



TALLER PRÁCTICO WPO – Meetup WP Valencia

28/11/2015

Fernando Serer
@fserer

twitter: #wpvalencia
@_wpvalencia

INDICE

1. CREACION SERVIDOR CLOUD
2. CONFIGURACION SISTEMA BASE (LINUX)
3. INSTALACION STACK (LEMP)
4. INSTALACION WORDPRESS
5. OPTIMIZACIONES WORDPRESS

1. CREACION SERVIDOR CLOUD

Digital Ocean

Linode

Amazon Web Services

etc...

Como ejemplo, lanzamos una instancia de 5\$ en Digital Ocean basada en CentOS (distribución basada en Red Hat)

2. CONFIGURACION SISTEMA BASE (LINUX)

Antes de empezar...

Algunos comandos genéricos:

```
w  
free -m  
df -h  
cat  
vi  
nano  
less  
ifconfig
```

```
ls  
ll  
cd
```

1.1. Actualización sistema

```
yum update  
yum -y update
```

1.2. Fecha / Hora

```
rm /etc/localtime  
ln -s /usr/share/zoneinfo/Europe/Madrid /etc/localtime
```

1.3. Añadir repositorios adicionales y activarlos

Instalar

EPEL → <https://fedoraproject.org/wiki/EPEL/es>
yum install epel-release

REMI → <http://rpms.famillecollet.com/>
rpm -ivh <http://rpms.famillecollet.com/enterprise/remi-release-7.rpm>

Activarlos

```
cd /etc/yum.repos.d/  
vi remi.repo  
nano remi.repo  
less remi.repo  
cat epel.repo
```

```
activar REMI y PHP  
enabled 1  
yum -y update
```

3. INSTALACION STACK (LEMP)

3.1. Instalacion software

instalamos por separado y con confirmación:

```
yum install phpMyAdmin
```

```
yum install nginx
```

```
yum install mariadb-server
```

```
yum install php-fpm
```

```
yum install unzip
```

o todo junto sin confirmación:

```
yum -y install phpMyAdmin nginx mariadb-server php-fpm unzip
```

3.2. Iniciar servicios

```
systemctl start nginx  
systemctl start mariadb  
systemctl start php-fpm
```

3.3. Directorio Web

```
/usr/share/nginx/html/
```

3.4. Enlace phpMyAdmin

```
mkdir old  
mv * old  
ln -s /usr/share/phpMyAdmin/ BD
```

3.5. Password de root en MariaDB

```
mysqladmin -u root password MeetUP49
```

3.6. Configurar NGINX para usar PHP-FPM

fichero: /etc/nginx/nginx.conf

bloque server {}

debajo de root añadimos fichero por defecto

```
index    index.php;
```

añadir location para gestión de PHP

```
location ~ \.php$ {
    fastcgi_pass 127.0.0.1:9000;
    fastcgi_index index.php;
    fastcgi_param SCRIPT_FILENAME $document_root$fastcgi_script_name;
    include fastcgi_params;
}
```

Reiniciamos nginx:

```
systemctl restart nginx
```

3.7. Cargamos phpMyAdmin

<http://146.185.144.199/BD/>

usuario: root

pass: password puesto

4. INSTALACION WORDPRESS

4.1. Bases de datos

Crear base de datos: wp_meetup
cotejamiento: utf8_general_ci

4.2. Privilegios de usuario

Nuevo: Agregar usuario

Nombre: usuario_wp
Servidor: local (localhost)
Contraseña: Generar y copiar
Privilegios globales: Nada
Continuar
Base de datos: wp_meetup
Continuar

4.3. Descargar WordPress

```
https://es.wordpress.org/  
cd /tmp/  
wget https://es.wordpress.org/wordpress-4.3.1-es_ES.zip
```

```
unzip wordpress-4.3.1-es_ES.zip
```

```
cd wordpress  
mv * /usr/share/nginx/html/  
chown apache.apache -R /usr/share/nginx/html/
```

4.4. Activar Permalinks

En el bloque `location / {}`
añadir línea

```
try_files $uri $uri/ /index.php?$args;
```

```
systemctl restart nginx  
systemctl reload nginx
```

4.5. Ajustes extra

```
/etc/php.ini  
upload_max_filesize = 8M
```

```
/etc/nginx/nginx.conf (bloque http)  
client_max_body_size 8M;
```

5. OPTIMIZACION WORDPRESS

5.1. Añadir número de queries y tiempo de ejecución de la página

```
footer.php del theme activo  
<!-- <?php echo get_num_queries(); ?> consultas en <?php timer_stop(1); ?> segundos -->
```

5.2. Activamos PHP OP CODE CACHE

```
yum install php-opcache
```

```
fichero de configuración: /etc/php.d/10-opcache.ini
```

```
systemctl restart php-fpm
```

5.3. Activar Query Cache

```
fichero: /etc/my.cnf.d/server.cnf
```

```
query_cache_type = 1  
query_cache_size = 32M  
query_cache_limit = 2M
```

5.4. Activar otros parámetros de MySQL

```
key_buffer = 16M  
sort_buffer_size = 4M  
read_buffer_size = 4M  
thread_stack = 128K  
table_cache = 128  
thread_cache = 256  
thread_concurrency = 2  
myisam_sort_buffer_size = 1M  
tmp_table_size = 12M  
max_heap_table_size = 12M
```

```
systemctl restart mariadb
```

5.5. Activar compresión nginx

```
fichero: /etc/nginx/nginx.conf (bloque http)
```

```
gzip on;  
gzip_static on;  
gzip_proxied any;  
gzip_comp_level 9;  
gzip_min_length 0;  
gzip_types text/plain text/css application/javascript text/xml application/xml+rss  
image/svg+xml;
```


5.6. Activar cache de navegador

añadir location en el bloque server {}

```
location ~* \.(ogg|ogv|svg|svgz|eot|otf|woff|mp4|ttf|css|rss|atom|js|jpg|jpeg|gif|png|ico|zip|tgz|gz|rar|bz2|doc|xls|exe|ppt|tar|mid|midi|wav|bmp|rtf)$ {  
    expires max;  
}
```

5.7. Activar W3 Total Cache, Autooptimize y ewww image optimizer y configurarlos

en el bloque server {}

```
include /path/to/wordpress/installation/nginx.conf;
```

(Rewrite Rules → Menú Install W3 Total Cache)

5.8. Finalizamos activando servicios en el arranque y reiniciando

```
systemctl enable nginx
```

```
systemctl enable mariadb
```

```
systemctl enable php-fpm
```

```
reboot
```

Varios extras fuentes Web:

FOUT (Flash of Unstyled Text)

FOIT (Flash of Invisible Text)

FOFT (Flash of Faux Text)

<https://css-tricks.com/fout-foit-foft/>

<https://github.com/typekit/webfontloader>

<http://help.typekit.com/customer/portal/articles/6852>

<http://www.frikipandi.com/public/post/optimizar-fuentes-web-en-la-carga-de-una-web-se-debe-usar-web-fonts/>

<http://keithclark.co.uk/articles/loading-css-without-blocking-render/>

<http://fontfamily.io/>