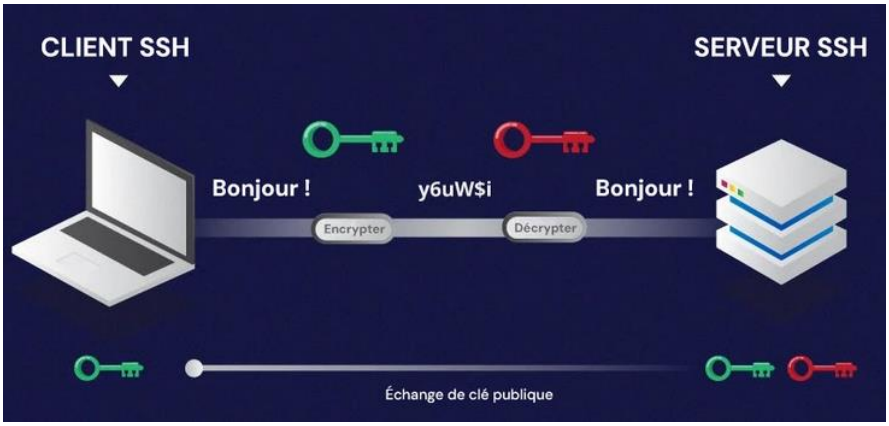
Exercice - Le chiffrement RSA

Le **chiffrement RSA** (nommé par les initiales de ses trois inventeurs) est un algorithme de cryptographie **asymétrique**, très utilisé dans le commerce électronique, et plus généralement pour échanger des données confidentielles sur Internet. Il utilise une paire de clés (des nombres entiers) composée d'une clé publique pour chiffrer et d'une clé privée pour déchiffrer des données confidentielles.



1- RAPPEL DE QUELQUES NOTIONS D'ARITHMETIQUES :

Définition : Les nombres **premiers** sont des entiers naturels supérieurs ou égal à 2, qui n'ont pas d'autres diviseurs qu'eux-mêmes et 1. Dans cette famille, on a les entiers suivants : $2, 3, 5, 7, 11, 13, 17, 19, 23, 29, 31, 37, 41, 43, 47$

Propriété : 2 nombres n et m sont **premiers entre eux** s'ils n'ont aucun diviseur en commun, autre que 1. Par exemple :

- 6 et 35 sont premiers entre eux : 6 = 2x3 et 35 = 7x5
- 6 et 27 ne sont pas premiers entre eux car ils ont 3 comme diviseur commun : 6 = 2x3 et 27 7x3

2- DECOUVERTE AVEC UN CAS SIMPLIFIE :

a- Génération d'une clé publique (n, e) et d'une clé privée (n, d) :

Méthode	Application
1- On choisit au hasard deux nombres premiers p et q	On peut choisir par exemple : $p = 11$ et $q = 17$
2- On calcule les entiers n et m suivants : $n = p\,q$ et $m = (p-1)(q-1)$	On a alors : $n = 11 \times 17 = 187$ et $m = (11-1)(17-1) = 10 \times 16 = 160$
3- On choisit un entier e appelé *exposant de chiffrement* qui respecte les conditions suivantes : $1 < e < m$ et e **premier** avec m	On peut choisir : $e = 3$
4- On choisit un entier d appelé exposant de déchiffrement qui respecte la condition $(d \times e)\%m = 1$	On peut choisir : $d = 107$ Car on a bien : $(107 \times 3)\%187 = 321\%187 = 1$

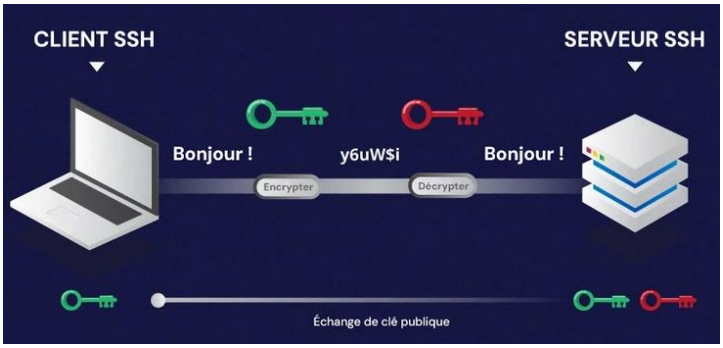
On suppose dans la suite que l'on utilise les clés publique (n, e) et privé (n, d) **du serveur**.

Lorsque le client réalise une demande au serveur (requête GET), celui-ci lui répond en lui envoyant « en clair » sa clé publique. Le client répond en utilisant cette clé pour chiffrer un premier envoi. Dans notre exemple ci-dessous, ce premier envoi sera « Bonjour ! ». Le client va le chiffrer en utilisant la clé publique (n, e). Par simplicité, on ne traite ci-dessous que le chiffrement de la première lettre qui est ici « B ».

b- Chiffrement du caractère « B » en utilisant la clé publique (n, e) : clé publique $(187,3)$

Méthode	Application
1- On convertit le caractère en nombre que l'on nommera ici num en utilisant son code ASCII. En python, on peut utiliser la fonction *ord()* qui renvoie le code UNICODE du caractère : $num = ord(..)$	`>>> ord("B")` `66`
2- On calcule l'entier naturel C égal à $C = (num^e)\%n$ Le caractère est chiffré avec le nombre C .	`>>> (66**3)%187` `77`

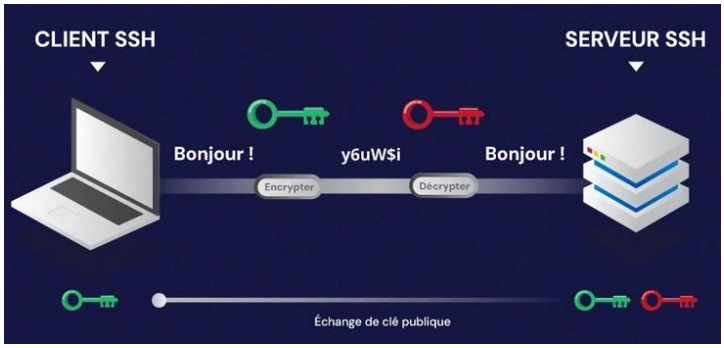
On suppose que cette valeur chiffrée C est envoyée sur le réseau internet. Elle est reçue par le serveur qui va la déchiffrer en utilisant sa clé privée (n, d) secrète. Seul le serveur la connait car il l'a générée lui-même.



c- Déchiffrement de la valeur chiffrée C utilisant la clé privée (n, d) : clé privée $(187,107)$

Méthode	Application
1- On calcule l'entier naturel num égal à $num = (C^d)\%n$ Si la paire de clés public-privée est correcte, on retrouve le code UNICODE envoyé par le client.	`>>> (77**107)%187` `66`
2- On convertit ce code en caractère. En python, on utilise la fonction chr() qui renvoie le caractère en retour.	`>>> chr(66)` `'B'`

Le serveur a pu retrouver la lettre « B » avec sa clé privée qu'il est le seul à connaître. Le serveur est ainsi le seul à pourvoir déchiffrer un message qui a été chiffré avec sa clé publique.



3- LIMITES DE CETTE METHODE SIMPLIFIEE :

Si on utilise la même méthode sur tout le message « Bonjour ! », cela donne :

Chiffrement de « Bonjour ! » avec la clé publique $(n, e) = (187, 3)$

	'B'	'o'	'n'	'j'	'o'	'u'	'r'	' '	'!'
$num = ord(..)$	66	111	110	106	111	117	114	32	33
$C = (num^e)\%n$	77	100	121	13	100	145	130	43	33

Déchiffrement de la suite de nombre chiffrés avec clé privée $(n, d) = (187, 107)$

C	77	100	121	13	100	145	130	43	33
$num = (C^d)\%n$	66	111	110	106	111	117	114	32	33
caractère	'B'	'o'	'n'	'j'	'o'	'u'	'r'	' '	'!'

On peut mettre en avant les limites suivantes :

- Il y a plus de 100 000 codes UNICODE différents. Comme le caractère chiffré C est le résultat d'un calcul modulo n $(\%n)$, pour éviter que 2 caractères aient la même valeur chiffrée, il faut que n soit au-moins supérieur à 100 000.

- Le caractère 'o' de Bonjour se retrouve 2 fois dans le mot 'Bonjour !'. En réalisant une analyse fréquentielle sur la liste de nombre chiffrés, il serait possible de casser ce chiffrement. Pour palier à ce problème, on peut fixer un nombre de chiffres alloué à chaque caractère et ensuite regrouper les caractères. Par exemple si on chiffre des groupes de 2 caractères, chacun représenté par un UNICODE complété à 3 chiffres, cela donnerait pour le mot 'Bonjour !' :

- Chiffrement de « Bonjour ! » avec la clé publique $(n, e) = (1592537, 16349)$

	'B'	'o'	'n'	'j'	'o'	'u'	'r'	' '	'!'
$num = ord(..)$	66	111	110	106	111	117	114	32	33
	66111		110106		111117		114032		33
$C = (num^e)\%n$	`>>> (66111**16349)%1592537` `546825` 546825		`>>> (110106**16349)%1592537` `1025144` 1025144		`>>> (111117**16349)%1592537` `68388` 68388		`>>> (114032**16349)%1592537` `1459330` 1459330		33

Déchiffrement de la suite de nombre chiffrés avec clé privée $(n, d) = (1592537, 389)$

C	546825		1025144		68388		1459330		33
$num = (C^d)\%n$	>>> (546825**389)%1592537 66111 66111		>>> (1025144**389)%1592537 110106 110106		>>> (68388**389)%1592537 111117 111117		>>> (1459330**389)%1592537 114032 114032		33
num	66	111	110	106	111	117	114	32	33
caractère	'B'	'o'	'n'	'j'	'o'	'u'	'r'	' '	'!'

3- ROBUSTESSE ET FAILLES POSSIBLES :

Pour « casser » RSA il est nécessaire de pouvoir factoriser le nombre n pour retrouver le produit initial des nombres *p* et *q*. Avec les algorithmes classiques, le temps que prend cette factorisation croît exponentiellement avec la longueur de la clé. Ce qui fait le succès du RSA est qu'il n'existe pas d'algorithme connu de la communauté scientifique pour réaliser une attaque force brute avec des ordinateurs classiques. On peut trouver la factorisation d'une clé de taille inférieure à 256 bits en quelques minutes sur un ordinateur individuel, en utilisant des logiciels librement disponibles. Pour une taille allant jusqu'à 512 bits, il faut faire travailler conjointement plusieurs centaines d'ordinateurs. Par sûreté, il est couramment recommandé que la taille des clés RSA soit au moins de 2 048 bits .

4- AUTRE EXERCICE DE CHIFFREMENT RSA SIMPLIFIE ET TRAITE A LA MAIN :

Alice décide de choisir $(n\,;\,e) = (21\,;\,5)$ comme clé publique.

a) Peut-elle faire ce choix ? Expliquer pourquoi ce choix de clé publique ne garantit pas la sécurité du message.

La valeur de n est trop faible car on peut facilement factoriser pour retrouver $p = 3$ et $q = 7$

b) Rappeler les valeurs de $p, q, m.$: $p = 3$, $q = 7$ donc $m = 2 \times 6 = 12$

c) Parmi les clés suivantes déterminer celle qu'Alice peut choisir comme clé privée :

$(21\,;6)$; $(21\,;11)$; $(21\,;13)$; $(21\,;15)$; $(21\,;17)$; $(21\,;19)$.

On vérifie si la relation $(d \times e)\%m = 1$ est respectée. Si on teste pour chacune des valeurs de d données, on constate que seule la valeur $d = 17$ fonctionne : $(17 \times 5)\%12 = 85\%12 = 1$

Dans la suite on admet qu'Alice choisit la seule clé privée possible parmi les propositions précédentes.

Bob veut envoyer à Alice le message (*M*) : PAL. Il va utiliser la clé publique (21 ; 5).

b) Associer à chaque lettre du message, sa **place** dans l'alphabet. Par exemple la lettre C est codée par 3. Obtenir ainsi trois entiers naturels M_1, M_2, M_3.

Pour 'P' , on a $M_1 = 16$	Pour 'A' , on a $M_2 = 1$	Pour 'L' , on a $M_3 = 12$

c) Déterminer les 3 nombres C_1, C_2, C_3 qui chiffrent les nombres M_1, M_2, M_3.

$C_1 = (16^5)\%21 = 4$	$C_2 = (1)\%21 = 1$	$C_3 = (12^5)\%21 = 3$

d) En déduire le message chiffré (*C*) que Bob envoie à Alice :

message envoyé constitué des nombres : 4 1 3

Alice reçoit le message (C) : 4 1 3. Elle veut déchiffrer ce message avec sa clé privé $(21\ ;17)$

e) Déterminer M_1, M_2, M_3 en déchiffrant les 3 nombres

$C_1 = (4^{17})\%21 = 16$	$C_2 = (1^{17})\%21 = 1$	$C_3 = (3^{17})\%21 = 12$

f) En déduire le message en clair que Bob a envoyé à Alice.

On retrouve les lettres initiales : 16ième lettre : 'P' , 1ère lettre : 'A' , 12ième lettre : 'L'