B01lers CTF 2020

b01lers

Welcome to b01lers CTF presented by the
Purdue University Capture the Flag Team!

Matryoshka

Challenge 13 Solves ×

Matryoshka

200

Super-secret password: один два ... ...

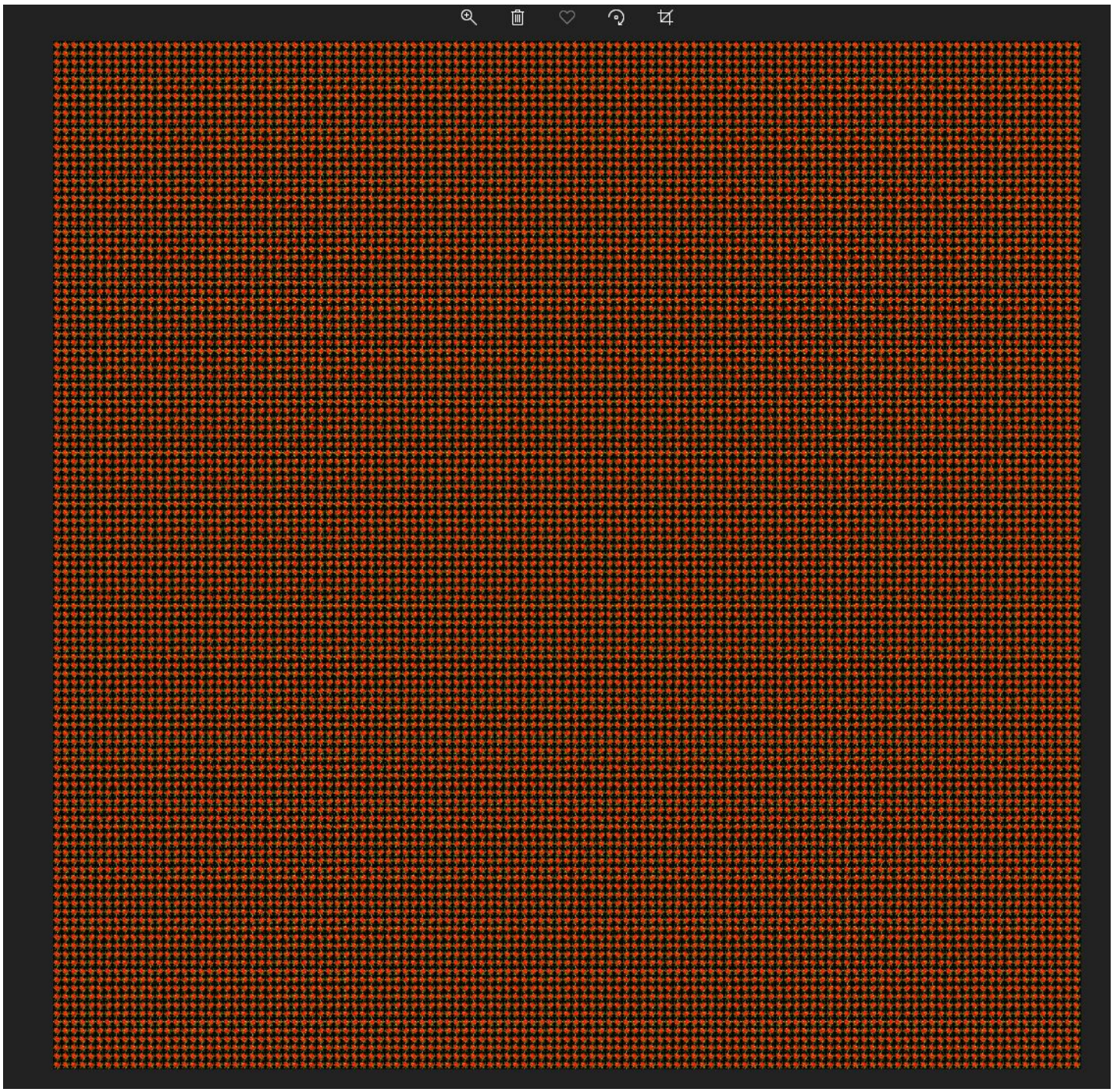
0d975215bc72fd3d13205218c00a1b23

(created by dm)

Flag Submit

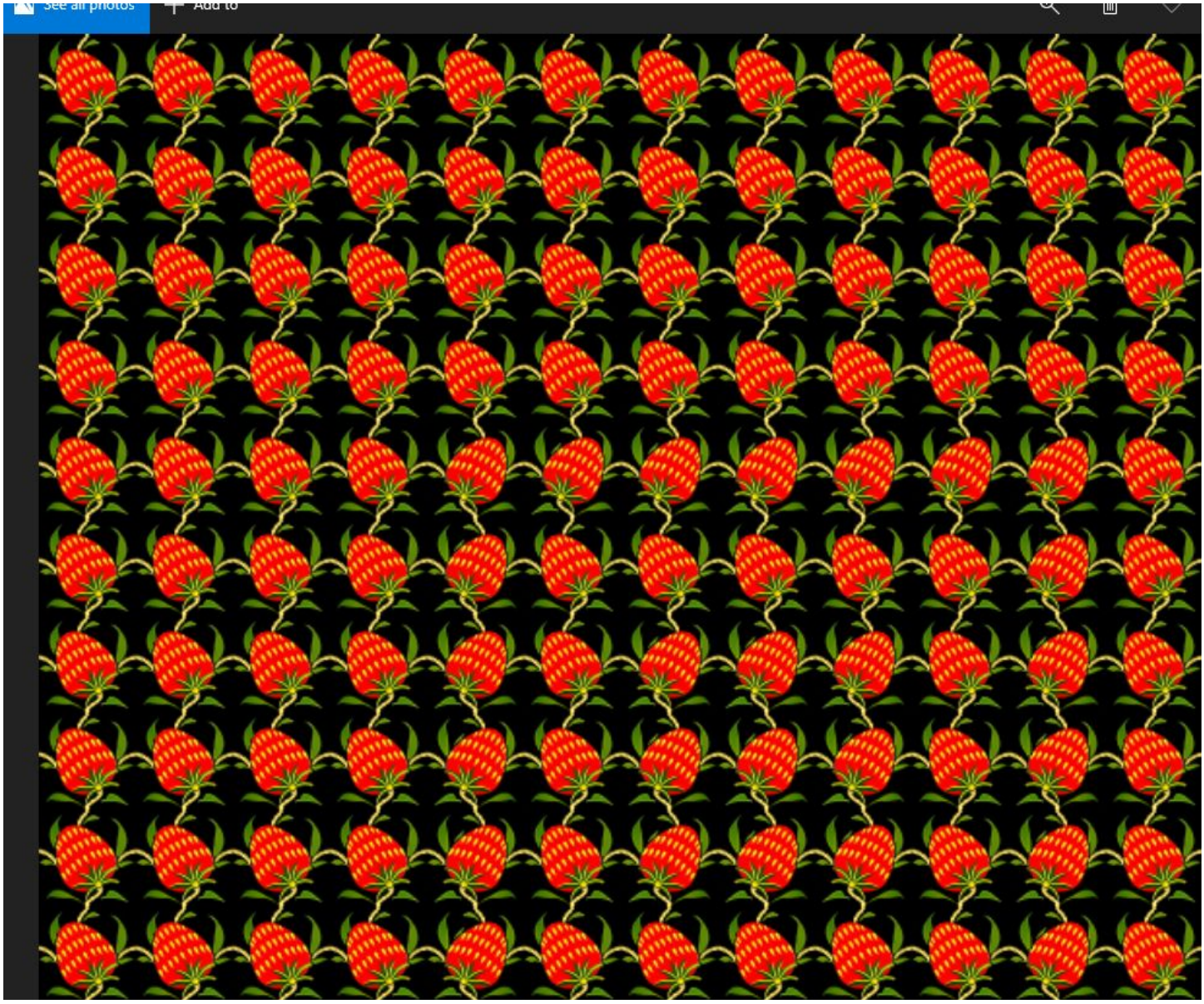
Download

Download the zip and open it up. One file inside. Looks like an image.



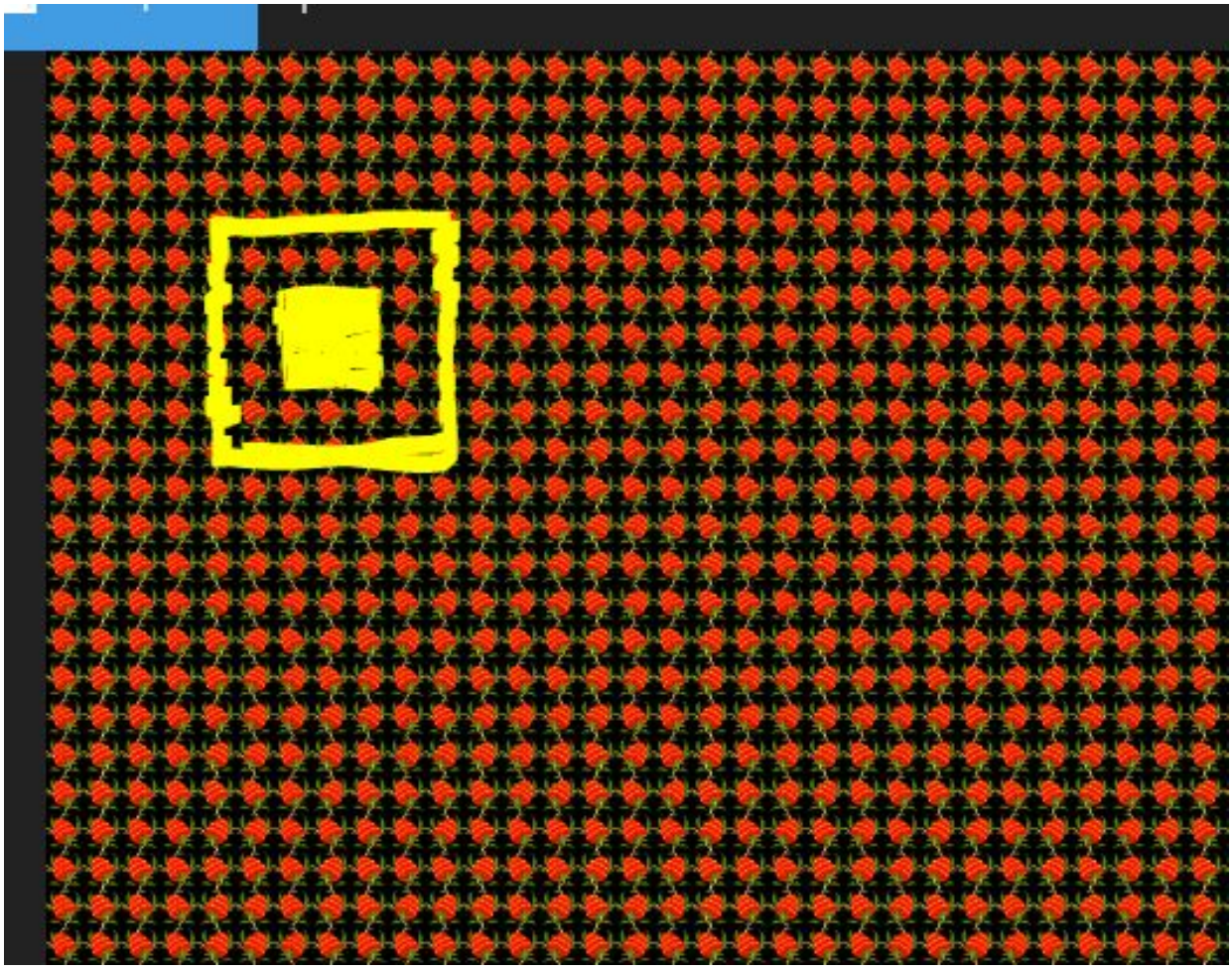
Strawberries

Zoom in a bit.



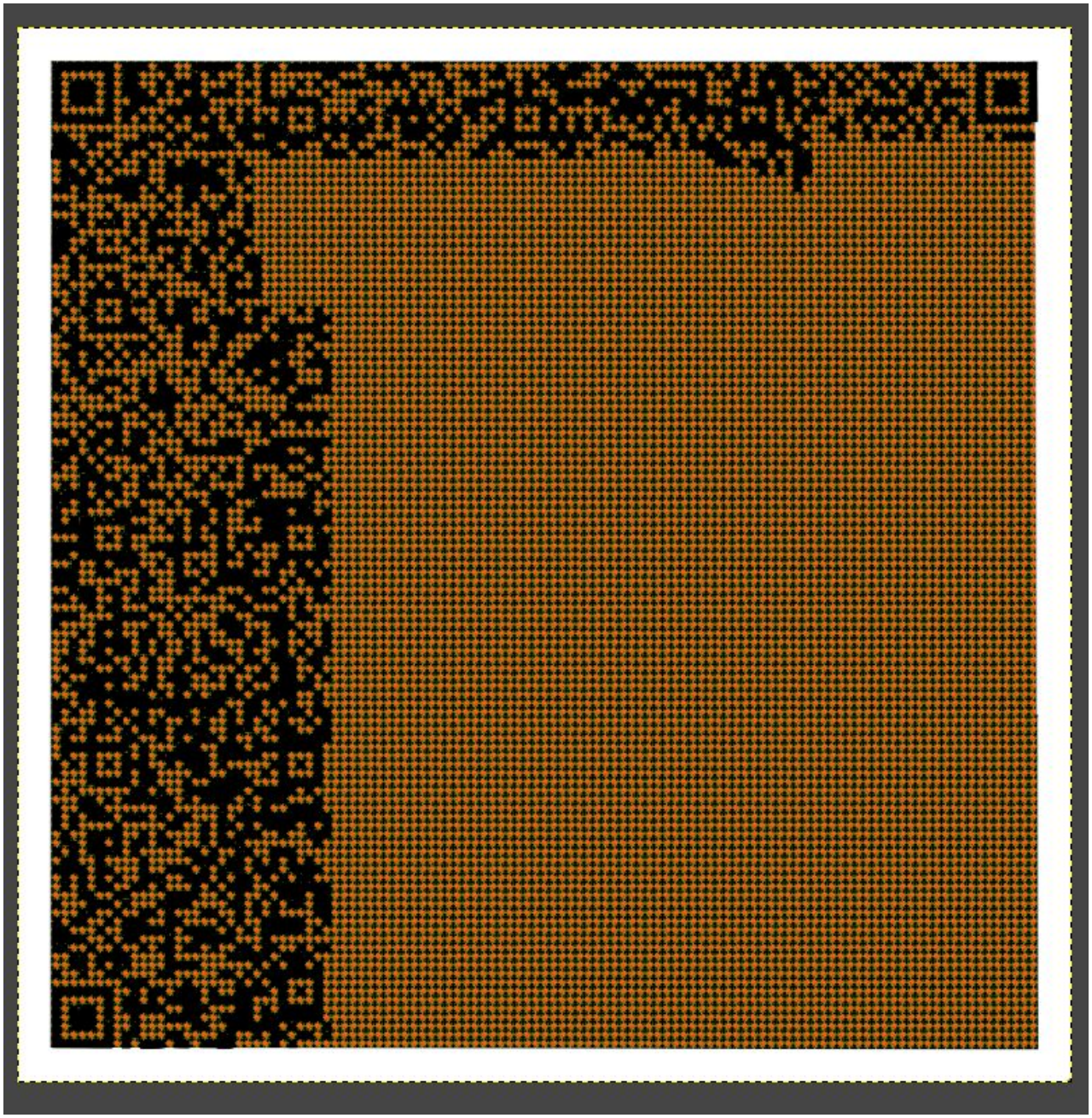
QR Code

Some strawberries are facing left and some facing right. Weird…. Zoom out a little. I decided to color the strawberries facing right since the left strawberries are framing the outside. Looks like a QR code.



Bad idea

So, I thought at first that I'd just bruteforce it by coloring the strawberries by hand. I mean how many can there be right? OK, so not all ideas pan out. Geez. This took me a while and I thought what if I make a mistake.



Coding time

Time to do this through code. I surprised myself with this one. I finished the code in about 20 mins.

First, an explanation of what I'm doing with the code and then the code to follow.

So, first I need to figure out how big the strawberries are to make sure that I can detect the correct location in the image for determining left from right strawberries. The picture below shows some of that rough work. Through this I was able to determine that they are 50X50 pixels in size. I also need to know which strawberry I'm looking at left vs. right. So, I'm counting over to the top left most red pixel to get the color and decide if it's red or black. Based on that count we're looking at x=16 and y=10 (counting up while going down on the y axis).

Steps:

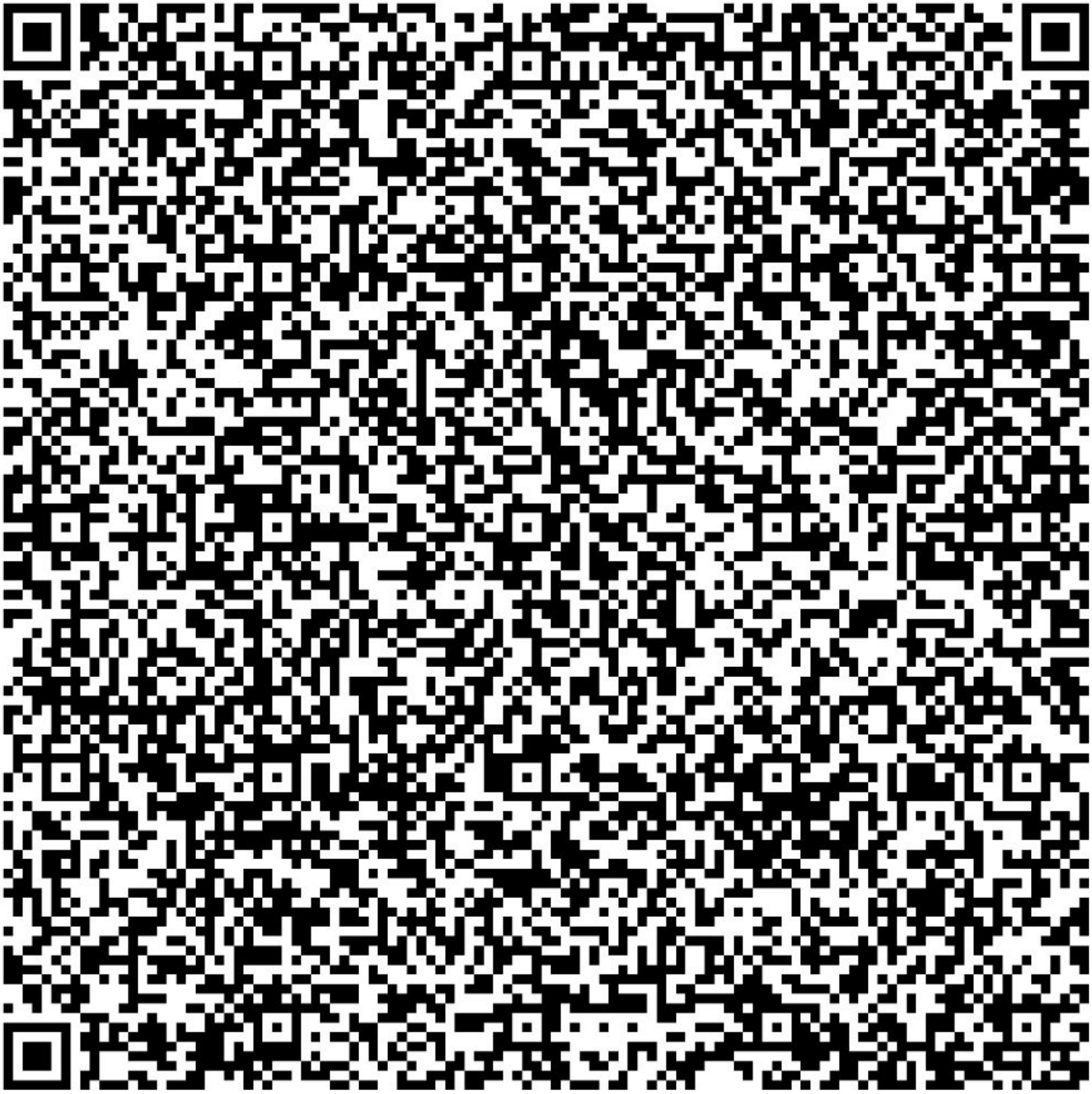
1. Look at that pixel on every 50X50 square and determine red or black.
2. Set new values for the entire 50X50 square of that strawberry to either all black or all white.

I'm able to do this by running through a loop on each axis and stepping by 50 for each. That will land me on the next strawberry either down or to the right. While, we're there, call a function and iterate over another double loop and fill in the new color. We just send over the current location and desired color. The function counts left 16 and up 10 to get to the top leftmost pixel and then fills in the color of each pixel in the square.

Code:

```
from PIL import Image

im = Image.open('matryoshka.png') # Can be many different formats.
pix = im.load()

var_dimensions = im.size # Get the width and height of the image for iterating over


def colorblock(var_x, var_y, var_newcolor):

  if var_newcolor == "Black":
     var_colorTuple = (0, 0, 0)
  else:
     var_colorTuple = (255, 255, 255)

  for q in range(var_x-16, var_x+34):
     for z in range(var_y-10, var_y+40):
        pix[q, z] = var_colorTuple


for x in range(16, var_dimensions[0], 50):
  for y in range(10, var_dimensions[1], 50):

     var_color = pix[x, y] # Get the RGBA Value of a pixel of an image

     if var_color[0] > 20:
        colorblock(x, y, "white")
     else:
        colorblock(x, y, "Black")

im.save('qr.png')  # Save the modified pixels as .png
```

The result:



I spent a while on this one. It would not scan with my phone. I downloaded a couple more QR Scanner apps and nothing. I double checked my work and it was correct, like scary perfect correct. So, I thought maybe it's my monitor or something. So tried uploading to an online QR scanner and continued to get errors. Until… I came across this site: https://online-barcode-reader.inliteresearch.com/

Glorious, Superior, The Bomb QR Scanner

Free Online Barcode Reader

To get such results using ClearImage SDK use TBR Code 103.

If your **business** application needs barcode recognition capabilities,
email your technical questions to support@inliteresearch.com
email your sales inquiries to sales@inliteresearch.com

File: qr.png New File
Pages: 1 **Barcodes:** 1

Barcode: 1 of 1 **Type:** QR Page 1 of 1
Length: 1165 **Rotation:** none
Module: 50.0pix **Rectangle:** {X=225,Y=225,Width=5597,Height=5597}
Barcode Text processing:
Converted Character Set: ISO-8859-1
Formatted: specialChar

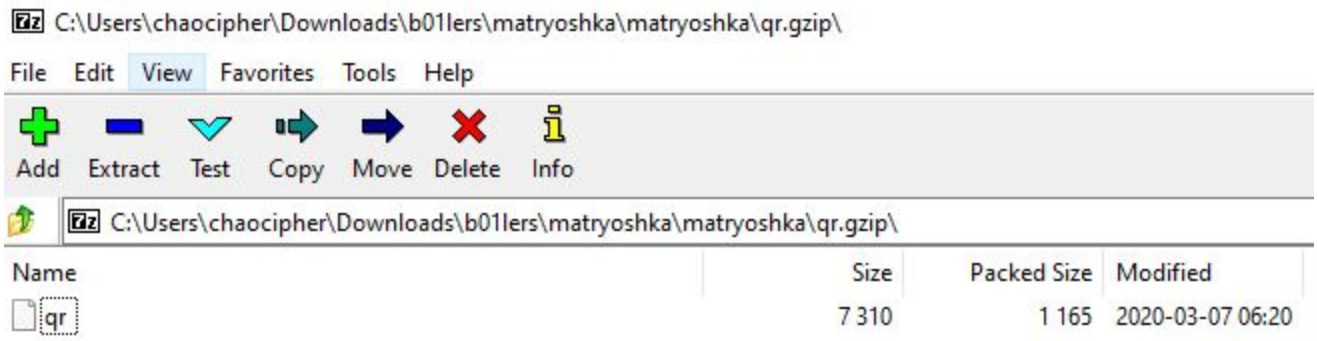
Binary Data in barcode (Hex-ASCII display)

```
0000  1f 8b 08 00 86 9f 63 5e  00 03 ed 99 4b 6e 1b 41  | ~~~~~~c^~~~~Kn~A |
0010  0c 44 f7 ba 0d ef c5 fb  af 13 60 9a af 1e 7b 94  | ~D~~~~~~~~`~~~{~ |
0020  55 92 45 80 c8 b0 2d 8f  7a c8 62 7d a8 d8 e9 fe  | U~E~~~-~z~b}~~~~ |
0030  0b 8f cf df 28 fa ef 55  ad e7 d1 7c fe fc e8 e7  | ~~~~(~~U~~~|~~~~ |
0040  eb 73 a9 5b af 9d 57 9e  6f e7 8a ab 50 b5 9b 2a  | ~s~[~~W~o~~~P~~* |
0050  fd bc c2 1d 35 d5 7b be  9d 3b d2 a5 82 e4 d4 3a  | ~~~~5~{~~;~~~~~: |
0060  55 cf f5 01 f6 94 04 c8  29 39 d7 e9 fc 1c 2f 01  | U~~~~~~~)9~~~~/~ |
0070  af ef 55 0f 24 8f 33 97  0e 90 f9 a4 dd 73 ec d7  | ~~U~$~3~~~~~~s~~ |
0080  55 33 58 85 a7 e9 d0 73  a4 66 e2 5a e0 bf 54 0d  | U3X~~~~s~f~Z~~T~ |
0090  57 53 ed 19 f0 4d f4 d1  6b 98 14 dd 2f 5e 8b fe  | WS~~~M~~k~~~/^~~ |
00a0  bf f3 71 79 40 8f 94 0f  d6 a3 56 49 c0 a8 3f 54  | ~~qy@~~~~~VI~~?T |
00b0  7c f1 6b 6f a4 21 bf 52  4a 34 c5 db a8 18 29 84  | |~ko~!~RJ4~~~~)~ |
00c0  b5 86 f7 cb 44 7b bc 4e  3d e8 2c 1f 9d de a8 85  | ~~~~D{~N=~,~~~~~ |
00d0  c6 e3 03 20 72 7d 70 a3  ca f2 0a 91 b3 5a f6 f8  | ~~~ r}p~~~~~~Z~~ |
00e0  f2 60 6e 8e ef 95 17 f5  0e 7f c3 6b b7 4f e2 a2  | ~`n~~~~~~~~k~O~~ |
00f0  54 d6 12 f0 10 5c c9 98  56 ab 3d c1 f8 0e 5b c7  | T~~~~\~~V~=~~~[~ |
0100  aa 94 1e 31 58 08 39 7c  d4 ca 5c 09 78 a7 d1 4a  | ~~~1X~9|~~\~x~~J |
0110  c3 4b 3f b9 e3 c6 9a 2b  ed 73 6b 9f 2d 47 44 d7  | ~K?~~~~+~sk~-GD~ |
0120  fb 65 39 cb 7e cd d6 1b  77 66 c0 82 99 bc d8 11  | ~e9~~~~~wf~~~~~~ |
```

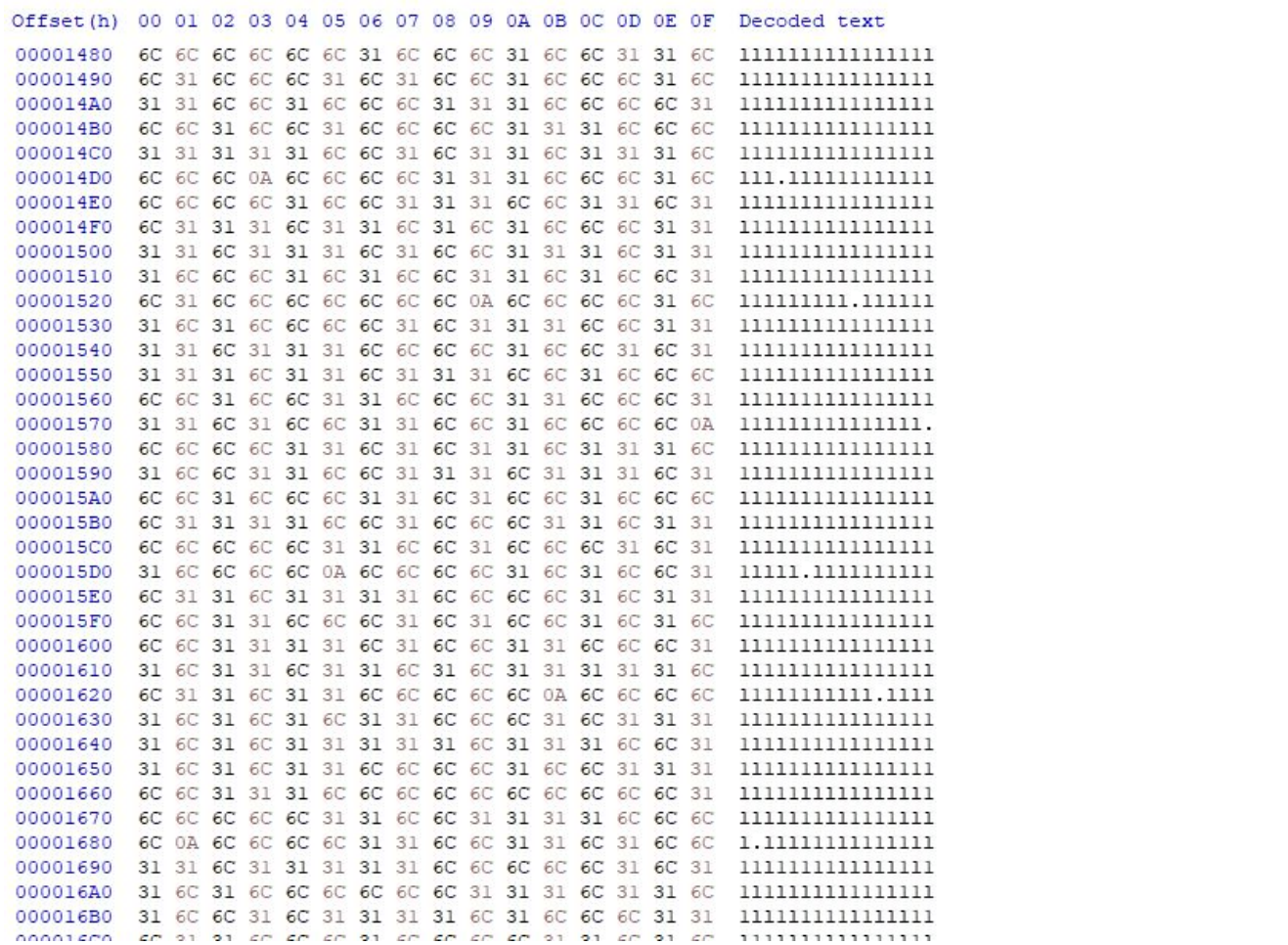
So, HEX dump. Yea!! Check out the header bytes: **1f 8b 08**

GZIP

That's right for the nerds in the audience, that's a gzip magic header. Grab the bytes in the middle and save out to a file using a hex editor.



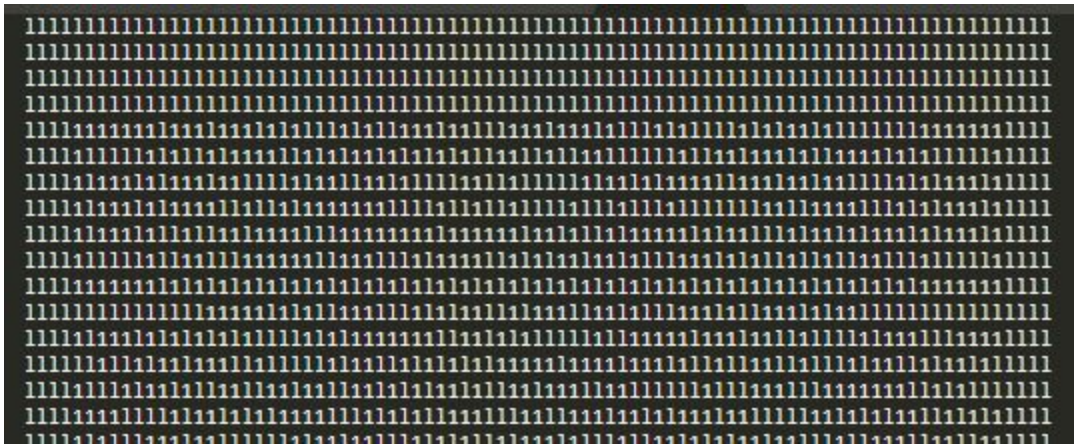
OK. One file. Let's see what it looks like in a hex editor.



Offset(h)	00	01	02	03	04	05	06	07	08	09	0A	0B	0C	0D	0E	0F	Decoded text
00001480	6C	6C	6C	6C	6C	6C	31	6C	6C	6C	31	6C	6C	31	31	6C	1111111111111111
00001490	6C	31	6C	6C	6C	31	6C	31	6C	6C	31	6C	6C	6C	31	6C	1111111111111111
000014A0	31	31	6C	6C	31	6C	6C	6C	31	31	31	6C	6C	6C	6C	31	1111111111111111
000014B0	6C	6C	31	6C	6C	31	6C	6C	6C	6C	31	31	31	6C	6C	6C	1111111111111111
000014C0	31	31	31	31	31	6C	6C	31	6C	31	31	6C	31	31	31	6C	1111111111111111
000014D0	6C	6C	6C	0A	6C	6C	6C	6C	31	31	31	6C	6C	6C	31	6C	111.111111111111
000014E0	6C	6C	6C	6C	31	6C	6C	31	31	31	6C	6C	31	31	6C	31	1111111111111111
000014F0	6C	31	31	31	6C	31	31	6C	31	6C	31	6C	6C	6C	31	31	1111111111111111
00001500	31	31	6C	31	31	31	6C	31	6C	6C	31	31	31	6C	31	31	1111111111111111
00001510	31	6C	6C	6C	31	6C	31	6C	6C	31	31	6C	31	6C	6C	31	1111111111111111
00001520	6C	31	6C	6C	6C	6C	6C	6C	6C	0A	6C	6C	6C	6C	31	6C	111111111.111111
00001530	31	6C	31	6C	6C	6C	6C	31	6C	31	31	31	6C	6C	31	31	1111111111111111
00001540	31	31	6C	31	31	31	6C	6C	6C	6C	31	6C	6C	31	6C	31	1111111111111111
00001550	31	31	31	6C	31	31	6C	31	31	31	6C	6C	31	6C	6C	6C	1111111111111111
00001560	6C	6C	31	6C	6C	31	31	6C	6C	6C	31	31	6C	6C	6C	31	1111111111111111
00001570	31	31	6C	31	6C	6C	31	31	6C	6C	31	6C	6C	6C	6C	0A	111111111111111.
00001580	6C	6C	6C	6C	31	31	6C	31	6C	31	31	6C	31	31	31	6C	1111111111111111
00001590	31	6C	6C	31	31	6C	6C	31	31	31	6C	31	31	31	6C	31	1111111111111111
000015A0	6C	6C	31	6C	6C	6C	31	31	6C	31	6C	6C	31	6C	6C	6C	1111111111111111
000015B0	6C	31	31	31	31	6C	6C	31	6C	6C	6C	31	31	6C	31	31	1111111111111111
000015C0	6C	6C	6C	6C	6C	31	31	6C	6C	31	6C	6C	6C	31	6C	31	1111111111111111
000015D0	31	6C	6C	6C	6C	0A	6C	6C	6C	6C	31	6C	31	6C	6C	31	11111.1111111111
000015E0	6C	31	31	6C	31	31	31	31	6C	6C	6C	6C	31	6C	31	31	1111111111111111
000015F0	6C	6C	31	31	6C	6C	6C	31	6C	31	6C	6C	31	6C	31	6C	1111111111111111
00001600	6C	6C	31	31	31	31	6C	31	6C	6C	31	31	6C	6C	6C	31	1111111111111111
00001610	31	6C	31	31	6C	31	31	6C	31	6C	31	31	31	31	31	6C	1111111111111111
00001620	6C	31	31	6C	31	31	6C	6C	6C	6C	6C	0A	6C	6C	6C	6C	1111111111.1111
00001630	31	6C	31	6C	31	6C	31	31	6C	6C	6C	31	6C	31	31	31	1111111111111111
00001640	31	6C	31	6C	31	31	31	31	31	6C	31	31	31	6C	6C	31	1111111111111111
00001650	31	6C	31	6C	31	31	6C	6C	6C	6C	31	6C	6C	31	31	31	1111111111111111
00001660	6C	6C	31	31	31	6C	6C	6C	6C	6C	6C	6C	6C	6C	6C	31	1111111111111111
00001670	6C	6C	6C	6C	6C	31	31	6C	6C	31	31	31	31	6C	6C	6C	1111111111111111
00001680	6C	0A	6C	6C	6C	6C	31	31	6C	6C	31	31	6C	31	6C	6C	1.111111111111111
00001690	31	31	6C	31	31	31	31	31	6C	6C	6C	6C	6C	6C	31	31	1111111111111111
000016A0	31	6C	31	6C	6C	6C	6C	6C	6C	31	31	31	6C	31	31	6C	1111111111111111
000016B0	31	6C	6C	31	6C	31	31	31	31	6C	31	6C	6C	6C	31	31	1111111111111111

Fun little obfuscation. Off to the text editor.

1’s vs. l’s



Replace the l’s with spaces:

Awe, another QR code. This is in keeping with the theme right.

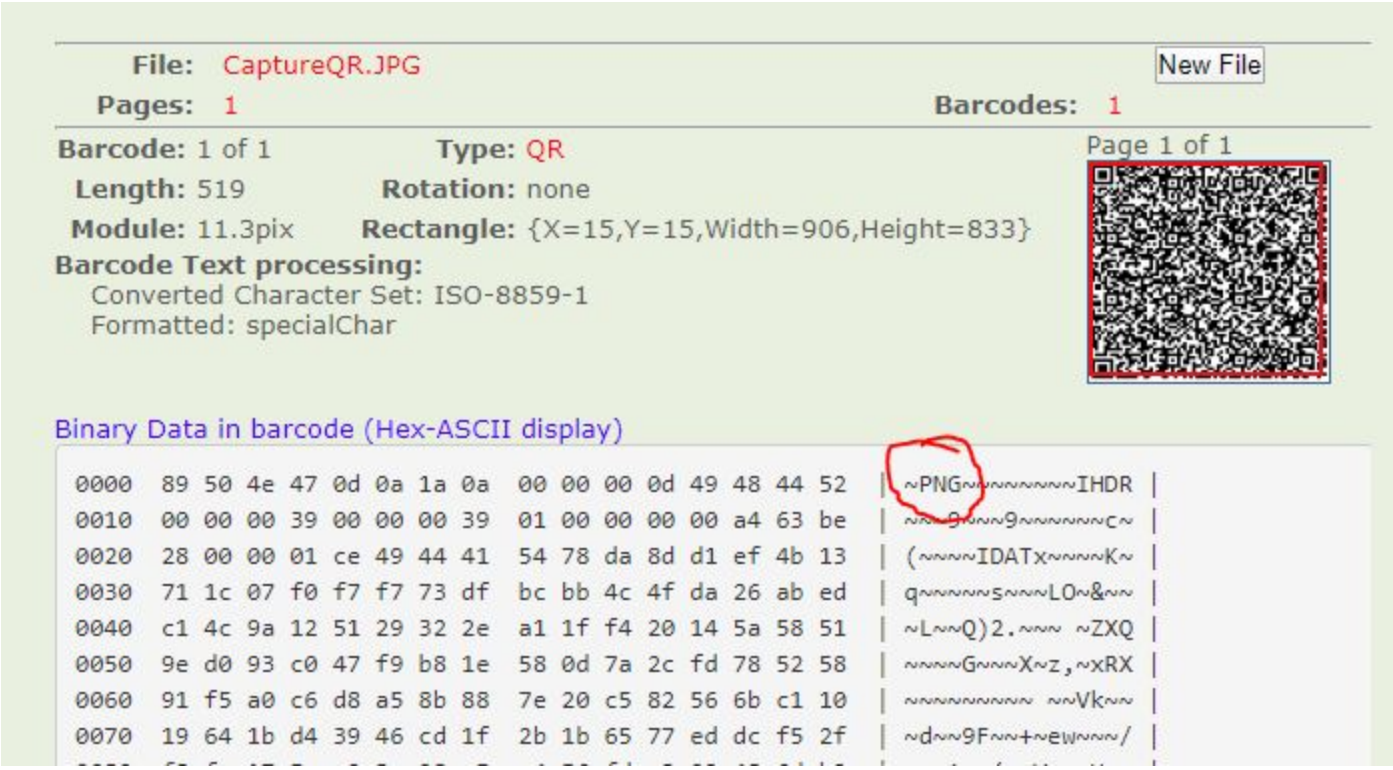
Add commas to make changes in a Google sheets:

Google sheets

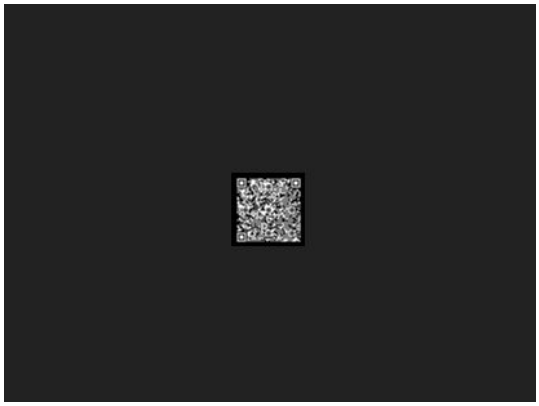
Text to columns and conditional formatting. Shrink the columns a bit to make everything more square.

PNG file

OK. Back to the same QR scanner:



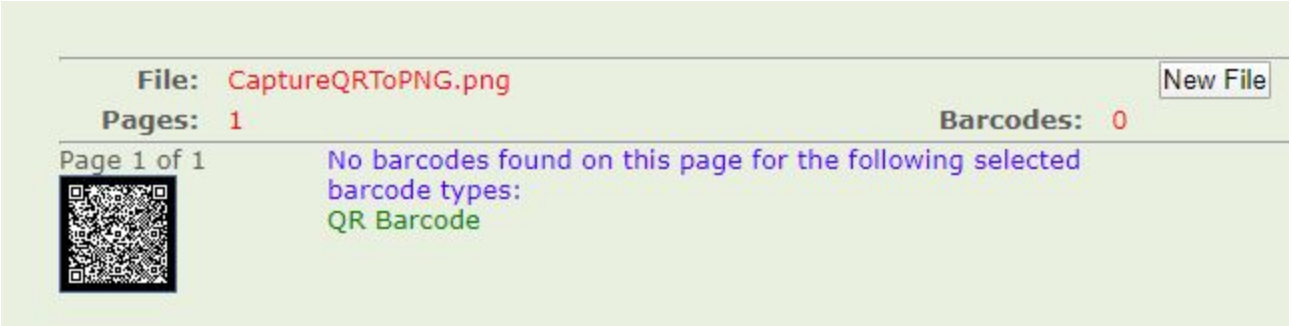
Boom - PNG file. That one's easy. Crack it open.



What the heck? Zoom in.

Well another QR code. Still in keeping with the theme.

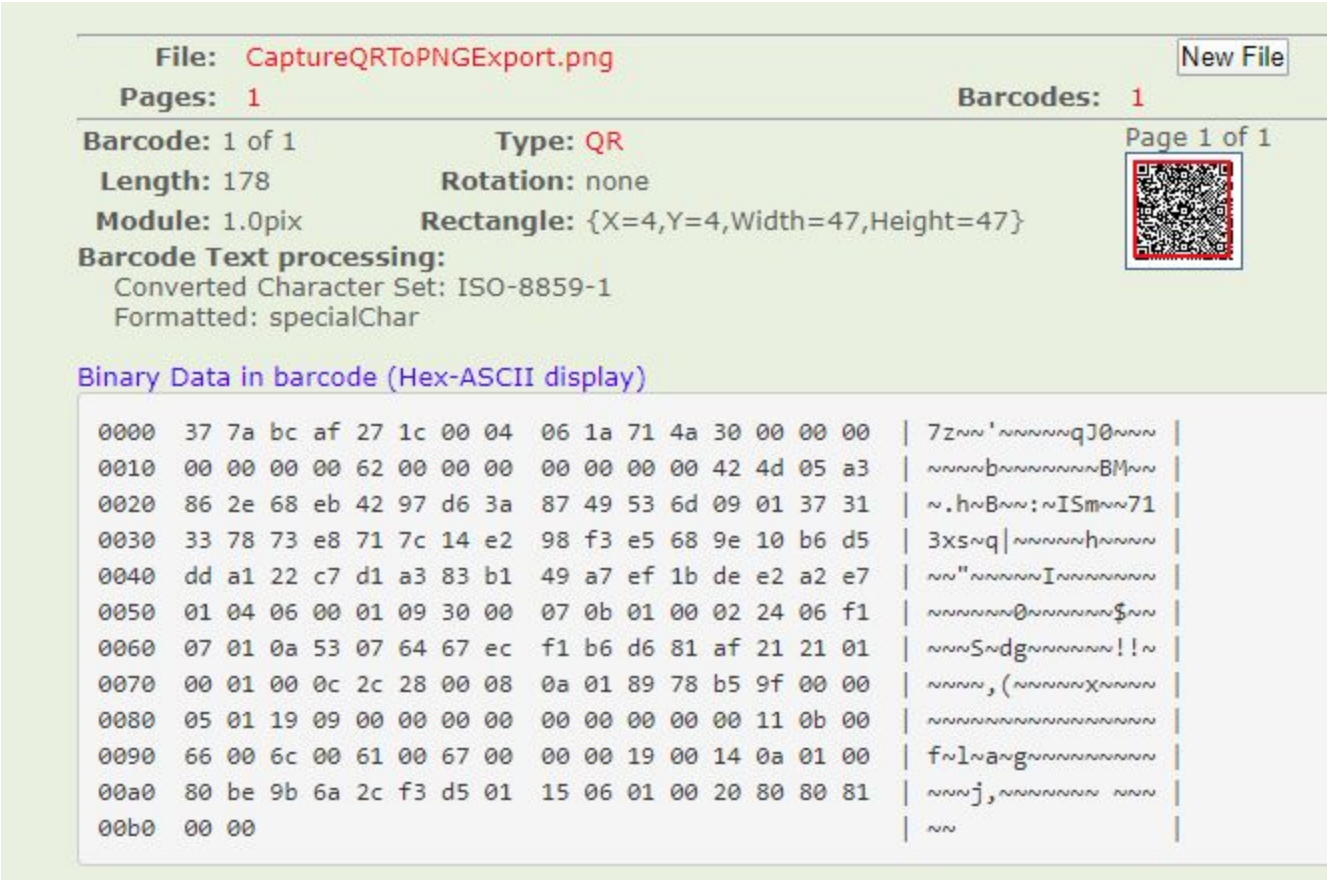
Upload it to the scanner and no dice.



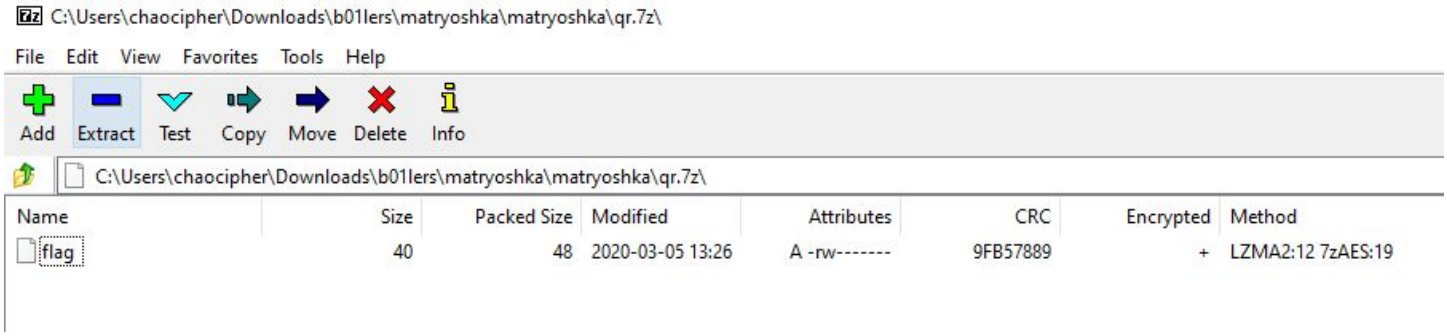
I tinkered with this for a bit and then it hit me. These are the wrong colors for a QR code. Need to invert the colors first.

7z file

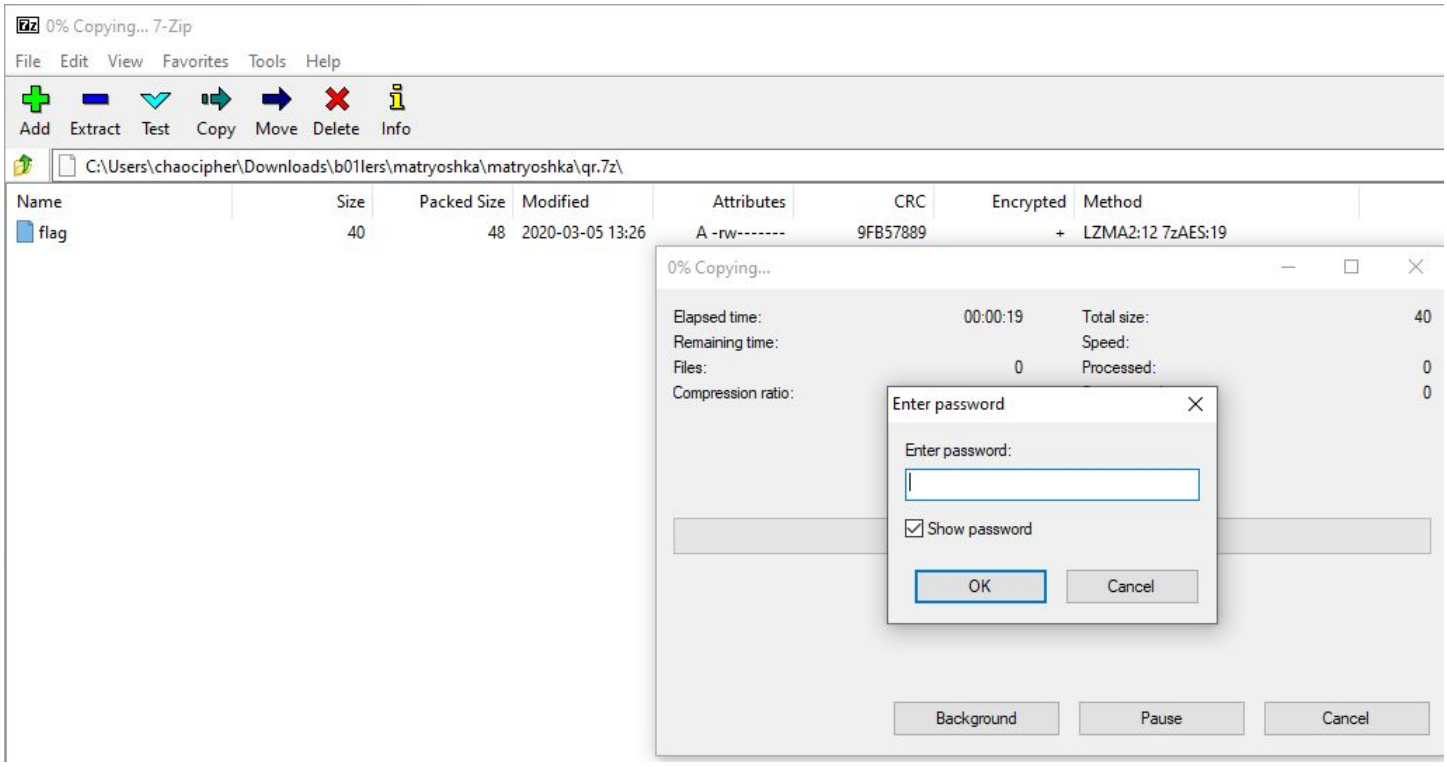
Upload again:



Nice! 7z file, cool. Same hex editor to file stuff.

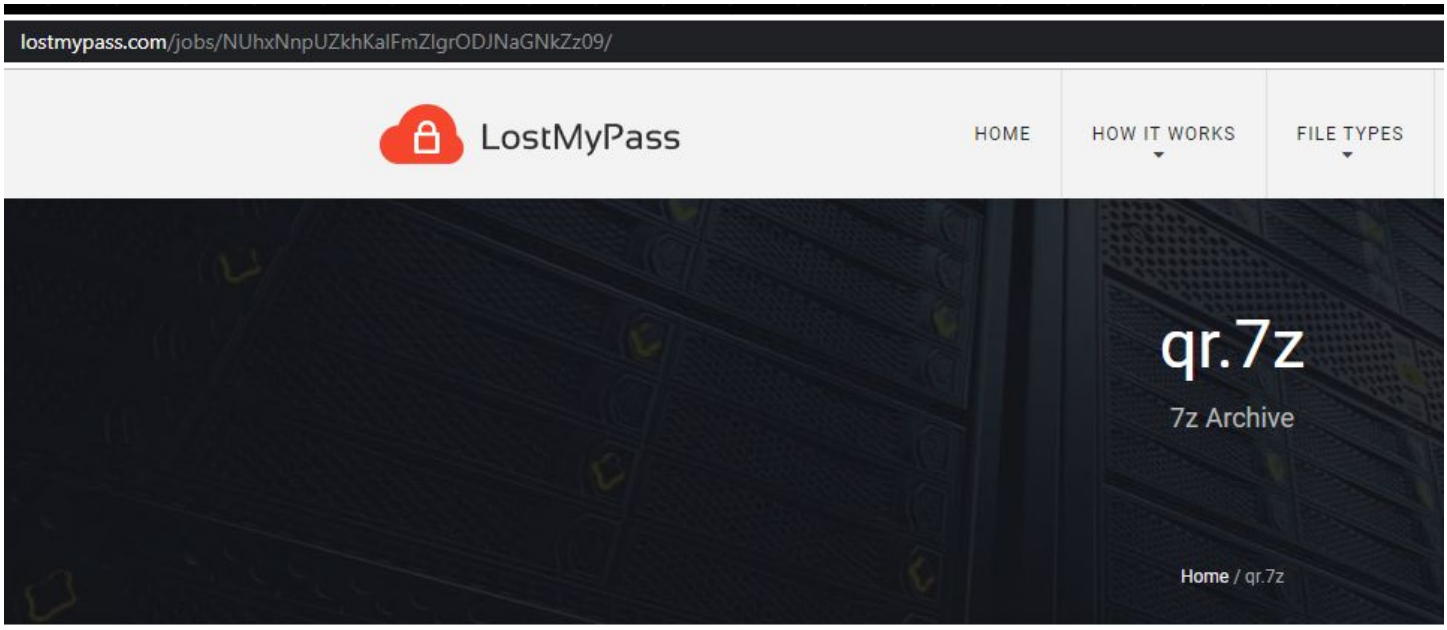


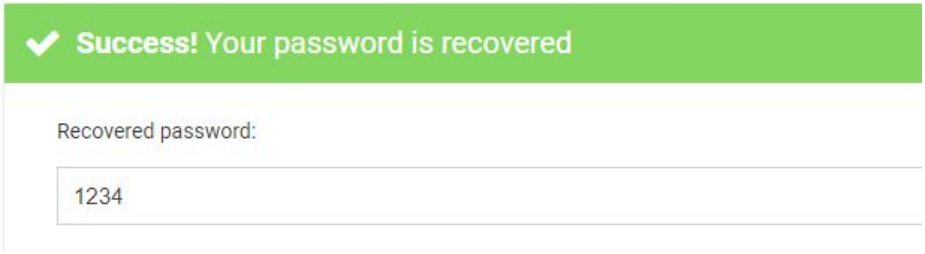
Flag!!! Gimmie….



This ladies and gentlemen, this where I hit the wall. I spent the next 4 or 5 hours trying to get my bad Kali image from OSCP to run an old Perl script that I found online. I gave up on that and found a python script out there, but it would fail because the headers were not what the script expected. It was brutal. I finally decided to look for some online tools that I could upload the file to. The first 4 or 5 gave me errors like the file was bad. Finally, one of them broke the password.

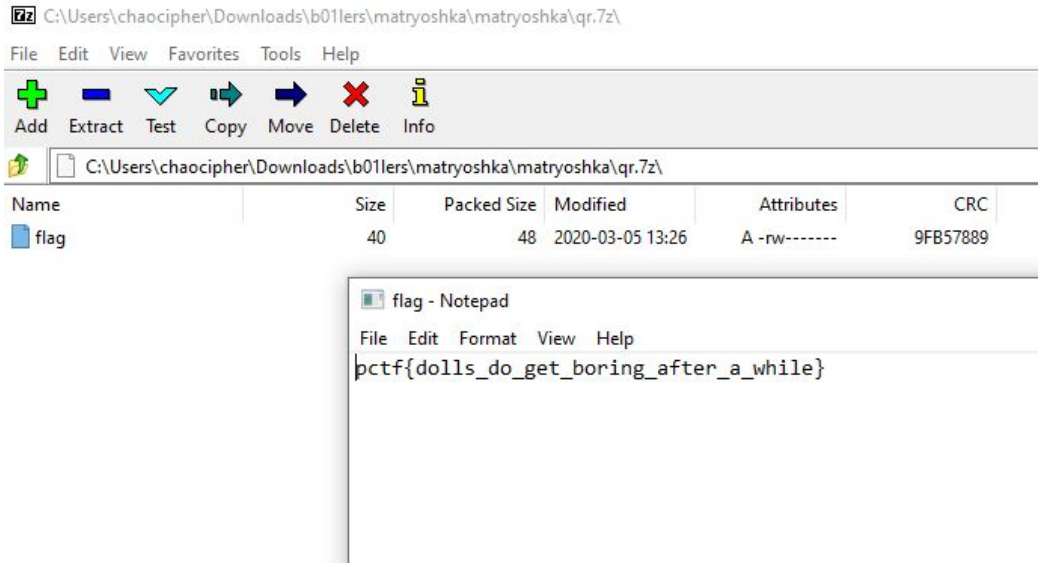
LostMyPass





Seriously?!

Flag


C:\Users\chaocipher\Downloads\b01lers\matryoshka\matryoshka\qr.7z\
File Edit View Favorites Tools Help
Add Extract Test Copy Move Delete Info
C:\Users\chaocipher\Downloads\b01lers\matryoshka\matryoshka\qr.7z\
Name Size Packed Size Modified Attributes CRC
flag 40 48 2020-03-05 13:26 A -rw------- 9FB57889
flag - Notepad
File Edit Format View Help
pctf{dolls_do_get_boring_after_a_while}

Reflection

So, looking back the theme would have helped me if I had really kept that fresh in my head..
So the theme:

Matryoshka doll

Matryoshka dolls, also known as Babushka dolls, stacking dolls, nesting dolls, Russian tea dolls, or Russian dolls, are a set of wooden dolls of decreasing size placed one inside another. The name matryoshka, literally "little matron", is a diminutive form of Russian female first name "Matryona" or "Matryosha". Wikipedia

And…do you remember the text on the challenge tile?

Russian - detected	English
super-secret password один два	super-secret password one two ... ...

Come ooooonnnnnnn!!!!… Looks so easy after the fact.

-chaocipher