Pico CTF 2013 Answers –By Chris Issing

1. Failure to Boot 20:

ANSWER: FAT

1. Read the Manual 30:

ANSWER: cbcfebeaeeed

Copy the text to the manual and passed it to a text file. Use the Caesar Cipher to decrypt it and then send it to another text file to read later. Find out the shift by figuring out the most common word. The most common word was ‘the’ and in the ciphered text; it translated to ‘aol’. Use that as the key to figure out what the shift was, then make a python script to decode it using the Caesar Cipher made earlier

1. XMLOL 30:

ANSWER: d6b6aba9c44bf3dd58809c46e2c0ffba

Download the file and then read it with a text editor.

1. Technician Challenge 30:

ANSWER: [Nissan 350Z](http://en.wikipedia.org/wiki/Nissan_350Z)

I just looked it up because it was simple.

1. First Contact 40:

ANSWER: 37 14'06"N 115 48'40"W

I just looked through the pcap file for the new locations that’s all.

1. GREP IS YOUR FRIEND 40:

ANSWER: fHYYpdrfeOCHyQicfe96xfw==

Download all the files and then grep them in the terminal

1. TRIVIAL 45:

ANSWER: th4ts\_w0rs3\_th4n\_DES

I modified the source code to do the revese of the encryption. It is in the folder with this.

1. TRY THEM ALL 45:

ANSWER: dog

I put it in an online decryptor and removed the ‘37’ at the end because there was a salt of 37

1. GETKey 50:

ANSWER: 9fa449c061d64f58de600dfacaa6bd5d

I changed the url to say ‘admin=true&compietion=picoctf’

1. Spaceport Map 55:

ANSWER: Do passports let you fly interstellar?

I took a video of the gif and slowed it down to see what the key was.

1. Bitwise 55:

ANSWER: ub3rs3cr3t

I wrote a python code that test all the possibilities and prints out he correct one if it equals the original hash.

1. Yummy 60:

ANSWER:

I used a chrome plugin that allowed me to edit the cookies, so I added a cookie that had the name ‘Authorization’ and the key ‘Administrator’

1. SPAMCARVER 60:

ANSWER:

We used a filecarver software to recover the txt file in the picture.

1. CLASSIC 70:

ANSWER: when\_do\_we\_get\_to\_the\_hard\_stuff

I found a website that I used to decrypt the cipher by finding the most common word in the text and using it as a starting point to decrypt it all.

1. BYTECODE 70:

ANSWER:

I used a software called jd-gui to decompile the .class file and then read the contents of the Authenticator.class.

1. RSA 70:

ANSWER: UNSOLVED ☹

I could not figure out how to do this math.

1. NAVASAT 70:

ANSWER: UNSOLVED ☹

I could not find any software that could recover the txt file.

1. Client-Side is the Best 75:

ANSWER: cl13nt\_s1d3\_1s\_w0rst\_s1d3

I used google chrome to inspect the element and then I found the location of the flag in the source code.

1. Pilot Logic 75:

ANSWER: You can't take the sky from me

I went to the pilotbots home on the pilot\_logic.img and did ‘ls –a’ to display all the hidden and non hidden folders. I found a folder called ‘.secret’ so I cd’ed into it and catted the file called ‘key’.

1. Robocommunication 80:

ANSWER: boopbeep

We listened to the sound file and found out that it was in morse code so we translated it to find out the flag.

1. Python Eval 1 85:

ANSWER: eval\_is\_best\_thing\_evar

In the input of the first number I put in the string ‘int(flag)

1. avaJ 85:

ANSWER:

I used the jd-gui application to decompile it.

1. PHP2 85:

ANSWER: UNSOLVED ☹

Can’t figure it out.

1. DDOS DETECTION 85:

ANSWER: UNSOLVED ☹

We were very close but we never figured it out.

1. Second Contact 85:

ANSWER: I don’t remember, Hector solved it

1. hotcoffe 85:

ANSWER:

Used jd-gui.

1. In Hex No One Can Hear You Complain 90:

ANSWER: UNSOLVED ☹

1. Overflow 1 90:

ANSWER: overflow\_is\_best\_flow

I catted the source code and saw that the buffer was at 64, so I used python to print a bunch of ‘A’’s to overflow the buffer.