16-124-836
Marcel Matti ZAUDER

# 12.5 Question 5

## 12.5.A Onion Routing (GROUP R3)

Public Key: e=511, n=851
Private Key: p=37, q=23, d=31 ⇒ n = 37*23 = 851

Received Message from R2: [699, 759, 306, 537, 225, 277, 277, 815, 437, 432, 47, 47, 288, 288, 288, 460, 301, 487, 277, 460, 144, 232, 47, 319, 599, 336, 404, 587, 313, 516]

Decryption process: $m = c^d \ mod \ n$ using a calculator and in the end mapping the results to ASCII characters using the ASCII table.

Decrypted message [numbers]: [71, 69, 84, 32, 104, 116, 116, 112, 115, 58, 47, 47, 119, 119, 119, 46, 98, 105, 116, 46, 108, 121, 47, 51, 70, 67, 118, 50, 113, 89]
Decrypted message [ascii]: [G, E, T, _, h, t, t, p, s, :, /, /, w, w, w, ., b, i, t, ., l, y, /, 3, F, C, v, 2, q, Y]

⇒ «GET https://www.bit.ly/3FCv2qY»

## 12.5.B The encryption scheme in question 5 A isn't particularly secure. What are its weaknesses? How would you make it more secure?

The problem with this encryption scheme is that there is a one-to-one correspondence between the ASCII character and its cipher, meaning that the same cipher corresponse to the same character. Therefore an analysis of the ciphertext can be performed and a mapping of each cipher character to its original plain character can be made.
A solution for this exploit would be to add some kind of randomness that can be reverted like using a seed or a predefined random function in order to mitigate such an analysis attack.