

**PONTIFICIA UNIVERSIDAD CATÓLICA DEL
PERÚ**

Facultad de Ciencias e Ingeniería

ADMINISTRACIÓN DE SISTEMAS OPERATIVOS

LABORATORIO 4

Alumno: Saymon Nicho

Profesor: Mario Carpio

Horario: 0781

Lima, 29 de octubre del 2024

Tabla de Contenidos

PREGUNTA 1	3
Figura 1.1. Base de datos creada	3
Figura 1.2. Actualización de la variable ORACLE_SID	3
Figura 1.3. Consulta a la base de datos con sqlplus	4
PREGUNTA 2	5
Figura 2.1. SELECT de tablespaces y datos asociados	5
Figura 2.2. Archivos de control y muestra de parámetros	5
PREGUNTA 3	6
Figura 3.1. Verificación de que el archivo /etc/oratab tenga Y	6
Figura 3.2. Contenido de los scripts start.sh y stop.sh	7
Figura 3.3. Prueba manual de que los scripts funcionan correctamente	7
Figura 3.4. Configuración del servicio dbora	7
Figura 3.5. Prueba de que el servicio funciona correctamente	8
PREGUNTA 4	9
Figura 4.1. Acceso a la instancia Ubuntu con ssh	9
Figura 4.2. Instalación de apache2 y openssl	9
Figura 4.3. Habilitación del servicio apache2	10
Figura 4.4. Resultado de la generación de una clave privada	10
Figura 4.5. Configuración que permite redireccionar de HTTP a HTTPS	11
Figura 4.6. Habilitación del servicio	11
Figura 4.7. Servidor levantado con IP 100.29.110.48	12
Figura 4.8. Página principal de noip.com	12
Figura 4.9. Hostname apuntando a la dirección pública de la instancia	13
Figura 4.10. Verificación de que se logró enlazar la dirección IP con nslookup	13
Figura 4.11. Obtención del certificado gratuito de seguridad	14
Figura 4.12. Certificado de seguridad visto desde el navegador	14
Figura 4.13. Comprobación de que el servidor tiene soporte para HTTPS	15

PREGUNTA 1

En primer lugar, se crea la base de datos según las especificaciones de la guía. A continuación se muestra el mensaje de éxito al finalizar su creación.

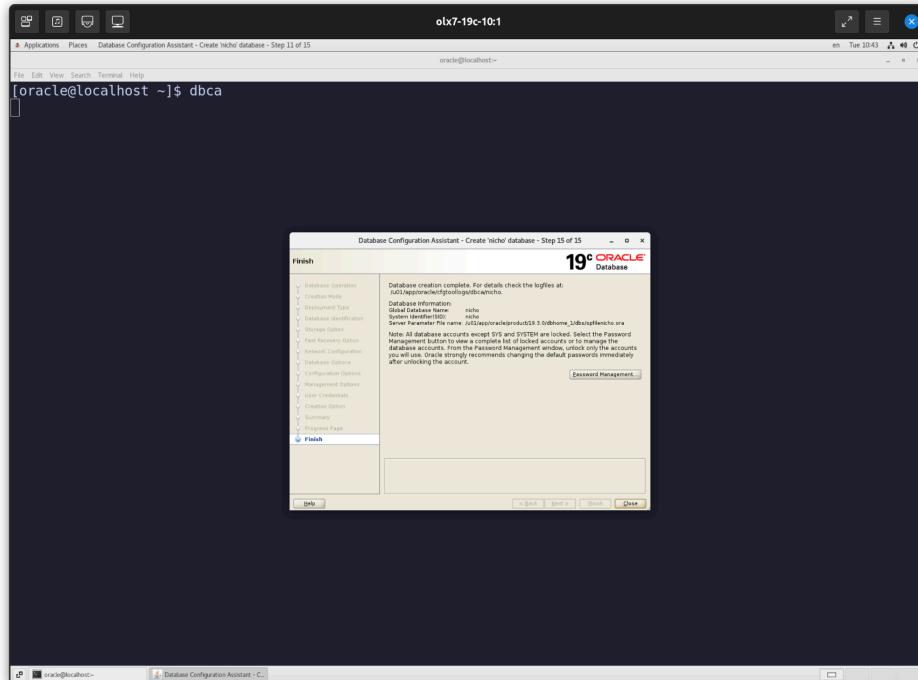


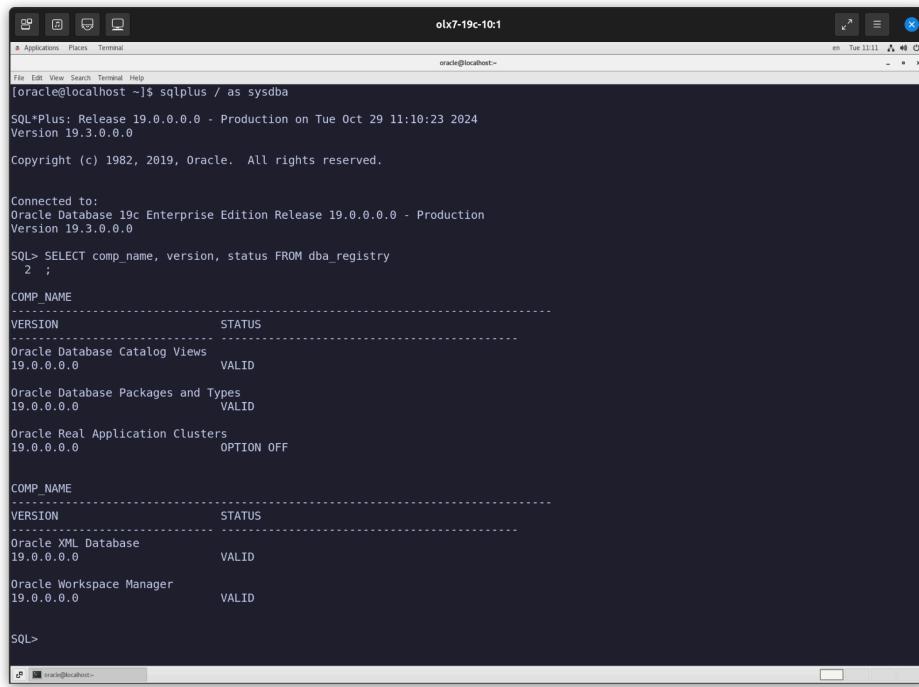
Figura 1.1. Base de datos creada

Luego, se actualiza el valor de la variable de entorno `ORACLE_SID` en el archivo `.bash_profile`. Se le da el valor de `nicho`.

A screenshot of a terminal window titled 'Terminal' showing the execution of several commands. The user runs dbca, then echo \$ORACLE_HOME to verify it's /u01/app/oracle/product/19.3.0/dbhome_1. They open a vim editor on their .bash_profile file. Inside, they add a line setting ORACLE_SID=nicho. They then run cat -n .bash_profile to view the changes, source the profile with source .bash_profile, and finally echo \$ORACLE_SID to confirm it has been set to nicho.

Figura 1.2. Actualización de la variable `ORACLE_SID`

Después, se realiza la consulta a la vista `dba_registry` en la base de datos Oracle, y esta devuelve el nombre del componente, su versión y su estado actual.



The screenshot shows a terminal window titled "olx7-19c-10:1" running on an Oracle Database 19c Enterprise Edition. The session is connected as "oracle@localhost" with a sysdba privilege. The user has run the command "SELECT comp_name, version, status FROM dba_registry" and the results are displayed in two sections. The first section lists components under "Oracle Database Catalog Views" and "Oracle Database Packages and Types". The second section lists components under "Oracle Real Application Clusters" and "Oracle XML Database". All components shown have a status of "VALID".

```
[oracle@localhost ~]$ sqlplus / as sysdba
SQL*Plus: Release 19.0.0.0.0 - Production on Tue Oct 29 11:10:23 2024
Version 19.3.0.0.0
Copyright (c) 1982, 2019, Oracle. All rights reserved.

Connected to:
Oracle Database 19c Enterprise Edition Release 19.0.0.0.0 - Production
Version 19.3.0.0.0

SQL> SELECT comp_name, version, status FROM dba_registry
2 ;
COMP_NAME
-----
VERSION          STATUS
-----
Oracle Database Catalog Views
19.0.0.0.0        VALID
Oracle Database Packages and Types
19.0.0.0.0        VALID
Oracle Real Application Clusters
19.0.0.0.0        OPTION OFF

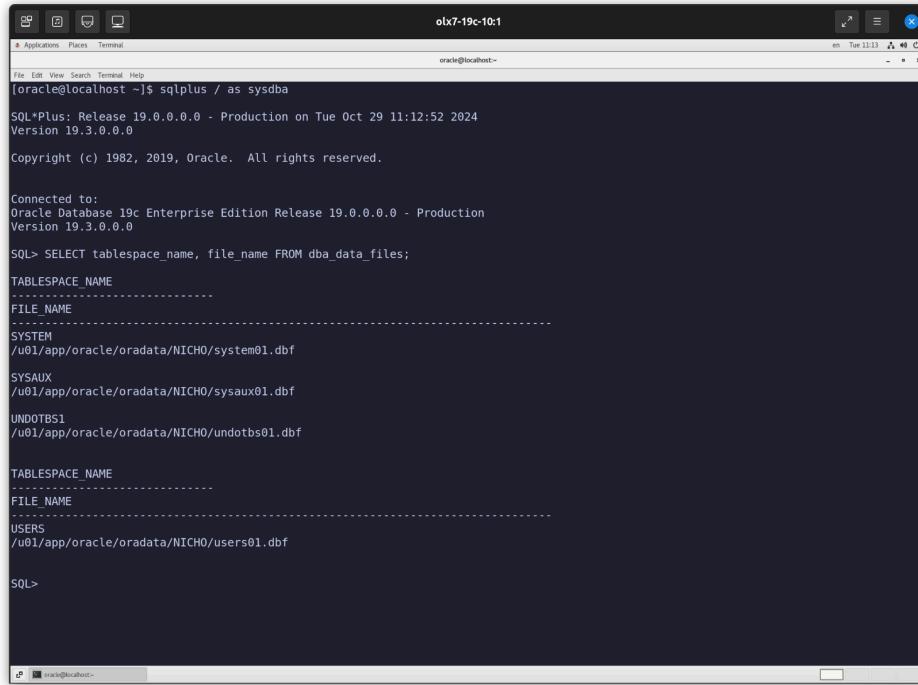
COMP_NAME
-----
VERSION          STATUS
-----
Oracle XML Database
19.0.0.0.0        VALID
Oracle Workspace Manager
19.0.0.0.0        VALID

SQL>
```

Figura 1.3. Consulta a la base de datos con `sqlplus`

PREGUNTA 2

Se realizan las consultas pedidas. En primer lugar se consultan los tablespaces y sus archivos de datos asociados.



```
oracle@localhost ~]$ sqlplus / as sysdba
SQL*Plus: Release 19.0.0.0.0 - Production on Tue Oct 29 11:12:52 2024
Version 19.3.0.0.0
Copyright (c) 1982, 2019, Oracle. All rights reserved.

Connected to:
Oracle Database 19c Enterprise Edition Release 19.0.0.0.0 - Production
Version 19.3.0.0.0

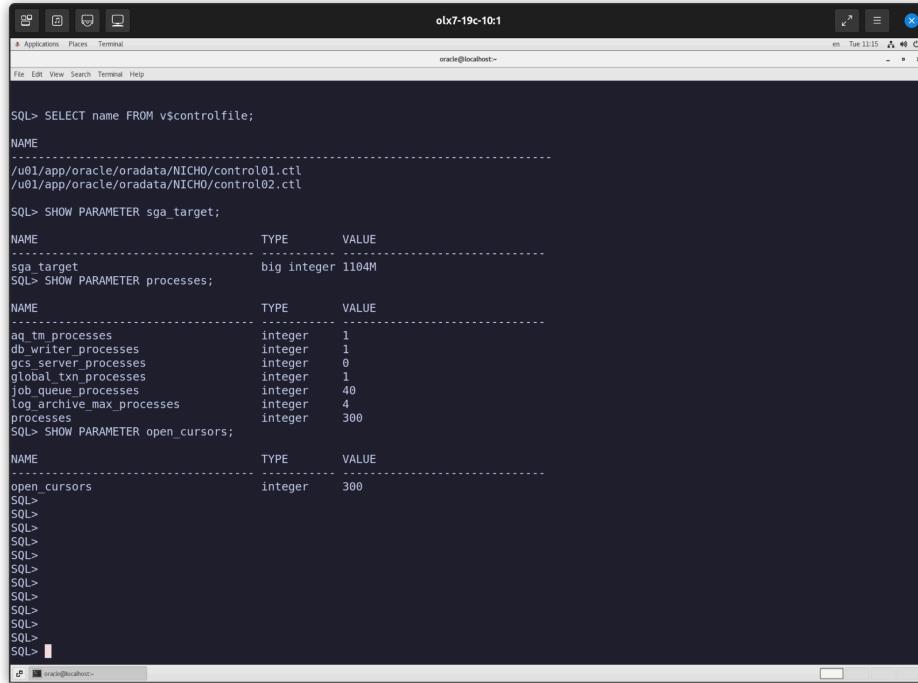
SQL> SELECT tablespace_name, file_name FROM dba_data_files;
TABLESPACE_NAME
-----
FILE_NAME
-----
SYSTEM
/u01/app/oracle/oradata/NICHO/system01.dbf
SYSAUX
/u01/app/oracle/oradata/NICHO/sysaux01.dbf
UNDOTBS1
/u01/app/oracle/oradata/NICHO/undotbs01.dbf

TABLESPACE_NAME
-----
FILE_NAME
-----
USERS
/u01/app/oracle/oradata/NICHO/users01.dbf

SQL>
```

Figura 2.1. **SELECT** de tablespaces y datos asociados

Luego, se obtiene la ubicación de los archivos de control con **SELECT** y se usa **SHOW PARAMETER** para mostrar los valores pedidos.



```
oracle@localhost ~]$ SELECT name FROM v$controlfile;
NAME
-----
/u01/app/oracle/oradata/NICHO/control01.ctl
/u01/app/oracle/oradata/NICHO/control02.ctl

SQL> SHOW PARAMETER sga_target;
NAME          TYPE        VALUE
sga_target    big integer 1104M
SQL> SHOW PARAMETER processes;
NAME          TYPE        VALUE
sga_target    big integer 1104M
processes     integer   300
SQL> SHOW PARAMETER open_cursors;
NAME          TYPE        VALUE
open_cursors  integer   300
SQL>
```

Figura 2.2. Archivos de control y muestra de parámetros

PREGUNTA 3

En primer lugar, se verifica el contenido de /etc/oratab contenga una Y. Esto permite que dbshut y dbstart incluyan la instancia **nicho** al momento de ejecutar los scripts.

Figura 3.1. Verificación de que el archivo `/etc/oratab` tenga Y

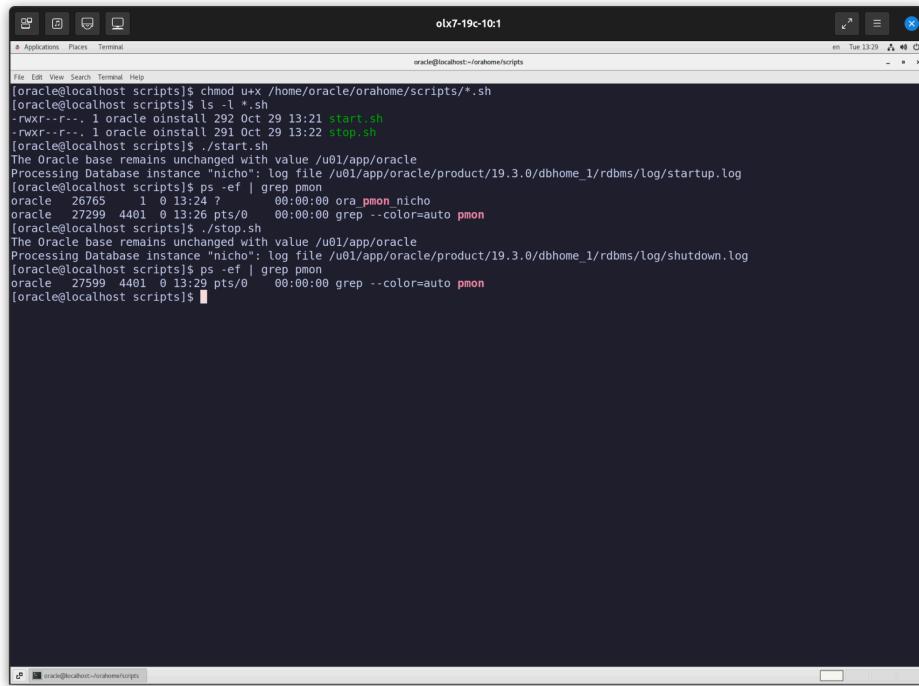
Luego, se crean los scripts con los permisos necesarios y se prueban de forma manual. Para comprobar que funcionan correctamente se usa `ps` con `grep` para observar si el proceso efectivamente inicia y se detiene.

```
File Edit View Search Terminal Help
[oracle@localhost ~]$ ls
Desktop Documents Downloads Music orahome Pictures Public Templates Videos
[oracle@localhost ~]$ cd orahome/scripts/
[oracle@localhost scripts]$ vim start.sh
[oracle@localhost scripts]$ cat -n start.sh
 1 #!/bin/bash
 2
 3 export TMP=/tmp
 4 export TMPDIR=$TMP
 5
 6 export ORACLE_HOME=/u01/app/oracle/product/19.3.0/dbhome_1
 7 export ORACLE_SID=nicho
 8
 9 export PATH=/usr/sbin:/usr/local/bin:$PATH
10 export PATH=$ORACLE_HOME/bin:$PATH
11
12 export ORAENV_ASK=NO
13 source oraenv
14 export ORAENV_ASK=YES
15
16 dbstart $ORACLE_HOME
17

[oracle@localhost scripts]$ vim stop.sh
[oracle@localhost scripts]$ cat -n stop.sh
 1 #!/bin/bash
 2
 3 export TMP=/tmp
 4 export TMPDIR=$TMP
 5
 6 export ORACLE_HOME=/u01/app/oracle/product/19.3.0/dbhome_1
 7 export ORACLE_SID=nicho
 8
 9 export PATH=/usr/sbin:/usr/local/bin:$PATH
10 export PATH=$ORACLE_HOME/bin:$PATH
11
12 export ORAENV_ASK=NO
13 source oraenv
14 export ORAENV_ASK=YES
15
16 dbshut $ORACLE_HOME
17

[oracle@localhost scripts]$
```

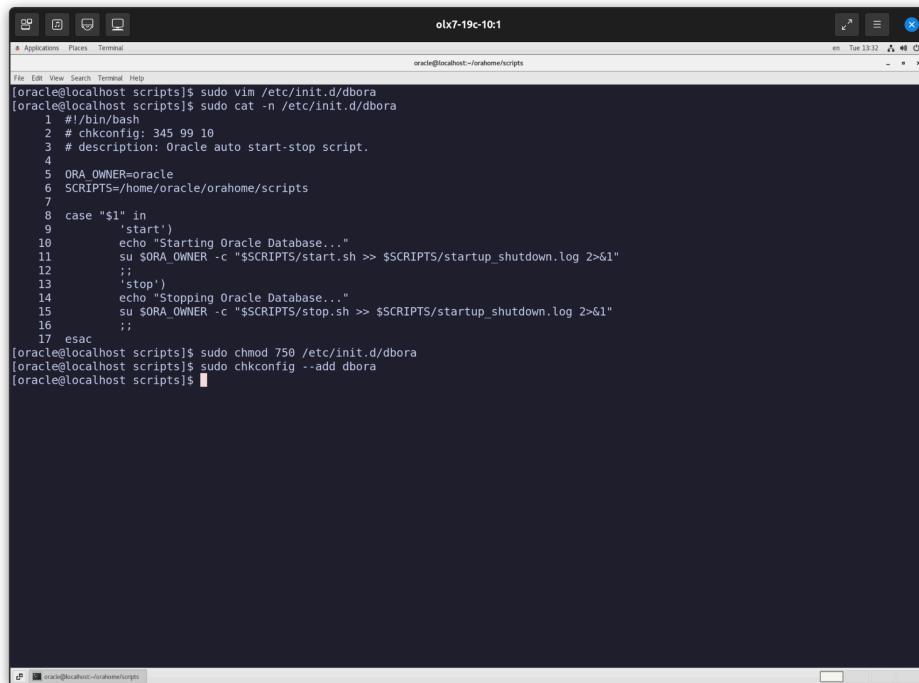
Figura 3.2. Contenido de los scripts **start.sh** y **stop.sh**



```
oracle@localhost:~/orahome/scripts$ chmod u+x ./home/oracle/orahome/scripts/*.sh
[oracle@localhost scripts]$ ls -l *.sh
-rwxr--r-- 1 oracle oinstall 292 Oct 29 13:21 start.sh
-rwxr--r-- 1 oracle oinstall 291 Oct 29 13:22 stop.sh
[oracle@localhost scripts]$ ./start.sh
The Oracle base remains unchanged with value /u01/app/oracle
Processing Database Instance "nicho": log file /u01/app/oracle/product/19.3.0/dbhome_1/rdbms/log/startup.log
[oracle@localhost scripts]$ ps -ef | grep pmon
oracle 25765 1 0 13:29 ? 00:00:00 ora_pmon nicho
oracle 27299 4401 0 13:29 pts/0 00:00:00 grep --color=auto pmon
[oracle@localhost scripts]$ ./stop.sh
The Oracle base remains unchanged with value /u01/app/oracle
Processing Database Instance "nicho": log file /u01/app/oracle/product/19.3.0/dbhome_1/rdbms/log/shutdown.log
[oracle@localhost scripts]$ ps -ef | grep pmon
oracle 27599 4401 0 13:29 pts/0 00:00:00 grep --color=auto pmon
[oracle@localhost scripts]$
```

Figura 3.3. Prueba manual de que los scripts funcionan correctamente

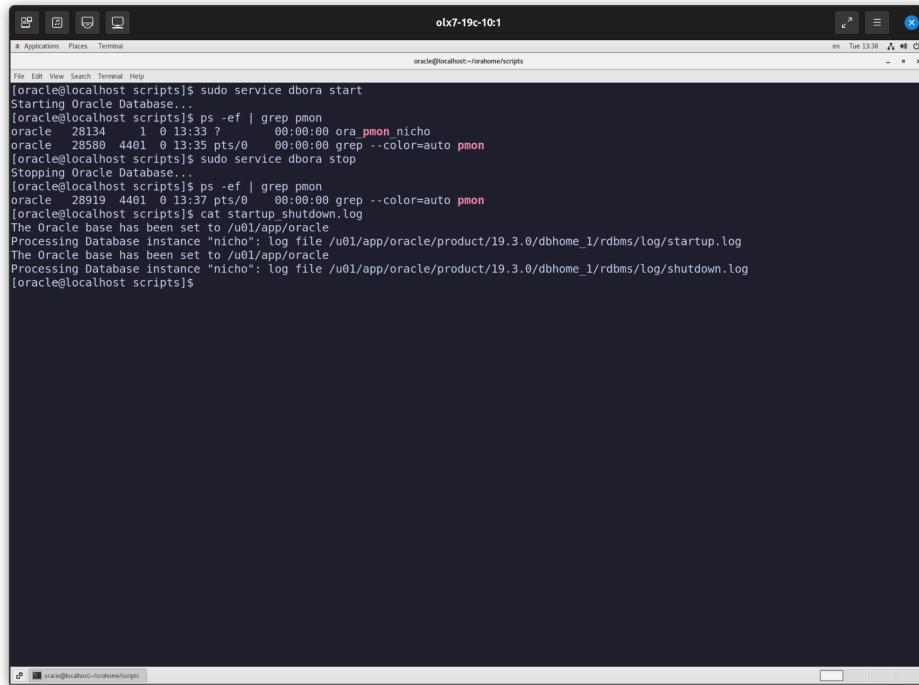
Despues, se añade el servicio de **dbora** con **chkconfig**. Aquí se usan los scripts creados anteriormente.



```
oracle@localhost:~/orahome/scripts$ sudo vim /etc/init.d/dbora
[oracle@localhost scripts]$ sudo cat -n /etc/init.d/dbora
 1 #!/bin/bash
 2 # chkconfig: 345 99 10
 3 # description: Oracle auto start-stop script.
 4
 5 ORA_OWNER=oracle
 6 SCRIPTS=/home/oracle/orahome/scripts
 7
 8 case "$1" in
 9   "start")
10     echo "Starting Oracle Database..."
11     su $ORA_OWNER -c "$SCRIPTS/start.sh >> $SCRIPTS/startup_shutdown.log 2>&1"
12     ;;
13   "stop")
14     echo "Stopping Oracle Database..."
15     su $ORA_OWNER -c "$SCRIPTS/stop.sh >> $SCRIPTS/startup_shutdown.log 2>&1"
16     ;;
17 esac
[oracle@localhost scripts]$ sudo chmod 750 /etc/init.d/dbora
[oracle@localhost scripts]$ sudo chkconfig --add dbora
[oracle@localhost scripts]$
```

Figura 3.4. Configuración del servicio **dbora**

Finalmente, se verifica que el servicio pueda ser iniciado y pausado correctamente. También, se verifica que el log contenga el resultado de ejecutar las operaciones de **start** y **stop**.

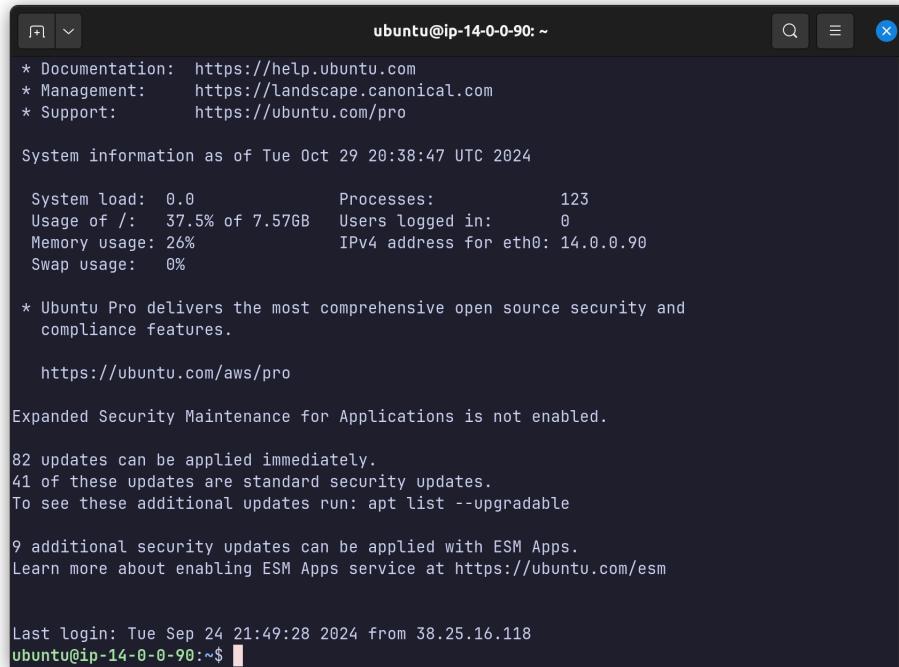


```
[oracle@localhost scripts]$ sudo service dbora start
Starting Oracle Database ...
[oracle@localhost scripts]$ ps -ef | grep pmon
oracle  28134      1  0 13:33 ?    00:00:00 ora_pmon nicho
oracle  28580  4401  0 13:35 pts/0    00:00:00 grep --color=auto pmon
[oracle@localhost scripts]$ sudo service dbora stop
Stopping Oracle Database ...
[oracle@localhost scripts]$ ps -ef | grep pmon
oracle  28919  4401  0 13:37 pts/0    00:00:00 grep --color=auto pmon
[oracle@localhost scripts]$ cat startup_shutdown.log
The Oracle base has been set to /u01/app/oracle
Processing Database instance "nicho": log file /u01/app/oracle/product/19.3.0/dbhome_1/rdbms/log/startup.log
The Oracle base has been set to /u01/app/oracle
Processing Database instance "nicho": log file /u01/app/oracle/product/19.3.0/dbhome_1/rdbms/log/shutdown.log
[oracle@localhost scripts]$
```

Figura 3.5. Prueba de que el servicio funciona correctamente

PREGUNTA 4

En primer lugar, se realiza la conexión con la instancia de Ubuntu.



```
ubuntu@ip-14-0-0-90: ~
* Documentation: https://help.ubuntu.com
* Management: https://landscape.canonical.com
* Support: https://ubuntu.com/pro

System information as of Tue Oct 29 20:38:47 UTC 2024

System load: 0.0          Processes: 123
Usage of /: 37.5% of 7.57GB  Users logged in: 0
Memory usage: 26%          IPv4 address for eth0: 14.0.0.90
Swap usage: 0%

* Ubuntu Pro delivers the most comprehensive open source security and
  compliance features.

https://ubuntu.com/aws/pro

Expanded Security Maintenance for Applications is not enabled.

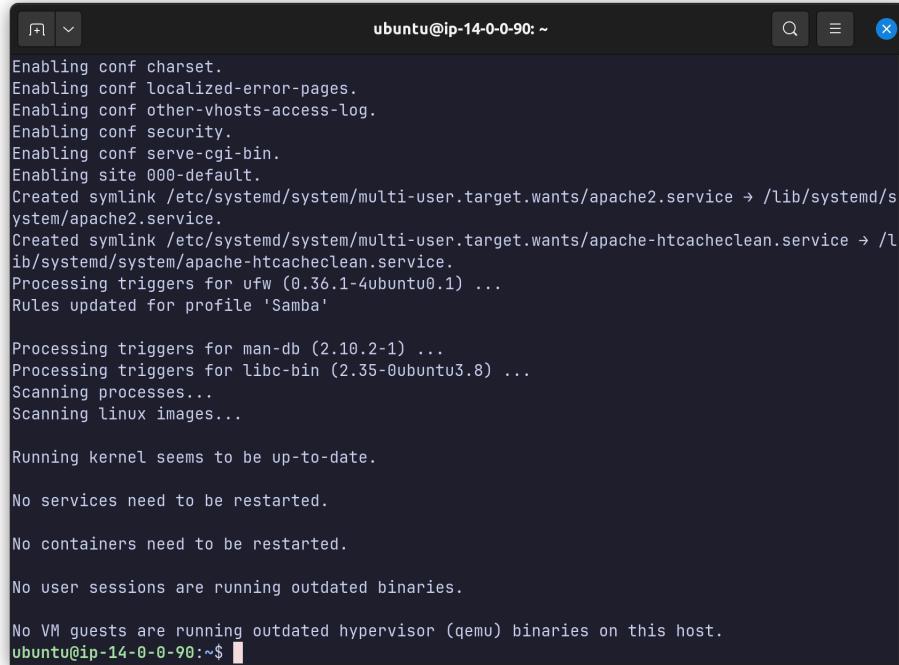
82 updates can be applied immediately.
41 of these updates are standard security updates.
To see these additional updates run: apt list --upgradable

9 additional security updates can be applied with ESM Apps.
Learn more about enabling ESM Apps service at https://ubuntu.com/esm

Last login: Tue Sep 24 21:49:28 2024 from 38.25.16.118
ubuntu@ip-14-0-0-90:~$
```

Figura 4.1. Acceso a la instancia Ubuntu con `ssh`

Luego, se instalan los paquetes necesarios para obtener el servidor Apache.



```
ubuntu@ip-14-0-0-90: ~
Enabling conf charset.
Enabling conf localized-error-pages.
Enabling conf other-vhosts-access-log.
Enabling conf security.
Enabling conf serve-cgi-bin.
Enabling site 000-default.
Created symlink /etc/systemd/system/multi-user.target.wants/apache2.service → /lib/systemd/system/apache2.service.
Created symlink /etc/systemd/system/multi-user.target.wants/apache-htcacheclean.service → /lib/systemd/system/apache-htcacheclean.service.
Processing triggers for ufw (0.36.1-4ubuntu0.1) ...
Rules updated for profile 'Samba'

Processing triggers for man-db (2.10.2-1) ...
Processing triggers for libc-bin (2.35-0ubuntu3.8) ...
Scanning processes...
Scanning linux images...

Running kernel seems to be up-to-date.

No services need to be restarted.

No containers need to be restarted.

No user sessions are running outdated binaries.

No VM guests are running outdated hypervisor (qemu) binaries on this host.
ubuntu@ip-14-0-0-90:~$
```

Figura 4.2. Instalación de `apache2` y `openssl`

Después, se habilita e inicia la ejecución del servicio.

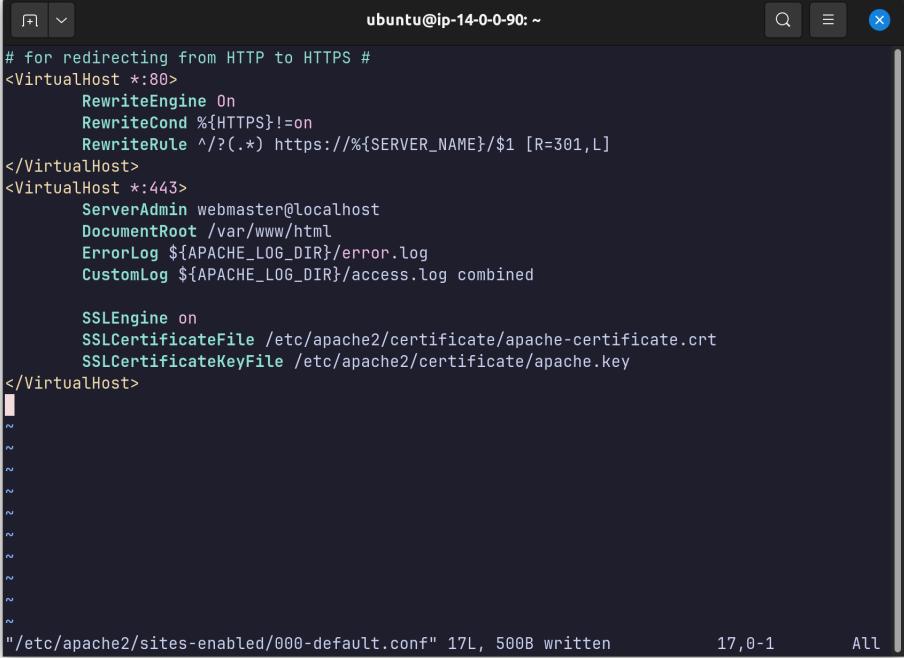
```
[+] ~ ubuntu@ip-14-0-0-90:~$ sudo a2enmod ssl
Considering dependency setenvif for ssl:
Module setenvif already enabled
Considering dependency mime for ssl:
Module mime already enabled
Considering dependency socache_shmcb for ssl:
Enabling module socache_shmcb.
Enabling module ssl.
See /usr/share/doc/apache2/README.Debian.gz on how to configure SSL and create self-signed certificates.
To activate the new configuration, you need to run:
  systemctl restart apache2
ubuntu@ip-14-0-0-90:~$ sudo systemctl restart apache2
ubuntu@ip-14-0-0-90:~$ sudo a2enmod ssl
Considering dependency setenvif for ssl:
Module setenvif already enabled
Considering dependency mime for ssl:
Module mime already enabled
Considering dependency socache_shmcb for ssl:
Module socache_shmcb already enabled
Module ssl already enabled
ubuntu@ip-14-0-0-90:~$ sudo a2enmod rewrite
Enabling module rewrite.
To activate the new configuration, you need to run:
  systemctl restart apache2
ubuntu@ip-14-0-0-90:~$ sudo systemctl restart apache2
ubuntu@ip-14-0-0-90:~$
```

Figura 4.3. Habilitación del servicio apache2

A continuación, se genera la clave privada siguiendo las indicaciones de la documentación brindada. Los valores usados son ficticios.

Figura 4.4. Resultado de la generación de una clave privada

Tras esto, se configura el archivo `000-default.conf` de manera que permita redireccionar de HTTP a HTTPS cuando se acceda a la dirección IP del servidor.



```
# for redirecting from HTTP to HTTPS #
<VirtualHost *:80>
    RewriteEngine On
    RewriteCond %{HTTPS}!=on
    RewriteRule ^/(.*) https:// %{SERVER_NAME}/$1 [R=301,L]
</VirtualHost>
<VirtualHost *:443>
    ServerAdmin webmaster@localhost
    DocumentRoot /var/www/html
    ErrorLog ${APACHE_LOG_DIR}/error.log
    CustomLog ${APACHE_LOG_DIR}/access.log combined

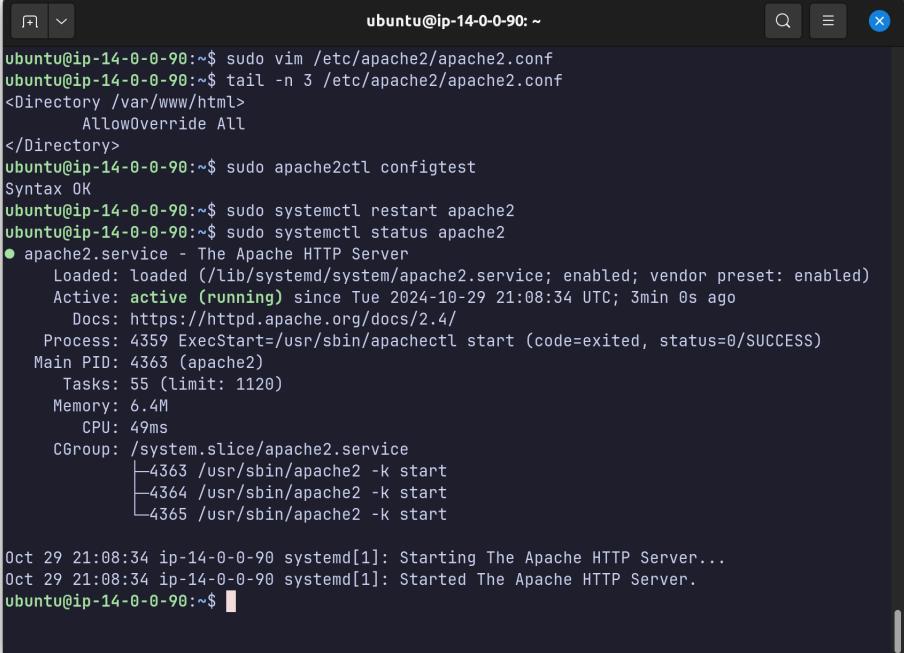
    SSLEngine on
    SSLCertificateFile /etc/apache2/certificate/apache-certificate.crt
    SSLCertificateKeyFile /etc/apache2/certificate/apache.key
</VirtualHost>

```

"./etc/apache2/sites-enabled/000-default.conf" 17L, 500B written 17,0-1 All

Figura 4.5. Configuración que permite redireccionar de HTTP a HTTPS

Luego se reinicia y se vuelve a habilitar el servicio.



```
ubuntu@ip-14-0-0-90:~$ sudo vim /etc/apache2/apache2.conf
ubuntu@ip-14-0-0-90:~$ tail -n 3 /etc/apache2/apache2.conf
<Directory /var/www/html>
    AllowOverride All
</Directory>
ubuntu@ip-14-0-0-90:~$ sudo apache2ctl configtest
Syntax OK
ubuntu@ip-14-0-0-90:~$ sudo systemctl restart apache2
ubuntu@ip-14-0-0-90:~$ sudo systemctl status apache2
● apache2.service - The Apache HTTP Server
   Loaded: loaded (/lib/systemd/system/apache2.service; enabled; vendor preset: enabled)
   Active: active (running) since Tue 2024-10-29 21:08:34 UTC; 3min 0s ago
     Docs: https://httpd.apache.org/docs/2.4/
 Process: 4359 ExecStart=/usr/sbin/apachectl start (code=exited, status=0/SUCCESS)
 Main PID: 4363 (apache2)
    Tasks: 55 (limit: 1120)
   Memory: 6.4M
      CPU: 49ms
     CGroup: /system.slice/apache2.service
             └─4363 /usr/sbin/apache2 -k start
                 ├─4364 /usr/sbin/apache2 -k start
                 ├─4365 /usr/sbin/apache2 -k start

Oct 29 21:08:34 ip-14-0-0-90 systemd[1]: Starting The Apache HTTP Server...
Oct 29 21:08:34 ip-14-0-0-90 systemd[1]: Started The Apache HTTP Server.
ubuntu@ip-14-0-0-90:~$
```

Figura 4.6. Habilitación del servicio

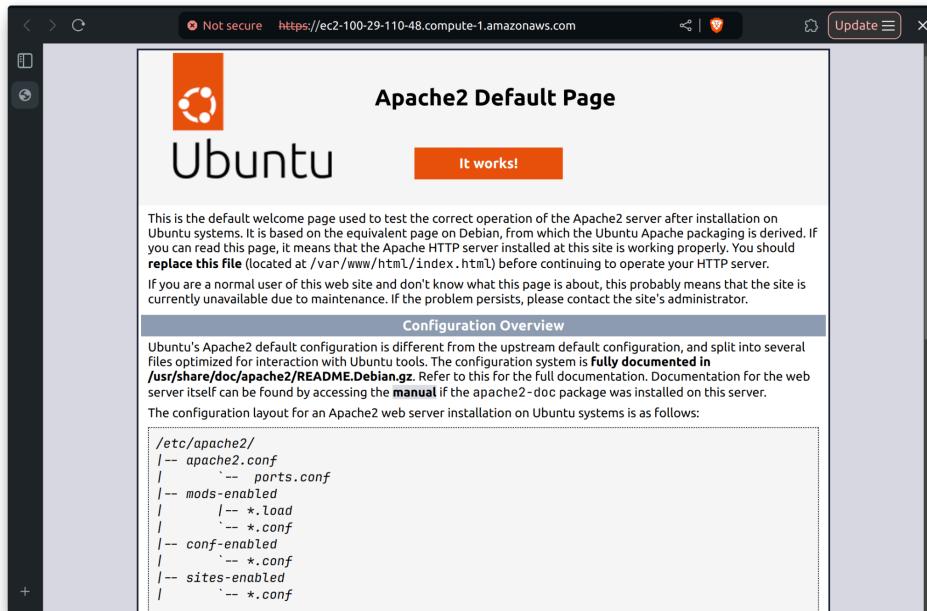


Figura 4.7. Servidor levantado con IP 100.29.110.48

Se opta por usar el servicio de [noip.com](#) en lugar del de [cloudns.net](#). Este permite tener el dominio gratuito para el hosteo de nuestra dirección IP.

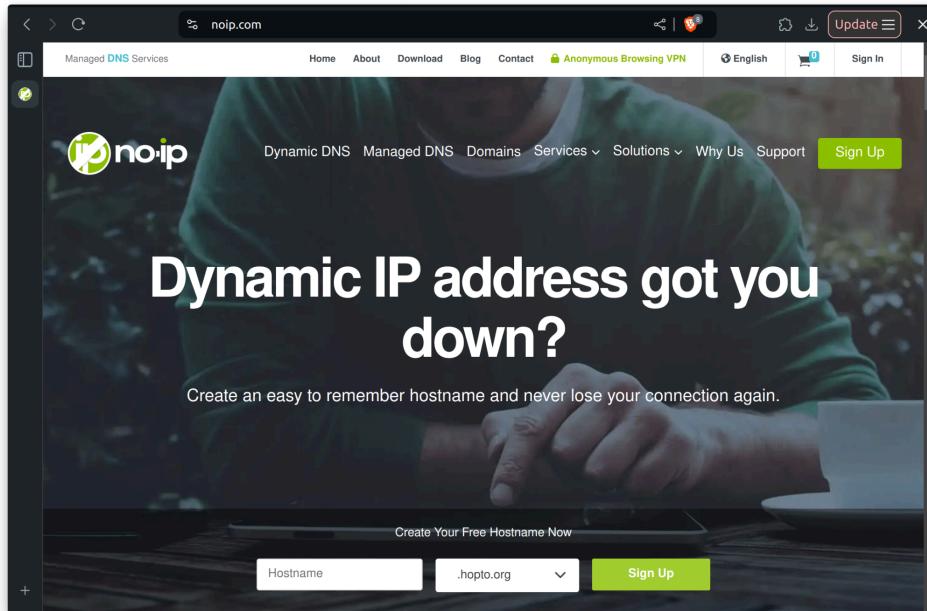


Figura 4.8. Página principal de [noip.com](#)

A continuación se muestra el hostname lab04o20211866@hopto.org, el cual está configurado para apuntar a la dirección IP de nuestra instancia.

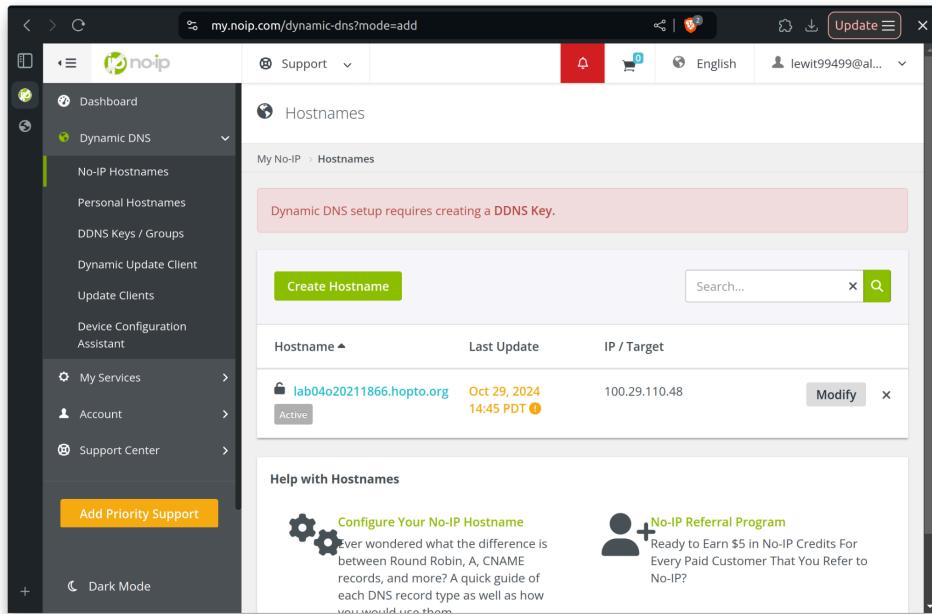


Figura 4.9. Hostname apuntando a la dirección pública de la instancia

Se usa `nslookup` para consultar la información sobre el dominio usado.

```
ubuntu@ip-14-0-0-90:~$ nslookup lab04o20211866.hopto.org
Server:      127.0.0.53
Address:     127.0.0.53#53

Non-authoritative answer:
Name:   lab04o20211866.hopto.org
Address: 100.29.110.48

ubuntu@ip-14-0-0-90:~$
```

Figura 4.10. Verificación de que se logró enlazar la dirección IP con `nslookup`

Luego, se obtiene el certificado de seguridad SSL con la aplicación `cerbot`. Se elige la opción 2: HTTPS.

```

Certificate is saved at: /etc/letsencrypt/live/lab04o20211866.hopto.org/fullchain.pem
Key is saved at:      /etc/letsencrypt/live/lab04o20211866.hopto.org/privkey.pem
This certificate expires on 2025-01-27.
These files will be updated when the certificate renews.
Certbot has set up a scheduled task to automatically renew this certificate in the background.

Deploying certificate

We were unable to find a vhost with a ServerName or Address of lab04o20211866.hopto.org.
Which virtual host would you like to choose?

1: 000-default.conf | | Enabled
2: 000-default.conf | | HTTPS | Enabled

Select the appropriate number [1-2] then [enter] (press 'c' to cancel): 2
Successfully deployed certificate for lab04o20211866.hopto.org to /etc/apache2/sites-enabled/000-default.conf
Added an HTTP->HTTPS rewrite in addition to other RewriteRules; you may wish to check for overall consistency.
Congratulations! You have successfully enabled HTTPS on https://lab04o20211866.hopto.org

If you like Certbot, please consider supporting our work by:
* Donating to ISRG / Let's Encrypt: https://letsencrypt.org/donate
* Donating to EFF: https://eff.org/donate-le

```

Figura 4.11. Obtención del certificado gratuito de seguridad

Finalmente, se puede acceder al sitio desde el navegador y revisar que posee un certificado válido generado por Let's Encrypt. También se puede ver que estará vigente hasta el 27 de enero del 2025.

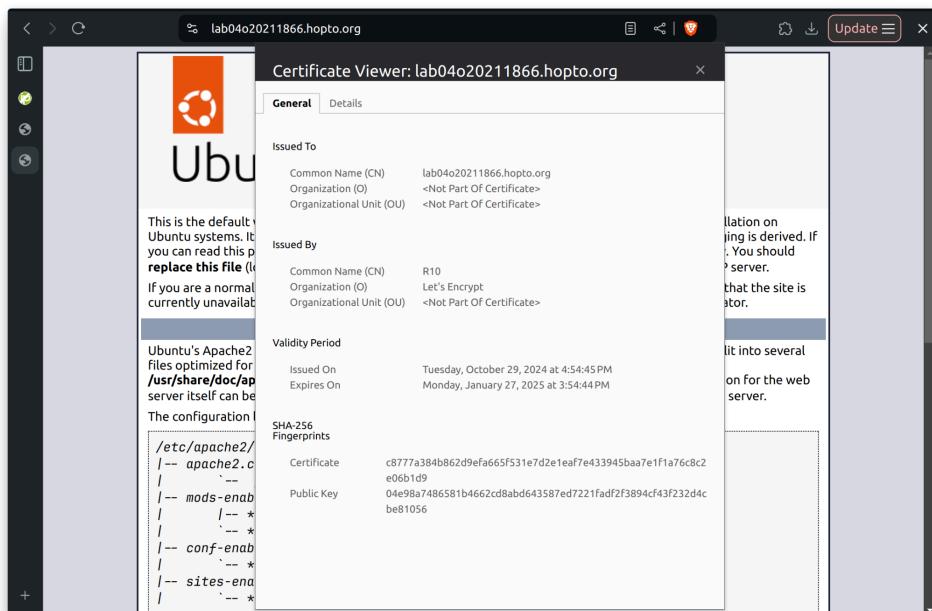
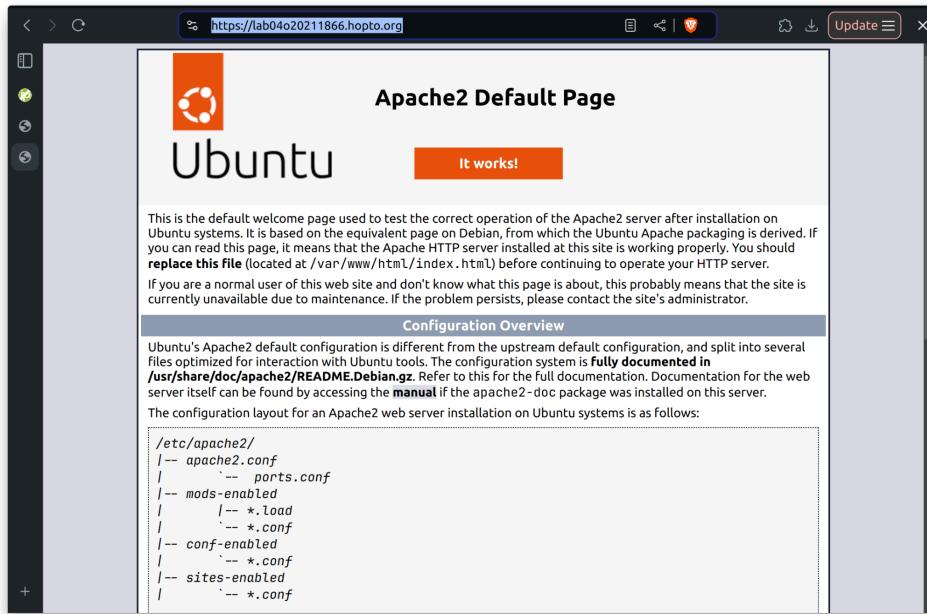


Figura 4.12. Certificado de seguridad visto desde el navegador

También se puede observar cómo al acceder a la página con HTTP, esta nos redirige a HTTPS, razón por la que ya no se muestran advertencias de seguridad.



```
/etc/apache2/  
|-- apache2.conf  
|   '-- ports.conf  
|-- mods-enabled  
|   '-- *.load  
|   '-- *.conf  
|-- conf-enabled  
|   '-- *.conf  
|-- sites-enabled  
|   '-- *.conf
```

Figura 4.13. Comprobación de que el servidor tiene soporte para HTTPS