Setting Up an NGINX Demo Environment

Configure NGINX Open Source as a web server and NGINX Plus as a load balancer, as required for the sample deployments in NGINX deployment guides.

The instructions in this guide explain how to set up a simple demo environment that uses NGINX Plus to load balance web servers that run NGINX Open Source and serve two distinct web applications. It is referenced by some of our deployment guides for implementing highly availability of NGINX Plus and NGINX Open Source in cloud environments.

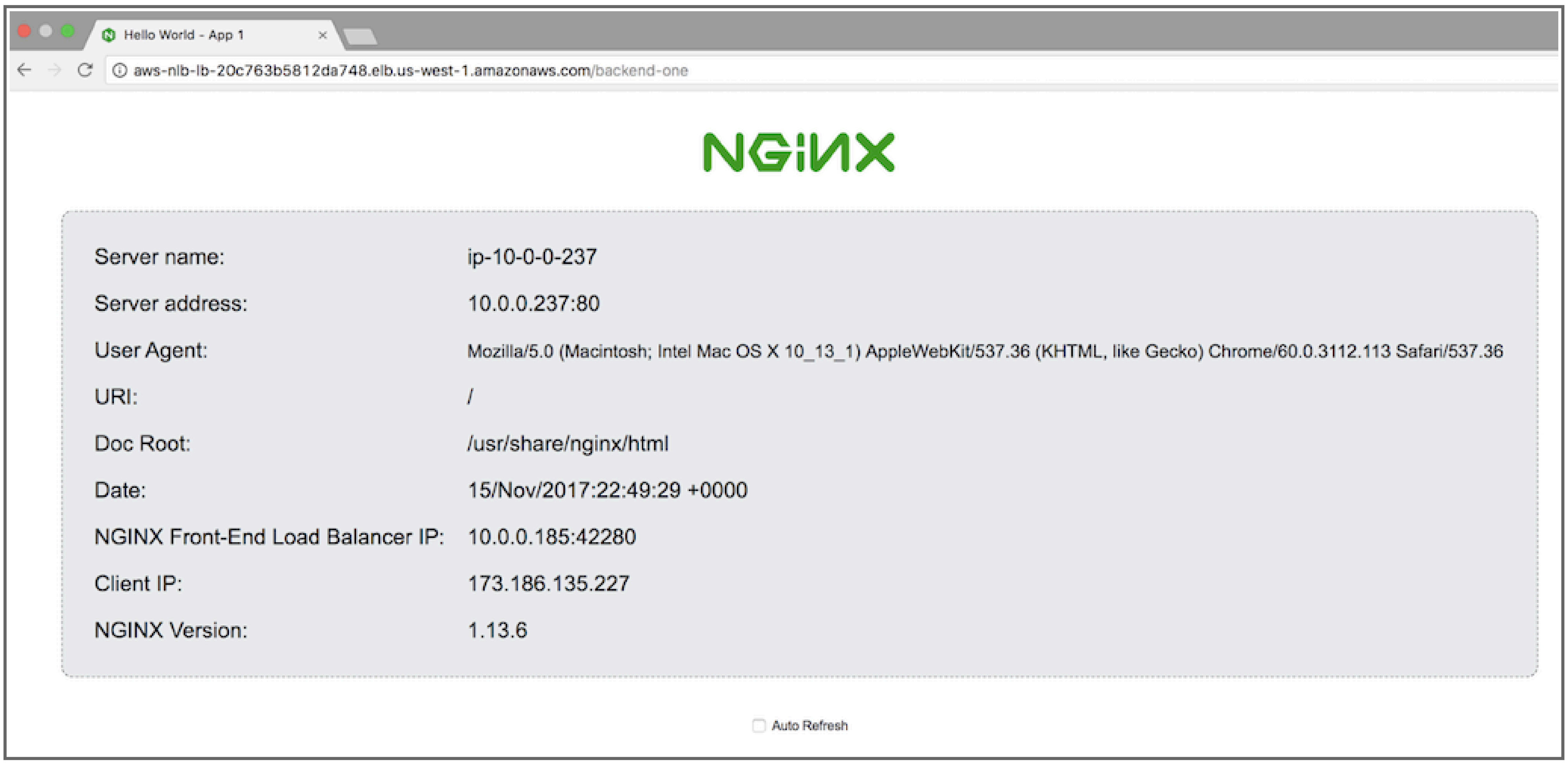
Prerequisites

This guide assumes you have already provisioned a number of host systems (physical servers, virtual machines, containers, or cloud instances) required for a deployment guide (if applicable) and installed NGINX Open Source or NGINX Plus on each instance as appropriate. For installation instructions, see the NGINX Plus Admin Guide.

Some commands require `root` privilege. If appropriate for your environment, prefix commands with the `sudo` command.

Configuring NGINX Open Source for Web Serving

The steps in this section configure an NGINX Open Source instance as a web server to return a page like the following, which specifies the server name, address, and other information. The page is defined in the **demo-index.html** configuration file you create in Step 4 below.


Hello World - App 1
aws-nlb-lb-20c763b5812da748.elb.us-west-1.amazonaws.com/backend-one
NGINX
□ Auto Refresh

Server name:	ip-10-0-0-237
Server address:	10.0.0.237:80
User Agent:	Mozilla/5.0 (Macintosh; Intel Mac OS X 10_13_1) AppleWebKit/537.36 (KHTML, like Gecko) Chrome/60.0.3112.113 Safari/537.36
URI:	/
Doc Root:	/usr/share/nginx/html
Date:	15/Nov/2017:22:49:29 +0000
NGINX Front-End Load Balancer IP:	10.0.0.185:42280
Client IP:	173.186.135.227
NGINX Version:	1.13.6

If you are using these instructions to satisfy the prerequisites for one of our deployment guides, the Appendix in the guide specifies the name of each NGINX Open Source instance and whether to configure **App 1** or **App 2**.

Note: Some commands require `root` privilege. If appropriate for your environment, prefix commands with the `sudo` command.

1. Open a connection to the NGINX Open Source instance and change the directory to **/**etc/nginx/conf.d**:

   Copy

   ```
   cd /etc/nginx/conf.d
   ```

2. Rename **default.conf** to **default.conf.bak** so that NGINX Plus does not use it.

   Copy

   ```
   mv default.conf default.conf.bak
   ```

3. Create a new file called **app.conf** with the following contents.

   Copy

```
server {
    listen 80 default_server;
    server_name app_server;

    root /usr/share/nginx/html;
    error_log /var/log/nginx/app-server-error.log notice;
    index demo-index.html index.html;
    expires -1;

    sub_filter_once off;
    sub_filter 'server_hostname' '$hostname';
    sub_filter 'server_address'  '$server_addr:$server_port';
    sub_filter 'server_url'      '$request_uri';
    sub_filter 'remote_addr'     '$remote_addr:$remote_port';
    sub_filter 'server_date'     '$time_local';
    sub_filter 'client_browser'  '$http_user_agent';
    sub_filter 'request_id'      '$request_id';
    sub_filter 'nginx_version'   '$nginx_version';
    sub_filter 'document_root'   '$document_root';
    sub_filter 'proxied_for_ip'  '$http_x_forwarded_for';
}
```

Directive documentation: error_log, expires, index, listen, root, server, server_name, sub_filter

4. Include the following directive in the top-level ("main") context in **/etc/nginx/nginx.conf**, if it does not already appear there.

   Copy

   ```
   include conf.d/*.conf;
   ```

   Directive documentation: include

5. In the **/usr/share/nginx/html** directory, create a new file called **demo-index.html** with the following contents, which define the default web page that appears when users access the instance.

   In the `<title>` tag, replace the comment with `1` or `2` depending on whether the instance is serving **App 1** or **App 2**.

   Copy

```
<!DOCTYPE html>
<html>
    <head>
        <title>Hello World - App X <!-- Replace 'X' with '1' or '2' as appropriate --></title>
        <link href="data:image/png;base64,iVBORw0KGgoAAAANSUhEUgAAAEAAAABACAYAAACqaXHeAAAGPElEQVR42u1bDUyUdRj/iwpolN
        <style>
            body {
                margin: 0px;
                font: 20px 'RobotoRegular', Arial, sans-serif;
                font-weight: 100;
                height: 100%;
                color: #0f1419;
            }
            div.info {
                display: table;
                background: #e8eaec;
                padding: 20px 20px 20px 20px;
                border: 1px dashed black;
                border-radius: 10px;
                margin: 0px auto auto auto;
            }
            div.info p {
                display: table-row;
                margin: 5px auto auto auto;
            }
            div.info p span {
                display: table-cell;
                padding: 10px;
            }
            img {
                width: 176px;
                margin: 36px auto 36px auto;
                display:block;
            }
            div.smaller p span {
                color: #3D5266;
            }
            h1, h2 {
                font-weight: 100;
            }
            div.check {
                padding: 0px 0px 0px 0px;
                display: table;
                margin: 36px auto auto auto;
                font: 12px 'RobotoRegular', Arial, sans-serif;
            }
            #footer {
                position: fixed;
                bottom: 36px;
                width: 100%;
            }
            #center {
                width: 400px;
                margin: 0 auto;
                font: 12px Courier;
            }
        </style>

        <script>
            var ref;
            function checkRefresh() {
                if (document.cookie == "refresh=1") {
                    document.getElementById("check").checked = true;
                    ref = setTimeout(function(){location.reload();}, 1000);
                } else {
                }
            }
            function changeCookie() {
                if (document.getElementById("check").checked) {
                    document.cookie = "refresh=1";
                    ref = setTimeout(function(){location.reload();}, 1000);
                } else {
                    document.cookie = "refresh=0";
                    clearTimeout(ref);
                }
            }
        </script>
    </head>

    <body onload="checkRefresh();">
        <img alt="NGINX Logo" src="data:image/png;base64,iVBORw0KGgoAAAANSUhEUgAAAWAAAABICAMAAAD/N9+RAAAAVFBMVEUAAA
        <div class="info">
            <p><span>Server name:</span> <span>server_hostname</span></p>
            <p><span>Server address:</span> <span>server_address</span></p>
            <p class="smaller"><span>User Agent:</span> <span>client_browser</span></p>
            <p class="smaller"><span>URI:</span> <span>server_url</span></p>
            <p class="smaller"><span>Doc Root:</span> <span>document_root</span></p>
            <p class="smaller"><span>Date:</span> <span>server_date</span></p>
            <p class="smaller"><span>NGINX Frontend Load Balancer IP:</span> <span>remote_addr</span></p>
            <p class="smaller"><span>Client IP:</span> <span>proxied_for_ip</span></p>
            <p class="smaller"><span>NGINX Version:</span> <span>nginx_version</span></p>
        </div>
        <div class="check">
            <input type="checkbox" id="check" onchange="changeCookie()"> Auto Refresh</input>
        </div>
        <div id="footer">
            <div id="center" align="center">
                Request ID: request_id<br/>
                © NGINX, Inc. 2018
            </div>
        </div>
    </body>
</html>
```

Configuring NGINX Plus for Load Balancing

The steps in this section configure an NGINX Plus instance to load balance requests across the group of NGINX Open Source web servers you configured in the previous section.

If you are using these instructions to satisfy the prerequisites for one of our deployment guides, the Appendix in the guide specifies the names of the NGINX Plus instances used in it.

Repeat these instructions on each instance. Alternatively, you can configure one instance and share the configuration with its peers in a cluster. See the NGINX Plus Admin Guide.

1. Open a connection to the NGINX Plus instance and change the directory to ** **/etc/nginx/conf.d**:

   Copy

   ```
   cd /etc/nginx/conf.d
   ```

2. Rename **default.conf** to **default.conf.bak** so that NGINX Plus does not use it.

   Copy

   ```
   mv default.conf default.conf.bak
   ```

3. Create a new file called **lb.conf** with the following contents.

Note: In the `upstream` blocks, include a server directive for each NGINX Open Source instance that serves the relevant application.

Copy

```
# in the 'http' context
upstream app1 {
    server <internal IP address of NGINX Open Source instance serving App 1>;
    # 'server' directives for additional App 1 servers, if using
    zone app1 64k;
}

upstream app2 {
    server <internal IP address of NGINX Open Source instance serving App 2>;
    # 'server' directives for additional App 2 servers, if using
    zone app2 64k;
}

server {
    listen 80;
    status_zone backend;
    root /usr/share/nginx/html;

    location / {
        # directives for serving the site's HTML landing page
    }

    location /application1 {
        proxy_set_header Host $host;
        proxy_set_header X-Forwarded-For $remote_addr;
        proxy_pass http://app1/;
    }

    location /application2 {
        proxy_set_header Host $host;
        proxy_set_header X-Forwarded-For $remote_addr;
        proxy_pass http://app2/;
    }

    location /api {
        api write=on;
        # directives to control access, such as 'allow' and 'deny'
    }

    location = /dashboard.html {
        root /usr/share/nginx/html;
    }

    location = /status.html {  # redirect requests that are made to pre-R14 dashboard
        return 301 /dashboard.html;
    }
}
```

Directive documentation: api, listen, location, proxy_pass, proxy_set_header, return, root, server (upstream),server (virtual), server_name, status_zone, upstream, zone

4. Include the following directive in the top-level ("main") context in **/etc/nginx/nginx.conf**, if it does not already appear there.

Copy

```
include conf.d/*.conf;
```

Directive documentation: include

Revision History

- Version 2 (April 2019) – Generalized instructions for use with deployment guides

- Version 1 (April 2018) – Initial version