常见问题

目录
Composer 在 Automattic\Jetpack\Autoloader\AutoloadGenerator 上的错误 VVV: HostsUpdater vagrant 插件错误
VVV: install-wp-tests.sh 错误
运行端到端测试时出现超时 /404 错误
Docker 容器构建失败，导致端到端测试无法执行
设置 WooCommerce Payments 开发模式
获取 WooCommerce 管理员安装时间戳
重置引导向导
启用控制台中的跟踪调试
使用 Vagrant (VVV) 运行 PHP 单元测试

本页面旨在记录已知的常见问题、遇到的问题以及它们的解决方案或解决方法。
如果您遇到了此处未提及但应该记录的问题，请随时将其添加到列表中。

Composer 在

Automattic\Jetpack\Autoloader\Autol oadGenerator 上的错误

[ErrorException]
Declaration of Automattic\Jetpack\Autoloader\AutoloadGenerator::dump(Composer\Config
$config, Composer\Repository\Inst
alledRepositoryInterface $localRepo, Composer\Package\PackageInterface $mainPackage, Composer\Installer\InstallationMa
nager $installationManager, $targetDir, $scanPsrPackages = false, $suffix = NULL) should be compatible with Composer\A
utoload\AutoloadGenerator::dump(Composer\Config $config, Composer\Repository\InstalledRepositoryInterface $localRepo,

```
  Composer\Package\RootPackageInterface $rootPackage, Composer\Installer\InstallationManager
$installationManager, $targ
  etDir, $scanPsrPackages = false, $suffix = '')
```

Composer 2.0.7 □□□ □□ □□□□□□□□□□□□□ `AutoloadGenerator` □□□□□□ □□□□□
composer 2.0.6□□□ `composer self-update 2.0.6` □□□

# VVV: HostsUpdater vagrant □□□□

```
...vagrant-hostsupdater/HostsUpdater.rb:126:in ``digest': no implicit conversion of nil into
String (TypeError)
```

□□□□□□□□□□□□ Vagrant □□□ □□□□□□ VVV □ Vagrant 2.2.7 □□□ □□□ VVV □ □□ □

# VVV: `install-wp-tests.sh` □□

```
mysqladmin: CREATE DATABASE failed; error: 'Access denied for user 'wp'@'localhost' to
database 'wordpress-one-tests''
```

□□□□□

- □□ `sudo mysql` □□ MySQL□
- □□ `GRANT ALL PRIVILEGES ON * . * TO 'wp'@'localhost';` □ □□□□ `exit;` □□□
- □□□□ `install-wp-tests.sh` □□□

# □□□□□□□□□□□ /404 □□

```
 Store owner can complete onboarding wizard › can complete the product types section

    TimeoutError: waiting for function failed: timeout 30000ms exceeded

      1 | export const waitForElementCount = function ( page, domSelector, count ) {
```

```
    > 2 | □return page.waitForFunction(
        | □           ^
      3 | □□( domSelector, count ) => {
      4 | □□□return document.querySelectorAll( domSelector ).length === count;
      5 | □□},
```

端到端测试中的超时或 404 错误表明现有的构建可能已损坏。 运行 `npm install && npm run clean && npm run build` 以生成一个全新的构建。 值得注意的是，我们的某些 npm 脚本也会删除当前的构建，因此，在运行端到端测试之前，始终构建是一个好习惯。

# Docker

## 容器构建失败，导致端到端测试无法执行

```
Thu Dec  3 11:55:56 +08 2020 - Docker 容器正在构建中
Thu Dec  3 11:56:06 +08 2020 - Docker 容器正在构建中
Thu Dec  3 11:56:16 +08 2020 - Docker 容器正在构建中
Thu Dec  3 11:56:26 +08 2020 - Docker 容器构建失败
npm ERR! code ELIFECYCLE
npm ERR! errno 1
npm ERR! @woocommerce/e2e-environment@0.1.6 test:e2e: `bash ./bin/wait-for-build.sh &&
./bin/e2e-test-integration.js`
npm ERR! Exit status 1
```

请确保 Docker 正在运行。虽然脚本显示 “Docker 容器正在构建中”，但它实际上并不负责运行 Docker；它只是等待现有的 Docker 实例做出响应。如果未运行，请运行 `npm run docker:up`。

# 设置 WooCommerce Payments 开发模式

将以下内容添加到 `wp-config.php`：

```
define( 'WCPAY_DEV_MODE', true );
```

另请参阅 此文档____ 。

# 获取 WooCommerce

## 管理员安装时间戳

要获取安装时间戳（例如，在 `NoteTraits` 中的 `wc_admin_active_for()` 函数中），请尝试以下 SQL 语句：

```
SELECT * FROM wp_options WHERE option_name = 'woocommerce_admin_install_timestamp'
```

## 重置引导向导

删除 `woocommerce_onboarding_profile` 选项：

```
DELETE FROM wp_options WHERE option_name = 'woocommerce_onboarding_profile'
```

## 启用控制台中的跟踪调试

```
localStorage.setItem( 'debug', 'wc-admin:tracks' );
```

并将 Chrome 的日志级别设置为 “详细”（verbose），并选中 “已选中”状态。

# 使用 Vagrant (VVV) 运行 PHP

## 单元测试

1. 通过 SSH 连接到 Vagrant 虚拟机 (`vagrant ssh`)
2. `cd /srv/www/<WP_INSTANCE>/public_html/wp-content/plugins/woocommerce-admin`
3. 执行设置: `bin/install-wp-tests.sh wc-admin-tests root root`
4. 快速测试: `./vendor/bin/phpunit --group fast`
5. 所有测试: `./vendor/bin/phpunit`

如果 `phpunit` 不存在，您可能需要运行 `composer install`。

□□□ #2
□ □□□ □□□ 2026-03-09 11:53:58 UTC
□ □□□ □□□ 2026-03-09 11:53:58 UTC