

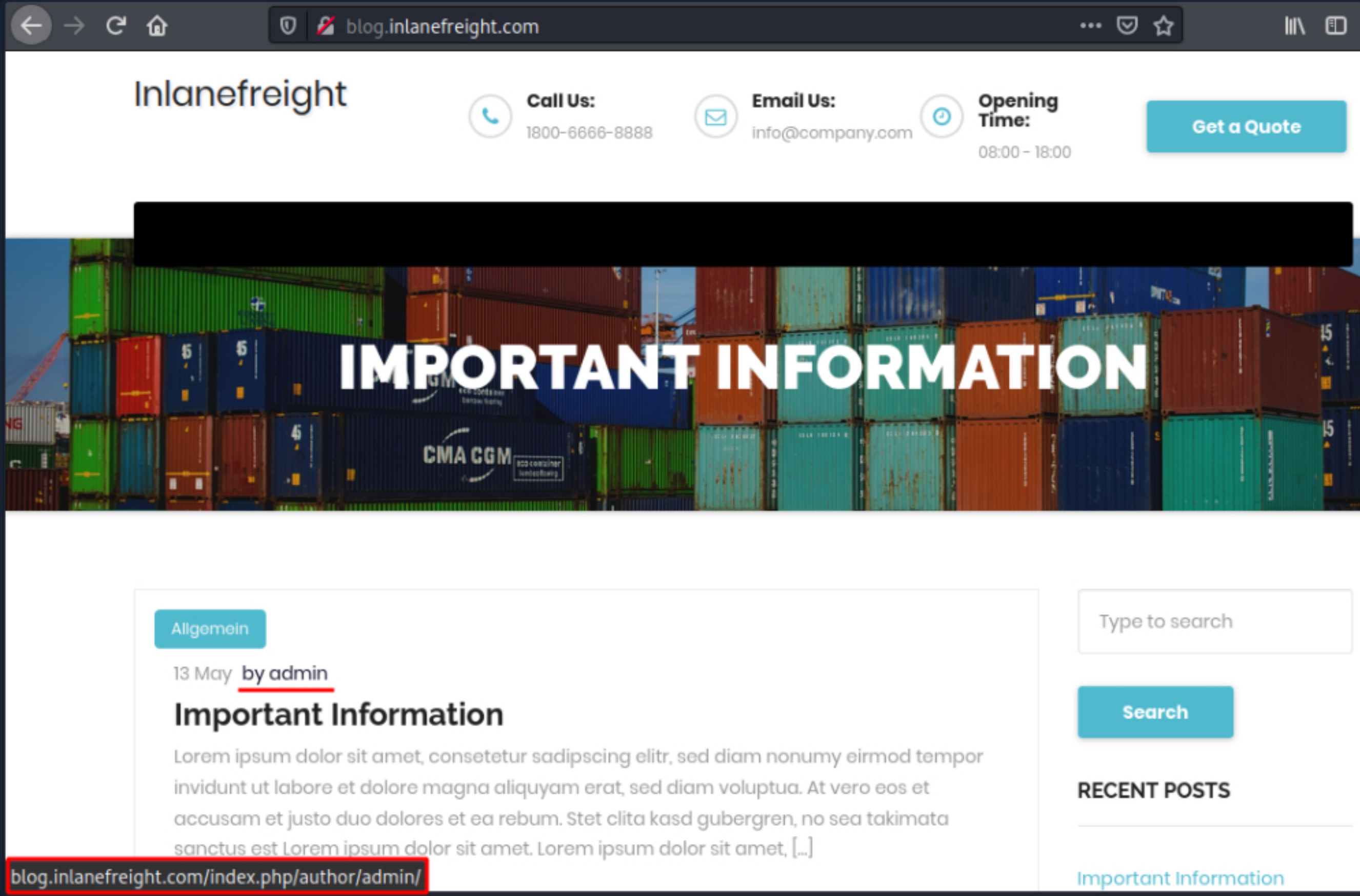
# User Enumeration

Enumerating a list of valid users is a critical phase of a WordPress security assessment. Armed with this list, we may be able to guess default credentials or perform a brute force password attack. If successful, we may be able to log in to the WordPress backend as an author or even as an administrator. This access can potentially be leveraged to modify the WordPress website or even interact with the underlying web server.

There are two methods for performing manual username enumeration.

## First Method

The first method is reviewing posts to uncover the ID assigned to the user and their corresponding username. If we mouse over the post author link titled "by admin," as shown in the below image, a link to the user's account appears in the web browser's lower-left corner.



The **admin** user is usually assigned the user ID **1**. We can confirm this by specifying the user ID for the **author** parameter in the URL.

http://blog.inlanefreight.com/?author=1

This can also be done with **cURL** from the command line. The HTTP response in the below output shows the author that corresponds to the user ID. The URL in the **Location** header confirms that this user ID belongs to the **admin** user.

### Existing User

```
Govardhan Gujji22@htb[/htb]$ curl -s -I -X GET http://blog.inlanefreight.com/?author=1

HTTP/1.1 301 Moved Permanently
Date: Wed, 13 May 2020 20:47:08 GMT
Server: Apache/2.4.29 (Ubuntu)
X-Redirect-By: WordPress
Location: http://blog.inlanefreight.com/index.php/author/admin/
Content-Length: 0
Content-Type: text/html; charset=UTF-8
```

The above **cURL** request then redirects us to the user's profile page or the main login page. If the user does not exist, we receive a **404 Not Found error**.

### Non-Existing User

```
Govardhan Gujji22@htb[/htb]$ curl -s -I -X GET http://blog.inlanefreight.com/?author=100

HTTP/1.1 404 Not Found
Date: Wed, 13 May 2020 20:47:14 GMT
Server: Apache/2.4.29 (Ubuntu)
Expires: Wed, 11 Jan 1984 05:00:00 GMT
Cache-Control: no-cache, must-revalidate, max-age=0
Link: <http://blog.inlanefreight.com/index.php/wp-json/>; rel="https://api.w.org/"
Transfer-Encoding: chunked
Content-Type: text/html; charset=UTF-8
```

## Second Method

The second method requires interaction with the **JSON** endpoint, which allows us to obtain a list of users. This was changed in WordPress core after version 4.7.1, and later versions will only show a user if configured or not. Before this change, all users who had published a post were shown by default.

### JSON Endpoint

```
Govardhan Gujji22@htb[/htb]$ curl http://blog.inlanefreight.com/wp-json/wp/v2/users | jq

[
  {
    "id": 1,
    "name": "admin",
    "url": "",
    "description": "",
    "link": "http://blog.inlanefreight.com/index.php/author/admin/",
    <SNIP>
  },
  {
    "id": 2,
    "name": "ch4p",
    "url": "",
    "description": "",
    "link": "http://blog.inlanefreight.com/index.php/author/ch4p/",
    <SNIP>
  },
  <SNIP>
]
```

### Questions

Answer the question(s) below to complete this Section and earn cubes!

+ 1 From the last cURL command, what user name is assigned to User ID 2?

Submit your answer here...

Submit

Previous

Next

Cheat Sheet

Go to Questions

### Table of Contents

#### Introduction

Intro

WordPress Structure

WordPress User Roles

#### Enumeration

WordPress Core Version Enumeration

Plugins and Themes Enumeration

Directory Indexing

User Enumeration

Login

WPScan Overview

WPScan Enumeration

#### Exploitation

Exploiting a Vulnerable Plugin

Attacking WordPress Users

RCE via the Theme Editor

Attacking WordPress with Metasploit

#### Security Measures

WordPress Hardening

#### Skills Assessment

Skills Assessment - WordPress

### My Workstation

OFFLINE

Start Instance

1 spawns left

