1 Cryptography PicoCTF Challenges

1.1 Easy1

1.1.1 Description

The one time pad can be cryptographically secure, but not when you know the key. Can you solve this? We've given you the encrypted flag, key, and a table to help {encrypted text} with the key of {key}. Can you use this table to solve it?

1.1.2 Solution

Look up a one-time pad decrypter. I like: www.boxentriq.com/code-breaking/one-time-pad. It should be pretty simple to just plug in values and get the key!

1.2 Mr-Worldwide

1.2.1 Description

A musician left us a message. What's it mean?

1.2.2 Solution

This one is a bit daft but that's okay! It'll teach you how to deal with the average user when you end up working in IT. The flag is in the form

$$\mathsf{picoCTF}\{(x_1, y_1), (x_2, y_2), \ldots, (x_k, y_k)\}$$

The x, y pairs can be interpreted in many ways (I originally thought they were the frequency bands for musical notes) but they're really coordinates in $\mathsf{Lat}, \mathsf{Lon}$ form. If you take the first character of the city these coordinates land in, you'll have your flag!

1.3 Flags

1.3.1 Description

What do the flags mean?

1.3.2 Solution

Time to put r/vexillologycirclejerk knowledge to use. While it may seem these are national flags, especially since you should be able to recognize the French *Tricolore*, they're really not. A key observation for that claim is that most national flags follow $2:3$ or $3:5$ dimension ratio however most of these flags are squares. Instead, they're International Maritime Signal flags and you can look up a table to decode them. Wikipedia has a handy one!

1.4 john_pollard

1.4.1 Description

Sometimes RSA certificates are breakable

1.4.2 Solution

An actual cryptography challenge! The challenge file is cert. You can use openssl to get the public key from the certificate. Try these commands:

```
openssl x509 -pubkey -in cert > key
openssl rsa -pubin -in key -text | grep Modulus
```

If you're familiar with RSA encryption, you'll realise that this modulus is rather small. Now, you can use factorDB to find the factors or, if you're interested, implement the Pollard ρ algorithm to factor the number. You should have your flag! PS: This algorithm is called the 'rho' algorithm because if you draw out the path it takes, you'd end up with the Greek letter ρ.

1.5 miniRSA

1.5.1 Description

Let's decrypt this: ciphertext? Something seems a bit small.

1.5.2 Solution

This one might be a bit involved. As described by the RSA encryption algorithm, $c = m^e \bmod N$. Usually, when e is a large number, raising m^e results in a wrap around due to N. For example, $2^5 = 32$ but if you set $N = 5$, then $2^5 \equiv_5 32 \equiv_5 2$. However, if the power is small, say 2 in our example, then there is no wrap around, $2^2 = 4 \equiv_5 4$. This is exacerbated when N is as large as it is in the challenge. As such, this problem reduces to finding the cube root of c, formally $m = \sqrt[3]{c}$. A quick note on floats: if you attempt to take the cube root directly, you're likely to run into errors; I suggest using a binary search approach that exclusively uses ints. Once you get m, it should be as simple as taking the hex value and getting the alphanumeric representation through something like: www.duplichecker.com/hex-to-text.php