Problèmes avec le chiffrement RSA de clés

- Rappelons ce que nous disions précédemment : RSA est un système qui peut être considéré sûr si le message à chiffrer est aléatoire dans l'intervalle 1...N-1.
  - Ceci n'est certainement pas le cas lorsque le message consiste en une clé pour AES de 128 bits ou, pire, une clé DES de 64 bits.
  - La méthode qui consiste à ajouter des «0» au bout de la clé n'est pas sûre!
- Nous devons donc trouver une méthode de remplissage pour allonger les messages et les rendre aussi aléatoires que possible...

OAEP: Remplissage optimal pour chiffrement asymétrique

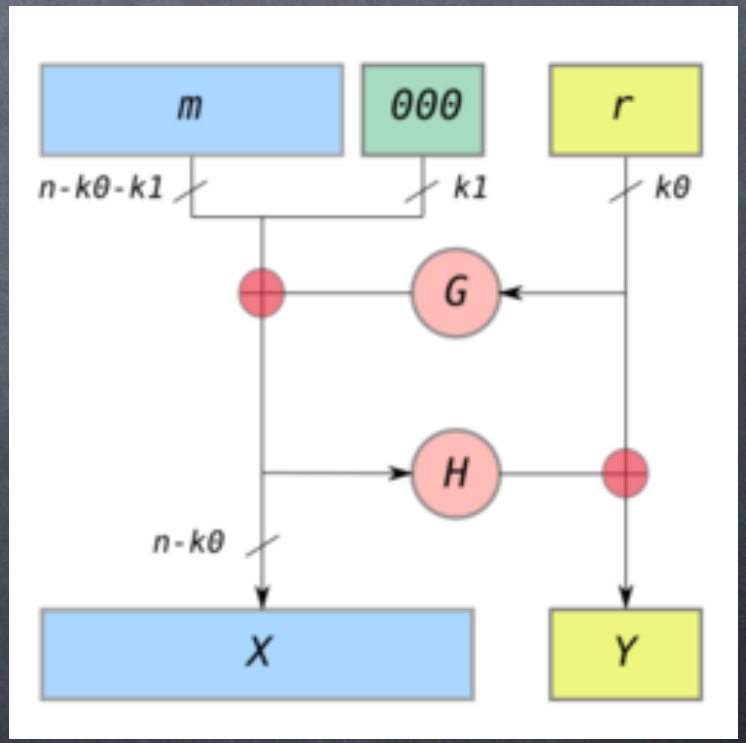
- OAEP est un standard qui permet de chiffrer des messages beaucoup plus courts que ce que la taille de N (dans RSA) permet.

Supposons que N a n bits de long.

Supposons les messages à envoyer ont m bits de long t.q. m=n-k0-k1.

$G:\{0,1\}^{k0} \rightarrow \{0,1\}^{n-k0}$ Est une fonction d'expansion cryptographique : Un générateur pseudo-aléatoire

$H:\{0,1\}^{n-k0} \rightarrow \{0,1\}^{k0}$ fonction de hachage cryptographique (p.ex. SHA256)



OAEP (II)

- Soient :
  - $G:\{0,1\}^{k0} \rightarrow \{0,1\}^{n-k0}$ un générateur pseudo-aléatoire
  - $H:\{0,1\}^{n-k0} \rightarrow \{0,1\}^{k0}$ une fonction de hachage cryptographique comme SHA256.
- Alors, le message $m \in \{0,1\}^{n-k0-k1}$ sera codé pour r aléatoire de longueur k0 par :
  - $OAEP(m) = (m \,||\, 0^{k1}) \oplus G(r) \,||\, H((m \,||\, 0^{k1}) \oplus G(r)) \oplus r$
  - Sur réception de OAEP(m) le destinataire sépare le bourrage en deux parties $O_1 || O_2 = OAEP(m)$.
    - Calculer $H(O_1) \oplus O_2 = r$, et
    - $G(r) \oplus O_1 = m \,||\, 0^{k1}$.