

# Node application Hosting

Last updated by | shakibesaib | Sep 29, 2023 at 3:08 PM GMT+6

## ON APACHE:

- Installing node npm

```
devops@devops-Lenovo-G410:~$ sudo apt install nodejs npm
[sudo] password for devops:
Reading package lists... Done
Building dependency tree... Done
Reading state information... Done
The following additional packages will be installed:
  build-essential g++ g++-11 gyp libjs-events libjs-highlight.js
  libjs-inherits libjs-is-typedarray libjs-psl libjs-source-map
  libjs-sprintf-js libjs-typedarray-to-buffer libnode-dev libnode72 libssl-dev
  libstdc++-11-dev libuv1-dev node-abab node-abbrev node-agent-base
  node-ansi-regex node-ansi-styles node-ansistyles node-aproba node-archy
  node-are-we-there-yet node-argparse node-arrify node-asap node-asynckit
  node-balanced-match node-brace-expansion node-builtins node-cacache
  node-chalk node-chownr node-clean-yaml-object node-cli-table node-clone
  node-color-convert node-color-name node-colors node-columnify
  node-combined-stream node-commander node-console-control-strings
  node-copy-concurrently node-core-util-is node-coveralls node-cssom
  node-cssstyle node-debug node-decompress-response node-defaults
  node-delayed-stream node-delegates node-depd node-diff node-encoding
  node-end-of-stream node-err-code node-escape-string-regexp node-esprima
  node-events node-fancy-log node-fetch node-foreground-child node-form-data
```

- Creating a Node.js application

```
devops@devops-Lenovo-G410:~$ npm init -y
Wrote to /home/devops/package.json:

{
  "name": "devops",
  "version": "1.0.0",
  "description": "",
  "main": "index.js",
  "scripts": {
    "test": "echo \"Error: no test specified\" && exit 1"
  },
  "keywords": [],
  "author": "",
  "license": "ISC"
}

devops@devops-Lenovo-G410:~$ npm install express

added 58 packages, and audited 59 packages in 8s

8 packages are looking for funding
  run `npm fund` for details

found 0 vulnerabilities
```

- Creating a new file called app.js in our project directory and add the following code & Testing the Node.js application:

```
const express = require('express');
const app = express();

app.get('/', (req, res) => {
  res.send('Hello, Node.js!');
});

app.listen(3000, () => {
  console.log('Server is running on http://localhost:3000');
});
```

```
devops@devops-Lenovo-G410:~/Desktop$ ls
Backup.sql      Laravel         Nodejs         School.sql
devopsIntern.sql mywebsite.html QUICKLY        'Tittle diya ki hbe'
devops@devops-Lenovo-G410:~/Desktop$ cd Nodejs
devops@devops-Lenovo-G410:~/Desktop/Nodejs$ node app.js
Server is running on http://localhost:3000
^C
```

- Configuring Apache to proxy requests

```
devops@devops-Lenovo-G410:~/Desktop/Nodejs$ sudo a2enmod proxy proxy_http
Module proxy already enabled
Considering dependency proxy for proxy_http:
Module proxy already enabled
Enabling module proxy_http.
To activate the new configuration, you need to run:
systemctl restart apache2
```

- Creating a new Apache configuration file by running the following command:

```
devops@devops-Lenovo-G410:~$ sudo nano /etc/apache2/sites-available/nodeapp.conf
```

Adding the following configuration:

```
<VirtualHost *:80>
  ServerName localhost

  ProxyPass / http://localhost:3000/
  ProxyPassReverse / http://localhost:3000/
</VirtualHost>
```

- Enabling the Apache configuration

```
devops@devops-Lenovo-G410:~$ sudo a2ensite nodeapp.conf
Enabling site nodeapp.
To activate the new configuration, you need to run:
systemctl reload apache2
devops@devops-Lenovo-G410:~$ sudo service apache2 restart
```

- Updating hosts-

Adding our server name.

```
devops@devops-Lenovo-G410:~$ sudo nano /etc/hosts
```

- Restarting Apache:

```
devops@devops-Lenovo-G410:~$ sudo service apache2 restart
```

## Nginx configuration:

```
server {  
    listen 80;  
    server_name nodejsnginx.local;  
  
    location / {  
        proxy_pass http://localhost:3000;  
        proxy_http_version 1.1;  
        proxy_set_header Upgrade $http_upgrade;  
        proxy_set_header Connection 'upgrade';  
        proxy_set_header Host $host;  
        proxy_cache_bypass $http_upgrade;  
    }  
}
```