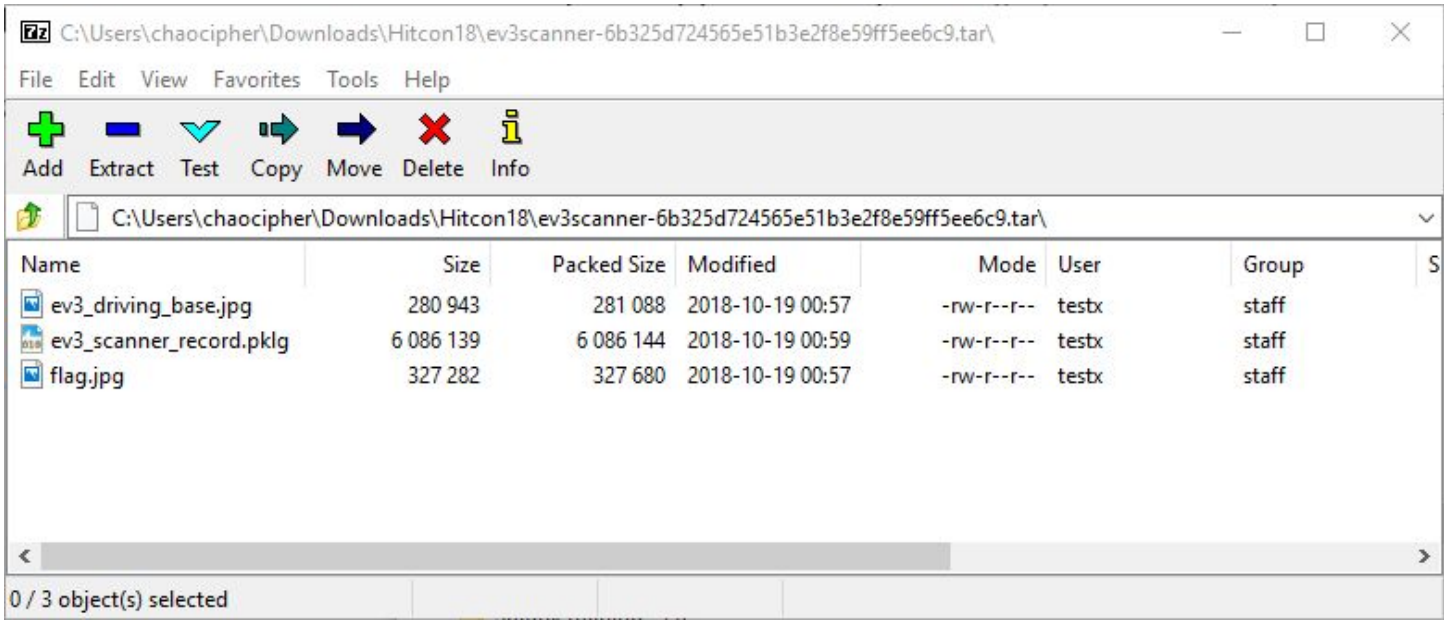
EV3 Scanner

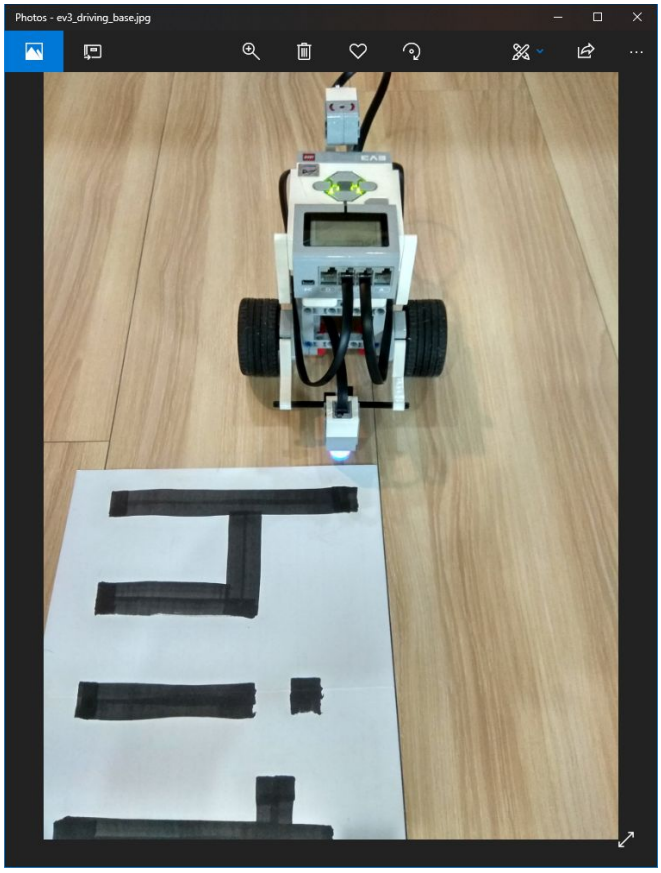
Finishing points = 180

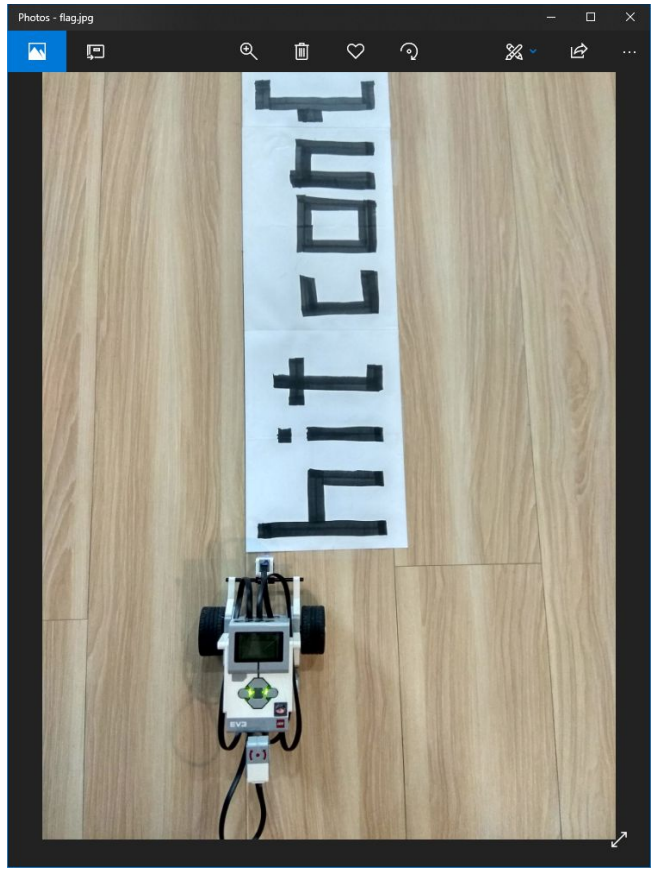
Find the flag.
[ev3scanner-6b325d724565e51b3e2f8e59ff5ee6c9.tar.gz]
Author: Jeffxx

Open the archive and you get three files:

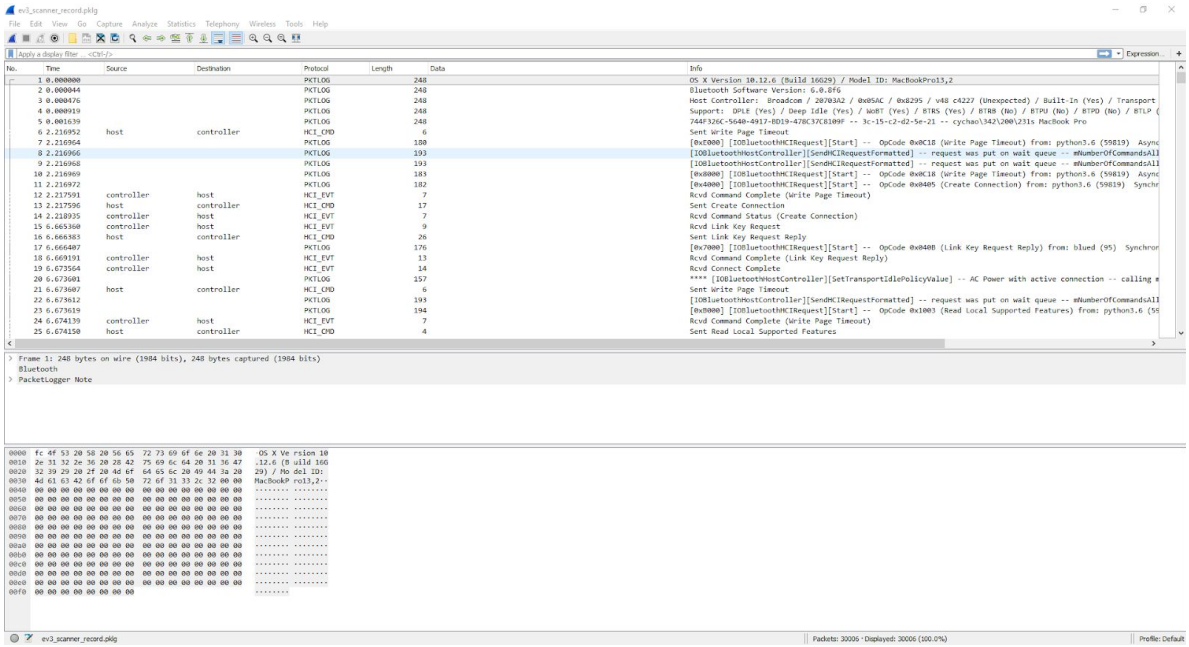


Let’s look at the two pictures real fast:

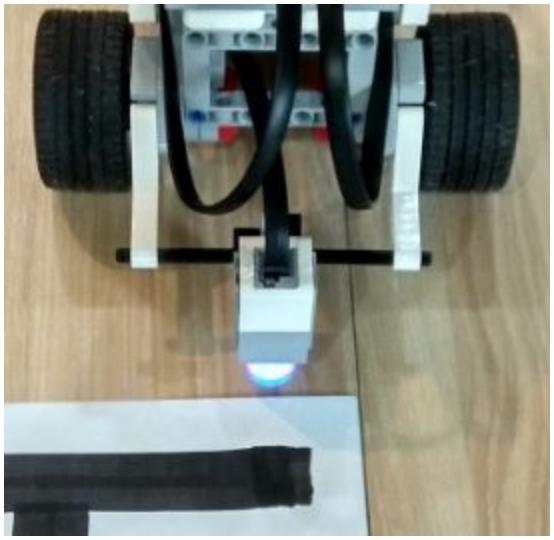




I didn’t know what to think about those yet. Let’s crack open the pklg file with Wireshark.



So, here's where I stopped to think about what I was actually doing and seeing. What's happening here is that the EV3 robot is being controlled by a computer using Bluetooth. So, the packet capture is grabbing that communication to see what the robot is doing and what it's communicating back.



So, what communication do I need to look at specifically? Well, let's take a look at the robot again.
That's a color sensor on the bottom not a marker. So, we want to know what the robot sees. That will be in the communication back to the computer.

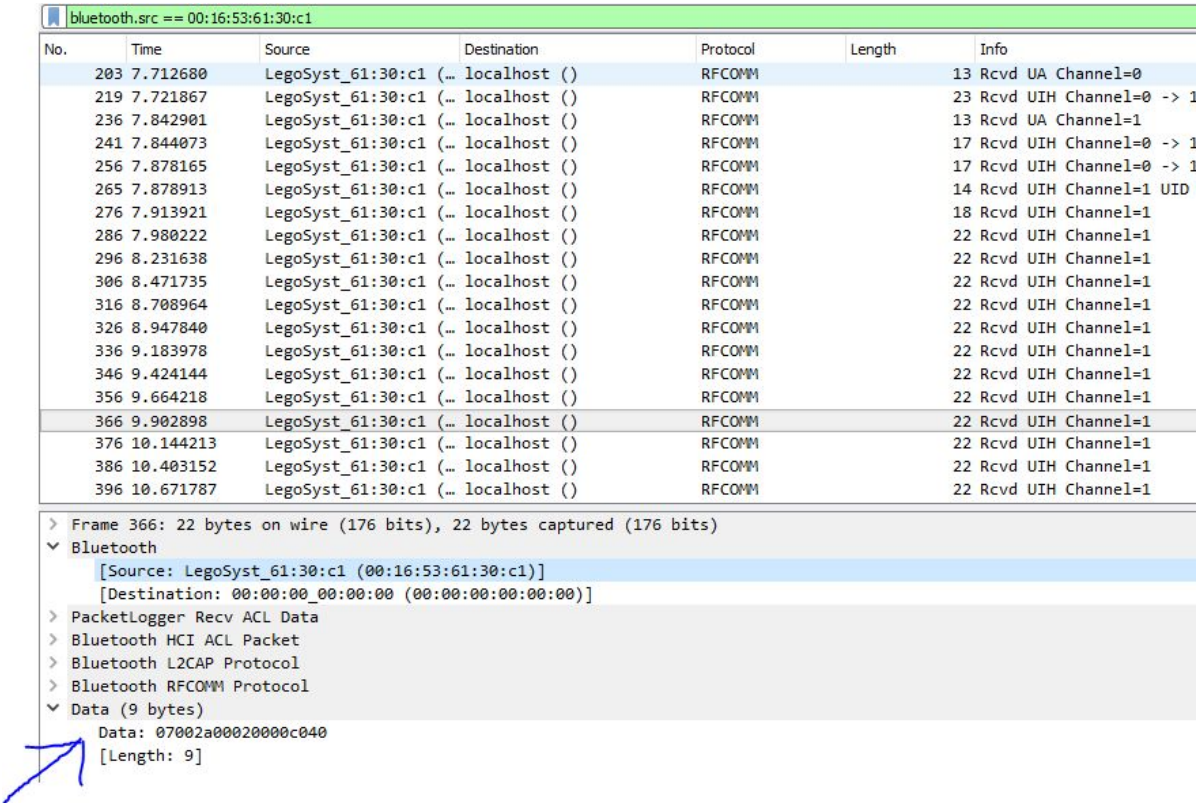
So, let's filter Wireshark down to that.

bluetooth.src == 00:16:53:61:30:c1

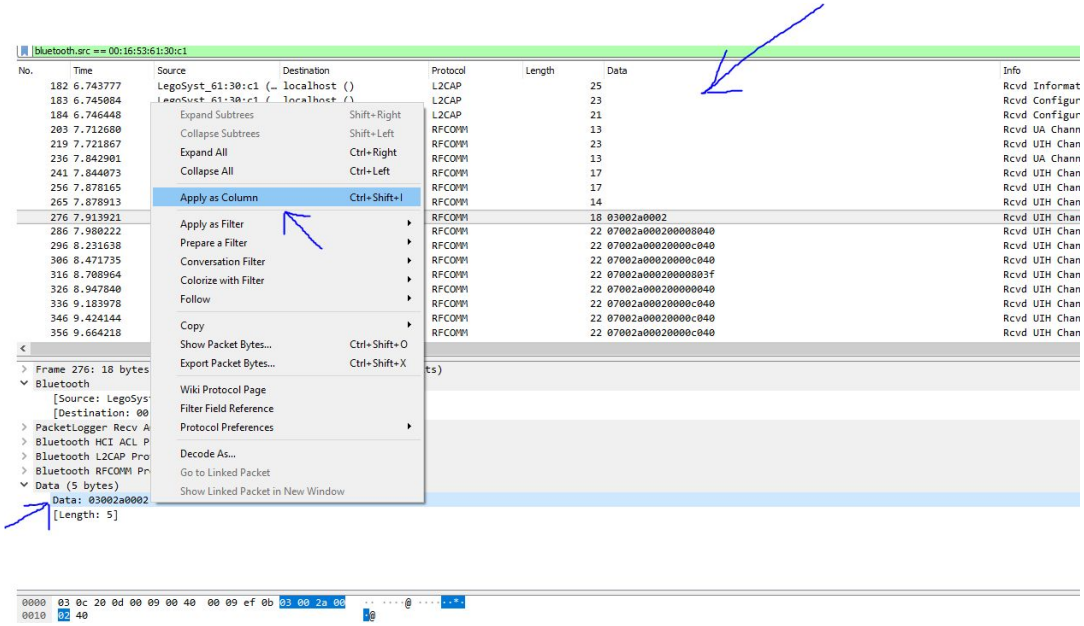
No.	Time	Source	Destination	Protocol	Length	Info
151	6.716440	LegoSyst_61:30:c1 (…	localhost ()	L2CAP	21	Rcvd Information Response (Extended Features Mask, Success)
171	6.721381	LegoSyst_61:30:c1 (…	localhost ()	L2CAP	21	Rcvd Connection Response - Success (SCID: 0x0040, DCID: 0x004
181	6.741301	LegoSyst_61:30:c1 (…	localhost ()	L2CAP	25	Rcvd Information Response (Fixed Channels Supported, Success)
182	6.743777	LegoSyst_61:30:c1 (…	localhost ()	L2CAP	25	Rcvd Information Response (Fixed Channels Supported, Success)
183	6.745084	LegoSyst_61:30:c1 (…	localhost ()	L2CAP	23	Rcvd Configure Response - Success (SCID: 0x0040)
184	6.746448	LegoSyst_61:30:c1 (…	localhost ()	L2CAP	21	Rcvd Configure Request (DCID: 0x0040)
203	7.712680	LegoSyst_61:30:c1 (…	localhost ()	RFCOMM	13	Rcvd UA Channel=0
219	7.721867	LegoSyst_61:30:c1 (…	localhost ()	RFCOMM	23	Rcvd UIH Channel=0 -> 1 MPX_CTRL DLC Parameter Negotiation (PN
236	7.842901	LegoSyst_61:30:c1 (…	localhost ()	RFCOMM	13	Rcvd UA Channel=1
241	7.844073	LegoSyst_61:30:c1 (…	localhost ()	RFCOMM	17	Rcvd UIH Channel=0 -> 1 MPX_CTRL Modem Status Command (MSC)
256	7.878165	LegoSyst_61:30:c1 (…	localhost ()	RFCOMM	17	Rcvd UIH Channel=0 -> 1 MPX_CTRL Modem Status Command (MSC)
265	7.878913	LegoSyst_61:30:c1 (…	localhost ()	RFCOMM	14	Rcvd UIH Channel=1 UID
276	7.913921	LegoSyst_61:30:c1 (…	localhost ()	RFCOMM	18	Rcvd UIH Channel=1
286	7.980222	LegoSyst_61:30:c1 (…	localhost ()	RFCOMM	22	Rcvd UIH Channel=1
296	8.231638	LegoSyst_61:30:c1 (…	localhost ()	RFCOMM	22	Rcvd UIH Channel=1
306	8.471735	LegoSyst_61:30:c1 (…	localhost ()	RFCOMM	22	Rcvd UIH Channel=1
316	8.708964	LegoSyst_61:30:c1 (…	localhost ()	RFCOMM	22	Rcvd UIH Channel=1
326	8.947840	LegoSyst_61:30:c1 (…	localhost ()	RFCOMM	22	Rcvd UIH Channel=1
336	9.183978	LegoSyst_61:30:c1 (…	localhost ()	RFCOMM	22	Rcvd UIH Channel=1
346	9.424144	LegoSyst_61:30:c1 (…	localhost ()	RFCOMM	22	Rcvd UIH Channel=1

OK, from here we need to look for data.

Bingo.



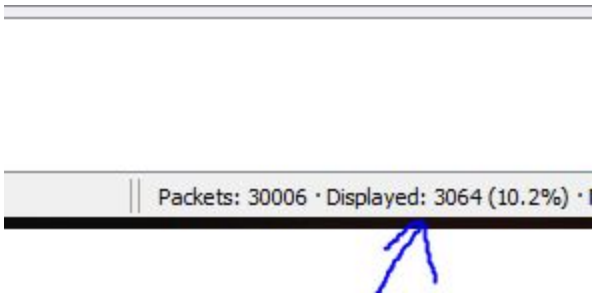
OK, let's set that as a column in our view. Right click on the data itself, click "Apply as Column" and you can see it in the top panel now.



OK, now what. Well, based on the pictures that we were given it's clear that white should be the most given symbol and anything else is likely the black lines. Well, the first number 03002a0002 is maybe the wood color to start, but it's clear that numbers ending in 40 will be white and 3f will be black.

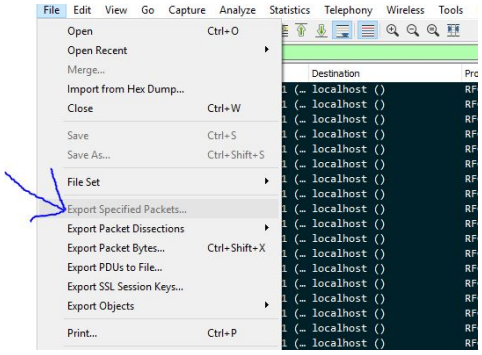
```
18 03002a0002
22 07002a000200008040
22 07002a00020000c040
22 07002a00020000c040
22 07002a00020000803f
22 07002a000200000040
22 07002a00020000c040
22 07002a00020000c040
22 07002a00020000c040
22 07002a00020000c040
22 07002a00020000c040
22 07002a00020000c040
22 07002a00020000c040
22 07002a00020000c040
22 07002a00020000c040
22 07002a00020000c040
22 07002a00020000c040
22 07002a00020000c040
22 07002a00020000c040
22 07002a00020000c040
22 07002a00020000c040
22 07002a00020000c040
22 07002a00020000c040
22 07002a00020000c040
22 07002a00020000c040
22 07002a00020000c040
22 07002a00020000c040
22 07002a00020000c040
22 07002a00020000c040
14
22 07002a00020000c040
22 07002a00020000c040
22 07002a00020000c040
22 07002a00020000c040
22 07002a00020000c040
22 07002a00020000c040
22 07002a00020000c040
22 07002a00020000c040
```

OK, now to plot this out. How many packets are there?



Ouch.

It was at this point that I tried to use Python to read the packets and plot out the grid. Well, I ran into quite a few problems. The pypcap module would not read the pklg file at all. I tried exporting all the packets, but that option was removed from Wireshark. See the screenshot below:



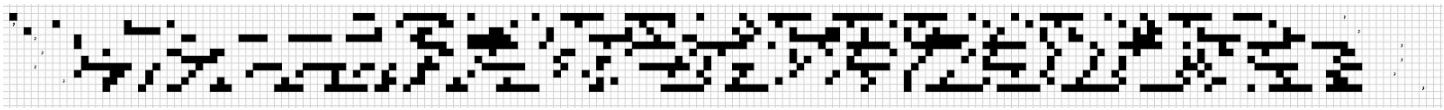
OK, sometimes for the CTF you can just bruteforce it by hand rather than spending hours trying to get a round peg in a square hole. So, off to Excel I went.

So, time for the guessing. I guessed that this message would be laid out left to right and then on the next pass would go back from right to left. This is the method I used until I ran out of packets.

At the end of each row is a bunch of other numbers that I just figured was the robot getting turned around for its next pass so I just wrote a question mark for each of those to mark the end of a row.

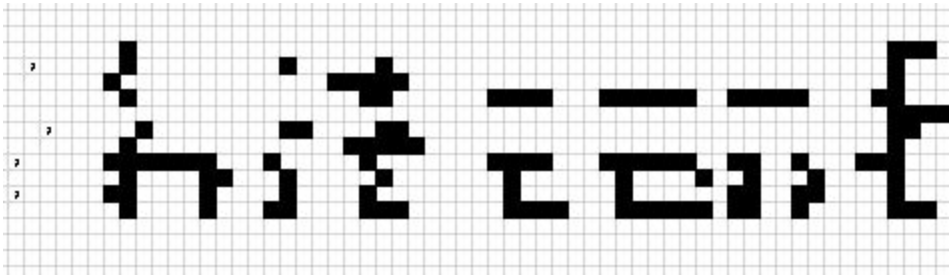
```
22 07002a00020000803f
22 07002a00020000803f
22 07002a00020000803f
14
22 07002a000200000040
22 07002a00020000c040
22 07002a00020000c040
22 07002a00020000c040
22 07002a00020000c040
22 07002a00020000c040
22 07002a00020000c040
22 07002a00020000c040
22 07002a00020000c040
22 07002a00020000c040
22 07002a00020000c040
22 07002a00020000c040
22 07002a000200d00345
18 03002a0002
22 07002a000200d00345
22 07002a000200e00345
22 07002a000200f00345
22 07002a000200100445
22 07002a000200400445
22 07002a000200700445
22 07002a000200c00445
22 07002a000200f00445
22 07002a000200400545
22 07002a000200700545
22 07002a000200b00545
22 07002a000200f00545
22 07002a000200300645
22 07002a000200700645
22 07002a000200b00645
22 07002a000200000745
```

This is what I ended up with.

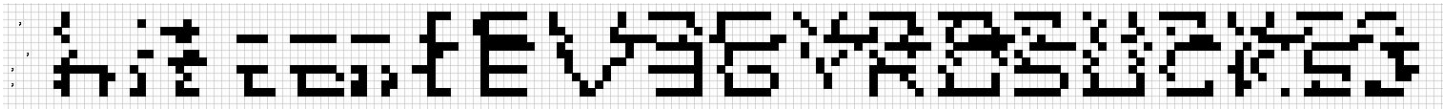


At this point I thought that the “hitcon{“ might be there but I wasn’t sure. None of the question marks lined up and I certainly didn’t see hitcon on the left. So, I started tinkering which each row moving them left or right to see how things lined up. I actually wrote a couple macros to make that process faster.

After some tinkering I ended up with this on the left:



That’s cool so I knew I had something, but not sure if I needed to move rows up or down too, but I just zoomed out a bit to take a look at it and I saw the flag:



hitcon{EV3GYROSUCKS}

That’s some nerd humor I can get behind.
-Chaocipher