Sécurité des SI

TD5

1) Secret sharing scheme : Alice a un secret qu'elle ne veut pas garder sur elle. Elle décide de le diviser en fragments de secrets et distribue un fragment à chacun de ses quatre plus proches amis. Elle veut pouvoir reconstituer le secret avec trois fragments. Pour cela, elle a juste à contacter trois amis parmi les quatre et à partir des trois fragments reçus, utiliser l'interpolation de Lagrange pour obtenir le secret.
   Alice choisit p=7. Elle distribue un point à chacun de ses quatre amis.
   Alice distribue (1,4), (2,1), (3,2), (6,1)
   On rappelle que le secret est le terme constant du polynôme construit par Alice.
   Quel est le secret d'Alice ?
   Rappel : Si les points sont notés (xi, yi) le secret se retrouve par la formule

$$d_0 = \sum_{i=0}^{k} \left(\prod_{\substack{j=1 \\ j \neq i}}^{k} \frac{-x_j}{x_i - x_j} \right) y_i \pmod p$$

2) L'algorithme ElGamal utilise un groupe Z*p et un générateur g du groupe.
   1) Détaillez l'algorithme
   2) Comment trouver le générateur g facilement ?
3) Fault attack on RSA. L'algo de signature utilise souvent CRT pour aller plus vite (environ 4 fois plus rapide). Il s'agit de calculer Sp=c^d dans Zp et Sq=c^d dans Zq et d'en déduire S=c^d dans Zn (n=p.q). Un attaquant peut provoquer une faute lors du calcul de Sq (par exemple avec un laser). Comment, à partir de la valeur Sp et de la valeur fausse Sq, un attaquant peut-il en déduire la factorisation de n ?
4) Fonction d'accumulation. Un dictionnaire authentifié est une structure de données qui supporte les requêtes authentifiées d'appartenance. Par exemple, si Alice émet une requête pour obtenir un document, elle le recevra (s'il existe) et obtiendra une preuve d'authenticité du document. Le dictionnaire pourrait signer tous les documents comme preuve d'authenticité. Mais cette solution est lourde si le nombre de documents est grand et que ceux-ci sont mis à jour très souvent. Une autre solution consiste à prouver que le document reçu fait bien partie du dictionnaire. Pour cela on peut utiliser une fonction d'accumulation. Soit n=p.q et g un élément de Z*n. Chaque document a un identifiant i et son haché est hi. L'entité gérant le dictionnaire va signer g^A, où A est le produit de tous les hachés. Lorsqu'un client souhaite obtenir le document j, il le reçoit accompagné de la valeur Pj=g^B, où B est le produit de tous les hachés (sauf sauf celui de j). Le client hache le document reçu calcule (Pj)^hj et compare la valeur à g^A.
   Soit n=323, g^A=234. Le client demande le document doc1 de valeur de haché 47. Il le reçoit accompagné de la valeur 157.
   1) L'authentification est-elle correcte ?
   2) Quels sont les élément publics, secrets dans ce système pour obtenir une bonne sécurité ? Cette solution est arithmétique et donc un peu lourde.
   3) Existe-t-il une autre solution utilisant uniquement des fonctions de hachages et permettant de vérifier que le document reçu appartient bien au dictionnaire ?