Université de Sorbonne Paris Nord **CC2 de cryptologie** **Année 2023-2024**
Institut Galilée

Q1. On rappelle qu'un système de chiffrement asymétrique est la donnée de trois algorithmes :
— un algorithme $G()$ qui engendre aléatoirement une paire (pk, sk) de clés publique et privée.
— un algorithme $E(pk, m)$ qui étant donné une clé publique pk et un message clair m retourne un chiffré c.
— un algorithme $D(sk, c)$ qui étant donné une clé privée sk et un chiffré c retourne un texte clair m .

Dans le cas du RSA d'école, l'ensemble des textes clairs et chiffrés sont les entiers modulo $n = p.q$, où p et q sont deux nombres premiers.
— l'agorithme $G()$ choisit p et q, calcule $n = pq$ et $\phi(n) = (p-1)(q-1)$, puis calcule e et d tels que $ed = 1 \mod \phi(n)$. La clé publique est (n, e) et la clé privée est (n, d).
— $E((n, e), m) = m^e \mod n$.
— $D((n, d), m) = m^d \mod n$.

Programmez RSA d'école en python et vérifier la consistance du système. Vous pouvez utiliser les bibliothèques python math, sympy. Donner les trois fonctions G, E et D en python.

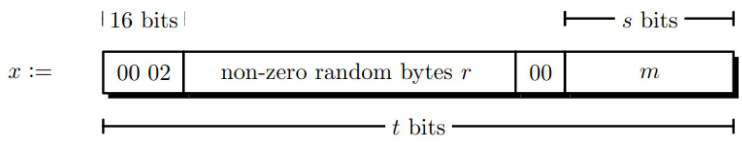
Le but de cet exercice est de montrer une raison supplémentaire de la non utilisabilité du RSA d'école : sa maléabilité. On va considérer la situation artificielle d'un service (appelé oracle dans la suite) qui pour chaque chiffré retourne la parité du texte clair (ou encore son bit le moins significatif). Ici, l'attaquant ne dispose initialement que d'un chiffré dont il veut retrouver le clair et de l'accès à l'oracle. Il est clair qu'il y a une fuite d'information avec cet oracle : si on détient un secret, il ne faut pas révéler d'information à son sujet. Ce qui l'est moins, c'est qu'on peut retrouver le texte clair bit à bit à partir de l'oracle de parité.

1. Soit $c = E((n, e), m)$. Quel est le chiffré de $2m$?
2. Peut-on calculer le chiffré de $2m$ à partir du chiffré de m ?
   On remarque que :
   — soit $2m < n$, donc $2m \mod n = 2m$ et le bit le moins significatif de $2m$ est 0.
   — soit $2m \geq n$, donc $2m \mod n = 2m - n$ qui est impaire (comme n est impair) et le bit le moins significatif de $2m$ est 1.
   Ainsi le bit le moins significatif de $2m$ nous informe si $0 \leq m < n/2$ ou $m \geq n/2$.
3. En déduire une attaque qui retrouve le message m en utilisant l'oracle de parité.
4. Programmez l'attaque en python.
5. Quelle est la complexité de cette attaque en fonction de la taille de n ?

Pour la culture générale : Dans la section 12.8.3 de A Graduate Course in Applied Cryptography, Boneh & Shoup 2023 on peut lire :

> Bleichenbacher's attack on the RSA-PKCS1 encryption scheme : RSA-PKCS1 encryption is not secure against chosen ciphertext attacks. We describe an attack, due to Bleichenbacher, as it applies to the SSL 3.0 protocol used to establish a secure session between a client and a server. The SSL 3.0 protocol was later replaced by an improved protocol called TLS 1.0 that defends against this attack, as discussed below. The latest version of TLS, called TLS 1.3, has moved away from RSA encryption altogether (see Section 21.10).

Le principe du padding PKCS1 est illustré ci-dessous. Le problème se situe dans les 16 premiers bits d'entête du message chiffré qui doivent être égaux à 0x0002 : si ce n'est pas le cas, les versions vulnérables de SSL renvoient une erreur ce qui conduit à une fuite d'information, non pas sur le bit de poids faible comme avec l'oracle de parité, mais sur les 16 bits de poids forts car on sait qu'on a réussi à obtenir un chiffré plausible. On remarque qu'avec RSA-PKCS1, on ne peut pas chiffrer des messages aussi long qu'avec textbook RSA car il faut environ 11 octets (2 d'en-tête + 8 du nombre aléatoire + 1 séparateur) pour le padding.



L'attaque est un peu plus compliquée à mettre en oeuvre, car il faut chercher des messages et l'information obtenue est moins simple que diviser l'espace de recherche en deux, mais elle est réalisable.

Q2.

— Décrire la fonction à sens unique avec trappe reliée à RSA.

— Supposons que le même message m est donnée à l'entrée de la fonction à sens unique reliée à RSA, avec deux clés publiques RSA (n, e) et (n, f) et e et f sont premiers entre eux. Montrer que l'on peut déduire le message m à partir des deux chiffrés. (Indication : utiliser l'algorithme d'Euclide étendu).

— Application : le même message m a été chiffré avec les clés $(493, 3)$ et $(493, 5)$. Les chiffrés sont respectivement 293 et 421. Retrouver le message m.

— Pourquoi ne doit-on pas utiliser la fonction à sens unique avec trappe reliée à RSA directement comme un système de chiffrement ?

— Proposer une méthode pour utiliser RSA comme système de chiffrement.

Q3.

— Les systèmes de chiffrement symétriques permettent de chiffrer des messages clairs de 32 bits en des chiffrés de 32 bits. Peut-on faire de même en asymétrique ? Pourquoi ?

— Soit (G, E, D) un système de chiffrement asymétrique sémantiquement sûr, où G est l'algorithme qui engendre la paire de clés publique et privée (pk, sk), E est l'algorithme de chiffrement et D est l'algorithme de déchiffrement. La fonction de chiffrement E peut–elle être déterministe ? (Justifier)

Q4. Montrer que 7 est un générateur de $\mathbb{Z}_{13}^{\times}$? 3 est-il générateur de $\mathbb{Z}_{13}^{\times}$? Quel est le logarithme discret de 10 en base 7 dans $\mathbb{Z}_{13}^{\times}$?

Q5. Calculer $2^{245} \mod 35$ en justifiant et en détaillant la trace d'exécution de l'algorithme. Montrer que 7 est un générateur de $\mathbb{Z}_{13}^{\times}$? 3 est-il générateur de $\mathbb{Z}_{13}^{\times}$? Quel est le logarithme discret de 10 en base 7 dans $\mathbb{Z}_{13}^{\times}$?