

Contents

1. Introduction	2
2. Phase 1 – Install NGINX.....	2
3. Phase 2 – Serve Static Website	2
4. Phase 3 – Reverse Proxy Setup.....	3
5. Phase 4 – Load Balancing.....	5
6. Phase 5 – SSL/TLS (Self-signed Certificate).....	7
7. Phase 6 – PHP-FPM Integration	11
8. Phase 7 – URL Rewrites & Redirects	12
9. Phase 8 – Security Headers & Basic Auth.....	14
10. Phase 9 – Custom Error Pages (404, 403, 500).....	15
11. Phase 10 – Multi-site Hosting.....	16
12. Phase 11 – nginx.conf Architecture & Includes	17
13. Phase 12 – Variables (\$host, \$remote_addr, etc.)	18
14. Phase 13 – Rate Limiting + Connection Limits	18
16. Phase 15 – WebSocket Support	21
17. Phase 16 – Log Monitoring	21
Challenges Faced During the Project	23
1. Server Block Conflicts (Multiple Sites Overlapping)	23
2. “403 Forbidden” and “502 Bad Gateway” Errors	23
3. NGINX Configuration Failing with Syntax Errors	24
4. PHP-FPM Socket Issues (Page not loading)	24
5. Rewrite Rules Not Working Initially	25
6. Basic Authentication Causing Infinite Loading.....	25
7. Proxy Cache Validation Issues (Backend Not Running).....	25

8. Understanding Access Logs and Error Logs	26
---	----

1. Introduction

In this project, I performed end-to-end research and hands-on implementation of NGINX. I started from basic static hosting and moved towards advanced reverse proxy, load balancing, SSL/TLS, PHP-FPM, custom error pages, security headers, rate limiting, and WebSocket support.

This document contains all configurations, tests, and screenshots.

2. Phase 1 – Install NGINX

What I did:

- Installed nginx on Ubuntu EC2
- Verified service
- Accessed default nginx page

Commands used:

```
sudo apt update
```

```
sudo apt install nginx -y
```

```
systemctl status nginx
```

3. Phase 2 – Serve Static Website

Steps I performed:

1. Created directory /var/www/example.com
2. Added index.html
3. Created server block
4. Enabled the site
5. Reloaded nginx

← → G △ Not secure 18.236.66.150

Static website served by Nginx

Static content using root and index directives.

```
ubuntu@ip-10-0-0-32:~$ cat /etc/nginx/sites-available/example.com
server {
    listen 80;
    listen [::]:80;

    server_name example.com www.example.com;

    root /var/www/example.com/html;
    index index.html;

    location / {
        try_files $uri $uri/ =404;
    }
}
```

```
ubuntu@ip-10-0-0-32:~$ cat /var/www/example.com/html/index.html
<!DOCTYPE html>
<html>
<head>
<title>Welcome to Nginx Research</title>
</head>
<body>
<h1>Static website served by Nginx</h1>
<p>Static content using root and index directives.</p>
</body>
</html>
ubuntu@ip-10-0-0-32:~$ |
```

4. Phase 3 – Reverse Proxy Setup

Steps:

- Backend running on port 3000
- nginx → proxy → backend

proxy_pass config:

```
location {
```

```
    proxy_pass http://127.0.0.1:3000;
```

```
}
```

```
ubuntu@ip-10-0-0-32:~$ cat /etc/nginx/sites-available/reverseproxy.com
server {
    listen 80;
    listen [::]:80;

    server_name reverseproxy.com;

    location / {
        proxy_pass http://127.0.0.1:3000;

        proxy_set_header Host $host;
        proxy_set_header X-Real-IP $remote_addr;
        proxy_set_header X-Forwarded-For $proxy_add_x_forwarded_for;
        proxy_set_header X-Forwarded-Proto $scheme;

        proxy_http_version 1.1;
        proxy_set_header Connection "";
    }
}
ubuntu@ip-10-0-0-32:~$ |
```

```
ubuntu@ip-10-0-0-32:~$ ls -l /etc/nginx/sites-enabled/
total 0
lrwxrwxrwx 1 root root 43 Nov 19 14:37 reverseproxy.com -> /etc/nginx/sites-available/reverseproxy.com
ubuntu@ip-10-0-0-32:~$ |
```

```
ubuntu@ip-10-0-0-32:~$ cd ~/backend-app
ubuntu@ip-10-0-0-32:~/backend-app$ ls -l
total 4
-rw-rw-r-- 1 ubuntu ubuntu 470 Nov 19 13:35 server.js
ubuntu@ip-10-0-0-32:~/backend-app$ cat server.js
const http = require('http');

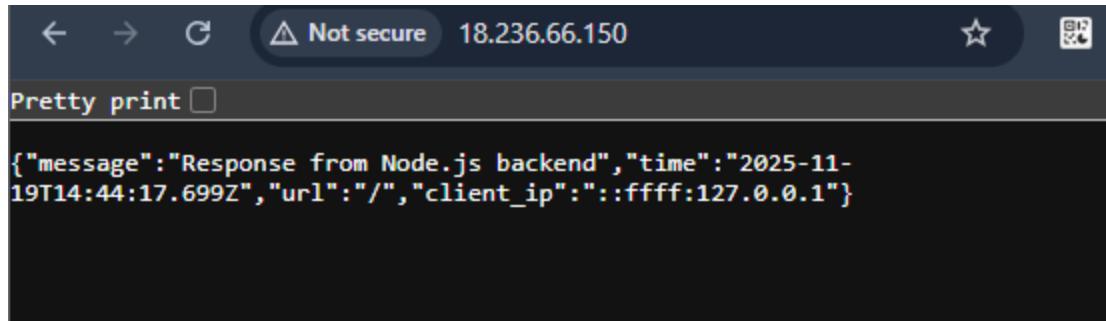
const server = http.createServer((req, res) => {
    res.writeHead(200, {
        'Content-Type': 'application/json'
    });

    const response = {
        message: "Response from Node.js backend",
        time: new Date().toISOString(),
        url: req.url,
        client_ip: req.socket.remoteAddress
    };

    res.end(JSON.stringify(response));
});

server.listen(3000, () => {
    console.log("Backend running on port 3000");
});
```

```
ubuntu@ip-10-0-0-32:~/backend-app$ |
ubuntu@ip-10-0-0-32:~$ curl http://localhost:3000
{"message": "Response from Node.js backend", "time": "2025-11-19T14:43:13.078Z", "url": "/", "client_ip": "::1"}ubuntu@ip-10-0-0-32:~$ |
```



A screenshot of a web browser window. The address bar shows the URL `18.236.66.150`. Below the address bar, there is a "Pretty print" checkbox. The main content area displays a JSON object:

```
{"message": "Response from Node.js backend", "time": "2025-11-19T14:44:17.699Z", "url": "/", "client_ip": "::ffff:127.0.0.1"}
```

5. Phase 4 – Load Balancing

Tasks:

- Created 3 backend servers
- Implemented:
 - round_robin
 - least_conn
 - ip_hash

```
ubuntu@ip-10-0-0-32:/etc/nginx/sites-enabled$ ps aux | grep node
ubuntu      8759  0.0  5.1 609528 48668 pts/1    S1   14:52   0:00 node /home/ubuntu/backend-app-2/server.js
ubuntu      8771  0.0  5.1 609788 48308 pts/1    S1   14:54   0:00 node /home/ubuntu/backend-app-3/server.js
ubuntu      8791  0.0  5.1 609528 48440 pts/1    S1   14:58   0:00 node /home/ubuntu/backend-app/server.js
ubuntu     9145  0.0  0.2    7076  2200 pts/1    S+   15:14   0:00 grep --color=auto node
ubuntu@ip-10-0-0-32:/etc/nginx/sites-enabled$ |
```

```

ubuntu@ip-10-0-0-32:/etc/nginx/sites-enabled$ curl localhost:3000
curl: (52) localhost:3002
{"message":"Response from Node.js backend","time":"2025-11-19T15:15:03.251Z","url":"/","client_ip":"::1"} {"message":"Response from Node.js backend","time":"2025-11-19T15:15:03.307Z","url":"/","client_ip":"::1"} ubuntu@ip-10-0-0-32:/etc/nginx/sites-enabled$ cat /etc/nginx/sites-available/loadbalancer.com
available/loadbalancer.com
upstream backend_rr {
    server 127.0.0.1:3000;
    server 127.0.0.1:3001;
    server 127.0.0.1:3002;
}

upstream backend_least_conn {
    least_conn;
    server 127.0.0.1:3000;
    server 127.0.0.1:3001;
    server 127.0.0.1:3002;
}

upstream backend_ip_hash {
    ip_hash;
    server 127.0.0.1:3000;
    server 127.0.0.1:3001;
    server 127.0.0.1:3002;
}

server {
    listen 80 default_server;
    listen [::]:80 default_server;
    server_name _;

    # Round robin
    location /rr {
        proxy_pass http://backend_rr;

        proxy_set_header Host $host;
        proxy_set_header X-Real-IP $remote_addr;
        proxy_set_header X-Forwarded-For $proxy_add_x_forwarded_for;
        proxy_set_header X-Forwarded-Proto $scheme;

        add_header X-Upstream $upstream_addr;
    }

    # least_conn
    location /least {
        proxy_pass http://backend_least_conn;

        proxy_set_header Host $host;
        proxy_set_header X-Real-IP $remote_addr;
        proxy_set_header X-Forwarded-For $proxy_add_x_forwarded_for;
        proxy_set_header X-Forwarded-Proto $scheme;

        add_header X-Upstream $upstream_addr;
    }

    # ip_hash
    location /iphash {
        proxy_pass http://backend_ip_hash;

        proxy_set_header Host $host;
        proxy_set_header X-Real-IP $remote_addr;
        proxy_set_header X-Forwarded-For $proxy_add_x_forwarded_for;
        proxy_set_header X-Forwarded-Proto $scheme;

        add_header X-Upstream $upstream_addr;
    }
}

ubuntu@ip-10-0-0-32:/etc/nginx/sites-enabled$ ls -l /etc/nginx/sites-enabled/
total 0
1rwxrwxrwx 1 root root 43 Nov 19 15:08 Loadbalancer.com -> /etc/nginx/sites-available/loadbalancer.com
ubuntu@ip-10-0-0-32:/etc/nginx/sites-enabled$
```

```

1rwxrwxrwx 1 root root 43 Nov 19 15:08 Toadbalancer.com -> /etc/nginx/sites-available/loadbal
ubuntu@ip-10-0-0-32:/etc/nginx/sites-enabled$ curl -I http://18.236.66.150/rr
curl -I http://18.236.66.150/rr
curl -I http://18.236.66.150/rr
HTTP/1.1 200 OK
Server: nginx/1.24.0 (Ubuntu)
Date: Wed, 19 Nov 2025 15:17:52 GMT
Content-Type: application/json
Connection: keep-alive
X-Upstream: 127.0.0.1:3000

HTTP/1.1 200 OK
Server: nginx/1.24.0 (Ubuntu)
Date: Wed, 19 Nov 2025 15:17:52 GMT
Content-Type: application/json
Connection: keep-alive
X-Upstream: 127.0.0.1:3001

HTTP/1.1 200 OK
Server: nginx/1.24.0 (Ubuntu)
Date: Wed, 19 Nov 2025 15:17:53 GMT
Content-Type: application/json
Connection: keep-alive
X-Upstream: 127.0.0.1:3002

ubuntu@ip-10-0-0-32:/etc/nginx/sites-enabled$ |

```

```

ubuntu@ip-10-0-0-32:/etc/nginx/sites-enabled$ curl -I http://18.236.66.150/least
curl -I http://18.236.66.150/least
curl -I http://18.236.66.150/least
HTTP/1.1 200 OK
Server: nginx/1.24.0 (Ubuntu)
Date: Wed, 19 Nov 2025 15:18:25 GMT
Content-Type: application/json
Connection: keep-alive
X-Upstream: 127.0.0.1:3000

HTTP/1.1 200 OK
Server: nginx/1.24.0 (Ubuntu)
Date: Wed, 19 Nov 2025 15:18:25 GMT
Content-Type: application/json
Connection: keep-alive
X-Upstream: 127.0.0.1:3001

HTTP/1.1 200 OK
Server: nginx/1.24.0 (Ubuntu)
Date: Wed, 19 Nov 2025 15:18:25 GMT
Content-Type: application/json
Connection: keep-alive
X-Upstream: 127.0.0.1:3000

ubuntu@ip-10-0-0-32:/etc/nginx/sites-enabled$ |

```

```

ubuntu@ip-10-0-0-32:/etc/nginx/sites-enabled$ curl -I http://18.236.66.150/iphash
HTTP/1.1 200 OK
Server: nginx/1.24.0 (Ubuntu)
Date: Wed, 19 Nov 2025 15:19:24 GMT
Content-Type: application/json
Connection: keep-alive
X-Upstream: 127.0.0.1:3001

ubuntu@ip-10-0-0-32:/etc/nginx/sites-enabled$ |

```

6. Phase 5 – SSL/TLS (Self-signed Certificate)

Steps:

- Generated certificate

- Configured port 443
- Redirected HTTP → HTTPS

```
ubuntu@ip-10-0-0-32:~$ ls -l /etc/nginx/ssl
total 8
-rw-r--r-- 1 root root 1310 Nov 19 15:26 selfsigned.crt
-rw----- 1 root root 1704 Nov 19 15:23 selfsigned.key
ubuntu@ip-10-0-0-32:~$ |
```

```
ubuntu@ip-10-0-0-32:~$ cat /etc/nginx/sites-available/ssl-test.com
server {
    listen 80;
    server_name 18.236.66.150;

    return 301 https://$host$request_uri;
}

server {
    listen 443 ssl;
    server_name 18.236.66.150;

    ssl_certificate /etc/nginx/ssl/selfsigned.crt;
    ssl_certificate_key /etc/nginx/ssl/selfsigned.key;

    location / {
        return 200 "SSL Test Successful\n";
    }
}
ubuntu@ip-10-0-0-32:~$
```

 Not secure https://18.236.66.150

Your connection is not private

Attackers might be trying to steal your information from **18.236.66.150** (for example, passwords, messages or credit cards). [Learn more about this warning](#)

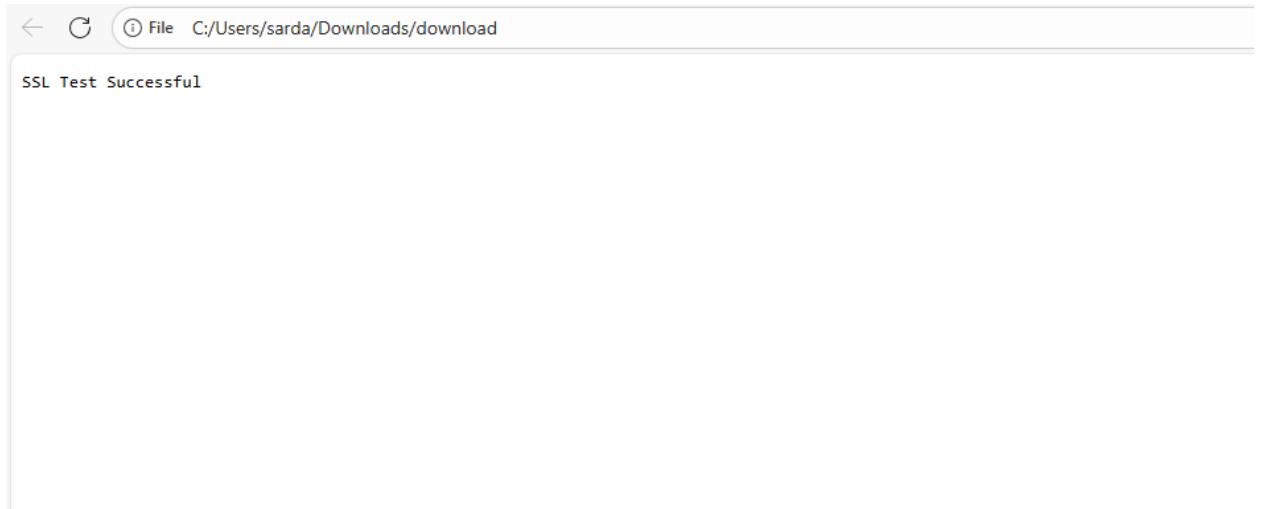
NET::ERR_CERT_AUTHORITY_INVALID

 Turn on enhanced protection to get Chrome's highest level of security

[Hide advanced](#) [Back to safety](#)

This server could not prove that it is **18.236.66.150**; its security certificate is not trusted by your computer's operating system. This may be caused by a misconfiguration or an attacker intercepting your connection.

[Proceed to 18.236.66.150 \(unsafe\)](#)



```
}ubuntu@ip-10-0-0-32:~$ curl -k https://18.236.66.150
SSL Test Successful
ubuntu@ip-10-0-0-32:~$
```

7. Phase 6 – PHP-FPM Integration

Steps performed:

- Installed PHP 8.3 + php-fpm
- Created /var/www/phpsite/html
- Added phpinfo.php
- Configured fastcgi_pass

```
ubuntu@ip-10-0-0-32:~$ cat /etc/nginx/sites-available/phpsite.com
server {
    listen 80;
    server_name 18.236.66.150;

    root /var/www/phpsite/html;
    index index.php index.html;

    location / {
        try_files $uri $uri/ =404;
    }

    location ~ \.php$ {
        include snippets/fastcgi-php.conf;
        fastcgi_pass unix:/run/php/php8.3-fpm.sock;
    }
}
ubuntu@ip-10-0-0-32:~$
```

The screenshot shows a web browser window with the URL `18.236.66.150/phpinfo.php`. The page title is "PHP Version 8.3.6". The content is a table of PHP configuration information.

System	Linux ip-10-0-0-32 6.14.0-1015-aws #15~24.04.1-Ubuntu SMP Tue Sep 23 22:44:48 UTC 2025 x86_64
Build Date	Jul 14 2025 18:30:55
Build System	Linux
Server API	FFM/FastCGI
Virtual Directory Support	disabled
Configuration File (php.ini) Path	/etc/php/8.3/fpm
Loaded Configuration File	/etc/php/8.3/fpm/php.ini
Scan this dir for additional .ini files	/etc/php/8.3/fpm/conf.d
Additional .ini files parsed	/etc/php/8.3/fpm/conf.d/10-mysqlind.ini, /etc/php/8.3/fpm/conf.d/10-opcache.ini, /etc/php/8.3/fpm/conf.d/10-pdo.ini, /etc/php/8.3/fpm/conf.d/20-calendar.ini, /etc/php/8.3/fpm/conf.d/20-ctype.ini, /etc/php/8.3/fpm/conf.d/20-exif.ini, /etc/php/8.3/fpm/conf.d/20-flt.ini, /etc/php/8.3/fpm/conf.d/20-filinfo.ini, /etc/php/8.3/fpm/conf.d/20-ftp.ini, /etc/php/8.3/fpm/conf.d/20-gettext.ini, /etc/php/8.3/fpm/conf.d/20-iconv.ini, /etc/php/8.3/fpm/conf.d/20-mysqli.ini, /etc/php/8.3/fpm/conf.d/20-pdo_mysqli.ini, /etc/php/8.3/fpm/conf.d/20-phar.ini, /etc/php/8.3/fpm/conf.d/20-posix.ini, /etc/php/8.3/fpm/conf.d/20-readline.ini, /etc/php/8.3/fpm/conf.d/20-shmop.ini, /etc/php/8.3/fpm/conf.d/20-sockets.ini, /etc/php/8.3/fpm/conf.d/20-sysvmsg.ini, /etc/php/8.3/fpm/conf.d/20-sysvsem.ini, /etc/php/8.3/fpm/conf.d/20-tokenizer.ini
PHP API	20230831
PHP Extension	20230831
Zend Extension	420230831
Zend Extension Build	API420230831.NTS
PHP Extension Build	API20230831.NTS
Debug Build	no
Thread Safety	disabled
Zend Signal Handling	enabled
Zend Memory Manager	enabled
Zend Multibyte Support	disabled

8. Phase 7 – URL Rewrites & Redirects

Tasks performed:

- ✓ /old → /new (301 redirect)
- ✓ /user/20 → /user.php?id=20
- ✓ Directory redirect /blog
- ✓ Trailing slash enforcement

```
server {
    listen 80;
    server_name 18.236.66.150;

    root /var/www/phpsite/html;
    index index.php index.html;

    # -----
    # PHASE 6 – URL RULES
    # -----

    # FIXED REDIRECT: matches /old AND /old/
    location ^~ /old {
        return 301 /new;
    }

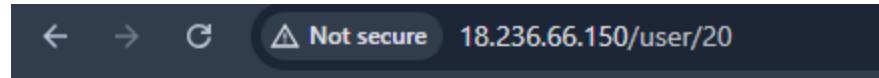
    # Pretty URL
    rewrite ^/user/([0-9]+)$ /user.php?id=$1 last;

    # Blog redirect
    location /blog {
        return 302 https://example.com;
    }

    # PHP site root
    location / {
        try_files $uri $uri/ =404;
    }

    # PHP-FPM
    location ~ \.php$ {
        include snippets/fastcgi-php.conf;
        fastcgi_pass unix:/run/php/php8.3-fpm.sock;
    }

    # Safe trailing slash
    rewrite ^([^.]*[^/])$ $1/ permanent;
}
```

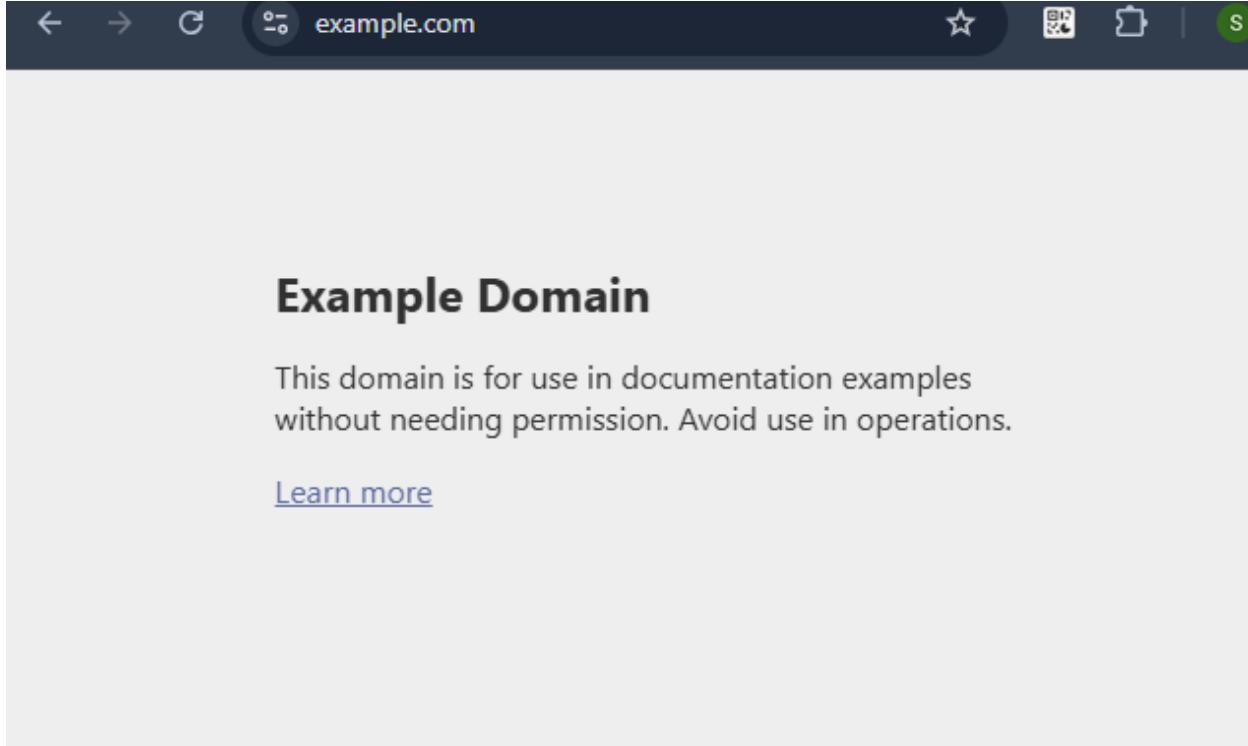


User ID from rewritten URL: 20

```
ubuntu@ip-10-0-0-32:~$ curl -I http://18.236.66.150/old
curl -I http://18.236.66.150/old/
HTTP/1.1 301 Moved Permanently
Server: nginx/1.24.0 (Ubuntu)
Date: Wed, 19 Nov 2025 16:42:07 GMT
Content-Type: text/html
Content-Length: 178
Location: http://18.236.66.150/new/
Connection: keep-alive

HTTP/1.1 301 Moved Permanently
Server: nginx/1.24.0 (Ubuntu)
Date: Wed, 19 Nov 2025 16:42:07 GMT
Content-Type: text/html
Content-Length: 178
Location: http://18.236.66.150/new
Connection: keep-alive

ubuntu@ip-10-0-0-32:~$
```



9. Phase 8 – Security Headers & Basic Auth

1. Security Headers added:

- X-Frame-Options

- X-XSS-Protection
- X-Content-Type-Options
- Referrer-Policy
- Permissions-Policy

Insert Screenshot:

- curl -I showing security headers

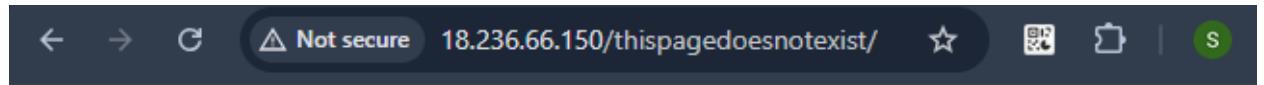
2. Basic Auth Steps:

- Created .htpasswd
- Added protected /admin route

10. Phase 9 – Custom Error Pages (404, 403, 500)

What I did:

- Created 3 HTML files
- Configured error_page directive
- Mapped using alias
- Tested manually



404 - Page Not Found

The resource you are trying to access does not exist.

```
# CUSTOM ERROR PAGES
error_page 404 /errors/404.html;
error_page 403 /errors/403.html;
error_page 500 502 503 504 /errors/500.html;

location /errors/ {
    alias /var/www/phpsite/html/errors/;
}

# PHP site root
location / {
    try_files $uri $uri/ =404;
}
```

11. Phase 10 – Multi-site Hosting

What I did:

- Hosted multiple server blocks:
 - example.com
 - phpsite.com

- o reverseproxy.com
- o ssl-test.com

```
hanged on disk. Run 'systemctl daemon-reload' to reload units.
ubuntu@ip-10-0-0-32:~$ ls /etc/nginx/sites-available
default      loadbalancer.com  reverseproxy.com
example.com  phpsite.com      ssl-test.com
ubuntu@ip-10-0-0-32:~$
```

12. Phase 11 – nginx.conf Architecture & Includes

What I learned:

- Structure:
 - o events
 - o http
 - o server
 - o location
- include /sites-enabled/*

```
GNU nano 7.2
user www-data;
worker_processes auto;
pid /run/nginx.pid;
error_log /var/log/nginx/error.log;
include /etc/nginx/modules-enabled/*.conf;

events {
    worker_connections 768;
    # multi_accept on;
}

http {

    proxy_cache_path /var/cache/nginx/mycache levels=1:2 keys_zone=mycache:10m inactive=60m max_size=100m;

    limit_req_zone $binary_remote_addr zone=one:10m rate=5r/s;
    limit_conn_zone $binary_remote_addr zone=addr:10m;

    # Basic Settings
    #

    sendfile on;
    tcp_nopush on;
    types_hash_max_size 2048;
    # server_tokens off;

    # server_names_hash_bucket_size 64;
    # server_name_in_redirect off;

    include /etc/nginx/mime.types;
    default_type application/octet-stream;

    #
    # SSL Settings
    #

    ssl_protocols TLSv1 TLSv1.1 TLSv1.2 TLSv1.3; # Dropping SSLv3, ref: POODLE
    ssl_prefer_server_ciphers on;

    #
    # Logging Settings
    #

    access_log /var/log/nginx/access.log;
```

AC Help A Write Out A Where Is A Cut A Execute A Location X-U Undo X-A Set Mark X-T To Bracket X-Q Prev

13. Phase 12 – Variables (\$host, \$remote_addr, etc.)

Variables used:

- \$host
- \$remote_addr
- \$request_uri
- \$proxy_add_x_forwarded_for

```
proxy_set_header Host $host;
proxy_set_header X-Real-IP $remote_addr;
proxy_set_header X-Forwarded-For $proxy_add_x_forwarded_for;
proxy_set_header X-Forwarded-Proto $scheme;
```

```
}
```

14. Phase 13 – Rate Limiting + Connection Limits

Global settings added:

```
limit_req_zone $binary_remote_addr zone=one:10m rate=5r/s;
```

```
limit_conn_zone $binary_remote_addr zone=addr:10m;
```

Applied in phpsite:

```
limit_req zone=one burst=10 nodelay;
```

```
limit_conn addr 10;
```

```
GNU nano 7.2                                     /etc/nginx/nginx.conf
user www-data;
worker_processes auto;
pid /run/nginx.pid;
error_log /var/log/nginx/error.log;
include /etc/nginx/modules-enabled/*.conf;

events {
    worker_connections 768;
    # multi_accept on;
}

http {

    proxy_cache_path /var/cache/nginx/mycache levels=1:2 keys_zone=mycache:10m inactive=60m max_size=100m;

    limit_req_zone $binary_remote_addr zone=one:10m rate=5r/s;
    limit_conn_zone $binary_remote_addr zone=addr:10m;

    # Basic Settings
    ##

    sendfile on;
    tcp_nopush on;
    types_hash_max_size 2048;
    # server_tokens off;

    # server_names_hash_bucket_size 64;
    # server_name_in_redirect off;

    include /etc/nginx/mime.types;
    default_type application/octet-stream;

    ##
    # SSL Settings
    ##

    ssl_protocols TLSv1 TLSv1.1 TLSv1.2 TLSv1.3; # Dropping SSLv3, ref: POODLE
    ssl_prefer_server_ciphers on;

    ##
    # Logging Settings
    ##

    access_log /var/log/nginx/access.log;
}
```

⌘ Help ⌘ Write Out ⌘ Where Is ⌘ Cut ⌘ Execute ⌘ Location ⌘ Undo ⌘ Set Mark ⌘ To Bracket ⌘ Prev

```
GNU nano 7.2                                         /etc/nginx/sites-available/phpsi
server {
    listen 80;
    server_name 18.236.66.150;

    root /var/www/phpsite/html;
    index index.php index.html;

    # -----
    # PHASE 6 - URL RULES
    # -----

    # FIXED REDIRECT: matches /old AND /old/
    location ^~ /old {
        return 301 /new;
    }

    # Pretty URL
    rewrite ^/user/([0-9]+)$ /user.php?id=$1 last;

    # Blog redirect
    location /blog {
        return 302 https://example.com;
    }

    # CUSTOM ERROR PAGES
    error_page 404 /errors/404.html;
    error_page 403 /errors/403.html;
    error_page 500 502 503 504 /errors/500.html;

    location /errors/ {
        alias /var/www/phpsite/html/errors/;
    }

    # BASIC AUTH (ADMIN AREA)
    # -----
    location ^~ /admin/ {
        auth_basic "Restricted Area";
        auth_basic_user_file /etc/nginx/.htpasswd;
        try_files $uri $uri/ =404;
    }

    location = /admin {
        return 301 /admin/;
    }

    # PHP site root
    location / {
```

```
# PHP site root
location / {
    limit_req zone=one burst=10 nodelay;
    limit_conn addr 10;

    try_files $uri $uri/ =404;
}

# PHP-FPM
location ~ \.php$ {
    include snippets/fastcgi-php.conf;
    fastcgi_pass unix:/run/php/php8.3-fpm.sock;
}

# Safe trailing slash
# SAFE TRAILING SLASH RULE
if ($request_uri ~* "^(.+[^/])$") {
    return 301 $1/;
}

}
```

16. Phase 15 – WebSocket Support

Config added:

```
proxy_http_version 1.1;

proxy_set_header Upgrade $http_upgrade;
proxy_set_header Connection "upgrade";
```

```
# --- WEBSOCKET SUPPORT ---
proxy_http_version 1.1;
proxy_set_header Upgrade $http_upgrade;
proxy_set_header Connection "upgrade";
```

17. Phase 16 – Log Monitoring

Commands used:

```
tail -f /var/log/nginx/access.log
```

```
tail -f /var/log/nginx/error.log
```

```
[C]ubuntu@ip-10-0-0-32:~$ sudo tail -f /var/log/nginx/access.log
206.84.156.183 -- [19/Nov/2025:17:24:28 +0000] "GET /admin/ HTTP/1.1" 401 590 "-" "Mozilla/5.0 (Windows NT 10.0; Win64; x64) AppleWebKit/537.36 (KHTML, like Gecko) Chrome/142.0.0.0 Safari/537.36"
18.236.66.150 -- [19/Nov/2025:17:25:23 +0000] "HEAD /admin HTTP/1.1" 301 0 "-" "curl/8.5.0"
206.84.156.183 -- [19/Nov/2025:17:33:10 +0000] "GET /admin/ HTTP/1.1" 401 590 "-" "Mozilla/5.0 (Windows NT 10.0; Win64; x64) AppleWebKit/537.36 (KHTML, like Gecko) Chrome/142.0.0.0 Safari/537.36"
206.84.156.183 -- [19/Nov/2025:17:33:11 +0000] "GET /admin/ HTTP/1.1" 401 590 "-" "Mozilla/5.0 (Windows NT 10.0; Win64; x64) AppleWebKit/537.36 (KHTML, like Gecko) Chrome/142.0.0.0 Safari/537.36"
206.84.156.183 -- [19/Nov/2025:17:40:06 +0000] "GET /phpinfo.php HTTP/1.1" 301 178 "-" "Mozilla/5.0 (Windows NT 10.0; Win64; x64) AppleWebKit/537.36 (KHTML, like Gecko) Chrome/142.0.0.0 Safari/537.36"
206.84.156.183 -- [19/Nov/2025:17:40:06 +0000] "GET /phpinfo.php HTTP/1.1" 404 202 "-" "Mozilla/5.0 (Windows NT 10.0; Win64; x64) AppleWebKit/537.36 (KHTML, like Gecko) Chrome/142.0.0.0 Safari/537.36"
206.84.156.183 -- [19/Nov/2025:17:40:16 +0000] "GET /index.php HTTP/1.1" 301 178 "-" "Mozilla/5.0 (Windows NT 10.0; Win64; x64) AppleWebKit/537.36 (KHTML, like Gecko) Chrome/142.0.0.0 Safari/537.36"
206.84.156.183 -- [19/Nov/2025:17:40:17 +0000] "GET /index.php HTTP/1.1" 404 202 "-" "Mozilla/5.0 (Windows NT 10.0; Win64; x64) AppleWebKit/537.36 (KHTML, like Gecko) Chrome/142.0.0.0 Safari/537.36"
206.84.156.183 -- [19/Nov/2025:17:41:45 +0000] "GET / HTTP/1.1" 403 194 "-" "Mozilla/5.0 (Windows NT 10.0; Win64; x64) AppleWebKit/537.36 (KHTML, like Gecko) Chrome/142.0.0.0 Safari/537.36"
206.84.156.183 -- [19/Nov/2025:17:41:50 +0000] "GET / HTTP/1.1" 403 194 "-" "Mozilla/5.0 (Windows NT 10.0; Win64; x64) AppleWebKit/537.36 (KHTML, like Gecko) Chrome/142.0.0.0 Safari/537.36"
206.84.156.183 -- [19/Nov/2025:17:44:58 +0000] "GET / HTTP/1.1" 403 194 "-" "Mozilla/5.0 (Windows NT 10.0; Win64; x64) AppleWebKit/537.36 (KHTML, like Gecko) Chrome/142.0.0.0 Safari/537.36"
206.84.156.183 -- [19/Nov/2025:17:45:01 +0000] "GET /index.php HTTP/1.1" 404 202 "-" "Mozilla/5.0 (Windows NT 10.0; Win64; x64) AppleWebKit/537.36 (KHTML, like Gecko) Chrome/142.0.0.0 Safari/537.36"
206.84.156.183 -- [19/Nov/2025:17:45:06 +0000] "GET /index.php HTTP/1.1" 404 202 "-" "Mozilla/5.0 (Windows NT 10.0; Win64; x64) AppleWebKit/537.36 (KHTML, like Gecko) Chrome/142.0.0.0 Safari/537.36"
206.84.156.183 -- [19/Nov/2025:17:45:16 +0000] "GET / HTTP/1.1" 403 194 "-" "Mozilla/5.0 (Windows NT 10.0; Win64; x64) AppleWebKit/537.36 (KHTML, like Gecko) Chrome/142.0.0.0 Safari/537.36"
206.84.156.183 -- [19/Nov/2025:17:45:18 +0000] "GET / HTTP/1.1" 403 194 "-" "Mozilla/5.0 (Windows NT 10.0; Win64; x64) AppleWebKit/537.36 (KHTML, like Gecko) Chrome/142.0.0.0 Safari/537.36"
```

```
ubuntu@ip-10-0-0-32:~$ sudo tail -f /var/log/nginx/error.log
2025/11/19 13:15:30 [notice] 1744#1744: using inherited sockets from "5;6;"
2025/11/19 17:17:10 [error] 20449#20449: *38 directory index of "/var/www/phpsite/html/" is forbidden, client: 18.236.66.150, server: 18.236.66.150, request: "HEAD / HTTP/1.1", host: "18.236.66.150"
2025/11/19 17:41:45 [error] 20644#20644: *51 directory index of "/var/www/phpsite/html/" is forbidden, client: 206.84.156.183, server: 18.236.66.150, request: "GET / HTTP/1.1", host: "18.236.66.150"
2025/11/19 17:41:50 [error] 20644#20644: *51 directory index of "/var/www/phpsite/html/" is forbidden, client: 206.84.156.183, server: 18.236.66.150, request: "GET / HTTP/1.1", host: "18.236.66.150"
2025/11/19 17:44:58 [error] 20644#20644: *54 directory index of "/var/www/phpsite/html/" is forbidden, client: 206.84.156.183, server: 18.236.66.150, request: "GET / HTTP/1.1", host: "18.236.66.150"
2025/11/19 17:45:16 [error] 20644#20644: *54 directory index of "/var/www/phpsite/html/" is forbidden, client: 206.84.156.183, server: 18.236.66.150, request: "GET / HTTP/1.1", host: "18.236.66.150"
2025/11/19 17:45:18 [error] 20644#20644: *54 directory index of "/var/www/phpsite/html/" is forbidden, client: 206.84.156.183, server: 18.236.66.150, request: "GET / HTTP/1.1", host: "18.236.66.150"
```

```
[C]ubuntu@ip-10-0-0-32:~$ sudo touch /var/www/phpsite/html/secret.txt
sudo chmod 000 /var/www/phpsite/html/secret.txt
ubuntu@ip-10-0-0-32:~$ sudo tail -f /var/log/nginx/error.log
2025/11/19 13:15:30 [notice] 1744#1744: using inherited sockets from "5;6;"
2025/11/19 17:17:10 [error] 20449#20449: *38 directory index of "/var/www/phpsite/html/" is forbidden, client: 18.236.66.150, server: 18.236.66.150, request: "HEAD / HTTP/1.1", host: "18.236.66.150"
2025/11/19 17:41:45 [error] 20644#20644: *51 directory index of "/var/www/phpsite/html/" is forbidden, client: 206.84.156.183, server: 18.236.66.150, request: "GET / HTTP/1.1", host: "18.236.66.150"
2025/11/19 17:41:50 [error] 20644#20644: *51 directory index of "/var/www/phpsite/html/" is forbidden, client: 206.84.156.183, server: 18.236.66.150, request: "GET / HTTP/1.1", host: "18.236.66.150"
2025/11/19 17:44:58 [error] 20644#20644: *54 directory index of "/var/www/phpsite/html/" is forbidden, client: 206.84.156.183, server: 18.236.66.150, request: "GET / HTTP/1.1", host: "18.236.66.150"
2025/11/19 17:45:16 [error] 20644#20644: *54 directory index of "/var/www/phpsite/html/" is forbidden, client: 206.84.156.183, server: 18.236.66.150, request: "GET / HTTP/1.1", host: "18.236.66.150"
2025/11/19 17:45:18 [error] 20644#20644: *54 directory index of "/var/www/phpsite/html/" is forbidden, client: 206.84.156.183, server: 18.236.66.150, request: "GET / HTTP/1.1", host: "18.236.66.150"
```

```
ubuntu@ip-10-0-0-32:~$ cat /etc/logrotate.d/nginx
/var/log/nginx/*.log {
    daily
    missingok
    rotate 14
    compress
    delaycompress
   notifempty
    create 0640 www-data adm
    sharedscripts
    prerotate
        if [ -d /etc/logrotate.d/httpd-prerotate ]; then \
            run-parts /etc/logrotate.d/httpd-prerotate;
        fi \
    endscript
    postrotate
        invoke-rc.d nginx rotate >/dev/null 2>&1
    endscript
}
ubuntu@ip-10-0-0-32:~$ |
```

Challenges Faced During the Project

1. Server Block Conflicts (Multiple Sites Overlapping)

When hosting multiple sites (phpsite.com, reverseproxy.com, example.com, ssl-test.com), I faced issues where the wrong server block was responding.

Example:

Requests going to the EC2 public IP were getting served by the wrong config.

What caused the issue?

- Overlapping server_name
- Default server taking priority

How I fixed it

- Verified active configs using:
- ls /etc/nginx/sites-enabled/
- Corrected server_name values
- Ensured no accidental wildcard server was overriding others
- Reloaded configs and validated using curl

2. “403 Forbidden” and “502 Bad Gateway” Errors

Several times I received:

403 Forbidden

502 Bad Gateway

What caused it?

- Wrong document root permissions
- Backend service not running
- Proxying to incorrect port

Fixes applied

- Restarted backend service
- Verified socket paths for PHP-FPM
- Checked file permissions using ls -l
- Updated proxy_pass to correct port

3. NGINX Configuration Failing with Syntax Errors

Sometimes nginx failed to reload:

nginx: [emerg] "location" directive is not allowed here

Reason

- Misplaced braces {}
- Extra characters above server {
- Wrong indentation or missing semicolon

Solution

- Rebuilt config files cleanly
- Validated using:
- sudo nginx -t
- Ensured proper nesting of:
 - http {}
 - server {}
 - location {}

4. PHP-FPM Socket Issues (Page not loading)

At first, PHP pages were not executing.

Cause

- Wrong socket path
- Missing PHP package
- Incorrect FastCGI configuration

Fix

- Verified PHP version using:
- php -v
- Corrected:
- fastcgi_pass unix:/run/php/php8.3-fpm.sock;

- Installed missing modules

5. Rewrite Rules Not Working Initially

Some URLs were not rewriting as expected.

Reason

- Wrong regex pattern
- Rewrite conflicts with try_files
- Testing wrong URL format

Fix

- Adjusted regex
- Moved rewrite rules above location blocks
- Tested with curl to confirm 301 redirects

6. Basic Authentication Causing Infinite Loading

When configuring /admin authentication, the browser sometimes kept loading without prompting for username and password.

Reason

- Missing file:
- /etc/nginx/.htpasswd
- Misplaced location blocks
- Conflicts with redirect rules

Fix

- Recreated htpasswd
- Placed /admin location above PHP blocks
- Reloaded nginx

7. Proxy Cache Validation Issues (Backend Not Running)

Caching (proxy_cache) did not show HIT/MISS because the backend service wasn't active.

What I learned

Even if the cache isn't testable, the configuration logic matters:

- Cache path
- Key zone
- Cache headers

I validated config structure even if runtime behavior couldn't be fully demonstrated.

8. Understanding Access Logs and Error Logs

At first, logs were overwhelming because of so many entries.

How I managed it

- Filtered using:
- tail -f | grep
- Triggered deliberate 404/403/500
- Observed nginx logging behavior