Módulo 06
Seguridad en Redes (Pt. 1)

Redes de Computadoras
Depto de Cs. e Ing. de la Comp.
Universidad Nacional del Sur

Copyright

Contenidos

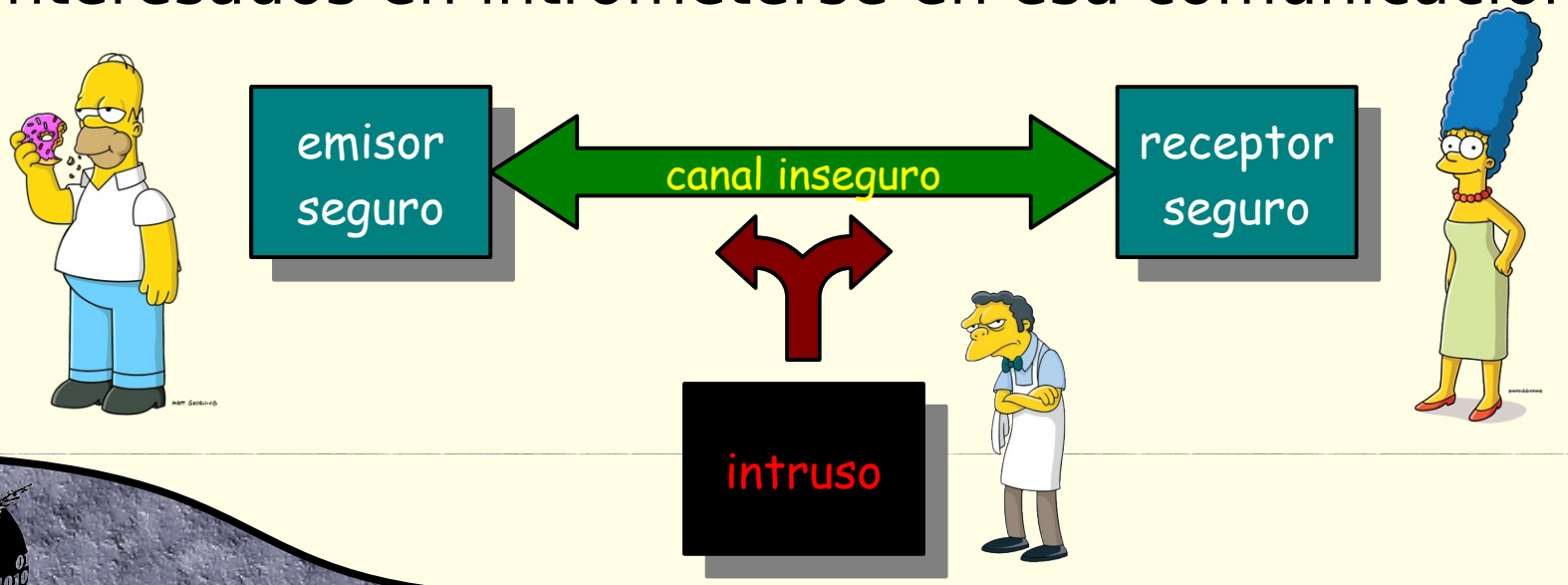
- Introducción a la seguridad en redes
- Principios de la criptografía
- Autenticación
- Integridad
- Distribución de claves y de certificados
- Seguridad multinivel

Seguridad en Redes

- La seguridad en redes abarca un conjunto heterogéneo de objetivos:
  - Confidencialidad: sólo el emisor y el receptor deben ser capaces de entender el contenido del mensaje
  - Autenticación: tanto emisor como receptor desean poder verificar la identidad de su interlocutor
  - Integridad de los mensajes: emisor y receptor también desean poder comprobar que el mensaje no ha sido alterado durante el transporte
  - Acceso y disponibilidad: los servicios provistos por la red deben estar a disposición de los usuarios

Actores

- Usaremos la siguiente analogía para ilustrar los principales actores en la seguridad en redes:
  - Supongamos que Homero y Marge desean poder comunicarse de manera segura
  - Es posible que existan terceros, por caso Moe, interesados en intrometerse en esa comunicación



Actores

- ¿A quién representan Homero y Marge en el mundo real?
  - A los navegadores y servidores **HTTP** utilizados en las transacciones electrónicas
  - A los clientes **DNS** (resolvers) y a la jerarquía de servidores **DNS**
  - A los routers cuando intercambian mensajes de actualización de sus tablas de ruteo
  - ...

Actores

- ¿Qué daño pueden causar los “Moes”?
  - Escuchas ilegales: interceptando mensajes privados
  - Adulteración: agregar o modificar los mensajes que se intercambian en una comunicación
  - Personificación: alterando la dirección de origen (o cualquier otro campo) de los paquetes
  - Secuestro: tomar una conexión activa y reemplazar al emisor o receptor de manera unilateral
  - Negación de servicio: impedir que un cierto servicio pueda ser accedido por sus usuarios legítimos

Principios de la criptografía

K_H clave de encriptado de Homero

clave de desencriptado de Marge K_M

mensaje original

emisor seguro

mensaje codificado

canal inseguro

receptor seguro

mensaje original

intruso

Principios de la criptografía

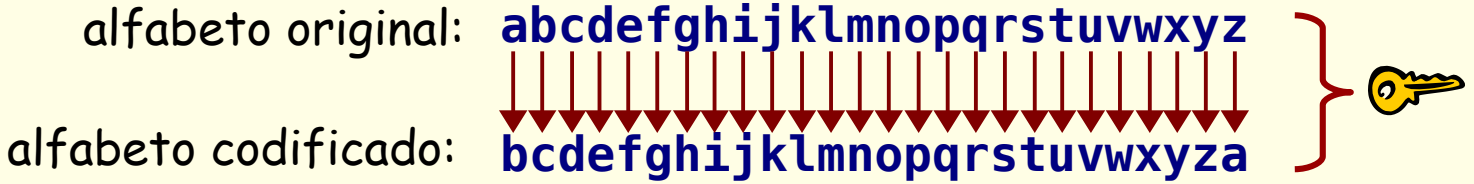
- El proceso de encriptado y de desencriptado dependen de tres parámetros de entrada:
  - El mensaje a ser enviado/que acaba de ser recibido
  - La clave de encriptado y/o desencriptado (ya que puede o no ser la misma)
  - El algoritmo de encriptado y/o desencriptado
- La resistencia del esquema de encriptación depende de mantener secreta la clave usada
  - El algoritmo de encriptado y desencriptado, en contraste, puede ser algo conocido por todos

Principios de la criptografía

- Los distintos esquemas de encriptado se distinguen observando la manera elegida para distribuir las claves:
  - Clave simétrica: tanto emisor como receptor hacen uso de la misma clave para encriptar y desencriptar el mensaje
  - Clave asimétrica: el emisor hace uso de la clave pública del receptor para encriptar el mensaje y el receptor desencripta el mensaje con su clave privada
  - No usa clave: por caso, una función hash. No brinda secreto, pero si resulta de utilidad en ciertos casos

Clave simétrica

- El cifrado por substitución es uno de los ejemplos más sencillos de clave simétrica
  - Julio César codificaba las órdenes que enviaba a sus generales usando este esquema de encriptado
- El cifrado monoalfabético consiste en encontrar un único mapeo entre las letras del alfabeto
  - Por caso:



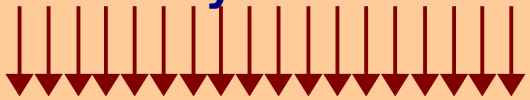
Clave simétrica

mensaje original: **Hola Marge!**

mensaje codificado: **Ipmb Nbshf!**

mapeo utilizado

abc...xyzABC...XYZ



bcd...yzaBCD...YZA

mensaje original: **odio a flanders**

mensaje codificado: **kvsk m xgmjvcoi**

mapeo utilizado

abcdefghijklmnopqrstuvwxyz

mnbvcxzasdfghjklpoiuytrewq

Criptoanálisis

- El criptoanálisis consiste en el estudio de los métodos que permiten descifrar el contenido de un mensaje encriptado
  - El uso ilegal del criptoanálisis posibilita sacar provecho de la información contenida en el mensaje encriptado
  - El uso legal del criptoanálisis, en contraste, permite identificar las vulnerabilidades de los distintos algoritmos de encriptado y/o claves empleadas antes de que la información protegida sea comprometida

Criptoanálisis

- ¿Qué tan difícil resulta romper en encriptado basado en la sustitución monoalfabética?
  - Para cifrar las letras del alfabeto de **a** a **z**, existen tantas claves como permutaciones, esto es, **26!**
  - Si ciframos mayúsculas y minúsculas la cantidad de claves crece a **52!**
  - Eventualmente, la cantidad de claves puede ser lo suficientemente grande como para hacer inviable al análisis por fuerza bruta (en tiempo o costo)
  - No obstante, existen otras estrategias, por caso el ataque basado en diccionarios

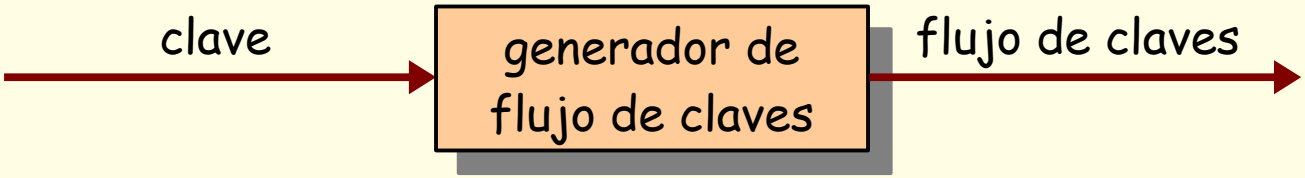
Clave simétrica

¿¿cómo hacen para ponerse de acuerdo
Homero y Marge acerca de qué clave usar??

Esquemas simétricos

- En la actualidad se han explorado dos grandes familias de esquemas de cifrado simétrico:
  - Cifradores de flujo de bits
  - Cifradores en bloques de bits
- Los crifradores de flujo de bits encriptan de a un bit a la vez
- Los cifradores en bloques de bits encriptan de a grupos de bits a la vez

Cifrador de flujo de bits



- Idea central: combinar cada uno de los bits del flujo de claves con los bits del mensaje original para obtener los bits de mensaje en encriptado
  - Sean **`m(i)`**, **`ks(i)`** y **`c(i)`** el i-ésimo bit del mensaje original, del flujo de claves y del mensaje encriptado, respectivamente
  - En este contexto, **`c(i) = ks(i) ⊕ m(i)`** y para recuperar el mensaje original **`m(i) = ks(i) ⊕ c(i)`**

Cifrador en bloque de bits

- El mensaje a ser encriptado se procesa en bloques de **k** bits
  - Se adopta un mapeo que convierta los **k** bits del mensaje original en los **k** bits del texto encriptado
- Por caso, para **k = 3**:

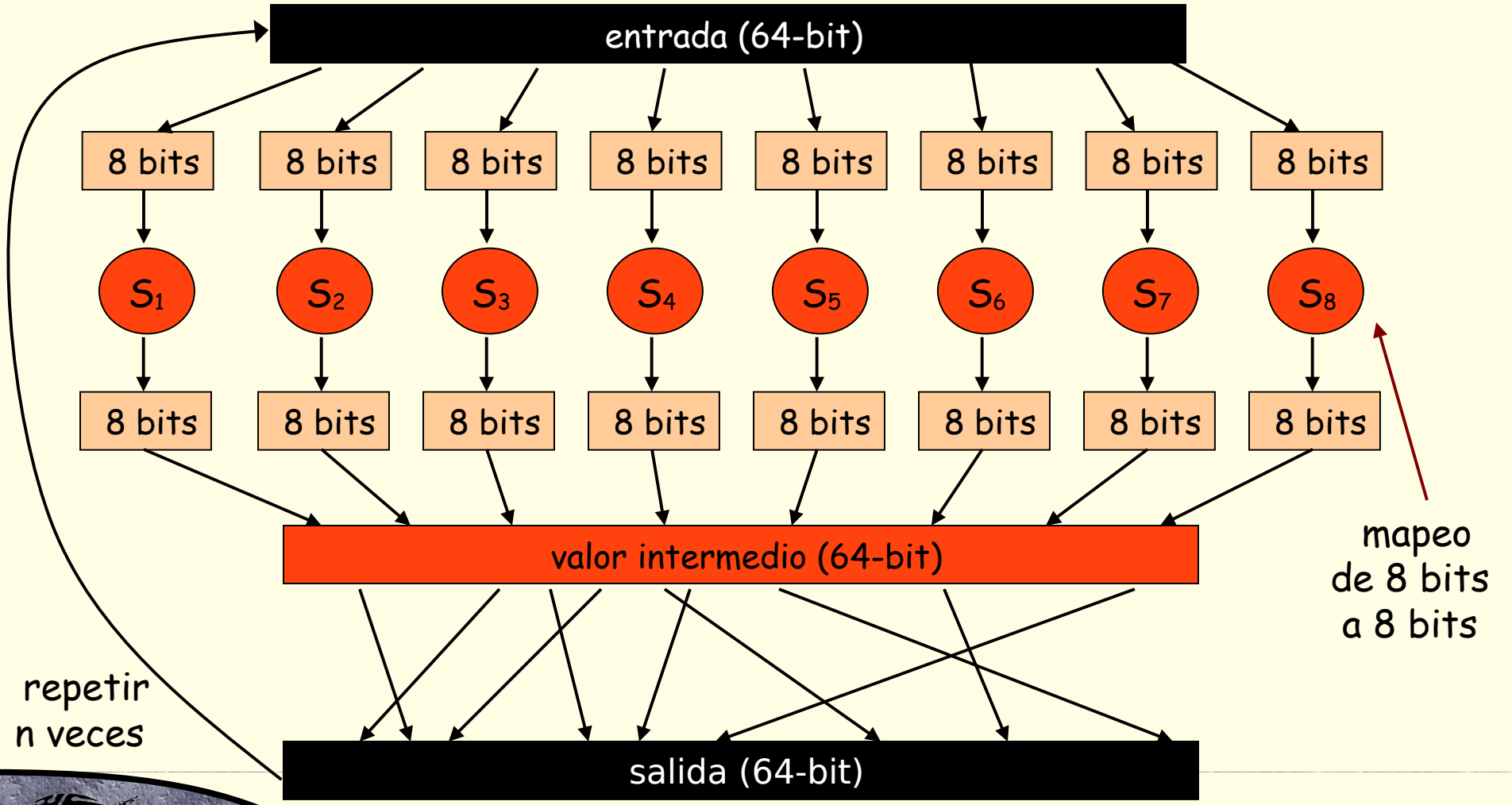
original	encriptado	original	encriptado
000	110	100	011
001	111	101	010
010	101	110	000
011	100	111	001

- ¿Cómo se codifica el mensaje **100110110111**?

Cifrador en bloque de bits

- ¿Cuántos mapeos se podrán construir para el ejemplo anterior donde **k = 3**?
  - ¿Cuántas entradas tiene la tabla?
  - ¿Cuántas permutaciones se pueden postular?
- En general, se pueden postular $2^k!$ mapeos
  - La tabla de un cifrador en bloques de 64 bits necesita de 2^{64} entradas
- Una alternativa es hacer uso de una función que simule la aplicación de este mapeo

Función prototípica

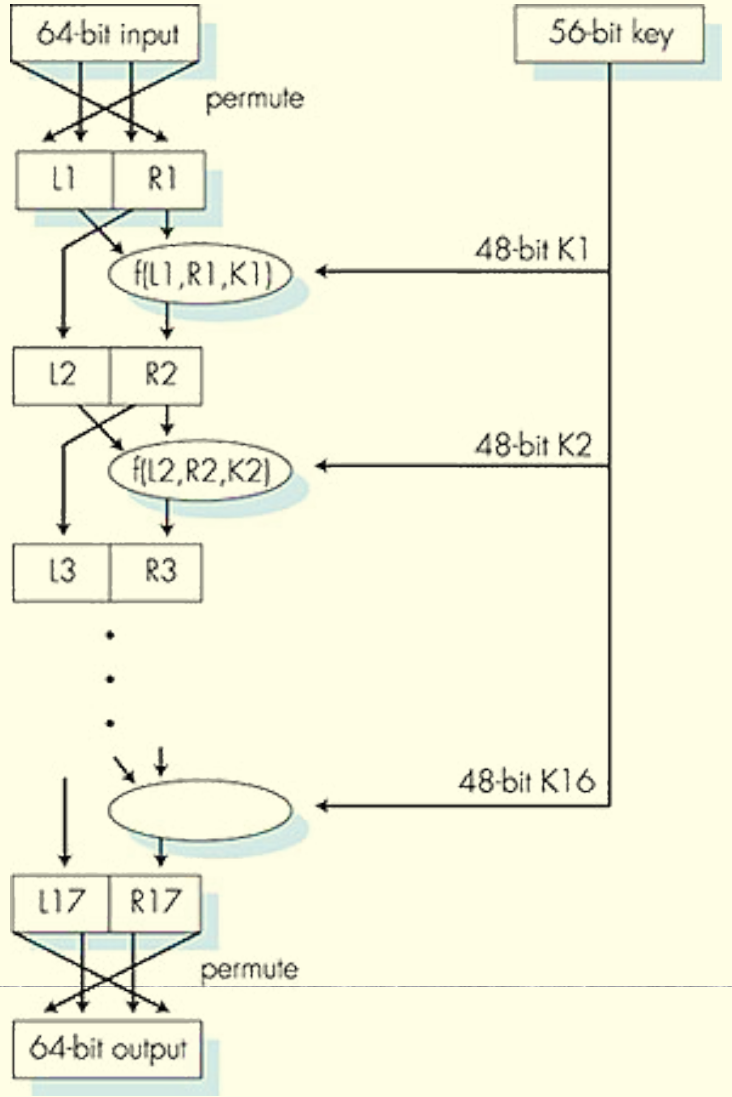


Data Encription Standard

- El algoritmo de encriptado **DES** (Data Encription Standard) es un estándar de encriptado adoptado por Estados Unidos
  - Surgió de una competencia organizada por el gobierno norteamericano en la década del 70'
  - Hace uso de una clave simétrica de 56 bits y codifica iterativamente de a bloques de 64 bits de datos
  - Resultó un estándar bastante controversial, se sugirió que la **NSA** contaba con un acceso oculto que les posibilitaba desencriptar la información protegida de manera directa, incluso sin conocer la clave

Data Encription Standard

- El proceso aplicado al mensaje a ser codificado es el siguiente:
  1) Permutación inicial
  2) 16 rondas de alteraciones, cada una usando una clave de 48 bits diferente, que se derivan de la clave original
  3) Permutación final



Data Encription Standard

- ¿Qué tan seguro es **DES** hoy en día?
  - El primer “Desafío **DES**” tomó en 1997 unos cuatro meses en ser quebrado usando fuerza bruta (tomaron parte unas 78.000 computadoras hogareñas)
  - El segundo tomó sólo 41 días en 1998, y la variante usando hardware especializado tomó sólo 56 horas
  - El tercer desafío, que combinó hardware especializado con computadoras hogareñas quebró la clave **DES** es apenas 22 horas, 15 minutos
  - Evidentemente a esa altura los 56 bits de seguridad provistos por **DES**... ¡resultaban escasos!

Triple DES

- El algoritmo triple **DES** (**3DES**) extiende la cantidad de bits de la clave:
  - La idea es en esencia aplicar tres veces el algoritmo **DES** a cada bloque de datos
  - Sean **K1**, **K2** y **K3** las tres claves **DES** y **m** el mensaje original; el mensaje encriptado se obtiene haciendo la siguiente conversión:

$$E_{K3}(D_{K2}(E_{K1}(m)))$$

  - Luego, para desencriptar se computa la reversa:

$$D_{K1}(E_{K2}(D_{K3}(m)))$$

Advanced Encription Standard

- El algoritmo de clave simétrica **AES** (Advanced Encription Standard) es el nuevo estándar que reemplaza los ya añejos **DES** y triple **DES**
  - Procesa el mensaje a ser codificado de a bloques de 128 bits
  - Existen tres versiones, **AES-128**, **AES-192** y **AES-256**, con claves de 128, 192 y 256 bits respectivamente
  - Cuando el criptoanálisis de un mensaje **DES** tome tan sólo 1 segundo, el criptoanálisis de un mensaje **AES** seguiría tomando 149.000.000.000.000 años

Clave asimétricas

- Para hacer uso de un esquema de clave simétrica es necesario que emisor y receptor acuerden qué clave privada usar
  - Esto es especialmente complicado de organizar si emisor y receptor no pueden reunirse “en persona”
- La alternativa es hace uso de un esquema radicalmente diferente, donde emisor y receptor no usen la misma clave
  - Este tipo de esquema se denomina clave asimétrica (también criptografía de clave pública)

RSA

- El algoritmo **RSA** (propuesto por Rivest, Shamir y Adleman), implementa el citado esquema de criptografía basada en claves públicas
  - Emisor y receptor no necesitan tener acceso a la misma clave privada
  - El emisor hace uso de la clave pública del receptor, la cual es conocida por todos, ¡incluso por los potenciales intrusos!
  - El receptor hace uso de su clave privada para desencriptar el mensaje

RSA

Homero hace uso de
la clave pública de Marge

clave pública

clave privada

K^+_M

K^+_M

K^-_M

mensaje codificado
$K^+_M(m)$

mensaje original
m

emisor seguro

canal inseguro

receptor seguro

mensaje original
$m = K^-_M(K^+_M(m))$

intruso

¿¿$K^+_M(m)$??

¿¿$K^+_M(K^+_M(m))$??

RSA

- Para poder hacer uso del esquema de clave asimétrica basado en una clave pública y otra privada se deben satisfacer dos condiciones:
  - Sea K^+ y K^- el par de claves pública y privada respectivamente. En este contexto, la primer condición es un tanto evidente:

$$m = K^-(K^+(m))$$

  - La segunda condición es que la clave privada K^- no debe poder ser derivada a partir de la clave pública

RSA

- Nociones básicas:
  - Un mensaje es en esencia una secuencia de bits
  - Toda secuencia de bits representa un único entero positivo
  - Es decir, encriptar un cierto mensaje consiste en encriptar un número
- Ejemplo:
  - Para el mensaje **`m = 10010001`**, **`m`** representa al número **`145`**; encriptar este número consiste en encontrar un segundo número que lo codifique

¿Cómo elegir las claves?

- Mecanismo para determinar una clave privada y otra pública que satisfagan las dos condiciones:

  1) Elegir dos número primos **p** y **q** bien grandes (por ejemplo, de unos 1000 dígitos cada uno)

  2) Computar **n = pq** y **z = (p-1)(q-1)**

  3) Elegir un **e < n** que no comparta factores en común con **z** (**e** y **z** deben ser primos entre sí)

  4) Elegir un **d** tal que **ed-1** sea divisible de manera exacta por **z** (es decir, **de mod z = 1**)

  5) La clave pública será **(n,e)** y la privada **(n,d)**

Encriptado y desencriptado

- Sean **(n,e)** y **(n,d)** las claves pública y privada, obtenidas de la forma indicada
- Para encriptar un patrón de bits **m**, se calcula **$c = m^e \bmod n$** (el resto de dividir **m^e** por **n**)
- Para desencriptar un patrón de bits **c**, se calcula **$m = c^d \bmod n$** (el resto de dividir **c^d** por **n**)
- Reemplazando iguales por iguales, nos queda que necesariamente **$m = (m^e \bmod n)^d \bmod n$**

Ejemplo

- Supongamos que se parte de elegir **p = 5** y **q = 7**, por lo que **n = 35** y **z = 24**
  - Eligiendo **e = 5**, **e** y **z** resultan primos entre sí
  - Luego, **d = 29** para que **ed - 1** sea divisible por **z**
- Encriptado usando la clave pública:

  m = "x" = 24 $\quad m^e = 24^5 = 7.962.624 \quad c = m^e \bmod n = 7.962.624 \bmod 35 = 19$

- Desencriptado usando la clave privada:

  c = 19 $\quad c^d = 19^{29} = 12.129.821.994.589.221.844.500.501.021.364.910.179$

  $m = c^d \bmod n = 12.129.821.994.589.221.844.500.501.021.364.910.179 \bmod 35 = 24$

Teoría de números

- ¿Por qué razón se verifica que las claves se complementan tan bien?
- Un resultado obtenido en la teoría de números resulta práctico para demostrar esa relación:
  - Si **p** y **q** son primos y **n = pq**, entonces se verifica que $x^y \bmod n = x^{y \bmod (p-1)(q-1)} \bmod n$
  - En base a este resultado se puede demostrar el correcto funcionamiento del algoritmo **RSA**:

$$
\begin{aligned}
(m^e \bmod n)^d \bmod n &= m^{ed} \bmod n \\
&= m^{ed \bmod (p-1)(q-1)} \bmod n \\
&= m^1 \bmod n \\
&= m
\end{aligned}
$$

propiedad anterior

ed - 1 es múltiplo de z

Propiedad a tener en cuenta

- La forma en la que se eligen las claves pública y privada permiten demostrar otra propiedad muy interesante:

$$K^-(K^+(m)) = m = K^+(K^-(m))$$

- En otras palabras, no importa cuál apliquemos primero, ya sea la clave pública o la clave privada, puesto que son intercambiables
  - Este resultado será usado más adelante al abordar el proceso de autenticación y de firma digital

¿Qué tan seguro es RSA?

- Un cierto atacante logra acceso a nuestra clave pública **(n,e)**... ¿qué probabilidad tiene de descubrir nuestra clave privada **(n,d)**?
  - Es decir, necesita factorizar **n** sin conocer previamente los valores **p** y **q**
  - Por suerte, encontrar los factores primos de un número arbitrario es un problema bastante complejo
- Recientemente se ha propuesto un algoritmo de factorización altamente eficiente:
  - Eso si, ¡sólo corre en computadoras cuánticas!

RSA en la práctica

- Debemos tener en cuenta que la función de exponenciación es computacionalmente intensiva
  - **DES** es al menos 100 veces más veloz que **RSA**, corriendo sobre el mismo hardware
- Idea brillante: ¡usar **RSA** para intercambiar la clave simétrica a ser usada durante el resto del intercambio cifrado!

¿Preguntas?