

Hosting a Project on Forge

Laravel Forge makes hosting a project super easy, but there are some things worth going over to make sure everything is done correctly.

Prerequisites

- You will need access to Laravel Forge.
- The project must have a domain set up on Cloudflare pointing to `134.209.65.245`.
- The project will require a repository hosted on GitHub.

Create a New Site

The first thing you'll need to do is create a new site. Expand the "New Site" form on the page for the Vegan Hacktivists server:

The screenshot shows the Forge web interface for a server named "vegan-hacktivists". At the top, there are buttons for "Restart" and "Stop". Below the server name, it shows "Public IP 134.209.65.245", "Private IP 10.136.98.115", and "Region New York 1". On the left is a sidebar with navigation links: Sites, Database, SSH Keys, Monitoring, Backups, PHP, Packages, Nginx Templates, Scheduler, Daemons, Network, Logs, Events, Integrations, and Meta. The main area is titled "New Site" and contains a blue informational box with a red square icon in its top right corner. The box contains text about sites and a note about Website Isolation. Below this are several form fields: "Root Domain" (veganproject.org), "Aliases" (second-domain.com,third-domain.com), "Project Type" (General PHP / Laravel), "Web Directory" (/public), "PHP Version" (PHP 8.1), "Allow Wildcard Sub-Domains" (unchecked), "Use Website Isolation" (checked), "Website Isolation Username" (veganprojectorg), "Create Database" (checked), and "Database Name" (veganproject). An "Add" button is at the bottom right.

vegan-hacktivists

Public IP 134.209.65.245 Private IP 10.136.98.115 Region New York 1

Sites

Database SSH Keys Monitoring Backups PHP Packages Nginx Templates Scheduler Daemons Network Logs Events Integrations Meta

New Site

Think of sites as representing each "domain" on your server. The "default" site is included with each freshly provisioned server; however, you should delete it and create a new site with a valid domain name when you are ready to launch your production site. If you need to host additional domains or sub-domains, you may add them here.

When using Website Isolation, Forge will automatically create an isolated PHP-FPM process for the given site.

Root Domain
veganproject.org

Aliases
second-domain.com,third-domain.com

Project Type
General PHP / Laravel

Web Directory
/public

PHP Version
PHP 8.1

☐ Allow Wildcard Sub-Domains

☒ Use Website Isolation

Website Isolation Username
veganprojectorg

☒ Create Database

Database Name
veganproject

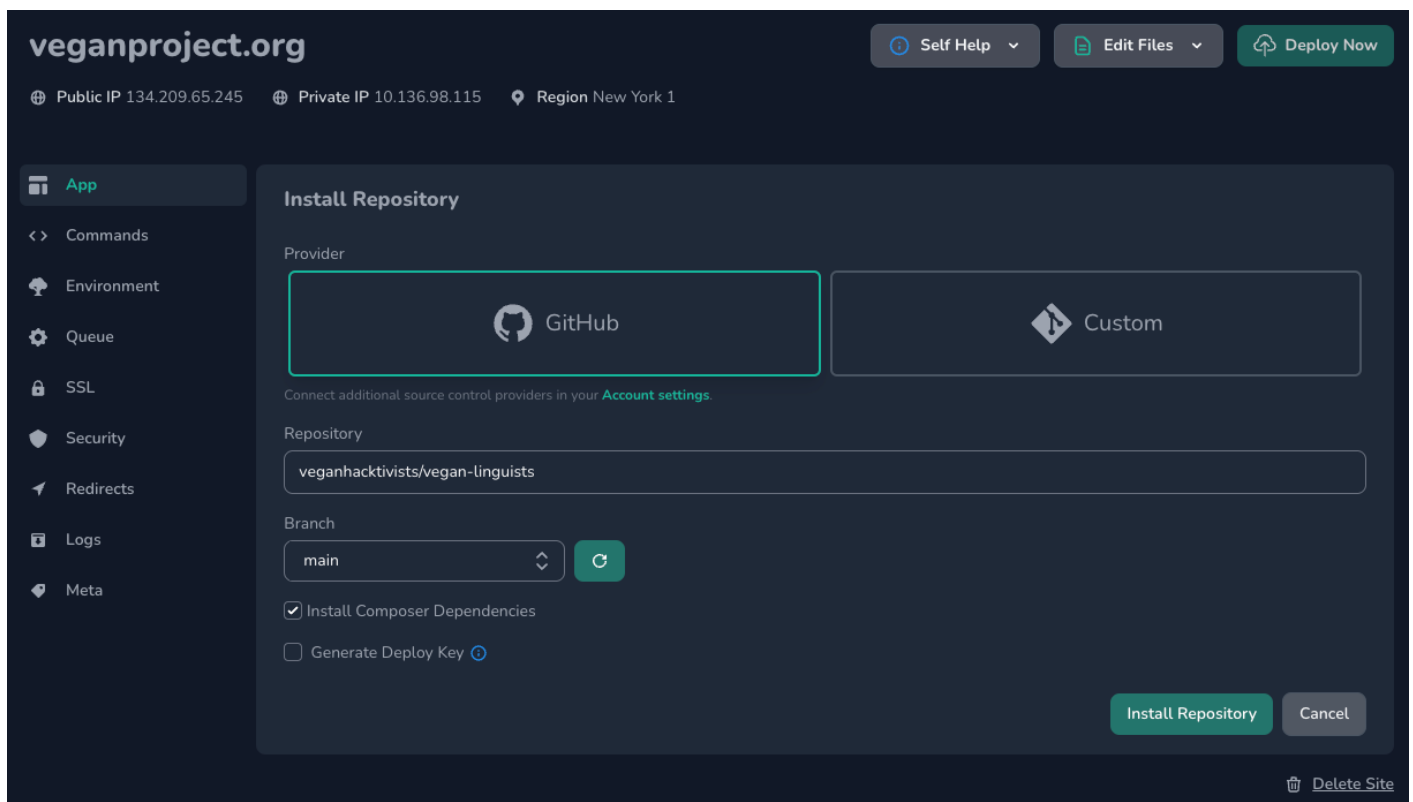
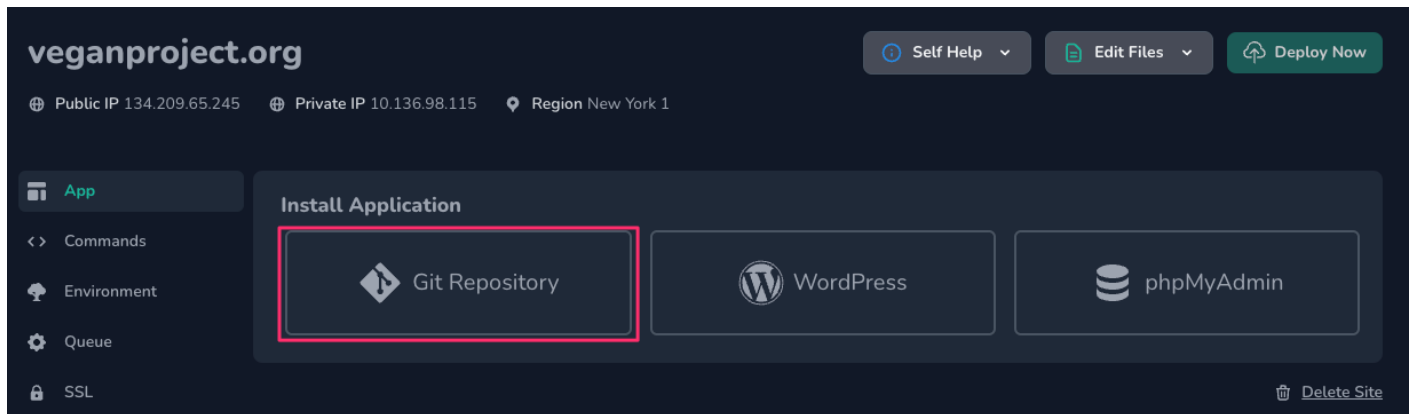
Add

Fill in the form using the project details.

- The "Root Domain" will of course be the domain name associated with the project.
- **The "Project Type" will likely be "General PHP / Laravel,"** but if it's a static website, choose "Static HTML." It will never be any of the Symfony options, and "Laravel Octane" should only be chosen if you *really* know what you're doing.
- **For non-static websites, make sure to check the "Website Isolation" option.** This will make sure that the project is running its own PHP-FPM instance, which will help prevent problems from one project affecting other projects.
- In the likely event that your project requires a database, now is a good time to do that. Typically, we name the database after the project name, with no underscores and only lowercase characters (e.g. "Vegan Linguists" becomes "veganlinguists").

Connect to the GitHub Repository

After creating the project, you should be redirected to the project's page where you can connect it to the GitHub Repository:



Make sure to fill in the "Repository" and "Branch" fields.

If this is a PHP project, make sure to keep "Install Composer Dependencies" checked. If it's a static website that doesn't require PHP packages, you can uncheck it.

⚠ Note

Forge confusingly shows `main` in the "Branch" field, but this is simply placeholder text, and not a default value. You will actually need to choose an option from the dropdown or type in the name of the branch manually.

Once you are done, click the "Install Repository" button.

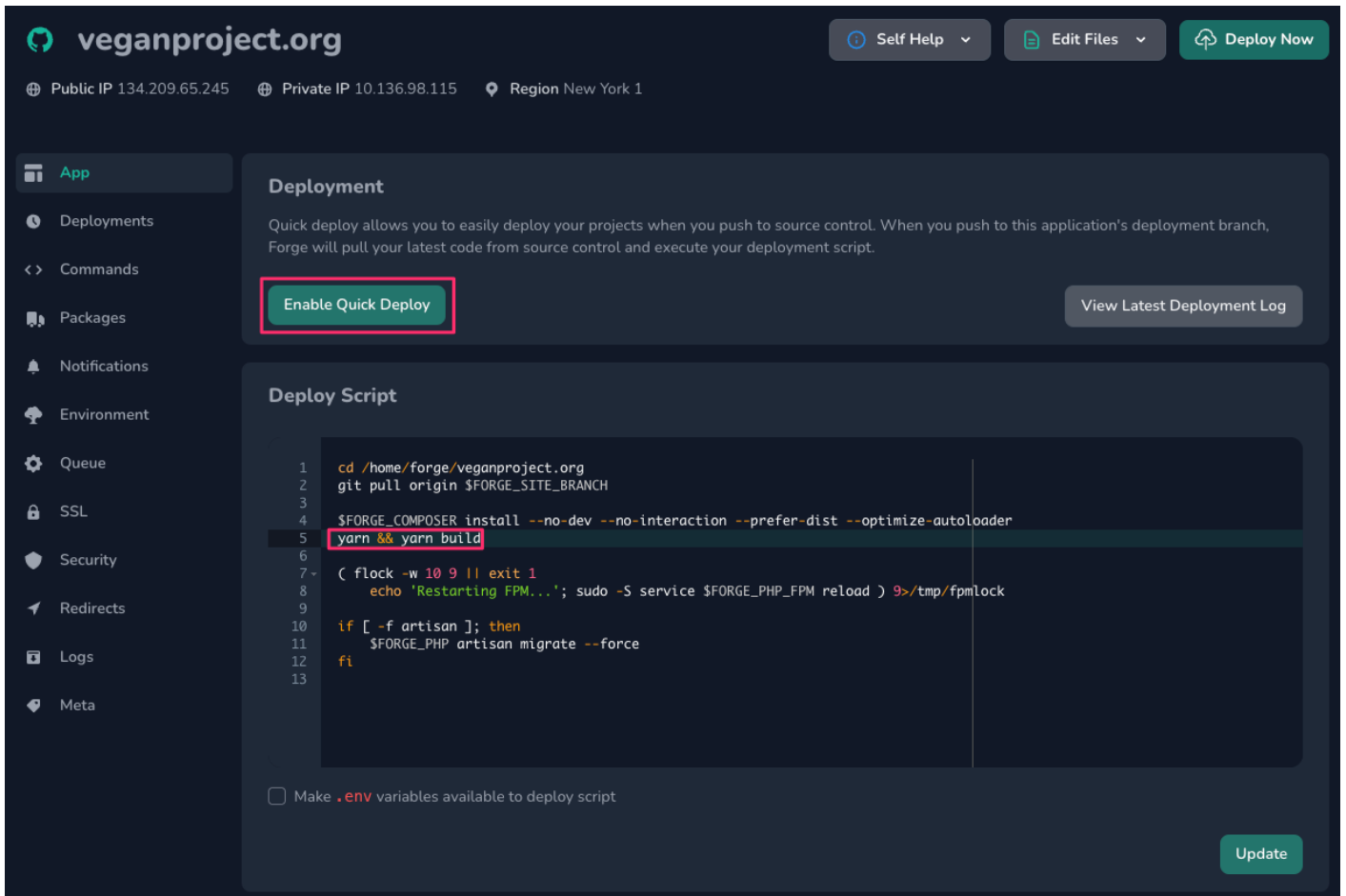
Setting Up Deployment

After the repository is installed, we need to make a quick change to the deploy script. By default, Node dependencies aren't installed and assets aren't built, so we need to make sure all of that happens.

After the line in the deploy script, add the following (or your project's equivalent if production assets are built differently):

```
yarn && yarn build
```

Also, you will likely want to click the "Enable Quick Deploy" button so that deployment happens as soon as new changes are pushed to the `main` branch.



The screenshot shows the Forge dashboard for the application `veganproject.org`. The top navigation bar includes links for `Self Help`, `Edit Files`, and `Deploy Now`. The left sidebar contains various management options like `App`, `Deployments`, `Commands`, `Packages`, `Notifications`, `Environment`, `Queue`, `SSL`, `Security`, `Redirects`, `Logs`, and `Meta`. The main content area is divided into two sections: **Deployment** and **Deploy Script**.

In the **Deployment** section, there is a description of quick deployment and a button labeled `Enable Quick Deploy`, which is highlighted with a red box. A `View Latest Deployment Log` button is also present.

The **Deploy Script** section displays a list of commands to be executed during deployment. The command `yarn && yarn build` on line 5 is highlighted with a red box. The script includes commands for pulling code, installing dependencies, building assets, and restarting the FPM service.

```
1 cd /home/forge/veganproject.org
2 git pull origin $FORGE_SITE_BRANCH
3
4 $FORGE_COMPOSER install --no-dev --no-interaction --prefer-dist --optimize-autoloader
5 yarn && yarn build
6
7 ( flock -w 10 9 || exit 1
8   echo 'Restarting FPM...'; sudo -S service $FORGE_PHP_FPM reload ) 9>/tmp/fpmlock
9
10 if [ -f artisan ]; then
11   $FORGE_PHP artisan migrate --force
12 fi
13
```

At the bottom of the script section, there is a checkbox labeled `Make .env variables available to deploy script` and an `Update` button.

Create a Database User

The next step is to create a user that can access the project's database. Head to the server's [databases section](#) and search for the "add database user" form:

Add Database User

Name
veganproject_user

Password
3mw0lw2bu9ttngwrpva0

Can Access

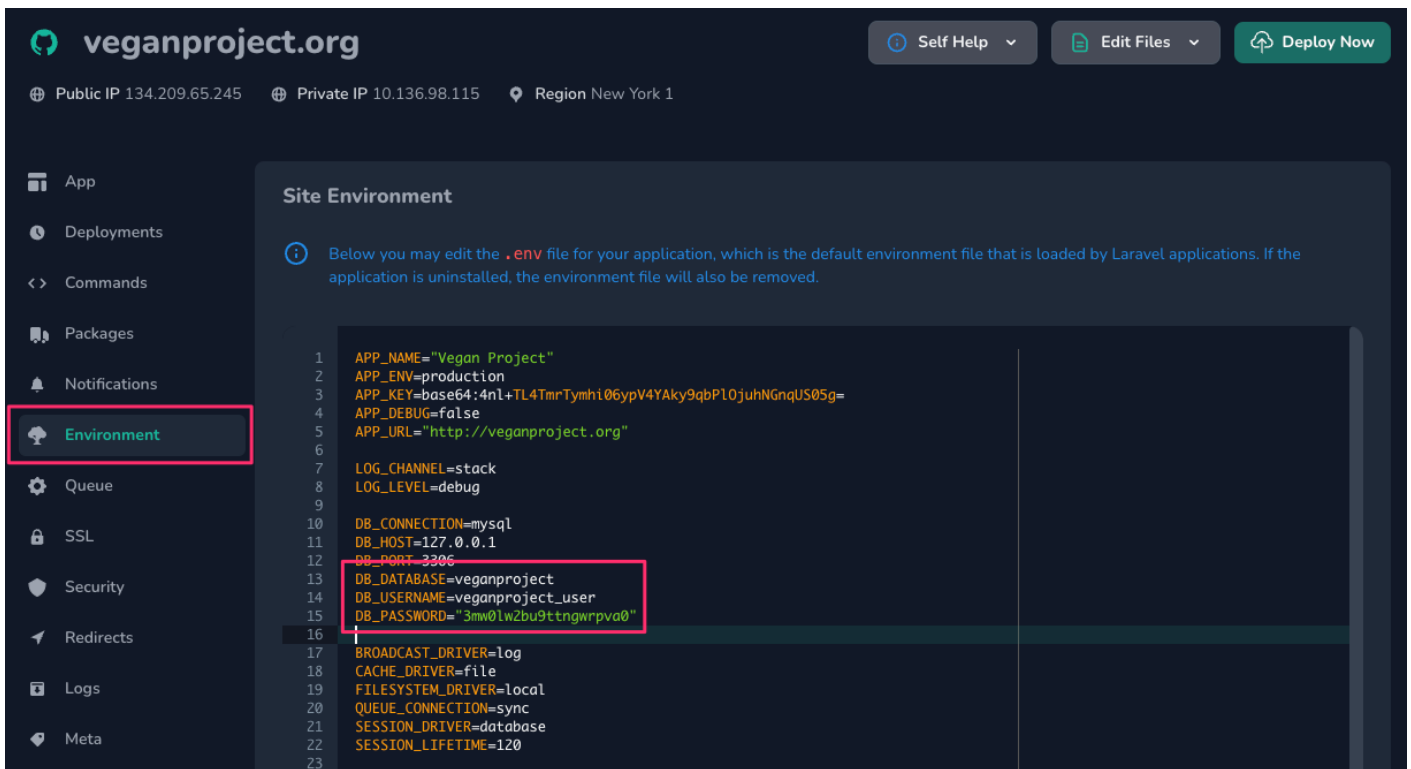
☐ 3movies	☐ 5m5v	☐ 5minutes5vegans
☐ activisthub_staging	☐ analytics_5m5v	☐ animalexploitationorg
☐ docsveganhacktivists	☐ fastcommunity	☐ karedo2_vegan
☐ meatthevictims	☐ mydailydozen	☐ pitchfta
☐ plantchallenge	☐ plantdiningpartnerships	☐ staging_veganlinguists
☐ startpressuring	☐ vb_community_phpbb	☐ veganactivism
☐ veganactivism_bot_click_counts	☐ veganbootcamp_dev	☐ veganbootcamp_staging
☐ veganfasttrack	☒ veganproject	☐ vrdntf_PicoGames
☐ vrdntf_veganactivism	☐ watchdominion	☐ wtfactivism
☐ Select All Databases		

Create

1. For the "Name" field, we typically use the name of the database followed by `_user`.
2. Click the ⚙ gear button next to the "Password" field to generate a password for this user.
3. **Select *only* the database associated with the project.**
4. **Copy the password to your clipboard for the next step.**
5. Click the "Create" button.

Update the `.env` File

Forge makes updating a project's environment variables very easy. Navigate to the "Environment" section of your project's page and update the `DB_Database`, `DB_USERNAME`, and `DB_PASSWORD` entries:

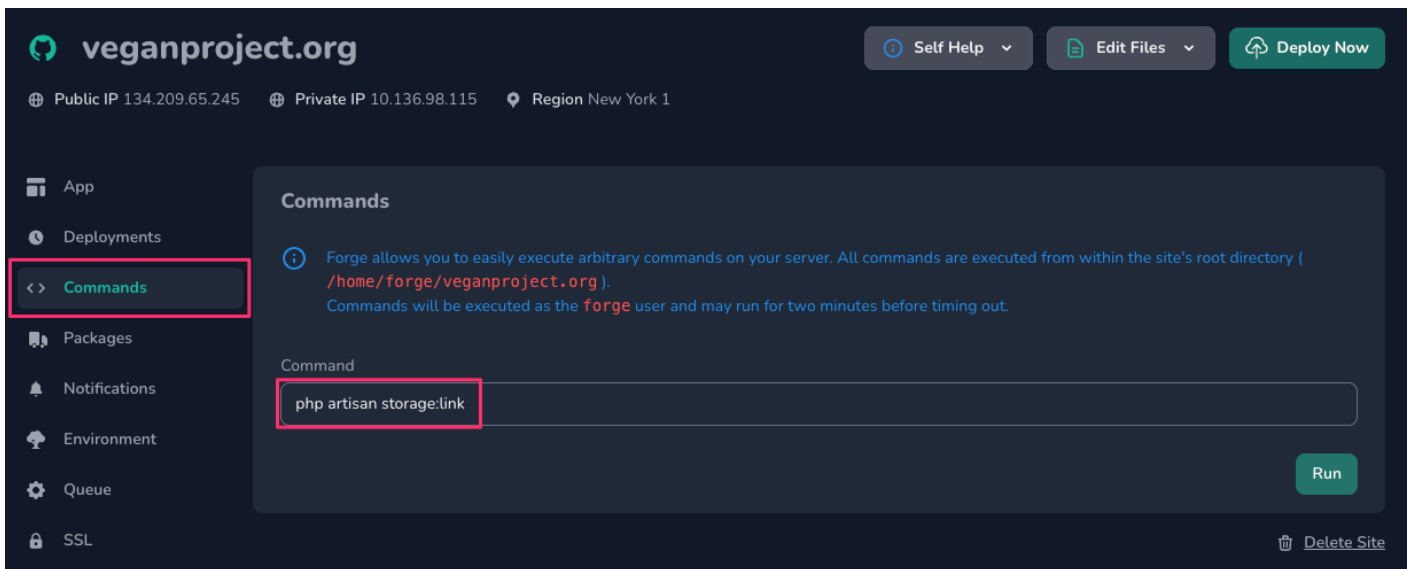


You can, of course, modify other variables, but the defaults should be good for most projects unless you know what you're doing! The main exception would be the `MAIL_` variables, which [we cover elsewhere](#).

Set Up the Storage Directory

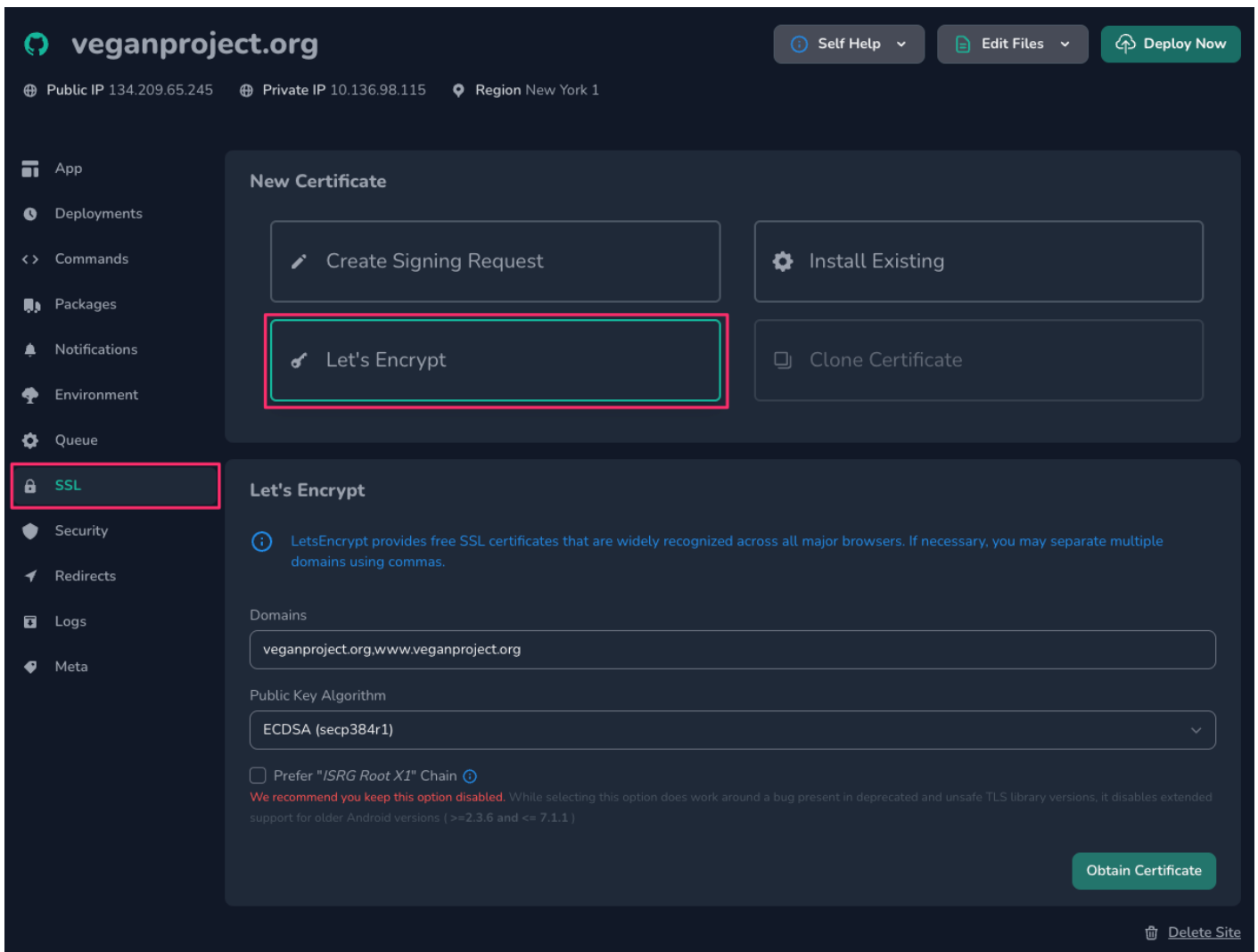
Laravel handles uploaded files using a [symlinked folder](#), which is something you must set up yourself. Luckily, there is an `artisan` command to do this, which we can run directly from Forge.

Head to the "Commands" section on your project's page and choose the `php artisan storage:link` command from the dropdown menu, then click "Run."



TLS

The last step is setting up TLS, often [incorrectly] referred to as SSL. Navigate to the "SSL" section, click "Let's Encrypt," and then "Obtain Certificate." There should be no reason to make modifications to the default values.



⚠ Note

If the domain has not been propagated, the certificate creation may fail. In this case, try again immediately, and if it fails again, try waiting an hour or so. If it continues to fail, double check to make sure that everything is set up correctly on Cloudflare.

That's It!

You now have a project hosted on Laravel Forge!

Revision #1

Created 11 December 2022 10:11:24

Updated 11 December 2022 18:11:29