ETHICAL HACKING

FOOTPRINTING

Banner Grabbing

By \
Ismail Ahmed
V 0.1
BANNER GRABBING
14:30:12 GMT\r\nServer: Apache/2.2.20
dified: Thu, 17 Nov 2005 17:20:54 G
t-Ranges: bytes\r\nContent-Length:
tent-Type: text/html\r\n\r\n<html>\
n</head>\r\n\r\n<body bgcolor="#ffff
height="100%" width="100%" align=center valign=middle>
\r\n<td width="100%" height="100%" align="center" valign="mi
ddle">\r\n <img src="faded_logo.png" >\r\n</td>\r\n</tr>\r\n</table>\r\n\r\n</body
>\r\n</html>\r\n'

Researcher By: Nin Kanong

1. Banner Grabbing

- What is a Banner?

A banner is the information displayed by software or service running on a specific port. This information involves the type of software running, version of the software running etc. This information is displayed by default by every software running for marketing purposes.

- What is banner grabbing?

Grabbing a banner is the first and apparently the most important phase in both the offensive and defensive penetration testing environments. In this article, we'll take a tour to "Banner Grabbing" and learn how the different command-line tools and web interfaces help us to grab the banner of a webserver and its running services.

- What is Banner grabbing?

Banner grabbing as its name implies, is grabbing this banner. A banner when displayed to a common user may provide information to the user. In the same way, by grabbing this banner, hackers and penetration testers can get information about the software running on it and the version of the software running. This allows them to search or research for any vulnerabilities in the software.

- Table of Content

Why Banner Grabbing?

Banner Grabbing allows an attacker to discover network hosts and running services with their versions on the open ports and moreover operating systems so that he can exploit the remote host server.

Banner Disclosure is the most common vulnerability with a **"CWE-200 i.e. Exposure of Sensitive Information to an Unauthorized Actor"** and a **"CVSS Score of 5.0 with the Risk factor as Medium."**

In order to clear the vision, we'll consider an attack scenario:

*As we all know that Microsoft Windows 7 are exploitable by Eternal Blue (**CVE-2017-0143**) directly with **SMBv1** service. In order to enumerate this server, the attacker needs to grabs a service banner which displays whether the SMB service with a vulnerable version is running over it or not. If running, he/she can easily exploit the Microsoft server directly with the Eternal Blue attack. You can learn more about this attack from **here**.*

- Type of Banner grabbing

Banner grabbing can be performed in two ways: Active & passive.

1.**Active banner grabbing:** In active banner grabbing, a hacker or penetration tester interacts with the software & target services to grab the banner.

2.**Passive banner grabbing:** In passive banner grabbing, a hacker or penetration tester doesn't interact with the target service while grabbing the banner. This can be done by packet sniffing on the network traffic of the network.

Although banner grabbing can be performed on almost all services running on all ports, the most common services that are used for banner grabbing are.

FTP-Port 21
SSH-Port 23
SMTP-Port 22
HTTP-Port 80

Banner grabbing using Kali Linux

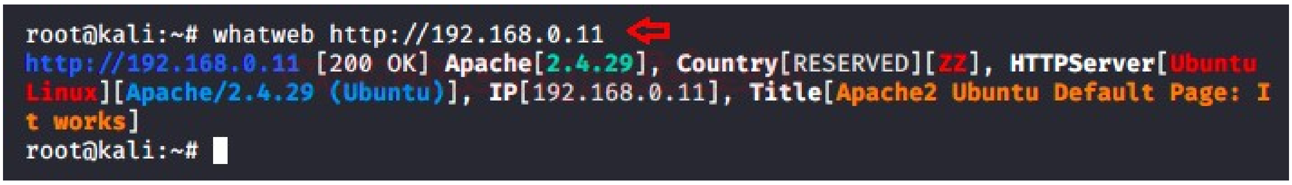
Whatweb

"WhatWeb" recognizes websites, which helps us to grab the web-applications banner by disclosing the server information with its version, the IP address, the webpage Title and running operating system.

Type the following command in order to capture the essentials.

- kali> whatweb <website URL>

- kali> whatweb http://192.168.0.11



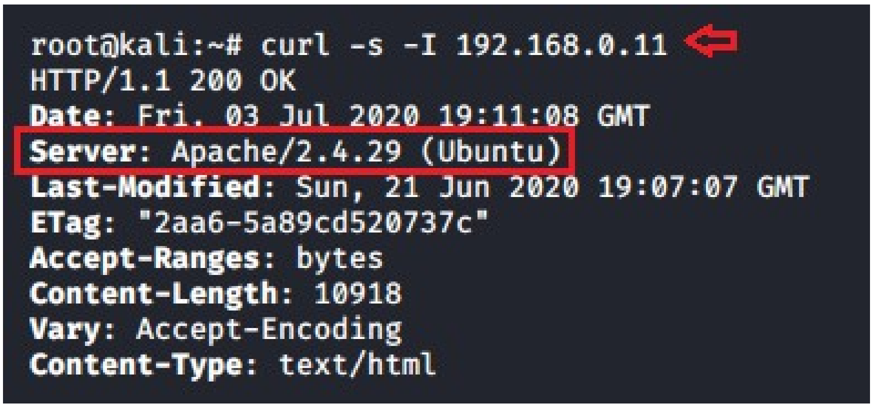
```
root@kali:~# whatweb http://192.168.0.11
http://192.168.0.11 [200 OK] Apache[2.4.29], Country[RESERVED][ZZ], HTTPServer[Ubuntu
Linux][Apache/2.4.29 (Ubuntu)], IP[192.168.0.11], Title[Apache2 Ubuntu Default Page: I
t works]
root@kali:~#
```

- cURL

The cURL command includes the functionality for retrieving the banner details from HTTP servers. Just execute the following command, and discover what we grab:

- kali> curl –s –I 192.168.0.11

However to fetch a clean result, we are using the -s flag to prevent the progress of the error messages from being displayed, and the -I flag to simply print out the header information of all requested pages.



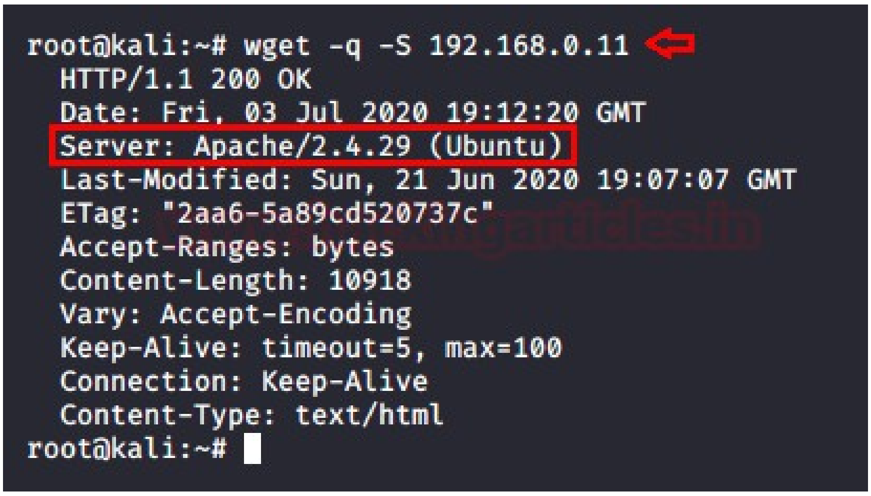
```
root@kali:~# curl -s -I 192.168.0.11
HTTP/1.1 200 OK
Date: Fri, 03 Jul 2020 19:11:08 GMT
Server: Apache/2.4.29 (Ubuntu)
Last-Modified: Sun, 21 Jun 2020 19:07:07 GMT
ETag: "2aa6-5a89cd520737c"
Accept-Ranges: bytes
Content-Length: 10918
Vary: Accept-Encoding
Content-Type: text/html
```

- Wget

We will be using the wget command to capture the **HTTP banner** of the remote server.

wget –q –S 192.168.0.11

The –q flag will cover-up the progress of our output, while the -S flag will print out the header information of all requested pages.



```
root@kali:~# wget -q -S 192.168.0.11
  HTTP/1.1 200 OK
  Date: Fri, 03 Jul 2020 19:12:20 GMT
  Server: Apache/2.4.29 (Ubuntu)
  Last-Modified: Sun, 21 Jun 2020 19:07:07 GMT
  ETag: "2aa6-5a89cd520737c"
  Accept-Ranges: bytes
  Content-Length: 10918
  Vary: Accept-Encoding
  Keep-Alive: timeout=5, max=100
  Connection: Keep-Alive
  Content-Type: text/html
root@kali:~#
```

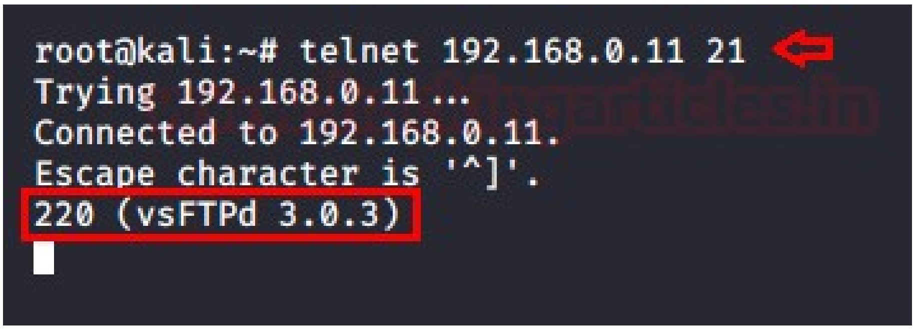
Telnet

We will be using the Telnet protocol in order to interact with services to grab their banners.

Type following command to grab **the FTP banner** of the remote server.

telnet 192.168.0.11 21

As a result, it will dumb “**220 (vsFTPd 3.0.3)**”



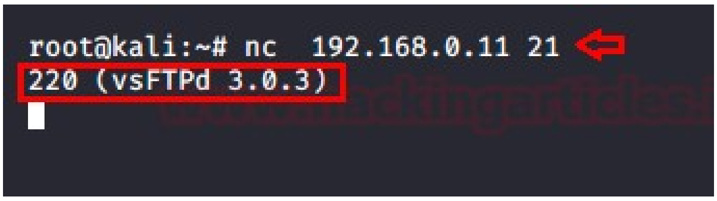
```
root@kali:~# telnet 192.168.0.11 21
Trying 192.168.0.11...
Connected to 192.168.0.11.
Escape character is '^]'.
220 (vsFTPd 3.0.3)
```

- Netcat

Netcat is a network utility that will again help us to grab the **FTP banner** of the remote host server.

nc 192.168.0.11 21

From the above image, you can check that it dumbs up **“220 (vsFTPd 3.0.3)”**



```
root@kali:~# nc  192.168.0.11 21
220 (vsFTPd 3.0.3)
```

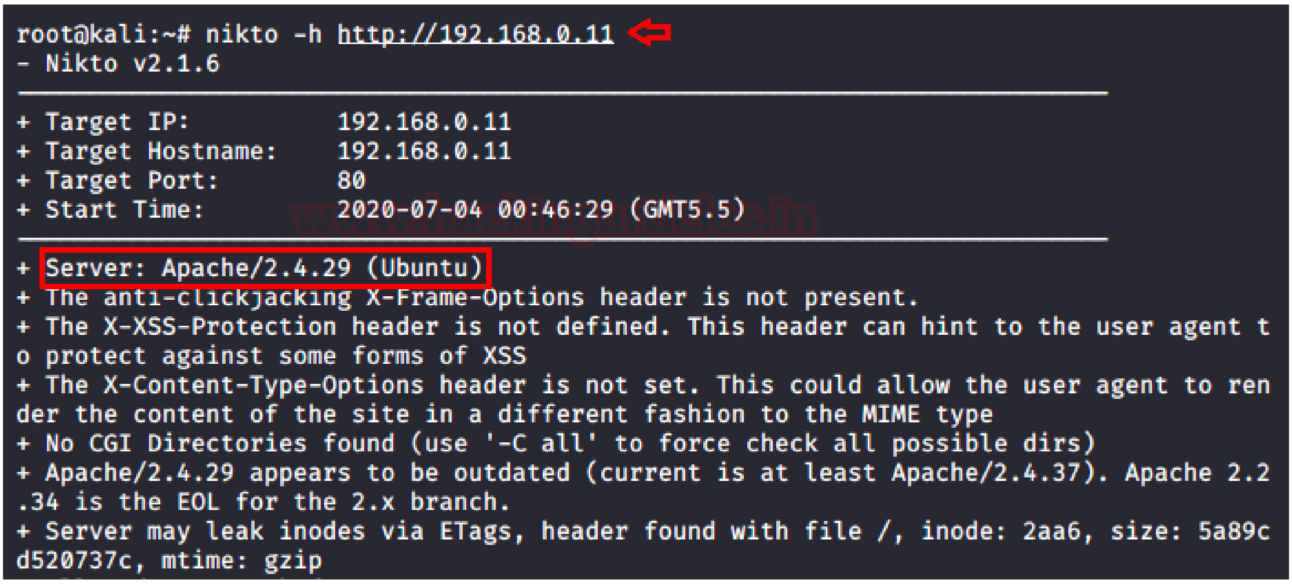
Nikto

Nikto is an open-source web-application scanner, which we'll be using to grab a banner of a website running on an Ubuntu server.

Type the following command in order to capture the installed web server – its version, the configuration index files, the HTTP server options and a list of other useful details.

nikto –h http://192.168.0.11

The –h flag is used to specify the host.



```
root@kali:~# nikto -h http://192.168.0.11
- Nikto v2.1.6
---------------------------------------------------------------------------
+ Target IP:          192.168.0.11
+ Target Hostname:    192.168.0.11
+ Target Port:        80
+ Start Time:         2020-07-04 00:46:29 (GMT5.5)
---------------------------------------------------------------------------
+ Server: Apache/2.4.29 (Ubuntu)
+ The anti-clickjacking X-Frame-Options header is not present.
+ The X-XSS-Protection header is not defined. This header can hint to the user agent t
o protect against some forms of XSS
+ The X-Content-Type-Options header is not set. This could allow the user agent to ren
der the content of the site in a different fashion to the MIME type
+ No CGI Directories found (use '-C all' to force check all possible dirs)
+ Apache/2.4.29 appears to be outdated (current is at least Apache/2.4.37). Apache 2.2
.34 is the EOL for the 2.x branch.
+ Server may leak inodes via ETags, header found with file /, inode: 2aa6, size: 5a89c
d520737c, mtime: gzip
```

Nmap

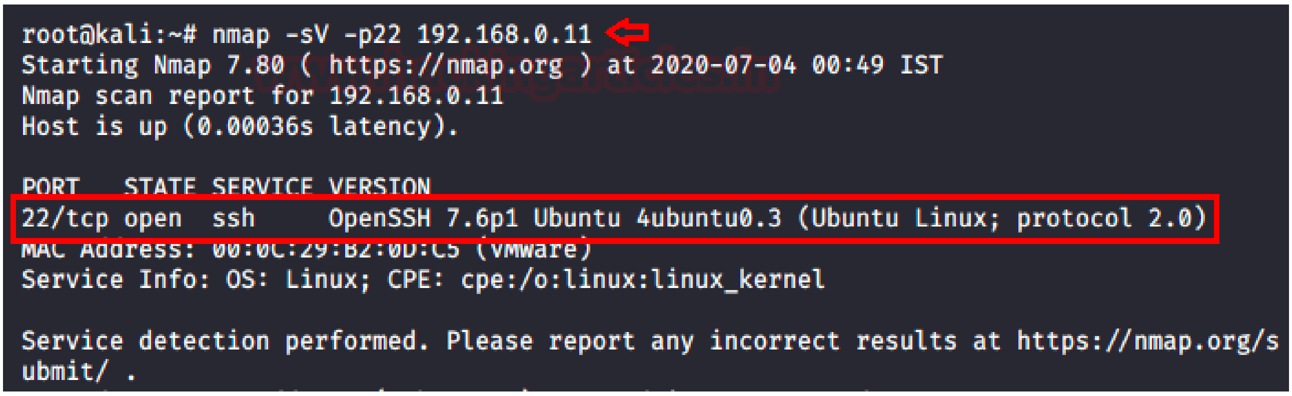
We'll use Nmap as a simple banner grabber which connects to an open TCP port and prints out anything sent by the listening service within a couple of seconds

Then, type following command which will grab banner for the **SSH** service running on port **22** in the remote host.

nmap -sV –p22 192.168.0.11

The -sV flag prints out the version of the running service.

From the above screenshot, you can read the SSH service and its version, fetched by NMAP as **"OpenSSH 7.6p1 Ubuntu 4ubuntu0.3"**



```
root@kali:~# nmap -sV -p22 192.168.0.11
Starting Nmap 7.80 ( https://nmap.org ) at 2020-07-04 00:49 IST
Nmap scan report for 192.168.0.11
Host is up (0.00036s latency).

PORT   STATE SERVICE VERSION
22/tcp open  ssh     OpenSSH 7.6p1 Ubuntu 4ubuntu0.3 (Ubuntu Linux; protocol 2.0)
MAC Address: 00:0C:29:B2:0D:C5 (VMware)
Service Info: OS: Linux; CPE: cpe:/o:linux:linux_kernel

Service detection performed. Please report any incorrect results at https://nmap.org/s
ubmit/ .
```

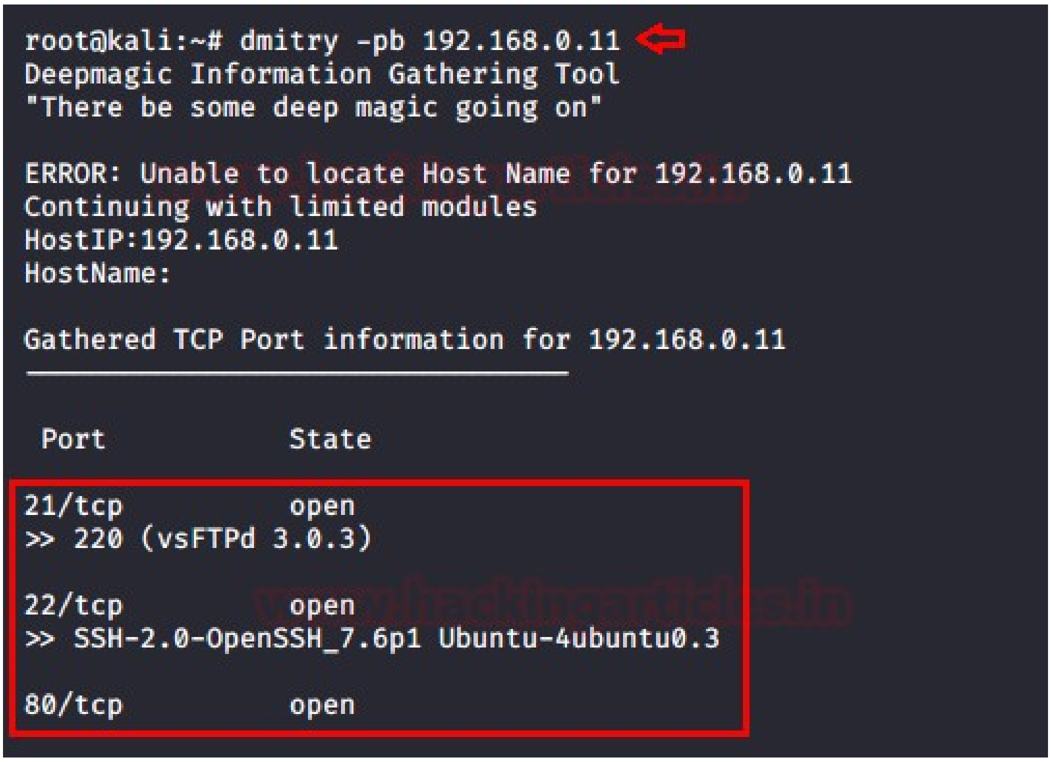
Dmitry

Dmitry (Deepmagic Information Gathering Tool) has the ability to gather as much information as possible about a host. Base functionality is able to gather possible subdomains, email addresses, uptime information, tcp port scan, whois lookups, and many more.

The –pb flag is used to grab the banner for all the open-ports of the remote host.

Fire the following command to grab the banners of the running services.

dmitry –pb 192.168.0.11



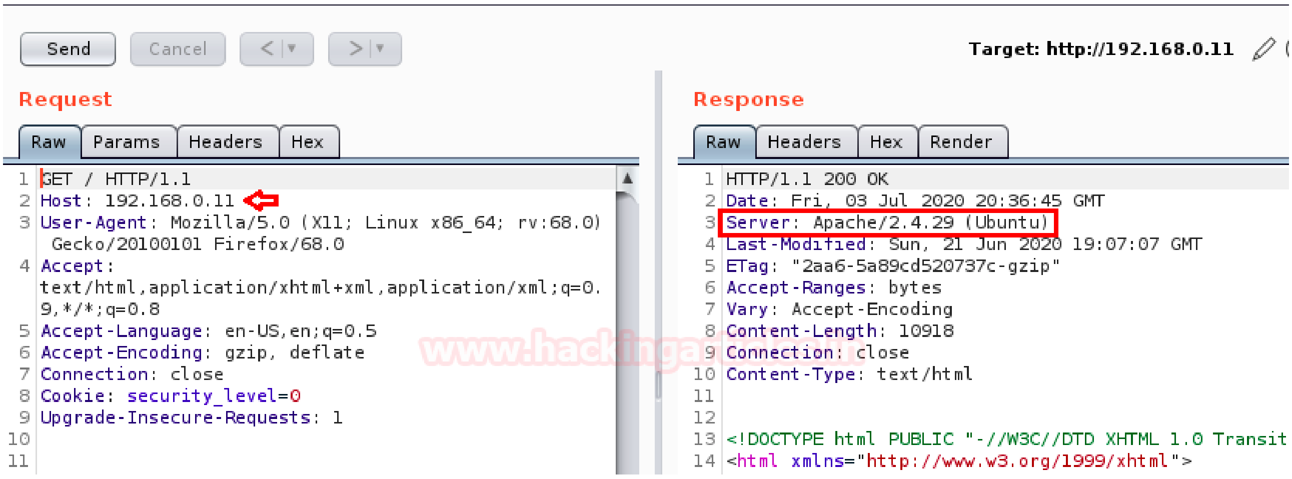
```
root@kali:~# dmitry -pb 192.168.0.11
Deepmagic Information Gathering Tool
"There be some deep magic going on"

ERROR: Unable to locate Host Name for 192.168.0.11
Continuing with limited modules
HostIP:192.168.0.11
HostName:

Gathered TCP Port information for 192.168.0.11
---------------------------------

 Port           State

21/tcp          open
>> 220 (vsFTPd 3.0.3)

22/tcp          open
>> SSH-2.0-OpenSSH_7.6p1 Ubuntu-4ubuntu0.3

80/tcp          open
```

+ Banner Grabbing over Burpsuite

While performing an attack or a penetration test, we all use burp suite somewhere or the other, but does it help us to identify the target's web server?

Yes, we can simply grab the server's information through the response generated by the **repeater.**

Consequently, from the below screenshot you can see that I've sent the interpreted request into the Repeater. As soon as I hit the **Send** button, the response will be executed, and on the right-hand side, you will get the captured server details as **Apache/2.4.29 (Ubuntu)**



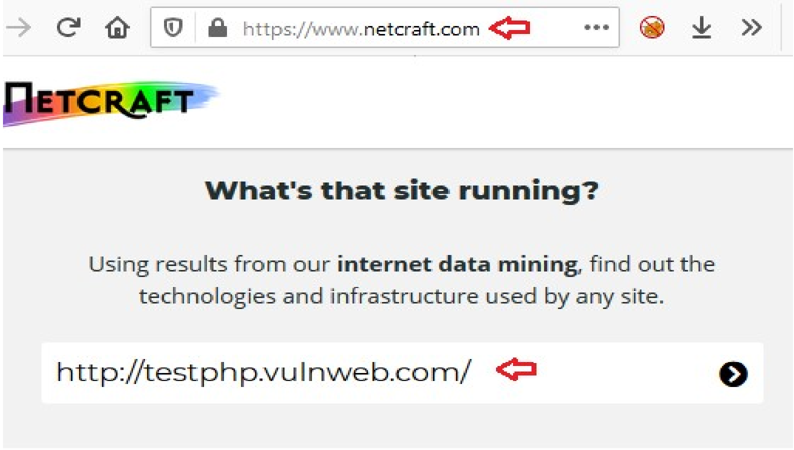
```
GET / HTTP/1.1
Host: 192.168.0.11
User-Agent: Mozilla/5.0 (X11; Linux x86_64; rv:68.0)
 Gecko/20100101 Firefox/68.0
Accept:
text/html,application/xhtml+xml,application/xml;q=0.
9,*/*;q=0.8
Accept-Language: en-US,en;q=0.5
Accept-Encoding: gzip, deflate
Connection: close
Cookie: security_level=0
Upgrade-Insecure-Requests: 1
```

```
HTTP/1.1 200 OK
Date: Fri, 03 Jul 2020 20:36:45 GMT
Server: Apache/2.4.29 (Ubuntu)
Last-Modified: Sun, 21 Jun 2020 19:07:07 GMT
ETag: "2aa6-5a89cd520737c-gzip"
Accept-Ranges: bytes
Vary: Accept-Encoding
Content-Length: 10918
Connection: close
Content-Type: text/html


<!DOCTYPE html PUBLIC "-//W3C//DTD XHTML 1.0 Transit
<html xmlns="http://www.w3.org/1999/xhtml">
```

+ Banner Grabbing using Netcraft

Netcraft is one of the most operatable information gathering web-interface which help us to check the technologies and the infrastructure of the web-applications.

So I'll be using a demo website over Netcraft in order to grab some service banners and capture all the possible information.

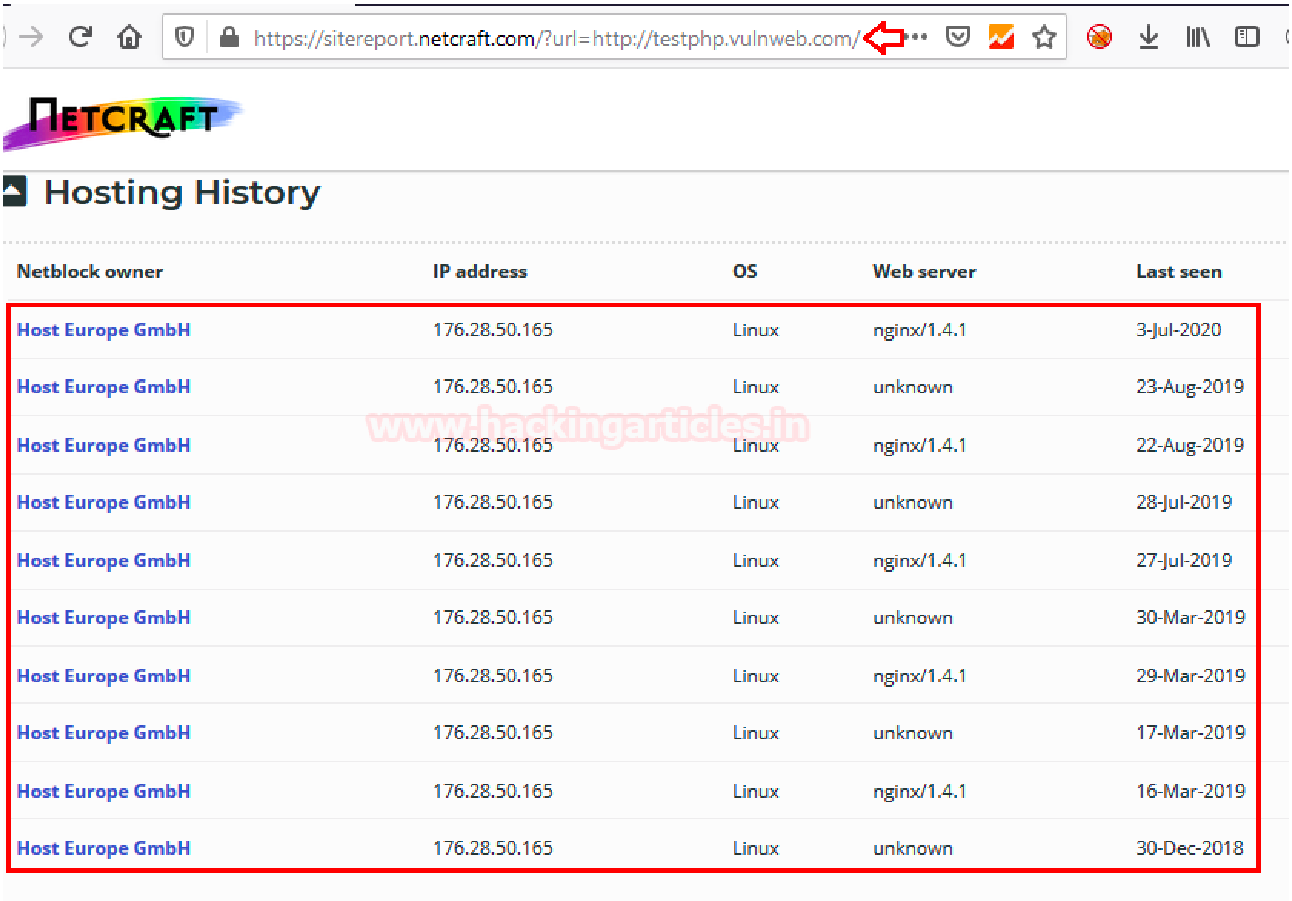


What's that site running?

Using results from our **internet data mining**, find out the technologies and infrastructure used by any site.

http://testphp.vulnweb.com/

From the above image, you can see that I have grabbed the **Hosting History** of testphp.vulnweb.com, which shows up the IP addresses, the operating systems and the webservers along with their last seen.



https://sitereport.netcraft.com/?url=http://testphp.vulnweb.com/

Hosting History

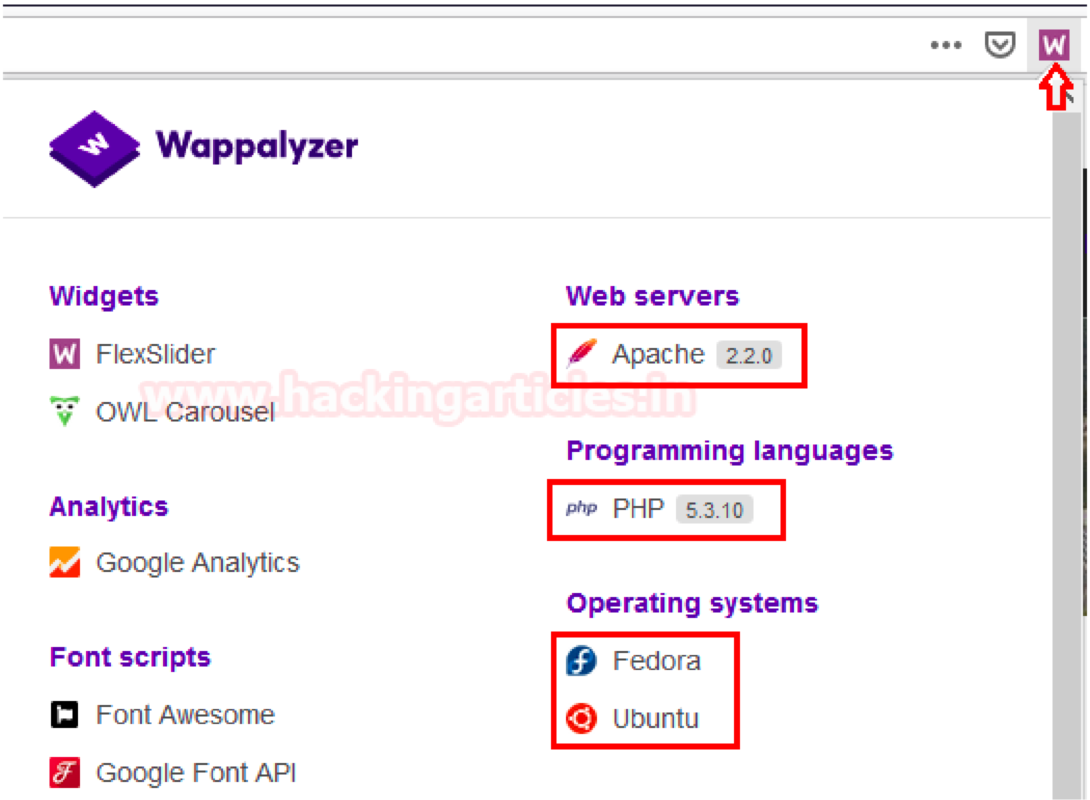
Netblock owner	IP address	OS	Web server	Last seen
Host Europe GmbH	176.28.50.165	Linux	nginx/1.4.1	3-Jul-2020
Host Europe GmbH	176.28.50.165	Linux	unknown	23-Aug-2019
Host Europe GmbH	176.28.50.165	Linux	nginx/1.4.1	22-Aug-2019
Host Europe GmbH	176.28.50.165	Linux	unknown	28-Jul-2019
Host Europe GmbH	176.28.50.165	Linux	nginx/1.4.1	27-Jul-2019
Host Europe GmbH	176.28.50.165	Linux	unknown	30-Mar-2019
Host Europe GmbH	176.28.50.165	Linux	nginx/1.4.1	29-Mar-2019
Host Europe GmbH	176.28.50.165	Linux	unknown	17-Mar-2019
Host Europe GmbH	176.28.50.165	Linux	nginx/1.4.1	16-Mar-2019
Host Europe GmbH	176.28.50.165	Linux	unknown	30-Dec-2018

+ Banner Grabbing through Browser Extensions

Sometimes it’s a bit time consuming while grabbing banners of multiple web applications. Thus in order to make our work faster, we will be setting up some browser extensions that will help us to capture the server information with their version numbers, the running operating systems and the other frameworks that drive up the web applications.

+ Wappalyzer

Wappalyzer is a free browser extension available for both Mozilla Firefox and Google Chrome. It helps us to check the technologies of the web-application, majorly the server with its version and the framework running on it. You can add this extension in your browser from here.



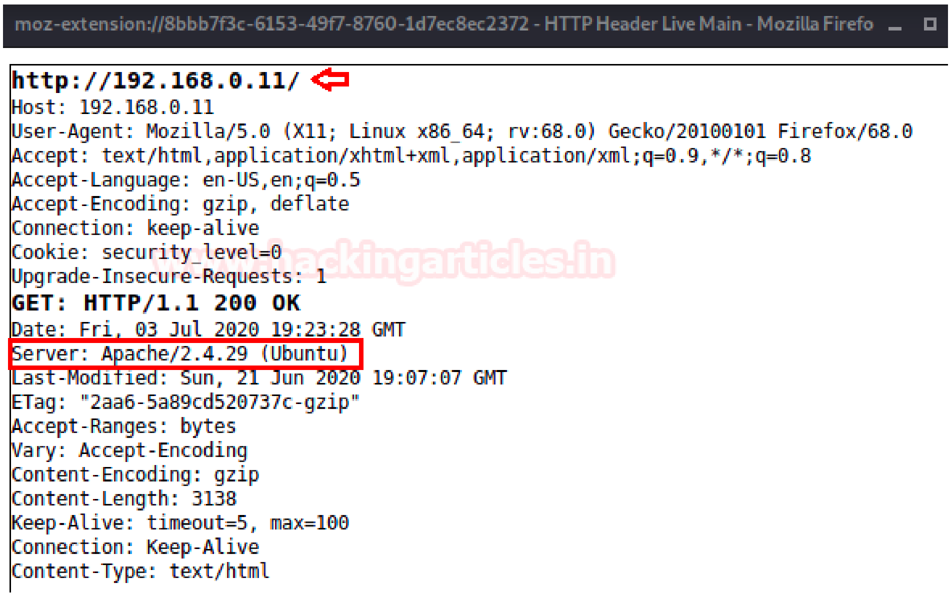
From the above image you can see that, we have easily captured “Apache 2.2.0” as the server, “PHP 5.3.10” as the programming language and “Ubuntu and Fedora” as the running operating systems.

HTTP Header Live

This extension gives us the power to capture the ongoing HTTP Requests before they are sent to the server.

Therefore we are going to garb some server banners through this HTTP Header extension. You can add it in your browser from here.

From the below image you can see that, as soon as I capture the HTTP request, I was presented with the target’s information containing the server and the operating system banners i.e. **Apache/2.4.29 (Ubuntu)**

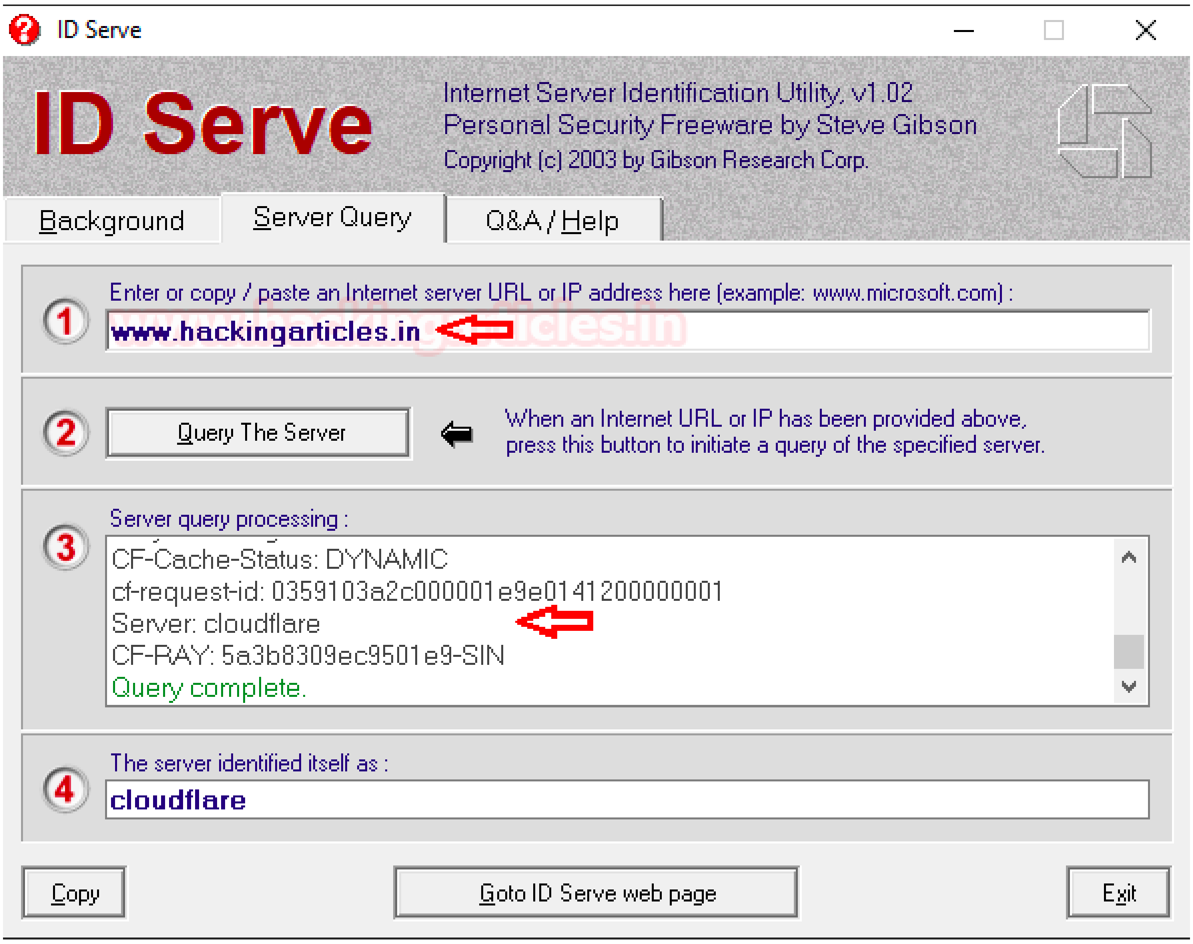

moz-extension://8bbb7f3c-6153-49f7-8760-1d7ec8ec2372 - HTTP Header Live Main - Mozilla Firefo

```
http://192.168.0.11/
Host: 192.168.0.11
User-Agent: Mozilla/5.0 (X11; Linux x86_64; rv:68.0) Gecko/20100101 Firefox/68.0
Accept: text/html,application/xhtml+xml,application/xml;q=0.9,*/*;q=0.8
Accept-Language: en-US,en;q=0.5
Accept-Encoding: gzip, deflate
Connection: keep-alive
Cookie: security_level=0
Upgrade-Insecure-Requests: 1
GET: HTTP/1.1 200 OK
Date: Fri, 03 Jul 2020 19:23:28 GMT
Server: Apache/2.4.29 (Ubuntu)
Last-Modified: Sun, 21 Jun 2020 19:07:07 GMT
ETag: "2aa6-5a89cd520737c-gzip"
Accept-Ranges: bytes
Vary: Accept-Encoding
Content-Encoding: gzip
Content-Length: 3138
Keep-Alive: timeout=5, max=100
Connection: Keep-Alive
Content-Type: text/html
```

+ Banner Grabbing using ID Serve

ID Server is a free and a general-purpose Internet server identification utility which helps us to grab the banner of a remote host. You can download the tool from **here.**

Just enter the target's website URL and hit the **"Query This Server"** button. And there it goes, it dumps everything it could, including the IP addresses, open ports, cookie and the server information.



Internet Server Identification Utility, v1.02
Personal Security Freeware by Steve Gibson
Copyright (c) 2003 by Gibson Research Corp.

1 Enter or copy / paste an Internet server URL or IP address here (example: www.microsoft.com) :
www.hackingarticles.in

2 When an Internet URL or IP has been provided above, press this button to initiate a query of the specified server.

3 Server query processing :

```
CF-Cache-Status: DYNAMIC
cf-request-id: 0359103a2c000001e9e0141200000001
Server: cloudflare
CF-RAY: 5a3b8309ec9501e9-SIN
Query complete.
```

4 The server identified itself as :
cloudflare

Author: Chiragh Arora is a passionate Researcher and Technical Writer at Hacking Articles. He is a hacking enthusiast. Contact **here**

REFERENCE

1. **HackArticle**: https://www.hackingarticles.in/multiple-ways-to-banner-grabbing/

2. **HackerMagazine**: https://www.hackercoolmagazine.com/banner-grabbing-for-beginners/

Contact:

Gmail: ninkanong200620@gmail.com

Facebook: នីន កាណុង(Nin Kanong)

Telegram: @ninkanong

Tel: 0978297806