Université Côte d'Azur — L3 Info, L3 MI, L3 ST
Cryptographie et calcul symbolique — 2023-24

TD n° 4

Cryptographie à clef publique

Exercice 1) Théorème d'Euler Mettons que l'on veuille connaître le chiffre unité de 7^{222}. Comment faire dans ce cas précis pour le trouver en n'effectuant que des calculs très simples ?

Exercice 2) Indicatrice d'Euler Soit le cryptosystème RSA avec la clef publique d'Alice suivante $K_{pub} = (n, e) = (319, 11)$.

1. Comment Bob chiffre-t-il le message $M = 100$ avec cette clef ?
2. Sans calculer les coefficients de Bézout, calculez d la clef privée de déchiffrement d'Alice correspondant à sa clef de chiffrement publique e.
3. Déchiffrez le message chiffré $C' = 133$ reçu par Alice.
4. Le message chiffré $C'' = 625$ peut-il résulter d'un chiffrement avec la clef publique e d'Alice ?

Exercice 3) Cryptosystème RSA Bob envoie le message chiffré $C = 55$ à Alice qui a publié sa clef publique $(n, e) = (187, 107)$. Melchior veut trouver la clef privée d'Alice afin de déchiffrer indûment ce message.

1. Aidez Melchior à calculer la clef privée d d'Alice.
2. Retrouvez le message clair correspondant au chiffré $C = 55$ que Bob a chiffré pour Alice.
3. Comment Bob avait-il fait pour chiffrer ce message.

Exercice 4) Factorisation En admettant que l'entier 22733 est le produit de deux nombres premiers, pouvez-vous le factoriser ? Si en outre, on vous révèle que $\varphi(22733) = 22428$, la factorisation devient-elle possible ?

Exercice 5) Diffie-Hellman Ce protocole permet à deux protagonistes de se mettre d'accord sur un secret à distance.

1. Pourquoi ce protocole n'est pas forcément adapté au partage d'une clef secrète en vue de faire du chiffrement symétrique ?
2. L'attaque MITM est-elle envisageable envers ce protocole ?

3. Alice et Bob se mettent d'accord pour utiliser ce protocole afin d'obtenir un secret en commun. Ils choisissent les paramètres $g = 2$ et $p = 11$. Eve espionne leurs communications, elle connaît donc les paramètres g et p. Elle voit passer un nouvel envoi d'Alice, elle en déduit que $g^a \mod p = 6$ sans connaître l'entier a choisi par Alice. De même elle déduit d'un envoi de Bob que $g^b \mod p = 3$ sans connaître le b choisi par Bob.
   Les paramètres sont petits là, Eve peut en déduire leur secret ... Calculez vous aussi ce secret en expliquant votre méthode.

Exercice 6) Coefficients de Bézout Vous connaissez l'*algorithme d'Euclide* pour calculer le *pgcd* de deux entiers naturels a et b. Il existe aussi un *algorithme d'Euclide étendu* qui fournit en prime les *coefficients de Bézout*. Ce sont des entiers relatifs u, v tels que :

$$pgcd(a, b) = u.a\ + b.v$$

Pour a et b donnés, ce couple de coefficients (u, v) n'est pas unique. La librairie `sympy` de PYTHON fournit la fonction `gcdex()` *(Greater Common Divisor extended)* permettant de les obtenir en plus du $pgcd$. Prouvez la correction de la version récursive de cet algorithme si utile en cryptographie :

```
def bezout(a,b) :
    if b == 0 :
        return (a,1,0)
    q = a // b
    r = a % b
    g,u,v = bezout(b,r)
    return g,v,u-q*v
```

Exercice 7) Partage de secret Dans le cadre d'un service financier, on souhaite utiliser le principe d'une clef publique et de plusieurs clefs privées appliqué à un problème de secret. Trouvez un système afin que deux directeurs au moins doivent être réunis pour obtenir et donc partager un secret leur permettant d'ouvrir un coffre. *Indication : la solution attendue ici est purement géométrique !*

Exercices complémentaires

Exercice 8) Déchiffrement RSA

La clef publique de Bob est $K_{pub} = (e, n) = (27, 55)$. Alice chiffre son message M et envoie à Bob le message chiffré correspondant $C = 4$. Vous interceptez ce chiffré C. Déduisez-en M.

Exercice 9) Coefficients de Bézout Comprenez les versions itératives plus ou moins *pythonesques* suivantes pour calculer ces coefficients :

```
def bezoutIt(a,b) :
    u0,v0,u1,v1 = 1,0,0,1
    while b != 0 :
        q,r = divmod(a,b)
        a,b = b,r
        u0,v0,u1,v1 = u1,v1,u0-q*u1,v0-q*v1
    return a,u0,v0


def bezoutIter(a,b) :
    if b == 0 :
        return (a,1,0)
    u0 = 1
    v0 = 0
    u1 = 0
    v1 = 1
    while b != 0 :
        q = a // b
        r = a % b
        u2 = u0 - q * u1
        v2 = v0 - q * v1
        a = b
        b = r
        u0 = u1
        v0 = v1
        u1 = u2
        v1 = v2
    return a,u0,v0
```

Exercice 10) Le protocole de Shamir Ce protocole qui date de 1979 est une simple évolution des méthodes géométriques de l'Exercice n° 7 précédent. Le réel S est le secret à partager. Il y a n participants et le *seuil* pour reconstituer le secret est de t participants. On se donne des réels choisis au hasard et de façon uniforme a_1, a_2, ..., a_{t-1}. On considère le polynôme :

$$P(X) = S + \sum_{j=1}^{t-1} a_j X^j$$

P est donc de degré $t-1$. Chaque participant reçoit un nombre x_i et son image $y_i = P(x_i)$. Si t participants rassemblent leurs informations, ils connaissent t points ainsi que leurs images par un polynome de degré $t-1$. Par la théorie des polynômes interpolateurs de Lagrange, on sait reconstituer P, et donc retrouver S ! On peut aussi démontrer que si l'on dispose seulement de $t-1$ points et de leurs images, on ne peut retrouver aucune information sur S.