Lab 13: HTML smuggling

Table of Contents

Goals

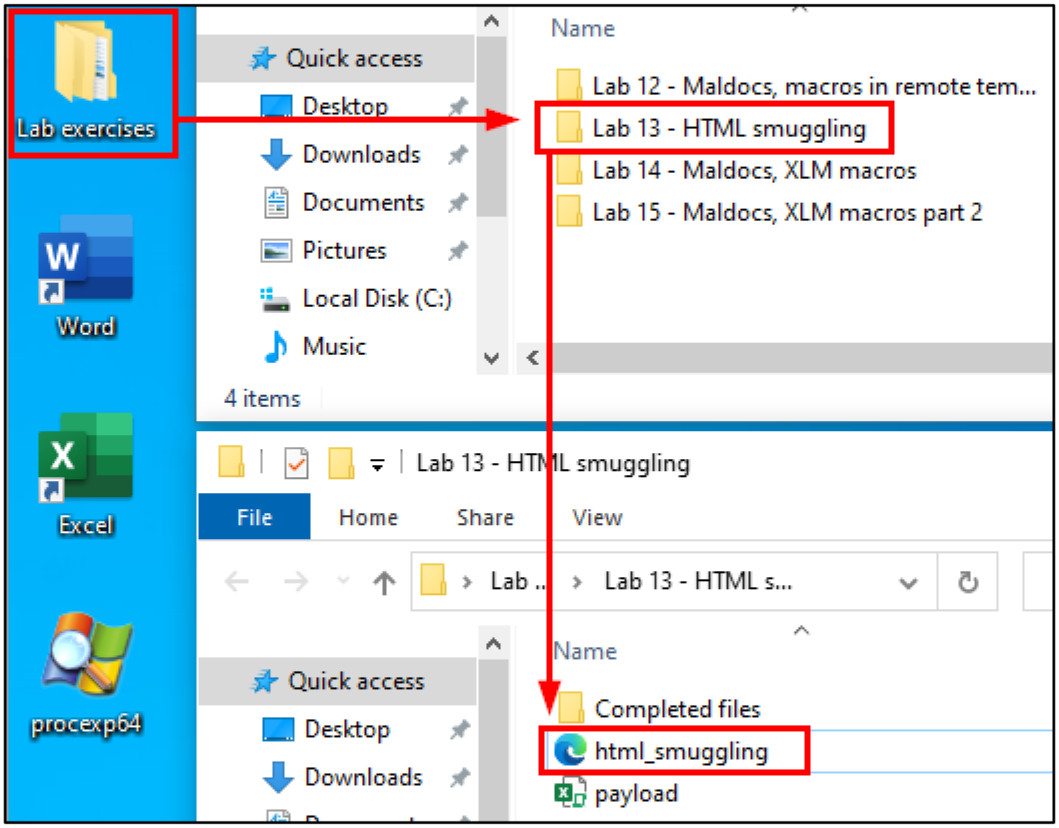
- Embed a payload file inside of an HTML document.

Requirements

- Windows 10 VM

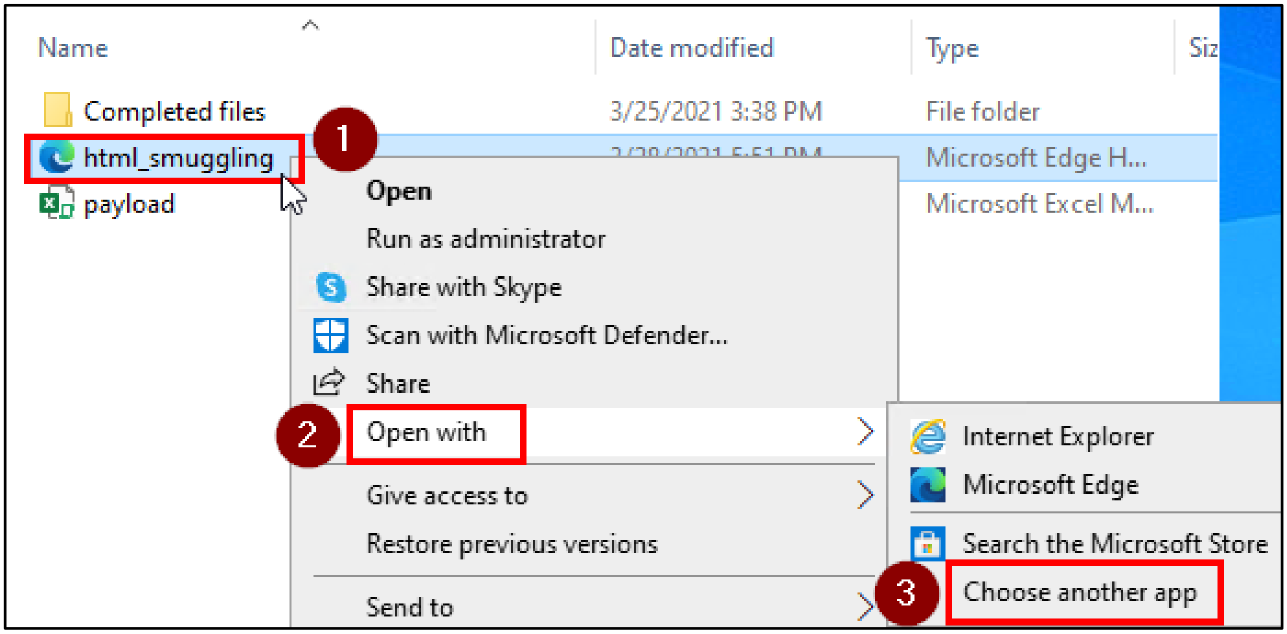
Instructions

1. In your Windows 10 VM, open the Lab 13 subfolder inside the Lab Exercises folder on your desktop. You should see the "html_smuggling.html" file inside.



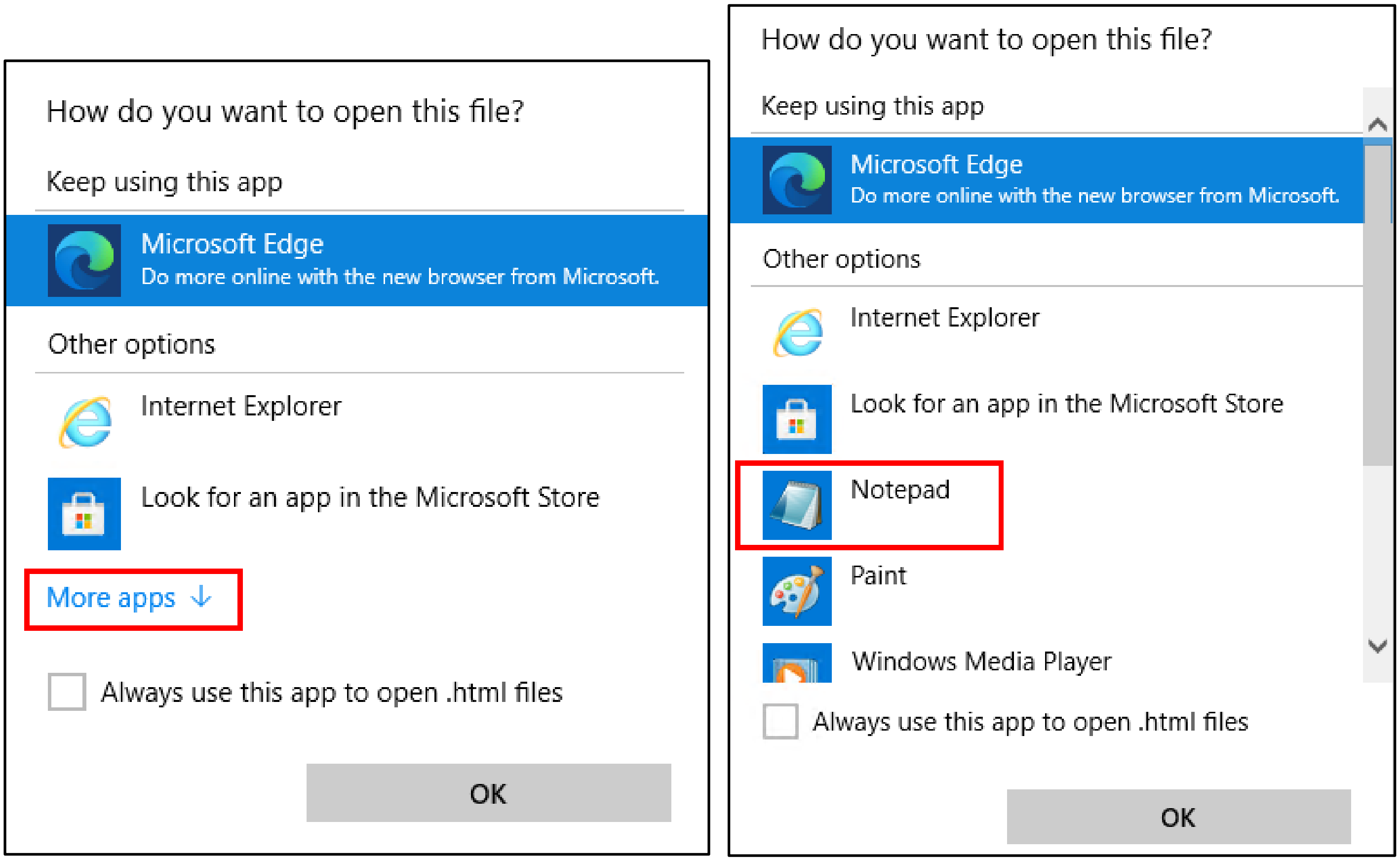
Location of "html_smuggling.html"

2. Right click on "html_smuggling.html", and in the menu that appears, click "Open with" and then "Choose another app".



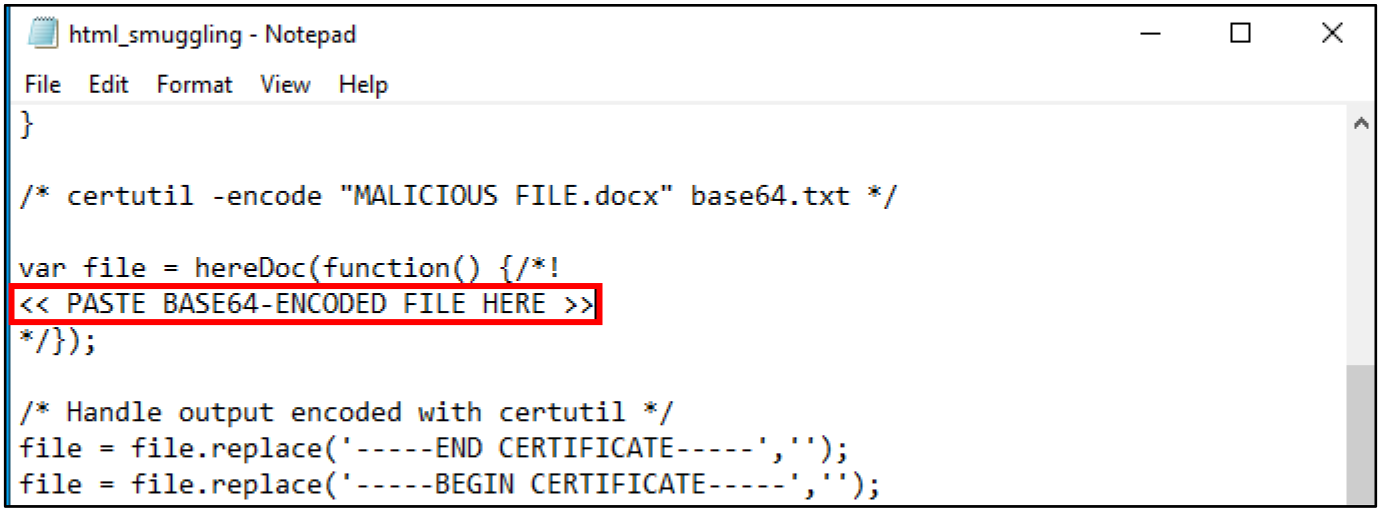
Open with Another App

3. Click "More apps" near the bottom of the window that appears. Then find "Notepad" in the list and double-click on it.


How do you want to open this file?
Keep using this app
Microsoft Edge
Do more online with the new browser from Microsoft.
Other options
Internet Explorer
Look for an app in the Microsoft Store
More apps ↓
□ Always use this app to open .html files
OK
How do you want to open this file?
Keep using this app
Microsoft Edge
Do more online with the new browser from Microsoft.
Other options
Internet Explorer
Look for an app in the Microsoft Store
Notepad
Paint
Windows Media Player
□ Always use this app to open .html files
OK

Open with Notepad

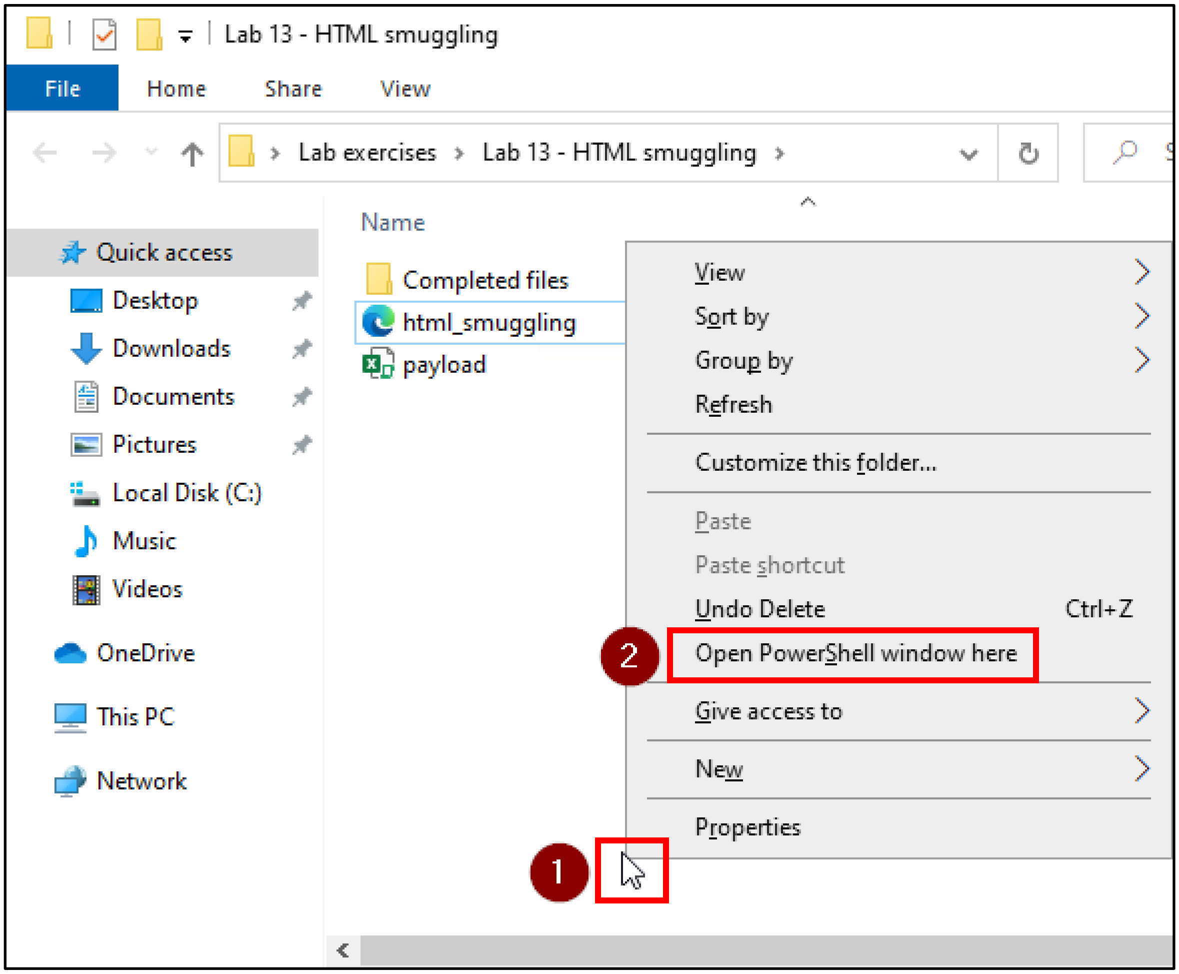
4. In the Notepad window, find the line in the file that says "<< PASTE BASE64-ENCODED FILE HERE >>". Your Base64-encoded payload file will replace this line in the next steps.



```
}

/* certutil -encode "MALICIOUS FILE.docx" base64.txt */

var file = hereDoc(function() {/*!
<< PASTE BASE64-ENCODED FILE HERE >>
*/});

/* Handle output encoded with certutil */
file = file.replace('-----END CERTIFICATE-----','');
file = file.replace('-----BEGIN CERTIFICATE-----','');
```

Line to be Replaced

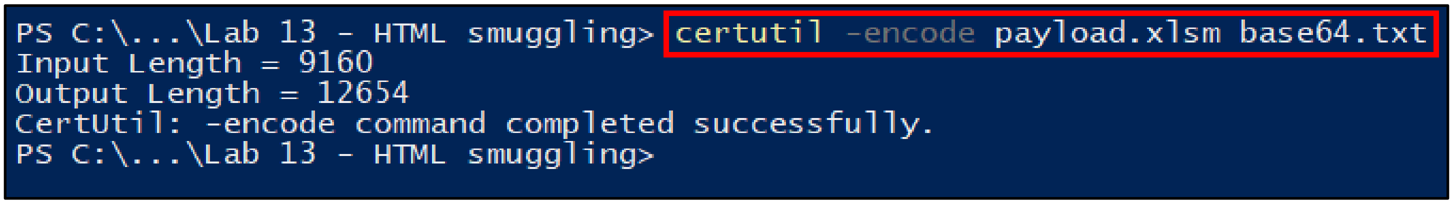
5. To generate the Base64-encoded file, first open a PowerShell window by holding the Shift key on your keyboard and simultaneously right-clicking the white background of the "Lab 13 - HTML smuggling" folder. Then choose "Open PowerShell window here" from the menu.



Open PowerShell Here

6. In the PowerShell window, run the command below to generate a Base64-encoded version of the "payload.xlsm" file.

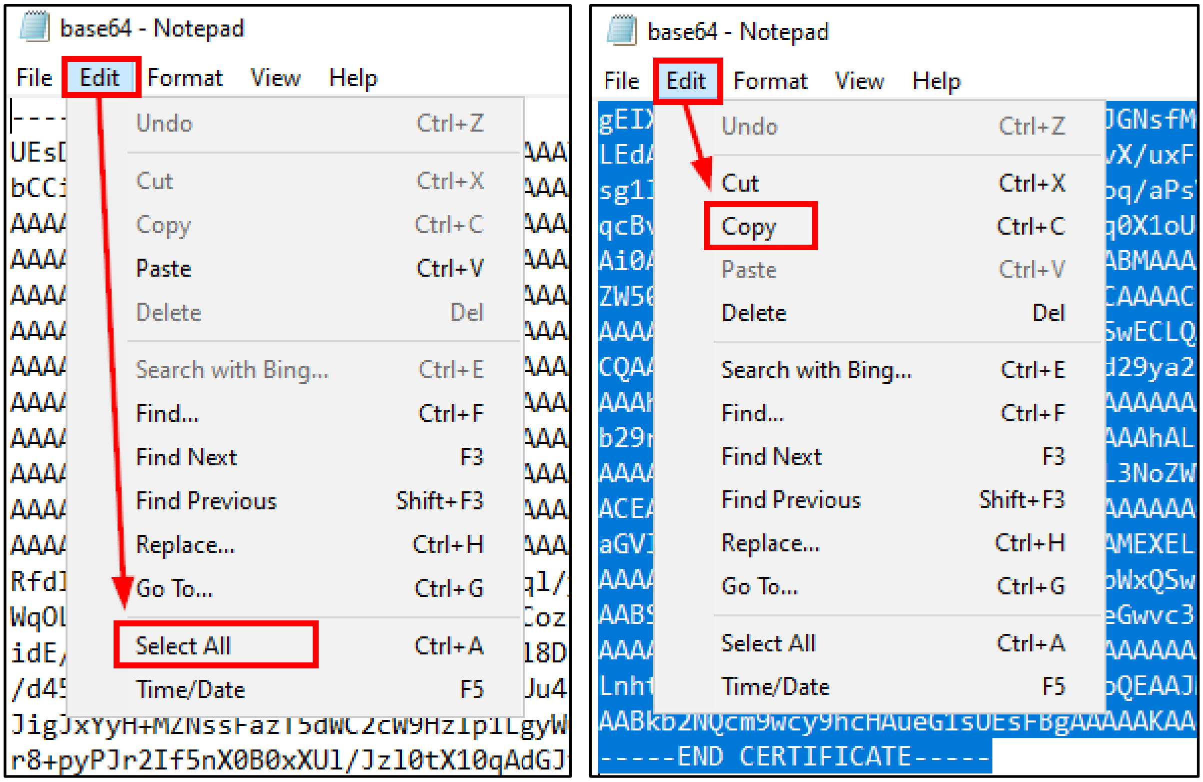
```
certutil -encode payload.xlsm base64.txt
```



```
PS C:\...\Lab 13 - HTML smuggling> certutil -encode payload.xlsm base64.txt
Input Length = 9160
Output Length = 12654
CertUtil: -encode command completed successfully.
PS C:\...\Lab 13 - HTML smuggling>
```

Base64 Encode "payload.xlsm"

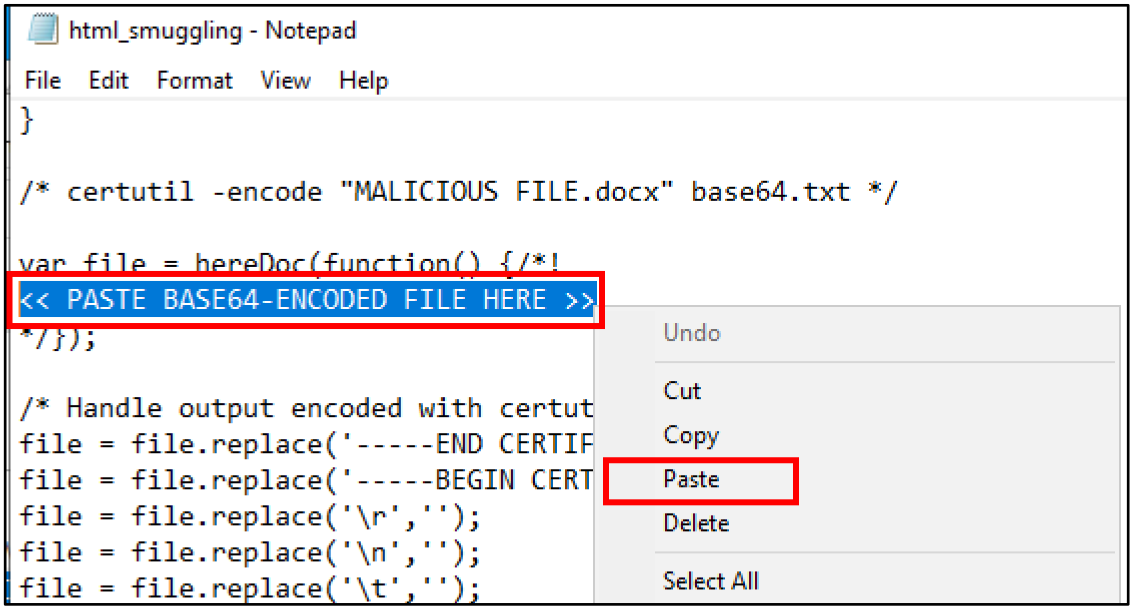
7. If you look in the Lab 13 folder, you should now see the file "base64.txt" listed there. Double-click on the Base64.txt file to open it in Notepad. Then select and copy all of the text in the file.



Copy All Contents in Base64.txt

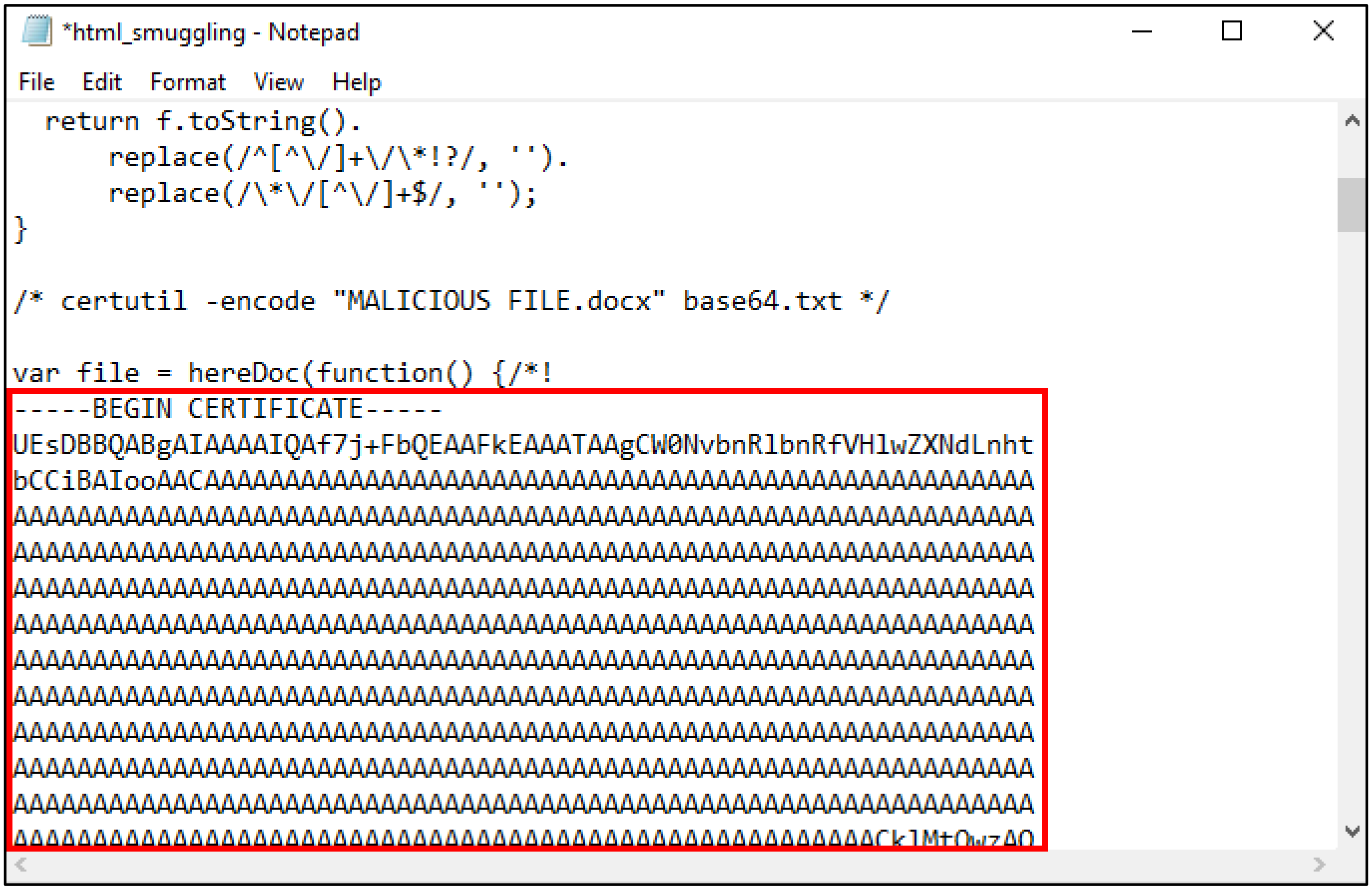
8. Now replace the "<< PASTE BASE64-ENCODED FILE HERE >>" line in the "html_smuggling.html" with the Base64-encoded data that you just copied.

NOTE: Normally you would want to remove the "BEGIN CERTIFICATE" and "END CERTIFICATE" lines from the Base64 output, as well as removing all the line breaks. However, to keep this exercise user-friendly, the "html_smuggling.html" template file has been modified to account for the difference. Just be aware of this difference when dealing with other tools that accept Base64 data on your own in the real world.



```
}

/* certutil -encode "MALICIOUS FILE.docx" base64.txt */

var file = hereDoc(function() {/*!
<< PASTE BASE64-ENCODED FILE HERE >>
*/});

/* Handle output encoded with certut
file = file.replace('-----END CERTIF
file = file.replace('-----BEGIN CERT
file = file.replace('\r','');
file = file.replace('\n','');
file = file.replace('\t','');
```

Paste the Base64 Data Here



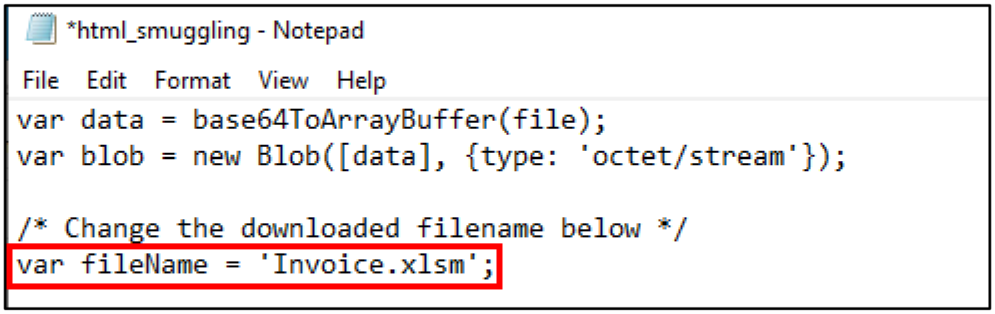
```
  return f.toString().
      replace(/^[^\/]+\/\*!?/, '').
      replace(/\*\/[^\/]+$/, '');
}

/* certutil -encode "MALICIOUS FILE.docx" base64.txt */

var file = hereDoc(function() {/*!
-----BEGIN CERTIFICATE-----
```

```
UEsDBBQABgAIAAAAIQAf7j+FbQEAAFkEAAATAAgCW0NvbnRlbnRfVHlwZXNdLnht
bCCiBAIooAACAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAA
AAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAA
AAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAA
AAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAA
AAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAA
AAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAA
AAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAA
AAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAA
AAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAA
AAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAA
AAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAACklMtQwzAQ
```

"html_smuggling.html" After Adding Base64-Encoded Data

9. After pasting in the Base64 data, scroll to the bottom of the "html_smuggling.html" file, and find the line that says "/* Change the downloaded filename below */". Then change the value of the "fileName" variable under this line to a filename with a ".xlsm" extension. (This is the filename that your payload file will be saved as when it is downloaded by the target, so it should have the same extension as the original payload file.) The filename I'm using for this example is "Invoice.xlsm".

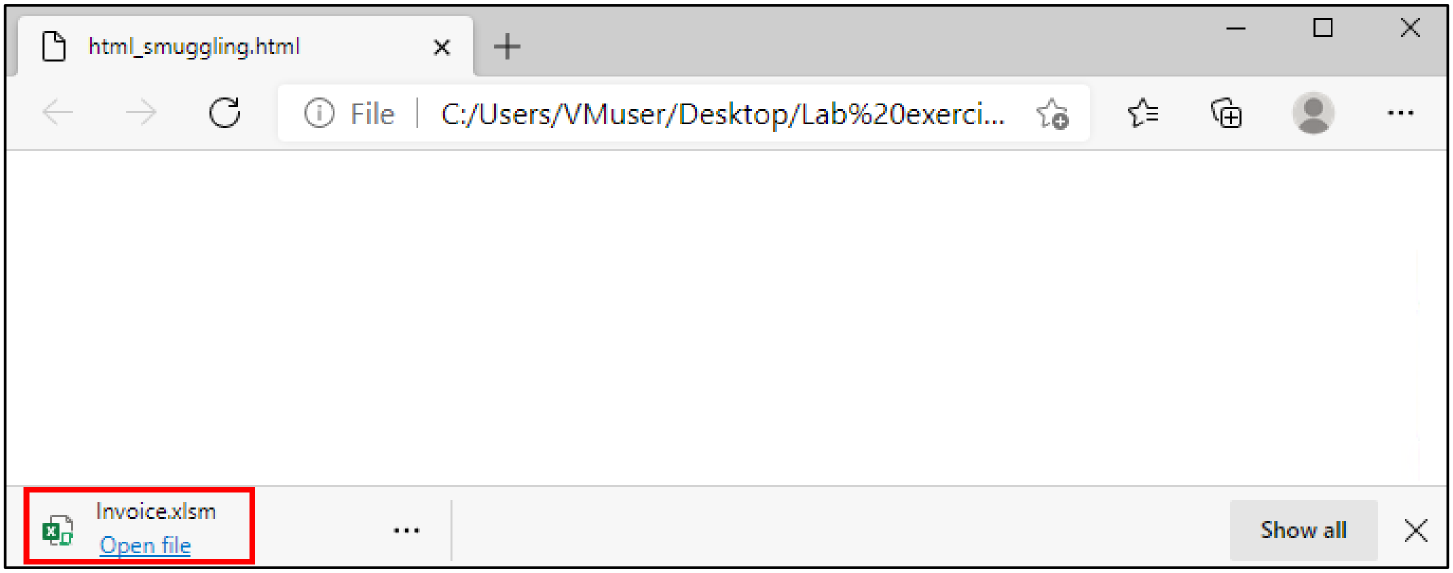
```
Invoice.xlsm
```



```
var data = base64ToArrayBuffer(file);
var blob = new Blob([data], {type: 'octet/stream'});

/* Change the downloaded filename below */
var fileName = 'Invoice.xlsm';
```

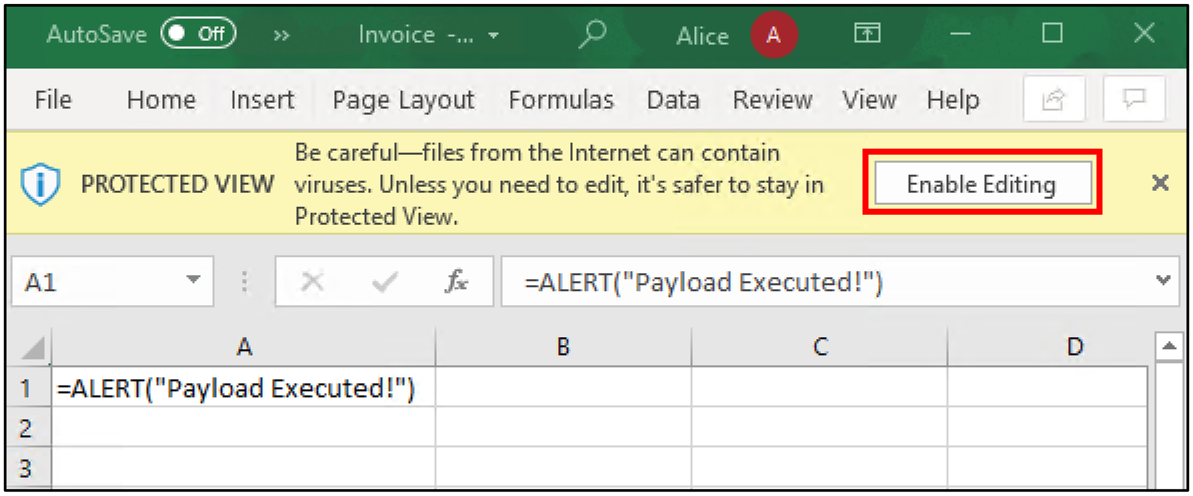
Totally Legit-Looking Filename Goes Here

10. Save your changes to the "html_smuggling.html" file when they are complete, and close the open Notepad windows.
11. Now double-click on the "html_smuggling.html" file to open the file in a web browser and observe how the embedded payload is automatically downloaded when the HTML file is opened.



Invoice.xlsm Automatically Downloaded When "html_smuggling.html" is Opened

12. Open the downloaded file to confirm that it was correctly decoded from the Base64 data you created and that it executes successfully.

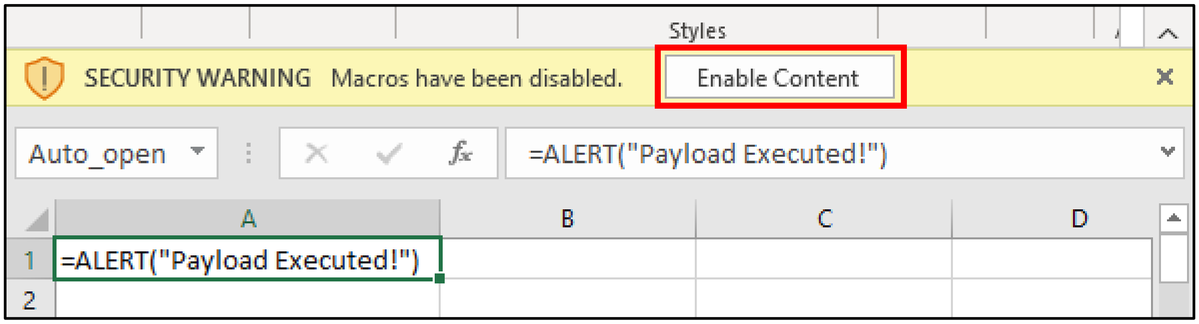


PROTECTED VIEW Be careful—files from the Internet can contain viruses. Unless you need to edit, it's safer to stay in Protected View.

A1 =ALERT("Payload Executed!")

	A	B	C	D
1	=ALERT("Payload Executed!")			
2				
3				

"Enable Editing" (Due to Mark of the Web)

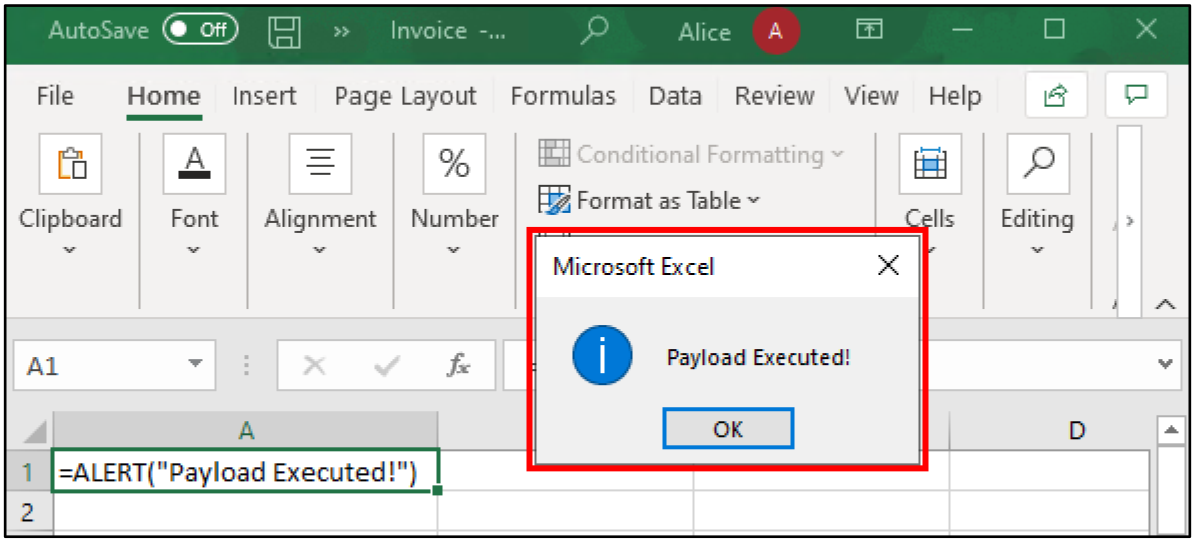


SECURITY WARNING Macros have been disabled.

Auto_open =ALERT("Payload Executed!")

	A	B	C	D
1	=ALERT("Payload Executed!")			
2				

"Enable Content" to Run Macros



Microsoft Excel

Payload Executed!

A1

	A	D
1	=ALERT("Payload Executed!")	
2		

Successful Payload Execution