IIC3253

Diffie-Hellman y ElGamal

RSA se basa en la idea de que multiplicar dos primos grandes es fácil, pero encontrarlos dada su multiplicación es difícil

En este sentido, la multiplicación de dos primos grandes es una *one-way function*.

También, para encriptar un mensaje m usábamos la ecuación $c := m^e \mod N$.

¿Qué otra función estamos suponiendo que es una *one way function?*

$$f(x) = x^e \mod N$$

Para los e y N usados, si esta función se puede invertir tendríamos problemas...

Es natural esperar que buena parte de nuestras premisas criptográficas se basen en *one-way functions*

One-way function

From Wikipedia, the free encyclopedia

In computer science, a **one-way function** is a function that is easy to compute on every input, but hard to invert given the image of a random input. Here, "easy" and "hard" are to be understood in the sense of computational complexity theory, specifically the theory of polynomial time problems. Not being one-to-one is not considered sufficient for a function to be called one-way (see Theoretical definition, below).

? **Unsolved problem in computer science**:

Do one-way functions exist?

(more unsolved problems in computer science)

The existence of such one-way functions is still an open conjecture. Their existence would prove that the complexity classes P and NP are not equal, thus resolving the foremost unsolved question of theoretical computer science.[1]:ex. 2.2, page 70 The converse is not known to be true, i.e. the existence of a proof that P≠NP would not directly imply the existence of one-way functions.[2]

¿Qué otras herramientas criptográficas que hemos visto son *one-way functions*?

Para ciertos grupos, en particular subgrupos de $\mathbb{Z}_p^*$, creemos que es difícil calcular el *logaritmo discreto*

Dado $g \in G$ y el valor $y = g^x$, si $|\langle g \rangle|$ es *grande*, es difícil calcular x

Siendo $g \in G$ un elemento que genera un subgrupo *grande,* la función $f(x) = g^x$ es una *one-way function*

Ejemplo

Consideremos $\mathbb{Z}_p^*$ donde p es el primo

2184735958988820847550672491716226506357140198532537036763136178111402965302595681515760532819041114104
4160689815741319381196532979871500038979862309158738250945118554961626824152307536605872616502884288878
0624670527776052278467097818506147927484588389513422048126018381129378053717826003801060205228844064528
2381882445568398204288292818343119459318917143106637113851025297964851355307876258459614742745683728962
3008879364829477705183636149304120998948654278133874026711188494311770883514889363351380064520413459602
696141353949407971810071848354127868725934057811052285511726070951954828625761984797831079801857828431

Consideremos el valor g igual a

2174464614324321605702022855115620875270394288720730886866444527554867473662050873292576435751519954730
3283870847514971207187185912917434889899462163342116463504651187567271577773370136574456671482796328194
6984303144643072394262976090391828780001136731637603815756299285930385635362349585632133854954455419111
6841474125049441861570488354829672808054579585984332040507247226675344890671460563730864246842289855863
0812487636188819677130134963833040948411243908028200183454403067866539747291394732970142401544187137624
428138444276721310399530477238861596789940953323090393313600101710523922727140772179016720953265564666

El subgrupo generado $\langle g \rangle$ tiene orden

13491513086924420379699774282445616590110876328163828635542747312619 $> 2^{223}$

El subgrupo generado $\langle g \rangle$ tiene orden

13491513086924420379699774282445616590110876328163828635542747312619 $\approx 2^{223}$

Ahora nos entregan el valor de g y nos dicen que el valor de y ($g^x \bmod p$ para algún x) es igual a

1536476230038846372453692052975822982216352914989956039610111965258846739270632359082997656587785541146
0404795470021081424113665835216208812609454341730731152625410757805611096732628352026138173163850936514
2972776197933119290552906093126645803829658706720060523443906450845950541976058455694606511147844946799
0700701510854682860235742761960466183026116266076083313253895843453059934637080837623653743059754476906
5148667866961431930701752171665789415576103861339420573330180827030167025617510940422605291486028835373
482572429815605657042144465177500680554535410615558841089156813927431336187658507588690514238359496142

¿Cuánto valía x?

1073047948652475552916726959330258486833057832169442702837091821008073877695134995329494190617270381242
4113590135611300896946910511672767341496249447139480832759489717781717366435764805390482608595152793454
9444105493776617607799301400283393374479400301012298030832336662879611592005488674389774116106703340022
1946758267329105548464433403828011812711827922291897705278017299704851528386224528600741080582768528528
3803090275486719473186614608098570879968482055934388686340093829677116684111318091921710929784849921861
265475473551891121985559541852576660111216196950860288307666933919442980522109794494422597424335681405

Un ejemplo más simple en vivo...

¿Número primo favorito?

$$p = 97$$

Usemos el generador $g = 5$

Dado que $y = g^x \bmod p = 31$, ¿Cuánto vale x?

¿Por qué querríamos usar elementos de orden primo en $\mathbb{Z}_p^*$?

¿Quién nos asegura cuál es el orden de $\langle g \rangle$?

Diffie-Hellman (y Merkle)

Un protocolo para compartir un secreto en un canal público

Es un protocolo para intercambiar una llave secreta (simétrica) en un canal inseguro

Usemos este G y este g

Dale!

$x \in \{1, \ldots, |\langle g \rangle|\}$ al azar

$y \in \{1, \ldots, |\langle g \rangle\}$ al azar

g^x

g^y

¿Cuál es la llave secreta compartida?

¿Cuál es la llave secreta compartida?

Alice tiene x, g^y

Bob tiene y, g^x

$$S = g^{x \cdot y}$$

$S = (g^y)^x$

$S = (g^x)^y$

¿Y qué ve un atacante?

Tiene los valores g, g^x y g^y

¿Puede descubrir algo en base a esto?

Ejercicio: describa un juego que defina
la seguridad de este protocolo

DH Assumption: Un atacante (de tiempo polinomial) **no** puede ganar el juego con probabilidad no despreciable

¿Para qué queremos DH? Podríamos usar RSA...

Mi llave pública: P_A

Bacán! La mía es P_B

Listo, podemos comunicarnos de forma segura!

Es menos eficiente...

¿Y qué pasa si se roban una llave secreta?

¿Y si usamos DH?

¿Si un atacante gana acceso a x?

¡Esperamos que no sea suficiente para poder leer conversaciones presentes/pasadas!

Forward secrecy

From Wikipedia, the free encyclopedia

In cryptography, **forward secrecy** (**FS**), also known as **perfect forward secrecy** (**PFS**), is a feature of specific key agreement protocols that gives assurances that session keys will not be compromised even if long-term secrets used in the session key exchange are compromised. For HTTPS, the long-term secret is typically the private key of the server. Forward secrecy protects past sessions against future compromises of keys or passwords. By generating a unique session key for every session a user initiates, the compromise of a single session key will not affect any data other than that exchanged in the specific session protected by that particular key. This by itself is not sufficient for forward secrecy which additionally requires that a long-term secret compromise does not affect the security of past session keys.

Diffie-Hellman (y Merkle)

Acordamos un grupo $\langle g \rangle$ de orden q

Alice genera $x \in \{1, \ldots, q\}$

Bob genera $y \in \{1, \ldots, q\}$

Alice envía g^x

Bob envía g^y

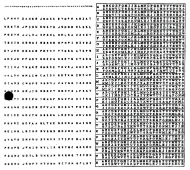
$S = (g^y)^x$

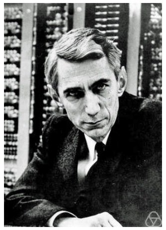
$S = (g^x)^y$

$$S = g^{x \cdot y}$$

ElGamal

Un protocolo de criptografía de clave pública



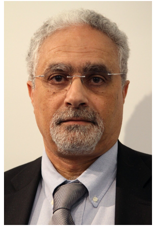


100 - 44 a.C. | 1882 | 1945

Diffie-Hellman

RSA

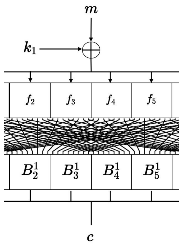
ElGamal



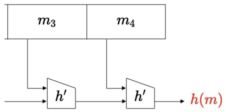
HMAC

$$HMac_k(m) = h(k_2 \mid\mid h(k_1 \mid\mid m))$$

AES



SHA-256



1976 | 1978 | 1985 | 1996 | 2001

ElGamal se basa en usar Diffie-Hellman, donde quien envía el mensaje genera una llave *efímera*

ElGamal

Acordamos un grupo $\langle g \rangle$ de orden q

Alice genera $x \in \{1, \ldots, q\}$, esa será su llave secreta

Alice publica g^x, que será su llave pública

ElGamal

Supongamos que Bob quiere encriptar un elemento m del grupo.

Bob genera $y \in \{1, \ldots, q\}$ (llave efímera)

Le envía a Alice el par $(m * g^{xy}, g^y)$

¿Cómo puede Alice recuperar m?

ElGamal

Le envía a Alice el par $(m * g^{xy}, g^{y})$

¿Cómo puede Alice recuperar m?

Podría intentar buscar el inverso de g^{xy}

¿Cómo lo hacemos?

ElGamal

¿Algoritmo extendido de Euclides?

¿Y si no estamos en $\mathbb{Z}_p^*$?

Recordar: q es el orden del grupo $\langle g \rangle$

$$g^{xy} * g^{y(q-x)}$$

$$= g^{xy+yq-xy} = g^{yq}$$

ElGamal

$$g^{xy} * g^{y(q-x)}$$

$$= g^{xy+yq-xy} = g^{yq}$$

Como yq es múltiplo del orden del grupo, $g^{yq} = e$

$g^{y(q-x)}$ es el inverso de g^{xy}

ElGamal

Grupo $\langle g \rangle$ de órden q

Alice: llave secreta $x \in \{1, \ldots, q\}$, llave pública g^x

Encriptar m:

1. Generar $y \in \{1, \ldots, q\}$ al azar
2. Calcular $s = g^{xy}$
3. Enviar $(m * s, g^y)$

Decriptar $(m * s, g^y)$:

1. Calcular $s^{-1} = g^{y(q-x)}$
2. Calcular $m * s * s^{-1} = m$

Ejemplo

Consideremos $\mathbb{Z}_{107}^*$ y el grupo generado por el 3

¿Cuántos elementos podría tener $\langle 3 \rangle$?

106, 53 ó 2

```
>>> pow(3, 53, 107)
1
```

¿Es suficiente para decir que tiene orden 53?

✅

Ejemplo

Supongamos que la llave secreta de Alice es 27

Con su llave pública $3^{27} \bmod 107$ encriptemos el 35

No es necesario que el 35 esté en $\langle 3 \rangle$, basta que sea un elemento del grupo $\mathbb{Z}_{107}^{*}$

Bob genera $y = 19$ y envía $(35 \cdot (3^{27})^{19}, 3^{19}) = (47, 75)$

Alice calcula $s^{-1} = 75^{53-27} = 75^{26} = 44$

Alice calcula $47 \cdot 44 = 35$

Ejemplo

Supongamos que la llave secreta de Alice es 27

Con su llave pública $3^{27} \bmod 107$ encriptemos el 35

No es necesario que el 35 esté en $\langle 3 \rangle$, basta que sea un elemento del grupo $\mathbb{Z}_{107}^*$

Bob genera $y = 19$ y envía $(35 \cdot (3^{27})^{19}, 3^{19}) = (47, 75)$

a $s^{-1} = 75^{53-27} = 75^{26} = 44$

calcula $47 \cdot 44 = 35$

¿Qué información tiene un atacante?

$$g^x, m * g^{xy}, g^y$$

Si suponemos que g^{xy} se ve aleatorio para un atacante, entonces $m * g^{xy}$ también

¿Por qué?

¿Cuántos elementos tiene $m\langle g\rangle = \{m * s \mid s \in \langle g\rangle\}$?