Security Essentials Achievement

CTF Problems

Cookies

```
File  Edit  View  Search  Terminal  Help
root@mightythor:~# curl GET -v --cookie "amount=7" http://shell.fullstackacademy
.com:21081/
```

I used ‘curl’ to access the site and GET a specified cookie. I used the -v option to ask for verbose information. I specified cookie and saw that I needed to give “name=value”. To adhere to the directions in the challenge problem I typed in “amount=7” followed by the url.

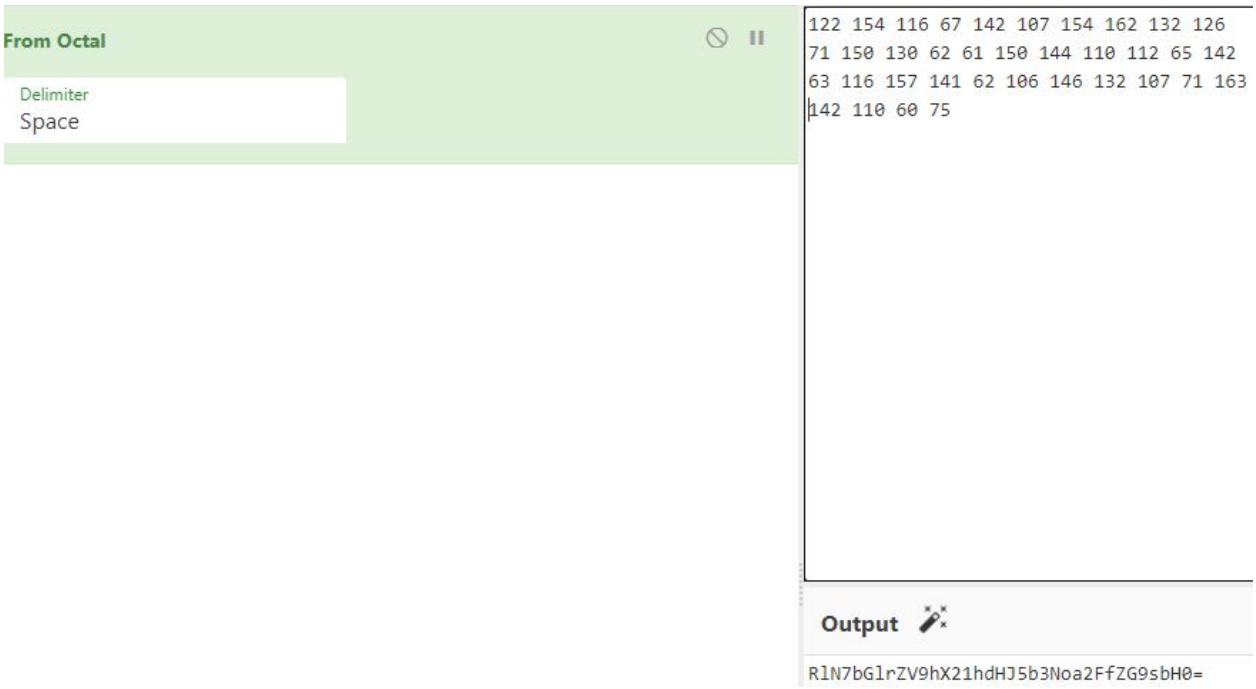
```
      <footer class="footer">
          <p>&copy; Company 2017</p>
          <p>Flag: RlN7SV93aWxsX3Rha2VfdHdvfQ==</p>
      </footer>

    </div> <!-- /container -->
* Connection #1 to host shell.fullstackacademy.com left intact
</body>root@mightythor:~#
```

After sending I noticed this line of text next to Flag: that ended in two = signs. This made me think of Base64 encoding so I ran the command below to decode.

```
root@mightythor:~# echo 'RlN7SV93aWxsX3Rha2VfdHdvfQ==' | base64 -d
FS{I_will_take_two}root@mightythor:~#
```

Finally, giving the flag: FS{I_will_take_two}

Encoding Round 2



I recognized the text in the problem as being in octal format. I was looking for a way to convert octal format into plain text in Linux but the closest I found was “od -x”, but couldn't get that to work. I remembered Cyber Chef had a “From Octal” recipe and chose to use that. This output ending in an = sign made me think of Base64.

```
root@mightythor:~# echo 'RlN7bGlrZV9hX21hdHJ5b3Noa2FfZG9sbH0=' | base64 -d
FS{like_a_matryoshka_doll}root@mightythor:~#
```

Echoing that string into ‘base64 -d’ to decode gives us the Flag:
FS{like_a_matryoshka_doll}

GET Requests

I tried a couple things, but I'll explain what worked first. The solution that worked was typing the request into the url

http://shell.fullstackacademy.com:61751/?usertype=admin&school=fullstack%20cyber%20bootcamp&access-level=super%20duper%20top%20secret

I initially typed the spaces in, but it converted it with the %20. When I googled GET with parameters I found this solution of adding ?param1=value1&param2=value2 etc onto the end of the url

Flag:

```
RlN7cmVhbGx5X3Nob3VsZF9ub3RfdXNlX3F1ZXJ5X3BhcmFtc190b19hdXRoZW50aWNhdGV9
```

The site gave me this, which I recognized as Base64. I echoed this into base64 -d in terminal to get Flag: FS{really_should_not_use_query_params_to_authenticate}

The method that I could not get to work (which I assume is the intended way) was:

curl -X GET
-H "Content-type: application/json"
-H "Accept: application/json"
-d '{usertype:admin,school:"fullstack cyber bootcamp",access-level:"super duper top secret"}'
"http://shell.fullstackacademy.com:61751"

I remember having a similar problem in the workshop with JSON requests but there is something wrong in my syntax. Maybe it's how I'm doing my spacing? I tried using "" around my parameters but that didn't work either.

Hashing

root > Python > dict_labs > apache.py > ...

Set as interpreter

```python
#!/usr/bin/env python3
import sys
import hashlib

def hasher(file):
    #Open our file
    f = open(file)

    #Store the read contents of the file in a variable
    f_read = f.read()

    #Turn those contents into bytes
    f_bytes = f_read.encode()

    #Get the MD5, SHA1, SHA256, and SHA512 hashes of the contents
    md5_hash = hashlib.md5(f_bytes).hexdigest()
    sha1_hash = hashlib.sha1(f_bytes).hexdigest()
    sha256_hash = hashlib.sha256(f_bytes).hexdigest()
    sha512_hash = hashlib.sha512(f_bytes).hexdigest()

    #Print the first index of each, in order, concatenated together
    print (str(md5_hash[0]) + str(sha1_hash[0]) +
    str(sha256_hash[0]) + str(sha512_hash[0]))

def main():
    file = sys.argv[1]
    hasher(file)
main()
```

Seeing that we needed four different types of hashes this made me think of the hashing import from some of the python challenges. I made a script that got all four hashes of the contents then concatenated the first index of each hash in the order requested in the problem

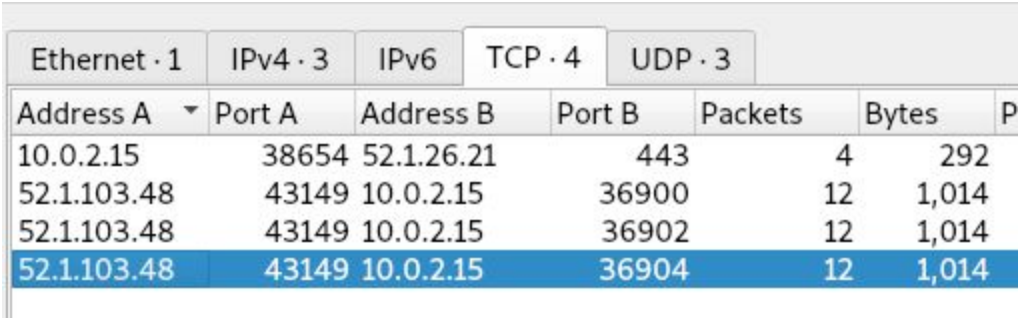
```
root@mightythor:~/Python/dict_labs# ./apache.py hash_this.txt
f5ce
```

Running my script with the contents of the file gives the Flag: FS{f5ce}

Logging

```
Access_code is following_the_green_machine
.....................................................Access Code:
1
.....................................................................................Welcome. Please choose from the
follow menu:
[1]: Estimate Mission Success
[2]: Display Current Time
[3]: Get Flag
```

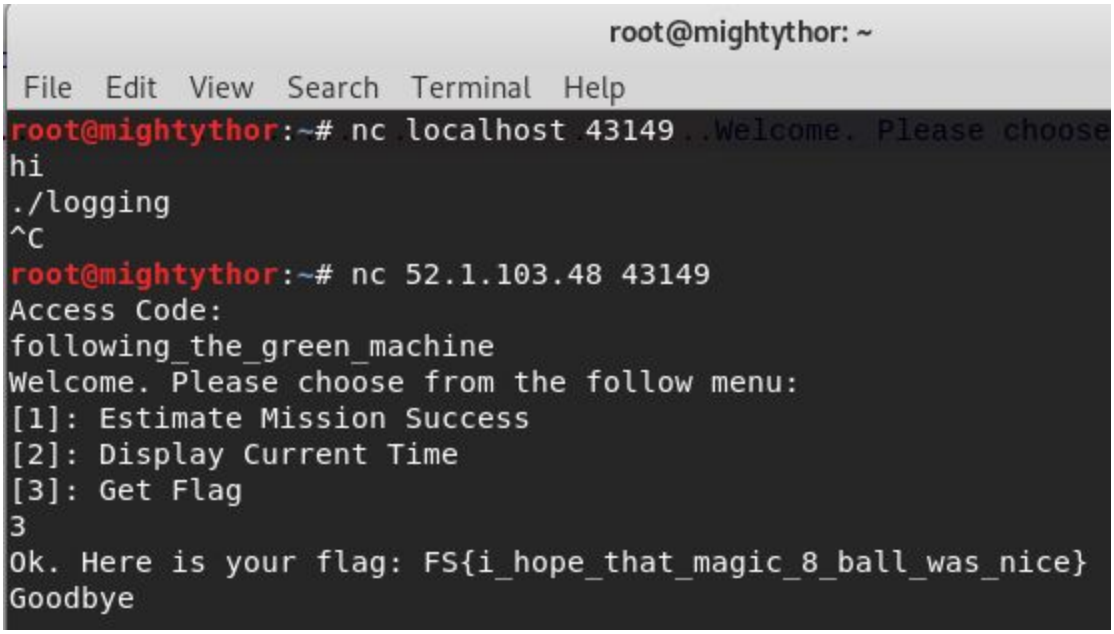
When the problem said to download and capture traffic, I knew this would involve Wireshark. I downloaded the file, booted up Wireshark, ran the file and went to go look at the TCP stream for clues. I found the text above talking about an access code and a menu that involved getting a flag, so I knew I was on the right track.



Ethernet · 1 | IPv4 · 3 | IPv6 | TCP · 4 | UDP · 3

Address A	Port A	Address B	Port B	Packets	Bytes	P
10.0.2.15	38654	52.1.26.21	443	4	292	
52.1.103.48	43149	10.0.2.15	36900	12	1,014	
52.1.103.48	43149	10.0.2.15	36902	12	1,014	
52.1.103.48	43149	10.0.2.15	36904	12	1,014	

I went under Analyze and into the Conversations to see where this was coming from. This got me the IP of 52.1.103.48 on port 43149. This led me to try a few different things that didn't work. I tried to curl into the site at that port and echo “following_the_green_machine” into the curl, but was instantly denied. I tried in the browser to go to this address with the parameters ?access_code=following_the_green_machine but that did the same.



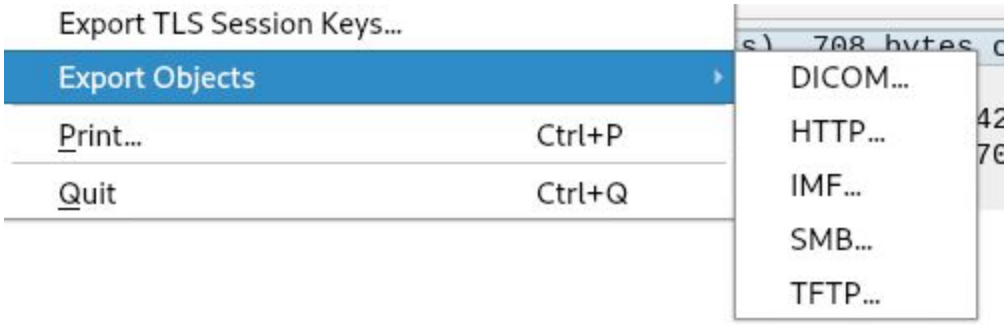
I needed a way to get access to the site and port and be able to input information. I remembered netcat(and that we hadn't used it in the achievement yet), typed in the given access code, then 3 for Get Flag: FS{i_hope_that_magic_8_ball_was_nice}

Name Server 2

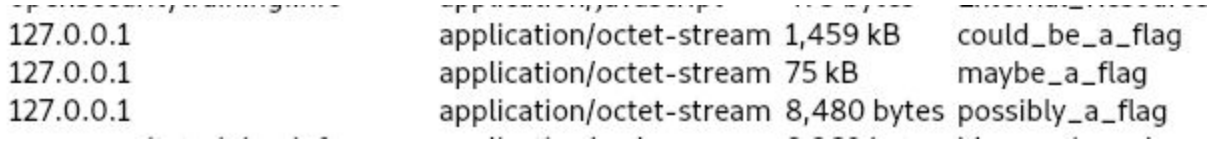
```
root@mightythor:~/Python/dict_labs# host defender.fullstackacademy.com
defender.fullstackacademy.com has address 13.37.13.37
defender.fullstackacademy.com has IPv6 address ::ffff:19.55.19.55
```

I knew I could use nslookup, host, or dig to grab this info. I chose host and saw two different addresses. Knowing the top is formatted in IPv4, Flag: FS{13.37.13.37}

Packet Capture



From the start this problem reminded me of the Stay Out of the Water 2 problem where there was a jpeg Jaws picture we had to export from Wireshark. So I used wget to download the capture, then opened it in Wireshark. Once there I went to File and down to the Export tab for HTTP traffic

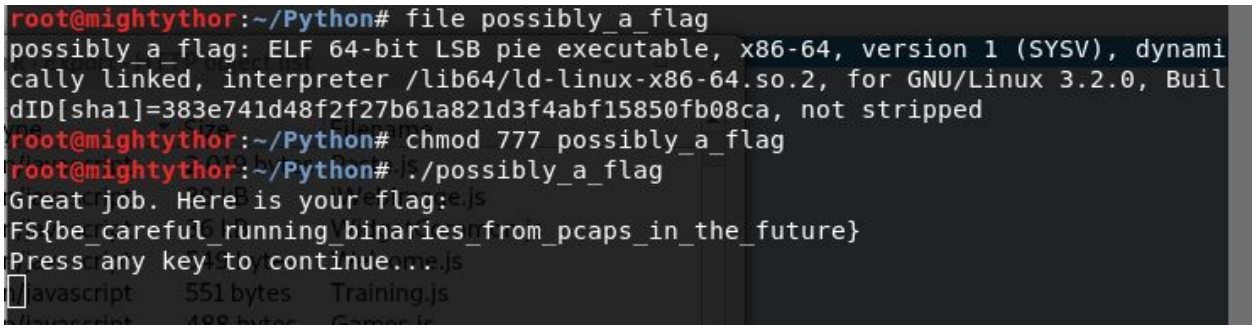


I found these 3 interesting things:

maybe_a_flag: an excellent Yoda meme letting me know to move on

could_be_a_flag: an even greater joke of a massive text file, when grepped into for 'FS' finds you a flag written as 'FS{this_is_not_the_flag}'

Possibly_a_flag: couldn't be opened so it had me question what type of file it actually was, so I used the file command in Linux to check and saw it was an executable



So i changed the permissions on the file and ran it to get Flag:
FS{be_careful_running_binaries_from_pcaps_in_the_future}

Services

Unsuccessful (Edit: I solved it! But wanted to leave the documentation of my 4 day insanity adventure down many wrong rabbit holes)

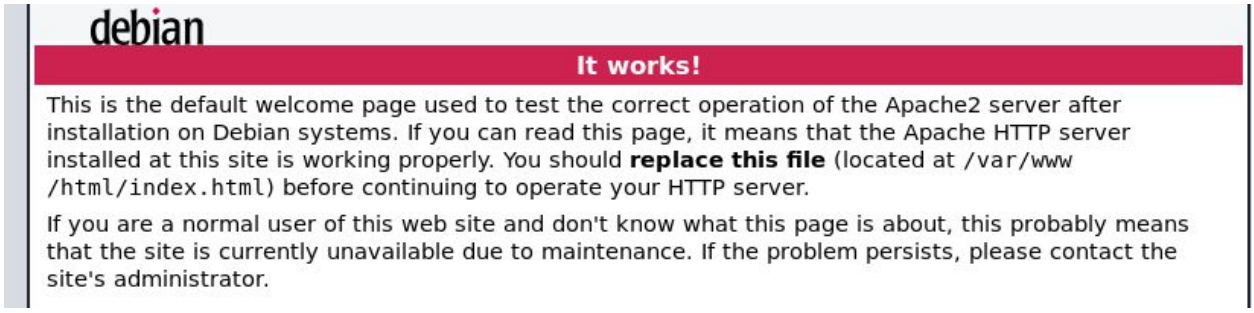
Steps Attempted:

- I was successfully able to use scp to secure copy the .white-hat-manifesto.txt file from the shell to my VM with:
    - Scp myshellusername@cyberlabhost:whitemanifesto.txt /local/VM/path
- The biggest roadblock I had was interpreting " serve this file up via **HTTP** at http://127.0.0.1/ethics/white-hat-manifesto.txt" from the Challenge Problem.
- My first issue was that it says HTTP but the methods for this problem I've been using were secure. And when I used nmap I didn't see port 80 even operational. 22 was my only option.
- I tried a lot of similar things that all led to the same result of me not having a working password to connect to 127.0.0.1 I tried both connecting as my shell

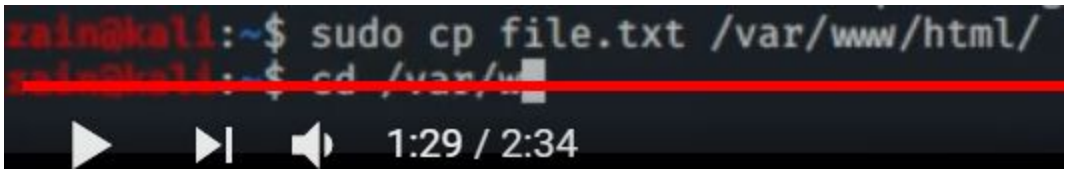
username (andrews1mon) and root. It always required a username, but no password I gave was allowed permission. I tried this with both scp to this location and just ssh to that server.
    - I thought maybe the secret was using the hash in the “special” manifesto text. On every 3rd failed password attempt it prompted (public key,password) I thought maybe that this hash in the text was the public key, but nothing I tried worked.
    - Then I thought maybe the key is to generate a set of keys, but how would I do that without access to the second server
- A previous bonus problem gave me the idea to go into my ssh_config and/or sshd_config files and change PasswordAuthentication to ‘no’, thinking this would bypass the need to enter a password, but this also was unsuccessful
- When I try to use scp to serve up the file from my shell I do have permission but it says where is no path /ethics/white-hat-manifest.txt
    - So this led me to try ‘ssh 127.0.0.1’ in the shell
    - I was able to create a directory named ethics and a folder named white-hat-manifesto.txt. (new path for scp being /home/andrews1mon/ethics/white-hat-manifesto.txt) which worked….. But running services still failed so it was wrong
    - I also tried creating a rsa key pair in this ssh server and sharing the public key with my Kali VM for access but couldn’t get that to work either
- My last thought is that it did say HTTP in the problem so maybe the second half is some sort of POST or PUT request where I can put this data in somehow, but nothing worked
    - This led to the thought of checking the “Inspect” in the browser. When I checked under Network I did see the white-manifesto.txt under Headers but no way to access it

ACTUAL SUCCESS

- It was at this time of desperation that I realized, if stuck, I should take a look at the recommended previous Cyberlabs Problems we were told to review (shocker, I know lol). I found the problem Uget asks for the almost exact thing to be done… I should’ve remembered this when I saw that it asks via HTTP not ssh. I should’ve never been trying to ssh or scp the file to the address
- I already had apache2 on my VM from the Uget problem, I started it up, ran nmap again and 80 was open, and opened apache in my VM Firefox browser



When opening localhost in my apache browser the top section of the default page gave me this info. I found the html file for this page in /var/www. I deleted it like it said, but was a little confused how to properly get files to the server. I found a YouTube video that linked it together for me https://www.youtube.com/watch?v=Nj3lhHTdsBk



In the video he did an example where he just moved a sample file to that directory and was able to access it through apache. So based on what was asked in the problem (http://127.0.0.1/ethics/white-hat-manifesto.txt) I made a folder called “ethics”, changed the name of the file to just white-hat-manifesto.txt, and moved it into this ethics folder in the html folder. This gave me something like this on the browser:

With the white-hat-manifesto.txt inside this folder. I then went back to my Kali machine, ran wget on the services executable, ran services and got:

```
root@mightythor:~/Python# ./services
Command not found or exited with error status
Error: Success
result = -1Error number = 32677
root@mightythor:~/Python# ./services
result = 1Great job. Here is your flag:
FS{im_now_seeing_that_manifesto_in_my_sleep}
Press Enter to continue...
```

Flag: FS{im_now_seeing_that_manifesto_in_my_sleep}

Sockets

```
root@mightythor:~/Python/dict_labs# echo 'Y' | nc shell.fullstackacademy.com 423
93 > trash
```

Just running ‘nc address port’ required me to give an input. This reminded me of the Plumbing problem a while back. I used the same solution of echoing ‘Y’ into that command, then exporting the output into a file (‘trash’)

```
root@mightythor:~/Python/dict_labs# grep -v 'rubbish' trash
Are you ready to start [Y or N]?
Prompt :Alright, here we go!
F
S
{
m
a
n
```

I knew that I needed to read the contents of this file with every ‘rubbish’ removed. I knew that the command for searching through a file is ‘grep’ and through the manpage I found -v lets us grep the opposite or ‘not this’.

```
root@mightythor:~/Python/dict_labs# grep -v 'rubbish' trash | tr -d '\n'
Are you ready to start [Y or N]?Prompt :Alright, here we go!FS{man,_this_challen
ge_was_a_load_of_rubbish}root@mightythor:~/Python/dict_labs#
```

I remembered from a past challenge problem to get rid of newlines was “tr -d ‘\n’. So I piped than in to give us Flag: FS{man,_this_challenge_was_a_load_of_rubbish}

Web

```
    Here is your flag Part1: FS{web_f67f787e99

    Looking at the source code is a good way to get started on a web challenge.
-->
```

After following the hyperlink to the sight and doing right-click to Inspect, I was greeted with the first part of the flag

```
29 /*
30  * Here is your flag part 3: 516b8ecb}
31  *
```

I went over to the Sources tab and saw this Part 3 of the flag

```
95 /*  Here is your flag Part 2: 7dee69cbaeeade
96  *
```

Tabbing down into the css page in Sources I found the Part 2

Putting the HTML, JavaScript, and CSS pieces together gives us
Flag: FS{web_f67f787e997dee69cbaeeade516b8ecb}

Python Coding

Fixstart

```
    Set as interpreter
1   #!/usr/bin/env python3
2   import sys
3
4   def fixstart(string):
5       #Define a new string to put characters into
6       newstring = ''
7       #Filter through each character, gettign value and index
8       for index, char in enumerate(string):
9           #If its the first character, put it in newstring
10          if index == 0:
11              newstring += char
12          #If the character is not the first char, put it in newstring
13          elif char != string[0]:
14              newstring += char
15          #Else add a * to newstring instead
16          else:
17              newstring += '*'
18      #Return the newstring
19      return newstring
20
21
22  string = sys.argv[1]
23  print(fixstart(string))
```

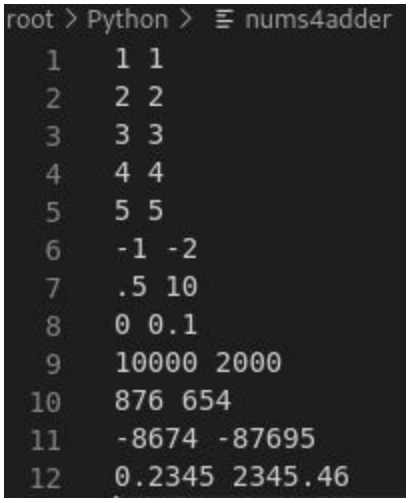
Text Version

```
#!/usr/bin/env python3
import sys

def fixstart(string):
    #Define a new string to put characters into
    newstring = ''
    #Filter through each character, getting value and index
    for index, char in enumerate(string):
        #If its the first character, put it in newstring
        if index == 0:
            newstring += char
        #If the character is not the first char, put it in newstring
        elif char != string[0]:
            newstring += char
```

```
        #Else add a * to newstring instead
        else:
            newstring += '*'
    #Return the newstring
    return newstring


string = sys.argv[1]
print(fixstart(string))
```

Simple Addition

```
Set as interpreter
#!/usr/bin/env python3
import sys

def adder(path):
    #Specify the file as the last element of the path split at each "/" mark
    file = path.split('/')[-1]
    #Open the file up
    f = open(file, 'r')
    #Filter through every line in the file
    for line in f:
        #Split the two numbers into elements of a list
        nums = line.split()
        #Place the addition of the floats of the two numbers in a variable
        newline = float(nums[0]) + float(nums[1])
        #Print that variable (the sum) out for the current line (This is done for each line,
        #printing out all sums in float form)
        #In Chrome theres a single white space in the output lines so I thought this might help, but no
        #newline = str(newline) + ''
        print(newline)

def main():
    path = sys.argv[1]
    adder(path)
main()
```



This is an example text file that I used for testing my code. Lines 1-8 are identical to the test shown in the learndot example. I added some more to test it farther

Lines 1-8 output the exact same as learndot wanted them too. My extra tested numbers seem to run the wanted math just fine. The error I'm getting is:

Submission Time	Status
Sep 7, 2020 5:44 PM	Dies While Running: Script can handle integers
Sep 3, 2020 12:17 PM	Dies While Running: Script can handle integers

In my code the line starts out as a string (ex. Line 1 being '1 1'). When I split each line at each space in Line 12 of the code that line now becomes ['1','1']. Then I take those two strings, cast them into floats, concatenate them, and store the sum as a new variable. (Line 14 of code) With the error it's giving me, maybe it doesn't like the way I'm doing Line 14? Maybe if the numbers are whole it wants them to be seen as integers and only their total as a float? Wanting me to only cast newline as a float after concatenation. But this would become a problem with the later examples having decimals being added together. Only thing I can think of to fix that is some complex if/then scenario where I

say something like if the index's value includes a "." cast it as a float and if not cast it as an integer. Other than that, I can't think of anything that could be the problem involving the script running integers

Text Version

```
#!/usr/bin/env python3
import sys

def adder(path):
    #Specify the file as the last element of the path split at each "/" mark
    file = path.split('/')[-1]
    #Open the file up
    f = open(file, 'r')
    #Filter through every line in the file
    for line in f:
        #Split the two numbers into elements of a list
        nums = line.split()
        #Place the addition of the floats of the two numbers in a variable
        newline = float(nums[0]) + float(nums[1])
        #Print that variable (the sum) out for the current line (This is done for each line,
        #printing out all sums in float form)
        print(newline)

def main():
    path = sys.argv[1]
    adder(path)
main()
```