

# Gestión de un Servidor OpenBSD

Nicolás A. Ortega Froysa

2 de noviembre de 2022

# Índice

<b>1. Introducción</b>	<b>2</b>
<b>2. Configuración Inicial</b>	<b>2</b>
<b>3. Instalación de Paquetes</b>	<b>3</b>
<b>4. Configuración de Servicios</b>	<b>4</b>
4.1. Nginx . . . . .	4
4.2. MariaDB . . . . .	5
4.3. Servicio RSync . . . . .	6
<b>5. Conclusión</b>	<b>6</b>
<b>6. Derechos de Autor y Licencia</b>	<b>7</b>

# 1. Introducción

En esta práctica vamos a ver cómo es la administración de sistema de un servidor de *OpenBSD*. Los objetivos son los siguientes:

- Aprender los comandos básicos del sistema.
- Ver en qué se diferencia OpenBSD de un sistema Linux.
- Ver cómo cambia la carga del sistema mientras más servicios vayamos proveyendo.

La instalación en sí no forma parte de este documento, pero generalmente el instalador guiado es bastante intuitivo.

Lo que caracteriza a OpenBSD especialmente, respecto a otros sistemas UNIX (y otros sistemas que hemos visto en clase) son dos cosas. Primero, que no es una distribución de Linux, sino de BSD. Son sistemas completamente distintos, aunque siguen (generalmente) una filosofía/arquitectura similar, basándose los dos en UNIX. Segundo, que OpenBSD, entre los otros sistemas BSD, es el más especializado en seguridad, hasta el punto de sacrificar cualquier otro aspecto de la computación a favor siempre de la seguridad. Es por este motivo que deshabilitaron en su momento el *hyper-threading* de los procesadores Intel por miedo a *Spectre*.<sup>1</sup>

# 2. Configuración Inicial

Generalmente la configuración de OpenBSD ha sido bastante simple, y la mayoría de las opciones están en su valor por defecto. El teclado – i.e. keymap – usado es el estadounidense, simplemente porque para la informática es la más cómoda. La máquina lleva el *hostname* de **puffy-nortega**.

En cuanto a los usuarios, existe un usuario *root* con contraseña **toor**, y otro usuario *nicolas* con contraseña **nicolas**. Durante la instalación se ha configurado que no sea posible acceder a *root* por SSH, ya que esto hace que el sistema sea más vulnerable – más todavía con este tipo de contraseña.

El disco se ha particionado con una sola partición para simplificar este proceso de instalación. Se podría instalar con más particiones, repartiendo así los datos y haciendo más fácil el proceso de hacer copias de respaldo (que sean de una sola partición), pero eso no entra dentro de los objetivos de esta práctica.

---

<sup>1</sup>[https://www.theregister.com/2018/06/20/openbsd\\_disables\\_intels\\_hyperthreading/](https://www.theregister.com/2018/06/20/openbsd_disables_intels_hyperthreading/)

```

load averages: 0.00, 0.00, 0.00          puffly-nortega.my.domain 19:28:38
32 processes: 31 idle, 1 on processor      up 0 days 00:37:48
CPU:  0.0% user,  0.0% nice,  0.0% sys,  0.0% spin,  0.0% intr, 100% idle
Memory: Real: 31M/905M act/tot Free: 977M Cache: 392M Swap: 0K/0K

  PID USERNAME  PRI  NICE   SIZE   RES STATE   WAIT   TIME   CPU COMMAND
    1 root         10    0   656K   628K idle    wait   0:01  0.00% init
27022 _pflogd      4    0   772K   1576K sleep   bpf     0:00  0.00% _pflogd
25429 _ntp         2   -20  1188K  3020K sleep   kqread  0:00  0.00% ntpd
77046 root        18    0   956K   936K idle    sigsusp 0:00  0.00% ksh
 1542 root        28    0   960K   2184K onproc  -       0:00  0.00% top
 7358 _syslogd     2    0  1228K  1516K sleep   kqread  0:00  0.00% syslogd
42906 root        2   -20  1076K  1744K idle    kqread  0:00  0.00% ntpd
92240 root        2    0   876K   1516K idle    kqread  0:00  0.00% cron
47390 root        2    0   848K   1412K idle    kqread  0:00  0.00% dhcpleased
44031 _ntp         2    0  1076K  2736K idle    kqread  0:00  0.00% ntpd
98759 _smtpd       2    0  1936K  4720K idle    kqread  0:00  0.00% smtpd
89999 _smtpq       2    0  1652K  3396K idle    kqread  0:00  0.00% smtpd
74067 root        2    0   760K   2472K idle    netio   0:00  0.00% syslogd
  288 _smtpd       2    0  1452K  3224K idle    kqread  0:00  0.00% smtpd
32276 root        2    0   524K   1112K idle    kqread  0:00  0.00% resolvd
28058 _smtpd       2    0  1660K  3448K idle    kqread  0:00  0.00% smtpd
75849 dhcp       2    0   864K   1684K idle    kqread  0:00  0.00% dhcpleased
16431 _smtpd       2    0  1524K  3296K idle    kqread  0:00  0.00% smtpd
31197 _smtpd       2    0  1452K  3244K idle    kqread  0:00  0.00% smtpd

```

Figura 1: Carga inicial de OpenBSD.

Finalmente, en OpenBSD se dividen los archivos de sistema en diferentes *sets*. Éstos se pueden usar para instalar diferentes componentes del sistema, desde el *kernel* hasta algunos juegos de consola. Éstos están los *sets* instalados:

- `bsd`: *kernel* de BSD (**obligatorio**).
- `bsd.mp`: *kernel* multi-procesador.
- `bsd.rd`: *kernel* de uso en memoria.
- `base72.tgz`: sistema base o *userland* (**obligatorio**).
- `comp72.tgz`: colección de compiladores.
- `man72.tgz`: páginas de manual.

Al inicializar la máquina lleva una carga de sistema extremadamente baja, ya que está corriendo tan sólo lo mínimo (figura 1).

### 3. Instalación de Paquetes

Para administrar paquetes en OpenBSD se hace uso de las herramientas de paquete de OpenBSD. Los comandos se pueden reconocer porque llevan la patrón `pkg_*`. En particular, las que nos interesan son tres:

- `pkg_add`: instalar y actualizar.
- `pkg_delete`: eliminar/desinstalar.

- `pkg_info`: buscar información.

Para instalar los paquetes que nos interesan vamos a correr el comando siguiente:

```
root# pkg_add nginx mariadb-server mariadb-client \  
rsync
```

## 4. Configuración de Servicios

El control de los servicios en OpenBSD se hace con el comando `rcctl`, ya que OpenBSD no usa Systemd. Generalmente se usa de una forma muy similar a Systemd, usando los subcomandos siguientes:

- `start`: iniciar un servicio.
- `stop`: parar un servicio.
- `enable`: habilitar un servicio para iniciarse al arrancar la máquina.
- `disable`: deshabilitar un servicio para que no se inicie cuando arranque la máquina.
- `ls`: mostrar una lista de los servicios.

### 4.1. Nginx

<b>Nombre</b>	Nginx
<b>Descripción</b>	Un servidor web moderno.
<b>Dir. Config.</b>	/etc/nginx/
<b>Dir. Logs</b>	/etc/nginx/logs/

Cuadro 1: Datos sobre Nginx.

En nuestra configuración de Nginx vamos a crear una página simple que se pueda acceder desde fuera en el puerto 80. Por defecto, Nginx en OpenBSD usa el directorio `/var/www/htdocs/` como raíz del servidor. Aquí añadiremos un archivo `index.html` para que responda con un código 200 OK. Iniciamos el servicio (y lo habilitamos) con el comando siguiente:

```
root# rcctl enable nginx  
root# rcctl start nginx  
nginx(ok)
```

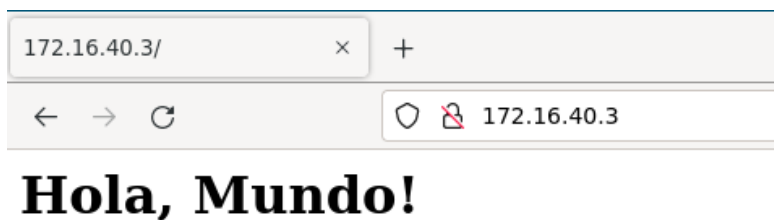


Figura 2: Prueba de Nginx.

```

load averages:  0.49,  0.27,  0.11                puffu-nortega.my.domain 18:1
38 processes: 37 idle, 1 on processor              up 0 days 00:3
CPU states: 13.7% user,  0.0% nice, 50.8% sys,  0.0% spin, 14.7% intr, 20.8% idle
Memory: Real: 36M/993M act/tot Free: 969M Cache: 392M Swap: 0K/0K
  PID USERNAME PRI NICE  SIZE  RES STATE   WAIT   TIME   CPU COMMAND
61856 www        2    0 2020K 2584K sleep   kqread 0:47 44.07% nginx
1 root         0    0  652K  824K idle     wait    0:01  0.00% init
49721 nicolas   2    0 4395K 3240K sleep   kqread 0:00  0.00% sshd
19220 pflogd    4    0  760K 1572K sleep   bpif    0:00  0.00% pflogd
68770 ntp        2 -20 1204K 2880K sleep   kqread 0:00  0.00% ntpd
12450 root       18    0  956K  948K idle     sigsusp 0:00  0.00% ksh
98929 root       28    0 1488K 2732K onproc   -       0:00  0.00% top
78180 root        2    0 1164K 3900K idle     kqread 0:00  0.00% sshd
62801 root        3    0  960K  932K idle     ttuin  0:00  0.00% ksh

```

Figura 3: *Load* resultante de una prueba con *wrk*.

Ahora, si probamos acceder a nuestro servidor, veremos el archivo que hemos creado (figura 2).

A este servidor le podemos hacer un *stress test* utilizando la herramienta *wrk*, que sirve para hacer *benchmarking* de servicios HTTP. Desde un cliente corremos el comando siguiente, que nos hará una prueba de 50 conexiones paralelas, sobre 10 hilos, durante 20 segundos. Vemos que el *load* aumenta de forma considerable (figura 3).

```
$ wrk -c 50 -t 10 -d 20 http://172.16.40.3/
```

## 4.2. MariaDB

<b>Nombre</b>	MariaDB
<b>Descripción</b>	Sistema de gestión de bases de datos.
<b>Archivo Conf.</b>	/etc/my.cnf

Cuadro 2: Datos sobre MariaDB.

MariaDB requiere de una configuración inicial más manual, así que en primer lugar se tendrá que inicializar (y habilitar) el servicio, que se denomina *mysqld* – esto es porque MariaDB tiene como objetivo ser un reemplazo *in-place* de MySQL.

```
root# rcctl enable mysqld
root# rcctl start mysqld
mysqld(ok)
```

Posteriormente lo configuramos usando el comando `mysql_secure_install`. Esto instalará las tablas dentro del directorio `/var/mysql/` y podremos definir la contraseña de `root`, que será `Rooty` (y algunas opciones de MariaDB). Podemos acceder a este base de datos usando el comando siguiente desde el usuario `root`, e introduciendo la contraseña que hemos asignado:

```
puffy-nortega# mysql -u root -p
Enter password:
Welcome to the MariaDB monitor.  Commands end with ; or \g.
Your MariaDB connection id is 13
Server version: 10.9.3-MariaDB OpenBSD port:
mariadb-server-10.9.3v1
```

```
Copyright (c) 2000, 2018, Oracle, MariaDB Corporation Ab and
others.
```

```
Type 'help;' or '\h' for help. Type '\c' to clear the current
input statement.
```

```
MariaDB [(none)]>
```

### 4.3. Servicio RSync

## 5. Conclusión

## **6. Derechos de Autor y Licencia**

Copyright © 2022 Nicolás A. Ortega Froya <nicolas@ortegas.org>

Este documento se distribuye bajo los términos y condiciones de la licencia Creative Commons Attribution No Derivatives 4.0 International.