

Códigos y criptografía: Curso 2022-2023

Práctica 4: Intercambio de claves. Protocolo de Diffie y Hellman.

- NOTA: Tanto en esta práctica como en posteriores se trabajará con números naturales grandes. Para evitar que MATLAB aproxime los números obtenidos debemos trabajar en formato uint64.
- Asimismo, siempre que las operaciones sean modulares debemos tomar módulos operación a operación, para que el cálculo sea más eficiente y que no se saturé.

1. Función *power_mod*

```
1 function pow = power_mod (b,e,m)
```

Se trata de una función que aplica el método de potenciación rápida o binaria para calcular potencias modulares.

Entradas:

b : un número natural, la base de la potencia.

e : un número natural, el exponente de la potencia.

m : un número natural, el módulo de trabajo.

Salida: un número natural, el resultado de la potencia b^e módulo m .

Ejemplo:

```
1 >> pow = power_mod (34237778, 38472317, 101010331)
2 pow =
3     uint64
4     25000315
```

2. Función *generate_0*

```
1 function gen = generate_0 (g,p)
```

Se trata de una función que comprueba, mediante la definición de generador, si el número natural g es generador del grupo multiplicativo determinado por el primo p .

Entradas:

g : el número natural candidato a generador.

p : el número primo que determina el grupo multiplicativo.

Salidas:

$gen = 0$ en caso de que no sea generador.

$gen = g$ en caso de que sea generador.

- La función también debe indicar el tiempo empleado.

Ejemplos:

```
1 >> gen = generate_0 (7,100003)
2 Elapsed time is 0.380462 seconds.
3 gen = 7
```

```
1 >> gen = generate_0 (18,100003)
2 Elapsed time is 0.227053 seconds.
3 gen = 0
```

```
1 >> gen = generate_0 (15,1234547)
2 Elapsed time is 3.218184 seconds.
3 gen = 15
```

3. Función *generate*

```
1 function gen = generate (g,p)
```

Se trata de una función que comprueba, mediante el criterio alternativo estudiado en clase, si el número natural g es generador del grupo multiplicativo determinado por el primo p .

Entradas:

g : el número natural candidato a generador.

p : el número primo que determina el grupo multiplicativo.

Salidas:

$gen = 0$ en caso de que no sea generador.

$gen = g$ en caso de que sea generador.

- La función también debe indicar el tiempo empleado.

Ejemplos:

```
1 >> gen = generate (7,100003)
2 Elapsed time is 0.085435 seconds.
3 gen = 7
```

```
1 >> gen = generate (18,100003)
2 Elapsed time is 0.002836 seconds.
3 gen = 0
```

```
1 >> gen = generate (15,1234547)
2 Elapsed time is 0.005067 seconds.
3 gen = 15
```

4. Programa *diffie_hellman*

```
1 diffie_hellman
```

Se trata de un programa en el que se implementa el protocolo de Diffie y Hellman para el intercambio de claves. Para ello:

- Se deben pedir los elementos comunes g y p y comprobar que sean válidos.
- El agente A escoge un número entre 2 y $p - 2$ (podría ser de forma aleatoria) y envía a B la potencia correspondiente.
- El agente B realiza el procedimiento análogo.
- Tanto A como B obtienen la clave compartida.

Ejemplo:

```
1 >> diffie_hellman
2
3 COMMON ELEMENTS
4 Introduce a prime number p:
5 1999
6 Introduce a generator g of the multiplicative group determine by 1999:
7 33
8
9 AGENT A
10 A should introduce its private number a between 2 and 1997:
11 557
12 A sends to B power_mod (33,557,1999) = 1185
13
14 AGENT B
15 B should introduce its private number b between 2 and 1997:
16 1093
17 B sends to A power_mod (33,1093,1999) = 1448
18
19 BOTH AGENTS
20 A obtains power_mod (1448,557,1999) = 946
21 B obtains power_mod (1185,1093,1999) = 946
```

5. Programa *man_in_the_middle*

```
1 man_in_the_middle
```

Se trata de un programa que muestra la vulnerabilidad del protocolo de Diffie y Hellman para el intercambio de claves. Para ello:

- Se deben pedir los elementos comunes g y p y comprobar que sean válidos.
- Tanto el agente A como el B escogen sus números privados entre 2 y $p - 2$ (podría ser de forma aleatoria) y calculan las potencias respectivas, que las interceptará el espía C .
- El espía C escoge su número entre 2 y $p - 2$ (podría ser de forma aleatoria) e interactúa con A y con B suplantando la identidad de estos.
- A y C deben obtener una clave común.
- B y C deben obtener otra clave común.

Ejemplo:

```
1 >> man_in_the_middle
2
3 COMMON ELEMENTS
4 Introduce a prime number p:
5     1999
6 Introduce a generator g of the multiplicative group determine by 1999:
7     33
8
9 AGENT A
10 A should introduce its private number a between 2 and 1997:
11     1599
12 A sends to B power_mod (33,1599,1999) = 789
13 This number is intercepted by C.
14
15 AGENT B
16 B should introduce its private number a between 2 and 1997:
17     843
18 B sends to A power_mod (33,843,1999) = 1874
19 This number is intercepted by C.
20
21 AGENT C
22 C intercepts 789 from A and 1874 from B and saves these numbers.
23 C should introduce its private number c between 2 and 1997:
24     284
25 C sends to A and B power_mod (33,284,1999) = 225
26
27 B thinks he has received such number from A and A from B.
28
29 AGENTS A AND B
30 A obtains power_mod (225,1599,1999) = 204
31 B obtains power_mod (225,843,1999) = 738
32
33 AGENT C
34 For A computes power_mod (789,284,1999) = 204
35 For B computes power_mod (1874,284,1999) = 738
```