Question 1

27 June 2025 15:26

- The given question is to decode the following:
  01000011 01010011 01001111 01000011 00110010 00110101 173 61 144 63 156 67 61 146 171 49 110 103 95 100 49 66 66 33 72 33 6e 37 5f 33 bmMwZDFuZzV9

- So we can observe that the first part is binary and the last is base64
  On decoding using dcode.fr both we get the respective decoded answers.

- Moving on to the middle part it seems to be a mix of ASCII numbers and HEX format ascii numbers, as when the encoding changes there seems to be a double space between the different strings. Which also on decoding we get the respective part, finally on concatenating all the strings, we get the flag.

- Therefore the answer is:

  CSOC25{1d3n71fy1ng_d1ff3r3n7_3nc0d1ng5}

Question 2

27 June 2025 15:35

- We are given a source and output.
- On trying to decode the source, we first identify its in base 64 encoding by either trial and error or passing through dcode.fr cipher identifier.
- We get a python script which shows how the output was created
- Now we have to reverse engineer the input.
- The logic of the reverse engineered python script is that, in the original every four characters the first two are jus converted to integer then again hex and the other two characters are made to go through a XOR function with the first two then converted hex.
- So reversing it would mean making The first two digits in the converted go through XOR function with the second two and then adding the first , which would give the flag.
  CSOC25{y0u_kn0w_X0r_4nd_b45364}