Lab 9: Routing HTTP Requests

Estimated time for completion: **20 minutes**

Requirements

The following tasks must be completed before beginning this lab:

- Getting Started with NGINX, (the Getting Started Guide in LearnF5)
- Log into Hosted Environment, your lab initialization instructions are located in the LearnF5 course

Scenario

You set up a location using a regular expression that captures incoming HTTP requests looking for the **`/pictures`** URI, followed by any character before a dot, followed by either an extension of **`gif`**, **`jpe`**, **`jpg`**, or **`png`**. In other words, the location block serves URI's that have **`/pictures/<file_name>.<picture_file_extension>`** in them. You also specify a replacement path using the alias directive.

Objectives

At the end of this lab you will be able to:

- Use NGINX directives to reroute traffic
- Define URL rewrites
- Determine rewrite request processing

Lab Contents

Exercise 1: Use the `alias` directive

IMPORTANT
You can copy and paste the commands and text from the examples to your terminal or editor, (just make sure you don't copy and paste the $ prompt!)

Exercise 1: Using the `alias` directive

Learning Objective:

Use the **`alias`** directive with a location defined by a regular expression.

Scenario

You set up a location using a regular expression that captures incoming HTTP requests looking for the **`/pictures`** URI, followed by any character before a dot, followed by either an extension of **`gif`**, **`jpe`**, **`jpg`**, or **`png`**. In other words, the location block serves URI's that have **`/pictures/<file_name>.<picture_file_extension>`** in them. You also specify a replacement path using the alias directive.

1. Rename your **`default.conf`** file to **`default.bak`**:

```
$ cd /etc/nginx/conf.d
$ sudo mv default.{conf,bak}
```

2. Rename your **`mywebserver.bak`** file to **`mywebserver.conf`**:

```
$ sudo mv mywebserver.{bak,conf}
```

3. Open the **`mywebserver.conf`** configuration file:

```
$ sudo vim /etc/nginx/conf.d/mywebserver.conf
```

4. Comment out the existing **`/images`** location block.

```
# location /images {
# root /data;
# }
```

5. Create a new **location** (positioned directly below the log directive lines, as shown below), using the following case in sensitive regular expression and the **`alias`** directive:

```
location ~ ^/pictures/(.+\.(gif|jpeg|png))$ {
    alias /data/images/$1;
}
```

Your file look be like this:

```
# This is the http context

log_format test_log ' "Request: $request\n Status: $status\n
Request_URI: $request_uri\n Hosts: $host\n Client_IP:
$remote_addr\n Proxy_IP(s): $proxy_add_x_forwarded_for\n Proxy_Host
name: $proxy_host\n Real_IP: $http_x_real_ip" ';

server {
listen 80;
root /home/ubuntu/public_html;

error_log /var/log/nginx/server1.error.log info;
access_log /var/log/nginx/server1.access.log test_log;

location ~ ^/pictures/(.+\.(gif|jpe?g|png))$ {
alias /data/images/$1;
  }

  location /application1 {
  index app1.html;
  proxy_pass http://localhost:8080/sanokeApp;
  }

  location /application2 {
  index app2.html;
  }

# location /images {
# root /data;
# }
}
```

6. Save the file and reload NGINX. (**esc** and **:wq**).

```
$ sudo nginx -s reload
```

7. In a browser, test for the correct results:

   http://localhost/pictures/logo.png

   You should get the NGINX logo in your browser.

Expected Results

You set up a location using a regular expression that captures incoming HTTP requests looking for the **/pictures** URI, followed by any character before a dot, followed by either an extension of **gif**, **jpe**, **jpg**, or **png**. You then specified a replacement path using the alias directive.