**1: SECRET SHARING - BASIC EXAMPLE**

Assume a (3,5) Secret Sharing scheme, with standard numbering of parties. i.e., Pi 🡪 xi = {1,2,3,4,5}. Using arithmetic modulus 11, the dealer hides a secret S in the range [0,10]. At reconstruction time, Parties 1,2 and 4 show their shares:

P1 🡪 share1 = 4

P2 🡪 share2 = 7

P4 🡪 share4 = 9

What is the secret?

[answer: 7]

**2: SECRET SHARING – Information leakage**

Assume the same SS scheme as in exercise 1 above but in this case using **ordinary arythmetics – NO MODULUS**! The dealt secret S was the range [0,10]. At reconstruction time, Parties 2 and 4 show their shares:

P2 🡪 share2 = 29

P4 🡪 share4 = 75

Prove that even if a third share is missing, this information is sufficient to completely reveal the secret!

[*hint: write an expression parametric in the hidden secret S, and show that only one value in the range [0,10] satisfies a condition…*]

**3: Common Modulus Attack**

An RSA scheme uses modulus n = 77; A same message M is RSA-encrypted using two different public keys e1 = 17 and e2 = 23, but same modulus n=77. The two resulting ciphertexts are: c1=60 and c2=53. Decrypt the message applying the Common Modulus Attack.

[Answer: M=37]

**4: verificable Secret Sharing**

Let p=83 and q=(p-1)/2 = 41.

Assume a (2,n) Verificable Secret Sharing scheme with Modulus 41 which uses the Feldmann scheme with g=10 and modulus 83. The commitments are:

C0 = 23

C1 = 4

Party P7 (x=7) receives share s7 = 29: verify that this is a valid share.

[NOTE: computations here are NOT IMMEDIATE (somewhat large numbers), so if this type of exercise will occur in the exam, I’ll select very special numbers so as to further reduce the computation complexity – In this exercise I haven’t spent too much time to find “good” numbers]

**5: Find elliptic Curve Points**

Consider the Elliptic curve

Y2 = x3 + 2x + 1

defined over the modular integer field Z5:

find all the points EC(Z5)

[answer: O, {0,1}, {0,4}, {1,2}, {1,3},{3,2},{3,3}]

**6: Pairing Based crypto**

Being e: GxG🡪Gt a bilinear map, and g a generator of G, simplify the expression:

e(gx x gy, gz)w / e(gwz, gx)