23 DE FEBRERO DE 2022

PRÁCTICA 8

Ansible, Docker, GitHub
y
Script Final

HECHO POR:

Carolina Valencia Tienda
2° G.S ASIR

Contenido

Ansible

Lo primero de todo, será generar unos ficheros nuevos donde concretaremos las conexiones

Fichero host (servidor)

```
[webservers]
192.168.0.35 ansible_ssh_user=carol ansible_ssh_pass=Carola25
```

Fichero ansible.cfg (servidor)

```
[defaults]

inventory=./host
host_key_checking = False
retry_files_ensable = False
~
```

Test DHCP (servidor)

```
---
- hosts: webservers
  tasks:
          - name: run echo command
            command: /bin/echo hello world
```

Instalación DHCP (servidor)

```
---
- hosts: webservers
  become: yes
  tasks:
          - name: INSTALAR DHCP
            apt: name=isc-dhcp-server update_cache=yes state=latest
          - name: REINICIAR DHCP
            service: name=isc-dhcp-server state=restarted
            become: true
```

Desinstalación DHCP

```
---
- hosts: webservers
  become: yes
  tasks:
          - name: DESINSTALAR DHCP
            apt: name=isc-dhcp-server update_cache=yes state=absent
```

Comprobación de conexión con el cliente

```
carola@carola-VirtualBox:~/ansible-apache$ ansible webservers -m ping
192.168.0.35 | SUCCESS => {
    "ansible_facts": {
        "discovered_interpreter_python": "/usr/bin/python3"
    },
    "changed": false,
    "ping": "pong"
}
```

Comprobación conexión con Ansible

```
carola@carola-VirtualBox:~/ansible-apache$ ansible webservers -m command -a "/bi
n/echo Hola Mundo"
192.168.0.35 | CHANGED | rc=0 >>
Hola Mundo
```

Comprobación instalación del servicio (cliente)

```
carol@carol-VirtualBox:~$ sudo service isc-dhcp-server status
● isc-dhcp-server.service - ISC DHCP IPv4 server
     Loaded: loaded (/lib/systemd/system/isc-dhcp-server.service; enabled; vend>
     Active: failed (Result: exit-code) since Sat 2022-02-19 10:26:06 CET; 1min>
       Docs: man:dhcpd(8)
    Process: 3419 ExecStart=/bin/sh -ec      CONFIG_FILE=/etc/dhcp/dhcpd.conf; >
   Main PID: 3419 (code=exited, status=1/FAILURE)

feb 19 10:26:06 carol-VirtualBox dhcpd[3419]:
feb 19 10:26:06 carol-VirtualBox dhcpd[3419]: If you think you have received th>
feb 19 10:26:06 carol-VirtualBox dhcpd[3419]: than a configuration issue please>
feb 19 10:26:06 carol-VirtualBox dhcpd[3419]: bugs on either our web page at ww>
feb 19 10:26:06 carol-VirtualBox dhcpd[3419]: before submitting a bug.  These p>
feb 19 10:26:06 carol-VirtualBox dhcpd[3419]: process and the information we fi>
feb 19 10:26:06 carol-VirtualBox dhcpd[3419]:
feb 19 10:26:06 carol-VirtualBox dhcpd[3419]: exiting.
feb 19 10:26:06 carol-VirtualBox systemd[1]: isc-dhcp-server.service: Main proc>
feb 19 10:26:06 carol-VirtualBox systemd[1]: isc-dhcp-server.service: Failed wi>
```

Al no estar configurado, nos sale como “failed”, pero se ha instalado

Docker

Empezaremos creando un Docker sencillo

```
carola@carola-VirtualBox:~$ docker run hello-world
Unable to find image 'hello-world:latest' locally
latest: Pulling from library/hello-world
2db29710123e: Pull complete
Digest: sha256:97a379f4f88575512824f3b352bc03cd75e239179eea0fecc38e597b2209f49a
Status: Downloaded newer image for hello-world:latest

Hello from Docker!
This message shows that your installation appears to be working correctly.

To generate this message, Docker took the following steps:
 1. The Docker client contacted the Docker daemon.
 2. The Docker daemon pulled the "hello-world" image from the Docker Hub.
    (amd64)
```

Buscamos el dhcp que queremos, en mi caso voy a elegir Ubuntu

- docker search [lo que queremos buscar]

```
carola@carola-VirtualBox:~$ docker search ubuntu
NAME                              DESCRIPTION                                     STARS     OFFICIAL   AUTOMATED
ubuntu/nginx                      Nginx, a high-performance reverse proxy & we…   31
ubuntu/mysql                      MySQL open source fast, stable, multi-thread…   23
ubuntu/apache2                    Apache, a secure & extensible open-source HT…   23
ubuntu/prometheus                 Prometheus is a systems and service monitori…   22
kasmweb/ubuntu-bionic-desktop     Ubuntu productivity desktop for Kasm Workspa…   17
ubuntu/postgres                   PostgreSQL is an open source object-relation…   13
ubuntu/squid                      Squid is a caching proxy for the Web. Long-t…   12
ubuntu/bind9                      BIND 9 is a very flexible, full-featured DNS…   11
ubuntu/redis                      Redis, an open source key-value store. Long-…   8
ubuntu/prometheus-alertmanager    Alertmanager handles client alerts from Prom…   5
ubuntu/grafana                    Grafana, a feature rich metrics dashboard & …   5
ubuntu/telegraf                   Telegraf collects, processes, aggregates & w…   3
circleci/ubuntu-server            This image is for internal use                  3
ubuntu/memcached                  Memcached, in-memory keyvalue store for smal…   3
ubuntu/cortex                     Cortex provides storage for Prometheus. Long…   2
ubuntu/cassandra                  Cassandra, an open source NoSQL distributed …   1
ubuntu                            Ubuntu is a Debian-based Linux operating sys…   0         [OK]
ubuntu-upstart                    DEPRECATED, as is Upstart (find other proces…   0         [OK]
open-liberty                      Open Liberty multi-architecture images based…   0         [OK]
neurodebian                       NeuroDebian provides neuroscience research s…   0         [OK]
ubuntu-debootstrap                DEPRECATED; use "ubuntu" instead                0         [OK]
bitnami/ubuntu-base-buildpack     Ubuntu base compilation image                   0                    [OK]
websphere-liberty                 WebSphere Liberty multi-architecture images …   0         [OK]
snyk/ubuntu                       A base ubuntu image for all broker clients t…   0
rancher/ubuntuconsole                                                             0
```

Hacemos el pull para descargarnos en este caso ubuntu

```
carola@carola-VirtualBox:~$ docker pull ubuntu
Using default tag: latest
latest: Pulling from library/ubuntu
08c01a0ec47e: Already exists
Digest: sha256:669e010b58baf5beb2836b253c1fd5768333f0d1dbcb834f7c07a4dc93f474be
Status: Downloaded newer image for ubuntu:latest
docker.io/library/ubuntu:latest
```

Comprobamos las imágenes que tenemos y comprobamos que la hemos descargado correctamente

```
carola@carola-VirtualBox:~$ docker images
REPOSITORY    TAG       IMAGE ID       CREATED         SIZE
ubuntu        latest    54c9d81cbb44   2 weeks ago     72.8MB
hello-world   latest    feb5d9fea6a5   4 months ago    13.3kB
```

Implementamos la imagen

Con Docker run empezaríamos el Docker, pero hay veces que necesitamos agregar run -it para entrar

```
carola@carola-VirtualBox:~$ docker run ubuntu
```

```
carola@carola-VirtualBox:~$ docker run -it ubuntu
root@2e0330c00ae3:/#
```

Podemos observar, que tenemos literalmente un sistema operativo dentro de otro sistema operativo

```
root@2e0330c00ae3:/# ls
bin  boot  dev  etc  home  lib  lib32  lib64  libx32  media  mnt  opt  proc  root  run  sbin  srv  sys  tmp  usr  var
root@2e0330c00ae3:/#
```

Al ser otro sistema, inicio la actualización del “sistema”

A su vez, utilizo la combinación de teclas Cntrl+P y Cntrl+Q, ya que necesito el sistema principal mientras que nuestra jaula sigue funcionando, pero si queremos volver a la jaula que hemos dejado en segundo plano, deberemos utilizar:

- **docker attach[nombre_docker]**

Para salir finalmente de nuestro Docker, tendremos que utilizar Cntrl+D ó Cntrl+C ó escribiendo exit en la “terminal”. Podremos comprobar si nuestros Docker están Up o no con:

- **docker ps -a**

```
root@93b96a1c6f37:/# exit
exit
carola@carola-VirtualBox:~$ docker ps -a
CONTAINER ID   IMAGE         COMMAND    CREATED          STATUS                      PORTS     NAMES
93b96a1c6f37   ubuntu        "bash"     18 minutes ago   Exited (0) 8 seconds ago              silly_johnson
116d143910d0   hello-world   "/hello"   52 minutes ago   Exited (0) 52 minutes ago             objective_bassi
```

En cambio, si queremos volver a iniciarlo, deberemos utilizar:

- **docker start [Nombre_docker]**

```
carola@carola-VirtualBox:~$ docker start silly_johnson
silly_johnson
```

Y mas tarde

- **docker attach [Nombre_docker]**

```
carola@carola-VirtualBox:~$ docker attach silly_johnson
```

Dentro de la jaula, voy a iniciar la instalación de un par de servicios, en mi caso apache2 y DHCP

Instalación de servicios

Instalo el servicio DHCP en nuestro docker

```
root@2e0330c00ae3:/# apt install isc-dhcp-server -y
Reading package lists... Done
Building dependency tree
Reading state information... Done
```

Inicio el servicio DHCP

```
root@93b96a1c6f37:/# service isc-dhcp-server start
Launching both IPv4 and IPv6 servers (please configure INTERFACES in /etc/default/isc-dhcp-server if you only want one or the other).
 * Starting ISC DHCPv4 server dhcpd
```

Este servicio, estaría completamente funcional si hacemos la configuración, por eso no esta activo como tal.

Instalo el servicio Apache2

```
root@93b96a1c6f37:/# apt-get install apache2
```

Compruebo que el servicio apache2 está activo

```
root@93b96a1c6f37:/# service apache2 status
 * apache2 is not running
root@93b96a1c6f37:/# service apache2 start
 * Starting Apache httpd web server apache2
AH00558: apache2: Could not reliably determine the server's fully qualified domain name, using 172.17.0.5. Set the 'ServerName' directive globally to suppress this message
 *
root@93b96a1c6f37:/# service apache2 status
 * apache2 is running
root@93b96a1c6f37:/#
```

Para seguir haciendo pruebas, he hecho un pequeño script para comprobar el estado del servicio instalado en nuestra jaula

```
root@93b96a1c6f37: /
  GNU nano 4.8                          comprobacion_apache.sh
#!/bin/bash

if [ $1 == "apache2" ]; then
        echo "Hora de comprobar el servicio apache2"
        service $1 status
else
        echo "No has puesto parametro"
        echo "Opcion:"
        echo "          apache2"
fi
```

Comprobacion de errores:

```
root@93b96a1c6f37:/# ./comprobacion_apache.sh
No has puesto parametro
Opcion:
                apache2
root@93b96a1c6f37:/#
```

Comprobacion normal de script:

```
root@93b96a1c6f37:/# ./comprobacion_apache.sh apache2
Hora de comprobar el servicio apache2
 * apache2 is not running
root@93b96a1c6f37:/#
```

(No esta funcionando, ya que hemos salido previamente del Docker y se han detenido los servicios)

En el siguiente script, voy a iniciar el servicio

```
  GNU nano 4.8
#!/bin/bash

if [[ $1 == "apache2" ]]; then
        echo "Vamos a activar el servicio apache2"
        service $1 start
else
        echo "No me has puesto ningun parametro aceptable"
        echo "Opciones:"
        echo "          apache2"
fi
```

Comprobación de error

```
root@93b96a1c6f37:/# ./activar_apache.sh
No me has puesto ningun parametro aceptable
Opciones:
                apache2
```

Comprobación normal:

```
root@93b96a1c6f37:/# ./activar_apache.sh apache2
Vamos a activar el servicio apache2
 * Starting Apache httpd web server apache2
```

Comprobamos que ha funcionado finalmente con el script anterior:

```
root@93b96a1c6f37:/# ./comprobacion_apache.sh apache2
Hora de comprobar el servicio apache2
 * apache2 is running
root@93b96a1c6f37:/#
```

Script final

Con este script vamos a recoger un conjunto de acciones. Teniendo así, un menú con 8 opciones distintas para completar las necesidades del usuario.

```
while [ 0 -eq 0 ]
do
    echo ""
    echo "[--------------------------- +MENU+ -------------------------------]"
    echo ""
    echo "[------------------- SERVICIOS --------------------]"
    echo "1. Comprobar actualizaciones del equipo e instalarlas"
    echo "2. Ver servicios del equipo instalados"
    echo "3. Mostrar estado de un servicio [X]"
    echo "4. Instalación de DHCP"
    echo "5. Datos DHCP"
    echo "5.1 Que es DHCP"
    echo ""
    echo "[------------------- PAQUETES --------------------]"
    echo "6. Mostrar paquetes instalados"
    echo "7. Mostrar paquete y posible instalacion [X]"
    echo "8. Salir"
    echo ""
    echo "[-----------------------------------------]"
    read -r -p "Dime que opcion quieres: " opcion

    if [ $opcion == "1" ]; then
        echo "[=======================================]"
        echo "[===== Se va a actualizar el equipo =====]"
        echo "[=======================================]"
        sudo apt update -y
        sudo apt upgrade -y
        sleep 2
        echo "[=======================================]"
        echo "[----- Tu equipo se ha actualizado ------]"
        echo "[=======================================]"
        sleep 4
        echo ""
    fi


    if [ $opcion == "2" ]; then
        echo ""
        echo "[=======================================]"
        echo "[==== Se van a mostrar los servicios ====]"
        echo "[=======================================]"
        service --status-all
        sleep 2
        echo ""
        echo "[=======================================]"
        echo "[===== Esos son todos los servicios =====]"
```

```
            echo "[===== Esos son todos los servicios =====]"
            echo "[========================================]"
            sleep 3
        else
            echo ""
            echo "[========================================]"
            echo "NO se mostraran los servicios en ejecucion"
            echo "[========================================]"
            sleep 3
        fi
    fi

    if [ $opcion == "3" ]; then
        read -r -p "Que servicio quieres comprobar?: " servicio1
        if [ $servicio1 ]; then
            echo ""
            echo "[=================================================]"
            echo "[===== Se va a comprobar el servicio $servicio1 =====]"
            echo "[=================================================]"
            echo ""
            sleep 3
            sudo systemctl status $servicio1
            sleep 3
        else
            echo ""
            echo "[++++++++++++++++++++++++++++++++++++]"
            echo "ERROR: No has puesto el servicio a comprobar"
            echo "[++++++++++++++++++++++++++++++++++++]"
            echo ""
            sleep 2
        fi
    fi

    if [ $opcion == "4" ]; then
        echo ""
        echo "[================================================]"
        echo "[==== Va a comenzar la instalación de DHCP ======]"
        echo "[================================================]"
        sleep 3
        echo ""
        echo "[============================================================]"
        echo "[==== Va a comenzar la comprobacion de actualizaciones ======]"
        echo "[============================================================]"
        sleep 3
        sudo apt update; sudo apt upgrade -y
        echo ""
        echo "[=========================================================]"
        echo "[===== Tu sistema se ha actualizado correctamente =====]"
```

```
        echo "[===== Tu sistema se ha actualizado correctamente =====]"
        echo "[======================================================]"
        echo ""
        sleep 3
        echo "++++ Inicio de instalacion del servicio DHCP ++++"
        echo ""
        sleep 3
        sudo apt-get install isc-dhcp-server -y
        echo ""
        echo "[======================================================]"
        echo "[========== Servicio instalado correctamente ==========]"
        echo "[======================================================]"
        echo ""
        sleep 2
        read -r -p "Quieres comprobarlo? (Y/N): " comprob
        if [ $comprob == "Y" ] || [ $comprob == "y" ]; then
            echo ""
            sudo /etc/init.d/isc-dhcp-server status
            echo ""
            sleep 3
        else
            echo ""
            echo "[++++++++++++++++++++++++++++++++++++++++]"
            echo "ERROR: No se va a comprobar el servicio DHCP"
            echo "[++++++++++++++++++++++++++++++++++++++++]"
            echo ""
        fi

    fi


    if [ $opcion == "5" ]; then
        echo ""
        echo "[================================================]"
        echo "[===== Se va a mostrar informacion de DHCP  =====]"
        echo "[================================================]"
        sleep 3
        echo ""
        echo "------------------------------------------------"
        echo "Estas conectado ahora mismo como:"
        whoami
        echo "------------------------------------------------"
        echo "Tu IP actual es:"
        hostname -I
        echo "------------------------------------------------"
        sleep 2
        echo "Tu IP de loopback es:"
        hostname -i
```

```
        hostname -i
        echo "------------------------------------------------"
        sleep 2
        echo "Tu mascara de red y tu IP de broadcast es:"
        ifconfig | grep netmask
        echo "------------------------------------------------"
        sleep 2
        echo "Interfaz de conexion:"
        if [[ `ifconfig | grep eth0` ]]; then
            echo "eth0"
        elif [[ `ifconfig | grep enp0s3` ]]; then
            echo "enp0s3"
        else
            echo "lo"
        fi
        echo "------------------------------------------------"
        sleep 7
        echo ""
        read -r -p "Quieres ver logs de DHCP? (Y/N): " logs

        if [ $logs == "Y" ] || [ $logs == "y" ]; then
        cat /var/log/syslog | grep -E "dhcp"
        sleep 10
        echo ""
        read -r -p "Quieres ver los dispositivos de red? (Y/N): " int
            echo ""
            if [ $int == "Y" ] || [ $int == "y" ]; then
            arp -a
            sleep 4
            else
                echo "[++++++++++++++++++++++++++++++++++++++]"
                echo "[+ No se van a mostrar los dispositivos +] "
                echo "[++++++++++++++++++++++++++++++++++++++]"
                echo ""
                sleep 2
            fi
        echo ""

        else
            echo ""
            echo "[++++++++++++++++++++++++++++++++++++++]"
            echo "[++ No se van mostrar los logs de DHCP ++] "
            echo "[++++++++++++++++++++++++++++++++++++++]"
            echo ""
            sleep 3
        fi
    fi
```

```
if [ $opcion == "5.1" ]; then
    echo ""
    echo "[========================================]"
    echo "El protocolo de configuración dinámica de host es un protocolo de red de tipo cliente/servidor"
    echo "mediante el cual un servidor DHCP asigna dinámicamente una dirección IP y otros parámetros de configuración de red a cada dispositivo en una red para que puedan"
    echo "comunicarse con otras redes IP"
    echo "[========================================]"
    echo ""
    sleep 7
fi

if [ $opcion == "6" ]; then
    echo ""
    echo "[========================================]"
    echo "[==== Se van a mostrar los paquetes =====]"
    echo "[========================================]"
    echo ""
    dpkg -l
fi

if [ $opcion == "7" ]; then
    echo ""
    read -r -p "Que paquete quieres comprobar?: " paquete1
    if [ $paquete1 ]; then
        echo ""
        echo "[========================================]"
        echo "[============ Comprobacion ============]"
        echo "[========================================]"
        echo ""
        sleep 3
        #Comprobar si esta en el sistema y mostrarlo, sino, dar opcion a instalarlo.
        if [[ `dpkg -l $paquete1` ]]; then
            sleep 5
            echo ""
            echo "[========================================]"
            echo "[====== Tu paquete esta instalado =======]"
            echo "[========================================]"
            sleep 4

        else
            read -r -p "Quieres instalarlo? (Y/N): " opcion2
            if [[ $opcion2 == "Y" ]] || [[ $opcion2 == "y" ]]; then
                echo "[========================================]"
                echo "[==== Se va a instalar $paquete1 =====]"
                echo "[========================================]"
                sudo apt install $paquete1 -y
                sudo apt update
                sudo apt upgrade
```

```
                    sudo apt upgrade
                    sleep 3
                    echo "[========================================]"
                    echo "[====== Tu paquete esta instalado =======]"
                    echo "[========================================]"
                else
                    echo ""
                    echo "[++++++++++++++++++++++++++++++++++++++++]"
                    echo " No se va a instalar el paquete $paquete1 "
                    echo "[++++++++++++++++++++++++++++++++++++++++]"
                    echo ""
                    sleep 3
                fi
            fi
        else
            echo ""
            echo "[++++++++++++++++++++++++++++++++++++++++]"
            echo "ERROR: No has puesto el paquete a comprobar"
            echo "[++++++++++++++++++++++++++++++++++++++++]"
            echo ""
            sleep 3
        fi

    fi

    if [ $opcion == "8" ]; then
        echo ""
        echo "[========================================]"
        echo "[============ Fin del script ============]"
        echo "[========================================]"
        echo ""
        sleep 1
        break
    fi
done
```

Funcionamiento

Empezaremos lanzando un menú, en el que el usuario podrá elegir la opción que necesite

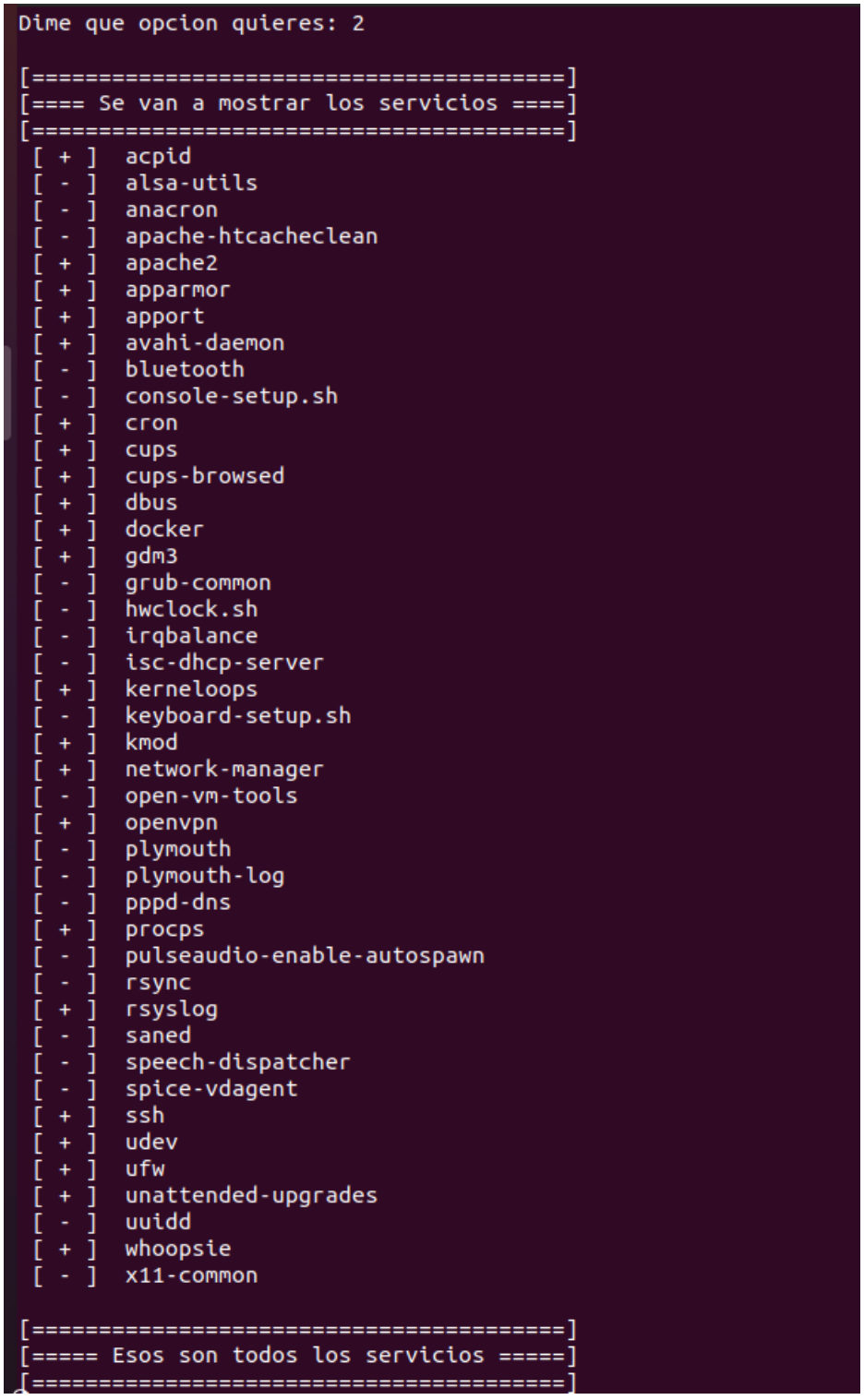
```
carola@carola-VirtualBox:~/ansible-dhcp/script_final_DHCP$ sudo ./script.sh

[---------------------------- +MENU+ ------------------------------]

[------------------- SERVICIOS --------------------]
1. Comprobar actualizaciones del equipo e instalarlas
2. Ver servicios del equipo instalados
3. Mostrar estado de un servicio [X]
4. Instalación de DHCP
5. Datos DHCP
5.1 Que es DHCP

[-------------------- PAQUETES --------------------]
6. Mostrar paquetes instalados
7. Mostrar paquete y posible instalacion [X]
8. Salir

[----------------------------------------]
Dime que opcion quieres:
```

Con la opción 1, comprobaremos las actualizaciones del sistema y en caso de haber, estas se instalaran

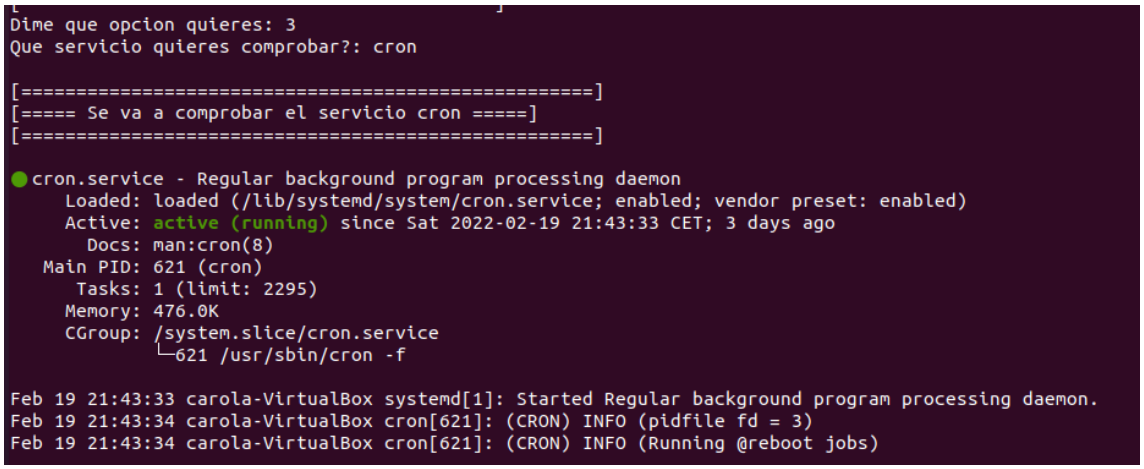
```
Dime que opcion quieres: 1
[======================================]
[===== Se va a actualizar el equipo =====]
[======================================]
Hit:1 https://download.docker.com/linux/ubuntu focal InRelease
Get:2 http://security.ubuntu.com/ubuntu focal-security InRelease [114 kB]
Hit:3 http://es.archive.ubuntu.com/ubuntu focal InRelease
Get:4 http://es.archive.ubuntu.com/ubuntu focal-updates InRelease [114 kB]
Get:5 http://es.archive.ubuntu.com/ubuntu focal-backports InRelease [108 kB]
Get:6 http://security.ubuntu.com/ubuntu focal-security/main amd64 Packages [1,265 kB]
Get:7 http://security.ubuntu.com/ubuntu focal-security/main i386 Packages [383 kB]
Get:8 http://security.ubuntu.com/ubuntu focal-security/main Translation-en [221 kB]
Get:9 http://security.ubuntu.com/ubuntu focal-security/main amd64 DEP-11 Metadata [40.7 kB]
Get:10 http://security.ubuntu.com/ubuntu focal-security/main amd64 c-n-f Metadata [9,624 B]
Get:11 http://security.ubuntu.com/ubuntu focal-security/restricted amd64 Packages [764 kB]
Get:12 http://security.ubuntu.com/ubuntu focal-security/restricted Translation-en [109 kB]
Get:13 http://security.ubuntu.com/ubuntu focal-security/universe i386 Packages [535 kB]
Get:14 http://security.ubuntu.com/ubuntu focal-security/universe amd64 Packages [679 kB]
Get:15 http://security.ubuntu.com/ubuntu focal-security/universe Translation-en [116 kB]
```

```
Setting up libexpat1:amd64 (2.2.9-1ubuntu0.2) ...
Setting up libsasl2-modules:amd64 (2.1.27+dfsg-2ubuntu0.1) ...
Setting up libsasl2-modules-db:amd64 (2.1.27+dfsg-2ubuntu0.1) ...
Setting up libsasl2-2:amd64 (2.1.27+dfsg-2ubuntu0.1) ...
Processing triggers for libc-bin (2.31-0ubuntu9.2) ...
[========================================]
[----- Tu equipo se ha actualizado ------]
[========================================]
```

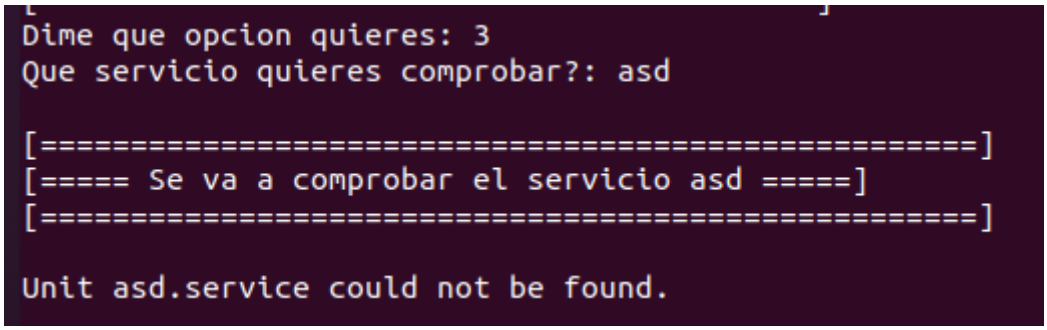
Con la opción 2, comprobaremos los servicios instalados, y a continuación nos dará la opción de si queremos ver solo los servicios en ejecución



Con la tercera opción, comprobaremos el estado de un servicio en concreto puesto por el usuario



En caso de no existir, saldrá un error de comando



En la opción 4, instalaremos el servicio DHCP pero sin configurarlo, y posteriormente nos saldrá la opción de si queremos comprobar el que se haya actualizado. Antes de instalar el servicio, el equipo comprobara actualizaciones y de ser así, las instalara

```
Quieres comprobarlo? (Y/N): Y

● isc-dhcp-server.service - ISC DHCP IPv4 server
     Loaded: loaded (/lib/systemd/system/isc-dhcp-server.service; enabled; vendor preset: enabled)
     Active: failed (Result: exit-code) since Sat 2022-02-19 21:43:38 CET; 3 days ago
       Docs: man:dhcpd(8)
    Process: 877 ExecStart=/bin/sh -ec      CONFIG_FILE=/etc/dhcp/dhcpd.conf;      if [ -f /etc/ltsp/dhcpd.conf ]; then CONFIG_FILE=/etc/ltsp/dhcpd.conf; fi;      [ -e /var/lib/dhcp/dhcpd.leases ] || tou
ch /var/lib/dhcp/dhcpd.leases;      chown root:dhcpd /var/lib/dhcp /var/lib/dhcp/dhcpd.leases;      chmod 775 /var/lib/dhcp ; chmod 664 /var/lib/dhcp/dhcpd.leases;      if test -n "$INTERFACES" -a -z "$I
NTERFACESv4"; then INTERFACESv4="$INTERFACES"; fi;      exec dhcpd -user dhcpd -group dhcpd -f -4 -pf /run/dhcp-server/dhcpd.pid -cf $CONFIG_FILE $INTERFACESv4 (code=exited, status=1/FAILURE)
   Main PID: 877 (code=exited, status=1/FAILURE)

Feb 19 21:43:38 carola-VirtualBox dhcpd[877]:
Feb 19 21:43:38 carola-VirtualBox dhcpd[877]: If you think you have received this message due to a bug rather
Feb 19 21:43:38 carola-VirtualBox dhcpd[877]: than a configuration issue please read the section on submitting
Feb 19 21:43:38 carola-VirtualBox dhcpd[877]: bugs on either our web page at www.isc.org or in the README file
Feb 19 21:43:38 carola-VirtualBox dhcpd[877]: before submitting a bug.  These pages explain the proper
Feb 19 21:43:38 carola-VirtualBox dhcpd[877]: process and the information we find helpful for debugging.
Feb 19 21:43:38 carola-VirtualBox dhcpd[877]:
Feb 19 21:43:38 carola-VirtualBox dhcpd[877]: exiting.
Feb 19 21:43:38 carola-VirtualBox systemd[1]: isc-dhcp-server.service: Main process exited, code=exited, status=1/FAILURE
Feb 19 21:43:38 carola-VirtualBox systemd[1]: isc-dhcp-server.service: Failed with result 'exit-code'.
```

En la opción cinco, veremos información relevante sobre nuestro sistema referido a las interfaces, IP's y usuario. Posteriormente, nos dará la opción de comprobar los logs de DHCP y la opción de poder ver los logs de DHCP

```
Dime que opcion quieres: 5

[================================================]
[===== Se va a mostrar informacion de DHCP  =====]
[================================================]

--------------------------------------------------
Estas conectado ahora mismo como:
root
--------------------------------------------------
Tu IP actual es:
192.168.0.38 172.17.0.1
--------------------------------------------------
Tu IP de loopback es:
127.0.1.1
--------------------------------------------------
Tu mascara de red y tu IP de broadcast es:
        inet 172.17.0.1  netmask 255.255.0.0  broadcast 172.17.255.255
        inet 192.168.0.38  netmask 255.255.255.0  broadcast 192.168.0.255
        inet 127.0.0.1  netmask 255.0.0.0
--------------------------------------------------
Interfaz de conexion:
enp0s3
--------------------------------------------------

Quieres ver logs de DHCP? (Y/N):
```

```
Quieres ver logs de DHCP? (Y/N): y
Feb 23 11:00:34 carola-VirtualBox NetworkManager[624]: <info>  [1645610434.4669] dhcp4 (enp0s3): canceled DHCP transaction
Feb 23 11:00:34 carola-VirtualBox NetworkManager[624]: <info>  [1645610434.4669] dhcp4 (enp0s3): state changed bound -> done
Feb 23 11:00:34 carola-VirtualBox NetworkManager[624]: <info>  [1645610434.4674] dhcp4 (enp0s3): activation: beginning transaction (timeout in 45 seconds)
Feb 23 11:00:35 carola-VirtualBox NetworkManager[624]: <info>  [1645610435.7336] dhcp4 (enp0s3): option dhcp_lease_time      => '86400'
Feb 23 11:00:35 carola-VirtualBox NetworkManager[624]: <info>  [1645610435.7347] dhcp4 (enp0s3): option domain_name_servers  => '212.166.210.80 212.166.132.104'
Feb 23 11:00:35 carola-VirtualBox NetworkManager[624]: <info>  [1645610435.7350] dhcp4 (enp0s3): option expiry              => '1645696834'
Feb 23 11:00:35 carola-VirtualBox NetworkManager[624]: <info>  [1645610435.7353] dhcp4 (enp0s3): option ip_address          => '192.168.0.38'
Feb 23 11:00:35 carola-VirtualBox NetworkManager[624]: <info>  [1645610435.7397] dhcp4 (enp0s3): option next_server         => '192.168.0.1'
Feb 23 11:00:35 carola-VirtualBox NetworkManager[624]: <info>  [1645610435.7401] dhcp4 (enp0s3): option requested_broadcast_address => '1'
Feb 23 11:00:35 carola-VirtualBox NetworkManager[624]: <info>  [1645610435.7403] dhcp4 (enp0s3): option requested_domain_name => '1'
Feb 23 11:00:35 carola-VirtualBox NetworkManager[624]: <info>  [1645610435.7404] dhcp4 (enp0s3): option requested_domain_name_servers => '1'
Feb 23 11:00:35 carola-VirtualBox NetworkManager[624]: <info>  [1645610435.7405] dhcp4 (enp0s3): option requested_domain_search => '1'
Feb 23 11:00:35 carola-VirtualBox NetworkManager[624]: <info>  [1645610435.7405] dhcp4 (enp0s3): option requested_host_name  => '1'
Feb 23 11:00:35 carola-VirtualBox NetworkManager[624]: <info>  [1645610435.7405] dhcp4 (enp0s3): option requested_interface_mtu => '1'
Feb 23 11:00:35 carola-VirtualBox NetworkManager[624]: <info>  [1645610435.7406] dhcp4 (enp0s3): option requested_ms_classless_static_routes => '1'
```

```
Quieres ver logs de DHCP? (Y/N): y
Feb 23 11:00:34 carola-VirtualBox NetworkManager[624]: <info>  [1645610434.4669] dhcp4 (enp0s3): canceled DHCP transaction
Feb 23 11:00:34 carola-VirtualBox NetworkManager[624]: <info>  [1645610434.4669] dhcp4 (enp0s3): state changed bound -> done
Feb 23 11:00:34 carola-VirtualBox NetworkManager[624]: <info>  [1645610434.4674] dhcp4 (enp0s3): activation: beginning transaction (timeout in 45 seconds)
Feb 23 11:00:35 carola-VirtualBox NetworkManager[624]: <info>  [1645610435.7336] dhcp4 (enp0s3): option dhcp_lease_time      => '86400'
Feb 23 11:00:35 carola-VirtualBox NetworkManager[624]: <info>  [1645610435.7347] dhcp4 (enp0s3): option domain_name_servers  => '212.166.210.80 212.166.132.104'
Feb 23 11:00:35 carola-VirtualBox NetworkManager[624]: <info>  [1645610435.7350] dhcp4 (enp0s3): option expiry              => '1645696834'
Feb 23 11:00:35 carola-VirtualBox NetworkManager[624]: <info>  [1645610435.7353] dhcp4 (enp0s3): option ip_address          => '192.168.0.38'
Feb 23 11:00:35 carola-VirtualBox NetworkManager[624]: <info>  [1645610435.7397] dhcp4 (enp0s3): option next_server         => '192.168.0.1'
Feb 23 11:00:35 carola-VirtualBox NetworkManager[624]: <info>  [1645610435.7401] dhcp4 (enp0s3): option requested_broadcast_address => '1'
Feb 23 11:00:35 carola-VirtualBox NetworkManager[624]: <info>  [1645610435.7403] dhcp4 (enp0s3): option requested_domain_name => '1'
Feb 23 11:00:35 carola-VirtualBox NetworkManager[624]: <info>  [1645610435.7404] dhcp4 (enp0s3): option requested_domain_name_servers => '1'
Feb 23 11:00:35 carola-VirtualBox NetworkManager[624]: <info>  [1645610435.7405] dhcp4 (enp0s3): option requested_domain_search => '1'
Feb 23 11:00:35 carola-VirtualBox NetworkManager[624]: <info>  [1645610435.7405] dhcp4 (enp0s3): option requested_host_name  => '1'
Feb 23 11:00:35 carola-VirtualBox NetworkManager[624]: <info>  [1645610435.7405] dhcp4 (enp0s3): option requested_interface_mtu => '1'
Feb 23 11:00:35 carola-VirtualBox NetworkManager[624]: <info>  [1645610435.7406] dhcp4 (enp0s3): option requested_ms_classless_static_routes => '1'
Feb 23 11:00:35 carola-VirtualBox NetworkManager[624]: <info>  [1645610435.7406] dhcp4 (enp0s3): option requested_nis_domain => '1'
Feb 23 11:00:35 carola-VirtualBox NetworkManager[624]: <info>  [1645610435.7406] dhcp4 (enp0s3): option requested_nis_servers => '1'
Feb 23 11:00:35 carola-VirtualBox NetworkManager[624]: <info>  [1645610435.7406] dhcp4 (enp0s3): option requested_ntp_servers => '1'
Feb 23 11:00:35 carola-VirtualBox NetworkManager[624]: <info>  [1645610435.7406] dhcp4 (enp0s3): option requested_rfc3442_classless_static_routes => '1'
Feb 23 11:00:35 carola-VirtualBox NetworkManager[624]: <info>  [1645610435.7407] dhcp4 (enp0s3): option requested_root_path  => '1'
Feb 23 11:00:35 carola-VirtualBox NetworkManager[624]: <info>  [1645610435.7407] dhcp4 (enp0s3): option requested_routers    => '1'
Feb 23 11:00:35 carola-VirtualBox NetworkManager[624]: <info>  [1645610435.7407] dhcp4 (enp0s3): option requested_static_routes => '1'
Feb 23 11:00:35 carola-VirtualBox NetworkManager[624]: <info>  [1645610435.7407] dhcp4 (enp0s3): option requested_subnet_mask => '1'
Feb 23 11:00:35 carola-VirtualBox NetworkManager[624]: <info>  [1645610435.7407] dhcp4 (enp0s3): option requested_time_offset => '1'
Feb 23 11:00:35 carola-VirtualBox NetworkManager[624]: <info>  [1645610435.7408] dhcp4 (enp0s3): option requested_wpad       => '1'
Feb 23 11:00:35 carola-VirtualBox NetworkManager[624]: <info>  [1645610435.7408] dhcp4 (enp0s3): option routers             => '192.168.0.1'
Feb 23 11:00:35 carola-VirtualBox NetworkManager[624]: <info>  [1645610435.7408] dhcp4 (enp0s3): option subnet_mask         => '255.255.255.0'
Feb 23 11:00:35 carola-VirtualBox NetworkManager[624]: <info>  [1645610435.7408] dhcp4 (enp0s3): state changed unknown -> bound

Quieres ver los dispositivos de red? (Y/N):
```

```
Quieres ver los dispositivos de red? (Y/N): y

? (192.168.0.1) at d0:57:94:78:d2:24 [ether] on enp0s3
```

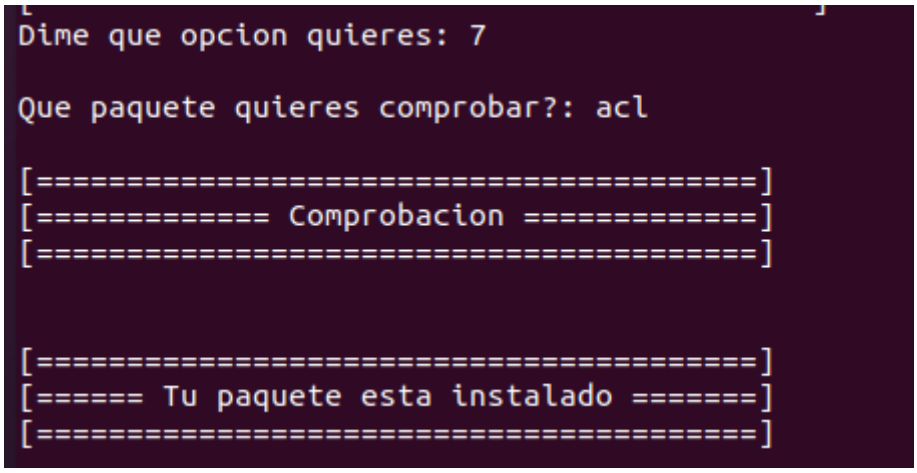
Con la opción 5.1, nos dirá información sobre que es DHCP

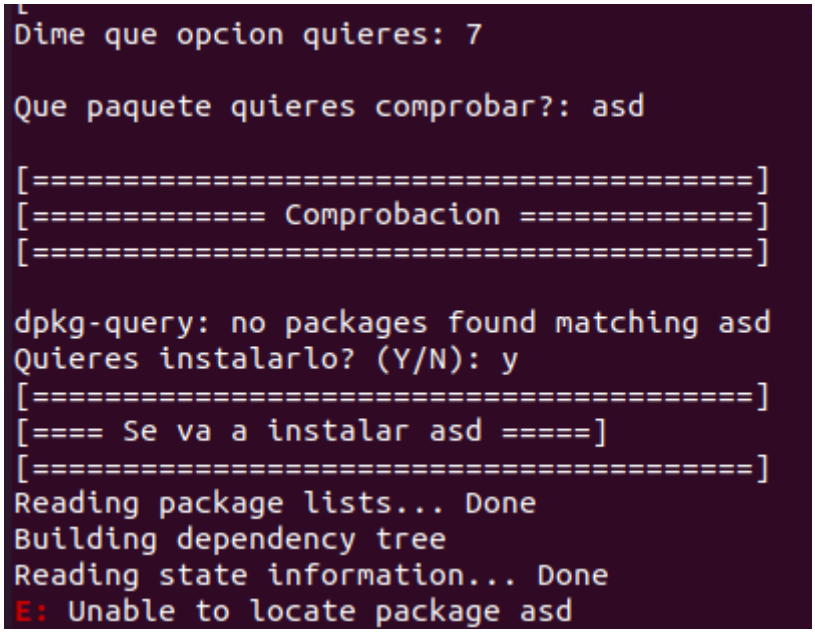
```
Dime que opcion quieres: 5.1

[========================================]
El protocolo de configuración dinámica de host es un protocolo de red de tipo cliente/servidor
mediante el cual un servidor DHCP asigna dinámicamente una dirección IP y otros parámetros de configuración de red a cada dispositivo en una red para que puedan
comunicarse con otras redes IP
[========================================]
```

Con la sexta opción, podremos observar los paquetes instalados en el sistema

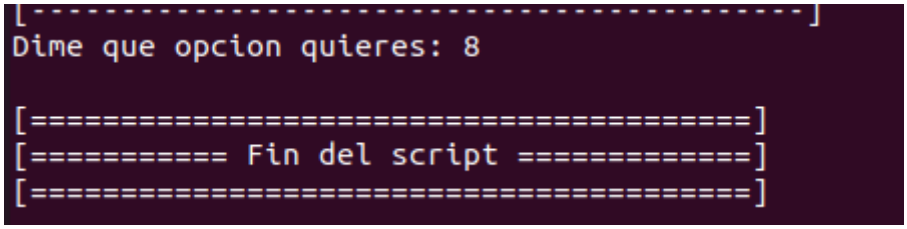
```
Dime que opcion quieres: 6

[========================================]
[==== Se van a mostrar los paquetes =====]
[========================================]

Desired=Unknown/Install/Remove/Purge/Hold
| Status=Not/Inst/Conf-files/Unpacked/half-conf/Half-inst/trig-aWait/Trig-pend
|/ Err?=(none)/Reinst-required (Status,Err: uppercase=bad)
||/ Name                                  Version                                 Architecture Description
+++-=====================================-=======================================-============-===============================================================================================
ii  accountsservice                       0.6.55-0ubuntu12~20.04.5                amd64        query and manipulate user account information
ii  acl                                   2.2.53-6                                amd64        access control list - utilities
ii  acpi-support                          0.143                                   amd64        scripts for handling many ACPI events
ii  acpid                                 1:2.0.32-1ubuntu1                       amd64        Advanced Configuration and Power Interface event daemon
ii  adduser                               3.118ubuntu2                            all          add and remove users and groups
ii  adwaita-icon-theme                    3.36.1-2ubuntu0.20.04.2                 all          default icon theme of GNOME (small subset)
ii  alsa-base                             1.0.25+dfsg-0ubuntu5                    all          ALSA driver configuration files
ii  alsa-topology-conf                    1.2.2-1                                 all          ALSA topology configuration files
ii  alsa-ucm-conf                         1.2.2-1ubuntu0.11                       all          ALSA Use Case Manager configuration files
ii  alsa-utils                            1.2.2-1ubuntu2.1                        amd64        Utilities for configuring and using ALSA
ii  amd64-microcode                       3.20191218.1ubuntu1                     amd64        Processor microcode firmware for AMD CPUs
ii  anacron                               2.3-29                                  amd64        cron-like program that doesn't go by time
ii  ansible                               2.9.6+dfsg-1                            all          Configuration management, deployment, and task execution system
ii  apache2                               2.4.41-4ubuntu3.9                       amd64        Apache HTTP Server
ii  apache2-bin                           2.4.41-4ubuntu3.9                       amd64        Apache HTTP Server (modules and other binary files)
ii  apache2-data                          2.4.41-4ubuntu3.9                       all          Apache HTTP Server (common files)
ii  apache2-utils                         2.4.41-4ubuntu3.9                       amd64        Apache HTTP Server (utility programs for web servers)
ii  apg                                   2.2.3.dfsg.1-5                          amd64        Automated Password Generator - Standalone version
ii  app-install-data-partner              19.04                                   all          Application Installer (data files for partner applications/repositories)
ii  apparmor                              2.13.3-7ubuntu5.1                       amd64        user-space parser utility for AppArmor
ii  apport                                2.20.11-0ubuntu27.21                    all          automatically generate crash reports for debugging
ii  apport-gtk                            2.20.11-0ubuntu27.21                    all          GTK+ frontend for the apport crash report system
ii  apport-symptoms                       0.23                                    all          symptom scripts for apport
ii  appstream                             0.12.10-2                               amd64        Software component metadata management
ii  apt                                   2.0.6                                   amd64        commandline package manager
ii  apt-config-icons                      0.12.10-2                               all          APT configuration snippet to enable icon downloads
ii  apt-config-icons-hidpi                0.12.10-2                               all          APT configuration snippet to enable HiDPI icon downloads
ii  apt-transport-https                   2.0.6                                   all          transitional package for https support
ii  apt-utils                             2.0.6                                   amd64        package management related utility programs
ii  aptdaemon                             1.1.1+bzr982-0ubuntu32.3                all          transaction based package management service
ii  aptdaemon-data                        1.1.1+bzr982-0ubuntu32.3                all          data files for clients
ii  apturl                                0.5.2ubuntu19                           amd64        install packages using the apt protocol - GTK+ frontend
ii  apturl-common                         0.5.2ubuntu19                           amd64        install packages using the apt protocol - common data
ii  aspell                                0.60.8-1ubuntu0.1                       amd64        GNU Aspell spell-checker
ii  aspell-en                             2018.04.16-0-1                          all          English dictionary for GNU Aspell
ii  at-spi2-core                          2.36.0-2                                amd64        Assistive Technology Service Provider Interface (dbus core)
ii  avahi-autoipd                         0.7-4ubuntu7.1                          amd64        Avahi IPv4LL network address configuration daemon
ii  avahi-daemon                          0.7-4ubuntu7.1                          amd64        Avahi mDNS/DNS-SD daemon
ii  avahi-utils                           0.7-4ubuntu7.1                          amd64        Avahi browsing, publishing and discovery utilities
ii  base-files                            11ubuntu5.5                             amd64        Debian base system miscellaneous files
```

Con la séptima opción, podremos comprobar si un paquete esta instalado y en caso de que no, lo podremos instalar. Si ese paquete no es encontrado, dpkg nos dará error.





Con la octava opción, saldremos del script



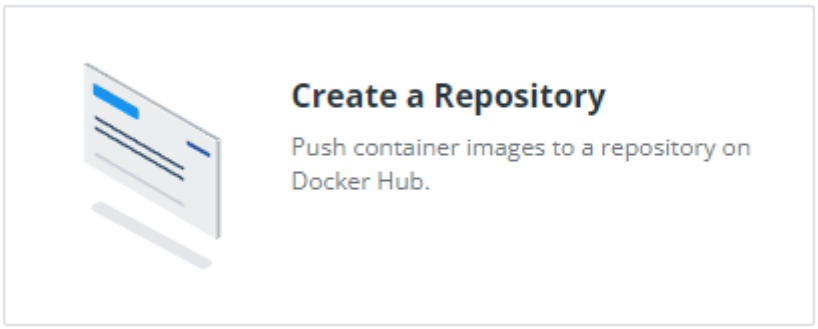
Subir archivos

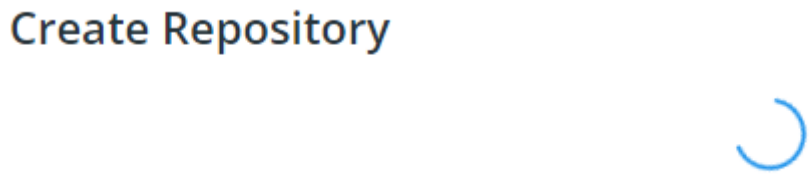
Subir Docker

Agregaremos un comentario a nuestra imagen sobre nuestro usuario

```
carola@carola-VirtualBox:~/.docker$ docker commit -m "ubuntu con cosas" -a "shayinn" 93b96a1c6f3
7 "ubuntu_final"
sha256:ec80d2381e899364931e3773fdbbf55b01a0eac73a0ce2560e69ee558f73ee55
```

```
carola@carola-VirtualBox:~/.docker$ docker images
REPOSITORY                    TAG       IMAGE ID       CREATED          SIZE
ubuntu_final                  latest    ec80d2381e89   27 seconds ago   287MB
ubuntu_apache2_dhcp_scripts   latest    f8095b015a12   20 minutes ago   287MB
ubuntu                        latest    54c9d81cbb44   2 weeks ago      72.8MB
hello-world                   latest    feb5d9fea6a5   4 months ago     13.3kB
carola@carola-VirtualBox:~/.docker$
```





(no puedo crearlo ya que no me carga)

Generaremos un PAT (Personal Access Token)

New Access Token

A personal access token is similar to a password except you can have many tokens and revoke access to each one at any time. Learn more

Access Token Description *

tocken_apache_dhcp

Access permissions

Read, Write, Delete

Read, Write, Delete tokens allow you to manage your repositories.

Iniciaremos sesión en Docker con nuestro nombre de usuario y nuestra contraseña

```
carola@carola-VirtualBox:~/ansible-dhcp$ docker login -u shayinn
Password:
WARNING! Your password will be stored unencrypted in /home/carola/.docker/config.json.
Configure a credential helper to remove this warning. See
https://docs.docker.com/engine/reference/commandline/login/#credentials-store

Login Succeeded
```

Crearemos una imagen con nuestro nombre de usuario

```
carola@carola-VirtualBox:~/.docker$ docker tag ubuntu_final:latest shayinn/ubuntu_final
carola@carola-VirtualBox:~/.docker$ docker images
REPOSITORY                  TAG       IMAGE ID       CREATED          SIZE
shayinn/ubuntu_final        latest    ec80d2381e89   2 minutes ago    287MB
ubuntu_final                latest    ec80d2381e89   2 minutes ago    287MB
```

Subiremos la imagen

```
carola@carola-VirtualBox:~/.docker$ docker push shayinn/ubuntu_final
Using default tag: latest
The push refers to repository [docker.io/shayinn/ubuntu_final]
51bb2a302593: Pushed
36ffdceb4c77: Mounted from shayinn/ubuntu_apache2_dhcp_scripts
latest: digest: sha256:28bc86441d9e3597d9da0fcb018297fd1d8d324a7538e2632b2be1d4735a1cd6 size: 74
1
```

Buscamos las imágenes de nuestro usuario

```
carola@carola-VirtualBox:~/.docker$ docker search shayinn
NAME      DESCRIPTION   STARS     OFFICIAL   AUTOMATED
carola@carola-VirtualBox:~/.docker$
```

Debería de salirnos nuestro repositorio, pero parece ser que no es posible, al no haber creado el repositorio previamente ya que no he podido

Subir Ansible y script final a GitHub

Lo primero de todo será agregar todo el directorio (.)

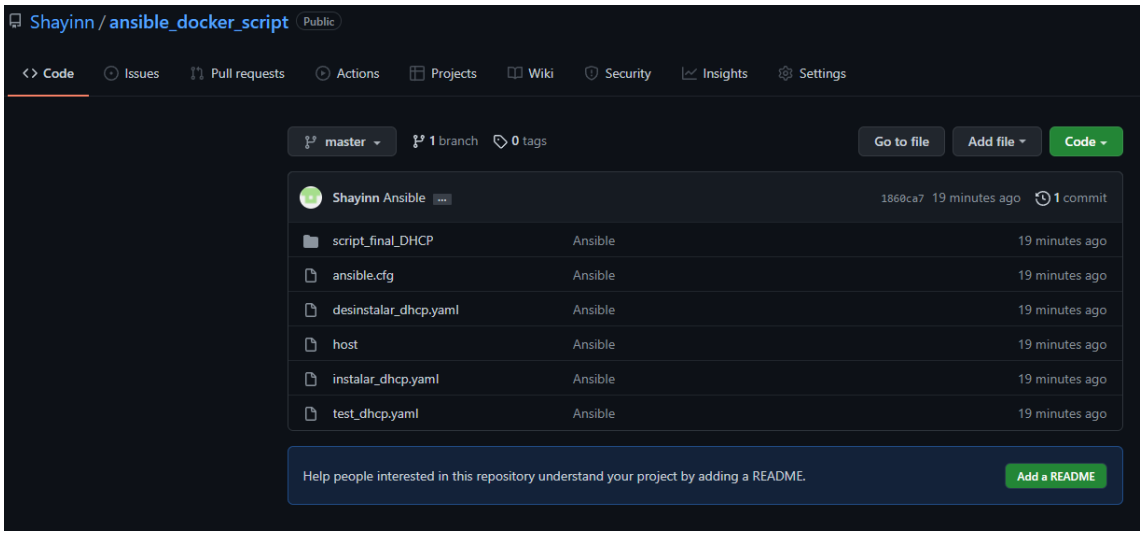
Posteriormente, agregaremos un comentario, en mi caso “Ansible”

```
carola@carola-VirtualBox:~/ansible-dhcp$ git add .
carola@carola-VirtualBox:~/ansible-dhcp$ git commit -m "Ansible"
[master (root-commit) 4685450] Ansible
 7 files changed, 316 insertions(+)
 create mode 100644 ansible.cfg
 create mode 100644 desinstalar_dhcp.yaml
 create mode 100644 host
 create mode 100644 instalar_dhcp.yaml
 create mode 100644 script_final_DHCP/git.txt
 create mode 100644 script_final_DHCP/script.sh
 create mode 100644 test_dhcp.yaml
```

Agregaremos de manera remota el repositorio y lo subiremos a la rama principal

```
carola@carola-VirtualBox:~/ansible-dhcp$ git remote add origin https://github.com/Shayinn/ansible_docker_script.git
```

```
carola@carola-VirtualBox:~/ansible-dhcp$ git push --set-upstream origin master
Username for 'https://github.com': Shayinn
Password for 'https://Shayinn@github.com':
Enumerating objects: 10, done.
Counting objects: 100% (10/10), done.
Compressing objects: 100% (10/10), done.
Writing objects: 100% (10/10), 2.72 KiB | 2.72 MiB/s, done.
Total 10 (delta 1), reused 0 (delta 0)
remote: Resolving deltas: 100% (1/1), done.
To https://github.com/Shayinn/ansible_docker_script.git
 * [new branch]      master -> master
Branch 'master' set up to track remote branch 'master' from 'origin'.
```



Subir Docker a GitHub

Lo primero de todo es realizas un PAT (Personal Access Token) y lo guardaremos en un fichero, en mi caso PAT.txt

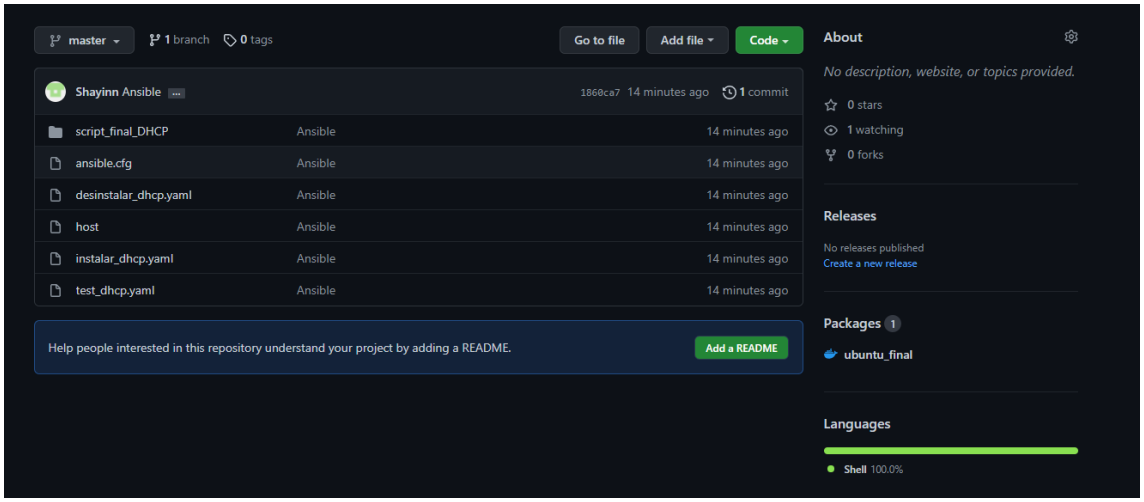
Con el siguiente comando, haremos que nuestro PAT de GitHub, sea “recogido” y directamente autenticado nuestro usuario al haberle puesto el cat PAT.txt

```
carola@carola-VirtualBox:~$ cat PAT.txt | docker login https://docker.pkg.github.com -u shayinn
--password-stdin
WARNING! Your password will be stored unencrypted in /home/carola/.docker/config.json.
Configure a credential helper to remove this warning. See
https://docs.docker.com/engine/reference/commandline/login/#credentials-store

Login Succeeded
```

```
carola@carola-VirtualBox:~$ docker tag ubuntu_final docker.pkg.github.com/shayinn/ansible_docker
_script/ubuntu_final:v1
```

Subimos el paquete de Docker a nuestro GitHub finalmente agregando la version

```
carola@carola-VirtualBox:~$ docker push docker.pkg.github.com/shayinn/ansible_docker_script/ubun
tu_final:v1
The push refers to repository [docker.pkg.github.com/shayinn/ansible_docker_script/ubuntu_final]
51bb2a302593: Pushed
36ffdceb4c77: Pushed
v1: digest: sha256:b8ce2d7096d074c45149123a3b26f3d5b9c298dbed2cb4151893bbefe20c0eb0 size: 741
```



Link a mi GitHub: https://github.com/Shayinn