3 extended apache configuration

restart kills the procese and initialice the daemon

reload restores the config info

Extended Apache VirtualHost Configuration Tutorial (HTTP only)

1. Introduction

In this tutorial, new directives are explained:

- ErrorDocument
- Options -Indexes
- ServerAlias yourname.com yourname.es
- Redirect 301 /search http://www.google.com/
- DirectoryIndex main.html

2. Website Directory

Asure the directory that will hold your website files:

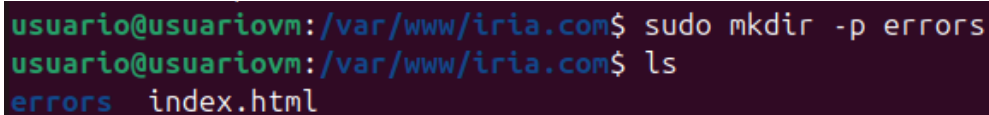sudo mkdir -p /var/www/yourname.com/

Then add a test homepage:

echo "<h1>Welcome to yourname.com!</h1>" | sudo tee /var/www/yourname.com//main.html

To set up a custom error page for the 404 (Page Not Found) error in Apache2, you can modify the default site configuration to add the `ErrorDocument` directive. Here are the steps:

Create the Custom 404 Error Page

Create an HTML file to be displayed when a 404 error occurs:

```
sudo mkdir -p /var/www/html/errors
sudo nano /var/www/html/errors/404.html
```


Add custom content to `404.html`. For example:

```
<html>
    <head><title>Page Not Found</title></head>
    <body>
        <h1>404 - Page Not Found</h1>
        <p>Sorry, but the page you are looking for does not
exist.</p>
        <a href="/">Go back to the homepage</a>
    </body>
</html>
```

Save and close the file.

Configure the Error Page in the Site Configuration File

1. Open the default site configuration file:
2. `sudo nano /etc/apache2/sites-available/000-default.conf`
3. Add the following line inside the `<VirtualHost>` block to specify the custom error page:
4. `ErrorDocument 404 /errors/404.html`

The configuration file should look something like this:

```
<VirtualHost *:80>
    ServerName example.com
    ServerAdmin webmaster@example.com
    DocumentRoot /var/www/html/example

    # Custom error page
    ErrorDocument 404 /errors/404.html

    ErrorLog ${APACHE_LOG_DIR}/error.log
    CustomLog ${APACHE_LOG_DIR}/access.log combined
</VirtualHost>
```

5. Save and close the file.

ERORS. Test the Custom Error Page

1. Try accessing a URL that does not exist on your domain, like
    `http://example.com/does-not-exist.`
2. You should see the custom 404 error page you created at
    `/var/www/html/errors/404.html.`

This sets up a custom 404 error page in Apache2 for the default site.

3. Explanation of Each Directive

Directive	Description
ServerName yourname.com	The main domain for this VirtualHost.
ServerAlias yourname.com yourname.es	Allows the same site to respond to both yourname.com and yourname.es.
DocumentRoot	The folder where your web files are stored.
Options -Indexes	Prevents Apache from listing files if there's no index file.
AllowOverride All	Allows .htaccess files to override settings if needed.
Require all granted	Lets everyone access the site.
DirectoryIndex main.html	Sets main.html as the default page.
Redirect 301 /search http://www.google.com/	Redirects all requests to /search to Google permanently.
ErrorLog / CustomLog	Specifies log file locations for errors and access.

4. Create the Apache VirtualHost File

Now create a configuration file for your site:

sudo nano /etc/apache2/sites-available/yourname.com.conf

```
usuario@usuariovm:/var/www/iria.com/errors$ sudo nano /etc/apache2/sites-available/iria.com.conf
```

Paste this full configuration:

```
<VirtualHost *:80>

    ServerAdmin admin@yourname.com

    ServerName yourname.com

    ServerAlias yourname.com yourname.es


    DocumentRoot /var/www/yourname.com/

    <Directory /var/www/yourname.com >

        Options -Indexes

        AllowOverride All

        Require all granted

        DirectoryIndex main.html

    </Directory>

    # create 404.html in ./errors

    ErrorDocument 404 /errors/404.html

    # Redirect only the /search path to Google

    Redirect 301 /search http://www.google.com/

    ErrorLog ${APACHE_LOG_DIR}/yourname_error.log

    CustomLog ${APACHE_LOG_DIR}/yourname_access.log combined

</VirtualHost>
```
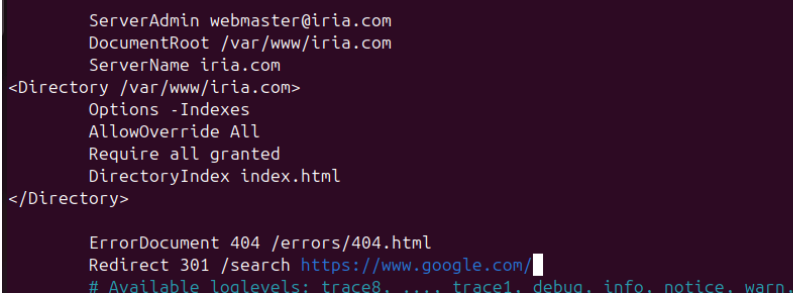
Save and close (Ctrl + O, Enter, Ctrl + X).

```
        ServerName iria.com
        ServerAlias iria.com iria.es
```

5. 🚀 Enable and Activate the Site

Enable your VirtualHost:

sudo a2ensite yourname.com.conf


```
usuario@usuariovm:/etc/apache2/sites-available$ sudo a2ensite iria.com.conf
Site iria.com already enabled
usuario@usuariovm:/etc/apache2/sites-available$
```

Then reload Apache to apply the changes:

sudo systemctl reload apache2

You can disable the default Apache page if you like:

sudo a2dissite 000-default.conf

sudo systemctl reload apache2

6. 🧪 Test the Configuration

First, check for syntax errors:

sudo apache2ctl configtest


```
usuario@usuariovm:/etc/apache2/sites-available$ sudo apache2ctl configtest
AH00558: apache2: Could not reliably determine the server's fully qualified domain name, using 127.0
rName' directive globally to suppress this message
Syntax OK
usuario@usuariovm:/etc/apache2/sites-available$
```

Expected output:

Syntax OK

Now test in your browser:

Test URL	Expected Result
http://yourname.com/	Displays your main.html page.
http://yourname.com/search	Redirects to http://www.google.com/.
http://yourname.com/folder/	Returns "403 Forbidden" (no directory listing).

Test URL	Expected Result
http://yourname.es/	Loads the same website as yourname.com.

7. 🧾 Summary of Configuration

Here's the final VirtualHost block for clarity:

```
<VirtualHost *:80>

    ServerAdmin admin@yourname.com

    ServerName yourname.com

    ServerAlias yourname.com yourname.es


    DocumentRoot /var/www/yourname.com


    <Directory /var/www/yourname.com >

        Options -Indexes

        AllowOverride All

        Require all granted

        DirectoryIndex main.html

    </Directory>


    Redirect 301 /search http://www.google.com/


    ErrorLog ${APACHE_LOG_DIR}/yourname_error.log

    CustomLog ${APACHE_LOG_DIR}/yourname_access.log combined

</VirtualHost>
```

Now your working Apache VirtualHost

- Hosts your site at yourname.com and yourname.es
- Uses main.html as the homepage
- Manages errors
- Disables directory listing
- Redirects /search to Google
- Works entirely on **HTTP (no SSL)**