



Manual de Configuración de Sitios Servidor Apache

Zweicom S.A

París 720, Piso 4. Santiago de Chile

Tel: +562 6328415 Fax: +562 6641935

Email: info@zweicom.com Home: www.zweicom.com

Santiago, 02 de Junio 2011

Contenido

Contenido	2
1 Descripción.....	4
1.1 Configuraciones sugeridas:.....	4
2 Estructura	5
2.1 Permisos.....	5
2.2 Estructura de directorios	7
2.3 Configuración con Hosts Virtuales (vhost).....	8
2.4 Configuración a nivel de alias.....	10
2.5 mod_userdir o alias con mod_rewrite.....	11
3 Referencias	12

Control de cambios

Fecha	Autor	Observaciones
27/02/11	Rodrigo Riquelme	Primera versión del documento

1 Descripción

A continuación se sugieren algunas directrices para la configuración de sitios usando el servidor web Apache de forma de mantener todos los proyectos agrupados en una estructura fácilmente comprensible y tomando en cuenta los siguientes requerimientos.

- Todos los proyectos de Zweicom deben estar agrupados dentro de una carpeta única.
- Cada proyecto debe tener una carpeta “public_html” o “web/public” (Según framework web) sobre la cual se montará la plataforma web.
- La plataforma web no debe estar separada del resto de la plataforma.
- Se debe poder actualizar plataforma y web conjuntamente a través de un repositorio único de control de versiones.
- Se deben poder considerar la posibilidad de configurar variables de entorno en el servidor, fuera de las carpetas que estén bajo control de versiones.

1.1 Configuración utilizada:

Apache 2.2 con mod_rewrite habilitado.

2 Estructura

2.1 Permisos

Dentro de la configuración de permisos podemos autorizar a nivel de Apache o a nivel de sitio.

En distribuciones basadas en Debian, el usuario de Apache se llama `www-data` y debe encontrarse en el grupo `www-data`. En otras distribuciones basadas en Red Hat se llama `nobody` o `apache`.

Este usuario `www-data` debe tener acceso a los directorios públicos del sitio web para poder leer y mostrar sus contenidos.

Para facilitar la administración del sitio web, se puede agregar al usuario `root` u otro al grupo `www-data`...

```
sudo usermod -a -G www-data usuario_sitio
```

Para verificar si se ha añadido correctamente, iniciar sesión como `usuario_sitio` y ejecutar en consola

```
groups
```

El usuario debe ser miembro `www-data` y el comando anterior debe listar su nombre.

Si no lo hemos creado, según la estructura propuesta en el paso anterior, podemos proceder a crear el directorio `public_html` dentro del directorio de nuestro usuario en `/home`:

A continuación, debemos asegurarnos de que nuestro directorio web o `public_html` es parte del grupo `www-data` de Apache:

```
sudo chgrp -R www-data /home/usuario-admin/public_html
```

Otorgamos los permisos adecuados para que el directorio web y los archivos y directorios creados dentro del mismo sean visibles en la web, usamos el comando...

```
sudo chmod -R 2755 /home/usuario-admin/public_html
```

Si lo deseamos, podemos crear un directorio en el cual puedan almacenarse y grabar archivos, por ejemplo con el nombre de files y otorgar los permisos adecuados.

```
mkdir ~/public_html/files  
sudo chmod -R 2775 /home/usuario-admin/public_html/files
```

El 2 inicial hace que sea permanente el permiso para los archivos creados en el futuro, el primer 7 da todos los permisos al usuario dueño del archivo, el segundo 7 da todos los permisos a los usuarios del grupo al cual pertenece el dueño (www-data de Apache) y el 5 indica permisos de lectura (r) y ejecución (x), pero no de escritura (w), para el resto del mundo.

2.2 Estructura de directorios

Configuración de sitios en Apache

En caso de usar Zend Framework u otro Framework MVC en una aplicación en la cual exista lógica web fuera de la carpeta pública se usará esta estructura

```
/home
  /ambientes
    /{proyecto}
      /web
        /public
```

En otro caso

```
/home
  /ambientes
    /{proyecto}
      /public_html
```

2.3 Configuración con Hosts Virtuales (vhost)

Para poder usar hosts virtuales basados en DNS (tipo `http://subdominio.dominio.com`) en primer lugar debemos tener la disponibilidad de ocupar el dominio o DNS público o en caso contrario poder modificar el archivo hosts del terminal cliente (lo que normalmente solo es aplicable para ambiente de desarrollo), en caso contrario no podremos usar esta modalidad de configuración.

```
<VirtualHost *:80>
  DocumentRoot "/home/ambientes/bconsumo/web/public"
  ServerName bconsumo.tesla

  # This should be omitted in the production environment
  SetEnv APPLICATION_ENV development

  <Directory "/home/ambientes/bconsumo/public_html/public">
    Options Indexes MultiViews FollowSymLinks
    AllowOverride All
    Order allow,deny
    Allow from all
  </Directory>

  #ErrorLog "/proyectos/bconsumo/web/log/error"
  #CustomLog "/proyectos/bconsumo/web/log/access" common
</VirtualHost>
```

En el ejemplo anterior tenemos las siguientes directivas de interés.

DocumentRoot: indica la carpeta donde estarán los scripts web públicos

ServerName: indica la URL pública

AllowOverride: “all” permite usar `mod_rewrite` en la aplicación para personalizar URLs y extender la funcionalidad de Apache mediante un archivo `.htaccess`

ErrorLog: Ruta de log de errores de Apache para ese sitio (opcional).

CustomLog: Ruta de log de accesos de Apache para ese sitio (opcional).

SetEnv: Podemos configurar variables (para efectos prácticos constantes) globales de apache. En PHP se puede obtener el valor de estas con `$ip = getenv('APPLICATION_ENV')`.

En el siguiente ejemplo se define la constante (global) `APPLICATION_ENV` para ser usada en PHP:

```
<?php
defined('APPLICATION_ENV') || define('APPLICATION_ENV', (getenv('APPLICATION_ENV') ?
getenv('APPLICATION_ENV') : 'production'));
```

En el siguiente ejemplo se inicializan las variables de configuración según `APPLICATION_ENV` en Zend Framework.

```
<?php
$config = new Zend_Config_Ini(ROOT_DIR.'/application/configs/application.ini',
APPLICATION_ENV);
$hosts = $config->db->params->host;
```

Ejemplo `application.conf`

```
[production]
{...}
db.adapter = pdo_mysql
db.params.host = 127.0.0.1
db.params.port = 3453
db.params.username = usuario_produccion
db.params.password = password_produccion
db.params.dbname = DB_PRODUCION

[testing : production]
db.params.username = usuario_testing
db.params.password = password_testing
db.params.port = 3306
phpSettings.display_startup_errors = 1
phpSettings.display_errors = 1

[development : testing]
db.params.username = usuario_development
db.params.password = password_development
db.params.dbname = DB_DEVELOPMENT
resources.frontController.params.displayExceptions = 1
```

Para tener los virtualhost separados de la configuración general se puede crear en la carpeta de Apache una subcarpeta `vhost` e incluirla en la configuración de apache.

`Include vhost/*.conf`

Si se prefiere hacer una configuración XML se puede usar [Zend Config Xml](#) e iterar de exactamente la misma forma.

2.4 Configuración a nivel de alias

Si no tenemos el DNS registrado ni tenemos acceso al archivo hosts cliente, no podremos configurar subdominios, pero podremos elegir en que carpeta será visible en el navegador web a través de la directiva Alias

```
Alias /bconsumo "/proyectos/bconsumo/web/public"  
<Directory "/proyectos/bconsumo/web/public">  
# This should be omitted in the production environment  
SetEnv APPLICATION_ENV development  
DirectoryIndex index.html index.php  
</Directory>
```

Alias: El primer parámetro es la ruta pública, el segundo es la ruta interna.

Directory: Ruta interna

SetEnv: ver capítulo anterior

Para tener los Alias separados de la configuración general se puede crear en la carpeta de Apache una subcarpeta vhost e incluirla en la configuración de apache.

```
Include alias/*.conf
```

2.5 mod_userdir o alias con mod_rewrite

Debe añadirse la directiva **RewriteBase** al archivo .htaccess dentro de la carpeta public con la ruta web absoluta del archivo index.php.

```
RewriteBase /bconsumo/  
RewriteEngine On  
RewriteCond %{REQUEST_FILENAME} -s [OR]  
RewriteCond %{REQUEST_FILENAME} -l [OR]  
RewriteCond %{REQUEST_FILENAME} -d  
RewriteRule ^.*$ - [NC,L]  
RewriteRule ^.*$ index.php [NC,L]
```

3 Referencias

<http://httpd.apache.org/docs/2.0/vhosts/examples.html>
<http://www.php.net/manual/es/function.apache-getenv.php>
http://httpd.apache.org/docs/2.0/mod/mod_env.html
<http://framework.zend.com/manual/en/zend.config.adapters.xml.html>
<http://framework.zend.com/manual/en/zend.config.adapters.ini.html>