- We were given those steps from the challenge

```
>>> import base64
>>> msg = msg + " "+ base64.b64encode(data)
>>> y = []
>>> for i in range(0,len(msg)):
y.append(ord(msg[i]) + i)
>>> y = y[::-1]
>>> y
```

[249, 264, 274, 285, 287, 256, 233, 279, 289, 245, 245, 277, 288, 241, 241, 273, 280, 271, 241, 242, 217, 224, 253, 266, 267, 215, 249, 250, 266, 225, 279, 247, 278, 204, 204, 241, 200, 237, 259, 248, 254, 233, 196, 244, 247, 242, 251, 239, 252, 243, 188, 236, 248, 255, 243, 222, 181, 197, 202, 202, 250, 235, 197, 198, 232, 201, 194, 220, 232, 239, 168, 206, 224, 181, 197, 214, 217, 177, 177, 199, 216, 189, 157, 204, 222, 207, 173, 174, 221, 185, 148, 194, 214, 169, 204, 174, 195, 172, 139, 187, 162, 153, 207, 174, 196, 203, 191, 170, 129, 145, 148, 150, 195, 149, 125, 171, 140, 137, 153, 151, 100, 164, 182, 162, 164, 95, 165, 1 71, 165, 178, 169, 165, 164, 166, 156, 85, 153, 155, 166, 81, 149, 165, 143, 149, 76, 144, 129, 73, 154, 140, 145, 136, 133, 139, 66, 130, 64, 132, 144, 126, 60, 144, 137, 1 46, 56, 139, 119, 125, 136, 51, 137, 128, 126, 122, 46, 86, 44, 57, 124, 110, 111, 117, 103, 119, 120, 86, 34, 106, 72]

- So lets start to go with the steps in reverse order

- First we are going to reverse the numbers again to get the original list

```
>>> y  = [249, 264, 274, 285, 287, 256, 233, 279, 289, 245, 245, 277, 288, 241, 241, 273, 280, 271, 241, 242, 217, 224, 253, 266, 267, 215, 249, 250, 266, 225, 279, 247,
278, 204, 204, 241, 200, 237, 259, 248, 254, 233, 196, 244, 247, 242, 251, 239, 252, 243, 188, 236, 248, 255, 243, 222, 181, 197, 202, 202, 250, 235, 197, 198, 232, 201,
194, 220, 232, 239, 168, 206, 224, 181, 197, 214, 217, 177, 177, 199, 216, 189, 157, 204, 222, 207, 173, 174, 221, 185, 148, 194, 214, 169, 204, 174, 195, 172, 139, 187,
162, 153, 207, 174, 196, 203, 191, 170, 129, 145, 148, 150, 195, 149, 125, 171, 140, 137, 153, 151, 100, 164, 182, 162, 164, 95, 165, 171, 165, 178, 169, 165, 164, 166,
156, 85, 153, 155, 166, 81, 149, 165, 143, 149, 76, 144, 129, 73, 154, 140, 145, 136, 133, 139, 66, 130, 64, 132, 144, 126, 6 0, 144, 137, 146, 56, 139, 119, 125, 136, 51,
137, 128, 126, 122, 46, 86, 44, 57, 124, 110, 111, 117, 103, 119, 120, 86, 34, 106, 72]

>>> y = y[::-1]
```

- Now the list looks like this

```
>>> y
```

[72, 106, 34, 86, 120, 119, 103, 117, 111, 110, 124, 57, 44, 86, 46, 122, 126, 128, 137, 51, 136, 125, 119, 139, 56, 146, 137 , 144, 60, 126, 144, 132, 64, 130, 66, 139, 133, 136, 145, 140, 154, 73, 129, 144, 76, 149, 143, 165, 149, 81, 166, 155, 153, 85, 156, 166, 164, 165, 169, 178, 165, 171, 165, 95, 164, 162, 182, 164, 100, 151, 153, 137, 140, 171, 125, 149, 195, 150, 148, 145, 129, 170, 191, 203, 196, 174, 207, 153, 162, 187, 139, 172, 195, 174, 204, 169, 214, 194, 148, 185, 221, 174, 173, 207, 222, 204, 157, 189, 216, 199, 177, 177, 217, 214, 197, 181, 224, 206, 168, 239, 232, 220, 194, 201, 232, 198, 197, 235, 250, 202, 202, 197, 181, 222, 243, 255, 248, 236, 188, 243, 252, 239, 251, 242, 247, 244, 196, 233, 254, 248, 259, 237, 200, 241, 204, 204, 278, 247, 279, 225, 266, 250, 249, 215, 267, 266, 253, 224, 217, 242, 241, 271, 280, 273, 241, 241, 288, 277, 245, 245, 289, 279, 233, 256, 287, 285, 274, 264, 249]

- Second we are going to get the ascii value for each number in the list

```
>>> for I in range(len(y)):
        y[i] = chr(y[i] - i)
```

- Here is the message looks like now

```
>>> y
```

['H', 'i', ' ', 'S', 't', 'r', 'a', 'n', 'g', 'e', 'r', '.', ' ', 'I', ' ', 'k', 'n', 'o', 'w', ' ', 't', 'h', 'a', 't', ' ', 'y', 'o', 'u', ' ', 'a', 'r', 'e', ' ', 'a', ' ', 'h', 'a', 'c', 'k', 'e', 'r', ' ', 'W', 'e', ' ', 'h', 'a', 'v', 'e', ' ', 't', 'h', 'e', ' ', 'f', 'o', 'l', 'l', 'o', 'w', 'i', 'n', 'g', ' ', 'd', 'a', 't', 'a', ' ', 'R', 'S', 'B', 'D', 'b', '3', 'J', 'w', 'I', 'F', 'B', '1', 'Y', 'm', 'x', 'p', 'Y', 'y', 'B', 'J', 'b', '1', 'Q', 'g', 'Q', 'n', 'J', 'v', 'a', '2', 'V', 'y', 'I', 'G', 'h', 'v', 'c', '3', 'R', 'I', 'Z', 'C', 'B', 'i', 'e', 'S', 'B', 'I', 'Y', '2', 'x', 'p', 'c', 'H', 'N', 'I', 'I', 'G', 'I', 'z', 'I', 'H', 'B', '1', 'Y', 'm', 'x', 'p', 'c', '2', 'h', 'p', 'b', 'm', 'c', 'g', 'c', '2', 'V', 'j', 'c', 'm', 'V', '0', 'X', '2', '1', 'z', 'Z', 'y', 'B', 'j', 'Y', 'W', '4', 'g', 'e', 'W', '9', '1', 'I', 'G', 'd', 'I', 'd', 'C', 'B', 'p', 'd', 'C', 'B', 'm', 'b', '3', 'I', 'g', 'd', 'X', 'M', '=']

To show it in a better way we can convert the list to a string

```
>>> print("".join(y))
```

Hi Stranger. I know that you are a hacker We have the following data
RSBDb3JwIFB1YmxpYyBJb1QgQnJva2VyIGhvc3RlZCBieSBlY2xpcHNlIGlzIHB1Ymxpc2hpbmcgc2VjcmV0X21zZyBjYW4geW91IGdldCBpdCBmb3lgdXM=

- Now let's extract the base64 part of the message to decode it

```
>>> msg = "RSBDb3JwIFB1YmxpYyBJb1QgQnJva2VyIGhvc3RlZCBieSBlY2xpcHNlIGlzIHB1Ymxpc2hpbmcgc2VjcmV0X21zZyBjYW4geW91IGdldCBpdCBmb3 lgdXM="

>>> import base64

>>> print(base64.b64decode(msg).decode('utf-8'))
```

E Corp Public IoT Broker hosted by eclipse is publishing secret_msg can you get it for us

- Challenge started to get spicy from now on
- Let's continue by searching google what is "public iot broker" actually means
- From the results I got this nice tutorial which helped me a lot
http://www.steves-internet-guide.com/into-mqtt-python-client/

- So after installing the paho-mqtt python library I used this script from the tutorial and adjusted it to fit my needs for the challenge and finally it loo ks like this

```python
def on_message(client, userdata, message):
    print("message received " ,str(message.payload.decode("utf-8")))
    print("message topic=",message.topic)
    print("message qos=",message.qos)
    print("message retain flag=",message.retain)


topic = "secret_msg"

broker_address="iot.eclipse.org"

print("creating new instance")

client = mqtt.Client("P1")
client.on_message=on_message
print("connecting to broker")

client.connect(broker_address)
client.loop_start()
print("Subscribing to topic",topic)

client.subscribe(topic,qos=1)

time.sleep(200) # wait

client.loop_stop() #stop the loop
```

- After I ran the script it showed me the banner of the topic I subscribed on the broker it showed me this

```
creating new instance
connecting to broker
Subscribing to topic secret_msg
message received  VGhpcyBGbGFnIHdpbGwgYmUgc2VudCBldmVyeSAyIG1pbg==
message topic= secret_msg
message qos= 1
message retain flag= 1
```

- It is another base64 message so let's decode it

>>> print(base64.b64decode('VGhpcyBGbGFnIHdpbGwgYmUgc2VudCBldmVyeSAyIG1pbg==').decode('utf-8'))
This Flag will be sent every 2 min

- Yeah we are so close we just have to make our script to wait more than 2 minutes to get the flag
- After I adjusted the time I ran it and waited 2 minutes and the flag started to show up (every character's ascii value in a separate message)
- So I collected all the messages in a string and made a regular expression to gather the ascii values of the flag

```
message received  Hacker You Have gone so far Time to Send you some good stuff
message topic= secret_msg
message qos= 1
message retain flag= 0
message received  83
message topic= secret_msg
message qos= 1
message retain flag= 0
message received  97
message topic= secret_msg
message qos= 1
message retain flag= 0
message received  108
message topic= secret_msg
message qos= 1
message retain flag= 0
message received  117
message topic= secret_msg
message qos= 1
message retain flag= 0
message received  115
message topic= secret_msg
message qos= 1
message retain flag= 0
message received  108
message topic= secret_msg
message qos= 1
message retain flag= 0
message received  97
message topic= secret_msg
message qos= 1
message retain flag= 0
message received  98
message topic= secret_msg
message qos= 1
message retain flag= 0
message received  123
message topic= secret_msg
message qos= 1
```

```
>>> import re
>>> flag = re.findall('received .*', s)
>>> flag
['received 83', 'received 97', 'received 108', 'received 117', 'received 115', 'received 108', 'received 97', 'received 98', 'received 123', 'received 49', '
received 95', 'received 83', 'received 116', 'received 48', 'received 108', 'received 51', 'received 95', 'received 66', 'received 114', 'received 48', 'rece
ived 107', 'received 51', 'received 114', 'received 95', 'received 70', 'received 108', 'received 52', 'received 103', 'received 125']
>>> for i in range(len(flag)):
        x = flag[i].split("  ")
        flag[i] = chr(int(x[1]))


>>> flag
['S', 'a', 'l', 'u', 's', 'l', 'a', 'b', '{', 'l', '_', 'S', 't', '0', 'l', '3', '_', 'B', 'r', '0', 'k', '3', 'r', '_', 'F', 'l', '4', 'g', '}']
>>> print("".join(flag))
Saluslab{l_St0l3_Br0k3r_Fl4g}
```

Done :)