James Gouveia

Professor Jun Dai

CSC138

04/22/2021

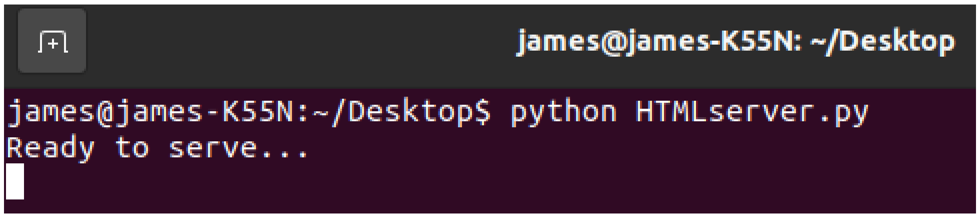
Web Server

Lab Overview:

In this lab we will create a simple web server. The server will be hosted on my computer and will only be available to other processes on the the same computer. In order to achieve this goal, the program will reference localhost where the IP address normally goes in the socket.bind method. The server will return a basic web page based on the sample in the assignment's instructions. The server will have an exception handler that will return a 404 message if the user requests a page that the server does not have.

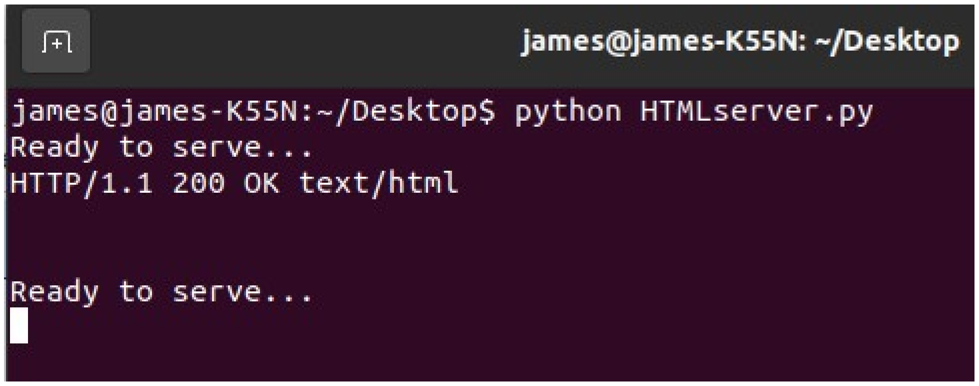
Lab Procedure:

1. Start the server.



```
james@james-K55N:~/Desktop$ python HTMLserver.py
Ready to serve...
```

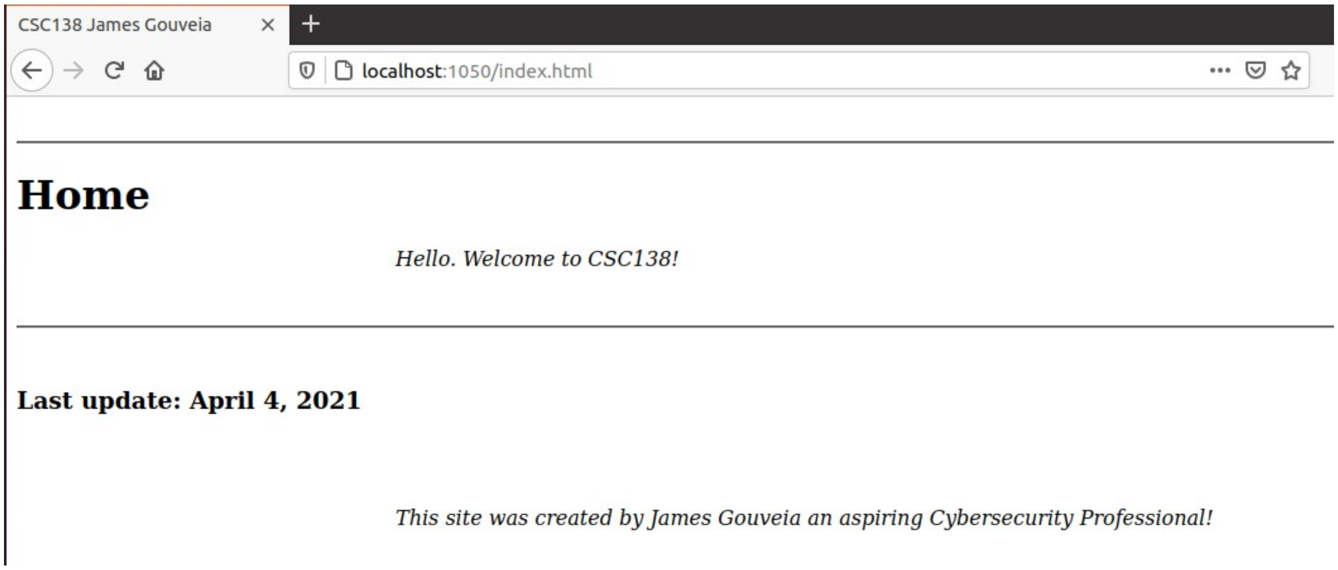
2. Open a web browser and type in ‘localhost:1050/index.html’



```
james@james-K55N:~/Desktop$ python HTMLserver.py
Ready to serve...
HTTP/1.1 200 OK text/html


Ready to serve...
```

Report Continued on next page


CSC138 James Gouveia

localhost:1050/index.html

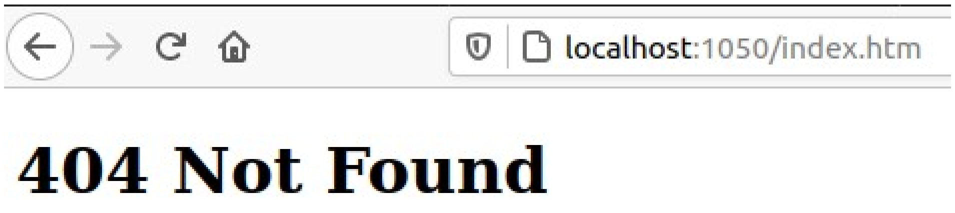
Home

Hello. Welcome to CSC138!

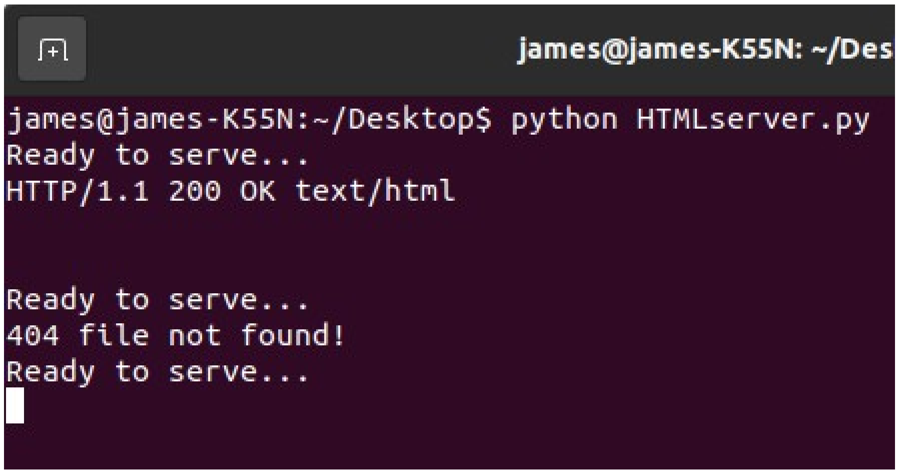
Last update: April 4, 2021

This site was created by James Gouveia an aspiring Cybersecurity Professional!

3. In order to test the exception handler, change the text in the browser to: 'localhost:1050/index.htm'


localhost:1050/index.htm

404 Not Found


james@james-K55N: ~/Des

```
james@james-K55N:~/Desktop$ python HTMLserver.py
Ready to serve...
HTTP/1.1 200 OK text/html


Ready to serve...
404 file not found!
Ready to serve...
```

Report Continued on next page

Python Code:

Note: Please find my explanation of the code annotated in the program.

```
#import socket module
from socket import *

#This is the port for the server, I have chosen a number greater
#than 1024 so I do not interfere with reserved ports
serverPort = 1050

#Create a variable called serverSocket
#Using the python libraries call the socket function
#Add the arguments AF_INET and Sock_STREAM to the function
#AF_INET sets the the IP system to IPv4
#SOCK_STREAM sets the protocol to TCP
serverSocket = socket(AF_INET, SOCK_STREAM)

#This function associates this socket with the contents
#of the variable serverPort, in this case 1050 and also
#binds to the local host so this server is only visible
#to other processes on this computer
serverSocket.bind(('localhost',serverPort))

#This function creates the welcoming door by telling the socket
#to start listening
serverSocket.listen(1)

#This while loop keeps the server alive after a Get request is made
while True:

        #Establish the connection
        #Prints to terminal so I can see the server is ready
        print 'Ready to serve...'
```

```
#This function receives the incoming connection request
#and creates a new socket for this communication
connectionSocket, addr = serverSocket.accept()

#This try/catch is to gracefully handle a file not found (404) error
try:
        #This variable accepts the name of the file the user is trying to access
        message = connectionSocket.recv(1024)

        #This variable holds the filename broken into an array
        filename = message.split()[1]

        #This function takes the filename and opens the file
        f = open(filename[1:])

        #This variable holds the contents of the file
        outputdata = f.read()

        #Send one HTTP header line into socket
        #A sucessful file read will return code 200
        connectionSocket.send('HTTP/1.1 200 OK text/html\n\n')

        #Print message to terminal so I can see what the server
        #is doing
        print 'HTTP/1.1 200 OK text/html\n\n'



        #Send the content of the requested file to the client
        for i in range(0, len(outputdata)):
                connectionSocket.send(outputdata[i])

        #This function closes the connection socket
```

```
			connectionSocket.close()

		#If an error occurs this code gracefully handles it
		except IOError:
			#send response message for file not found
			#In order for the 404 message to be displayed, a simple page was created
			connectionSocket.send("HTTP/1.1 404 Not Found\r\nContent-Type: text/html\r\n\r\n<!
doctype html><html><body><h1>404 Not Found<h1></body></html>")

			#Print message to terminal so I can see what the server
			#is doing
			print '404 file not found!'

			#close client socket
			connectionSocket.close()

#Close the server Socket
#Since the while loop is infinite, this will never be called but the server will remain up
#for multiple Get requests
serverSocket.close()
```