Université Paris 8 J. Lavauzelle

Master 1 Mathématiques et applications — année 2022–23

Cryptographie à clé publique – Feuille de TD 1

27/01/2023

Le corrigé de certains exercices sera disponible à l'adresse suivante :

`www.math.univ-paris13.fr/~lavauzelle/teaching/2022-23/clef-publique.html`

$(\star)$ exercice fondamental $(\star\star)$ pour s'entraîner $(\star\star\star)$ pour aller plus loin 💻 sur machine

Exercice 1. $(\star)$ Application du protocole de Diffie–Hellman.

On exécute le protocole de Diffie–Hellman dans le groupe multiplicatif $\mathbb{G} = \mathbb{F}_p^\times$ avec les paramètres suivants :

- $p = 19$,
- le générateur du groupe est $g = 2$,
- la valeur secrète d'Alice est $a = 4$,
- la valeur secrète de Bob est $b = 6$.

Question 1.– Quelle est la valeur commune obtenue par Alice et Bob ?

On souhaite maintenant exécuter le protocole de Diffie–Hellman dans le sous-groupe des résidus quadratiques de $\mathbb{F}_p^\times$, que l'on note $\mathrm{QR}_p^\times$. Pour cela on garde la valeur $p = 19$.

Question 2.– L'élément $g = 2$ reste-t-il un générateur de $\mathrm{QR}_p^\times$?

Question 3.– Démontrer que 5 est un générateur de $\mathrm{QR}_p^\times$.

Question 4.– Exécuter le protocole de Diffie–Hellman dans $\mathrm{QR}_p^\times$ avec les valeurs suivantes :

- $p = 19$,
- $g = 5$,
- la valeur secrète d'Alice $a = 4$,
- la valeur secrète de Bob est $b = 6$.

Exercice 2. ($\star\star$) Protocole de Diffie–Hellman dans $(\mathbb{Z}/n\mathbb{Z}, +)$.

On considère une variante additive du protocole de Diffie–Hellman dans le groupe fini $\mathbb{G} = (\mathbb{Z}/n\mathbb{Z}, +)$ où n est un très grand nombre entier. Pour cela, on fixe un générateur $M \neq 0$ du groupe $\mathbb{G}$, que l'on publie. Dans cette version « additive », le protocole devient donc :

Alice		Bob
engendre $a \in \mathbb{Z}/n\mathbb{Z}$		engendre $b \in \mathbb{Z}/n\mathbb{Z}$
	$u = aM \mod n$ $\longrightarrow$	
	$\longleftarrow$ $v = bM \mod n$	
calcule $av \mod n$		calcule $bu \mod n$

Question 1.– Donner une caractérisation simple des générateurs du groupe additif $(\mathbb{Z}/n\mathbb{Z}, +)$. Puis, proposer un exemple de générateur qui convient pour tout n.

Question 2.– Vérifier que lorsqu'ils suivent le protocole ci-dessus, Alice et Bob détiennent bien un secret commun.

Question 3.– Rappeler un algorithme qui calcule l'inverse modulaire dans $\mathbb{Z}/n\mathbb{Z}$ (autrement dit, qui effectue l'opération $x \mapsto x^{-1} \mod n$). La complexité de cet algorithme est-elle polynomiale en $\log n$?

Question 4.– Selon vous, la variante du protocole de Diffie–Hellman proposée dans cet exercice est-elle sûre ? Justifier.

Exercice 3. ($\star\star$) Échange de clefs et masque jetable.

On note $\mathcal{X} = \mathbb{F}_2^n$ l'ensemble des chaînes de bits de longueur n, munies de l'opération de `xor` bit-à-bit, notée $\oplus$. On considère le protocole d'échange de clefs suivant.

Alice		Bob
engendre aléatoirement $\boldsymbol{k}, \boldsymbol{r} \in \mathcal{X}$		engendre aléatoirement $\boldsymbol{t} \in \mathcal{X}$
	$\boldsymbol{s} = \boldsymbol{k} \oplus \boldsymbol{r}$ $\longrightarrow$	
	$\longleftarrow$ $\boldsymbol{u} = \boldsymbol{s} \oplus \boldsymbol{t}$	
	$\boldsymbol{v} = \boldsymbol{u} \oplus \boldsymbol{r}$ $\longrightarrow$	
détient $\boldsymbol{k}$		détient $\boldsymbol{v} \oplus \boldsymbol{t}$

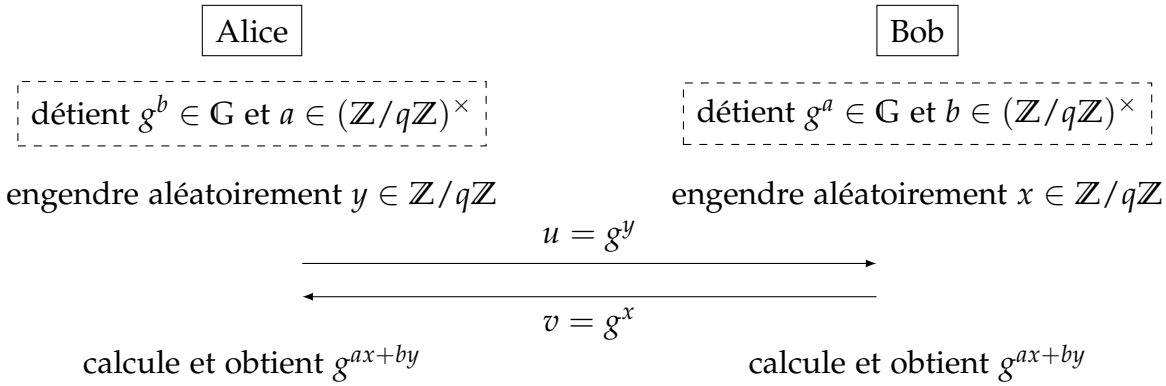
Question 1.– Vérifier qu'Alice et Bob détiennent bien un secret commun.

Question 2.– Le protocole est-il sûr ? Justifier.

Exercice 4. $(\star\star)$ Une variante de Diffie–Hellman.

Alice et Bob souhaitent échanger un secret commun. Pour cela, ils mettent en place une variante du protocole de Diffie–Hellman. On se place donc dans un groupe cyclique $\mathbb{G}$ d'ordre q. **On suppose que q est premier.** Enfin, un générateur g de $\mathbb{G}$ est donné publiquement.

On suppose qu'Alice a engendré une valeur secrète aléatoire $a \in (\mathbb{Z}/q\mathbb{Z})^\times$, et a publié $g^a \in \mathbb{G}$. De même, Bob a engendré une valeur secrète $b \in (\mathbb{Z}/q\mathbb{Z})^\times$ et publié $g^b \in \mathbb{G}$. Le protocole d'échange de secret commun est décrit ci-dessous.



Question 1.– Détailler les calculs permettant à Alice et Bob de calculer la valeur commune g^{ax+by}, à partir des valeurs que chacun détient. En donner la complexité.

Question 2.– Une fois que les entiers a et b sont fixés, quelle est la taille de l'ensemble des valeurs communes possibles ?

Question 3.– Pourquoi l'attaque de la personne au milieu (*man-in-the-middle*), telle que décrite dans le cours pour le protocole de Diffie–Hellman, ne fonctionne-t-elle pas dans le cas présent ?

Question 4.– Supposons que l'on détienne un algorithme $\mathcal{A}$ qui sache résoudre efficacement le problème CDH (Diffie–Hellman calculatoire). Démontrer qu'on peut alors retrouver la valeur commune détenue par Alice et Bob en observant leurs échanges (c'est-à-dire en procédant à une attaque passive).

Exercice 5. $(\star\star\star)$ Problèmes DH et sqDH.

Soit $\mathbb{G}$ un groupe cyclique d'ordre r, et $g \in \mathbb{G}$ un générateur du groupe. On rappelle que le problème calculatoire de Diffie–Hellman dans $\mathbb{G}$ est :

$$\mathsf{CDH} : \text{étant donné } (g, g^a, g^b), \text{ calculer } g^{ab}$$

On définit maintenant le problème de « Diffie–Hellman carré » dans $\mathbb{G}$ comme :

$$\mathsf{sqDH} : \text{étant donné } (g, g^a), \text{ calculer } g^{a^2}.$$

Question 1.– Expliquer en quoi sqDH peut être vu comme un sous-problème de CDH.

Question 2.– Supposons que l'on détienne un algorithme A qui résout efficacement sqDH. Démontrer qu'à partir d'une instance (g, g^a, g^b) de CDH, on peut calculer g^{2ab}.

Question 3.– Pour simplifier, on suppose ici que r est impair. On dit que x est un carré dans $\mathbb{G}$ s'il existe un $y \in \mathbb{G}$ tel que $x = y^2$. Décrire un algorithme qui calcule une racine carrée d'un carré $x \in \mathbb{G}$.

Question 4.– Toujours dans le cas où r est impair, en conclure que CDH se réduit à sqDH.

Exercice 6. $(\star\star)$ 💻 Implantation de Diffie–Hellman dans $\mathrm{QR}_p^\times$.

Dans cet exercice on suppose que p est un nombre premier de taille importante **tel que** $p' = (p-1)/2$ **est également un nombre premier**. Un tel nombre premier p est appelé **nombre premier sûr**, tandis que p' est un **nombre premier de Sophie Germain**.

Question 1.– Trouver dans les bibliothèques `random` et `math` de `python` comment :

- tirer uniformément un entier entre 1 et x,
- calculer efficacement une puissance modulaire,
- calculer efficacement un inverse modulaire.

Question 2.– Écrire une fonction `is_square(x, p)` qui teste si un entier `x` est un résidu quadratique modulo `p`.

Question 3.– Écrire une fonction `random_QR_generator(p)` qui retourne un générateur aléatoire du groupe des résidus quadratiques modulo `p`. On rappelle que l'on suppose que $p' = (p-1)/2$ est également un nombre premier.

Question 4.– Écrire les fonctions suivantes du protocole de Diffie–Hellman :

- une fonction `system_parameters(t)` qui définit les paramètres du protocole : un nombre premier p aléatoire de taille t bits tel que $p' = (p-1)/2$ est aussi premier, et le générateur aléatoire g de $\mathrm{QR}_p^\times$,
- une fonction `random_secret(p)` qui définit la valeur secrète x engendrée par un participant au protocole,
- une fonction `to_send(x, g, p)` qui déduit de la valeur secrète x et du générateur g la valeur à transmettre à l'autre participant,
- une fonction `shared_value(gy, x, p)` qui déduit de la valeur reçue `gy` $(= g^y)$ et de lm valeur secrète x, la valeur commune aux deux participants.

Pour l'implantation de la fonction `system_parameters(t)`, on pourra notamment s'aider de la fonction `Crypto.Util.number.getPrime(t)` de la bibiothèque `Crypto.Util`, voir :

`https://pycryptodome.readthedocs.io/en/latest/src/util/util.html#Crypto.Util.number.getPrime`

Question 5.– En choisissant une taille $t = 64$ (exemple-jouet), tester vos fonctions en exécutant les actions successives d'Alice et Bob dans le protocole de Diffie–Hellman.