Canape,git、Python[pickle漏洞]

```
└──# nmap -sCV -p80,65535 -A 10.10.10.70
Starting Nmap 7.94SVN ( https://nmap.org ) at 2024-07-21 05:42 PDT
Nmap scan report for 10.10.10.70
Host is up (0.34s latency).

PORT      STATE SERVICE VERSION
80/tcp    open  http    Apache httpd 2.4.29 ((Ubuntu))
|_http-title: Simpsons Fan Site
|_http-server-header: Apache/2.4.29 (Ubuntu)
|_http-trane-info: Problem with XML parsing of /evox/about
| 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
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)
Warning: OSScan results may be unreliable because we could not find at least 1 open
and 1 closed port
Device type: general purpose
Running (JUST GUESSING): Linux 5.X|4.X|2.6.X (97%)
OS CPE: cpe:/o:linux:linux_kernel:5.0 cpe:/o:linux:linux_kernel:4
cpe:/o:linux:linux_kernel:2.6.32
Aggressive OS guesses: Linux 5.0 (97%), Linux 4.15 - 5.8 (90%), Linux 5.0 - 5.4 (90%),
Linux 5.3 - 5.4 (89%), Linux 2.6.32 (89%), Linux 5.0 - 5.5 (88%)
No exact OS matches for host (test conditions non-ideal).
Network Distance: 2 hops
Service Info: OS: Linux; CPE: cpe:/o:linux:linux_kernel

TRACEROUTE (using port 80/tcp)
HOP RTT       ADDRESS
1   367.37 ms 10.10.14.1
```

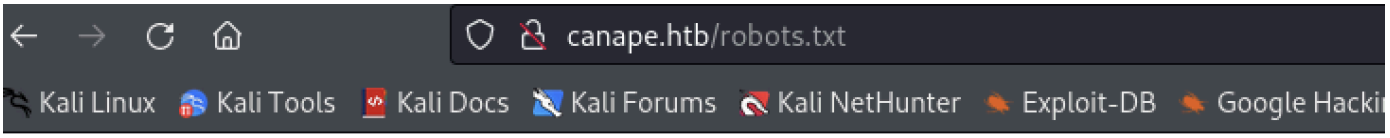
```
2    367.56 ms 10.10.10.70

OS and Service detection performed. Please report any incorrect results at
https://nmap.org/submit/ .
Nmap done: 1 IP address (1 host up) scanned in 24.60 seconds
```

WEB

爆破跟網站無啥重點

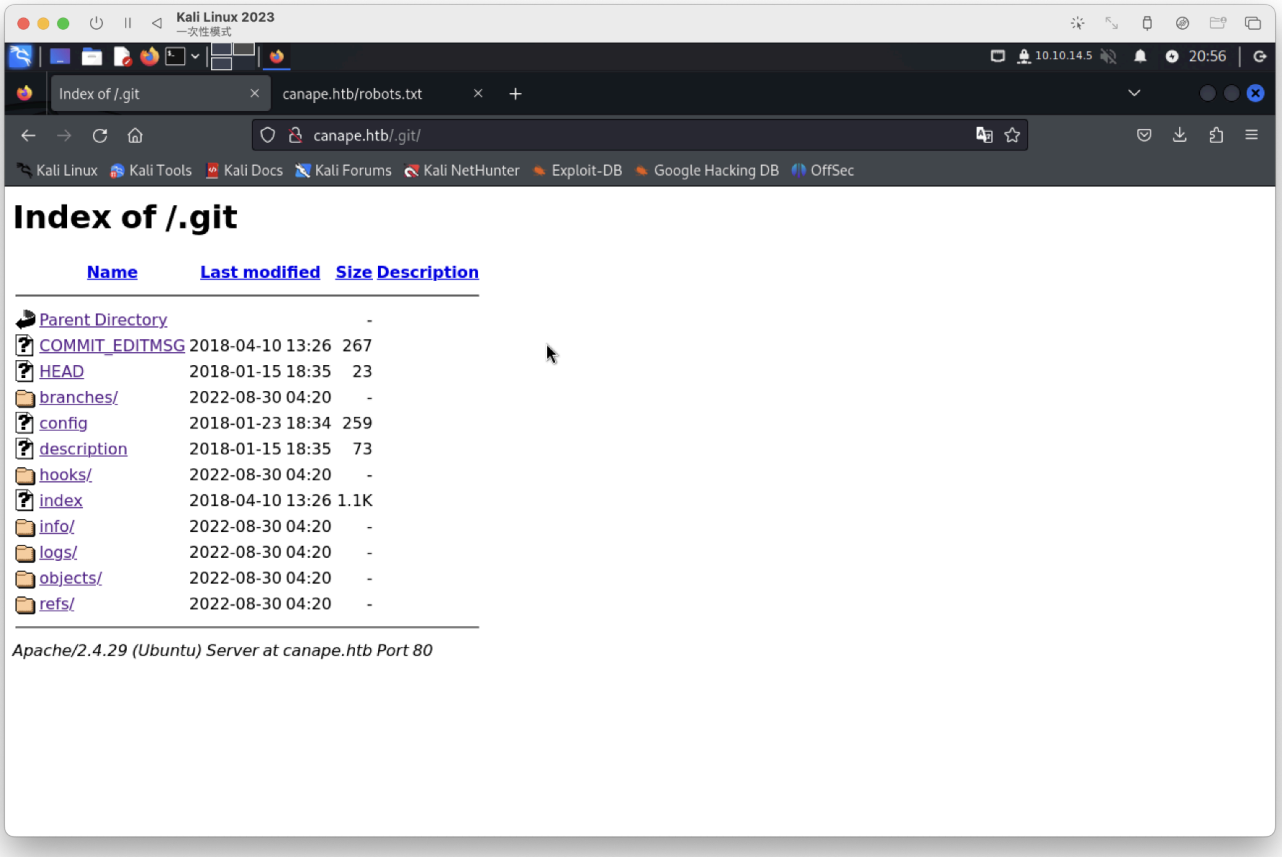
使用index.html or php 、robots.txt有時會出現隨機亂碼，

多刷幾次就不見了



R354KK9Y2HMIWKQZJPSQRQ1EM5EJWFA2A1G8A05SJA23D0DRCKS8XBXSYG8DCNZQSMH

nmap有.git檔案+增加hostsname



(無發現特別，除了config跟)

nmap 也有`http://git.canape.htb/simpsons.git`

進行檔案下載

```
┌──(root㉿kali)-[~]
└─# git clone http://git.canape.htb/simpsons.git
正複製到 'simpsons' ...
remote: Counting objects: 49, done.
remote: Compressing objects: 100% (47/47), done.
remote: Total 49 (delta 18), reused 0 (delta 0)
展開物件中：100% (49/49), 163.16 KiB | 57.00 KiB/s, 完成.

┌──(root㉿kali)-[~]
└─# ls
simpsons

┌──(root㉿kali)-[~]
└─# cd simpsons

┌──(root㉿kali)-[~/simpsons]
└─# ls
__init__.py  static  templates
```

進行code分析

```
└──# cat __init__.py
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()
```

int_py發現
這是處理/submit頁面的部分，它接受POST值character和quote，如果其中任何一個為null以及白名單中不存在該字符，則拋出錯誤，這裡可以引用名稱，如果它們都有效就會p_id使用char+quote資料的md5sum初始化一個變量，然後在tmp資料夾中以.p副檔名建立一個同名檔案（該檔案應該是一個pickle檔案），然後將其填入char+quote資料…

Python的pickle庫有助於序列化資料和存儲，並且像大多數使用各種語言的序列化庫一樣容易受到攻擊，例如Celestial上的NodeJs序列化漏洞…
可以到google搜尋相關pickle漏洞資訊