solución"** dentro de la cual contiene la carpeta **Mirror** la cual incluye su Dockerfile y una carpeta para el manejo de su clave y otra para las configuraciones de Aptly, el Entrypoint y el publish_snapshot.

4) Archivos para la prueba del servicio:

Para realizar dicha prueba se que el mirror funcionaba y se podian obtener los servicios ofrecidos se configuro un cliente el cual accedia a dicho mirror para descargar algunos archivos, los archivos utilizados para la configuracion de dicho cliente se encuentran en la carpeta llamada **"Solución"** dentro de la cual contiene la carpeta **Cliente**

5) y 6) Verificación de la Solución:

Para verificar dicho funcionamiento nos ubicamos en la carpeta principal donde se encuentra el archivo **docker-compose.yml** y ejecutamos el comando **docker-compose -p "RepositoryName" up** y el empieza a subir el servicio donde se evidencia que al mirror subir sus servicios el cliente puede acceder a este y obtener el servicio. A continuación se muestra dicho funcionamiento:



```
distribuidos@redes1: ~/Escritorio/prueba1/ProyectoMirror
distribuidos@redes1: ~/Escritorio/prueba1/ProyectoMirror$ docker-compose -p "nuevo" up
Recreating nuevo_mirror_1
Recreating nuevo_client_1
Attaching to nuevo_mirror_1, nuevo_client_1
nuevo_mirror_1 | postgresql python3
nuevo_mirror_1 | Priority (required) | Priority (important) | Priority (standard) | postgresql | python3
nuevo_mirror_1 | Mirror [mirror-xenial]: http://mirror.upb.edu.co/ubuntu/ xenial successfully updated.
nuevo_mirror_1 | Downloading http://mirror.upb.edu.co/ubuntu/dists/xenial/InRelease...
nuevo_client_1 | Waiting for mirror || Waiting for mirror || Waiting for mirror
nuevo_mirror_1 | gpgv: Signature made Thu Apr 21 23:25:09 2016 UTC using DSA key ID 437D0585
nuevo_mirror_1 | gpgv: Good signature from "Ubuntu Archive Automatic Signing Key <ftpmaster@ubuntu.com>"
nuevo_mirror_1 | gpgv: Signature made Thu Apr 21 23:25:09 2016 UTC using RSA key ID C0B21F32
nuevo_mirror_1 | gpgv: Good signature from "Ubuntu Archive Automatic Signing Key (2012) <ftpmaster@ubuntu.com>"
nuevo_mirror_1 | Downloading & parsing package files...
nuevo_client_1 | Waiting for mirror || Waiting for mirror || Waiting for mirror
nuevo_client_1 | Waiting for mirror || Waiting for mirror || Waiting for mirror
nuevo_client_1 | Waiting for mirror || Waiting for mirror || Waiting for mirror
nuevo_client_1 | Waiting for mirror || Waiting for mirror || Waiting for mirror
```

7) Problemas encontrados:

- En la realización de dicho proyecto se presento confusión en el manejo de las keys debido a que era necesario manejar una key privada y una publica, donde la publica fue impuesta a los clientes y la privada por

motivos de seguridad se implanto en el mirror. Al inicio no era complejo la generación y el uso de las llaves pero después de algunos ejemplos dados por algunos compañeros, de donde se llego a que la manera mas aproximada era la mencionada anteriormente.