

# **Production Server Tuning & Deployments**

**Why?**

# Ensure Production Configuration

```
APP_NAME="Laravel Reports"
```

```
APP_ENV=production
```

```
APP_KEY=base64:vr1Fu9KKSh1NB0vbqzsXoXXEyIVfRVc3RwsZxcHuYjk=
```

```
APP_DEBUG=false
```

```
APP_URL=https://laravel.report
```

```
APP_ADMIN_URL=admin
```

```
APP_TIMEZONE=Asia/Dhaka
```

```
APP_LOCALE=en
```

# Use Framework Level Caches

```
php artisan route:cache
```

```
php artisan config:cache
```

```
php artisan view:cache
```

# Optimize Composer Autoloading

```
composer install --optimize-autoloader --no-dev
```

# Optimize Eloquent Queries

<https://github.com/barryvdh/laravel-debugbar>

Messages

Timeline

Exceptions

Views83

Route

Queries7

Models34

Mails

Gate

Session

Request

GET forum

6MB

96.33ms

7.4.3

7 statements were executed

13.25ms

select count(\*) as aggregate from `threads` left join `replies` on `threads`.`id` = `replies`.`replyable\_id` and `replies`.`replyable\_type` = 'threads' group by `threads`.`id`

7.03ms /app/Models/Thread.php:156 laravelio

select `threads`.\*,  
CASE WHEN COALESCE(MAX(replies.created\_at), 0) > threads.created\_at  
THEN COALESCE(MAX(replies.created\_at), 0)  
ELSE threads.created\_at  
END AS latest\_creation  
from `threads` left join `replies` on `threads`.`id` = `replies`.`replyable\_id` and  
`replies`.`replyable\_type` = 'threads' group by `threads`.`id` order by `latest\_creation` desc limit 20  
offset 0

1.42ms /app/Models/Thread.php:156 laravelio

select \* from `replies` where 0 = 1

420µs /app/Models/Thread.php:156 laravelio

select \* from `replies` where `replies`.`replyable\_id` in (4, 5, 6, 7, 12, 13, 14, 15, 20, 21, 22, 28, 29, 30, 36, 37, 38, 44, 45, 46) and `replies`.`replyable\_type` = 'threads'

740µs /app/Models/Thread.php:156 laravelio

select `tags`.\*, `taggables`.`taggable\_id` as `pivot\_taggable\_id`, `taggables`.`tag\_id` as  
`pivot\_tag\_id`, `taggables`.`taggable\_type` as `pivot\_taggable\_type`, `taggables`.`created\_at` as  
at, `taggables`.`updated\_at` as `pivot\_updated\_at` from `tags` inner join `taggables` on

2.06ms /app/Models/Thread.php:156 laravelio



# Optimize Eloquent Queries

<https://github.com/beyondcode/laravel-query-detector>

```
11 175 169 1 10 GET / 5MB 289ms 8.2.16
i Model: App\Models\Course\Course => Relation: Spatie\MediaLibrary\MediaCollections\Models\Media - You... INFO
i Model: App\Models\Course\Course => Relation: teachers - You should add `with(teachers)` to eager-loa... INFO
i Model: App\Models\Course\Course => Relation: App\Models\Payment\Price - You should add `with(App\Mod... INFO
i Model: App\Models\Course\Course => Relation: App\Models\Payment\Price - You should add `with(App\Mod... INFO
i Model: App\Models\Payment\Price => Relation: currency - You should add `with(currency)` to eager-loa... INFO
i Model: App\Models\Payment\Price => Relation: priceable - You should add `with(priceable)` to eager-l... INFO
i Model: App\Models\Course\Course => Relation: Spatie\MediaLibrary\MediaCollections\Models\Media - You... INFO
i Model: App\Models\Course\Course => Relation: Spatie\MediaLibrary\MediaCollections\Models\Media - You... INFO
i Model: App\Models\Course\Course => Relation: Spatie\MediaLibrary\MediaCollections\Models\Media - You... INFO
i Model: App\Models\Course\Course => Relation: Spatie\MediaLibrary\MediaCollections\Models\Media - You should add `with(Spatie\MediaLibrary\MediaCollections\Models\Media)` to eager-load this relation. INFO
```



# Tune MySQL configuration

<https://github.com/major/MySQLTuner-perl>

```
----- Performance Metrics -----
[--] Up for: 1d 9h 9m 8s (27K q [0.228 qps], 9K conn, TX: 3M, RX: 4M)
[--] Reads / Writes: 100% / 0%
[--] Binary logging is disabled
[--] Total buffers: 169.0M global + 1.1M per thread (151 max threads)
[OK] Maximum reached memory usage: 170.1M (17.13% of installed RAM)
[OK] Maximum possible memory usage: 338.9M (34.11% of installed RAM)
[OK] Slow queries: 0% (0/27K)
[OK] Highest usage of available connections: 0% (1/151)
[OK] Aborted connections: 0.24% (22/9085)
[!!] Query cache is disabled
[OK] Sorts requiring temporary tables: 0% (0 temp sorts / 60 sorts)
[OK] Temporary tables created on disk: 5% (1K on disk / 24K total)
[OK] Thread cache hit rate: 99% (1 created / 9K connections)
[OK] Table cache hit rate: 29% (2K open / 6K opened)
[OK] Open file limit used: 2% (139/5K)
[OK] Table locks acquired immediately: 100% (9K immediate / 9K locks)
```

```
----- MyISAM Metrics -----
[!!] Key buffer used: 18.3% (1M used / 8M cache)
[OK] Key buffer size / total MyISAM indexes: 8.0M/41.0K
[OK] Read Key buffer hit rate: 99.7% (2K cached / 7 reads)
```

```
----- InnoDB Metrics -----
```



# Configure PHP-FPM

```
[www]
user = www-data
group = www-data
listen = /run/php/php8.1-fpm.sock
pm = dynamic
pm.max_children = 50
pm.start_servers = 5
pm.min_spare_servers = 5
pm.max_spare_servers = 35
pm.max_requests = 500
```

`pm = dynamic`: Adjusts number of child processes based on load

`pm.max_children = 50`: Never exceed 50 PHP processes

`pm.start_servers = 5`: Start with 5 processes when PHP-FPM launches

`pm.min_spare_servers = 5` and `pm.max_spare_servers = 35`: Keep between 5 and 35 idle processes

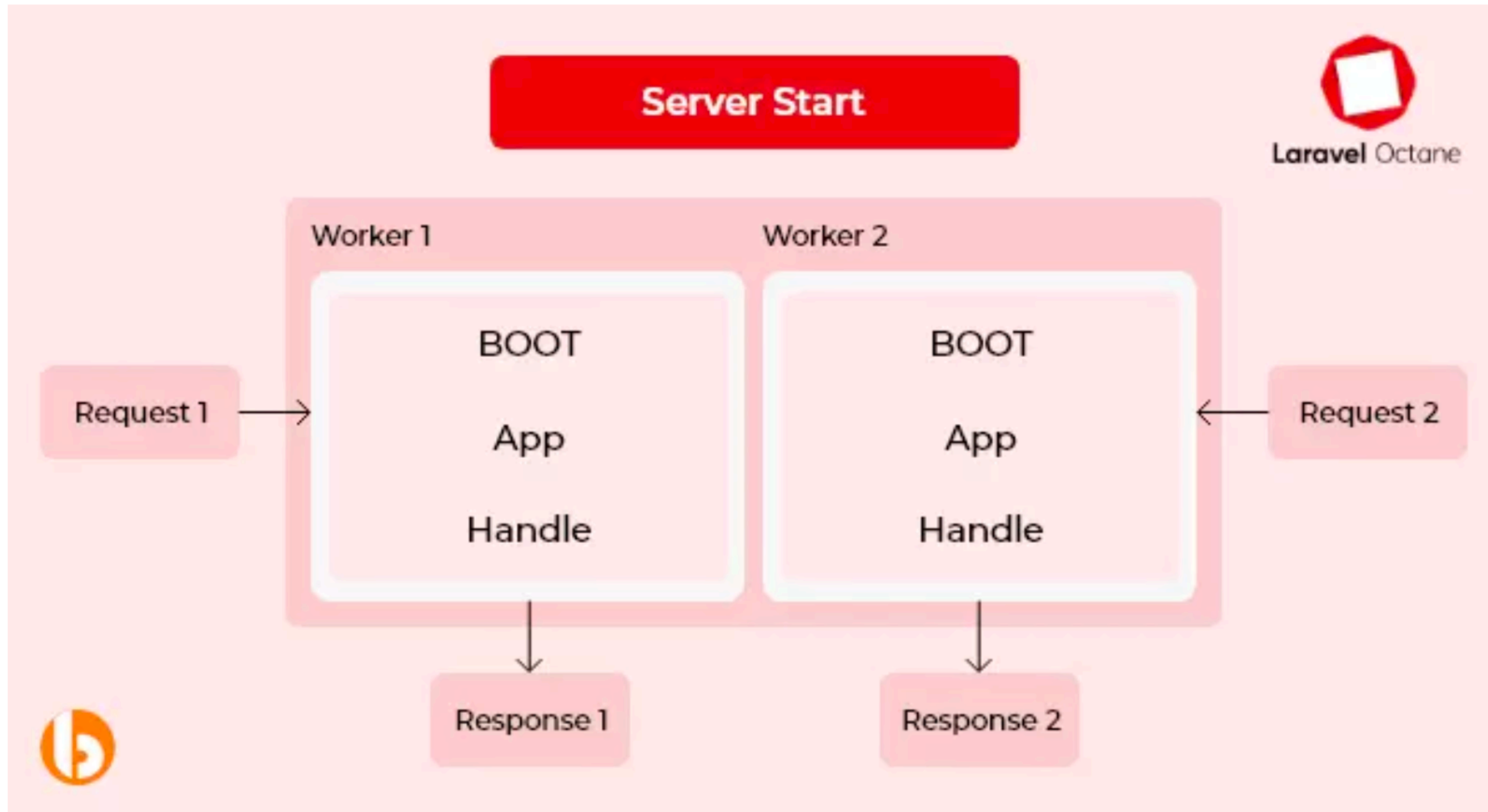
# Use a modern webserver



**Caddy**<sup>®</sup>  
THE ULTIMATE SERVER



# Switch to Laravel Octane



# Basic Deployment using Git

```
#!/bin/bash

# Navigate to the project directory
cd /path/to/your/laravel/project

# Pull the latest changes
git pull origin main

# Install dependencies
composer install --no-interaction --prefer-dist --optimize-autoloader

# Run migrations
php artisan migrate --force

# Clear caches
php artisan optimize:clear

# Restart services
sudo systemctl restart php8.1-fpm
sudo systemctl reload nginx

echo "Deployment completed successfully!"
```



# Serious Deployment using Deployer

<https://deployer.org/>

Deploy **PHP**  
everywhere



The PHP deployment tool with support for [popular frameworks](#) out of the box.

Get Started

## Why Deployer?

### Provision

Deployer will configure your server, setup a **firewall**, install **PHP** and **MySQL** or **Postgres**, configure **HTTPS**, etc.

### Zero downtime

Zero downtime deployments. Did something go wrong? **Rollback** to the previous release.

### Recipes

Ready to use recipes for **most frameworks** and **applications**.

```
1 <?php declare(strict_types=1);
2
3 namespace Deployer;
4
5 use Deployer\Utility\Httppie;
6
7 require 'recipe/laravel.php'; /** phpcs:disable PSR1.Files.SideEffects */
8 |
9 /**
10  * We should close repo deep enough to get info about the latest available tag.
11  * It's hacky solution, but if you know better – please update the related code.
12  * @psalm-internal
13  */
14 define('GIT_CLONE_DEPTH_ENOUGH_FOR_TAGS', 300);
15
16 /**
17  * @see https://deployer.org/docs
18  *
19  * How to start deployment examples:
20  * ./deployer.phar deploy dev-01
21  * ./deployer.phar deploy dev-02
22  * ./deployer.phar deploy dev-03
23  * ./deployer.phar deploy develop
24  * ./deployer.phar deploy production
25  * ▲ 'deploy' task will ship your CURRENT git branch
26  *
27  * Rollback latest deployment:
28  * ./deployer.phar rollback production
29  *
30  * Run a single (any) task on a host:
31  * ./deployer.phar {taskname} [{hostname}='localhost']
32  * Examples:
33  * ./deployer.phar hello
34  * ./deployer.phar hello production
35  * ./deployer.phar phpmetrics develop
36  * ./deployer.phar deploy dev-01
37  * ./deployer.phar artisan:queue:restart production
38  * ./deployer.phar laravel:config-changed production
39 */
40
41 // Project name
42 set('application',          );
43
44 // Project repository
45 set('repository', 'git@github.com:  .git');
46
47 // Number of releases to keep
48 set('keep_releases', 2);
49
50 // No need to use sudo to make writable directories
51 set('writable_use_sudo', false);
52 set('http_user', 'forge');
53
54 set('default_timeout', 1000);
55 set('ssh_multiplexing', true);
56
```