Crocodile Write-up

Created	@November 25, 2025 8:54 PM

Introduction

Tier I is all about exploitation vectors that chain together to offer you the possibility of gaining a foothold on the target from one service to another. Credentials could be lost somewhere in a publicly accessible folder which would let you login through a remote shell left untended and unmonitored. A misconfigured service could be leaking information that might allow you to impersonate the digital identity of a victim. Any number of possibilities exist in the real world. However, we will start with some simpler ones.Tackling an example sewed together from two other previous targets, we will be looking at an insecure access configuration on FTP and an administrative login for a website. Let us proceed to deconstruct this vector and analyze its' components.

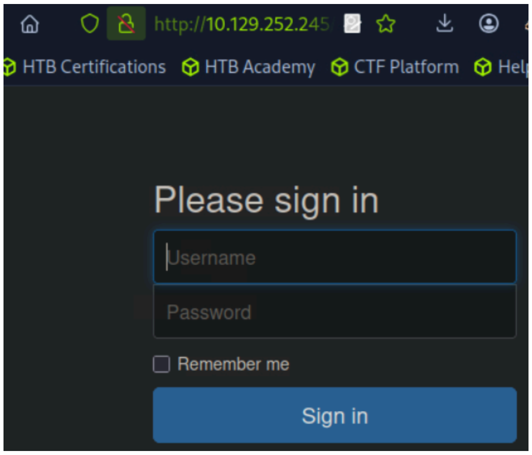
Enumeration

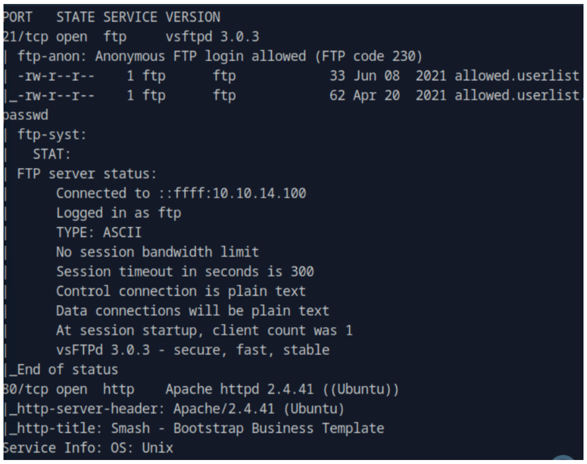
We will start by enumerating the target. Our first step is, as always, a thorough nmap scan. By using the following two switches for the scan, we ensure that our nmap script analyses the service being run on any port found in the open state and returns a mostly exact service version value in the output and that all of the default analysis scripts are run against the target, as we are not constrained on how intrusive we can be

with our scan. Running the scan as mentioned, we can receive results as seen below, with snippets of directories the scan has even found for us!

-sC: Performs a script scan using the default set of scripts. It is equivalent to --script=default. Some of the scripts in this category are considered intrusive and should not be run against a target network without permission.

-sV: Enables version detection, which will detect what versions are running on what





```
PORT   STATE SERVICE VERSION
21/tcp open  ftp     vsftpd 3.0.3
| ftp-anon: Anonymous FTP login allowed (FTP code 230)
| -rw-r--r--    1 ftp      ftp            33 Jun 08  2021 allowed.userlist
|_-rw-r--r--    1 ftp      ftp            62 Apr 20  2021 allowed.userlist.
passwd
| ftp-syst:
|   STAT:
| FTP server status:
|      Connected to ::ffff:10.10.14.100
|      Logged in as ftp
|      TYPE: ASCII
|      No session bandwidth limit
|      Session timeout in seconds is 300
|      Control connection is plain text
|      Data connections will be plain text
|      At session startup, client count was 1
|      vsFTPd 3.0.3 - secure, fast, stable
|_End of status
80/tcp open  http    Apache httpd 2.4.41 ((Ubuntu))
|_http-server-header: Apache/2.4.41 (Ubuntu)
|_http-title: Smash - Bootstrap Business Template
Service Info: OS: Unix
```

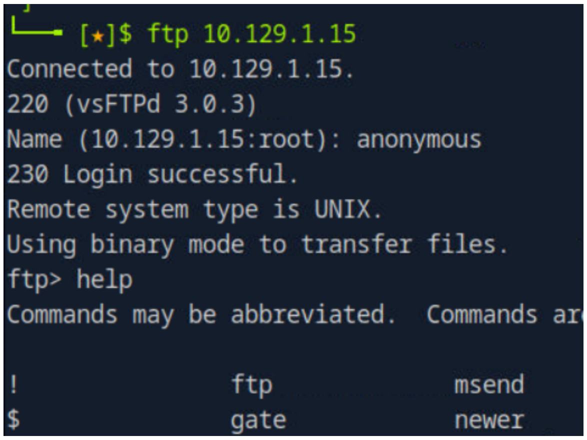
We have two open ports: 21 and 80. Port 21 is the port dedicated to FTP (File Transfer Protocol), meaning that its' primary use is to transfer files between hosts on the same network.
According to Wikipedia, a quick reminder:

The File Transfer Protocol (FTP) is a standard communication protocol used to transfer computer files from a server to a client on a computer network. FTP users may authenticate themselves with a clear-text sign-in protocol, generally using a username and password. However, they can connect anonymously if the server is configured to allow it.

Users could connect to the FTP server anonymously if the server is configured to allow it, meaning that we could use it even if we had no valid credentials. If we look back at our nmap scan result, the FTP server is indeed configured to allow anonymous login:

> ftp-anon: Anonymous FTP login allowed (FTP code 230)

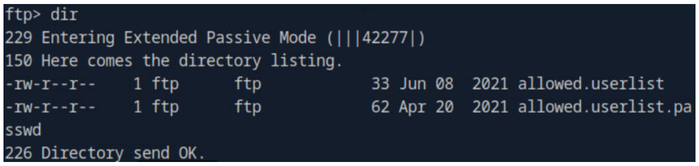
The File Transfer Protocol (FTP) is a standard communication protocol used to transfer computer files from a server to a client on a computer network. FTP users may authenticate themselves with a clear-text sign-in protocol, generally using a username and password. However, they can connect anonymously if the server is configured to allow it.
If you need a refresher, the ftp -h command will help you figure out the available commands for the FTP service on your local host.



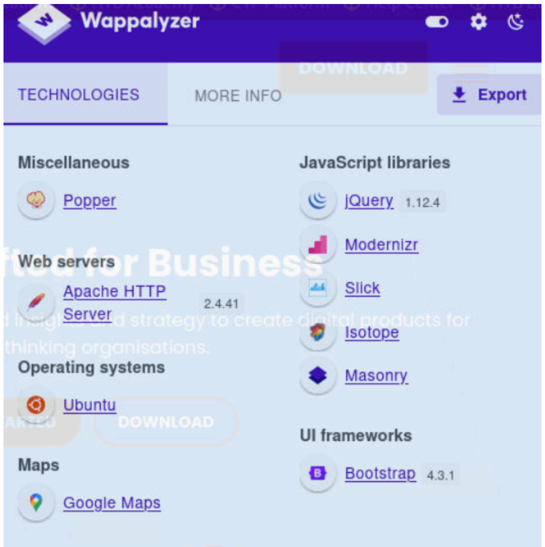
```
└──╼ [★]$ ftp 10.129.1.15
Connected to 10.129.1.15.
220 (vsFTPd 3.0.3)
Name (10.129.1.15:root): anonymous
230 Login successful.
Remote system type is UNIX.
Using binary mode to transfer files.
ftp> help
Commands may be abbreviated.  Commands ar

!               ftp             msend
$               gate            newer
```

Foothold
After the credentials have been obtained, the next step is to check if they are used on the FTP service for
elevated access or the webserver running on port 80 discovered during the nmap scan. Attempting to log in
with any of the credentials on the FTP server returns err



```
ftp> dir
229 Entering Extended Passive Mode (|||42277|)
150 Here comes the directory listing.
-rw-r--r--    1 ftp      ftp            33 Jun 08  2021 allowed.userlist
-rw-r--r--    1 ftp      ftp            62 Apr 20  2021 allowed.userlist.pa
sswd
226 Directory send OK.
```



Miscellaneous
Popper

Web servers
Apache HTTP Server 2.4.41

Operating systems
Ubuntu

Maps
Google Maps

JavaScript libraries
jQuery 1.12.4
Modernizr
Slick
Isotope
Masonry

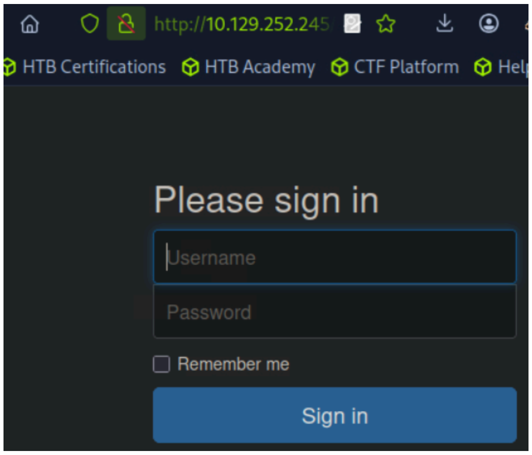
UI frameworks
Bootstrap 4.3.1

From the output of Wappalyzer, we can note some of the more interesting items, specifically the PHP programming language used to build the web page. However, nothing gives us a direct plan of attack for now. Meanwhile, navigating around the page using the tabs and buttons provided on it leads us nowhere.

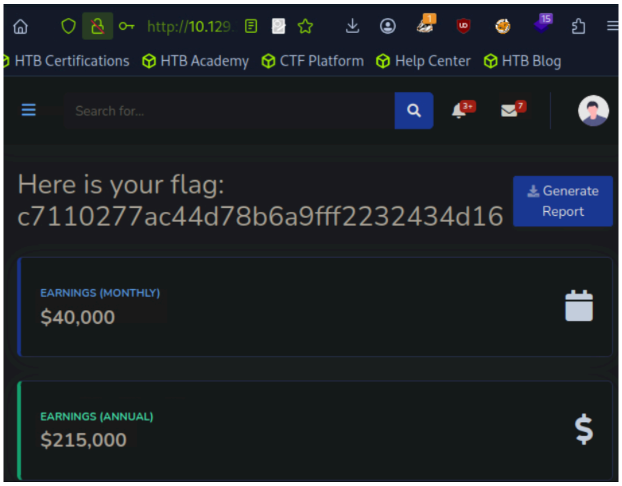
Referencing previous write-ups, there is mention of a different, more direct way of navigating any hidden or hardly accessible directories and pages, and that is through dir busting. Using gobuster as our tool of choice, we can use the following switches for the script to get the fastest, most accurate results.

For the -x switch, we can specify php and html to filter out all the unnecessary clutter that does not interest us. PHP and HTML files will most commonly be pages. We might get lucky and find an administrative panel login page that could help us find leverage against the target in combination with the credentials we extracted from the FTP server.

dir : Uses directory/file enumeration mode.
--url : The target URL.
--wordlist : Path to the wordlist.
-x : File extension(s) to search for.



One of the most interesting files gobuster retrieved is the /login.php page. Navigating manually to the URL, in the form of http://{target_IP}/login.php , we are met with a login page asking for a username/password combination.



Here is your flag:
c7110277ac44d78b6a9fff2232434d16

EARNINGS (MONTHLY)
$40,000

EARNINGS (ANNUAL)
$215,000